在衰變通道上有效率 polling 演算法

王三元 [1]、蔡福文 [2]

義守大學 資訊工程系 [1,2]

E-mail: sywang@isu.edu.tw[1] fuwem@yahoo.com.tw[2]

摘要

藍芽(Bluetooth)是一種新興的短距離、低成本無線通訊技術，其最小的網路單位是 piconet；piconet 是由一個 master 以及最多七個 slaves 組合而成的。然而在 piconet 中每對 master 與 salve 的 traffic 速率是非對稱的。此外受到地形地物的影響之下，造成路徑遺失(path loss)和衰變(fading)的現象，進而導致在傳送資料的時候，無法達到通道最佳效能。

根據 traffic 速率以及 fading 來製作一個有效率的 polling 演算法。透過動態選擇合適的封包型態和動態調整 polling 的間隔來達到通道最佳效能。

最後我們在 Rayleigh 衰變通道環境中模擬傳送效能，結果顯示在衰變現象愈嚴重且 piconet traffic load 愈大時我們演算法的產能會比 PMP 演算法還要高。

第一章 概論

藍芽(Bluetooth)是一種短距離、低成本、低耗電量的無線通訊技術。藍芽使用的頻帶是在免執照的 ISM (Industrial， Scientific and Medical) 2.4GHz 頻段上，在 2.4GHz 的頻道上會分成 79 個不同的頻率通道 (frequency channel)，每一個的通道的頻寬是 1MHz，使用跳頻的方式去與另外的藍芽裝置建立一個連結，跳頻的速率是每秒 1600 次，每個時槽(time slot)的長度是 625μs。其最小的網路單位是 piconet (圖 1.1)；piconet 是由一個 master 以及最多七個 active slaves 組合而成的，當空間中裝置超過八個的時候，便會形成兩個以上的 piconet，此時 piconet 會透過 bridge 裝置去連結，而形成一個 scatternet (圖 1.2)。藍芽是使用分時雙工方式，master 是在偶數時槽傳送資料，slave 是在奇數時槽傳送資料，且藍芽使用以 master 為集權的機制，因此 master 使用輪循的方式來與 slave 互相傳送資料。藍芽的連接方式分成 SCO (Synchronous Connection Oriented)、ACL (Asynchronous Connection-Less)兩種，SCO 是在傳送即時性(real-time)的資料，如語音資料，而且是一對一的方式在傳送，每個 slave 會支援 2 個或 3 個 SCO links；ACL 是在傳送非同步的資料，它在 master 與 slave 之間是一對多的 link。藍芽的封包是由 Access Code、Header、Payload 三個部份所組成的，封包分成 DH_x 封包與 DM_x 封包兩種類型，X 是封包的長度，分為 1、3、5 slots，DH_x、DM_x 的差異是在 payload 的部份，DH_x 是沒有用任何的糾錯方法，而 DM_x 是用了 2/3 FEC 的糾錯方法。

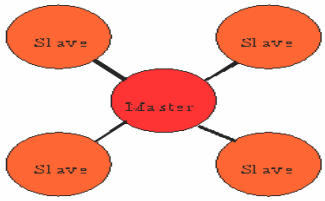

圖 1.1 piconet 圖

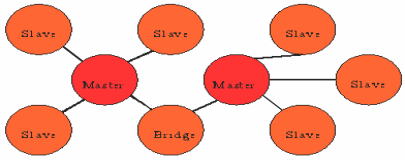

圖 1.2 scatternet 圖

藍芽是使用 master 集權的排程機制，並且使用分時雙工(TDD)的方式，master 是在偶數時槽(time slots)傳送資料給 salve，slave 則是在奇數時槽傳送資料給 master。由於藍芽是使用集權機制，因此 master polling 的時間安排是一項十分重要的事情，假如時間安排不佳的時候便會造成功率(power)損耗、頻道的浪費等，因此 polling 演算法是十分重要的。

現有 polling 演算法主要分成三大類型：第一個類型是根據 traffic 來製作藍芽 polling 演算法[2][7][9][10]，這類型的缺點是無法根據通道狀況去選擇合適的封包型態，優點是在沒考慮通道衰變狀況下可以傳出最佳的效能並且可以降低電量的浪費；第二個類型是根據電量的損耗來製作藍芽 polling 演算法

[3][4][5][8]，這類型的缺點是無法大量提升通道的效能，優點是降低電量浪費，第三個類型是根據頻道衰減程度來製作 polling 演算法[1][6][11]，缺點是由於沒有配對出 master 和 slave 兩者最佳的傳送封包，導致傳送時會有頻道的浪費，優點是隨著通道衰變狀況可以傳送出最佳的封包。

在無線通道上傳送資料會遭遇到多種不同因素的干擾，以及訊號本身隨著傳送距離導致訊號衰減，因此我們在傳送資料時必須考慮通道的衰變狀況來選擇合適的封包來傳送資料。

藍芽裝置在傳送資料時，訊號會受到地形地物的影響，導致訊號的消耗，進而使得傳輸效能的不佳，然而封包形式對於通道衰變狀況會有不同的效能，因此根據衰變的狀況來選定合適的封包形式來傳送資料，提升網路的效能。藍芽是使用 master 集權的排程機制，因此 master 的 traffic rate 與 slave 的 traffic rate 是不同的，所以在傳送資料的時候必須根據 traffic rate 來選擇封包形式，提升傳送資料時網路的效能。因此根據 traffic rate 以及考量通道的 fading 現象來設計一個有效率的 polling 演算法，提高網路效能。

由於 Bluetooth 裝置日漸普及，設計其 piconet 上之排程演算法是重要的研究課題。我們研發目標是設計出一個根據通道的衰變狀況以及 master 與 slave 的 traffic 速率而做一個有效率的演算法。這個演算法是以不改變系統原有設定而出發點，最終的目的是要讓藍芽排程演算法在現實環境使用時都可以有最佳的效能，而且在理想的環境中也是有最佳效能的。

研究目標如下：

- 在衰變通道環境之下，根據 master 與 slave 的 traffic 速率來設計一個有效率的 polling 演算法
- 發展一個效能模擬平台

在 Rayleigh 衰變通道環境中模擬傳送效能，結果我們的演算法可以根據 fading 現象選擇適當的封包型態提升傳送效能。本文的貢獻是在現實環境使用時都可以有較佳的傳送效能。

本篇 paper 的章節安排編制如下：在第二章的部分是訊號傳送特性簡介，在第三章的部分是詳細描述本篇論文所提的演算法，在第四章是模擬效能評估，最後在第五章是本篇論文的結論與未來的工作。

第二章訊號傳送特性簡介

在無線通道上傳送資料會遭遇到多種不同因素的干擾，以及訊號本身隨著傳送距離或通道品質而導致衰減，如此一來對於通訊品質及網路傳輸效能有很大的影響，因此我們就探討無線通道上傳送資料時，訊號的衰減和干擾影響，以及訊號 path-loss 模式。

2.1.1 訊號的干擾

訊號在大氣中傳送資料時會因為環境而造成反射、繞射、折射等現象產生，而使的訊號重疊。訊號因為反射或其他現象會使接收端收到多個信號，此情形稱為多重路徑(Multi-path)。當接收端在同一時間因為接收到不同相位信號重疊，便會使得訊號強度重疊增強或是互相抵消而衰減，而導致訊號的失真。

除了訊號本身影響外，訊號也會受到其他外界因素而干擾損壞訊號。例如在相同的頻道上如有其他無線裝置也使用此頻道再進行傳送訊號時，此時兩個訊號會因為在相同頻道中使用而互相碰撞受到干擾。

另外家中有一些使用在 ISM 頻段的電器用品也會干擾到訊號。當電器用品運作時會產生電磁波，而電磁波的頻率剛好跟訊號傳送的通道頻率相同時，訊號就會遭受到干擾。

除了以上方面會產生干擾之外，如果鄰近的頻道太過接近也會受到影響，因此藍芽系統才會使用跳頻方式傳送資料來避免固定頻道上傳送的干擾。

2.1.2 訊號的衰減

在無線網路訊號傳送上，訊號傳送範圍會受到地形地物的影響，因此受到環境影響的衰減模式可以分成 free space、two-ray 以及 shadowing，接下來分別介紹這三種訊號 path-loss 模式。除此之外，訊號在大氣中傳送資料時會有反射、繞射、折射等現象產生，而使的訊號重疊。訊號因為反射或其他現象會使接收端收到多個信號，此情形稱為多重路徑(Multi-path)。當接收端在同一時間因為接收到不同相位信號重疊，便會使得訊號強度重疊增強或是互相抵消而衰減，而導致訊號的失真。而多重路徑的衰變主要分成兩種類型：Rayleigh Fading 以及 Ricean Fading。

2.1.2.1 Free Space Fading

Free Space Fading 是最基本的訊號損耗模式(收送兩端無阻隔)，接收訊號強度和距離是成平方比的關係，當天線是理想的天線時，在 Free Space Fading 模式下傳送出來的訊號和接收到的訊號之間的關係如公式 2.1。

$$\frac{P_t}{P_r} = \frac{(4\mathrm{d}p)^2}{l^2} = \frac{(4\mathrm{fd}p)^2}{c^2} \qquad (2.1)$$

在這裡的 P_r 指的是接收的功率，P_t 是傳送訊號的功率，d 是傳送端和接收端之間的距離，l 是訊號波長，$l = c/f$，c 是光速(3×10^8)，f 是訊號傳送的頻率。

對於其他非理想天線，我們必須要考慮天線功率增益，那麼傳送出來的訊號和接收到的訊號之間的關係如公式 2.2。

$$\frac{P_t}{P_r} = G_t G_r \frac{(4\mathrm{d}p)^2}{l^2} = G_t G_r \frac{(4\mathrm{fd}p)^2}{c^2} \qquad (2.2)$$

在這裡的 G_t 是傳送端天線的功率增益，G_r 是接收端天線的功率增益。

2.1.2.2 Two-Ray Fading

Free Space Fading 距離和功率的關係式是無法滿足實際環境，在實際環境情況下，訊號從傳送端到達接收端的路徑是會有很多不同方向。傳送端和接收端的天線高度是影響訊號接收時的強度。根據 Two-Ray Fading 定義傳送出來的訊號和接收到的訊號之間的關係如公式 2.3。

$$\frac{P_t}{P_r} = G_t G_r \frac{h_t^2 h_r^2}{d^4} \qquad (2.3)$$

一般在 Two-Ray Fading 下，我們通常把功率與距離關係式以 dB 表示，如公式 2.4：

$$10\log(P_r) = 10\log(P_0) - 10a\log(d) \qquad (2.4)$$

這裡 P_r 是在 d 公尺所接收到的訊號強度，P_0 是在 1 公尺所接收到的訊號強度。

2.1.2.3 Shadow Fading

當傳送端傳送出一個訊號時，即使在相同的距離接收到的訊號強度會隨著接收端的周圍環境不同而有不同的訊號強度。我們可以從公式 2.4 的之傳送端與接收端距離 d 公尺所得訊號強度的平均值，可是實際上所接收到的訊號強度會在這平均值上下變動，這個變動值以 X 來表示，X 在這裡是一個隨機變數，他所顯現出來的分佈是一個 log-normally 分佈，shadowing fading 的 path-loss 如下：

$$L_p = 10\log(P_t) - 10\log(P_0) + 10b\log(d) + X \qquad (2.5)$$

2.1.2.4 Rayleigh Fading

訊號從發送端經由通道到接收端之間會受到許多干擾，如受到地形地物的影響，而導致訊號出現許多傳送路徑，所以訊號經過不同路徑到達接收端的時間不同，進而使訊號的振幅與相位出現不同，如此一來造成相互干擾，這樣的情況就是多重路徑衰變(multipath fading)，在傳送端與接收端之間只有許多間接的傳送訊號，但是沒有直接的傳送訊號，並且當訊號振幅大小呈現 Rayleigh 分布，那麼我們把這種多重路徑衰變稱為 Rayleigh fading。Rayleigh fading 振幅的分布圖可以經由公式 2.6 得到。

$$f_{ray} = \frac{r}{s^2}\exp\left(-\frac{r^2}{2s^2}\right), \quad r \ge 0 \qquad (2.6)$$

2.1.2.5 Ricean Fading

在傳送端與接收端之間有一個直接傳送的訊號而且有許多間接傳送的訊號，因此 Ricean Fading 和 Rayleigh Fading 兩者最大的不同市在於有無直接傳送的訊號。Ricean Fading 振幅的分布圖可以經由公式 2.7 得到。

$$f_{ric}(r) = \frac{r}{s^2}\exp\left(-\frac{r^2+K^2}{2s^2}\right) I_0\left(\frac{Kr}{s^2}\right), \quad r \ge 0, K \ge 0 \qquad (2.7)$$

第三章 TRA:在衰變通道上根據 traffic rate 的 polling 演算法

本論文是在 fading channel 環境及考量 master 與 slave 兩者的 traffic rate 不同之下的藍芽網路(piconet)中，設計一個有效率的 polling 演算法。這個演算法會根據衰變通道的情況來調整傳送的封包型態，並且根據 traffic rate 以及所選出來的封包型態來安排下次 polling 的時間，接下來我將在第一節中介紹各種封包型態在衰變通道中傳送出來的效能以及在第二節介紹如何根據 traffic rate 以及封包型態來選定下次 polling 的時間，然後在第三節介紹本演算法的整個流程。

第一節 各種封包型態在衰變通道中傳送出來的效能

我們要如何知道衰變通道的狀況，知道通道狀況的方法大致有兩種，第一種是根據接收到封包的位元錯誤率，第二種是根據接收到傳送信號的電量強度以及干擾信號的電量強度的比率，這個比率通常是稱之為 SNR 值。由於 SNR 值是比較容易可以得知，因此本演算法是根據 SNR 值的方式，接下來我將分別在 4.1.1 介紹位元錯誤率與封包之間的效能以及在 4.1.2 介紹 SNR 值與封包之間的效能。

4.1.1 位元錯誤率與封包之間的效能

圖 4.1 [7]是位元錯誤率跟各種封包型態的效能圖，從圖 4.1 我們發現當錯誤率變高那便會導致所有的封包效能變低，而且也發現封包與錯誤率有四個區間：第一個區間是當錯誤率低於 0.00019 時，DH5 封包型態是所有封包型態中可以有最佳的傳送效能；第二個區間是當錯誤率在 0.00019 跟 0.0065 之間時，DM5 的封包型態會是所有封包型態中可以有最佳的傳送效能；第三個區間是當錯誤率在 0.0065 跟 0.019 之間時，DM3 封包是所有封包型態中擁有最佳的傳送效能；第四個區間是當錯誤率大於 0.019 時，DM1 封包是所有封包型態擁有最佳的傳送效能。

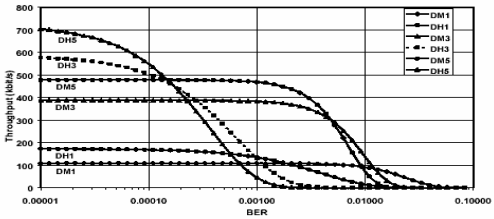

圖 4.1 錯誤率與封包型態的效能 [12]

4.1.2 SNR 值與封包之間的效能

圖 4.2 [12]是 SNR 值跟各種封包型態的效能圖，從這個圖我發現 SNR 可以分成四個區間：第一個區間是 SNR < 9.67 dB，當 SNR 在這個區間時 DM1 封包的效能是最佳的；第二個區間是 9.67 dB ≤ SNR < 10.79 dB，當 SNR 在這個區間時 DM3 封包的效能是最佳的；第三個區間是 10.79 dB ≤ SNR < 13.52 dB，當 SNR 在這個區間時 DM5 封包的效能是最佳的；第四個區間是 13.52 dB ≤ SNR，當 SNR 在這個區間時 DH5 封包的效能是最佳的。

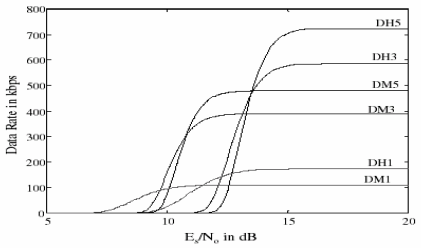

圖 4.2 SNR 值對於各種封包型態的效能 [12]

第二節 根據 traffic rate 和選擇的封包型態來選定下次 polling 的時間

藍芽 ACL 的封包型態分成 DH1、DH3、DH5、DM1、DM3、DM5 六種，下面分別介紹六種封包最大可以傳送資料量，DH1 的 payload 部分最大傳送量為 27 bytes，DM1 最大傳送量為 17 bytes，DH3 最大傳送量為 183 bytes，DM3 最大傳送量為 121 bytes，DH5 最大傳送量為 339 bytes，DM5 最大傳送量為 224 bytes。接下來我們將利用例子來敘述如何根據 traffic rate 和選定的封包型態來安排下次 polling 的時間，例如：master 的 traffic rate 為 20 Bytes/slot，slave 的 traffic rate 為 10 Byte/slot，而根據 SNR 值選擇出來的封包形式為 DH5，由於 DH5 的 payload 最多可以放 339 Bytes，那麼我們的方法是等到資料量至少達到 339 Bytes 時才會去做 polling 的動作，資料量要達到 339 Bytes 的時間為 339 / 20 = 16.95，由於 master 只有在偶數時槽才會做傳送的動作，那麼 master 會經過 18 slots 才去做 polling 的動作。

第三節 TRA 演算法流程

在前面兩節我們已經敘述通道狀況對於封包的效能影響以及如何去安排 polling 的時機，而本演算法是架構在這兩個因素之下所發展出來的。

當 piconet 內裝所有的裝置都連線成功並且 master 與 slave 角色確定之後，這個時候我們是設定為 t = 0 秒，在這個時候 master 與 slave 都是不知道彼此 traffic rate，而且也不知道通道狀況的，因此我們 TRA 演算法為了解決這個問題，我們首先會在 t = 0 的時候 master 會根據使用 Round-Robin 的方式來和所有的 slave 去取得每個 salve 的 traffic rate，而使用的封包形式是 DM1，我們使用 DM1 來取得資料的原因有兩個，第一個原因是我們傳送的資料只有 traffic rate 的資料，所以使用一個時槽的封包型態是比較佳；第二個原因是因為避免通道狀況不佳而造成資料錯誤。這個做法如圖 4.3 所示。

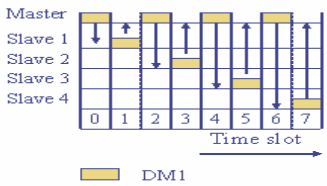

圖 4.3 TRA 演算法 RR 做法

Step 1. Initial，master 使用 Round-Robin 的方式去取得每個 salve 的 traffic rate 以及由 slave 傳送的封包知道通道 SNR 值，而傳送的封包形式是 DM1

Step 2. master 會根據通道 SNR 值來選擇 master 的傳送封包型態以及告知 slave 要選擇何種封包型態來傳送 slave 要傳給 master 的資料

Step 3. 根據 traffic rate 以及選定的封包形式來計算何時封包資料量可以充滿，假設充滿封包資料量所需要的時間為 k slot，那麼下次 polling slave 的時間是第 t + k slot

Step 4. 判斷每個 slave 的 polling 時間是否有重疊，並且調整重疊的 polling 時間

Step 5. 當時間為 polling 的時間，那麼傳送選定的封包和接收 slave 傳送的封包，並且根據 slave 傳送封包所得到的 SNR 值來更新 SNR 值而且 go to Step 2.

接下來我將透過流程圖來細部介紹演算法的操作，圖 4.4 是在介紹當 master 的 traffic rate(TRM) ≥ slave 的 traffic rate(TRS)時 Master 如何選定傳送封包，圖 4.5 是介紹 TRM< TRS 時如何去選定傳送封包，圖 4.6 是在介紹當 TRS>TRM 時 Slave 如何選定封包來傳送，圖 4.7 是介紹 slave 在 TRS ≤ TRM 時如何去選定傳送封包，圖 4.8 是介紹當 polling 時間重疊時解決的流程圖。

SNR (dB)	SNR<9.67	9.67<=SNR <10.79	10.79<=SNR <13.52	13.52<=SNR
傳送封包型態	DM1	DM3	DM5	DH5

圖 4.4 當 master 的 traffic rate(TRM) ≥ slave 的 traffic rate(TRS)時 Master 如何選定傳送封包

Data (Bytes) / SNR (dB)	Data<121	121<=Data <224	224<=Data	Data<183	183<=Data< 224	183<=Data< 339	339<=Data
SNR<9.67	DM1	DM1	DM1	DM1	DM1	DM1	DM1
9.67<= SNR<10.79	DM1	DM3	DM3				
10.79<= SNR<11.3	DM1	DM3	DM5				
11.3<= SNR<12.9	DH1	DM3	DM5				
12.9<= SNR<13.52			DM5	DH1	DH3		
SNR>= 13.52				DH1		DH3	DH5

圖 4.5 master 在 TRS > TRM 時如何選擇封包

SNR (dB)	SNR<9.67	9.67<=SNR <10.79	10.79<=SNR <13.52	13.52<=SNR
傳送封包型態	DM1	DM3	DM5	DH5

圖 4.6 在介紹當 TRS>TRM 時 Slave 如何選定封包來傳送

Data (Bytes) / SNR (dB)	Data<121	121<=Data <224	224<=Data	Data<183	183<=Data< 224	183<=Data< 339	339<=Data
SNR<9.67	DM1	DM1	DM1	DM1	DM1	DM1	DM1
9.67<= SNR<10.79	DM1	DM3	DM3				
10.79<= SNR<11.3	DM1	DM3	DM5				
11.3<= SNR<12.9	DH1	DM3	DM5				
12.9<= SNR<13.52			DM5	DH1	DH3		
SNR>= 13.52				DH1		DH3	DH5

圖 4.7 slave 在 TRS ≤ TRM 時如何選擇封包

	TRMa>TRMb	TRMa<TRMb	TRMa=TRMb
Polling開始時間相同	Slave a先poll	Slave b先poll	根據Slave ID 來做polling
Polling開始時間不同	Polling開始最早的先Poll	Polling開始最早的先Poll	Polling開始最早的先Poll

圖 4.8 解決 polling 時間重疊的流程圖

第四章模擬效能評估

敘述完 TRA 演算法後，接下來我們將 TRA 演算法模擬出來，看看這個方法是否可行，以及這個方法的效能為何，是否如我們預期一樣提升了頻道傳送效能。本模擬實驗是在 NS2 下的 Bluehoc 2.0 來做的。

第一節模擬環境

我將會使用二種 Traffic type 來比較 PMP[10]跟 TRA 效能：模擬環境參數表如:表 5.1。

表 5.1 模擬環境參數

模擬參數	數值
Traffic type	CBR 、FTP
Slave個數	7
TRM/TRS	1/7 、1/4 、1/3 、1、3、4、7
Piconet traffic load (bytes/slot)	14、28、42、56、70、84
transmission Power (mw)	100
Fading	Rayleigh Fading
Master與Slave的距離	1 、3 、5 、6 m
Noise (mw)	0.0095

第二節模擬結果

在 master 與 slave 的距離為 3 公尺且每個 node 傳送的應用層是利用 CBR 來當傳送工具的頻道效能，我們可以由圖 5.1(TRM/TRS: 1、3、4、7)以及圖 5.2(TRM/TRS: 1/3、1/4、1/7)來看，當 traffic load 變大時，我們的演算法跟 PMP 的方法的 Piconet throughput 是都會增加。當 piconet traffic load (PTL) 高於 42(bytes/slot)時，我們演算法產能會比 PMP 演算法還要高，這是因為我們演算法 polling 間隔時間與 PMP 演算法的間隔時間差距變小，而且我們演算法每次 polling 的封包資料都是滿的，所以我們演算法產能會比較高，而當 PTL 低於 42(bytes/slot)時，我們演算法產能跟 PMP 演算法效能是差不多的。

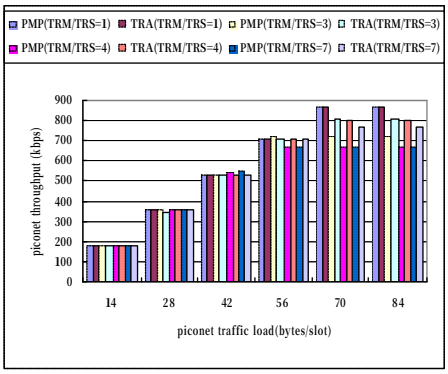

圖 5.1 距離 3 公尺、CBR 且 TRM >= TRS，PTL 與 piconet throughput 的關係圖

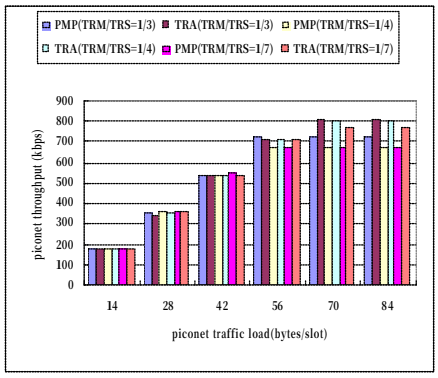

圖 5.2 距離 3 公尺、CBR 且 TRM <TRS，PTL 與 piconet throughput 的關係圖

在 master 與 slave 的距離為 3 公尺且每個 node 傳送的應用層是利用 FTP 來當傳送工具的頻道效能，我們可以由圖 5.3(TRM/TRS: 1、3、4、7)以及圖 5.4(TRM/TRS: 1/3、1/4、1/7)來看，當 traffic load 變大時，我們的演算法跟 PMP 的方法的 Piconet throughput 是都會增加。當 TRM/TRS=1/4、1/3、3、4 時，我們產能會比 PMP 演算法還要低，這是因為我們配對封包型態有用到 DH1 的封包而導致產能會比 PMP 演算法還要低，但是當 TRM/TRS=1/7 或是 7 而且 PTL > 42 (bytes/slot)時，我們演算法的產能會比 PMP 還要高，這是因為我們配對封包類型有使用到 DH3，因此我們產能才會比 PMP 高。

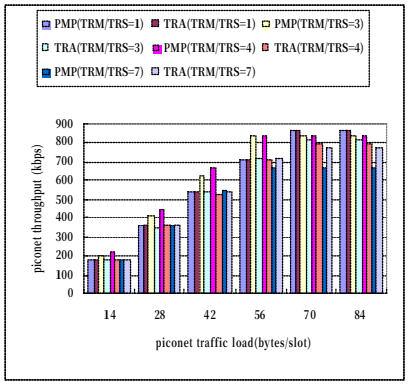

圖 5.3 距離 3 公尺、FTP 且 TRM >= TRS，PTL 與 piconet throughput 的關係圖

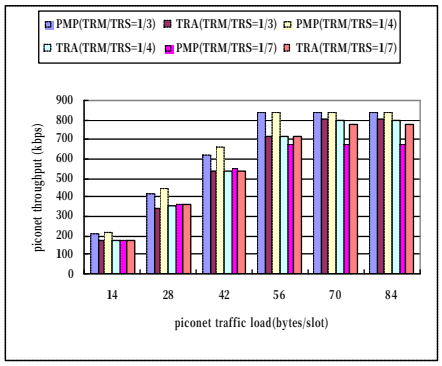

圖 5.4 距離 3 公尺、FTP 且 TRM < TRS，PTL 與 piconet throughput 的關係圖

在 master 與 slave 的距離為 5 公尺，且每個 node 傳送的應用層是利用 CBR 來當傳送工具的頻道效能，我們可以由圖 5.5(TRM/TRS: 1、3、4、7)以及圖 5.6(TRM/TRS: 1/3、1/4、1/7)來看，當 PTL 低於 42(bytes/slot)時，我們的演算法產能都是會比 PMP 低。這是因為 PMP 演算法 PMP polling 的間隔時間比我們演算法的間隔時間還要短，因此才會導致 PMP 的產能比較高。而當 PTL 超過 42(bytes/slot) 以及 TRM/TRS 比值變大時，我們演算法的產能才會有高於 PMP 產能，這是因為我們 polling 間隔時間變小以及所使用得封包型態效能比較佳。

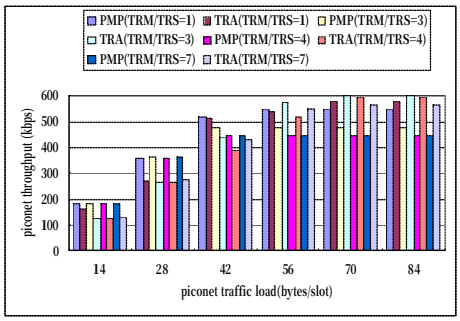

圖 5.5 距離 5 公尺、CBR 且 TRM >= TRS，PTL 與 piconet throughput 的關係圖

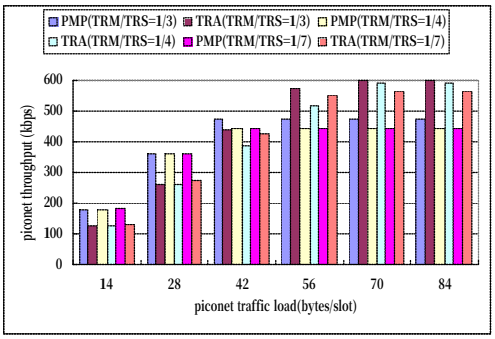

圖 5.6 距離 5 公尺、CBR 且 TRM < TRS，PTL 與 piconet throughput 的關係圖

在 master 與 slave 的距離為 5 公尺，而每個 node 傳送的應用層是利用 FTP 來當傳送工具的頻道效能，我們可以由圖 5.7(TRM/TRS: 1、3、4、7)以及圖 5.8(TRM/TRS: 1/3、1/4、1/7)來看，當 traffic load 變大時，我們的演算法跟 PMP 的方法的 Throughput 是都會增加，在 traffic load 比較低的時候，我們的演算法產能都會比較差，原因是我們演算法的 polling 間隔時間比較長，因此導致傳送次數比較少，進而使得 piconet throughput 會比較低。但是當 PTL 變大以及 TRM/TRS 比值變大時，我們的演算法產能會比 PMP 的產能好。

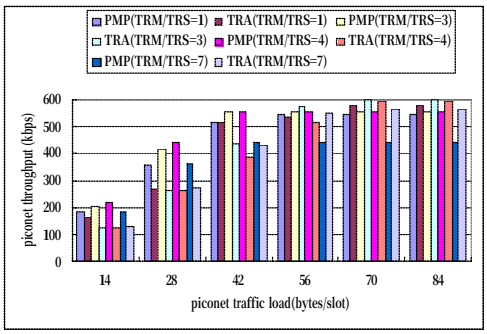

圖 5.7 距離 5 公尺、FTP 且 TRM >= TRS，PTL 與 piconet throughput 的關係圖

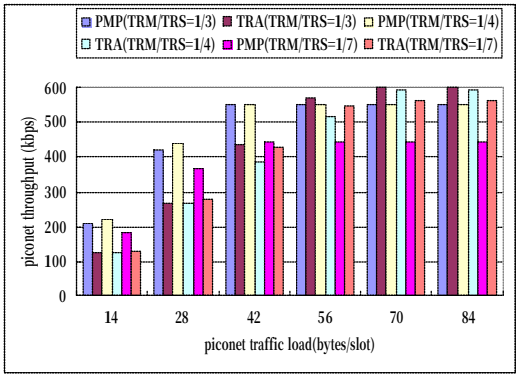

圖 5.8 距離 5 公尺、FTP 且 TRM < TRS，PTL 與 piconet throughput 的關係圖

在 master 與 slave 的距離為 6 公尺且每個 node 傳送的應用層是利用 CBR 來當傳送工具的頻道效能，我們可以由圖 5.9(TRM/TRS: 1、3、4、7)以及圖 5.10(TRM/TRS: 1/3、1/4、1/7)來看，當 TRM/TRS 值是 1/7 或是 7 且 PTL >= 42(bytes/slot)時，我們要產能會比 PMP 演算法還要高一點，我們產能會比較高的原因是因為 traffic rate 低的裝置有時候有用到 DM3 的封包來傳送資料，因此我們產能會比 PMP 演算法高一點，但是 PTL < 42(bytes/slot)時，我們要產能會比 PMP 演算法還要低一點，這是因為我們 polling 間隔時間比較長一點，因此才會導致產能低一點。除了 TRM/TRS 為 1/7、7，我們演算法的產能都會比 PMP 演算法還要高，這是因為我們所選擇的封包型態是根據通道狀況而選定的。

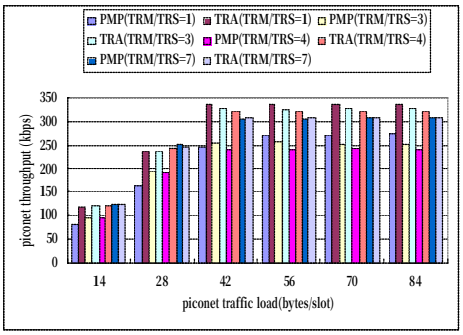

圖 5.9 距離 6 公尺、CBR 且 TRM >= TRS，PTL 與 piconet throughput 的關係圖

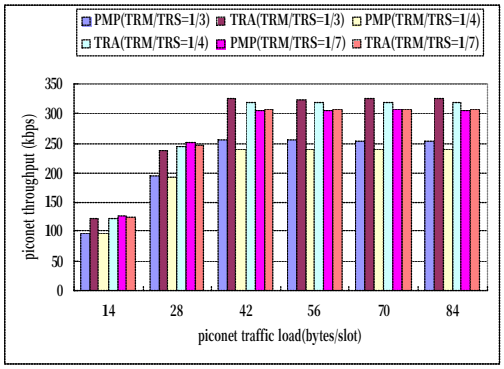

圖 5.10 距離 6 公尺、CBR 且 TRM < TRS，PTL 與 piconet throughput 的關係圖

在 master 與 slave 的距離為 6 公尺且每個 node 傳送的應用層是利用 FTP 來當傳送工具的頻道效能，我們可以由圖 5.11 (TRM/TRS: 1、3、4、7)以及圖 5.12 (TRM/TRS: 1/3、1/4、1/7)來看，當 traffic load 變大時，我們的演算法跟 PMP 的方法的 Piconet throughput 是都會增加，當 TRM/TRS 值是 1/7 或是 7 且 PTL >= 42(bytes/slot)時，我們要產能會比 PMP 演算法還要高一點，我們產能會比較高的原因是因為 traffic rate 低的裝置有時候有用到 DM3 的封包來傳送資料，因此我們產能會比 PMP 演算法高一點，但是 PTL < 42(bytes/slot)時，我們要產能會比 PMP 演算法還要低一點，這是因為我們 polling 間隔時間比較長一點，因此才會導致產能低一點。除了 TRM/TRS 為 1/7、7，我們演算法的產能都會比 PMP 演算法還要高，這是因為我們所選擇的封包型態是根據通道狀況而選定的。

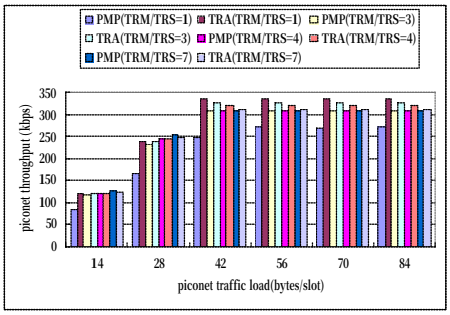

圖 5.11 距離 6 公尺、FTP 且 TRM >= TRS，PTL 與 piconet throughput 的關係圖

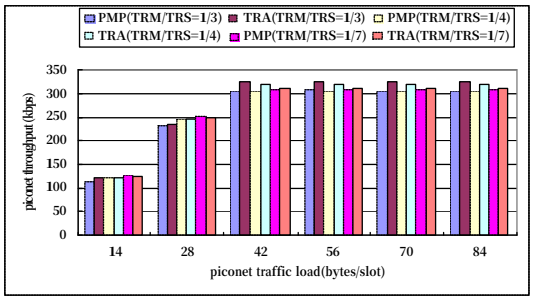

圖 5.12 距離 6 公尺、FTP 且 TRM < TRS，PTL 與 piconet throughput 的關係圖和 Throughput 的關係圖

第三節 總結

由第四節的各種數據圖可以整理出 TRA 演算法產能優於 PMP 演算法的情況如表 5.2。

表 5.2 TRA 優於 PMP 的情況

D / Traffic Type	1 m	3 m	5 m	6 m
CBR	PTL > 42 (B/s)	PTL > 42 (B/s)	PTL > 42 (B/s)	PTL > 28 (B/s)
FTP	TRM/TRS =7 or 1/7 且 PTL > 42 (B/s)	TRM/TRS =7 or 1/7 且 PTL > 42 (B/s)	PTL > 56 (B/s)	PTL > 28 (B/s)

PTL: piconet traffic load (Bytes/slot)

由表 5.2 知道當衰變通道的衰變狀況是不嚴重的，只有當 PTL 愈大時我們演算法的傳送產能才會比 PMP 演算法還要高，而當衰變通道的衰變狀況愈嚴重而且 PTL 愈大時，我們演算法的傳送產能會比 PMP 還要高，所以我們的貢獻是在衰變通道的衰變狀況愈嚴重而且 PTL 愈大時，我們演算法是可以有提升傳送產能。

第五章結論與未來發展

由於現今生活中，藍芽無線系統技術日益提升，裝置之間的傳輸速率也再日漸加快，而且現在藍芽無線技術也日漸普遍，因此我們必須要探討藍芽傳送的情況。藍芽裝置是在 ISM 的頻道上傳送資料，這個頻道的衰變現象是很明顯的，而在衰變通道中傳送資料是會大大影響傳輸效能。在藍芽 piconet 內每對 master、slave 的 traffic rate 是非對稱的，因此要根據兩者的 traffic rate 做適當的傳送封包配對，進而達到產能的提升。所以我們根據通道衰變現象以及 master、slave 兩者 traffic rate 發展一個有效率的藍芽 polling 演算法(TRA 演算法) ，在藉由 TRA 演算法來提升在衰變通道的傳輸產能。

我們在 NS2 以及 Bluehoc 平台上模擬我們演算法在 Rayleigh 衰變通道環境中傳送的效能，在傳送距離為 1、3 公尺，裝置的傳送工具為 CBR， piconet traffic load >= 42 Bytes/slot

時，我們演算法產能都會比 PMP 演算法高；piconet traffic load <42 Bytes/slot 時，我們演算法產能和 PMP 產能差不多。傳送距離為 1、3 公尺而且裝置是使用 FTP 為傳送工具時，TRM/TRS 的比值為 1/4、1/3、3、4，我們演算法產能會比 PMP 要低，但是當 TRM/TRS 的比值為 1/7、7 而且 piconet traffic load >= 42 Bytes/slot，我們演算法的產能會比 PMP 還要高。在傳送距離為 5 公尺且 piconet traffic load >= 42 Bytes/slot 時，不論傳送工具是 CBR 還是 FTP，我們產能都會比 PMP 演算法高。在傳送距離為 6 公尺時，我們演算法的產能都會比 PMP 演算法高。本論文的貢獻是在通道衰變現象愈嚴重而且 PTL 愈大時提升傳送產能。

未來研究工作上，希望能將這個演算法的概念拓展到 Scatternet 上，進而讓藍芽網路在衰變通道中都可以有最佳的傳送效能。

參考文獻

[1] Farooq Anjum, Dave Famolari, “A Robust And Adaptive Algorithm For Enabling Efficient Communication In Bluetooth Networks”, in IEEE GLOBCOM 2002. IEEE, Volume: 1, Nov 17-21 2002. Page.17-21

[2] Manish Kalia, Deepak Bansal, Rajeev Shorey, “Data scheduling and SAR for Bluetooth MAC”, in Proceeding of IEEE Vehicular Technology Conference, 2000, Page. 716-720

[3] Indraneel Chakraborty, Abhisek Kashyap, Anupam Rastogi, Huzar Saran, Rajeev Shorey, Apurva Kumar, “Policies for increasing throughput and decreasing power consumption in Bluetooth MAC”, in IEEE International Conference on Personal Wireless Communications, 2000. Dec. 17-20, 2000, Page. 90-94

[4] Indraneel Chakraborty, Abhishek Kashyap, Apurva Kumar, Anupam Rastogi, Huzur Saran, Rajeev Shorey, “MAC scheduling policies with reduced power consumption and bounded packet delays for centrally”, in IEEE International Conference on Communications, 2001, Volume: 7, June. 11-14, 2001, Page. 1980-1984

[5] Hao Zhu, Guohong Cao , George Kesidis, Chita Das, “An adaptive power-conserving service discipline for Bluetooth”, in IEEE International Conference on Communication, 2002. Volume: 1, 2002, Page. 312-321

[6] T. Y. Chui, F. Thaler, W. G.. Scanlon, “A Novel Channel Modeling Technique for Performance Analysis of Bluetooth Baseband Packets”, IEEE International Conference on Communications, Volume: 1 , 2002, Page. 308-312

[7] Yang-Ick Joo, Jong-Soo Oh, OH-Seok Kwon, Yongsuk Kim, Tae-Jin Lee, Kyun Hyon Techah, “An efficient and Qos-aware scheduling policy for Bluetooth”, in Proceedings of 2002 IEEE 56th VTC, Volume: 4, Sept. 24-28, 2002, Page. 2445-2448

[8] Mark Perillo, Wendi B. Heinzelman, “ASP: an adaptive energy-efficient polling algorithm for Bluetooth piconets” in Proceedings of the 36th Annual Hawaii International Conference on System Sciences, Jan 6-9, 2003

[9] Jong-Soo Oh, Yang-Ick Joo, Oh-Seok Kwon, Yongsuk Kim, Tae-Jin Lee, Kyun Hyon Tchah,“ Differentiated fairness guaranteeing scheduling policies for Bluetooth ”, in Proceedings of IEEE Vehicular Technology Conference, 2002. Volume: 2, 2002, Page: 923-926

[10] Ting-Yu Lin, Yu-Chee Tseng, Yuan-Ting Lu, “An efficient link polling policy by pattern matching for Bluetooth piconets”, in Proceedings of the 36th Annual Hawaii International Conference on System Sciences, Jan 6-9, 2003

[11] Joao H. Kleinschmidt, Marcelo E. Pellenz ,Luiz A. de Paula Lima Jr., “Evaluating and Improving Bluetooth Piconet Performance over Nakagami-m Fading Channels”, The Ninth IEEE Symposium on Computers and Communications, 2004, Page.639-644

[12] Matthew C. Valenti, Max Robert, Jeffrey H. Reed, “On the throughput of Bluetooth data transmissions”, IEEE Wireless Communications and Networking Conference, 2002, Page. 119-123