

# 第一章 導論

## 1.1 研究動機

無線所帶來的最大的優點，莫過於提供使用者較高的方便性。在基於需求的考量下，無線的發展是一個必然的趨勢，尤其在網際網路及寬頻網路發展日趨成熟的今天，無線網路的發展將成為下一波的重要的技術[1][2]。其中最為我們熟知的是無線區域網路。無線區域網路(IEEE 802.11)規格的制定是 97 年開始[3]。近幾年來，經過修改，已經大致完備。有些廠商已經著手產品的研發和製造。在面臨無線與有線技術整合的趨勢下，針對是否能在無線環境中依然順利使用網際網路，以及如何在無線環境中增加效能都是需要研究的重要問題。

## 1.2 在無線環境下提升 TCP 效能的方法

在無線環境下，由於位元錯誤率較高，使得封包遺失率相對的比較高，在這種環境下，對 TCP 而言，不僅僅是封包遺失，TCP 會將封包遺失誤以為是網路壅塞，壅塞控制(Congestion control)便開始啟動，降低傳送的速率，使 TCP 效能嚴重下降。針對這個現象，在學術界提出了相當多改進的方法，來改善 TCP 的效能。綜合各種方法，大致可以分為兩

類，一種是在 TCP 層進行改善，另一種是在資料鏈結層來進行。

我們由前人研究知[4][5][6][7]，在位元錯誤率較低的情形下，這些方法的確都會對 TCP 效能有所改進。但是，在高位元錯誤率的環境，這些在 TCP 層的改善方法並不能真正的改善效能。這是由於這些方法都是在 TCP 層運作，屬於 end-to-end 的機制，反應時間上比較慢，必須將訊息傳回傳送端才能做出反應，因此錯誤更正能力有一個上限。

在資料鏈結層加入錯誤重傳的方法，來改善 TCP 的效能，經由模擬證明，在 end-to-end delay 較大且在高位元錯誤率的環境下，使用這種方式仍能改善 TCP 的效能，而且使用這種方法最大的優點是能夠局部的重傳遺失的封包，對於封包遺失的反應時間上相對的比較快，且並不更動現有 TCP 的機制，不過此類方法並不能禁止壅塞控制的啟動，因此我們預估在錯誤更正能力上，會存在一個上限。但由於在無線鏈結上傳送一個封包來回的時間相對比較小時(相對於 End-to-End)，可以明顯看到此法的效果。

### **1.2.1 於傳輸層層進行改進的方法**

在無線環境高位元錯誤率的影響下，TCP 效能將受到很嚴重的影響。近年來，頗多的研究都在更動 TCP 的機制，以適應無線環境。這類方法在實作上屬於端對端(End-to-End)的機制。

#### **1.2.1.1 Snoop Protocol**

網路架構如圖 1.1，Snoop Protocol[4]主要是在無線接取點端 (Access Point) 設立一個監測裝置(Snoop Agent)，將網際網路伺服器端送出的資料先保留在記憶緩衝區中，對於由無線使用者端送回的 ACK 封包做監測，判斷封包是否在無線鏈結上遺失了。若封包遺失了，便局部的重傳遺失的封包，若由送回的 ACK 封包判斷封包已經成功抵達，便在記憶體中清除資料。使用這種方法的好處是它並未改變原本 TCP，只是額外的加上改善的方法，而且可以局部端重傳未送達的封包。

不過這個方法有些缺點，它只能針對下行資料的傳送做保障，而對於上行資料的傳送此法並不能改善效能，而且這個方法主要針對的是 TCP，因為他是經由 ACK 封包來判斷封包是否遺失，若要用在非 TCP 的資料傳送，只能採用其他方式來增加效能，不過對於上行的資料流仍是無法改善效能。

### 1.2.1.2 ELN 法:

在傳輸層(Layer 4)的另一個改善的方法，是加入”ELN”bit[4]，觀念上是將由位元錯誤率所造成的封包遺失區隔出來，避免 TCP 在重傳遺失封包時啟動壅塞控制。實作上，主要在無線接取點端來做改變以及對現行 TCP 的機制修改，假設位在網際網路上的伺服器建立一條 TCP 至無線網路上的使用者端，如圖 1.1，步驟如下：

- (1) 在無線接取點端設立一個監測裝置，觀察由無線端傳回的 ACK 封包，若 ACK 重複傳回而且指向同一封包，則表示有封包遺失在無線這端，而且是由於位元錯誤率所造成。

- (2) 若顯示有封包遺失是由於位元錯誤率，便將傳回傳送端的 ACK 封包上加入'ELN' 位元。
- (3) 當 sender 端收到封包(ACK)，而其中含有 ELN 位元，TCP 將不啟動壅塞控制，不減低傳送資料的速率，只是將資料重送。

使用這種方法可以有效的提升效能，主要是將封包遺失的原因區格出來，TCP 傳送端並不啟動壅塞控制，不過此法尚有幾個缺點。此法必須對現行 TCP 的機制進行修改，在實作上是一件比較麻煩的事。此法只能善 TCP 的效能，對於 UDP 或其他類型的封包並不能改善效能。就錯誤更正能力而言，此法是屬端對端的方法，在端對端傳輸時間比較長時，錯誤更正能力上仍會有所限制。

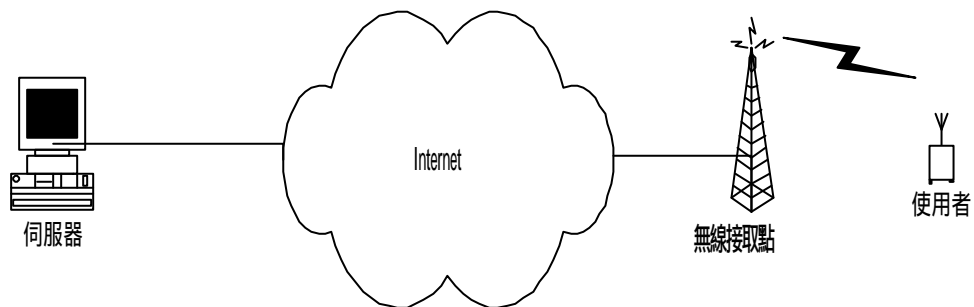


圖 1.1 Snoop Protocol 及 ELN bit 法網路架構示意圖  
：網際網路上的伺服器經由無線接取點連接至無線網路上的使用者。

### 1.2.2 於資料鏈結層進行改善的方法

在現行的網際網路中，由於封包遺失的原因主要是由於網路的壅塞，因此在資料鏈結層並沒有錯誤重傳的功能，但在無線環境中由於位元錯誤率比較高，因此在資料鏈結層執行重傳遺失的封包有其必要性，此法主要是當資料傳出時，必須收到接送端傳回 ACK(資料鏈結層)才能再傳下一個資料，好處是可以局部重傳遺失的封包，而且並不更動現行 TCP 的機制，此外 UDP 等其他類型的封包亦可使用這種方法來改善效能，不過此種方法仍有些缺點：

- (1) 封包遺失率較大時，重傳次數也增加，相對的傳送一個封包的時間也相對增加，對於目前網際網路上一些有關即時(Real-time)的服務，便需在封包遺失率和延遲(delay)上做取捨。
- (2) 對於 TCP 而言此法並非避免壅塞控制的啟動，只在局部端重傳遺失封包，因此在錯誤更正能力上，仍會存在一個上限。不過一般而言，局部重傳遺失的封包所花費的時間並不多(因為是 LAN 的環境，相對於 End-to-End 並不算長)，因此仍有可觀的效能。

在資料鏈結層進行改善的方法，其基本的觀念是在資料鏈結層加上錯誤重傳的概念，由這觀念下所衍生的方法有相當多[5]，Select repeat、RLP(Radio Link Protocol)..，還有其他在資料鏈結層類似如 TCP 的機制，它最大的好處是可以局部重傳遺失的封包。在網路中的無線的鏈結上，無論是上行或是下行皆可有效的增進效能。

### 1.3 無線區域網路概述

無線區域網路最為大家熟知的是 IEEE802.11[3]，802.11 與傳統有線網路中的乙太網路相似，最大的不同是使用了 CSMA/CA，很多人一定會問，為何不用 CSMA/CD 而是 CA(Carrier avoidance)，主要是在無線的介質中，無法對傳送在無線通道上的信號去偵測。除了網路協定上制定規格的不同外，最大的不同是無線通道錯誤率的變化性及其對傳輸效能的影響。在標準中所及的並非我們所重視的地方，我們著重的是無線通道的錯誤率變化，以及這些因素對於目前網路中傳輸所造成的影響。

在封包切割與封裝方面，如圖 1.2，IP 封包傳送至資料鏈結層，經由封包切割成若干無線封包，並加上標頭傳送至無線網路中，當其中一個無線封包遺失，便會造成整個 IP 封包的遺失，在本論文中我們假設一個 IP 封包恰好可封裝成一個無線封包，且將標頭忽略不計。我們會做這個假設主要是無線區域網路存在相當多的規格，封裝一個無線封包的方式也有所不同，在不同的規格和標準之下，僅數學換算的不同而已，並不影響我們這篇論文的討論。

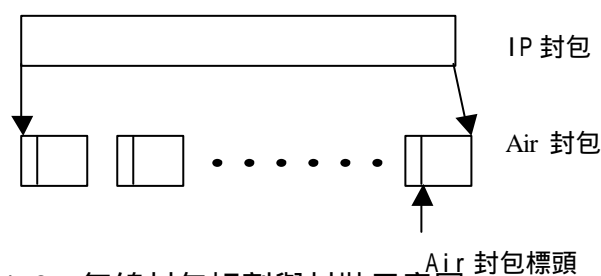
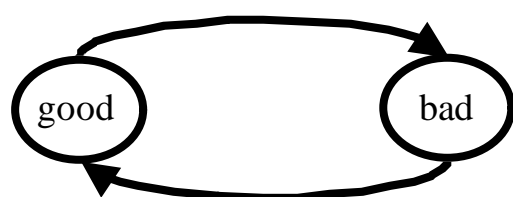


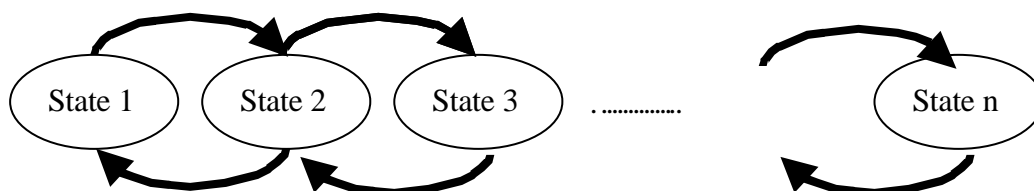
圖 1.2 無線封包切割與封裝示意圖

### 1.3.1 無線通道的特性與模型

無線通道的品質由於多路徑干擾的效應，位元錯誤率不僅比一般無線環境來的大，位元錯誤率更會隨著時間而改變。經由研究發現，可以用有限狀態馬可夫模型(finite state Markovian model)來模擬通道的特性[8][9]。其中，每一個狀態對應一個封包遺失率，停留在每一個狀態的時間是一個隨機變數，通道中的位元錯誤率便隨著這樣的模型來變化，如圖 1.3 的二元通道模型來說，當鏈結(Link)處於”good”的狀態時，封包遺失率較低，封包大部份可以成功的傳送，有較佳的效能；當鏈結處於”bad”的狀態時，封包遺失率較高，傳輸時會造成大量的封包遺失。我們在這篇論文中所討論的重點是針對封包遺失率的變化對傳輸效能的影響以及如何增進效能的方法，在這篇論文中採用了兩種狀態的變化來做為通道的模型。



二元狀態變化的通道模型



多元狀態通道模型

圖 1.3 通道模型

### 1.3.2 無線區域網路通道特性與影響

在無線區域網路中，除了擁有無線通道的特性外，到達每個使用者鏈結上的通道錯誤率變化又各自獨立。在這種複雜的環境下，若不依據網路狀況變化的情形來調整送出資料的順序，會造成無線頻寬使用的浪費，造成整體效能的下降。最糟的情形是都在某一鏈結(Link)處在”bad”的狀態下傳送封包至該使用者，將造成許多不必要的重傳，使得有效使用頻寬降低(從 IP 層觀點的有效頻寬)。對 TCP 使用者而言，連續的錯誤重傳延長了封包抵達使用者的時間，甚至封包遺失，使得 TCP 良流通率下降。對多媒體使用者而言，複雜多變的錯誤率變化使得在服務品質上也受到很大的影響。此外，由於複雜多變的錯誤率，在網際網路發展蓬勃的今天，對於保障服務品質將是一大挑戰。

## 1.4 章節架構與問題描述

提到網際網路，大家都想到 TCP/IP，在無線環境下，使用傳統有線環境的通訊協定如 TCP/IP，最大的問題莫過於無線通道的品質所造成的影響。錯誤率常是不斷變化的，這對網路效能會是很大的影響，如何讓使用者使用時不受通道品質變化所造成的影響，將會是需要考量的一大問題。

此外，綜觀目前網際網路的研究中，如何保障服務品質，一直是一個重要的研究論題，因此我們可以斷言，當無線和有線面臨整合之後，除了增加無線上傳輸的可靠性之外，下一個重要的論題便是如何在這樣




複雜的環境中去保障服務品質。在無線環境中，對服務品質最大的挑戰，是多變的錯誤率，錯誤率的變化讓服務品質變得難以保障。

針對以上所提及的幾個問題我們將研究分成兩個部份來進行，如何在無線區域網路中增加 TCP 良流通率，以及如何在無線區域網路中保障服務品質。

在第二章中我們探討了無線區域網路中保障 TCP 效能的排程方法，在 TCP 效能方面我們設計的目標是達到最大的網路整體 TCP 的良流通率，我們設計了幾種排程方法。在第二章最末我們討論了在設計的架構中的一些參數討論以及通道模型變化參數對我們設計架構的檢驗。

在第三章中我們探討了無線區域網路中保障服務品質的排程方法，我們將即時資料流列入研究。我們將無線網路中區分為 TCP 和即時資料流兩種類型的服務，在即時資料流方面我們採用了 VoIP，設計的目標是達成在使用者端播放時的較低封包遺失率以及達成最大的網路整體 TCP 良流通率，當設計目標有衝突時以達成較低的有效封包遺失率做為我們首要的目標。



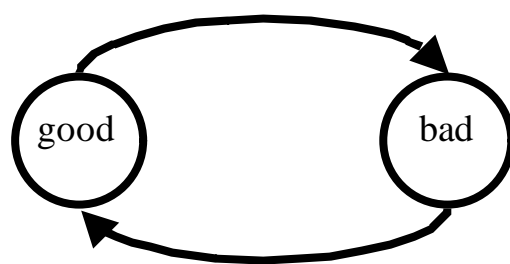
## 第二章

# 無線區域網路依通道特性之傳輸控制方法

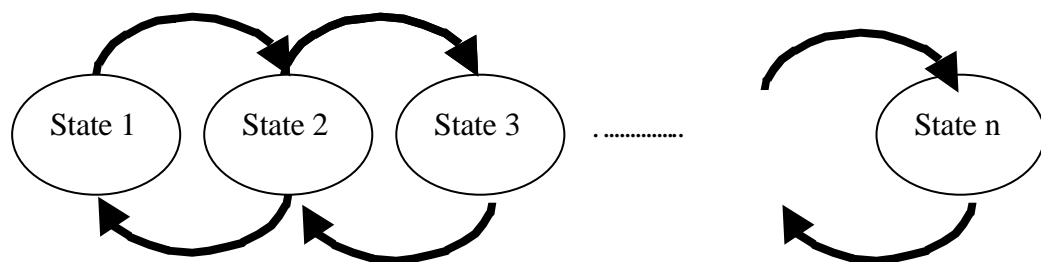
本章節主要探討無線通道的特性，以及如何在無線網路中增進網路整體 TCP 良流通率的方法，以達成最大的整體網路 TCP 良流通率為我們的設計目標。設計的部份主要是在資料鏈結層，我們提出了可適性資料鏈結層的架構，而在針對網路狀況變化的估計上我們提出了二種方法。

### 2.1 無線區域網路網路特性與對 TCP 效能的影響

在無線區域網路中，除了擁有一般無線通道的特性外，到達每個使用者鏈結上的通道狀態變化又各自獨立，在這種環境下若不依據網路狀態變化的情形來調整送出資料的順序，會造成無線頻寬使用的浪費，造成整體效能的下降，最糟的情形是都在每一鏈結(Link)處在”bad”的狀態下傳送封包，將造成許多不必要的重傳，使得有效頻寬降低(從 IP 層觀點的有效頻寬)，對 TCP 使用者而言，將延長封包送達的時間甚而封包遺失，使得 TCP 良流通率下降。在這種考量下，提出了可適性資料鏈結層的架構(Adaptive data Link Layer)，估測網路狀態變化的資訊，傳送時依此來調整傳送順序，以達到無線頻寬上較高的使用率。



二元狀態變化的通道模型



多元狀態通道模型

圖 2.1 通道模型

## 2.2 改善效能的方法

在此複雜多變的環境下，提供如同網際網路上的服務，品質容易受環境本身變化的影響，因此我們必須有所修正，將網路中位元錯誤率變化的衝擊減至最低。[10][11][12]

我們的想法是依據網路狀態變化的情形調整傳送封包，基於這個問

題我們區分為幾個設計重點：

- (1) 效能提升：無論是對 TCP、UDP、WWW、多媒體使用者而言，高的位元錯誤率使整體效能下降，我們首先必須提升效能。
- (2) 對網路網路狀態做良好的估測：由於在無線區域網路中複雜的錯誤率變化，使得封包在傳送時效能容易受到網路中錯誤率變化的影響。因此，我們在每個鏈結上的通道品質做一個適當的估計以做為資源分配時的依據。
- (3) 有效的分配網路資源：得知網路的狀態後，在資料傳送，對每個使用者做適當的頻寬資源分配。

由這些重點我們設計了可適性資料鏈結層架構，在考量大部份交通量由無線接取點至用戶端的下行方向，我們在無線接取點(AP, Access Point)做適當的傳輸控制，我們將控制的裝置設計在資料鏈結層(Data Link Layer)，主要是希望與現今網際網路做一個完整的結合，且在實作上有較大的可行性，這裝置區分為以下模組：

- (1) 遺失封包重傳模組：我們在資料鏈結層使用了重傳的機制，對於遺失在無線通道上的封包，我們用重傳的方式來提升效能，在第一章中我們曾經介紹過幾個增進效能的方法，最後我們決定使用這種方式來增加效能。
- (2) 通道狀態估測器：無線介質中錯誤率的變化將對我們傳送資料有相當大的影響，當通道中的錯誤率處於較高的狀態時，傳送資料常是無用的，除了造成頻寬和資源的浪費外，也會造成整體效能的下降，因此我們在無線接取點上增加了通道品質估測器，來預估通

道錯誤率變化的情形。

- (3) 排程器：排程器設計的目的是為了達到良好的資源分配，當到某一使用者的無線通道處於錯誤率較高的狀態時，傳送資料到這個使用者，將會犧牲相當多的無線頻寬。因此我們的想法是，經由網路品質的估測後我們選擇通道品質較好的優先傳送，藉由這個想法，設計了幾種排程器，在之後的章節一一描述。

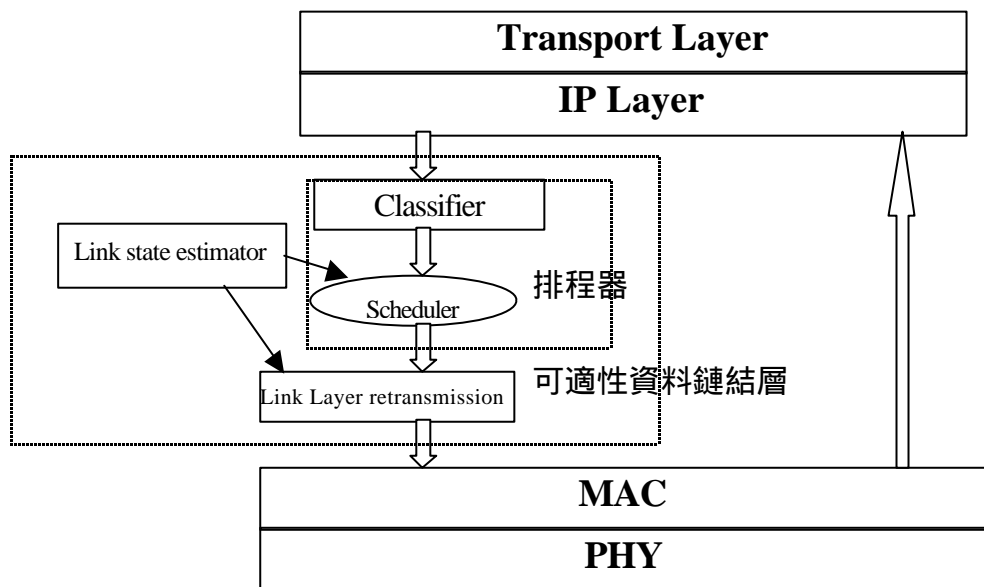


圖 2.2 可適性資料鏈結層架構圖

在實作上，將架構分成驅動程式(driver)和網路卡兩個部份，驅動程式除了我們一般驅動程式所定義的功能和界面外，我們設計的重點是在其中加入了排程器和估測器，而網路卡的部份負責傳送封包。另外一個我們說明的重點是網路卡和驅動程式的界面，驅動程式會給予網路卡

兩項資訊，欲傳送的封包和給予傳送此封包的最大傳送次數，而網路卡在工作完成後將給予驅動程式資訊，傳送是否成功和傳送該封包花費的傳送次數。

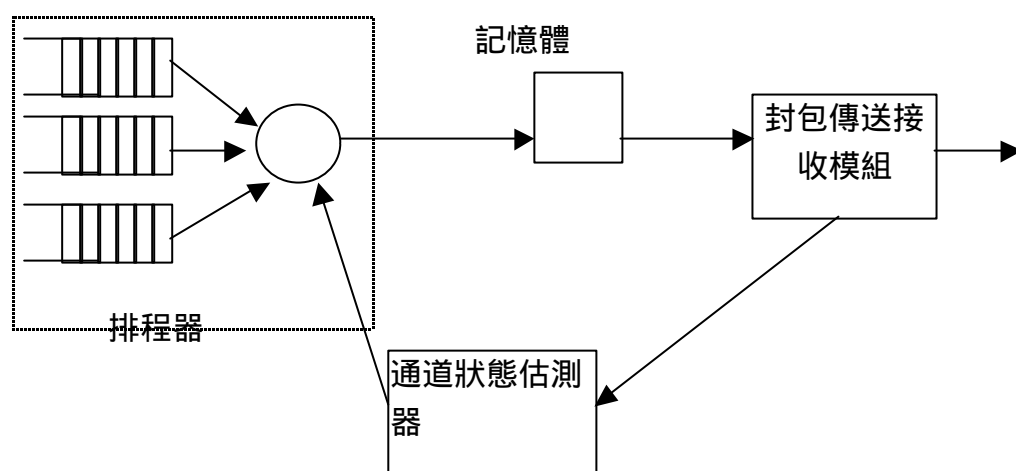


圖 2.3 可適性資料鏈結層實作架構圖

### 2.2.1 通道品質估測器設計

當網路鏈結(Link)上處於錯誤率較高的狀態時，送出資料是徒勞無功，我們的理念是判定網路鏈結狀態，來決定送出資料的先後順序，當狀況不好時將延後傳送或壓縮頻寬(減少傳送)，我們首先要知道的是網路狀況的資訊。基於這個想法，我們設計了如下二種通道狀態估計的方法，其中除了估測網路狀態外，並一方面調整所給予的傳送次數。

(1) 估測器一：線性通道狀態估測器

(2) 估測器二：二元通道狀態估測器

在設計通道狀態估測器的時後，我們將重傳次數視為無線上傳輸的

資源，無論是估測器一或估測器二，在通道狀態好時我們都給予較多的資源，在通道狀態不好時(傳送失敗)，壓縮給予的資源，兩者之間的不同僅是壓縮或放寬資源時的幅度和方式。反過來說，所分配的資源多者，代表了通道的品質較佳，分配資源較少者，通道品質較差，估測器所扮演的角色便是將這類資訊傳達給排程器，而排程器便將估測器所給予的資訊來做有效的資源分配。

假設通道狀態變化的速率並不大快，使得估測器能較完整的估測通道狀態的資訊，若通道變化過快時，估測器的效能較不佳，但預期這類狀況不常發生。

### 2.2.1.1 通道狀態估測方法一：線性通道狀態估測器

當通道的品質變得不好時，封包遺失率相對的增加，這時需要重傳遺失封包來提升效能。因此，在估測器中紀錄了每個鏈結上花費的重傳封包次數，來判定網路的狀態是否良好，假設最大重傳次數為  $R_{\max}$ ，針對每個鏈結紀錄了最大重傳次數為  $R$  和目前重傳次數為  $r$ ，當重傳次數至紀錄的最大值  $R$  仍未將封包成功傳出時，則判定網路狀態走向錯誤率高的狀況，這時我們設  $R=R-1$ ，反之我們設  $R=R+1$ ，我們的目的是當網路狀況不好時便限制其傳送次數來壓縮使用的頻寬，當狀態好時增加其重傳次數。

我們將此法做如下的說明：

#### (1) 變數定義：

<1>  $N$  :通道獨立變化的鏈結數，可能到某一群使用者鏈結上的通道狀態變化是相同的，在本文中假設到達每一個使用者鏈結上的通道狀態變化都是獨立的。

<2>  $R_{\max}$  :在網路中，假設最大重傳的次數為  $R_{\max}$ ，若封包重傳次數超過  $R_{\max}$ ，卻無法將封包成功送出，便停止傳送此封包。

<3>  $R_i$  :給予網路中第  $i$  鏈結的最大重傳次數。在此法中，我們經由網路狀態的估計，來回授調整  $R_i$  的大小。

<4>  $r_i$  :網路中第  $i$  鏈結上的上次重傳次數，每當有封包從第  $i$  鏈結送出時便紀錄  $r_i$ ，在此法中，我們以  $r_i$  的值來判定網路的狀態。

## (2) 估測器演算法描述：

假設目前有  $N$  個獨立變化的鏈結(或  $N$  個使用者，Link)，每個鏈結最大重傳次數  $R_i$ ，而網路的最大重傳次數  $R_{\max}$ ，在某一個傳送過程中，第  $i$  條鏈結上的重傳次數是  $r_i$ ，我們可以得到以下的關係：

$$1 \leq R_i \leq R_{\max} \quad i = 1, 2, 3, \dots, N$$

$$1 \leq r_i \leq R_i \quad i = 1, 2, 3, \dots, N$$

當在第  $i$  條鏈結上，封包傳送  $R_i$  次仍無法有效傳送，便判定此鏈結處於封包遺失率較高的狀態，流程圖如圖 2.4。當傳送成功，則增加  $R_i$



，設定  $R_i = R_i + 1$ ，但最大不能超過  $R_{\max}$ ；當傳送失敗，便減少  $R_i$ ，設定  $R_i = R_i - 1$ ，但最小不能低於 1。

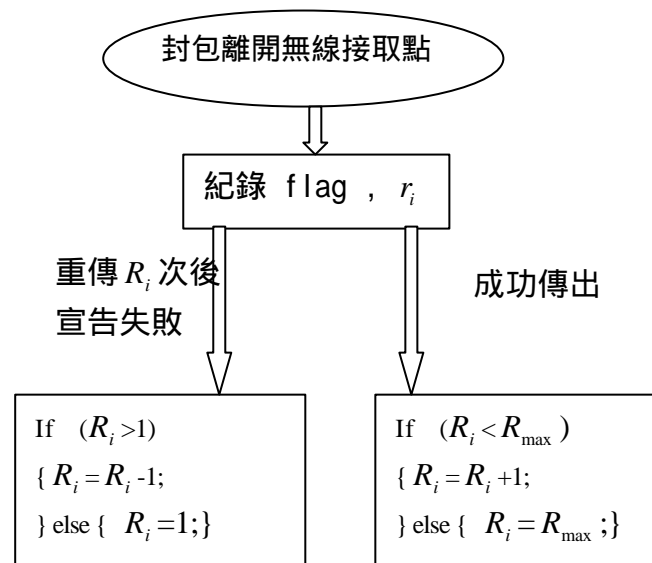


圖 2.4 線性通道狀態估測器

### 2.2.1.2 通道狀態估測方法二：二元通道狀態估測器

變數定義如 2.2.1.2，當封包重傳次數達  $R_{\max}$  仍未將封包成功傳出便判定網路鏈結處於較不好的狀態，此時便當此鏈結上的最大重傳次數設為一次，也就是僅允許此狀態處於不好的鏈結上最多只能重傳一次，當此鏈結上成功傳送封包，便判定此鏈結已經回到錯誤率較低的狀況，因此再將此鏈結上的最大重傳次數設為  $R_{\max}$ ，此法所使用網路狀況估計與前一種方式是相同的，所不同的是當得到網路狀態的估計後所做的調

整不同，在此法中，一旦以知網路狀況處於封包遺失率較高的狀態時，便將重傳次數減為一次，我們將此法做如下的說明：

(1) 變數定義：

<1>  $N$ ：通道獨立變化的鏈結數，可能到某一群使用者鏈結上的通道狀態變化是相同的，在本文中假設到達每一個使用者鏈結上的通道狀態變化都是獨立的。

<2>  $R_{\max}$ ：在網路中，假設最大重傳的次數為  $R_{\max}$ ，若封包重傳次數超過  $R_{\max}$ ，卻無法將封包成功送出，便停止傳送此封包。

<3>  $R_i$ ：對網路中第  $i$  鏈結所給予的最大重傳次數，在此法中，我們經由網路狀態的估計，來回授調整  $R_i$  的大小。

<4>  $r_i$ ：網路中第  $i$  鏈結的上次重傳次數，每當有封包從第  $i$  鏈結送出時便紀錄  $r_i$ ，在此法中，我們以  $r_i$  得值來判定網路的狀態。

(2) 演算法描述：

假設目前有  $N$  個獨立變化的鏈結(或個使用者，Link)，每個鏈結最大重傳次數  $R_i$ ，而網路的最大重傳次數  $R_{\max}$ ，在某一個傳送過程中，第

$i$  條鏈結上的重傳次數是  $r_i$ ，我們可以得到以下的關係：

$$1 \leq R_i \leq R_{\max} \quad i = 1, 2, 3, \dots, N$$

$$1 \leq r_i \leq R_i \quad i = 1, 2, 3, \dots, N$$

換言之，當在第  $i$  條鏈結上，封包傳送  $R_i$  次仍無法有效傳送，便判

定此鏈結處於封包遺失率較高的狀態，流程圖如 2.5。當傳送成功，則增加  $R_i$

，設定  $R_i = R_{\max}$ ；當傳送失敗，便減少  $R_i$ ，設定  $R_i = 1$ 。

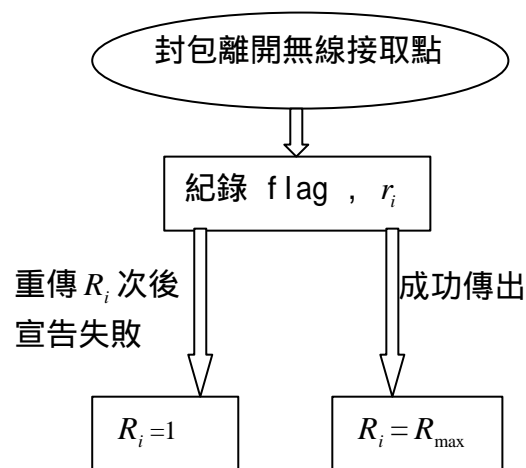


圖 2.5 二元通道狀態估測器

### 2.2.2 排程器設計

排程器的設計，是希望能將無線上的資源做有效的分配，除了將網路狀態估測做為回授控制的一環外，排程器的選擇也是一個重點，良好的排程器才能將無線上的資源做最恰當的分配，排程器我們設計的架構中，扮演了資源分配的角色。

在研究之初，將目前幾種常見的排程器來做比較，如循環法(Round-Robin)，權重式循環法(Weighted Round-robin)，並設法與我們設計

的估測器搭配來完成我們的設計。此外，在基於整體設計的考量下，我們更從中做了修正，設計了兩個排程器：依通道狀態傳輸循環法(Channel-Controlled Round Robin, CCRR)，依通道狀態傳輸權重式循環法(Channel-Controlled Weighted Round Robin, CCWRR)。

在設計排程器之前，我們先討論通道狀態的估測方面，在設計通道狀態估測器的時後，我們將重傳次數視為無線上傳輸的資源，無論是估測器一或估測器二，在通道狀態好時我們都給予較多的資源，在通道狀態不好時(傳送失敗)，我們就減少給予的資源，兩者之間的不同僅是壓縮或放寬資源時的幅度，理念上是相同的。因此，反過來說，所分配的資源多者，代表了通道的品質較佳，分配資源較少者，通道品質較差，估測器所扮演的角色便是將這類資訊傳達給排程器，而排程器便將估測器所給予的資訊來做有效的資源分配。

#### 2.2.2.1 通道控制排程設計一：依通道狀態傳輸循環法

我們所設計的第一種排程器是從循環法中改良過來的，循環法所使用的原理很簡單，在很多交換機或路由器都有類似的功能，它的基本觀念就是輪流，但我們設計的目標是良好的資源分配，所以純粹的循環法並不能達到我們的目標，我們將循環法做了一些修正。

- (1) 由估測器所傳來的資訊來判斷每個鏈結上的通道狀態。
- (2) 當鏈結上的通道品質屬於較佳者優先傳送。
- (3) 我們定義一個循環(Round)為所有使用者輪流一次。

(4) 已傳送過者在這一輪不能再被傳送。

依通道狀態傳輸循環法與循環法都是輪流的概念，所不同的僅是輪流先後順序的問題，CCRR 除了輪流外，將通道狀態做了考量而排出了另一個順序。我們經由以上這四點想法來做修正，其流程圖 2.7。

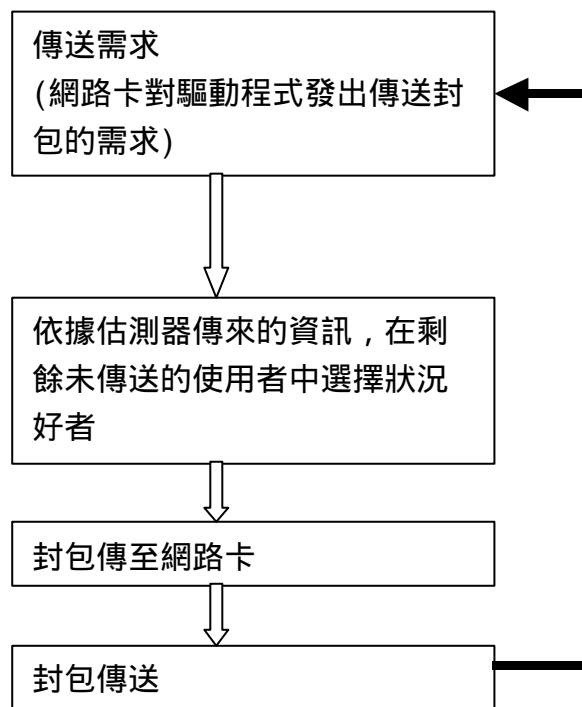


圖 2.7 依通道狀態傳輸循環法

## 2.2.2.2 通道控制排程設計二：依通道狀態傳輸權重式

### 循環法

我們設計的第二種排程器是依通道狀態傳輸權重式循環法(Channel Controlled Weighted Round Robin, CCWRR)，權重式循環法一個常見的排程器，在交換機、路由器中都有所使用，它最大功能是用來做頻寬分配，我們採用權重式循環法加上估測器的資訊來完成我們的設計，設計的重點與 CCRR 相似，最大的關鍵是權數的設定，權數的大小將決定我們對通道品質好壞的不同待遇，有關權數參數的探討我們將在本章最後做探討，設計的重點如下：

- (1) 將估測器的資訊傳送給排程器作為排程的依據。
- (2) 狀況較佳的通道我們給予一個權數，狀況較差的通道給予另一個權數。
- (3) 權數在每傳送一個封包出去(由網路卡傳出)時更新一次。

由上面三點我們設計了 CCWRR，其流程圖如 2.8 所示。此外，排程器資源分配演算法如下：

#### (1) 變數定義

$f_g$  (good 狀態權數)：排程器中給予通道狀態較佳者的權數。

$f_b$  (bad 狀態權數)：排程器中給予通道狀態較差者的權數。

$T_h$ ：由估測器判定通道狀態的權數臨界值。

$R_{\max}$ ：網路中的最大重傳次數。

$R_i$  : link  $i$  的最大重傳次數。

## (2) 排程演算法

經由  $R_i$  和  $T_h$  的比較來動態調整不同的權數”Weight”。

*if*  $R_i \geq T_h$ , *set*  $weight = f_g$

*if*  $R_i < T_h$ , *set*  $weight = f_b$

*set*  $T_h = R_{\max} / 2$

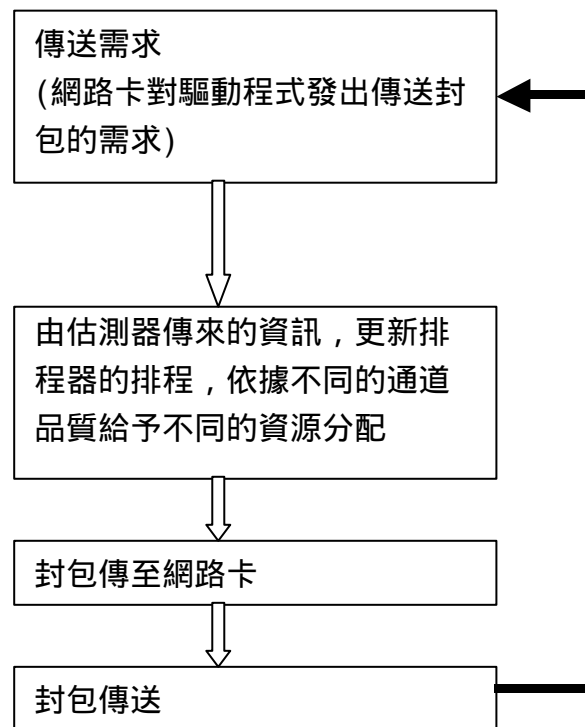


圖 2.8 依通道狀態傳輸控制權重式循環法

## 2.3 模擬環境

我們模擬環境的建立是用 NS(Network Simulator)[18]來建構，我們更改了其中 link 的模組來建立我們需要的模擬。建立環境可分成四個部份：無線區域網路、無線通道模型、排程器、估測器。在網路環境方面，我們假設使用者位於無線區域網路中，經由無線界面和網際網路下載網際網路上的資訊，我們將探討使用者所獲得的頻寬，和網路整體的使用情形。

### 2.3.1 模擬環境描述

模擬環境是 IEEE 802.11 的網路，考慮由網際網路的伺服器傳送資料至無線環境的使用者端，在 IEEE 802.11 的無線網路中，在無線通道中至每一個使用者間的錯誤率是各自獨立且會隨時間變化，我們將比較以上提出幾種設計組合和 FIFO 做比較，傳送資料的型態為 TCP。

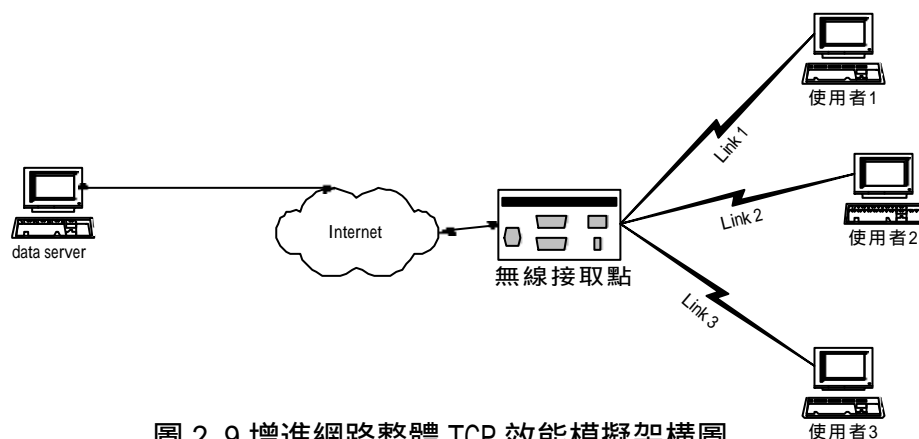


圖 2.9 增進網路整體 TCP 效能模擬架構圖



網路參數設定如下所示：

### (1) 通道模型

通道變化的參數如上所示，分為 good 和 bad 兩個狀態，各對應封包遺失率 0% 和 100%，兩個狀態間的變換是採指數分配，停留在 good 和 bad 狀態的平均時間為 1.0sec 和 0.1sec。

通道狀態	參數	值
good	狀態停留平均時間(sec)	1.00
	封包遺失率	0%
bad	狀態停留平均時間(sec)	0.10
	封包遺失率	100%

表 2.1 通道模型參數

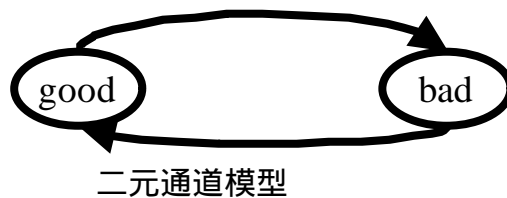


圖 2.10 通道模型

### (2) 網路參數

在無線區網路中給予  $R_{\max} = 8$ ，上下行無線頻寬為 750kbps。

### (3) CCWRR 的參數

在 CCWRR 參數的設計上，主要是通道狀態權數  $f_g$ 、 $f_b$  的設計。這兩個參數決定了我們對於通過狀態的好或壞的不同頻寬分配，而其中  $f_g$  的設計中我們在本章最後做一個完整討論。在這個階段

的模擬中，我們取了兩個值，3 和 8。我們在經過模擬證實 good 狀態參數為 2、3、4 在效能上差距不大，而 good 狀態參數為 8~12 的效能差距亦不大，因此我們取這兩個值做為代表。

參數	數值
$f_g$ (good 狀態權數)	2~12
$f_b$ (bad 狀態權數)	1

表 2.1 依通道狀態傳輸權重式循環法參數設計

### 2.3.2 模擬結果

模擬時間為 20 秒，結果如表 2.2、圖 2.11、圖 2.12。

排程器	估測器	使用者1 TCP 良流通 率	使用者 2 TCP 良流通 率	使用者 3 TCP 良好流 通率	網路整體 TCP 良流通 率
FIFO		154.090	148.101	174.492	476.683
RR	估測器(一)	164.744	145.910	156.603	467.256
	估測器(二)	170.894	149.769	175.377	496.040
CCRR	估測器(一)	165.166	201.628	208.543	575.337
	估測器(二)	198.513	220.945	142.854	562.312
CCWRR(3)	估測器(一)	180.080	175.558	184.462	613.246
	估測器(二)	203.437	208.020	197.930	609.387
CCWRR(8)	估測器(一)	217.608	210.492	201.166	629.266
	估測器(二)	227.819	202.111	226.653	656.583

表 2.2 TCP 良流通率：單位 kbps

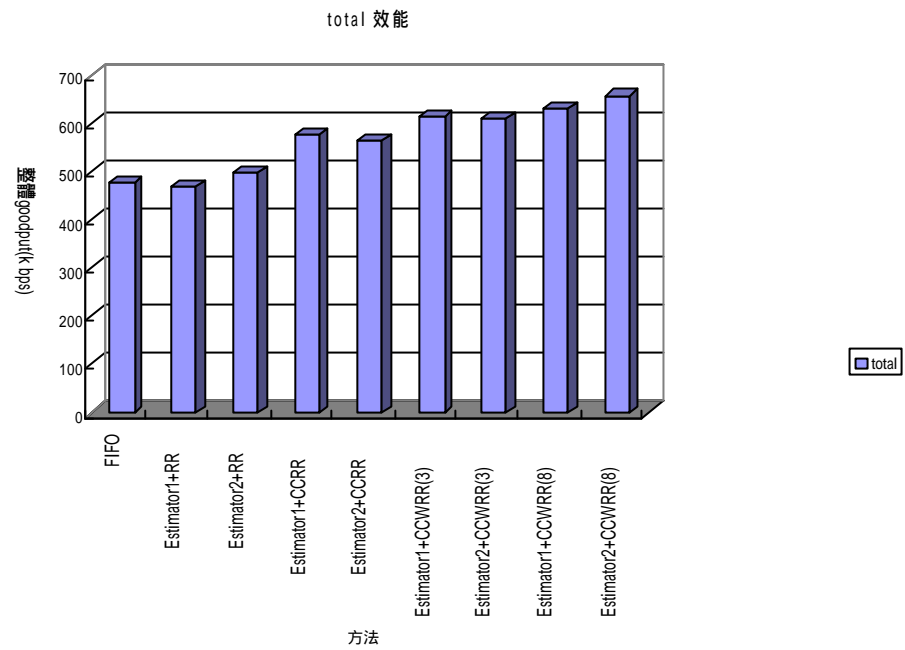


圖 2.11 提升 TCP 效能在各方法下之模擬結果：縱軸為網路整體 TCP 良流通率，橫軸表示所使用的方法。

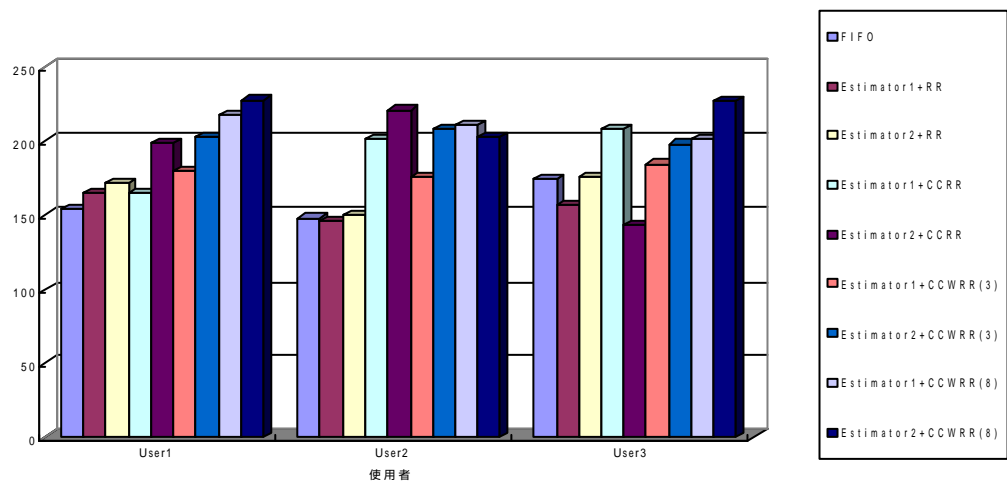


圖 2.12 模擬結果  
在各種組合下，各用戶之良流通率(kbps)

由模擬結果中，我們可以知道，使用循環法搭配兩種估測器的網路整體 TCP 良流通率都比 FIFO 來的更低，這顯示了一件事，在我們估測器的設計上有依網路狀態調整傳送次數，很顯然的此法並不能增加效能。使用 CCWRR 無論是和線性通道估測器、二元通道估測器搭配都可以達到最高的效能，其中以搭配二元通道估測器最佳。在參數的設定上，我們可以看到  $f_g=3, 8$  之間的比較，8 時顯然較好。在公平性的考量上，由於我們在設計這個架構時並未有加入，因此未加以討論，這將可以是下一階段的研究方向。

## 2.4 參數設計的討論

在以上的討論中我們證實使用我們所設計架構的確可以有效的增加使用效能，然而有幾個參數的設定也是設計的重點。

### (1) $R_{\max}$ ，網路最大重傳次數

我們設定這個參數的目的，是給予每次傳送封包時的最大重傳上限，是為了避免 H0L(Head Of Line)的效應，當網路狀態處於較差的狀態時過多的重傳是徒勞無功的，我們在以上的實驗中都將其值設為 8，但在事後我們都對此產生質疑。

### (2) $f_g$ ，good 狀態權數

此參數設計的目的，是在給予狀態好的通道使用者較多的頻寬資源，在模擬時我們發現使用  $f_g=8$  可以達到最大的效能。

### 2.4.1 模擬環境

我們模擬的環境如表 2.2 節所設定，用 NS 架構。我們將改變網路最大重傳次數  $R_{\max}$  和 good 狀態權數  $f_g$ ，來觀察對網路整體 TCP 良流通率的影響。

### 2.4.2 模擬結果

結果如上所示， $R_{\max}=4, 8, 12$  對應  $f_g=2, 4, 6, 8, 10\ldots, 16$ ，所顯示的結果為整體 TCP 良流通率，可以觀察到幾個現象：

#### (1) good 狀態權數的設定

我們比較權數大於 6 和小於 6，無論網路最大重傳次數為何，以權數大於 6 者效能為佳。而在權數設為 6 以上時，網路整體 TCP 良流通率都在 650kbps 附近。因此 good 狀態權數設計的最佳值是大於 6。

#### (2) 網路最大重傳次數 $R_{\max}$

由模擬結果知，最大重傳次數設為 4 和設為 8 的效能頗接近，但設為 8 的較好一些。最大重傳次數設為 8 和 12 相比較，明顯以設為 8 的比較佳。因此可以證實在我們給予的環境下用  $R_{\max}=8$  是最佳選擇。

$f_g$	Rmax (網路最大重傳次數)		
	4	8	12
2	521.1	540.5	552.3
4	621.1	609.4	594.5
6	647.4	666.7	648.9
8	653.8	656.6	631.1
10	650.6	660.4	666.1
12	668.4	657.7	635.5

表 2.3 模擬結果：在不同網路最大重傳次數  $R_{\max}$ 、good 狀

態權數  $f_g$  下的整體 TCP 良流通率(kbps) 圖

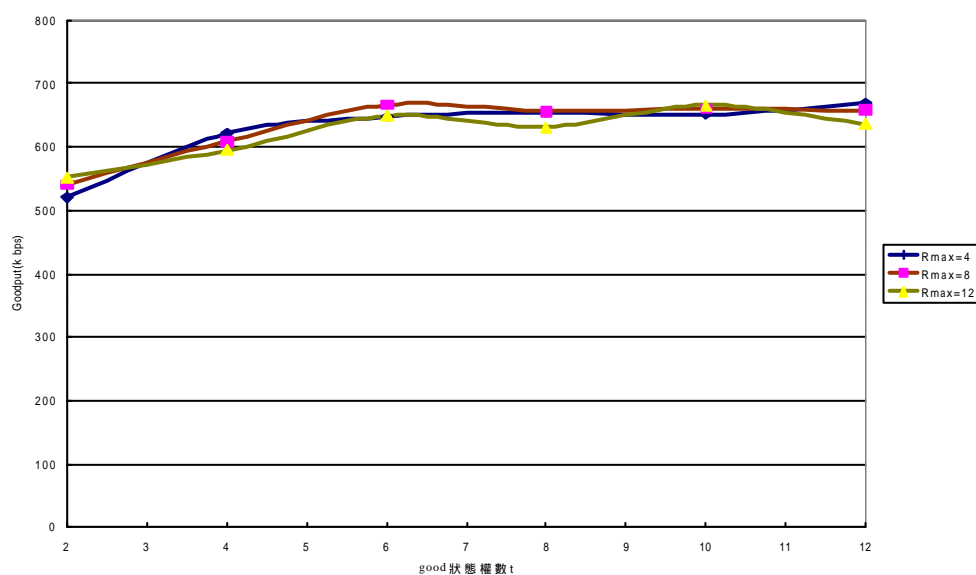


圖 2.13 模擬結果：在不同網路最大重傳次數  $R_{\max}$ 、good 狀態參數  $f_g$  下的網路整體 TCP 良流通率(kbps) 圖

## 2.5 通道模型參數之討論

### 2.5.1 通道狀況參數之檢驗

在之前的設計模擬中，我們在通道狀態的參數上給予某些特定的數值，我們在這節中將討論當通道停留在每個狀態的參數在不同的數值下，我們的設計是否依然能增加效能，網路架構的建立如前所述。

在驗證的過程中我們改變了通道模型參數中的 good 狀態平均停留時間和 bad 狀態停留時間，分兩方面進行：固定 good 狀態平均停留時間改變 bad 狀態停留時間以及固定 bad 狀態平均停留時間改變 good 狀態停留時間參數，如表 2.4 和 2.5。

經由模擬結果表 2.6~2.10 發現，無論通道狀態停留時間如何變化，相對於 FIFO，依通道狀態傳輸控制權重式循環法都能有效的增進效能，達到比較佳的網路整體 TCP 良流通率。

通道狀態	通道狀態參數	數值
good	狀態停留平均時間(sec)	1.0
	封包遺失率	0%
bad	狀態停留平均時間(sec)	0.01, 0.1, 1.0
	封包遺失率	100%

表 2.4 通道狀態參數表

通道狀態	通道狀態參數	數值
good	狀態停留平均時間(sec)	0.01, 2.0
	封包遺失率	0%
bad	狀態停留平均時間(sec)	0.1
	封包遺失率	100%

表 2.5 通道狀態參數表

mean( bad)=0.01					
		使用者1	使用者2	使用者3	整體網路
FIFO		186.80	190.00	199.36	576.16
CCWRR weight=3	估測器一	227.06	225.82	228.54	681.42
	估測器二	237.04	237.32	236.58	710.94
CCWRR weight=8	估測器一	230.56	230.32	229.82	690.70
	估測器二	239.96	237.68	240.02	717.66

表 2.6 Good,bad 狀態平均停留時間為 1.0sec, 0.01sec 時的 TCP 良流通率(kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

mean( bad)=0.1					
		使用者1	使用者2	使用者3	整體網路
FIFO		153.12	146.74	172.30	472.16
CCWRR weight=3	估測器一	177.86	172.66	182.96	533.48
	估測器二	202.26	206.42	196.02	604.70
CCWRR weight=8	估測器一	215.96	208.54	198.50	623.00
	估測器二	226.52	200.32	225.36	652.20

表 2.7 Good,bad 狀態平均停留時間為 1.0sec, 0.1sec 時的 TCP 良流通率(kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

mean( bad)=1.0					
		使用者1	使用者2	使用者3	整體網路
FIFO		153.12	146.74	172.30	472.16
CCWRR weight=3	估測器一	177.86	172.66	182.96	533.48
	估測器二	202.26	206.42	196.02	604.70
CCWRR weight=8	估測器一	215.96	208.54	198.50	623.00
	估測器二	226.52	200.32	225.36	652.20

表 2.8 Good,bad 狀態平均停留時間為 1.0sec, 1.0sec 時的 TCP 良流通率(kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器



mean( good)=0.1					
		使用者1	使用者2	使用者3	整體網路
FIFO		39.62	11.86	24.86	76.34
CCWRR weight=3	估測器一	72.60	62.24	63.56	198.40
	估測器二	80.34	74.62	68.26	223.22
CCWRR weight=8	估測器一	84.10	74.74	74.16	233.00
	估測器二	75.02	81.94	82.94	239.90

表 2.9 Good,bad 狀態平均停留時間為 0.1sec, 0.1sec 時的 TCP 良流通率(kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

mean( good)=2.0					
		使用者1	使用者2	使用者3	整體網路
FIFO		194.20	211.44	159.70	565.34
CCWRR weight=3	估測器一	213.90	187.50	213.34	614.74
	估測器二	205.82	215.04	216.30	637.16
CCWRR weight=8	估測器一	228.90	223.12	211.70	663.72
	估測器二	235.04	221.84	243.66	700.54

表 2.10 Good,bad 狀態平均停留時間為 2.0sec, 0.1sec 時的 TCP 良流通率(kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

## 2.5.2 通道模型變化速率之研究

在本節中我們將討論當網路狀態變化的頻率不同時對我們設計的影響。我們設定通道的平均封包遺失率相同，更改通道狀態轉換的速率，觀察在設計架構下的網路整體 TCP 良流通率。我們更改了通道狀態轉換的頻率，且變換的時間並不會造成 TCP timeout 的現象。

觀察模擬結果表 2.12 、 2.13 ，我們發現通道狀態變化頻率對 FIFO 並沒有影響，網路整體 TCP 良流通率都維持在 70~80 kbps，而使用 CCWRR 仍可達到比較高的網路整體 TCP 良流通率，值得注意的是，當變化速率較高時，能增加的效能幅度就比較有限，這是因為當通道狀態變化過快時，估測器就比較不能抓到網路狀態的資訊。

通道狀態	通道狀態參數	數值
good	狀態停留平均時間(sec)	0.1 及 0.001
	封包遺失率	0%
bad	狀態停留平均時間(sec)	0.1 及 0.001
	封包遺失率	100%

表 2.11 通道狀態參數表，在本節的實驗我們將改變狀況轉換時間，分別為 0.1 和 0.001

mean( good)=mean(bad)=0.001					
		使用者1	使用者2	使用者3	整體網路
FIFO		40.22	10.66	24.88	75.76
CCWRR weight=3	估測器一	47.2	28.16	36.14	111.5
	估測器二	42.5	30.6	38.1	111.2
CCWRR weight=8	估測器一	49.46	27.56	39.68	116.7
	估測器二	51	28.08	40.28	119.36

表 2.12 Good,bad 狀態平均停留時間皆為 0.001sec 時的 TCP 良流通率(kbps)，估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

mean( good)=mean(bad)=0.1					
		使用者1	使用者2	使用者3	整體網路
FIFO		39.62	11.86	24.86	76.34
CCWRR weight=3	估測器一	72.60	62.24	63.56	198.40
	估測器二	80.34	74.62	68.26	223.22
CCWRR weight=8	估測器一	84.10	74.74	74.16	233.00
	估測器二	75.02	81.94	82.94	239.90

表 2.13 Good,bad 狀態平均停留時間皆為 0.1sec 時的 TCP 良流通率 (kbps)，估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

### 2.5.3 估測器檢驗

在之前的設計中我們將通道狀態封包遺失率設為 100%和 0%，我們試著改變封包遺失率為 99%和 1%，來檢驗我們估測器是否能良好的運作。觀察模擬結果 2.15、2.16，當改變通道狀態的封包遺失率之後，線性通道狀態估測器能得到比較佳的效能，這與之前的模擬有所不同，這是由於當通道狀態的封包遺失率並非絕對的 100%和 0%時，二元通道狀態估測器對通道狀態會有所誤判而降低了效能，而相對的線性通道狀態估測器反而能適應這個變化。

通道狀態	通道狀態參數	數值
good	狀態停留平均時間(sec)	1.0
	封包遺失率	99%
bad	狀態停留平均時間(sec)	0.5, 1.0
	封包遺失率	1%

表 2.14 通道狀態參數表

mean( good)=1.0,mean(bad)=0.5					
		使用者1	使用者2	使用者3	整體網路
FIFO		104.66	64.18	156.38	325.22
CCWRR weight=3	估測器一	149.46	123.80	122.04	395.30
	估測器二	127.32	127.62	134.60	389.54
CCWRR weight=8	估測器一	150.84	149.94	187.46	488.24
	估測器二	110.24	153.76	116.52	380.52

表 2.15 Good,bad 狀態封包遺失率為 99%和 1%時的 TCP 良流通率 (kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

mean( good)=1.0,mean(bad)=1.0					
		使用者1	使用者2	使用者3	整體網路
FIFO		58.72	20.30	55.08	134.10
CCWRR weight=3	估測器一	95.74	68.48	82.04	246.26
	估測器二	70.60	47.40	74.54	192.54
CCWRR weight=8	估測器一	168.00	60.62	103.14	331.76
	估測器二	115.54	95.50	94.64	305.68

表 2.16 Good,bad 狀態封包遺失率為 99%和 1%時的 TCP 良流通率 (kbps) , 估測器一和估測器二分別為線性通道狀態估測器和二元通道狀態估測器

## 2.6 觀察總結

在這一章中我們探討了如何在無線區域網路中增加 TCP 效能的方法，我們設計了幾種架構，以達到最大的網路整體 TCP 良流通率為我們的目標，設計主要在資料鏈結層，包含排程器、網路狀態估測器、遺失

封包重傳模組。

首先在排程器設計方面，不論通道模型是否完美，CCWRR"依通道狀態傳輸權重式循環法都可有效增加網路整體 TCP 良流通率。在完美通道模型下，對於 CCWRR 的參數我們做了檢驗，參數主要是 good 狀態權數  $f_g$  和網路最大重傳次數  $R_{\max}$ ，發現  $R_{\max}=8$  和  $f_g \geq 6$  會有較佳的表現。

在估測器設計上，在完美通道模型下，以二元通道狀態估測器表現為佳，但在不完美通道模型下，以線性通道狀態估測器表現較為強韌。

通道狀態變換頻率若有不同時，使用一般 FIFO 並不影響 TCP 良流通率；使用 CCWRR，相較於 FIFO 都能增加效能，但在通道狀態變換較緩和的情形下，我們的設計可較有效提升效能。



## 第三章

# 無線區域網路保障多重服務品質之排程方法

在無線環境中，對服務品質最大的挑戰，是多變的錯誤率，錯誤率的變化讓服務品質變得難以保障，我們在這章討論了如何在這樣環境中保障服務品質。[20][21]

在第二章的討論中主要針對的是保障無線上頻寬的使用率，在服務的類型方面主要針對一般的檔案資料傳輸的資料流。面臨目前多樣化的服務品質需求下，不同的服務類型所須保障的服務品質也有所不同，亦要使用不同的方式來保障。在這章中主要討論是如何針對不同的服務類型保障服務品質的方法。

我們在這章中，提出了多重服務可適性資料鏈結層的架構，在無線環境中來保障服務品質，在本章中的討論只是一個概念性的考量，在面臨往後的多樣化的網路服務，仍有許多修正的空間。在本章的設計中，設計的目標以達到最大的整體網路 TCP 良流通率和即時資料流的有效封包遺失率為目標，有效封包遺失率是將延遲變異的影響考慮在內。

### 3.1 網際網路服務品質需求

在現今網際網路中，服務品質的研究一直是一個重要的論題，無論在使用者端、路由器..，都加入一些保障服務品質的方法。不同的服務類型需要不同的服務品質要求，亦需要不同的方法來保障。

我們將目前網際網路的服務區分成兩大類，一般資料(data)服務和即時(real-time stream)資料服務，就資料服務而言，所須保障的是傳輸時的可靠性，對於傳輸過程中花費多少時間並不是重要的要素，因此在設計的理念上，我們沿用第二章的設計來看這類的服務，在第二章的討論中我們也證明了當使用我們所設計的架構的確能保障這類的服務(從 TCP 良流通率的服務品質要求角度來看)。針對即時的服務，所考量的是延遲時間(delay time)、延遲變異(delay Jitter)、封包遺失率等，服務品質需求的是在封包遺失率可容量的範圍內，達到延遲時間和延遲變異所要求的上限之內。

#### 3.1.1 語音資料的規格

表 3.1 中列出目前 ITU 所制定的一些語音編碼(Voice Coder and decoder ,VoCodec)標準所使用之技術的位元速率[1]。我們將採用 G.729 CS-ACELP 語音編碼解碼技術。當語音資料封包經過層層標頭封裝成 IP 封包，再經過切割和加入無線標頭，便送入無線通道中。假設我們 voice 資料流的 IP 封包大小為 50 bytes，扣去 UDP 和 RTP 的 header 後剩下 10 bytes 的語音資料，欲達到 8 Kbps 的位元速率，我

們必須每 10ms 傳送一個 50bytes 的語音 IP 封包，可以將這個資料流視為 40kbps 的固定速率資料流，我們在模擬時忽略了無線封包的標頭和對於 IP 封包的切割。

Codec		位元速率(kbps)
G.711	PCM	64
G.726	ADPCM	32
G.728	LD-CELP	16
G.729	CS-ACELP	8
G.729a	CS-ACELP	8
G.723.1	MPMLQ	6.3
G.723.1	ACELP	5.3

表. 3.1 : 語音編碼解碼

Version	Length	TOS	Total length	
Authentication code			Flag	Segment offset
Time-to-live		Protocol	Header checksum	
Source address				
Destination address				

IP header  
12bytes

Source port	Destination Port
Total length	Header checksum

UDP header  
8 bytes

V	P	X	CC	M	PT	Sequence number
Timestamp						
Synchronous source identifier						
Contributing source identifier						

RTP header  
20 bytes

Voice encoding data
---------------------

圖 3.1 語音資料 IP 封包



### 3.2 多重服務可適性資料鏈結層架構

在第二章中探討了如何增加無線頻寬的使用率的方法，由模擬驗證，此設計有效的提升了 TCP 良流通率，保障了 TCP 的服務品質。設計架構包含了重傳遺失封包模組、估測器和排程器。由這設計出發，探討是否需要修正以及如何修正。

我們首先將目前的網際網路服務類型分為資料和即時兩類。針對傳輸資料的服務，所需保障的是資料的完整性，即時的服務，重視的是延遲、延遲變異和封包遺失率。由這需求下開始考量設計。在估測器方面，由於對所提供的服務而言，皆需要通道狀態變化的資訊，因此並沒有改變的必要，我們仍沿用第二章設計的估測器，使用二元通道狀態估測器。將設計重點放在遺失封包重傳模組和排程器方面：

#### (1)重傳遺失封包模組

在無線上，我們將無線的重傳視為一種資源，針對不同的服務是否需要不同的傳送次數限制，對於傳送即時資料流而言，重傳次數上限越大，對即時資料流的延遲變異產生的變異越大。在設計時，這也是一個重要的問題。

#### (2)排程器

在第二章中我們證明了我們設計的排程方法 CCRR"依通道狀態傳輸循環法"和 CCWRR"依通道狀態傳輸書輪權重式循環法"可以增加 TCP 的良流通率，有效保障資料傳輸服務的服務品質。針對要求不同服務品質需求的即時資料流而言，原先的設計是否會有不良影響以及如何去修正排程器的設計重點。

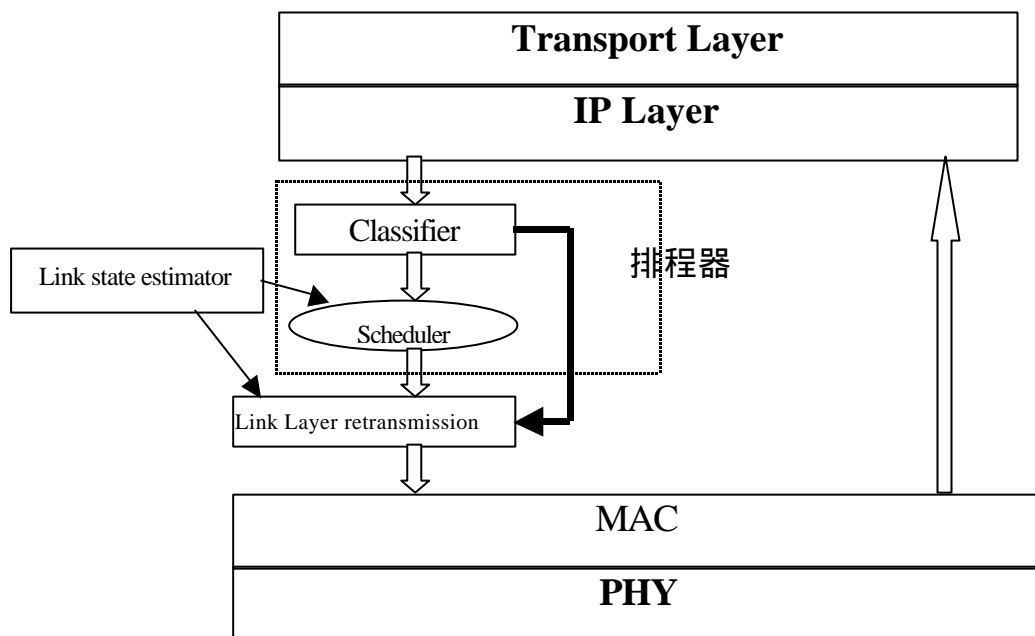


圖 3.2 多重服務資料鏈結層架構圖

### 3.2.1 重傳次數的差別待遇

我們在第二章中，使用了資料鏈結層的錯誤重傳來保障資料傳輸在無線通道上的可靠性，在資料型態為 TCP 時，可以將資料較可靠的傳送，付出的代價是傳送時的花費的延遲，這是由於每發生一次封包遺失，便需要重傳一次，相對的，成功傳送一個封包的時間也就加大，這對於即時的服務而言會是一個很大的問題。因此我們考慮對遺失封包重傳模組做出修正。對於資料的服務方面，我們不限制重傳次數，對於即時的服務，給予較少的重傳次數。我們對遺失封包重傳模組研究的問題

是 1. 是否該對重傳次數做限制 2. 重傳次數的上限的設定。在研究中，我們將即時資料流的上限設為 1，這是我們經過多次模擬驗證的結果。

### 3.2.2 排程器設計

我們在第二章中設計了二種排程器：CCRR"依通道狀態傳輸循環法"和 CCWRR"依通道狀態傳輸權重式循環法"，主要針對的服務型態是以 TCP 為主。我們這個架構出發來設計我們的排程器。在通道狀態估器方面使用了第二章設計的二元通道估測器。

#### 3.2.2.1 排程器架構一：依通道狀態傳輸共用佇列排程器

這個排程器沿用第二章所設計的架構。在佇列的分類上，主要依使用者的不同分類，將歸屬於同一使用者的即時資料流和 TCP 放置在同一佇列中。在排程方法上，使用了第二章設計的 CCWRR"依通道狀態傳輸權重式循環法"來調整封包傳送的先後順序。

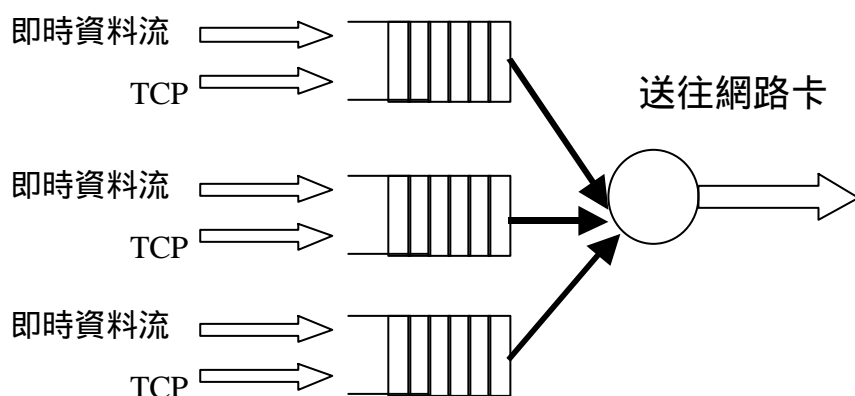


圖 3.3 依通道狀態傳輸之共用佇列排程器架構

### 3.2.2.2 排程器二：依通道狀態傳輸並指定即時資料流

#### 為高優先權之共用佇列排程器

在這個排程器的設計上，如圖 3.4、3.5，主要分成兩級。在佇列的分類上，分成兩級，第一級是依使用者分類，同一使用者下再依服務類型不同分類，即時資料流和 TCP 各放置在不同的佇列中。在排程方法上，我們分成兩級，針對同屬同一使用者的不同服務的佇列，給予即時資料流較高優先權。在不同使用者的佇列，用 CCWRR 來調整傳送順序。

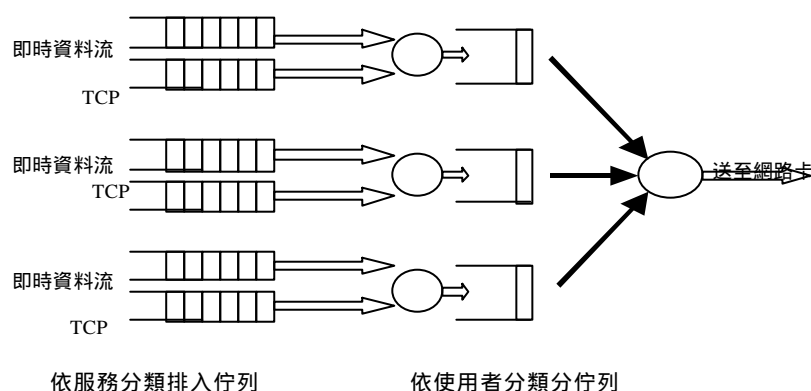


圖 3.4 依通道狀態傳輸並指定即時資料流為高優先權之共用佇列排程器架構圖

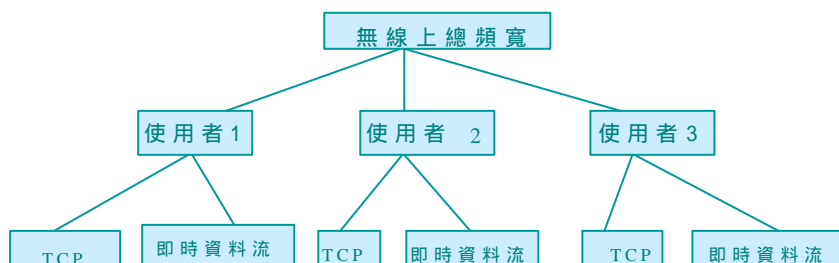


圖 3.5 排程器二頻寬分配階層圖

### 3.2.2.3 排程器三：依服務類型分流並指定即時資料流為高優先權之分級排程器

本排程器在設計上亦分為兩級，如圖 3.6、3.7。在佇列設計上分為兩級，第一級依服務類型分類，分為 TCP 和即時資料流，第二級方面，在 TCP 的佇列中，依使用者不同分佇列而即時資料流則同屬同一佇列。在排程方法上，針對 TCP 不同使用者間的佇列採用之前設計的 CCWRR，在 TCP 和即時資料流的佇列中，使用給予即時資料流較高優先權的方式。

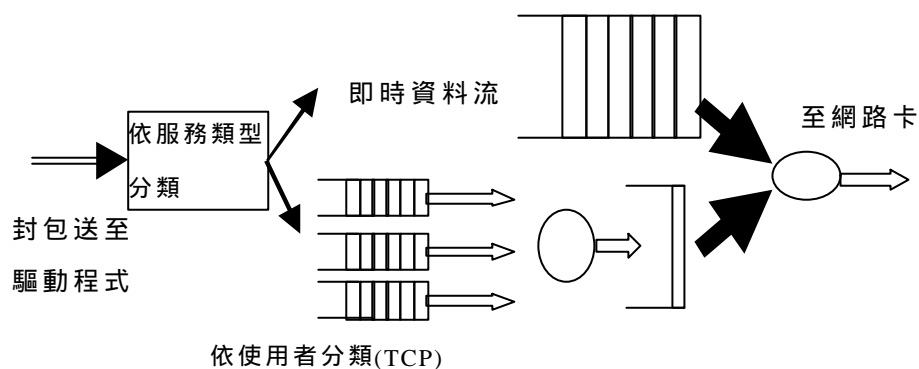


圖 3.6 依服務類型分流並指定即時資料流為高優先權之分級排程器架構圖

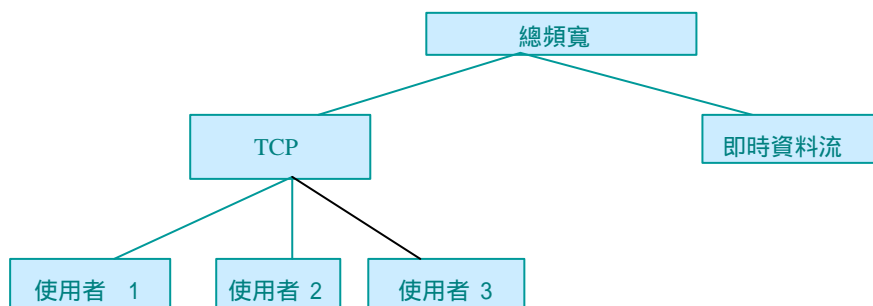


圖 3.7 排程器三頻寬分配階層圖

#### 3.2.2.4 排程器四：依服務類型分流並指定即時資料

##### 流為高優先權且依通道狀態傳輸之排程器架構圖

本排程器的設計從排程器三修正而來，亦區分為兩級。在佇列的設計上，第一級中，依服務類型分類，分成 TCP 和即時資料流兩部份，第二級中 TCP 和即時資料流都各依使用者的不同再分佇列。在排程方法的設計上，針對 TCP 和即時資料流兩佇列間給予即時資料流較高的優先權，在同屬 TCP 服務各使用者間的佇列，則使用 CCWRR"依通道狀態傳輸權重式循環法的排程方法，在同屬即時資料流的各使用者間，使用了 CCRR"依通道狀態傳輸循環法"的排程方法。

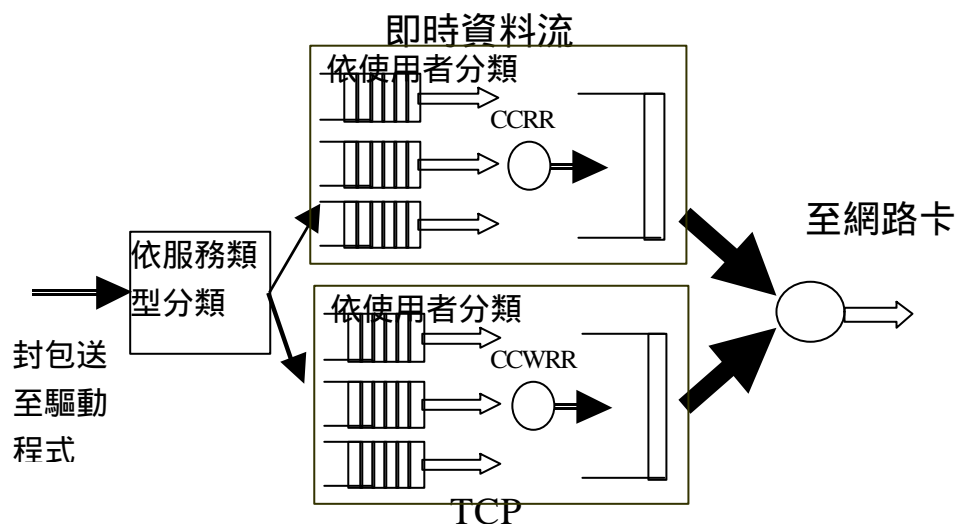


圖 3.8 依服務類型分流並指定即時資料流為高優先權且依通道狀態傳輸之排程器

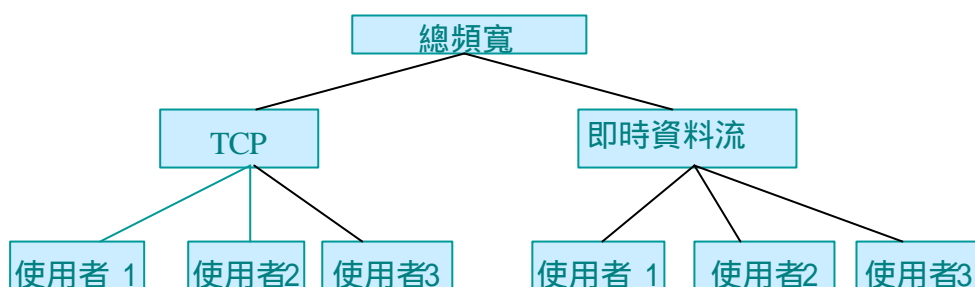


圖 3.9 排程器四頻寬分配階層圖

### 3.3 模擬環境

我們模擬環境的建立是用 NS(Network Simulator)來建構，建立環境分成四個部份，無線區域網路，通道模型，排程器，估測器。在網路環境方面，我們假設使用者位於無線區域網路中，經由無線界面和網際網路下載網際網路上的資訊。在無線區域網路和通道模型方面我們沿用在第二章中所建立的模型，我們將加入在本章節中所設計的排程器和重傳次數的控制，我們所關心的加入即時資料流之後服務品質的需求是否滿足，以及對網路整體的影響，我們將觀察的部份分為二個：

- (1)即時資料流加入後對 TCP 的效能是否有所影響，對網路整體使用率是否有影響。
- (2)即時資料流的延遲變異是否能達到合理的範圍，即時資料流的封包遺失率是否滿足需求。

觀察模擬結果是否達成設計目標，當設計目標有所衝突時，以第二個目標為首要目標。

### 3.3.1 模擬環境描述

模擬環境是 IEEE 802.11 的網路，考慮由網際網路的伺服器傳送資料至無線環境的使用者端，在 IEEE 802.11 的無線網路中，在無線上至每一個使用者間的錯誤率是各自獨立此會隨時間變化，我們將比較以上提出幾種設計組做比較，我們將傳送資料型態分為兩類

(1) TCP：IP 封包設為大小為 512 bytes。

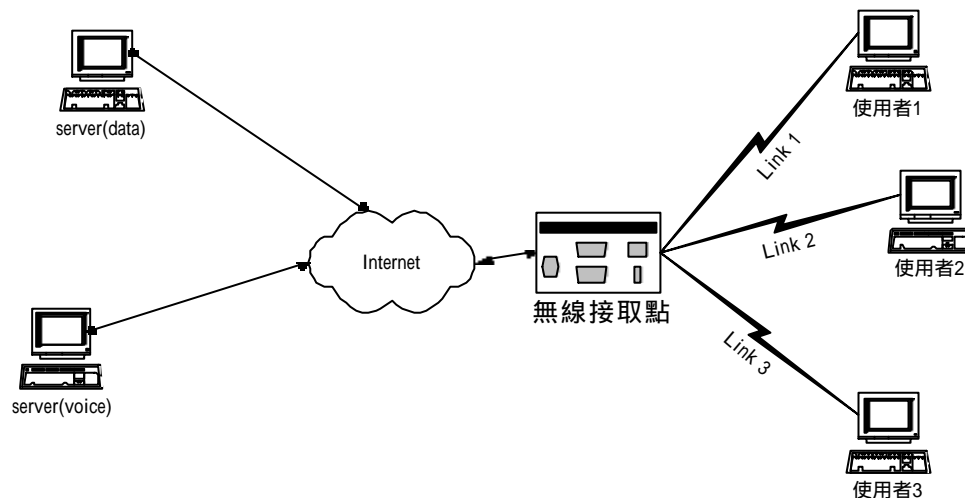


圖 3.10 模擬網路架構圖

(2) 即時資料流，用來模擬 VoIP，我們由 3.1.1 的語音編碼規格中，我們選擇了 G.729，所用的 IP 封包大小為 50 bytes，扣除 IP, UDP, RTP 的 header 剩 10 bytes，因此我們可以知每 10 ms 傳送一個 50 bytes 的 IP 封包(為達成 G.729 8kbps)，因此可視為 40kbps 的固定速率資料流。

在網路參數設定通道變化模型我們採用二元通道模型，設定參數如



表 3.2 所示，通道狀態分為 good 和 bad 兩個狀態，各對應封包遺失率 0%和 100%，兩個狀態間的變換時間採指數分配，狀態平均停留時間為 1.0sec 和 0.1 sec。估測器方面二元通道狀態估測器，網路最大重傳次數  $R_{\max}$  設 8。無線區域網路上下行無線頻寬為 750kbps。在 CCWRR 的參數如表 3.3。

通道狀態	參數	值
good	狀態停留平均時間(sec)	1.00
	封包遺失率	0%
bad	狀態停留平均時間(sec)	0.10
	封包遺失率	100%

表 3.2 通道模型參數

參數	數值
$f_g$ (good 狀態權數)	8
$f_b$ (bad 狀態權數)	1

表 3.3 依通道狀態傳輸權重式循環法參數設計

### 3.3.2 模擬結果

模擬時間是 20 秒，服務型態分為資料傳輸(TCP)和即時資料流，

觀察 TCP 和即時資料流的效能。我們首先討論觀察的方向。在 TCP 效能方面，加入即時資料流後，觀察整體 TCP 良流通率，以此作為 TCP 服務品質是否保障的指標。

在即時資料流的使用效能方面，重視的是延遲變異和封包遺失率。對即時資料流而言，延遲變異一直是一個很重要的參數，延遲變異過大時，我們就必須在使用者端使用較多緩衝和較慢的播放速度，即時資料流的效能也因此下降。封包遺失率對即時資料流而言也是一個重要的效能參數，當封包遺失率超過界限時，語音會造成失真，視訊會有馬賽克無法辨識。在表 3.5、3.6、3.7 所提的封包遺失率是針對 IP 層而言，不考慮應用層或 RTP 的封包遺失率，當 IP 層的封包遺失率已經過大時，比較上層的封包遺失率也相對變大。對即時資料流的效能而言，不單單考慮封包遺失率，當封包延遲變異度過大也不堪使用，因此，在表 3.8 中，我們將延遲變異大於某設定時限的封包亦納入封包遺失率的考慮中，我們將這個有效封包遺失率做為即時資料流的效能參數，我們將模擬結果區分為 TCP 和即時資料流兩方面來討論是否達成我們的設計目標。

在 TCP 效能的討論方面如表 3.4，為我們設計的八種架構下的 TCP 良流通率(kbps)，依服務類型傳輸即時資料流依通道狀態高優先權排程器(排程器四)的網路整體 TCP 良流通率為 510~530kbps，相對於幾種架構的組合，擁有較高的 TCP 良流通率。在即時資料流是否要重傳限制的討論上，由結果知，無論在何種架構下，對於即時資料流做重傳次數的限制都會比不限制的效能來的佳。

類別		使用者1 TCP良好流 通率 (kbps)	使用者2 TCP良好流 通率 (kbps)	使用者3 TCP良好流 通率 (kbps)	網路整體TCP 良好流通率 (kbps)
排程器	即時資料流控制				
排程器一	重傳次數不限制	168.844	166.171	166.854	501.869
	重傳次數限制	163.960	190.070	157.467	511.497
排程器二	重傳次數不限制	189.286	138.452	160.563	488.302
	重傳次數限制	177.226	145.447	170.090	492.764
排程器三	重傳次數不限制	143.236	141.910	148.623	433.769
	重傳次數限制	154.111	170.412	156.583	481.106
排程器四	重傳次數不限制	173.809	167.739	177.005	518.553
	重傳次數限制	191.518	179.176	158.894	529.588

表 3.4 網路 TCP 效能圖 (kbps)：在設計各種架構下的 TCP 良好流通率(kbps)。

## (2)即時資料流的效能

表 3.5~3.7 顯示了即時資料流的效能，由兩方面來分析，延遲變異和封包遺失率，封包遺失率是指 IP 封包的封包遺失率，由以上的模擬我們發現當延遲變異和封包遺失率呈現一種互補的現象，當封包遺失率低時，延遲變異也就變大，延遲變異大者的缺點是我們必須在使用者端使用較多的緩衝和較慢的播放率，這會存在某一臨界點，當超過時，即時資料流便不堪使用。

針對即時資料流而言，表 3.6~3.7 中的封包遺失率是指 IP 層封包接受的遺失率，當遺失率過大時，品質有就不堪使用。因此在探討即時

資料流的效能時，我們在使用者端假設存在一個播放的應用程式，且擁有緩衝區。應用程式可以看做是一個固定播放的 clock，當播放的同時，封包卻尚未抵達，我們便將這個封包視為封包遺失。由這個方式我們決定了即時資料流的效能，結果如 3.8。

在表 3.8 中可以發現無論是預留 20ms 或 30ms 的緩衝區，排程器四和排程器二加上封包重傳不限制的封包遺失率是最低的，針對排程器四而言，即對即時資料流是否要重傳限制並不會有影響。針對排程二，即時資料流加上重傳限制，會有比較低的封包遺失率。以即時資料流封包是否該設重傳限制來看，排程器一似乎看不出影響，排程器二、排程器三和排程器四我們很顯然發現，重傳限制對封包遺失率和延遲變異有產生了某種程度的效果。

類別		使用者 1平均 延遲變 異(s)	使用者 2平均 延遲變 異(s)	使用者 3平均 延遲變 異(s)	網路整體 平均延遲 變異(s)	網路整體 平均延遲 變異(ms)
排程器	即時資料流控制					
排程器一	重傳次數不限制	0.028	0.027	0.017	0.024	24.025
	重傳次數限制	0.028	0.010	0.019	0.019	18.858
排程器二	重傳次數不限制	0.008	0.022	0.011	0.014	13.813
	重傳次數限制	0.012	0.016	0.006	0.011	11.436
排程器三	重傳次數不限制	0.000	0.001	0.002	0.001	1.061
	重傳次數限制	0.000	0.001	0.001	0.001	0.659
排程器四	重傳次數不限制	0.008	0.012	0.010	0.010	9.972
	重傳次數限制	0.004	0.008	0.013	0.009	8.532

表 3.5 即時資料流延遲變異(delay jitter)。

類別		使用者1 封包遺失率	使用者2 封包遺失率	使用者3 封包遺失率	網路整體 平均封包 遺失率
排程器	即時資料流控制				
排程器一	重傳次數不限制	0.01%	0.05%	0.3%	0.15%
	重傳次數限制	0.8%	0.2%	0.2%	0.40%
排程器二	重傳次數不限制	0.05%	0.05%	0.05%	0.05%
	重傳次數限制	0.90%	2.91%	1.36%	1.73%
排程器三	重傳次數不限制	19.55%	20.65%	15.53%	18.58%
	重傳次數限制	18.04%	7.04%	17.54%	14.20%
排程器四	重傳次數不限制	0.45%	0.35%	0.35%	0.39%
	重傳次數限制	0.40%	0.30%	0.35%	0.35%

表 3.6 即時資料流封包遺失率比較表：IP 層所觀察的封包遺失率  
不將封包延遲超過時限者列入。

類別		平均延遲 變異(ms)	平均封包遺失 率
排程器	即時資料流控制		
排程器一	重傳次數不限制	24.025	0.15%
	重傳次數限制	18.858	0.40%
排程器二	重傳次數不限制	13.813	0.50%
	重傳次數限制	11.436	1.73%
排程器三	重傳次數不限制	1.061	18.58%
	重傳次數限制	0.659	14.20%
排程器四	重傳次數不限制	9.972	0.39%
	重傳次數限制	8.532	0.35%

表 3.7 即時資料流效能表：IP 層封包遺失率和延遲變異比較表

類別		延遲變異超過時限 併入之封包遺失率	
排程器	即時資料流控制	≥ 20ms	≥ 30ms
排程器一	重傳次數不限制	20.20%	16.63%
	重傳次數限制	21.82%	15.34%
排程器二	重傳次數不限制	5.48%	4.98%
	重傳次數限制	8.11%	6.64%
排程器三	重傳次數不限制	18.58%	18.58%
	重傳次數限制	14.20%	14.20%
排程器四	重傳次數不限制	5.86%	5.86%
	重傳次數限制	4.34%	3.84%

表 3.8 即時資料流效能表： 將延遲變異超過時限後，視為封包遺失

### 3.4 不完美通道模型的影響

在之前的模擬中，我們使用了完美通道模型，在本節我們使用了不完美通道模型來檢驗我們的設計，通道模型的參數如表 3.9，模擬結果如表 3.10。

通道狀態	通道狀態參數	數值
Good	狀態停留平均時間(sec)	1.0
	封包遺失率	1%
Bad	狀態停留平均時間(sec)	0.1
	封包遺失率	99%

表 3.9 通道狀態參數表

類別		網路整體TCP良流通率(kbps)		有效封包遺失率(%)	
排程器	即時資料流控制	估測器一	估測器二	估測器一	估測器二
排程器一	不限制重傳次數	534.89	461.47	17.7%	17.1%
	限制重傳次數	536.66	491.16	15.6%	16.0%
排程器二	不限制重傳次數	536.42	520.38	8.4%	7.9%
	限制重傳次數	540.49	507.88	8.7%	8.1%
排程器三	不限制重傳次數	494.58	461.69	14.6%	14.6%
	限制重傳次數	508.56	505.11	11.4%	11.4%
排程器四	不限制重傳次數	542.84	519.80	6.0%	6.9%
	限制重傳次數	543.19	524.71	4.9%	5.0%

表 3.10 在不完美通道模型下的模擬結果:估測器一為線性通道狀態估測器，估測器二為二元通道狀態估測器

模擬結果如表 3.10。首先觀察估測器效能 :在保障 TCP 服務品質方面，我們發現使用線性通道狀態估測器能達到較高的網路整體 TCP 良流通率;在保障即時資料流服務品質方面，兩種估測器的表現頗為相近，綜合結果，在估測器效能方面以線性通道狀態估測器表現最好。在排程器的設計上，以排程器四表現最好。這方面跟之前完美通道模型的結果一致，是由於排程器四對於 TCP 和即時資料流都依通道狀態來控制傳輸排程，因此即使在不完美通道模型下，效能仍比其他排程器佳。觀察對 TCP 良流通率的影響，排程器二比排程器一來的佳，與之前在完美通道模型下顯示的結果不同。在這兩個排程器的設計上，最大的不同是排程器二對即時資料流給予較高優先權，在完美通道模型下，故其 TCP 良流通率比排程器一來的低，但由於不完美通道模型使得估測器對網路狀態

的估計較不準確，給予即時資料流的優先權在較不確定的通道狀態下變得較為隨機，對 TCP 反而有利，故排程器二反而比排程器一達到較高的網路整體 TCP 良流通率。

### 3.5 觀察總結

我們在這章節中討論了如何在無線區域網路中，提供不同服務的各種排程方法，在服務的類型上，我們參考了現今網際網路的服務，將服務類型區分成 TCP 和即時資料流兩部份，對於 TCP 而言，我們在第二章中討論了幾種設計，我們採用 CCWRR 來做改進。對於即時資料流的服務類型，我們質疑原先設計的架構是否合適，是否需要修改，以及如何修改。

因此，在這章中，我們針對即時資料流的加入，設計了幾種架構，我們所使用的方法都是目前網際網路中比較常見的。即時資料流方面我們採用了 VoIP，40k 固定速率資料流，在效能方面，TCP 由於在第一章中討論，且我們亦沿用上一章的架構，因此我們把重點放在即時資料流上，我們觀察的重點有三個，1. 對 TCP 的影響，2. 即時資料流的延遲變異 3. 資料流的封包遺失率。

在 TCP 效能的評估上，排程器四的網路整體 TCP 良流通率是最佳的，其次是排程器一。而且值得注意的是，當我們使用排程器四時，不僅比一般 FIFO 來的好，更比排程器一來的佳。

在即時資料流的效能上，在即時資料上面我們定義了播放時的封包



遺失率做為效能好壞的評估。如表 3.8 我們發現排程器四和排程器二的封包遺失率都在 5%左右。對排程器四而言，是否對即時資料流做重傳限制並不會造成影響，而對排程器二來說，唯有對重傳不限制能有效降低即時資料流的封包遺失率。此外在 3.4 節使用了不完美通過模型來檢驗我們的設計，結果證實排程器四仍能有效的達到我們的設計目標。綜合各項討論，排程器四的效能最佳。

## 第四章

### 結果與展望

近年來，無線網路研發與製造產品的技術日趨成熟。無線網路的發展，所帶來的最大的優點，莫過於提供使用者較佳的方便。在基於需求的考量下，無線網路的發展是一個必然的趨勢，尤其在網際網路及寬頻網路發展日趨成熟後，無線網路的發展將成為下一波的重要技術。在面臨無線網路有線網路整合的趨勢下，針對是否能在無線環境依然有效使用網際網路，無線網路網路特性的研究，在無線區域網路中如何提升效能及保障服務品質都是重要的研究論題。我們在這篇論文中探討了這些問題，我們將問題區分成兩部份，分別為如何改善 TCP 效能的方法和如何保障多重服務品質的排程方法。

第二章中，我們探討了在無線區域網路中提升 TCP 效能的排程方法，設計的目標是達到無線區域網路最大的整體 TCP 良流通率。我們設計了可適性資料鏈結層的架構，設計的模組分為資料鏈結層重傳模組、傳送排程器、通道狀態估測模組。在估測器方面，設計了兩個通道狀態估測模組，二元狀態估測器和線性通道狀態估測器。在排程器設計上，我們依據目前常用的幾種排程器來做修正。綜合完美二元通道模型的模擬結果，我們發現"依通道狀態傳輸權重式循環法"(CCWRR)的排程器搭配二元通道估測器的表現最佳，但在不完美二元通道模型下，線性狀態

估測器表現較為強韌。在針對完美通道模型下，CCWRR 排程器參數 good 狀態權數  $f_g$  和最大重傳次數  $R_{\max}$  的討論中，發現當  $R_{\max}=8$  和  $f_g \geq 6$  會有較佳的表現，因此我們建議 CCWRR 的 good 狀態參數最好設在 6 以上。

在通道狀態變換頻率的討論中，使用一般 FIFO 並不影響 TCP 良流通率；使用 CCWRR，相較於 FIFO 都能增加效能，但在狀態變換較緩和的情形下，我們的設計可較有效增加效能。

第三章中探討了在無線區域網路中保障多重服務品質的排程方法，設計的目標是保障無線區域網路的服務品質，包含封包延遲變異度、封包遺失率。在服務型態上我們區分成一般資料傳輸服務(TCP)和即時資料流的服務，於即時資料流中採用 VoIP。我們設計了差別服務可適性資料鏈結層的架構來達成我們的設計，設計的重要模組分為資料鏈結層重傳控制模組、傳送排程器、通道狀態估測模組。在估測器設計上主要沿用第二章的設計，修改的部份是針對排程器的設計和資料鏈結層重傳控制模組。在資料鏈結層重傳模組方面，我們針對不同服務類型給予差別待遇，對即時資料流而言，我們探討了重傳次數限制和重傳次數不限制對效能的影響。在排程器的設計上，我們共設計了四種排程器。綜合所有模擬結果，無論通道是否為完美模型，以排程器四："依服務類型分流並指定即時資料流為高優先權且依通道狀態傳輸之分級排程器"的效能為最佳。

在未來的設計上，我們可以考慮將目前網際網路中的眾多服務加入討論，如 MP3、MPEG、多媒體，在未來寬頻發展日趨成熟後，在無線網路中提供這些類型的服務將勢在必行。因此，針對在無線網路中提供服

務及保障服務品質將在是未來非常值得研究的問題。在通道狀態估測模組上，本論文中設計了兩種排程器，線性通道狀態估測器和二元通道狀態估測器，在這方面的設計仍有相當大的空間。

## 參考文獻

- [1] J. C. Haartsen, "The Bluetooth Radio System," IEEE Personal Communications, Feb. 2000.
- [2] K. J. Negus, A. p. Stephens, Jim Lansford, "HomeRF Wireless Networking for the Connected Home," IEEE Personal Communications, Feb. 2000.
- [3] IEEE Standards Department. 802.11: IEEE Standard for Wireless LAN Media Access Control and Physical Layer Specification, Nov. 1997.
- [4] G. Xylomenos and G. C. Polyzos, "TCP and UDP Performance over Wireless LAN," Proc. IEEE INFOCOM '99, pp. 439-446, March 1999.
- [5] G. Xylomenos and G. C. Polyzos, "Internet Protocol Performance over Networks with Wireless Links," IEEE Network, vol. 13, No. 5, July 1999.
- [6] H. Balakrishnan, "A Comparison of Mechanisms for Improve TCP Performance over Wireless Links," Proc. ACM SIGCOMM '96, pp. 256-267, Aug. 1996.
- [7] H. Balakrishnan, V. N. Padmanabhan, S. Seshan and R. H. Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links," IEEE/ACM Trans. on Networking, Vol 5, No6, December 1997.
- [8] F. Swarts and H. Ferriira, "Markov characterization of digital

Fading Mobile VHF Channels,"IEEE Trans. Vehic.,pp. 977-985 ,November 1994.

- [9] H. Wang and N. Moayeri, "Finite State Markov Channel – a Useful Model for Radio Communication Channels," IEEE Trans. Vehic., pp. 163-171, February 1995.
- [10] P. Bhagwat, P. Bhattacharya, A. Krishma and S.K. Tripathi, "Using Channel State Dependent Packet Scheduling to Improve TCP Throughput over Wireless LANs," IEEE Wireless Network, pp. 91-102, March 1997.
- [11] S. Desilva, S. R. Das, "Experimental Evaluation of Channel State Dependent Scheduling in an in-building Wireless LAN," Computer Communication, pp. 414-421, Nov. 1998.
- [12] B. P. Crow, "Performance Evaluation of the IEEE 802.11 Wireless Local Area Network Protocol," Master's thesis, Dept. of Electrical and Computer Engineering, University of Arizona, 1996.
- [13] G. Bianchi, "Throughput Evaluation of the IEEE 802.11 Distribution Coordination Function," IN Proc. 5<sup>th</sup>. Int. Wshp. On Mobile Multimedia Communication, pp. 307-318, Berlin, 1998.
- [14] F. Calio, M. Conti, and E. Gregori, "IEEE802.11 Wireless LAN : Capacity Analysis and Protocol Enhancement," INFOCOMM '98, San Francisco, USA, 1998.
- [15] H. S. Chaya and S. Gupta, "Performance Modeling of the Asynchronous Data Transfer Methods of IEEE 802.11 MAC Protocol," IEEE Wireless Network, vol.3, pp. 217-234, 1997.
- [16] C. Fragouli, V. Sivaraman and M. B. Srivastava, "Controlled

Multimedia Wireless Link Sharing via Enhanced Class-Based Queueing with Channel State Dependent Scheduling," INFOCOM '98, vol. 2, pp572-580, 1998.

[17]S. Floyd and V. Jacobson, "Link-sharing and Resource Management Models for Packet Network," IEEE/ACM Trans. on Networking vol. 3, no. 4, pp365-386, August 1995.

[18]<http://www-mash.cs.berkeley.edu/ns/>, Homepage of UCB/LBNL/VINT Network Simulator-ns(version 2), Virtual InetrNetwork Testbed project Lawrence Berkeley National Laboratory, U.C.Berkeley.

[19]H. Schulzrinne, GMD Fokus, RFC 1890, "RTP Profile for Audio and Video Conferences with Minimal control"

[20]K. W. Lee , M. Cheng and L. F. Chang, "Wireless QoS Analysis for a Rayleigh Fading Channel," ICC '98, vol. 2, pp. 1089-1093, 1998.

[21]A. Pajares, J. C. Guerri, M. Esteve, C. Palau, A. Leon, N. Cardona, "Dynamic Frequency and Resource Allocation with Adaptive Error Control Based on RTP for Multimedia QoS Guarantees in Wireless Networks," ICC '99, vol. 2 , pp. 333-337, 1999.

