

國立中山大學電機工程研究所

碩士論文

指導教授：李錫智 博士



*A Ratio-based Call Admission Control  
for ATM Networks*

研究生 陳宗群 撰  
中華民國 九十年 五月

# 摘要

在這一篇論文中，我們利用 Ratio-Based 允入控制( Call Admission Control, CAC)來計算既有連線所需求的頻寬，並依此來判斷當新的連線要求到達 switch 時，switch 是否該允許建立新的連線。我們的方法計算了實際在 switch 上傳輸的流量和 UPC 參數所宣告的流量，用這兩者的關係來計算既有連線所需求的頻寬。過去 Model-Based 允入控制是利用 UPC 參數所宣告的流量來計算既有所需要的頻寬，但是 UPC 參數內所宣告的流量往往是最差情況，以至於使用 Model-Based 的方法頻寬無法有效利用。Measure-Based 允入控制則量測過去的實際流量來計算既有連線所需要的頻寬，在一般情況下 Measure-Based 能夠更準確計算實際需求的頻寬，達到更好的頻寬使用率；然而當 Measure-Based 允許一個新的連線建立後，必需要一段時間估計值才會穩定，這時就很容易造成判斷錯誤而允許過多的連線建立。我們的方法結合了 Model-Based 和 Measure-Based 兩種方法的優點和特性，在沒有新連線加入的情況下 Ratio-Based 所估算的跟 Measure-Based 一樣，當有新連線加入時能夠像 Model-Based 一樣立即反映在需求頻寬上，並且快速的達到估計值的穩定。

# Abstract

We propose a novel call admission control which makes use of ratio-based traffic measurement to estimate the required bandwidth when a new call is issued. Existing approaches fail to estimate properly the required bandwidth. To alleviate the problem, we calculate the ratio between the measured mean rate and the mean rate declared by UPC parameters. The ratio and the target cell loss rate are used to estimate the required bandwidth to make decision if a new call is accepted or rejected. Because of more accurate estimation of required bandwidth, our method can provide a better control on quality of service.

# 目錄

摘要.....	i
Abstract.....	ii
第一章 簡介.....	1
第二章 CAC 概論.....	4
2.1 CAC 簡介及研究動機.....	4
2.2 Model-Based 的機制.....	8
2.3 Measure-Based 的機制.....	12
2.4 Model-Based 和 Measure-Based 的優缺點.....	15
第三章 Ratio-Based 的機制.....	17
第四章 分析與模擬.....	21
4.1 數學分析.....	21
4.2 模擬結果.....	27
第五章 總結.....	36

參考文獻.....38

# 圖目錄

圖 2.1	on/off 模型 .....	9
圖 2.2	原始的 on/off 模型和改進的 on/off 模型 .....	10
圖 2.3	Measure window .....	12
圖 3.1	Ratio-Based 的演算法 .....	20
圖 4.1	Measure-Based 的連線 .....	22
圖 4.2	Ratio-Based 的連線 .....	25
圖 4.3	Measure-Based 的模擬結果 1 .....	29
圖 4.4	Ratio-Based 的模擬結果 1 .....	29
圖 4.5	Measure-Based 的模擬結果 2 .....	30
圖 4.6	Ratio-Based 的模擬結果 2 .....	30
圖 4.7	Measure-Based 的模擬結果 3 .....	31
圖 4.8	Ratio-Based 的模擬結果 3 .....	31
圖 4.9	使用率的比較 .....	32
圖 4.10	細胞遺失率的比較 .....	32
圖 4.11	過度負載下 Measure-Based 的表現 .....	34
圖 4.12	過度負載下 Ratio-Based 的表現 .....	34

# 第一章 簡介

為了讓高速網路能支援多媒體應用程式，網路必須提供一些必要的服務品質保證(QoS)，如細胞遺失率(cell loss rate)、延遲限制 (delay bound)等。ATM 網路是一種現在已經使用的網路技術，它的特色就在於能夠提供良好的服務品質保證(QoS)，因此有很多的多媒體應用的研究都是以 ATM 網路為基礎。不同類型的網路服務所要求的服務品質保證也不同，而 ATM 網路將這些網路服務分成以下幾種類別：固定傳輸率(CBR)、即時的變動傳輸率(rt-VBR)、非即時的變動傳輸率(nrt-VBR)、最低可得到的傳輸率(ABR)、不保證傳輸率(UBR)。這些類別的網路服務都有各自的服務品質保證(QoS)，為了能夠達到這些保證並且得到最佳的網路使用率，各種方法被提出，如允入控制 (call admission control)、排程法(scheduling)等等。我們在這篇論文中將針對 ATM 網路下 VBR 服務的允入控制來進行改良。

在 ATM 網路下，任何一個點對點的連線必需先建立連線才能進行資料的傳輸，允入連線控制必需依現有系統上的資源以及連線所要求的資源來判斷是否連線可被建立。它的原理是計算在交換器(switch)上剩餘的頻寬，再計算新連線所要求的頻寬，如果剩餘的頻寬能夠滿足新連線的要求時，新的連線就會被允許在 switch 上建立；反之如果剩餘的頻寬不能滿足新連線的要求時，新的連線就不會被允許在

switch 上建立，以避免新的連線和舊有的連線爭奪有限的頻寬造成服務品質低落，進而無法達到品質的保證。

現今已有不少允入控制機制被提出來，其中主要可分為 Measure-Based 以及 Model-Based 的方法。早期 Model-Based 的方法認為只 traffic source 才能正確的了解它所要傳輸的流量(traffic)特性，因此利用每個連線所宣告的 UPC 參數(Usage Parameter Control):最高速率(peak rate)、平均速率(mean rate)、最大 burst(burst size)這三項來建立一個 on/off 的模型，而這模型的特性即是 traffic source 的流量特性。所以它由每一個連線宣告的參數來計算相對應的頻寬，這種方法相當的簡單、快速，但是這種結果卻是完全忽視了實際的傳輸情況。尤其是 VBR 的流量特性難以用簡單的參數來表示，而 UPC 參數所描述的僅僅只有三項太過於稀少，再者流量在經過網路設備上暫存器(buffer)的處理後特性會變得更不相同，所以 Model-Based 的方法在這方面便產生了問題。為了因應 Model-Based 在這方面的缺陷 Measure-Based 的方法被提出來。Measure-Based 的方法量測實際的流量，計算出實際流量的特性，以此來計算相對應所需要的頻寬。很明顯的這個方法與 Model-Based 相反，它只考慮實際的流量如何不管 UPC 參數如何宣告，它的好處在於能夠反應出實際的流量，而且可以動態的保留頻寬。但這種方法仍有缺點，它是計算過去實際的流量來估計所需要的頻寬，但是對新的連線而言由於才建立沒多久，系統無法掌握到新連線的狀態，於是就會產生盲點。對於系統而言，只好等到這些新連線建立一段夠久的時間才能得到足夠的取樣，而在系統完全掌握到新連線的流量特性之前，系統所估計的需求頻寬都會有估計錯誤的危險。



而在我們的想法裡，我們認為應該把 UPC 參數和實際的流量特性整合在一起，我們所提的新方法計算”實際的平均速率”和”UPC 所宣告的平均速率”兩者之間的比率，再由這個比率關係跟最後系統的連線狀態算出相對應的所需頻寬，這個方法即是 Ratio-Based CAC。我們的方法繼承了 Measure-Based 能夠準確、動態保留頻寬的優點，也結合了 Model-Based 能夠立即反應新加入連線的優點，更重要的是，我們的方法實作的複雜度和 Measure-Based 相近並沒增加多少複雜的步驟。

本篇論文的結構如下，第二章將介紹連線控制機制研究背景，也一併將 Model-Based 和 Measure-Based 的機制、優缺點做說明。第三章描述 Ratio-Based 的如何改進 Measure-Based 的缺點，並對演算法進行詳細說明。第四章則是數學的分析和模擬結果，證明我們的方法不論是理論上還是實際上的確都是優於舊有的 Measure-Based。第五章則是對本篇全文做總結。

## 第二章 CAC 概論

### 2.1 CAC 簡介和研究動機

多媒體時代的來臨，網路使用者對網路有了更多的需求，例如視訊及音訊都需要即時的傳送以便接收端能在時效內順利撥放，而讀取圖片則需要使用大量的頻寬，高速的區域網路於是在此需求下因應而生。而隨著高速網路來的問題是：我們需要建立一些有效的制度及方法來管理整個網路，並且保證服務品質能滿足所有使用者的要求。

允入控制是網路管理制度中的一部分，它針對每個連線的不同特性及需求來保留相對應的頻寬，如果再配合上其他的網路管理技術，網路的整體運作就會井然有序且保持最佳效能。

那麼究竟什麼是允入控制，我們做個簡單的敘述：對 ATM 網路的各個使用者而言，點對點要傳輸資料之前必需先建立連線，將需要的傳輸服務類型、最高傳輸速率(peak rate)、平均傳輸速度(mean rate)、可允許的延遲範圍(delay bound)、可允許的細胞遺失率(cell loss rate)等等的資料傳至 switch，並且要求建立連線。在接到新連線的建立要求後，switch 先計算已建立的連線所需要的頻寬，扣去這一部分

即是系統所剩餘能給新連線的頻寬，再由新連線所宣告的各種資料來判斷剩餘的頻寬是否能夠滿足新連線的要求，假設剩餘的頻寬可以滿足新連線的要求時就會允許新線建立，反之就必需拒絕連線要求。我們可以想見的是，如果系統允許了超過它可以容納的連線建立，同時間內會有大量的資料進到系統內，但系統並無法在短時間送出去，於是無法傳送出去的資料就會被丟棄或是延遲傳送，就會造成網路傳輸上大量的資料遺失和延遲。相反的，如果系統只允許極少的連線建立，則網路頻寬大部分時間都會處在閒置的狀態，但卻有許多的使用者在等待系統允許他們的連線。因此，綜合以上兩點，我們可以得到一個結論，一個最理想的連線允入控制應該在不影響連線服務品質下盡可能的允許最多的連線建立。

另外考慮到每種網路傳輸服務的特性都不一樣，而允入控制機制大多是依照特定類型的網路傳輸服務來設計，如果把所有種類的連線都由同一個連線允入機制來控制並不適合，一個比較好的方法是把各種連線服務分類，如 CBR、VBR、UBR、ABR 等四種連線服務可以分別用四個適合的允入連線機制來控制。事實上也可以再把這些再做分類，例如可以把 VBR 傳輸服務再分成隨選視訊、語音等，當分類分得愈細時，同一類的連線其流量特性亦會愈相近，對於連線允入控制機制而言更能準確的掌握流量的特性，達到更好的效能。

衡量一個 CAC 的優劣主要有三項參考指標：**使用率(Utilization)**、**細胞遺失率(Cell loss rate)**以及**複雜度(Complexity)**。在此敘述它們在 CAC 中扮演的角色：

## ● Utilization

網路設備是一項極為昂貴的投資，為了能夠達到最大的效益，自然是希望網路頻寬能夠盡可能的被利用。在前面已經說過在連線建立的時候 CAC 必須保證連線建立後能夠提供必要的頻寬，否則不能讓新的連線建立，可以想見的是如果每個連線都以它最大可能的速率來保留一定能提供最好的服務品質，但是這樣卻會犧牲珍貴的網路頻寬，畢竟連線並不是一直以最高速率來傳送資料。若保留給每個連線的頻寬過少，想必大部分時間頻寬的使用率會是滿檔，但是服務品質卻會無法滿足保證。因此使用率和服務品質往往都是權量如何才能達到平衡。而使用率可以用以下式子來表示

$$U = \frac{M_{avg}}{C_{max}} \text{-----}(1)$$

其中 U 為使用率， $M_{avg}$  表示實際連線的平均速率， $C_{max}$  為系統最大可傳送的頻寬容量。

若是實際連線的平均速率愈高，使用率就會愈理想。在最理想的情況下當然是希望系統的頻寬完全被使用到，即  $U=1$ ，一般來說這是不可能達成的理想值。

## ● Cell Loss Rate

細胞遺失率(cell loss rate)是網路分析裡最重要的一項數據，細胞遺失率會直接影響的是傳送的資料品質，如果細胞遺失率過大會使得資料無法使用或必需重傳。一個跟它直接相關的名詞是”延遲”，延遲簡單的來說即是一筆資料從某一點完整傳輸到另一點所需的時間。延遲產生的原因相當多，為了簡單說明，在這裡我們僅討論與連線允入控制有關的等待延遲這一項。當進入系統的輸入速率大於系統的輸出速率時，無法立即傳送出去的資料必需暫存在暫存器內，直到先前的資料都被傳送後才能被傳送出去，這是等待延遲產生的主因，如果延遲過大就會造成多媒體服務品質的低落，如影像的停格、破碎和聲音的失真、無法辨識。一般來說資料的傳送對細胞遺失率比較敏感、對延遲不敏感，但多媒體服務則是剛好相反對細胞遺失率比較不敏感、對延遲敏感。在這篇論文中我們所討論的是 VBR 的流量，而 VBR 多半是用在多媒體服務，為了適合多媒體服務的特性，因此我們設定的系統記憶體有一上限，公式如下

$$\frac{L}{C_{\max}} \leq D \text{-----}(2)$$

其中，D 是最大可允許的延遲，L 為使用的記憶體大小。在這個限制下，因暫存在記憶體內而造成的等待延遲一定小於多媒體服務的要求，如果有過多的細胞無法立即被傳送出去就會表現在細胞遺失率上，所以這篇論文只從細胞遺失率來做比較，不從延遲

方面來討論。

## ● Complexity

複雜度也是 CAC 中一項最重要的指標，許多研究在理論上都能夠達到相當良好的效果，但複雜度高的方法往往就會造成硬體實作上的困難，甚至會成為整個系統的瓶頸。即使現在的硬體科技突飛猛進，但考量到硬體成本的因素時，較簡單的方法法就能有效降低成本並具有實際效益。

從以上幾點，我們將研究的目標定為在複雜度不超過現有方法過多的情況下，盡可能的增進使用率和細胞遺失率的表現。接下來，我們介紹以前所使用的方法，並比較兩者在這三方面的優缺點。

## 2.2 Model-Based 的機制

Model-Based 的機制是利用 on/off 模型來模擬一個 traffic source，[1]與[2]。當一個連線要建立之前必需先將 UPC 參數傳至 switch 即：最高速率( $\lambda_p$ )、平均速率( $\lambda_s$ )、最大 burst 大小( $B_s$ )。Model-Based 利用這些參數建立一相對應的 on/off 模型，再計算出所需的頻寬為何。圖 2.1 為一 on/off 的模型

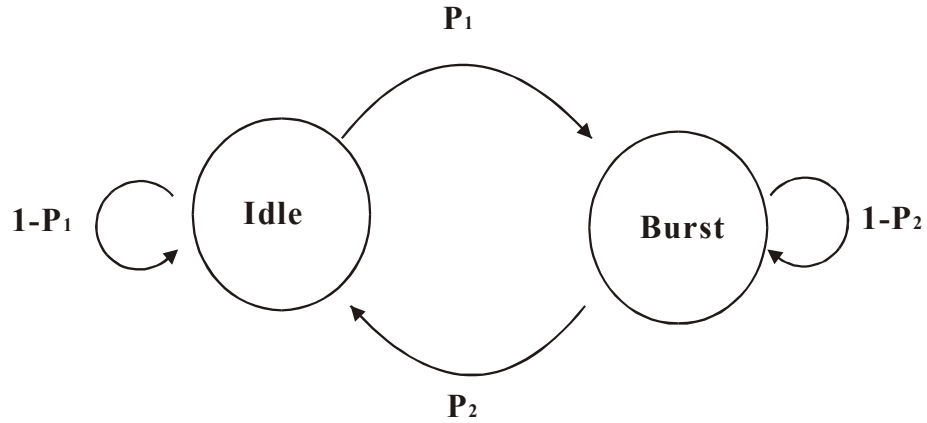


圖2.1 on/off 模型

$P_1$ 代表現在是 Idle 狀態變換為 Burst 狀態的機率為多少， $P_2$ 則代表現在是 Burst 狀態變換為 Idle 狀態的機率為多少。

在 on/off 模型下，系統如果在 Idle 狀態時完全不送資料，如果是在 Burst 狀態時會以最大速率傳送資料。Doshi [6] 證明了 Idle 和 Burst 兩個狀態變化的機率為

$$P_1^{-1} = \frac{B_s}{\lambda_p} \frac{(\lambda_p - \lambda_s)}{\lambda_s} \text{-----}(3)$$

$$P_2^{-1} = \frac{B_s}{\lambda_p} \text{-----}(4)$$

而  $P_1^{-1}$  除了是  $P_1$  的倒數外，也會等於系統在 Idle 狀態的平均週期長度  $T_{off}$ ， $P_2^{-1}$  則會等於系統在 Burst 狀態的平均週期長度  $T_{on}$ 。因此可以計算系統在 on 的機率為

$$P_{on} = \frac{P_2^{-1}}{P_1^{-1} + P_2^{-1}} = \frac{\lambda_s}{\lambda_p} \text{-----(5)}$$

在這個模型中系統的傳輸速率只有  $\lambda_p$  和 0 兩種，然而這個模型並沒有考慮到 Buffer 的效應，一般網路裝置都會有 Buffer 以暫存輸入的資料，經過 Buffer 之後的流量就會變得更平均，而原本的模型只能代表在**最差的情況下**所可能出現的狀況，考慮這兩種情況可能的速率如圖 2.2

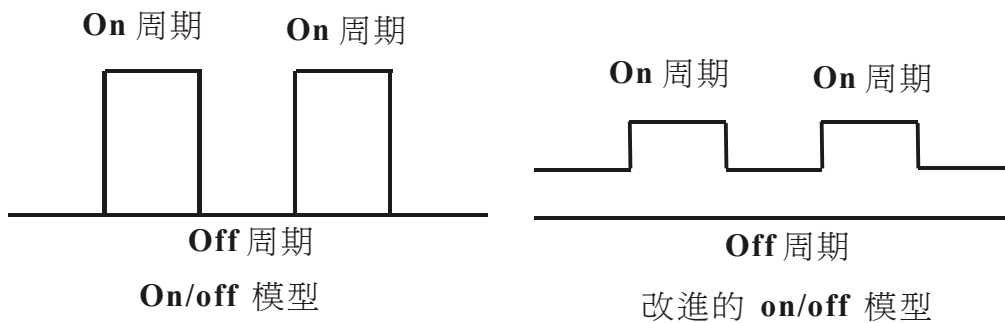


圖2.2 原始的on/off模型和改進的on/off模型

為了反映 Buffer 的作用，改進的 on/off 模型被提出來，它假設系統中有 Buffer，並且假設在一般情況下為半滿，則在 on 週期的高速  $\lambda_H$  及在 off 週期的低速  $\lambda_L$  分別為

$$\lambda_L = \max \left[ 0, 1 - \frac{T_{on}}{T_b} \right] \lambda_s \text{-----(6)}$$



$$\lambda_H = \min \left[ 1, \frac{T_{on}}{T_b} \right] \lambda_p + \lambda_L \text{-----}(7)$$

其中

$$T_b = \frac{L_b}{2C_{\max}} \text{-----}(8)$$

在這裡  $L_b$  為 Buffer 的大小。一個沒有細胞損失的流量不管 Buffer 的大小為何它的平均值應該都一樣，會改變的是它的速率的高低的機率。在 Lee 和 Song [5] 的研究中說明了，所有連線的變異數如同(10)式所示

$$\overline{M}^{UPC} = \sum_{i=1}^n \lambda_s^i \text{-----}(9)$$

$$\left( \sigma^{UPC} \right)^2 = \sum_{i=1}^n \left( \lambda_H^i - \lambda_L^i \right)^2 P_{on}^i (1 - P_{on}^i) \text{-----}(10)$$

上式中  $\lambda_s^i$  為第  $i$  個連線的平均速率， $\lambda_H^i$ 、 $\lambda_L^i$  分別為第  $i$  個連線在高、低速狀態時的速率， $P_{on}^i$  為第  $i$  個連線在 on 狀態的機率， $\overline{M}^{UPC}$  和  $\sigma^{UPC}$  為 Model-Based 下所有連線的平均速率和標準差。

## 2.3 Measure-Based 的機制

Measure-Based 的精神在於藉由量測最近使用的頻寬量測出連線實際的流量特性，[4]、[5]即為這一類型。

這個方法把時間軸切成一段一段等距的時間間距  $T$ ，這個間距被稱為 sub-window，再取連續  $N$  個 sub-window 稱為一個 window。當系統已有連線建立並開始傳輸資料，處理程序量測在每一個 sub-window 總共有多少的資料量進入，計算進入的資料量和時間的關係可以得到在這個 sub-window 的平均輸入速率，如同圖 2.3 所示。

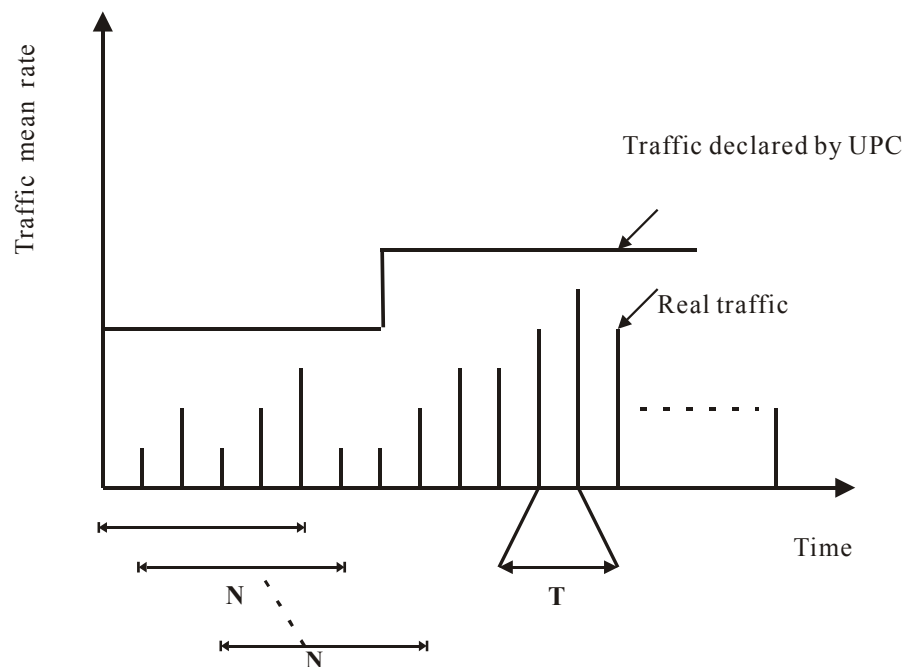


圖2.3 Measure window

平均速率可由以下公式求得

$$\lambda_i^{mes} = \frac{K_i * L_{cell}}{T} \text{-----(11)}$$

$\lambda_i^{mes}$  為在第  $i$  個 sub-window 的平均速率， $T$  表示 sub-window 的週期， $K_i$  為在第  $i$  個 sub-window 內進入的總細胞數， $L_{cell}$  為每一個細胞的長度。

當計算了  $N$  個 sub-window 的平均速率後，再由這  $N$  個樣本計算整個 window 的平均速率和標準差，如(12)、(13)

$$\overline{M}^{mes} = \frac{1}{N} \sum_{i=1}^N \lambda_i^{mes} \text{-----(12)}$$

$$\left(\sigma^{mes}\right)^2 = \frac{1}{N-1} \sum_{i=1}^N \left(\lambda_i^{mes} - \overline{M}^{mes}\right)^2 \text{-----(13)}$$

其中  $M^{mes}$  代表 Measure-Based 在這個 window 內所估測的平均輸入速率， $\sigma^{mes}$  代表這 window 內速率的標準差。在 Ren 和 Ramamurthy [7] 的論文中提到，當知道平均傳輸速率和標準差時，可以用這兩者及細胞遺失率計算出要保留的頻寬

$$C_{mes} = \overline{M}^{mes} + \zeta \sigma^{mes} \text{-----(14)}$$

$C_{mes}$  代表需要保留給現有全部連線的頻寬。而  $\zeta$  為一個與細胞遺失率有關的參數，其公式如下

$$\zeta \approx 1.8 - \log_{10}(\eta) \text{-----(15)}$$

其中  $\eta$  代表可允許的細胞遺失率。

如果是有新連線要求建立，由於無法得知其真實的平均速率和標準差，因此利用 Model-Based 所使用的方法來計算平均速率和標準差，再使用(14)式來估算所需的頻寬。令  $C_{in}$  是由 Model-Based 計算出來的頻寬，若要保證現有連線的服務品質不變，而且能夠滿足新連線所要求的服務品質，則  $C_{max}$  必需大於現有連線要求的頻寬及新連線所需求的頻寬之和，即

$$C_{max} > C_{in} + C_{mes} \text{-----(16)}$$

如果成立的話，即允許新的連線成立，否則就必需拒絕。

必需補充的是，假設輸入速率穩定而且沒有很大變化，當  $N$  愈大時系統愈能得到足夠的樣本，所估測的頻寬也愈準確。相反的，假設輸入速率變動很大， $N$  愈小愈能夠立即反應連線速率的變動。如何在這兩者間取得平衡也是一個議題。

## 2.4 Model-Based 和 Measure-Based 的優缺點

在前面我們介紹了 Model-Based 和 Measure-Based 的方法。在這

一小節裡將就這兩個方法有何優缺點來分析。

Model-Based 的優缺點:

優點:只有新連線加入時才會計算需求的頻寬，因此計算量和複雜度都很低。一般都會以最壞情況來保留頻寬，因此都能達到服務品質保證。而且能夠立即保留足夠的頻寬給新的連線。

缺點:只依 UPC 參數來保留頻寬，無法反應實際的連線速率，也無法動態的分配頻寬。如我們在 2.2 節所說的，traffic 在經過網路裝置後它的特性已經跟原本不同了，在無法知道實際的特性下，為了達到服務品質保證常常必需考慮最壞的情況，因此它的使用率並不高，會浪費網路頻寬。

Measure-Based 的優缺點:

優點:藉由量測實際的 traffic 來估算所需的頻寬，因此可以較為準確的計算出真正需求的頻寬，也可以動態的分配頻寬，頻寬使用率較高。大部分的計算過程並不複雜，因此可行性亦可。

缺點:在大部分的情況下能夠運作良好，但是它的計算過程是由過去的實際輸入速率來計算的，因此當有新連線被建立時系統無法立即的反應，當 window 的大小  $N$  愈大時愈不容易立即的反應。因此靜態連線時表現優異，但動態連線時易計算到錯誤的結果甚至允許過多連線建立。

從以上的分析，我們可以知道 Measure-Based 的方法有較佳的頻寬使用率，在靜態連線時會有最優異的表現，但動態連線時卻不容易立即反應新的連線狀態。相反的，Model-Based 無法達到良好的頻寬使用率，但是卻可以立即反應新的連線狀態。我們的目標即是希望新

的機制能夠有像 Measure-Based 般良好的頻寬使用率，以及可以像 Model-Based 般能夠立即反應新連線狀態。

## 第三章 Ratio-Based 的機制

我們在前面解釋了 Measure-Based 的缺陷，為了解決這些缺陷，我們提出了新的機制，並在這一章開始說明新機制的作法。

我們將 Ratio-Based 的方法說明如下：

1. 將時間軸切成同等大小的 sub-window，長度為  $T$ 。集合  $N$  個 sub-window 即為一個 window。
2. 當有新連線要求建立時，必需在該 sub-window 結束時才會處理。
3. Ratio-Based 量測每一個 sub-window 內的平均速率，再將這個平均速率和宣告的平均速率相除，實際平均速率/宣告平均速率即是在這個 sub-window 內的取樣結果，如下式

$$\beta_i = \frac{\lambda_i^{mes}}{\lambda_i^{UPC}} \text{-----}(17)$$

其中， $\lambda_i^{UPC}$  是在第  $i$  個 sub-window 內所有連線的宣告平均速率。 $\beta_i$  是在第  $i$  個 sub-window 的實際平均速率/宣告平均速率之比值。

4. 利用在第 3 步驟所得到的比值，將過去  $N$  個 sub-window 的取

樣的結果再行計算這些比值的平均值和標準差，如(19)、(20)式所示

$$\beta = \frac{1}{N} \sum_{i=1}^N \beta_i \text{-----}(19)$$

$$(\sigma_\beta)^2 = \frac{1}{N-1} \sum_{i=1}^N (\beta_i - \beta)^2 \text{-----}(20)$$

$\beta$  代表在過去這個 window 周期內實際平均速率和宣告平均速率比值的平均， $\sigma_\beta$  代表在這個 window 內比值的標準差。將這兩個數再乘上所有連線宣告的速率，即為 Ratio-Based 的方法所估測的平均速率和標準差，即(21)、(22)所示

$$\overline{M}_r = \lambda_{N+1}^{UPC} \beta \text{-----}(21)$$

$$\sigma_r = \lambda_{N+1}^{UPC} \sigma_\beta \text{-----}(22)$$

$\lambda_{N+1}^{UPC}$  為在第 N+1 個 sub-window 內宣告平均速率的總合， $\overline{M}_r$  代表估算的平均速率， $\sigma_r$  代表速率的標準差。

- 如同(14)式中 Measure-Based 估算頻寬的做法，我們可以得到現有連線需求的頻寬為

$$C_r = \overline{M}_r + \zeta \sigma_r \text{-----}(23)$$



若有一新的連線要求被提出，假設保留給現有連線的頻寬加上新連線所要求的頻寬小於系統的容納時，則可以允許新的連線建立，反之則拒絕連線的要求。

$$C_{\max} > C_{in} + C_r \text{-----}(24)$$

圖 3.1 為整個 Ratio-Based 的流程圖。從以上所說的步驟，可以知道 Ratio-Based 的方法基本上和 Measure-Based 的作法很相近，但是 Ratio-Based 的作法也會整合連線的變動狀況。當有新連線加入時，透過 UPC 參數所宣告的速率，新的連線狀態會立即反應在保留的頻寬上，這一點是 Measure-Based 的方法無法達到的。

要說明的是，我們的方法是使用實際值和宣告值的關係，如果每個連線的比值關係愈相近，我們的方法會有愈好的效能。

## Ratio-Based CAC Procedure

---

### VARIABLES

$C_{max}$  : Maximum bandwidth

$C_{in}$  : Estimated bandwidth for new calls

$C_r$  : Estimated bandwidth for existing connections

### MEASURE

Monitor input traffic to measure mean rate and variance. This measurement must be repeated irrespective of the rest steps;

### EVENT

#### **New call request**

Estimate  $C_r$  for the existing connections based on previous  $N$  sub-windows;

Estimate  $C_{in}$  for the new call;

**if**  $C_{max} \geq C_{in} + C_r$

**then** the new call is admitted;

    Add new call's UPC mean rate to total UPC mean rate;

**else** the new call is rejected;

#### **Call leave**

Total UPC mean rate minus leaving connection's UPC mean rate;

---

圖 3.1 修正後的演算法

## 第四章 分析與模擬

### 4.1 數學分析

在這一章裡我們將先用數學分析的方式來證明我們的方法能夠提供更好的服務品質保證。我們分別對兩種情況來進行分析：靜態連線以及動態連線。

#### I. 靜態連線

首先考慮的是在 window 內連線固定沒有任何新的連線加入或舊的連線離去。在這種情況下，由 UPC 參數所計算出來的平均速率在任何一個 sub-window 都是固定的，即

$$\lambda_1^{UPC} = \lambda_2^{UPC} = \dots = \lambda_N^{UPC} = \lambda_{N+1}^{UPC}$$

在這裡  $\lambda_1^{UPC}$  為在第 1 個 sub-window 內的宣告平均速率總合。從第三章內的 Ratio-Based 的公式可以得到下面的結論

$$\overline{M}_r = \beta \lambda_{N+1}^{UPC} = \frac{1}{N} \sum_{i=1}^N \lambda_i^{mes} = \overline{M}^{mes} \text{-----}(25)$$

以及

$$\begin{aligned}
(\sigma_r)^2 &= \frac{(\lambda_{N+1}^{UPC})}{N-1} \sum_{i=1}^N (\beta_i - \beta)^2 = \frac{1}{N-1} \sum_{i=1}^N \left( \lambda_i^{mes} - \overline{M}^{mes} \right)^2 \\
&= (\sigma^{mes})^2
\end{aligned}
\tag{26}$$

由(25)、(26)兩式可以知道，在靜態連線下 Ratio-based 和 Measure-based 所估測的需求頻寬是一樣的。因此，在這種情況下這兩者會有同樣的效能。

## II. 動態連線

接下來，我們考慮當連線為動態的情況，可能有新的連線加入或是舊的連線離去。我們考慮當有新連線加入的情況，假設如圖 4.1 的情況，在某一 window 內第 1 到第  $N$  個 sub-window 皆沒有連線變動，在第  $N$  個 sub-window 時有一個連線要求到達。

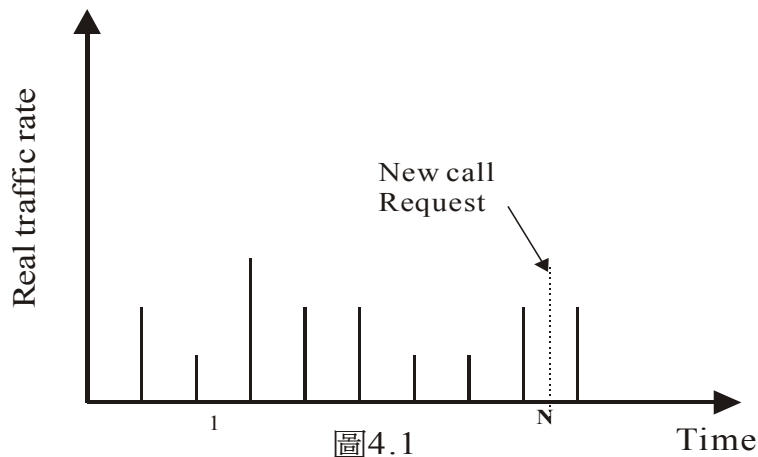


圖4.1

假設這個新連線的要求被接受，並且在第  $N$  個 sub-window 內即開始傳送資料。我們將之前已建立的連線稱為舊連線，在第  $N$  個 sub-window 建立的連線叫新連線，我們將這兩種連線從 Measure-Based 的式子中分出，並且分析在第  $N$  個 sub-window 結束時新連線所造成的變化。

要說明的是，在第 1 至第  $N-1$  個 sub-window 只有舊連線存在，新連線只有在第  $N$  個 sub-window 才建立，因此第 1 至第  $N-1$  個 sub-window 所計算到的速率皆為舊連線的速率，即

$$\begin{cases} \lambda_i^{mes} = \lambda_{old,i}; \text{ for } i = 1 \sim N-1 \\ \lambda_N^{mes} = \lambda_{old,N} + \lambda_{new} \end{cases} \text{-----}(27)$$

其中  $\lambda_{old,i}$  為舊連線在第  $i$  個 sub-window 內估測的平均速率， $\lambda_{new}$  為新的連線在第  $N$  個 sub-window 內實際傳輸的位元數除以 sub-window 的週期長度  $T$ 。把(27)式代入 Measure-Based 的平均速率公式內

$$\overline{M}^{mes} = \frac{1}{N} \sum_{i=1}^N \lambda_i^{mes} = \frac{1}{N} \sum_{i=1}^N \lambda_{old,i} + \frac{1}{N} \lambda_{new} = \overline{M}_{old} + \frac{1}{N} \lambda_{new} \text{-----}(28)$$

$\overline{M}_{old}$  為只計算舊連線時在整個 window 內的平均速率。再將(27)、(28)式代入 Measure-Based 的變異數公式內，則可以得到以下結果

$$\begin{aligned}
(\sigma^{mes})^2 &= \frac{1}{N-1} \sum_{i=1}^N \left( \lambda_i^{mes} - \overline{M}^{mes} \right)^2 \\
&= \frac{1}{N-1} \sum_{i=1}^N \left( \lambda_i^{mes} - \overline{M}^{old} - \frac{1}{N} \lambda_{new} \right)^2 \\
&= \frac{1}{N-1} \sum_{i=1}^{N-1} \left[ \left( \lambda_{old,i} - \overline{M}^{old} \right) - \frac{1}{N} \lambda_{new} \right]^2 + \frac{1}{N-1} \left[ \left( \lambda_{old,N} - \overline{M}^{old} \right) + \lambda_{new} - \frac{1}{N} \lambda_{new} \right]^2 \\
&= \frac{1}{N-1} \sum_{i=1}^N \left( \lambda_{old,i} - \overline{M}^{old} \right)^2 - \frac{2}{(N-1)N} \sum_{i=1}^N \left( \lambda_{old,i} - \overline{M}^{old} \right) \lambda_{new} + \frac{1}{(N-1)N^2} \sum_{i=1}^N \lambda_{new}^2 \\
&\quad + \frac{2}{N-1} \left[ \left( \lambda_{old,N} - \overline{M}^{old} \right) - \frac{1}{N} \lambda_{new} \right] \lambda_{new} + \frac{1}{N-1} \lambda_{new}^2 \\
&= (\sigma_{old})^2 - \frac{2}{N-1} \left[ \frac{1}{N} \sum_{i=1}^N \left( \lambda_{old,i} - \overline{M}^{old} \right) \lambda_{new} - \left( \lambda_{old,N} - \overline{M}^{old} \right) \lambda_{new} \right] + \frac{1}{N} \lambda_{new}^2 \\
&= (\sigma_{old})^2 - \frac{2}{N-1} \left[ \frac{1}{N} \sum_{i=1}^N \left( \lambda_{old,i} - \overline{M}^{old} \right) - \left( \lambda_{old,N} - \overline{M}^{old} \right) \right] \lambda_{new} + \frac{1}{N} \lambda_{new}^2 \\
&= (\sigma_{old})^2 - \frac{2}{N-1} \left[ \left( \overline{M}^{old} - \overline{M}^{old} \right) - \left( \lambda_{old,N} - \overline{M}^{old} \right) \right] \lambda_{new} + \frac{1}{N} \lambda_{new}^2 \\
&= (\sigma_{old})^2 + \frac{2}{N-1} \left( \lambda_{old,N} - \overline{M}^{old} \right) \lambda_{new} + \frac{1}{N} \lambda_{new}^2
\end{aligned}
\tag{29}$$

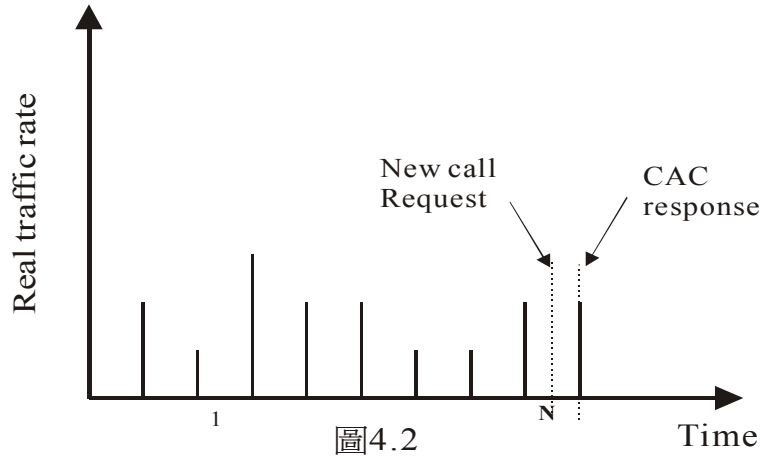
在上式中， $\sigma_{old}$  為只計算舊連線時的標準差。 $\sigma_{old}$  為

$$\sigma_{old}^2 = \frac{1}{N-1} \sum_{i=1}^N \left( \lambda_{old,i} - \overline{M}^{old} \right)^2$$

由(28)、(29)式得知，在第 N 個 sub-window 結束時，N 愈大新連線對平均速率的貢獻愈小，相同的 N 愈大新連線對標準差的貢獻也愈小。為了使用估測的結果可以更快的反映新連線的建立，Measure-Based 必須要把取樣的數目 N 減少，愈少的取樣數目愈能立即反應新的連線狀態，但是這卻會產生另一個問題，如果樣本的數目過少，利用這些樣本所估算的結果也就愈不準。

接下來，我們考慮上面同樣的情況在 Ratio-Based 下的結果會如

何。



在 4.2 圖，當在第  $N$  個 sub-window 時有一個新的連線要求到達，如同在 Measure-Based 所做的假設：這個新的連線會被允許建立。但是在第三章時我們提到了 Ratio-Based 的方法必需要等到這個 sub-window 結束才會建立。因此，第 1 至第  $N$  個 sub-window 內都只有舊連線存在。在第  $N$  個 sub-window 結束時因為允許了新連線的建立，所以新連線 UPC 參數所宣告的平均速率會被加入宣告平均速率的總告。如下式所表示

$$\begin{cases} \lambda_i^{mes} = \lambda_{old,i}; \text{ for } i = 1 \sim N \\ \lambda_1^{UPC} = \lambda_2^{UPC} = \dots = \lambda_N^{UPC}, \lambda_{N+1}^{UPC} = \lambda_N^{UPC} + \lambda_{new}^{UPC} \end{cases} \text{-----(30)}$$

上式裡  $\lambda_{new}^{UPC}$  代表的是新連線所宣告的平均速率。把上式代入

Ratio-Based 的平均速率和標準差的公式，可以得到

$$\begin{aligned}\overline{M}_r &= \frac{\lambda_{N+1}^{UPC}}{N} \sum_{i=1}^N \frac{\lambda_i^{mes}}{\lambda_i^{UPC}} = \frac{(\lambda_{old}^{UPC} + \lambda_{new}^{UPC})}{N} \sum_{i=1}^N \frac{\lambda_{old,i}}{\lambda_N^{UPC}} \\ &= \left(1 + \frac{\lambda_{new}^{UPC}}{\lambda_N^{UPC}}\right) \overline{M}_{old}\end{aligned}\tag{31}$$

$$\begin{aligned}(\sigma_r)^2 &= \frac{(\lambda_{N+1}^{UPC})^2}{N-1} \sum_{i=1}^N \left(\frac{\lambda_i^{mes}}{\lambda_i^{UPC}} - \beta\right)^2 \\ &= \frac{(\lambda_N^{UPC} + \lambda_{new}^{UPC})^2}{N-1} \sum_{i=1}^N \left(\frac{\lambda_{old,i}}{\lambda_N^{UPC}} - \frac{1}{N} \sum_{j=1}^N \frac{\lambda_{old,j}}{\lambda_N^{UPC}}\right)^2 \\ &= \left(1 + \frac{\lambda_{new}^{UPC}}{\lambda_N^{UPC}}\right)^2 (\sigma_{old})^2\end{aligned}\tag{32}$$

由上式可得

$$\sigma_r = \left(1 + \frac{\lambda_{new}^{UPC}}{\lambda_N^{UPC}}\right) \sigma_{old}$$

使用了 Ratio-Based 後，估測的平均速率和標準差會立即因新連線的加入而產生變化，增加的幅度會與新連線所宣告的平均速率有關，而且估算的平均速率和標準差不與 N 成反比，可以改善 Measure-Based 缺陷。



## 4.2 模擬結果

在這一節我們模擬 Ratio-Based 的允入控制和 Measure-Based 的允入控制機制，分別在不同的環境比較兩者所產生的差異。

在前面，我們證明了當靜態連線時 Ratio-Based 和 Measure-Based 的方法所估算到的結果是一樣了，只有當動態連線時兩者才會產生不同的結果。為了明顯比較兩種允入控制的效能，在這一節的模擬裡我們將環境設定成不斷有新連線要求建立，並且使用 on/off 模型來模擬每個連線的流量，而每個連線的生命期都很短，因此系統中不斷有新連線加入和舊連線離開。模擬的環境設定如下：

● Switch 上的設定

最大頻寬( $C_{\max}$ )	20Mbps
系統分配的 buffer(L)	250cells
sub-window 的週期(T)	5msec

● Traffic source 的設定(使用改進的 on/off 模型)

每個流量源宣告的 UPC 平均速率( $\lambda_s$ )	2.5Mbps
每個流量源宣告的 UPC 最高速率( $\lambda_p$ )	5.0Mbps
每個流量源的低速率( $\lambda_L$ )	2Mbps
每個流量源的高速率( $\lambda_H$ )	3Mbps
細胞最大可允許延遲	5msec
最大可允許的細胞遺失率	$10^{-5}$
每個流量源宣告的最大 burst 大小(B)	500cells
每次連線傳送的細胞數	20000~30000cells

當 window size  $N=10$

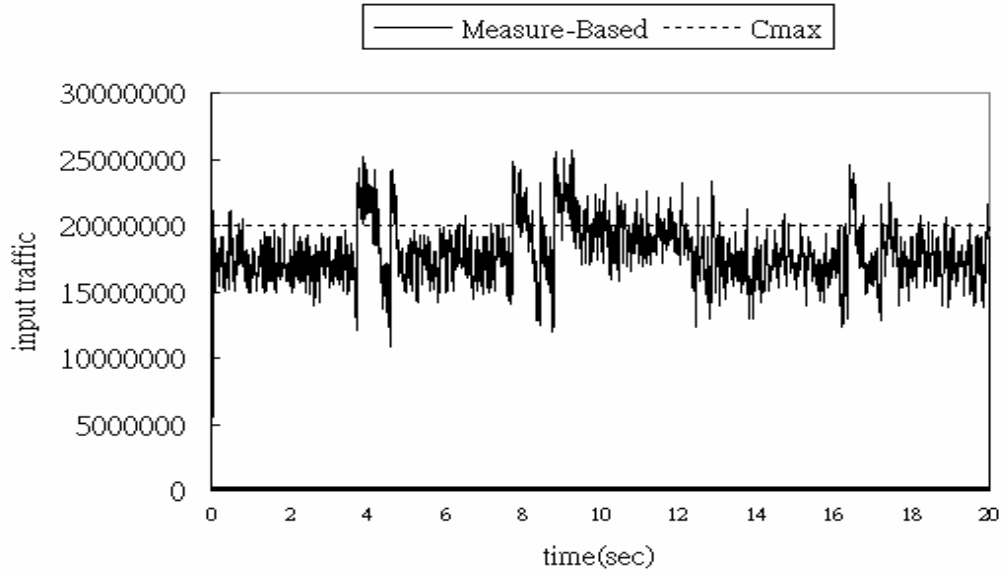


圖 4.3 使用 Measure-Based 管理的 input traffic

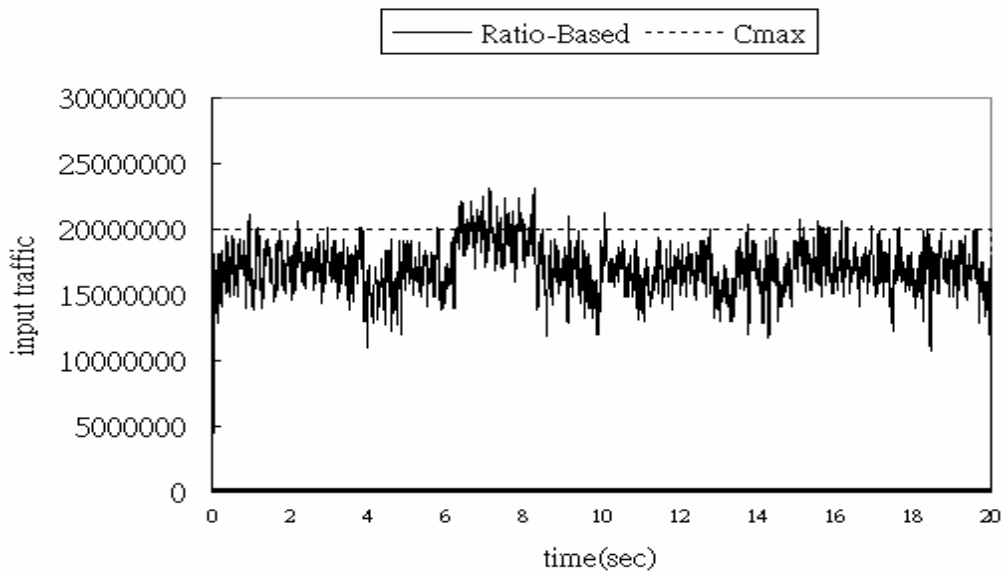


圖 4.4 使用 Ratio-Based 管理的 input traffic

當 window size  $N=50$

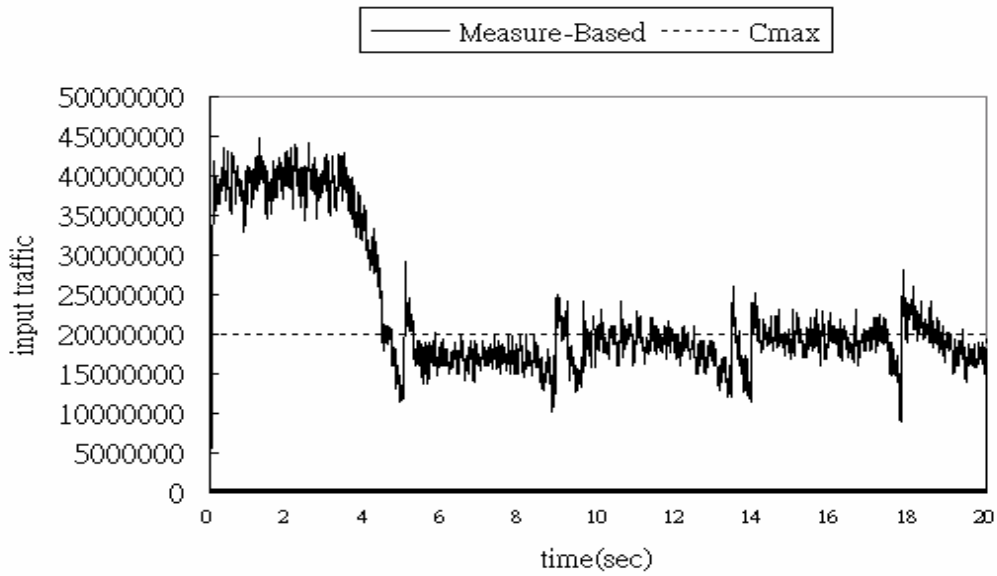


圖 4.5 使用 Measure-Based 管理的 input traffic

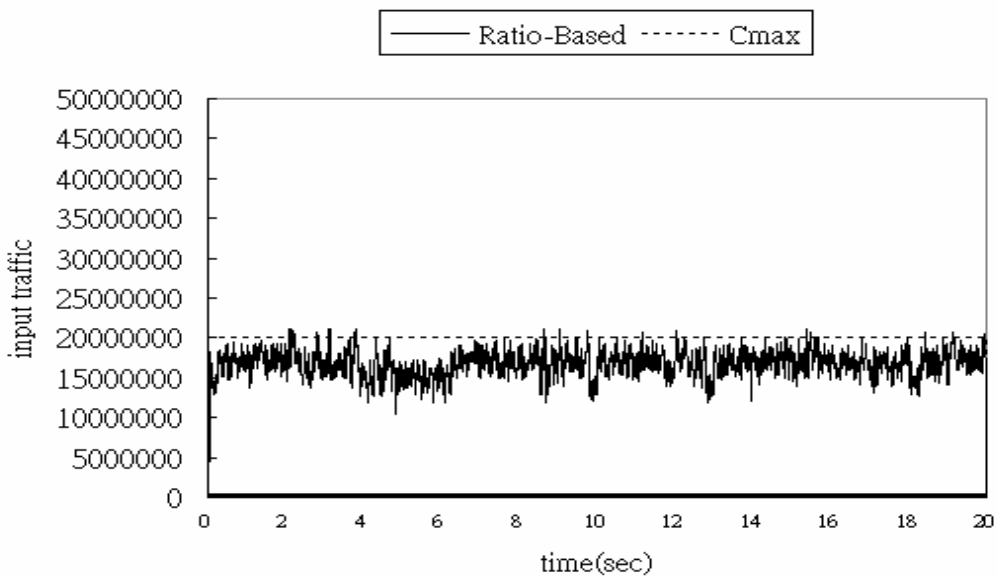


圖 4.6 使用 Ratio-Based 管理的 input traffic

當 window size  $N=100$

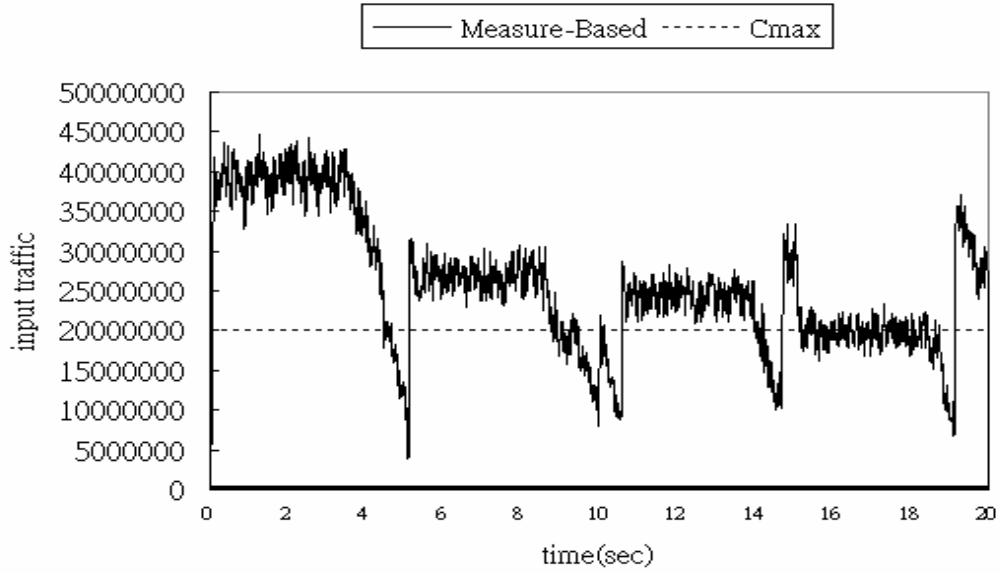


圖 4.7 使用 Measure-Based 管理的 input traffic

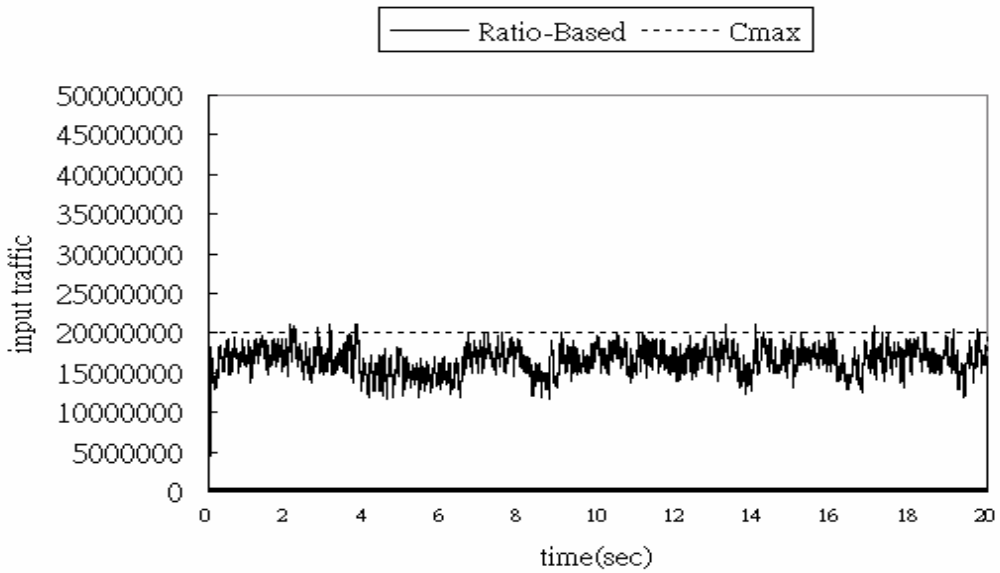


圖 4.8 使用 Ratio-Based 管理的 input traffic

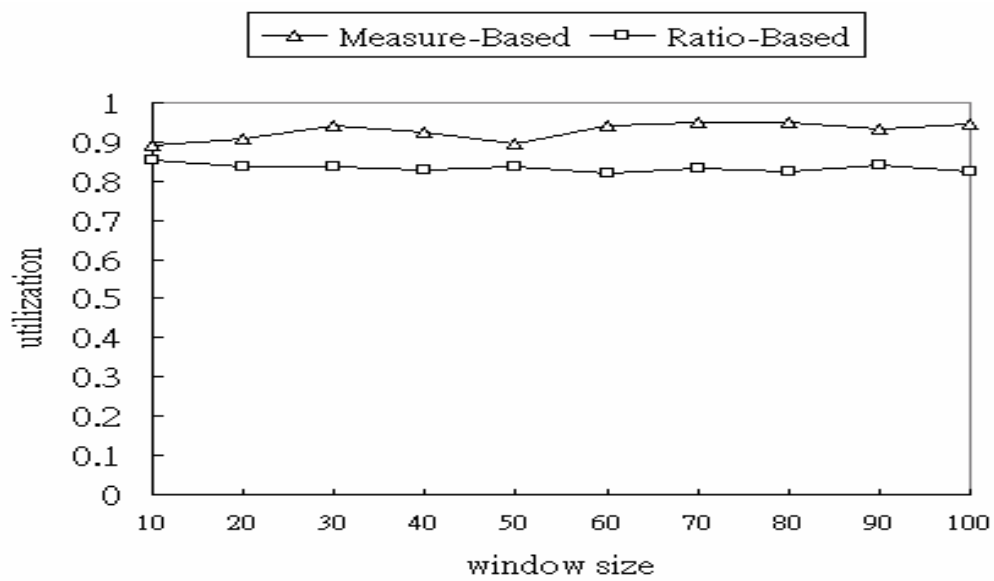


圖 4.9 Measure-Based 與 Ratio-Based 的使用率比較



圖 4.10 Measure-Based 與 Ratio-Based 的細胞遺失率比較

在實際的網路中，網路上的流量會被 switch 的 buffer 所緩衝，原本相當 busy 的流量會得穩定許多。因此，我們認為使用改進的 on/off 模型來模擬實際的網路流量會比用原始的 on/off 模型更準確。另外，因為每個連線的生命週期不一致，在模擬裡我們設定了每個連線總共傳送的細胞數為 20000~30000 之間的隨機數。

從模擬中，明顯的可以看得出來當 window 的大小  $N=10$  時，使用 Measure-Based 和 Ratio-Based 來管理的 switch 其連線的結果差異並不大，然而當 window 的大小逐漸增加時，Measure-Based 逐漸無法準確地管理連線的建立。相反的，Ratio-Based 卻沒有這個問題。在模擬中，當  $N=100$  時，實際的輸入流量比  $N=10$  時更為穩定、更加靠近系統的頻寬上限。在圖 4.10 裡我們比較了在穩定狀態下 Measure-Based 和 Ratio-Based 的細胞遺失率，Measure-Based 的細胞遺失率會隨著  $N$  增加而愈變大，Ratio-Based 的細胞遺失率則不會因為  $N$  增加而提高。

在上面的模擬中，我們假設一開始並沒有任何連線被建立，在接下來的圖 4.11、圖 4.12 將 switch 的初始狀態設定成已經建立過多連線，並且模擬在這種情況下 Measure-Based 和 Ratio-Based 兩種方法在過度負載時的表現。

N=100，每次連線傳送的細胞數為 5000~10000

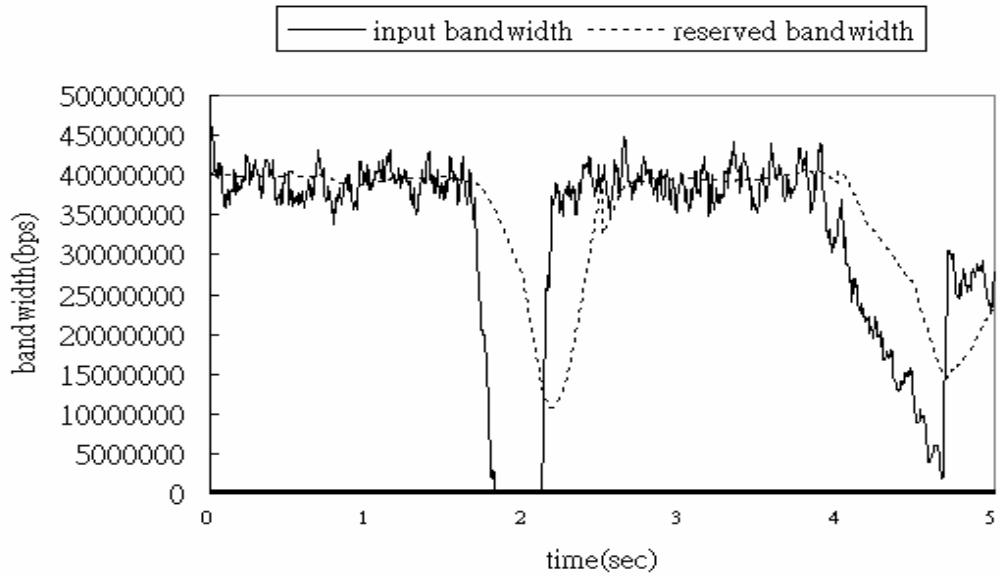


圖 4.11 初始為過度負載下使用 Measure-Based 的方法

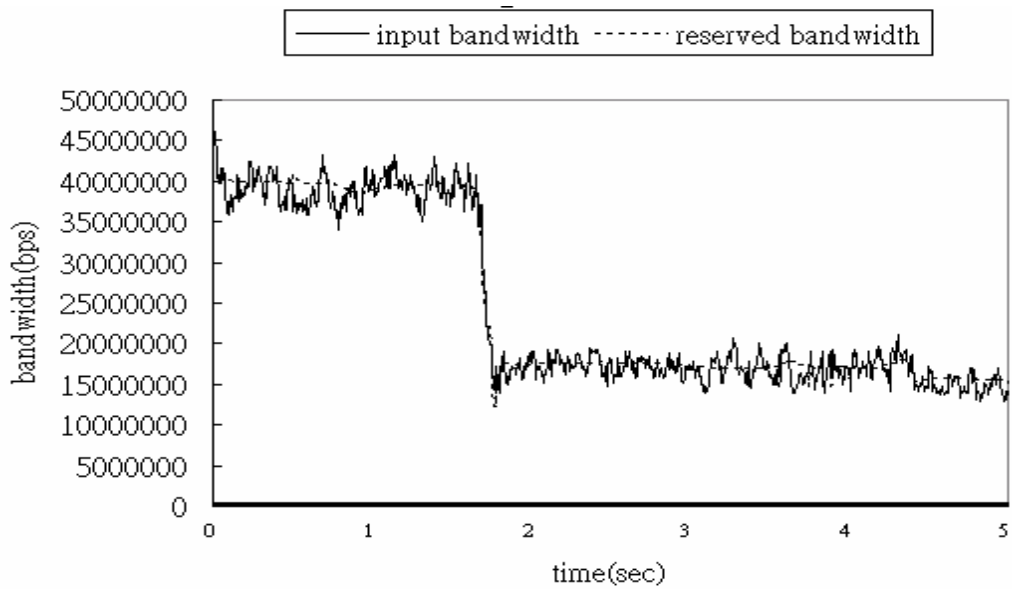


圖 4.12 初始為過度負載下使用 Ratio-Based 的方法



在前面提到，Measure-Based 的估測結果跟  $N$  直接相關，當  $N$  愈大時 Measure-Based 對連線的變化愈遲鈍，而 Ratio-Based 的估測結果不是跟  $N$  直接相關，因此  $N$  加大不會使 Ratio-Based 變得遲鈍。圖 4.11、圖 4.12 裡清楚地顯示了 Measure-Based、Ratio-Based 的保留頻寬和實際輸入速率的關係，很明顯的 Measure-Based 的反應總是較為緩慢，因此當有一段時間為過高負載時，接下來雖然連線中止了但保留的頻寬無法立即降下來，於是只會允許極少的連線建立。相同的，極少的負載過後，又因為反應過慢的緣故接下來又會允許過多的連線建立，所以在過高的連線後只會允許極少的連線建立，而極少的連線後容易允許過高的連線建立，當  $N$  愈大時這種情況愈明顯。但是 Ratio-Based 就沒有這種問題，保留的頻寬能夠快速的反應連線的變化，所以在圖 4.12 裡面當過多的連線中止後，switch 可以立即的回復正常的連線狀態。

## 第五章 總結

Measure-Based 連線允入控制具有簡單、高效率的優點，在原本的概念中它能夠改良過去 Model-Based 連線允入控制效能不佳的缺點，就這一點來看它的方法非常的優秀。Measure-Based 概念是藉由量測過去傳輸流量的特性來估測現有連線對於頻寬的要求為何，所以它的本質是無法去估測那些過去沒有在系統上傳輸過的新連線。這便使得 Measure-Based 在有新連線加入時就會產生問題。

在可見的未來，網路的使用情況想必更為擁塞，數個連線要求同時到達的情況必定會更加常見，如果回頭使用 Model-Based 的方法雖然可以解決細胞遺失率過大的問題保證傳輸品質，但是卻會讓原本就頻寬不足的網路浪費更多頻寬，因此走回頭路並不可行。

在這論文中我們提出了一個新的方法，Ratio-Based 連線允入控制。我們相信只是單純的量測流量的特性並不足夠，重要的應該是要考慮實際的流量和連線一開始所宣告的值兩者之間的關係，把兩者接上關係才能真正的估測到傳輸流量的特性。在這篇論文中，我們從數學分析和模擬分析上證實這個想法是正確的，比照過去 Measure-Based 允入連線控制的方法，我們的方法在靜態連線的情況下會與 Measure-Based 的方法具有同等的效能，而在動態連線的情況下我們的方法能夠克服 Measure-Based 無法立即反應新連線加入的缺點，而在計算量方面我們的方法所必需用到的計算式與 Measure-Based 的計算式相差無幾，每一個過程僅僅多一個乘法或是除法，因此運算的複雜度與 Measure-Based 同等。從以上幾點來看，我們認為 Ratio-Based 連線允入控制比過去的 Model-Based、

Measure-Based 連線允入控制更可以增進網路效能。

Ratio-Based 的基本精神是測量實際傳輸流量與宣告傳輸流量的關係，因此在這篇論文中我們所引用來計算所需頻寬的公式與 Ratio-Based 是可以獨立的。在論文中我們所引用的只是許多種估算需求頻寬的一種，這種估算法因此計算式相當簡單而被採用，但我們相信 Ratio-Based 也能夠適用其他種更複雜，但更準確的估算法。

最後，如我們在第四章內所做的模擬，當 window 的大小改變時所得到的結果也會不同，當 window 拉大時可以得到更多的取樣值，對於生命周期長的連線和生命周期短的連線所得到的結果不一定相同，而不同類型的網路服務所適用的 window 大小也未必一樣，如何能夠依不同類型的網路服務找出最合適的 window 大小，我們認為這是另一個可以研究的方向。

## 參考文獻

- [1]. R. Guerin, H. Ahmadi, and M. Nagshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," IEEE Journal on Selected Areas in Communications, Vol. 9, pp.968-981, September 1991.
- [2]. A. Elwalid, D. Mitra, and R. Wentworth, "A new approach for allocating buffers and bandwidth to heterogeneous regulated traffic in an ATM node," IEEE Journal on Selected Areas in Communications, Vol. 13, pp. 1115-1127, August 1995.
- [3]. Z. Fan and P. Mars, "Fast and accurate estimation of ATM quality-of-service parameters with applications to call admission control," IEEE Proc. Communications, vol. 145, pp.133-140, June 1998.
- [4]. H. Saito and K. Shiomoto, "Dynamic call admission control in ATM networks," IEEE Journal on Selected Areas in Communications, vol. 9, pp.982-989, September 1991.
- [5]. S. Lee and J. Song, "An integrated call admission control in ATM networks," IEICE Transaction on Communication, vol. E82-B, pp. 704-711, May 1999.
- [6]. B. T. Doshi, "Deterministic rule based traffic descriptors for broadband ISDN : worst case behavior and connection acceptance control," Proc. Globecom, vol. 3, pp. 1759-1764, November 1993.
- [7]. O. Ren and G. Ramamurthy, "A real-time dynamic connection admission controller based on traffic modeling, measurement, and fuzzy logic control," IEEE Journal on Selected Areas in Communications, vol. 18, pp. 184-196, February 2000.
- [8]. Y. Kawamura and H. Saito, "VP bandwidth management with dynamic connection admission control in ATM networks," in Local

and Metropolitan Communication Systems, T. Hasegawa *et al.*, Eds. London, U.K.: Chapman & Hall, 1995, vol. 3.

- [9]. H. Kobayashi and Q. Ren, "A diffusion approximation analysis of an ATM statistical multiplexer with multiple types of traffic. - I. Equilibrium stat solutions," IEEE International Conference on Communications, vol. 2, pp. 1047–1053, May 1993.
- [10]. K. Kobayashi, "Steady-state analysis of ATM multiplexer with variable input rate through diffusion approximation," Perform. Eval., vol. 23, no. 2, pp. 163–184, 1995.
- [11]. B. Mark and G. Ramamurthy, "Real-time estimation and dynamic renegotiation of UPC parameters for arbitrary traffic sources in ATM networks," IEEE/ACM Transactions on Networking, vol. 6, issue. 6, pp. 811–827, Dec. 1998.
- [12]. G. Ramamurthy and Q. Ren, "Multi-class connection admission control policy for high speed ATM switches," INFOCOM'97, vol. 3, pp.963-972, Apr. 1997.
- [13]. Q. Ren and H. Kobayashi, "Diffusion approximation modeling for Markov modulated bursty traffic and its applications to bandwidth allocation in ATM networks," IEEE Journal on Selected Areas in Communications., vol. 16, issue 5, pp. 679–691, June 1998.