

國立台灣大學資訊工程學研究所
碩士論文

IEEE 802.11 無線區域網路之服務品質保證控制

**Quality of Service Control for IEEE 802.11
Wireless LANs**



指導教授：許清琦 博士

傅楸善 博士

研究生：徐信能

中華民國九十二年七月

Acknowledge

在此謝謝所有我要感謝，以及所有我應該感謝的人。謝謝你們！

首先我要感謝我的指導教授許清琦博士，在我學習的過程中給予督促及指導，且在論文撰寫的過程中給予專業的建議，使得研究成果得以更趨成熟。不但讓我學習到許多做研究的方法，及研究的思考角度，更讓我在無線區域網路這方面的技術著實加深了不少。在此同時也要特別感謝傅揪善教授對本論文的指導與建議。

再者，要感謝和我共同奮鬥的張明智和陳建華同學，希望他們可以繼續努力。還有我的工作伙伴們，包括何聰、周博(志昇)的支持，澤欽、正忠、元民的鼓勵，以及燕星在英文上的指導，也因為有你們的存在讓我獲得不少東西，也拓展了我的視野，更重要的是因為有你們而添加了許多的笑聲及歡樂。

感謝我的家人，榮煥、鳳蘭、國鐘、美雯、國媛、榮厚、佳琳及佳暉與各位長輩的鼓勵及深切的期許，讓我能堅持對於自我的要求。並分擔我的壓力，分享我的喜悅。在這段日子不斷的給我鼓勵及精神上的支持。

再來，我要感謝我的愛犬們-多莉、皮皮、小皮。感謝牠們陪我度過碩士生活中的每一天，感謝牠們陪伴我完成論文。雖然牠們有時候會耍耍性子，雖然牠們有時候會調皮搗蛋；雖然皮皮在我快畢業的期間離我遠去，但是牠們依然是我最佳的戰友，謝謝牠們。

最後也最重要的是，感謝辛苦撫育我並培養我接受高等教育的父母親，謝謝您們毫不保留的付出，謝謝您們寬容我的任性不羈，更謝謝您們耐心的給我空間與時間去成長蛻變，讓我學會去體會生命歷程中的每一點滴。並願將這小小的成果與我在天上的母親分享。

謹獻上這小小的成果，並再一次獻上最誠摯的謝意，謝謝您們。

徐信能 謹誌於

國立台灣大學資訊工程研究所
2003/7/28

摘要

IEEE 802.11 無線區域網路 (Wireless Local Area Networks, 簡稱 WLANs) 標準在媒體存取控制 (Media Access Control, 簡稱 MAC) 制定了分散式協調功能 (Distributed Coordination Function, 簡稱 DCF) 和集中式協調功能 (Point Coordination Function, 簡稱 PCF) 等兩種不同的資料傳輸方式。DCF 採用避免碰撞之載波偵測多重存取 (Carrier Sense Multiple Access with Collision Avoidance, CSMA/CA) 的方式進行資料的傳輸，在一般的情況下，雖然可以讓大部份的訊框 (Frame) 避免不必要的碰撞，然而網路的負載變大，仍免不了會讓訊框碰撞的機會變大，如此不但造成資料流量 (Throughput) 的降低，同時讓資料的傳送時機變得更難以預測，所以此方式只適合用來傳送非即時性的資料。而 PCF 採用輪詢 (Polling) 的方式進行資料的傳輸，並由網路協調者 (Point Coordinator, 簡稱 PC) 安排工作站 (Station, 簡稱 STA) 的資料傳送，以及傳送的順序及時機，因而資料被傳送的時間比較容易掌控，即使如此，由於 PC 對於想要進行輪詢的 STA 沒有做數目上的限制，結果將造成資料傳送的時間無法得到保證，所以此方式還是無法讓即時性的資料可獲得即時的傳送。為此，我們針對原來 PCF 機制的設計進行改善，結果不但能準確的掌控資料傳送的時間，並且可以依照傳送的資料是聲音 (Audio) 或影像 (Video)，給予不同的頻寬保證，所以新的機制非常適合用來傳送即時性的多媒體 (Multimedia) 資料，也由於提供了服務品質保證 (Quality of Service, 簡稱 QoS)，故此機制命名為 Quality of Service-PCF (簡稱 Q-PCF)。

我們將以模擬實驗的結果來證明 Q-PCF 的確可以達到以下幾點保證：第一、提供不同優先權 (Priority) 等級的資料進行傳輸，有定速率 (CBR) 及變速率 (VBR) 兩種，並保證在相同的時間，優先權較高的資料會比優先權較低的資料先進行傳送。第二、讓想要獲得服務的 STA，可以在有限時間內得到服務，並不會因為網路負載變大而無法獲得服務。第三、讓已經獲得服務的 STA，資料可以在有限的時間內進行傳送，並不會因為網路負載變大而產生延遲的現象。第四、保證 STA 的使用頻寬不會因為網路負載變大而減少。

雖然由實驗的結果我們可以知道 Q-PCF 確實可以達到 QoS，但是對於一些假設性的問題，將造成與 IEEE 802.11 的標準無法相容，譬如像是為了要完成資料即時傳輸的目的，便假設所有的 STA 沒有省電 (Power Save) 的機制，另外為了要增加網路的資料流量，而在有基礎建設 (Infrastructure) 的網路架構下，假設相鄰的 STA 彼此之間可以直接進行資料的傳輸，這些似乎與 IEEE 802.11 的標準完全不同。為此，我們將進行深入的探討，並提出一些修正，讓 Q-PCF 與 IEEE 802.11 之間更具相容性，然而經過修正過後的機制命名為 Enhanced Q-PCF (簡稱 EQ-PCF)。

EQ-PCF，除了要讓 WLANs 更適合多媒體的傳輸，對於與 IEEE 802.11 的相容性和提升網路的資料流量也是我們的研究重點之一，最後本研究將會再以模擬實驗的數據來證明我們的推論及所有的研究成果。

關鍵字：IEEE 802.11、無線區域網路、媒體存取控制、分散式協調功能、集中式協調功能、避免碰撞之載波偵測多重存取、輪詢、多媒體、服務品質保證、定速率、變速率

Abstract

The WLANs international standard of IEEE 802.11 has defined two mechanisms for implementing the Media Access Control (MAC); one is distributed coordination function (DCF) and another is point coordination function (PCF). The DCF mechanism employs the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) strategy to provide data transmission. In general, the majority of frames transmitted via CSMA/CA strategy would not cause collision; but the probability of collision occurrence would get greater as the network traffic becomes larger. In that case, the network throughput isn't only lower but also the latency of transmitting data successfully would be more unpredictable. Thus DCF mechanism is only appropriate for non-real time data service. In PCF mechanism, the latency of transmitting data service is predictable since in PCF mechanism, there is a pointer coordinator (PC), which shall be implemented at the AP, to decide the order and timing of data transmissions of STAs via polling method. However, it results in that the latency of data transmission is unable to be guaranteed since PC doesn't apply any limitations to the number of STAs which want to join the polling list of PC. Thus PCF mechanism is unable to provide real-time transmission for real-time data. According to these reasons, we improve the PCF mechanism to provide the exact timing and period of data transmission and bandwidth guarantees upon different transmitted data, such as audio and video data. Thus this proposed mechanism is appropriate for real-time multimedia transmission. We nominate this mechanism as Quality of Service –PCF, abbreviated as Q-PCF, since it provides Quality of Service.

We design a simulation program to see if Q-PCF provides QoS and has the following functions. First, the different priorities for variety data services that include constant bit rate (CBR) and variable bit rate (VBR) data services are supported. The data with higher priority are guaranteed to be transmitted before lower priority data. Second, the STA attempting to acquire the QoS service should be served in a bounded time regardless of the loading of network. Third, the data are guaranteed to be transmitted within finite time regardless of network loading. Last, the bandwidth of some STA is guaranteed regardless of network loading. With providing these functions, Q-PCF mechanism can be called as QoS data service supporting protocol.

Although Q-PCF mechanism provides QoS data service, it is limited by too many hypotheses and this results in incompatibility with the IEEE 802.11 standard. For instance, all STAs in Q-PCF mechanism must disable power-saving mechanism, which is a critical issue for portable computers and equipments, to arrive to real-time

data transmission. Another hypothesis is that the two adjacent STAs attempting to transmit data must recognize each other and their radio waves range must cover the other one in order to increase the network throughput and this hypothesis violates the IEEE 802.11 standard. Hence we give a further discussion and provide some amendment to become more compatible with IEEE 802.11. We nominate this revised mechanism as Enhanced Q-PCF, abbreviated as EQ-PCF.

EQ-PCF mechanism provides more compatibility with IEEE 802.11 and increases the network throughput besides that it is more appropriate for the real-time multimedia transmission.

keywords : IEEE 802.11, WLAN, MAC, DCF, PCF, CSMA/CA, polling, multimedia , QoS, CBR, VBR

目錄索引

Acknowledge	i
摘要.....	ii
Abstract	iv
目錄索引.....	vi
圖目錄.....	viii
表目錄.....	x
第 1 章 導論.....	1
1.1. 研究動機.....	1
1.2. 研究目的.....	1
第 2 章 背景與相關文獻.....	3
2.1. 無線區域網路.....	3
2.2. 分散協調式功能.....	4
2.2.1. 載波偵測.....	5
2.2.2. 訊框間隔.....	5
2.2.3. 後退時間.....	6
2.2.4. DCF 存取程序.....	7
2.3. 集中協調式功能.....	8
2.3.1. CFP 結構與時機	8
2.3.2. PCF 存取程序	10
2.4. 相關文獻.....	11
第 3 章 Q-PCF	13
3.1. CFP 結構與時機	13
3.2. 訊框格式.....	13
3.3. 優先權程序.....	14
3.4. 碰撞程序.....	15
3.5. 輪詢程序.....	16
3.6. 頻寬分配程序.....	17
3.7. 允入控制.....	19
第 4 章 模擬模式 (Simulation Model).....	23
4.1. 交易模式 (Traffic Models).....	23
4.1.1. 資料 (Data) 交易模式	23
4.1.2. 固定速率 (CBR) 交易模式.....	24

4.1.3. 變動速率 (VBR) 交易模式.....	24
4.2. 事件驅動機制 (Event-Driven Scheme)	25
4.2.1. 資料結構.....	27
4.2.2. DCF 有限狀態機.....	28
4.2.3. Q-PCF 有限狀態機	33
4.2.4. IEEE 802.11 有限狀態機	42
第 5 章 模擬結果.....	47
5.1. DCF 的資料流量	47
5.2. Q-PCF	48
5.2.1. 允入控制能力	48
5.2.2. DCF 的影響	48
5.2.3. CBR 之 ε 的影響	49
5.2.4. VBR 之 ε 的影響	50
5.2.5. 資料流量和資料遺失率的關係.....	51
第 6 章 EQ-PCF.....	53
6.1. 註冊階段.....	53
6.2. 資料傳送.....	54
6.3. 省電管理.....	56
6.4. DCF 存在的必要性.....	57
第 7 章 效能評估.....	59
7.1. EQ-PCF 有限狀態機.....	59
7.2. 模擬結果.....	60
7.2.1. 註冊時間.....	61
7.2.2. 資料流量.....	62
第 8 章 結論與未來努力的方向.....	63
參考文獻.....	64
附錄 A.....	66

圖目錄

圖 2.1 IEEE 802.11 網路架構.....	4
圖 2.2 MAC 架構.....	5
圖 2.3 超級訊框.....	5
圖 2.4 DCF 基本存取方法.....	6
圖 2.5 後退時間.....	7
圖 2.6 RTS、CTS 和 NAV 的關係.....	8
圖 2.7 CFP 和 CP 的關係.....	9
圖 2.8 Beacon 和 CFP 的關係.....	9
圖 2.9 Beacon 的延遲.....	10
圖 2.10 PCF 訊框轉送.....	11
圖 3.1 超級訊框的結構.....	13
圖 3.2 訊框格式.....	14
圖 3.3 優先權程序.....	15
圖 3.4 碰撞程序.....	16
圖 3.5 輪詢程序.....	17
圖 3.6 需求頻寬.....	18
圖 3.7 剩餘頻寬分配.....	19
圖 3.8 加入輪詢方法的比較.....	20
圖 3.9 允入控制.....	20
圖 3.10 允入控制演算法.....	21
圖 4.1 資料結構變數.....	27
圖 4.2 DCF 有限狀態機.....	29
圖 4.3 碰撞的處理.....	30
圖 4.4 Q-PCF 有限狀態機.....	34
圖 4.5 模擬程式的結果.....	35
圖 4.6 Q-PCF 在 PCF 的有限狀態機.....	36
圖 4.7 Q-PCF 在 DCF 的有限狀態機.....	41
圖 4.8 IEEE 802.11 在 PCF 的有限狀態機.....	43
圖 4.9 IEEE 802.11 在 DCF 的有限狀態機.....	45
圖 5.1 DCF 的資料流量.....	48
圖 5.2 允入控制的能力.....	48
圖 5.3 DCF 的 λ 對 PCF 和 QPCF 的影響.....	49
圖 5.4 CBR 的資料流量和資料遺失率的關係.....	50
圖 5.5 VBR 的資料流量和資料遺失率的關係.....	51
圖 5.6 效能評估.....	52
圖 6.1 超級訊框的結構.....	54

圖 6.2 訊框格式.....	54
圖 6.3 RF 的涵蓋範圍.....	55
圖 6.4 距離的量測.....	55
圖 6.5 資料傳輸路徑.....	56
圖 6.6 訊框格式.....	57
圖 6.7 M-POLL 訊框格式.....	57
圖 7.1 EQ-PCF 有限狀態機.....	59
圖 7.2 平均註冊時間.....	61
圖 7.3 效能評估.....	62

表目錄

表 4.1 系統參數.....	23
表 4.2 CBR 參數.....	24
表 4.3 VBR 參數	25

第1章 導論

隨著無線區域網路的蓬勃發展，加上傳輸速度不斷的提升，在無線區域網路上要傳送聲音、影像等即時性的多媒體資料，將會逐漸成為一個熱門的議題，然而以目前的 IEEE 802.11 而言，如果要傳送即時性的多媒體資料，似乎是有困難，原因並不在於頻寬的不夠，而是在於資料的傳輸過程中無法給予傳送時間及傳送頻寬的保證。為此，我們不但要證明 Q-PCF [6] 機制可以達到以上兩項保證，同時還要能夠針對不同優先權等級的資料給予不同的頻寬保證，此外 EQ-PCF 更強調與 IEEE 802.11 之間的相容性。

1.1. 研究動機

IEEE 802.11 所提出的 DCF 完全是採用公平競爭的方式進行資料傳輸，所以在設計上並沒有考慮到優先權 [4] 的問題，相對的更不可能提供即時性的資料傳輸，並且如果參與競爭的 STA 數目太多，會造成資料流量嚴重的下降。PCF 是採輪詢的機制，似乎已經考慮了優先權的問題，但是至於什麼樣的 STA 可以參與輪詢，並沒有明確的加以規定，所以優先權在這裏似乎變的不具任何意義，此外以即時傳送資料的能力而言，由於 PC 對於要參與輪詢的 STA 數目也沒有加以限制，如此將會造成參與輪詢的 STA 數目過多，即使 STA 已經被 PC 列為輪詢的成員，但是由於所有參與輪詢傳送的 STA 必須在一個免競爭週期 (Contention-Free Period, 簡稱 CFP) 的時間內完成，所以還是免不了會發生 STA 無法在限定時間內傳送資料的困境，相對的 STA 也無法根據動態的需求，來要求 PC 給予傳送時間與傳送頻寬的保證。此外由於每個 STA 一次最多只能送出一個媒體存取控制服務資料單元 (MAC Service Data Unit, 簡稱 MSDU)，其 MSDU 的大小又有所限制，如此的特性根本無法真正做到即時傳輸的要求。為此，我們設計了新的機制，不但能讓網路的資料流量增加，而且最重要的是要讓傳送時間及傳送頻寬獲得保障，並期望可以達到以即時性的多媒體資料傳輸為目的，而且與 IEEE 802.11 相容。

1.2. 研究目的

目前的 DCF 和 PCF 除了無法真正做到即時傳輸的要求外，似乎還有其它的部分仍有討論的空間，為此我們設定了最終的目的，主要有三個：

目的一：讓無線區域網路傳送即時的多媒體資料

在不增加網路負載的情況下增加服務品質的保證，所謂服務品質的保證包括以下四點：第一、支援不同優先權等級的資料進行傳送。第二、讓想要參與輪詢

的 STA，保證在有限時間內完成輪詢的登記。第三、對於已經完成輪詢登記的 STA，給予傳送時間的保證。第四、除了給予傳送時間的保證外還需要給予傳送頻寬的保證。除了以上這些保證外，對於剩餘頻寬也可以進行更有效率的分配，以便讓無線區域網路更適合傳送即時性的多媒體資料。

目的二：讓無線區域網路的資料流量提升

由於任何的資料都需要透過擷取點 (Access Point, 簡稱 AP) 的轉送，即使包括相鄰間的 STA 彼此間要互相傳送資料也是如此，這樣的結果，將會造成頻寬的浪費，事實上這個問題並不是完全無法克服，而所遭遇的最大問題，會是發生在省電的機制上，因為 STA 彼此間在互相的傳送資料前，傳送者並不會知道接收者是否進入省電模式 (Power Save Mode, 簡稱 PS Mode)，而且如果此時接收者已經進入省電模式，將會造成資料的無法傳送。在此，如果能克服省電的問題，就可以讓彼此間相鄰的 STA 直接進行資料的傳輸，如此便可以讓頻寬能進行更有效率的分配。

目的三：讓新的機制與 IEEE 802.11 更具相容性

任何新的理論如果可以已經存在，並且廣泛被使用的機制相容，便可以提高其實用性，所以在這篇論文當中，不但要讓無線區域網路傳送即時性的多媒體資料，更讓無線區域網路的資料流量提升，同時在此也強調與 IEEE 802.11 的相容性。

針對目的一，我們將透過 Q-PCF 的機制來完成，針對目的二與目的三我們將透過 EQ-PCF 的機制來完成。

第2章 背景與相關文獻

WLANs 和傳統的有線網路 (Wired LANs) 之間最大的不同點，就在於使用了不同的傳輸媒介，由原先以實體線路為主的傳輸媒介轉換為以光或無線電波為主的傳輸媒介，如此讓網路的建置變得更具有彈性，此外 STA 也可以在移動當中使用，這的確也讓整個網路變得更具有機動性，基於以上的優勢，因而造就了目前 WLANs 的產品能在全世界蓬勃的發展，但是由於 WLANs 在訊號傳輸的品質上較難掌控及頻寬遠不如 Wired LANs 的事實，因此 WLANs 的確還有相當大的成長空間，基於此原因，IEEE 802.11 規範便是以制定適合無線區域網路傳輸為主的通訊協定之一，涵蓋範圍包括實體層 (Physical Layer, 簡稱 PHY) 以及媒體存取控制 (Media Access Control, 簡稱 MAC)，簡單而言，PHY 最主要是負責傳送及接收訊框的工作，在傳送方面，接收來自 MAC 的訊框，經過調變、編碼處理後傳送出去，在接收方面，將接收到的訊框，經解調、解碼後再傳給 MAC 處理，而 MAC 最主要的工作則是負責控制傳輸媒介的存取方式，讓資料可以獲得正確無誤的傳送及接收。

為了可以讓 WLANs 能夠提供最佳的服務，IEEE802 委員會自 1997 年 6 月為無線區域網路制定規範以來，到目前為止，仍不斷的在 PHY 及 MAC 上進行修正，如附錄 A 所示。在 PHY 的部分有 802.11b 和 802.11a 與 802.11g。802.11b 的傳輸速度由原來 802.11 的 1~2 Mbps 提升到 11Mbps，802.11a 與 802.11g 的傳輸速度更提升到了 54Mbps，此外在 MAC 部分也做了蠻多的修正，如 802.11f 草案針對漫遊 (Roaming) 的議題進行探討，802.11i 草案則是針對加密 (Security) 的議題進行探討，並觀察這個發展趨勢，可以得知在 MAC 的部分所修正的地方大多與傳輸速度沒有什麼關係。但是其實傳輸速度的快慢不只有決定在 PHY，事實上在 MAC 的部分也扮演著相當重要的角色，我們可以做個小小的比喻，如果 PHY 是一台性能絕佳的賽車，MAC 就是一個賽車好手，彼此間如果沒有良好的搭配，是無法使系統發揮到最大的功效。為此，這篇論文便是以 MAC 為發點，進行深入的探討。

2.1. 無線區域網路

IEEE 802.11 是一個以單一頻道 (Single Channel) 為主的環境進行資料傳輸，意謂的在同一個時間內只允許一台 STA 或 AP 進行資料的傳輸，而 STA 或 AP 在資料傳輸前要先握有控制權。然而就網路架構而言又可區分為無基礎建設 (Ad Hoc) 及有基礎建設 (Infrastructure) 兩種，如圖 2.1 所示，並使用相同的基本存取協定。無基礎架構的無線區域網路主要是提供 STA 和 STA 之間進行點對點的資料傳輸，傳送端的 STA 將資料送出後，接收端的 STA 便可以直接進行資

料的接收，由於資料不可以透過第三者轉送，因此網路變的無法繼續擴充，但是因為較具彈性，所以 Ad Hoc 比較適合在臨時性的場合中使用，如臨時性會議。所謂的有基礎建設，指的是在分散式系統的網路，會存在一個擷取點 (Access Points, AP)，一台 AP 除了要負責提供一群 STA 在無線上進行資料的存取與轉送，我們把這一群 STA 稱為基本服務集合 (Basic Service Set, BSS)，更最重要的是要將數個獨立的 BSS 整合成一個擴充服務集合 (Extended Service Set, ESS) 以便和整個分散式系統的網路進行連結，如此即使是來自不同 BSS 的 STA 也可以進行資料的交換，最後整個分散式系統的網路會透過埠接器 (Portal) 與有線網路進行連結，不過就目前而言分散式系統大部分都是由有線網路所組成，所使用的通訊協定是以乙太網路 (Ethernet) 為主，所以目前埠接器的工作便由 AP 所取代。

由於 Ad Hoc 的網路架構，只可以使用 CSMA/CA 的機制進行資料的傳輸，而 Infrastructure 的網路架構則是可以透過 AP 的控管，使用輪詢的機制進行資料的傳輸，為了完成以即時為目的的資料傳輸方式，所以這篇論文便是以 Infrastructure 為出發點，進行深入的探討。

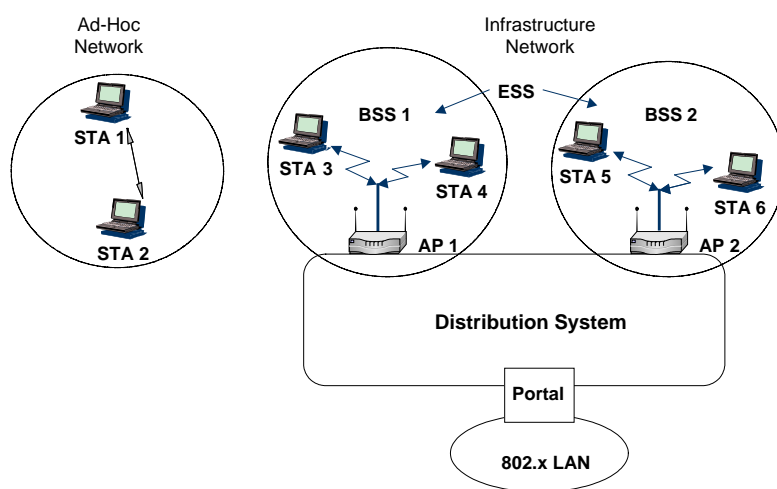


圖 2.1 IEEE 802.11 網路架構

2.2. 分散協調式功能

IEEE 802.11 的 MAC 提供了 DCF 和 PCF 二種資料存取方法:DCF 是透過競爭 (Contention) 的方式來進行資料的傳輸，PCF 則是透過免競爭 (Contention Free) 的方式來進行資料的傳輸，因此 PCF 不會像 DCF 有訊框碰撞的情形發生，而 PCF 只可使用在 Infrastructure 的無線區域網路中，這是因為整個傳送機制需要靠 PC 來進行掌控。如圖 2.2 所示，PCF 是透過 DCF 來完成。

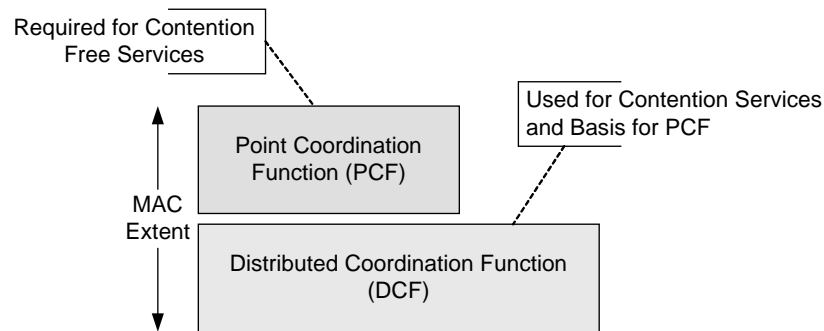


圖 2.2 MAC 架構

DCF 和 PCF 可以同時存在系統內，只是分別在不同的時間內使用，DCF 被使用在競爭週期 (Contention Period)，PCF 被使用在免競爭週期 (Contention Free Period)，這兩個週期合稱為一個超級訊框 (Superframe)，如圖 2.3 所示。

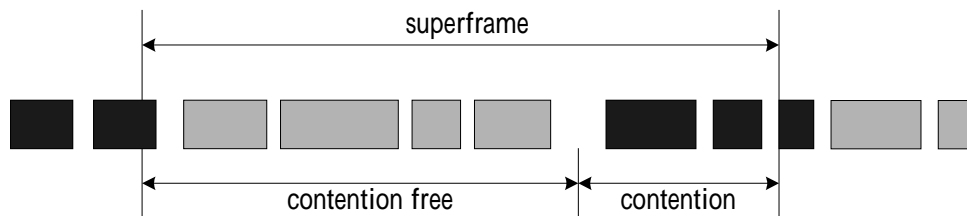


圖 2.3 超級訊框

2.2.1. 載波偵測

DCF 使用 CSMA/CA 的載波偵測技術，判斷該頻道是否處於忙碌或空間的狀態，如果所偵測到的訊號強度在標準值以下，就表示該頻道目前正處於空間的狀態，此時 STA 或 AP 便可以利用該頻道來進行資料的傳輸。相對的，若偵測到的訊號強度在標準值以上，則表示該頻道目前正處於忙碌的狀態，此時 STA 或 AP 必需延緩訊框傳送的時間，直到發現該頻道是空間的，才能夠進行訊框的傳送。

2.2.2. 訊框間隔

IEEE 802.11 根據不同的訊框給予不同的等待時間，就是所謂的訊框間隔 (Interframe space, 簡稱 IFS)，在等待時間結束的同時才可進行訊框的傳送，而不同的等待時間，又可區分為不同等級的優先權，如圖 2.4 所示，第一，短訊框間隔 (Short IFS, 簡稱 SIFS)，負責使用在需要做立即回應的訊框，在傳送訊框前，必須等待的時間，如回覆訊框 (ACK)。第二，PCF 訊框間隔 (PCF IFS, 簡稱 PIFS)，負責使用以 PCF 的方式傳送訊框前，必須等待的時間。第三，DCF

訊框間隔 (DCF IFS, 簡稱 DIFS)，負責使用以 DCF 的方式傳送訊框前，必須等待的時間。此外還有延長訊框間隔 (Extended IFS, EIFS)，負責使用在重送訊框時所必須等待的時間。優先權由大到小依序為 SIFS、PIFS、DIFS、EIFS。由於優先權等級越高的訊框在等待時間越短的情況下，會比優先權等級較低的訊框，更容易將訊框送出，如此不但可以讓優先權較高的訊框獲得較快的服務，同時可以避免不同優先權等級的訊框發生碰撞，但是對於 DCF 而言，相同優先權等級的訊框發生碰撞的機會仍然無法避免，因為相同優先權等級的訊框在等候了相同的訊框間隔之後，如果發現這段時間內該頻道是處於空閒的狀態，就會同時將訊框送出造成碰撞。為解決這個問題，IEEE 802.11 規定在 DCF 中，訊框傳送前必需等待一段由亂數決定的時間才將訊框傳送出去，這個時間就稱為後退時間 (Backoff Time)，由於 STA 或 AP 產生的後退時間可能不同，所以訊框發生碰撞的機會就會降低。

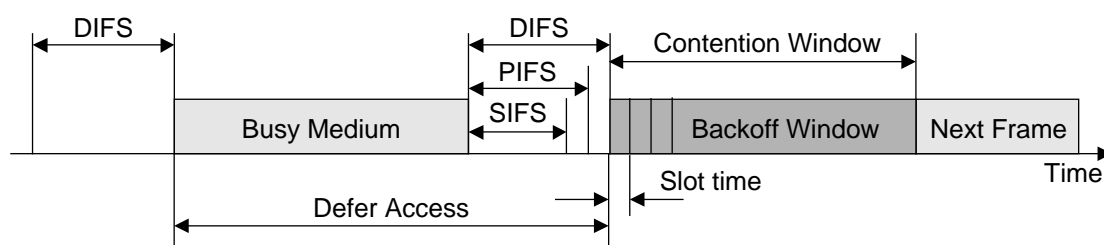


圖 2.4 DCF 基本存取方法

2.2.3. 後退時間

在 DCF 中，傳送者 (包括 STA 或 AP) 在傳送訊框前，即使偵測到頻道處於空閒的狀態也不能立即傳送訊框，必須開始等待一個 DIFS 的時間。在等待的過程中，如果頻道仍然處於空閒的狀態，則可立即傳送訊框。如果傳送者在傳送訊框前，偵測到頻道處於忙碌的狀態，或在等待 DIFS 時間的過程中，如果有其它傳送者先行傳送訊框，則必須等待訊框傳送結束後再繼續等待一個 DIFS 時間。如果在等待的過程中，仍然沒有其他傳送者要傳送訊框，緊接著便進入競爭視窗 (Contention Window, 簡稱 CW)。此時傳送者會產生隨機的 Backoff Time，如圖 2.5 所示，此 Backoff Time 會在 CW 中隨著時間遞減，傳送者必須等到該 Backoff Time 減為零時才能進行訊框的傳送。但是如果在遞減的過程中，只要有其他傳送者處於傳送訊框的狀況，就會將 CW 結束，此時就必須暫停 Backoff Time 遞減的工作，等待下次再進入 CW 時才繼續進行遞減。

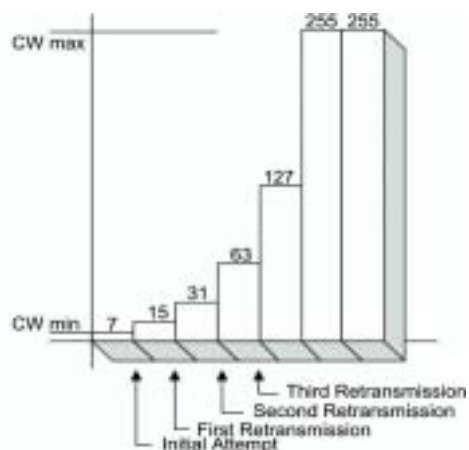


圖 2.5 後退時間

2.2.4. DCF 存取程序

IEEE 802.11 在 DCF 中除了使用 CSMA/CA 的技術來減少訊框碰撞的機會，同時使用了 RTS/CTS 的機制來減少訊框碰撞的成本，其基本原理就是在傳送者要進行資料訊框傳送前，會先行送出要求傳送的控制訊框 (Request to Send, 簡稱 RTS)，而接收者在收到這個控制訊框後，則在等一個 SIFS 後會立即回應允許傳送的控制訊框 (Clear to Send, 簡稱 CTS)。如果傳送者能收到來自接收者所回應的 CTS 訊框，則表示傳送端所傳送的 RTS 並沒有發生碰撞，此時傳送者才可以將資料訊框送出。由於 RTS 只有 20 Byte，CTS 只有 14 個 Byte，所以即使訊框發生碰撞，其碰撞成本都會比資料訊框發生碰撞來的要低。

此外利用 RTS 訊框所攜帶的持續時間 (Duration) 欄位可以用來預估資料傳送結束的時間，如圖 2.6 所示，此時所有聽到此 RTS 訊框的所有的 STA 或 AP 都要根據 Duration 欄位來進行網路配置向量 (Net Allocation Vector, 簡稱 NAV) 的設定。相同的，在接收者回送的 CTS 訊框中所攜帶的 Duration 欄位，其內容也是用來預估資料傳送結束的時間，此時所有聽到此 CTS 訊框的所有 STA 或 AP 都要根據 Duration 欄位來進行 NAV 的修正，此時由於在 NAV 值不等於零之前不可以傳送訊框，因此所有傳送者間所傳送的訊框發生碰撞的機會就會降低。

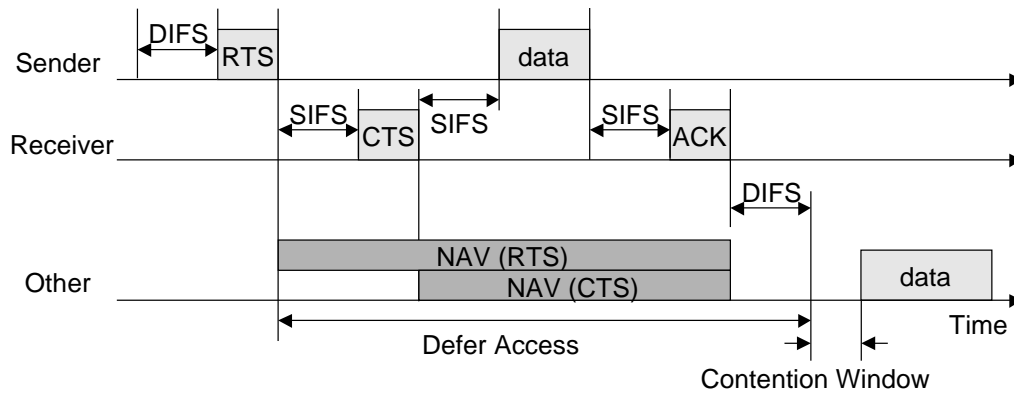


圖 2.6 RTS、CTS 和 NAV 的關係

由於 DCF 採用 CSMA/CA、Backoff Time、RTS/CTS 和 NAV 這四個機制來減少訊框在傳送前或傳送中發生碰撞的次數，或當訊框發生碰撞時可以讓碰撞成本降低，即使如此，當網路負載變大，由於訊框碰撞的機會也變大，此時情況似乎會變得更難掌控，如此將造成傳送者無法將資料在有限的時間內送出。不但如此，假設此時有 STA_i 要透過 AP 把資料送給 STA_j，若此時 AP 已經收來自 STA_i 的資料，因為不知道 STA_j 是否已經進入省電模式，所以會先把接收到的資料存入緩衝器 (Buffer) 內，並以 Beacon 訊框不斷的進行通知 STA_j 的動作，直到 STA_j 從省電模式醒來，再以 CSMA/CA 進行回應，此時的 AP 在收到 STA_j 的回應後才會再以 CSMA/CA 的機制與全部想要進行資料傳送的 STA 共同競爭後，才有可能把資料送給 STA_j。由以上這麼多不確定的因素可以知道，此機制非常不適合用來傳送即時性的資料。

2.3. 集中協調式功能

IEEE 802.11 除了採用 DCF 以競爭的方式來傳送訊框，同時採用 PCF 以支援免競爭的方式來傳送訊框，並以傳送具有時限性 (Time Bounded) 的訊框為主。其原理主要是由需要傳送即時資料的 STA 向 AP 進行聯結 (Association) 成功後，PC 會把該 STA 的聯結 ID (Association Identifier, 簡稱 AID) 登錄在輪詢名單 (Polling List) 中，接著 PC 會在進入 CFP 時，依據這個登錄在輪詢名單的 AID 次序進行輪詢，此時只有獲得 PC 授權輪詢的 STA 才可以進行訊框的傳送。

2.3.1. CFP 結構與時機

在 PCF 的 CFP 中不需要藉著 CSMA/CA、Backoff Time 和 RTS/CTS 等機制來減少訊框的碰撞，最主要的是因為輪詢的機制中不會有訊框碰撞的問題發生，其運作原理為 STA 如果要參與輪詢，首先要向 PC 進行登錄，並且要在 CP 間以

DCF 進行登錄，STA 要送出聯結 (Association) 或重新聯結 (Reassociation) 兩種訊框向 PC 進行登錄，而取消登錄資料，STA 則要送出重新聯結 (Reassociation) 訊框向 PC 進行取消登錄，如圖 2.7 所示。如果 STA 登錄成功，便要等候 PC 通知後才可進行資料的傳送，所傳送的資料最多以一個 MSDU 為限。

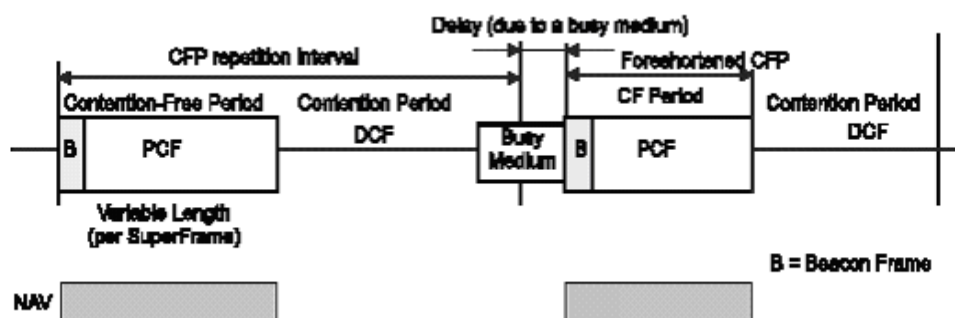


圖 2.7 CFP 和 CP 的關係

STA 由 AP 所發出 Beacon 訊框內所帶的 CFPMaxDuration 參數來得知免競爭週期的長度為何，而在執行的過程中如果免競爭週期的長度大於 Beacon 訊框的傳送間隔 (Beacon Interval)，AP 還是會在免競爭週期間送出 Beacon 訊框，此時 STA 可以由 AP 所發出 Beacon 訊框內帶的 CFPMaxDuration 參數來得知此免競爭週期剩餘的時間為何，此時只要沒有加入輪詢名單的 STA 不可把資料送出。

如圖 2.8 所示，一個超級訊框等於兩個 DTIM 期間，也就每經過兩個 DTIM 期間，就必須開始啟動一個免競爭週期，而一個 DTIM 期間等於三個 Beacon 期間。也就是一個超級訊框的長度等於六個 Beacon 期間。協調者在此期間將傳送六個 Beacon 訊框，由圖中我們可以得知免競爭週期大約是 2.5 個 Beacon 期間。

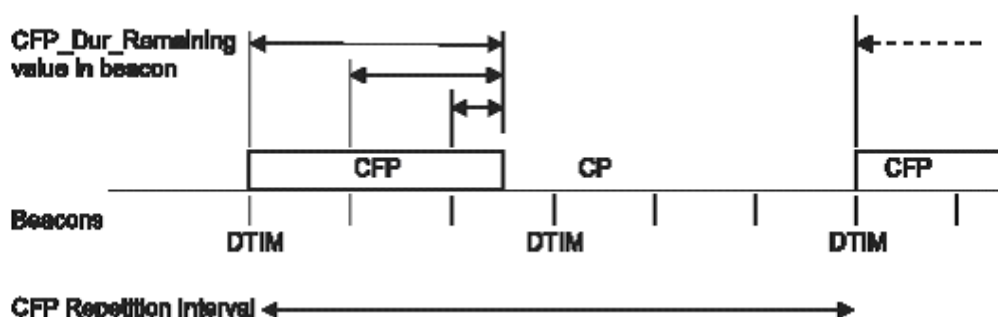


圖 2.8 Beacon 和 CFP 的關係

由於在預定傳送 Beacon 訊框的時間 (Target Beacon Transmission Time, 簡稱 TBTT) 到來時，因為其他訊框還在傳送資料，導致傳送 Beacon 訊框的時間可能會被延後，這使得此免競爭週期必須被迫減少，而縮短的時間就是 Beacon

訊框被延後傳送的時間。此外事實上 PC 真的會如原先預期的結果，在 CFPMaxDuration 的時間結束免競爭週期，所以此時的剩餘時間的參數 CFPDurRemaining，一定要扣除 Beacon 訊框被延遲的時間才可保證免競爭週期會在 CFPMaxDuration 時間結束，而延遲時間最大等於 RTS 訊框+CTS 訊框+MSDU+ACK 訊框等等，如圖 2.9 所示。除此之外如果輪詢的 STA 數目不多，免競爭週期是可提前結束。

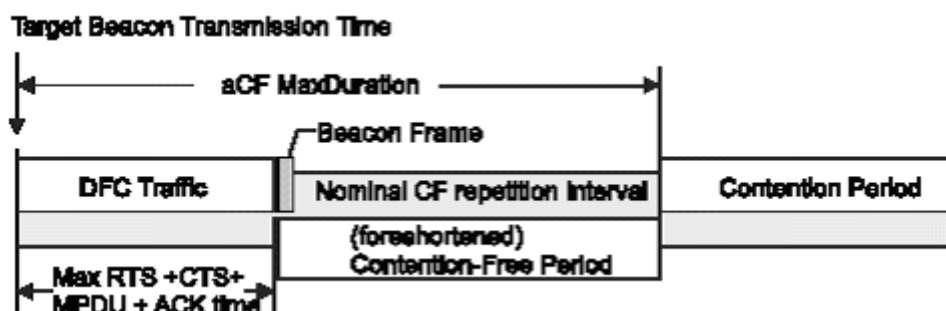


圖 2.9 Beacon 的延遲

2.3.2. PCF 存取程序

PC 依輪詢名單的順序以 CF-Poll 訊框通知已經加入輪詢名單的 STA 可以進行資料的傳送，接著該 STA 如果沒有資料要傳送則回應 Null 訊框，如果有資料要傳送則回應 Data。如果 PC 也有資料要傳送給該 STA，則會在發出 CF-Poll 的訊框順便送出資料，而接著該 STA 如果沒有資料要傳送則回送 ACK 訊框，如果有資料要傳送則回送 Data 和 ACK 訊框。如果 AP 收到了 Data，此時 AP 除了會回送 ACK 訊框外，同時發出會 CF-Poll 訊框，加上如果又有資料要傳给对方，會順便送出資料，如圖 2.10 所示為免競爭週期中訊框傳送的典型範例，PCF 傳送訊框的次序通常是先由 PC 送給 STA，然後由 STA 送給 PC，如此重複交換進行。

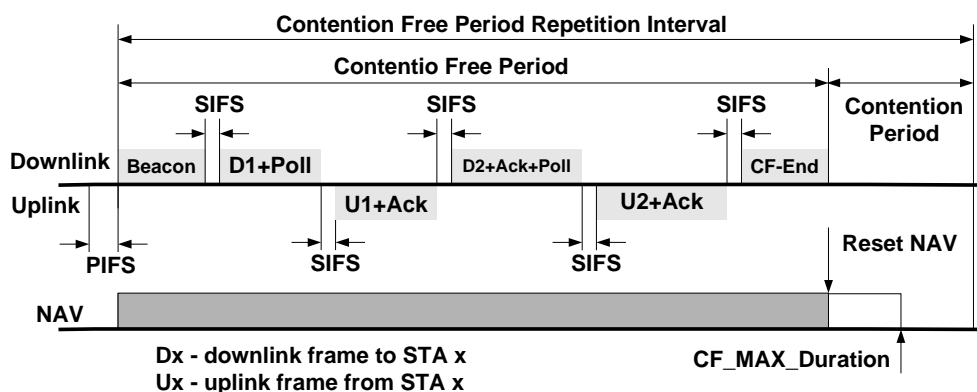


圖 2.10 PCF 訊框轉送

PCF 似乎是個蠻有效率的機制，但是在此得考慮四個問題：第一，STA 想要加入輪詢名單要透過 DCF 機制，但是網路負載一旦變大，要加入輪詢名單的時間會變的難以預測。第二，當 AP 收到來自傳送者所送出的資料，此時如果接收者並沒有加入輪詢名單，就會發生 AP 無法在 CFP 中把資料送出的困境，而是先放入緩衝器，等待 CP 的到來，再以 DCF 的方式送出，如此資料便無法即時的被送出。第三，如果接收者已經是參與輪詢的成員，但是由於已經在這個 CFP 中傳送過資料，如果此時又有其它 STA 要把資料送給這個 STA，因為 AP 在同一個 CFP 中不會重複傳送兩次資料給同一台 STA，所以 AP 便無法在這個 CFP 中把資料傳給這個接收者，而是先放入緩衝器，等待下一個 CFP 的到來，再以輪詢的方式把資料送出。第四，如果傳送者已經是參與輪詢的成員，而且在這個 CFP 也還沒有傳送過任何資料，但是由於 IEEE 802.11 沒有對加入輪詢的成員數目進行限制，所以有可能造成傳送者還沒有把資料送出，CFP 的時間已經結束，由以上這些結果可以得知 PCF 也無法保證資料能在有限的時間內被送出。

2.4. 相關文獻

為了改善 PCF 的效能，文獻 [17] 提出 STRP 機制。在 STRP 中，PC 將 STA 分為 Active Ring 和 Idle Ring 兩類。PC 在輪詢的過程同時針對兩台 STA 進行通知的動作，其中一個屬於 Active Ring，另一個屬於 Idle Ring。被通知的 STA 如果是 Idle Ring 就必須回應 Jamming Noise，一旦 PC 聽到 Jamming 訊號，就會將這個 Idle Ring 的 STA 加入 Active Ring 的行列。其好處是縮短加入 Active Ring 的時間，但是 STRP 會遭受 Near-Far Problem 的困擾。文獻 [10] 提出 SuperPoll，最主要的目的，在於減少通知的次數，並進而達到增加頻寬的使用率，其原理是 PC 在一個超級訊框中只需要發出一通知即可，因為在 SuperPoll 的通知訊框中會包含所有在輪詢名單裡頭的成員及次序，而每個被通知的成員就必等到自己專屬的時段才能傳送資料，這和 IEEE 802.11 對於在輪詢期間內的成員都要各別進行通知，如此比較起來的確是可以減少頻寬的浪費，但是如果在 CFP 的過程中對於已經加入輪詢名單的 STA 又不想傳送資料，卻無法順利的退出輪詢名單的情況下，即使如此，其它的 STA 也不可以提前把資料送出，在此還是會造成頻寬的浪費。

為了要支援 QoS，系統必定也要支援優先權，文獻 [8] 提出 Priority DCF，其主要原理是讓優先權高的 STA 等待較短的 Backoff Time。但是由於 CW 有部分重疊的現象，當兩個優先權較高的訊框發生碰撞，極有可能會讓優先權較低的 STA 先進入輪詢名單的現象發生。文獻 [19] 提出 DBASE 的機制，可去解決優先權較低的 STA 會先加入輪詢名單的現象，其原理是讓傳送及時資料的 STA 在

PIFS 和 DIFS 之間的 Slots 競爭。所以 DBASE 假設傳送及時的 CW 的大小「固定」，而且只有 3 個 Slots，並且設定 $DIFS = SIFS + 5 * SlotTime$ ，如此少的 Slots 在很多 STA 要加入輪詢名單的同時有可能會造成更大的碰撞，其次如果幾乎沒有 STA 要加入輪詢名單，那麼 DCF 的 STA 也要浪費這些 Slots 的時間。

為了要支援 QoS，系統除了要有支援優先權的機制外，同時系統一定也要讓一些傳送及時資料的 STA 很快的加入輪詢名單，其碰撞解析 (Collision Resolution) 的方法會比碰撞避免 (Collision Avoidance) 的方法來的好些，因為碰撞解析的方法可以根據 STA 的反應來作適當的調整。像文獻 [18] 直接修改初始值的機制 [15] 的作法，利用丟銅板來解決碰撞。由於「丟銅板」的過程有相當的不確定性，所以此碰撞避免的方法有可能永遠停不下來。

第3章 Q-PCF

3.1. CFP 結構與時機

如圖 3.1 所示，Q-PCF 將整個 CFP 切分為三個階段，分別為優先權階段 (Prioritization Period)，碰撞解析階段 (Collision Resolution Period)，和輪詢階段 (Polling Period)。前面的兩個階段又稱為註冊階段 (Registration Period)。在優先權階段，PC 會執行一連串的握手協定 (Handshake) 用來確保優先權高的 STA 能比優先權低的 STA 先加入輪詢名單。在碰撞解析階段，PC 會執行一連串的 Handshake 用來讓想加入輪詢名單的 STA 能在有限時間之內加入。在輪詢階段，PC 會發出一個 M-POLL (Multi Poll) 訊框用來讓輪詢名單內的 STA 可以按照次序發送資料訊框，在輪詢階段結束時，PC 會發出 CF-End 來結束整段 CFP，並重設 NAV。為了和 IEEE 802.11 相同，在此設定 CP 的最短時間為「傳送一個最大的 MPDU 及其 ACK 所需的時間」，此外值得注意的是，有可能會發生因 DCF 時間延遲而導致 CFP 時間減少。

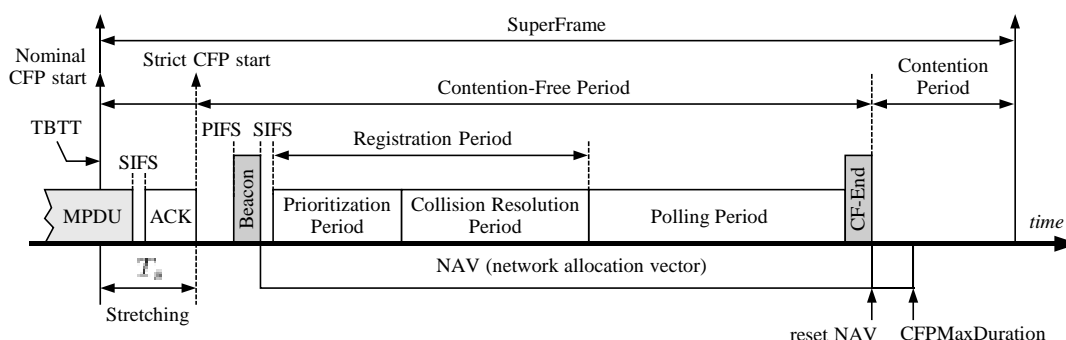


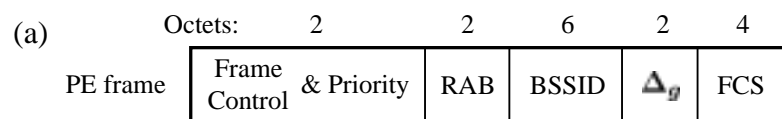
圖 3.1 超級訊框的結構

所有的 STA 於免競爭週期開始時都將其 NAV 值設為免競爭最大週期 (CFPMaxDuration)，這樣可以避免 STA 未被輪詢到卻把訊框傳出的問題發生。此外由於免競爭最大週期的時間是有限的，而註冊週期的時間如果沒有進行限制，可能會導致在輪詢名單的 STA 無法獲得頻寬的保證。因此我們必需設計動態允入控制，讓 PC 有能力決定什麼時機該終止註冊程序。

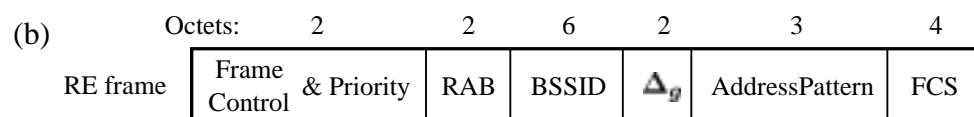
3.2. 訊框格式

Q-PCF 的 MAC 訊框格式，如圖 3.2 所示，其訊框分別為 PE、PR、RE、RR、M-POLL 及 Data 六種：PE 和 PR 使用在優先權階段，PE：PC 以優先權為主進行詢問，PR：STA 則負責回覆。RE 和 RR 使用在碰撞解析階段，RE：PC 以 Address Pattern 為主進行詢問，RR：STA 則負責回覆。M-POLL 和 Data 使用在輪詢階段，M-POLL：PC 通知 STA 傳送順序及傳送時間，Data：STA 開始進行資料的傳送。

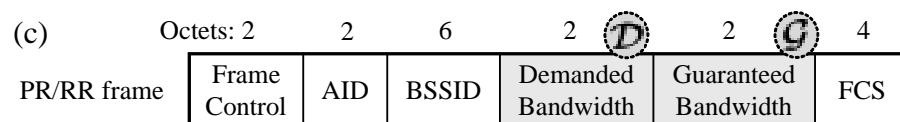
PE (Priority Enquiry)



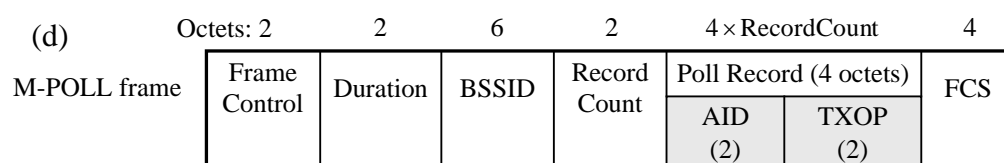
RE (Registration Enquiry)



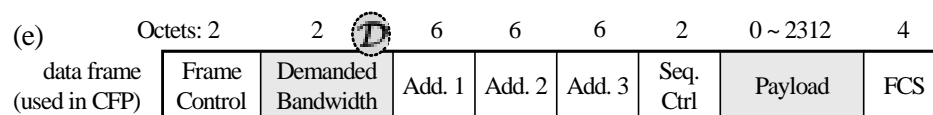
PR (Priority Response) 、RR (Registration Response)



M-POLL



Data



To DS	From DS	Add. 1	Add. 2	Add. 3	Usage
0	1	DA	BSSID	SA	AP-to-STA traffic in a BSS
1	0	BSSID	SA	DA	STA-to-AP traffic in a BSS
0	0	DA	SA	BSSID	STA-to-STA traffic in a BSS

圖 3.2 訊框格式

3.3. 優先權程序

Q-PCF 將優先權等級分為 H+1 個，從 0 到 H，如果 STA 有一連串的资料要進行傳送，而且等級比 0 還大，便有機會加入輪詢名單。以圖 3.3 為例來進行觀察，Q-PCF 是如何支援優先權。一開始 PC 會發出優先權詢問 (Priority Enquiry, 簡

稱 PE) 訊框，先調查一下有沒有優先等級為 H 的 STA 要加入輪詢名單，如果沒有的話，在 PIFS 之後，PC 繼續調查是否有優先權為 H-1 的 STA 要加入輪詢名單，如果剛好只有一台 STA 送出回應 (Priority Response, 簡稱 RR) 訊框的話，那麼 STA 便會被加入輪詢名單。PC 繼續調查優先權為 H-1 的情況，如果此時有兩台以上的 STA 同時送出回應，那麼 PC 就會發出註冊查詢 (Registration Enquiry, 簡稱 RE) 訊框，宣告進入碰撞解析階段，在碰撞解析階段，PC 會展開一連串的 Handshake 找出到底是那些 STA 發出 PR 造成碰撞。

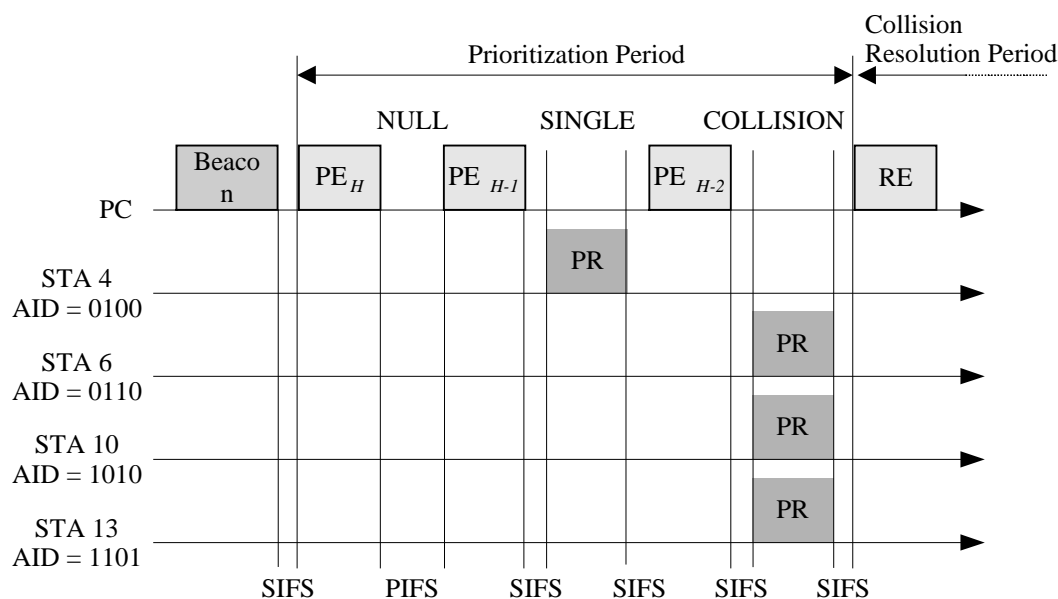


圖 3.3 優先權程序

3.4. 碰撞程序

我們所採用的方法稱為 FDS (Depth-First-Search) Traversal Dimension Tree Splitting。基本原理是利用 DFS 的方法逐步把造成碰撞的 STA 找出來，PC 會執行一連串的 Tree Splitting 演算法，一開始 PC 會發出 RE 的詢問訊框，內容包括優先權為 H-2，而且 Address-Pattern 為{***0}的 STA 請回應，如果剛好只有一台 STA 回應 RR (registration response) 訊框，那麼 STA 便會被加到輪詢名單。PC 會接著詢問 Address-Pattern 為{***1}的 STA 請回應，但是如果發生碰撞，PC 會把 Address-Pattern 的範圍減半，改為{**00}，如此依序找下去，便可找出所有造成碰撞的 STA。如圖 3.4 所示，為了把 STA 6、10、13 找出來，第一次 PC 會以 Address-Pattern 為{***0}進行詢問，此時會發生碰撞，於是第二次，PC 會以 Address-Pattern 為{**00}進行詢問，此時沒有回應，於是第三次，PC 會以 Address-Pattern 為{*0*0}進行詢問，此時 STA 10 回應，於是 PC 會把 STA 10 加入輪詢名單，於是第四次，PC 會以 Address-Pattern 為{*1*0}進行詢問，此時 STA 6 回應，於是 PC 會把 STA 6 加入輪詢名單，於是第五次，PC 會以 Address-Pattern

為{***1}進行詢問，此時 STA 13 回應，於是 PC 會把 STA 13 加入輪詢名單，於是所有碰撞的 STA 都被找出。

Q-PCF 和 IEEE 802.11 DCF 進行比較，主要是因為 DCF 採用碰撞避免的策略，這意味著 DCF 想靠著時間來解決碰撞的問題，然而 Q-PCF 採用碰撞解析的策略 [2][5][11][12][15]，這意味著，如果發生碰撞，Q-PCF 嘗試著用動態調控的方法來解決碰撞的問題。Q-PCF 的方法最糟糕的情況，會發生在大部分的 STA 想在同一個時間加入輪詢名單，並且優先權都相同。然而，這種情況發生的機率較低，而且 STA 一旦加入輪詢名單後，便不會再參與競爭。

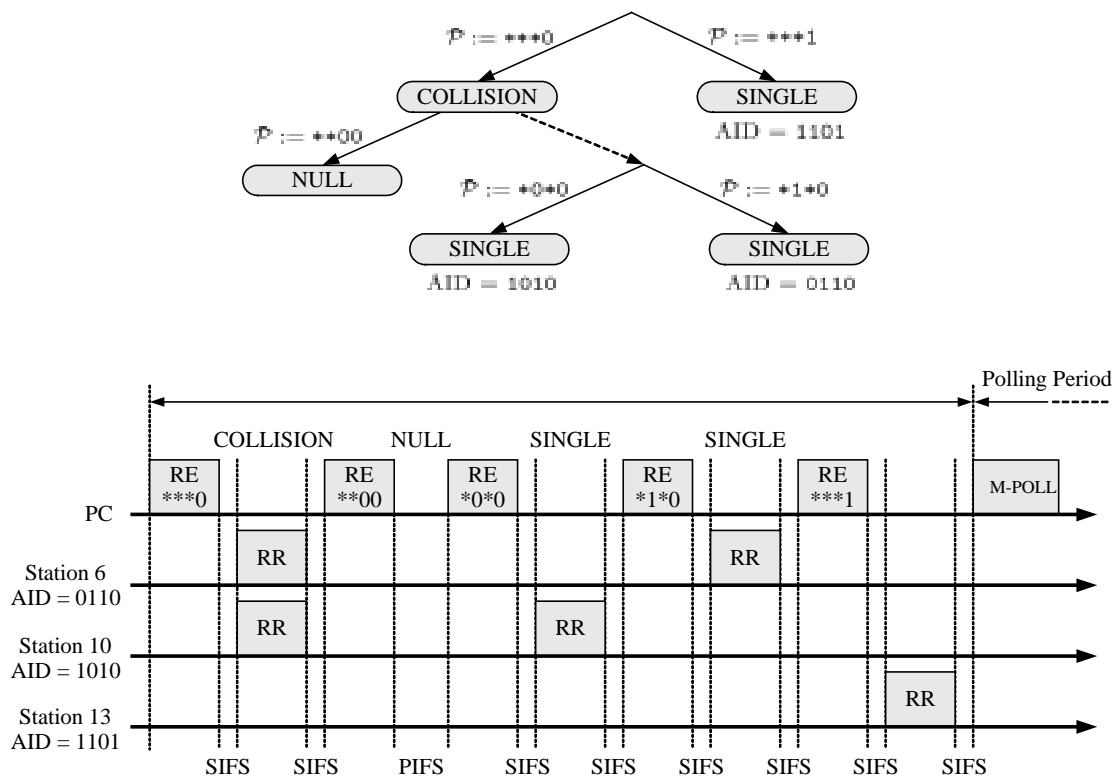


圖 3.4 碰撞程序

3.5. 輪詢程序

一旦碰撞解析階段結束，接下來 PC 就會發出 M-POLL，進入輪詢階段，M-POLL 會宣告 STA 被輪詢的次序，以及被允許傳送的時間（Transmission Opportunity, 簡稱 TXOP） [9][14]。此時所有輪詢的 STA 在這段時間都必須仔細聆聽網路的情況，在聽到前一順位的 STA 把資料傳完時，再等待一個 SIFS 後，就必須把資料送出。但是無線網路通常是處於「被干擾」的情況。如果 PC 發現某個被輪詢的 STA 沒有回應，那麼 PC 就必須在 PIFS 後介入，重新發出 M-POLL，這個 M-POLL 會包含那些還沒有傳送過資料的 STA。因為 Q-PCF 允

許 PC 在輪詢階段介入，所有 PC 能夠保有「媒介控制權」，也由於 PC 能夠隨時保有控制權，所以當參與輪詢的 STA 如果沒有回應，其它的 STA 便可以提前把資料送出，如此便不會造成頻寬的浪費。圖 3.5 顯示輪詢階段的整個運作過程，(a)發完 M-POLL 之後，STA 4、6、10、13 依序傳送資料。(b)由於 STA 10 沒反應，在 PIFS 之後，PC 奪回控制權，並重新發出 M-POLL。

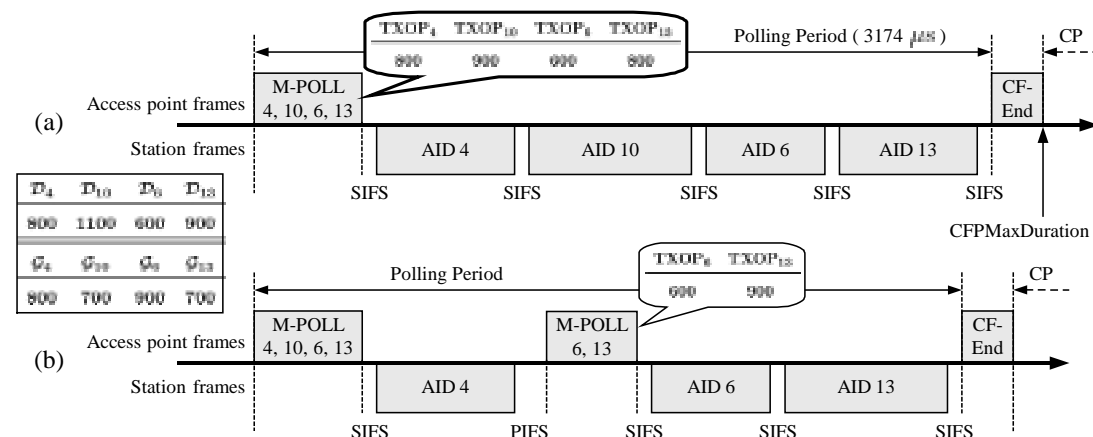


圖 3.5 輪詢程序

3.6. 頻寬分配程序

Q-PCF 嘗試支援多媒體資料的傳輸，多媒體的資料通常有二大特色：(1) 資料有時間上的限制，如果 STA 沒辦法在有限時間內將資料傳送出去，很可能造成那些沒被傳送出去的資料被丟棄。(2) 多媒體的資料量通常相當的大，所以必須持續性的傳送。而多媒體的資料通常分為定速率 (CBR) 和變速率 (VBR) 兩類。CBR 在每個 CFP 階段所需的頻寬（又稱為 Demanded Bandwidth, 簡稱 D）相同。而 VBR 在每個 CFP 階段所需的頻寬可能會不相同。為了確保獲得頻寬的保障，STA 必須在註冊的階段向 PC 提出頻寬保證 (Bandwidth Guarantee, 簡稱 G)。此時如果 $D > G$ ，那麼 STA 的頻寬需求可能無法獲得滿足。因此，VBR 必須仔細評估 G 該設定多少比較好。當然，G 愈大愈好。但是 G 愈大，輪詢名單的 STA 數目就越小，G 愈小，則資料被丟棄的機會就變大。假設 ε [16] 是 VBR 在一個 CFP 裡頭頻寬無法獲得滿足的機率；也就是說 $\Pr[D > G] < \varepsilon$ 。VBR 自行評估多少的 ε 值是可以被允許的範圍。然後再依據 ε 提出 G 值。至於 CBR，由於每次所需的頻寬都一樣，所以可令 $G(\text{CBR}) = D * (1 - \varepsilon)$ 。在目前已知文獻皆無法「針對個別的使用者」提供頻寬的保障。而 Q-PCF 利用 (D, G) 的宣告，很容易地就達成這個任務。

至於如何提出 D 和 G 呢？如圖 3.6 顯示 STA 提出 (D, G) 的過程：一開始註冊時須提出 (D, G)，往後每次在送資料的同時，必須提出「下個 CFP 所需

的 D」。另外，當 STA 想離開輪詢名單時，只要在傳送資料時順便 (piggyback) 講一聲「More Data = 0」即可。PC 看到 More Data = 0 的訊號，就會將這個 STA 從輪詢名單去除。

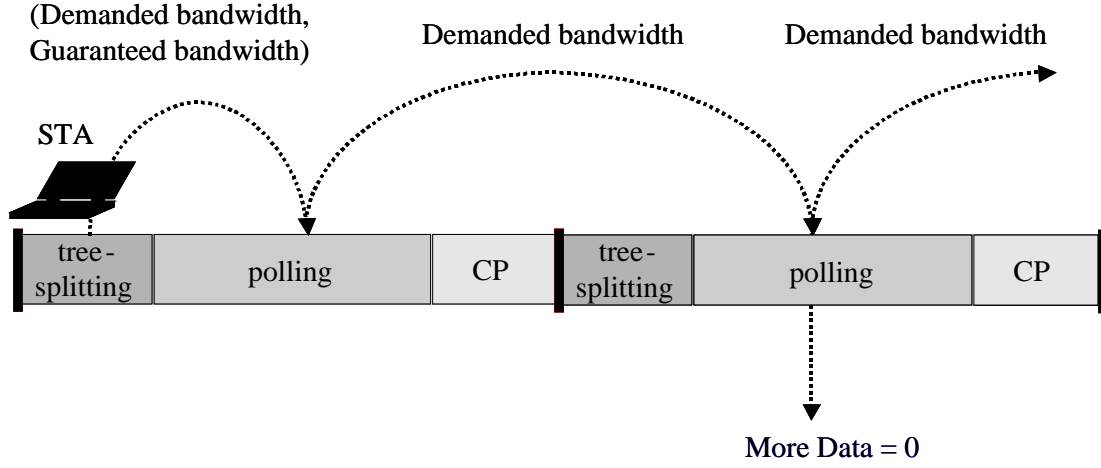


圖 3.6 需求頻寬

一旦 STA 宣告 (D, G)，PC 必須計算一下該給每個 STA 多少頻寬 (TXOP) 較為恰當。Q-PCF 所採用的策略是「按超額需求比例均分」的原則。假設輪詢名單裡頭分別為 {1, 2, ..., m}，我們把 STA i 的 (D, G) 表示為 (D_i, G_i)。PC 在進入輪詢階段之前，先計算 Y 值 (如下)

$$Y = \text{CFPMaxDuration} - (T_s + \text{PIFS} + T_{\text{beacon}} + \text{SIFS} + T_{\text{reg}} + T_{\text{M-POLL}} + \text{SIFS} + T_{\text{CF-End}})。$$

接著 PC 計算一下在輪詢階段還剩多少時間 (Residually Sharable bandwidth, 簡稱 RSB) 可以給「超額需求」的 STA。

$$RSB = Y - \sum_{i=1}^m (\min\{D_i, G_i\} + \text{SIFS})$$

有了 RSB，PC 計算 TXOP 的公式如下：

$$TXOP_i = \begin{cases} D_i & \dots\dots\dots D_i < G_i \\ \min \left\{ D_i, G_i + \left[RSB \times \frac{D_i - G_i}{\sum_{D_i > G_i} D_i - G_i} \right] \right\} & \dots\dots\dots D_i > G_i \end{cases}$$

如圖 3.7 的例子所示，STA 6 和 10 有超額需求，因為 G₆ 和 G₁₀ 都是 10 (單位)，但是 D₆ 和 D₁₀ 卻分別為 14 和 16。此時 RSB 是 5，於是將 5 按 4：6 分配

給 STA 6 和 10，於是 $TXOP_6$ 和 $TXOP_{10}$ 分別為 12 和 13。

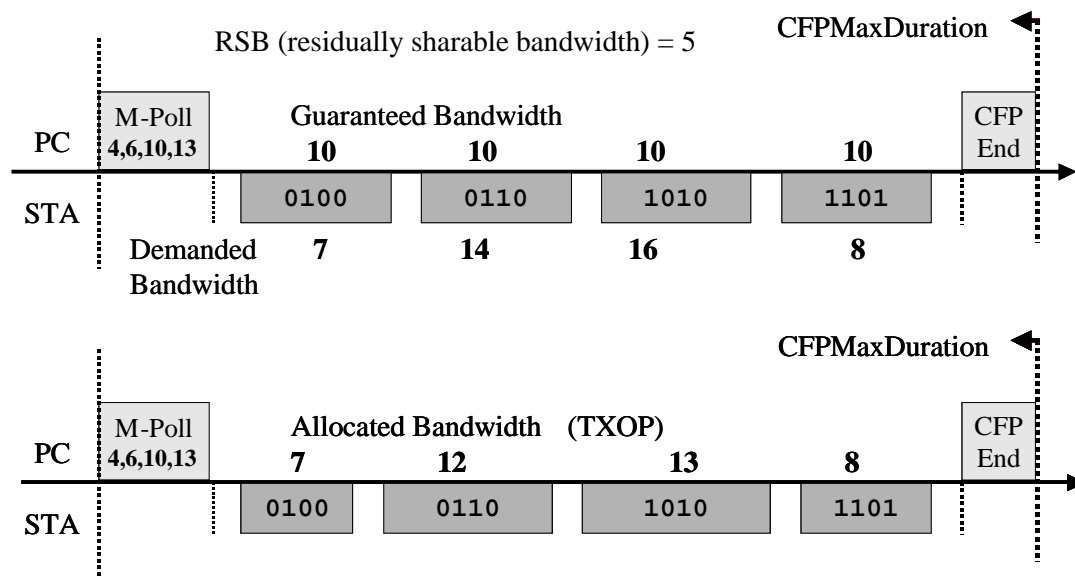


圖 3.7 剩餘頻寬分配

3.7. 允入控制

為了允諾保障 STA 基本需求的頻寬（對 STA 來說，基本需求為 $\min\{D_i, G_i\}$ ），註冊階段不可以執行太久，否則會佔用輪詢階段的預設時間，此外，PC 也必須掌控整著註冊階段，不可以因為新的 STA 加入，而導致允諾給 STA 的基本需求受損。然而，傳統的允入控制 [5][7][12] 在這裏並不適用。如圖 3.8 所示，左邊的部分為傳統作法：STA 送出頻寬保留的要求訊框，PC 發現自己所剩餘的時間不夠，便回應拒絕的訊框，如此將導致失敗的詢問會造成頻寬的浪費，我們可發現，在右邊的 Q-PCF 中，詢問都會是「有效的」，不會發生頻寬的浪費。

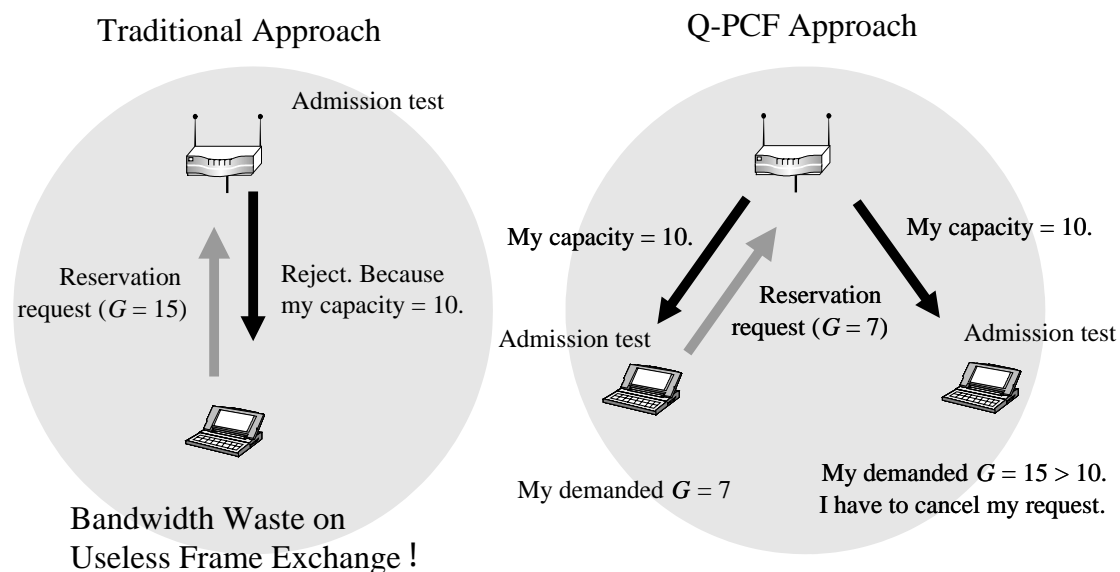


圖 3.8 加入輪詢方法的比較

如圖 3.9 所示，Q-PCF 採用比較有效率的作法，PC 定期公告剩餘頻寬（Remaining Available Bandwidth, 簡稱 RAB），STA 必須自己衡量，如果需求大於 RAB，就不可以發出 PR 或 RR，其 Q-PCF 對於允入控制的作法如下，由 PC 發出 PE 或 RE，順便公告 RAB，每個想加入輪詢名單的 STA 必須作允入檢查。如果 $\min\{G, D\} < \text{RAB}$ ，便可回應 PR 或 RR，否則必須放棄，等待下一個 CFP，PC 再根據 STA 的反應決定是否要執行下一個 PE/PR 或者 RE/RR。

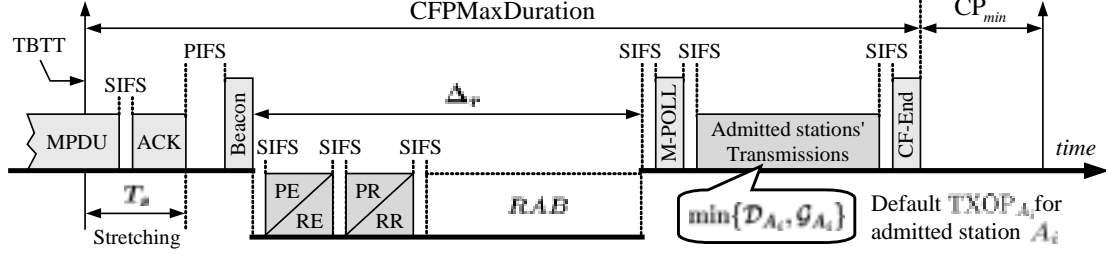


圖 3.9 允入控制

具體的演算法參見圖 3.10。注意，PC 必須在發出 Beacon 之前先計算出兩個輔助變數， Δ_r 和 Δ_g 。

$$\Delta_r = \text{CFPMaxDuration} - \left[T_s + O_{CFP} + \sum_{i \in L} (SIFS + \min\{G_i, D_i\}) \right]$$

$$\Delta_g = \text{CFPMaxDuration} - \left[\hat{T}_s + O_{CFP} + \sum_{i \in L} (SIFS + G_i) \right]$$

其中 $O_{CFP} = \text{PIFS} + T_{\text{beacon}} + T_{\text{M-POLL}} + T_{\text{CF-End}} + 2 * \text{SIFS}$ 。

```

01  After broadcasting the beacon, the PC computes  $\Delta_g$ ,  $\Delta_r$ , and the variable
     $RAB := \Delta_r - (\delta_1 + \delta_2 + 3 \times SIFS)$ ;
    /* The variable  $RAB$  denotes the remaining available bandwidth
    if the PC proceeds to the next PE/PR or RE/RR handshake. */
02  while ( $\Delta_g > 0$  and  $RAB > 0$  and (registration process is not finished)) {
03      The PC sends the PE/RE frame and announces  $(\Delta_g, RAB)$ ;
    /* On receiving the PE/RE frame, each active real-time station, say  $A_k$ ,
    takes the following admission test. */
04      if ( $\mathcal{G}_{A_k} \leq \Delta_g$  and  $\min\{\mathcal{G}_{A_k}, \mathcal{D}_{A_k}\} \leq RAB$ )
05          Station  $A_k$  replies the PR/RR frame and declares  $(\mathcal{D}_{A_k}, \mathcal{G}_{A_k})$ ;
06       $status := \text{receive}(\text{PR or RR})$ ;
    /* The PC updates the channel state variable  $status$  according to
    received PR/RR frames. */
07      switch ( $status$ ) {
08          case SINGLE:
09              The PC places the real-time station  $A_k$  on the polling list;
10               $\Delta_g := \Delta_g - (SIFS + \mathcal{G}_{A_k} + \frac{4 \times 8}{CDR})$ ;
11               $\Delta_r := \Delta_r - (\delta_1 + \delta_2 + \min\{\mathcal{G}_{A_k}, \mathcal{D}_{A_k}\} + 3 \times SIFS + \frac{4 \times 8}{CDR})$ ; break;
    /* Note that the length of the M-POLL frame will increase by 4 bytes
    (32 bits) if a new real-time station is admitted. */
12          case NULL:
13               $\Delta_r := \Delta_r - (\delta_1 + \text{SlotTime})$ ; break; // PIFS = SIFS + SlotTime.
14          case COLLISION:
15               $\Delta_r := \Delta_r - (\delta_1 + \delta_2 + 2 \times SIFS)$ ; break;
16      }
17       $RAB := \Delta_r - (\delta_1 + \delta_2 + 3 \times SIFS)$ ;
18  }

```

圖 3.10 允入控制演算法

第4章 模擬模式 (Simulation Model)

模擬器 (Simulator) 是使用 C 語言撰寫完成，除了可以精準模擬 MAC 的運作外，還可以進一步的進行效能評估 (Performance Evaluation)，所使用的方法是 Event-Driven Scheme，而模擬的參數是以 IEEE 802.11b 的直接序列展頻 (Direct Sequence Spread Spectrum, 簡稱 DSSS) 為主 [13]，模擬的時間長度是 1.8×10^8 us，如果換算成超級訊框的數目，因為每個超級訊框的長度是 25ms，所以共有 7200 個超級訊框，此外 STA 的連線數目原則上是以 256 台為主，所傳送的資料包含 Data、CBR 與 VBR 三種。

在這個模擬器中，為了能讓複雜的環境獲得簡化，我們作了一些假設：(1) 每個 STA 在初始化時便要決定所傳送的資料是屬於 Data、CBR 或 VBR 三個優先權的其中一個，而且在執行的過程中不可以更換。(2) 程式進行前，假設所有的 STA 都已經和 AP 完成聯結 (Association) (3) 忽略訊號在空中傳輸的時間，同時也忽略距離會影響訊號的品質。

4.1. 交易模式 (Traffic Models)

交易模式依優先權的等級可以區分為 Data、CBR、VBR 三種模式，彼此之間由於應用的不同，所以會產生不同的特性，在撰寫上自然就會有所差異。

4.1.1. 資料 (Data) 交易模式

優先權最低，是利用 DCF 的機制來傳送不需要即時傳送的資料，針對 Data 模式，我們另外作了一些假設：(1) STA 在進行資料傳輸前，一定要先使用 RTS 及 CTS 的機制來確認是否可以進行資料的傳輸。(2) 每個 STA 固定只能送出 2312 Byte 的資料長度。(3) 每個 STA 所傳送資料的時間間隔是利用卜氏分配 (Poisson Distribution) 來決定。其參數如表 4.1 所示。

表 4.1 系統參數

Parameter	Value	Unit
Channel bit rate	11	Mbps
Superframe length	25	ms
SIFS	10	us
PIFS	30	us
DIFS	50	us
SlotTime	20	us

RTS frame length	20	bytes
CTS frame length	14	bytes
ACK frame length	14	bytes
(CWmin, CWmax)	(31;1023)	slots
Reassociation Request frame length	38	bytes
Reassociation Response frame length	34	bytes
Beacon frame length	57	bytes
PE frame length	18	bytes
PR frame length	16	bytes
RE frame length	20	bytes
RR frame length	19	bytes
M-POLL frame length	16 + 4 * polling list size	bytes
CF-End frame length	20	bytes

4.1.2. 固定速率 (CBR) 交易模式

優先權最高，是利用 PCF 的機制來傳送需要即時傳送的資料，如 Audio，而針對 CBR 模式，我們另外作了一些假設：(1)假設傳輸速度是固定不變。(2)分為送資料 (Talkspurt) 及不送資料 (Silent) 兩種狀態，所佔用的時間分別為 1×10^6 us (40 個訊框) 及 1.35×10^6 us (54 個訊框)，並不斷的交互執行，在 Talkspurt 狀態表示 STA 要傳送資料，而在 Silent 狀態表示 STA 不要傳送資料。其參數如表 4.2 所示。

表 4.2 CBR 參數

CBR Traffic Parameter	Value	Unit
Conversation length	1.8×10^8	us
Principle talkspurt	1.0×10^6	us
Principle silent gap	1.35×10^6	us
Data bit rate (CBR)	64	Kbps
Maximum voice frame tolerable delay	25	ms

4.1.3. 變動速率 (VBR) 交易模式

優先權低於 CBR，高於 Data，也是利用 PCF 的機制來傳送需要即時傳送的資料，如 Video，而針對 VBR 模式，我們另外作了一些假設：(1)以截斷式指數分配 (Truncated Exponential Distribution) 的方式產生 VBR 的資料，因此傳輸速度不固定，而且要落在 120 Kbps 和 420 Kbps 的範圍之間。(2)根據多媒體的特性，

以指數分配 (Exponential Distribution) 的方式不斷產生需求頻寬 (Demanded Bandwidth) 的持續值 (Holding Value) [1][19]，而在每次 Holding Value 倒數計時結束後，STA 才會另外產生 VBR 的資料。其參數如表 4.3 所示。

表 4.3 VBR 參數

VBR Traffic Parameter	Value	Unit
Peak bit rate	420	Kbps
Minimum bit rate	120	Kbps
Mean bit rate	240	Kbps
Mean state holding time	160	ms
Mean video call length	1.8×10^8	us
Maximum video frame tolerable delay	50	ms

4.2. 事件驅動機制 (Event-Driven Scheme)

一般而言，新的理論並不能馬上在真實的環境中進行測試，所以如果要知道其效能為何，一定要藉助模擬器的撰寫來獲得結果，但是模擬的過程要與理論不能有所差異，否則模擬器所製造出來的結果，是完全不具任何意義，但是如果模擬的過程只是表達的方式與真實的環境有所不同，但是其模擬的過程並不與原來的理論相違背，而且模擬結果的數據是正確的，也算是個可以讓人接受的結果，而 Event-Driven Scheme 的概念就是如此。此外由於網路上的環境是隨時都在變化，如果模擬器在每個時間都要處理所有的變化，這會讓模擬器在處理上是個相當沈重的負擔，事實上我們可以想出一個辦法來解決這樣的情況，所以決定採取以事件驅動為架構的模擬器，並把這個模擬器命名為 Event-Driven Scheme，顧名思義，這是一個以事件驅動為主的模擬器，不但可以模擬在真實環境所發生的狀況，而且由於不是隨時都要針對所有 STA 或 AP 進行處理，所以在大部分的情況下，程式執行的速度會比真實環境所發生的時間還要更快。

要了解 Event-Driven Scheme 的基本運作原理，首先必須要知道 Event-Driven 中的最主要的資料結構型態 (struct)，如以下的程式碼所示，這是一個動態的資料結構，並利用鏈結串列，將所有資料存在動態資料結構上，由於可以動態的配置記憶體，所以在程式執行期間可以不需要佔用太多的記憶體。

```
struct event_list_type
{
    char state;        // block
    char event;        // process
    int clock;         // event 發生的時間
```

```

short int nodeid;    // Association ID
struct event_list_type *next;    // 指向下一個指標
};typedef struct event_list_type e_type;
e_type *first_d;

```

我們先來看 block (state) 與 process (event) 的關係，block 是一個基本的功能單位，而一個 block 需要一個或多個 process 來協助實際的工作處理，並以具體的有限狀態機 (Finite State Machines, 簡稱 FSM) 來描述整個處理過程。我們可以再舉一個實際的例子，具體的說明 block 和 process 的關係，我們可以把整個傳送 Data 的 block 叫做 SEND，而在這個 block 中會有兩件事需要處理，第一件事是要處理 Data 的傳送，第二件事是要處理來自接收端的回應，所以我們把這兩件事的處理過程叫做 process (SEND_DATA, RECV_ACK)，也就是模擬器中的 event，至於這個 event 要在什麼時間發生，接下去我們需要來看 block、event、clock 及 nodeid 之間的關係。

在初始化的過程中，我們要先設定一個叫 gclock 的時間變數，並且把值設為 0，此外每個 STA 都會先安插一個 event，而在安插 event 的同時，還要安插這個 event 屬於那一個 block，它的 AID (nodeid) 是什麼，至於這個 event 必需在何時要被執行，便要把想要發生的時間填入 clock，所以在安插的過程中要針對這個 clock 進行排序，把 clock 最小的排在第一筆，依此類推，如圖 4.1 所示，等到全部的 STA 都已安插 event，並且在結束後，程式便會開始到結構變數中取出第一筆資料，這筆資料包含了那一個 STA 要進入那一個 block 要處理什麼 process，並在取出 event 的同時要把 gclock 的變數值改為和 clock 的變數值相同，接著再按照 block 和 process 進行處理，等到處理結束後，會再產生下一個 event，接著再到結構變數找出第二筆資料進行處理。如此每個 STA 不斷的在處理完一個 event，會產生下一個 event，所以可以讓程式不斷的繼續執行到 gclock 等於預設的結束時間，程式才會停止。

此外，很重要的觀念，程式在進行 event 安插 (insert) 的同時，程式的主控權是握在 STA 或 AP 的手上，如果程式在進行 event 取出的同時，程式的主控權是握在主程式手上





 first_d->next	0x00441e20
state	1 ' '
event	1 ' '
clock	1890
nodeid	48
 next	0x00441d20
state	1 ' '
event	1 ' '
clock	2177
nodeid	63
 next	0x00441d60
state	1 ' '
event	1 ' '
clock	2281
nodeid	58
 next	0x004400e0

圖 4.1 資料結構變數

4.2.1. 資料結構

在 DCF、QPCF 及 PCF 運作中的 STA 分別有記錄自己狀態的主要資料結構，而這些資料結構與整個模擬過程息息相關。

DCF：

如以下的程式碼所示，這個資料結構，最主要是紀錄在 DCF 中的 STA 狀態。

```
typedef struct
{
    int nav;           // 存放 NAV 的值
    short int retry;   // 這裏是指 RTS 或 Reassociation 重傳的次數，預設值為 6
    short int partner; // 指定接收者
}node_type;
```

QPCF：

如以下的程式碼所示，這個資料結構，最主要是紀錄在 QPCF 中的 STA 狀態。

```
short int  IDbit[LOGMAX]; // 二進制的 AID
short int  active;         // 是否為 CBR 或 VBR 的成員
short int  pri;            // 其值等於 2 表示為 CBR，其值等於 1 表示為 VBR
int        db;            // 需求頻寬
short int  enter;         // 用以表示是否已經加入輪詢名單
```

```

short int  more_data;           // 判斷 CBR 是否要加入或退出輪詢名單
short int  more_data_count;     // CBR 加入或退出輪詢名單的計數
int        loss;                // 被丟棄的資料
short int  hold;                // 需求頻寬的持續時間 (Holding time)
int        TXOP;                // 剩餘頻寬分配
int        db2;                 // 需求頻寬加上原來緩衝器的值
double     regist_time;         // 加入輪詢名單所花費的時間
int        B[2];                // VBR 把沒有送出的資料先放入緩衝器
};

```

PCF

如以下的程式碼所示，這個資料結構，最主要是紀錄在 PCF 中的 STA 狀態。

```

struct node_state{
    short int  IDbit[LOGMAX];    // 二進制的 AID
    short int  active;           // 是否為 CBR 或 VBR 的成員
    short int  pri;              // 其值等於 2 表示為 CBR，其值等於 1 表示為 VBR
    int        db;               // 需求頻寬
    short int  enter;            // 用以表示是否已經加入輪詢名單
    short int  REASSOC_IN;       // 判斷 CBR 是否要加入輪詢名單
    short int  REASSOC_OUT;      // 判斷 CBR 是否要退出輪詢名單
    short int  succ_send;        // 記錄在這個 CFP 是否已經成功的把資料送出
    short int  unpoll_count;      // VBR 並沒有被輪詢的次數
    int        buffer;           // VBR 把沒有送出的資料先放入緩衝器
    short int  more_data;        // 判斷 CBR 是否要加入或退出輪詢名單
    short int  more_data_count;   // CBR 加入或退出輪詢名單的計數
    int        loss;             // 被丟棄的資料
    short int  hold;             // 需求頻寬的持續時間 (Holding time)
    short int  generate;          // 判斷是否已經成功的加入或退出輪詢名單
    int        db2;              // 需求頻寬加上原來緩衝器的值
    double     regist_time;       // 加入輪詢名單所花費的時間
};

```

4.2.2. DCF 有限狀態機

如圖 4.2 所示，這是一個 DCF 的有限狀況機。

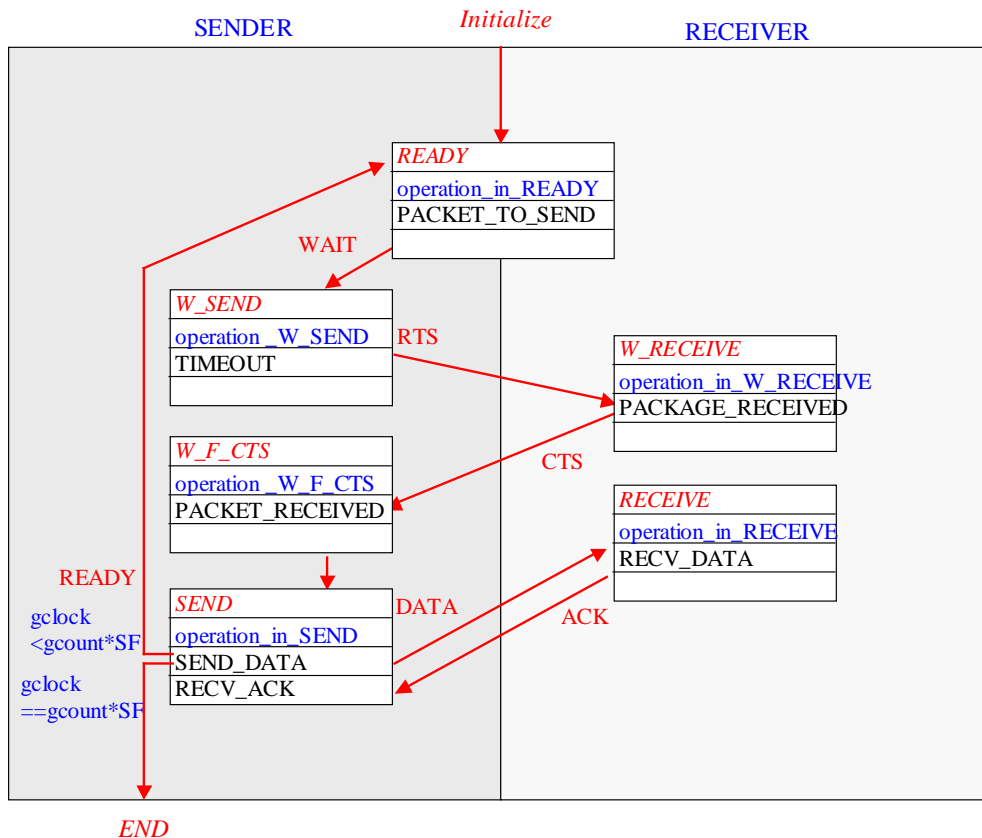


圖 4.2 DCF 有限狀態機

Initialize :

如以下的程式碼所示，以 STA 數目 (MAX_NODE) 為次數，分別以卜氏分配所產生的時間安插 backoff time 事件 (PACKET_TO_SEND EVENT)。

```
for (i=0;i<MAX_NODE;i++){
    t=(int) poisson(rate);
    event_insert_d(i,PACKET_TO_SEND,t,READY);
}
```

READY STATE :

PACKET_TO_SEND EVENT :

這個 process 最主要是讓傳送者經過 backoff time 的計算，安插 RTS 事件 (TIMEOUT EVENT)，如以下的程式碼所示，先以隨機的方式選擇接收者 (partner)，接著在時間的計算上，要把目前的時間，加上一個 DIFS 的時間，再加上以 RTS 重傳的次數當成 backoff time 的參數所產生的時間。

```
node_d[id].partner=choose_a_partner_d(id);
```



```
event_insert_d(id, TIMEOUT, gclock_d + DIFS + backoff(node_d[id].retry), W_SEND);
```

W_SEND STATE :

TIMEOUT EVENT :

這個 process 最主要是讓傳送者送出 RTS，並安插 CTS 事件 (PACKET_RECEIVED EVENT)，如以下的程式碼所示，先將 NAV 的時間與目前的時間 (gclock_d) 進行比較，如果 NAV 大於或等於目前的時間，表示 NAV 已經被設定，所以此時不可以將 RTS 送出，所以此時要重新安插 RTS 事件 (TIMEOUT EVENT)，在時間的計算上，要把 NAV 的時間 (這裏的 NAV 已經包含了目前的時間)，加上一個 DIFS 的時間，再加上以 RTS 重傳的次數當成 backoff time 的參數所產生的時間。

```
if (node_d[id].nav >= gclock_d)
    event_insert_d(id, TIMEOUT, node_d[id].nav + DIFS + backoff(node_d[id].retry), W_SEND);
```

假設如果 NAV 小於目前的時間，傳送者可以傳送 RTS，但是否會發生碰撞，要在收到 CTS 之後才會知道。如圖 4.3 所示，時間點(1)的時候，STA 1 和 STA 2 發生碰撞，可是這個碰撞會持續到時間點(2)才會結束。但是 STA 1 和 STA 2 並不知道碰撞是否有發生。另一方面，他們都認為：如果 RTS 沒有碰撞，那麼時間點(3)應該會收到 CTS 才對。假如一直到時間點(4)STA 1 和 STA 2 都沒收到 CTS，那麼 STA 1 和 STA 2 就大膽預測：應該是 RTS 發生碰撞才對，於是準備重送 RTS。至於 STA 3，他自從時間點(2)之後就發現網路是空間的狀態，沒有 STA 在傳資料，於是等個 DIFS + backoff 之後，STA 3 就在時間點(5)發送 RTS。

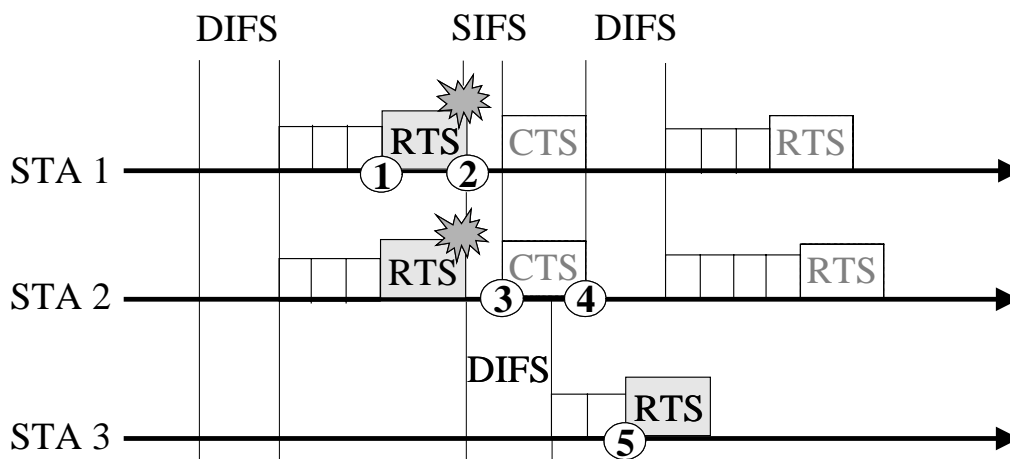


圖 4.3 碰撞的處理

但是在模擬程式，並不能模擬訊號在空中碰撞的情形，所以採用先到結構變數查看的方式，一方面不但可以解決以上的問題，另外一方面也可以減少 RTS

在模擬程式中碰撞的機會，節省程式的執行時間，所謂到結構變數查看最主要的理由，是由於 event 都是事先安排，在 event 觸發前便可以查到，所以在此很容易知道傳送者的 RTS 是否發生碰撞，如以下的程式碼所示。但是在同一個時間發生的 event 除了 RTS，還會有別的 event，所以安插的程式 (event_insert_d) 除了要按照時間發生的先後來進行排序，還要把 RTS 事件 (TIMEOUT EVENT) 排在前面。

```
while ((first_d->next != NULL)&&(first_d->next->clock == gclock_d)
      &&(first_d->next->event == TIMEOUT))
```

除了以上兩種狀況，事實上還有第三種狀況，如以下的程式碼所示，就是已經確定沒有碰撞會發生，於是設定 NAV，並且安插 CTS 的事件 (PACKAGE_RECEIVED EVENT)，在時間的計算上要把目前的時間加上傳送 RTS 的時間 (RTS_time)。

```
for(j=0;j<MAX_NODE;j++)
    node_d[id].nav=gclock_d+RTS_time+SIFS+CTS_time+SIFS+PACKET_time+SIFS+ACK_time;

event_insert_d(j,PACKAGE_RECEIVED,gclock_d+RTS_time,W_RECEIVE);
```

W_RECEIVE STATE :

PACKAGE_RECEIVED EVENT :

這個 process 最主要是讓接收者在收到 RTS 之後再把 CTS 送出，並安插 CTS 接收事件 (PACKAGE_RECEIVED EVENT)，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 的時間和 CTS (CTS_time) 時間。

```
event_insert_d(psrc,PACKAGE_RECEIVED,gclock_d+SIFS+CTS_time,W_F_CTS);
```

W_F_CTS STATE :

PACKAGE_RECEIVED EVENT :

這個 process 最主要是要處理 CTS 事件，並安插資料傳送的事件 (SEND_DATA EVENT)，如以下的程式碼所示，在時間上的計算上，是沒有任何的時間需要被計算。

```
event_insert_d(psrc,SEND_DATA,gclock_d,SEND);
```

SEND STATE :

SEND_DATA EVENT:

這個 process 最主要是讓傳送者把資料傳送出，並安插接收資料事件 (RECV_DATA EVENT)，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 再加上資料的傳送時間 (PACKET_time)。

```
event_insert_d(pdst,RECV_DATA,gclock_d+SIFS+PACKET_time,RECEIVE);
```

RECV_ACK EVENT:

這個 process 最主要是讓傳送者接收來自接收者送出的回應，再以卜式分配安插 backoff time 事件 (PACKAGE_TO_SEND EVENT)，如以下的程式碼所示，在此表示資料已經成功的被送出，所以要計算資料的數目 (succ_AP)，其實成功的被送出有兩種，一個是 STA 一個是 AP，這裏只負責計算 AP (id 為 0 表示為 AP) 的部分，另外在此還要卜氏分配產生傳送下一個資料的 event，AP 則不用需要使用卜氏分配。STA 在時間的計算上，要把目前的時間加上卜氏分配的時間。另外要把 NAV 都設為目前時間，表示所有 STA 又可以開始參與競爭。

```
if (id==0){
    t=0;
    succ_AP++;
}else{
    t=(int) poisson(rate);
}
event_insert_d(id,PACKAGE_TO_SEND,gclock_d+t,READY);

for(j=0;j<MAX_NODE;j++)
    node_d[j].nav=gclock_d;
```

RECEIVE STATE :

RECV_DATA EVENT:

這個 process 最主要是讓接收者在收到資料之後再安插回應事件 (RECV_ACK EVENT)，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 的時間和 ACK (ACK_time) 的時間。

```
event_insert_d(psrc,RECV_ACK,gclock_d+SIFS+ACK_time,SEND);
```

END :

如以下的程式碼所示，當目前時間大於所設定的時間便結束程式，但在結束前要進行資料流量 (throughput_AP) 的計算。

```
if (gclock_d>SIMU_TIME_d){  
    Throughput_AP=succ_AP*(PACKET_time-(PHYheader+MACheader));  
}
```

4.2.3. Q-PCF 有限狀態機

由於 DCF 和 PCF 使用相同的結構，如以下的程式碼所示。但是為了達到獨立的運作的目的，所以宣告成兩個結構變數。

```
struct event_list_type  
{  
    char state;          // block  
    char event;          // process  
    int clock;           // event 發生的時間  
    short int nodeid;    // Association ID  
    struct event_list_type *next;  
};typedef struct event_list_type e_type;  
  
e_type *first_d;        // 使用在 DCF  
e_type *first;          // 使用在 PCF
```

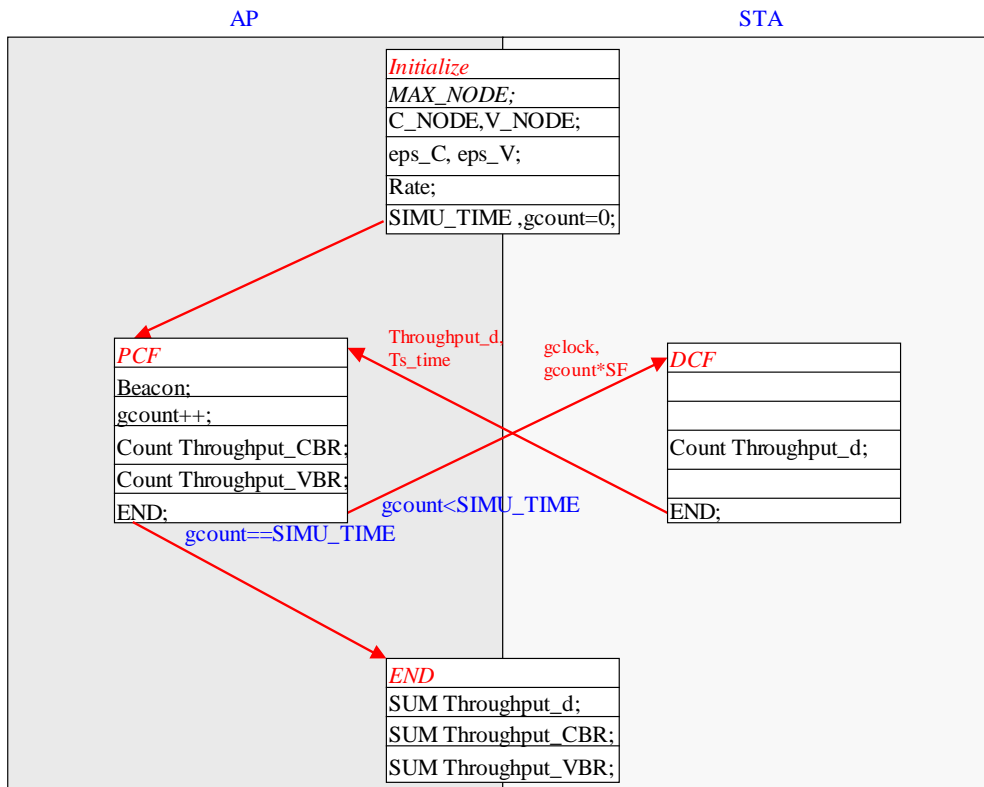


圖 4.4 Q-PCF 有限狀態機

Q-PCF 在加入輪詢名單是利用 PCF 來完成，而不是透過 DCF 來完成。如圖 4.4 所示，Q-PCF 除了 Initialize 與 END 以外共分為 PCF 與 DCF 等 2 個部分，而且互相交錯執行，並在每個 Beacon 結束後便進入 PCF 部分，而在 CFPmaxDuration 時間還未送完便會提前進入 DCF 部分，或是在 CFPmaxDuration 時間內還未送完資料也會強迫進入 DCF 部分，其傳遞的參數有兩個，一為系統目前的時間 (gclock)，一為 DCF 應該結束的時間，計算方法是以超級訊框的次數乘以超級訊框的長度 (gcount*SF)，並等待 DCF 的時間結束便要進入 PCF 部分時間，此時的傳遞參數為 Dealy 的時間 (Ts_time) 及此次所完成的資料流量 (Throughput_d)。

Initialize：

MAX_NODE、C_NODE、V_NODE 的決定：

先確定全部 STA 的總數 (MAX_NODE) 和 CBR 的數目 (CBR_NODE)，及 VBR 的數目 (VBR_NODE)，再分別以隨機的方式決定所有的 STA 屬於那一種優先權。

eps_C、eps_V 的決定：

藉著 eps_C 和 eps_V (ε_{CBR} 、 ε_{VBR}) 的設定，分別探討 ε_{CBR} 和 ε_{VBR} 與資料遺失率的關係。

Rate 的決定：

指的是資料到達率 (Arrival Rate)，依卜氏分配所產生的時間。

SIMU_TIME 的決定：

指的是模擬程式執行的時間 180 秒，由於一個超級訊框是 25ms，所以整個模擬程式執行的時間為 7200 個超級訊框。

END：

如圖 4.5 所示，程式最後的執行結果。

```
CBR # in Polling List : 100
CBR # Register Time : 4673.640000

VBR # in Polling List : 5
VBR # Register Time : 93472.800000

DCF Succ=24964244, DCF Throughput=0.138671

CBR Succ=103435620, CBR Throughput=0.574563
CBR Loss=962640, CBR ErrorRatio=0.009221

VBR Succ=14917908, VBR Throughput=0.082866
VBR Loss=0, VBR ErrorRatio=0.000000

ErrorRatio=0.008068
Throughput=0.796099
```

圖 4.5 模擬程式的結果

State machines for Q-PCF MAC access point

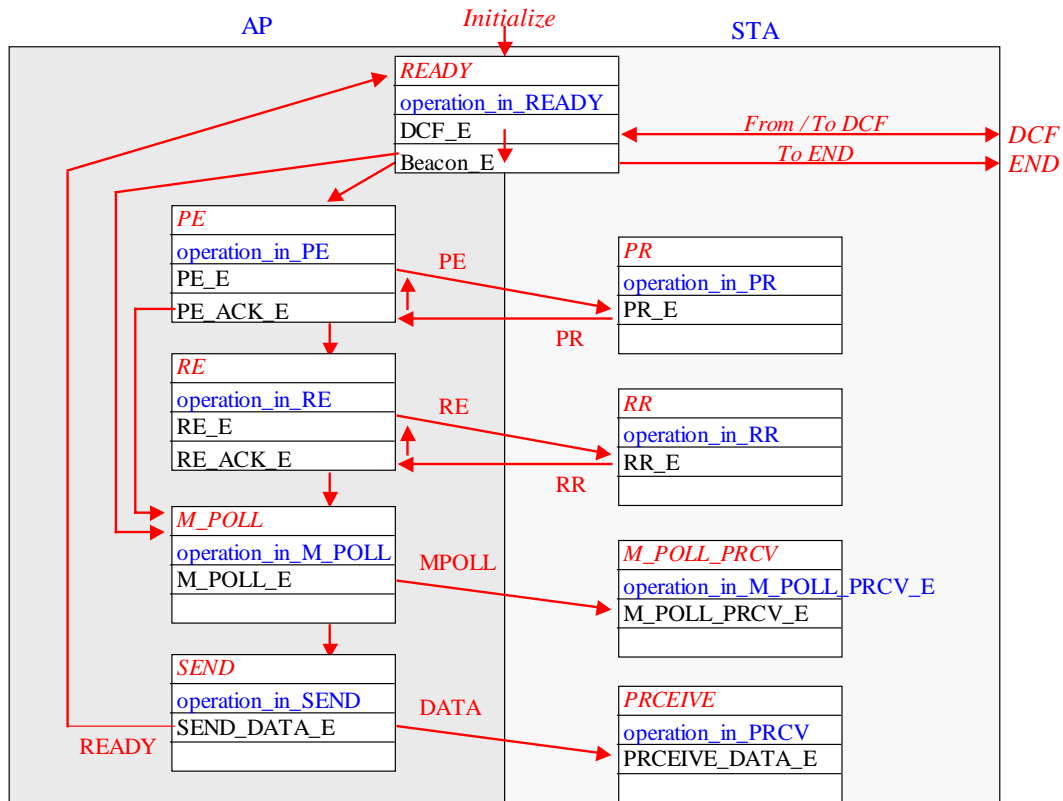


圖 4.6 Q-PCF 在 PCF 的有限狀態機

如圖 4.6 所示，與 IEEE 802.11 不同的是把加入輪詢名單的工作交由 PCF 來執行。

READY STATE :

DCF_E EVENT :

這個 process 最主要是讓已經結束的 CFP 進入 CP，再等待 CP 的結束再回到 CFP，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 DCF 所延遲 (T_s_time ; T_s) 的時間，再加上一個 PIFS 的時間。

如以下的程式碼所示，進入這個 process，代表 PCF 已經結束，現在要直接進入 DCF，等待 DCF 結束後，再回到 PCF。

```
Throughput_d=DCF(gclock,gcount*SF,&Ts_time);
event_insert(id,Beacon_E,gclock+Ts+PIFS,PRADY_S);
```

Beacon_E EVENT :

這個 process 最主要是讓 PC 發出 beacon，如以下的程式碼所示，並計算 d_g 、 d_r 和 RAB 後，再判斷 d_g 和 RAB 只要其中一個等於 0 或小於 0，代表頻寬已經被佔滿，所以要進入 M-POLL，否則要進入 PE。在時間上的計算上，要把目前的時間加上 beacon 的傳送時間，再加多一個 SIFS 的時間。

```
d_g = CFPmaxDuration-(RTS+CTS+maxMPDU+ACK+3*SIFS+OCFP+(3*k)+Gi_SUM);
d_r = CFPmaxDuration-(Ts+OCFP+(3*k)+Di_SUM);
RAB=d_r-(PE+PR+3*SIFS);
if ((d_g<=0)|| (RAB<=0))
    event_insert(id,M_POLL_E,gclock+Beacon+SIFS,M_POLL_S);
else
    event_insert(id,PE_E,gclock+Beacon+SIFS,PE_S);
```

PE STATE :

PE_E EVENT :

這個 process 最主要是讓 PC 送出優先權的詢問訊框，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 PE 的傳送時間，再加多一個 SIFS 的時間。

```
event_insert(id,PR_E,gclock+PE+SIFS,PR_S);
```

PE_ACK EVENT :

這個 process 最主要是讓 PC 接收優先權的回應訊框，如以下的程式碼所示，先判斷 d_g 和 RAB 只要其中一個等於 0 或小於 0，代表頻寬已經被佔滿，所以要進入 M-POLL，否則要依照 PR 所回應的 6 種狀況進入 PE 或 RE，所以在時間的計算上也會有所不同。

```
if ((d_g<=0)|| (RAB<=0))
    event_insert(id,M_POLL_E,gclock+M_POLL+SIFS,M_POLL_S);
    break;

if ((pri==2)&&(collision==0)){pri--;event_insert(id,PE_E,gclock,PE_S);
} else if ((pri==2)&&(collision==1)){pri--;event_insert(id,PE_E,gclock,PE_S);
} else if ((pri==2)&&(collision==2)){event_insert(id,RE_E,gclock,RE_S);
} else if ((pri==1)&&(collision==0)){event_insert(id,M_POLL_E,gclock,M_POLL_S);
} else if ((pri==1)&&(collision==1)){event_insert(id,M_POLL_E,gclock,M_POLL_S);
```



```

    }else if ((pri==1)&&(collision==2)){event_insert(id,RE_E,gclock,RE_S);
}

```

PR STATE :

PR_E EVENT :

這個 process 最主要是讓 STA 送出優先權的回應訊框，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 的時間和 PR 的時間。由於 STA 的回應狀況會影響到 Δg (d_g)、 Δr (d_r)和 RAB。

```

collision=2;
event_insert(id,PE_ACK_E,gclock+PR+SIFS,PE_S);
d_r = d_r-(PE+RE+2*SIFS);
collision=0;
event_insert(id,PE_ACK_E,gclock+Slot_Time,PE_S);
d_r = d_r-(PE+Slot_Time);
collision=1;
event_insert(id,PE_ACK_E,gclock+PR+SIFS,PE_S);
if (pri==2){
    d_g = d_g-(SIFS+CBRGB+3);
    d_r = d_r-(PE+PR+CBRGB+3*SIFS+3);
}else{
    d_g = d_g-(SIFS+VBRGB+3);
    d_r = d_r-(PE+PR+Dmin(node[AID].db,VBRGB)+3*SIFS+3);
}
RAB=d_r-(PE+PR+3*SIFS);

```

RE STATE :

RE_E EVENT :

這個 process 最主要是讓 PC 送出 Address Pattern 的詢問訊框，如以下的程式碼所示，如果已經結束 Address Pattern 的詢問，則要進入 M-POLL，否則要進入 RR，在時間上的計算上，要把目前的時間加上 RE 的傳送時間，再加多一個 SIFS 的時間。

```

if ( stack_empty() !=1 )
    event_insert(id,RR_E,gclock+RE+SIFS,RR_S);
else

```

```
event_insert(id,M_POLL_E,gclock,M_POLL_S);
```

RE_ACK EVENT :

這個 process 最主要是讓 PC 接收 Address Pattern 的回應訊框，如以下的程式碼所示，先判斷 d_g 和 RAB 只要一個等於 0 或小於 0，代表頻寬已經被佔滿，所以要進入 M-POLL，否則進入 RE，在時間方面不需要被計算。

```
if ((d_g<=0)|| (RAB<=0))
    event_insert(id,M_POLL_E,gclock,M_POLL_S);
else
    event_insert(id,RE_E,gclock,RE_S);
```

RR STATE :

RR EVENT :

這個 process 最主要是讓 STA 送出 Address Pattern 的回應訊框，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 的時間和 PR 的時間。由於 STA 的回應狀況會影響到 $\Delta g(d_g)$ 、 $\Delta r(d_r)$ 和 RAB。

```
collision=1;
if (pri==2){
    d_g = d_g-(SIFS+CBRGB+3);
    d_r = d_r-(RE+RR+CBRGB+3*SIFS+3);
}else{
    d_g = d_g-(SIFS+VBRGB+3);
    d_r = d_r-(RE+RR+Dmin(node[AID].db,VBRGB)+3*SIFS+3);
}
collision=0
d_r = d_r-(RE+Slot_Time);
collision=2;
d_r = d_r-(RE+RR+2*SIFS);
RAB=d_r-(RE+RR+3*SIFS);

if (STATUS==0){
    event_insert(id,RE_ACK_E,gclock+RR+Slot_Time,RE_S);
else
    event_insert(id,RE_ACK_E,gclock+RR+SIFS,RE_S);
```

```
}
```

M-POLL STATE :

M-POLL EVENT :

這個 process 最主要是讓傳送者送出 M-POLL，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 M_POLL 的傳送時間，及每增加一個 STA 要多 3us 及一個 SIFS 的時間。

```
event_insert(id,M_POLL_PRCV_E,gclock+M_POLL+(j*3)+SIFS,M_POLL_PRCV_S);
```

最重要的是在 TXOP 的計算之前一定要先進行 gamma 的計算，如以下的程式碼所示，這個 gamma 值就是所謂的剩餘頻寬。

```
gamma=CFPmaxDuration - ((gclock - gzero) + (M_POLL + j * 3) + (j * SIFS) + (SIFS + CF_End));
```

M-POLL_PRCV STATE :

M-POLL_PRCV EVENT :

這個 process 最主要是讓接收者處理來自 PC 所送出的 M-POLL，如以下的程式碼所示，因為接收者在接收 M-POLL 後不需要做任何的回應，所以在此是沒有任何的時間需要被計算。

```
event_insert(id,SEND_DATA_E,gclock,SEND_S);
```

SEND STATE :

SEND_DATA EVENT :

這個 process 最主要是讓傳送者送出資料，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上資料被允許傳送的時間 (TXOP) 及一個 SIFS 的時間。

```
event_insert(node[i].rpacket.dst,PRCV_DATA_E,gclock+node[ActiveSet[STA]].TXOP+SIFS,PRCEIVE_S);
```

在此，最重要是資料遺失率的計算，CBR 與 VBR 不同，CBR 是在一個 CFP 沒有送出才算資料遺失，VBR 則要兩個 CFP 沒有送出才算資料遺失。

PRCEIVE STATE :

PRCEIVE_DATA EVENT :

這個 process 最主要是讓接收者處理來自傳送者所送出的資料，如以下的程式碼所示，因為接收者在接收資料後不需要做任何的回應，所以在此是沒有任何的時間需要被計算。

```
event_insert(psrc,SEND_DATA_E,gclock,SEND_S);
```

State machines for Q-PCF MAC STA

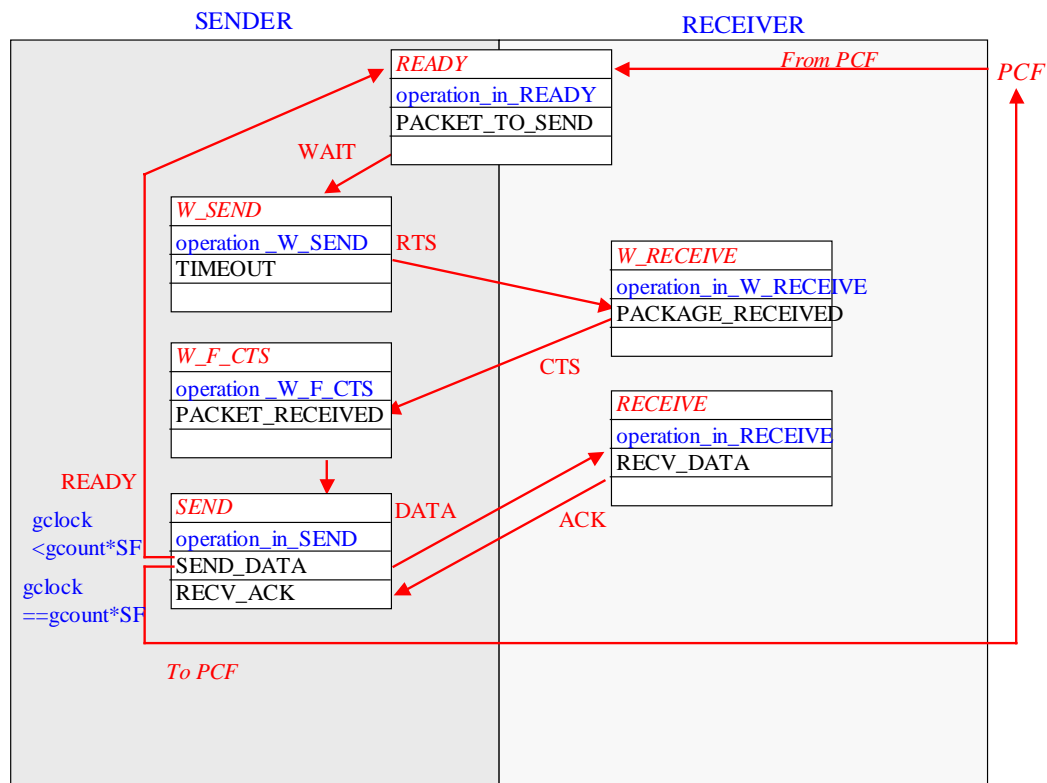


圖 4.7 Q-PCF 在 DCF 的有限狀態機

如圖 4.7 所示，大部分的內容都與先前針對 DCF 所敘述的內容相同，最主要不同的部分在於 PCF 有可能會提前結束而把控制權交給 DCF，如以下的程式碼所示，PCF 要傳遞二個參數，目前時間 (gclock) 和預定結束時間 (gcount*SF)，而 STA 則要回傳這次的資料流量 (Throughput_d) 和回傳延遲 (&Ts_time) 的時間

```
Throughput_d=DCF(gclock,gcount*SF,&Ts_time); // 回傳給 PCF
```

在進入 DCF 時先要處理在上一次未結束，並且會在這個 CFP 時間內發生的 event 全部刪除，在將這些被刪除的 event，再重新安插準備 backoff time 的 event。

```
while ((first_d->next != NULL)&&(first_d->next->clock < gclock_p)){
    nid=first_d->next->nodeid;
    event_delete(first_d->next->nodeid,first_d->next->event);
    event_insert_d(nid,PACKET_TO_SEND,gclock_p,READY);
}
```

RTS 只要在時間結束前 (SIMU_TIME_d) 送出，所以在傳送資料結束時，要知道是否已經超過原先預定的時間。

```
if (gclock_d > SIMU_TIME_d)
    delay_time=gclock_d-SIMU_TIME_d;
```

最後再把這次 CP 所成功送出的資料次數 (succ_d) 乘以資料傳送時間 (PACKET_time)，這個傳送時間要扣除 Header 的部分，再加上延遲的時間回傳給 PCF。

```
Throughput=succ_d*(PACKET_time-(PHYheader+MACheader));
*Ts_time=delay_time;
```

4.2.4. IEEE 802.11 有限狀態機

IEEE 802.11 的主要架構和 QPCF 相同，不同的是 IEEE 802.11 在加入輪詢名單是利用 DCF 來完成，而不是透過 PCF 來完成，所以第一個超級訊框是以 DCF 為主，等有 STA 加入輪詢名單後，接著在第二個超級訊框才開始使用 PCF 和 DCF 交錯的方式執行。

State machines for 802.11 MAC access point

為了簡化程式，在這個部分我們假設所有傳送 CBR 和 VBR 以及所有接收者都要加入輪詢名單，並且所有的接收者也剛好都會有資料要送出。

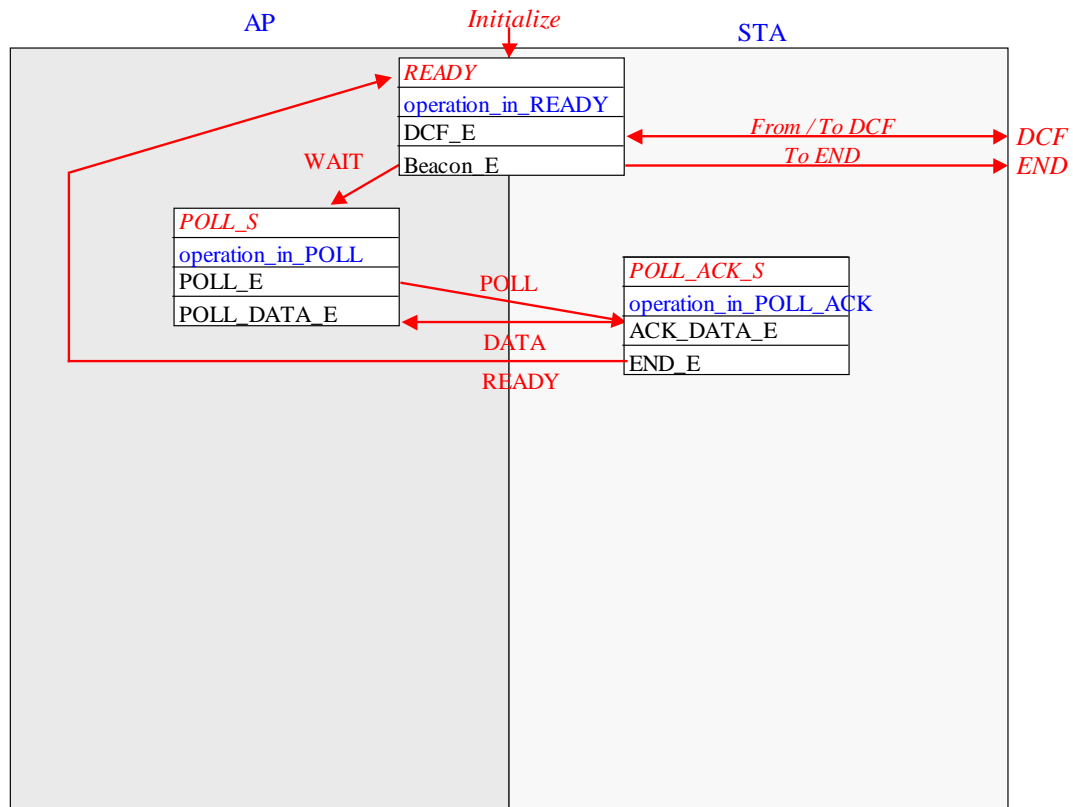


圖 4.8 IEEE 802.11 在 PCF 的有限狀態機

如圖 4.8 所示，與 Q-PCF 不同的是把加入輪詢名單的工作交由 DCF 來執行。

READY STATE :

DCF_E EVENT :

這個 process 最主要是讓 PCF 和 DCF 進行參數的交換為主。

Beacon_E EVENT :

這個 process 最主要是把 Beacon 送出。

POLL_S STATE :

POLL_E EVENT:

這個 process 最主要是讓 PC 送出 POLL (CF_POLL) 訊框，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 POLL 的時間及一個 SIFS 的時間。

```
event_insert(psrc,ACK_DATA_E,gclock+POLL+SIFS,POLL_ACK_S);
```

POLL_DATA_E EVENT:

這個 process 最主要是讓 PC 送出 POLL+Data+ACK，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 POLL+Data+ACK 及一個 SIFS 的時間 (POLL 和 ACK 已包含在 Header 內)。

```
event_insert(psrc,ACK_DATA_E,gclock+node[ActiveSet[STA]].db+SIFS,POLL_ACK_S
);
```

POLL_ACK_S STATE :**ACK_DATA_E EVENT :**

這個 process 最主要是讓 STA 送出 Data+ACK，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 Data+ACK 及一個 SIFS 的時間 (ACK 已包含在 Header 內)。

```
event_insert(psrc,ACK_DATA_E,gclock+node[ActiveSet[STA]].db+SIFS,POLL_ACK_S);
```

END_E EVENT:

這個 process 最主要是讓 PC 送出 CF_END，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 CF_END 的時間。

```
event_insert(pdst,DCF_E,gclock+CF_End,PRADY_S);
```

State machines for 802.11 MAC STA

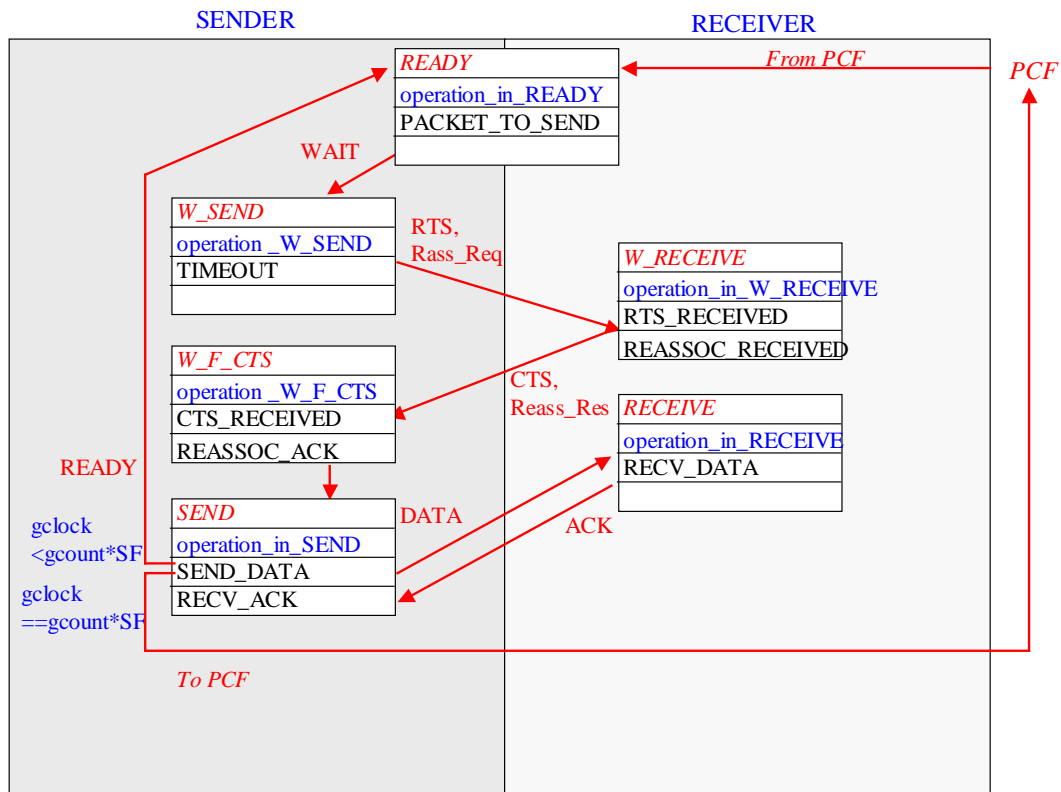


圖 4.9 IEEE 802.11 在 DCF 的有限狀態機

如圖 4.9 所示，大部分的内容都與 QPCF 針對 DCF 部分所敘述的相同，最主要不同的部分在接收 RTS 的部分還要處理重新聯結要求 (Reassociation Request) 的部分，而在接收 CTS 的部分還處理重新聯結回應 (Reassociation Response) 的部分。

W_SEND STATE :

TIMEOUT EVENT :

這個 process 的内容都與 QPCF 針對 DCF 在 RTS 部分是相同，所不同的是在於 STA 在要加入或退出輪詢名單的同時要發出 Reassociation Request 的訊框，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 Reassociation Request (REASSOC_REQ_Time) 的時間。

```
event_insert_d(id, REASSOC_RECEIVED, gclock_d + REASSOC_REQ_Time, W_RECEIVE);
```

W_RECEIVE STATE

RTS_RECEIVED EVENT:

這個 process 最主要是讓接收者在收到 RTS 之後再把 CTS 送出，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 的時間和 CTS 的時間 (CTS_time)。

```
event_insert_d(psrc,CTS_RECEIVED,gclock_d+SIFS+CTS_time,W_F_CTS);
```

REASSOC_RECEIVED EVENT:

這個 process 最主要是讓接收者在收到 Reassociation Request 之後再把 Reassociation Response 送出，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上一個 SIFS 的時間和 ACK 的時間 (ACK_time)。

```
event_insert_d(psrc,REASSOC_ACK,gclock_d+SIFS+ACK_time,W_F_CTS);
```

W_F_CTS STATE :

CTS_RECEIVED EVENT :

這個 process 最主要是讓傳送者準備把資料傳送出去，如以下的程式碼所示，在時間上的計算上，是沒有任何的時間需要被計算。

```
event_insert_d(id,SEND_DATA,gclock_d,SEND);
```

REASSOC_ACK EVENT :

這個 process 最主要是讓 Reassociation 不管是要加入或退出輪詢名單，在成功之後可以停止，此外要把 STA 的狀態變為已經加入或退出輪詢名單的狀態，如以下的程式碼所示。

```
if (node[id].REASSOC_IN==0){
    node[id].REASSOC_IN=1;
    node[id].enter=1;
}
if (node[id].REASSOC_OUT==0){
    node[id].REASSOC_OUT=1;
    node[id].enter=0;
}
```

第5章 模擬結果

首先我們要替資料流量 (Throughput, 又稱 Goodput) [3][20] 和資料遺失率 (又稱 Frame Delay Dropped Rate) 給予明確的定義。資料流量指的是在程式執行過程中, 資料成功被送出的時間除以全部的執行時間, 就是單位時間內網路傳輸的資料位元數 (bps), 包括 DCF 和 CBR 及 VBR 三種不同優先權的資料, 同時排除負載 (Overhead)、碰撞 (Collision) 和重傳 (Retransmission) 三種狀況 [12], 加上在轉送過程中所送出的資料。資料遺失率指的是在應該送出的資料却沒有送出。CBR 是以一個 CFP 沒有把資料送出就要算資料遺失, VBR 則是以兩個 CFP 沒有把資料送出才要算資料遺失。

5.1. DCF 的資料流量

目的：

測試 DCF 在透過 AP 轉送的資料流量

條件：

DCF： $\lambda = 10^2$ frame / sec (DCF)

STA：1,2,4,8,16,32,64,128

結果：

我們可以由圖 5.1 可以得知資料流量隨著 STA 數目的增加而下降迅速, 這是由於 STA 所送出的資料不但要透過 AP 的轉送, 而且 AP 在轉送資料的過程還要參與 DCF 的競爭。

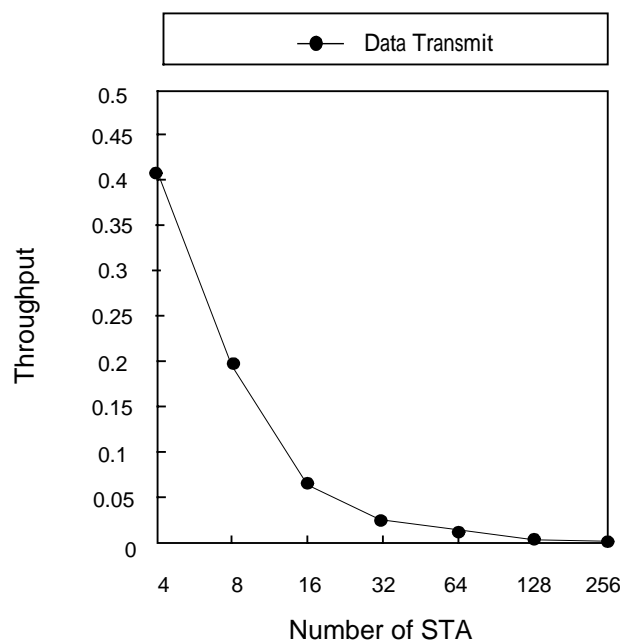


圖 5.1 DCF 的資料流量

5.2. Q-PCF

5.2.1. 允入控制能力

目的：

- (a)測試 CBR 和 VBR 分別在 Q-PCF 和 PCF 中加入輪詢名單的數目。
- (b)測試 CBR 和 VBR 分別在 Q-PCF 和 PCF 中資料遺失率的數目。

條件：

STA=256, $\varepsilon_{CBR}=0$, $\varepsilon_{VBR}=0.5$, $\lambda=0.1$ frame / sec (DCF) , 並分兩成種狀況, CBR=150,VBR=0 及 CBR=0,VBR=150。

結果：

我們可以由圖 5.2 可以得知：

(a)CBR 和 VBR 在 Q-PCF 中加入輪詢名單的數目, 如理論的 Upper Bound 相同, CBR 和 VBR 在 PCF 中加入輪詢名單的數目似乎不受到限制, 除非 DCF 的 STA 數目太多才會受到影響。

(b)在 Q-PCF 中的 CBR 由於 $\varepsilon_{CBR}=0$, 所以沒有資料遺失, VBR 則因為由於 $\varepsilon_{VBR}=0.5$, 所以也不會有資料遺失, 而 PCF 的 CBR 和 VBR 則不然, 資料遺失率隨著加入輪詢名單的數目變多迅速增加。

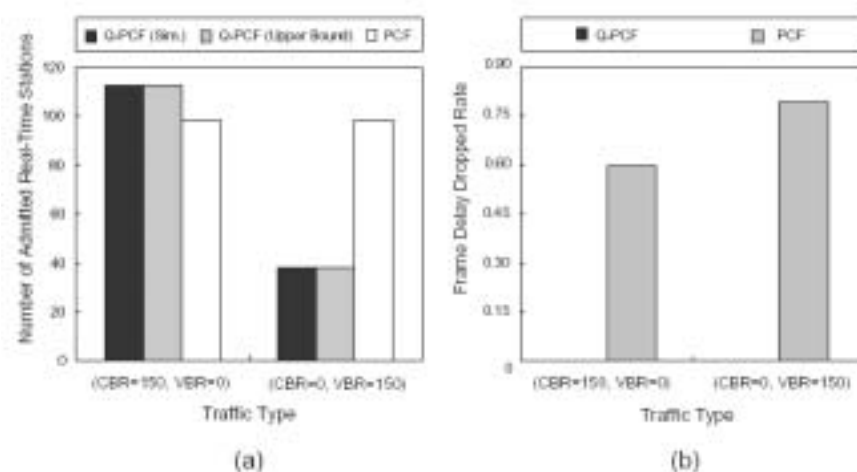


圖 5.2 允入控制的能力

5.2.2. DCF 的影響

目的：

- (a)DCF 的資料產生間隔由長變短, 藉此測試 CBR 和 VBR 在 PCF 中加入輪詢名單的數目。

(b)DCF 的資料產生間隔由長變短，藉此測試 CBR 和 VBR 在 Q-PCF 中加入輪詢名單的數目。

條件：

STA=256, $\varepsilon_{CBR}=0$, $\varepsilon_{VBR}=0.5$, $\lambda=1$ 或 10 frame / sec (DCF) 並分兩成種狀況，CBR=150,VBR=0 及 CBR=0,VBR=150。

結果：

我們可以由圖 5.3 可以得知：

(a)PCF 的 CBR 和 VBR 分別如預測的結果一樣，因為 DCF 的資料產生間隔由長變短，讓 CBR 和 VBR 加入輪詢名單的數目受到影響。

(b)QPCF 的 CBR 和 VBR 分別如預測的結果一樣，雖然 DCF 的資料產生間隔由長變短，讓 CBR 和 VBR 加入輪詢名單的數目不受任何影響。

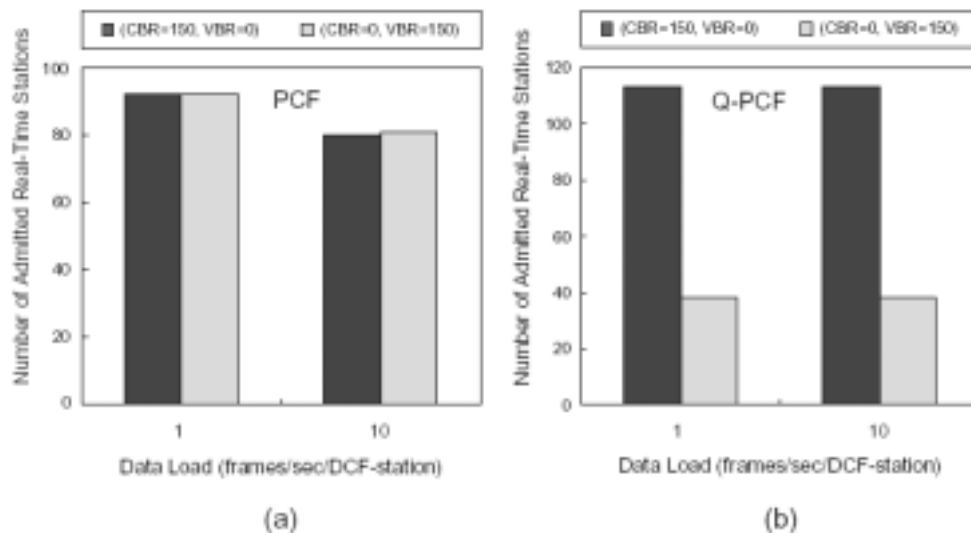


圖 5.3 DCF 的 λ 對 PCF 和 QPCF 的影響

5.2.3. CBR 之 ε 的影響

目的：

(a)測試 CBR 在 Q-PCF 中的資料流量和資料遺失率的關係

(b)相對的測試 CBR 可以加入輪詢名單的數目

條件：

STA=256, CBR=150, VBR=0, $\lambda=0.1$ frame / sec (DCF)

$\varepsilon_{CBR}=0, 0.02, 0.04, 0.08, 0.1, 0.12$

結果：

我們可以由圖 5.4 可以得知：

(a) ε 的值和資料遺失率成正比。基本上如果 ε 為 0.1，其資料遺失率會等於 10%。

(b) 當 ε 變大，加入輪詢名單的 CBR 數目會變多，事實上資料遺失率也相對的會變大。

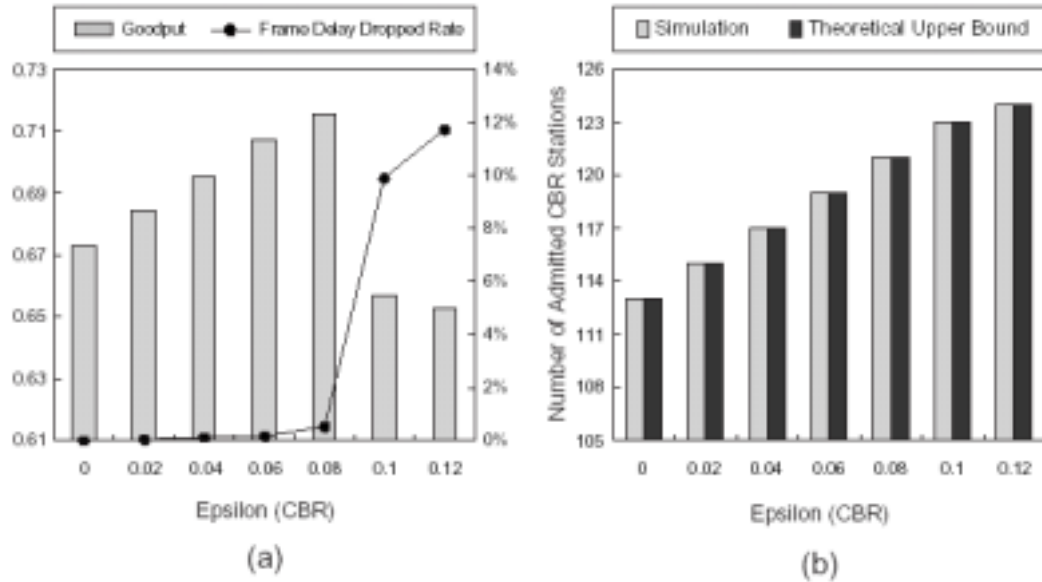


圖 5.4 CBR 的資料流量和資料遺失率的關係

5.2.4. VBR 之 ε 的影響

目的：

(a) 測試 VBR 在 Q-PCF 中的資料流量和資料遺失率的關係

(b) 相對的測試 VBR 可以加入輪詢名單的數目

條件：

STA=256, CBR=0, VBR=150, $\lambda=0.1$ frame / sec (DCF)

$\varepsilon_{VBR}=0.475, 0.5, 0.55, 0.6, 0.625, 0.65$

結果：

我們可以由圖 5.5 可以得知：

(a) ε 的值和資料遺失率成正比。基本上如果 ε 為 0.5 以上，其資料遺失率會才會逐開始產生。

(b) 當 ε 變大，加入輪詢名單的 VBR 數目會變多，事實上資料遺失率也相對的會變大。

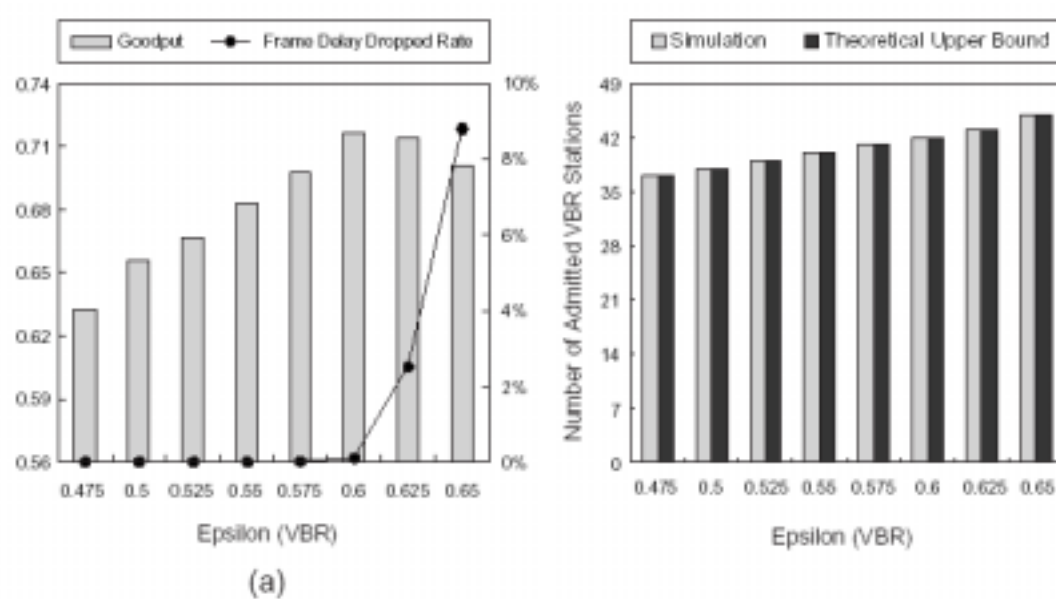


圖 5.5 VBR 的資料流量和資料遺失率的關係

5.2.5. 資料流量和資料遺失率的關係

目的：

PCF 和 Q-PCF 之間資料流量的關係

(a)測試分別在 Q-PCF 和 PCF 中的資料流量

(b)測試分別在 Q-PCF 和 PCF 中的資料遺失率

條件：

STA=256, ϵ CBR=0:02, ϵ VBR=0.5, CBR=150, λ =1 frame / sec (DCF)

結果：

我們可以由圖 5.6 可以得知：

(a)Q-PCF 的資料流量是 PCF 的資料流量的兩倍，最主要還是在於 Q-PCF 可以使用點對點的方式進行資料的傳輸。

(b)Q-PCF 中加入輪詢名單的數目似乎接近飽合，而且 CBR 中途因為有退出輪詢名單的機會，所以即使 ϵ =0.02，也沒有資料遺失的現象發生，VBR 則因為由於 ϵ VBR=0.5，雖然傳輸速度不固定，而且落在 120 Kbps 和 420 Kbps 的範圍之間，但是由於平均值是落在 240Kbps，所以也不會有資料遺失的現象發生，而 PCF 的 CBR 和 VBR 則不然，資料遺失率相當的大。

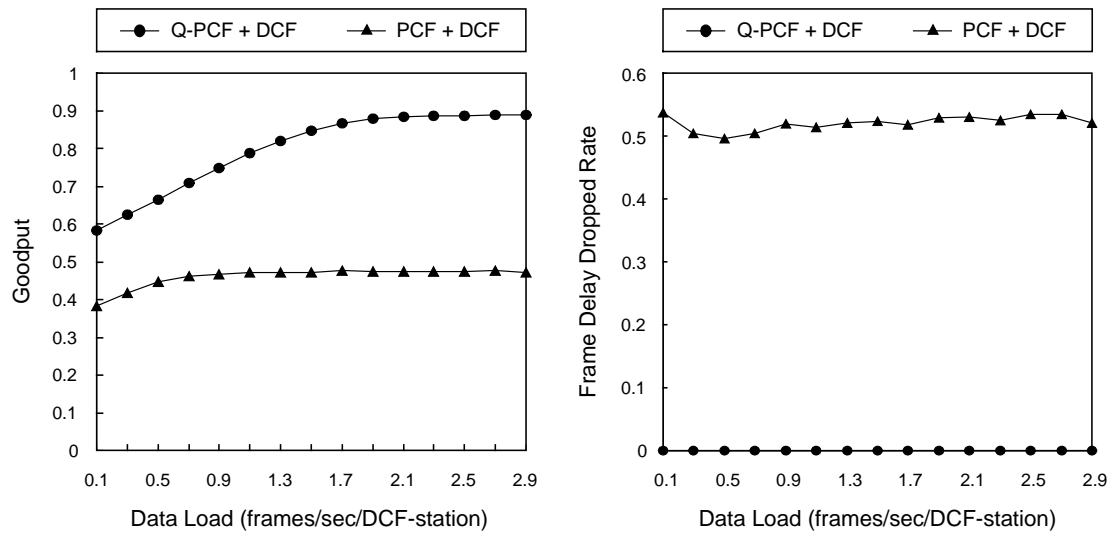


圖 5.6 效能評估

第6章 EQ-PCF

Q-PCF 對於頻寬控管方面確實是相當有效，但是由於我們先前做了一些假設，這些假設對於 IEEE 802.11 來說，似乎有些衝突，所以對於與 IEEE 802.11 的相容性而言，是個值得探討的問題，因此這個章節最主要的以相容性為出發點，並進行深入的探討，且將原來的機制進行修正，讓 Q-PCF 與 IEEE 802.11 之間更具有相容性，然而修改過後的機制命名為 Enhanced Q-PCF 簡稱 EQ-PCF。

6.1. 註冊階段

按照 Q-PCF 的理論來說，即使已經沒有 STA 要傳送 CBR 或 VBR 的資料，也就是已經沒有 STA 要加入輪詢名單，同時輪詢名單也還有剩餘的空間，此時在每個 Beacon 結束後，還是會進行優先權的詢問，如此則會造成 Overhead 的增加，尤其優先權等級如果很多，情況會更糟，像 IEEE 802.11e 草案就把分為優先權等級分為 8 個 [14]。

EQ-PCF 將整個 CFP 切分為三個部分中的第一個部分的 prioritization period 再分成連結 (Join) 和 Prioritization 兩個階段，如圖 6.1 所示，Join 階段和 prioritization 階段最大的不同在於 AP 會發出 JE (Join Enquiry) 訊框，這個訊框會詢問 STA 是否要加入輪詢名單，收到此訊框的 STA，可以不管自己優先權的等級，發出 JR (Join Response) 的回覆訊框，訊框格式如圖 6.2 所示。換句話而言，在 Join 階段，PC 會和想要加入輪詢名單的 STA 先進行一個 Handshake 後，才會決定是否要繼續執行 Prioritization period，所以 Prioritization period 未必會被執行，只要在 Join 的階段沒有 STA 回應，或是只有一台 STA 要求加入輪詢名單，便會跳過 Prioritization 和 Collision Resolution Period 階段，若此時輪詢名單先前已經有 STA 加入，便要執行 M-POLL，否則就會結束 CFP。此外在 Prioritization Period 的部分則維持不變，PC 會執行一連串的 Handshake 用來確保優先權高的 STA 能比優先權低的 STA 先加入輪詢名單。

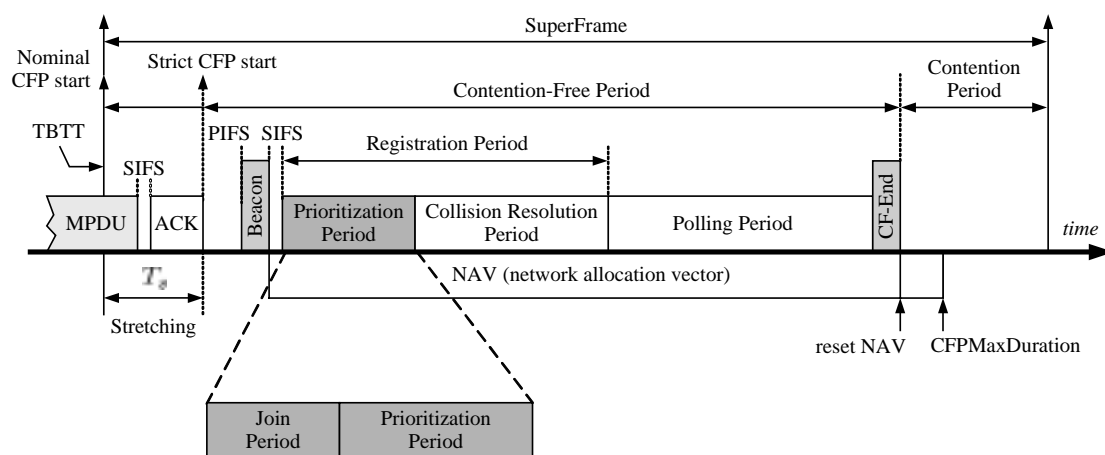


圖 6.1 超級訊框的結構

增加了 JE 訊框詢問的理由很簡單，因為 STA 只要想加入輪詢名單而且頻寬沒有剩餘，通常會很快的加入輪詢名單，而 STA 不要想加入輪詢名單的機會更大，所以 CFP 大部分的時間都是處於這種狀態，以 CBR 和 VBR 只有兩個優先權的情況來說，似乎不是太明顯，但是一但優先權很多，差異就會很大。

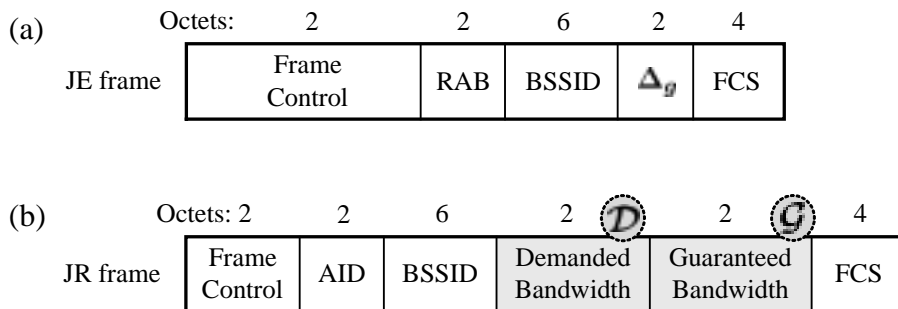


圖 6.2 訊框格式

6.2. 資料傳送

Q-PCF 假設所有的 STA 彼此之間要相鄰，所謂相鄰，不只是射頻 (Radio Frequency, 簡稱 RF) 所涵蓋的範圍有重複的現象，還包括兩台 STA 的 Receiver 都可以接收到對方 Transmitter 所發出的訊號，如此在理論上資料便可以進行點對點 (STA 對 STA) 的直接傳送，完全不需要經過 AP 轉送，不過根據 IEEE 802.11 的標準，STA 即使不相鄰也可以透過 AP 傳送資料，而且 STA 和 STA 是不可以直接進行資料的傳送。

在此，首先要討論的是 STA 即使不相鄰也可以透過 AP 傳送資料，如此的機制存在的理由為何，其實在於 STA 和 STA 之間即使不相鄰也可以進行資料的傳送的好處，可以增加 STA 的傳輸範圍，其實我們可以透過兩個具體的數據來知道其差異為何。

第一，以 RF 的涵蓋範圍而言：

以 IEEE 802.11 的規範而言，AP 和 STA RF 所發射的範圍相同，另外如果在真實的環境要做到 STA 之間一定要相鄰才可以向 AP 註冊，在真實的環境要做到只能把 AP 的 RF 發射強度降低，讓 AP 的涵蓋範圍縮小到 STA 的一半或一半以下才可以做到，如圖 6.3 所示，所以 IEEE 802.11 的 RF 所涵蓋的面積是 Q-PCF 的四倍。

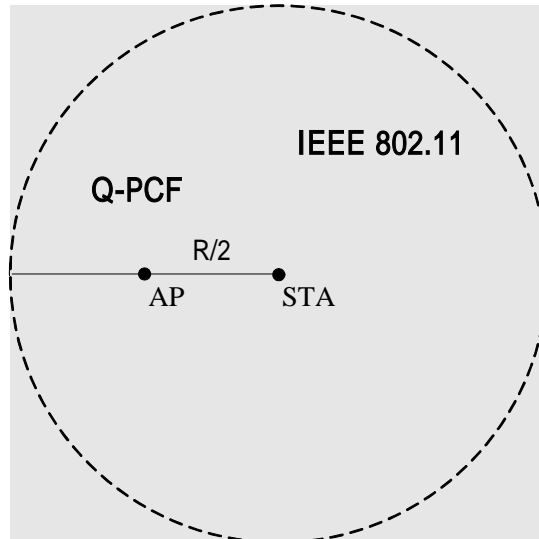


圖 6.3 RF 的涵蓋範圍

第二，以直接進行通訊的機率而言：

透過演算法來得到一個比較具體的結論，我們先給定 AP 和 STA 的 RF 發射範圍，並且假設其大小相同，在以隨機的方式分別產生 STA1 和 STA2 的坐標軸，而這個坐標軸一定要落在 AP 的 RF 所涵蓋的範圍內，此時在把再計算這兩個點的距離是否落在彼此的涵蓋範圍，如圖 6.4 所示，並由數據來判斷是否可以互相涵蓋對方，如此重複 100 萬次，結果發現如果在一個 AP 的範圍下 STA 和 STA 可以直接通訊的比例是 58%，而不可以直接通訊的比例是 42%，換句話來說，如果有 AP 的轉送可以增加 42% 的資料流量。

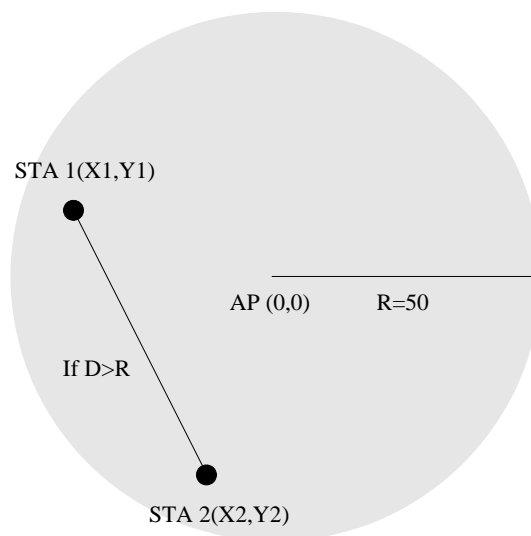


圖 6.4 距離的量測

其次我們要探討為什麼 STA 和 STA 是可以直接進行資料的傳送的優點為

何，其實很簡單，資料流量會是完全要透過 AP 轉送的一倍。

所以我們的目標在於如果相鄰的 STA 可以直接進行資料的傳輸，否則便要透過 AP 的轉送，如圖 6.5 所示。

接著我們要探討為什麼 STA 和 STA 是不可以直接進行資料的傳送，其實問題在於是否可以克服省電的問題，因此傳送者根本不知道接收者目前是否已經進入省電模式的狀態，這個部分我們會在下一個章節介紹。

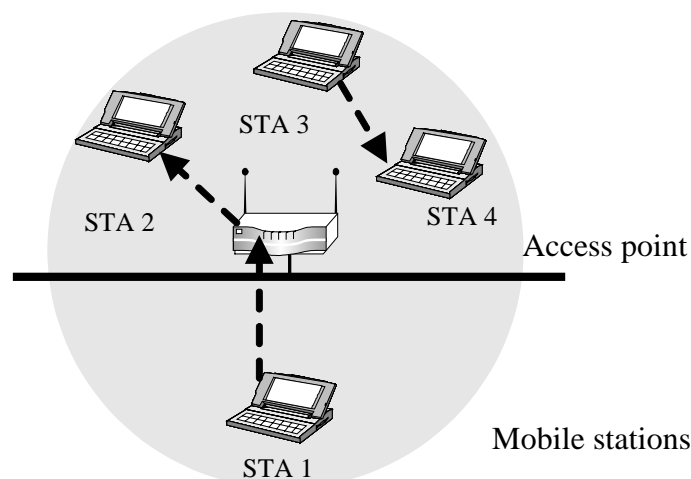


圖 6.5 資料傳輸路徑

6.3. 省電管理

不管 STA 和 STA 是否能直傳送，或者是透過 AP 轉送，其接收者都不可以進入省電模式，但是對於整個 Q-PCF 的機制，也是要在沒有省電的機制下完成，但是所有 STA 都不知道自己是否會成為接收者，所以都不可以進入省電模式，但是省電的機制確是 IEEE 802.11 扮演著相當重要的角色，尤其是再沒有電源的地方。

以目前 IEEE 802.11 的機制，如果 AP 要傳送資料給 STA，由於 AP 不知道目前 STA 是否是處於省電模式，所以都先由發 Beacon 來進行通知，STA 再接受通知後會發出 PS-Poll 代表可以接收 AP 傳送資料，

針對 DCF 部分，由於 STA 沒有即時的資料要進行傳送，所以在不確定對方是否處於省電模式的狀況，先把資料存在緩衝器也是一個不錯的選擇，所以在此只針對 PCF 進行討論。

首先我們要在 JR 和 PR 和 RR 的訊框加入 6 個 Byte 的目的位址 (Destination

Address) 如圖 6.6 所示。

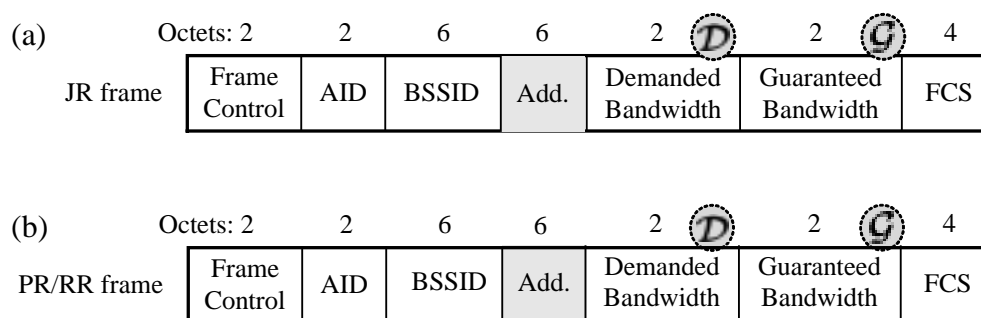


圖 6.6 訊框格式

其次在通訊的過程中，M-POLL 除了要告知 STA 的順序和傳輸時間，同時也要告知接收者是誰 (Destination Address)，如圖 6.7 所示，將原本 2 個 byte 變為 3 個 byte，前面 12 個 bit 指的是傳送者，後面 12 個 bit 指的接收者，當所有 STA 接收到這個訊框，便會進行判斷，準備接收資料的 STA 一律不可以進入省電模式，沒有要接收資料的 STA 便可以進入省電模式。

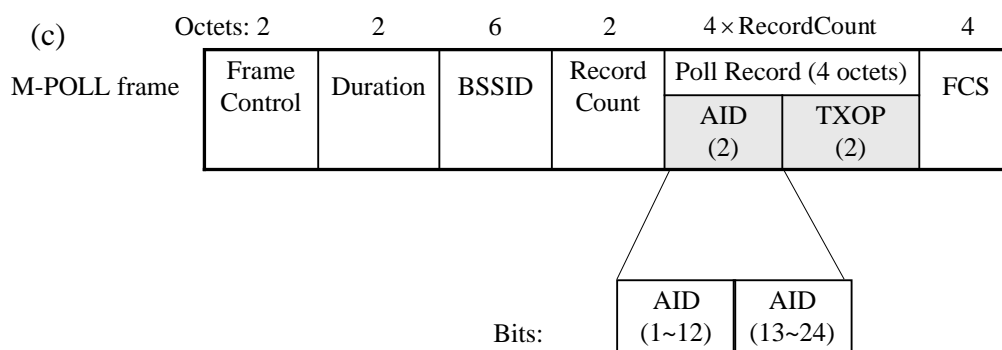


圖 6.7 M-POLL 訊框格式

更進一步來看，我們剛才假設所有的 STA 在每個 Beacon 都要醒來，這似乎會造成相當大的耗電量，所以基於 STA 可以更省電，STA 不一定要在每個 Beacon 醒來，而且在那一個 Beacon 醒來的時間也都不同，所以此時 PC 需要一個清單來記錄 STA 目前省電的模式，以便 STA 在要在加入輪詢名單時進行判斷，如果此時該接收者是進入省電模式，便拒絕，如果此時該接收者是沒有進入省電模式，便接受，如此 STA 可能會在加入輪詢名單的時間會稍微拉長一點，最長也只有處於省電模式之 STA 應該醒來接收 Beacon 訊框的時間間隔 (Listen Interval) 而已，最重要的是可以保證沒有加入輪詢名單的 STA 仍保有省電的機制。

6.4. DCF 存在的必要性

由於加入輪詢名單的工作已經由 DCF 的轉為 PCF，所以似乎 DCF 已經沒有

存在的價值，但是事實上目前還有頻道的選擇和認證 (Authentication) 的部分是由 DCF 來執行，這些是否也可以考慮改由 PCF 的階段來進行，而把原來 DCF 的部分捨棄不用，這是個值得探討的問題，在技術上是不會有問題，但是我們必須要考慮兩個現實的問題，第一，由於不屬於同一個 BSS 的 STA 也可以進行頻道的選擇和認證，這會使得一旦在這個部分負載太大，會讓碰撞的機會變大，讓原本想要加入輪詢名單的 STA 變得比較的困難。第二，由於有些資料是偶爾才送，譬如像郵件，可能不是那麼的急時，但是資料還是需要送出去，所以 DCF 還有存在的必要，除非一開始的環境就是只有多媒體的資料，是可以考慮。

第7章 效能評估

7.1. EQ-PCF 有限狀態機

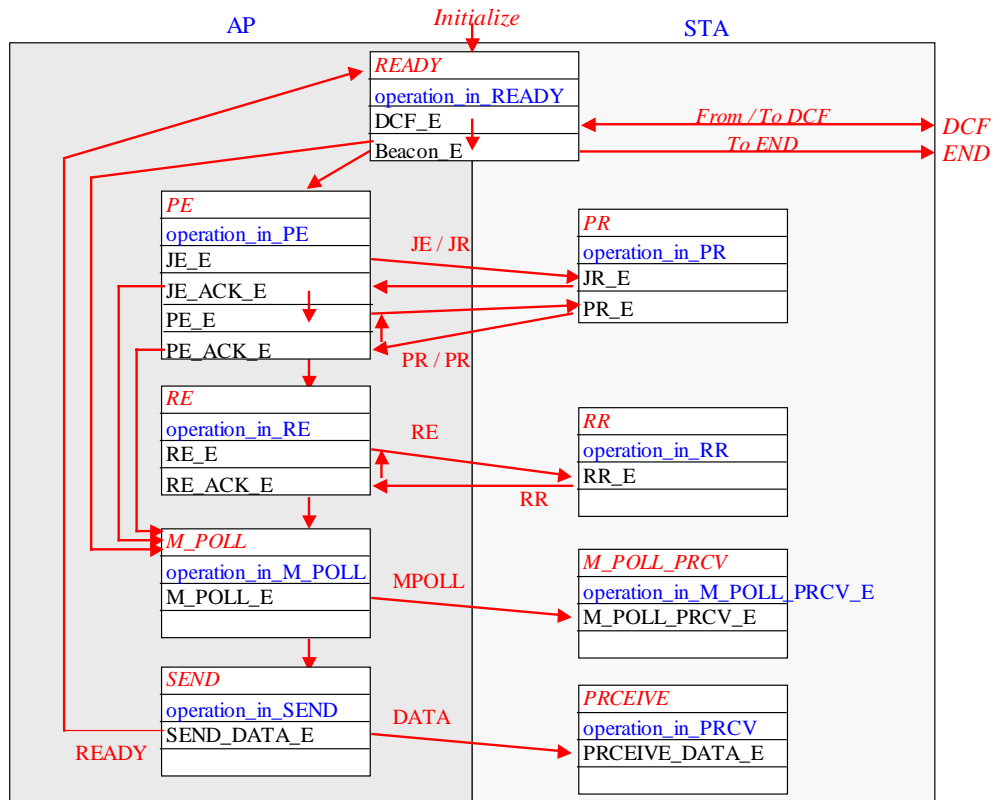


圖 7.1 EQ-PCF 有限狀態機

如圖 7.1 所示，在此大部分的內容都與 QP_PCF 部分所敘述的相同，最主要不同的部分在 PE 和 RE 的部分。

PE STATE :

JE EVENT :

這個 process 最主要是詢問是否有 STA 要加入輪詢名單，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 JE 及一個 SIFS 的時間。

```
event_insert(id,JR_E,gclock+JE+SIFS,PR_S);
```

JE_ACK EVENT :

這個 process 最主要是接收來自 STA 的回應，如以下的程式碼所示，在時間上的計算上，要依狀況而定。

```

if ((d_g<=0)|| (RAB<=0))
    event_insert(id,M_POLL_E,gclock+M_POLL+SIFS,M_POLL_S);

if (collision==0){
    event_insert(id,M_POLL_E,gclock,M_POLL_S);
} else if (collision==1){
    event_insert(id,M_POLL_E,gclock,M_POLL_S);
} else if (collision==2){
    event_insert(id,PE_E,gclock,PE_S);
}

```

PR STATE :

JR EVENT :

這個 process 最主要是回應是否有 STA 要加入輪詢名單，如以下的程式碼所示，在時間上的計算上，要把目前的時間加上 JR 及一個 SIFS 的時間。

```

collision=2;
event_insert(id,JE_ACK_E,gclock+JR+SIFS,PE_S);
d_r = d_r-(JE+JR+2*SIFS);

collision=0;
event_insert(id,JE_ACK_E,gclock+Slot_Time,PE_S);
d_r = d_r-(JE+Slot_Time);

collision=1;
event_insert(id,JE_ACK_E,gclock+JR+SIFS,PE_S);
if (pri==2){
    d_g = d_g-(SIFS+CBRGB+4);
    d_r = d_r-(JE+JR+CBRGB+3*SIFS+4);
} else {
    d_g = d_g-(SIFS+VBRGB+4);
    d_r = d_r-(JE+JR+Dmin(node[AID].db,VBRGB)+3*SIFS+4);
}

```

7.2. 模擬結果

7.2.1. 註冊時間

目的：

分別測試 EQ-PCF 和 Q-PCF 與 PCF 的平均註冊時間。

條件：

STA=256, ε CBR=0, ε VBR=0, CBR=100, VBR=0, $\lambda=1$ frame / sec
(DCF)

結果：

當輪詢名單已經額滿，就是頻寬已經沒有剩餘，如果 PC 經常處於這種狀態，Q-PCF 和 EQ-PCF 的註冊時間會差不多，因為 PC 在頻寬已經沒有剩餘，便不會進行詢問，但是只要頻寬還有剩餘，可是事實上已經沒有 STA 要加入輪詢名單，此時的 PC 還是會依照優先權的高低進行詢問，等於只要系統分為幾個優先權等級，就會詢問幾次，此時的 EQ-PCF 會比 Q-PCF 來的要好。如這個測試，只設定 CBR 的數目，並沒有設定 VBR 的數目，但是 PC 還是會進行 VBR 的詢問。

我們可以由圖 7.2 可以得知，在連續 180 秒的測試中，EQ-PCF 比 Q-PCF 多出 100us 的註冊時間，這是可以接受的結果，但是如果優先權等級超過 3 個或以上，其結果將會不同。此外這個結果也顯示在 128 台 STA 時 PCF 比 EQ-PCF 多了 8500 us 的註冊時間，而在 256 台 STA 時 PCF 比 EQ-PCF 多了 12200 us 註冊時間，由此可以得知 PCF 的部分的確是相當的不理想，尤其是在 256 台的 STA 的狀況下。

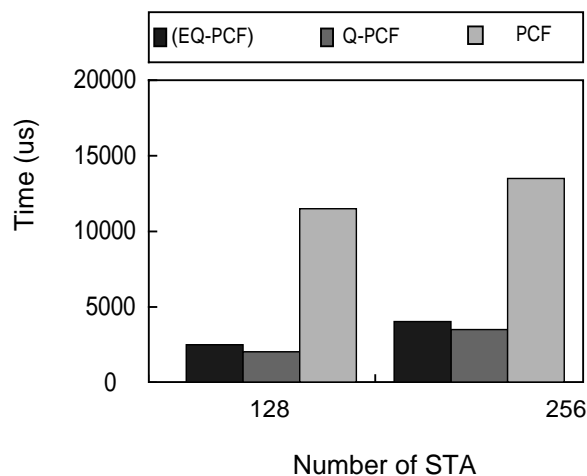


圖 7.2 平均註冊時間

7.2.2. 資料流量

目的：

分別測試 EQ-PCF 和 Q-PCF 與 PCF 的資料流量

條件：

STA=256, ε CBR=0:02, ε VBR=0.5, CBR=100, VBR=30

結果：

我們可以由圖 7.3 可以得知，Q-PCF 的資料流量是 PCF 的兩倍，最主要是由於 Q-PCF 在資料的傳輸上是以點對點的直接對傳，而 PCF 則是透過 AP 轉送的緣故，而 EQ-PCF 的資料流量比 Q-PCF 的要小，但是它的優點在於可以和 IEEE 802.11 相容，而且資料流量比 PCF 要大很多。

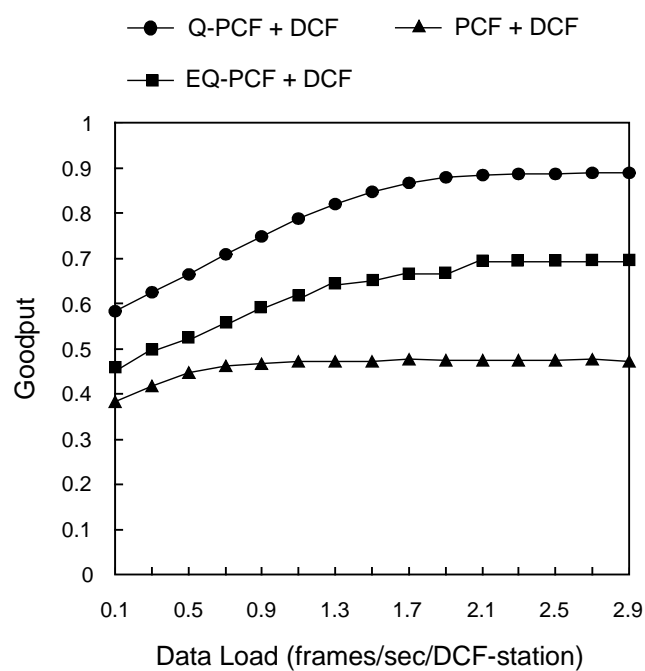


圖 7.3 效能評估

第8章 結論與未來努力的方向

在這篇論文所提出的方法中我們根據動態的需求來要求 Q-PCF 給予傳送時間與傳送頻寬的保證，結果已經讓無線網路可以傳送即時的多媒體資料，另外我們設計讓彼此相鄰的 STA 可以進行直接的傳輸，也讓資料流量能大幅度的提升，從我們的模擬結果得知，其資料流量比起傳統的網路提升的相當的多，而且最重要的是已經讓無線網路可以傳送即時的多媒體資料，除此之外，在 EQ-PCF 中更強調與 IEEE 802.11 的相容性，我想這些是我們最大的貢獻。

由模擬結果得知我們已經達到了我們當初所預設的目標，不過我想有些部分還是有討論的空間，如下列幾點：一、當 DCF 負載太高，是否可以設計讓部分的 STA 自動轉入 PCF，而不是一定只允許傳送多媒體的資料可以加入 PCF，因為即使不是要傳送即時的多媒體資料，還是不能太差，如此可以避免訊框的碰撞。二、在相同時間可以讓優先權較高的 STA 先行加入，但是如果有優先權較低的 STA 已經先行佔滿了所有 CFP 的時間，此時即使優先權較高的 STA 想要再加入，也無法加入，我想此時是否要讓優先權較低的 STA 退出，可能要考慮到其應用是否有其必要性，再行決定。三、雖然目前這是個以應用程式為主的應用，事實上也可以把這個概念應用在以人為主的應用，如在一個有無線上網的咖啡廳的收費。四、在 QoS 的要求下其省電模式相對會比較不理想，所以針對兩者都能兼顧的設計，還是有討論的空間。

參考文獻

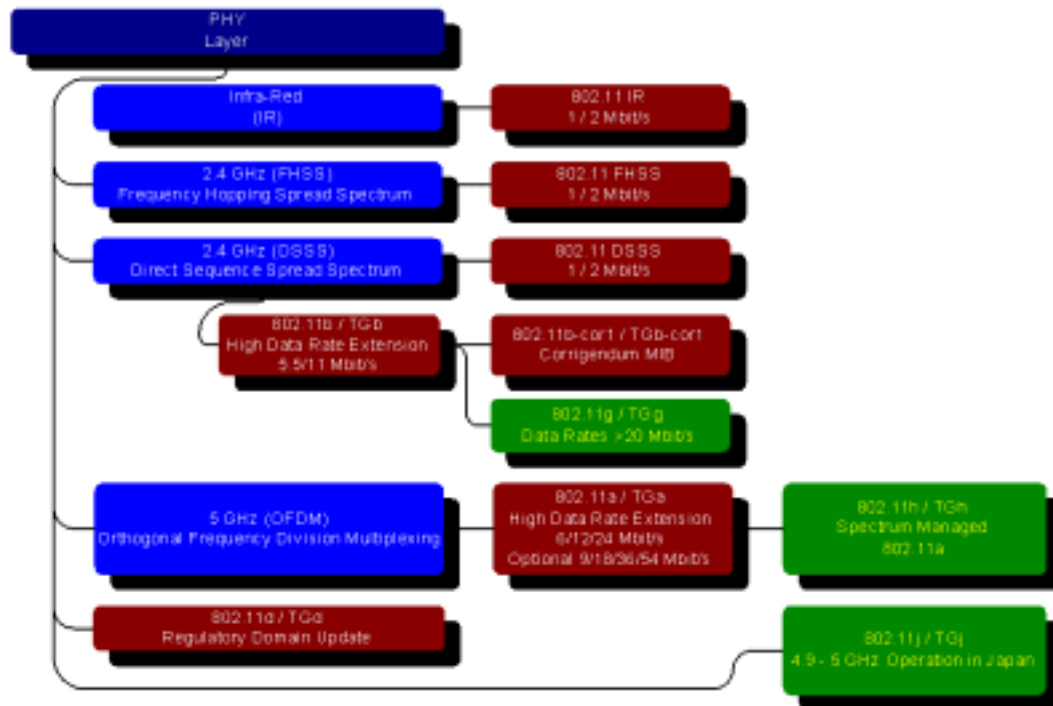
- [1] I. F. Akyildiz, D. A. Levine, and I. Joe. A Slotted CDMA Protocol with BER Scheduling for Wireless Multimedia Networks. *IEEE Transactions on Networking*, Vol. 7, No. 2, pp. 146–158, April 1999.
- [2] D. Bertsekas and R. Gallager. *Data Networks*, Second Edition, Prentice-Hall, 1992.
- [3] Giuseppe Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 3, pp. 535–547, March 2000.
- [4] Sunghyun Choi. Emerging IEEE 802.11e WLAN for Quality-of-Service (QoS) Provisioning. *SK Telecom Telecommunications Review*, Vol. 12, No. 6, pp. 894–906, December 2002.
- [5] S. Choi and K. G. Shin. A Unified Wireless LAN Architecture for Real-Time and Non-Real-Time Communication Services. *IEEE/ACM Transactions on Networking*, Vol. 8, No. 1, pp. 44–59, February 2000.
- [6] Zi-Tsan Chou, Ching-Chi Hsu, and Ferng-Ching Lin : A New Quality-of-Service Point Coordination Function for IEEE 802.11 Wireless Multimedia LANs
- [7] C. Coutras, S. Gupta, and N. B. Shro®. Scheduling of Real-Time Traffic in IEEE 802.11 Wireless LANs. *ACM/Baltzer Wireless Networks*, Vol. 6, pp. 457–466, 2000.
- [8] D.-J. Deng and R.-S. Chang. A Priority Scheme for IEEE 802.11 DCF Access Method. *IEICE Transactions on Communications*, pp. 96–102, January 1999.
- [9] M. Fischer. QoS Baseline Proposal. *IEEE Document*, IEEE 802.11-00/360.
[Online] <http://grouper.ieee.org/groups/802/11/Documents/DT351-400.html>
- [10] A. Ganz, A. Phonphoem, and Z. Ganz. Robust SuperPoll with Chaining Protocol for IEEE 802.11 Wireless LANs in Support of Multimedia Applications. *ACM/Kluwer Wireless Networks*, Vol. 7, pp. 65–73, 2001.

- [11] J. Y. Juang and B. W. Wah. Unified Window Protocols for Contention Resolution in Local Multiaccess Networks. IEEE INFOCOM, pp. 97–104, April 1984.
- [12] S. Keshav. An Engineering Approach to Computer Networking: ATM Networks, the Internet, and the Telephone Network. Addison-Wesley Publishing Company, 1997.
- [13] LAN MAN Standards Committee of the IEEE Computer Society. IEEE Standard 802.11-1999 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE, November 1999.
- [14] LAN MAN Standards Committee of the IEEE Computer Society. IEEE 802.11e/D3.2, Draft Supplement to Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS). IEEE, August 2002.
- [15] K. Nakano and S. Olariu. Randomized Initialization Protocols for Ad Hoc Networks. IEEE Transactions on Parallel and Distributed Systems, Vol. 11, No. 7, pp. 749–759, July 2000.
- [16] S. Ross. A First Course in Probability, Fifth Edition. Prentice-Hall, Inc., 1998.
- [17] O. Sharon and E. Altman. An Efficient Polling MAC for Wireless LANs. IEEE/ACM Transactions on Networking, Vol. 9, No. 4, pp. 439–451, August 2001.
- [18] J.-P. Sheu, C.-H. Liu, S.-L. Wu, and Y.-C. Tseng. A Priority MAC Protocol to Support Real-Time Multimedia Traffic in Ad Hoc Networks. ACM/Kluwer Wireless Networks, to appear, 2003.
- [19] S.-T. Sheu and T.-F. Sheu. A Bandwidth Allocation/Sharing/Extension Protocol for Multimedia Over IEEE 802.11 Ad Hoc Wireless LANs. IEEE Journal on Selected Areas in Communications, Vol. 19, No. 10, pp. 2065–2080, October 2001.
- [20] Y. C. Tay and K. C. Chua. A Capacity Analysis for the IEEE 802.11 MAC Protocol. ACM/Kluwer Wireless Networks, Vol. 7, pp. 159–171, 2001.

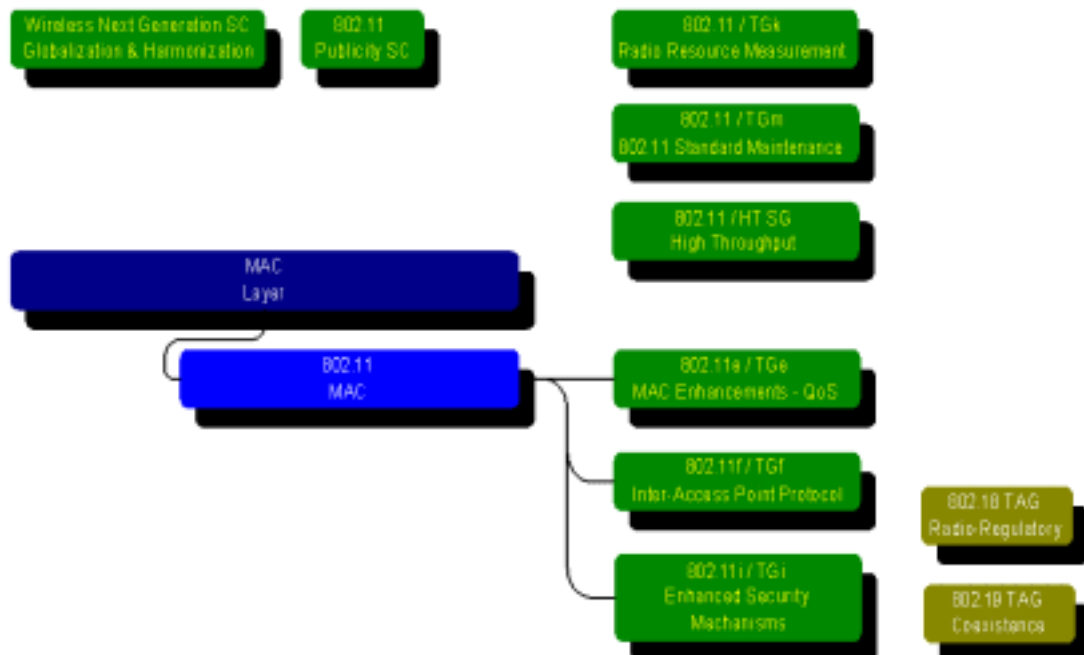
附錄 A

以下是 IEEE 802.11 Family 的在 PHY 和 MAC 的圖和 Status，資料來源 <http://www.ieee802.org/11/>。

802.11 Activities - PHY:



802.11 Activities - MAC & Others:



Status

- 802.11a : The scope of the project is to develop a PHY to operate in the newly allocated UNII band.
- 802.11b : Work has been completed and is now part of the Standard as an amendment - Published as IEEE Std. 802.11b-1999
- 802.11b-Cor1 : Work has been completed and is now part of the Standard as an amendment - Published as IEEE Std. 802.11b-cor1 2001
- 802.11c : Work has been completed and is now part of the ISO/IEC 10038 (IEEE 802.1D) Standard
- 802.11d : Work has been completed and is now part of the Standard as an amendment - Published as IEEE Std. 802.11d 2001
- 802.11e : Ongoing - Note: the Security portion of the TGe PAR was moved to the TGi PAR as of May 2001. TGe has completed letter ballot 51 with a 83% approval rate and is now in comment resolution.
- 802.11f : Work has been completed and is now part of the Standard as a recommended practice.
- 802.11g : Work has been completed and is now part of the Standard as an amendment.
- 802.11h : TGh has completed the 2nd Sponsor Recirculation Ballot with a 98% approval rating and is now in the comment resolution phase.
- 802.11i : Ongoing - Note: the Security portion of the TGe PAR was moved to the TGi PAR as of May 2001. TGi has completed WG Recirculation Letter Ballot 57 with a 78% approval rating and is now in the comment resolution phase.
- 802.11j : Ongoing - Initial meeting January 2003 TGj has completed Letter Ballot 56 with an approval rating of 79% and is now in the comment resolution phase.
- 802.11k : Ongoing - Initial meeting January 2003 and is preparing its first draft.
- 802.11m : Initial meeting March 2003 (Subject to SEC approval of the Task Group PAR)