

國立中山大學電機工程學系

碩士論文



指導教授：李錫智 博士

一個減低 ATM 網路上

連線速率變化的調整-排程機制

A Rate-Based Regulation-Scheduling Scheme  
for Rate-Jitter Control in ATM Networks

研究生 林毅杰 撰

中華民國 九十一年 六月

## 摘要

在 ATM (asynchronous transfer mode) 網路中, Rate-based 調整-排程機制在以控制標定連線的細胞被交換機傳送的速率而達到某種服務品質的保證。只考慮該標定連線本身的調整-排程機制, 可以使標定連線的細胞從調整器佇列被換手到排程器佇列的行為得到良好的控制, 但由於可能有許多連線在同一條件下換手, 而共同競爭其在排程器佇列的位置, 使得標定連線的細胞實際被交換機傳送出去的行爲和被換手的行爲有很大的變形, 而與預期的標定連線細胞被傳送的行爲有相當程度的差異。已有相關的方法被提出, 藉由考慮其他連線的細胞被換手的狀況來降低標定連線的細胞被傳送之速率變化 (rate jitter) 而得到相當好的效果, 但在非工作保持 (non-work-conserving) 的調整-排程機制下, 排程器的狀況的改變除了肇因於由調整器換手過來的細胞外, 還有排程器佇列因為已在調整器等待的細胞皆未合格而被清空造成的閒置狀況, 未考慮此閒置狀況必定造成標定連線的細胞被傳送的行爲與被預期的行爲不必要的差異。我們提出一種新的調整-排程機制, 完整地考慮排程器狀況的改變, 使得標定連線的細胞被傳送的行爲和被換手的行爲之間的差異更小, 使得細胞被傳送的速率更能被掌握, 進而達到更好的服務品質保證。這個機制不論在時間或空間的複雜度都相當的小, 足以符合 ATM 高速網路的要求。

# Abstract

In ATM (asynchronous transfer mode) networks, rate-based regulation-scheduling (R&S) mechanisms guarantee certain quality of service (QoS) with controlling the rate of a tagged connection serviced by the switches. The R&S mechanisms, which consider only the issues of one tagged connection itself but the situation of the scheduling queue, can behave hand-over well. However, there may be many connections hand over in identical condition and compete for the position in the scheduler queue. It makes the transmission to be distorted greatly from the hand-over. There is already a scheme, dynamic R&S, to smooth rate jitter with counting the hand-over of other connections. Dynamic R&S doesn't count the idle slots, which also make the interval between two consecutive tagged cells, and results in unnecessary misses. The new scheme proposed considers the system time and scheduler queue's length. It has less difference between transmission intervals and the premised one of a tagged connection, so that is better in rate-jitter control. This mechanism has  $O(1)$  time-complexity, so conforms to the demand of high-speed ATM networks.

# 目錄

摘要

Abstract

第一章 簡介	1
第二章 已經提出的方法.....	6
第一節 系統架構.....	6
第二節 Static Regulation and Scheduling (Static R&S).....	7
第三節 Dynamic R&S.....	8
第四節 Static/Dynamic R&S 的缺點.....	9
第五節 Modified Dynamic R&S (with Jitter Control).....	11
第三章 我們的新方法：Sum of Time and Queue-Length (TQS)	14
第一節 TQS 機制	14
第二節 關於 (式 11) 中的 “=”	16
第三節 每個時槽中合格檢查的次數	18
第四節 考慮標定連線所分配到的頻寬的改進	
Modified TQS (MTQS)	21
第五節 降低太晚換手的機率的改進	
Conditional Accelerated TQS (CATQS)	24
第四章 數學分析	26
第五章 模擬結果	31
模擬一 每時槽能做的合格檢查次數的影響	33
模擬二 頻寬使用率與暴衝的影響	37
模擬三 MTQS 連線得到的平均頻寬	57
模擬四 CATQS 連線得到的平均頻寬	59
第六章 總結與未來展望	62

# 第一章：簡介

隨著科技的發展，電腦網路的傳輸速率也越來越高，因此除了單純的資料交換以外，多媒體和即時資料在電腦網路上傳輸流量也日益增大，而由於這些資料的即時性，或者是資料傳輸終端的軟硬體的限制，網路的高傳輸率不再是唯一的要求，其服務品質(Quality of Service, QoS) [1]保證也越來越重要。非同步傳輸模式 (asynchronous transfer mode, ATM) [1]架構和技術，符合高速網路網路的要求，而被廣為使用。非同步傳輸模式網路的服務品質保證依傳輸資料的要求而有許多不同項目，除了最基本的吞吐率 (throughput) 保證之外，還包括細胞遺失率 (cell loss rate, CLR)、端對端延遲 (end-to-end delay)、延遲變化 (delay jitter)、速率變化 (rate jitter) …等等。

達成服務品質保證的方法，包括連線允入控制 (Connection Admission Control, CAC)、管制連線使用資源 (Policing)、調整 (regulation or shaping)、和排程 (scheduling) …等等。

連線允入控制主要在考慮交換機的資源，如頻寬 (bandwidth) 及緩衝區容量 (buffer capacity)，保留足以應付原有的連線 (包括達到連線的服務品質要求) 的資源之後，還足夠讓新的連線加入 (亦包括新連線的服務品質要求)，若能，則讓新的連線加入，若不能，就拒絕該連線建立的要求。

管制連線使用資源在監視已建立的連線的實際行為是否符合在要求建立連線時所提供的參數，因為在建立連線時，交換機以細胞發送端所提供的參數做為連線允入控制的依據，若連線實際行為和原來提供的參數不符，可能會佔用交換機過多的資源而影響其他連線所得到的服務品質；若連線有佔用過多資源的情形產生，將會對該連線做出懲罰的動作，如丟棄該連線的細胞，並可能回授一些訊息給連線的發送端告知其應修正其發送細胞的行為。

調整的目的在調整封包被送出的行為，使其不要直接受封包到達的行為影

響，或符合許多連線之間的資源的分配。在電腦網路中，暴衝 (burst) 是一定會發生的，即使產生封包的伺服器以緩慢的速率產生封包，在通過網路中高速交換機的緩衝 (buffering) 之後，可能將相當多的封包集中在一起以高速率送出。著名的典型調整機制之一為漏水桶 (Leaky Bucket) 機制[1]。就像一個底部破了小洞的大水桶，若桶中有水則以穩定的速率洩水，當進水速率比洩水速率大時，水桶中的水位不斷升高，到水桶滿時再進來的水會溢出。在漏水桶機制中，封包就好比水，緩衝區的容量就是水桶的容積，調整器在緩衝區內有封包時，將封包以穩定的速率送出，此機制可避免該連線的流量暴衝，且在封包來到該交換機的暴衝持續時間過長時，有限的緩衝區使該連線的封包被丟棄，因此亦可達到管制連線使用資源的效果。

排程的目的在解決通過同一交換機各連線競爭的仲裁，仲裁的方式與優先權 (priority) 有關。靜態優先權 (static priority) 仲裁方式是將各連線事先標定好優先權的順序，當兩個以上連線同時要求服務時，就以事先標定好的優先權順序決定各連線被服務的順序。動態優先權 (dynamic priority) 的優先權順序隨一些即時的狀態而改變，例如讓 CLR 最高的連線擁有最高的優先權。也有無優先權考慮的排程機制，如最簡單的先到先服務。

相關的調整-排程演算法，主要有 frame-based 及 rate-based 兩種。frame-based 下的 stop and go queueing (SGQ)[2]、delayed frame queueing (DFQ)[3]、及 multilayer gated frame queueing (MGFQ)[4]…等等。frame-based 的演算法的特色是將交換機的頻寬在時間上做分段 (framing)，將必須在該段時間內傳送出去的細胞做排程，若在該段時間內不能被傳送出去的細胞即丟棄，藉以達到單節點的延遲保證，並有些演算法 (如 DFQ 及 MGFQ) 藉著在節點之間交換細胞的延遲訊息，而達到端對端 (end-to-end) 的延遲保證。

我們將 rate-based 的演算法分成幾類，第一類有 packet-by-packet generalized processor sharing (PGPS)[5]、delay-earliest due date (D-EDD)[6]、及 jitter-earliest due date (J-EDD)[7]…等等，此類演算法利用 earliest due date (EDD)[8] 的觀念，比較

細胞該被傳送出去的時間底限 (deadline) 之先後而做排程，細胞該被傳送出去的時間之計算由演算法著眼的服務項目而定。PGPS 以頻寬分配為主要目的，故以該連線的頭端 (head of line，或 HOL)細胞大小除以該連線分配到的頻寬計算該封包該被傳送出去的時間；D-EDD 以各封包的被指定的時間底限做先後排程，但在封包被服務的時間有重疊時，要將該被先傳送的細胞的時間底限調整為較早的值；J-EDD 配合 D-EDD，利用下游節點將細胞緩衝，使連線的延遲變化在限定範圍之內。

rate-based 演算法的第二類與 round-robin 有關，有 hierarchical round-robin (HRR)[9]、weighted round-robin (WRR)[10]、deficit round-robin (DRR)[11]、及 interleaved round-robin (IRR)[12]…等等。HRR 將交換機的頻寬分段，分段中的頻寬分配給不同等級的連線，在同等級中的排程以 round-robin 的方式達成，在各等級之間以等級高低順序完成傳送。WRR 依各連線的權重 (weight) 決定在一個循環 (cycle) 是否該被詢訪 (visit)，一個循環中每個連線最多被詢訪一次，權重較高者在往後的較多循環中有被詢訪的機會，權重較低者則在往後的較多循環中有被詢訪的機會，WRR 的權重未正規化 (un-normalized)。HRR 與 WRR 的差異在 HRR 對各等級的頻寬分配是固定的，同等級中的各連線頻寬分配也是固定的，所以各連線的頻寬並不會互相影響，若連線在分配到的頻寬中未有足夠多的細胞讓系統傳送，系統即在應為該連線傳送細胞時間閒置；而 WRR 中各連線所得到的頻寬是該連線的權重佔所有連線的權重總和的比例，以及連線佇列的狀況，故各連線的頻寬會互相影響，且系統為工作保持 (work-conserving)。DRR 在解決 round-robin 中因為封包大小不一而產生的頻寬分配不均的狀況，在每回合 (round)分配給各連線固定的額度 (quantum)，該連線在該回合可將該額度用到不能用 (如接下來的封包長度大於剩下的額度)，即詢訪下一個連線，而各連線在該回合未用完的額度可留到下個回合使用。IRR 將交換機頻寬分段，在每分段中依各連線分配到的頻寬將細胞平均安排在該分段中的各時間點，當各連線的細胞被服務的時間有重疊時，依其被安排的時間點先後決定被服務的順序。

在 rate-based 演算法中有部分較難歸類者，我們參考到的其他的演算法有 credit-based fair queueing (CBFQ)[13] 及 rotating-priority-queues (RPQ)[14]。CBFQ 在每個單位時間內依各連線其分配到的頻寬，給各連線相當的額度，與 DRR 分配額度的方式類似，只是 CBFQ 的單位時間是以傳送封包的單位時間（如位元時間或位元組時間）。CBFQ 以各連線的「頭端細胞長度減去已得的額度」除以該連線分配到的頻寬，其值最小者被服務。RPQ 將各連線到達的細胞緩衝 (buffering) 在不同的佇列，在一週期性的時間，將所有佇列旋轉，同一佇列中因為旋轉的關係將緩衝來自不同連線的細胞，由於每個佇列的長度固定且相同，故可達到單節點的延遲保證。

在 rate-based 演算法中的第三類，也是我們提出的演算法所屬於的，是直接控制連線的細胞被傳送的速率，換句話說就是控制兩個細胞被傳送出去的時間間隔。此類演算法是使標定連線 (tagged connection) 的細胞與細胞之間被交換機傳送的時間盡量接近一個靜態的值。最直觀的方法就是 Static Regulation and Scheduling (Static R&S) [15]，調整器在固定的時間間隔就將一個細胞換手到排程器；其他著名的方法如 Rate Controlled Static Priority (RCSP) [16]，Dynamic Regulation and Scheduling (Dynamic R&S) [15]……等等。RCSP 控制細胞進入排程器的行為，使細胞與細胞之間被換手的間隔在限定的範圍之內。Dynamic R&S 利用偵測其它連線換手的細胞數，判斷是否與上一個細胞之間已有足夠的間隔而決定換手與否。

在 rate-based 中的第三類調整-排程演算法中，RCSP 及 Static-D&S 由於缺乏由排程器回授的資訊，所以其控制的只是標定連線由調整器換手至排程器的行為，而實際細胞被傳送的行為卻沒有被適當地控制。Dynamic R&S 雖在判斷細胞合格與否的條件有考慮排程器的狀況，但由於只考慮了其它連線換手的細胞，未將排程器閒置的狀況也考慮進去；排程器的閒置也佔據時槽而影響同一標定連線的細胞被傳送的間隔，故 Dynamic R&S 使得細胞被傳送的間隔有一些不必要的失誤。我們提出一個新的演算法，同時考慮其他連線換手的細胞及排程器閒置



的狀況，而改進了 Dynamic R&S 中不必要的失誤，使該連線被傳送出去的速率更為穩定，比其他的演算法更接近我們想要的傳送速率。

本文之後的架構如下：第二章回顧以前已經提出的方法，第三章是我們提出的新方法，並和已提出的調整-排程演算法做進一步的討論，第四章用數學分析我們的方法如何優於以前的方法，第五章提供我們做模擬所得的數據並加以討論，第六章對本文做總結。



## 第二章：已經提出的方法

### 第一節：系統架構

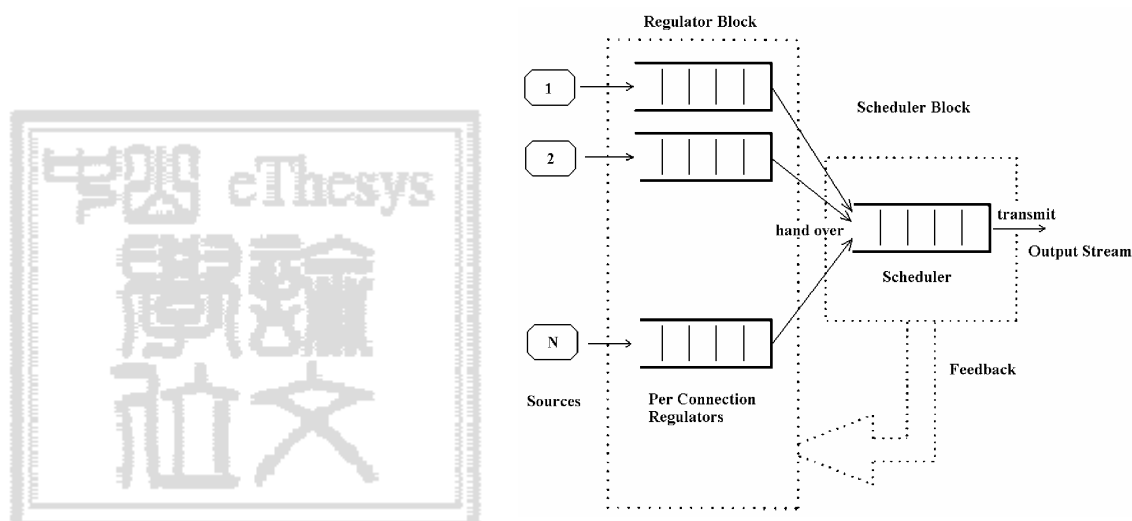


圖 1：系統架構

圖 1 為本文中討論的交換機系統架構。交換機有兩個主要的區塊，一為調整器 (regulator)，一為排程器 (scheduler)。每個細胞到達交換機時都先進入調整器佇列中，在符合調整器的調整機制後，送入排程器佇列中等待被傳送。排程器依先到先服務(First Come First Serve, FCFS) 的方式將其佇列中的細胞依其到達排程器的順序傳送到下一個交換機。排程器依各機制的設計不同，而可能對調整器回授某些必要的資訊。

每個連線在調整器有一個專用的佇列，每個細胞到達交換機都被放在調整器的佇列中，直到其合格 (eligible) 為止。一個合格的細胞會被換手 (handover) 到排程器，依先到先服務 (FCFS) 的次序被傳送出去 (transmit)。

以下的討論，關於時間都以時槽 (time-slot) 為單位，交換機的頻寬為每個時槽傳送一個細胞；關於佇列長度都以槽 (slot) 為單位，一個槽可儲存一個細

胞。若一個連線欲以某種機制來控制其各細胞被傳送的時間，我們稱該連線為**標定連線** (tagged connection)。在此文中，沒有上下標的  $t$  是指系統時間，沒有上下標的  $q$  是指排程器的佇列長度。 $t_k^r$  為連線  $r$  的第  $k$  個細胞被換手時的系統時間， $q_k^r$  為連線  $r$  的第  $k$  個細胞被換手前的  $q$  值，即該細胞被換手至排程器的第  $q_k^r + 1$  個槽。由於  $t$  和  $q$  分別以時槽及槽為單位，故兩項可疊加(superposition)。 $T^r$  表示標定連線  $r$  所預先設定好的  $T$  值，即我們期望標定連線  $r$  的細胞被交換機傳送的時間間隔。 $C_k^r$  表示標定連線  $r$  的第  $k$  個細胞。

實際上交換機的排程器與調整器，細胞的接收與傳送都可以平行運作。每個時槽中交換機需要完成以下動作：1. 傳送細胞，2. 將合格細胞換手，3. 接收細胞。在每個時槽，交換機須查看排程器佇列是否有細胞等待被傳送，若有則將排程器佇列的頭端細胞 (Head of Line, 或 HOL) 傳送出去。同時，各連線調整器中合格 (eligible) 的細胞換手至排程器佇列中。進入交換機的細胞則被存放在該連線的調整器佇列中。

## 第二節：Static Regulation and Scheduling (Static R&S)

Static regulation and scheduling (Static R&S)中，標定連線  $r$  的調整機制是每隔  $T^r$  個時槽，將調整器佇列的頭端細胞換手至排程器，排程器並沒有任何回授資訊給調整器。我們以  $ET_k^r$  表示  $C_k^r$  合格的時間，則在 Static R&S 中， $ET_k^r$  定義如下：

$$ET_1^r = AT_1^r. \quad (式 1)$$

$$ET_k^r = \max(ET_{k-1}^r + T^r, AT_k^r) \quad k > 1. \quad (式 2)$$

其中  $AT_k^r$  是  $C_k^r$  到達交換機的時間。為了方便討論，我們忽略細胞在網路上被丟棄的狀況，也就是標定連線  $r$  到達交換機的第  $k$  個細胞，就是標定連線  $r$  的第  $k$  個細胞。(式 1) 的涵義是當  $C_1^r$  到達時，該細胞立即合格；若  $C_k^r$  的  $k \neq 1$ ，則其合格時間為  $C_{k-1}^r$  的合格時間加上  $T^r$ ，但若這個合格時間已到，該細胞仍未到達該交換機，則該細胞在到達時立即合格。

### 第三節：Dynamic R&S

在 Static R&S 中，兩個細胞被傳送的間隔大於  $T^r$  的其中一種情形是：當  $C_k^r$  被換手之後，其他連線在  $T^r - 1$  個時槽內，有超過  $T^r - 1$  個細胞被換手至排程器。Dynamic R&S 即針對 Static R&S 的此項缺點改良而來。除了每隔  $T^r$  個時槽將標定連線  $r$  的頭端細胞換手之外，為了減少兩個細胞之間實際被傳送出去的時間大於  $T^r$  個時槽的情形，在  $C_{k-1}^r$  被換手之後，若其他連線換手的細胞個數大於  $T^r - 2$  時，若  $C_k^r$  已在調整器佇列中，就將  $C_k^r$  換手至排程器。

以  $ET_{k-1}^r + B_{k-1}^r$  表示在  $ET_{k-1}^r$  之後，其他連線 (除了標定連線  $r$  之外的所有連線) 換手到排程器的細胞數大於  $T^r - 2$  個的時間，則  $ET_k^r$  在 Dynamic R&S 定義如下：

$$ET_1^r = AT_1^r. \quad (\text{與式 1 同})$$

$$ET_k^r = \max(ET_{k-1}^r + \min(T^r, B_{k-1}^r), AT_k^r) \quad k > 1. \quad (\text{式 3})$$

也就是說，Dynamic R&S 一樣是每隔  $T^r$  個時槽換手一個標定連線  $r$  的細胞到排程

器，但 Dynamic R&S 另外加上了一個機制，就是在將  $C_{k-1}^r$  換手之後，若其他的連線換手的細胞總數超過  $T^r - 2$  個，即使距  $C_{k-1}^r$  被換手還不到  $T^r$  個時槽，也要將  $C_k^r$  換手至排程器。Dynamic R&S 以 (式 1) 與 (式 3) 定義  $ET_k^r$ ，在細節上與上一段的文字描述有一點不同。上一段文字是說在  $C_{k-1}^r$  被換手之後...，但 (式 1) 與 (式 3) 的含義是  $C_{k-1}^r$  被換手之後的“時槽”之後...。因為在  $ET_k^r$  的時槽中，被換手的細胞，除了  $C_{k-1}^r$  之外，可能還有來自其他連線的其他細胞；若標定連線  $r$  的細胞總是會在排程器中排在這些同時被換手的眾多細胞的最後面一個，那麼就和 (式 1) 與 (式 3) 吻合，否則，因為同時換手的細胞數無法確定，且  $C_{k-1}^r$  會被排在同時換手的眾多細胞的第幾個也不確定，和上一段的文字描述就會有差異。由於該篇文章並沒有界定得十分清楚，故依其模擬結果以及合理性推測，應是文字描述的內容較為正確，但不論是哪一個較正確，兩者的概念都是相同的。

Dynamic R&S 在有  $N$  個標定連線的情形下，需要維護  $2N$  個計數器，其中  $N$  個記錄  $ET_{k-1}^r + T^r$  使  $C_{k-1}^r$  與  $C_k^r$  被換手的時間間隔不大於  $T^r$  個時槽，另外  $N$  個記錄自  $C_{k-1}^r$  換手後，其他連線換手至排程器的細胞數，使能察覺其他的連線換手的細胞總數超過  $T^r - 2$  個，將  $C_k^r$  換手。本段中  $r = 1, 2, \dots, N$ 。

#### 第四節：Static/Dynamic R&S 的缺點

我們的控制目的是要使標定連線  $r$  的細胞在某一交換機被傳送的時間為預先設定的靜態值  $T^r$ ，由於交換機需要同時處理許多連線，在只有一個排程器的架構下，無論控制方法如何，都不可能保證每個標定細胞被傳送的時間一定是  $T^r$ ，請見圖 2。

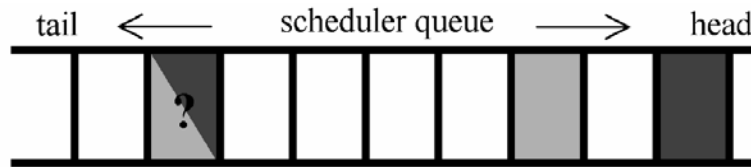


圖 2：控制細胞被傳送的時間無法解決的問題。

深色的細胞來自  $T=7$  的標定連線，

淺色的細胞來自  $T=5$  的標定連線。

從圖 2 可以看出，我們不可能總是同時滿足兩個以上的連線。然而，Static/Dynamic R&S 卻因為本身機制上的缺點，造成了不必要的失誤。我們將失誤分成兩類，一是**太早換手**，二是**太晚換手**。Dynamic R&S 已對 Static R&S 太晚換手的部分做了改進，故我們略過 Static R&S 太晚換手的部分，直接對 Dynamic R&S 的太晚換手部分作討論；而這兩種演算法太早換手的情形是完全一樣的。舉例說明這兩種失誤如圖 3 及圖 4。

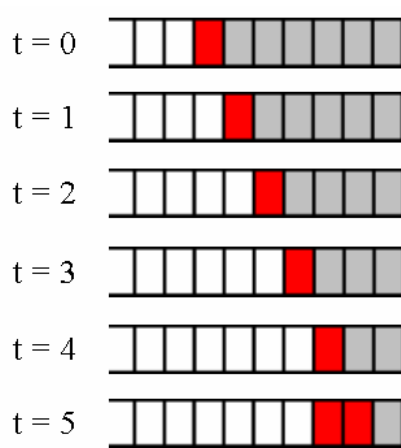


圖 3：Dynamic R&S 太早換手一例

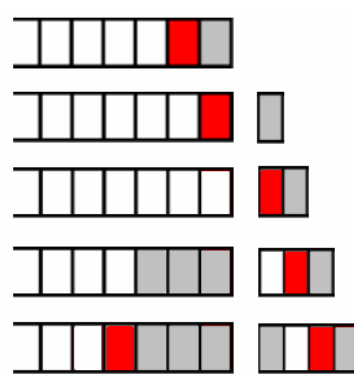


圖 4：Dynamic R&S 太晚換手一例

圖 3 及圖 4 分別表示排程器佇列兩個不同的狀況，深色方塊為來自同一個標定連線的標定細胞，淺色方塊為來自任何其他連線的細胞。預定的標定細胞傳送

間隔  $T = 5$ 。在圖 3 的例中，我們假設在標定細胞換手後，就沒有其他細胞被換手，如此一來，調整器在 5 個時槽後，即換手下一個標定細胞，如此一來，雖然標定細胞被換手的時間間隔是 5，但兩個細胞的被傳送的時間將會是 1 而不是 5，這一點 Static R&S 和 Dynamic R&S 是相同的；在圖 4 的例中，右側為被傳送的細胞，空白的表示交換機在該時槽閒置，雖然沒有傳送細胞，但仍是佔據了一個時槽，在例中若在  $t = 3$  時換手即能使細胞傳送間隔為 5，而 Dynamic R&S 卻沒有能偵測到交換機的閒置而錯失了讓細胞被傳送的時間符合預定值 5 的機會，讓其他連線有更多的機會換手更多的細胞，而使細胞傳送間隔變成 6 或更大，Static R&S 則是不論其它連線換手了多少個細胞到排程器，都是在  $t = 5$  的時候為標定連線作第二次換手。

## 第五節：Modified Dynamic R&S (with Jitter Control)

本實驗室上一屆畢業的石翔任學長，提出了兩個方法，分別為 Modified Dynamic R&S (MDRS) [17] 和 Modified Dynamic R&S with Jitter Control (MDRSJ)[17]，主要在改善 Dynamic R&S 的吞吐率而犧牲了原來 Dynamic R&S 對連線傳送速率變化控制，與本文目標不同，但其概念及演算法與本文有相近之處，故提出討論。

MDRS 中， $C_k^r$  合格的條件如下：

$$t_i^r = 0 \quad \text{或} \quad (式 4)$$

$$q \geq q_{k-1}^r + t_i^r \quad (式 5)$$

其中  $t_i^r$  的維護方式如下：

$$t_i^r \leftarrow T^r \quad \text{當 } C_{k-1}^r \text{ 被換手時} \quad (\text{式 } 6)$$

$$t_i^r \leftarrow t_i^r - 1 \quad \text{若 排程器在該時槽未閒置} \quad (\text{式 } 7)$$

$$t_i^r \leftarrow t_i^r - \min_r(t_i^r) \quad \text{若 排程器在該時槽閒置} \quad (\text{式 } 8)$$

$t_i$  是用來計數距  $C_k^r$  被換手之還要等幾個時槽，由 (式 6) 與 (式 7) 共同達成 MDRS 下標定連線  $r$  的兩個細胞被換手的間隔不得大於  $T^r$  的機制，所以在太早換手的部分和 Static/Dynamic R&S 有著相同的問題，而且在 (式 8) 中更加劇了太早換手的狀況，這並不算是缺點，因為 MDRS 主要著眼在吞吐率而不是速率的穩定。在太晚換手的部分，在 Dynamic R&S 中可看出，Dynamic R&S 造成不必要的太晚換手，是因為忽略了排程器閒置所佔據的時槽，MDRS 雖然考慮了排程器閒置的狀況，但由於 MDRS 在排程器閒置時對各連線的合格條件作了變動，將所有的  $t_i^r$  向前平移了  $\min_r(t_i^r)$ ，而使得排程器在閒置一個時槽後必定有細胞被換手（當然必須至少有一個調整器佇列裡有細胞等待被換手），而不會連續閒置兩個時槽，而失去了穩定速率的效果。

MDRS 在有  $N$  個標定連線的情形下，需要維護  $2N$  個計數器，其中  $N$  個記錄  $t_i^r$  使  $C_{k-1}^r$  與  $C_k^r$  被換手的時間間隔不大於  $T^r$  個時槽，另外  $N$  個記錄自  $q_{k-1}^r$ ， $r = 1, 2, \dots, N$ 。

在 MDRSJ 中，限制 (式 8) 對  $t_i^r$  的作用，在  $t_i^r$  未被 (式 6) 更新之前， $t_i^r$  因為 (式 8) 而減少的值不得超過一個預先設定的靜態值  $J^r$ （當然  $J^r < T^r$  才有意義），然而，這導致 MDRSJ 成爲一個無法穩定地持續工作的演算法。請注意在排程器閒置時，(式 7) 是不被作用的，而 MDRSJ 對 (式 8) 的作用的限制，使得排程器閒置時，不如 MDRS 能讓至少一個連線合格。很容易地可以想見，若該



交換機只處理一個 MDRSJ 的連線，其  $T^r > 1$ ，且完全沒有其他連線，那麼該標定連線只會被換手第一個細胞後，因為沒有來自其他連線的細胞，所以該連線永遠不會再合格，第二個細胞永遠送不出去。在有多個連線的狀況，且**每個連線都執行 MDRSJ 演算法**的話，當排程器佇列被清空時，若每個連線都  $J^r < t_i^r$ 。那麼就沒有連線合格，就沒有細胞換手，排程器的閒置就一直持續下去，最後導致系統掛掉。石的論文中的模擬沒有發生這個狀況是因為有  $T^r = 1$  的連線參與競爭，這樣的連線在  $C_{k-1}^r$  被換手後  $C_k^r$  同時就合格了，故當排程器持續閒置的情形發生時（此時  $T^r = 1$  的連線之調整器佇列必然都是空的），只要有  $T^r = 1$  的連線又有細胞到達交換機，即可再啓動（式 7）的作用。Dynamic R&S 沒有這個問題是因為其保留了 Static R&S 中以固定時間間隔  $T^r$  為最遲的換手底限。

### 第三章：我們的新方法：

## Sum of Time and Queue-Length (TQS)

### 第一節：TQS 機制

要使  $C_{k-1}^r$  與  $C_k^r$  被傳送の間距為  $T^r$ ，就要使  $C_k^r$  在  $t = t_{k-1}^r + T^r - i$  時，被放在排程器佇列的第  $q_{k-1}^r + 1 + i$  個槽中，其中  $i$  為整數；若此換手的條件達成時  $C_k^r$  尚未到達該交換機，則當其到達時，立刻換手至排程器。這個觀念在石學長的論文中有提及，但在上一章已陳述，其演算法因 (式 8) 在排程器閒置時對各標定連線的合格條件作了平移動作，且由 (式 6) 與 (式 7) 看出其仍保留了兩細胞被換手的時間間隔不得大於  $T^r$  的機制，故未能表現出速率穩定的效果。上述內容可以 (式 9) 表示：

$$q_k^r + t_k^r = q_{k-1}^r + t_{k-1}^r + T^r \quad (\text{式 9})$$

若 (式 9) 狀況達成，則  $C_{k-1}^r$  與  $C_k^r$  被傳送の間隔就會是  $T^r$ 。故我們將  $C_k^r$  合格的條件  $TQ_k^r$  定為：

$$TQ_k^r = q_{k-1}^r + t_{k-1}^r + T^r \quad (\text{式 10})$$

故  $C_k^r$  合格的時機就是當

$$t + q = TQ_k^r \quad (\text{式 11})$$

成立時。而若當 (式 11) 成立時， $C_k^r$  尚未到達該交換機，或  $k = 1$  的情形下，則該細胞到達交換機時即合格。 $TQ_k^r$  由調整器來維護， $TQ_k^r$  代表的就是  $C_k^r$  合格時「系統時間及排程器佇列的長度的和」。調整器須為每個連線  $r$  維護一個值來記住  $TQ_k^r$ ，不過調整器並不需要記住該連線已經換手了幾個細胞，只需要記錄每個連線的下一個細胞該在什麼樣的狀況合格即可，故嚴格說起來調整器維護的值是  $TQ^r$  (把下標  $k$  拿掉)，但為避免符號太多，以下的討論仍以  $TQ_k^r$  表示之。

在此我們必須強調一點， $TQ_k^r$  是以 (式 10) 計算出來，是我們“希望”  $C_k^r$  被換手時的「系統時間與排程器佇列長度的和」，而  $q_k^r + t_k^r$  則是恰在  $C_k^r$  被換手之前，“實際的”「系統時間與排程器佇列長度的和」，由於不同連線可能以同樣的  $TQ_k^r$  值在等待將其調整器佇列的頭端細胞換手，而實際被換手後在排程器佇列的位置必定有先後，故  $TQ_k^r$  不一定能等於  $q_k^r + t_k^r$ 。舉例如圖 5：

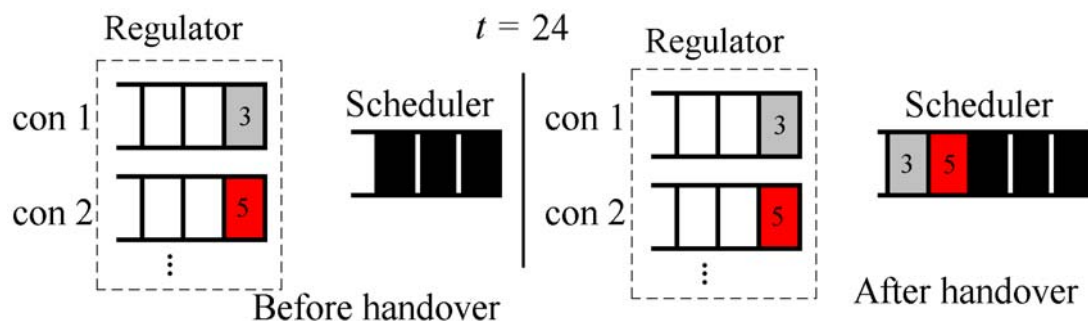


圖 5：  $TQ_k^r$  和  $q_k^r + t_k^r$  的關係

圖 5 中，標定連線 1 的細胞以淺色方塊表示，標定連線 2 的細胞以深色方塊表示，其他在換手前就已經在排程器佇列的細胞以黑色表示。在  $t = 24$  時，標定連線 1

和標定連線 2 在換手前，其調整器佇列的頭端細胞分別為該連線的第 3 個及第 5 個細胞，且  $TQ_3^1 = TQ_5^2 = 27$ ，在換手後， $t_3^1 + q_3^1 = 24 + 4 = 28$ ，且  $t_5^2 + q_5^2 = 24 + 3 = 27$ ，由此可看出

$$TQ_k^r \leq q_k^r + t_k^r \quad (\text{式 12})$$

也就是說，這個演算法完全沒有太早換手的失誤，太晚換手的失誤是因為  $C_k^r$  在剛好會和  $C_{k-1}^r$  間隔為  $T^r$  時加入與其他連線的競爭，而在競爭中未能成為最勝者而造成。由於此演算法如 (式 11) 中所示，以  $t + q$  的和為判斷標定細胞合格的依據，我們稱之為 **Sum of Time and Queue-Length** 演算法，簡稱為 **TQS**。

我們仍以圖 3 及圖 4 為例說明 TQS 不會有 Dynamic R&S 中不必要的失誤。在圖 3 中， $t_{k-1}^r = 0$  且  $q_{k-1}^r = 6$ ，故由 (式 10) 得  $TQ_k^r = 0 + 6 + 5 = 11$ ，在  $t = 0 \sim 5$  之間，隨著時間的前進， $t$  在每個時槽遞增 1，而  $q$  在每個時槽遞減 1，所以  $t + q$  值一直維持為 7，所以  $C_k^r$  不會在  $t = 5$  時被換手造成和  $C_{k-1}^r$  緊緊相連的狀況。

以圖 4 為例， $t_{k-1}^r = 0$  且  $q_{k-1}^r = 1$ ，故  $t_{k-1}^r + q_{k-1}^r = 1$  且  $TQ_k^r = 6$ 。在  $t = 0 \sim 2$  之間，同樣的隨著時間的前進， $t$  在每個時槽遞增 1，而  $q$  在每個時槽遞減 1，所以  $t + q$  值一直維持為 2。TQS 與 Dynamic R&S 不同之處發生在  $t = 3$  時，在  $t = 1 \sim 2$  之間再也沒有其他連線換手，造成了  $t = 3$  時排程器閒置，使得  $t$  遞增 1，但  $q$  值卻仍然和  $t = 2$  時一樣是 0，因此  $t + q$  值增加了。在  $t = 3$  其他連線換手之後， $t + q = 3 + 3 = 6 = TQ_k^r$ ，使  $C_k^r$  合格，而不會錯失正確的換手機會。

## 第二節：關於 (式 11) 中的 “=”

我們在 (式 11) 中使用 “=” ，而不若 Static/Dynamic R&S 或 MDRS 使用 “≥” 。這表示我們對所有的  $t + q$  值都必須檢查。當調整器為各連線維護的  $TQ_k^r$  有兩個以上相同時，這些連線同時換手其調整器佇列中的頭端細胞，而使得  $t + q$  值一次增加許多而不是 1，TQS 須逐一檢查每一個  $t + q$  值；再以圖 5 舉例來說，在系統時間為 24 時，排程器佇列為 3， $t + q = 24 + 3 = 27$ ，若此時連線 1 及連線 2 的  $TQ_k^r$  值皆為 27，調整器以  $t + q = 27$  代入 (式 11) 供各連線作合格檢查，此狀況下連線 1 及連線 2 一起換手，使得  $t + q = 24 + (3 + 2) = 29$ ，但在 TQS 下，下一次的檢查不能直接用  $t + q = 29$  代入 (式 11)，而是調整器仍先以  $t + q = 28$  代入供各連線作合格檢查，完畢後再以  $t + q = 29$  代入供各連線作合格檢查。

由上一段可以看出，其實 TQS 代入 (式 11) 供各連線作合格檢查的值，是一個以 1 遞增的值，在此設為  $z$ ，此值並非即時的  $t + q$  值，而即時的  $t + q$  值是一個上限值，如上例  $t + q = 24 + 5 = 29$  的狀況，調整器代入 (式 11) 的值不能一直遞增到 30 或更大，除非在以  $z = 28$  或  $z = 29$  作合格檢查時有更多的細胞被換手增加了  $q$  值。我們將 (式 11) 改寫為

$$z = TQ_k^r \quad (式 13)$$

而  $z$  就是上述「一個以 1 遞增且以  $t + q$  值為上限的值」。

既然每個時槽需要做的合格檢查次數可能不只一次，這就牽涉到另一個問題，一個時槽裡到底要作幾次合格檢查 (即一個時槽中要代入幾個不同的  $z$  值到 (式 13) 供各連線做合格檢查)? 這個問題在下一節中討論。

TQS 在有  $N$  個標定連線的情形下，調整器需要維護  $N + 1$  個計數器，其中  $N$  個為每個連線記錄  $TQ_k^r$ ， $r = 1, 2, \dots, N$ ，另一個記錄  $z$  值。

### 第三節：每個時槽中合格檢查的次數

所有將排程器佇列的狀況列入合格檢查條件的排程演算法，包括 Dynamic R&S、MDRS、和 TQS，都必定遭遇每個時槽中該做或可以做多少次合格檢查的問題，每個時槽中能做多少次合格檢查，和交換機內部的處理能力有直接的關係。在某個時槽中，一但有細胞換手，不論是一個或是多個細胞，排程器佇列的狀況都會改變，而可能需要以新的排程器的狀況再做一次合格檢查。除了排程器佇列是空的狀況下，換手的細胞可能會是排程器佇列的頭端細胞，而在下一個時槽開始時立刻被傳送，其他狀況下，換手只是讓細胞換個地方等待而已，早檢查或晚檢查未必會影響交換機傳送細胞的行為，然而，在 Dynamic R&S 和 TQS 下卻明顯地會受影響。Dynamic R&S 在 (式 3) 的狀況達成時，即  $t = ET_k^r$  時，若  $C_k^r$  仍未到達調整器佇列等待被換手 (即連線  $r$  的調整器佇列是空的)，則  $C_k^r$  到達交換機時立即就合格，TQS 在 (式 11) 達成時亦是同樣的狀況，我們稱這樣的細胞  $C_k^r$  為一個**新到細胞**。若交換機內部處理能力無限大 (快)，或者說其內部處理能力遠大於傳送細胞的能力，即交換機在每個時槽能做的合格檢查次數沒有上限，**總是能在該時槽結束前，確認在最新排程器狀況下沒有細胞合格**，那麼，在此時槽尚未到達交換機的細胞很顯然地不可能與此時槽合格的細胞競爭在排程器中的位置；然而若每個時槽能做的合格檢查次數有限，原本該在某個時槽合格的細胞可能必須等到較遲的時槽才被換手到排程器佇列，那麼在這兩個時槽 (可能是連續的兩個時槽，也可能不是) 之間若有有新到細胞到達交換機，此新到細胞就會與原本早該合格且被換手的細胞競爭在排程器佇列中的位置，這個情形在內部處理能力無限大時是不會發生的。更多的競爭必定造成交換機傳送標定連線的細胞之行為有更多的失誤。MDRS 在演算法中明定每個時槽只做一次合格檢查故沒有這個問題，但由以上討論我們可知若 MDRS 在每個時槽做更多次的合格檢查，

效能應該會更好。

Dynamic R&S 和 TQS 在每個時槽該做多少次合格檢查才能「總是」確認在該時槽的最新排程器佇列狀況下沒有細胞合格？在  $\sum_r \frac{1}{T^r} > 1$  的狀況下，這個答案就是該交換機的全部調整器佇列可容納的細胞數，因為被換手一個細胞就造成的合格細胞數平均為  $\sum_r \frac{1}{T^r} > 1$ ，再將合格的細胞換手又造成更多的細胞合格，在本文假設交換機的緩衝區無限大的狀況下，將連線  $r$  的細胞來源端平均速率寫做  $\lambda^r$ ，由於系統穩定的關係，必須  $\sum_r \lambda^r < 1$ ，所以不會每個調整器佇列都總是有細胞等待被換手，故所需的合格檢查次數仍然是有限的，但和細胞來源端的行為有關。

至此介紹完了 TQS 的細節，在圖 6 寫出 TQS 的詳細演算法。在各連線之間，其合格檢查是沒有任何交互關係的，可以平行處理，在圖 6 中限於表達方式，故寫成依連線的編號順序做檢查。

常數：

$T^r$ ：第  $r$  個連線所希望其兩個細胞之間被傳送的時間。

$P$ ：每個時槽可做合格檢查的次數上限。

變數：

$t$ ：系統時間，以時槽為單位

$q$ ：排程器佇列長度(佇列內的細胞數)，以槽為單位。

$N$ ：已建立的標定連線數，各連線分別被標為連線 1、連線 2、...、連線  $N$ 。

$TQ^r$ ：標定連線  $r$  下一次換手時機的  $t + q$  值。

$z$ ：用來與  $TQ^r$  比對判斷標定連線  $r$  的換手時機是否已成立。

$i$ ：用來計數該時槽已經做過幾次合格檢查。

初始： $t \leftarrow 0$ ； $q \leftarrow 0$ ； $z = 0$ ； $TQ^r \leftarrow null$ ；

排程器：

在每個時槽，若佇列中有細胞，將頭端細胞傳送出去；/\*  $q \leftarrow q - 1$  \*/

調整器：每個時槽執行以下動作

$i \leftarrow 0$ ；

**while** ( $z \leq t + q$  **and**  $i < P$ )

{

$r \leftarrow 1$ ；

**while** ( $r \leq N$ )

    {

**if** ( $z = TQ^r$  **or**  $TQ^r = null$ )

        {

**if** ( 標定連線  $r$  的調整器佇列中有細胞 )

            {

$TQ^r \leftarrow t + q + T^r$ ；

                將調整器  $r$  的頭端細胞換手；/\*  $q \leftarrow q + 1$  \*/

            }

**else**

$TQ^r \leftarrow null$ ；

        }

$r \leftarrow r + 1$ ；

    }

$z \leftarrow z + 1$ ；

$i \leftarrow i + 1$ ；

}

---

圖 6：TQS 演算法



## 第四節：考慮標定連線所分配到的頻寬的改進

### Modified TQS (MTQS)

由 (式 12) 得知，在 TQS 下，標定連線  $r$  的細胞被傳送的時間大於等於  $T^r$ ，故其得到的頻寬勢必小於  $1/T^r$ ；另一個更直觀的想法是，TQS 完全沒有太早換手的失誤，卻有因為其他連線以同樣的  $TQ_k^r$  等待被換手而因為競爭未得到最勝而造成太晚換手的失誤，很明顯地標定連線  $r$  所得到的頻寬勢必小於  $1/T^r$ 。雖然頻寬保證是另外一個主題，但是希望一個連線的細胞被傳送的時間接近  $T^r$ ，其中可能也暗示了對該連線得到的頻寬能接近  $1/T^r$  的期待，只要小小的修正即可達到此要求。

若我們將 (式 10) 改爲

$$TQ_k^r = TQ_{k-1}^r + T^r \quad (\text{式 14})$$

也就是說， $TQ_k^r$  值的不再是由  $C_{k-1}^r$  被換手時的實際情形決定，而改成由  $C_{k-1}^r$  被期望換手時的情形決定，那麼，若  $C_{k-1}^r$  發生太晚換手的情形 (即  $C_{k-1}^r$  與  $C_{k-2}^r$  被傳送的時間大於  $T^r$ )，則  $C_{k-1}^r$  與  $C_k^r$  被傳送的時間就可以小於  $T^r$ 。假設  $C_{k-2}^r$  被換手時沒有失誤 (即 (式 15))，且  $C_{k-1}^r$  被換手時失誤晚 2 個時槽 (即 (式 16))，

$$q_{k-2}^r + t_{k-2}^r = TQ_{k-2}^r \quad (\text{式 15})$$

$$q_{k-1}^r + t_{k-1}^r = TQ_{k-1}^r + 2 \quad (\text{式 16})$$

$$TQ_k^r = TQ_{k-1}^r + T^r = q_{k-1}^r + t_{k-1}^r + T^r - 2 \quad (\text{式 17})$$

我們稱此演算法為 Modified TQS 或 MTQS。在此也看得出來，標定連線  $r$  的細胞被傳送的時間小於  $T^r$ ，即  $q_k^r + t_k^r - q_{k-1}^r - t_{k-1}^r < T^r$ （太早換手）是伴隨著前面的細胞被傳送的時間大於  $T^r$ （太晚換手）的狀況發生。而綜合前面所述，我們可以知道，若標定連線  $r$  的調整器佇列永遠細胞等待傳送，則

$$TQ_k^r = t_1^r + q_1^r + (k-1) \times T^r \quad (\text{式 18})$$

如此一來，若是標定連線  $r$  的調整器佇列持續保持有細胞等待換手的狀態，則其兩個細胞的傳送間隔的平均值就會是  $T^r$ ，也就是說該連線得到的頻寬為  $1/T^r$ 。

MTQS 所須維護的計數器的用途及數目和 TQS 一樣為  $N+1$  個，圖 7 為 MTQS 的演算法。

常數：

$T^r$ ：第  $r$  個連線所希望其兩個細胞之間被傳送的時間。

$P$ ：每個時槽可做合格檢查的次數上限。

變數：

$t$ ：系統時間，以時槽為單位

$q$ ：排程器佇列長度(佇列內的細胞數)，以槽為單位。

$N$ ：已建立的標定連線數，各連線分別被標為連線 1、連線 2、...、連線  $N$ 。

$TQ^r$ ：標定連線  $r$  下一次換手時機的  $t + q$  值。

$i$ ：用來計數該時槽已經做過幾次合格檢查。

初始： $t \leftarrow 0$ ； $q \leftarrow 0$ ； $z = 0$ ； $TQ^r \leftarrow null$ ；

排程器：

在每個時槽，若佇列中有細胞，將頭端細胞傳送出去；/\*  $q \leftarrow q - 1$  \*/

調整器：每個時槽執行以下動作

$i \leftarrow 0$ ；

**while** ( $z \leq t + q$  **and**  $i < P$ )

{  $r \leftarrow 1$ ；

**while** ( $r \leq N$ )

{ **if** ( $z = TQ^r$ )

{ **if** ( 標定連線  $r$  的調整器佇列中有細胞 )

{  $TQ^r \leftarrow TQ^r + T^r$ ；

將調整器  $r$  的頭端細胞換手；/\*  $q \leftarrow q + 1$  \*/

}

**else**

$TQ^r \leftarrow null$ ；

}

**else if** ( $TQ^r = null$  **and** 標定連線  $r$  的調整器佇列中有細胞 )

{  $TQ^r = t + q + T^r$ ；

將調整器  $r$  的頭端細胞換手；/\*  $q \leftarrow q + 1$  \*/

}

$r \leftarrow r + 1$ ；

}

$z \leftarrow z + 1$ ；

$i \leftarrow i + 1$ ；

}

---

圖 7：MTQS 演算法

## 第五節：降低太晚換手的機率的改進

### Conditional Accelerated TQS (CATQS)

倘若我們需要的連線服務，除了讓該連線的細胞被傳送間隔接近  $T'$  的前提下，較可容許太早換手而較不願接受太晚換手的情形，我們可在 TQS 特別容易造成太晚換手的情形下為該連線加速換手，也就是在特定條件下讓未達到 TQS 合格條件的連線進行換手。

我們以  $r'$  標示「在調整器佇列有細胞等待被換手的連線」。我們觀察到在 TQS 下，當  $\sum \frac{1}{T_{r'}} > 1$  時，當每一個細胞被換手，都將造成多於一個連線合格，亦即有多於一個細胞競爭排程器佇列的下一個位置，這正是 TQS 太晚換手的唯一原因。我們在 CATQS 中，當  $\sum \frac{1}{T_{r'}}$  值較高的狀況下，我們讓所有連線  $r$  的  $T'$  值降低，待狀況解決，再將  $T'$  值恢復。至於在  $\sum \frac{1}{T_{r'}}$  多高的時候  $T'$  該降低多少，可以監測各連線的即時行為做動態調整，在本文我們不詳細討論其監測及調整的方式。

關於降低  $T'$  值，我們不必太介意  $r$  與  $r'$  的分別，因為「不是  $r'$  的連線  $r$ 」沒有細胞等待被換手，不論其  $T'$  值為何都一樣沒有細胞會加入競爭。另外，由於 MTQS 主要在使連線  $r$  被服務的平均速率為  $\frac{1}{T'}$ ，加速後即偏離了 MTQS 原本的控制目的，故我們不討論在 MTQS 加入加速的機制。

CATQS 所須維護的計數器的用途及數目和 TQS 一樣為  $N + 1$  個。表一為相關演算法的比較。

	Dynamic R&S	MDRS	TQS 系列
控制目的	使標定連線的細胞被傳送的時間間隔接近定值。	以改善 Dynamic R&S 的交換機吞吐率為主。	使標定連線的細胞被傳送的時間間隔接近定值。
調整器決定換手的複雜度	$O(1)$ ，只與該連線本身有關。	$O(N)$ ，在排程器被清空時，要找出 $\min_r(t'_i)$ 。	$O(1)$ ，只與該連線本身有關。
Work Conserving	NOT	調整器有細胞時排程器連續閒置最多 1 時槽。	NOT
每個時槽需檢查所有調整器佇列的次數	$\geq 1$	1	$\geq 1$
計數器數目	$2N$	$2N$	$N + 1$

表一：各演算法的比較，在  $N$  個標定連線的情形下

## 第四章：數學分析

既然我們的控制目的是要讓標定連線  $r$  的間隔盡量接近  $T^r$ ，我們在此證明在 TQS 演算法下， $t_k^r - t_{k-1}^r = T^r$  的機率較在 Dynamic R&S 演算法下為高。由於  $T^r = 1$  時， $C_{k-1}^r$  換手後  $C_k^r$  即合格，並沒有任何控制連線傳送速率的意義，故在此不討論  $T^r = 1$  的情形。

我們以  $X_k$  表示  $C_{k-1}^r$  與  $C_k^r$  被傳送的時間間隔，加上上標， $X_k^d$  表示 Dynamic R&S 下  $C_{k-1}^r$  與  $C_k^r$  被傳送的時間間隔， $X_k^t$  表示 TQS 下  $C_{k-1}^r$  與  $C_k^r$  被傳送的時間間隔；以  $A^m$  表示第  $k-1$  個細胞被換手後的連續  $m$  個時槽中，其它連線換手至排程器佇列的細胞數，且定義  $A^0 = 0$ ；以  $S^m$  表示在第  $k$  個細胞被換手後連續  $m$  個時槽中排程器閒置的時槽數。此分析中和前面的討論一樣，每個時槽中，交換機工作程序為先處理傳送，再處理換手，所以被換手的細胞最快要等到下一個時槽才會被傳送。為了簡化分析，在此我們討論通過該交換機的全部  $N$  個連線中只有一個標定連線的狀況，且當包括此標定連線的許多連線都合格時，此標定連線為其中最後換手的一個，並假設此標定連線的調整器佇列永遠有細胞等待被換手。在此我們都省去所有符號中指出標定連線  $r$  的標籤，如  $T^r$  改記為  $T$ ， $q_k^r$  改記為  $q_k$ ， $C_k^r$  改記為  $C_k$  …等等。

由於 Dynamic R&S 最慢會在  $C_{k-1}$  被換手後的第  $T$  個時槽將  $C_k$  換手，故我們只要計算  $C_{k-1}$  被換手後的第  $T$  個時槽的狀況，因此，在  $q_k = i$  的狀況下， $X_k^d = T$  的機率是：

$$\begin{aligned}
& \Pr\{X_k^d = T \mid q_k = i\} \\
&= \Pr\{A^T = A^{T-1} = T - S^T - 1, S^T = S^{T-1} \geq 1\} \\
&+ \sum_{m=1}^T \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - 2, A^m + S^m > A^{m-1} + S^{m-1}\}
\end{aligned} \tag{式 19}$$

請注意 (式 19) 等號右邊的第一項沒有累加符號。由於  $A^m \geq A^{m-1} \geq 0$  且  $S^m \geq S^{m-1} \geq 0$ ，故「 $A^m = A^{m-1}$  且  $S^m = S^{m-1} \geq 1$ 」、「 $A^m + S^m > A^{m-1} + S^{m-1}$ 」、和「 $A^m = A^{m-1}$  且  $S^m = 0$ 」三者之間是互斥的，且涵蓋了  $A^m$ 、 $A^{m-1}$ 、 $S^m$ 、及  $S^{m-1}$  所有可能的組合情形。「 $A^m = A^{m-1}$  且  $S^m = 0$ 」在式中沒有出現，是因為在 Dynamic R&S 之下，這樣的情形不會造成「 $C_k$  在  $C_{k-1}$  換手後的第  $m$  個時槽合格且  $X_k^d = T$ 」，在  $S^m = 0$ ， $C_k$  和  $C_{k-1}$  之間的時間間隔完全靠  $A^m$  填充，所以  $A^m$  一定要等於  $T - 1$  才使  $X_k^d = T$ ，若  $A^m = A^{m-1} = T - 1$ ，那就在  $C_{k-1}$  換手後的第  $m - 1$  個時槽 (或更早) 就合格了，而在第  $m$  個時槽。「 $A^m = A^{m-1}$  且  $S^m = S^{m-1} \geq 1$ 」在等號右邊以  $m = T$  出現，是因為當  $S^m \geq 1$  時，要使  $X_k^d = T$  則必須  $A^m = T - S^m - 1$ ，此時  $A^m$  必定小於  $T - 1$ ， $C_k$  不會因為其他連線換手的細胞數合格，而是因為距  $C_{k-1}$  換手  $T$  時槽而換手，故在此情形下，「 $C_k$  在  $C_{k-1}$  換手後的第  $m$  個時槽合格」中的  $m$  值必然等於  $T$ 。

同樣地我們以  $X_k^t$  表示 TQS 中， $C_{k-1}$  與  $C_k$  被傳送的時間間隔，由於 TQS 會偵測到閒置的時槽，且沒有最慢會在  $C_{k-1}$  被換手後的第  $T$  個時槽將  $C_k$  換手的機制，因此，在  $q_k = i$  的狀況下， $X_k^t = T$  的機率是：

$$\Pr\{X_k^t = T \mid q_k = i\} = \sum_{m=1}^{\infty} \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - 2 - S^{m-1}\} \tag{式 20}$$

當  $S^m = 0$  時

$$\Pr\{X_k^d = T, S^m = 0 | q_k = i\} = \sum_{m=1}^T \Pr\{A^m = T - 1, A^{m-1} \leq T - 2, S^m = 0\} \quad (\text{式 21})$$

$$\Pr\{X_k^t = T, S^m = 0 | q_k = i\} = \sum_{m=1}^{\infty} \Pr\{A^m = T - 1, A^{m-1} \leq T - 2, S^m = 0\} \quad (\text{式 22})$$

(式 21) 及 (式 22) 只有  $m$  值的上限的不同，(式 22) 中的  $m$  值的上限用  $\infty$  來表示只要符合右邊大括號裡的條件的  $m$  值皆計入累加，而 (式 21) 中  $m$  有一明確的上限值  $T$ 。在此我們要確定，TQS 在符合右邊大括號裡的條件的狀況下， $m$  值有可能大於  $T$ ；當  $i > T$  且  $A^i \leq T - 2$ ，(式 22) 中的  $m$  就必定包含大於  $T$  的值，所以我們得到：

$$\Pr\{X_k^t = T, S^m = 0 | q_k = i\} > \Pr\{X_k^d = T, S^m = 0 | q_k = i\} \quad (\text{式 23})$$

再考慮  $S^m \neq S^{m-1}$  的狀況， $S^m - S^{m-1}$  在任何情況下都只會有等於 0 或 1 兩種結果，故若  $S^m \neq S^{m-1} \Rightarrow S^m - S^{m-1} = 1 \Rightarrow S^m \geq 1$ 。

$$\begin{aligned} & \Pr\{X_k^d = T, S^m \neq S^{m-1} | q_k = i\} \\ &= \sum_{m=1}^T \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - 2, S^m \neq S^{m-1}\} \end{aligned} \quad (\text{式 24})$$

$$\begin{aligned} & \Pr\{X_k^t = T, S^m \neq S^{m-1} | q_k = i\} \\ &= \sum_{m=1}^{\infty} \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - 2 - S^{m-1}, S^m \neq S^{m-1}\} \end{aligned} \quad (\text{式 25})$$

(式 24) 中，由於  $S^m > S^{m-1} \Rightarrow A^m + S^m > A^{m-1} + S^{m-1}$ ，故 (式 19) 中等號右邊第二項的第三個條件不必再寫出。由於  $S^m - S^{m-1} = 1$ ，(式 24) 及 (式 25) 的第一個條件成立時，第二個條件也同時成立了，所以兩式的條件都一樣，而 (式 25) 的累加上限值較高，故



$$\Pr\{X_k^t = T, S^m \neq S^{m-1} | q_k = i\} > \Pr\{X_k^d = T, S^m \neq S^{m-1} | q_k = i\} \quad (\text{式 26})$$

再來考慮  $S^m = S^{m-1} \geq 1$  的情形

$$\begin{aligned} & \Pr\{X_k^d = T, S^m = S^{m-1} \geq 1, A^m \neq A^{m-1} | q_k = i\} \\ &= \sum_{m=1}^T \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - 2, S^m = S^{m-1} \geq 1, A^m - A^{m-1} \geq 1\} \quad (\text{式 27}) \\ &= \sum_{m=1}^T \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - S^m - 2, S^m = S^{m-1} \geq 1\} \end{aligned}$$

$$\begin{aligned} & \Pr\{X_k^d = T, S^m = S^{m-1} \geq 1, A^m = A^{m-1} | q_k = i\} \\ &= \Pr\{A^T = A^{T-1} = T - S^T - 1, S^T = S^{T-1} \geq 1\} \\ &= \sum_{j=1}^{T-1} \Pr\{A^T = A^j = T - S^T - 1, A^j > A^{j-1}, S^T = S^j \geq 1\} \quad (\text{式 28}) \\ &= \sum_{j=1}^{T-1} \Pr\{A^T = A^j = T - S^j - 1, A^{j-1} \leq T - S^j - 2, S^T = S^j \geq 1\} \end{aligned}$$

$$\begin{aligned} & \Pr\{X_k^t = T, S^m = S^{m-1} \geq 1 | q_k = i\} \\ &= \sum_{m=1}^{\infty} \Pr\{A^m = T - S^m - 1, A^{m-1} \leq T - 2 - S^m, S^m = S^{m-1} \geq 1\} \quad (\text{式 29}) \end{aligned}$$

(式 27) 最終的第二個條件是來自  $A^m = T - S^m - 1$  及  $A^m - A^{m-1} \geq 1$ ，該條件比原來的  $A^{m-1} \leq T - 2$  嚴格。(式 28) 中由於  $S^T = S^{T-1}$  表示在  $C_k$  被換手的時槽排程器沒有閒置，且  $X_k^d = T > 1$ ，這表示在  $C_{k-1}$  換手後且  $C_k$  合格之前一定有其他連線的細胞進入排程器佇列，所以我們一定能找到 (式 28) 中  $A^j > A^{j-1}$  且  $A^j = T - S^T - 1$  的  $j$  值；且由於  $S^T = S^{T-1}$ ，表示所有的閒置都發生在這第  $j$  個時槽之前，因為若有閒置發生此之後，則第  $T$  個時槽要不閒置，就必須再有其他連線的細胞進到排程器佇列，那麼再有細胞進來的這個時槽才是真正的「第  $j$  個時槽」，這第  $j$  個

時槽進來的細胞要**剛好** (不能多不能少) 使排程器到第  $T$  個時槽不閒置，且讓「已經發生的閒置時槽數與其他已經換手的細胞數的和」恰等於  $T-1$ ，所以  $S^T = S^{T-1} = \dots = S^j$ ，且  $A^T = A^{T-1} = \dots = A^j$ 。

(式 27) 的最終條件和 (式 29) 都相同，很明顯的符合 (式 27) 條件的狀況也都被涵蓋在 (式 29) 的條件當中，且 (式 29) 的累加項較多。而將 (式 29) 中的  $m$  寫成  $j$  之後，也和 (式 28) 幾乎相同，差別只在 (式 28) 仍要求  $S^T = S^j$  且  $A^T = A^j$  但 (式 29) 不必，故 (式 28) 的狀況也被涵蓋在 (式 29) 的條件當中。由於 (式 27) 及 (式 28) 是的條件是互斥的，且都被涵蓋在 (式 29) 當中，故

$$\begin{aligned}
& \Pr\{X_k^t = T, S^m = S^{m-1} \geq 1 \mid q_k = i\} \\
& > \Pr\{X_k^d = T, S^m = S^{m-1} \geq 1, A^m \neq A^{m-1} \mid q_k = i\} \\
& + \Pr\{X_k^d = T, S^m = S^{m-1} \geq 1, A^m = A^{m-1} \mid q_k = i\} \quad (\text{式 30}) \\
& \Rightarrow \\
& \Pr\{X_k^t = T, S^m = S^{m-1} \geq 1 \mid q_k = i\} > \Pr\{X_k^d = T, S^m = S^{m-1} \geq 1 \mid q_k = i\}
\end{aligned}$$

綜合以上討論，我們可以得到：

$$\begin{aligned}
& \Pr\{X_k^t = T\} = \Pr\{X_k^t = T \mid q_k = i\} \times \Pr\{q_k = i\} \\
& = \Pr\{X_k^t = T, S^m = 0 \mid q_k = i\} \times \Pr\{q_k = i\} \\
& + \Pr\{X_k^t = T, S^m \neq S^{m-1} \mid q_k = i\} \times \Pr\{q_k = i\} \\
& + \Pr\{X_k^t = T, S^m = S^{m-1} \geq 1 \mid q_k = i\} \times \Pr\{q_k = i\} \\
& > \Pr\{X_k^d = T, S^m = 0 \mid q_k = i\} \times \Pr\{q_k = i\} \\
& + \Pr\{X_k^d = T, S^m \neq S^{m-1} \mid q_k = i\} \times \Pr\{q_k = i\} \\
& + \Pr\{X_k^d = T, S^m = S^{m-1} \geq 1 \mid q_k = i\} \times \Pr\{q_k = i\} \\
& \Rightarrow \Pr\{X_k^t = T\} > \Pr\{X_k^d = T\} \quad (\text{式 31})
\end{aligned}$$

TQS 下的標定連線中的兩個相鄰細胞被傳送的時間隔恰好為預設的  $T$  值的機率，比 Dynamic R&S 下的標定連線要高。

## 第五章：模擬結果

我們使用 OPNET 來模擬交換機的行爲。ON\_OFF Markov model[18]來做模擬的細胞發送端，其細胞發送的行爲以一個有限狀態機表示，如圖 8。在 ON 狀態的時候，發送端以最大速率發送細胞，在我們的模擬中以每個時槽一個細胞為最大速率，在 OFF 狀態的時候，發送端不傳送細胞。每個時槽都要以原來的狀態判斷接下來該是什麼狀態，當原來是 ON 狀態時，下一個狀態是 ON 的機率是  $a$ ，是 OFF 的機率是  $1 - a$ ；當原來是 OFF 狀態時，下一個狀態是 OFF 的機率是  $b$ ，是 ON 的機率是  $1 - b$ 。當最大速率為每個時槽發送一個細胞時，此發送端處在 ON 狀態的機率也就是發送細胞的平均速率，可以 (式 32) 計算。

$$\lambda = \frac{1 - b}{1 - a + 1 - b} = \frac{1 - b}{2 - a - b} \quad (\text{式 32})$$

由於  $T' = 1$  連線，不論是 Dynamic R&S、MDRS、TQS、或是 MTQS，標定連線的一個細胞合格後，該連線的下一個細胞就合格了，無所謂對速率的控制，所以我們在以下的部份模擬中，以  $T' = 1$  的連線提供背景流量。

我們將各組模擬中，各連線的細胞和細胞之間被傳送的時間間隔記錄下來，做成 PMF (Probability Mass Function)，以下的模擬的記錄都已將因為細胞發送端在 OFF 狀態時的細胞間隔排除在外，不列入統計。因為連線 1~3 只是提供背景流量，故不討論其記錄資料。在以下所有的 PMF 中，橫軸為細胞被傳送的時間間隔，縱軸為機率值。所有的模擬時間都為  $10^8$  個時槽。以下共四組模擬。

既然以速率穩定作為我們的演算法的控制目的，我們以同一標定連線  $r$  的細胞被交換機傳送的時間間隔的標準差作為判定演算法優劣的依據，較小者為優，另外平均值接近  $T'$  也列為一項重要的參考依據。

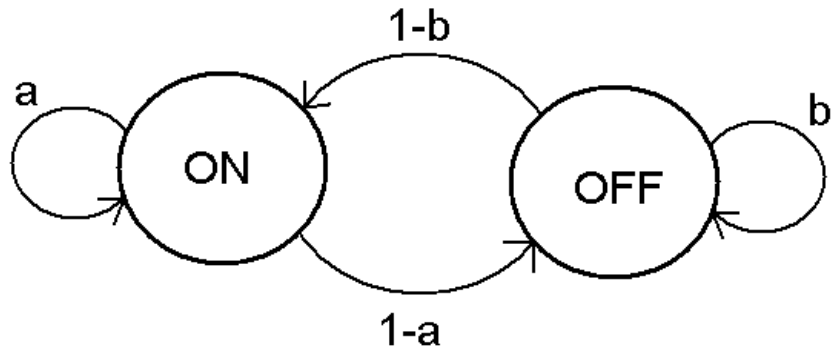


圖 8 : ON\_OFF Markov model

編號 $r$	$T^r$	$\lambda^r$	$a$	合格檢查次數上限	演算法
1	1	0.247	0.1	1、2、3、 $\infty$	Dynamic R&S TQS MTQS
2	1	0.195	0.1		
3	1	0.200	0.1		
4	10	0.043	0.1		
5	10	0.042	0.1		
6	5	0.111	0.1		
7	5	0.112	0.1		
狀況一				1	Dynamic R&S
狀況二				2	Dynamic R&S
狀況三				3	Dynamic R&S
狀況四				$\infty$	Dynamic R&S
狀況五				1	TQS
狀況六				2	TQS
狀況七				3	TQS
狀況八				$\infty$	TQS
狀況九				1	MTQS
狀況十				2	MTQS
狀況十一				3	MTQS
狀況十二				$\infty$	MTQS

表二：第一組模擬的參數

## 模擬一：每時槽能做的合格檢查次數的影響

首先我們以第一組模擬觀察 Dynamic R&S、TQS、及 MTQS 在每個時槽中能做的合格檢查次數有限的情形下，不同的檢查次數對其效果的影響。設定參數如表二所列， $a$  即為 ON\_OFF Markov model 中的  $a$ 。各連線在每個時槽能做的合格檢查次數上限和演算法都相同，如表二共列出 12 種狀況，我們將這 12 種狀況下連線 4 ( $T' = 4$ ) 模擬結果展示於圖 9 到圖 14。在圖 9 到圖 11，可看出 Dynamic R&S、TQS、及 MTQS 都在每個時槽可做合格檢查次數的上限到達 3 次時的結果，接近沒有次數上限的結果。在圖 16 到圖 18 中看到，雖然在每個時槽只能做一次合格檢查的情形下 TQS 和 MTQS 的效果明顯地較 Dynamic R&S 差，但在將上限提升至 2 次或更大時，TQS 和 MTQS 的細胞傳送間隔就較 Dynamic R&S 來得集中。本文自此以後的模擬，都是沒有合格檢查次數上限的狀況。

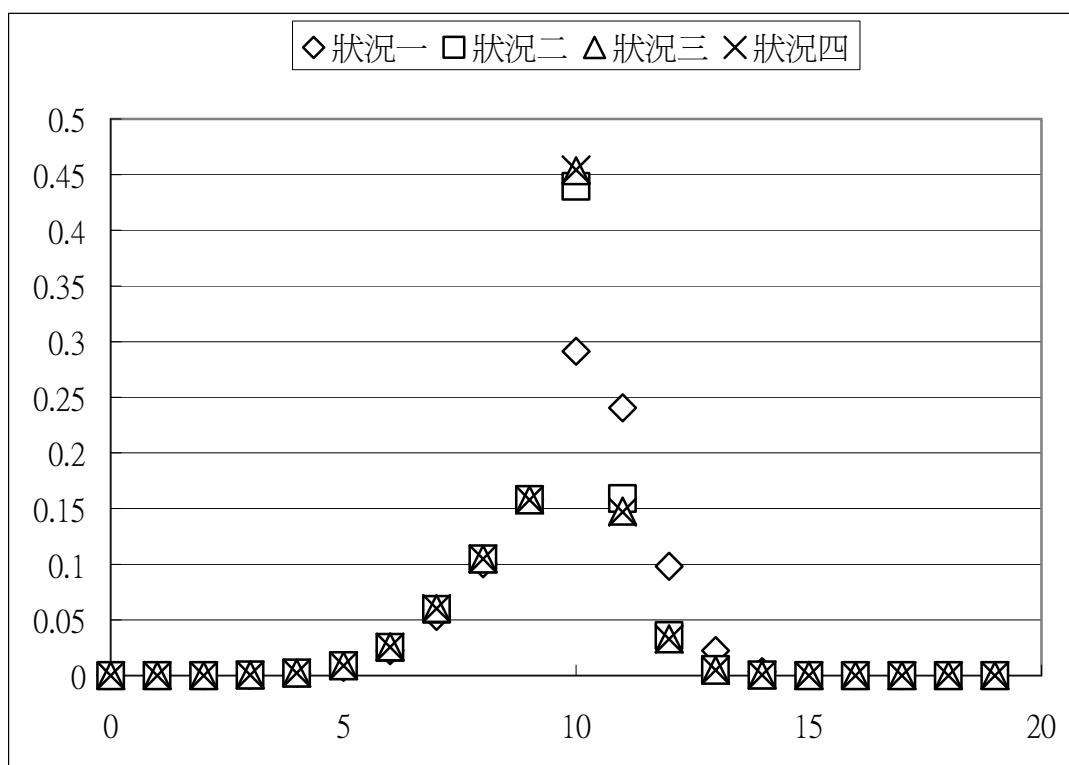


圖 9：第一組模擬中，Dynamic R&S 合格檢查次數的比較

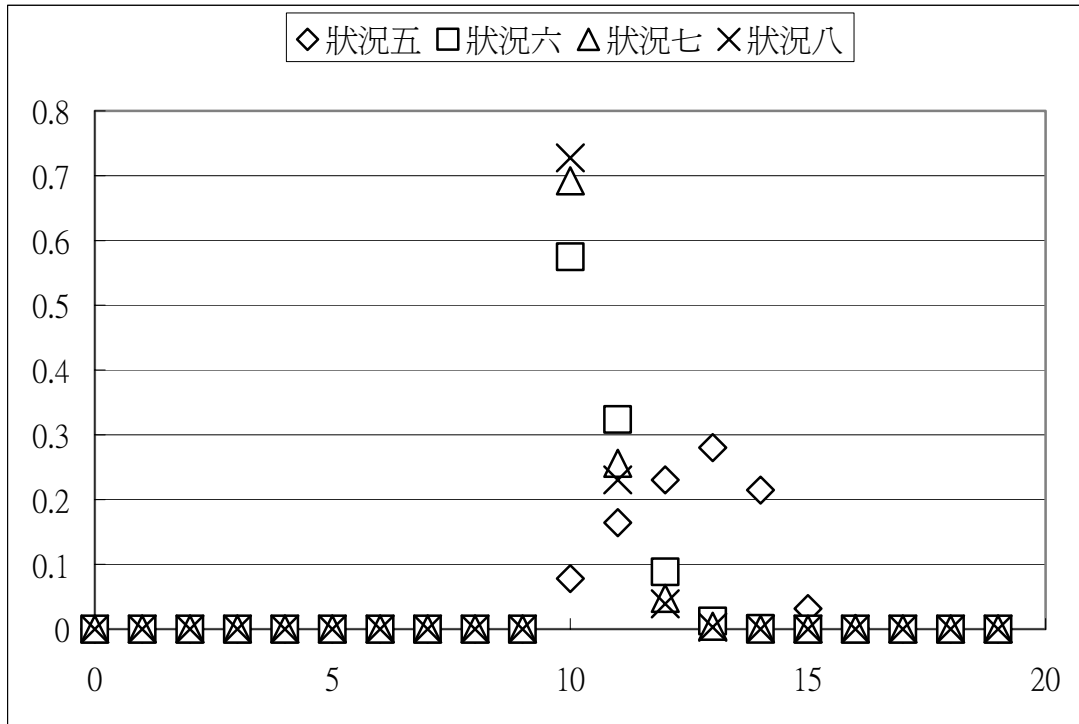


圖 10：第一組模擬中，TQS 合格檢查次數的比較

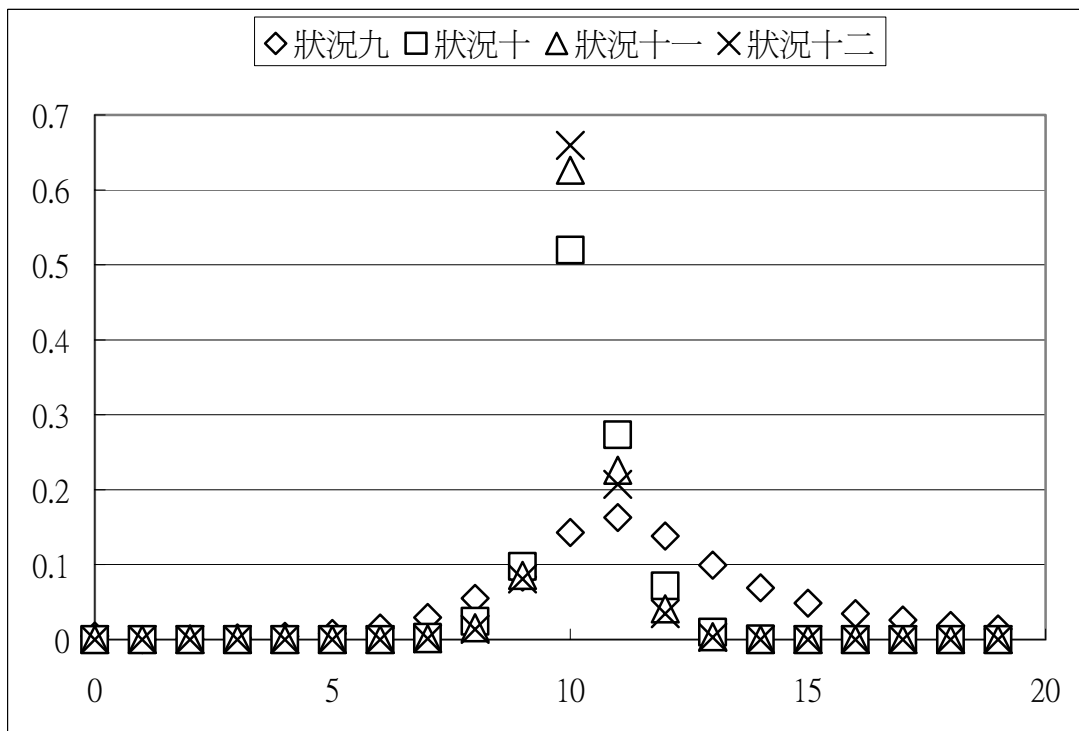


圖 11：第一組模擬中，MTQS 合格檢查次數的比較

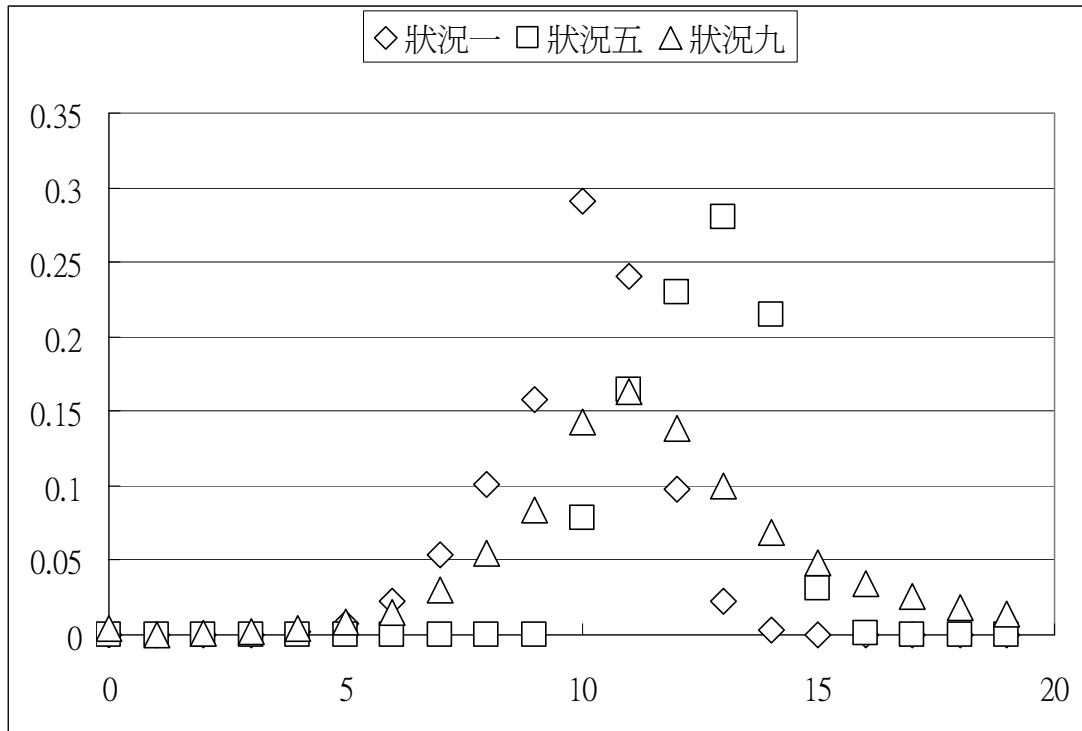


圖 12：第一組模擬中，每時槽合格檢查一次，各演算法的比較

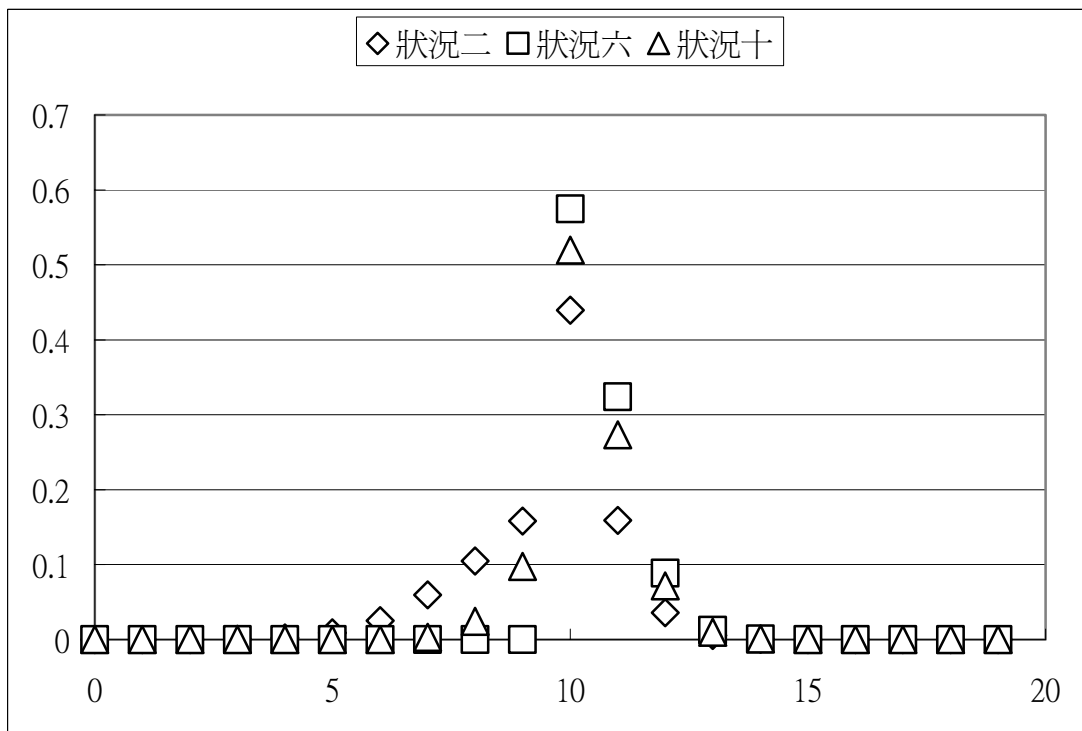


圖 13：第一組模擬中，每時槽合格檢查最多二次，各演算法的比較

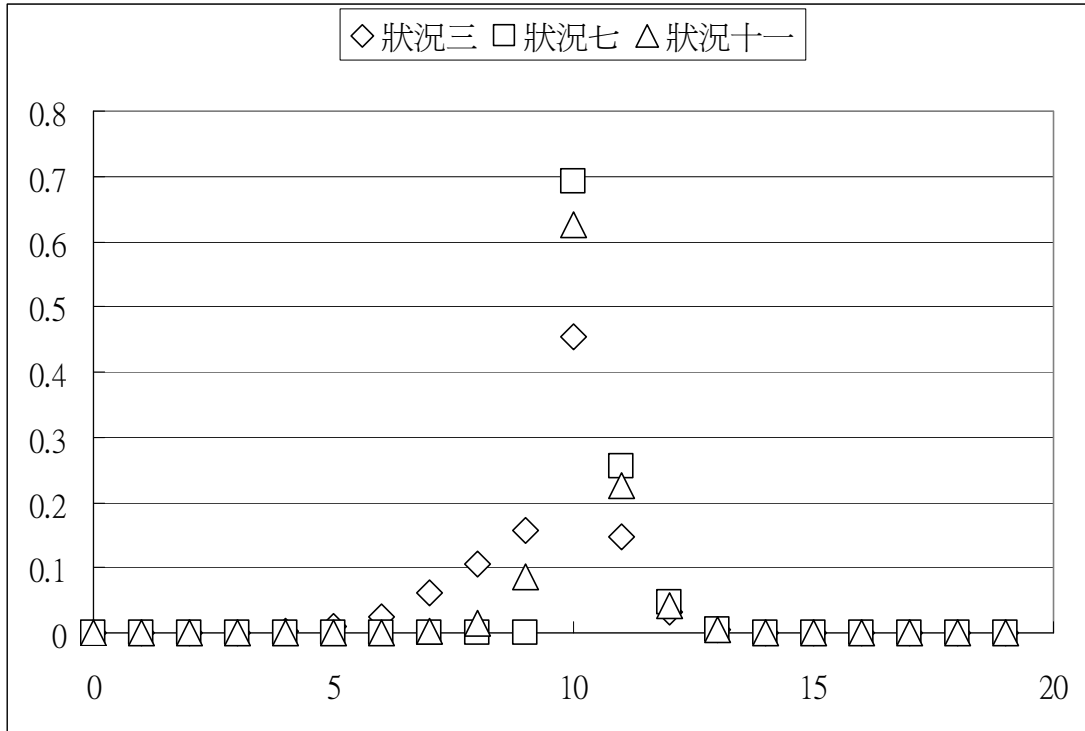


圖 14：第一組模擬中，每時槽合格檢查最多三次，各演算法的比較



## 模擬二：頻寬使用率與暴衝的影響

編號 $r$	$T^r$	$\lambda^r$		$a^r$	
		狀況一、二	狀況三、四	狀況一、三	狀況二、四
1	1	0.185	0.247	0.1	0.9
2	1	0.145	0.195	0.1	0.9
3	1	0.150	0.200	0.1	0.9
4	10	0.030	0.043	0.1	0.9
5	10	0.029	0.042	0.1	0.9
6	5	0.080	0.111	0.1	0.9
7	5	0.081	0.112	0.1	0.9

表三：第二組模擬中的設定

在第二組的模擬中，分別以不同的  $\sum_r \lambda^r$  及  $a^r$  值，共有四種狀況，每個狀況將所有的連線都用 Dynamic R&S、MDRS、TQS、及 MTQS 模擬，參數設定如表二。狀況一的頻寬使用率較低且細胞發送端暴衝較緩和， $\sum_r \lambda^r = 0.70$ ，各連線在 ON\_OFF Markov model 的  $a = 0.1$ 。狀況二的頻寬使用率較低且發送端暴衝較劇烈， $\sum_r \lambda^r = 0.70$ ，各連線在 ON\_OFF Markov model 的  $a = 0.9$ 。狀況三模擬的頻寬使用率較高且發送端暴衝較緩和， $\sum_r \lambda^r = 0.95$ ，各連線在 ON\_OFF Markov model 的  $a = 0.1$ 。狀況四模擬的頻寬使用率較高且發送端暴衝較劇烈， $\sum_r \lambda^r = 0.95$ ，各連線在 ON\_OFF Markov model 的  $a = 0.9$ 。由於連線 1 到連線 3 僅用作提供背景流量，故結果不在此列出，僅列出連線 4 到連線 7 的結果。

第二組模擬結果 PMF 作圖於圖 15 到圖 30 中。我們將結果的標準差與平均值列於表四及表五，並將標準差及「平均值與  $T^r$  的誤差百分比」作圖於圖 31 到

圖 38，因為 MDRS 的值較大，列入圖中會妨礙讀圖，故在圖 31 到圖 38 中未列出 MDRS 的標準差及平均值。

圖 15 到圖 30 中可見 Dynamic R&S 在  $X_k = T$  的機率較 TQS 來得小，與第五章的分析結果 (式 31) 相符合；狀況二、四中 Dynamic R&S 太早換手的情形明顯較狀況一、三多，這可能是因為暴衝劇烈而使排程器佇列變長的機會較多，而一但其中某些連線的暴衝結束後，細胞進入排程器的佇列的速率小於排程器傳送細胞的速率，即在  $T^r$  個時槽內，排程器傳送了  $T^r$  個細胞，而收到了細胞小於  $T^r$ ，而 Dynamic R&S 因為換手間隔不會大於  $T^r$ ，在細胞進入排程器的佇列的速率小於排程器傳送細胞的速率的狀況做出換手，則太早換手的情形就會較多。

圖 15 到圖 30 中看到 TQS 如前所述完全沒有太早換手的情形。MTQS 的太早換手伴隨著太晚換手的情形出現，在狀況一及狀況三的結果中，其圖形並未對稱於  $X_k = T$ ，這是因為連線的暴衝較緩和，即 ON\_OFF Markov model 中的 ON 持續的時間較短，在太晚換手發生後，還來不及發生伴隨的太早換手，就因為發送端的暴衝結束而沒有細胞可送，故太晚換手發生的機率較太早換手為大，這和其得到的頻寬應為  $\frac{1}{T^r}$  並沒有直接關係，我們在第三組模擬將展示出 MTQS 得到的頻寬為  $\frac{1}{T^r}$ ；在狀況二及狀況四中，我們看到由於暴衝時間加長，MTQS 的圖形就幾乎對稱於  $X_k = T$ 。

MDRS 由於對連線的控制目的不同，僅將其結果加入供參考；MDRSJ 已在前述，並不是一個可以穩定工作的演算法，細胞進入調整器後就無法換手，在模擬上來說，就是發送端產生的細胞佔據的記憶體不斷地變多，最後的結果是根本跑不完模擬就被作業系統中斷，故無法列出結果。

在圖 31 到圖 34 中，Dynamic R&S 的標準差明顯地高於 TQS 與 MTQS，這表示在速率穩定的表現上，TQS 和 MTQS 較 Dynamic R&S 為優；TQS 的標準差較 MTQS 略低，是因為 MTQS 中的太晚換手和太早換手伴隨著發生，不若 TQS 中太晚換手不影響之後的細胞是否失誤。Dynamic R&S 在暴衝程度增加時，標

準差亦顯著增加，而 TQS 和 MTQS 則略有減少，請注意兩者的標準差的變化的成因是完全不同的；Dynamic R&S 的失誤包含了太早換手和太晚換手的情形變多，而且因為其限制兩個細胞被換手的間隔不大於  $T'$  的情形下，太早換手的成分較太晚換手的成分高許多，Dynamic R&S 的標準差提高和太早換手的情形增加關係較大；TQS 完全沒有太早換手的情形，故 TQS 的標準差變小是因為太晚換手的情形變少；MTQS 的太早換手肇因於之前的太晚換手而伴隨發生，和 Dynamic R&S 下太早換手獨立發生的狀況不同；Dynamic R&S 太早換手的情形變多的可能原因已在本組模擬討論中的第一段指出可能的情形，TQS 和 MTQS 太晚換手的情形變少，可能因為在 ON\_OFF Markov model 下，平均速率不變但暴衝程度較低，則表示細胞發送端在 ON 和 OFF 狀態切換較頻繁，加入競爭（有細胞在調整器佇列中等待換手）的連線數變化亦較頻繁，故顯示出暴衝程度變大（ON 和 OFF 切換不頻繁），細胞被傳送間隔的標準差反而降低的情形。

在圖 35 到圖 38 中，Dynamic R&S 的細胞被傳送間隔的平均值的變化亦是因為太早換手的情形變化的結果，已在上一段討論，TQS 的細胞被傳送間隔的平均值的變化亦是因為太晚換手的情形變化的結果，亦已在上一段討論。而 MTQS 的平均值應是等於  $T'$ ，已在本組模擬討論中的第二段指出原因，並在下一組模擬展示出來。

另外我們在第二組模擬加上狀況五，同樣地模擬時間為  $10^8$  個時槽，其頻寬使用率與狀況一和二相同，但其暴衝程度 ( $a$  值) 隨著時間改變，其決定方式為

$$a = 0.05 + 0.9 \times \sin\left(\frac{\pi \times t}{10^8}\right) \quad (\text{式 33})$$

以狀況五來模擬連線的暴衝狀況變化時各連線之間的差異，其結果列在圖 39 到圖 44。暴衝程度不固定的狀況下，其結果的討論與此模擬的其它狀況相同。

	演算法	標準差			
		連線 4	連線 5	連線 6	連線 7
狀況一	Dynamic R&S	1.26	1.27	1.05	1.07
	TQS	0.56	0.57	0.59	0.61
	MTQS	0.67	0.68	0.73	0.77
	MDRS	3.69	3.72	2.05	1.95
狀況二	Dynamic R&S	2.63	2.63	1.43	1.43
	TQS	0.36	0.36	0.35	0.35
	MTQS	0.46	0.46	0.51	0.51
	MDRS	39.20	41.66	15.25	14.47
狀況三	Dynamic R&S	1.40	1.40	1.04	1.06
	TQS	0.56	0.56	0.58	0.58
	MTQS	0.70	0.70	0.75	0.77
	MDRS	8.64	8.67	4.38	4.25
狀況四	Dynamic R&S	2.95	2.95	1.58	1.60
	TQS	0.45	0.45	0.44	0.44
	MTQS	0.56	0.57	0.63	0.63
	MDRS	41.36	43.28	13.47	13.16

表四：第二組模擬結果的標準差

	演算法	平均值			
		連線 4	連線 5	連線 6	連線 7
狀況一	Dynamic R&S	9.87	9.88	4.88	4.89
	TQS	10.33	10.34	5.35	5.37
	MTQS	10.21	10.22	5.20	5.22
	MDRS	7.67	7.67	4.49	4.47
狀況二	Dynamic R&S	8.83	8.83	4.40	4.41
	TQS	10.13	10.13	5.12	5.12
	MTQS	10.01	10.01	5.01	5.01
	MDRS	8.20	8.41	4.77	4.67
狀況三	Dynamic R&S	9.52	9.54	4.74	4.77
	TQS	10.32	10.32	5.34	5.34
	MTQS	10.17	10.18	5.16	5.17
	MDRS	10.78	10.80	5.68	5.64
狀況四	Dynamic R&S	8.21	8.24	4.17	4.18
	TQS	10.21	10.21	5.20	5.20
	MTQS	10.01	10.01	5.01	5.02
	MDRS	15.41	15.78	6.99	6.93

表五：第二組模擬結果的平均值

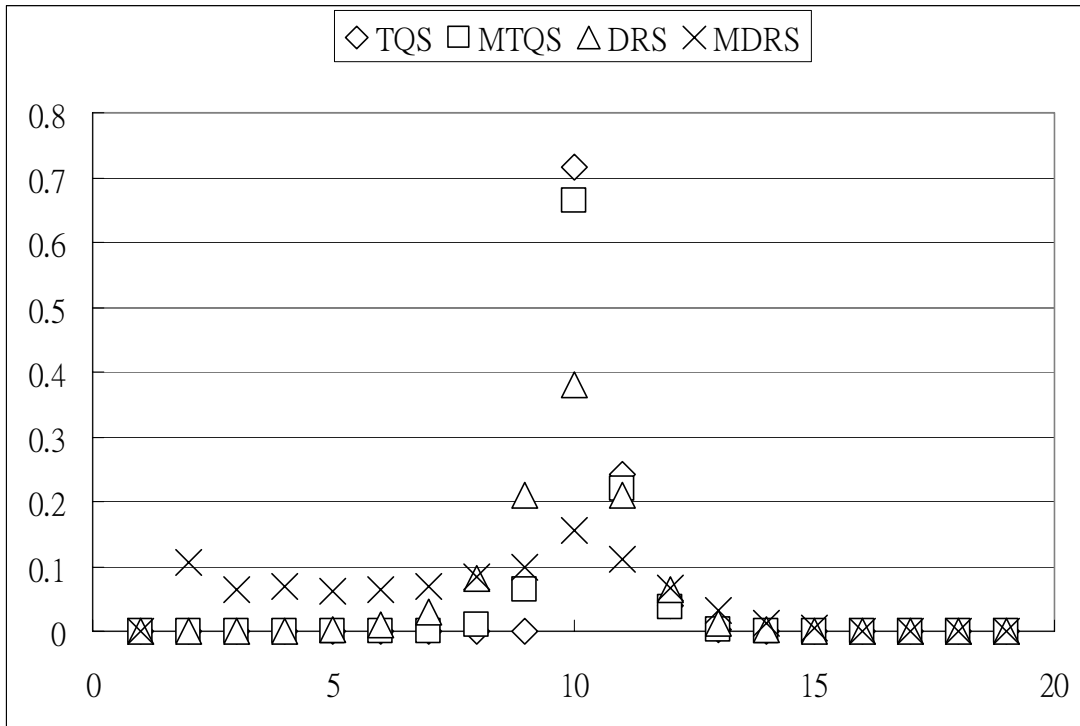


圖 15：第二組模擬狀況一的連線 4

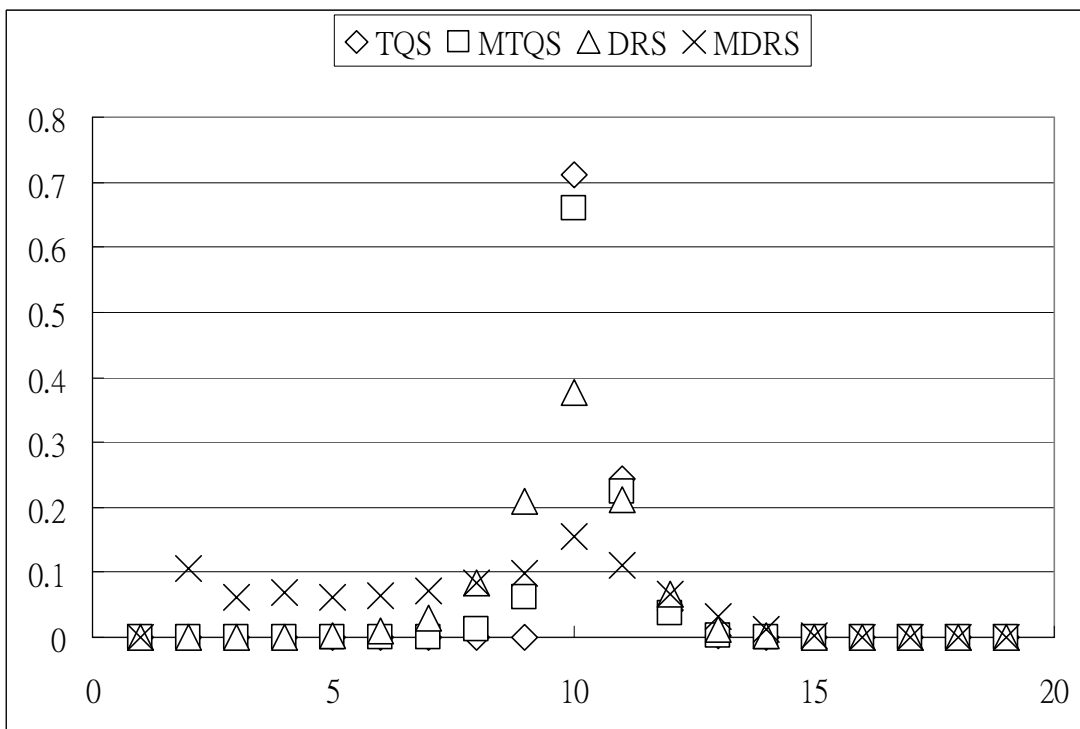


圖 16：第二組模擬狀況一的連線 5

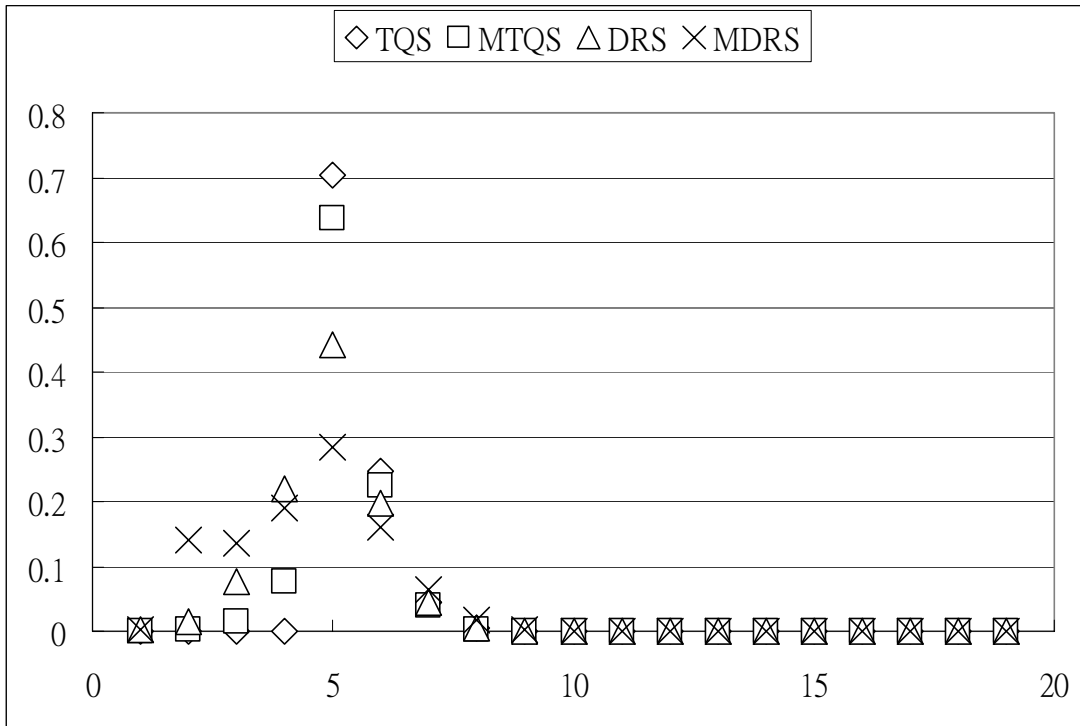


圖 17：第二組模擬狀況一的連線 6

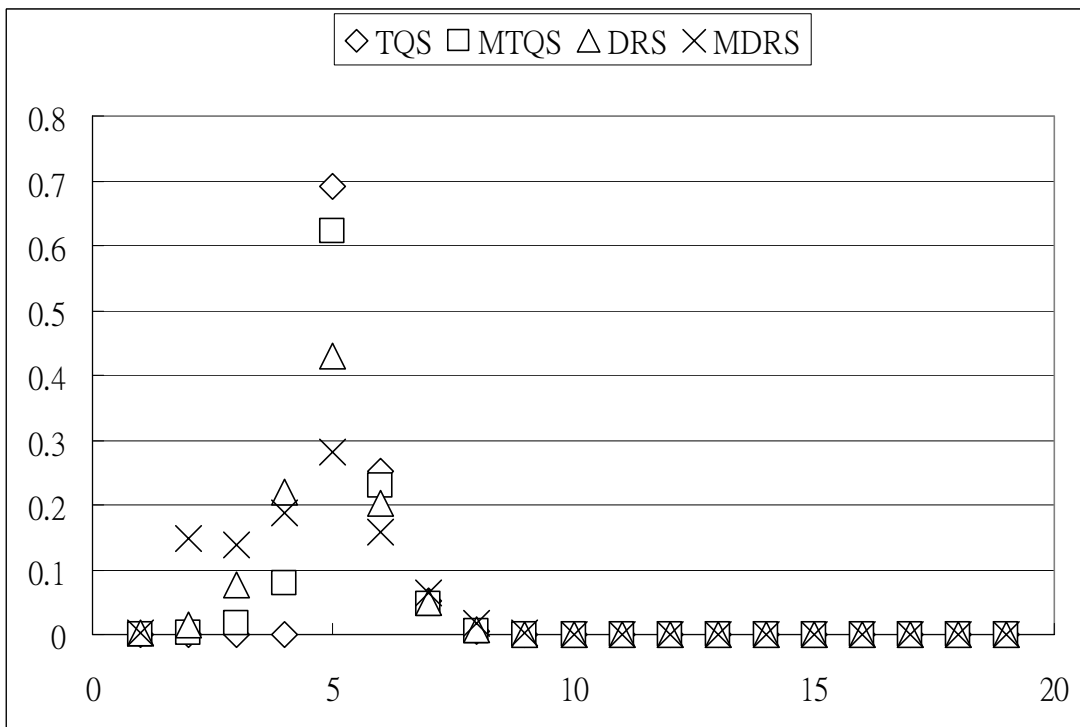


圖 18：第二組模擬狀況一的連線 7

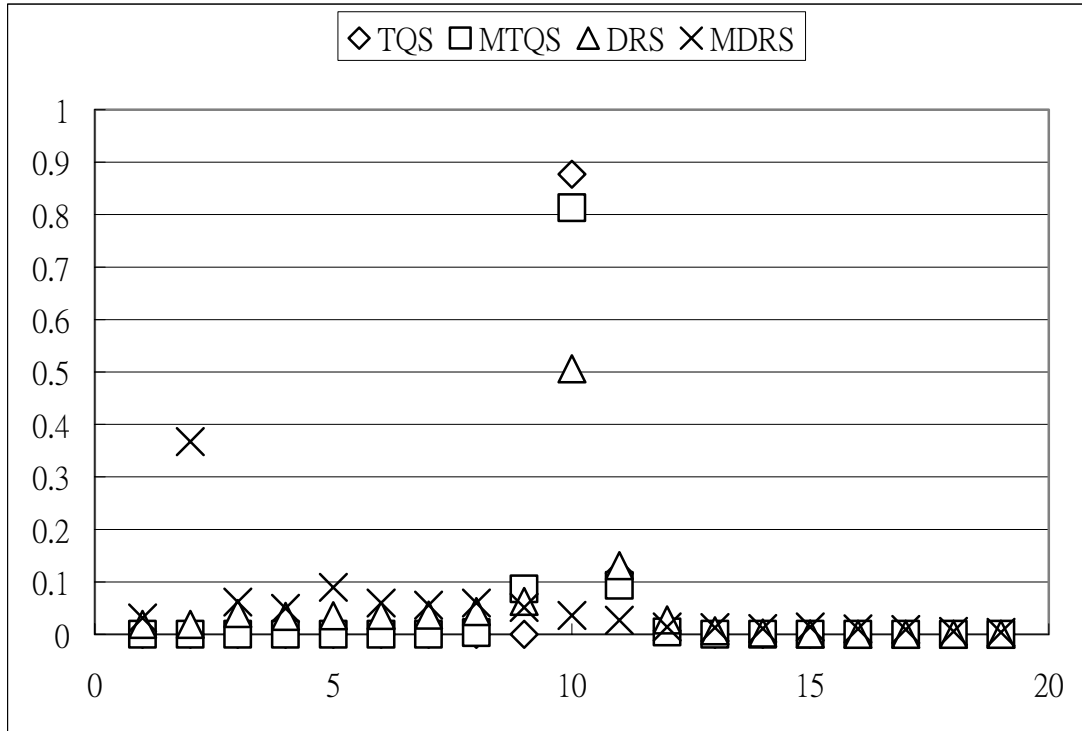


圖 19：第二組模擬狀況二的連線 4

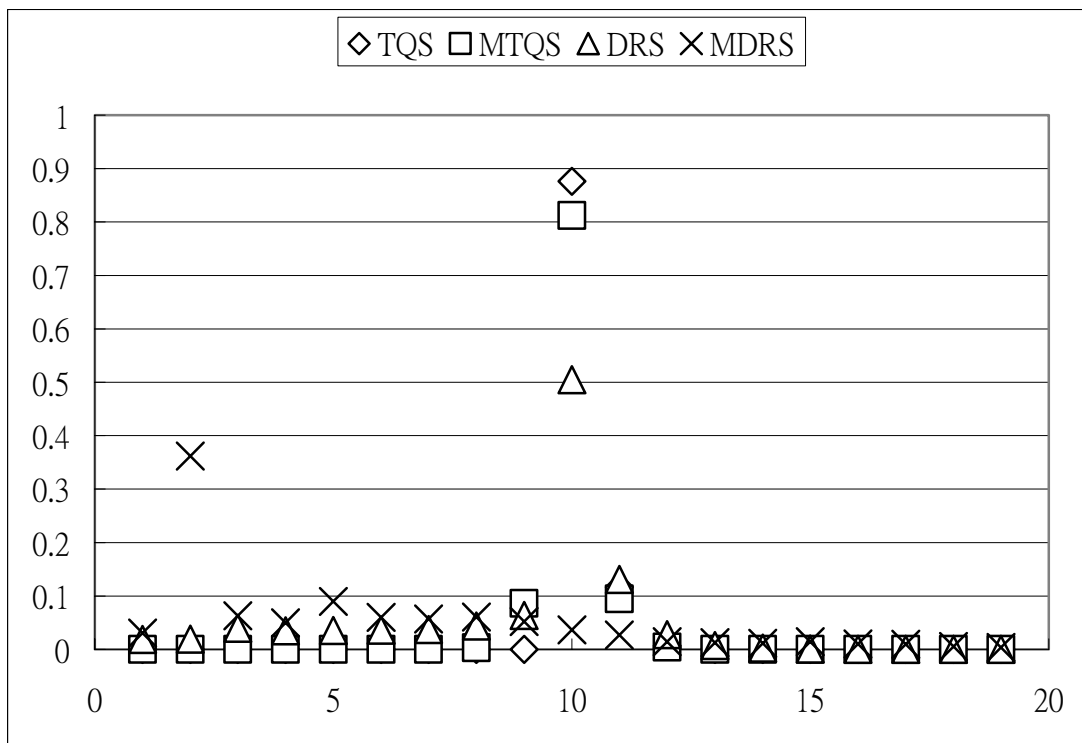


圖 20：第二組模擬狀況二的連線 5



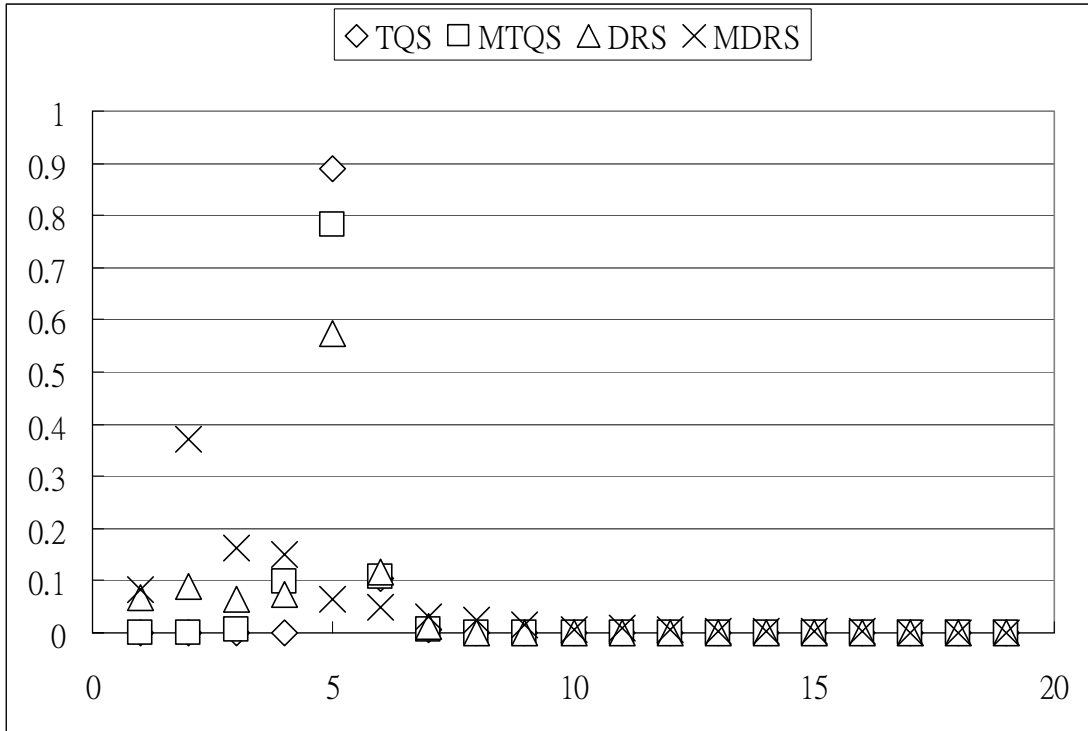


圖 21：第二組模擬狀況二的連線 6

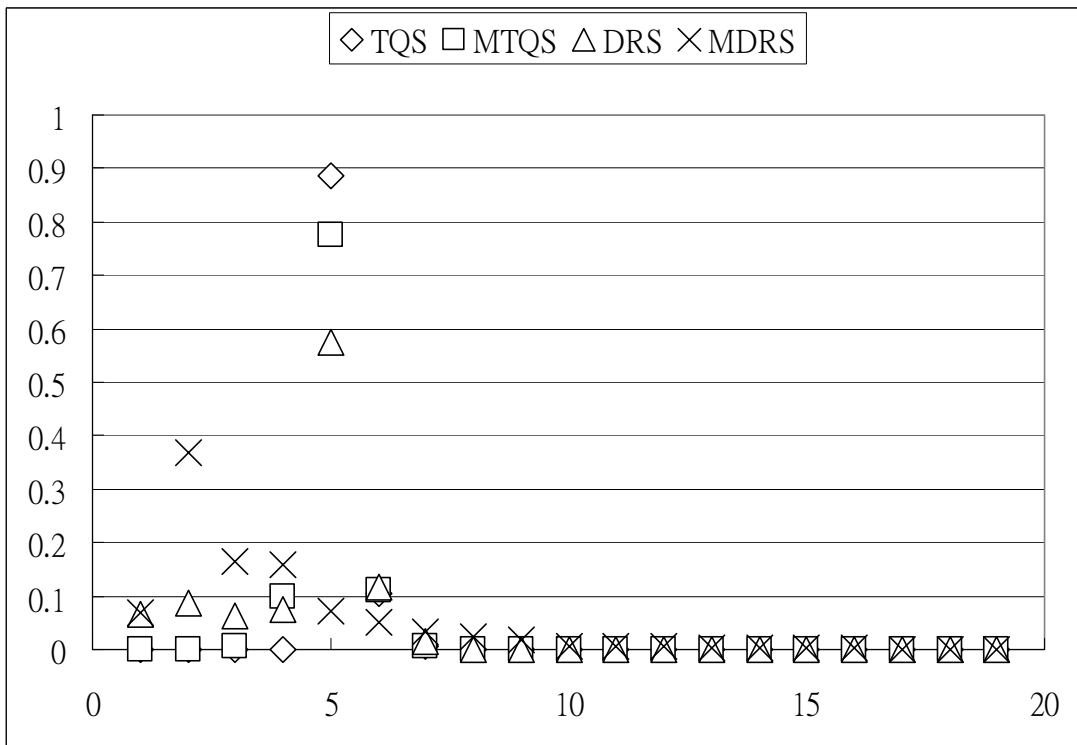


圖 22：第二組模擬狀況二的連線 7

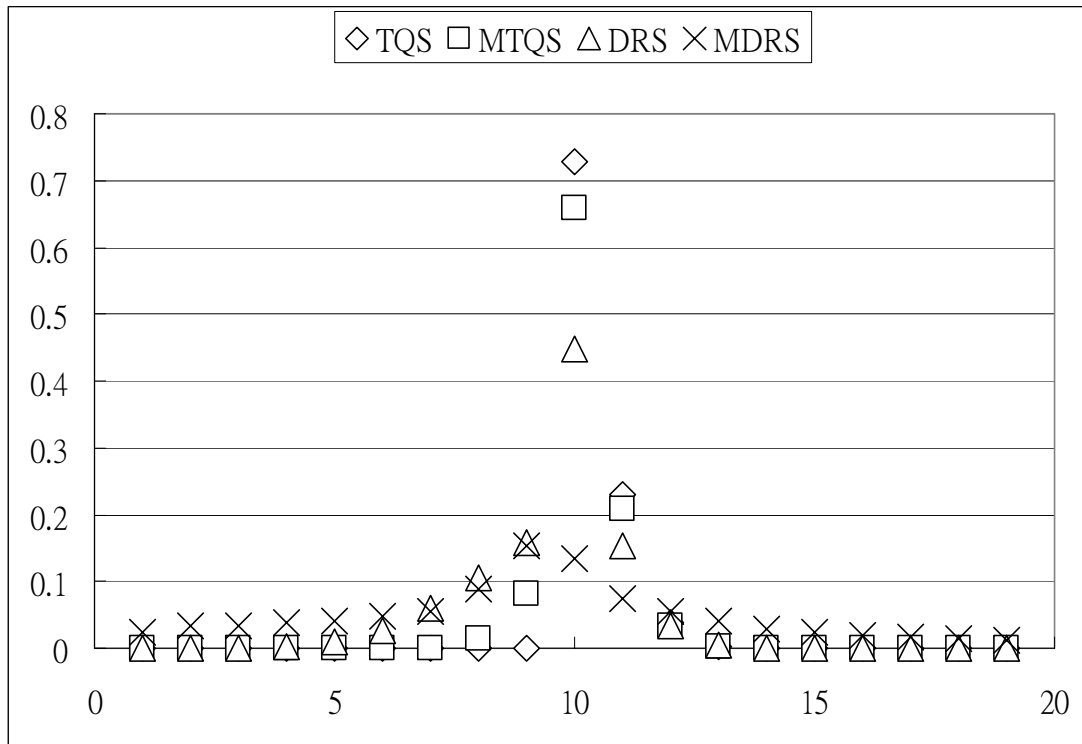


圖 23：第二組模擬狀況三的連線 4

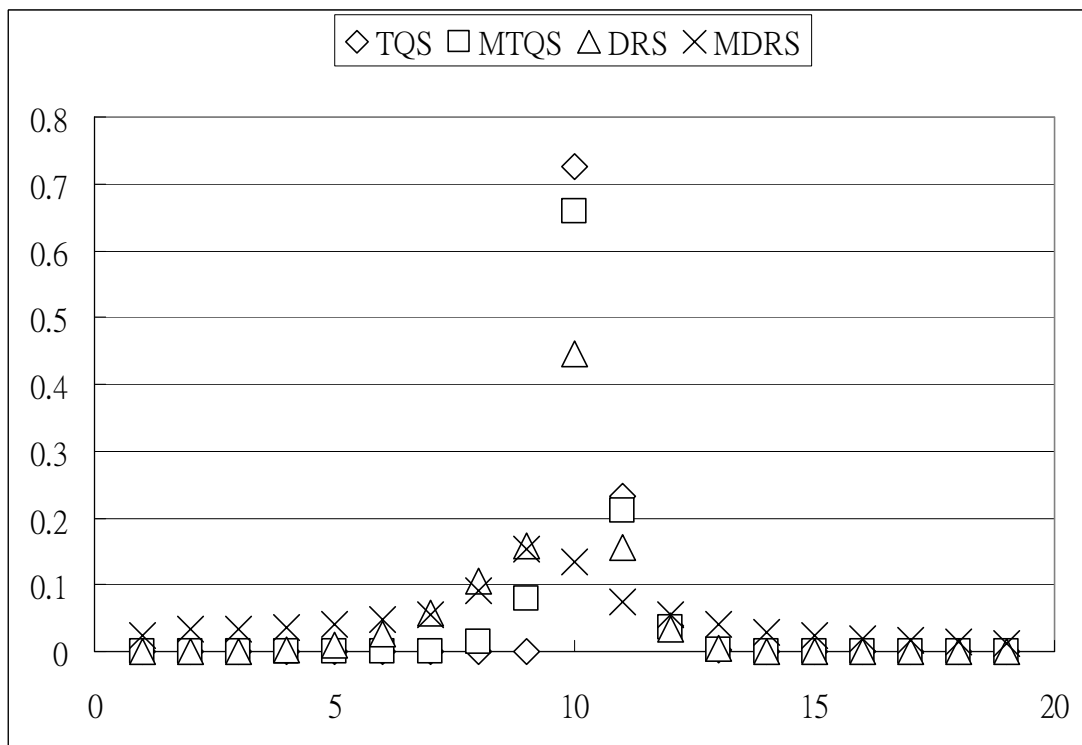


圖 24：第二組模擬狀況三的連線 5

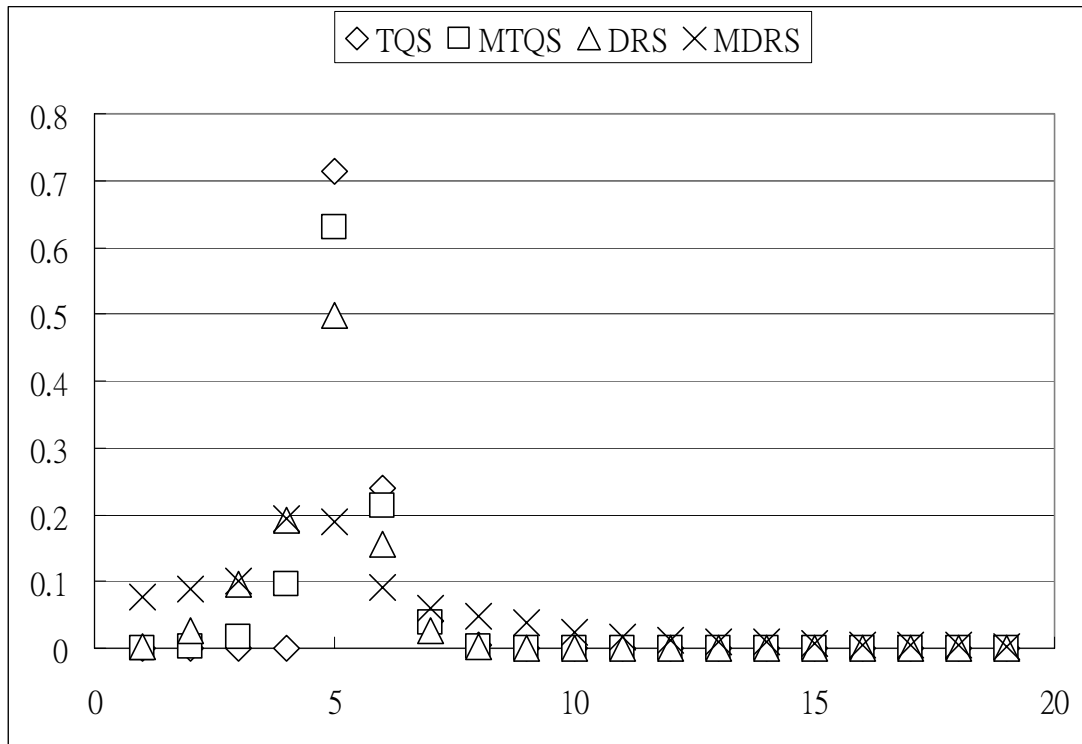


圖 25：第二組模擬狀況三的連線 6

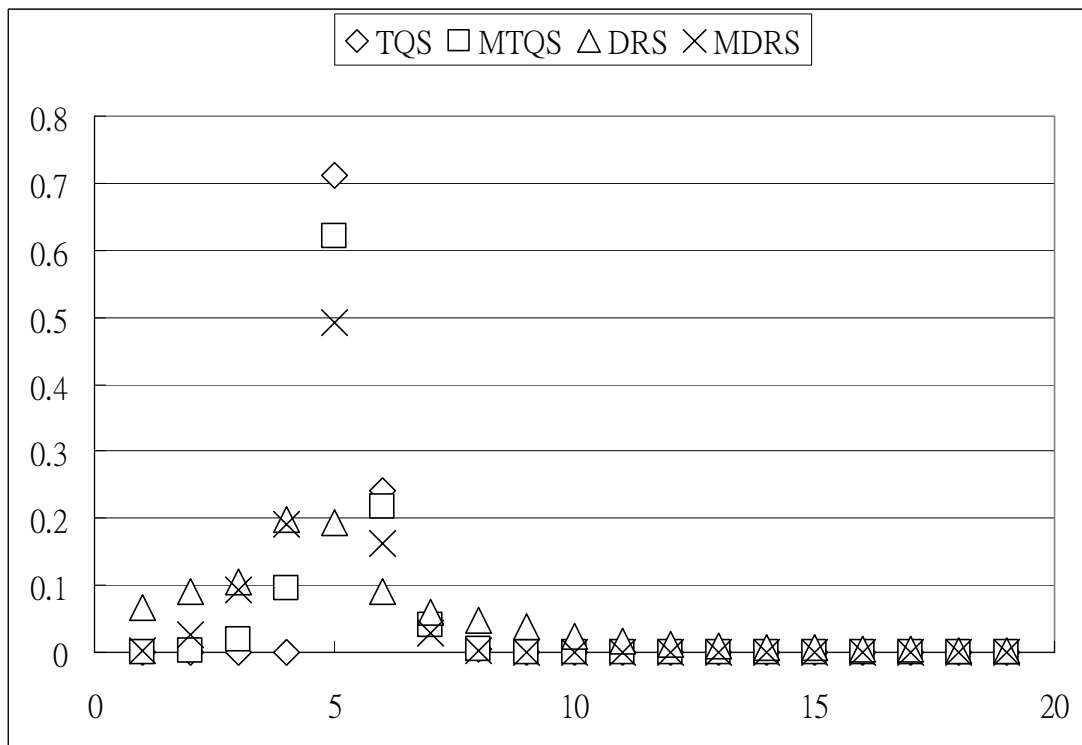


圖 26：第二組模擬狀況三的連線 7

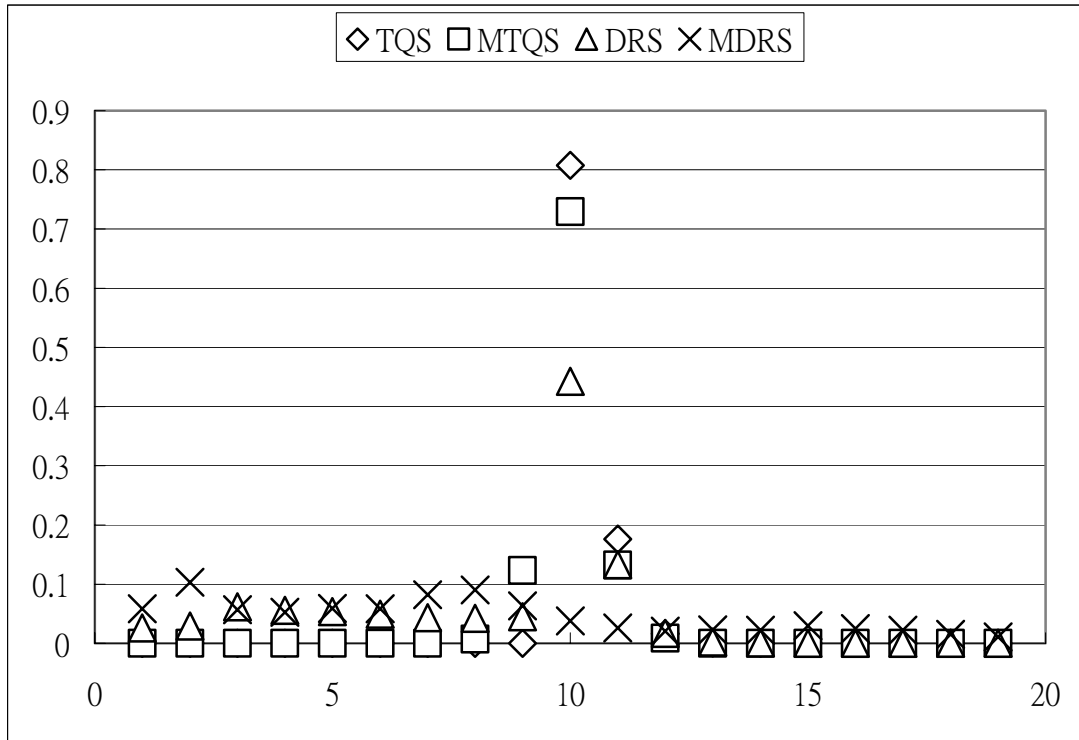


圖 27：第二組模擬狀況四的連線 4

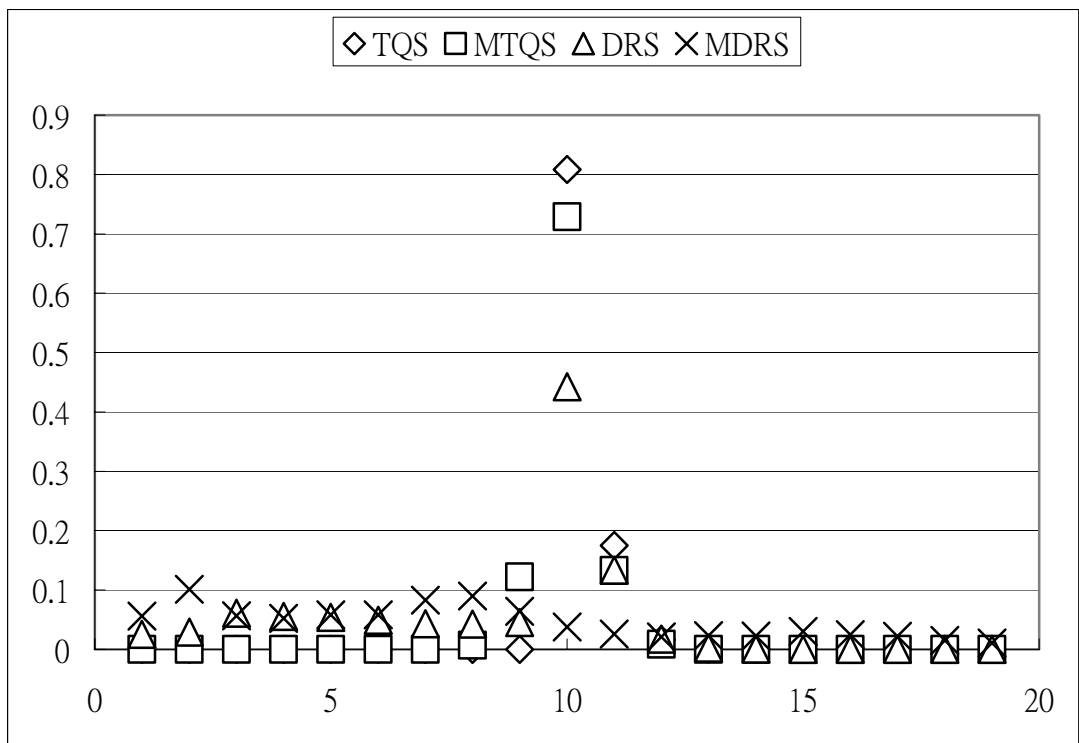


圖 28：第二組模擬狀況四的連線 5

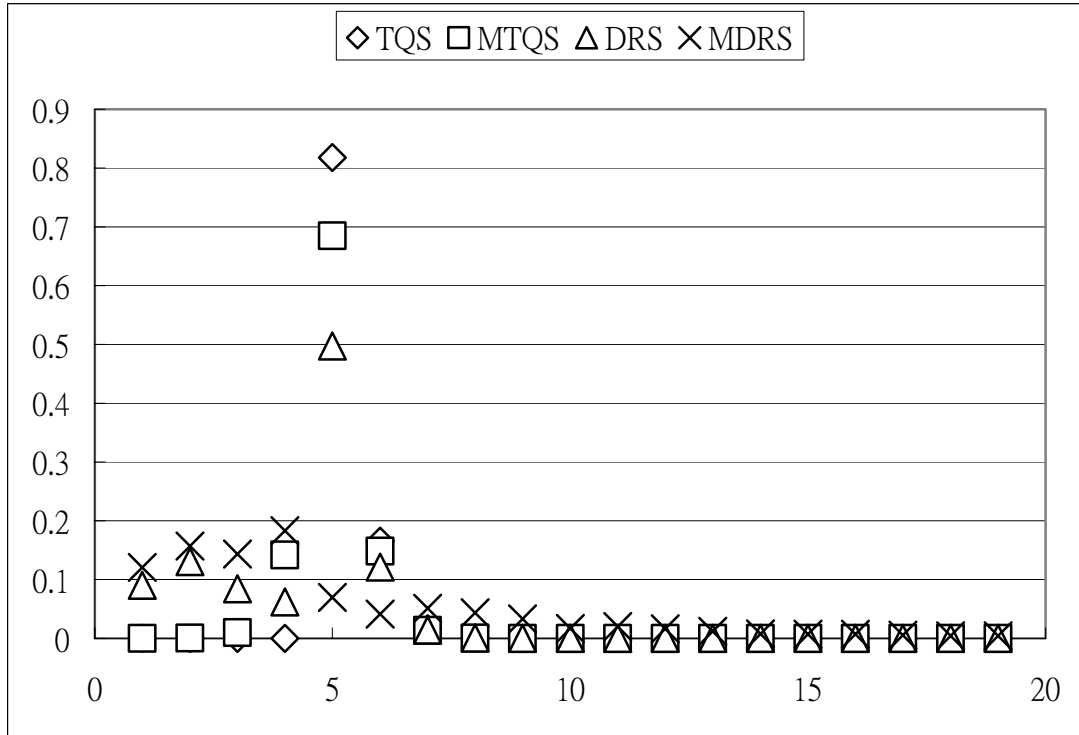


圖 29：第二組模擬狀況四的連線 6

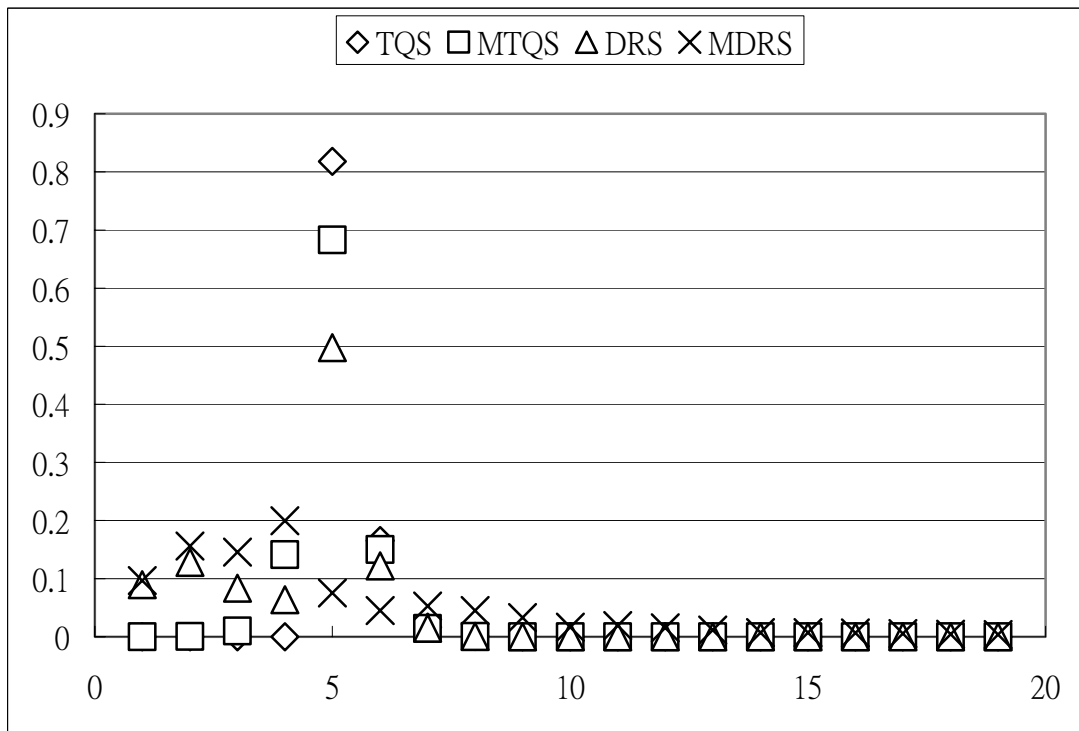


圖 30：第二組模擬狀況四的連線 7

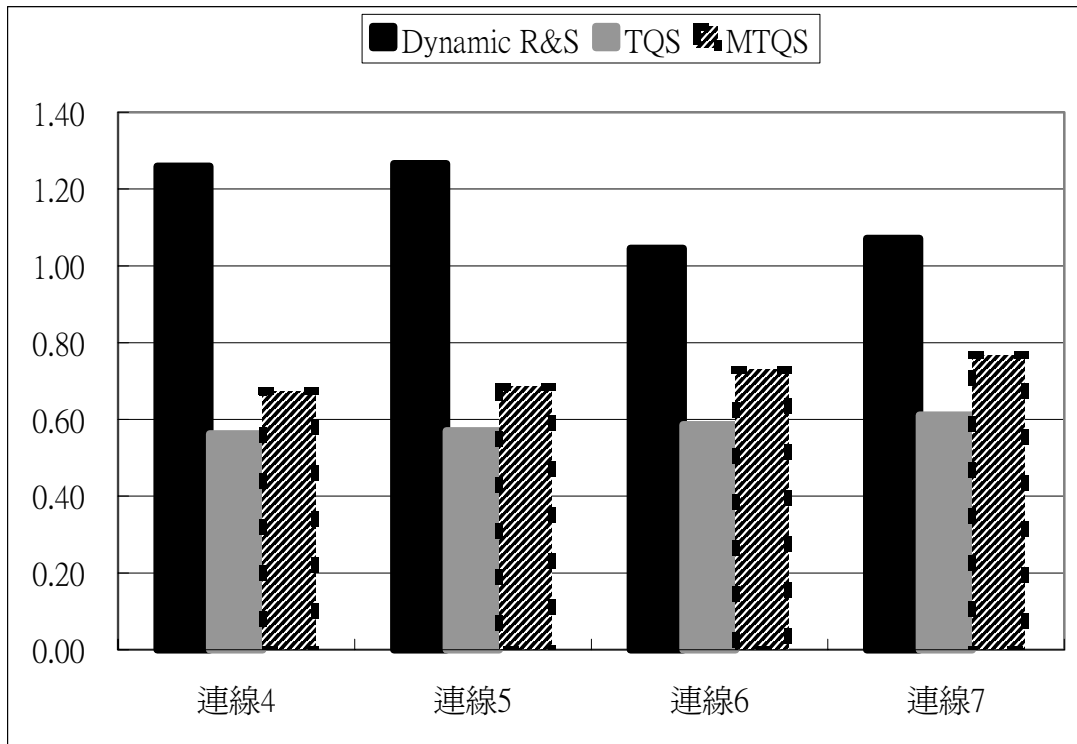


圖 31：第二組模擬狀況一的標準差

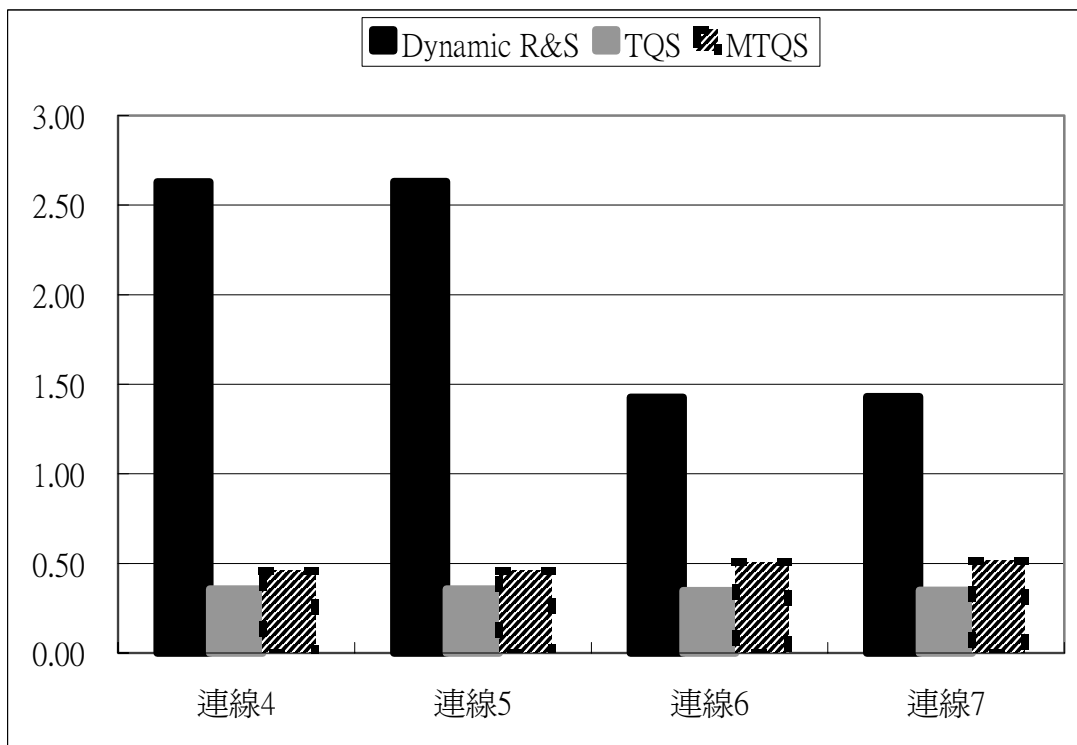


圖 32：第二組模擬狀況二的標準差

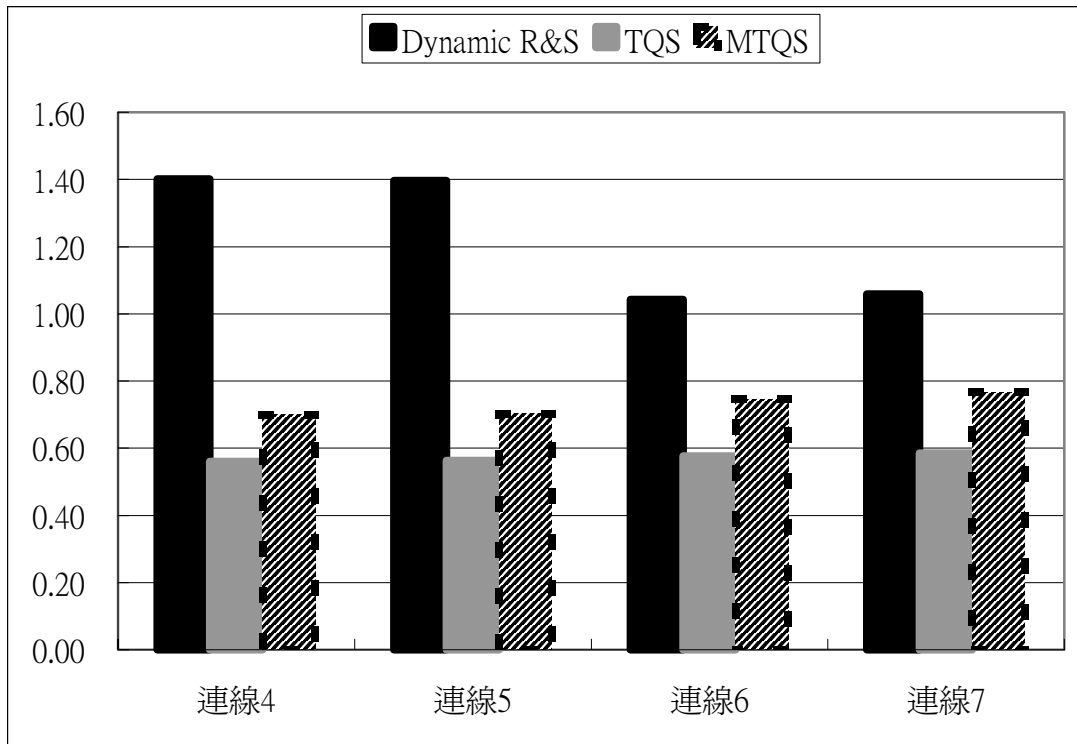


圖 33：第二組模擬狀況三的標準差

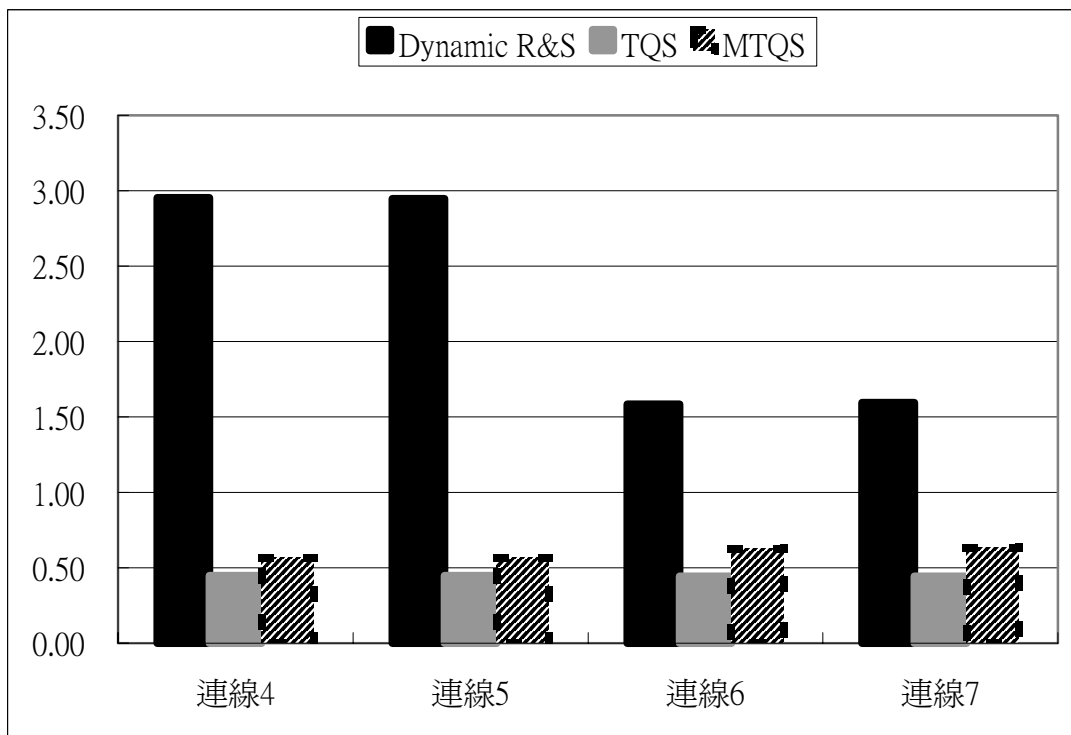


圖 34：第二組模擬狀況四的標準差

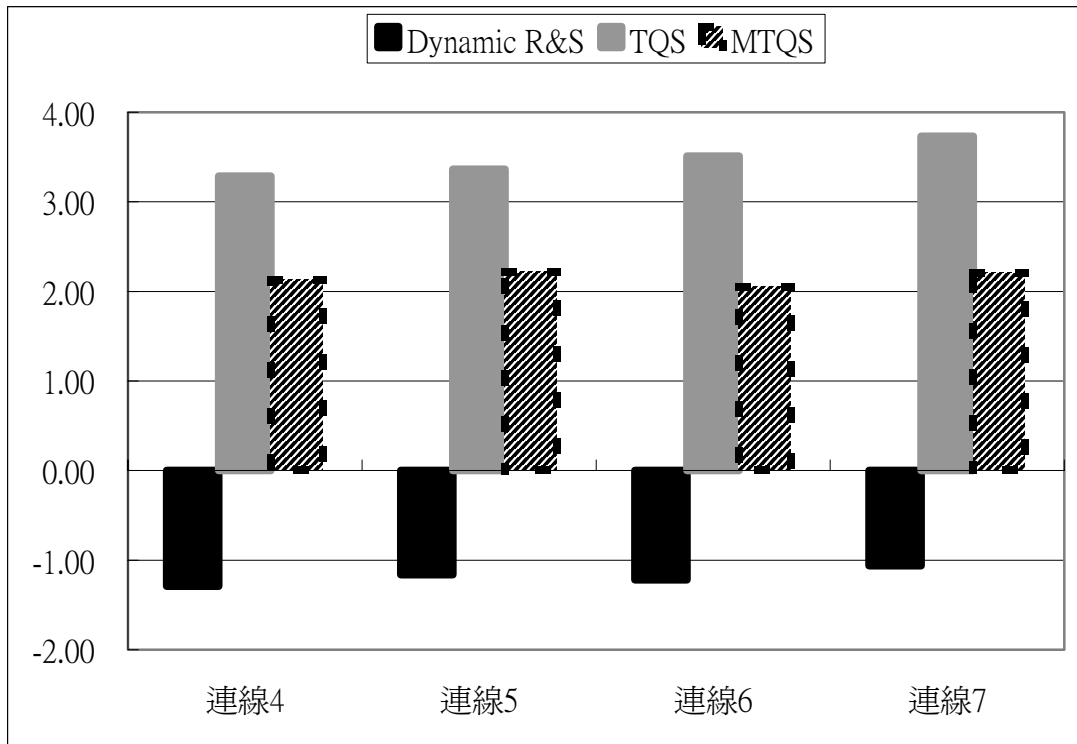


圖 35：第二組模擬狀況一的平均值與  $T'$  的誤差百分比

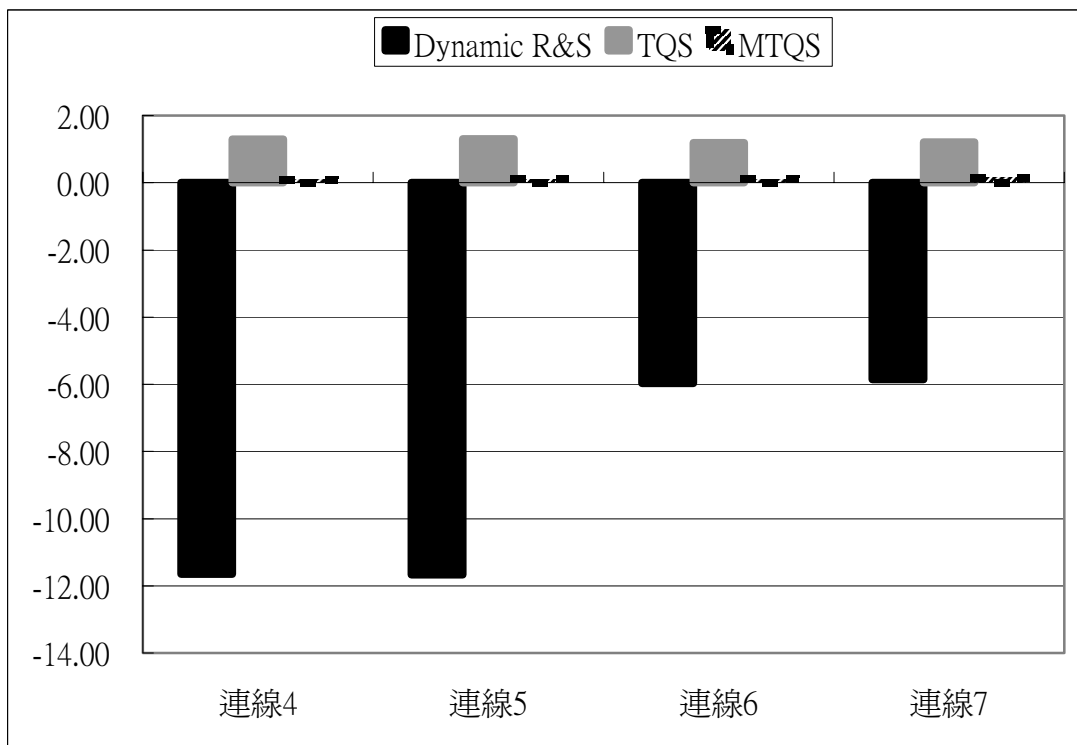


圖 36：第二組模擬狀況二的平均值與  $T'$  的誤差百分比



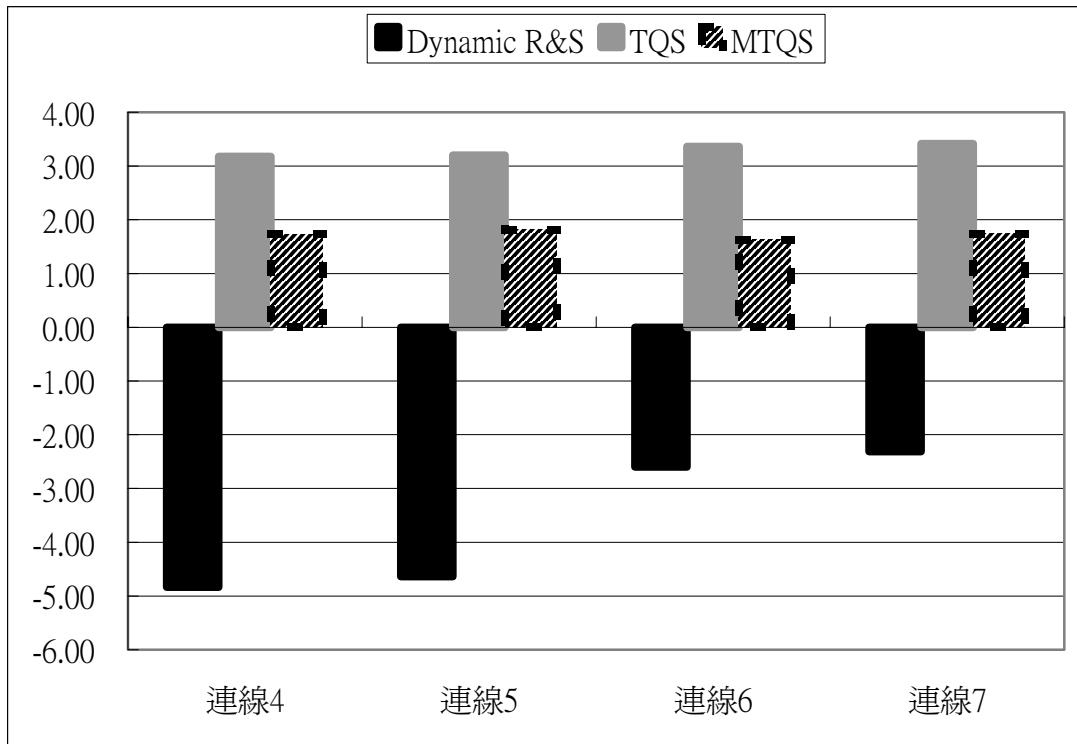


圖 37：第二組模擬狀況三的平均值與  $T'$  的誤差百分比

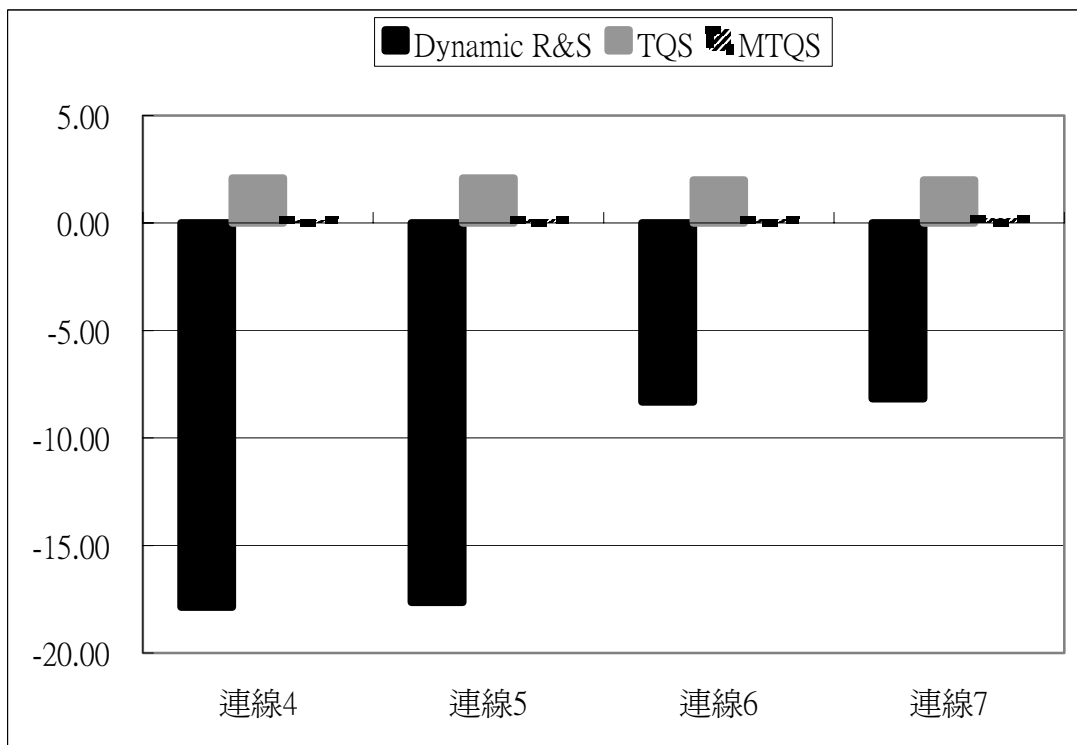


圖 38：第二組模擬狀況四的平均值與  $T'$  的誤差百分比

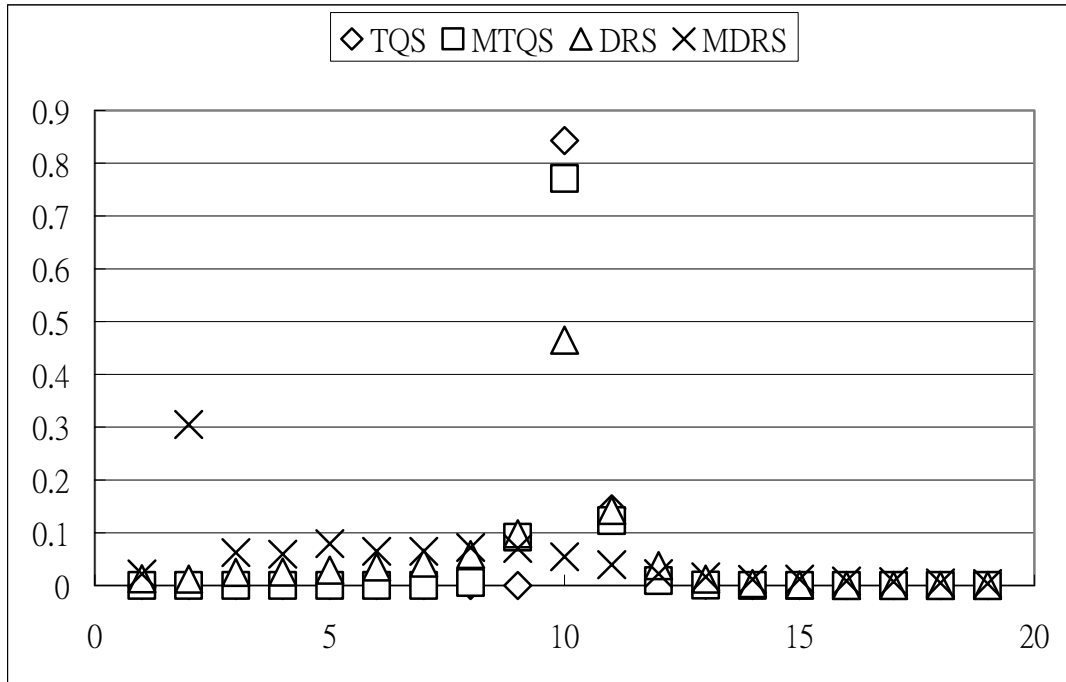


圖 39：第二組模擬狀況五的連線 4

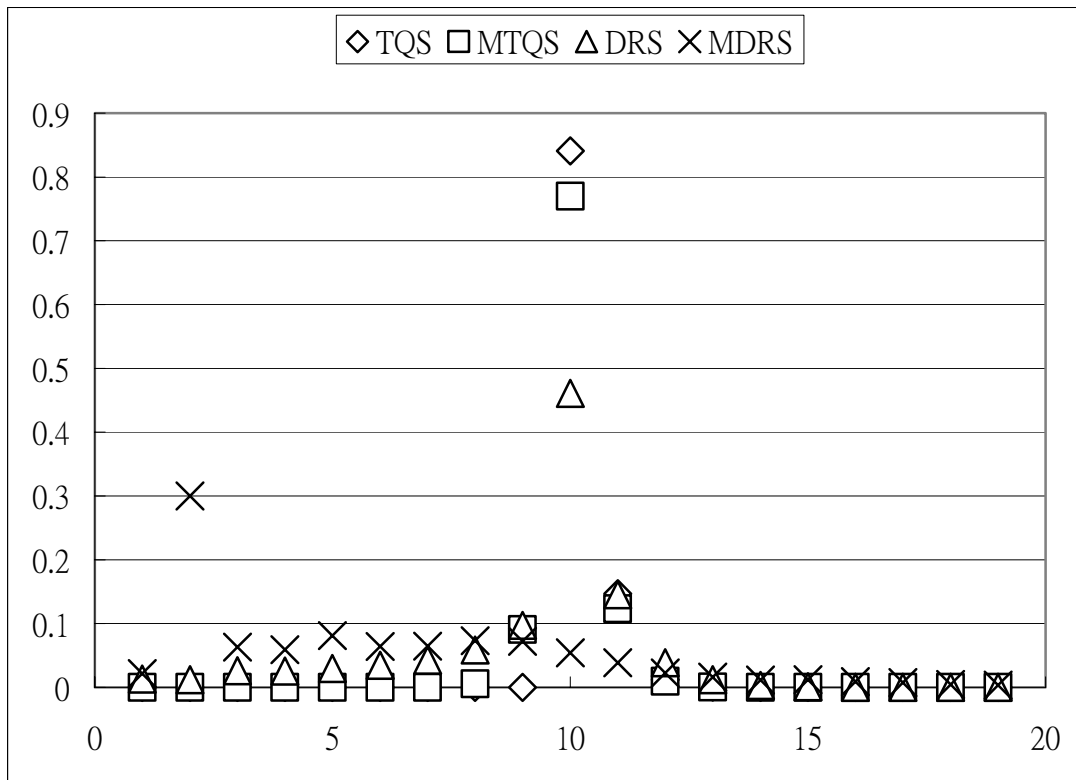


圖 40：第二組模擬狀況五的連線 5

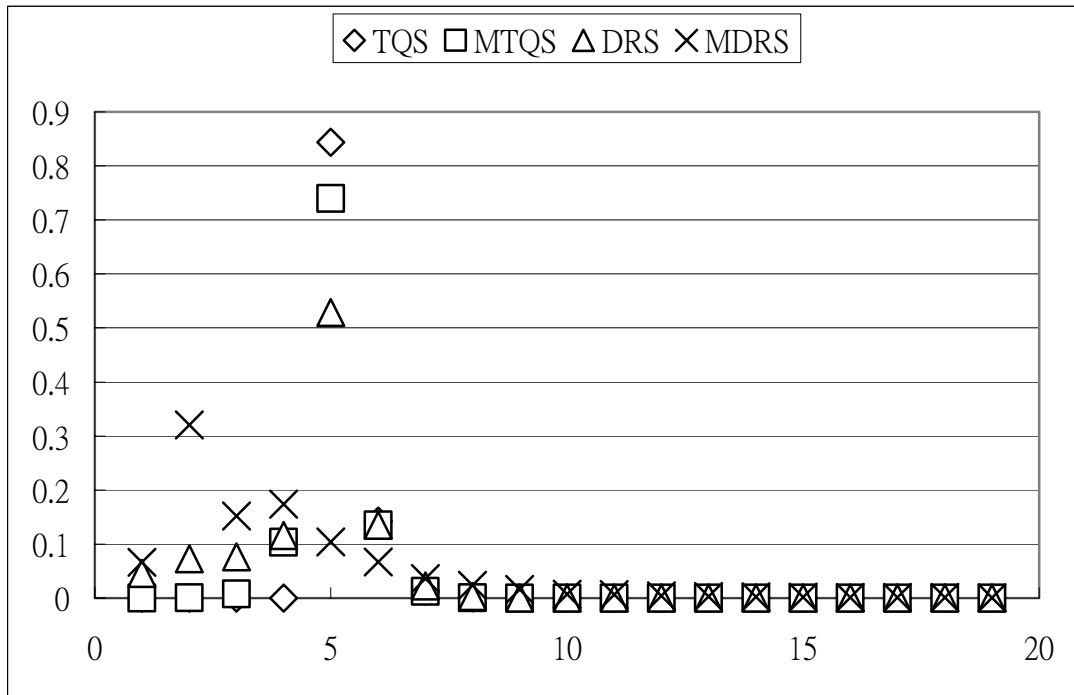


圖 41：第二組模擬狀況五的連線 6

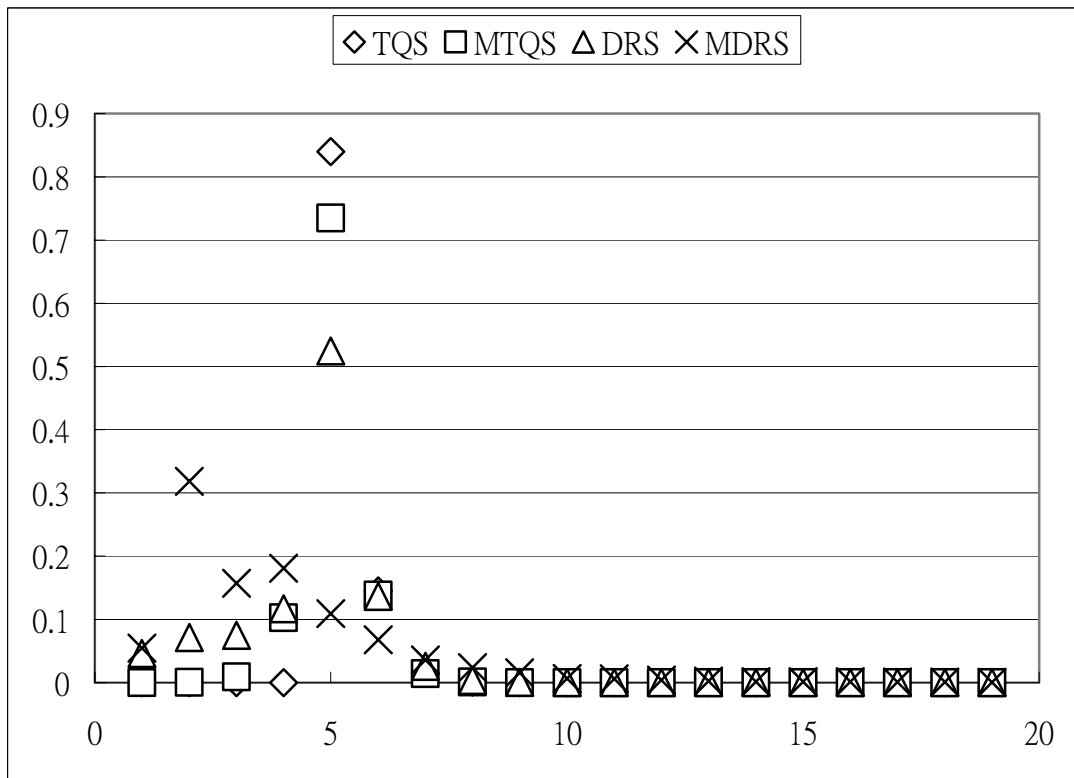


圖 42：第二組模擬狀況五的連線 7

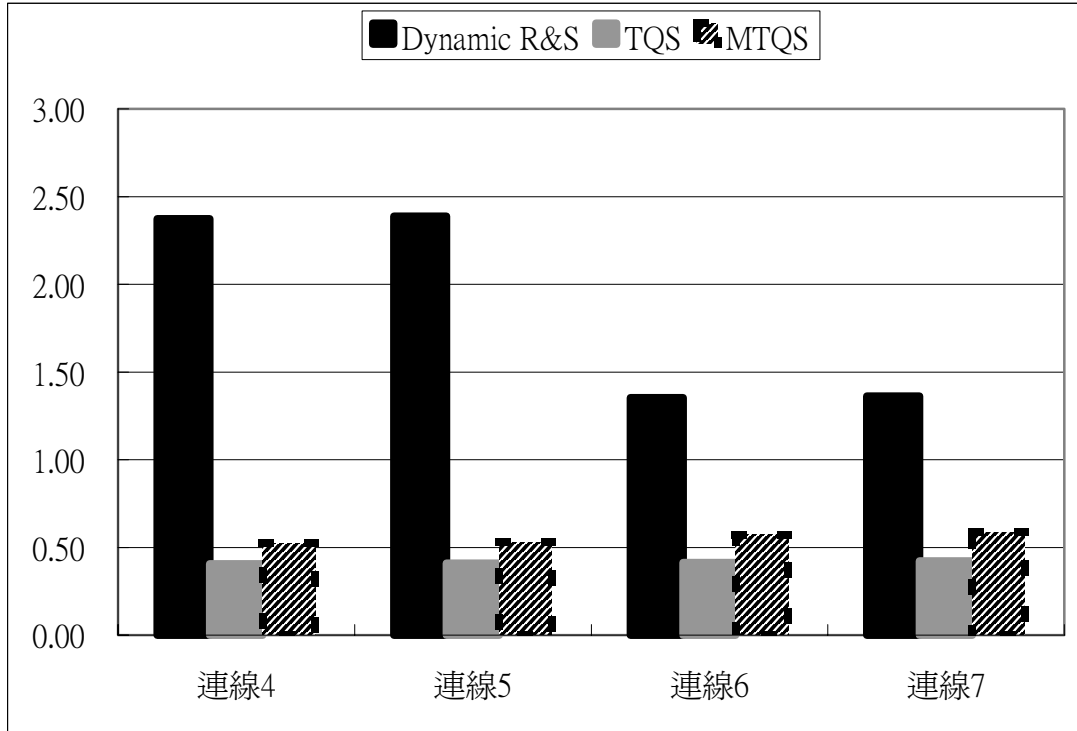


圖 43：第二組模擬狀況五的標準差

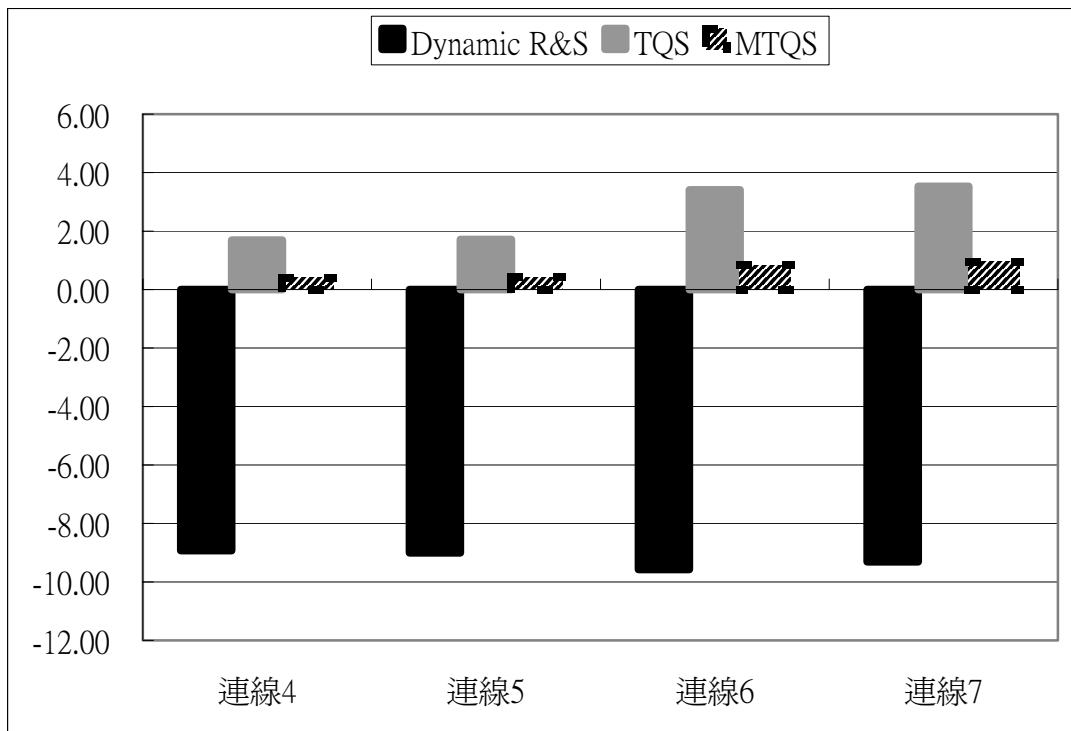


圖 44：第二組模擬狀況五的平均值與  $T'$  的誤差百分比

### 模擬三：MTQS 連線得到的平均頻寬

第三組模擬展示 MTQS 下細胞被傳送的時間之平均值等於  $T^r$ ，為避免前述因為暴衝結束而未能發生伴隨的太早換手的情形，我們讓欲觀察的連線 7 之調整器佇列永遠保持在飽和的狀態，亦即永遠有細胞等待被換手。由於在第二組模擬中，MTQS 下細胞被傳送的時間之平均值與  $T^r$  的誤差最大的情形發生在頻寬使用率較低且暴衝程度較低的情形，做調整參數如表六，使其頻寬使用率接近 0.70，各連線的暴衝仍用 0.1。由於設定連線 7 的調整器佇列為飽和，故其細胞發送端的行為參數沒有意義。

編號 $r$	$T^r$	$\lambda^r$	$a^r$
1	1	0.185	0.1
2	1	0.145	0.1
3	1	0.150	0.1
4	10	0.030	0.1
5	10	0.029	0.1
6	5	0.061	0.1
7	10	/	/

表六：模擬三的參數

本組模擬同樣是執行  $10^8$  個時槽，每個連線都套用同樣的演算法做模擬，在表七中列出各種演算法的結果，可看出 MTQS 得到的平均頻寬為  $\frac{1}{T^r}$ ，其他演算法的結果一併列出做為參考。綜合第二組及三組模擬的結果，我們可以看出 TQS 和 MTQS 下，標定連線的細胞被傳送的時間之標準差較 Dynamic R&S 低約 30% 到 80% 不等，且平均值更接近  $T^r$ ，TQS 和 MTQS 下標定連線  $r$  的細胞被傳送的

間隔集中在  $T^r$  附近的情形較 Dynamic R&S 為佳，TQS 和 MTQS 比 Dynamic R&S 更能提供標定連線接近  $\frac{1}{T^r}$  的穩定傳送速率。

	標準差	平均值	傳送細胞
Dynamic R&S	1.29	9.88	10,120,805
TQS	0.62	10.39	9,627,965
MTQS	0.87	10.00	10,000,000
MDRS	3.66	6.48	15,424,095

表七：模擬三連線 7 的細胞被傳送間隔

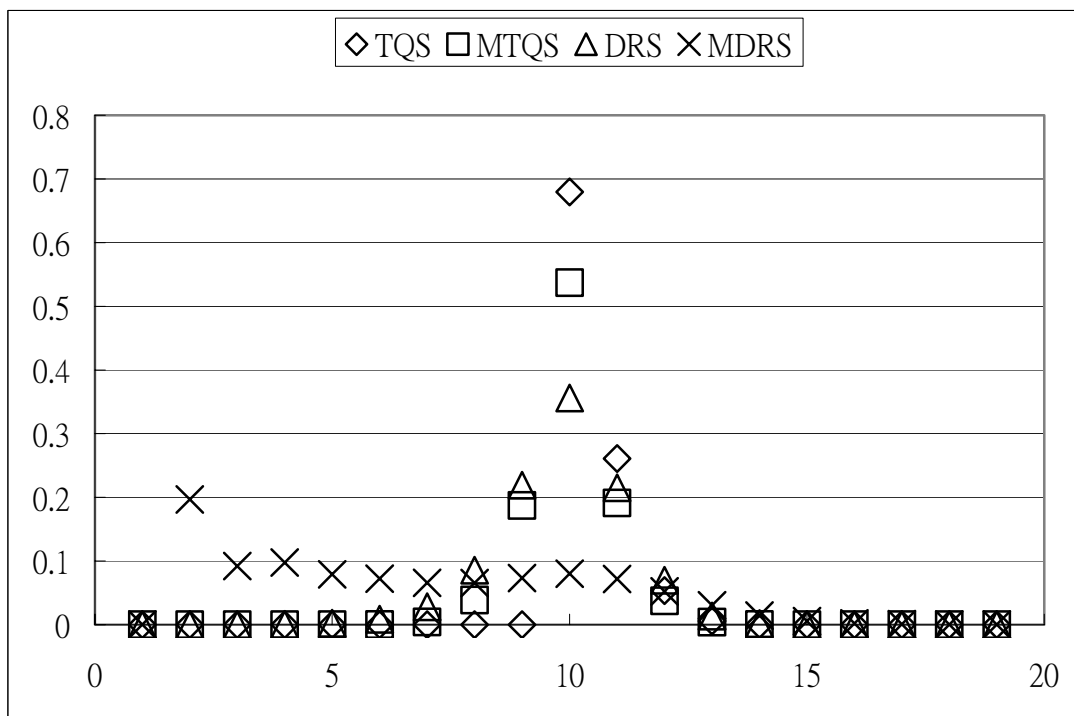


圖 45：模擬三的連線 7

## 模擬四：CATQS 連線得到的平均頻寬

第四組模擬展示 CATQS 下的一些例子，各連線的設定和模擬二的第一個狀況相同。我們設定加速的門檻值為  $\sum \frac{1}{T^r} > x$ ，加速的方式為將所有的  $T^r$  都暫時減少  $y$ ，在圖中記為  $x-y$ 。我們可以將加速機制設定為許多階段，如在圖中我們記為  $x_1 - y_1 / x_2 - y_2$ ，其中  $x_1 < x_2$  且  $y_1 < y_2$  的狀況下，意為當  $\sum \frac{1}{T^r} < x_1$  時不加速，當  $x_1 \leq \sum \frac{1}{T^r} < x_2$  時將所有的  $T^r$  都暫時減少  $y_1$ ，當  $\sum \frac{1}{T^r} \geq x_2$  時將所有的  $T^r$  都暫時減少  $y_2$ 。以下圖示皆為連線 4 的結果。

在模擬四的第一個狀況，我們將加速方式設定為將所有的  $T^r$  都暫時減少 1，結果示於圖 46 中。我們可以看出門檻值越小，被加速(即細胞被傳送的時間為  $T^r - 1$ )的機率越高，太晚換手的機率也越小，但沒有失誤的機率也被降低了；傳送門檻值越大時，其 PMF 越接近原來未加速的 TQS。

在模擬四的第二個狀況，我們將加速方式設定為將所有的  $T^r$  都暫時減少 2，結果示於圖 47 中。由於細胞以兩種速率被傳送，一是  $\frac{1}{T^r}$ ，一是  $\frac{1}{T^r - 2}$ ，因此在圖 47 中顯示出兩個峰值，一是  $T^r$ ，一是  $T^r - 1$ 。

在模擬四的第三個狀況，我們將加速方式設定為兩階段，其中第一階段為當  $1.8 \leq \sum \frac{1}{T^r} < 2.3$  時，將所有的  $T^r$  都暫時減少 1，第二階段為當  $\sum \frac{1}{T^r} \geq 2.3$  時，將所有的  $T^r$  都暫時減少 2。在此種加速的方式下，沒有失誤的機率雖較 TQS 低但仍高於 Dynamic R&S，太晚換手的機率則明顯小於 Dynamic R&S。

由第四組模擬可以看出，善用此加速的機制，可以減少太晚換手的機率，但是沒有失誤的機率也跟著被犧牲掉一部分。多階段的加速機制可以使太晚換手的機率更為降低，可將細胞被傳送的時間控制成只有一個峰值。

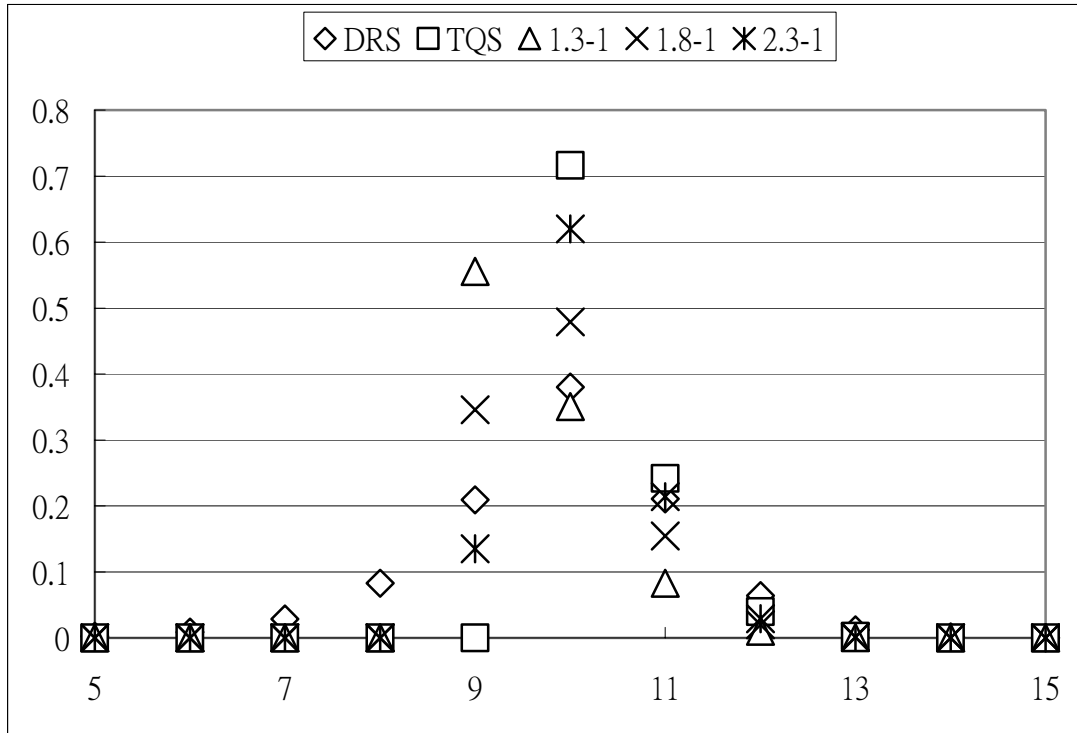


圖 46：設定 1 組加速參數，將連線的  $T'$  暫時減少 1

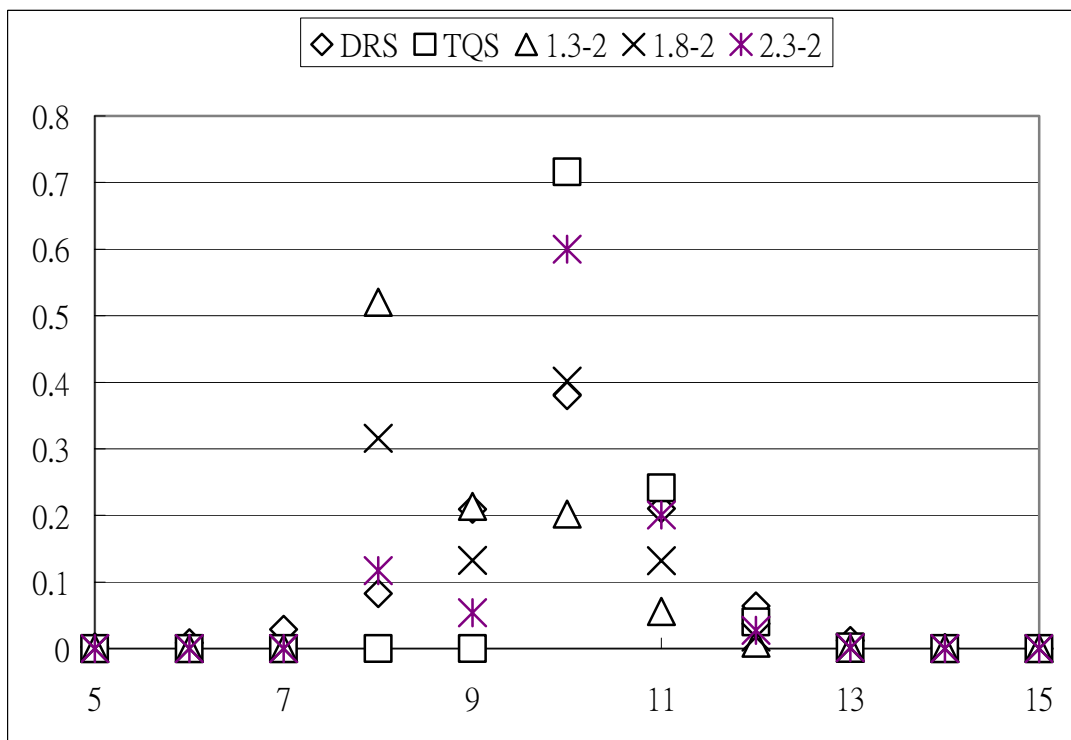


圖 47：設定 2 組加速參數，將連線的  $T'$  暫時減少 2



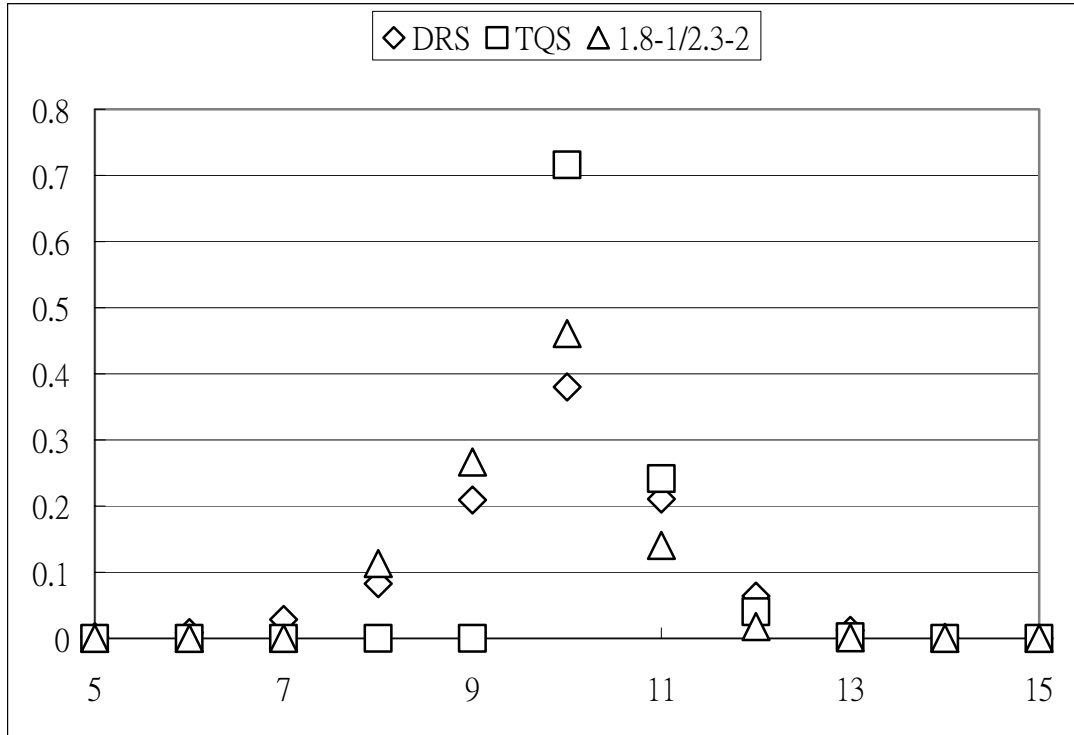


圖 48：設定 2 組加速參數

## 第六章：總結及未來展望

我們提出的 TQS、MTQS 及 CATQS 機制，在調整機制內加入考慮排程器佇列長度的考量，有效集中了標定連線的細胞間被傳送的時間之分布，進而改善了標定連線的速率變化情形。

最基本的 TQS 以「系統時間與排程器佇列長度的和」為判斷連線合格 (eligible) 的條件，每次連線合格且將其調整器佇列的頭端細胞換手後，以該細胞換手的實際情形加上「我們所希望得到的細胞被傳送的時間」來維護各連線的合格條件。

而由於 TQS 以各連線換手頭端細胞的實際情形維護合格條件，故得到的平均頻寬小於被希望得到的細胞被傳送的時間的倒數，因此我們在 MTQS 中，將 TQS 以實際情形維護合格條件改為以預期情形維護合格條件

在即時流量 (real-time traffic) 的應用下，我們可能較不希望看到太晚換手的情形，而在 TQS 由於各連線的頭端細胞都在恰好的狀態下加入競爭，一但競爭未成為最勝者，就會有太晚換手的情形，故我們用 CATQS，在競爭劇烈時，將各連線的頭端細胞都提早加入競爭，也就是在維護各連線的合格條件時，加上一個比「我們所希望得到的細胞被傳送的時間」較小的值，如此一來，即便競爭未成為最勝者，也未必會發生太晚換手。

在模擬中可看到 TQS 和 MTQS 下，標定連線的細胞被傳送的時間之標準差較 Dynamic R&S 低約 30%到 80%不等，且平均值更接近我們期望的時間。TQS 和 MTQS 下標定連線  $r$  的細胞被傳送的時間集中在我們期望的細胞被傳送時間附近的情形較 Dynamic R&S 為佳，TQS 和 MTQS 更能提供標定連線穩定且接近我們期望的值的傳送速率。

目前 TQS 及 MTQS 只有排程機制，尚有連線允入控制的問題有待解決，以及在端對端的行為尚未了解，這些都是目前我們正在努力的目標。

關於將 TQS 及 MTQS 推展到處理不固定封包大小的狀況，目前我們的想法是將系統時間、佇列長度及合格檢查的單位由時槽改爲一個最小的封包大小計量單位，如位元組；將排程器佇列長度改爲其實際長度減去正在傳送的封包已經傳送出去的部分，並將原本期望的同連線兩個細胞傳送間隔改爲同連線兩封包開始傳送的間隔除以兩封包的前者的長度。這一部分的機制尚未完整，亦正在努力。

## 參考文獻

- [1] ATM Forum, *Traffic Management Specification Version 4.1*. March 1999.  
<ftp://ftp.atmforum.com/pub/approved-specs/af-tm-0121.000.pdf>
- [2] S. J. Golestani, “Congestion-free communication in high-speed packet networks,” *IEEE Trans. Networking*, vol. 39, pp. 1801–1812, December 1991.
- [3] H. L. Pocher, V. C. M. Leung, and D. W. Gillies, “An efficient ATM voice service with flexible jitter and delay guarantees,” *IEEE J. Select. Areas Commun.*, vol. 17, pp. 51–62, January 1999.
- [4] F.-M. Tsou, H.-B. Chiou, and Z. Tsai, “Design and simulation of an efficient real-time traffic scheduler with jitter and delay guarantees,” *IEEE Trans. Multimedia*, vol. 2, pp. 255–266, December 2000.
- [5] A. K. Parekh and R. G. Gallager, “A generalized processor sharing approach to flow control in integrated services networks: The single-node case,” *IEEE/ACM Trans. Networking*, vol. 1, pp. 344–357, June 1993.
- [6] D. Ferrari and D. Verma, “A scheme for real-time channel establishment in wide-area networks,” *IEEE J. Select. Areas Commun.*, vol. 8, pp. 368–379, April 1990.
- [7] D. Verma, H. Zhang, and D. Ferrari, “Guaranteeing delay jitter bounds in packet switching networks,” in *Proc. Tricomm '91*, Chapel Hill, NC, April 1991, pp. 35–46.
- [8] C. L. Liu and J. W. Layland, “Scheduling algorithms for multiprogramming in a hard-real-time environment,” *J. ACM*, vol. 20, no. 1, pp. 46–61, January 1973.
- [9] C. R. Kalmanek, H. Kanakia, and S. Keshav, “Rate controlled servers for very high-speed networks,” in *Proc. IEEE Global Telecommunication Conf.*, San

- Diego, CA, December 1990, pp. 300.3.1–300.3.9.
- [10] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis, “Weighted round-robin cell multiplexing in a general-purpose ATM switch chip,” *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1265–1279, October 1991.
- [11] M. Shreedhar and G. Varghese, “Efficient fair queuing using deficit round-robin,” *IEEE/ACM Trans. Networking*, vol. 4, pp. 375–385, June 1996.
- [12] C.-S. Wu, J.-C. Jiau, and K.-J. Chen, “Providing traffic smoothness and service guarantees within ATM networks through a rate-controlled service discipline,” *Comput. Networks and ISDN System*, vol. 29, pp. 2119–2138, 1998.
- [13] B. Bensaou, D. H. K. Tsang, and K. T. Chan, “Credit-based fair queueing (CBFQ): A simple service-scheduling algorithm for packet-switched networks,” *IEEE/ACM Trans. Networking*, vol. 9, pp. 591–604, October 2001.
- [14] J. Liebeherr, D. E. Wrege, and D. Ferrari, “Exact admission control for networks with a bounded delay service,” *IEEE/ACM Trans. Networking*, vol. 4, pp. 885–901, December 1996.
- [15] S. Iatrou and I. Stavrakakis, “A dynamic regulation and scheduling scheme for real-time traffic management,” *IEEE/ACM Trans. Networking*, vol. 8, pp. 60-70, February 2000.
- [16] H. Zhang and D. Ferrari, “Rate-controlled static-priority queueing,” in *Proc. IEEE INFOCOM '93*, September 1993, pp. 227-236.
- [17] 石翔任, “一個簡易的利用排程器狀態資訊的動態網路流量調整機制,” M.S. thesis, 國立中山大學.
- [18] D. E. McDysan and D. L. Spohn, *ATM theory and applications*. McGraw-Hill, September 1994.