

符合 Max-Min fairness 的流量控制演算法 A Flow Control Scheme on ATM Networks with Max-Min fairness

賴威光

Wei Kuang Lai

國立中山大學資訊工程研究所
Institute of Computer and Information
Engineering Nation Sun Yat-Sen University
Kaohsiung, Taiwan, R.O.C
wkilai@cie.nsysu.edu.tw

蔡駿逸

Jiun-Yih Tsai

國立中山大學資訊工程研究所
Institute of Computer and Information
Engineering Nation Sun Yat-Sen University
Kaohsiung, Taiwan, R.O.C

摘要

ERAMM (Explicit Rate calculating Algorithm with Max-Min fairness) 乃是針對快速收斂性及低振盪性目標所設計的 Rate-based flow¹ control algorithm, 可以迅速進入收斂, 低振盪, 使每段 link bandwidth 均得到充份的利用, 降低 buffer 的變動。

關鍵字: Flow Control, Rate-based

Abstract

The ERAMM has the advantages of fast convergence, low oscillations, high link bandwidth utilization, and fewer buffer requirements

Keyword: Flow Control, Rate-based

1. 序論

在 ATM 網路上的重要特色之一, 是藉由提供各種不同型態 QoS[1] 等級的資訊服務以符合適用於不同應用系統的傳輸延遲需求及丟棄優先權等要求。在 ATM 網路所提供的諸多服務型態中, 以可利用位元速率(Available Bit Rate; ABR) 服務[2], 由於具有傳輸前不需和系統建立流量契約(traffic contract), 及可為其動態調整適當發送速率及配置剩餘頻寬等特性, 因此適合用於資料檔案的傳輸, 也最需要流量控制。

本 proposal 所提出的 ABR 流量控制演算法可

以有效解決頻寬的分配公平性(fair bandwidth allocation), 降低封包流失的數量(low cell loss), 簡化 switch 的設計, 並且可以快速收斂(fast convergence)到穩定傳輸狀態的能力。

2. 設計考量

我們對於一個 rate-based flow control 的演算法可以用下列幾個評估觀點來作比較 [4][5][8][9]:

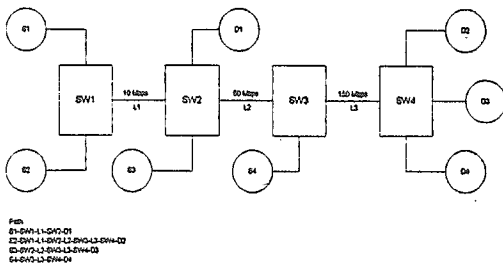
2.1 公平性(Fairness)

所謂公平性指的是如何為諸多不同的 VC 配置適當的傳輸速率, 使其得以滿足某一公平法則, 在這裡, 我們所採用的公平法則(fairness criterion)是『最大-最小公平性』(Max-Min fairness)[3][10], 所謂 Max-Min fairness 是指若有一段 link 對於所經過它的一些 VC 造成 bottleneck(一條 VC 所經過最小頻寬的 link, 謂之該 VC 的 bottleneck link)[6], 那麼此段 link 上的頻寬必需平均地配置給這些 VC, 而由於受到此 bottleneck link 的影響, 這些 VC 在它們所經過的其它 link 上, 最多也只能使用同樣大小頻寬, 使得整條 VC 在每一段 link 上可以均等的速率傳送資料。關於 Max-Min fairness 我們可以圖(一)為實例來說明。

可依下列步驟運算:

- (1) 先將每一段 link 的 bandwidth 除以經過它的 VC 個數, 使得每一段 link 均得到一個平均值。
- (2) 找出這些平均值當中的最小值(以上例而言為 5), 將所有經過這段 link 的 VC 的 allocated rate 設為此最小值。
- (3) 將這些 VC 所經過的其它 link 均減去此 allocated rate, 並將已確定 allocated rate 的 VC 減去不計後, 將剩餘頻寬平均分配給在此 link 上的其它 VC, 反覆(1)-(3)的步驟, 直到每段 link 的頻寬已經完成配置。

¹ This paper was supported in part by the National Council of the Republic of China under contract NSC86-2213-E-110-002.



圖(一)網路實例

計算過程如下：

Iteration	S1	S2	S3	S4
1	5	5	25	50
2	5	5	45	72.5
3	5	5	45	100

所以最後 vc1、vc2、vc3、vc4 的分配值分別為 (5,5,45,100)，亦即滿足 Max-Min fairness 的頻寬配置法。

2.2 收斂性(Convergence)

基於公平性，我們希望每一個 rate-based flow control 演算法均可以走到“趨近”或甚至完全符合 Max-Min fairness 的最終穩定狀態，若演算法的設計欠佳，缺乏公平分配頻寬，調整發送速率的機制，可能收斂到一個不公平的狀態，也可能無法收斂至穩定狀態，導致 ACR 不斷地大起大落，使 SES 無法穩健的速率發送資料，cell loss 嚴重，也影響產量，例如 EFCI marking scheme。也有可能因為缺乏平均速率計算的過程，而發生擊倒(beat-down)問題，造成頻寬分配不均，而無法趨近公平的結果。如 PRCA scheme。反之若設計得當，將可使整個系統得以迅速收斂(Fast convergence)，進入穩定狀態，即使網路 configuration 有所更動，也可以立即在短時間內重新配置頻寬，而調整至一個新的公平穩定狀態。

2.3 振盪性(oscillations)

雖然我們可以盡可能以平均速率的計算來趨近收斂狀態，但如果在接近收斂狀態下，一些發送端的發送速率卻一直在滿足 Max-Min fairness 所配置的頻寬上下巨幅地振盪，以估計(Approximation) 參數調整平均速率以求致收斂狀態的演算法如 EPRCA，在一些 case 中便會發生這樣的問題，這樣的情況並不能算是一個好的演算法，因為這樣使得傳輸速率不能穩定，勢必使得 link utilization 受

到影響，同樣也會導致低產量以及 buffer overflow，致 cell loss 嚴重的問題。

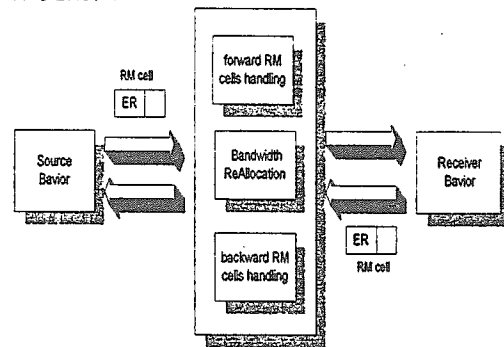
2.4 參數敏感性(Parameter sensitivity)

另外，一個良好的 rate-based flow control 演算法也必需考慮到在調整演算法中所使用的一些參數時，例如調升/調降參數，平均參數等，應不致對整體系統有太大的影響，也就是當某些參數一旦更動的話，若造成其 performance 相差太多，則我們稱此流量控制演算法為具有參數敏感性。特別是當網路的 configuration 有變動時，是否這些參數也不致需要太大的變動，亦是一個重要的設計考量。

3. ERAMM

3.1 設計構想

從滿足 Max-Min fairness 的思考點出發，我們可以設計一個藉由精確速率計算(Explicit Rate Calculating)來作流量控制的演算法。圖(二)是發送端，交換機與接收端所構成的系統方塊圖。由圖(二)的構想來看，我們可以利用 RM cell 上的 ER 欄位，在 forward path 上攜帶要求分配頻寬值，而在 backward path 上則是攜帶已經被計算過的分配值用以通知發送端。此外，我們在每一個 switch 對於經過它的 vc 作狀態維持(state-maintaining)，以利作頻寬配置。



圖(二)系統構想方塊圖

3.2 ERAMM 變數制訂

以下是演算法必要的一些變數制訂與初始狀況：

● Variables

- 1.B Bandwidth of the output link
- 2.Nvc Number of total VCs maintained by the switch
- 3.Nus Number of Unsatisfied VCs
- 4.state[i] State of VC i
- 5.Beq equal share of the link bandwidth
- 6.Req [i] Request temp variable for VC i

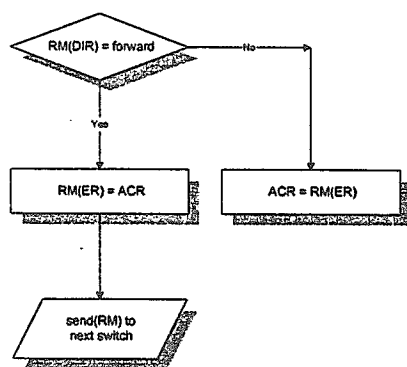
- 7. RM(ER) Explicit Rate field in RM cell format
- 8. RM(DIR) Direction Bit field in RM cell format
- 9. Alloc[i] Current bandwidth allocation for VC i
- 10. Bav Free bandwidth available in the output link

Initial condition

- Beq=B/Nvc;
- Nus=0;
- state[i]=satisfied; for all VC i

3.3 發送端演算法

在圖(三)的發送端演算法中，發送端為產生正向(forward)RM cell的單位，在發送 forward RM cell之前只要將目前的可允許速率(Allowed Cell Rate; ACR)的值置於其 ER 欄位中即可。而在接收到回向(backward)RM cell之後，便將其 ER 欄位讀出，並當作下一個時刻發送一般資料封包的 ACR。



圖(三)發送端演算法流程圖

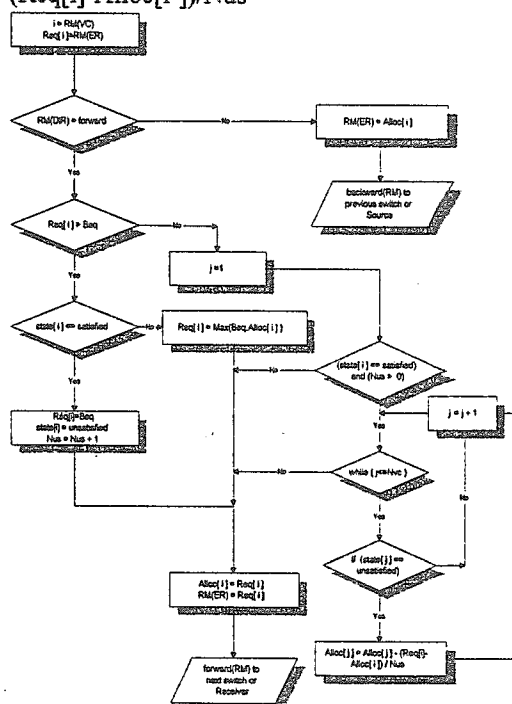
3.4 交換機演算法

交換機演算法的分作 RM cells handling, 圖(四)與 Bandwidth Reallocation 圖(五)兩部份。RM cells handling 主要是針對所經過的 forward 與 backward RM cells 的資料讀出，運算，寫入等處理，而 Bandwidth Reallocation 則是對一條 link 的頻寬依據其上各 vc 的狀態來重新整理、配置頻寬。

在 RM cells handling 中，每一個 switch 均可知道它所 maintain 的 output link 與其上的 vc 個數，所以我們採用在每個 switch 上對每一條 vc 作 state-maintaining。當 forward RM cells 抵達時，將其 ER 的值讀出並置於 Req[i] 的需求陣列中，若此要求小於或等於頻寬之平均值(Beq)，則可以答應其要求，將 Alloc[i] 設為 Req[i]，反之，若超過 Beq，再判斷此 vc 的原來狀態是否為 satisfied，若是，則將此 vc 的 state[i] 設為 unsatisfied，然後暫時僅能分配最大為 Beq 頻寬給此 vc，並將不滿足的 vc 計數器 Nus 加一。若原來已為 unsatisfied，則分配給它 Beq 與

前一次分配的 Alloc[i] 兩者的最大值。最後將 Req[i] 記錄回 RM cell 的 ER 欄位，然後繼續把它送給下一個 switch 或是接收端。

另外有一種情形是出現在當某條 vc 的 Req[i] 大於前一次的配置 Alloc[i-1] 但仍小於 Beq 時，若判斷此 vc 是否仍處於“滿足”(satisfied)狀態，則表示它應該還可以再獲得(Req[i]-Alloc[i-1])，而這部份的頻寬應該由其它那些“不滿足”(unsatisfied)的 vc 來平均釋出，所以每個 unsatisfied vc 必須再釋出 (Req[i]-Alloc[i-1])/Nus



圖(四)交換機 RM cells handling 演算法流程圖

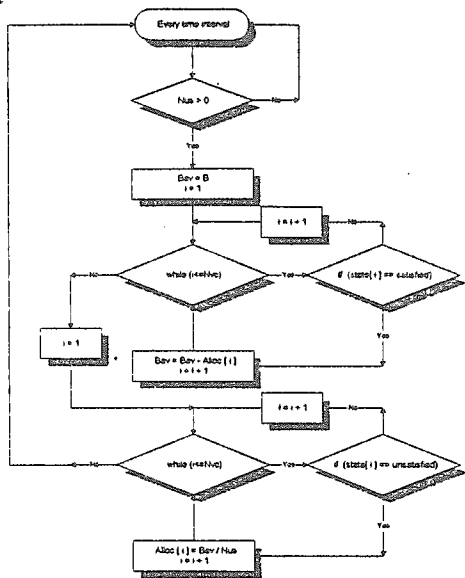
而在圖(五)Bandwidth Reallocation 方面所執行的工作，是在每一段時間間隔後，對於那些狀態為 unsatisfied 的 vc 作一次頻寬重新調整分配。其方法是將某 link 的總頻寬減去所有已配置給 satisfied vc 的 Alloc[i]，得到一可用頻寬值 Bav，再將此值平均分配給 unsatisfied 的 vc。

所以，我們可以用如下的公式來描述在圖(五)中所執行的頻寬分配：

$$Alloc[j](t) = \frac{B - \sum_{i \in Sa(t)} Alloc[i](t-)}{Nus(t)}$$

其中，t 與 t- 各代表此次與上次執行 Bandwidth Reallocation 的時刻，B 代表 link 總頻寬，Us(t) 代表 unsatisfied vc 對 t 的集合，Sa(t) 是代表 satisfied vc 對 t 的集合，而 Nus(t) 是代表在時間 t 的 Unsatisfied

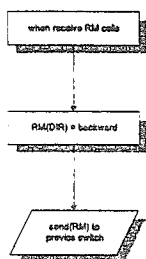
vc 個數。



圖(五)交換機 Bandwidth Reallocation 流程圖

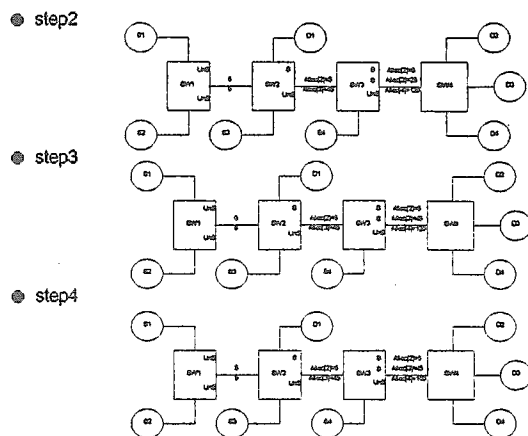
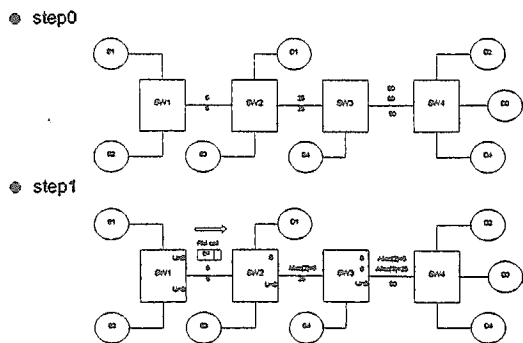
3.5 接收端演算法

圖(六)為接收端的演算法，接收端的行為只要將 forward RM cell 中的方向位元(DIR)設為 backward，也就是將此 RM cell return 回上一個 switch 即可。



圖(六)接收端演算法流程圖

以下我們將 ERAMM 流量控制演算法應用於圖(一)的範例，下圖(七)為其中一種可能的執行步驟。



圖(七)演算法執行步驟範例圖

起始時，每段 link bandwidth 均先計算出其平均值，每條 vc 發送 RM cell 經過 Switch 時，由 Switch 根據 RM cells handling 演算法判斷應該配置多少頻寬給該 vc，例如在 step1 裡，當 vc1 的 RM cell 中的 ER 值小於等於某個經過的 Switch 的 Beq 時，則將該 Switch state[vc1] 設為 "satisfied"，反之則設為 "unsatisfied"，並根據圖(四)的演算法配置頻寬，當 RM cells 抵達 Dnode 時再折返回 Snode，取出 RM cell 中的 ER 值作為更新 ACR 的依據。Switch 亦在一段時間內反復進行 Bandwidth Reallocation，以輔助調整 link 的頻寬分配，其它的 step 亦同 step1，整個過程不斷反復執行圖(四)與圖(五)的演算法使得整體網路的頻寬分配能滿足 Max-Min fairness。

4 模擬結果與分析

4.1 模擬環境

我們採用圖(一)作為模擬演算法的網路系統架構，在這個系統上，分別以 EPRCA 流量控制演算法，和本論文所提出的精確速率計算演算法 ERAMM 來作流量控制，並比較優缺。在圖(五)中，共有四條 ABR 的 VC(以 VC1、VC2、VC3、VC4 為代表，分別由發送端(Snode)，S1、S2、S3、S4 欲將資料封包送達(Dnode)，D1、D2、D3、D4。此外系統共有四個 Switch，其中 VC1 經過 SW1、SW2。VC2 經過 SW1、SW2、SW3、SW4。VC3 經過 SW2、SW3、SW4。VC4 經過 SW3、SW4。以下是一些基本設定[7]：

每一個 Switch 的 buffer 均設為 1000 個 cell 的容量。每一個 Snode 與 Switch，Dnode 與 Switch 均是以 150Mbps 的專線相連。SW1 與 SW2 之間的 link 頻寬為 10Mbps，SW2 與 SW3 之間的 link 頻寬為

50Mbps，SW3 與 SW4 之間的 link 頻寬為 150Mbps。Switch 之間的距離所造成之 Link Propagation Delay，Switch 的 Fabric Delay 假設可以省略不計。Snode 的 Traffic Pattern 是由亂數產生，在任意時刻，Snode 的 output buffer 幾乎都處於滿載，屬於 always-busy 的狀態。

4.2 模擬結果與觀察

在 EPRCA 演算法是採用估計平均式的速率控制演算法，我們將幾個重要的參數設定如下：
在 Snode 共同部份：

PCR (Peak Cell Rate) : 150Mbps

MCR(Mean Cell Rate) : 3Mbps

AIR(Additive Increase Rate) : 10Mbps

在 Snode 個別部份：

ICR(Initial Cell Rate)

	S1	S2	S3	S4
ICR	3Mbps	3Mbps	5Mbps	5Mbps

在 Switch 共同部份：

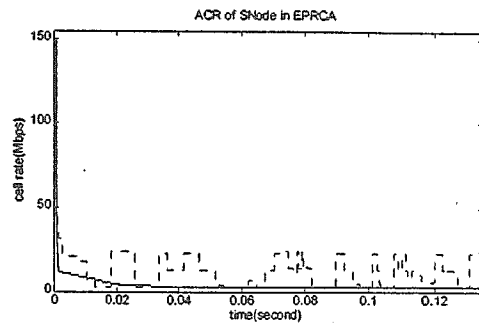
1. AVF (Average Factor) : 0.0625
2. DPF (Down Pressure Factor) : 0.875
3. Nrm : 32

在 Switch 個別部份：

	SW1	SW2	SW3	SW4
(QT、DQT)	(200、400)	(500、700)	(400、700)	(500、700)
ERF	0.725	0.825	0.825	0.825
MRF	0.325	0.625	0.625	0.625

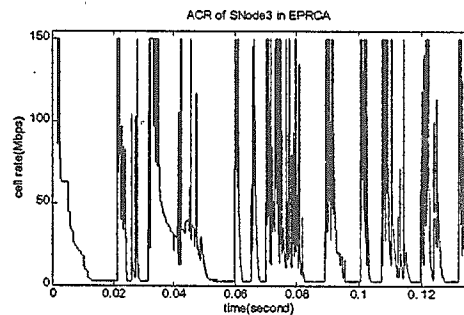
4.2.1 constant ready source traffic type

我們先考慮以 constant ready source traffic type 來作模擬，圖(八)、圖(九)、圖(十)是 EPRCA 在各個 Snode 的 ACR(Allowed Cell Rate)狀況模擬結果，由於撰寫的模擬程式採相對時間設計，在 150Mbps 的 link 上，6 個單位迴圈時間(loop time)可發送一個 cell，經換算一個 loop time 約等於 $4.4928e-7$ 秒，乘上經過 299700 個 loop time，所以全程模擬約進行了實際時間的 0.134651184082 秒。

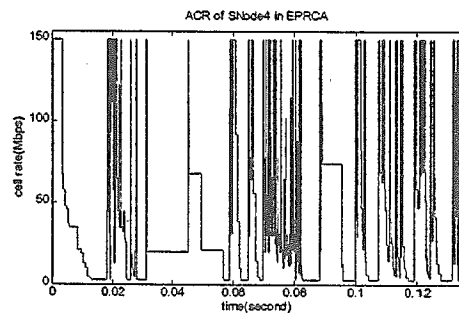


Snode 1 : _____
Snode 2 : _____

圖(八) ACR of Snode1 and Snode 2 in EPRCA

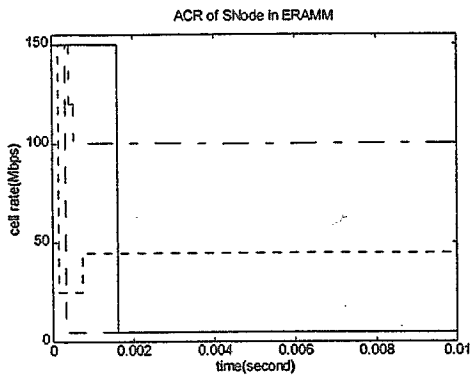


圖(九) ACR of Snode3 in EPRCA



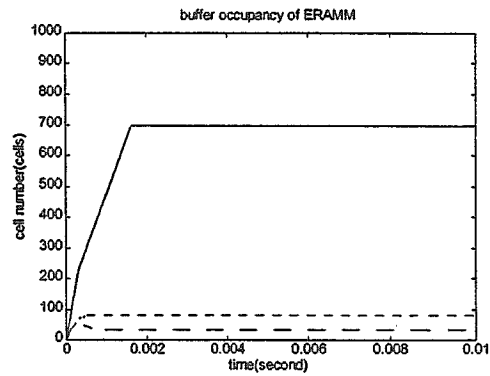
圖(十) ACR of Snode4 in EPRCA

我們可以立即看出各個 Snode 的可發送速率 (ACR) 呈現大幅劇烈的不斷變化，尤其圖(九)、圖(十)中，Snode3 與 Snode4 的 ACR 變化最明顯，不斷在 MCR 與 PCR 之間劇烈調整，並且直到模擬時間結束也沒有收斂的跡象。而 Snode 1 則幾乎經常只處於以 MCR 為 ACR 的最差狀況。圖(十一)為 ERAMM 在各個 Snode 的 ACR(Allowed Cell Rate)狀況模擬結果，(由於在約 0.0018 秒以後均呈收斂狀態，所以為利於觀察收斂情形，圖十一、十三均只列出 0 至 0.01 秒的模擬結果)。



Snode 1 : _____
 Snode 2 : _____
 Snode 3 : _____
 Snode 4 : _____

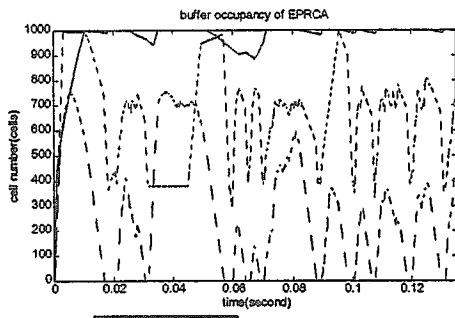
圖(十一) ACR for each Snode in ERAMM



SW1 : _____
 SW2 : _____
 SW3 : _____

圖(十三) buffer occupancy for each Switch in ERAMM

觀察發現，各條 VC 在經過起初短暫的一段時間(約為 0.0018 秒)RM cells 的調整之後，立即迅速達到收斂的狀態(Snode1 與 Snode2 重疊)，並精準地收斂至完全符合 Max-Min fairness 的公平標準。並以穩定的速率發送封包。圖(十二)是 EPRCA 在各級 Switch 造成的 buffer occupancy 狀況之模擬結果，我們可以發現 EPRCA 由於是利用平均速率的估計方法來作 ACR 的計算，因此除了 SW4 因為其連接三個 Dnode 的 link 均為獨立的 150Mbps，可以快速將 cell output，使 buffer occupancy 大多時間為零之外，其它各級 Switch 的 buffer 均呈現巨幅的震盪(oscillation)

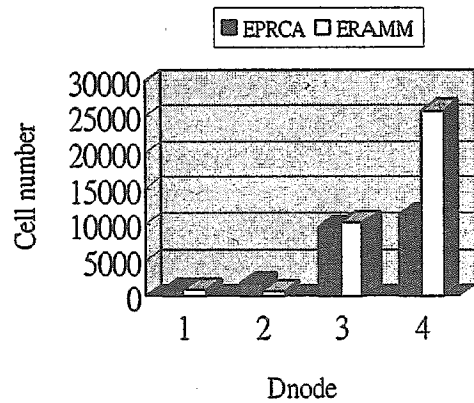


SW1 : _____
 SW2 : _____
 SW3 : _____

圖(十二) buffer occupancy for each Switch in EPRCA

圖(十三)是 ERAMM 之各級 Switch 的 buffer occupancy :

觀察發現，由於各個 VC Snode 的發送速率與各個 Switch 的 output link 頻寬迅速完成 negotiation，達到 Max-Min fairness 的滿足，因此各個 Switch 的 buffer occupancy 立即幾乎不再變化 (Input Rate = Output Rate)。圖(十四)是 EPRCA 與 ERAMM 各個 Dnode 在模擬時間內所收到的封包產量：



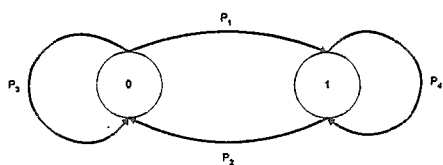
圖(十四) EPRCA 與 ERAMM 之產量比較圖

我們可以發現，在 Dnode1、Dnode2、Dnode3 收到的封包產量比較接近，但是在 Dnode4 卻有極為明顯的差異，若分別計算 EPRCA 與 ERAMM 四條 VC 的平均產量，可以得到 EPRCA 為 5909 個 cells，而 ERAMM 有 9450 個 cells，約為 EPRCA 的 1.5 倍，可以得知當各個 Snode 均處於 heavy traffic 的狀態下時，ERAMM 比 EPRCA 易獲得較高產量。此外在 cell loss 方面，EPRCA 在各級 Switch 的 cell loss 各別為 1951、43、121、0，而在 ERAMM 方面，各級 Switch 的 cell loss 均為 0。這是因為在一開始迅速達到 Max-Min fairness 之後，由於 buffer 不再振盪趨於穩定，所以除非 buffer 容量很小，才

有可能在一開始的調整階段有 cell loss 的可能，但即使在調整階段有少許 cell loss 產生，只要一旦趨於穩定收斂狀態，就不會再有 cell loss 發生。

4.2.2 Interrupted Bernoulli Process source traffic type

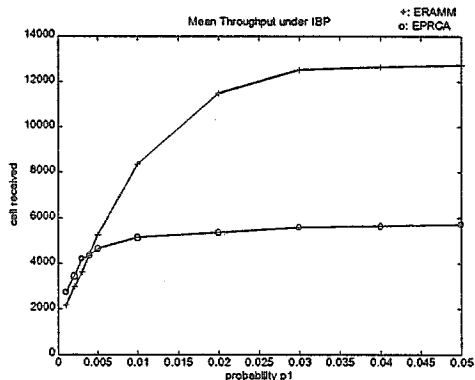
另外我們也可以使用 Interrupted Bernoulli Process (IBP)來作 source traffic type，這是一種經常用於模擬 multi-media 的資料流狀態。IBP state diagram 如下圖(十五)：



圖(十五) IBP State Diagram

在 0 的狀態表示 silent state，而在 1 的狀態則表示 burst state， p_1 、 p_2 、 p_3 、 p_4 則分別代表在 0 與 1 狀態之間轉換的機率，其中 $p_1+p_3=1$ ， $p_2+p_4=1$ 。為方便起見，我們定 p_1 與 p_2 的值時令 $p_1+p_2=1$ 。

模擬結果如下圖(十六)：



圖(十六) Mean Throughput under IBP

我們可以發現，當處於 burst state 的機率越高，ERAMM 比 EPRCA 可以獲得更高的平均產量，而只有在處於 silent state 的機率越高時(p_1 約介於 0 與 0.005 之間)，EPRCA 才能與 ERAMM 有較接近的平均產量或超越之。

4.3 分析比較

根據在第四章所提到的幾項評估觀點，我們可以對 EPRCA 和 ERAMM 作一分析比較：

4.3.1 參數敏感性

由於 EPRCA 是利用 Switch 的 MACR 來作經過它 vc 的平均速率計算，並且根據 DPF、QT、DQT 的大小來判斷目前 buffer 的擁塞狀況，然後再根據 AIR 調升速率，或用 ERF、MRF 來調降 MACR。由於 EPRCA 並沒有對於可用頻寬作分析調整，而且在這麼多參數的設定中，最令人困擾的是並沒有一個可依尋的設定參數的標準或依據，只能藉由大致的人為觀察網路型態及既有的一些經驗來設定。例如，若把 DPF、QT、DQT、ERF、MRF 設得太高，則很可能立刻就發現 cell loss 很嚴重，反之，若調得太低，又使得 Snode 發送速率降得過低，頻寬利用率不佳，產量過低，不能令人滿意。此外，不同的 Switch 的參數設定也不一定相同，往往一個 switch 的參數若改變亦將影響其它 Switch 的運作，所以，Switch 彼此之間的參數交互影響關係密切，具有高度的參數敏感性，也就是想獲取一組能接近既不要 cell loss 太多，又能兼顧使用率與產量的參數設定，並不是一件容易的事。所以這樣的演算法在使用上相當不便。而 ERAMM 因為分別對可用的 link 頻寬作計算調整，因此各 vc 可以獲得一個既穩定而公平的傳輸速率，並不須要調整太多複雜的參數。

4.3.2 收斂性、振盪性、與公平性

在各 ACR 的表現行為方面，EPRCA 的估計式計算，由於缺乏對與 link 頻寬分析，而只針對目前 switch buffer 的擁塞狀況、MACR 作調整 ACR 的依據，很可能使得發送端的 ACR 一直處於極不穩定的大幅變化之中，而難以獲得收斂，或是收斂至不公平的狀態，這跟我們所提的參數敏感性有極密切的關係。當某條 vc 的 ACR 得以提高的時候，可能某個 switch 的 buffer 的空間其實正以極快的速度被別的 vc 所傳進來的 cell 所塞滿，等到 cell loss 發生時，又發現可能為避免擁塞而將 ACR 降到太低，甚至是 MCR 的地步，此時的利用效率也因此降低，而連帶也會影響產量，即使有時雖然可以充份利用頻寬，但卻可能造成搶用頻寬不公平的狀態。此外由於 ACR 的影響，也造成了 switch 的 buffer occupancy 經常作巨幅的振盪，如此造成惡性循環。

反之，ERAMM 在各 vc 開始傳送不久之後，即可在每個 switch 的協調之下，獲得公平的頻寬配置，並以完全收斂的穩定速率傳送資料。可以達到 no cell loss、no buffer oscillation、high throughput 的目標。

4.3.3 overhead 問題

另外，由於在 EPRCA 中，發送端必須以一固定封包數間隔(Nrm)，不斷地發送 RM cells，然而

這個 N_{rm} 的值不能太大，通常為 32，這是為了不斷可以供 switch 計算最新的平均速率值，以使發送端的 ACR 能獲得較快反應(fast response)的調整，而這些 RM cell 是屬於額外的 control cell，無形中成為 overhead，造成頻寬使用上的浪費，使得接收端的產量也因而受到影響。而在 ERAMM 中，則不需像 EPRCA 那般密集地發送 RM cells，只要一段間隔才發送一個 RM cell 即可，其目的是用於檢查是否有後來才加進網路系統的 ABR VC，以便再重新作頻寬分配(bandwidth reallocation)，相較之下 ERAMM 有較少的 RM cell overhead。

5 結論與未來發展

本文介紹多種過去在處理 ATM 網路上的 ABR 流量控制所被提出的相關研究及方法，分別評估比較諸多方法之間的優缺，並針對一些設計考量，結合速率式(rate-based)控制與頻寬配置的觀念，提出 ERAMM 流量控制演算法。藉由 ERAMM，我們可以輕易獲得各 vc 之間的公平配置頻寬，穩定傳輸速率和 buffer occupancy，零封包流失等優點，因此 ERAMM 是一個可靠性與穩定性極佳的 ABR 流量控制演算法。

關於本文亦有尚待改進之處，首先是在 ERAMM 運作上可能發生的主要缺點是若發送端的 traffic type 並不一定處於 busy 狀態的話，可能有些 vc 的頻寬利用率太低，而其它的 vc 有又無法有效利用的情況，會造成頻寬配置上的浪費。所以若處於其它狀態之下的 traffic type，應再多作這方面的改進設計，或可以考量採用其它不是固定配置頻寬的演算法來輔助。因此，根據何種型態的 source traffic type，而採用適合的演算法來，是屬於取捨的問題，也是一個值得探討的問題。此外，在 Error handling 方面，例如 RM cell 在傳送過程中的處理問題，以及在雙向同時傳送資料傳送方面，還需要再作設計與考量。

6. 參考文獻

- [1] The ATM Forum Technical Committee, "The ATM Forum Traffic Management Specification," V4.0, 1996.
- [2] Raj jain et al., "Source Behavior for ATM ABR Traffic Management : An Explanation", IEEE Communication Magazine, Nov. 1996.
- [3] A. Arulambalam et al., "Allocating Fair Rates for Available Bit Rate Service in ATM Networks", IEEE Communication Magazine, Nov. 1996.
- [4] Thomas M. Chen et al., "The Available Bit Rate Service for Data in ATM Networks", IEEE Communication Magazine, May 1996.
- [5] Flavio Bonomi and Kerry W.Fendick, "The

- Rate-Based Flow Control Framework for the Available Bit Rate ATM Service", IEEE Network, March/April 1995.
- [6] Yoon Chang, Nada Golmie and David Su, "A Rate Based Flow Control Switch Design for ABR Service in an ATM Network", Twelfth International Conference on Computer Communication (ICCC'95), August 1995.
- [7] R. Walthall and M. Clement, "Simulation and Analysis of the performance of EPRCA in a Wide Area ATM Network Consisting of Both ER and EFCI Switch Mechanisms", Fourth International Conference on Telecommunications System, Modelling and Analysis, March 1996.
- [8] Raj Jain, "Congestion Control and Traffic Management in ATM Networks:Recent Advances and A Survey", Computer Networks and ISDN Systems, February 1995.
- [9] Aleksandar Kolarov and G. Ramamurthy, "End-to-End Adaptive Rate-Based Congestion Control Scheme for ABR Service in Wide Area ATM Networks", Technical Report, February 1995.
- [10] J.M. Jaffe, "Bottleneck Flow Control" IEEE Trans. Commun., vol. 29, July 1981.