

在 IEEE 802.16 中截斷二元指數後退演算法之研究

The Study of Truncated Binary Exponential Backoff Algorithm in IEEE 802.16

朱國志

龍華科技大學電子工程研究所

kcchu@mail.lhu.edu.tw

鄭宇哲

龍華科技大學電子工程研究所

Jammi56@yahoo.com.tw

摘要

IEEE 802.16 網路由於具有高頻寬及涵蓋範圍大的特性，所以目前成為廣受注目的一項網路技術。在 IEEE 802.16 中用戶要傳送資料，必須要先傳送頻寬請求給基地台，再由基地台保留頻寬供其傳送資料。在用戶傳送頻寬請求時，可能會和其他用戶同時傳送的頻寬請求發生碰撞，此時在 IEEE 802.16 中是使用截斷二元指數後退演算法來解決碰撞。在 IEEE 802.16 中提供了參數，可以讓管理者依照網路的運作情況來動態調整截斷二元指數後退演算法的運作過程，使得整個演算法可以有比較好的效能。在本篇論文中，我們將利用模擬找出在不同的網路負載下，哪一種參數設定值可以讓整個系統得到較好的傳輸量及延遲時間，以提供給管理者做設定參數上的建議。

Abstract

Due to the characteristics of high data rates and wide coverage area, the IEEE 802.16 network becomes a popular technology for wireless network. In IEEE 802.16, when there are data in a Subscribe

Station (SS) need to be transmitted, the SS will transmit a request Packet Data Unit (PDU) to the Base Station (BS). After the BS receives the request PDU from a SS, it reserves bandwidth for that SS to send data PDU. Collision occurs when there is more than one SS transmitting request PDUs to the BS in the same time. In IEEE802.16, Truncated Binary Exponential Back-off (TBEB) algorithm is adapted to resolve collision. The performance of this algorithm will drop dramatically when the load is high. Therefore, in the IEEE 802.16 standard some parameters are reserved for future use. They can be used to improve the performance of the original algorithm. In this paper, we use simulations to find the parameter values that generate the fine throughput and delay time under different load. The present simulation results provide network administrators with the ability to choose the most appropriate parameter values for different network loads.

關鍵詞: IEEE 802.16、碰撞解決演算法、截斷二元指數後退演算法

Keyword: IEEE 802.16, Contention Resolution Algorithm, Truncated Binary Exponential Backoff Algorithm

一、簡介

近年來，無線網路蓬勃發展，使得使用無線網路的用戶快速增加。在人們享受無線網路帶給我們的便利性之後，人們也開始對無線網路有更多的要求了。大家都希望所使用的無線網路有很快的傳輸速率、很大的涵蓋範圍、支援服務品質保證並可以支援高速漫遊等許多功能。然而目前最多人使用的 IEEE 802.11 無線網路，因為規格上的限制，使得它有著基地台涵蓋範圍小的問題，也因此讓某些網路應用不能使用在 IEEE 802.11 上。為了要讓無線網路能有更大的涵蓋範圍，IEEE 802.16 [3] 也因此孕育而生。

IEEE 802.16 是屬於無線都會網路 [1, 2]，它的涵蓋的範圍可以廣達 50 公里，因此一個城市只要架構好一個基地台，使用者都可以在城市內的任意地點連接上網。其次，它的資料傳輸速率也高達 75Mbps，這個速率可以滿足目前大多數使用者的需求。除此之外，它也支援服務品質保證，這項特點讓 IEEE 802.16 網路很適合傳輸多媒體的應用，如：影片及語音服務。最後 IEEE 802.16 也支援高速漫遊，因此使用者可以在移動的車上使用網路，在不同的城市中漫遊。因為 IEEE 802.16 具有上述許多的優點，所以近幾年來成為許多國家積極投入發展的一項網路技術。許多人士都看好未來 IEEE 802.16 將取代目前的 ADSL 及 Cable Network 成為家中最後一哩(last mile)所使用的網路技術[8]。對於一些有線網路基礎建設發展還沒很成熟的發展中國家，IEEE 802.16 更可以協助他們快

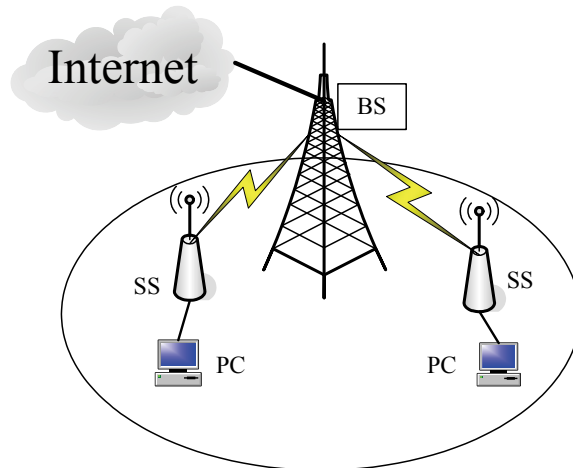


圖 1. IEEE 802.16 架構圖

速佈建整個網路架構，讓網路快速的普及。

圖 1 是 IEEE 802.16 網路的架構圖。整個網路架構中 BS (Base Station)端連接著 Internet，而用戶的電腦則連接到 SS (Subscriber Station)，BS 和 SS 之間則是依靠無線電波訊號來傳輸資料。IEEE 802.16 是屬於集中管理式的網路架構，整個網路是由 BS 管理著所有用戶的 SS。因此當用戶有資料要傳送時，SS 得先傳送頻寬需求給 BS，再經過 BS 排程後，BS 會保留頻寬供 SS 傳送該筆資料。因為 SS 傳遞資料所使用的頻寬，都是 BS 所事先預留的，所以在 IEEE 802.16 網路中傳遞資料時並不會有碰撞發生。但是，在同一時間內如果有超過一台的 SS 向 BS 提出頻寬請求，這個頻寬請求的訊息則會造成碰撞 (collision) 的發生。

在 IEEE 802.16 中是採用了截斷二元指數後退 (Truncated Binary Exponential Backoff, TBEB) 演算法來解決碰撞問題。TBEB 演算法在之前的研究中發現它的效能並不是很好 [9]。為了改善這個問題，IEEE 802.16 中的 TBEB 演算法裡提供了幾個參數，管理者可以根據網路的負載，動態的改變參數的設定值，藉此可以改變 TBEB 演算法的運作過程，以提升演算法

解決碰撞能力。因此在這篇論文中，我們將探討在不同的負載下，該如何設定這些參數，才能使得整個系統可以獲得比較好的效能。

接下來的論文架構如下。第二章將要對於 IEEE 802.16 的實體層及媒體存取層的運作方式作一個簡介。第三章中我們將針對 TBEB 演算法的運作方式做一個詳細的介紹。在第四章中，我們將會利用模擬找出在不同負載下，哪一種參數設定可以讓 TBEB 演算法能有較好的效能。最後第五章則是我們的結論與未來工作。

二、IEEE 802.16 簡介

IEEE 802.16 [3]在 2001 年 11 月最早的版本被通過，在這個版本主要是使用了 10~66MHz 的無線頻譜，傳輸距離最遠可達 5km，最大速率可以達 134Mbps。在這個版本只支援 LOS (Line of Sight)及點對點的傳輸方式。在 2003 年則通過了 802.16a [4]的版本，在這個版本中則支援了 NLOS (Non-Line of Sight)的傳輸方式，並且提供了單點對多點的傳輸方式。在 802.16a 的版本中是使用了 2-11GHz 的頻譜範圍，平均傳輸距離可達 6~9 公里，最高速率則可到達到 75Mbps。2004 年則推出了 IEEE 802.16-2004 [5]版本，這個版本整合了前面的版本，並也加入了次頻道的架構以及多重輸入輸出(MIMO)天線的概念，成為目前最多廠商遵循的一項標準。此外，為了讓 IEEE 802.16 提供支援行動工作站的能力，IEEE 802.16e-2005 [6]在 2005 年推出，這個標準可以支援行動工作站以 120km/h 的速度移動。IEEE 802.16e-2005 採用了 OFDMA 的調變方式，運作在 2~6MHz 的範圍，如果以一

個 5MHz 的頻道可以提供 15Mbps 的頻寬。

IEEE 802.16 的傳輸頻道分為上行頻道(Uplink, UL)及下行頻道(Downlink, DL)。SS 是利用上行頻道將資料傳送至 BS，而 BS 則是利用下行頻道傳送資料給 SS。BS 為了要控制所有 SS 的運作，因此會定期傳送 UCD (Uplink Channel Descriptor)、DCD (Downlink Channel Descriptor)、UL-MAP (Uplink Map) 及 DL-MAP (Downlink Map)等訊息給 SS。其中 UCD 是描述上行頻道實體層運作參數的訊息，而 DCD 則是描述下行頻道實體層運作的參數。SS 可以根據 UCD 及 DCD 的訊息調整自己實體層的調變及編碼方式。UL-MAP 則是描述媒體存取層中，上行頻道頻寬的分配結果，DL-MAP 則是描述下行頻道頻寬的分配結果。SS 可以觀看 UL-MAP 得知自己何時可以傳送資料，且觀看 DL-MAP 知道下行頻道何時有資料傳送給自己。

在 IEEE 802.16 中，SS 傳送資料的過程如下：首先 SS 查看 BS 傳送來的 UL-MAP，看看接下來上行頻道頻寬分配的結果為何。其次，SS 則選擇在 UL-MAP 中指定給 SS 傳送頻寬請求的時間，傳送頻寬請求給 BS。之後，SS 則會再查看之後送來的 UL-MAP，看看 BS 是否有為其保留頻寬傳送資料。如果 BS 有為其保留頻寬，SS 則會在 BS 指定的時槽傳送資料。如果 SS 沒有看到 BS 為其保留頻寬，那則代表剛剛 SS 傳送的頻寬請求發生了碰撞(collision)，對於可能發生的碰撞 IEEE 802.16 則是採用 TBEB 演算法來解決這個問題。下一個章節，我們將介紹 TBEB 演算法。

三、截斷二元指數後退演算法

IEEE 802.16 採用 TBEB 演算法[3]，來解決不同 SS 同時發送頻寬請求時，所可能產生的碰撞問題。圖 2 為 TBEB 演算法的流程圖。首先我們先介紹圖中的參數。RBS (Request_Backoff_Start) 的作用是決定初始的後退視窗值，而 RBE (Request_Backoff_End) 則是決定了最大的後退視窗值，RBS 及 RBE 值在 IEEE 802.16 規格書中定義的範圍均為 0~15。當網路負載低的時候，RBS 及 RBE 要設定的小一點，免得造成資料在佇列中過多的等待延遲；而當網路在負載重時，RBS 及 RBE 則要設定的大一些，好減低碰撞發生的機率。BE (Backoff Exponential)值代表的是此次傳輸所需使用的後退視窗指數值。NB (Number of Backoff)則代表目前傳輸的這筆資料已經重送了幾次，NB 的初始值被設為 1。Request Retries 則代表了最大的重傳次數，在 IEEE 802.16 中的預設值為 16。

當SS有新資料要傳送時，首先SS會先把BE值設為RBS。之後，SS從0到 $2^{BE}-1$ 中隨機選取一個數字並等待該數字個競爭傳輸機會 (contention transmission opportunities)經過後才可以傳送頻寬請求 (request PDU)。如果傳送的頻寬請求發生了碰撞，則SS會進行以下步驟。首先SS會將NB值加1。之後如果SS的BE值小於RBE值，則BE值加1，否則維持原BE值。最後則重複前面隨機選取數字及等待傳送的步驟。整個碰撞解決演算法將會持續直到頻寬請求傳送成功或重傳超過 Request Retries 為止，則此時資料會被丟棄。

這裡我們舉例說明，假設 RBS=5 且 RBE=10。SS 剛開始要傳送資料時，會先

把 BE 設為 5，且隨機從 $0 \sim (2^5-1)$ 中選取一個數值。假設 SS 選到的數字為 7，那就代表 SS 要等待 7 個競爭傳輸機會經過後，才可以傳送頻寬請求。如果傳送發生碰撞，則 NB 值要加上 1，且 BE 值也要增加 1 變成 6，之後重新進行以上請求頻寬動作。

TBEB 演算法在負載重時，效能的表現很不理想。為了追求更好的效能，在 IEEE 802.16 中，可以讓 BS 依照目前網路的流量大小，利用 UCD message 動態調整所有 SS 的 RBS 及 RBE 值。在下一個章節，我們將探討在不同負載下該如何設定 RBS 及 RBE 值以使得系統獲得較好的效能。

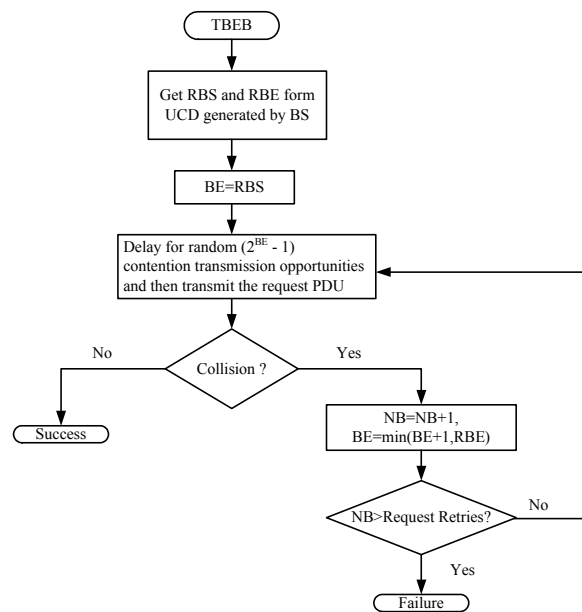


圖 2.TBEB 演算法流程圖

四、模擬與討論

在這一個章節中，我們將利用模擬來探討在不同負載下，RBS 及 RBE 該做如何的設定，才能讓系統得到較佳的 throughput、delay time 及 jitter。而在之後的討論中為了更清楚的表示每一個 RBS 和 RBE 的組合，因此我們採用了(RBS, RBE)的表示法，代表 BS 此次選擇的 RBS 及 RBE 值。例如: (0, 15)則代表此次 BS 選擇的是 RBS=0 及 RBE=15 的組合。

(一) 系統模擬架構

在這篇論文中，我們採用了 QualNet [7] 這一套模擬軟體來模擬 IEEE 802.16-2004 網路的運作。在模擬中整個網路有 1 台 BS 及 20 台 SS。網路的頻道大小為 20MHz，而每一個 Frame 的間隔為 20ms。SS 每次傳送的封包大小為 512Bytes，如果傳送不成功，重送的最大次數為 16 次。此外，每台 SS 的輸出佇列大小為 50000Bytes，並採用了 FIFO 的排程方法。至於其他相關參數的設定，如表 1 所示。

在模擬中，我們依照 SS 每次傳送資料的間隔，分成了三個實驗組別，如表 2 所示。在實驗 1 中每台 SS 每隔 0.01s 傳送 1 筆資料，此時系統是屬於輕負載的情況。實驗 2 則是每台 SS 間隔 0.005s 傳送 1 筆資料，這時整個系統將處於中負載的情況下。最後，在實驗 3 中每台 SS 傳遞資料的間隔則為 0.001s，在這種設定下，整個系統將會是在重負載。

在 802.16-2004 規格書中訂定 RBS 和 RBE 的範圍為 0~15。依照規格書的設定所能形成的 RBS 及 RBE 的組別將會非常的多。為了簡化整個設定，我們從 RBS 和

RBE 0~15 的範圍中選出四個數字分別為 0, 5, 10 及 15，並利用這四個數字排列出 RBS 及 RBE 的組合共 6 組分別為(0,5)、(0,10)、(0,15)、(5,10)、(5,15)及(10,15)。在這個章節中，我們將探討在三種不同的實驗下，這 6 種 RBS 及 RBE 的組合，何者可以讓整個系統有較好的 throughput、delay time 及 jitter。

表 1 模擬參數設定表

Parameters	Value
BS	1
SS	20
Channel Bandwidth	20MHz
Frame Duration	20ms
TDD Downlink Duration	10ms
DCD Broadcast Interval	5s
UCD Broadcast Interval	5s
Transmit/Receive Transition Gap (TTG)	10 μ s
Receive/Transmit Transition Gap (RTG)	10 μ s
SS Transition Gap (SSTG)	4 μ s
Packet Size	512Bytes
Request Retries	16
Queue Size	50000Bytes
Scheduling Algorithm	FIFO
Simulation Time	15s

表 2 實驗組別

實驗	Packet Interval
1	0.01s
2	0.005s
3	0.001s

(二) 模擬結果與討論

以下我們將分別根據不同的負載之下，TBEB 演算法使用了 6 種 RBS 和 RBE 組合做模擬，並對於模擬的結果做一個討論。

1. 實驗 1：輕負載情況

在實驗 1 中，每台 SS 以 0.01s 的間隔傳送資料。首先，我們先探討不同組合的 throughput。此處的 throughput 是指在模擬時間中，所有 SS 傳送資料成功的位元總和。圖 3 是不同組合在輕負載下的 throughput。其中 (0, 5) 這組的 throughput 為 8.46Mbps，是所有組別中最高的。這是因為在輕負載的情況，每台 SS 傳送的頻寬需求並不會遭遇到太多的碰撞便能傳送成功，因此 RBS 及 RBE 的值都不需訂的很大。而(10, 15)這組的 Throughput 為 7.8Mbps 則是最低的，這是因為這一組開始便使用了很大後退視窗值，因此 SS 要傳送頻寬需求前都必須要等待很久的時間，所以得到的 throughput 值會最低。

至於(0, 10)與(0, 15)這兩組會比(5, 10)及(5, 15)這兩組好的原因，是因為在輕負載的情況下，過多的起始後退視窗值，只會延誤傳遞的時間，造成 throughput 的下降。最後，對於(0, 10)與(0, 15)這兩組的數據接近相同的原因，是因為在輕負載時，碰撞的次數會很少，

因此 BE 值並不會有太多的成長，因此這兩組的數據也就差不多。至於(5, 10)及(5, 15)這兩組的數據也是因為同樣的原因。

圖 4 所示在低負載的情況下，不同組別的平均佇列等待時間。所謂的平均佇列等待時間是從資料抵達 SS 後，到 SS 成功傳輸資料，中間資料在佇列等待的平均間隔時間。由圖 4，我們可以發覺到除了(10, 15)這一組的平均佇列等待時間較長外，其他組別都是趨近於 0。這主要的原因是因為(10, 15)這一組的初始後退視窗值過大，造成過多的等待時間。

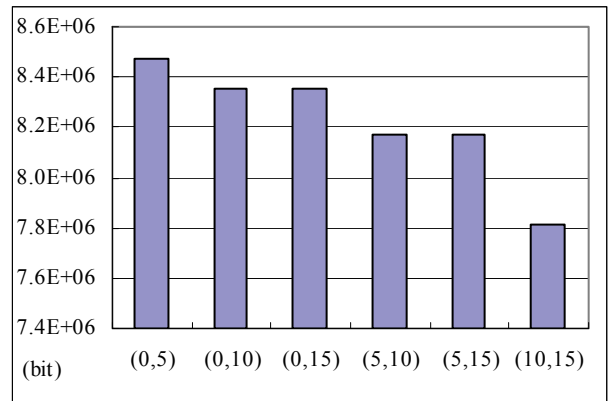


圖 3 不同 RBS 及 RBE 組合在輕負載下的 throughput

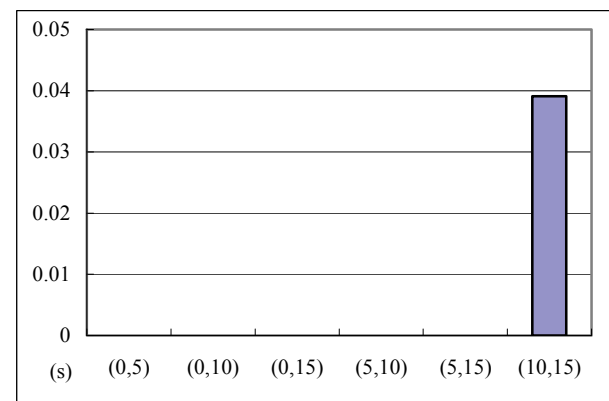


圖 4 不同 RBS 及 RBE 組合在輕負載下的平均佇列等待時間

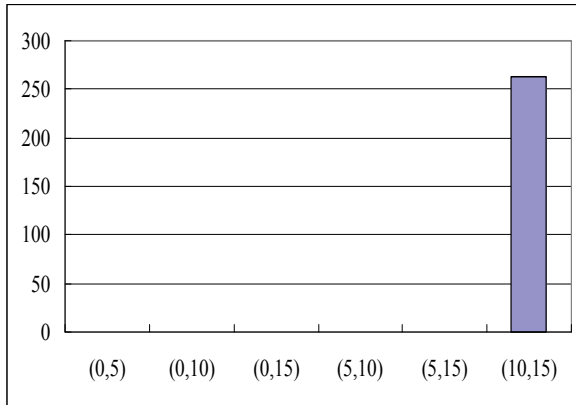


圖 5 不同 RBS 及 RBE 組合在輕負載下的封包丟棄量

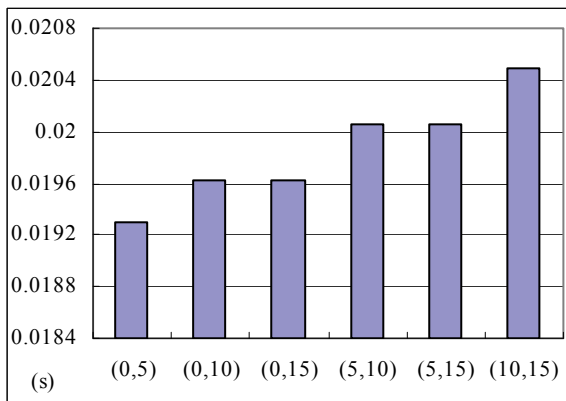


圖 6 不同 RBS 及 RBE 組合在輕負載下的平均 jitter

圖 5 是不同 RBS 及 RBE 組合在輕負載下的封包丟棄量。此處的封包丟棄量是指 SS 因為輸出佇列滿了而被丟棄的所有封包數量。會發生輸出佇列滿出來的原因，主要是因為 SS 遲遲未能成功取得頻寬傳送資料，在輸入的資料持續不斷進入下，便造成輸出佇列裝不下的現象發生。由圖中可看到只有(10, 15)這一組發生封包被丟棄的現象，主因也是過長的初始後退視窗值。由這一張圖，也間接的說明了在圖 3 中(10,15)這組 throughput 較低的原因。

圖 6 為不同 RBS 及 RBE 組合在輕負載下的平均 jitter 值。這裡的 jitter 是指資料在網路上傳輸的延遲時間差異值。這個值越大代表 SS 成功傳送資料的延遲時間值越不穩定。圖中我們可以發現(0, 5)這組的 jitter 最小，最高則為(10, 15)。這代表著(0, 5)這組可以提供 SS 在輕負載下較一致的延遲時間值。

總合以上的討論，我們可以發現在輕負載下，BS 選擇(0, 5)這一組將會是一個比較好的選擇。

2. 實驗 2：中負載情況

在實驗 2 中，每台 SS 以 0.005s 的間隔傳送資料。圖 7 顯示了在中負載流量之下，各組的 throughput。圖中的 throughput 的表現大概可以分成三組。首先表現較好的組別為(0, 5)、(0, 10)及(0, 15)，其次則為(5,10)及(5, 15)，最差的則為(10,15)這組。會有這樣表現的原因在於中負載時，頻寬請求的碰撞尚未太多，所以 RBS=0 比 RBS=5 讓 SS 可以更快傳送資料，因此可以獲得較好的 throughput。至於為何(0, 5)、(0, 10)及(0, 15)的表現會差不多，和實驗 1 的原因是一樣的，因為在中負載時碰撞還是不多，所以這三個組別的 BE 並未成長太多，因此也就表現的差不多。至於(10, 15)同樣因為起始後退視窗太大，造成過多延遲而影響到 throughput。

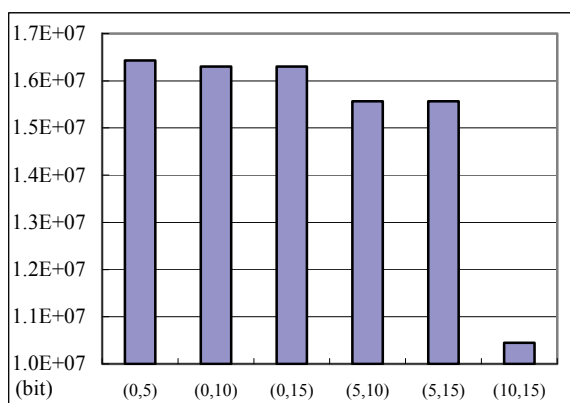


圖 7 不同 RBS 及 RBE 組合在中負載下的 throughput

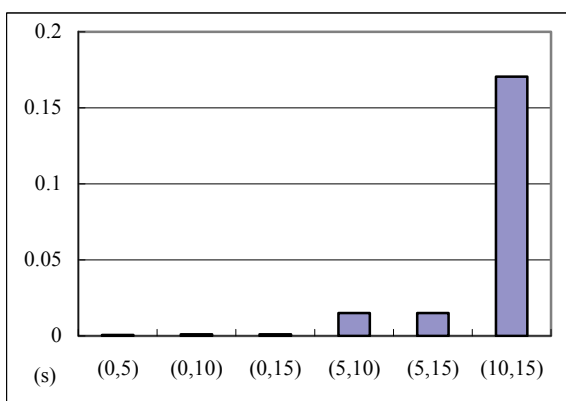


圖 8 不同 RBS 及 RBE 組合在中負載下的平均佇列等待時間

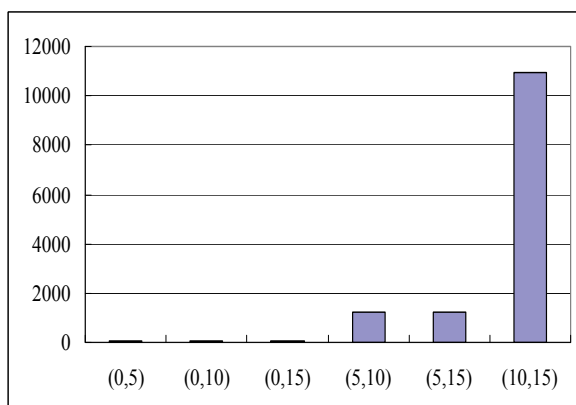


圖 9 不同 RBS 及 RBE 組合在中負載下的封包丟棄量

在中負載的情況下，不同組別的平均佇列等待時間顯示在圖 8。我們可以發現和 throughput 一樣，數據的表現也一樣分成三組。(0, 5)、(0, 10)及(0, 15)因為初始後退視窗值最小，在碰撞次數還沒很多時，等待時間最小。而(5, 10)及(5, 15)則居次。至於(10, 15)這組數據高達 170ms，資料停留在佇列的時間最長。主要的原因還是因為過大的初始後退視窗值。

圖 9 是各種組合在中負載下的封包丟棄量。由這張圖我們可以看出在中負載時，各種組合都已經有不同數量的資料被丟棄。整個看來，(0, 5)、(0, 10)及(0, 15)這三組表現較好。(10, 15)這組則表現的最不理想。

圖 10 為不同 RBS 及 RBE 組合在中負載下的平均 jitter 值。(0, 5)、(0, 10)、(0, 15)、(5, 10)及(5, 15)這五組的數據都趨近於 10ms，延遲時間變化量的差距幾乎一樣。至於(10, 15)這組的表現則明顯的不如其他組別。

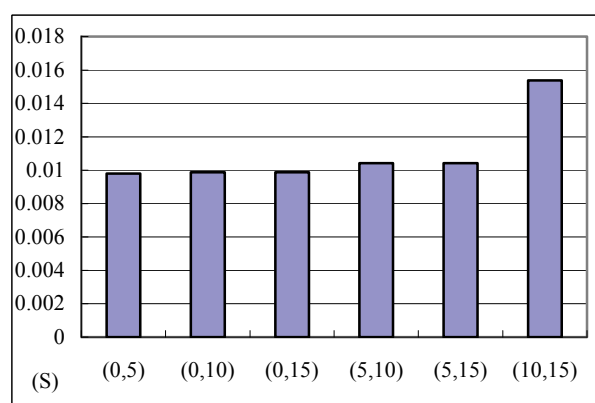


圖 10 不同 RBS 及 RBE 組合在中負載下的平均 jitter

根據以上的幾個模擬結果，我們建議在中負載的情況下，(RBS, RBE)可以考慮從(0, 5)、(0, 10)及(0, 15)中選擇一組做為設定值，所得到的效能會比其他三組好。

3. 實驗 3：重負載情況

在實驗 3 中，我們將讓每台 SS 以 0.001s 的間隔傳送資料，以探討六個組別在重負載下的效能表現。

圖 11 為不同組別在重負載下的 throughput。圖中的 throughput 的表現大概可以分成四組。首先表現較好的組別為(5, 10)及(5, 15)。其次為(0,10)和(0, 15)，這組的表現和前一組差距不大。第三則為(0, 5)，最差的則為(10,15)。系統在高負載時，SS 傳送頻寬請求會發生大量的碰撞，因此此時 RBS 最好大一點。(0, 5)這組的 RBS 則因為太小，導致每次傳送都很容易有碰撞發生，進而影響到 throughput。但 RBS 太大則又會讓每次傳送等太久，也將會降低 throughput，如(10, 15)。

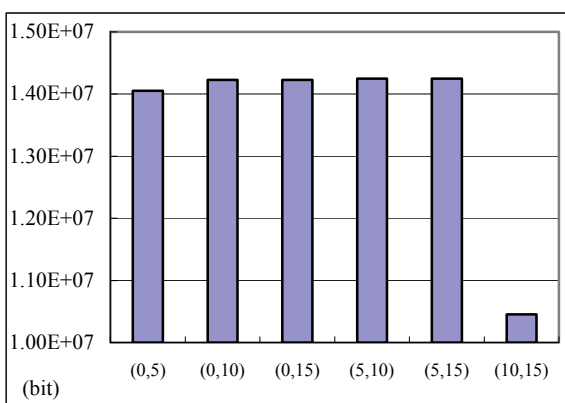


圖 11 不同 RBS 及 RBE 組合在重負載下的 throughput

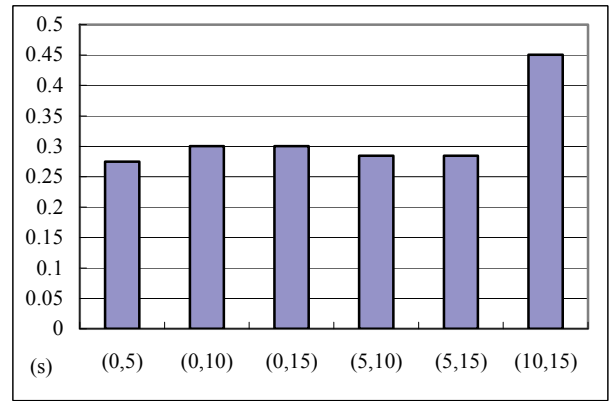


圖 12 不同 RBS 及 RBE 組合在重負載下的平均佇列等待時間

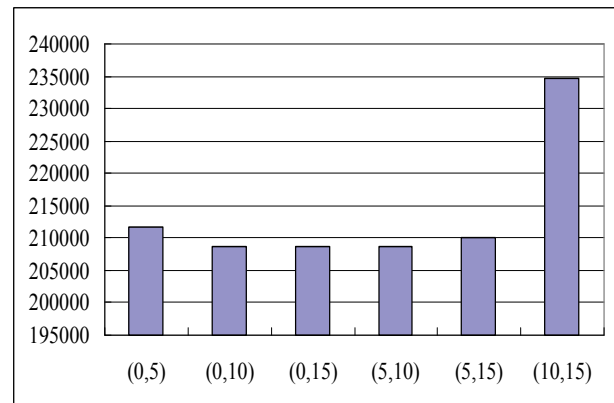


圖 13 不同 RBS 及 RBE 組合在重負載下的封包丟棄量

圖 12 是在重負載下不同組別的平均佇列等待時間。我們可以發現(0, 5)、(0, 10)、(0, 15)、(5, 10)及(5, 15)這幾組表現的差距不大。就只有(10, 15)這一組所顯示出的數據較大，代表有較多的平均佇列等待時間。

圖 13 為在高負載下各個組別的封包丟棄總量。我們可以發現六個組別都有大量的封包被丟棄，這是因為 SS 產生資料的速度過快，但資料卻不容易傳送成功，使得佇列長度持續累加，最後輸出佇列裝不下，導致新來的封包被丟棄。除了(10, 15)外，其他(0, 5)、(0, 10)、(0, 15)、(5, 10)及(5, 15)這幾組表現的差

距不是太大。至於為何(0, 10)及(0, 15)這兩組的結果會幾乎相同的原因，在於(0,15)這一組，當其 BE 值還未成長超過 10 時之前，便已發生佇列滿出來的現象，造成封包遺失，這也可以說明在圖 11 中兩者 throughput 相近的原因。最後對於(5, 15)這組的 RBE 值雖然比(5,10)這組大，但封包丟棄量還是比較大的原因如下：因為(5, 15)的 RBE 值較大，因此遇到碰撞時會有更大的後退視窗值，這也就是說在輸出佇列內的資料要多等一些時間才能傳送，也因如此使得佇列的長度很容易累積，造成封包被丟棄。

圖 14 則為不同 RBS 和 RBE 在重負載下的 jitter 表現。除了(10, 15)這組表現的比較不好外，其他各組的 jitter 其實差距都不大。

從以上的討論，我們發現在重負載的情況下(5, 10)這組可以說是所有組別中表現最好的，而其他(0, 5)、(0, 10)、(0, 15)及(5, 15)這四組的表現也都不會差距太大。至於(10, 15)的表現則就比較不理想了。

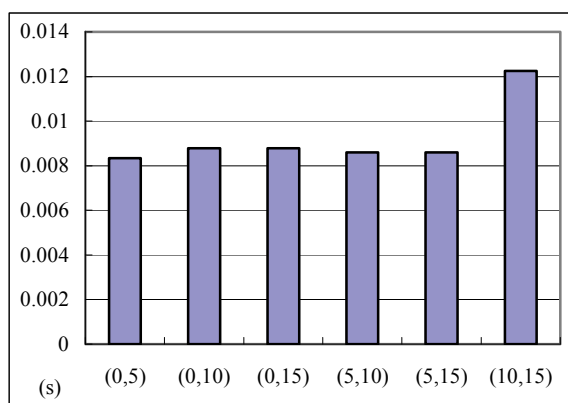


圖 14 不同 RBS 及 RBE 組合在重負載下的平均 jitter

最後，總結以上的三個實驗的模擬結果。我們可以發現當系統在輕負載的情況下時，BS 最好將(RBS, RBE)設定為(0, 5)。此外，在中負載時 BS 則可以從(0, 5)、(0, 10)及(0, 15) 三組中任選一組，都是不錯的選擇。而在重負載時(5, 10)則是所有組別中的首選。

五、結論與未來工作

近年來無線網路快速的興起，改變了我們的生活。在眾多新興的無線網路技術中，IEEE 802.16 由於具有高頻寬及涵蓋距離遠的特性，因此成為近年來頗受重視的網路技術之一。未來還有很大的機會取代目前的 ADSL 及 Cable Network 的地位。IEEE 802.16 是屬於集中管理的網路架構，SS 要取得頻寬傳送資料都要先向 BS 申請。當有超過一台的 SS 向 BS 提出頻寬請求時，此時則會有碰撞的發生。在 IEEE 802.16 中則是採用 TBEB 演算法來解決碰撞的問題。IEEE 802.16 針對 TBEB 演算法提供了部份參數，讓 BS 的管理者可以藉由設定這些參數改變 SS 端 TBEB 演算法的運作。藉此希望讓 SS 端的 TBEB 演算法能隨著網路負載的不同做出不同的調整，以得到較好的效能。

在這篇論文中我們設計了三個實驗，並在模擬中比較了 throughput、平均佇列等待時間、封包丟棄量及 jitter 等項目。試圖找出在不同的負載下，何種 RBS 及 RBE 的設定值可以讓系統得到較好的效能。最後，我們得到了在輕負載的情況下時，(0, 5)會是較好的設定。而在中負載時，(0, 5)、(0, 10)及(0, 15) 三組的效能因為差不多，所以可以從其中選擇

一組設定值。最候，是在重負載時，我們則選擇(5, 10)。

對於未來的工作，由於在這篇論文中，我們為了減低(RBS, RBE)設定值的複雜程度，而簡化了(RBS, RBE)的組別數。未來我們希望能夠在這篇論文的基礎上，能做在更多不同負載下所有(RBS, RBE)組別的模擬。以期能得到在各種負載下，所有合適的 RBS 及 RBE 的設定值。其次，我們將利用這些結果設計一個運作在 BS 端的演算法，這個演算法可以動態監測網路流量並找出該流量所對應的合適 RBS 及 RBE 值。之後由 BS 傳送給 SS，使得每一台 SS 上的 TBEB 演算法都能夠有效率的運作。

誌謝

本論文感謝國科會給予經費補助，計畫補助編號為 NSC-96-2221-E-262-004。

六、參考文獻

- [1] A. Ghosh, D.R. Wolter, J.G. Andrews, and R. Chen, "Broadband Wireless Access with WiMax/802.16: Current Performance Benchmarks and Future Potential," IEEE Comm. Magazine, vol. 43, no. 2, pp. 129-136, Feb. 2005.
- [2] C. Eklund, R.B. Marks, K.L. Stanwood, and S. Wang, "IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access," IEEE Comm. Magazine, vol. 40, no. 6, pp. 98-107, June 2002.
- [3] IEEE 802.16-2001, "IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air interface for fixed broadband wireless access systems," Apr. 8. 2002.
- [4] IEEE 802.16a-2003, "IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air interface for fixed broadband wireless access systems – Amendment 2: Medium access control modifications and addition physical layer specifications for 2-11 GHz," Apr. 1. 2003.
- [5] IEEE 802.16-2004, "IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air interface for fixed broadband wireless access systems," Oct. 2004.
- [6] IEEE 802.16e-2005, "IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air interface for fixed and mobile broadband wireless access systems, amendment for physical and medium access control layers for combined fixed and mobile operation in licensed bands," Feb. 2006.
- [7] Scalable Network Technologies, Inc., <http://www.scalable-networks.com/>
- [8] S. M. Cherry, "The wireless last mile," IEEE Spectrum, Vol. 40, Issue 9, pp. 18 – 22, Sep 2003.
- [9] Y. D. Lin, C. Y. Huang, and W. M. Yin "Allocation and Scheduling Algorithm for IEEE 802.14 and MCNS in Hybrid Fiber Coaxial Networks," IEEE Trans. on Broadcasting vol. 44, no. 4, pp. 427-35, Dec. 1998.