

適用於光學互連網路中群播分割機制之設計

The Design of Multicast Partitioning Mechanisms in Interconnected WDM Optical Networks

吳和庭(Ho-Ting Wu)

國立台北科技大學 資訊工程系
台北市大安區忠孝東路三段一號
National Taipei U. of Tech.
Email:htwu@en.ntut.edu.tw

洪博信(Po-Hsin Hong)

國立台北科技大學 電通所
台北市大安區忠孝東路三段一號
National Taipei U. of Tech.
Email:s8410013@ntut.edu.tw

摘要

本論文中我們提出數個適合光學星狀網路互連架構的群播分割存取協定，分別為回顧佇列存取協定、回顧比例存取協定以及雙重檢視存取協定。我們所考慮的星狀網路具有非同步時槽結構之特性。首先，我們所提出的群播存取協定，從模擬的結果中，可看到在網路區段上展現優異的網路效能。再者，我們又進一步改良此群播存取機制，使之適用於星狀網路互連架構中。模擬的結果可以看出此存取協定能夠發揮連結星狀網路中波長重複使用的特性，以維持良好的網路效能與通道使用率。

關鍵字:分波多工、星狀網路、群播、星狀互連網路。

ABSTRACT

In this paper, we first propose a few multicast partitioning access protocols, called Lookback Queue Access Protocol, Lookback Ratio Access Protocol and Double Check Head Access Protocol, respectively, for the employment in an asynchronous slotted optical star network. By using simulations, we compare and analyze channel utilization levels obtained

by deploying these mechanisms under three general traffic patterns. To be employed in interconnected optical star networks, these algorithms are modified such that wavelength reuse property can be efficiently exploited within the domain of a single star network range. Thus, high network throughput levels and superior channel utilization factors can be sustained in such environments.

Key words : WDM, star, multicast, interconnected optical star,.

二、前言

由於資訊的蓬勃發展，且隨著多媒體在網際網路的應用上與日遽增，比如隨選視訊、遠距教學或視訊會議等應用，使得寬頻通訊網路的建立與有效傳播群播封包的機制，成為熱門之研究話題。近年來分波多工(Wavelength Division Multiplexing, WDM)的技術的發展，在光纖內部，以不同的光波波長，形成多個傳輸通道，在同一時間中傳輸傳輸巨額資料，可以將光纖所提供的廣大頻寬做最佳的利用。也因此，在寬頻網路的研發中，大多以光學分波多工技術為基本的網路傳輸架構。

過去幾年的分波多工光學區域網路研究當中，常以星狀網路做為傳送群播封包的架構，這是因為星狀網路本身的廣播-選擇非常適合傳輸群播封包，而且設計簡易。廣播式的星狀架構下傳輸資料是用廣播的方式廣播到整個網路上，再由接收點接收資料，在此架構下節點若是沒有一個好的存取協定，網路效能將會因為接收器或波長通道發生碰撞而降低，因此設計一適當的存取協定，來避免或減少接收器及波長通道發生衝突的機會便非常重要。由於單一星狀網路有其節點上限[3]，因此存取協定是否適用於多重星狀互連網路，也是存取協定設計中重要的考量因素。

在星狀全光學網路中，網路效能跟網路節點的接收器與傳送器的功能有著相當程度的關連。在本文裡我們考慮的星狀光學網路為具有一集中式的排程控制器的星狀網路架構，網路上的各個節點皆具備有一組可調整至任一波長的發射器與接收器用來存取資料通道及另一組固定波長的發射器與接收器用來存取預約控制通道，透過控制通道網路節點可以傳送預約封包給中央排程控制器以及接收來自中央排程控制器的訊息。節點之間的通訊是先隨機存取方式傳送出預約封包到預約控制通道，來跟中央的排程控制器要求資料通道。當隨機存取成功之後，封包將傳入位於中央的排程控制器裡的佇列。中央排程控制器對於佇列裡的預約封包，依照所定義的排程演算法來排定哪一些節點可以調整到適當的波長資料通道來傳輸或接收資料封包，最後利用控制通道來通知網路上的節點調整發射器或接收器的波長，於某個時間點開始傳輸。在我們所考慮的星狀網路中，其節點是以非同步方式來連接網路，使得網路節點不用實現同步系統因而降低節點的複雜度與成本。

在[5]的研究報告中，作者指出以持續重傳存取協定來分配資料通道，會導致線頭佔線效應的問題，所以可利用隨機重傳隨機延遲存

取協定來分配資料通道。但以此方式也可能增加了不必要的封包傳輸延遲時間。在[4]中，當群播封包發生碰撞時，作者利用分割(partition)的方式先傳送封包至部份目的節點以提高波長的使用率。然而，當分波多工星狀網路架構中實際可用之波長通道數目相當有限時，分割群播封包可能導致接收器的使用率降低。在[1]中，作者提出了回顧存取協定，當碰撞發生時，藉由向後搜尋佇列其它的預約封包，來避免線頭佔線效應，也免除增加延遲時間的困擾以進一步提升網路效能，假如所有搜尋的封包皆有部份目的節點位置發生衝突，則利用分割的方式來傳送位於線頭的封包至部分目的節點。由模擬結果得知，回顧存取協定確實有較佳的效能，因此，只要善加利用中央排程控制器所記錄的資訊，便能有效的提高網路效益。在此篇論文中，我們更提出了數個適用於傳輸群播封包的存取協定-回顧佇列存取協定(LookBack Queue Access Protocol, LBQA)、回顧比例存取協定(LookBack Ratio Access Protocol, LBRA)、前視時槽存取協定(LookAhead Slot Access Protocol, LASA)、前視比例存取協定(LookAhead Ratio Access Protocol, LARA)以及雙重檢視存取協定(Double Check Head Access Protocol, DCHA)。經由模擬，我們比較了各種存取協定，在不同網路環境下，網路效能的表現。

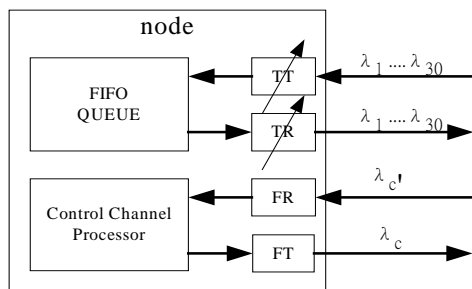
在接下來的幾節中，我們將在第三節介紹單星狀網路的架構及其群播存取協定，第四節則是說明單星狀網路的模擬結果。第五節首先介紹雙星狀網路的架構，接著說明如何修改第三節中所提出的存取機制，使其適用於雙星狀網路，並說明雙星狀網路的模擬結果。第六節為結論。

三、單星狀網路架構與群播存取協定

3.1 單星狀網路架構

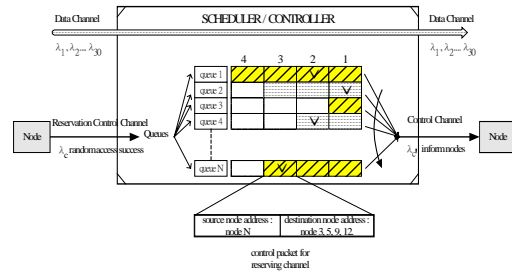
在廣播式的星狀架構下傳輸資料是用廣播的方式廣播到整個網路上，再由接收點調整接收器到適當的波長加以接收資料，在此架構下若是節點用隨機存取方式，因為封包碰撞的結果會導致網路的低效能表現，無法完全發揮 WDM 的特性。若是用預約(reservation)方式 [2]，則需要整個星狀網路其上的所有節點要同步化且須將資料傳輸通道與預約控制通道切割為不同大小的時槽，如此一來則增加網路節點的複雜度。這裡所提出的網路架構則是為了簡化節點與網路的設計複雜度並可以在有限的波長傳輸通道下用網路連結的方式來擴充增加網路的節點數目。

如圖一所示，我們所探討的網路在網路上的各個節點皆具備有一組可調波長的發射器與接收器用來存取資料通道及另一組固定波長的發射器與接收器用來存取預約控制通道。



圖一 網路上的節點結構

此星狀網路也包含了單一的中央排程控制器，其佇列架構如圖二所示。網路節點利用隨機存取方式傳送預約封包到波長 λ_c 的預約控制通道，來跟中央的排程控制器要求資料通道，當隨機存取成功之後，封包將傳入位於中央的排程控制器裡的佇列。而佇列只能儲存預約封包到一定長度，當對應到某一個節點的佇列裡儲放的封包已經飽和時，此時該節點後續傳送出的預約封包將會被丟棄掉，並由中央控制器通知該節點暫停存取預約控制通道，等到中央控制器裡的佇列有可用的空間才可重新存取預約控制通道。中央控制器對於佇列裡的

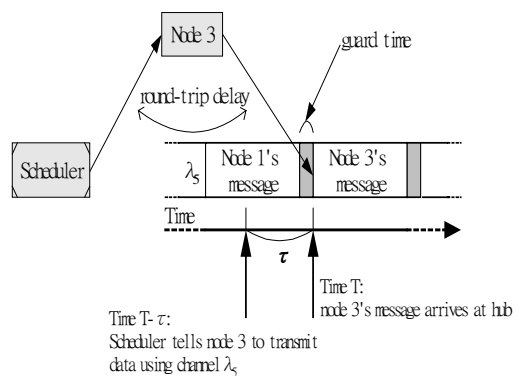


圖二 中央控制器的佇列結構

預約封包，依照所定義的排程演算法來排定哪一些節點可以調整到適當的波長資料通道來傳輸或接收資料封包，最後利用 λ_c' 控制通道來通知網路上的節點調整發射器或接收器的波長，於某個時間點開始傳輸。我們假設網路的存取方式是以非同步方式來存取資料以避免掉設計同步分時分波多工結構的複雜性，所以中央排程控制器必須計算及測量節點傳送時間來通知網路上節點的開始傳輸及接收時間，其方法如下所示。

3.2 測量節點傳送時間

所有節點均送出預約封包給中央排程控制器。因為節點間不需要維持同步，因此中央排程控制器在安排可使用的資料通道給節點後，須通知節點應調整之發射器或接收器的波長，而節點在收到中央排程控制器的回應預約控制封包後隨即調整波長開始傳送或接收資料。節點之間與節點和排程器之間必須克服傳輸延遲的問題，才能保證資料傳送的正確性。



圖三 中央排程器測量傳輸延遲時間之方式

利用測量節點傳送時間 (ranging)[2]的方法可以克服傳輸延遲問題的困擾。先計算每個節點到中央控制器單一回合傳輸過程的延遲時間(包括傳輸延遲與接收預約封包後的處理時間),這裡我們假設在節點上的可調式波長發射器與接收器具有快速調整波長的特性。如圖三, τ 為所量測出來的 Node 3 與中央排程器的傳輸延遲時間,為了讓 3 號節點可以在時間 T 開始傳送資料,中央排程控制器就必須在時間 $T - \tau$ 時送出預約封包給節點 3。利用如此的方式,不同的節點要彼此傳輸之間雖然沒有做同步化的動作,但利用中央的排程控制器量測傳輸延遲時間,可以形同分時多工方式,以時槽結構利用同一個波長通道來傳輸資料,而達到非同步但分時分波多工的目的。

用測量傳輸延遲時間的方式的優點是節點的傳輸設備可以簡化,也毋須節點與中央排程控制器同時維持同步即可有時槽的傳輸結構。

3.3 隨機存取預約控制通道

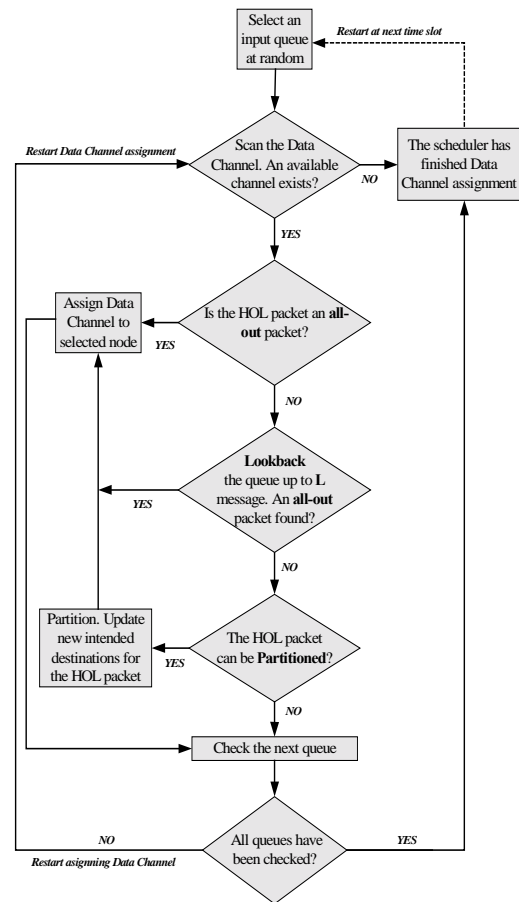
預約控制通道的存取方式是利用隨機存取協定來送出預約封包到波長 λ_c 的預約控制通道,所以有可能會與其他節點所送出的預約封包產生碰撞,一旦碰撞發生則預約封包將不被中央排程控制器所接收,若沒有發生碰撞的情形下,則中央排程控制器將順利接收預約封包置入佇列裡,並經由控制通道 λ_c 回應給節點一個回應預約控制封包,通知在中央控制器裡的佇列已經改變,節點可以送出下一個要求資料通道的預約封包。

3.4 群播排程演算法

3.4.1 回顧佇列存取協定(LBQA)

在群播環境中,可能同時會有數個封包傳送至相同的目的節點因而發生碰撞,為了提高網路的效能,我們必須避開線頭佔線效應以及減少傳輸目的節點衝突,在[1]中提出了回顧

佇列存取協定,在此協定中,作者藉由搜尋數個佇列內的預約封包來避免線頭佔線效應並減少傳輸目的節點衝突。在此,假設以隨機選擇起始佇列的方式來使所有節點均能公平地存取網路,圖四即為回顧佇列存取協定的流程圖。其中 all-out packet 定義成一個群播封包,而此封包的所有節點都能接收此封包,不致產生封包接收器碰撞的情形發生。



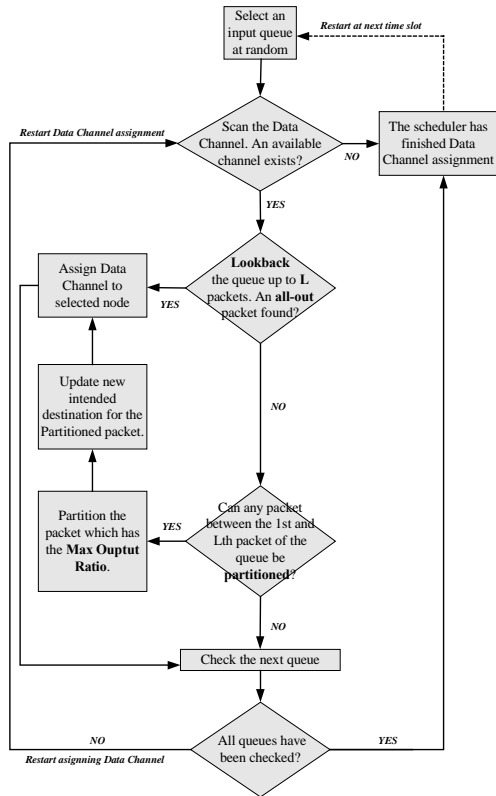
圖四 回顧佇列存取協定流程圖

3.4.2 回顧比例存取協定(LBRA)

在回顧比例存取協定中,我們為了能夠更充分的利用在回顧存取所獲取的資訊,所以當中央排程控制器在某一佇列的前 L 個封包中均搜尋不到 all-out packet 時,不再只會對佇列的線頭封包作切割的動作,而是切割前 L 個封包裡輸出比例最高的封包。切割輸出比例最高的封包除了考慮預約封包內的資訊之外

同時也考慮了接收器的使用情況，選擇切割最高輸出比例之封包，可以使得該封包在下次傳送時有效減少目的節點位址衝突的機會。

回顧比例存取協定流程如圖五所示，首先中央排程控制器隨機選擇一個佇列作為存取協定的起始位置。回顧比例存取協定首先會檢視其佇列的線頭封包內的目的節點位



圖五 回顧比例存取協定流程圖

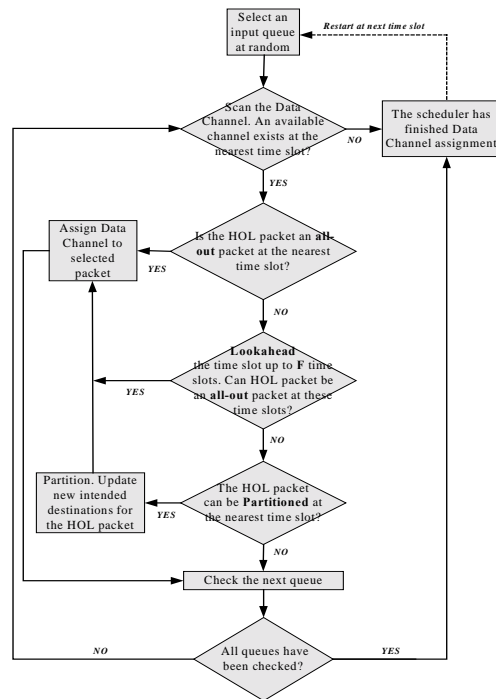
址是否與之前已分配資料通道的佇列其封包的目的節點位址相衝突，若是有發生衝突則再對該佇列後續的 $L-1$ 個封包進行檢查，若檢查的過程中有找到 all-out packet，則分配資料通道給該封包，若是所檢查的 L 個預約封包皆有部分的節點位址發生衝突，則計算該佇列裡前 L 個封包的輸出比例並將資料通道分配給擁有最高輸出比例(Max Output Ratio)的封包，其中輸出比例的定義如(3.1)式所示，如果佇列前 L 個封包內所有目的節點位址皆與之前已分配資料通道的佇列其封包的目的節點相衝突，則中央排程控制器就放棄將資料通道

$$\text{輸出比例} = \frac{\text{可傳送之目的節點位置數}}{\text{應傳送之目的節點位置數}} \quad (3.1)$$

分配給該佇列，轉而將使用資料通道的機會給下一個佇列。在下一個佇列重新執行上述的機制，直到所有未分配的資料通道分配完畢或搜尋完所有的佇列為止。

3.4.3 前視時槽存取協定(LASA)

在前二節中所提出的群播分割存取機制在碰撞發生時皆是藉由回顧佇列內後續的預約封包來避免碰撞的發生。在中央排程控制器中除了記錄了預約封包的資訊外也記錄了每個時槽發射器、接收器與波長通道的使用情況，因此，在接下來的兩節中，我們改變線頭



圖六 前視時槽存取協定流程圖

封包發生碰撞時的處理方式而再提出二個存取機制，分別為，前視時槽存取協定與前視比例存取協定。在上述的兩種存取協定中，當某佇列的線頭封包在最接近的時槽與其它佇列線頭封包的目的節點位址發生衝突時，存取協

定則往前檢視未來的時槽，察看線頭封包在未來時槽是否能以更有效率的方式來傳送以提高網路效能。

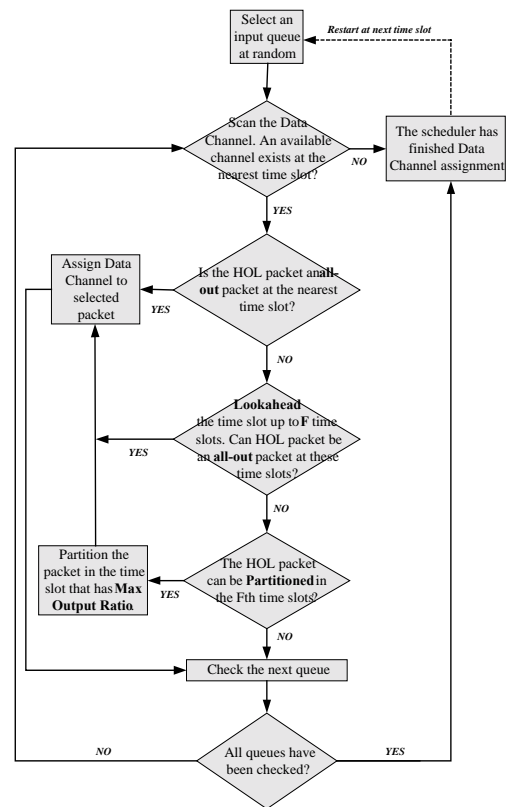
前視時槽存取協定流程如圖六所示，中央排程控制器一開始隨機選取一個佇列當作前視時槽存取協定的起始位置。前視時槽存取協定首先會檢視所選取佇列的線頭封包的目的節點位置是否與之前已分配資料通道的佇列其封包的目的節點位置在最接近的時槽相衝突，若是有發生衝突則檢查該封包在其它時槽是否為一 all-out packet，最多往前檢查到第 F 個未來時槽為止，此即為 Lookahead 的長度 F ，若檢查到線頭封包在其中一個時槽為一 all-out packet，則在該時槽分配一資料通道給該封包，若是檢查的結果是在前面的 F 個時槽裡，線頭封包的目的節點位址皆有部分會發生目的節點衝突的情況時，則視線頭封包的目的節點是不是可以在最接近的時槽用分割的方式來傳送資料，如果線頭封包的所有目的節點位址在最近的時槽都與其他佇列的線頭封包內的目的節位址發生衝突，則中央排程控制器就放棄將資料通道分配到該佇列。在下一個佇列重新執行上述的機制，直到分配完最接近的時槽所有的資料通道或搜尋完所有的佇列為止。

3.3.4 前視比例存取協定(LARA)

與回顧比例存取協定的想法類似，我們為了能夠更充分的利用在前視時槽時所獲取的資訊，所以當中央排程控制器發現某一佇列的線頭封包在搜尋的 F 個時槽內均不能以 all-out packet 形式傳送時，不再只會對該封包在最接近的時槽做切割的動作，而是將該封包在擁有最高輸出比例的時槽以切割的方式來傳送，希望更有效率的利用有限的資料通道。

圖七為前視比例存取協定的流程圖，首先中央排程控制器隨機選擇一個佇列作為存取協定的起始位置。前視比例存取協定與前視

佇取協定類似，前視比例存取協定首先會檢視所選取佇列的線頭封包內的目的節點位址是否與之前已分配資料通道的佇列其封包的目的節點位置相衝突，若是有發生衝突則再往前檢視 F 個時槽，若檢查到線頭封包可以在某個時槽以 all-out packet 形式來傳送，則分配一適當資料通道給該封包，若是所檢查的 F 個時槽內，線頭封包皆有部分的目的節點位址發生衝突，則計算該封包在 F 個時槽中個別的輸出比例，並在有最高輸出比例的時槽分配一資料通道給該封包，如果線頭封包在搜尋的 F 個時槽內所有的目的節點位址皆發生衝突，則中央排程控制器就放棄將資料通道分配給該佇列，轉而將使用資料通道的機會給下一佇列。



圖七 前視比例存取協定流程圖

3.4.5 雙重檢視存取協定(DCHA)

雙重檢視存取協定流程如圖八所示，首先中央排程控制器隨機選擇一個佇列作為存取協定的起始位置。雙重檢視存取協定首先會

檢視所選取佇列的線頭封包內的目的節點位址是否與之前已分配資料通道的佇列其封包的目的節點位址相衝突，假如有衝突發生則將傳送封包的機會讓給下一個佇列。假如在搜尋完所有的佇列之後，系統還有未被分配的資料通道，則轉而從起始佇列開對未被安排傳送的佇列的線頭封包進行檢查，查看是否能以切割的方式先傳送資料至部分目的節點位址，如果佇列的線頭封包無法用分割的方式利用兩個時槽以上的時間來傳送資料，則保留資料通道給其它未被安排傳送的佇列，重新執行此流程，直到所有的波長資料通道都分配完畢或檢查完所有的佇列為止。

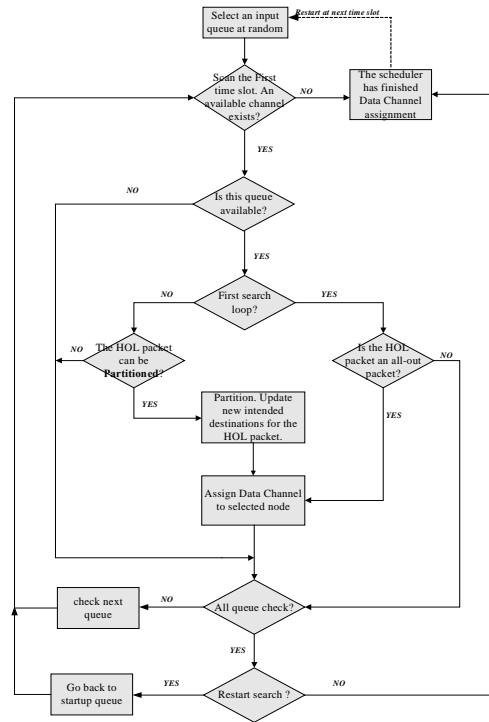
四、單星狀網路的模擬結果

這一節裡我們利用模擬方式來分析比較上述的幾種存取機制，這裡所模擬的網路狀況只考慮中央排程控制器的排程機制，沒有模擬預約控制通道的隨機存取情形，並假設中央排程控制器的佇列都是維持在滿載的狀態。

在開始分析網路效能之前，為了方便說明，我們將模擬時所使用的系統參數整理於表一並定義較簡短的符號來表示這些參數，另外，表二中為交通量模式的定義。我們還定義了網路效能(Efficiency)，如式 4.1 所示，作為

$$Efficiency = \frac{Max\left(\frac{E[K] \times W}{N}, 1\right)}{\text{平均傳送一個封包所需 的資料通道數}} \dots\dots\dots(4.1)$$

判斷存取協定好壞的參考。我們可以注意到，當 $W \times E[K] / N < 1$ 時(其中 $E[K]$ 為群播封包的平均目的節點數)，系統是受限於資料通道的數目與群播封包目的位置的數目乘積，我們稱為 channel limit，也就是波長資料太少，即使能傳送 W 個封包，每個封包平均送到 $E[K]$ 個位址，也不足以讓所有的接收點同時接收資料，



圖八 雙重檢視存取協定流程圖

因此如何善用每個資料通道便成為決定網路效能的重要因素。而當 $W \times E[K] / N \geq 1$ 時，系統則受到接收點數目的限制，我們稱為 receiver limit，代表者網路上有比接收點數目還多的封包等接收點接收，於是接收點的數目便成為決定網路效能的重要條件，因此為了有效的表達網路的效能，我們採用平均傳送一個封包所需資料通道數的理論值除以實際平均傳送一個封包所需的資料通道數來做網路效能的定義。

表一 系統參數與相對應之符號

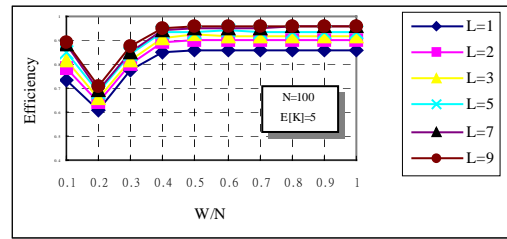
系統參數	說明	代表符號
Lookback length	回顧存取長度	L
Lookahead depth	時槽搜尋深度	F
Node	網路節點數	N
Multicast size	封包群播範圍	K
Wavelength	可用波長數	W
Unicast Ratio	單點傳輸封包與群播傳輸封包的比例	U(U6 表示系統 60%的封包為單點傳輸封包)

首先我們所要觀察的是回顧佇列存取協定在不同系統參數下網路效能變化。圖九表示回顧佇列存取協定在固定節點數，不同回顧長度對網路效能造成之影響。我們可以看出回顧佇列存取協定，其回顧長度跟網路效能提昇都是成正比關係，回顧長度愈長網路效能愈高，當回顧長度由 2 逐漸加大至 5 時，對於網路效能有著較顯著的增加，而之後網路效能的成長就趨向緩和，因此我們不需要太長的回顧長度使用有效的提高網路效能。另外值得注意的是，當 $E[K] \times W/N$ 接近 1 時網路效能會較低，這是因為在我們定義的網路效能中，當系統為 Channel limit 時，於是網路的效能主要是看演

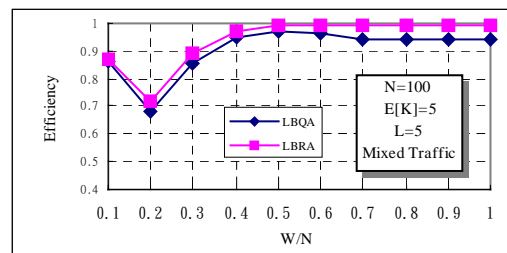
表二 交通量模式之定義

交通量模式	說明
Fixed Multicast Size Traffic	每個封包的群播範圍固定，且所有封包為群播封包。
Mixed Traffic	單點傳輸封包與群播傳輸封包成一定比例的交通量模式，其群播封包的群播範圍在 2 至 20 之間平均分佈。
Correlated Traffic	此交通量模式也是混合著單傳輸封包與群播傳輸封包，但相連封包的目的節點位置可能且有相關性，也就是說，可能會有連續數個封包具有相同的目的節點位置。

算法是否能夠有效的利用每個波長來傳送封包，當 $E[K] \times W/N \ll 1$ 時，回顧佇列存取協定較容易找到完全沒有碰撞發生的封包，使得平均傳送每一個預約封包的波長數減少，因而網路效能較高，當 $E[K] \times W/N$ 漸漸加大往 1 接近時預約封包發生碰撞的機會也慢慢加大，因而使得平均傳送封包的波長數加大，降低了網路效能，另一方面，在 receiver limit 時，我們可以藉由分割的機制來提高接收器的使用率，因此當波長愈多時網路效能會愈高。



圖九 比較回顧佇列存取協定不同的回顧長度在固定節點數目與固定平均群播範圍條件下的波長節點數比值對網路效能變化圖

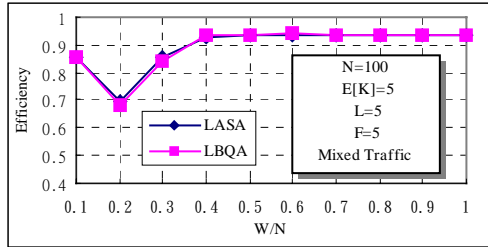


圖十 比較回顧比例存取協定(LBRA)與回顧佇列存取協定(LBQA)在固定回顧長度、固定節點數目及固定平均群播範圍條件下的波長節點數比值對網路效能變化圖

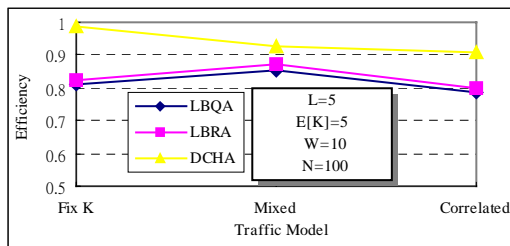
圖十表示回顧比例存取協定與回顧佇列存取協定在固定節點數和回顧長度網路效能的比較。在這裡我們設定回顧長度為 5，這是因為回顧比例存取協定的網路效能與回顧佇列存取協定類似，網路效能及回顧長度成一正向關係，且當回顧長度為 5 時已經顯示不錯的效能。由圖十可以看出回顧比例存取協定的網路效能較回顧佇列存取協定來的優異，這是因為切割擁有最高輸出比例的封包可以有效的減少該封包在下次傳送時發生碰撞的機會且降低資料通道的閒置。

回顧佇列及回顧比例存取協定中，皆是利用中央排程控制器裡預約封包的資訊來盡可能減少碰撞的發生以提高網路的效能。而以下所探討的兩種演算法中，在碰撞發生時我們藉由往前檢視數個時槽來避免碰撞，以期更有效率的利用波長。圖十一是設定回顧佇列存取協

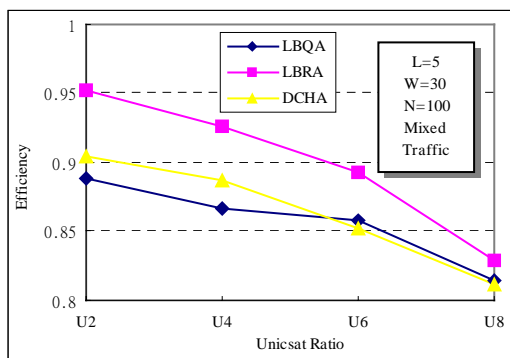
定的回顧長度為與前視時槽存取協定的前視時槽為 5 時，在固定平均群播範圍下此兩種存取協定在效益上的比較圖。當回顧佇列的長度



圖十一 比較前視時槽存取協定與回顧佇列存取協定在搜尋深度為 5 時固定節點數目、平均群播封包範圍下的波長節點數比值對網路效能變化圖



圖十二 比較回顧佇列、回顧比例及雙重檢視存取協定在不同交通量模式下固定節點數、固定資料通道數以及固定平均群播範圍時的網路效能變化圖



圖十三 比較回顧佇列、回顧比例及雙重檢視存取協定在 mixed traffic 下固定節點數以及固定資料通道數下改變平均群播範圍時的網路效能變化圖

等於前視時槽數時，回顧佇列存取協定與前視時槽存取協定的網路效能非常接近，因此在接下來的討論中，我們不討論前視時槽存取協定而是以回顧佇列存取協定來代替。而前視比例與回顧比例在相同搜尋深度下波長節點數效能也是相當近似，因此在之後的討論我們不再特別說明前視比例存取協定。

最後，我們藉由圖十二及圖十三來觀察雙重檢視存取協定的網路效能，並對各種演算法的適用網路環境作歸納。圖十二為波長節點數比值小時，不同演算法在二種交通量模式下網路效能的表現。如圖所示，雙重檢視存取協定的網路效能在三種交通量模式下都比回顧佇列及回顧比例存取協定來的優秀，這是因為雙重檢視存取協定在分配資料通道給 all-out packet 時，其搜尋範圍較另兩種存取協定來的廣，也就較容易找到 all-out packet，使得中央排程控制器能更有效的利用資料通道。在圖十二中，可以看出雙重存取協定在 correlated traffic pattern 中網路效能並沒有受到明顯的影響，這是由於雙重存取協定是透過搜尋不同佇列的線頭封包來避免碰撞的發生，因此可以避免搜尋到相同目的節點的預約封包造成無效的搜尋。圖十三為固定資料通道 $W=30$ 時，不同演算法在 mixed traffic pattern 下改變平均群播範圍時網路效能的表現，由圖中可以看出回顧比例的網路效能較雙重檢視及回顧佇列來的好。因此，為了得到較佳的網路效能，必須在不同的網路環境選擇適當的群播分割機制。

五、雙星狀網路架構與群播存取協定

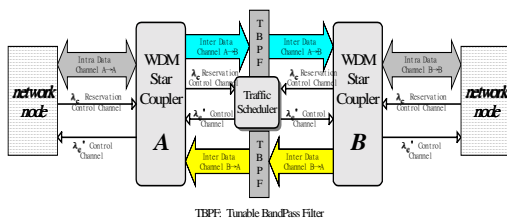
5.1 簡介

由於在非時槽單星狀光學網路下，網路上所有的節點所傳送的資料是利用被動星狀耦合器廣播至網路上其它節點，因此單一星狀光學網路的節點數有其上限。因此，在本節中我們將探討如何將星狀網路相互連結以增加網路上的節點總數，並且提出中央排程控制器的

演算法以提高互連星狀光學網路的網路效能，並與單一星狀光學網路下的網路效能做比較。

5.1 雙星狀網路架構

在雙星狀光學網路方面我們利用可調式光波濾波器來連接兩個子星狀網路。以圖十四為例，子星狀網路 A 可以透過 *Inter Data Channel A→B* 傳送資料封包到子星狀網路 B，同理子星狀網路 B 則可透過 *Inter Data Channel B→A* 傳送資料封包到子星狀網路 A，此外子星狀網路 A 與 B 皆可透過 *Intra Data Channel* 來傳送資料封包給位於本身子網路的節點，並且經由可調式的光波濾波器將彼此的 *Intra Data Channel* 加以阻隔，使得具群播性質的星狀光學網路所傳遞的相同波長的信號不會互相干擾。而當 *Inter Data Channel* 有閒置不用時，中央排程控制器便調整光波濾波器的範圍，並將該閒置波長歸到 *Intra Data Channel*，分別提供給兩個子星狀網路的內部節點來傳輸資料。在預約控制通道方面，兩個子星狀網路皆利用相同波長的預約控制通道 λ_c 作為預約封包的隨機存取通道，並連接到中央排程控制器。由於兩個子星狀網路的預約控制通道 λ_c 是不相通的，所以節點送出的預約封包發生碰撞的機率就大幅降低，便使網路總節點數目的上限就相對提高，同理，利用可調式的光波濾波器將子星狀網路各自的 λ_c' 控制通道加以阻隔，因此中央排程控制器同一時間可以送出兩個回應預約控制封包到不同的子星狀網路上，增加送出回應預約控制封包的速度。



圖十四 集中式排程控制的雙星狀光學網路

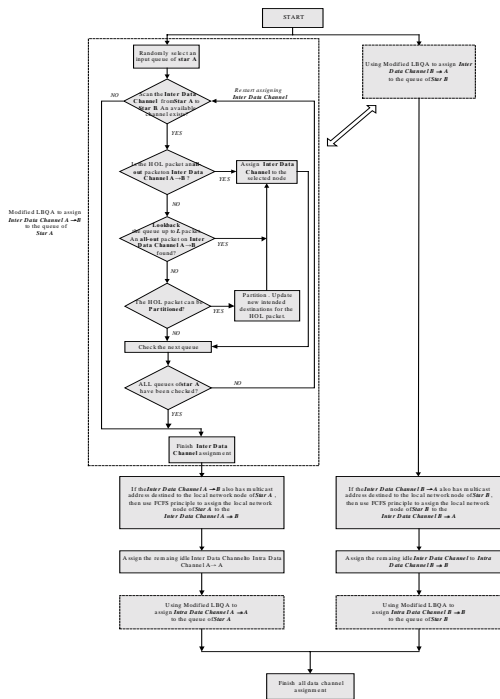
架構圖

在設計適用於雙星狀光學網路架構的群播機制時，我們先定義不同資料通道之優先分配順序。我們採用了 *Inter Data Channel* 的優先權高於 *Intra Data Channel* 的原則。

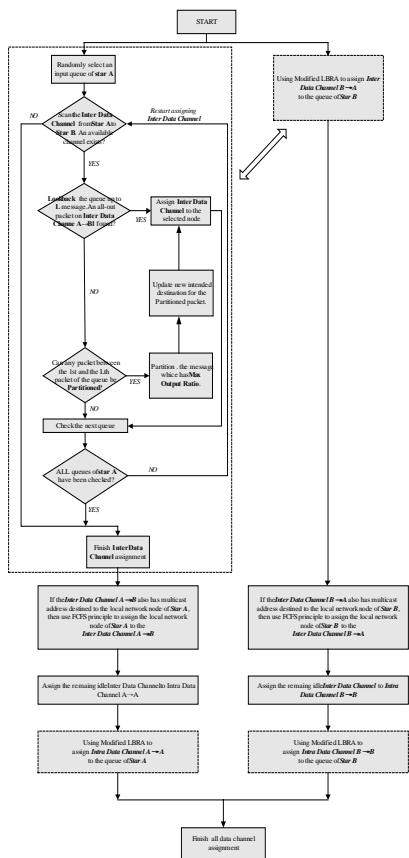
在雙星狀網路時，我們讓存取協定只單獨比較預約封包內往另一個子星狀網路的節點位址是否有發生衝突，而群播範圍中往自己本身子星狀網路的節點位址部分則忽略不予以比較。而在分配 *Intra Data Channel* 方面，同樣讓回顧存取協定只單獨比較預約封包內往自己本身子星狀網路的節點位址是否有發生衝突，至於群播範圍中往另一個子星狀網路的節點位址部分則忽略不予以比較，如此，可充分利用 *Intra Data Channel* 的使用效率。

5.2 雙星狀網路存取協定

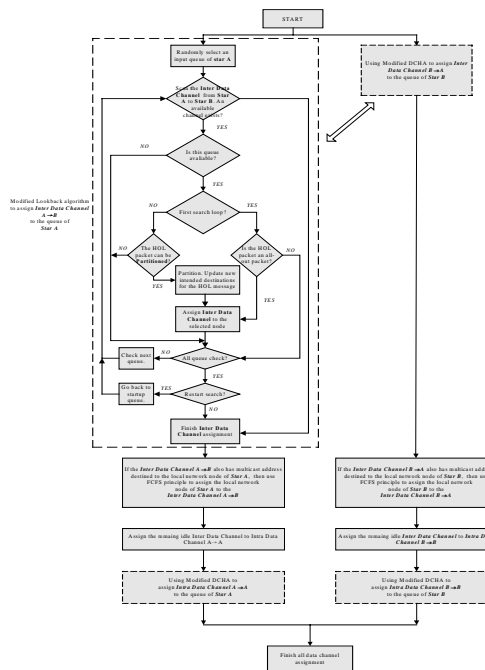
圖十五至圖十七分別為雙星狀回顧佇列存取協定的流程圖、雙星狀回顧比例存取協定流程圖與雙星狀雙重檢視存取協定流程圖。我們以雙星狀回顧佇列存取協定來說明如何將單星狀網路的存取協定加以修改使其適用於雙星狀網路。以圖十五的子星狀網路 A 為例，中央排程控制器會先檢查是否有空的 *Inter Data Channel A→B* 的波長資料通道，然後在對應到子星狀網路 A 的佇列中隨機選取一個其預約封包含有到子星狀網路 B 目的節點位址的佇列，此時因為是分配 *Inter Data Channel*，所以就檢查預約封包內的 *Inter* 目的節點位置是否與之前已分配好的傳輸節點發生 *Inter* 目的節點位置衝突，若是發生衝突則再對該佇列裡後續的封包做檢查，最多往後檢查到第 L 個封包為止。若在檢查的過程中，發現到某一預約封包內的 *Inter* 目的節點位址都沒有發生衝突，就可以將 *Inter Data Channel A→B* 分配給該預約封包。若是所有 L 個預約封包都有部分的 *Inter* 位址發生衝突，則中央排



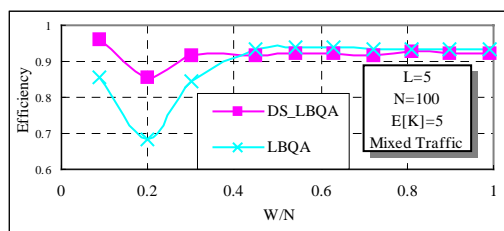
圖十五 雙星狀回顧佇列存取協定流程圖



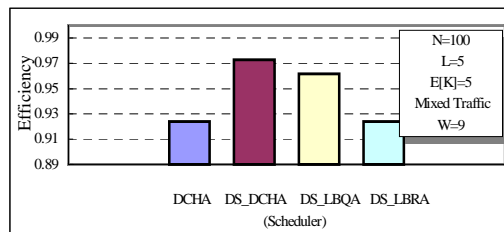
圖十六 雙星狀回顧比例存取協定流程圖



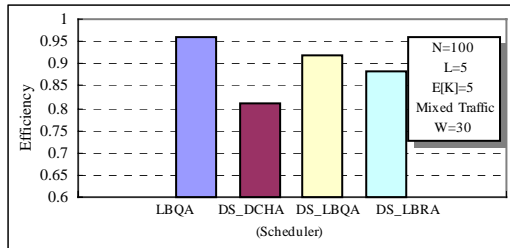
圖十七 雙星狀雙重檢視存取協定流程圖



圖十八 雙星狀光學網路與單星狀光學網路分別使用 DS_LBQA 及 LBQA 下，在固定節點數、固定平均群播範圍、固定回顧長度以及交通量模式為 mixed traffic 時的波長節點數比對網路效能變化圖



圖十九 DS_DCHA、DS_LBQA、DS_LBRA 及 DCHA，在固定節點數、固定平均群播範圍、固定回顧長度、固定波長數以及交通量模式為 mixed traffic 時的網路效能比較圖



圖二十 DS_DCHA、DS_LBQA、DS_LBRA 及 LBQA，在固定節點數、固定平均群播範圍、固定回顧長度、固定波長數以及交通量模式為 mixed traffic 時的網路效能比較圖

程器會檢查位於佇列中最前面的預約封包是否可以分割目的節點位址，如果可以分割預約封包，就將此 *Inter Data Channel* 分配給該預約封包，但若是無法將此預約封包分割則由下一個佇列，重新執行此流程來判斷是否可以分配 *Inter Data Channel*，直到所有 *Inter Data Channel* 分配完畢或佇列中沒有可傳輸的 *Inter* 封包為止。

5.3 模擬結果

由模擬結果得知，修改後的存取協定在雙星狀網路有著相當優異的效能。圖十八中為雙星狀回顧佇列存取協定與單星狀回顧佇列存取協定的網路效能比較圖，我們可以看到，在資料通道有限時，雙星狀網路的網路效能遠在單星狀網路來的好，這是因為我們有效的利用雙星狀網路資料通道重覆時用的特性使得可用的資料通道增加因而提高了網路效能，此一特性在圖十九中更可明白的表示出來。不過當資料通道達一定數量時，如圖二十所示，網路效能受實際可用接收器的限制，由於單一星狀網路對封包內的節點的檢查範圍比雙星狀光學網路大，因為雙星狀網路每次檢查預約封包只檢查一半的目的節點位址，因此使得單星狀網路效能較優異。綜合圖十九及圖二十，可以看到，為了得到較佳的網路效能，必須在不同的網路環境選擇適當的群播分割機制。

六、結論

在本文中，我們考慮星狀網路具有非同步時槽結構之特性，由中央排程控制器負責處理佇列裡的預約封包。透過模擬的方式，我們分析了所提出的存取協定，結果顯示各種存取協定各有其適用的網路環境，因此為維持網路在最佳的效能，我們必須依不同形態的網路來選擇最合適的存取協定。進一步的當網路上的節點總數變多時，隨著星狀網路的擴充，我們提出了改良後的存取協定，以適應多星狀群播網路的特性，而模擬的結果也顯示修改過後的存取協定的確能夠適用在雙星狀網路並發揮出空間再利用的特性，使網路能維持相當高的網路效能。

[1]黃舜雍，”多重星狀光學網路下群播協定之設計”，台北科技大學電腦通訊與控制研究所，2000

[2] M. S. Borella, and B. Mukherjee, “A reservation-based multicasting protocol for WDM local lightwave networks,” ICC’95, Vol. 2, pp. 1277-1281, 1995.

[3] N. F. Huang, and H. I. Liu, “An isochronous and asynchronous traffic scheduling algorithm for dual-star WDM Networks,” Journal of Lightwave Technology, Vol. 14, No. 3, March 1996.

[4] J. P. Jue and B. Mukherjee, “The advantages of partitioning multicast transmissions in a single-hop optical WDM network,” Proc. IEEE ICC’97, 1997, pp 427-431.

[5] E. Modiano, “Random algorithms for scheduling multicast traffic in WDM broadcast-and select networks,” IEEE/ACM transactions on Networking, Vol:73, pp. 425-434, 1999.