

國立交通大學

電信工程學系

碩士論文

乙太被動光纖網路動態頻寬分配-使用增強Deficit

Round Robin Algorithm

Dynamic Bandwidth Allocation Using Enhanced

Deficit Round Robin Algorithm for EPON

研究生：陳偉民

指導教授：田伯隆 博士

中華民國九十九年二月

# 乙太被動光纖網路動態頻寬分配-使用增強Deficit Round Robin Algorithm

學生：陳偉民

指導教授：田伯隆 博士

國立交通大學電信工程學系碩士班

## 摘要

目前各種類型的多媒體不斷的增加與發展，使得現在使用者對網路的頻寬需求大量增加，並且隨著各種多媒體有不同的連線品質需求，透過光具有傳輸速度快、錯誤率低，極低的訊號衰減等優勢，因此光纖網路便成為佈建下一代高速網路的首選。為了可以達到動態頻寬的分配以及滿足各種多媒體的服務品質需求，降低封包的延遲及縮小彼此封包延遲的差異就成為目前的首要任務。

本篇論文探討為乙太被動式光纖網路(EPON)的資料上傳管理，此演算法目的主要是可以公平的分配頻寬，透過將問題轉化以及適當的修改，成功的將封包的延遲降低並且同時減少封包的平均延遲變異數。之後在本篇論文會詳細的探討此演算法的詳細流程，最後透過模擬驗證確實可以有效達到我們所要求地結果。

# Dynamic Bandwidth Allocation Using Enhanced Deficit Round Robin Algorithm for EPON

Student : Wei-Ming Chen

Advisor : Dr. Po-Lung Tien

Department of Communication Engineering  
National Chiao Tung University

## ABSTRACT

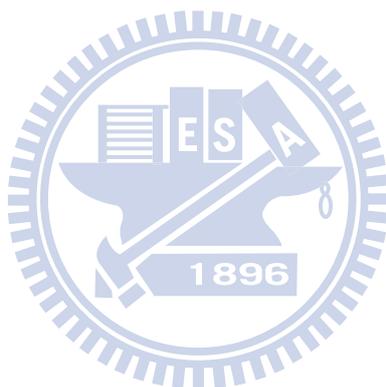
Due to multimedia programs are increased and developed nowadays, the required bandwidth to many users is increased rapidly, too. The advantage of light is high transmission rate, low bit error rate, and extremely low signal attenuation. Therefore, optical network is the best choice to build next generation high speed network. In order to achieve dynamic bandwidth allocation and satisfy the demand for quality of service, the decrease in packet delay and delay variance are both the most important issues for us to improve.

In this article, we focus mainly on management of upstream traffic on EPON. The purpose of this algorithm is that bandwidth can be allocated fairly through adaptive modified the original. Furthermore, we also successfully decrease the mean packet delay and delay variance. A detailed description of the algorithm will be given in the text and the simulation result will show it can really achieve our requirements.

## 誌謝

本篇論文得以完成，必須感謝指導教授田伯隆老師這一年多來的指導和教誨，透過與老師的討論和研究，讓我了解到許多研究與思考的方向，使得我在網路領域的認知方面又有上一層的了解。

另外也感謝實驗室的學長啟賢給我許多在研究上的建議，讓我在研究的上獲益匪淺，也同時感謝同學禕達、學弟柏宇與子凱在課後的研究與討論。最後也同時感謝我的父母對我的支持，可以讓我做研究時沒有後顧之憂，以及所有關心我的人。



## 目錄

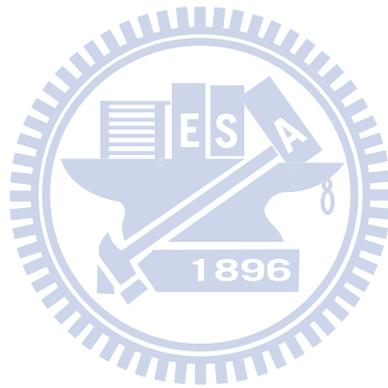
|                                                            |     |
|------------------------------------------------------------|-----|
| 中文摘要.....                                                  | i   |
| 英文摘要.....                                                  | ii  |
| 誌謝 .....                                                   | iii |
| 目錄 .....                                                   | iv  |
| 圖目錄.....                                                   | v   |
| 表目錄.....                                                   | vi  |
| Chapter 1 Introduction .....                               | 1   |
| Chapter 2 System Architecture.....                         | 3   |
| 2.1 Structure Of EPON.....                                 | 3   |
| 2.2 Downstream and Upstream over EPON.....                 | 4   |
| 2.3 Multi-Point Control Protocol(MPCP).....                | 6   |
| 2.4 EPON advantage and challenging.....                    | 7   |
| Chapter 3 Dynamic Bandwidth Allocation on EPON.....        | 8   |
| 3.1 Interleaved Polling with Adaptive Cycle Time.....      | 8   |
| 3.2 Some Dynamic Bandwidth Allocation Algorithm.....       | 11  |
| Chapter 4 Problems and Solution.....                       | 13  |
| 4.1 Deficit Round Robin.....                               | 13  |
| 4.2 Fairness issue.....                                    | 18  |
| 4.3 Modified Deficit Round Robin.....                      | 20  |
| 4.4 Dual Deficit Round Robin non reset to zero.....        | 22  |
| Chapter 5 Simulation Results and Performance Analysis..... | 26  |
| 5.1 Network parameter and traffic model.....               | 26  |
| 5.2 Simulation results.....                                | 28  |
| Chapter 6 Conclusion and Future Works.....                 | 35  |
| Reference .....                                            | 36  |

## 圖目錄

|        |                                                                    |     |
|--------|--------------------------------------------------------------------|-----|
| 圖 2.1  | Network Architecture of PON.....                                   | 3   |
| 圖 2.2  | Downstream traffic over EPON.....                                  | 4   |
| 圖 2.3  | Upstream traffic over EPON.....                                    | 5   |
| 圖 2.4  | GATE Operation.....                                                | 6   |
| 圖 2.5  | REPORT Operation.....                                              | 6   |
| 圖 3.1  | IPACT 演算法步驟(I).....                                                | 8   |
| 圖 3.2  | IPACT 演算法步驟(II).....                                               | 9   |
| 圖 3.3  | IPACT 演算法步驟(III).....                                              | 10  |
| 圖 3.4  | IPACT 演算法步驟(IV).....                                               | 10. |
| 圖 4.1  | Deficit Round Robin 步驟(I).....                                     | 14  |
| 圖 4.2  | Deficit Round Robin 步驟(II).....                                    | 15  |
| 圖 4.3  | Deficit Round Robin 步驟(III).....                                   | 15  |
| 圖 4.4  | Deficit Round Robin 步驟(IV).....                                    | 16  |
| 圖 4.5  | Code for Deficit Round Robin.....                                  | 17  |
| 圖 4.6  | ONU inter-transmission time.....                                   | 21  |
| 圖 4.7  | Dual DRR flow chart.....                                           | 24  |
| 圖 4.8  | ONU maximum waiting time for transmission.....                     | 25  |
| 圖 5.1  | Two state bursty traffic model.....                                | 27  |
| 圖 5.2  | Mean delay variance vs. Loading.....                               | 28  |
| 圖 5.3  | burstiness = 5 , Mean delay variance vs. Loading.....              | 29  |
| 圖 5.4  | burstiness = 10 , Mean delay variance vs. Loading.....             | 30  |
| 圖 5.5  | burstiness = 16 , Mean delay vs. Loading.....                      | 31  |
| 圖 5.6  | burstiness = 16 , Mean delay variance vs. Loading.....             | 32  |
| 圖 5.7  | burstiness = 21 , Mean delay variance vs. Loading.....             | 32  |
| 圖 5.8  | burstiness = 21 , Mean delay variance vs. Loading.....             | 33  |
| 圖 5.9  | burstiness = 80 , batch =10 , Mean delay vs. Loading.....          | 33  |
| 圖 5.10 | burstiness = 80 , batch =10 , Mean delay variance vs. Loading..... | 34  |

表目錄

Table 5.1 Network parameter used in simulation.....26



## Chapter 1 Introduction

近年來隨著網際網路的發展，各種整合了語音、影像以及數據的多媒體程式相繼的問世，這些即時性的多媒體需要不間斷且有效率的連線服務品質，對頻寬的要求也是一大挑戰，目前大部分解決寬頻骨幹網路方案為非數位用戶迴路(Digital Subscriber Line)與電纜數據機(Cable Modem)，但是近幾年隨著高品質數位電視(HDTV)、隨選視訊(Video on Demand)等高解析度影像的推出與應用，上述的方案已無法支援高度頻寬需求的多媒體程式[1][2]，故光纖的發展與應用已成為目前解決頻寬不足的最佳選擇及必經途徑，只需要透過一條光纖網路就可以傳輸聲音、影像以及資料類型的封包，這也是所有使用者期盼的方法。

在 1980 的末期，英國的電信公司開始利用被動光纖網路的技術提供網路及電信服務給客戶，此舉也開始了被動光纖網路技術的應用[3]，最後能夠實現光纖到府(Fiber-to-the-Home)或是光纖到大樓(Fiber-to-the-Building)的電信服務。此外，被動光纖網路之所以具有吸引力的部分，只要是在用戶端與中央交換機(Central Office)之間的所有元件都是屬於「被動式(Passive)」，所謂的「被動式」在於元件不需要額外提供電源即可動作，好處在於鋪設網路時不需要考慮複雜的電源問題，也可以大幅降低鋪設後的維修費用，總而言之，被動式光纖網路提供一種便利且容易安裝與維護的架構。

到了 2004 年的 6 月，電子電機工程師協會(IEEE)制定了名為 IEEE 802.3ah 的技術標準規範[6]，針對乙太被動式光纖網路(EPON)作完整的定義與規範[4]。所謂的乙太被動式光纖網路(EPON)，顧名思義就是結合了「乙太網路(Ethernet)」及「被動式光纖

網路(Passive Optical Network)」的技術，承襲了乙太網路既有的便宜、簡單、高頻寬等方式，並且使用被動的元件鋪設，可大幅降低鋪設的困難及機房維護成本，因此乙太被動式光纖網路已成為目前建置光纖網路架構的首選。

隨著乙太被動光纖網路(EPON)成為網路界的新寵兒，所推出滿足各種服務品質的演算法也如雨後春筍般的出現，但是在解決公平的分配頻寬(fairness)一直是一個很重要的議題，因此在我們所研究的主題中，將提出一個適用於乙太被動光纖網路的演算法來解決公平性(fairness)的問題，讓屬於負載(load)的末端使用者可以順利的傳送資料同時不影響到使用率(utilization)，然而當負載(load)增加的時候適當的做控制以期達到動態分配頻寬的原則。

我們將在接下來的章節介紹此演算法應用在乙太被動式光纖網路(EPON)上詳細的過程與步驟，並且最後再透過證明確實可以達到公平性原則和模擬的結果驗證對封包延遲的改善。

本篇論文如下面所介紹的方式編排，首先第二章先介紹被動式光纖網路的架構與下上傳的方式，第三章介紹利用 IPACT 的方法使得 OLT 可以管理每個 ONU 的傳頻寬與時間，第四章則是介紹 Deficit Round Robin 演算法的應用與特點，並且透過適當的改善可以更適合用於被動式光纖網路上，第五章是模擬所使用的環境介紹與結果的分析和比較，最後一章就是把前面的討論做整理和未來研究的方向探討。

## Chapter 2 System Architecture

### 2.1 Structure Of EPON

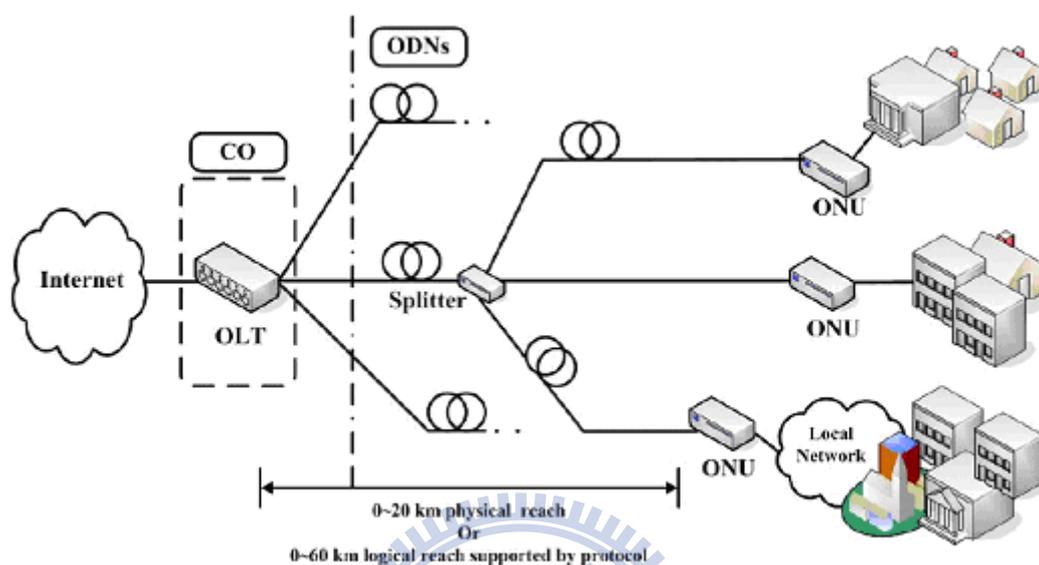


圖 2.1：Network Architecture of PON

在介紹以太被動式光纖網路(EPON)之前，我們必須要先對被動式光纖網路(PON)有徹底的了解，被動光纖網路是屬於樹狀的結構，如圖 2.1 所示。在中央局端(Central Office)部分的稱作光學式線路終端器(Optical Line Terminal，OLT)，負責分配和控制信號的傳遞，並且直接連結到骨幹網路。光學式分散網路(Optical Distribution Network)負責連接 OLT 到 ONU(Optical Network Unit)之間的所有網路設備，例如分配器(Splitter)，這部份的設備都是屬於被動式並且可提供的連線範圍從 0 到 20 公里不等。光學式網路終端器(Optical Network Terminal/Unit)負責接收來自 OLT 的所有信號，最後再送達給末端的使用者，除此之外，它也負責上傳使用者的資料到 OLT。

最後進入以太被動式光纖網路(EPON)的部份，和被動式光纖網路(PON)的差異在於可以透過目前以太網路的封包(Ethernet Network)直接存取網路。對目前世界各國來說，大部分都是以以太被動式光纖網路(EPON)為基礎並作建構，好處是可以直接和以太網路做結合，並且也可以大幅減少鋪設的成本與系統建構的時間。

## 2.2 Downstream and Upstream over EPON

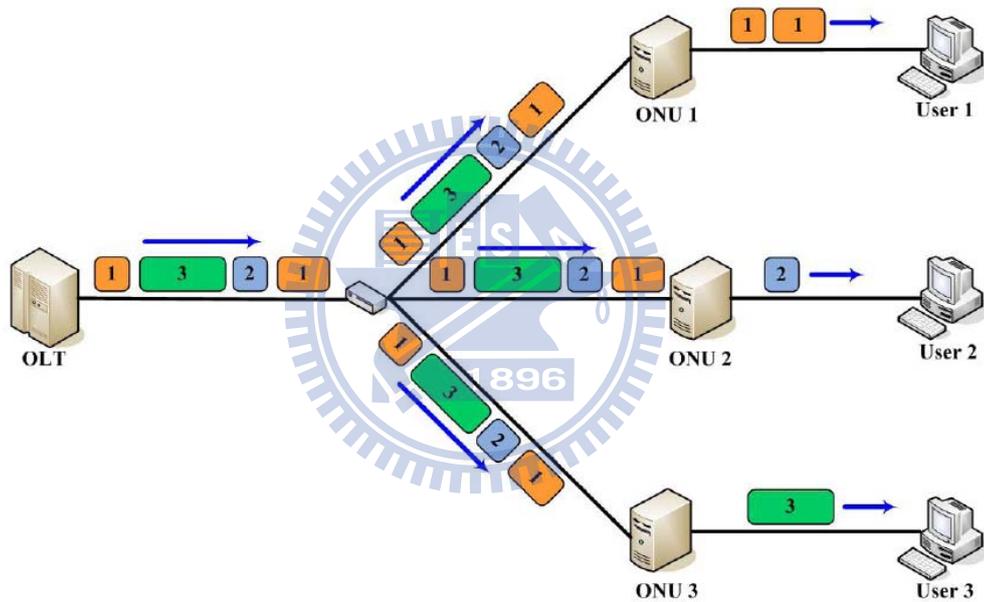


圖 2.2 : Downstream traffic over EPON

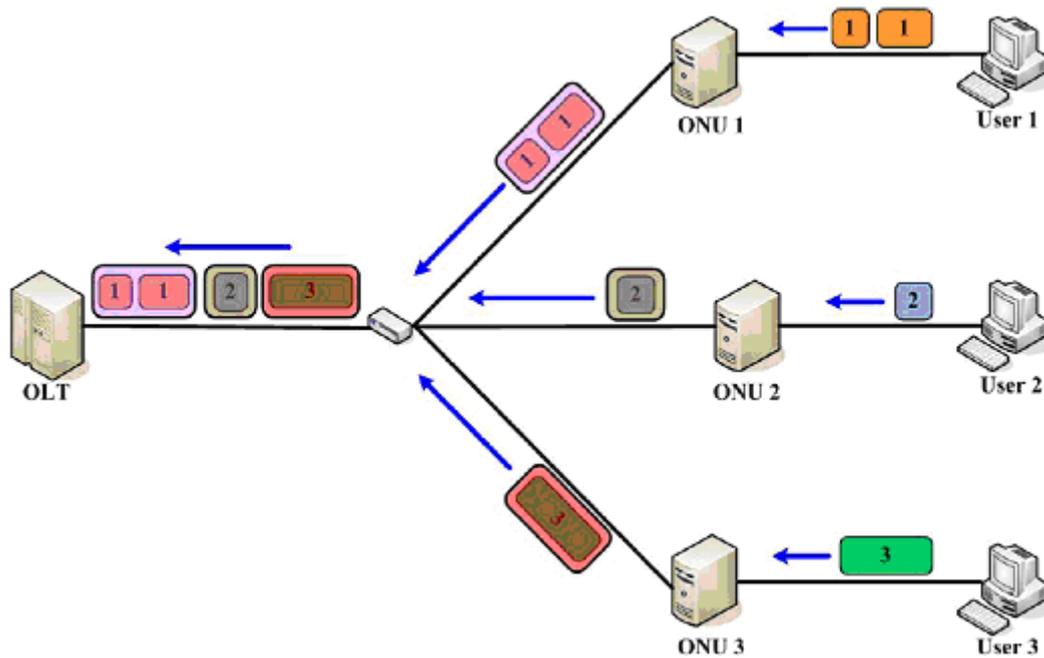


圖 2.3 : Upstream traffic over EPON

在 EPON 上，從 OLT 傳送到 ONU 的方向的封包(packet)稱為資料下傳(Downstream Traffic)；反之，如果是從 ONU 傳送到 OLT 方向的封包(packet)稱為資料上傳(Upstream Traffic)，圖 1.2 和圖 1.3 所代表的分別是在乙太光纖網路(EPON)上的資料下傳(Downstream)和資料上傳(Upstream)。資料下傳是使用廣播(broadcast)的方式，經過下傳路徑上的分配器(splitter)平均地把分配信號功率到每個輸出口[5]，最後讓所有的 ONU 收到由 OLT 送出的資料，然後 ONU 先將收到光訊號轉換成電的訊號並且把屬於自己的封包(packet)再繼續送給最後的使用者，也就是只會留下和本身擁有相同實體網路位址(MAC address)的封包，剩下的封包則是被捨棄掉。從上述的部分我們可以得知在資料下載(Downstream)時所有的封包(packet)都沒有發生碰撞的情形。然而，在資料上傳(Upstream)的方面就會有發生碰撞的問題，因此我們需要利用一些控制訊息讓每個 ONU 在上傳資料不會發生碰撞。

### 2.3 Multi-Point Control Protocol(MPCP)

Multi-Point Control Protocol(MPCP)[9]是 IEEE 802.3ah 的一種標準規範,目的是為了解決在被動式光纖網路上頻寬分配的問題,「GATE」和「REPORT」為 MPCP 的其中兩種控制訊息。OLT 會送出「GATE」訊息給 ONU,通知 ONU 何時可以開始上傳資料和允許上傳資料的時間(Transmission Window);至於「REPORT」則是 ONU 通知 OLT 目前本身所需要的頻寬大小、緩衝暫存區的容量大小、以及時間間隔等等。

## GATE Operation

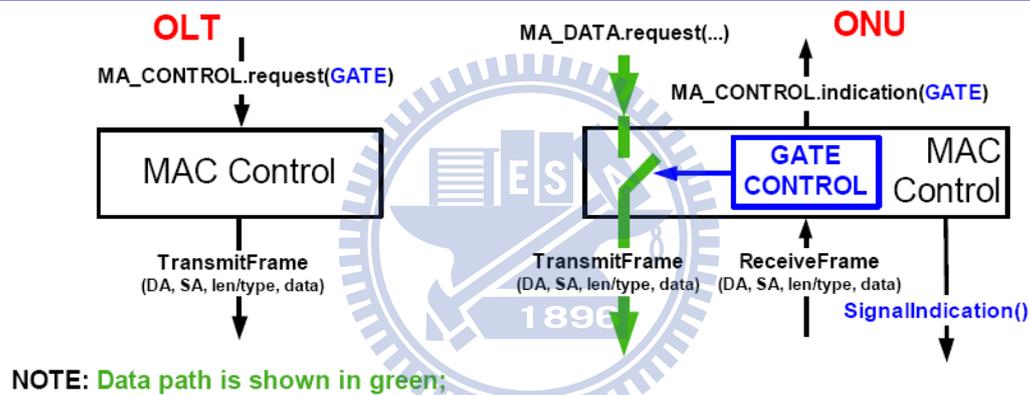


圖 2.4 : GATE Operation

## REPORT Operation

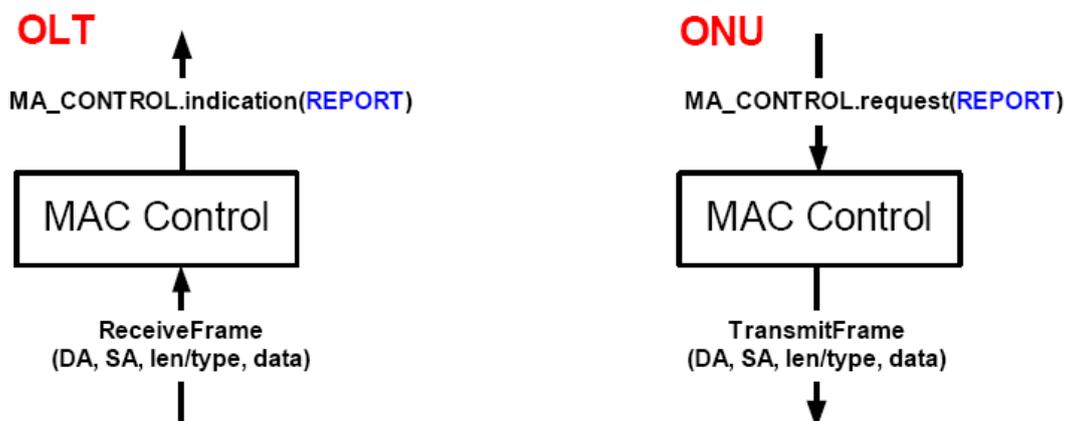


圖 2.5 : REPORT Operation

## 2.4 EPON advantage and challenging

與其他的網路比較，EPON 目前所擁有的優點如下：

- 與現有的乙太網路(Ethernet)相容能力：目前乙太網路是最普遍也最成熟的網路技術，基本上，EPON 上從 IEEE 802.3 標準所延伸出來的，因此，它必定可以相容於乙太網路的系統。
- 提供廣大的頻寬：EPON 可提供 Gigabit 等級的傳輸速率，可以應付目前所有的多媒體程式。
- 低成本：EPON 的網路架構中，都是由許多被動元件組成的，可以同時減少事前鋪設與事後維修的成本。



### Chapter 3 Dynamic Bandwidth Allocation

在目前許多的動態分配頻寬(Dynamic Bandwidth Allocation)方法中，由 Glen Kramer、Biswanath Mukherjee 和 Gerry Pesavento 所提出的 IPACT (Interleaved Polling with Adaptive Cycle Time)最受到大家所注目[4]，接下來我們將詳細介紹 IPACT 的步驟流程與方法。

#### 3.1 Interleaved Polling with Adaptive Cycle Time

這個小節我們將對 IPACT 動態頻寬分配方法做一個詳細的論述，這裡我們使用了一個 OLT 和三個 ONU 來做說明。

如圖 3.1 所示，假設 OLT 已經知道目前所有 ONU 想要傳送的資料量的大小，並且也知道每個 ONU 到 OLT 的 Round Trip Time(RTT)，這些訊息都會先紀錄在 OLT 的 polling table 中。

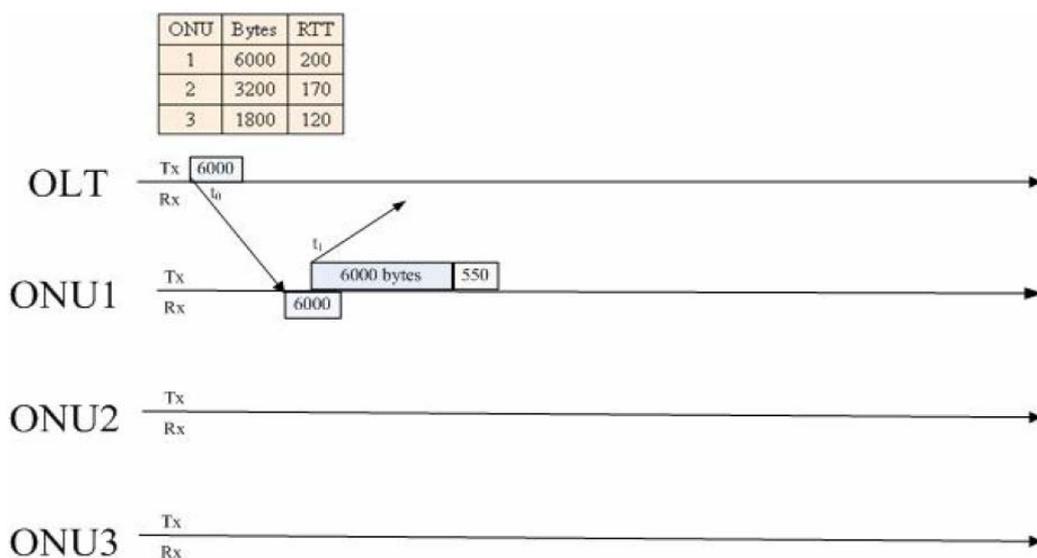


圖 3.1 IPACT 演算法步驟(I)

在時間  $t_0$  時，OLT 先送出一個 Grant 訊息給  $ONU_1$ ，通知它可以送出大小為 6000 Bytes 的資料，這個 Grant 訊息包含了目的地  $ONU_i$  的 ID 和傳送時間的大小 (Transmission Window)。當  $ONU_1$  在時間  $t_1$  收到由 OLT 送來的 Grant 訊息時，便會開始上傳資料直到最大可允許傳送的上限，在我們所舉的例子，這筆資料大小為 6000 Bytes。

當  $ONU_1$  上傳資料結束之後， $ONU_1$  還會在多送出一個「Request」控制訊息，目的是在告訴 OLT 目前還有多少資料量仍然在佇列(buffer)裡面，由圖 3.1 可得知，目前這筆資料量為 550 Bytes。

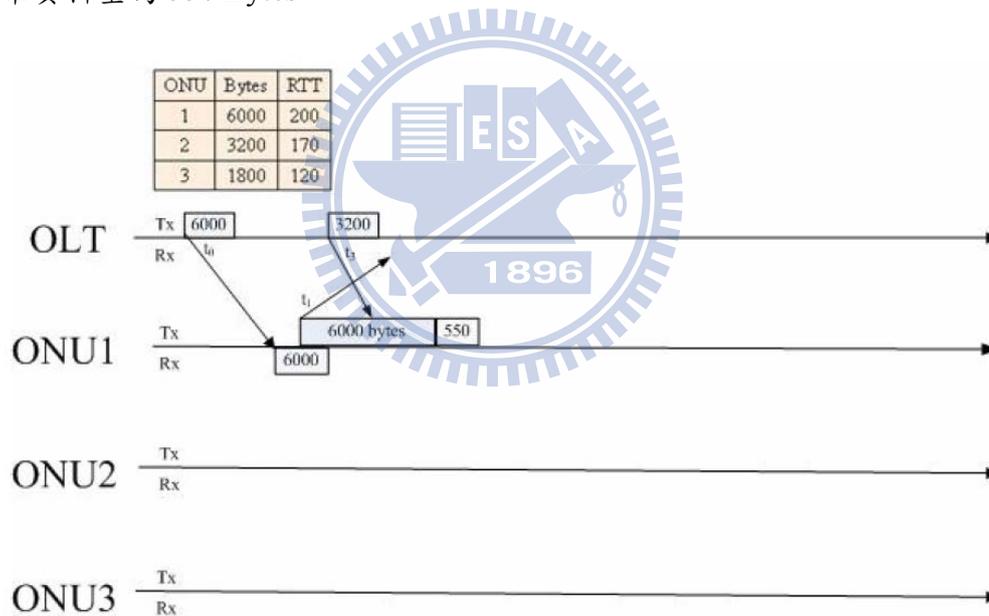


圖 3.2：IPACT 演算法步驟(II)

由於 OLT 會知道每個  $ONU_i$  的 Round Trip Time(RTT)時間和傳送結束的時間，於是會在適當的時間先送出 Grant 控制訊息給接下來需要上傳資料的  $ONU_i$ 。由圖 3.2 可得知，OLT 在時間  $t_3$  時，提早送出一個允許傳送 3200 Bytes 的 Grant 控制訊息給  $ONU_2$ 。

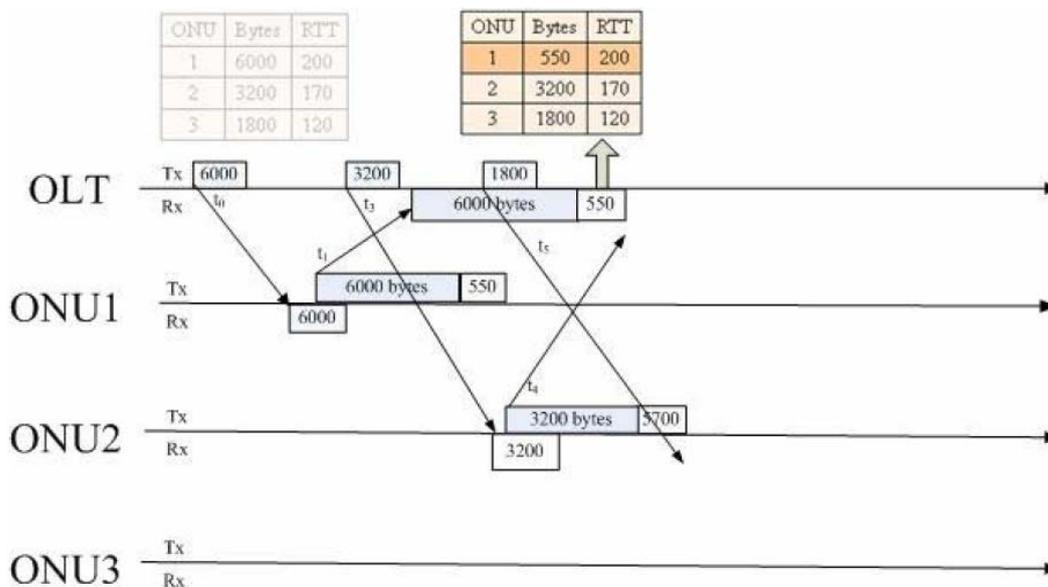


圖 3.3：IPACT 演算法步驟(III)

當 OLT 在接收 ONU<sub>1</sub> 的上傳資料和 Request 訊息時，會立刻更新 polling table 內的資訊，如圖 3.3，當 OLT 收到 ONU<sub>1</sub> 的 Request 訊息，立刻將原本的 6000 Bytes 改為 550 Bytes。由於 OLT 也會算出 ONU<sub>2</sub> 上傳資料的結束時間，在時間  $t_5$  會事先送出 Grant 訊息給 ONU<sub>3</sub> 並且允許傳送 1800 Bytes 的資料。

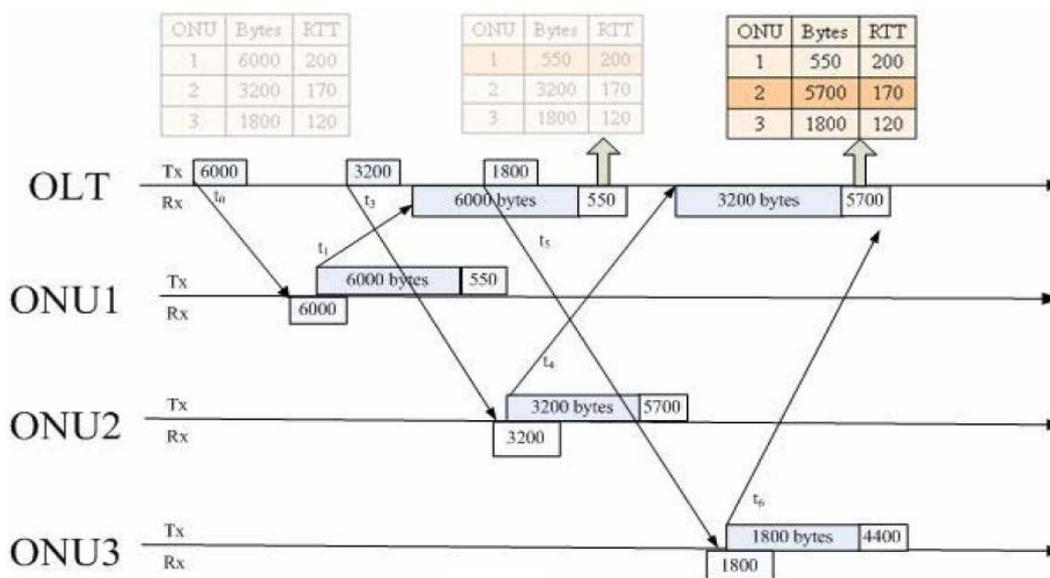


圖 3.4：IPACT 演算法步驟(IV)

這次和圖 3.3 類似，當 OLT 收到 ONU<sub>2</sub> 的 Request 訊息時，會立刻將 polling table 的資訊更新。圖 3.4 表示 OLT 將 ONU<sub>2</sub> 的 Request 數量由 3200 Bytes 改為 5700 Bytes。

### 3.2 Some Dynamic Bandwidth Allocation Algorithm

這個小節我們將提出一些現有的動態頻寬分配(Dynamic Bandwidth Allocation)的方法，並且根據所擁有的優勢和缺陷去做探討。

#### (i) Fixed Service

顧名思義為標準的 Time Division Multiplexing Access(TDMA)方法，就是不論 ONU 所要求的頻寬為多少，OLT 每次都分配相同大小的頻寬給 ONU 使用。好處是非常的公平(fairness)，但是使用效率(utilization)來說相對的低落。

#### (ii) Limited Service

針對 TDMA 的改善方法，代表 OLT 會滿足每個 ONU 的 Request，只要 Request 頻寬小於  $W_{max}$ ；反之，如果 Request 大於  $W_{max}$ ，ONU 還是只能獲得  $W_{max}$  大小的頻寬。

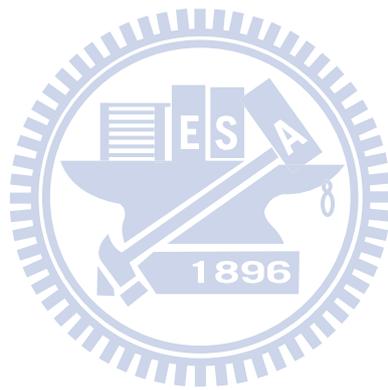
$$B_i = \begin{cases} Request, & \text{if } request < W_{MAX} \\ W_{MAX}, & \text{if } request > W_{MAX} \end{cases}$$

#### (iii) Elastic Service

這種分配頻寬的方法主要是移除每個 ONU 所得到的頻寬不能超過  $W_{max}$  的上限。假設  $T_{cycle} = N \times W_{max}$ ， $N$  為 ONU 的數目。只要所有的 ONU 分配的頻寬總和小於  $T_{cycle}$  即可，要是我們假設現在只有一個 ONU 要上傳資料，那麼這個 ONU 就可以獲得所有的頻寬，也就是  $N \times W_{max}$ 。

上面所提出的幾種動態分配頻寬的方法均互有利弊，但是都忽略掉了一個重要的

議題，那就是所謂的公平的分配頻寬，大部分的方法都只注重要如何滿足目前 ONU 所需要的頻寬，而沒有注意到經過長時間的服務後，是否每個 ONU 都能得到相同大小的頻寬，這是目前許多方法所沒有考慮到的。



## Chapter 4 Problems and Solution

在這章節裡我們將使用 M. Shreedhar and George Varghese 所提出的 Deficit Round Robin 演算法[7]，並且用 DRR 演算法做動態頻寬分配，最後針對所碰到的瓶頸與問題去做討論。

### 4.1 Deficit Round Robin

一開始 Deficit Round Robin，簡稱 DRR，主要是應用於網路的交換器，目的是為了公平的分配頻寬給每個輸入埠(input port)，現在我們將它用於 EPON 上去做討論。

假設每個  $ONU_i$  都會有封包來到，並且 OLT 會給每個 ONU 大小等於  $Q$  的籌碼， $Q$  就等於每個 ONU 可以得到的頻寬，當  $ONU_i$  收到籌碼  $Q$  之後，會將它放進存錢筒 (Deficit Counter) 裡面，簡稱  $DC_i$

$$DC_i = DC_i + Q$$

所以  $DC_i$  的大小就代表  $ONU_i$  這次可以傳送的資料數量；如果現在  $DC_i$  開始上傳資料，那麼  $ONU_i$  會將現在擁有的  $DC_i$  和排在最前面的封包去比較大小，假設  $DC_i$  的值大於封包大小  $L$ ，那麼立刻將封包送出並且  $DC_i$  相對的減少  $L$ ，

$$DC_i = DC_i - L$$

接下來再將  $DC_i$  和下一個封包大小做比較，持續做這個步驟直到  $DC_i$  小於封包的大小，或是已經沒有封包可以送出了，於是再換下一個  $ONU_j$  開始上傳資料。

但是發生  $DC_i$  大於零並且卻沒有任何的封包可以送出時，這時我們會將  $DC_i$  歸零，

$$DC_i = 0$$

下面我們使用例子說明 DRR 實際的運作方式

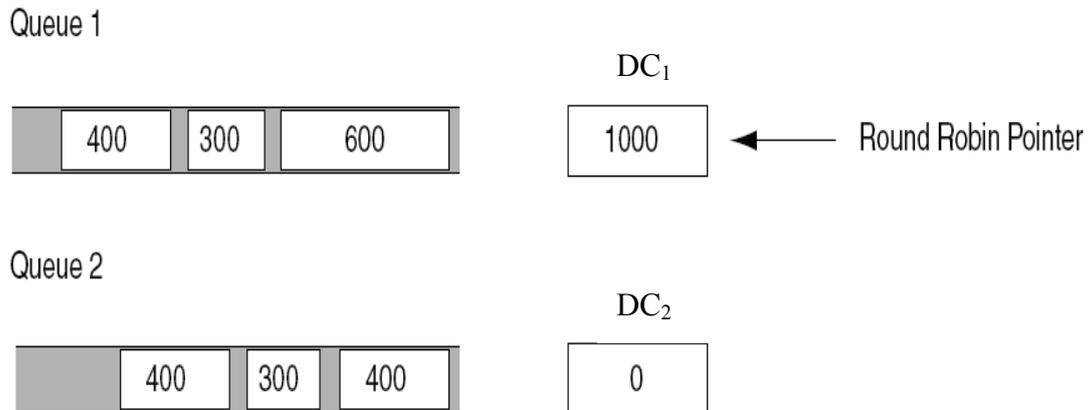


圖 4.1：Deficit Round Robin 步驟(I)

如圖 4.1，假設現在有兩個 ONU<sub>s</sub> 都有封包等待傳送，每次都可以獲得籌碼  $Q = 1000$  bytes。假使 ONU<sub>1</sub> 先開始傳送，ONU<sub>1</sub> 會先獲得 1000 bytes 的籌碼，並且將籌碼放入存錢筒 DC<sub>1</sub> 裡面，首先將 DC<sub>1</sub> 和第一個封包大小做比較，由於  $1000 > 600$ ，因此可以順利送出封包並且 DC<sub>1</sub> 也相對減少 600 bytes，繼續再把剩下 DC<sub>1</sub> 和下一個封包做比較，因為  $400 > 300$ ，所以第二個封包仍然可以順利送出，但是當封包大小為 300 bytes 送出之後，由於 DC<sub>1</sub> 只剩下 100 bytes，因為接下來封包的大小為 400 bytes 必需等到下一次才能送出，剩下的籌碼 100 bytes 也會被留下來提供下次使用。

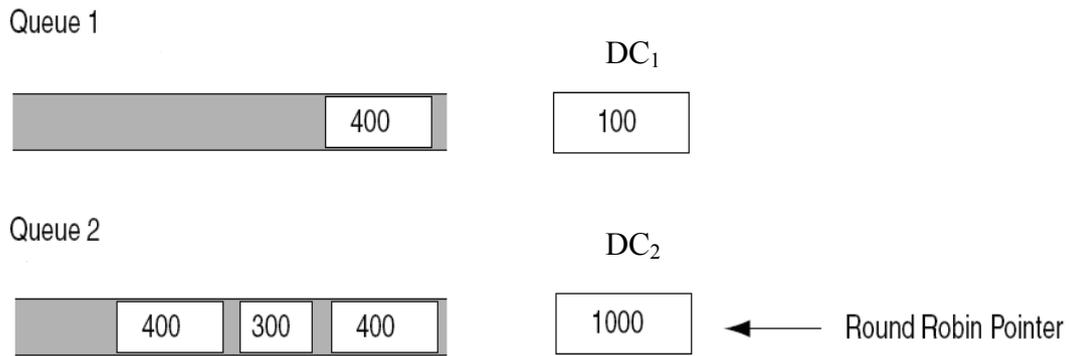


圖 4.2：Deficit Round Robin 步驟(II)

當  $ONU_1$  結束上傳後，接下來就換  $ONU_2$  開始送出資料，並且方法和之前所陳述的相同，如圖 4.2 所示。

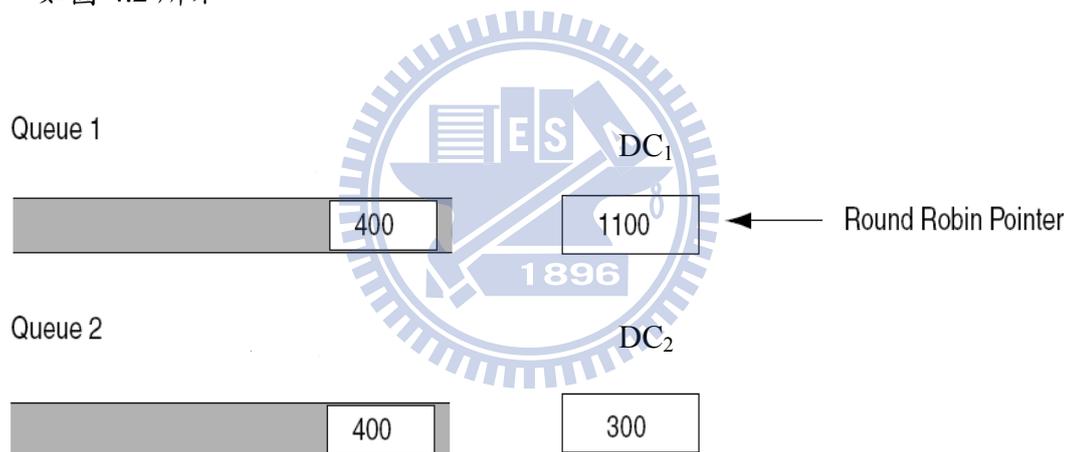


圖 4.3：Deficit Round Robin 步驟(III)

當  $ONU_2$  完成傳送資料後，接下來又會在輪到  $ONU_1$  送出封包，一樣是先給  $ONU_1$  籌碼  $Q=1000$  bytes，並且放入  $DC_1$  的存錢筒裡面，開始將  $DC_1$  和封包大小作比較，因為  $1100 > 400$ ，所以可以順利的送出封包同時  $DC_1$  也被更新為 700 bytes，但是  $ONU_1$  已經沒有任何的封包需要送出了，這時候我們就將  $DC_1$  的值設為零。如圖 4.3 和圖 4.4 所示。

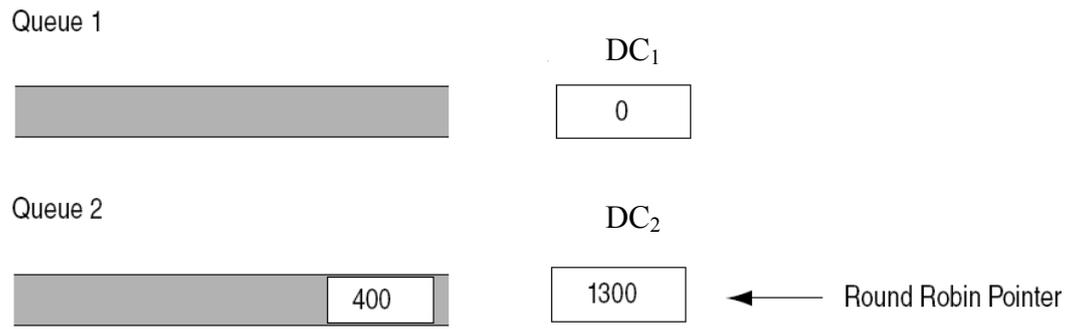
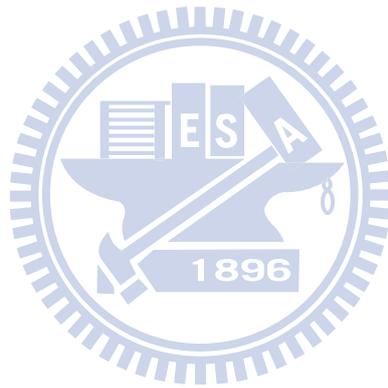


圖 4.4 : Deficit Round Robin 步驟(IV)



下面為 DRR 在實行上的演算方法

**Initialization:**

```
For ( $i = 0; i < n; i = i + 1$ )  
     $DC_i = 0;$ 
```

**Enqueuing module:** on arrival of packet  $p$

```
 $i = ExtractFlow(p)$   
If ( $ExistsInActiveList(i) == FALSE$ ) then  
     $InsertActiveList(i);$  (*add  $i$  to active list*)  
     $DC_i = 0;$   
If no free buffers left then  
     $FreeBuffer();$  (* using buffer stealing *)  
     $Enqueue(i, p);$  (* enqueue packet  $p$  to queue  $i$ *)
```

**Dequeuing module:**

```
While(TRUE) do  
    If  $ActiveList$  is not empty then  
        Remove head of  $ActiveList$ , say flow  $i$   
         $DC_i = Q_i + DC_i;$   
        while ( $(DC_i > 0)$  and  
            ( $Queue_i$  not empty)) do  
             $PacketSize = Size(Head(Queue_i));$   
            If ( $PacketSize \leq DC_i$ ) then  
                 $Send(Dequeue(Queue_i));$   
                 $DC_i = DC_i$   
                     $- PacketSize;$   
            Else break; (*skip while loop *)  
        If ( $Empty(Queue_i)$ ) then  
             $DC_i = 0;$ 
```

圖 4.5 : code for Deficit Round Robin

## 4.2 Fairness issue

了解 DRR 的演算法如何運作之後，我們接下來要開始證明 DRR 可以達成公平分配頻寬的原則。在證明之前，我們必須先討論之後證明會用到參數，其中參數可能擁有的上限值。

(i)對於所有的  $ONU_i$ ， $DC_i$  值的範圍一定會滿足  $0 \leq DC_i < L_{max}$ ， $L_{max}$  表示最大的封包尺寸

(proof) 一開始我們定義  $DC_i$  的起始值為零，所以可得到  $DC_i$  的最小值為零。

除此之外，由於所有的封包大小都不會超過  $L_{max}$ ，剩下的  $DC_i$  絕對不會超過  $L_{max}$ ，由此可得證。

(ii)在任意的一段時間( $t_1, t_2$ )內， $ONU_i$  所送出的資料量一定會大於  $m \cdot Q - L_{max}$ ，並且小於  $m \cdot Q + L_{max}$ ， $m$  代表在( $t_1, t_2$ )時間內被服務的次數，

$$m \cdot Q - L_{max} \leq \text{sent}_i(t_1, t_2) \leq m \cdot Q + L_{max}$$

(proof)假設在任何的( $t_1, t_2$ )時間內， $ONU_i$  都是屬於有封包需要被送出得情況。令  $DC_i(k)$  為  $ONU_i$  在完成第  $k$  次資料上傳後，所剩下的  $DC_i$  值； $\text{bytes}_i(k)$  為  $ONU_i$  第  $k$  次所送出的資料量，所以在時間( $t_1, t_2$ )內經過  $m$  次的服務之後所送出的資料量為

$$\text{sent}_i(m) = \sum_{k=1}^m \text{bytes}_i(k)$$

因此，從 DRR 的演算法可得知

$$\text{bytes}_i(k) + DC_i(k) = Q + DC_i(k-1)$$

$$\Rightarrow \text{bytes}_i(k) = Q + DC_i(k-1) - DC_i(k)$$

最後再將  $\text{bytes}_i(k)$  做累加函式，將  $k$  從 1 開始加到  $m$  為止，可得到

$$\text{sent}_i(m) = m \cdot Q + DC_i(0) - DC_i(m)$$

由(i)可得知  $0 \leq DC_i < L_{max}$ ，可推得

$$-L_{max} \leq DC_i(0) - DC_i(m) \leq L_{max}$$

代入後即可得証。

(iii)在任何的一段時間( $t_1, t_2$ )內， $ONU_i$ 和  $ONU_j$ ， $i \neq j$ ，彼此獲得頻寬相差一定不會大於  $2L_{max} + Q$ 。

(proof)

由前面(ii)的推導可得知

$$\text{sent}_i(t_1, t_2) \leq m \cdot Q + L_{max}$$

$$\Rightarrow \text{sent}_i(t_1, t_2) = (m-1) \cdot Q + Q + L_{max}$$

再從(ii)又可得知

$$\text{sent}_j(t_1, t_2) \geq n \cdot Q - L_{max}$$

$$\Rightarrow \text{sent}_i(t_1, t_2) - \text{sent}_j(t_1, t_2) \leq (m-n)Q + 2L_{max} + Q$$

由於  $ONU_i$  和  $ONU_j$  都是一直有封包需要送出，因此被服務的次數不會相差超過一

次， $|m-n| \leq 1 \Rightarrow n \geq m-1$

$$\Rightarrow \text{sent}_i(t_1, t_2) - \text{sent}_j(t_1, t_2) \leq (m-n)Q + 2L_{max} + Q$$

$$\leq 2L_{max} + Q$$

可得証。

在探討有關公平分配頻寬(fairness bandwidth allocation)的問題， $ONU_i$  和  $ONU_j$  一定要是屬於高負載(over loading)的情況，由於得到的頻寬無法滿足到來的封包(packet)數量，所以需要公平的分配頻寬給每個 ONU。

### 4.3 Modified Deficit Round Robin

經過 4.2 節的介紹之後，我們可以得知 Deficit Round Robin 演算法可以解決因為封包大小的變動而造成頻寬分配不公平(unfairness)的現象，並且在實際的運作上複雜度(complexity)也相對較低，因此在實現上的目標較容易達到。

但是如果考量封包來的情況的話，DRR 演算法就會無法完成有效率的分配頻寬，例如當封包出現的情況是 two-state model，就會發生只有在其中的一段時間內會有封包密集的到來並且等待被送出，而其他的時間則是看不見任何的封包，這時根據 DRR 演算法，ONU 所能得到的最大頻寬為  $L_{max}$ ，這樣的頻寬分配方法會無法滿足針對封包延遲很在意的應用程式。

所以在這部分我們提出了 DC 不歸零的機制，也就是當傳送完成之後，不會檢查是否有封包還在佇列(Queue)裡面。這個方法就是當  $ONU_i$  在沒有任何封包進入的這段時間內，不會將 OLT 每次所給的籌碼  $Q$  給收回去，可以一直無限的繼承下來，因此當 ONU 之後開始有封包到來的時候，可以使用的籌碼不會被限制只有  $L_{max}$ ，因此可以讓許多的封包在經過一次的 Report 之後，即可送出許多已經累積在佇列(Queue)內

的封包，這個方法我們稱為 Deficit Round Robin with non reset to zero。

經過上面的探討，可以得到其實不歸零這個機制對短時間內來很多封包的 ONU 來說，可以獲得很好的封包延遲。但是在被動光纖網路架構下，只要有一個  $ONU_i$  在上傳資料，其他的  $ONU_j$  是屬於無法送出的狀態，必須一定要等到  $ONU_i$  完成傳送之後，才可以換自己開始送出資料，這種情形會讓其他的  $ONU_j$  可能要等待很久的時間才會輪到自己開始送出資料，我們透過下面的圖來討論可能發生的情況

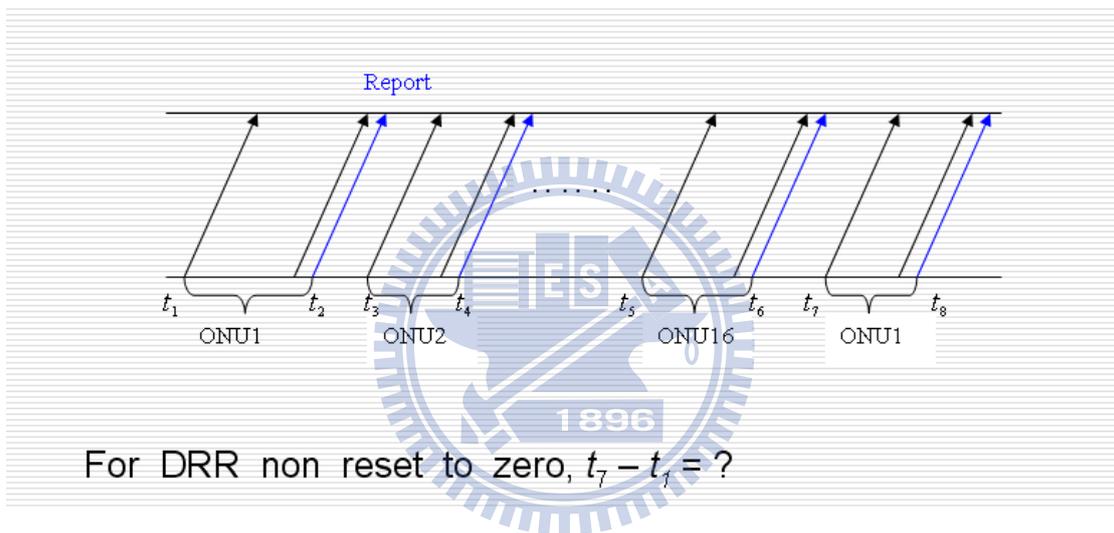


圖 4.6：ONU inter-transmission time

當  $ONU_I$  在時間  $t_2$  完成上傳資料時，並且送出 Report 下一次所需要的頻寬給 OLT，之後 OLT 會根據其他 ONU 的 Report 訊息，計算出  $ONU_I$  下一次可以開始送出資料的時間  $t_7$ ，可是時間  $t_7$  我們卻無法事先預知，因此會無法保證  $ONU_I$  有最長等待時間(waiting time)，因而導致無法滿足目前許多對封包的延遲有需求的串流媒體(video)等，因此在最後我們提出了 Dual Deficit Round Robin 演算方法，同時改善 DRR 和 DRR non reset to zero 所遇到的瓶頸及問題，以期能滿足目前網路流量的需求。

#### 4.4 Dual Deficit Round Robin non reset to zero

Dual DRR 演算法顧名思義就是使用了兩組的 DRR 去計算可送出的資料量，第一組 DRR 和前面所陳述的一樣，OLT 每次將大小等於  $Q$  的籌碼給  $ONU_i$ ，並且不會因為沒有封包可以傳送而將 DC 的值歸零，因此  $ONU_i$  所擁有的 DC 值一定會是相當可觀的數量，如果這時有很多封包抵達，那麼  $ONU_i$  就會運用前面所累積的 DC 值，將大量的封包一次全部送出。因此為了不讓這種可遇見的情形發生，於是我們設計了第二組 DRR，重新再計算一次  $ONU_i$  所允許傳送的資料量。

再這裡我們先定義幾組參數

MAX\_transmission：表示允許  $ONU_i$  每一次所能送出的最大資料量。

$Q_2$ ：第二組 DRR 每次可得到的籌碼大小。

$DC_{2,i}$ ：第二組 DRR 所累積的籌碼大小。

如果  $ONU_i$  的 Report 需要的頻寬小於 MAX\_transmission，那麼第二組 DRR 機制不會發生作用，直接 Report 需要的頻寬給 OLT；但是當 Report 需要的頻寬是大於 MAX\_transmission 的時候，這時就會將第二組的 DRR 機制啟動。

首先我們先將  $DC_i$  扣掉和  $Q_2$  相同大小的數值，並且將籌碼大小  $Q_2$  加到存錢筒  $DC_{2,i}$  裡面，最後再使用 DRR 演算法重新計算所需求的頻寬大小。在第二組 DRR 演算法的參數設定上，我們必須注意籌碼  $Q_2$  必須大於 OLT 所提供的籌碼大小  $Q$ ，主要的原因是如果  $Q_2$  小於  $Q$  的前提下，重新計算的結果就會和一開始的 DRR 演算法(DRR reset to zero)相同，那麼第二組 DRR 的演算法就會形同是虛設的。

#### 4.4.1

第二組 DRR 啟動機制與重新計算頻寬的演算法

**If** ( $Report_i > MAX\_transmission$ ) **then**

$$DC_i = DC_i - Q_2 ;$$

$$DC_{2,i} = DC_{2,i} + Q_2 ;$$

**do DRR using**  $DC_{2,i}$

**reset**  $Report_i = 0 ;$

**while**( $DC_{2,i} > 0$ ) **do**

$$PacketSize = Size(Head(Queue))$$

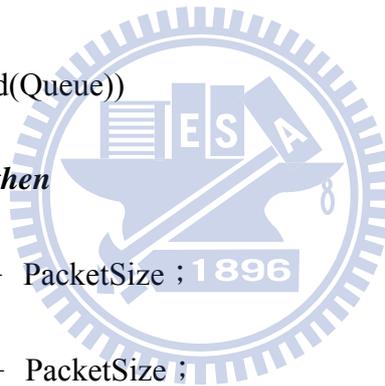
**If** ( $PacketSize \leq DC_{2,i}$ ) **then**

$$Report_i = Report_i + PacketSize ;$$

$$DC_{2,i} = DC_{2,i} - PacketSize ;$$

**Else** **break**;(\*break while loop\*)

**Sent**  $Report_i$  **to** OLT



4.4.2 Dual DRR flow chart

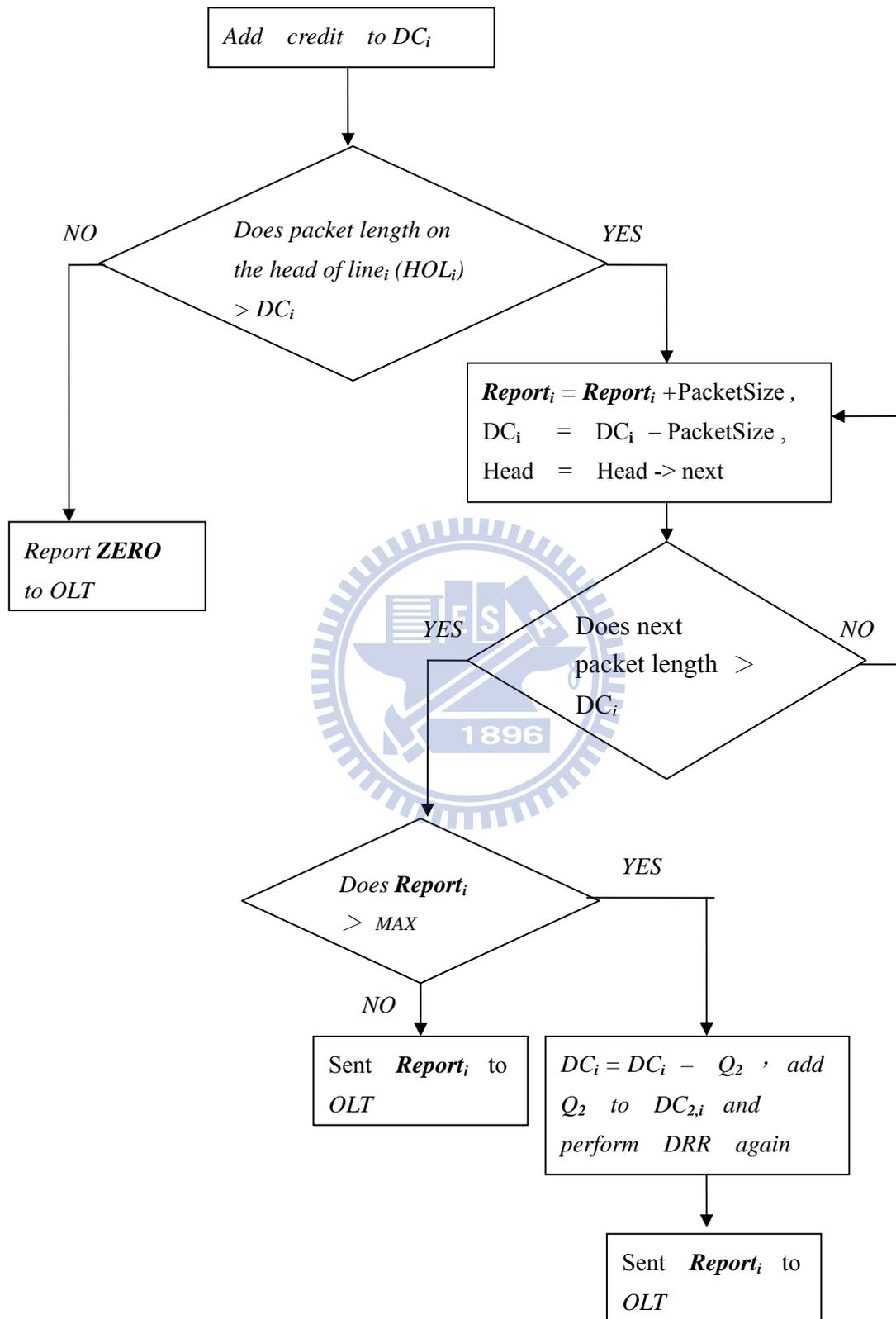


圖 4.7 Dual DRR flow chart

### 4.4.3 Dual DRR inter-transmission time

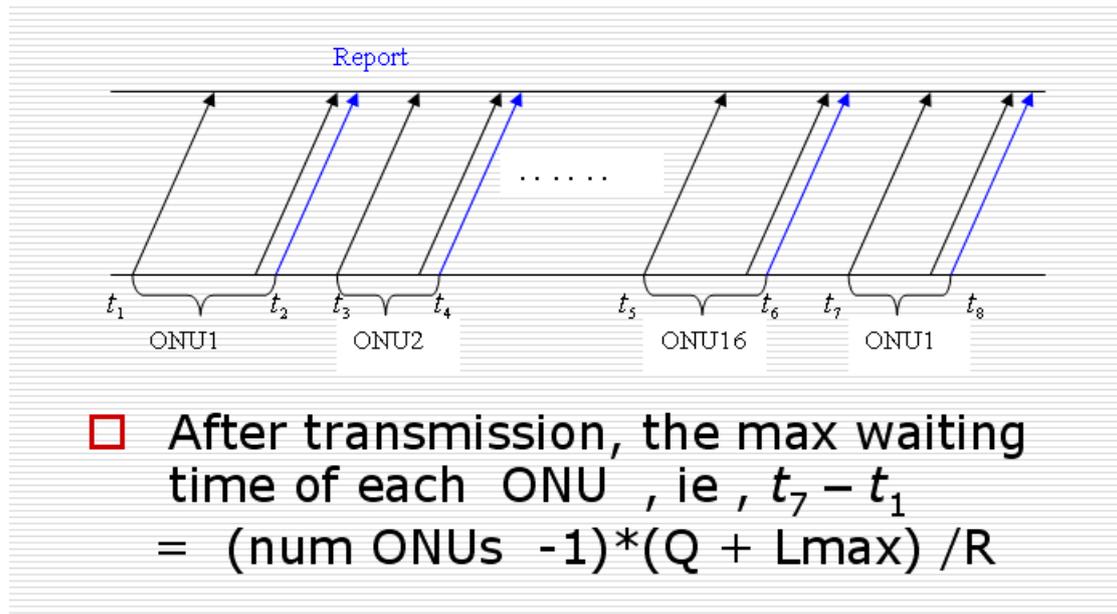


圖 4.8 : ONU maximum waiting time for transmission

將 Dual DRR 演算法使用在 PON 上，我們可以得知每個 ONU 每一次最多送出的資料量為  $Q_2 + L_{\max}$  (結果可參考證明 DRR fairness 部分)，因此可以得知當  $ONU_i$  結束第  $k$  次傳送資料後，則再次輪到自己可以開始傳送資料所需最長的等待時間為

$$(\text{ONUs 數目} - 1) \times (Q_2 + L_{\max}) / R$$

## Chapter 5 Simulation Results and Performance Analysis

這個章節我們主要是針對之前我們所提出的 Dual DRR 演算法進行模擬，並且根據模擬的結果做分析和討論，驗證是否可以達到我們之前所提出的結論。

### 5.1 Network parameter and traffic model

在開始探討模擬的結果之前，我們必須先設定好所使用的參數大小，如 Table 5.1 所示

| Parameter            | Value         |
|----------------------|---------------|
| Number of ONUs       | 16            |
| Transmission rate    | 1Gbps         |
| Per-timeslot         | 70bytes       |
| OLT to splitter      | 20km          |
| Splitter to each ONU | 4km           |
| Packet size          | 64~1518 bytes |
| Buffer size          | infinite      |
| Batch                | 10            |

Table 5.1 Network parameter used in simulation

我們所使用的 traffic 為 two state bursty traffic，代表了封包來的方式有兩種 arrival rate，這種模擬的結果比較能貼近網路上實際的封包流量情況；分別為 high state 和 low state，當在 high state 時有較高的 arrival rate  $\lambda_H$ ；反之，在 low state 時有較低的 arrival rate  $\lambda_L$ ，如圖 5.1 所示。

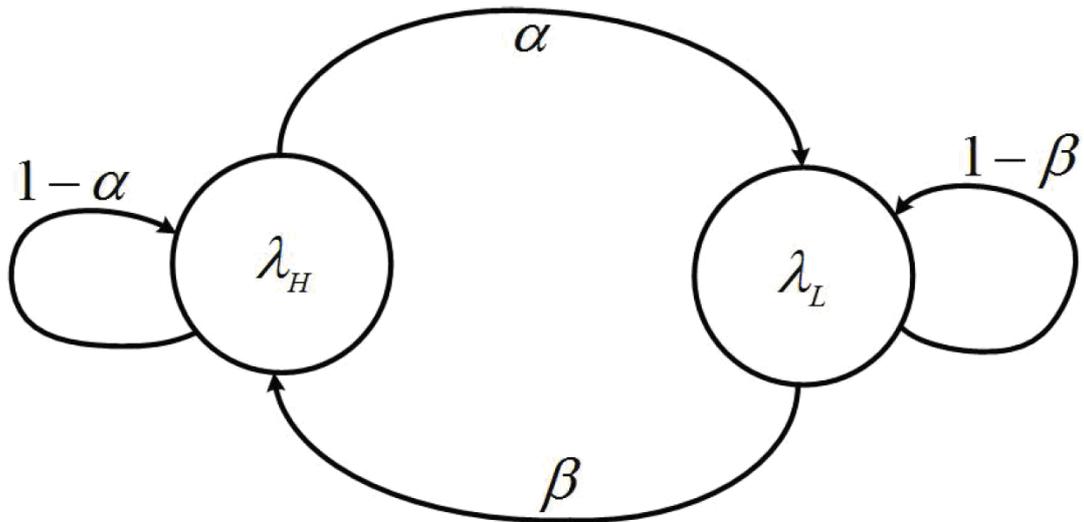


圖 5.1 Two state bursty traffic model

當 traffic 在 high state 時執行完產生封包(packet generation)的動作之後，會有  $\alpha$  的機率會跳到 low state，也會有  $1-\alpha$  的機率維持在 high state 的情況；同理，在 low state 會有  $\beta$  的機率跳到 high state，也會有  $1-\beta$  的機率維持在 low state 的情況。

根據 two state traffic model，我們可以推導出 mean arrival rate

$$\lambda = \frac{\alpha \cdot \lambda_L + \beta \cdot \lambda_H}{\alpha + \beta}$$

(proof)

By Queueing Theory

$$[\text{Prob}(\text{high}) \quad \text{Prob}(\text{low})] \begin{pmatrix} \alpha & 1-\alpha \\ \beta & 1-\beta \end{pmatrix} = [\text{Prob}(\text{high}) \quad \text{Prob}(\text{low})]$$

$$\text{Prob}(\text{high}) = \frac{\beta}{\alpha + \beta}, \quad \text{Prob}(\text{low}) = \frac{\alpha}{\alpha + \beta}$$

$$\text{Mean arrival rate } \lambda = \frac{\beta}{\alpha + \beta} \cdot \lambda_H + \frac{\alpha}{\alpha + \beta} \cdot \lambda_L = \frac{\alpha \cdot \lambda_L + \beta \cdot \lambda_H}{\alpha + \beta}$$

故可得證。

並且我們也定義了 Burstiness 代表這個 traffic bursty 的情形

$$B(\text{Burstiness}) = \frac{\lambda_H}{\lambda}$$

## 5.2 Simulation results

這部份為了簡化模擬的複雜性，我們將所有的單位一律改為 timeslot，所以經過正規化(normalize)之後，

Timeslot : 70 bytes

Packet size : 1 ~ 22 timeslots

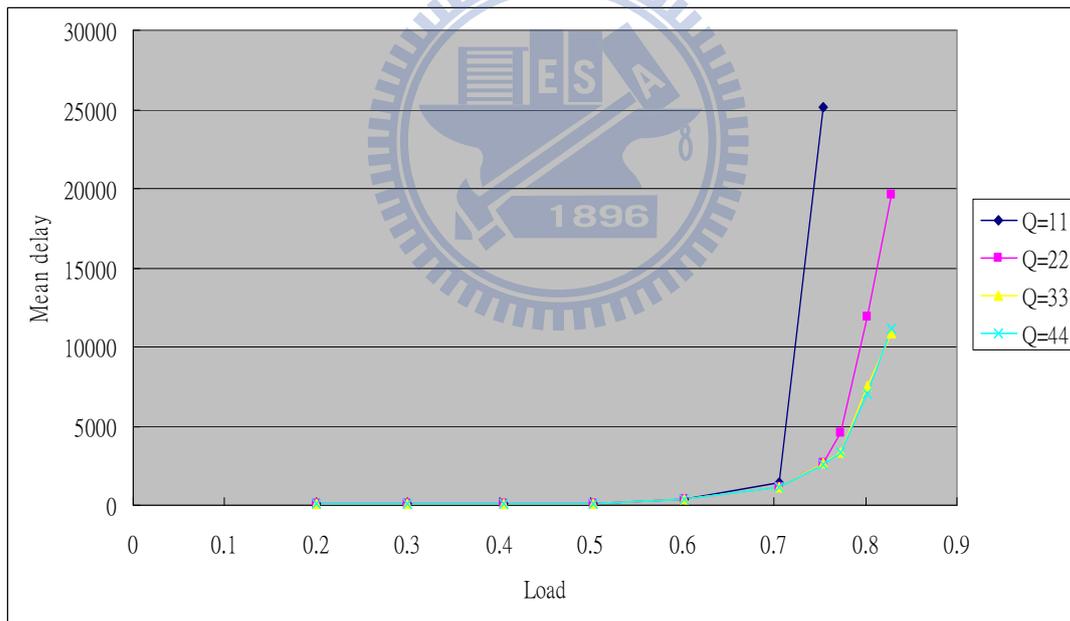


圖 5.2 Mean delay variance vs. Loading

由圖 5.2 為固定 burstiness =5，分別比較 OLT 給予不同的籌碼大小 Q 所造成的平均延遲(mean delay)影響。由於籌碼大小 Q 是 OLT 所給予的，會直接受到影響的就是可送出的總封包數量，如果 OLT 給予的籌碼太少會使得封包一直累積在 ONU 的佇

列(Queue)裡面，每個封包都必須等待很久的時間才會送出去，因此會造成延遲(delay)的大幅增加，除此之外，由於每個封包都無法得知需要等待多久才會被送出，因此每個封包的延遲就會屬於不穩定的狀態，造成的平均延遲(mean delay)的快速增加。

所以 OLT 只給予籌碼  $Q=11$  是無法滿足隨著 ONU 負載(load)逐漸的增加，當  $Q$  增加到 22 或 33 timeslots 的時候，可以發現平均延遲變異數(mean delay variance)會大幅的下降，表示累積的籌碼已經足夠應付所有的封包，並且就算增加到 44 timeslots 也不會再有更好的效能，所以之後模擬我們使用的籌碼大小  $Q$  就為 33 timeslots。

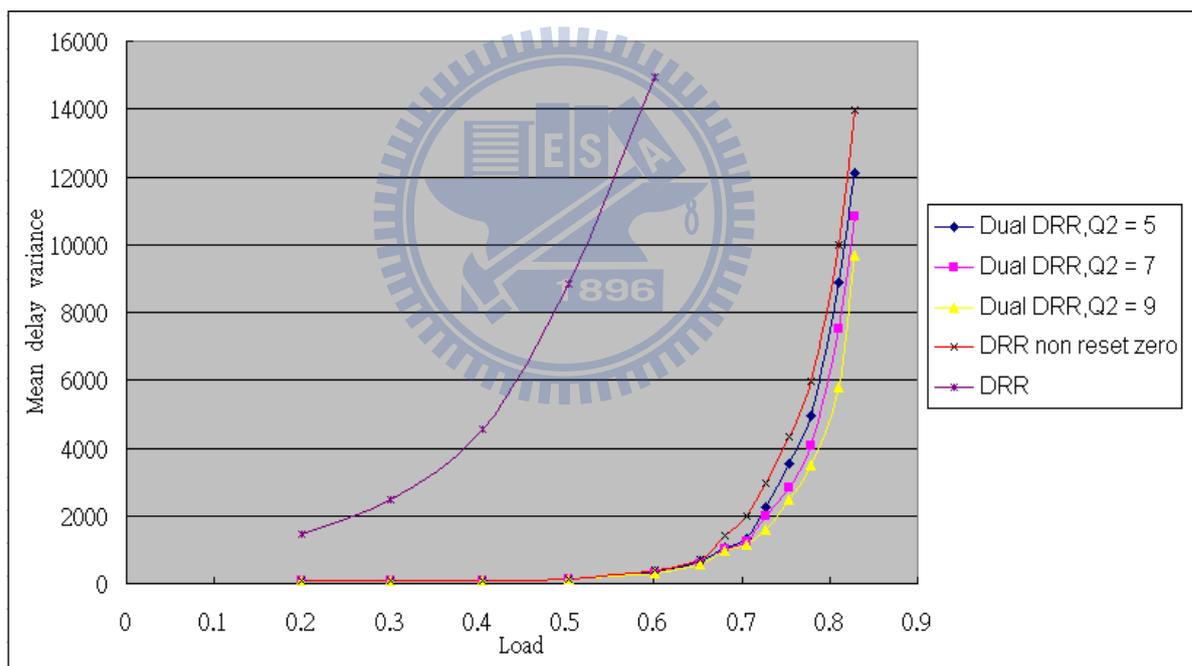


圖 5.3 burstiness = 5，Mean delay variance vs. Load

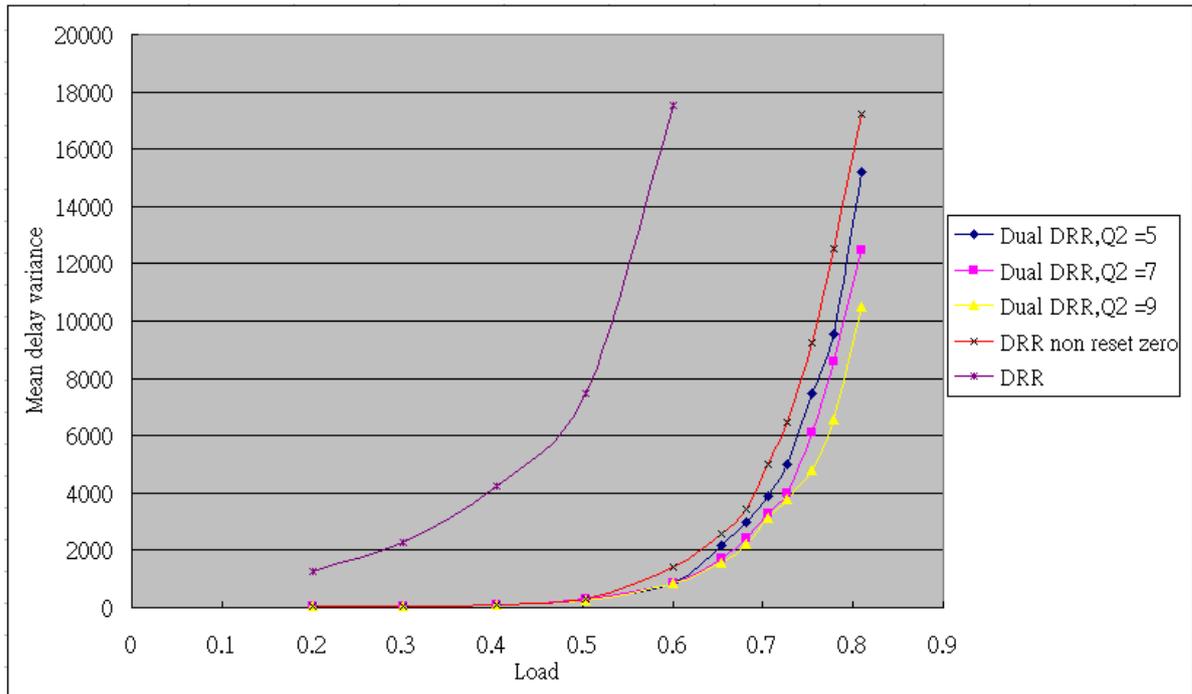


圖 5.4 burstiness = 10，Mean delay variance vs. Loading

由於使用了 Dual DRR 演算方法，每個 ONU 都有最長的等待再次傳送時間，所以彼此封包的延遲差異均不會很大，所以可以讓平均延遲變異數(mean delay variance)增加的速度較為緩慢。

由圖 5.4、圖 5.5 可以看出隨著 burstiness 的增加，改善的情形會變的更明顯，這是由於越高的 burstiness 代表相同的封包數量，但是會在越短的時間內到來，因此更加容易發生要等待很久的時間才可以再次送出封包，所以透過 Dual DRR 限制每次傳送的封包數量，可以讓每個 ONU 都公平的獲得頻寬。最後再看的是每次可以從 DC 拿取的籌碼  $Q_2$  大小分別為  $5L_{\max}$ 、 $7L_{\max}$ 、 $9L_{\max}$ ，代表每次最多所允許送出的封包數量為  $(Q_2 + L_{\max})$ ，當  $Q_2$  取的值太小會讓每次得到的頻寬太少，但是每個 ONU 的等待傳送時間會變小，所以  $Q_2$  的取捨是個比較尷尬的議題，原則上取  $Q_2 = 7L_{\max}$  會是個較好的選擇，由於  $Q_2 = 9L_{\max}$  改善情形已經不大，這部份在之後的圖還會有更進一步

的說明。

最後我們在考慮一種情況，就是假設在一個 timeslot 之間，可以產生的封包數目是超過一個以上，這種產生的封包數目我們稱為 Batch，產生封包數目的方法使用 Geometric generation，下面的圖我們假設 Batch = 10，表示做多一次產生的封包數目等於 10。

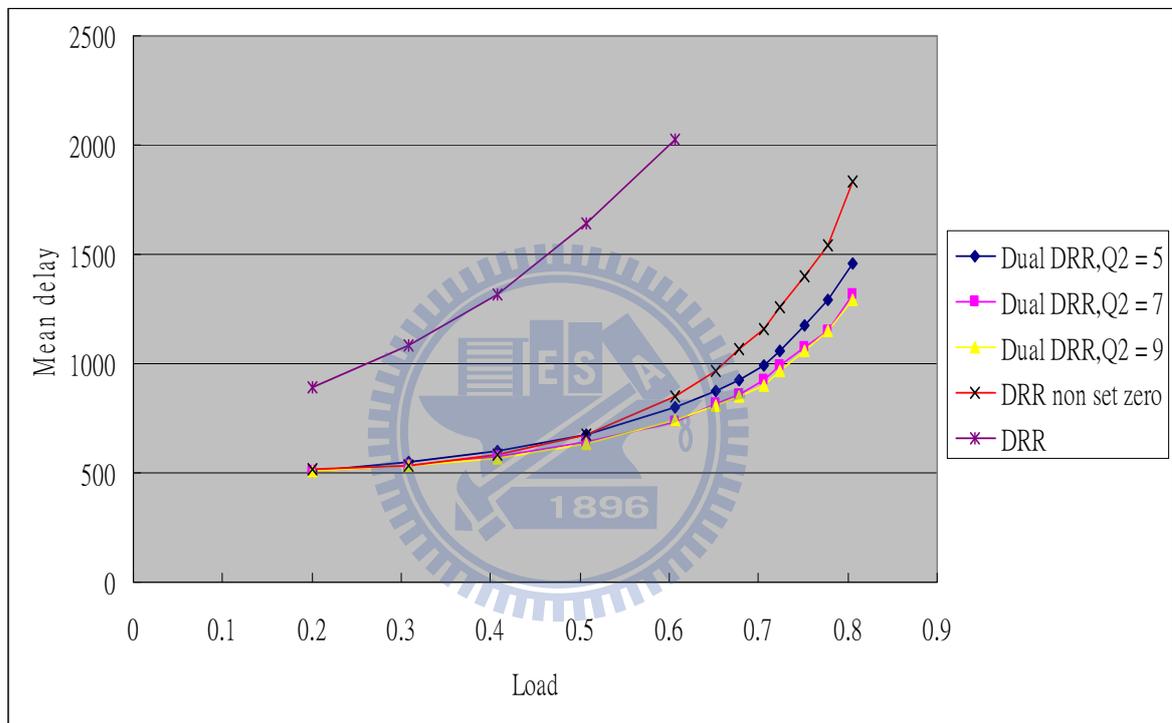


圖 5.5 burstiness = 16，batch = 10，Mean delay vs. Load

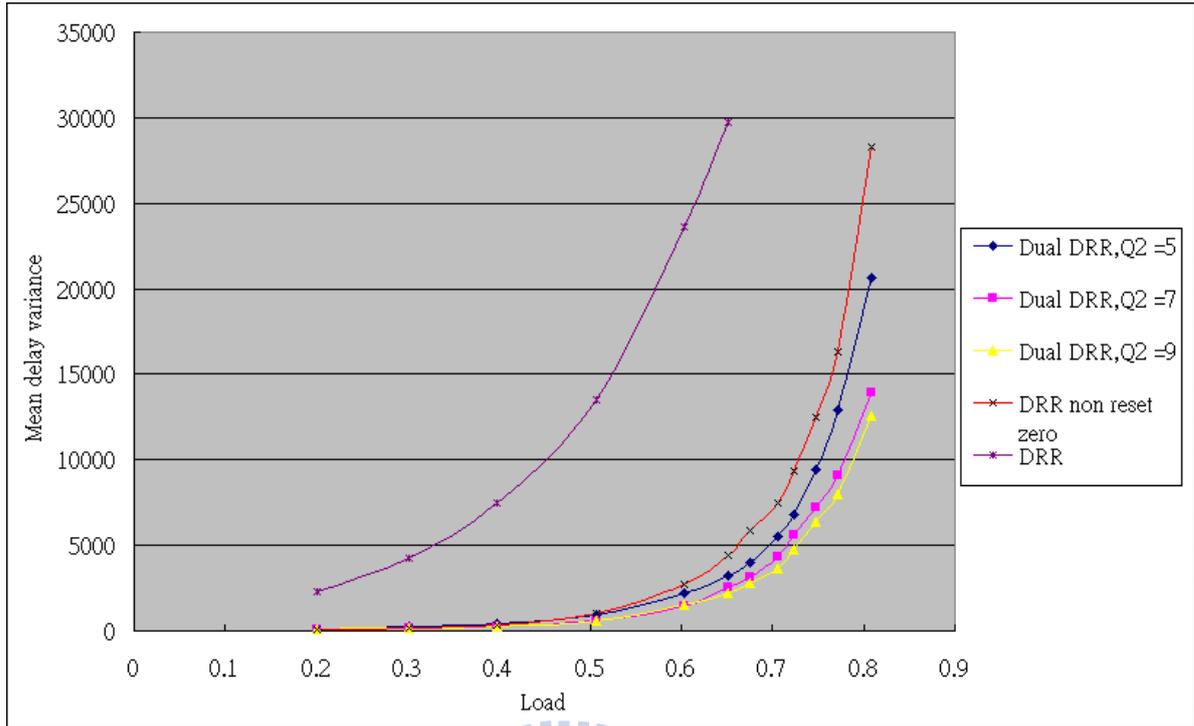


圖 5.6 burstiness = 16 , batch = 10 , Mean delay variance vs. Load

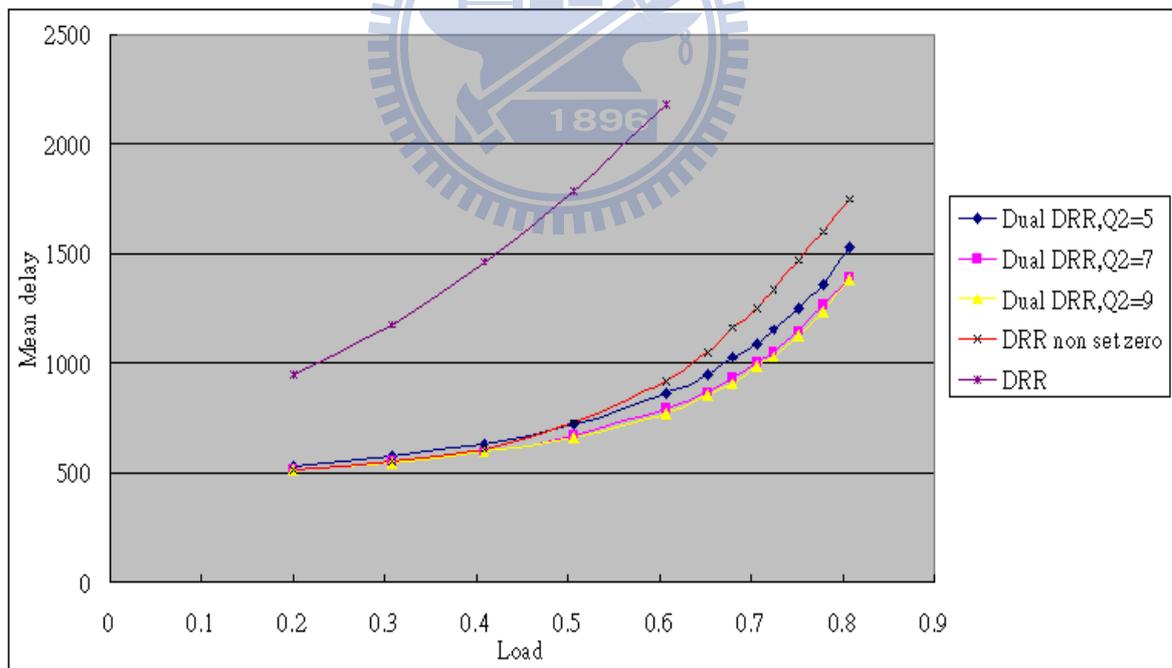


圖 5.7 burstiness = 21 , batch = 10 , Mean delay vs. Load

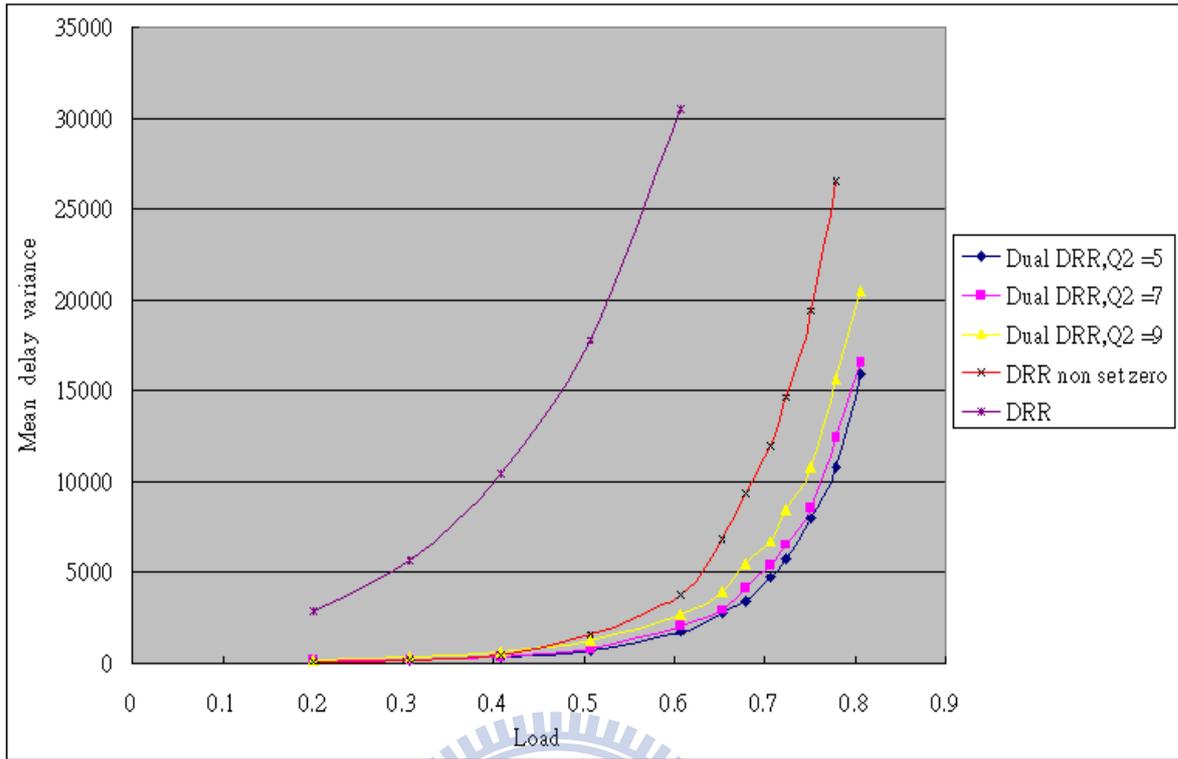


圖 5.8 burstiness = 21 , batch = 10 , Mean delay variance vs. Load

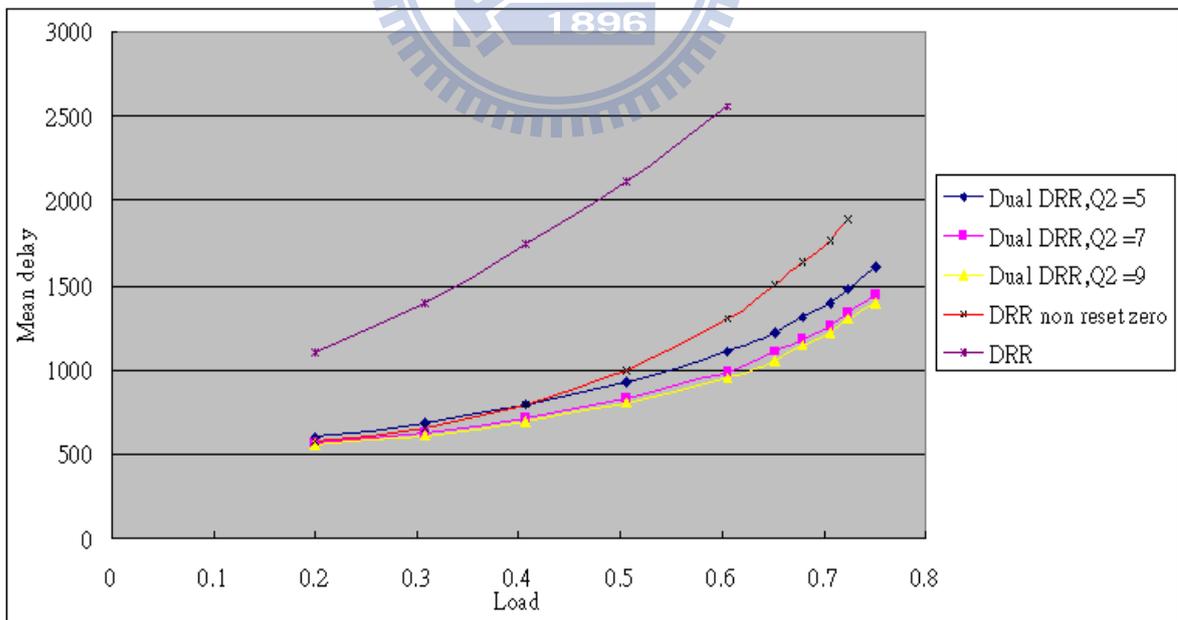


圖 5.9 burstiness = 80 , batch = 10 , Mean delay vs. Load

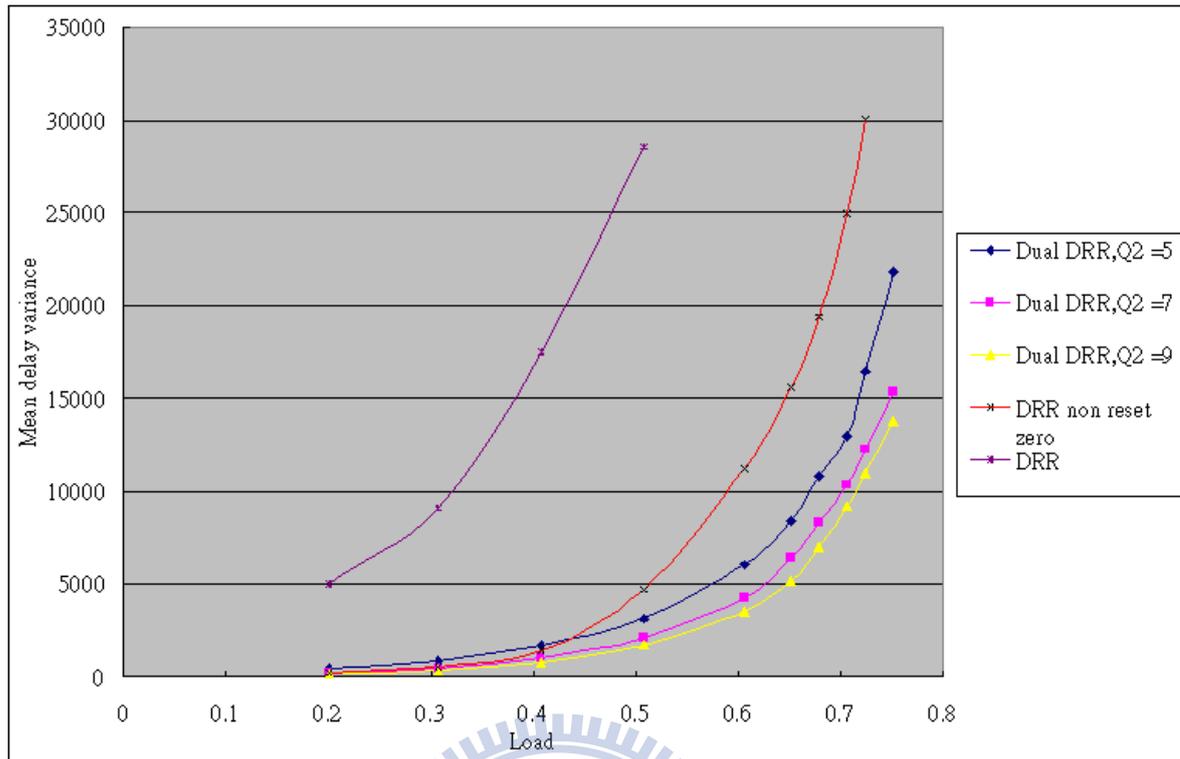


圖 5.10 burstiness = 80, batch = 10, Mean delay variance vs. Load

由於加了 Batch 的情況，代表 ONU 所看到的封包數目會更加的極端，也就是原本佇列(Queue)是屬於空的狀態，但是可能在下一次看到時就會是許多封包塞滿佇列(Queue)的情況，對公平的分配頻寬來說又是一項更嚴苛的挑戰，從圖 5.7 到圖 5.10 可以看出，DRR 和 Dual DRR 兩種演算方法的平均封包變異數(mean delay variance)差距變得更加明顯，尤其是在負載(load)等於 0.5 就可以開始看出變化，表示加了 Batch 之後 traffic bursty 的情況更為嚴重，透過 Dual DRR 演算方法限制的最大上傳頻寬，即可將不同封包的延遲差距給降低。

## Chapter 6 Conclusion and Future Works

在這篇的論文裡，我們主要是改善原本 Deficit Round Robin 演算法的缺點，目的是可以更加的適用於在被動式光纖網路(PON)上，不僅可滿足原先作者所提出的 fairness 要求，也改善了網路上的 bursty traffic 對頻寬分配上所產生的問題，畢竟降低封包的延遲(mean delay)和延遲變異數(mean delay variance)是我們目前所極需解決的問題。

在未來的研究上，隨著許多應用程式不斷的向前發展與進步，更多的標準也會慢慢的被許多人所制定出來，因此當使用單一的波長(wavelength)上傳資料已無法滿足使用者需求並且多波長技術(multi-wavelength)可以變的更趨成熟之後，這時多波長(multi-wavelength)技術和 Deficit Round Robin 演算法的整合又會是一個研究的新方向。

Reference:

- [1] Yu-mi Kim, Jung Yul Choi, Jeong-hee Ryou, Hyun-mi Baek, Ok-sun Lee, Hong-shil Park, and Minho Kang, “Cost Effective Protection Architecture to Provide Diverse Protection Demands in Ethernet Passive Optical Network”, in International Conference on Communication Technology Proceedings (ICCT 2003) (IEEE, 2003), Vol. 1, pp. 721–724.
- [2] Glen Kramer and Gerry Pesavento, Alloptic, Inc., “Ethernet Passive Optical Network(EPON): Building a Next-Generation Optical Access Network, Communications Magazine, IEEE, Feb 2002, Vol.40, pp.66–73.
- [3] C. F. Nche, C. Rodgers and D. J. Parish, “Implementing the Orwell Protocol over a fiber-based high-speed ATM network”, Electronics & Communication Engineering Journal, Dec 1992, Vol. 4, pp.345-350.
- [4] Glen Kramer and Biswanath Mukherjee, University of California, Davis, GerryPesavento, Alloptic, Inc. Livemore, “IPACT: A Dynamic Protocol for an Ethernet PON (EPON)”, Communications Magazine, IEEE, Feb 2002, Vol.40, pp.74-80.
- [5] Tang Shan Raycom Co., Ltd., Ji Yang, Cheng Sheng Beijing University of Posts and Telecommunications. “EPON Upstream Multiple Access Scheme”, Info-tech and Info-net, 2001. Proceedings. ICII 2001 - Beijing. 2001 International Conferences on, 2001, Vol. 2, pp.273-278.

- [6] IEEE Standards 802.3ah, 2004.
- [7] M. Shreedhar et al., "Efficient Fair Queuing Using Deficit Round-Robin", IEEE /ACM Trans on Networking, vol 4, no 3 (1996).
- [8] G. Kramer and G. Pesavento, "Ethernet Passive Optical Network : Building a Next-Generation Optical Access Network," IEEE Communications Magazine, Vol. 40, No. 2, pp. 66-73, Feb.,2002.
- [9] Mastrodonato, R. and Paltenghi, G., "Analysis of a bandwidth allocation protocol for Ethernet passive optical networks (EPONs),"Transparent Optical Networks, 2005, Proceedings of 2005 7th International Conference, July 2005, Vol. 1, pp.241-244.
- [10] Yongqing Zhu, Maode Ma, and Tee Hiang Cheng, "A novel multiple access scheme for Ethernet passive optical networks," Global Telecommunications Conference, 2003.GLOBECOM '03. IEEE, Dec 2003, Vol. 5, pp.2649-2653.
- [11] Yuanqiu Luo and Ansari, N. "Bandwidth management and delay control over EPONs,"High Performance Switching and Routing, 2005. HPSR. 2005 Workshop on, pp.457-461.