

國立交通大學

網路工程研究所

碩士論文

在 NCTUns 網路模擬器上支援 IEEE 802.11 無線感知
網路並進行效能評估



Supporting and Evaluating Performances of IEEE 802.11 Cognitive
Radio Networks over the NCTUns Network Simulator

研究生：黃昱銘

指導教授：王協源 教授

中華民國九十九年七月

在 NCTUns 網路模擬器上支援 IEEE 802.11 無線感知網路並進行
效能評估

Supporting and Evaluating Performances of IEEE 802.11 Cognitive Radio
Networks over the NCTUns Network Simulator

研究生：黃昱銘

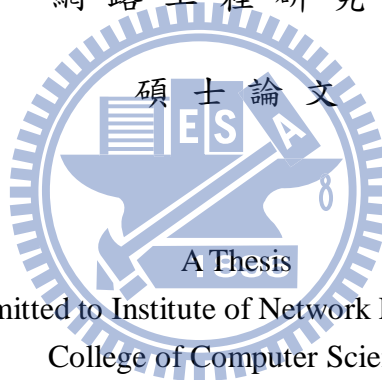
Student：Yu-Ming Huang

指導教授：王協源

Advisor：Shie-Yuan Wang

國立交通大學

網路工程研究所



Submitted to Institute of Network Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

July 2010

Hsinchu, Taiwan, Republic of China

中華民國九十九年七月

在 NCTUns 網路模擬器上支援 IEEE 802.11 無線感知網路並進行效能評估

研究生：黃昱銘

指導教授：王協源 教授

國立交通大學 網路工程研究所 碩士班

摘要

隨著無線網路愈來愈深入的研究與探討，目前可為人所用的無線網路傳輸之協定與其所使用之頻譜也隨之增加。根據現有的頻段用途與其分配方式，部份頻段需要經過申請執照的程序後才可使用。但在目前現有的研究就指出，並非所有頻段皆被充份的使用，這種情況在需要頻段使用執照的頻段更為明顯。因此，在近年來開始發展出一套稱作無線感知網路的技術，透過這種技術，無線感知網路的使用者可以在不干擾其他頻段的主要使用者來進行該頻段的使用；而當該位主要使用者欲使用此頻段時，無線感知網路的使用者便會主動退讓，使主要使用者在頻段的使用上不受干擾。

然而，在現階段對於無線感知網路的研究中所擁有的測試工具並不多，而且對於更高層的網路通訊協定之應用程式的支援也相當的有限，使得研究人員在驗證上難以利用更為複雜的情境來進行效能的評比。因此，本篇論文主要將透過 NCTUns 這套網路模擬器來進行平台的開發，並從目前既有的無線感測網路的相關研究進行實作，以提供給研究人員一套完整的無線感測網路之模擬環境，藉此能夠在沒有實體設備的狀況下進行通訊協定的開發。此外，本文也將基於該無線感測網路之通訊協定，從隱匿終端機問題的角度進行分析，並且對此通訊協定進行改善，期望能藉此使其效能提升。

關鍵字：無線感知網路、終端機隱匿問題、網路模擬器

Supporting and Evaluating Performances of IEEE 802.11 Cognitive Radio

Networks over the NCTUns Network Simulator

Student: Yu-Ming Huang

Advisor: Dr. Shie-Yuan Wang

Institute of Network Engineering
National Chiao Tung University

Abstract

According to a report about the utilization of spectrum, not all the range of spectrum is fully utilized, especially in licensed bands. In order to solve this problem, the concept of cognitive radio is proposed. Cognitive Radio is a technique which allows cognitive radio users to use channels that are not owned by them without interfering the primary users of channels. In other words, when a cognitive radio user is using a channel without any authorization, it should switch to other channels immediately if a licensed user wants to use this channel so that the licensed user will not be interfered by the cognitive radio users.

Nowadays, simulators that can support the research of cognitive radio networks are very rare. In addition, most of them can only simulate the physical and MAC layers of a cognitive radio network and lack the ability to simulate a cognitive network with high-layer protocols such as TCP/IP. In this thesis, we implement an IEEE 802.11 cognitive radio module on the NCTUns network simulator so that researchers can use it as a base to test their novel ideas. We also study the effects of hidden-terminal scenarios on the performances of cognitive radio networks and propose several enhancements to the basic scheme to improve its performances under hidden-terminal conditions.

Keywords: Cognitive Radio, Hidden-Terminal Problem, Network Simulator

誌 謝

在這兩年的研究所生涯中，首先我要感謝我的指導教授王協源老師，不斷的在研究上給予我指導，也在學術研究上教給我不少知識，使得我對於所鑽研的領域的根基能夠穩固且紮實。此外老師總是能在我退卻時鼓勵我再對於研究進行更深入的探討，使得我總是能夠持續進行研究，並且獲得非常豐碩的成果。

此外，在此也要感謝周承復老師、趙禧綠老師、鄭振牟老師，感謝你們願意擔任這次碩士口試的口試委員，並願意給予許多的意見，使得我能注意到很多疏漏的地方，或是考慮不盡周詳之處。有了各位老師們的指教，相信這篇論文會更好，感謝各位老師在口試時給予的指點。

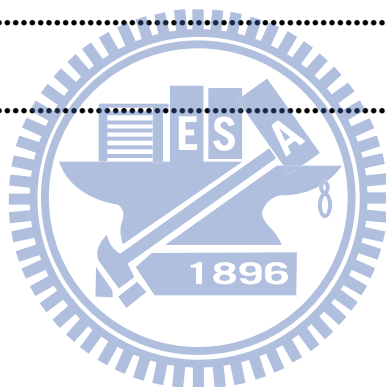
另外，也要感謝實驗室的學長、同學與學弟妹們在這段時間內的支持與幫助，學長們總是會在平常給予我們一些方向，並且肯花費自己的時間與我們進行討論，使我學到了很多的東西，也更能釐清自己的思路。而同學與學弟妹們總是能夠協助我分擔一些工作，並且在適當時機給予我意見或討論，感謝你們陪伴了這兩年的時光。

最後，我要感謝我的父母、以及以前在清華大學與中興大學的老師與同學們，在這兩年的時間內給予我不少的鼓勵與關心，使我能夠沒有任何的後顧並且往前衝刺，若沒有了你們，這段生活與研究的成果也將黯淡不少，感謝各位的陪伴。

目次

摘要.....	i
Abstract.....	ii
誌謝.....	iii
目次.....	iv
表目錄.....	vi
圖目錄.....	vii
1. 介紹.....	1
2. 背景.....	3
2.1. 相關研究.....	3
2.2. NCTUNS 網路模擬器介紹.....	5
3. 無線感知網路模擬模組之實作.....	8
3.1. 實作之無線感知網路機制介紹.....	8
3.1.1. 介紹.....	8
3.1.2. 封包傳輸之前置作業：快速頻道感測.....	9
3.1.3. 傳輸之後置作業：主動釋放頻道.....	11
3.2. 無線感知網路模組之實作.....	12
3.2.1. 概要.....	13
3.2.2. NCTUns 相關節點與模組.....	13
3.2.3. 實作.....	15
4. 原有機制之改良.....	26
4.1. 原有機制之效能增討論.....	26

4.2.	改進之方法與其設計實作.....	31
4.2.1.	頻段挑選機制.....	31
4.2.2.	資料傳輸頻段感測機制之改良.....	33
4.2.3.	SIFS _{CR} 後資料傳輸之改良.....	34
5.	效能評估與分析.....	36
5.1.	理論值之推導與估計.....	37
5.2.	無終端機隱匿問題的模擬情境之效能評估.....	45
5.3.	在終端隱匿之情境之效能比較.....	60
6.	未來展望.....	75
7.	結論.....	77
8.	參考文獻.....	78



表目錄

表 1、於效能評估中的參數設定。.....	37
表 2、改良機制與原有機制對無線感知網路在無終端機隱匿問題的 MAC 層之流量傳輸效能比較。(單位：Mbps).....	51
表 3、改良機制與原有機制對於無線感知網路在無終端機隱匿問題以 MAC 層之協定直接進行流量傳輸的 Round-Trip Delay 比較。(單位：ms).....	53
表 4、改良機制與原有機制對無線感知網路在無終端機隱匿問題的 TCP 之流量傳輸效能比較。(單位：Mbps).....	57
表 5、改良機制與原有機制對於無線感知網路在無終端機隱匿問題以 TCP 通訊協定來進行流量傳輸的 Round-Trip Delay 比較。(單位：ms).....	60
表 6、改良機制與原有機制對無線感知網路在終端機隱匿問題的 MAC 層之流量傳輸效能比較。(單位：Mbps).....	65
表 7、改良機制與原有機制對於無線感知網路在終端機隱匿問題下以 MAC 層之協定直接進行流量傳輸的 Round-Trip Delay 比較。(單位：ms).....	67
表 8、改良機制與原有機制對無線感知網路在終端機隱匿問題的 TCP 之流量傳輸效能比較。(單位：Mbps).....	72
表 9、改良機制與原有機制對於無線感知網路在終端機隱匿問題下以 TCP 通訊協定來進行流量傳輸的 Round-Trip Delay 比較。(單位：ms).....	74

圖目錄

圖 1、NCTUns 模組化之平台。.....	6
圖 2、實作之無線感知網路模組之機制流程圖。.....	9
圖 3、遭遇某一頻段無法使用之狀況的處理。.....	11
圖 4、在 NCTUns 中對於 IEEE 802.11b Ad-hoc 模式之節點的通訊協定堆疊。.....	14
圖 5、本次實作之類別圖。.....	16
圖 6、RTS _{CR} 之表頭結構。.....	17
圖 7、RTI 之表頭結構。.....	17
圖 8、發送端之行為狀態循環圖。.....	19
圖 9、接收端之行為狀態循環圖。.....	20
圖 10、不適用於原先 NCTUns 感測頻段忙碌流程之實例。.....	22
圖 11、由終端機隱匿問題之情境來考慮決策者的選定。.....	27
圖 12、當頻段感測到 ACK 後，即使頻段接下來無任何使用者使用，仍需切換頻段。	28
圖 13、在終端隱匿問題下，傳送端的行為反應。.....	30
圖 14、閒置記錄與評分方式之示意圖。.....	31
圖 15、修改後的 RTS _{CR} 之結構。.....	32
圖 16、修改後的 CTS _{CR} 之結構。.....	33
圖 17、修改後的頻段感測之決策機制。.....	34
圖 18、無終端機隱匿問題情境下的拓樸。.....	46
圖 19、原有機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線 感知網路使用者的效能。.....	47
圖 20、原有機制於無終端機隱匿問題之情境下，主要使用者的效能。.....	48
圖 21、改良之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無	

線感知網路使用者的效能。.....	49
圖 22、改良機制於無終端機隱匿問題之情境下，主要使用者的效能。.....	49
圖 23、原始之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的 Round-Trip 延遲時間。.....	52
圖 24、原始之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的 Round-Trip 延遲時間。.....	52
圖 25、原有之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。.....	54
圖 26、原有之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時，主要使用者的效能。.....	54
圖 27、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。.....	55
圖 28、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時，主要使用者的效能。.....	56
圖 29、原始之機制於無終端機隱匿問題之情境下，以 TCP 作為進行封包之方法時，無線感知網路使用者的 Round-Trip 延遲時間。.....	58
圖 30、改良之機制於無終端機隱匿問題之情境下，以 TCP 作為進行封包之方法時，無線感知網路使用者的 Round-Trip 延遲時間。.....	59
圖 31、終端機隱匿問題情境下的拓樸。.....	61
圖 32、原有機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的效能。.....	62
圖 33、原有機制於無終端機隱匿問題之情境下，主要使用者的效能。.....	62
圖 34、改良之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的效能。.....	63
圖 35、改良之機制於無終端機隱匿問題之情境下，主要使用者的效能。.....	64

圖 36、原始之機制於無終端機隱匿問題之情境下，無線感知網路使用者的 Round-Trip 延遲之量測。.....	66
圖 37、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者的 Round-Trip 延遲之量測。.....	66
圖 38、原有之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。.....	69
圖 39、原有之機制於無終端機隱匿問題之情境下，主要使用者的傳輸效能。.....	69
圖 40、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。.....	70
圖 41、改良之機制於無終端機隱匿問題之情境下，主要使用者的傳輸效能。.....	71
圖 42、原始之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時的 Round-Trip 延遲時間。.....	73
圖 43、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時的 Round-Trip 延遲時間。.....	73



1. 介紹

隨著無線網路愈來愈深入的研究與探討，目前可為人所用的無線網路傳輸之協定與其所使用之頻譜也隨之增加。然而隨著技術與其搭配的頻譜的不同，這些可使用的頻譜也被切割成多個區塊，並且被分配給特定的使用者，並給予其執照，這類使用者在本論文中稱之為主要使用者 (Primary User，簡稱 PU)；此外，某些頻段也被劃分為一般使用者皆可使用之頻段，如 2.4 GHz、5.8 GHz 等。但是，根據近年來對於這些頻譜使用情形之研究指出多數的頻段其使用率並不高，亦即這些頻段的使用者並非完整運用這份資源，而是這個頻段在多半的時間均維持在閒置的狀態，這樣的狀況尤其在需要持有合法執照才能使用的頻段較為嚴重；但在其他頻段中卻容易出現頻段使用擁塞之情況，使得在此頻段之可用資源嚴重不足[1]。因此在考慮在一般較少持有執照之主要使用者的情況下，若此處的使用者均使用某些特定頻段，但這些頻段無法承受如此多之使用者，在這樣的狀況下可見頻段的使用並不合實際需求。因此，便有人想到了無線感知網路 (Cognitive Radio) 之概念，希望在不影響正式使用者的情況下，能夠借用其頻段的資源來進行傳輸，藉此能夠使一般使用者能順利進行資料傳輸，也能提高其頻段的使用效能。

無線感知網路的實作概念最早是由 Joseph Mitola III 與 Gerald Q. Maguire, Jr 在 1999 年於[2]提出，內容主要是提出一套無線電知識表示語言 (Radio Knowledge Representation Language) 並搭配軟體無線電 (Software Radio) 之硬體平台來使得無線網路設備能利用軟體調整參數的方式來控制硬體所使用的頻段與訊號強度等之控制，如此能協助無線網路設備得以取得或分享其所需要之資訊，進而讓其能夠擁有找尋自己所需要之傳輸頻段的能力。

而無線感知網路的定義方面，在第一段中曾提及在近年來頻譜已受特定之機構進行切割與配置，因此在每個已分配的頻段中都會存在一些擁有合法使用執照的使用者，稱為主要的使用者 (Primary Users，簡稱 PU)。而對比於這些使用者，另外還有一些使用

者則為次要的使用者 (Secondary Users, 在本論文中這些使用者被稱作 CR Users), 這些使用者他們若需要使用那些需要授權才能使用的頻段時, 需要進行一些偵測的動作以找出主要使用者在傳輸時的空閒時間, 藉此以避免干擾到主要使用者利用這些頻段的動作, 畢竟這個頻段本身就並非這些次要的使用者獲得授權而可以使用的。

然而, 在現階段對於無線感知網路的研究中所擁有的測試工具並不多, 而且對於更高層的網路通訊協定之應用程式的支援也相當的有限, 使得研究人員在驗證上難以利用更為複雜的情境來進行效能的評比。因此, 本篇論文主要由目前既有的無線感測網路的相關研究進行實作, 以提供給研究人員一套完整個無線感測網路的模擬環境, 藉此能夠在沒有實體設備的狀況下進行通訊協定的開發。此外, 本文也將基於該無線感測網路之通訊協定進行改善, 期望能藉此使其效能提升。

因此, 接下來的章節配置如底下所示: 在第二章中將會介紹目前已有的無線感測網路的相關研究; 此外, 也會針對本篇論文於實作時所使用的網路模擬器—— NCTUns [3] 進行介紹。第三章將會介紹在本篇論文中用以實作的無線感知網路之通訊協定, 並會介紹如何將其實作於 NCTUns 這套網路模擬器內。而在接下來第四章的部份則會對於這一個通訊協定進行改善, 使得在終端機隱匿問題 (Hidden-Terminal Problem) 的情境下依然能夠維持其良好的傳輸效能。在第五章的部份, 本篇論文將提出數個模擬情境, 並針對於第三章與第四章進行之模組開發設計的結果進行模擬與分析。而在最後兩章則分別為結語與未來發展的部份。

2. 背景

承接第一章的部份，無線感知網路自概念發表至今已歷經了不短的時間，在這幾年之間，各種想法與實作技巧也日益增加與成熟，因此在本章節中將展示在進行此篇論文之研究過程中所進行的相關研究之探討。此外也將介紹在本論文中用以實作之模擬器—— NCTUns。

2.1. 相關研究

在相關研究的調查上，由於本篇論文在最初著重於開發無線感知網路之模擬環境，因此在進行探討上主要是先以無線感知網路相關模擬器為主，接著是進行是否有與此研究相關的論文提出了幾種不錯的設計機制。而在相關模擬器的部份，目前大多數已發表的期刊論文中大多都是以自行撰寫一支程式，並且套用自行研究之無線感知網路之模型以進行驗證。除此之外，仍然還有一些組織試圖進行更精細的模擬平台之開發，如[4]與[5]。在目前[4]已在 NS-2 進行相關的模組開發，當中設計了無線感知網路路由模組、MAC 模組、以及透過多重頻段之機制設計了 PHY 模組。但由於 NS-2 中對於較高層應用程式的支援在使用上較為繁雜，因此在此考慮使用其他模擬器來進行環境的開發。而在[5]中則提供了另一種稱作 Agent-Based 的模擬機制，並以訊號的角度進行精細的模擬，以挑選合適的頻段，但這可能無法對於更高層應用程式進行支援。

除了模擬平台的設計之外，目前已有不少成熟的無線感知網路在 MAC 機制之設計。其中在[6][7][8]中對無線感知網路提供了一個完整的介紹，並且探討了很多對於無線感知網路中的某些機制設計的相關文章。[6]本身較著重提供一個完整的流程，使得在進行相關的研究中可以知道無線感知網路之研究領域約可以劃分成哪些項目，並從中進行相關文章之整理。此外，[7]主要是針對頻段感測之演算法來進行相關研究的分類與探討，這篇文章對於這份論文在改進機制的設計與研究上給予了相當大的啟發。而[8]是針對多

頻段的無線感知網路之 MAC 來進行目前研究上探討並分類。而於[9][10][11]這幾篇論文中各自提供了一套完整的機制，藉著使用這套機制能使得感知網路的使用者在不干擾主要使用者的情況下進行傳輸。特別是在[9]中介紹了一套機制，使得在傳輸的過程中並不需要額外仰賴控制資料傳輸頻段。除此之外，在研讀這些相關研究中也發現，目前現存透過控制資料傳遞頻段進行無線感知網路機制設計之研究中，大多數的設計上均是透過類似 RTS-CTS 的機制先行在控制頻段進行溝通，之後才切換至資料控制頻段，如[10]與[11]中即如此。其中[10]這篇在無線感知網路的設計上保留了不少的彈性，使得無線感知網路的使用者在其效能上仍然有一定的水準，使人印象深刻。最後是[11]中提供了以統計的方式來進行頻段配置的一種無線感知網路的機制。

而除了完整的 MAC 機制設計之外，頻段的決定往往會影響日後無線感知網路的使用者在進行傳輸時的效能，為了檢視是否能有可以運用並與前段所找出的 MAC 機制結合的頻段存取機制，在頻段存取的相關研究方面也進行了一些相關研究的研讀，其中，[12][13][14]則是提出了多種頻段存取的演算法，藉由這些演算法能使得在進行頻段存取更有效率。其中[12]提出了基於傳輸能量與以實體層之 Preamble 封包為底的頻段存取與分析之演算法；而[13]則是提出了一套稱作預測性的動態頻段存取 (PDSA) 的想法，透過此想法可對主要使用者進行頻段使用進行分析，並嘗試找出可用的空間頻段；[14]則是以分析與歷史記錄的角度試圖逼近頻段預測的方式建造出一套預測模型。這些方式均有助於無線感知網路使用者進行頻段使用的規劃。

透過這些相關研究的研讀，可以了解到目前無線感知網路的機制設計上大略可分為頻譜的感測、頻譜的使用與頻譜的共享機制，透過這些機制的統合能夠給予無線感知網路的使用者一套完整的傳輸方式，使其能透過使用多個頻段來進行資料傳輸。而在經過仔細研讀後，由於[10]中在設計上具有清楚且完整的架構，此外其機制也提供了不小的彈性，因此本篇論文將接續[10]的研究，將此環境設計在 NCTUns 網路模擬器上，並在此環境中對此機制進行改良。

2.2. NCTUns 網路模擬器介紹

在本篇論文的實作主要是採用 NCTUns 網路模擬器，這一套網路模擬器主要是由國立交通大學所發展出的一套網路模擬器，至今支援了多種網路架構如：Ethernet、光纖網路、IEEE 802.11 a/b/e/p、GPRS、WiMAX (IEEE 802.16 d/e/j)、衛星網路 DVB-RCS 等，使得使用者能夠按照其所設計的模擬情境以進行模擬。而這套網路模擬器具備以下之優點，這也是在本篇論文研究中採用此模擬器之主因：

- 其使用了真實世界中所用的傳輸層與網路層的協定堆疊 (Protocol Stack)，進而能產生更精確的模擬結果。
- 真實世界的應用程式大多在不經修改的情形下可以直接運行於此模擬器上。
- 具有高度整合性的使用者圖形化介面 (Graphical User Interface)，使得模擬情境的設計更為簡潔。

而為了支援以上的優點，在其架構的設計上則包括了幾個基本的元件：(1) 修改後的 GNU/Linux 核心、(2) NCTUns 圖形化使用者介面、(3) NCTUns 模擬引擎、(4) NCTUns 協定模組 (Protocol Modules)。透過對於 GNU/Linux 核心的修改，其可讓真實世界中所用的利用 Socket 介面撰寫之網路應用程式能夠在不經修改的情況下，使其可直接做為模擬情境中的某一部網路設備上所運行的應用程式，藉此可模擬其在各種網路下的運作行為。而在圖形化使用者介面的部份則提供了一般使用者可直接以所見即所得的方式進行拓樸的設計規劃，使得使用者可以在不用學習任何腳本 (Script) 設計的情況下進行拓樸規劃。而在 NCTUns 模擬引擎的部份，則提供了對於時間、事件、排程、封包裝填等重要的部份一套完整的 API，使得開發者能夠對此容易上手。最後則是協定模組的部份，其對於任意模擬的網路設備的協定堆疊 (Protocol Stack) 採用了模組化的設計，透過此

設計可以使得使用者在開發自己所需的模組後，只需要以替換或加入此模組的方式，即可使得其所設計之模組得以直接在模擬世界中運行。

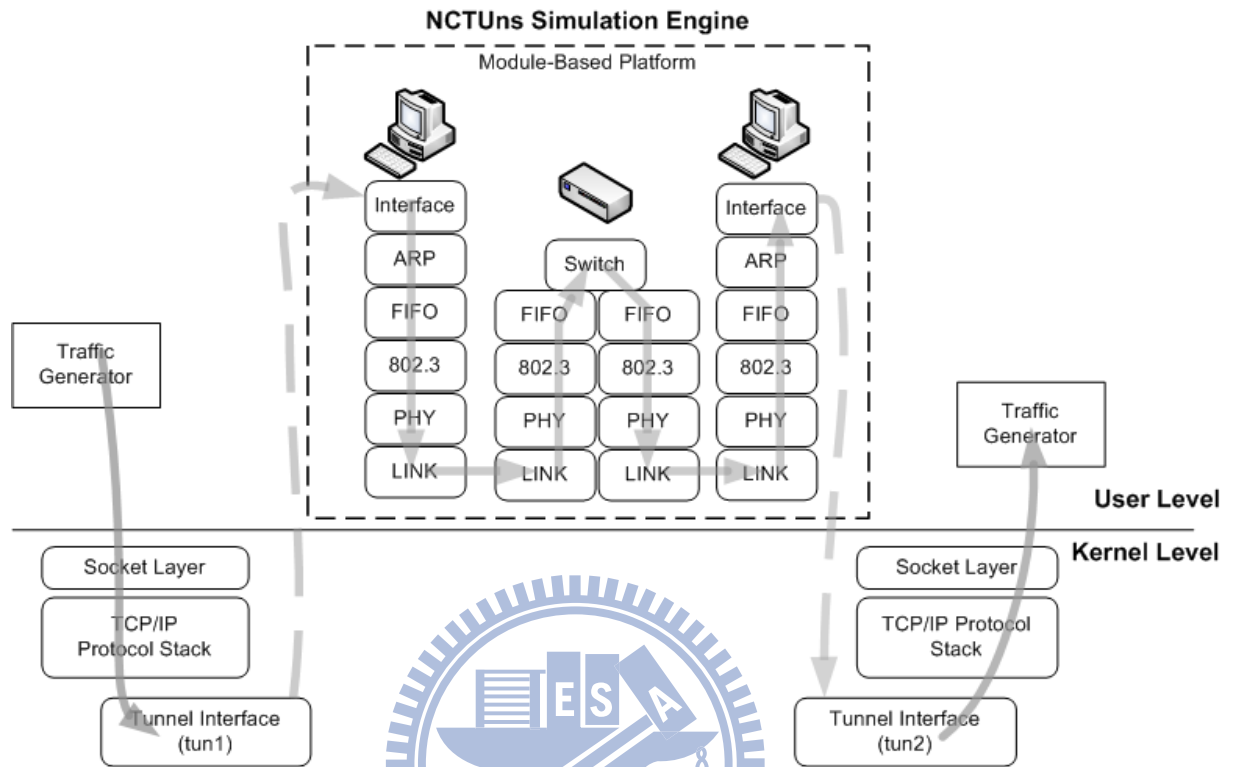


圖 1、NCTUns 模組化之平台。

圖 1 呈現了 NCTUns 之模組化模擬平台，這張圖分為上半部份與下半部份，其中上半部份為作業系統中使用者層級的應用程式，其中包括了流量產生的應用程式，在此稱為 Traffic Generator、NCTUns 模擬引擎，而底下的部份則為系統核心的部份。而由圖片左邊開始，可以看到一個在使用者層級之 Traffic Generator，這個程式是以 Socket 介面來撰寫的，並試圖透過 TCP/IP 的通訊協定將封包傳往圖片右側的應用程式接收端。當左方的應用程式送出封包時，它會先透過系統核心的 Socket 層，接下來則是傳輸層的 TCP 表頭 (Header) 的包裝與處理、網路層 IP 表頭的包裝，最後到達系統核心內的一個虛擬的網路介面卡稱作 Tunnel 介面，透過此介面送出的封包將會被導至 NCTUns 模擬引擎內進行模擬。當封包傳入模擬引擎時，此時封包會先經由 Interface 模組進入模擬引擎，並且經過模擬的協定堆疊：ARP、FIFO、802.3、PHY 及 LINK 模組，然後轉往中

間模擬的交換器 (Switch) ，同樣的這個封包也會遵守裡面所配置的模組，並依：LINK、PHY、802.3、FIFO、Switch 模組順序往上收，接下來再由交換器送出，並傳至右端的用戶端設備，其所經模組可以此類推。直至傳到右端的 Interface 模組，此時此模組會將此封包透過寫入 Tunnel 介面的方式送至系統核心，系統核心會進行判斷此是否為該設備之封包，是的話會往上層送；不是的話則會協助此封包之路由轉送。而接收的部份仍是經由 IP 層、TCP 層、Socket 層，最後會由右方 Traffic Generator 的 read()讀取。

而根據這樣的流程，為了在設計上的一致性與便利性，因此每一模組均繼承自 nsobject 這個類別，並根據模組在設計規劃上的不同，實作出不同的 send 與 recv 兩個成員函式，使得封包在抵達此模組時能夠順利的被處理。而在這些函式內可透過如計時器、事件的產生、新增封包的方式來達到在各種不同網路規格書內所刊載的處理程序，使得各式各樣的網路能夠輕易的在此平台上進行開發設計並模擬。

而由於本節主要用於概略介紹 NCTUns 這套網路模擬器，且在本論文中實作的部份主要是著重在模組的改寫上，因此更多關於模組設計上的細節則在此不進行更深入的描述，詳細的模組開發與設計方法可參考[15]。

3. 無線感知網路模擬模組之實作

本章主要將介紹在本論文中將實作之無線感知網路的通訊機制，並且說明如何將此實作於 NCTUns 這套網路模擬器上，使得一般研究人員能夠運用此模擬器與相關模組，進行更進一步的研究與效能評估。

3.1. 實作之無線感知網路機制介紹

3.1.1. 介紹

在第二章的相關研究成果的部份，其中，[10]在 IEEE 802.11 之無線網路中提供了一套極為清楚且簡單的機制，並在機制中留下了一些可進行設定與的變數。這套機制主要分成兩個階段：快速頻道感測 (Fast Channel Sensing) 與主動釋放頻道。透過這兩階段的評估與無線感知網路之使用者的行為，可以達到在不干擾主要使用者的情況下，無線感知網路之使用者在傳輸上的效能也能夠有一定的水準。因此，在本篇論文中將以這個協定為基礎，將其實作於 NCTUns 這套網路模擬器中。

而在這套機制中，它將使用之頻段分為控制封包傳輸頻段 (Control Channel) 與資料傳輸頻段 (Data Channel)，在最初的狀態中，所有的無線感知網路使用者均在控制封包傳輸頻段中交換訊息，直到當有使用者需要進行資料傳輸時，才會透過控制封包傳輸頻段向接收者約定接下來應該要在哪一個資料傳輸頻段進行資料傳輸。在約定完成後，雙方即會切換使用頻段至該資料傳輸用之頻段以進行資料傳輸，待傳輸完成，兩位使用者才會逕行回到控制封包傳輸頻段等待其他的資料傳輸。

整體傳輸之流程圖如下所示：

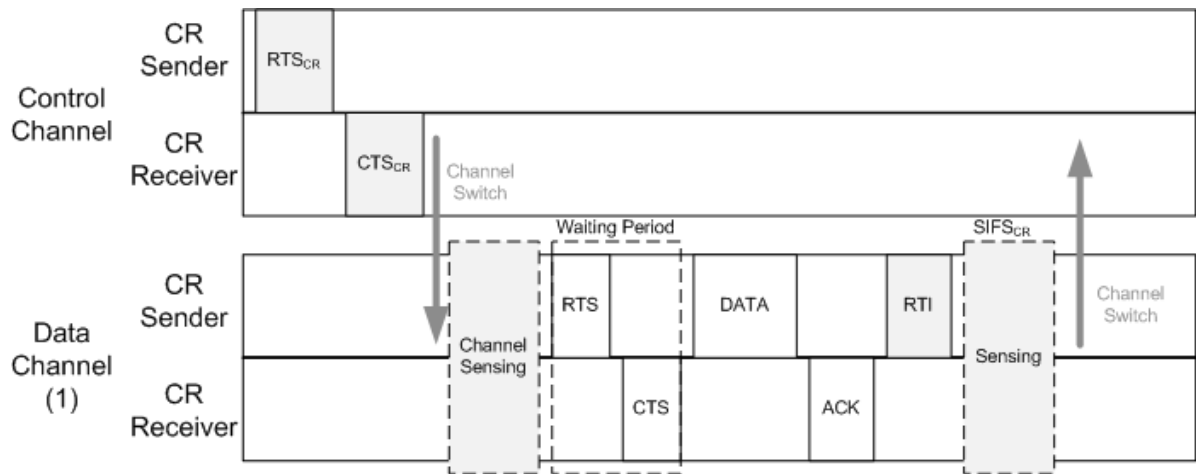


圖 2、實作之無線感知網路模組之機制流程圖。

3.1.2. 封包傳輸之前置作業：快速頻道感測

如圖 2 所示，這一套機制中提出了三種新型態的控制封包，這三種控制封包分別稱作 RTS_{CR} 、 CTS_{CR} 、 RTI 。而在其中進行握手動作 (Handshaking) 的控制封包則如下所示：

(1) RTS_{CR} ：如同在 IEEE 802.11 中制訂的 RTS 控制封包類似， RTS_{CR} 本身是用來進行資料交換前的溝通，但這個控制封包主要是由無線感知網路的發送端送出，使其能與接收端在控制封包傳輸頻段進行在後續資料的交換上的握手動作 (Handshaking)，如此能讓接收端得知接下來所使用的資料傳輸頻段之資訊。而在本設計機制中，在此使用了一套稱作快速頻段感測 (Fast Channel Sensing) 之機制，這套機制中，使用者僅需要提供兩種資料：初始頻段 (Initial Channel)、切換頻段遞增值。這兩個值與快速頻段感測之機制將會之後介紹。

(2) CTS_{CR} ：如同在 IEEE 802.11 中制訂的 CTS 控制封包類似， CTS_{CR} 本身是用來進行資料交換前的溝通，但和前面 RTS_{CR} 不同的地方則是：它本身是接收端用

以回應傳送端 RTS_{CR} 所用的，在內容的部份則不帶有任何與頻段相關之訊息。

快速頻段感測之機制內容主要由初始使用頻段、切換頻段遞增值構成，其代表的意義是在需要使用資料傳輸頻段時，最一開始使用的頻段即由初始頻段 (Initial Channel) 來決定。但若因為某些特殊理由而無法使用這些頻段，則代表傳送者與接收者需要再找下一個資料傳輸頻段來進行資料傳輸，此時它們所用到的頻段將會是目前所在頻段加上切換頻段遞增值，若相加後的值大於總使用之頻段數，則會以取餘數加一的方式處理。以數學示表示如下：〈 $Ch(i)$ 代表第 i 次切換所使用的頻段，其值域為 $1 \leq Ch(i) \leq N$ ； h 則代表切換頻段遞增值； N 則為總頻段數。〉

$$Ch(i+1) = \begin{cases} Ch(i) + h & Ch(i) + h \leq N \\ ((Ch(i) + h) \bmod N) + 1 & \text{Otherwise} \end{cases}$$

而為了避免在切換頻段時所切換的頻段只集中在某些特定頻段，因此另外的規定則是 N 與 h 需互質，亦即 $G.C.D(N, h) = 1$ 。



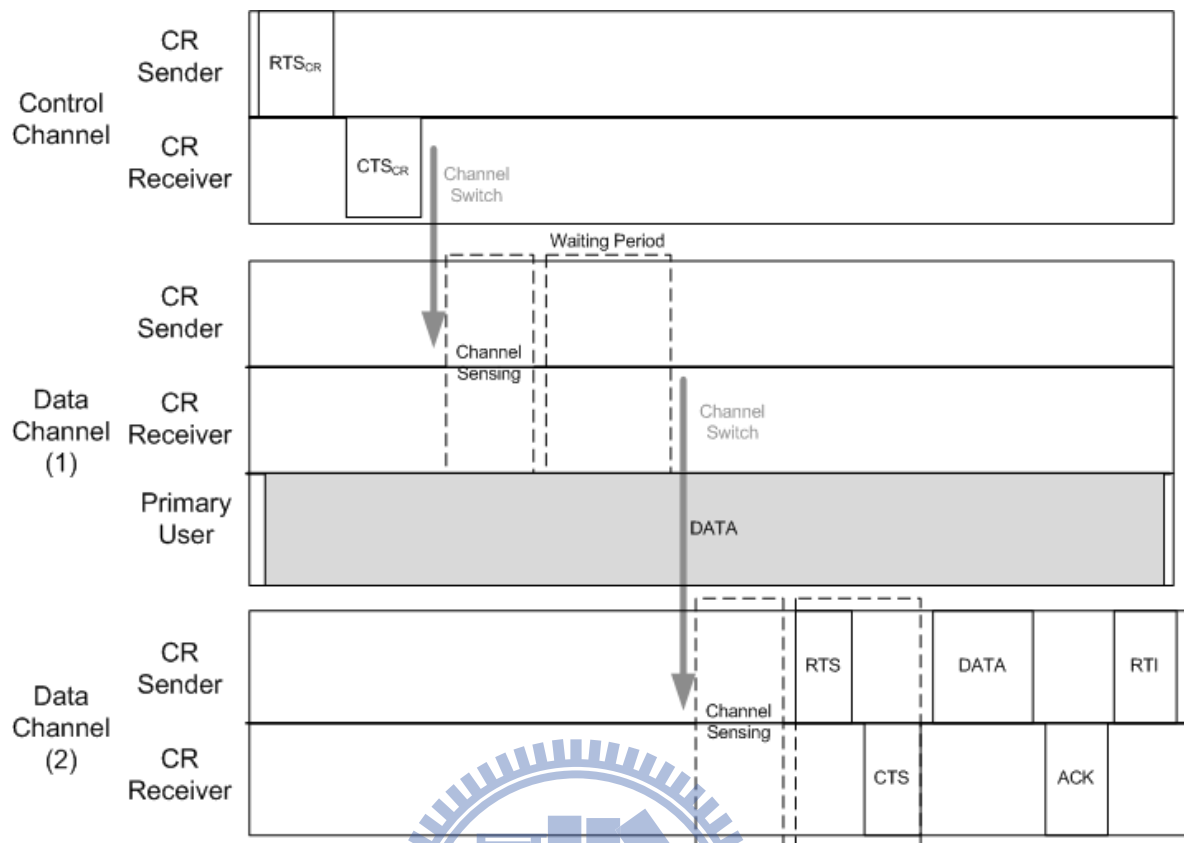


圖 3、遭遇某一頻段無法使用之狀況的處理。

而在此處所提到的「為某些特殊理由而無法使用這些頻段」的例子如圖 3 所示，圖中可以看到，在此無線感知網路之機制下，當傳送端與接收端切換至資料傳輸頻段時，它們會進行頻段感測的動作，之後並判斷是否能夠使用該頻段。若無法使用該頻段，則會在等待的時間區間 (Waiting Period) 之後，雙方一起切換至下一個頻段。而等待時間區間則是用以讓無線感知網路的使用者在資料控制頻段進行 RTS、CTS 之握手動作用的，若在此時間內未完成 RTS、CTS 之訊息交換，則代表此頻段可能正有使用者在使用中。

3.1.3. 傳輸之後置作業：主動釋放頻道

而在一次的資料傳輸完成後，為了避免干擾到此頻段的主要使用者，因此在 ACK 控制封包傳輸完成後，感知網路使用者的傳送端會送出一種控制封包稱作 RTI

〈Ready-To-be-Interrupted〉。當這個控制封包送出時，感知網路使用者的傳送端與接收端雙方均要等待一段稱作 $SIFS_{CR}$ 的時間。這段時間主要是用以讓主要使用者有辦法表達他想要使用該頻段的意願，亦即標題的「主動釋放頻道」。因此，若無線感知網路的使用者在此時間區間內收到任何的訊息，那很有可能就代表有主要使用者想使用該頻段，他們應該要回到控制封包傳輸頻段重新約定接下來應該要使用哪個頻段進行傳輸。

但是若此時沒有任何使用者在此頻段，或是主要使用者沒有意願傳輸的情況下，若只傳了一個資料封包即需要重新再次決定下一個封包應該使用哪一個頻段，這樣的情況下需要再次切換回控制封包交換頻段、交換 RTS_{CR} 、 CTS_{CR} 、切換至資料傳輸頻段、進行頻段感測，如此會需要額外再花費時間進行這些動作。因此為了減少這些時間的花費，在此定義一種稱作 $TxOP_{CR}$ 的參數，這個參數的意義主要是定義允許無線感知網路能在同一個資料傳輸頻段內進行資料傳輸的最大次數，當無線感知網路之使用者在等待完 $SIFS_{CR}$ 之時間間隔後，若此時沒有其他使用者進行資料傳輸，則傳送端即可直接再送出一個 DATA 封包給接收端，而後續動作仍為 ACK 與 RTI 的交換，如此即使用了第二次的傳送機會。若接下來仍然沒有其他使用者，則此狀況將會持續到其使用了第 $TxOP_{CR}$ 次的傳輸機會為止，屆時此二使用者仍需回到控制封包傳輸頻段再次進行 RTS_{CR} 、 CTS_{CR} 的交換。

3.2. 無線感知網路模組之實作

在了解所要實作的無線感知網路之機制後，本節將介紹如何將此實作於 NCTUns 這套網路模擬器內。因此本節會先為本篇論文實作的內容進行概略性的介紹，接下來會針對 NCTUns 內的環境進行更進一步的介紹，再來會說明如何進行實作。

3.2.1. 概要

根據 3.1 節的敘述，可以了解該無線感知網路之機制。而這些有別於原始 IEEE 802.11 DCF MAC 的部份可簡單區分為底下幾點：

- 控制封包 RTS_{CR} 、 CTS_{CR} 、 RTI 的新增。
- 多重頻段之切換與感測。
- Ready-To-be-Interrupted 之機制的實作。

這套機制主要是以 IEEE 802.11 所制定的規格為基礎來進行延伸。而在目前 NCTUns 這套模擬器上對於 IEEE 802.11 系列之網路的支援為 IEEE 802.11a、IEEE 802.11b、與無線車載網路 IEEE 802.11p，其中前二者的 MAC 層在實作時均為 DCF 的實作，與目前之情境較為相似，因此本次即先從中挑選其一。而為了進行驗證本身的實作的正確性，因此在此先以 IEEE 802.11b 的實作為主。此外，在此模擬器內使用 IEEE 802.11b 的節點中支援了 Infrastructure 模式與 Ad-hoc 模式，考慮到往後對於多段 (Multi-hop) 傳輸研究上的彈性，因此在本次的實作中會以 Ad-hoc 的網路節點為主。

而在實際應用的情況，這些機制可以在單一天線與多天線之情況下均適用。但考慮到 IEEE 802.11b 的實作中均以單一傳送接收設備為主，亦即在同一時間點內只能使用單一頻段。因此在實作中主要著重在「在同一時間點只能感測與使用單一頻段」的情境下，使得此無線感知網路之機制能在此模擬器上運作。

3.2.2. NCTUns 相關節點與模組

如 3.2.1 中所述，在此實作中主要是以 IEEE 802.11b 的 Ad-hoc 模式做為實作之主要對象，而在原本的節點中其在 IP 層以下的通訊協定之堆疊如圖 4 所示：

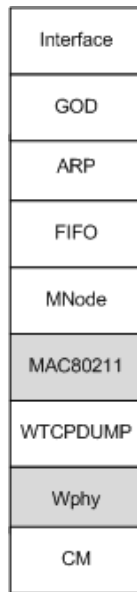


圖 4、在 NCTUns 中對於 IEEE 802.11b Ad-hoc 模式之節點的通訊協定堆疊。

於先前 2.2 節提到，NCTUns 本身使用了利用 GNU/Linux 本身所提供於真實世界中所使用的傳輸層與網路層的通訊協定堆疊的方式，協助我們避免需要自己模擬這些部份進而使得模擬嚴重失真的狀況。當一個封包由某個節點發送時，它會經過系統核心，並且送到對應的 Tunnel 介面。在送至 Tunnel 介面時，此時，該 Tunnel 介面的裝置驅動程式會發出通知，使得在圖 4 中最頂端的 Interface 模組會從對應的介面中將此封包導引至模擬的世界中。而在圖 4 所呈現的通訊協定堆疊模組各自的功用則如下所示：

- Interface 模組：用以記錄 IP 位址與 MAC 位址對應之關係，此外也協處理封包自 Tunnel 介面的讀取與寫入之工作，使得封包能順利被引導至模擬的世界中，或是送至系統核心內進行路由處理、甚至是被對應的應用程式接收。
- GOD 模組：此模組用以協助 Ad-hoc 模式下，MAC 層路由的處理。這個模組主要會透過 NCTUns 使用者圖形化介面在模擬前即計算好在模擬時哪些節點之間有能力直接進行傳輸，並在模擬時實際根據此運算結果進行處理。

- ARP 模組：此模組用以處理 ARP (Address Resolution Protocol) 之相關行為的模擬。
- FIFO 模組：用以模擬由網路層與資料連結層之間的緩衝佇列。
- MNode 模組：MNode 即為 Mobile Node，在此模組中主要負責從網路層傳下來的封包進行包裝 MAC 表頭 (Frame) 的處理；此外也隨著節點的不同，此模組會協助其發送 MAC 層的封包，如 BEACON 等，但不包括 RTS、CTS、ACK 等涉及於其動作行為封包的產生。
- MAC80211 模組：用以模擬 IEEE 802.11 DCF 之行為的模組。
- WTCPDUMP 模組：用以協助 tcpdump 這支工具程式對於無線網路封包的支援，使其能捕捉到在模擬世界中的封包。
- Wphy 模組：用以模擬 IEEE 802.11 之規格下所定義的實體層。
- CM 模組：用以模擬無線傳輸中訊號衰減處理的模組，目前支援多個在期刊與會議中已發表的理論或實際量測而得到的訊號模型。

由於在本次實作的內容均與 MAC 之行為與實體層之訊號感測之部份相關，因此在實作上只需要更改 MAC80211 與 Wphy 這兩個模組，其他模組可維持不變，如此即可完成整個系統的設計。在下一個段落中將介紹本論文對於這兩個模組進行哪些修改與設計，使得欲模擬的機制得以順利進行模擬。

3.2.3. 實作

承接 3.2.1 所描述的通訊協定堆疊，在本次的實作中僅需針對 MAC80211 與 Wphy 這兩個模組進行更改。但為了使得一般的使用者與使用無線感知網路的使用者能夠並存，因此勢必需要產生新的模組。因此，在最一開始即複製整個 MAC80211 與 Wphy 模組，

並在其後加入_CR 之字串以示區隔。而完成後的類別名稱則為 mac802_11dcf_CR 與 wphy_CR。而整個的修改架構則如圖 5 的簡化後之類別圖所示：

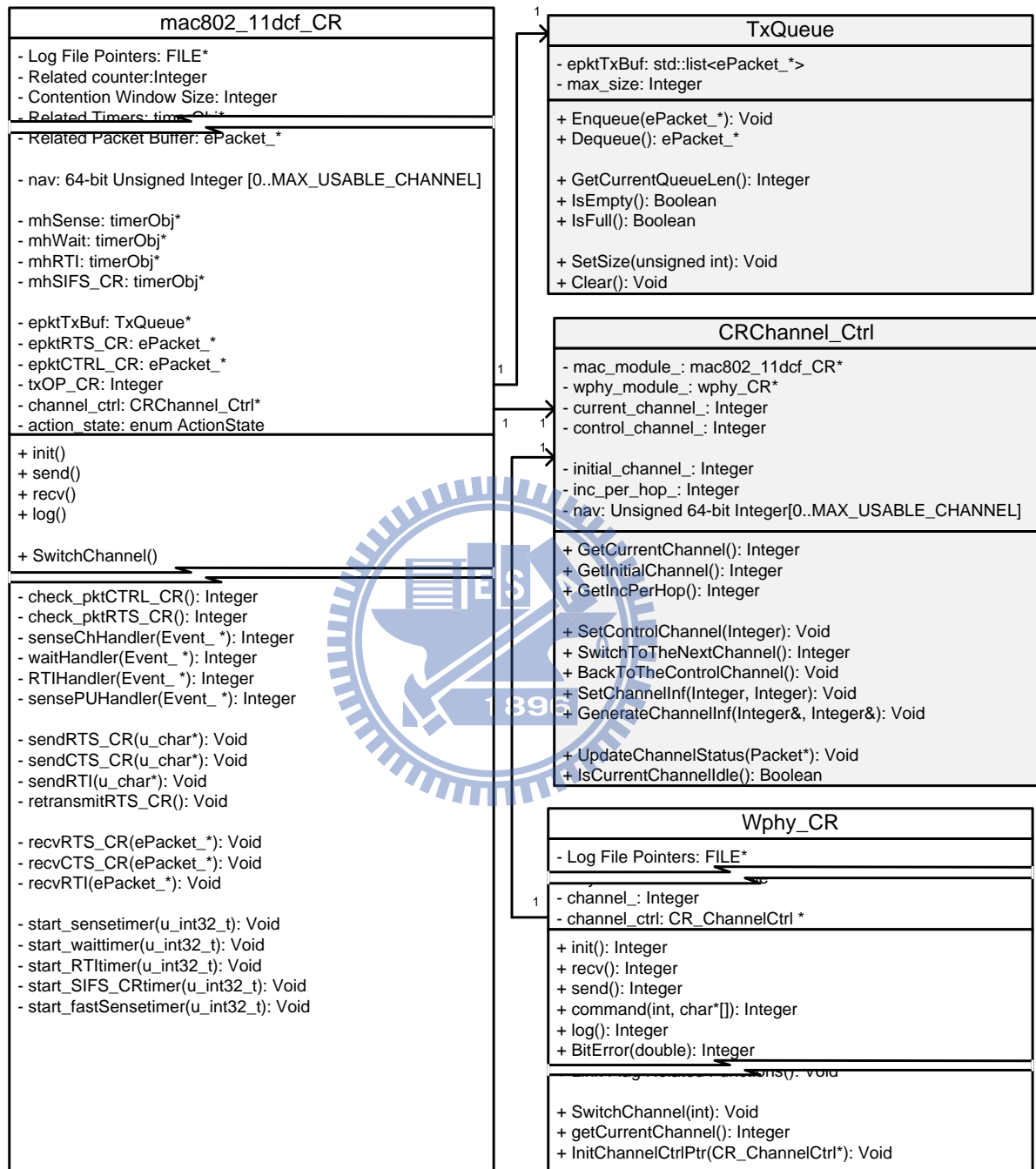


圖 5、本次實作之類別圖。

其中為了簡單呈現，因此在類別成員與類別方法僅詳細列出本次所新增的部份，其餘部份則予以簡化。從中可以看到在修改後，主要是以 mac802_11dcf_CR 做為

MAC80211CR 模組之實作類別，而 Wphy_CR 則為 WphyCR 模組之實作類別。至於其他尚有 CRChannel_Ctrl 與 TxQueue 這兩個類別，第一個類別主要是用來進行頻段操作之相關動作；而第二個類別 TxQueue 則是用來做為連續資料封包的緩衝區維護之類別。這兩個類別將分別在 3.2.3.2 與 3.2.3.3 中詳加描述。

3.2.3.1. 控制封包的新增與行為狀態之維護

● 控制封包的新增

在這次的實作中一共有三個控制封包需要新增，分別為 RTS_{CR} 、 CTS_{CR} 、RTI。針對前二者之封包，可利用原先之 RTS、CTS 封包進行更改。其中，由於 CTS_{CR} 僅用為回應傳送端，使傳送端了解接收端已同意這次的傳輸，因此整個表頭的設計與原本的 CTS 一樣之外， RTS_{CR} 則配合了該無線感知網路的機制而做些許修改，其表頭結構如下：

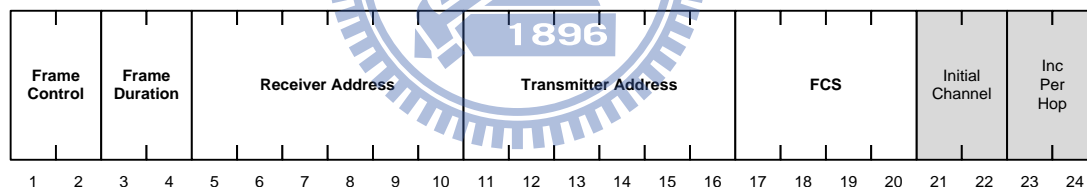


圖 6、 RTS_{CR} 之表頭結構。

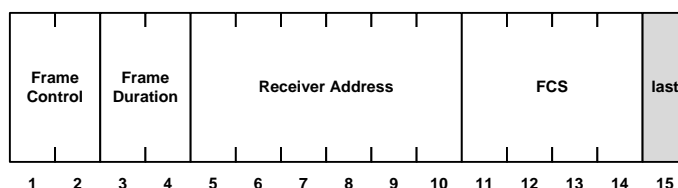


圖 7、RTI 之表頭結構。

修改後的結構仍與 RTS 相似，不同之處在於 RTS_{CR} 需要帶的兩個參數則置於最後面。而除了 RTS_{CR} 與 CTS_{CR} 之外，尚有 RTI 之控制封包。這個封包的部份以 ACK 之控制封包來進行實作，並加上通知接收端是否還有封包要傳送的資訊。

在決定完控制封包的資料結構後，僅需要參考之前處理 RTS、CTS、ACK 的設計，即可將此部份的產生控制封包、傳送控制封包的行為處理完成。對應先前的設計，為了產生與填寫控制封包，在此需要實作 sendRTS_CR、sendCTS_CR 與 sendRTI 這三個填寫表頭欄位之相關的函式，其中 sendRTS_CR 負責產生 RTS_{CR}；sendCTS_CR 則是用以產生 CTS_{CR} 之封包；最後 sendRTI 則是產生 RTI 而使用的。在實作完成後，需將此三個函式填入對應的地方，以使其能在正確的時間點被執行。其中 sendRTS_CR 主要是在當有封包抵達此模組時第一個進入的函式 send 進行呼叫；而 sendCTS_CR 則是在收到 RTS_{CR} 時對應的處理函式 recvRTS_CR 進行呼叫；最後在接收到 ACK 時的處理函式 recvACK 則是檢查是否還有其他封包要傳送與呼叫 sendRTI 函式。

除了填寫表頭欄位之相關函式外，由於這些封包仍需要送出至其他節點，因此需要進行封包傳輸函式之實作。實作的內容包括了：check_pktRTS_CR、check_pktCTRL_CR 這兩個函式，前者主要用來處理 RTS_{CR} 之封包，而後者則是負責傳送 CTS_{CR} 與 RTI 這兩種封包。而這兩個函式需要放在 deferHandler 與 backoffHandler 這兩個計時器逾時處理函式進行呼叫。

最後還需要增加 recvRTS_CR、recvCTS_CR、recvRTI 這三個接收 RTS_{CR}、CTS_{CR} 與 RTI 封包的對應處理函式，這些函式會透過送來的控制封包的內容來更新其狀態與頻段使用決策的相關參數。而被呼叫的位置則是在封包接收的處理函式 recvHandler 內。

● 行為狀態的維護

但單純加入產生封包、送出封包的處理函式是不夠的，這樣的實作很有可能發生狀態錯亂的狀況，原因是在 3.2.3 介紹的部份曾提到 MAC80211CR 並沒有與頻段相關的概念。此外，在先前的設計中，傳送端與接收端的定義主要是由封包類型的傳送優先順序來維護的。在這樣的狀態下，很有可能造成當傳送端與接收端進入資料傳輸頻段時，在頻段感測完成並進入 RTS-CTS 握手機制的狀況下，此時接收端有封包需要進行傳輸，

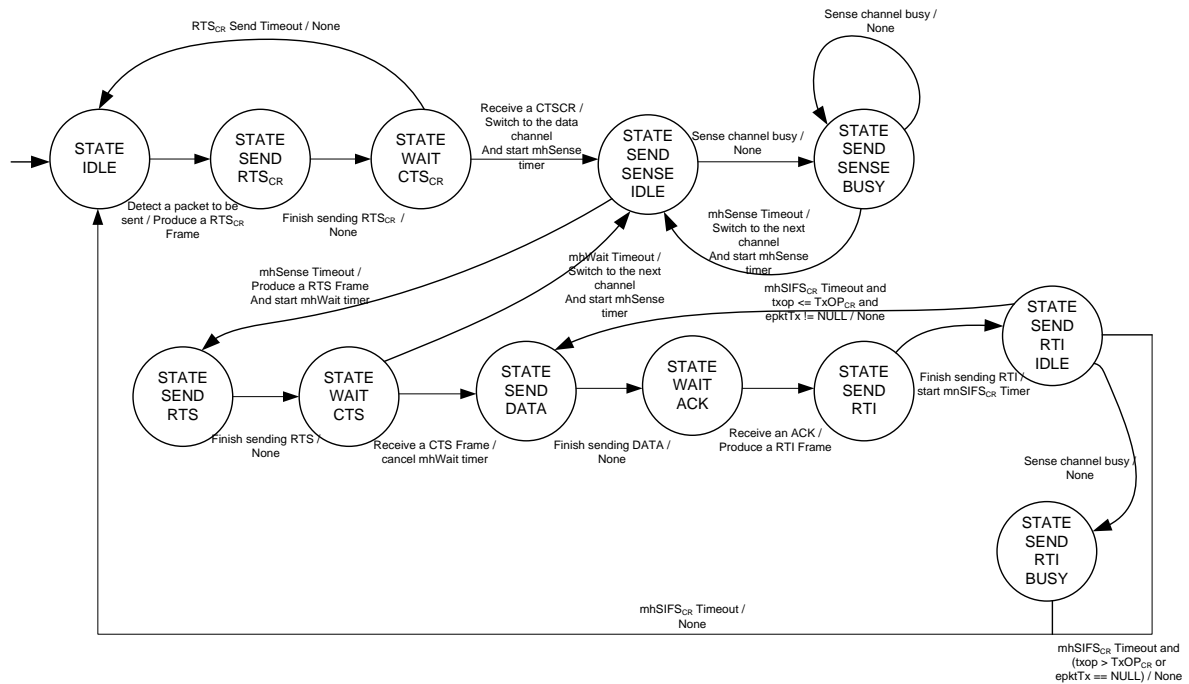


圖 9、接收端之行為狀態循環圖。

〈註：未標示之輸入者表示維持原本狀態，並且無任何輸出。〉

如圖 8、圖 9 所示，在 MAC80211CR 內，透過加入行為狀態的維護，可使得封包傳輸時能夠按照既定的流程進行，而不會產生行為狀態錯亂的狀況。此外也可協助於頻段感測階段進行頻段狀態之記錄，能夠有效的將各種資訊整合以方便設計實作此模組。

3.2.3.2. 頻段感測行為之實作與封包傳輸的特殊狀況的處理

- 頻段的切換與感測機制

在頻段的切換中，由於切換的決定權在 MAC80211CR 模組中，但目前使用之頻段資訊則是由 WphyCR 模組所維護，MAC80211 模組本身對於頻段是沒有概念的。因此在此處，本實作提出了在 MAC80211CR 模組中增加一個稱作 Cognitive Radio Channel Control 的類別 CRChannel_Ctrl，這個類別主要是在 MAC80211CR 模組中產生，負責與

頻段相關的處理事件。但是只有這樣的模組是不夠的，若需要更動目前所使用頻段還需要 WphyCR 模組之支援。故在此增加設定與取得目前相關頻段的函式，使得該類別之物件能直接呼叫相關函式 SwitchChannel 以協助 MAC80211CR 模組進行切換頻段之動作；此外在切換的過程中，為了避免切換頻段後仍在接收前一頻段資料之狀況，因此，在 MAC80211CR 內也另外實作一個函式 SwitchChannel 以處理這些動作。整體設計如圖 5 的 Wphy_CR 與 CRChannel_Ctrl 之類型描述區塊所示。

但在頻段的感測中，目前 NCTUns 對於本部份的實作主要是透過實體層模組來判斷這個封包是否與自己所屬之頻段符合，符合者則將此封包往上層送，如此 MAC 模組即能知道此時存在某些使用者正在使用該頻段；若不符合即直接將該封包丟棄。這樣的狀況在原本的使用情境——即該無線網路使用設備僅在某一特定頻段使用是沒有問題的設計方法，但應用在無線感知網路的情境中卻會導致極大的問題。

而這個問題的解釋主要在於模擬封包接收的方法上，承接上一段的敘述，若接收的封包與自己所在的頻帶吻合，則在 MAC 模組收到後，其流程主要為：

1. 由 MAC 模組的 recv 函式將此封包儲存在接收的緩衝區 epktRx 內，並且利用其封包大小與實體層相關參數來計算其接收時間，並利用此接收時間來啟動接收時所用的計時器 mhRecv。
2. 當 mhRecv 之計時器計時完畢時，此時該封包會透過 recvHandler 這個函式來判斷該封包是否為自己的，是的話即呼叫對應的處理函式；不是的話則在設定完 NAV (Network Allocation Vector) 後丟棄。

在這樣的處理下，由 mhRecv 是否正在倒數計時以及 MAC 的接收狀態即可判斷此時是否能夠感應得到資料的進入，但若考慮圖 10 之情境，這一套設計可能就無法正常運作。

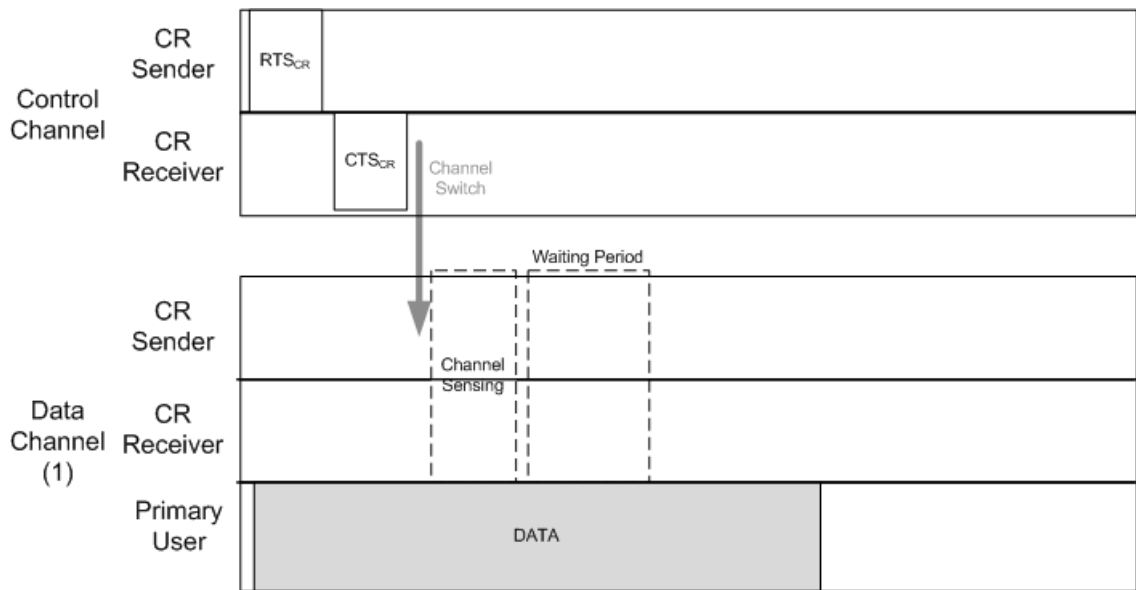


圖 10、不適用於原先 NCTUns 感測頻段忙碌流程之實例。

在圖 10 中，一開始無線感知網路的使用者正在控制封包傳輸頻段中進行封包的傳輸，而在同時，第一個資料傳輸頻段也有一個主要使用者正在傳送資料。假設此情境下，這三位使用者皆能完整接收到其它使用者所傳輸之封包，則在該時間點，無線感知網路之使用者均會收到由第一個資料傳輸頻段傳送來的封包，但是在實體層卻會因為它不是屬於目前使用之頻段的封包為理由而將其丟棄。之後，兩位無線感知網路之使用者切換至第一個資料傳輸頻段，在他們進行頻段感測時卻無法知道目前有一位主要使用者正在使用該頻段，因為該位使用者傳來的封包已被丟棄。

為了解決這樣的問題，最簡單的方式是改寫 WphyCR 模組，讓其完全不用判斷所用頻段，直接將其往上層送，但是這樣即不符合 MAC 的行為定義。因此在此處的設計主要是透過 CRChannel_Ctrl 來實作一個稱作 UpdateChannelStatus 的函式，任何進入 WphyCR 模組的封包都會被交由該函式處理，而該函式主要會計算出若要接收這個封包，則至少要到哪個時間點才能被接收完畢，並將該時間點儲存於該類別中負責儲存對應頻段的陣列中。此時若 MAC80211CR 模組需要檢查現在頻段是否為閒置狀態，則可直接透過 CRChannel_Ctrl 這個類別的物件來進行查詢，若現在時間比所記錄的時間小，則代

表此時仍有封包在傳輸。因此，此機制搭配原先儲存在 MAC80211CR 的 NAV 的機制在多重頻段上的改寫，即可正確判斷出目前頻段是否為忙碌狀態。

● 頻段感測時間區間之實作

在頻段由控制封包傳輸頻段切換至資料傳輸頻段後，根據[10]所提的機制，無線感測網路之傳送端與接收端雙方均要進行頻段感測之動作。若此時頻段為忙碌，則接下來應該要切換至下一個資料傳輸頻段；但若此時頻段為空閒，則傳輸可順利繼續進行。

在此部份的實作中，主要行為則是當接收端回應 CTS_{CR} 完成與傳送端收到 CTS_{CR} 後，雙方會切換至約定好的資料傳輸頻段。此時雙方即啟動一個稱作 mhSense 的計時器，並更動行為狀態為 STATE_SEND_SENSE_IDLE 或 STATE_WAIT_SENSE_IDLE，此外也會將 MAC80211CR 內用以記錄接收狀況之變數設為 MAC_SENSE 以代表它們將進入感測狀態。在此同時，MAC 模組會先檢查這個頻段是否為閒置狀態，若不是則會將行為狀態會更動為 STATE_SEND_SENSE_BUSY 或 STATE_WAIT_SENSE_BUSY。而計時器完成計時前，每一個傳送至 recv 函式的封包即會被丟棄，並且行為狀態會更動為 STATE_SEND_SENSE_BUSY 或 STATE_WAIT_SENSE_BUSY。

當 mhSense 這個計時器完成計時後，則會呼叫一個稱為 senseChHandler 的函式。這個函式會檢查目前的行為狀態是否為 IDLE，若是則會讓傳送端在此時填入 RTS 控制封包，並將行為狀態更動為 STATE_SEND_RTS 或 STATE_WAIT_RTS，使得後續的 RTS-CTS 握手動作能順利完成；若不是，則只單純將行為狀態更動為 STATE_SEND_RTS 或 STATE_WAIT_RTS。而在以上動作完成後，一個稱作 mhWait 的計時器會被啟動，若此計時器計時完成，RTS-CTS 握手動作卻沒有完成，則代表在這個頻段上的握手動作失敗，需要到下一個頻段重新進行頻段感測與 RTS-CTS 握手動作。

● 於接收時例外情況的處理

在封包的接收方面，為了避免發生在資料傳輸頻帶收到並非給自己的封包，以及收到並非本次傳輸時的傳輸對象之封包，因此在進行 RTS_{CR} 與 CTS_{CR} 的握手動作時，會另外將對方端的 MAC 位址記錄下來。若在接收完封包時，發現此封包並非本次傳送對象的狀況，則該位無線感知網路之使用者會直接將目前使用頻段切換至控制封包傳輸頻段。而若此狀況只有傳輸與接收兩端的其中一端知道，則按照原始機制，會使得原本的流程無法順利進行。

考慮到原本流程無法順利進行的狀況，例如：傳輸尚未完成，在接收端發生碰撞之狀況、沒有收到某些特定封包，如 ACK、RTI 等狀況，在實作的部份均是讓發生端在該時段回到控制封包傳輸頻段，使得其能重新進行封包的傳輸動作。

3.2.3.3. 主動釋出頻段之設計

而在最後的設計則是在進行完原始的資料傳輸程序，即 $RTS-CTS-DATA-ACK$ 之後，在 3.2.3.1 曾提到 RTI 這個表頭的設計。為了讓接收端知道在等待完 $SIFS_{CR}$ 之後是否仍要繼續在資料傳輸頻段繼續等待下一個資料，因此在 RTI 的設計上則是加入了 last 這一個欄位來表示這個資料是否為最後一個傳輸的資料，以及在這個 RTI 結束後，是否要返回控制封包傳輸頻段。為了能讓 RTI 內的欄位填入正確的值，因此在收到 ACK 之後，接收端需要在此時檢查他是否還有資料要傳、以及是否還有能夠運用此頻段的機會（即本次在此頻段的傳輸次數是否小於 $TxOP_{CR}$ 的定義值）。

而傳送端傳送完 RTI 與接收端在收到 RTI 後，如同在 3.2.3.2 提到的頻段感測之機制，兩者均需設定其接收狀態為 MAC_SENSE ，並將行為狀態設為 $STATE_SEND_RTI_IDLE$ 與 $STATE_WAIT_RTI_IDLE$ 以表示它們兩位使用者進正在進行頻段感測，並啟動 $SIFS_{CR}$ 的計時器。之後若感測到頻段有人在使用時，則將狀態切換為 $STATE_SEND_RTI_BUSY$ 與 $STATE_WAIT_RTI_BUSY$ 。

當 $SIFS_{CR}$ 的計時器逾時，即需要針對目前的狀態以及還有沒有資料封包需要傳輸。若目前的狀態為 $STATE_SEND_RTI_IDLE$ 或 $STATE_RECV_RTI_IDLE$ ，且在 RTI 封包中註明還有封包需要傳輸，則雙方繼續待在目前的資料傳輸頻道內；否則應該要回到控制封包傳輸頻帶再次決定應該要使用到哪個資料傳輸頻段進行後續的資料傳輸。

此外，為了避免只有傳送端發現有主要使用者想使用此資料傳輸頻帶之狀況，接收端在繼續待在此頻段時需要啟動 $mhWait$ 之計時器，並在接收到任意封包時取消此計時器的計時動作；若在計時器逾期前仍未取消使用此計時器，則表示傳送端可能偵測到主要使用者之訊號，並跳回控制封包傳輸頻段，故此時接收端應回到控制封包傳輸頻段進行等待或競爭傳輸之機會。而在此為了避免等待時間太短而造成接收端過早回到控制封包傳輸頻段，因此在此對於時間的設定與完成頻段感測時的等待時間區間相同。

在完成這些機制設計後， $MAC80211CR$ 與 $WphyCR$ 的模組即告完成，而模擬也能夠順利進行。但是，在進行模擬實驗的同時，也發現了在此機制下可能遭遇的某些問題使其效能降低，或是存在某些使其效能提升之可能機制，而這些議題的討論與其改良方案將於下一章中描述。



4. 原有機制之改良

4.1. 原有機制之效能增討論

在前一章中進行[10]之無線感知網路機制之設計時，發現有幾個部份的設計中存在著對於某些情境下效能的改良點。這些改良的部份根據其位於流程中位置來分類，可概略性分為三個部份：

- 快速頻段感測之頻段挑選機制

在 3.1.2 中曾提到快速頻段感測 (Fast Channel Sensing) 的機制，這個機制在實作上主要是採取亂數來做為決定初始頻段 (Initial Channel)，並且任取一個與可用資料頻段個數互值之數 h 來做為頻段切換遞增值。這樣的設計的好處是容易實作且在決策上較不耗費時間，但考慮到其取得之初始頻段正有使用者進行傳輸時，它將損失頻段感測所需時間外加等待時間區間的時間。如果很不湊巧的在切換頻段之後，仍舊感測到存在任意使用者正在使用該頻段，還是會造成該段時間的損失。因此，在這種情況下，若能夠找到一種方式來進行使用資料頻段的挑選會是較佳的解決方案。

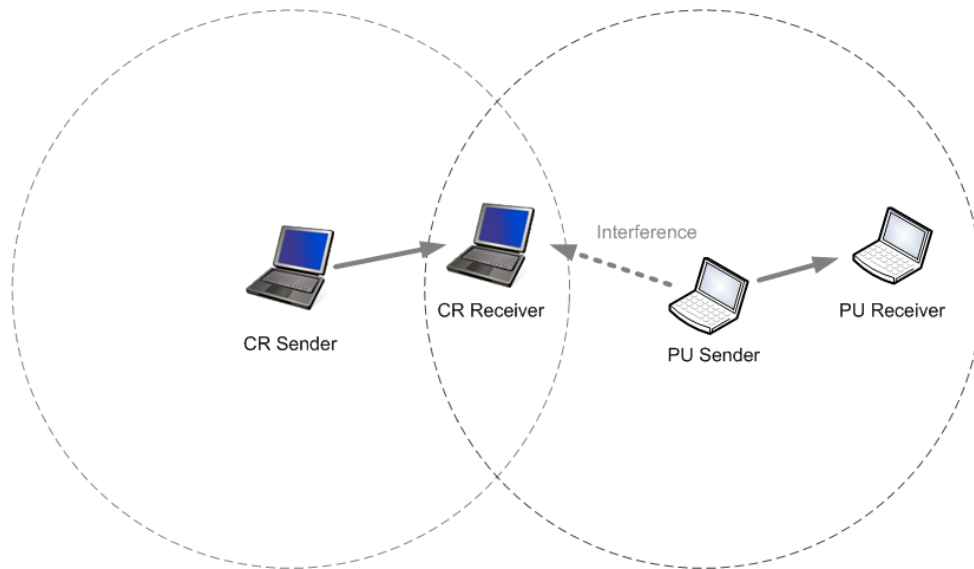


圖 11、由終端機隱匿問題之情境來考慮決策者的選定。

此外，在此機制下由於決策者為無線感知網路傳送端，若此時存在如圖 11 之終端機隱匿問題 (Hidden Terminal Problem)，假設目前已存在一種機制能使得傳送端能找到最適合自己的頻段，但由於傳送端無法知道接收端是否會被其他使用者干擾，因此傳送端仍舊無法保證接收者能夠順利接收到 RTS 控制封包，進而達到減少感測到頻段切換之損失。

因此，在解決方案的部分將設計兩個機制：第一個機制為建構於歷史感測資料之評分機制，透過此機制可以協助無線感知網路使用者挑選適合自己的頻段；第二個機制則是改寫 RTS_{CR} 與 CTS_{CR} 的設計，使得傳送端可以將「較適合自己使用」的頻段透過 RTS_{CR} 告知接收端，而接收端在此時會啟動「頻段感測快照 (Channel Sensing Snapshot)」的機制，使得接收端能在此時先了解目前周遭頻段使用情況，以決定頻段切換順序。此部份更進一步的設計說明將會在 4.2.1 中詳述。

● 資料傳輸頻段感測之機制

接下來則是探討在於頻段感測機制中對於資料傳輸感測頻段的部份，在這個部份中的設計中，原本的設計主要是在切換至任何資料傳輸頻段時需要進行頻段感測，以確定

是否存在主要使用者欲使用該頻段之狀況；若此時感測到任何訊息，則代表此時可能存在某些使用者欲使用此頻段。這樣的機制的設計主要是希望能儘量避免干擾主要使用者使用此頻段，不過在某些情境下它可能會降低無線感知網路使用者在進行傳輸時的效能。

情境如下圖所示：

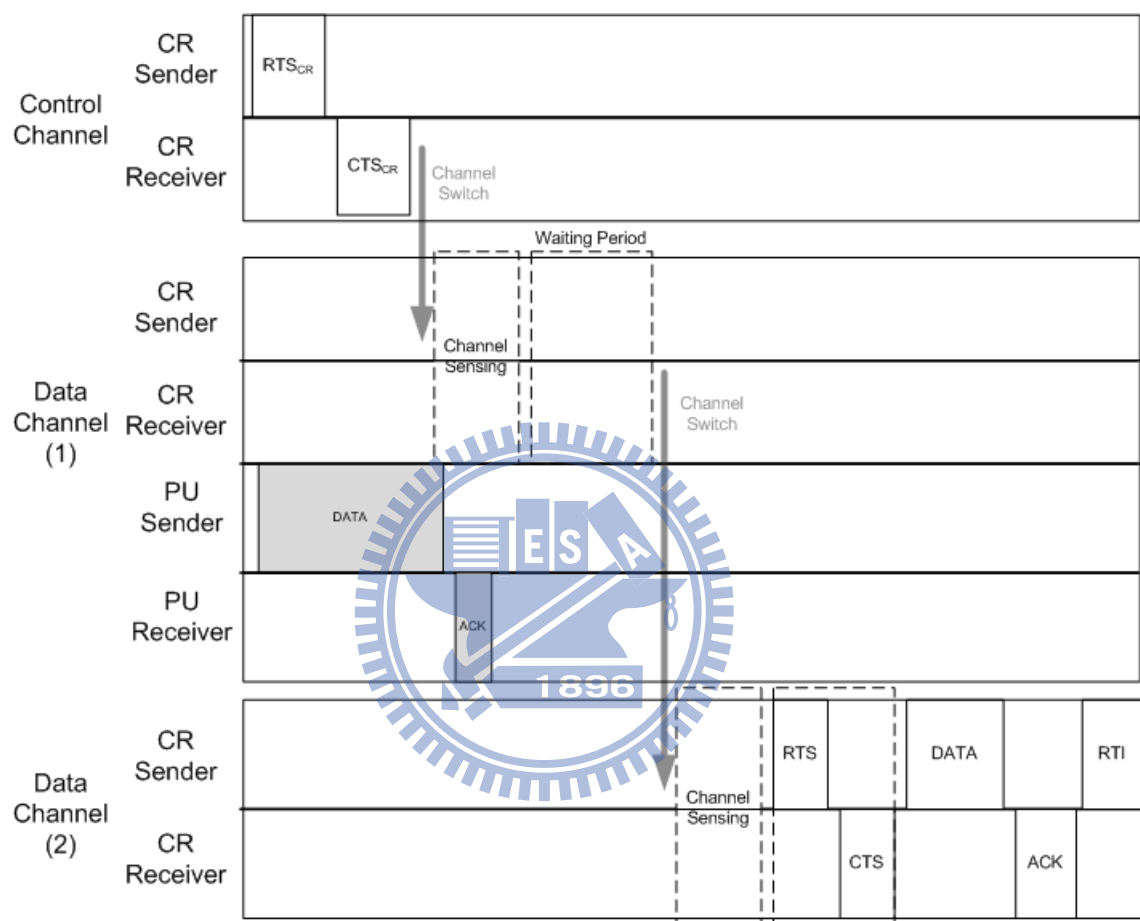


圖 12、當頻段感測到 ACK 後，即使頻段接下來無任何使用者使用，仍需切換頻段。

圖 12 展示了一種特殊的情境，在這種情境下，當無線感知網路使用者切換到資料傳輸頻段 1 時，此時存在一對主要使用者正在進行資料傳輸，並將進入尾聲。此時主要使用者中的接收端因為收到該資料後回傳 ACK 給傳送端，兩者在此後短時間內不再傳輸封包。但是在這個時候，無線感測網路之使用者卻感測到附近正有其它使用者在進行資料傳輸，於是在頻段感測完成且等完等待時間區間後隨即會放棄在此頻段的使用情況，直接切換至下一頻段。在此種情境下，無線感知網路的兩位使用者仍然損失了頻段感測

所需時間外加等待時間區間的時間，因此在此部份的機制是可以考慮進行改良的。

而在改良的方案主要有一個假設，這個假設即是在這些頻段內的使用者均為 IEEE 802.11 無線網路之使用者，亦即每一位使用者所傳的封包型態是所有使用者都知道的。因此，在此情況下，只需要針對頻段感測時所收到的封包進行型態的檢查，並且進行對應的處理即可。詳細的內容將在 4.2.2 中詳細描述。

● SIFS_{CR} 之後的資料傳遞與終端機隱匿問題之討論

最後則是在 SIFS_{CR} 之後關於資料傳輸方面的議題，這一部份的議題在原始的機制中主要是：在本次資料封包傳輸完成後，無線感知網路接收端會回應一個 ACK 以代表收到此資料傳輸封包，之後無線感知網路傳送端會發出一個稱作 RTI 的控制封包。在接下來 SIFS_{CR} 的時間內，雙方均需要進行頻段的感測，以確定此時是否存在主要使用者需要進行資料傳輸。

考慮到先前於圖 11 的終端隱匿問題之情境，在主要使用者的訊號只有無線感知網路接收者能夠感知的情況下，若此時確實存在任意主要使用者需要進行封包的傳輸，則該位使用者會發出 RTS 之控制封包。在此時此 RTS 只有無線感知網路接收端能夠感測到，傳送端則是維持認定「此頻段尚無其它使用者使用」之狀態。這樣的情境下，雖然在無線感測網路接收端返回控制資料傳輸頻段後，當無線感測網路傳送端送完資料封包卻沒有收到 ACK 控制封包時，傳送端能夠正常的返回控制資料傳輸頻段。整體流程可參考圖 13。

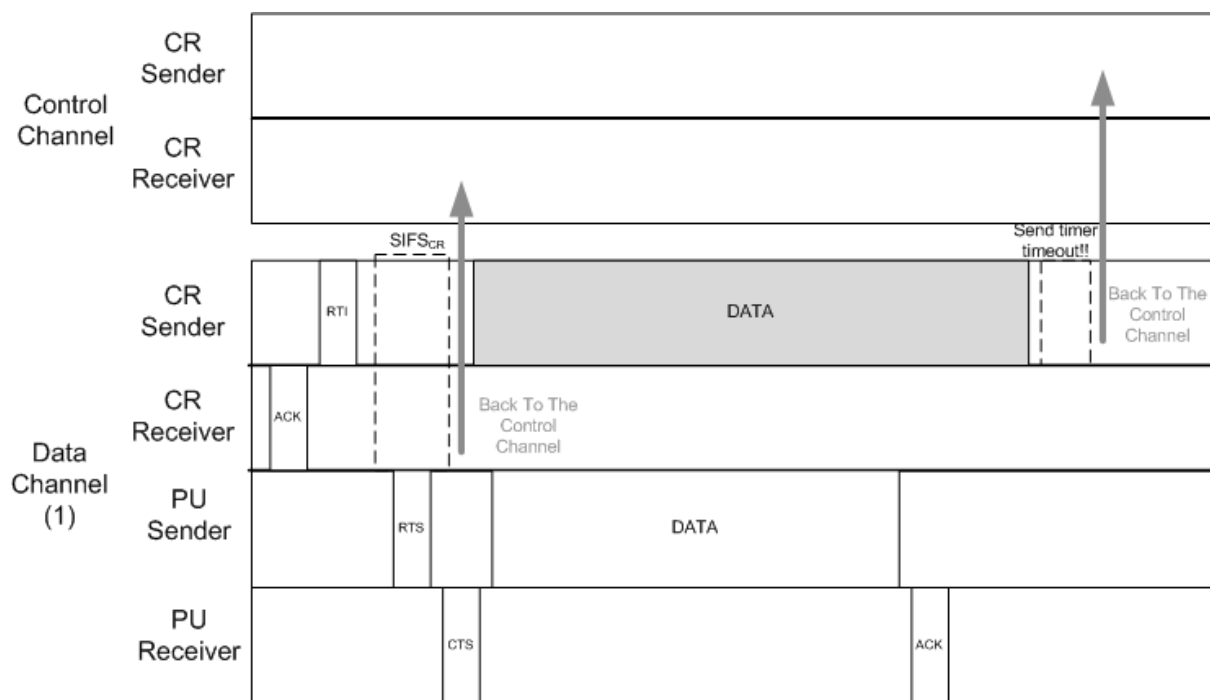


圖 13、在終端隱匿問題下，傳送端的行為反應。

現在假設資料封包的內容大小為 2048 bytes，且無線感知網路傳送端位於資料傳輸率上限為 2 Mbps 的環境中，則傳送端至少需要約 8000 微秒的時間進行此資料封包之傳輸，而且必定是在這之後才有機會知道接收端無法接收此資料。在如此之情境下，其返回控制封包所在頻段之代價相較之下似乎有點過高，因此若能在此更動其機制，勢必有機會能讓無線感測網路之使用者其效能提升，達到更有效的頻段使用率。而在改善方案的規劃上，本篇論文之設計主要是再次進行 RTS-CTS 握手動作為主，詳細設計於 4.2.3 中將詳細說明。

4.2. 改進之方法與其設計實作

4.2.1. 頻段挑選機制

在頻段挑選過程中一共有兩個機制，第一個機制稱為頻段評分機制，第二個則稱作頻段感測快照機制。底下將分別針對此二機制進行詳細的設計說明與實作方式：

● 頻段評分機制

在頻段評分機制的設計上主要是運用一種基於歷史資料觀測的方式來進行評分，在歷史資料觀測的部份，對於同一無線感知網路使用者，針對所使用的每一個頻段各有一組 32 位元稱作閒置記錄的記錄值，在每次的頻段感測中，皆會更新此記錄值。而在實作中，選擇最大 bit 數為最近一次的偵測結果，在每次進行偵測時，會先將此數值往右遞移一位元，之後當偵測到頻段為閒置狀態時即標示為 1，而偵測到忙錄狀態時則為 0。如圖 14 所示：

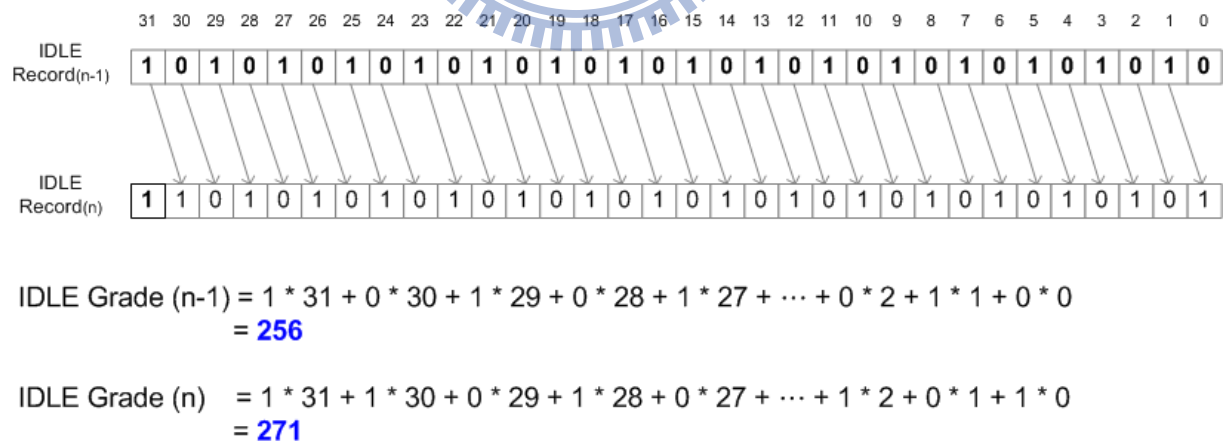


圖 14、閒置記錄與評分方式之示意圖。

圖 13 展示了一個例子，在某一個時間點，閒置記錄的記錄內容如圖中的 *IDLE Record(n-1)* 所示，內容為由 1010...10 循序組成，而在下一個時間點，該無線感知網路

使用者偵測到此頻段處於閒置狀態，則其會將該記錄往右移 1 位元，並且在最高位元的值填入 1，即如圖 14 中 *IDLE Record(n)* 所示。

而在進行評分值的方面，在此給與閒置記錄值的每 1 bit 一個權重，由最高的 bit 開始為 31，依序遞減。計算方式即是將所有的位元與其權重相乘後加總，即可得到一個值。以圖 14 為例，由 *IDLE Record(n-1)* 可以得到其分數為 256，而 *IDLE Record(n)* 則為 271。若其值愈大，則代表其具有在愈近的時間點具有較多的閒置機會，或是在歷史記錄中其閒置次數頻繁的特質。這樣的設計可以避免只有最後一個位元來決定評分的狀況。因此在此處主要將利用這個評分標準來進行閒置頻段的評估。

而在頻段選定的部份則交由無線感知網路傳送端按照評分的分數由高到低進行評分，並從中挑選部份頻段填入 RTS_{CR} 的控制封包內。為支援傳遞這些訊息， RTS_{CR} 的控制封包需要進行改變，改變的內容主要是將原先儲存於原機制的欄位所佔用的 4 個位元組改為儲存被選中的頻段。儲存方式主要是利用該欄位中一共 16 個位元分別代表不同的頻段已挑選或未挑選。而表頭的部份則如圖 15 所示：

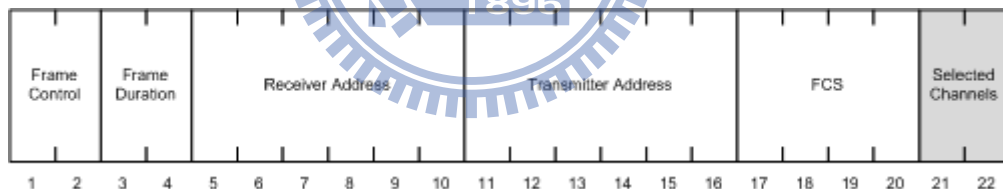


圖 15、修改後的 RTS_{CR} 之結構。

當無線感知網路使用者之接收端收到此 RTS_{CR} 的控制封包後，接收端即能知道傳送端選擇那些頻段，並且也能從這些頻段中挑選適合本次傳輸之頻段。而挑選的方式除了透過這套評分方式之外，尚還有在下個段落會提及的頻段感測快照機制。在挑選完畢之後，接收端會將接下來想要使用的頻段按照順序填入 CTS_{CR} 的控制封包內，以告知傳送端接下來的資料頻段切換順序，如此則可由接收端決定較有可能為空閒之頻段。而修改後的 CTS_{CR} 的結構則如圖 16 所示：

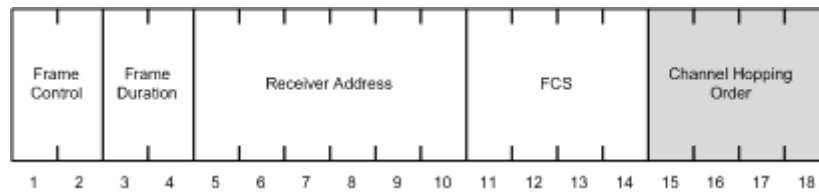


圖 16、修改後的 CTS_{CR} 之結構。

● 頻段感測快照機制

在上一段的無線感知網路接收端決定頻段切換順序時曾提到還需要配合頻段感測快照機制的使用，這個機制主要是為了讓感測者能夠以較短的時間為代價，藉此取得某些頻段的頻段使用情況。

當接收端收到 RTS_{CR} 之後，接收端會知道哪些頻段是傳送端所選擇的。此時接收端會對這些頻段進行快速的頻段感測。對於每個頻段，其感測時間為 100 微秒，並在這 100 微秒的時間結束時更新閒置記錄值，並切換到下一頻段。在此若該頻段有任何封包傳輸，則直接將該頻段標示為有其他使用者正在進行傳輸的情況；而未偵測到任何封包，則標示其為閒置狀態。藉由此方法，能夠讓接收端在短時間內對於可能用到的頻段進行其最近情況的分析，使得接收端判斷所用之資訊能夠更為準確。

4.2.2. 資料傳輸頻段感測機制之改良

在 4.1 的圖 12 中提到對於資料頻段感測時的某種特殊情境，這個情境主要發生在頻段感測時若接收到 ACK，且在此 ACK 之後，主要使用者在短期間內均不使用此頻段。若此時因為此 ACK 而導致頻段使用上的浪費則不太划算，因此在改善方案的部份則是著重在觀看封包內容並判斷接下來是否該使用。

根據各個接收到封包的意義，當收到一個 RTS 封包、CTS 封包或 DATA 封包，則代表接下來直到接收到一個 ACK 之前都不能使用此頻段。因此此處主要是利用頻段感

測時的最後一個封包來決定是否可以使用此頻段。但是封包的接收是需要時間的，若存在一個封包來不及在頻段感測時間結束前被接收完畢，則為了避免誤判的狀況，因此最後仍然不應該使用此頻段。根據以上的分析，可將決策以底下的圖來表示：

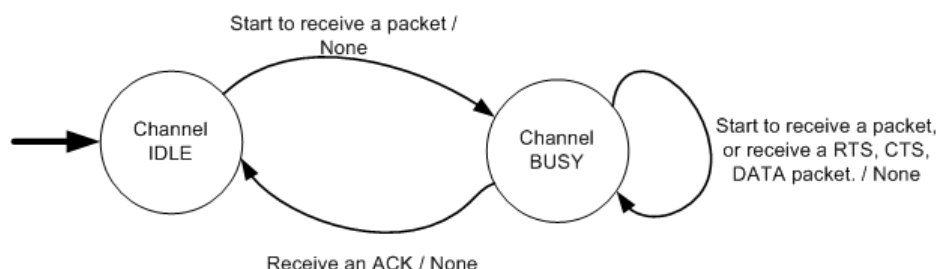


圖 17、修改後的頻段感測之決策機制。

4.2.3. SIFS_{CR} 後資料傳輸之改良

在 4.1 中，圖 13 顯示了在不考慮接收端的情況下，當 SIFS_{CR} 的等待時間過後可能會因為隱匿終端問題而導致傳送端需要不少的時間才能返回控制封包傳輸頻段的情境。而在解決方案的部份則是提出了「再次於同一頻段進行 RTS-CTS 握手動作」的方案做為解決方法。

而這樣的方案主要是為了能讓傳送端迅速察覺接收端出現異常，原因主要是因為這樣的程序所需要的時間約為 600 微秒，對比於先前提到在 DATA 為 2048 bytes 的情況下，如此可大幅縮小反應的時間；此外，利用 RTS 與 CTS 的優點也包括了能讓其他使用者知道這個頻段目前有其他使用者正在進行傳輸，使得本次的傳輸不會因為有其他使用者干擾而中斷，搭配原有機制在 RTI 的處理，將會使得有意願的使用者在這段時間之後能夠表達它們有使用此頻段的需求，使得整個無線感知網路的傳輸程序能順利完成，也能降低對於主要使用者的干擾程度。相信這樣的改良機制能使得無線感知網路使用者在此情境下的效能能夠提升。

在實作上，則只要在等待完 $SIFS_{CR}$ 的處理函式中，針對傳送端的部份加入產生 RTS 控制封包的呼叫函式，並且讓傳送端與接收端各自將其行為狀態更動為 $STATE_SEND_RTS$ 與 $STATE_WAIT_RTS$ ，如此在接下來的流程中，整個機制即會按照改良後的流程運作。

到此，對於整個機制的改良已告一段落，在接下來的章節中，本篇論文將呈現以原本的方法與改良後的方法在不同情境下的效能比較並進行分析。



5. 效能評估與分析

在本章節中將展示於本篇論文中所設計的無線感知網路模組與其經改良後之模組在效能上的比較。首先在 5.1 中，本篇論文將先以理想之情境來進行理論值的推導；而接下來，在情境的部份則將展示兩種情境，分別為 5.2 的無終端機隱匿問題的模擬情境，在此情境中，使用者彼此都在所有其他使用者的訊號傳輸干擾範圍內；第二個則是 5.3 的終端隱匿之情境，這個情境則已於 4.1 的圖 11 描述過，但此情境主要著重於多無線感知網路用戶的情境，而非僅有單一用戶的狀況。

而針對各個情境，在此主要使用者均為 IEEE 802.11b Ad-hoc 網路之使用者，其資料傳輸上主要是以 Poisson 機率分佈的封包產生間隔來產生封包，此封包主要是從 MAC 層產生的單純測試用之封包，而在長度方面，每個資料封包不包括 IEEE 802.11b 的表頭，其長度均為 2048 bytes，而所產生的資料傳輸率平均為 0.8 Mbps。在無線感知網路使用者的方面，則是存在兩種封包產生方法：第一種類型的流量測試樣式主要是產生封包到達率為 1 的封包，而這類型封包的其他參數與主要使用者使用的均相同，其封包長度在不包括 IEEE 802.11b 表頭的情況下，長度為 2048 bytes。在這一類型的情境下將可看出主要使用者是否會嚴重的受到無線感知網路使用者的影響。第二種則是透過真實世界的應用程式產生封包，這支應用程式主要稱作 stg，它可以協助我們產生任意傳輸樣式的 TCP 與 UDP 封包。在本章節中將使用其來不斷產生 1500 bytes 之 TCP 封包，藉此觀察在此無線感知網路的機制下，是否適合使用 TCP 的應用程式？

而在測試的情境中，除了拓樸不同的狀況之外，尚有頻段感測時間、快速頻段感測快照時間、RTS-CTS 握手等待時間等。底下為本次效能評估中所用之參數：

參數名稱	設定值
資料傳輸頻段總數 ($N_{DataChannel}$)	5
各頻段之最大資料傳輸率($Mbps$)	2
頻段感測時間 (μs)	2000
快速頻段感測快照時間 (μs)	100
$SIFS_{CR}$ (μs)	100 ($10 * SIFS$)
RTS-CTS 握手等待時間 (μs)	RTS 傳輸時間 + CTS 傳輸時間 + $12 * SIFS$
無線網路訊號傳輸模型	Two-Ray Ground

表 1、於效能評估中的參數設定。

5.1. 理論值之推導與估計

在效能評估前，本章節將先推導流量的理論值，藉此來進行模擬結果的驗證。而在本處理論值的推導上將著重於本章中接下來將提出的兩種流量產生樣式，這個情境則如下所示：

- 主要使用者 (於兩種流量中均固定)
 - 資料產生之模組：由 MAC 產生一般之資料，單向傳輸。
 - 資料大小：2048 bytes
 - 流量產生方式：封包產生時間間隔為 Poisson 機率分布，佔用 40% 的頻寬。
- 無線感知網路之使用者
 - 樣式一

- ◆ 資料產生之模組：由 MAC 產生一般之資料，單向傳輸。

- ◆ 資料大小：2048 bytes (MAC Payload)

- ◆ 流量產生方式：隨時均有資料產生。

- 樣式二

- ◆ 資料產生之模組：應用程式以 TCP 通訊協定進行資料交換。

- ◆ 資料大小：1474 bytes (TCP Payload)

- ◆ 流量產生方式：隨時均有資料產生。

而在推導式子前，為了簡化整體的狀況，使得推導內容能夠簡單易懂，因此在此進行底下的假設：

1. 假設所有控制封包均為相同大小，在此處以 ACK 為基準。

2. TCP 與 IP 表頭均無額外之欄位，亦及此二表頭長度為 20 bytes。

3. 推導環境

- i、 假設在此環境內，所有使用設備均可收到來自於所有其它設備傳遞之訊號。

- ii、 在整個情境中，所有傳輸均成功送達接收端，不會受到其他外在因素而產生碰撞或封包遺失等事件。

- iii、 假設每個頻段在其進入閒置狀況時，隨即會有一位使用者來進行使用。

- iv、 主要使用者均可達到其預估之平均流量值，即 40% 的頻寬。

在定義完流量產生樣式與推導之假設後，在此將開始針對兩種流量樣式來進行理論

流量之推導。推導流程在一開始會先著重在主要使用者為達到 40% 頻寬的傳輸量，其將會佔用 1 秒內多少時間的推算；接下來透過這個時間，將可以推論無線感知網路使用者在這 1 秒的時間內還有多少時間可供其傳送資料；最後即可推導出其理論上的流量值。

● 過程中所使用之參數定義與計算

■ 控制封包傳輸所需時間 (T_{RTS} , T_{CTS} , T_{ACK} , T_{RTT}) :

LEN_{ACK} : IEEE 802.11 ACK 之表頭長度 (112 bits)

Bandwidth : 頻段最大可傳輸之資料率，即 2 Mbps。

$$T_{PREMABLE} = 144 \mu s$$

$$T_{PCLP} = 48 \mu s$$

$$\begin{aligned} T_{ACK} &= \text{Ceiling}(LEN_{ACK} / \text{Bandwidth} * 10^6 + T_{PREMABLE} + T_{PCLP}) \\ &= \text{Ceiling}((112 / 10^6) / 2 * 10^6 + 144 + 48) \\ &= 248 (\mu s) \end{aligned}$$

■ MAC Data 2048 bytes 資料傳輸所需時間 (T_{DATA})

$$LEN_{DATA} = LEN_{MAC-Payload} + LEN_{MAC-Header}$$

$$= 2048 + 28$$

$$= 2076 \text{ (bytes)}$$

$$= 16608 \text{ (bits)}$$

$$T_{DATA} = \text{Ceiling}(LEN_{DATA} / \text{Bandwidth} * 10^6) + T_{PREMABLE} + T_{PCLP}$$

$$= \text{Ceiling}((16608 / 10^6) / 2 * 10^6) + 144 + 48$$

$$= 8496 (\mu s)$$

- TCP Payload 1474 bytes 資料傳輸所需時間 ($T_{TCP-DATA}$)

$$LEN_{TCP-DATA} = LEN_{TCP-Payload} + LEN_{TCP-Header} + LEN_{IP-Header} + LEN_{MAC-Header}$$

$$= 1474 + 20 + 20 + 28$$

$$= 1542 (\text{bytes})$$

$$= 12336 (\text{bits})$$

$$T_{TCP-DATA} = \text{Ceiling}(LEN_{TCP-DATA} / \text{Bandwidth} * 10^6) + T_{PREAMBLE} + T_{PCLP}$$

$$= \text{Ceiling}((12336 / 10^6) / 2 * 10^6) + 144 + 48$$

$$= 6360 (\mu s)$$

- TCP ACK 資料傳輸所需時間 ($T_{TCP-ACK}$)

$$LEN_{TCP-ACK} = LEN_{TCP-Header} + LEN_{IP-Header} + LEN_{MAC-Header}$$

$$= 20 + 20 + 28$$

$$= 68 (\text{bytes})$$

$$= 544 (\text{bits})$$

$$T_{TCP-ACK} = \text{Ceiling}(LEN_{TCP-ACK} / \text{Bandwidth} * 10^6) + T_{PREAMBLE} + T_{PCLP}$$

$$= \text{Ceiling}(544 / 10^6) / 2 * 10^6 + 144 + 48$$

$$= 464 (\mu s)$$

- 主要使用者為達到 40% 的頻寬之流量所需之時間 (T_{PU})

透過流量的換算，可以得到若 1 秒內要達到 40% 的資料傳輸，需要傳送的總資料量

為：

$$\begin{aligned} \text{LEN}_{\text{Total_DATA}} &= 0.4 \text{ (sec)} * \text{Bandwidth (Mbps)} \\ &= 0.4 * 2 \\ &= 0.8 \text{ (Mbits)} \\ &= 800000 \text{ (bits)} \end{aligned}$$

因此，在此可以得到於這 1 秒內，主要使用者需要傳送多少個 DATA Frame 才能達到此流量：

$$\begin{aligned} N_{\text{DATA_Frame}} &= \text{Ceiling}(\text{LEN}_{\text{Total_DATA}} / \text{LEN}_{\text{MAC_Payload}}) \\ &= \text{Ceiling}(800000 / 2048 / 8) \\ &= 49 \end{aligned}$$

由上面的 $N_{\text{DATA_Frame}}$ 可以得到主要使用者在 1 秒內至少需要傳送 49 個 DATA Frame 才能達到此流量，但是在一般 IEEE 802.11 中的 MAC 封包傳輸程序中不考慮 RTS Threshold 的狀況下，它本身是需要遵守 RTS, CTS, DATA, ACK 的傳輸程序的。因此為了傳送這樣的資料量，主要使用者需要使用的時間 T_{PU} 如底下式子所示：

$$\begin{aligned} T_{\text{PU}} &= (\text{sifs} + T_{\text{RTS}} + \text{sifs} + T_{\text{CTS}} + \text{difs} + T_{\text{DATA}} + \text{sifs} + T_{\text{ACK}}) * N_{\text{DATA_Frame}} \\ &= (10 + 248 + 10 + 248 + 20 + 8496 + 10 + 248) * 49 \\ &= 9290 * 49 \\ &= 455210 \text{ (}\mu\text{s)} \end{aligned}$$



故從中我們可以得到主要使用者需要使用 454720 μs ，才能使其資料傳輸量到達總資料頻寬的 40%；而這也代表了無線感知網路的使用者在在此 1 秒內於這個頻段可以使用的總時間(T_{CR})為：

$$\begin{aligned} T_{CR} &= 1000000 - 454720 \\ &= 544790 (\mu\text{s}) \end{aligned}$$

● 無線感知網路使用者在傳輸樣式一的流量理論值 (TH_{CR1})

在得到無線感知網路使用者在此頻段內可使用的時間 (T_{CR})後，接下來為了得到在這段時間內的最大資料傳輸量，此處仍需要再求出這段時間內可傳送 Data 的個數 $N_{CR-Transmit}$ 。而為了得到這個值，在一開始我們需要求出在資料傳輸之頻段時進行一次資料傳輸〈即從無線感知網路使用者切入資料傳輸頻段開始，至該位使用者返回控制封包傳輸頻段止〉的所需時間($T_{CR-Data1}$)：

$$\begin{aligned} T_{CR-Data1} &= T_{Sensing} + sifs + T_{RTS} + sifs + T_{CTS} + \\ &\quad (difs + T_{DATA} + sifs + T_{ACK} + sifs + T_{RTI} + sifs_{CR}) * TxOP_{CR} \\ &= 2000 + 10 + 248 + 10 + 248 + (20+8496+10+248+10+248+100) * TxOP_{CR} \\ &= 2516 + 9132 * TxOP_{CR} \end{aligned}$$

於此處， $TxOP_{CR}$ 為一變動值，其值為 1, 2, 3, 4。因此，可以求得 $T_{CR-Data1}$ 如下：

$$T_{CR-Data1} = \begin{cases} 11648 (\mu\text{s}) & , TxOP_{CR} = 1 \\ 20780 (\mu\text{s}) & , TxOP_{CR} = 2 \\ 29912 (\mu\text{s}) & , TxOP_{CR} = 3 \\ 39044 (\mu\text{s}) & , TxOP_{CR} = 4 \end{cases}$$

由以上之數值，可以求出這段時間內可傳送 Data 的個數 $N_{CR1-Transmit}$ ，於此先行定義

無線感知網路使用者在資料傳輸頻段的傳輸準備時間(T_{Prepare})與用來計算無線感知

網路使用者在該秒結束前若仍未返回控制封包交換頻段時傳輸的 Data 個數所用的

輔助函式 $f_I(T_{\text{CR-Data1}}, \text{TxOP}_{\text{CR}})$:

$$T_{\text{Prepare}} = T_{\text{Sensing}} + \text{sifs} + T_{\text{RTS}} + \text{sifs} + T_{\text{CTS}}$$

$$= 2516 (\mu\text{s})$$

$$T_{\text{DataSegment}} = \text{difs} + T_{\text{DATA}} + \text{sifs} + T_{\text{ACK}} + \text{sifs} + T_{\text{RTI}} + \text{sifs}_{\text{CR}} = 9132 (\mu\text{s})$$

$$f_I(T_{\text{CR-Data1}}, \text{TxOP}_{\text{CR}}) = \begin{cases} 0, & T_{\text{CR}} \% T_{\text{CR-Data1}} \geq 0 \vee \frac{T_{\text{CR}} \% T_{\text{CR-Data1}} - T_{\text{Prepare}}}{T_{\text{DataSegment}}} < 1 \\ 1, & T_{\text{CR}} \% T_{\text{CR-Data1}} > 0 \wedge 1 \leq \frac{T_{\text{CR}} \% T_{\text{CR-Data1}} - T_{\text{Prepare}}}{T_{\text{DataSegment}}} < 2 \\ 2, & T_{\text{CR}} \% T_{\text{CR-Data1}} > 0 \wedge 2 \leq \frac{T_{\text{CR}} \% T_{\text{CR-Data1}} - T_{\text{Prepare}}}{T_{\text{DataSegment}}} < 3 \\ 3, & T_{\text{CR}} \% T_{\text{CR-Data1}} > 0 \wedge 3 \leq \frac{T_{\text{CR}} \% T_{\text{CR-Data1}} - T_{\text{Prepare}}}{T_{\text{DataSegment}}} < 4 \end{cases}$$

$$N_{\text{CR1-Transmit}} = \text{floor}(T_{\text{CR}} / T_{\text{CR-Data1}} * \text{TxOP}_{\text{CR}} + f_I(T_{\text{CR-Data1}}, \text{TxOP}_{\text{CR}}))$$

$$= \begin{cases} 46 & , \text{TxOP}_{\text{CR}} = 1 \\ 52 & , \text{TxOP}_{\text{CR}} = 2 \\ 54 & , \text{TxOP}_{\text{CR}} = 3 \\ 55 & , \text{TxOP}_{\text{CR}} = 4 \end{cases}$$

因此，便可利用此求出在剩餘時間內 5 個資料傳輸頻段的總傳輸量理論值(TH_{CR1})：

$$\text{TH}_{\text{CR1}} = N_{\text{CR1-Transmit}} * 2048 * 8 / 10^6 * N_{\text{DataChannel}}$$

$$= \begin{cases} 3.76832 (\text{Mbps}) & , \text{TxOP}_{\text{CR}} = 1 \\ 4.25984 (\text{Mbps}) & , \text{TxOP}_{\text{CR}} = 2 \\ 4.42368 (\text{Mbps}) & , \text{TxOP}_{\text{CR}} = 3 \\ 4.50560 (\text{Mbps}) & , \text{TxOP}_{\text{CR}} = 4 \end{cases}$$

■

- 無線感知網路使用者在傳輸樣式二的流量理論值 (TH_{CR2})

基本上計算的方法同上，但此處由於 TCP 通訊協定涉及了擁塞視窗等機制，使得此

協定較為複雜，因此在此進行一個假設：在傳輸的過程中，一律是傳送端送出 $TxOP_{CR}$ 個資料，而接收端會在此時回傳 $TxOP_{CR}$ 個 TCP ACK。因此，基於這樣的模型，在此可以進行簡化後的 TCP 流量之理論推導，而在最一開始，也還是先從求出在資料傳輸之頻段時進行一次資料傳輸的所需時間($T_{CR-Data1}$)開始推導：

$$\begin{aligned}
 T_{CR-Data2} &= T_{Sensing} + sifs + T_{RTS} + sifs + T_{CTS} + \\
 &\quad (difs + T_{TCP-DATA} + sifs + T_{ACK} + sifs + T_{RTI} + sifs_{CR}) * TxOP_{CR} + \\
 &\quad T_{Sensing} + sifs + T_{RTS} + sifs + T_{CTS} + \\
 &\quad (difs + T_{TCP-ACK} + sifs + T_{ACK} + sifs + T_{RTI} + sifs_{CR}) * TxOP_{CR} \\
 &= 2000 + 10 + 248 + 10 + 248 + (20 + 6360 + 10 + 248 + 10 + 248 + 100) * TxOP_{CR} + \\
 &\quad 2000 + 10 + 248 + 10 + 248 + (20 + 464 + 10 + 248 + 10 + 248 + 100) * TxOP_{CR} \\
 &= 2516 + 6996 * TxOP_{CR} + 2516 + 1100 * TxOP_{CR} \\
 &= 5032 + 8096 * TxOP_{CR} \\
 &= \begin{cases} 13128 (\mu s) & , TxOP_{CR} = 1 \\ 21224 (\mu s) & , TxOP_{CR} = 2 \\ 29320 (\mu s) & , TxOP_{CR} = 3 \\ 37416 (\mu s) & , TxOP_{CR} = 4 \end{cases}
 \end{aligned}$$

由以上之數值，可以求出這段時間內可傳送 Data 的個數 $N_{CR-Transmit}$ 。但此處需要注意的：若成功傳送完一個 TCP Data，即將此資料傳輸量記錄下來。因此，使用的式子需要進行修正：

$$T_{DataSegment} = difs + T_{TCP-DATA} + sifs + T_{ACK} + sifs + T_{RTI} + sifs_{CR} = 6996$$

$$f_2(T_{CR-Data1}, TxOP_{CR}) = \begin{cases} 0, & T_{CR} \% T_{CR-Data1} \geq 0 \vee \frac{T_{CR} \% T_{CR-Data1} - T_{Prepare}}{T_{DataSegment}} < 1 \\ 1, & T_{CR} \% T_{CR-Data1} > 0 \wedge 1 \leq \frac{T_{CR} \% T_{CR-Data1} - T_{Prepare}}{T_{DataSegment}} < 2 \\ 2, & T_{CR} \% T_{CR-Data1} > 0 \wedge 2 \leq \frac{T_{CR} \% T_{CR-Data1} - T_{Prepare}}{T_{DataSegment}} < 3 \\ 3, & T_{CR} \% T_{CR-Data1} > 0 \wedge 3 \leq \frac{T_{CR} \% T_{CR-Data1} - T_{Prepare}}{T_{DataSegment}} < 4 \\ 4, & T_{CR} \% T_{CR-Data1} > 0 \wedge 4 \leq \frac{T_{CR} \% T_{CR-Data1} - T_{Prepare}}{T_{DataSegment}} \end{cases}$$

$$N_{CR2-Transmit} = \text{floor}(T_{CR} / T_{CR-Data1}) * TxOP_{CR} + f_2(T_{CR-Data2}, TxOP_{CR})$$

$$= \begin{cases} 41 & , TxOP_{CR} = 1 \\ 51 & , TxOP_{CR} = 2 \\ 55 & , TxOP_{CR} = 3 \\ 58 & , TxOP_{CR} = 4 \end{cases}$$

因此，便可利用此求出在剩餘時間內 5 個資料傳輸頻段的總傳輸量理論值(TH_{CR2})：

$$TH_{CR2} = N_{CR2-Transmit} * 1474 * 8 / 10^6 * N_{DataChannel}$$

$$= \begin{cases} 2.41736 \text{ (Mbps)} & , TxOP_{CR} = 1 \\ 3.00696 \text{ (Mbps)} & , TxOP_{CR} = 2 \\ 3.24280 \text{ (Mbps)} & , TxOP_{CR} = 3 \\ 3.41968 \text{ (Mbps)} & , TxOP_{CR} = 4 \end{cases}$$

■

藉由此推導，我們可以得到無線感知網路使用者在進行傳輸時的理論值。於後面的

分析中，將以此來進行模擬結果的驗證。

5.2. 無終端機隱匿問題的模擬情境之效能評估

在本章節將針對沒有終端隱匿問題之情境進行模擬測試，藉此能夠對於原始機制與

改良後之機制進行比較與分析。而在分析的過程中，將依序介紹模擬的拓樸以及兩種測試流量樣式之測試結果，並且展示其無線感知網路之使用者接收端所收到的總接收流量、主要使用者接收端之總接收流量，以進行相關之評比與分析。

- 模擬拓樸

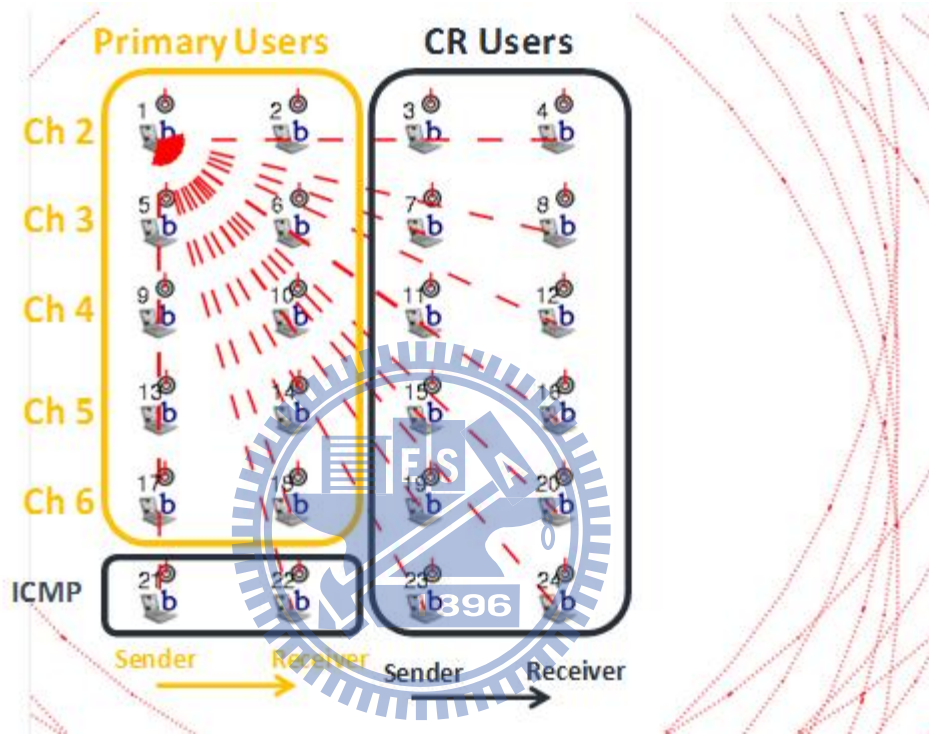


圖 18、無終端機隱匿問題情境下的拓樸。

此拓樸之設計主要是由 5 對主要使用者、7 對無線感知網路之使用者所構成，而在各個無線網路使用設備的擺放上主要是讓在此拓樸中的每一位成員，其均可接收到來自其他人傳送之訊號。設計的拓樸如圖 18 所示，以淡色線框住並標示者為主要使用者，其中置於左側者為傳送端，右側者為接收端；此外對於每一傳送端與接收端，其皆佔用一個頻段，如圖所示，佔用的頻段為頻段編號 2 至編號 6。而以深色線框住並標示者為無線感知網路之使用者，和主要使用者一樣，均是左側設備為傳送端，而右側設備為接收端；此外，於設備 21 與 22 處則是進行 Round-Trip 延遲時間量測之設備，而在圖中整

個右側的無線感知網路設備則進行一般封包的傳送與接收。而在圖中的虛線則是每一個無線網路設備其最大干擾範圍，此代表了所有使用者均聽得到其他使用者傳送之訊號。底下為其效能：

- 封包傳送樣式 1：無線感知網路使用者之傳輸樣式為封包到達率為 1 的情況。

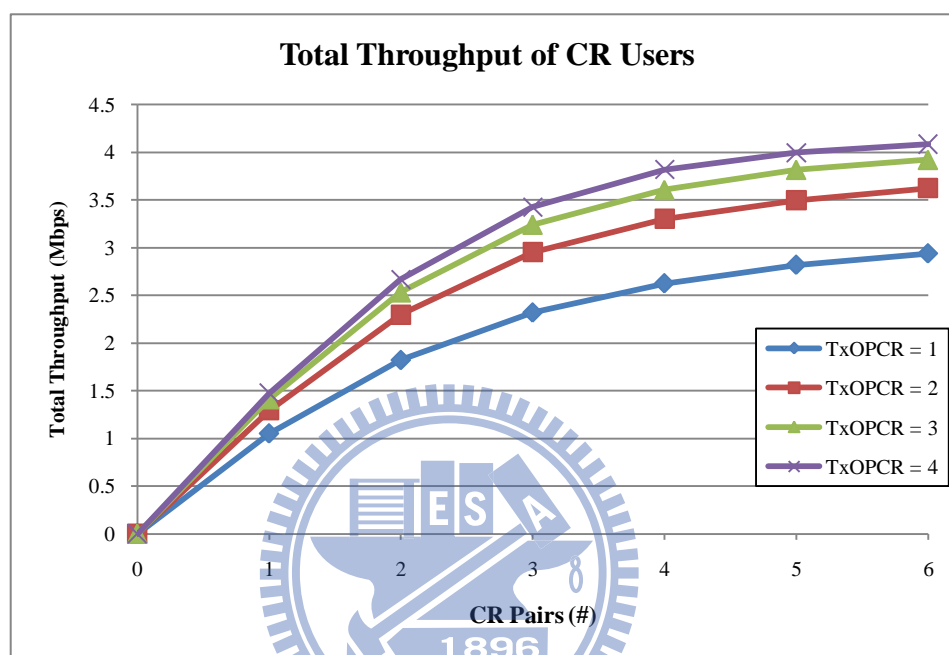


圖 19、原有機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的效能。

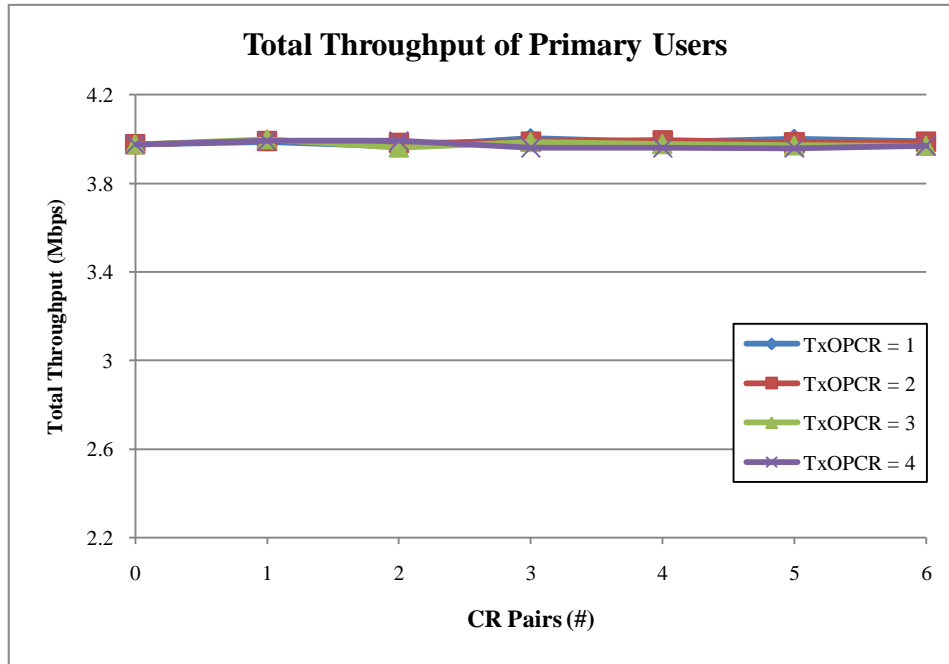


圖 20、原有機制於無終端機隱匿問題之情境下，主要使用者的效能。

圖 19 與圖 20 是在此情境下主要是用者與無線感知網路之使用者均採用由 MAC 層傳送封包的模擬結果，其中主要使用者的頻均產生封包之流量為 4 Mbps。從圖 19 中可看出無線感知網路之使用者其接收端之流量藉由 $TxOP_{CR}$ 的不同與其傳送-接收總對數之不同，而使得總傳送流量上升，並在總對數為 5 以後則趨於緩和。而在圖 20 的主要使用者方面則是維持總流量為 4 Mbps 左右的情境。

而在結果驗證的方面，由 5.1 的推導可以得到，在 $TxOP_{CR}$ 為 1~4 時，其最大可傳輸之流量分別為：3.77、4.26、4.42、4.51 Mbps，而在數據的呈現上，最大傳輸流量則分別為：2.94、3.63、3.92 與 4.09。以結果來說，模擬的數據確實是低於理論最大值，但卻有不小的落差，落差約落在 0.5 ~ 0.8 Mbps。在此推測應該是由切換至資料傳輸頻段的感測機制所導致的，這部份的例子如第四章的圖 12 所示。因此，在之後的改良機制的部份將會透過此部份機制的改善，使得流量能夠更為接近理論值。

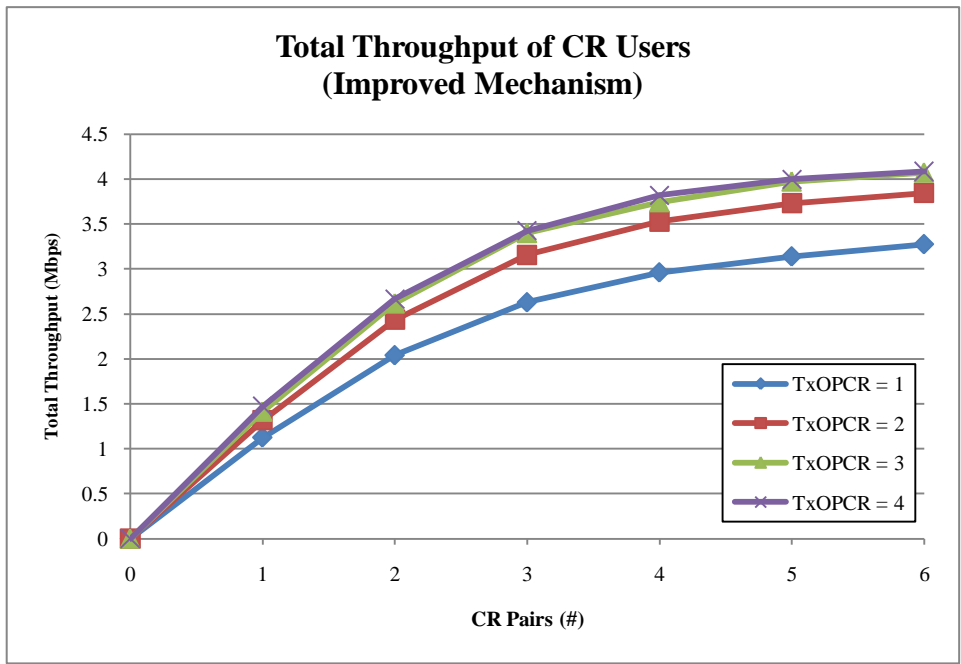


圖 21、改良之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的效能。

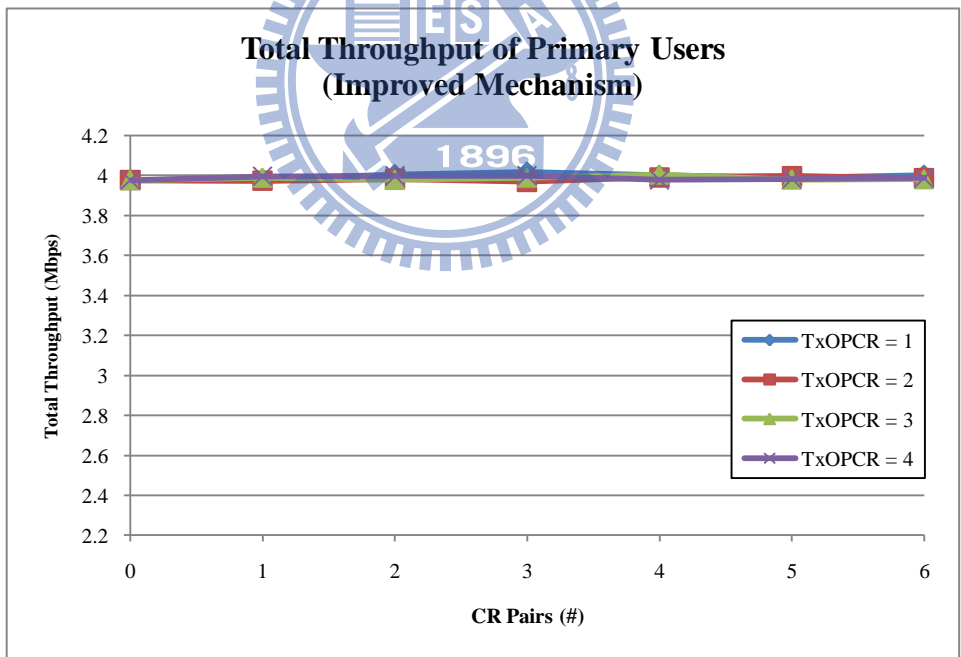


圖 22、改良機制於無終端機隱匿問題之情境下，主要使用者的效能。

圖 21 與圖 22 所呈現的結果為改良機制之效能比較，在圖 22 中主要使用者在傳輸的總效能方面則與原始之設計機制一樣，仍然維持在 4 Mbps 左右，此代表著改良之機

制仍然承襲了原有機制之優點，使得無線感知網路之使用者無法嚴重干擾到主要使用者進行資料傳輸。而在無線感知網路使用者的效能方面，由圖 21 與表 2 中可看出整體的傳輸效能均有所增長，改良的機制對於無線感知網路使用者在傳輸流量的改善幅度隨著 $TxOP_{CR}$ 的值上升而有明顯的改善，整體的改良幅度約落在 1% ~ 10% 左右。其增長的主要原因是位於 4.2.2 中所描述的頻段感測機制之改良上，此機制會導致無線感知網路使用者提高其競爭頻寬之能力。因此，由底下表 2 數據搭配理論值的結果可以看出，改良機制與理論值的差距縮減至 0.4 ~ 0.5 Mbps，這部份的原因主要有兩個，首先是在推導的過程中僅以頻段的使用作為主要角度來進行推導，並未考慮到無線感知網路使用者由於競爭頻段與在控制封包交換頻段時所需進行的控制封包的交換而產生的額外所需時間，這樣的現象將會使得無線感知網路的使用者在傳輸上無法達到完美的使用所有資料頻段的情況，進而使得某些無線感知網路使用者在該秒的傳輸量略為下降。

第二個原因則是出自於改良機制造成的負擔，其中包括了更改頻段挑選方式所造成的表頭的更動，以及在 $TxOP_{CR} > 1$ 的情況下，每需要在同一頻段上多傳輸一個資料，即需要再增加 RTS-CTS 的握手時間，在此情況下的確會造成此段時間內無法進行資料的傳輸。但是此機制是為了要解決終端機隱匿問題而設計的，可以視作其為解決此問題而產生之代價。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	1.051300	1.125995	7.11%	1.294494	1.317116	1.75%
2	1.820164	2.040894	12.13%	2.297984	2.433991	5.92%
3	2.318553	2.630007	13.43%	2.953147	3.157335	6.91%
4	2.625111	2.962662	12.86%	3.302856	3.529587	6.86%
5	2.819272	3.140714	11.40%	3.496109	3.732275	6.76%
6	2.938658	3.276997	11.51%	3.625996	3.846805	6.09%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	1.411434	1.413170	0.12%	1.475073	1.478492	0.23%
2	2.532572	2.617374	3.35%	2.666012	2.717138	1.92%
3	3.243045	3.400687	4.86%	3.426111	3.548656	3.58%
4	3.606731	3.744277	3.81%	3.819722	3.916882	2.54%
5	3.818459	3.974324	4.08%	3.996788	4.104883	2.70%
6	3.922764	4.074602	3.87%	4.085972	4.178236	2.26%

表 2、改良機制與原有機制對無線感知網路在無終端機隱匿問題，無線感知網路使用者在封包傳輸樣式一的流量傳輸效能比較。(單位：Mbps)

CR Nodes	TxOP = 1		TxOP = 2		TxOP=3		TxOP = 4	
	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism
0	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733
1	3.988418	3.982299	3.992050	3.972824	3.999664	3.988892	3.992445	3.995801
2	3.968007	4.004723	3.982102	3.983325	3.961611	3.977680	3.991419	3.997696
3	4.003341	4.017278	3.988853	3.968442	3.986523	3.989050	3.961020	3.998486
4	3.987313	4.001210	3.994380	3.990274	3.977088	4.004684	3.960269	3.977996
5	4.000736	3.989524	3.984194	3.996946	3.971166	3.979417	3.958374	3.983089
6	3.988497	4.001841	3.987866	3.986286	3.969389	3.981312	3.969823	3.987708

表 3、改良機制與原有機制無終端機隱匿問題時，主要使用者在封包傳輸樣式一的流量傳輸效能。(單位：Mbps)

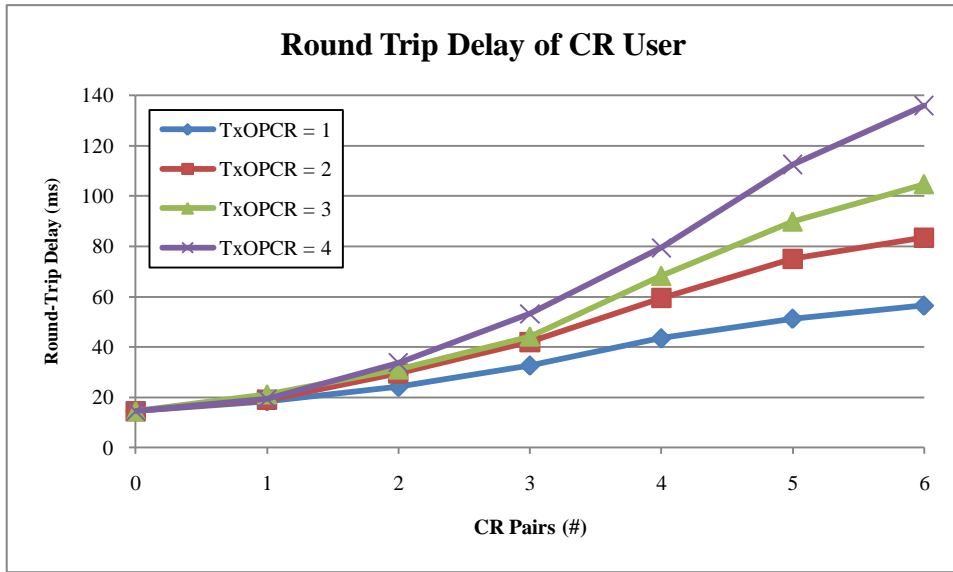


圖 23、原始之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的 Round-Trip 延遲時間。

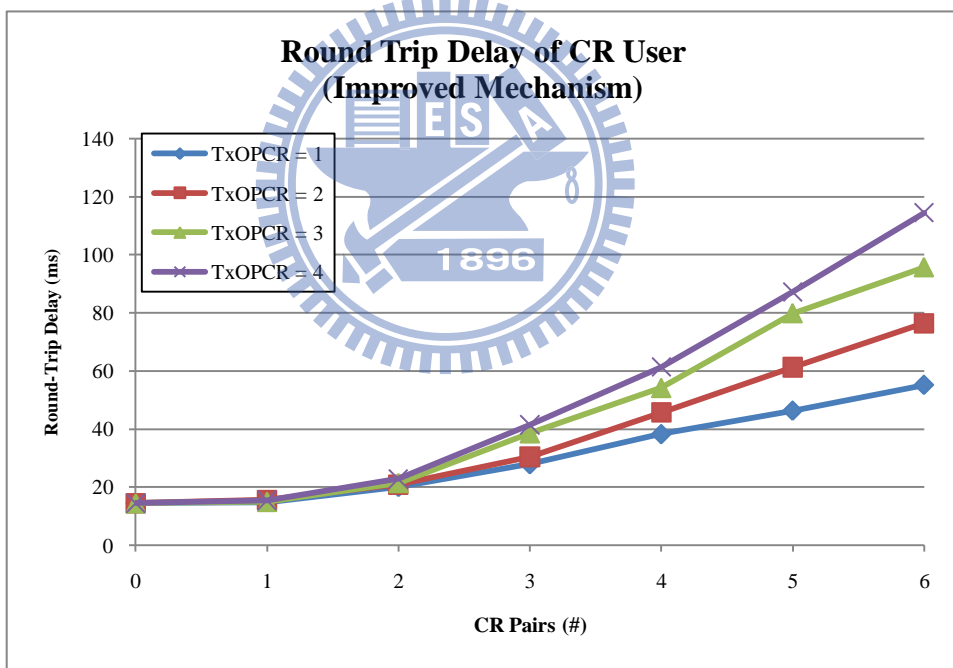


圖 24、原始之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的 Round-Trip 延遲時間。

而在圖 23、24 與表 3 中呈現的主要是原始機制與改良機制在 Round-Trip 延遲時間上的比較。由於在傳輸 ICMP 封包時所傳送的封包大小均一致，因此在此的延遲時間也

可作為在進行封包傳輸時，無線感知網路使用者所需要找尋可用頻段所需時間的一個指標。由圖中可以看出在改良後的機制也對於延遲時間有些許的改善，例如在 6 對無線感知網路進行傳輸的狀況下，其結果較原先少約 20 微秒。此外在整體的改善方面，幅度也大致落在 10~20% 左右。會導致這樣的效能改善的主要原因的部份應該是由於 4.2.2 的機制所造成的。雖然改善幅度仍不大，但若能更進一步的進行或許能使得需要較低延遲時間的應用能夠在此網路中使用。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	18.473	14.875	19.48%	18.995	15.635	17.69%
2	24.179	20.120	16.79%	29.575	20.930	29.23%
3	32.660	27.913	14.53%	42.011	30.458	27.50%
4	43.491	38.271	12.00%	59.439	45.728	23.07%
5	51.293	46.361	9.61%	75.128	61.290	18.42%
6	56.503	55.235	2.24%	83.543	76.419	8.53%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	21.255	14.992	29.47%	19.381	15.465	20.20%
2	31.142	21.473	31.05%	33.745	22.851	32.28%
3	44.221	38.577	12.76%	53.254	41.448	22.17%
4	68.266	54.182	20.63%	79.375	61.350	22.71%
5	89.862	79.895	11.09%	112.579	87.269	22.48%
6	104.693	95.709	8.58%	135.999	114.498	15.81%

表 4、改良機制與原有機制對於無線感知網路在無終端機隱匿問題以 MAC 層之協定直接進行流量傳輸的 Round-Trip Delay 比較。(單位：ms)

- 封包傳送樣式二：無線感知網路使用者之傳輸樣式為 TCP 封包。

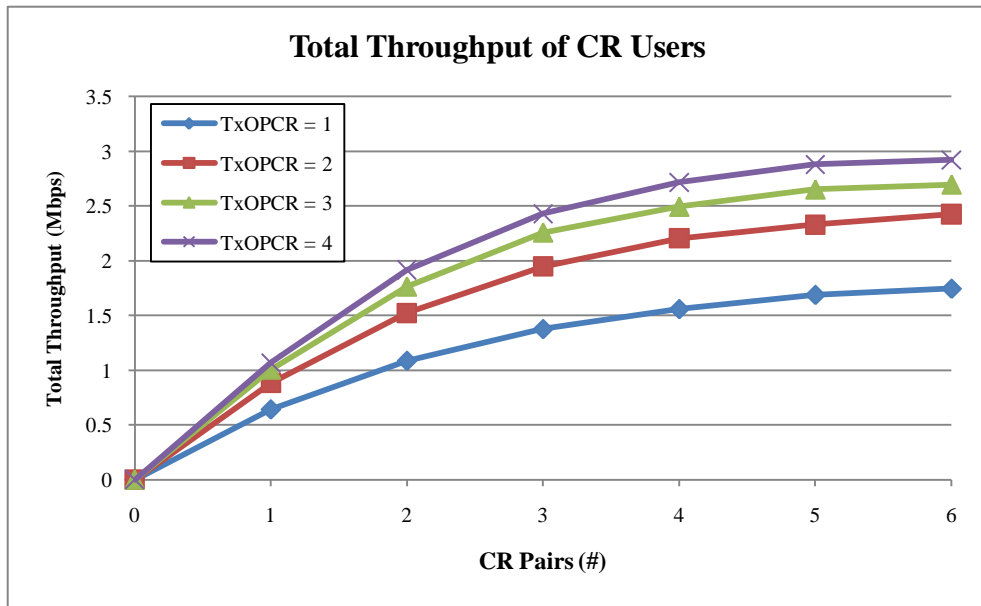


圖 25、原有之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。

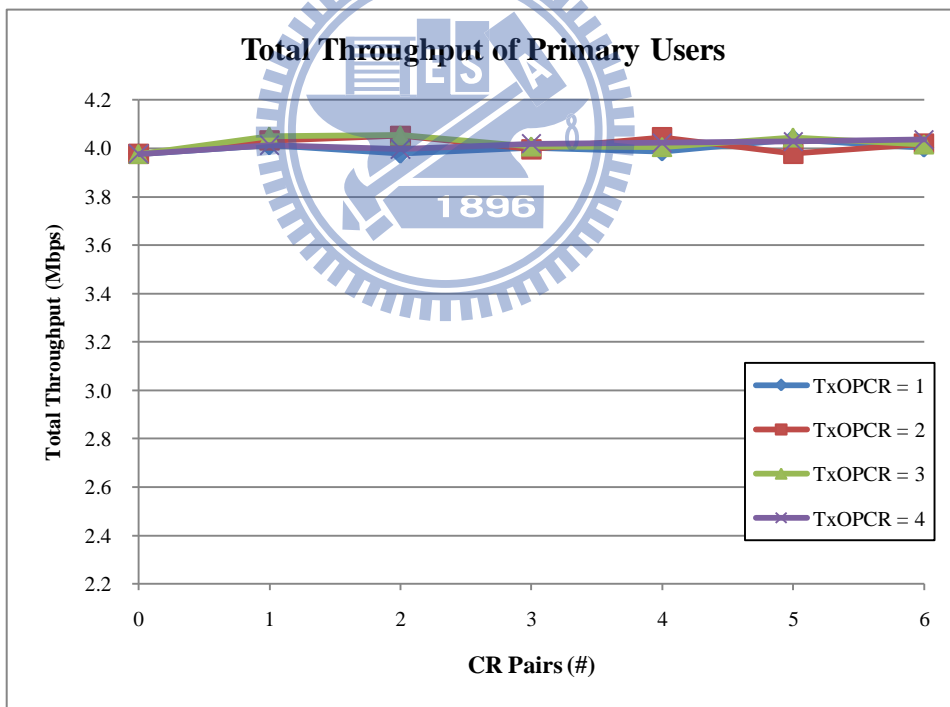


圖 26、原有之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時，主要使用者的效能。

圖 25 與圖 26 展示了在無線感測網路使用者使用 TCP 之傳輸層的通訊協定來進行

資料傳輸時的狀況。從圖 25 可以看到無線感知網路之使用者的傳輸效能按照 $TxOP_{CR}$ 的不同，約為 1.75、2.43、2.70、2.92 Mbps 左右，而理論值的推算則為 2.42、3.01、3.24、3.42 Mbps，由此可看出落差大約為 0.5 ~ 0.7 Mbps，原因與在前一個流量傳輸樣式時的理由大致上相同，其中一個因素主要是在 5.1 推導的式子本身是以頻段使用率為主要分析角度，並沒有考慮到無線感知網路使用者切換至資料傳輸頻段進行傳輸前所需要的額外時間，此外亦包括由於原有感測機制所導致的頻段無使用者使用之時間。這兩個因素會導致單一無線感知網路使用者於該時段無法有效運用頻段資源，進而降低每秒傳輸流量的平均值。

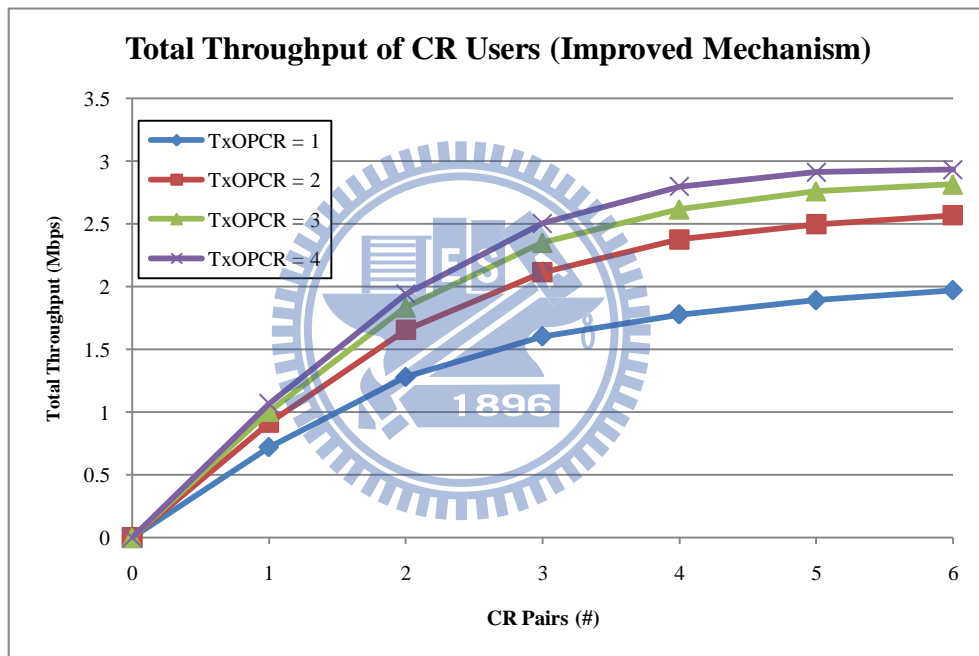


圖 27、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。

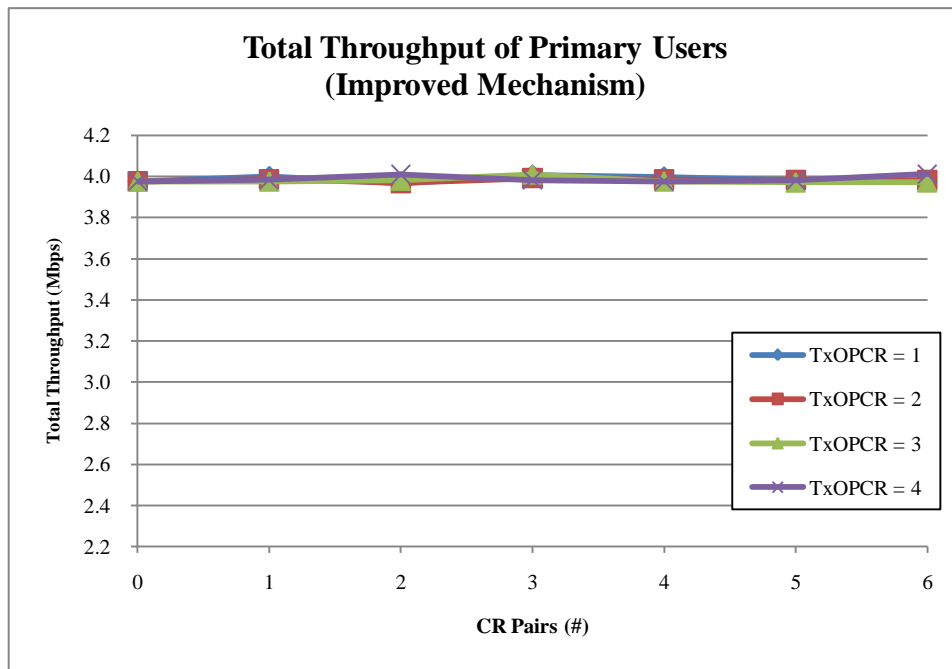


圖 28、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時，主要使用者的效能。

而圖 27、28 與表 5 則是利用改良後的方式來進行無線感知網路使用者以 TCP 之傳輸層通訊協定來進行傳輸資料的結果。從圖 27 搭配表 4 可以看到其結果在 $TxOP_{CR} < 4$ 的狀況下的傳輸效能有些許提升，改善幅度最大約為 15% 左右，而與離理論值的差距也縮小為 0.4 ~ 0.5 Mbps 左右。

改善幅度不大的原因主要是如同在第一種傳輸樣式時的狀況一樣，主要原因有兩個，第一個則是由更改頻段挑選方式所造成的表頭的更動，此部份造成的影響仍然不大，但第二個原因則是在 $TxOP_{CR} > 1$ 的情況下，每需要在同一頻段上多傳輸一個資料，即需要再增加 RTS-CTS 的握手時間，在此情況下的確會造成此段時間內無法進行資料的傳輸，且會延後接收端進行 TCP-ACK 封包傳輸之時間，而這些原因足以將 4.2.2 所帶來的好處抵消。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	0.641253	0.721959	12.59%	0.883140	0.915881	3.71%
2	1.085894	1.279962	17.87%	1.522860	1.656472	8.77%
3	1.376510	1.604029	16.53%	1.947094	2.114505	8.60%
4	1.559113	1.778758	14.09%	2.206231	2.378309	7.80%
5	1.689105	1.891323	11.97%	2.329771	2.496637	7.16%
6	1.745889	1.971469	12.92%	2.425653	2.569888	5.95%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	1.005344	1.005902	0.06%	1.067659	1.068200	0.05%
2	1.764372	1.836655	4.10%	1.916835	1.940230	1.22%
3	2.258417	2.352536	4.17%	2.429528	2.504623	3.09%
4	2.493995	2.618327	4.99%	2.716149	2.797973	3.01%
5	2.651758	2.763544	4.22%	2.879361	2.913220	1.18%
6	2.697166	2.816842	4.44%	2.920930	2.933091	0.42%

表 5、改良機制與原有機制對無線感知網路在無終端機隱匿問題時，無線感知網路使用者在封包傳輸樣式二的傳輸效能比較。(單位：Mbps)

CR Nodes	TxOP = 1		TxOP = 2		TxOP=3		TxOP = 4	
	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism
0	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733
1	3.996505	4.000381	3.987037	3.987313	3.969942	3.976575	3.985655	3.984391
2	3.978430	3.967179	4.012027	3.968678	3.985339	3.980522	3.960111	4.009777
3	3.980009	4.006776	3.980009	3.992287	3.954150	4.010803	3.995525	3.981865
4	3.993116	3.998328	3.968639	3.986050	3.996472	3.976693	3.991340	3.976851
5	3.968284	3.980404	3.980996	3.982496	3.967415	3.971916	3.986207	3.981786
6	3.972429	3.977443	3.981431	3.703063	4.004407	3.972350	4.001918	4.011632

表 6、改良機制與原有機制無終端機隱匿問題時，主要使用者在封包傳輸樣式二的流量傳輸效能。(單位：Mbps)

而在此情況下利用 TCP 進行效能傳輸並無法達到如傳輸樣式一的效能之原因主要是因為 TCP 本身由於需要保證接收端已收到該資料，因此在接收端收到資料後尚還需要回應 TCP-ACK，這樣的機制會使得在傳輸過程中，TCP 接收端仍需要向 TCP 傳送端發出 RTS_{CR} 、 CTS_{CR} 以及進行後續的頻段感測、傳送 TCP-ACK 等動作，使得 1 秒內能夠進行的傳輸量降低。

而在現在以 TCP 進行傳輸時所得到的 Round-Trip 延遲時間如圖 29、30 與表 7 所示，其中原始機制與改良機制的延遲時間差距不大。此外從中可看到其延遲時間較圖 23、24 低，原因主要是因為在此傳輸樣式中，無線感知網路使用者對於單一資料的傳輸時間較樣式一來得短，如此會使得其他使用者等待此頻段變成可用的狀況所需要等待的時間較短，進而降低整個 Round-Trip 延遲時間。

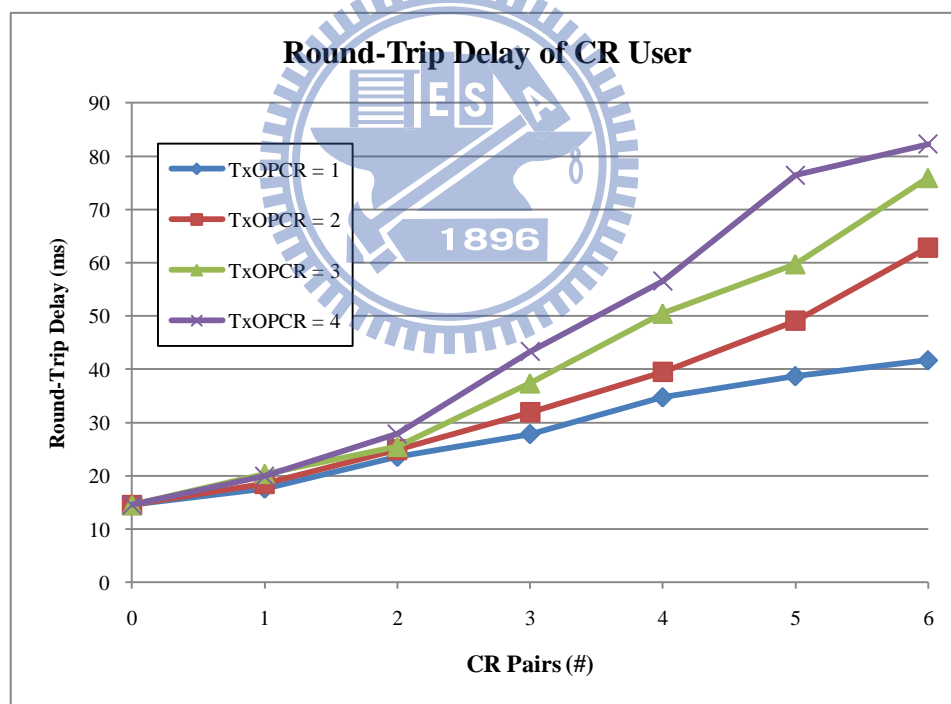


圖 29、原始之機制於無終端機隱匿問題之情境下，以 TCP 作為進行封包之方法時，無線感知網路使用者的 Round-Trip 延遲時間。

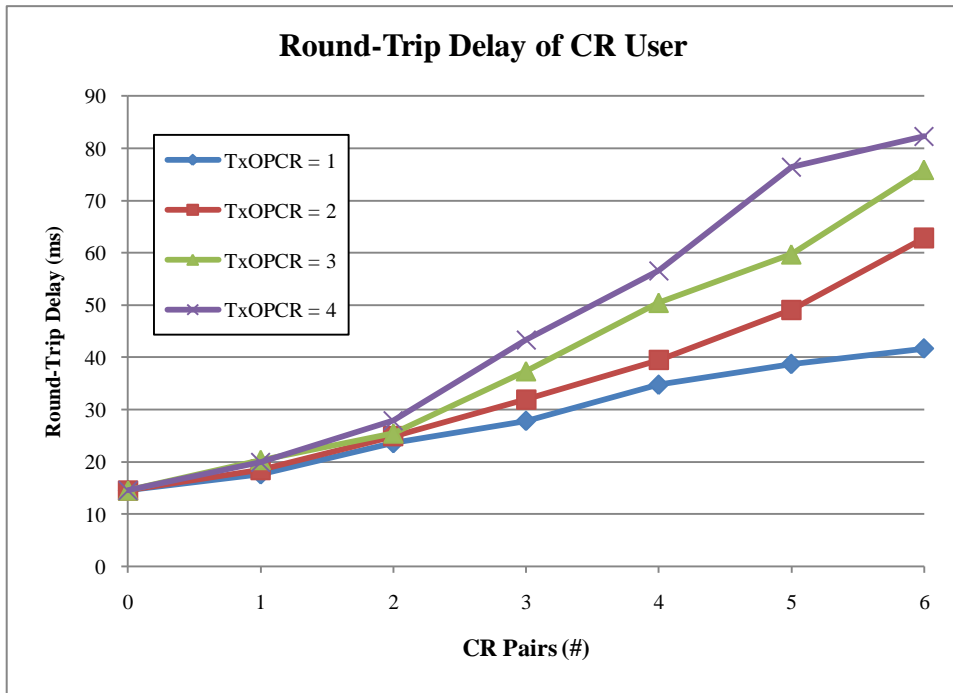


圖 30、改良之機制於無終端機隱匿問題之情境下，以 TCP 作為進行封包之方法時，無線感知網路使用者的 Round-Trip 延遲時間。



CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	17.604	15.631	11.21%	18.490	14.939	19.21%
2	23.569	19.660	16.58%	24.903	22.053	11.45%
3	27.834	25.182	9.53%	31.910	28.708	10.03%
4	34.735	31.974	7.95%	39.482	38.521	2.43%
5	38.710	36.045	6.88%	49.085	45.886	6.52%
6	41.679	41.648	0.07%	62.848	54.501	13.28%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	20.377	15.716	22.88%	19.913	15.226	23.54%
2	25.454	22.104	13.16%	27.866	22.851	17.99%
3	37.363	31.266	16.32%	43.311	35.588	17.83%
4	50.458	43.170	14.44%	56.564	52.182	7.75%
5	59.737	53.283	10.80%	76.402	66.867	12.48%
6	75.922	68.266	10.08%	82.270	79.500	3.37%

表 7、改良機制與原有機制對於無線感知網路在無終端機隱匿問題以 TCP 通訊協定來進行流量傳輸的 Round-Trip Delay 比較。(單位：ms)

5.3. 在終端隱匿之情境之效能比較

在本章節將針對具有終端隱匿問題之情境進行模擬測試，藉此能夠了解在原始機制與改良後之機制下，無線感知網路使用者與主要使用者之效能。而在分析的過程中，將依序介紹模擬的拓樸以及兩種測試流量樣式之測試結果，並且針對各個結果展示其無線感知網路之使用者接收端所收到的總接收流量、主要使用者接收端之總接收流量。

- 模擬拓樸

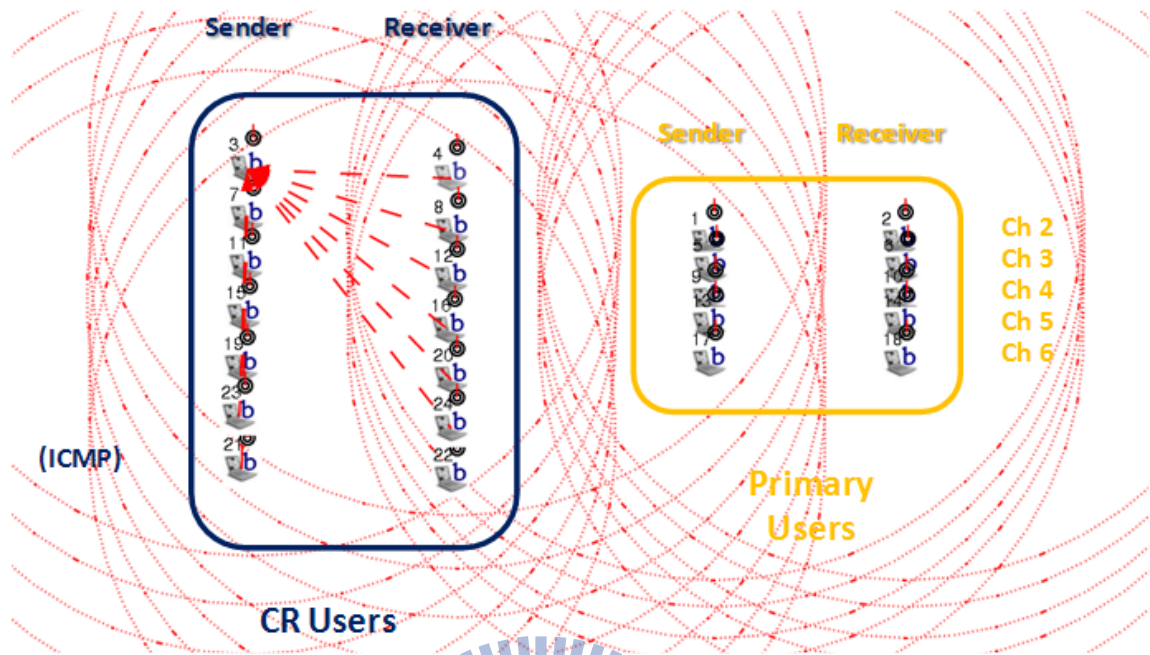


圖 31、終端機隱匿問題情境下的拓樸。

在此情境下，圖 31 最左側的部份為無線感知網路的使用者，傳輸與接收端之角色與 5.2 的拓樸相同，除了最底下的設備 21、22 主要是用以進行 Round-Trip 延遲時間的量測之外，其他的六組無線感知網路之設備均用於流量測試，左端為傳送端，而右端為接收端。而在右側的六組設備則是在此頻帶之主要使用者，其中左端為傳送端，右端為接收端。此外在此拓樸中的虛線則表示了各個網路設備的訊號傳輸最大干擾範圍，由圖中可知，無線感知網路之接收端會受到主要使用者的干擾，但其傳送端卻不會受到其干擾。因此，在此情境下即達到了圖 11 所陳述之隱匿終端機問題。在本節中，將針對此問題進行效能的評估，藉此判斷本篇論文所提出的改良方案是否能夠增進在此情境下之傳輸效能。

- 封包傳送樣式一：無線感知網路使用者之傳輸樣式為封包到達率為 1 的情況。

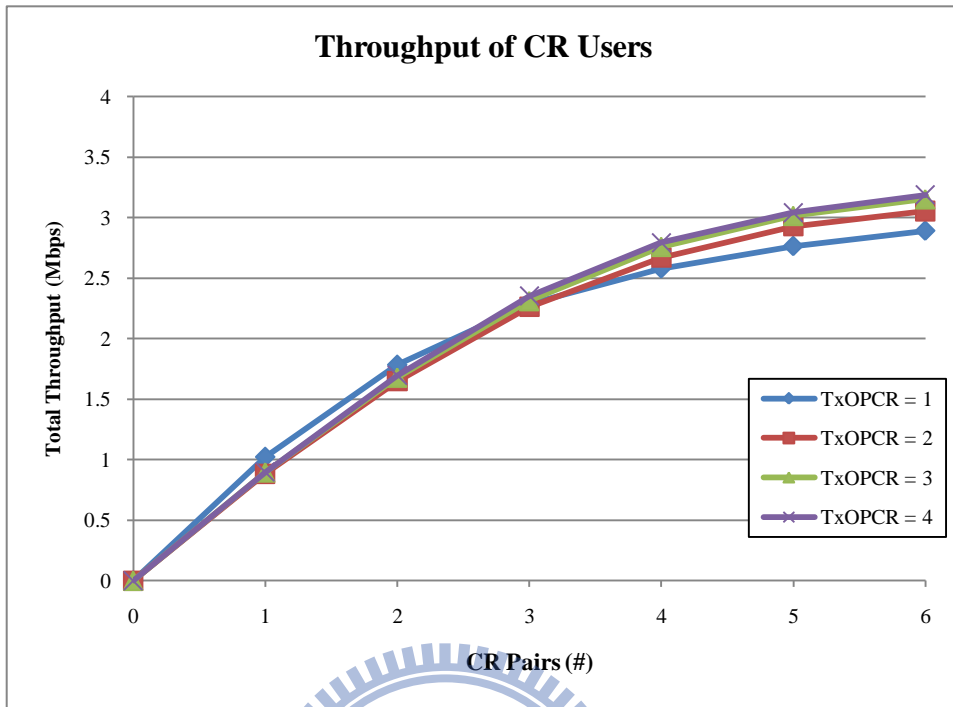


圖 32、原有機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的效能。

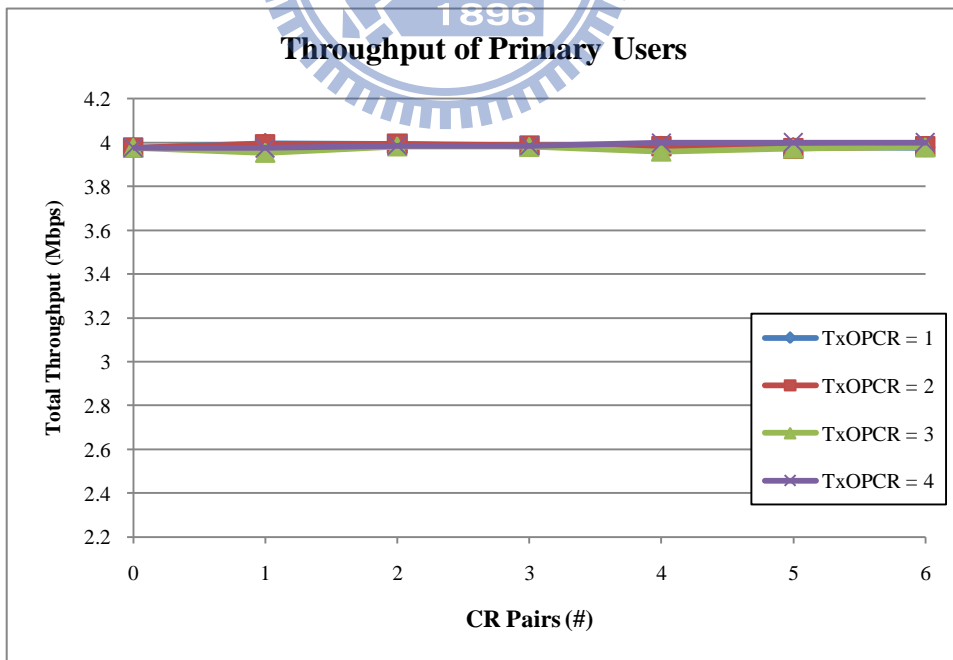


圖 33、原有機制於無終端機隱匿問題之情境下，主要使用者的效能。

圖 32、33 展示了原始機制在終端隱匿情境下的表現，從圖 33 可以看出在此情境下主要使用者仍舊不會受到無線感知網路使用者的干擾。但是從圖 32 可以看到在不同的 $TxOP_{CR}$ 之情況下，傳輸效能差異不大。會造成這樣的原因主要是因為在進行同一頻段第二次以後的傳輸，接收端可能在此時受到主要使用者傳輸端的影響，進而導致此次傳輸失敗，因此在結果的呈現上與 $TxOP_{CR} = 1$ 時僅有較小的差距，如此會使得當初設計在同一頻段中進行多次傳輸的機制無法發揮其效果。

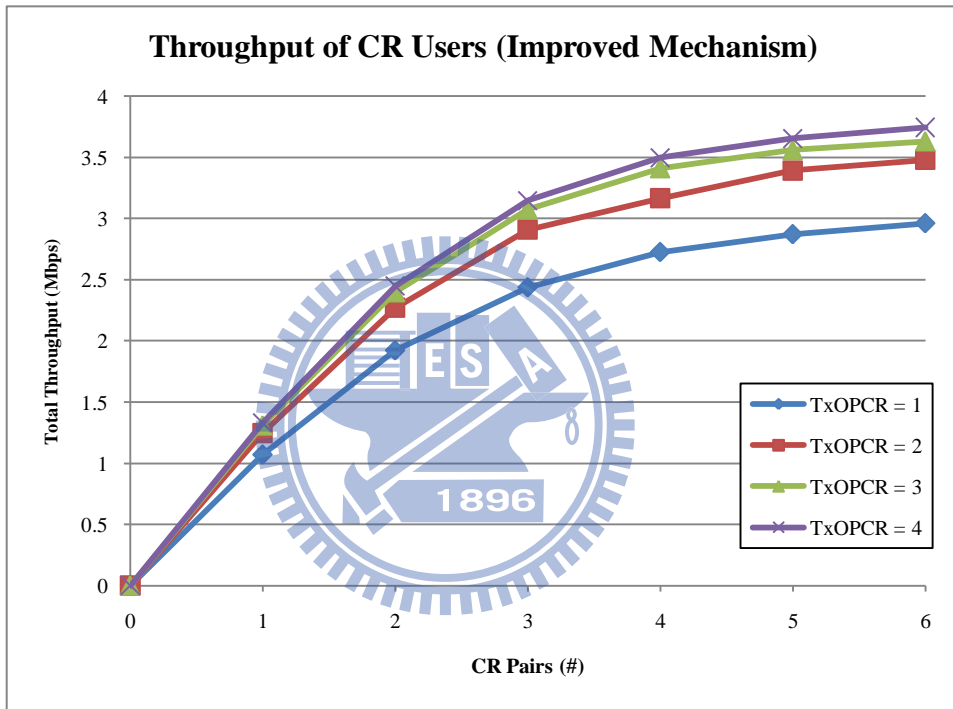


圖 34、改良之機制於無終端機隱匿問題之情境下，在封包到達率為 1 之流量樣式中，無線感知網路使用者的效能。

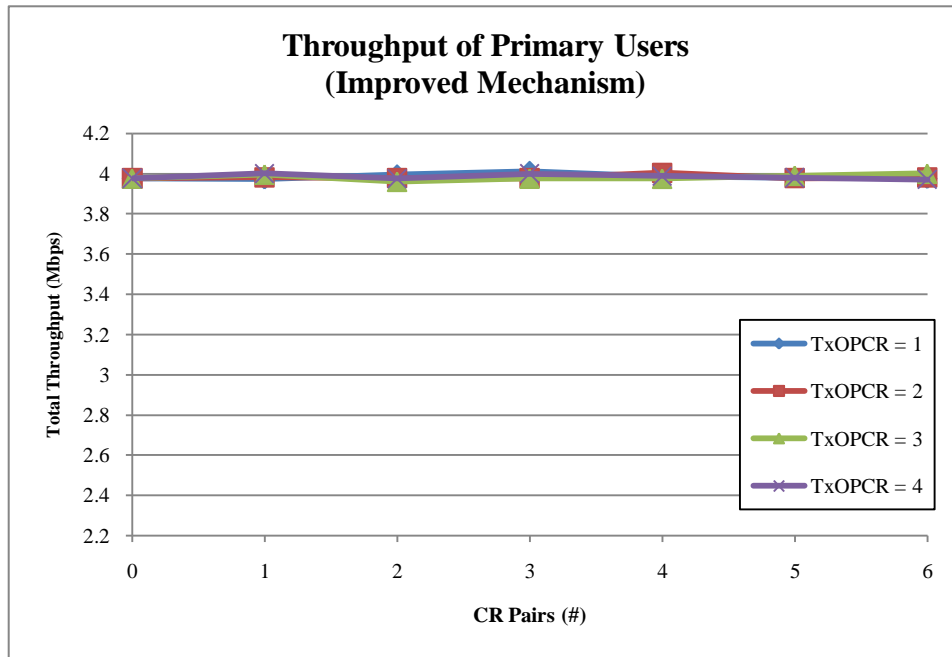


圖 35、改良之機制於無終端機隱匿問題之情境下，主要使用者的效能。

圖 34、35 與表 8 則呈現了改良機制的傳輸效能。在圖 35 中仍然可以看出主要使用者幾乎還是不會受到無線感知網路使用者的干擾；而在圖 34 與表 6 中很明顯可以看到在不同的 $TxOP_{CR}$ 的狀況下，其傳輸效能有較明顯的差異，例如在 $TxOP_{CR} = 4$ ，無線感知網路使用者總傳輸對數為 6 組的情況中可以看出，在原有機制時其傳輸效能約為 3.2 Mbps；而在改良機制上則上升到 3.7 Mbps，改良幅度約提高了 17% 左右。以這樣的結果來看，可以了解在增加了進行多次傳輸時 RTS-CTS 之握手機制後，傳送的資料可以受到完整的保護，使得其效能得以提升。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	1.022678	1.069618	4.59%	0.882722	1.246408	41.20%
2	1.783408	1.921587	7.75%	1.648437	2.270309	37.72%
3	2.285509	2.436597	6.61%	2.261663	2.908456	28.60%
4	2.580381	2.725074	5.61%	2.669565	3.166731	18.62%
5	2.764948	2.871187	3.84%	2.929025	3.392317	15.82%
6	2.891480	2.961043	2.41%	3.054886	3.480198	13.92%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	0.895593	1.309062	46.17%	0.896225	1.331921	48.61%
2	1.679104	2.398262	42.83%	1.695803	2.447888	44.35%
3	2.311289	3.074862	33.04%	2.352782	3.146202	33.72%
4	2.759658	3.412539	23.66%	2.793926	3.498043	25.20%
5	3.015683	3.560895	18.08%	3.041976	3.654738	20.14%
6	3.152756	3.632629	15.22%	3.188840	3.745619	17.46%

表 8、改良機制與原有機制對無線感知網路在終端機隱匿問題時，無線感知網路使用者在封包傳輸樣式一的傳輸效能比較。(單位：Mbps)

CR Nodes	TxOP = 1		TxOP = 2		TxOP=3		TxOP = 4	
	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism
0	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733
1	3.997064	3.972863	3.993116	3.981549	3.953755	3.996551	3.975035	4.001210
2	3.992429	3.994222	3.994617	3.977175	3.983207	3.960941	3.985379	3.976061
3	3.981115	4.011593	3.986326	3.977246	3.980444	3.976416	3.985339	4.000144
4	3.983641	3.984747	3.984155	4.004763	3.959322	3.975943	3.998209	3.988063
5	3.977759	3.980720	3.973298	3.978035	3.973061	3.990748	3.999986	3.979894
6	3.977167	3.975469	3.981983	3.980878	3.978706	4.000815	3.999749	3.971126

表 9、改良機制與原有機制在終端機隱匿問題時，主要使用者在封包傳輸樣式一的傳輸效能。(單位：Mbps)

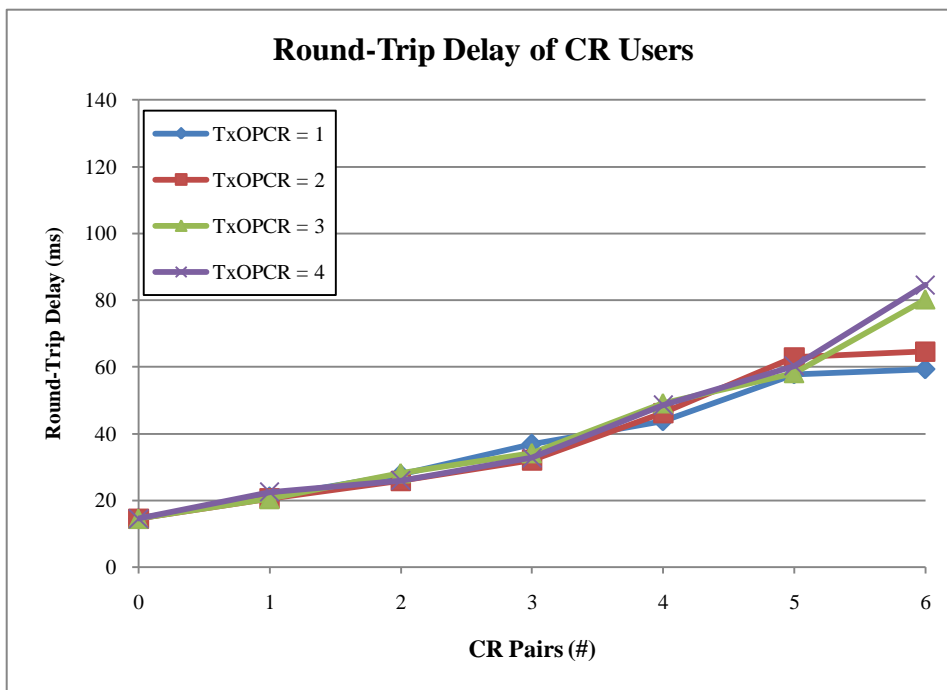


圖 36、原始之機制於無終端機隱匿問題之情境下，無線感知網路使用者的 Round-Trip 延遲之量測。

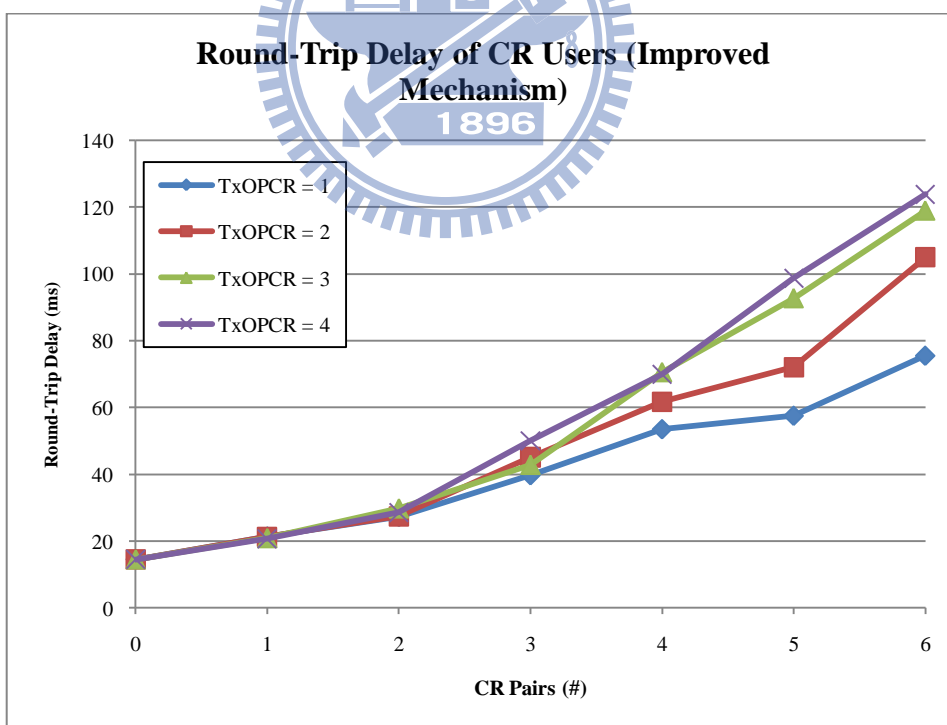


圖 37、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者的 Round-Trip 延遲之量測。

圖 36、37 與表 10 則展現了在此情境下的原始機制與改良機制下，無線感知網路使用者在 Round-Trip 延遲時間的量測，在圖 36 呈現了原始機制下的延遲的狀況，其值約落在 100 微秒以內且在不同 TxOP_{CR} 的情況下的數值較為集中；而圖 37 則是呈現了在改良機制之下的延遲狀況，其在不同 TxOP_{CR} 的情形下，其數據較分散，且其值較原始機制下來得大。這點從表 7 中很明顯可以看出，和前面幾個表格不同的地方在於：改良機制在此情境下對於延遲時間上的無法造成改善，使其改善比例皆為負值。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	21.110	21.327	-1.03%	20.588	21.310	-3.50%
2	27.800	27.227	2.06%	25.881	27.418	-5.94%
3	36.900	39.701	-7.59%	32.039	45.117	-40.82%
4	43.727	53.490	-22.33%	46.174	61.716	-33.66%
5	57.774	57.507	0.46%	62.852	72.120	-14.74%
6	59.281	75.501	-27.36%	64.577	104.922	-62.48%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	20.434	20.887	-2.22%	22.486	20.747	7.73%
2	28.181	29.779	-5.67%	25.990	28.598	-10.04%
3	34.192	42.809	-25.20%	32.783	50.039	-52.63%
4	49.130	70.539	-43.58%	48.529	69.975	-44.19%
5	58.185	92.674	-59.28%	60.306	98.749	-63.75%
6	80.269	118.892	-48.12%	84.550	123.855	-46.49%

表 10、改良機制與原有機制對於無線感知網路在終端機隱匿問題下以 MAC 層之協定直接進行流量傳輸的 Round-Trip Delay 比較。(單位：ms)

此部份原因主要是在原始機制下，對於 TxOP_{CR} > 1 的情況下其行為與 TxOP_{CR} = 1

的狀況較為類似，因此在進行頻寬的使用上因為頻段忙碌而導致使用時間延後的狀況較不容易發生。也因為如此的狀況，在數據的部份原始機制下，數據才顯得集中。而在改良機制下，數據則與在 5.2 的圖 23 描述的在原始機制下，無線感知網路的 Round-Trip 延遲時間類似，但比起圖 24 的在改良機制下的 Round-Trip 延遲時間部份來得大。原因主要是在此情境下，要進行封包的傳輸時仍需要進行頻段的感測，若此時存在任意無線感知網路傳送端傳送 RTS，但卻遲遲無法等到接收端回應 CTS，那麼在接下來聽得到該 RTS 的網路設備仍然會將接下來的時間認定為頻段忙碌，而非閒置狀態，直至 NAV 設定的時間結束為止，因此在此情況下仍需要將使用的資料頻段切換至其他的資料頻段。而在先前 5.1 的情境下不會發生的原因則是因為該情境並非隱匿終端機問題，因此傳送端仍然會感應得到頻段的忙碌，進而不會送出 RTS，使得其他傳送者也不會受到此干擾。



- 封包傳送樣式二：無線感知網路使用者之傳輸樣式為 TCP 封包。

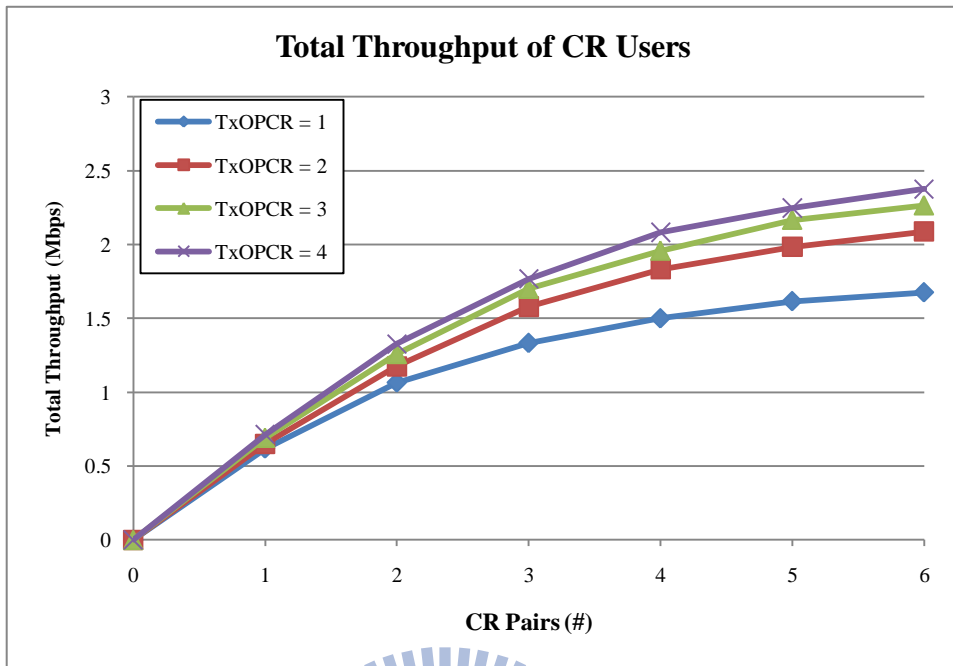


圖 38、原有之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。

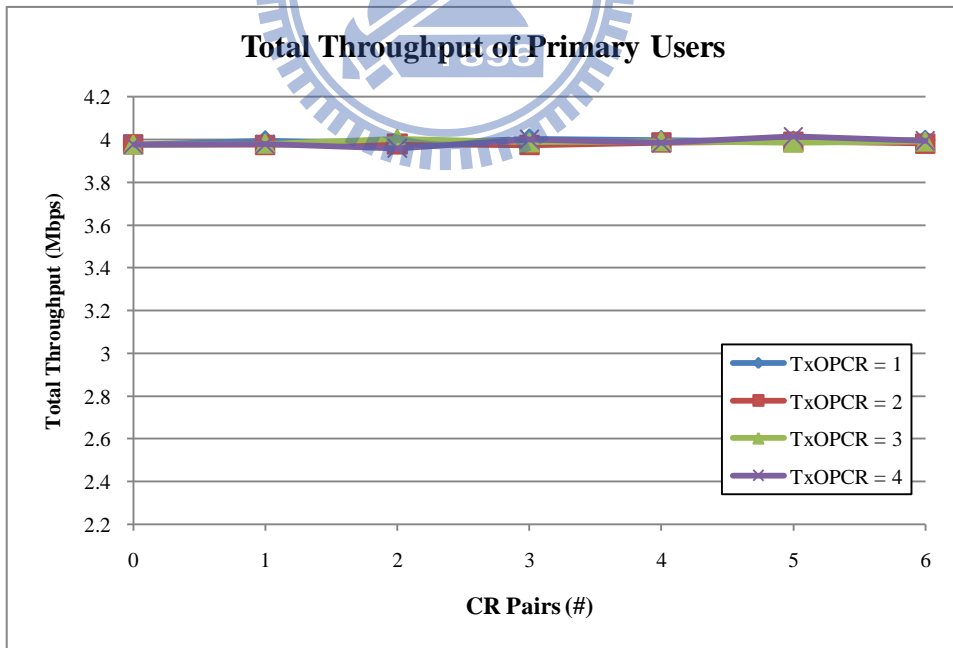


圖 39、原有之機制於無終端機隱匿問題之情境下，主要使用者的傳輸效能。

圖 38、39 則呈現了在終端機隱匿問題下，若無線感知網路之使用者利用 TCP 通訊協定來協助其進行資料傳輸之狀況。從中可以看到整體的傳輸效能雖然比傳輸樣式一的圖 32 對於不同的 $TxOP_{CR}$ 情況下的比較來得明顯，但整體傳輸效能僅落在 1.5 至 2.25 Mbps 內。而對於不同 $TxOP_{CR}$ 的情況下結果較容易分辨的原因主要在於 TCP 本身的機制，當有越多的使用者，其連續傳輸的次數越多的情況下，很有可能導致當某位使用者要進行傳輸時卻無法即時找到合適的頻段，此時在時間延遲變大的狀況下，是很有機會使得接收端無法即時回應 TCP-ACK，進而使得後續 TCP-DATA 無法立即送出進而減少傳輸量。這樣的狀況會導致其他人傳輸的機會增加，因此在 $TxOP_{CR}$ 的值不同的狀況下結果才不致於與最小 $TxOP_{CR}$ 的狀況一樣。而對於主要使用者的部份仍然未造成影響。

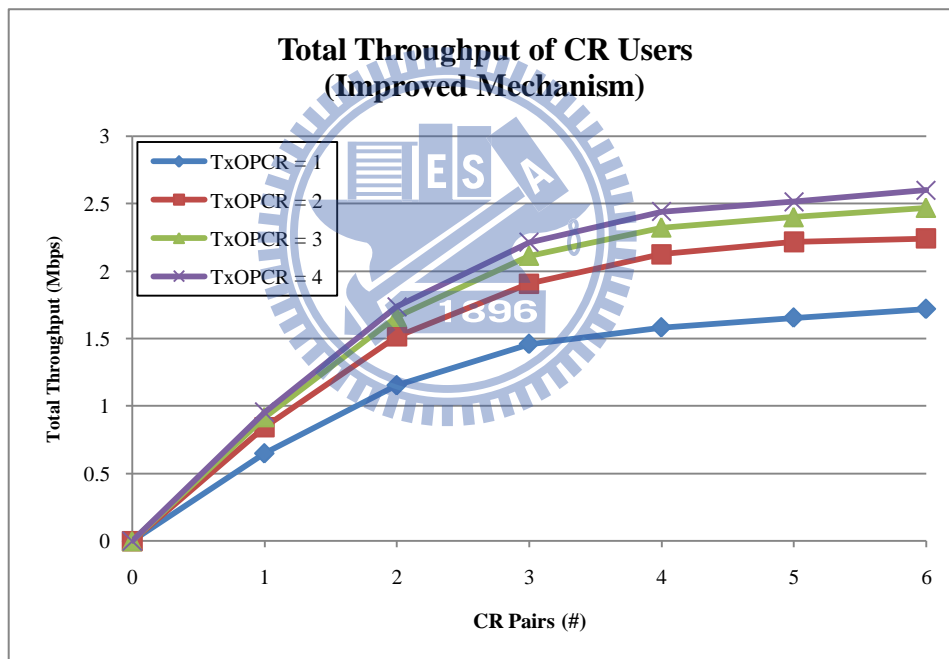


圖 40、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定的效能。

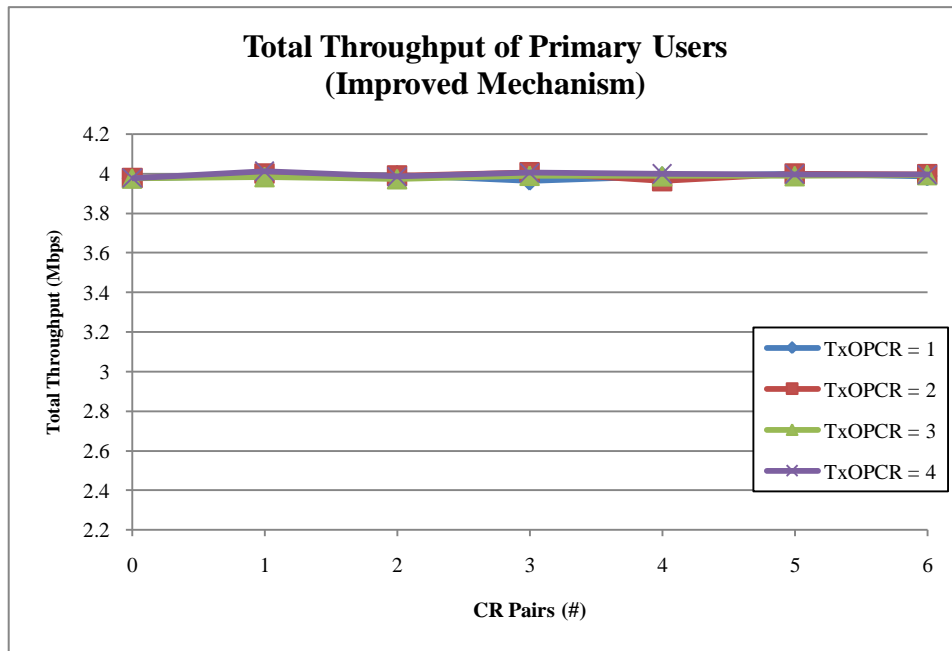


圖 41、改良之機制於無終端機隱匿問題之情境下，主要使用者的傳輸效能。

圖 40、41 與表 11 則展示了在終端機隱匿問題下，以改良機制進行無線感知網路實作後對於無線感知網路使用者與主要使用者的效能。由表 8 來進行兩者機制的比較，其中，在改良機制下，傳輸效能在 $TxOP_{CR} > 1$ 的情況有較為明顯的提升，例如在 $TxOP_{CR} = 4$ 且有 6 對無線感知網路使用者的狀況下，效能提升了 0.2 Mbps 左右，約上升了 9%。而整體的提升幅度則落在 2 ~ 30 %。但提升幅度之所以不如表 6 的主要原因應該是由於 TCP 傳送端與接收端無法在短時間內找到可以傳送資料與回傳 TCP-ACK 的頻段，使得在一秒內的個別傳送流量下降。而對於主要使用者的部份也如前面幾個情境一樣，仍未有所改變。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	0.619182	0.649157	4.84%	0.651074	0.844597	29.72%
2	1.065771	1.152897	8.17%	1.175949	1.516164	28.93%
3	1.333450	1.459410	9.45%	1.579856	1.906222	20.66%
4	1.499937	1.581370	5.43%	1.832405	2.125807	16.01%
5	1.616272	1.653590	2.31%	1.984077	2.216411	11.71%
6	1.676195	1.718147	2.50%	2.087723	2.240826	7.33%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	0	0	0.00%	0	0	0.00%
1	0.691602	0.918683	32.83%	0.713053	0.954609	33.88%
2	1.258489	1.661375	32.01%	1.327734	1.736743	30.80%
3	1.699493	2.113943	24.39%	1.767601	2.213310	25.22%
4	1.957895	2.323139	18.65%	2.082798	2.441061	17.20%
5	2.166563	2.399220	10.74%	2.247513	2.513088	11.82%
6	2.264252	2.468913	9.04%	2.375278	2.598715	9.41%

表 11、改良機制與原有機制對無線感知網路在終端機隱匿問題時，無線感知網路使用者在封包傳輸樣式二的傳輸效能比較。(單位：Mbps)

CR Nodes	TxOP = 1		TxOP = 2		TxOP=3		TxOP = 4	
	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism	Original Mechanism	Improved Mechanism
0	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733	3.976733
1	3.994340	3.987905	3.975666	4.000262	3.980601	3.983720	3.977956	4.011632
2	3.978746	3.991524	3.979377	3.989129	4.004092	3.973258	3.958967	3.985536
3	4.003144	3.964928	3.974206	4.005789	3.987115	3.990392	4.001684	4.004802
4	3.995841	3.984944	3.986326	3.962520	3.992248	3.987787	3.984036	3.998841
5	3.992603	3.997420	3.990392	3.999196	3.984983	3.988418	4.014712	3.997064
6	3.996669	3.987510	3.981391	3.996827	3.990747	3.993787	3.994261	3.994182

表 12、改良機制與原有機制在終端機隱匿問題時，主要使用者在封包傳輸樣式二的傳輸效能。(單位：Mbps)

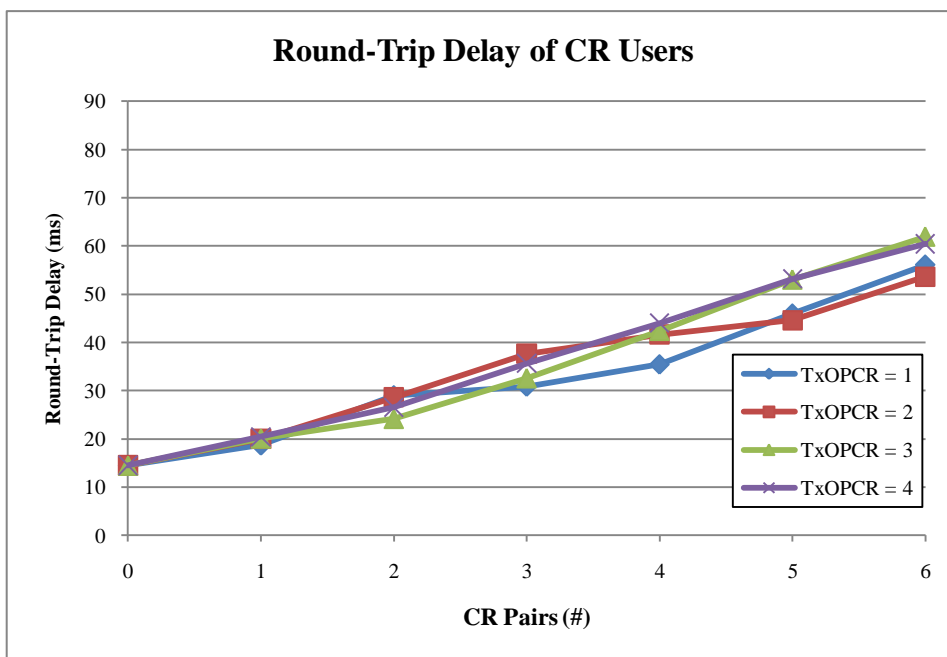


圖 42、原始之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時的 Round-Trip 延遲時間。

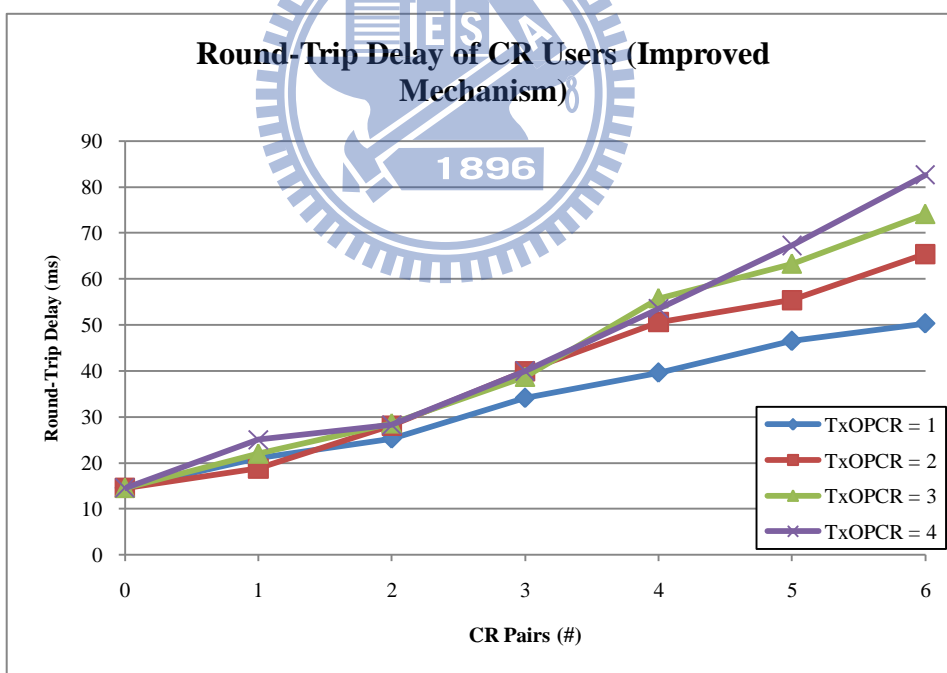


圖 43、改良之機制於無終端機隱匿問題之情境下，無線感知網路使用者以 TCP 做為傳輸協定時的 Round-Trip 延遲時間。

而在此情境下對於 Round-Trip 延遲時間則如圖 42、43 與表 13 所示。從中可以發現

對於延遲時間方面，雖然差距不是很大，但原始機制較改良機制來得低，原因主要是在上一段的分析中曾提到：改良機制的傳輸效能能在 $TxOP_{CR} > 1$ 的情況下，對比於原始機制，其傳輸效能有較為明顯的提升。在此情況下對於頻段的競爭與使用頻段的次數會較原始機制來的多，此會導致尋找可用頻段的找尋時間上升，進而使得延遲時間增加。

CR Nodes	TxOP _{CR} = 1			TxOP _{CR} = 2		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	18.705	21.069	-12.64%	20.021	18.759	6.30%
2	29.055	25.255	13.08%	28.631	28.100	1.85%
3	30.879	34.104	-10.44%	37.667	39.945	-6.05%
4	35.477	39.608	-11.65%	41.637	50.627	-21.59%
5	45.988	46.511	-1.14%	44.635	55.404	-24.13%
6	56.123	50.328	10.32%	53.638	65.452	-22.02%
CR Nodes	TxOP _{CR} = 3			TxOP _{CR} = 4		
	Original Mechanism	Improved Mechanism	Improvement	Original Mechanism	Improved Mechanism	Improvement
0	14.523	14.523	0.00%	14.523	14.523	0.00%
1	20.135	21.970	-9.11%	20.433	25.062	-22.66%
2	24.211	28.558	-17.96%	26.547	28.219	-6.30%
3	32.613	38.681	-18.61%	35.673	39.994	-12.11%
4	42.442	55.776	-31.42%	44.031	53.549	-21.62%
5	53.050	63.297	-19.31%	53.207	67.268	-26.43%
6	61.947	74.099	-19.62%	60.455	82.611	-36.65%

表 13、改良機制與原有機制對於無線感知網路在終端機隱匿問題下以 TCP 通訊協定來進行流量傳輸的 Round-Trip Delay 比較。(單位：ms)

6. 未來展望

在上述章節中介紹了如何在 NCTUns 這套網路模擬器進行無線感知網路相關模組之設計，並且進行改良。而在本章節中將介紹於本研究中發現的部份議題，這些議題在本篇論文中雖然考慮仍然不多，但其可作為日後進行更進一步研究與開發的一個方向：

- 終端機隱匿問題對主要使用者接收端的改善

在本篇論文中介紹了改良的機制，但是此機制主要是針對無線感知網路之使用者在終端機隱匿問題下的保護，透過此機制，確實能夠保護無線感知網路之使用者的傳輸流量，且於此同時，主要使用者所得到的流量也能夠盡可能不受干擾。但是無論是原始的機制，亦或是在改良的機制下，對於主要使用者在終端機隱匿問題下均未設計一套保護機制，這樣的情況會使得主要使用者在主要使用者傳輸端的訊號無法傳送給無線感知網路的使用者的情境下無法保障其傳輸效能。因此，在不改變主要使用者的機制下，如何有效的保護主要使用者將會是個極為重要的研究方向之一。

- 多天線支援情境下對於無線感知網路的效能增進與探討

在本篇論文中討論的主要是基於單一天線之情境為主，但在此情境下，頻段的感測也較為困難，無法在無線感知使用者位於控制頻段時進行偵測資料頻段的使用，使得無線感知網路使用者需要再花費額外的時間進行資料頻段的感測。但隨著技術的進步，使得在近來網路設備對於多天線的支援也愈來愈容易，因此在此趨勢之下，無線感知網路的設計上也隨之針對多天線來進行演算法的設計，目前即已存在數篇研究成果是基於多天線的假設。配合目前已有多個網路系統對於多天線傳輸進行規格的设计，如何為這些網路系統進行無線感知網路的设计以及如何在此網路下運用多天線提升效能將會是未來的發展方向之一。

- 適用於 Multi-hop 之無線感知網路之機制設計

在一般 Ad-hoc 的模式下，透過路由機制，如 ADOV，可以使得 Ad-hoc 設備能透過間接傳輸的方式使得封包能傳達到目的地。然而這樣的傳輸若需要過多額外的控制封包的協助，則可能會使得效能無法得以提升。然而這樣的功能卻是 Ad-hoc 模式中非常重要的一項功能，對於如何在 Multi-hop 的狀況下使得效能不受太多影響是非常重要的。

- 頻段感測機制之設計與實作

在本論文中提出了快速頻段快照的機制，但這樣的機制效率仍然不夠。然而在單一天線的假設之下，頻段感測與評估是很不容易進行的一項工作，雖然這個工作在設計上不容易，但這份的工作會協助無線感知網路使用者能更快找到適合使用的頻段，因此如何再進而發展出一套更有效率的頻段感測評估的方法是很重要的。此外，搭配於本章節第一點中提出的多天線之支援，即可使得在此部份的機制的設計上較為容易，且也將更有發展性。



7. 結論

在本篇論文中介紹了無線感知網路的基本概念，並且從研讀的相關研究中挑選一篇，使得無線感知網路的機制得以實作於 NCTUns 這套網路模擬器上。這套機制主要新增了 RTS_{CR} 、 CTS_{CR} 用來進行頻段的挑選，以及 RTI 用來使得主要使用者得以宣告其使用頻段之意願。而在實作中於主要以 802.11b Ad-hoc 模式實作了 MAC80211CR、Wphy_CR 這兩個模組，並且透過實作 Cognitive Radio Channel Control 的類別來解決了由於頻段的切換而導致的封包感測有誤之問題，且也協助了頻段的切換與挑選機制的使用。

接下來在第四章中，本篇論文對所實作的機制提出了改良的機制，這個改良的機制主要可分為：頻段挑選機制的改良、頻段感測機制的強化、在等待完主要使用者得以宣告其使用頻段之時間後的傳輸機制。頻段挑選的改良主要是透過快速頻段快照機制與讓傳送端進行頻段挑選，接收端負責頻段切換順序之決定，頻段感應機制的強化則是透過在頻段感測時改成以接收到的封包來做為是否能繼續傳送的依據，而最後所提的則是當在此頻段需要連續傳資料時均需要再次進行 RTS-CTS 握手動作，藉此降低因終端機隱匿問題而降低傳送效率。最後在第五章中提出了一系列的效能評估，證實了在這樣的改良方法中確實能有效的達到目的。

藉著這樣的模組的開發與設計，將可提供無線感知網路機制的研究人員有良好的平台可進行其機制的評估與改善，大幅縮減其進行複雜情境下的驗證所需之時間。

8. 參考文獻

- [1] FCC Spectrum Policy Task Force, “Report of the spectrum efficiency working group, “Technical Report 02-135, November 2002,” is available at http://www.fcc.gov/sptf/files/SEWGFfinalReport_1.pdf
- [2] Joseph Mitola III, and Gerald Q. Maguire, Jr. “Cognitive Radio: Making Software Radios More Personal,” in *IEEE Personal Communications*, August 1999.
- [3] NCTUns Network Simulator and Emulator is available at <http://nsl10.cs.nctu.edu.tw/>
- [4] Cognitive Radio Cognitive Network Simulator on NS2, is available at <http://stuweb.ee.mtu.edu/~ljialian/>
- [5] Jacek Dzikowski, and Cynthia Hood, “An agent-based simulation framework for cognitive radio studies,” in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, 2009.
- [6] Ian F. Akyildiz, Won-Yeol Lee, Mehmet C. Vuran, and Shantidev Mohanty, “A Survey on Spectrum Management in Cognitive Radio Networks,” in *IEEE Communication Magazine*, April 2008.
- [7] Tevfik Yücek, and Hüseyin Arslan, “A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications,” in *IEEE Communications Survey & Tutorials*, Vol 11, No. 1, First Quarter, 2009.
- [8] Jeonghoon Mo, Hoi-Sheung Wilson So, and Jean Walrand, “Comparison of Multi-Channel MAC Protocols,” in *IEEE Transaction on Mobile Computing*, Vol. 7, pp. 50-65, Jan 2008.

- [9] Hang Su, and Xi Zhang, "Channel-Hopping Based Single Transceiver MAC for Cognitive Radio Networks," in *Proc. IEEE Communication and Information Systems Security*, pp. 197-202, 2008
- [10] Alex Chia-Chun Hsu, David S. L. Wei, and C.-C. Jay Kuo, "A Cognitive MAC Protocol Using Statistical Channel Allocation for Wireless Ad-hoc Networks," in *Proc. of IEEE Wireless Communications & Networking Conference*, pp. 105-110, 2007.
- [11] Tak-Chon Lou, Hsi-Lu Chao, and Tzu-Jane Tsai, "On Synchronized Channel Sensing and Accessing for Cognitive Radio Users in IEEE 802.11 Wireless Networks," in *IEEE Personal, Indoor and Mobile Radio Communications*, pp.757-761, 2009.
- [12] Stefan Geirhofer, Lang Tong, and Brian M. Sadler, "Dynamic Spectrum Access in WLAN Channels: Empirical Model and Its Stochastic Analysis," in *Proceedings of the first international workshop on Technology and policy for accessing spectrum*, 2006.
- [13] T. Clancy, B. Walker, "Predictive Dynamic Spectrum Access," in *Proc. SDR Forum Technical Conference*, Orlando, Florida, U.S.A., Nov. 2006.
- [14] Serhan Yarkan, and Hüseyin Arslan, "Binary Time Series Approach to Spectrum Prediction for Cognitive Radio," in *Proc. IEEE Vehicular Technology Conference*, pp. 1563-1567, Sept.-Oct. 2007.
- [15] Shie-Yuan Wang, Chih-Liang Chou, Chih-Che Lin, and Chih-Hua Huang, "The Protocol Developer Manual for the NCTUns 6.0 Network Simulator and Emulator."