

# 改善 BlueTree 網路形成機制的設計與效能研究

邱奕維  
亞洲大學光電與通訊學系  
feyfey123@hotmail.com

余誌民  
中華大學通訊工程學系  
ycm@chu.edu.tw

蕭進松  
亞洲大學光電與通訊學系  
cshsiao@asia.edu.tw

## 摘要

現有藍芽系統已定義微網(Piconet)之架構，而如何將數個微網網路連接形成一個散網(Scatternet)是現在的研究議題之一，目前已經有許多學者針對散網形成進行研究，提出相關演算的方法，大概可以區分為樹狀(Tree)架構演算、環狀(Ring) 架構演算、星狀(Stars) 架構演算及網狀(Mesh) 架構演算等幾種類型。在本篇論文中，基於 BlueTree 的網路形成演算法，我們提出了一種網路回連的機制(BRC)，加入在 BlueTree 的演算法上，使 BlueTree 的散網結構，從樹狀變換成環狀的結構，以提升它的網路連線效率，而從模擬結果發現 BlueTree 的回連機制比 BlueTree 有更好的網路效能，有效縮短 BlueTree 節點間的平均路徑長度，減少 Blueroot 的訊務負載量和增加網路容量。

**關鍵詞：**藍芽、微網、散網

## Abstract

Bluetooth system has been defined piconet structure and how to connect several piconets into a scatternet is one of the research issues. Currently, there are already many scholars to study the scatternet formation and to propose their own algorithms. There are tree algorithm, ring algorithm, star algorithm, mesh algorithms and etc. In the paper, based on Bluetree design, a return connection mechanism is added to improve the scatternet performance. This mechanism can make Bluetree scatternet architecture from a tree structure into a ring structure and improve the connection efficiency. From the performance results, BlueTree of return connection (BRC) achieves better network performance than Bluetree by reducing the average path length among nodes. Reduce the Blueroot of traffic load and increase network capacity.

**Keywords:** bluetooth、piconet、scatternet

## 1. 簡介

近年來網際網路十分普及使得人們對資訊科技依賴度提升很多，藍芽是屬於一種短距離、低功率、低成本特性的無線通訊技術，其目標就是讓電子資訊產品裝置利用無線的模式連接到另一個裝置上，以取代有線的方式連結。

根據藍芽規格書[1]的定義，藍芽裝置要進行連結，必須要有一方扮演 Master，另一方扮演 Slaver 角色才能進行連結，且可以進行一對一(point to point)與一對多(point to multipoint)的連結如圖 1，互相連結後所形成一個微網(Piconet)，在微網中，一個 Master 裝置，最多可以跟七個 Slave 裝置連結形成。Master 裝置負責決定跳頻的序列，Slave 裝置跳頻頻率和時序必須與 Maser 裝置相同時，才能進行相連與傳輸，另外網路內的傳輸速率是 1M bit/s，由 Master 裝置分配給各個 Slave 裝置間的連線速率，所以說當網路內的 Slave 裝置越多時，所分配到的傳輸速率將隨之降低。

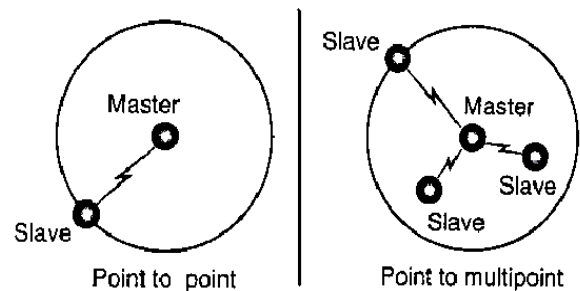


圖 1 單點對點與單點對多點的微網

散網(Scatternet)是由兩個以上的微網所組成，須經由一些中繼設備(Relay)節點來進行連結，而形成的網路架構，而 Relay 就是一個藍芽設備中扮演多重角色，Relay 同時可以屬於多個微網路，舉例來說：在某個微網路中的 Master 可以當另一個微網路的 Slave，相反的，一個微網路中的 Slave 也可以當另一個微網路的 Master，如下圖 2。當然也有不同的微網路共用同一個 Slave 的情況，如下圖 3。

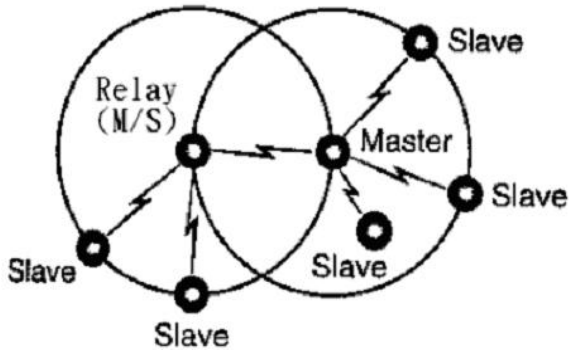


圖 2 Relay 扮演 Master 與 Slave 的角色

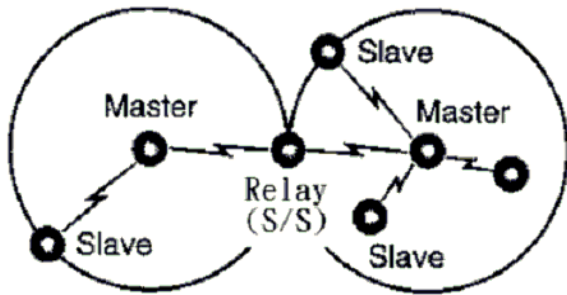


圖 3 Relay 扮演兩個微網中的 Slave 角色

藍芽規格書中並無明確定義藍芽散網 (Scatternet) 的形成，故大家只須遵守基本的規定限制，便可以依照自定的演算法去形成想要的散網架構，而目前藍芽散網研究已經發展出數多種架構形成演算法，大概可以分兩類，一種是集中式散網演算法，它會先選出 leader 節點，由它來控制分配散網的形成。像是 BlueTree[2]、BlueRing[3]。另外一種是分散式演算法，它就是不選出 leader 節點，讓所有的節點自行分散去運作，像是 BlueNet[4]、BlueStars[5]。

本篇論文所提出的演算法是以 BlueTree 為主體，然後再加入回連機制形成 BRC，以增加節點間的連線路徑，減輕 Blueroot 的資料傳送的負擔，同時具有 BlueTree 原有的優點。本篇論文在章節 2 會介紹 BlueTree 網路形成方法，在章節 3 將詳述 BRC 網路形成方法，在章節 4 研究 BRC 和 BlueTree 的效能比較，章節 5 做一個結論。

## 2. BlueTree 網路形成方法

由於本篇論文主要是探討改善 BlueTree 的效能，所以將簡單介紹一些 BlueTree 散網架構，BlueTree 是由三位美國德州 Dallas 大學學者 Gergely、Stefano 及 Imrich 所提出的樹狀架構演算。

他們假設每個節點最多扮演二或三個角色且節點移動性低，之後藍芽節點須隨機分布在一個事先給定的區域範圍內，還有在網路中的每一個節點都知道其附近節點的 MAC 位置 (BD\_ADDR)，且沒有孤立的節點 (isolated node)，假設完後才進行演算。

Bluetree 開始時會任意選取一個節點，指定為 Blueroot，然後與附近且只有 one-hop 的節點做連結，成為 Blueroot 的 Slave，之後這些 Slave 再轉成 Master 的角色 (S/M)，開始尋找附近且只有 one-hop 的節點成為它的 Slave，如此重複下去直到整個藍芽樹形成為止，如圖 4 所示。而這個架構的好處是網路架構容易建置，所需的時間也較短，而且幾乎每個節點都可以扮演 Master 的角色，Master 可以 Page 它 one-hop 可達的鄰居，故網路的連結機率相當高。但是也有缺點，Blueroot 節點容易形成傳輸時的瓶頸點 (bottleneck)，因為假如兩個節點分別在 Blueroot 的兩個分支，當這兩個節點有資料要傳送時，就必須經過 Blueroot，所以 Blueroot 的資料傳送量會相當大。

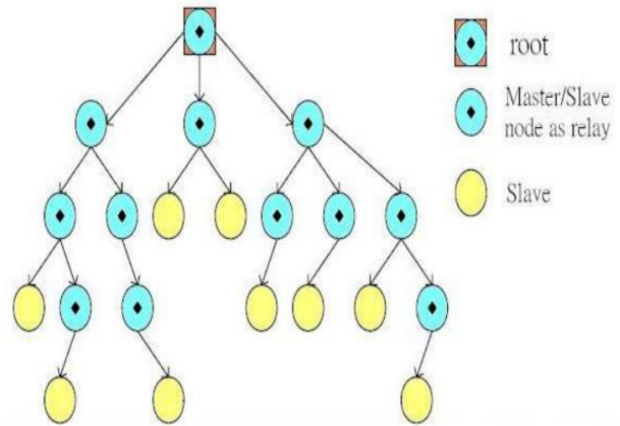


圖 4 BlueTree 的架構

## 3. BRC 網路形成方法

在此章節中，我們將詳述 BRC (Bluetree Return Connection) 的藍芽散網形成演算法的相關細節。BRC 演算法主要是根據 Bluetree 為主體，然後再加入回連機制所形成。

### 3.1 BRC 形成方法

首先任意挑選一個節點來當 Blueroot，利用這個來建立一棵衍生樹，Blueroot 會被指定為 Master 的角色，接下來，Blueroot 會 Page 附

近的鄰居，使得這些鄰居成為這個 Master 的 Slaves，這些節點的 Slave 將會轉換另一個角色 Master，然後再去 Page 還沒有被其它節點 Page 過附近的鄰居，使其成為它的 Slaves。而這樣的動作將會一直重複直到末端的節點 (leaf node)，也就是說明這些末端的節點只需要扮演 Slave 的角色，如此一來整棵樹就將可建立完成。如下圖 5 所示。

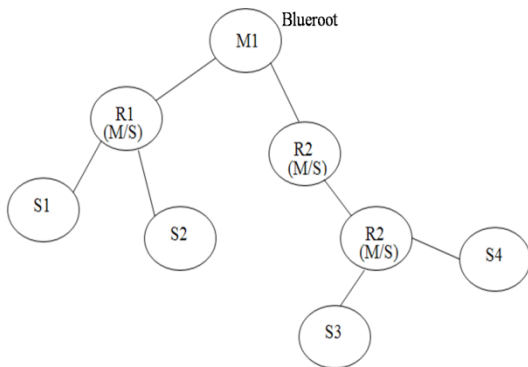


圖 5 未加入回連機制

當 Bluetree 連結到 leaf node 後，末端 Slave 節點會開始啟動回連機制，此 Slave 節點要進行回連時，它會先通知它的上一個 Master 節點，而這個 Master 節點收到回連的通知後，節點角色變成 Return Master 節點，開始進行 Page 的動作，以連結其它新的 Slaves，被連結的 Slave 就會改變角色成為 Relay(slave/slave)。Relay 就可以連結兩個新的微網，回連之後繼續通知上一個 Master 執行回連機制，直到 Blueroot 為止。形成結果如下圖 6 所示。

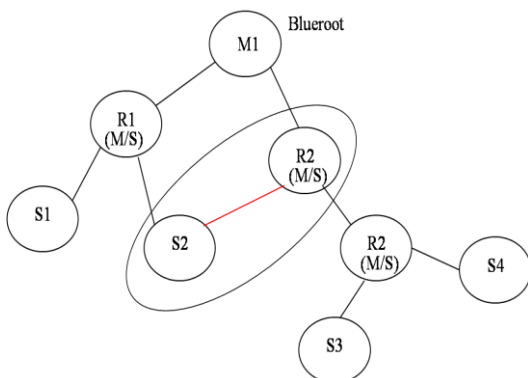


圖 6 加入回連機制

### 3.2 BRC 的演算流程

用幾個簡單的連結流程圖來介紹 BRC 的演算。圓圈代表的是角色的狀態

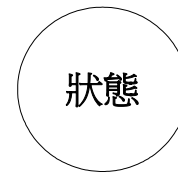


圖 7 狀態示意圖

一開始，所有的節點都處於初始狀態，在這個狀態裡，每一個節點有足夠的時間去做 Inquiry、Inquiry Scan、Inquiry Response 的動作，用以發掘它所有的一個 hop 可達的節點。然後，隨機選擇一個節點當 Blueroot，被選定的節點，開始 Master 流程，而其它節點開始 Slave 流程。如圖 8

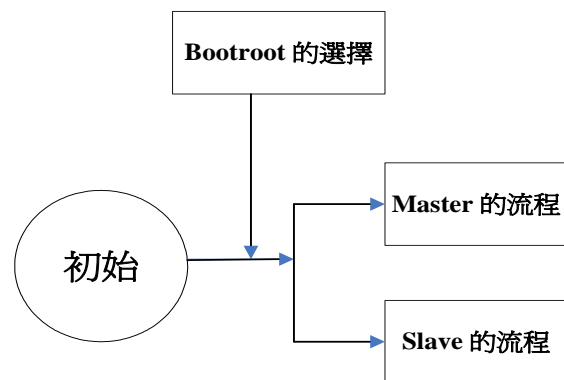


圖 8 BRC 的初始狀態

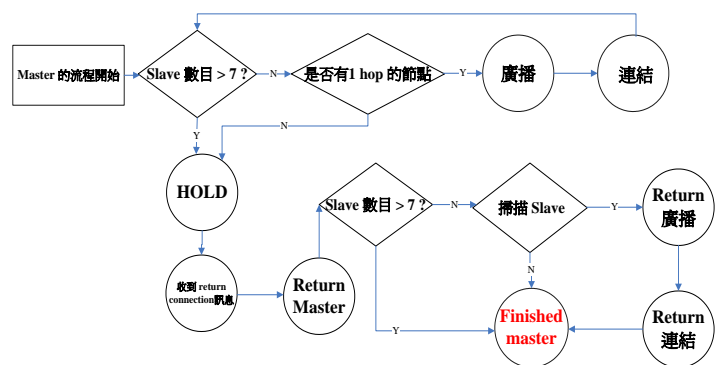


圖 9 Bluetree Master 連結流程圖

Master 會先檢查微網內的 Slave 個數是否大於七？如果沒有，則去檢查是否有一個 hop 的 slave 節點？若有，則 Page 此節點，並且和此節點連結，則此節點就成為這個 Master 的 Slave。若 Master 的 Slave 個數已經到達七個，或是已經沒有其它一個 hop 可達的 Slave 節點可以連結時，則 Master 進入 Hold 狀態。

接著，當 Master 收到 Return connection 的

訊息，則 Master 轉變角色成 Return Master，接著判斷微網內的 Slave 個數是否大於七？如果是，則 Master 動作完成。如果小於七，則開始掃描附近的 Slave 節點，若 Slave 節點已連結的 master 還沒有跟 Return Master 連結過，則去 Return Page 並且連結此節點，則此節點就成為這 Master 的中繼 Relay (Slave/Slave)。如圖 9

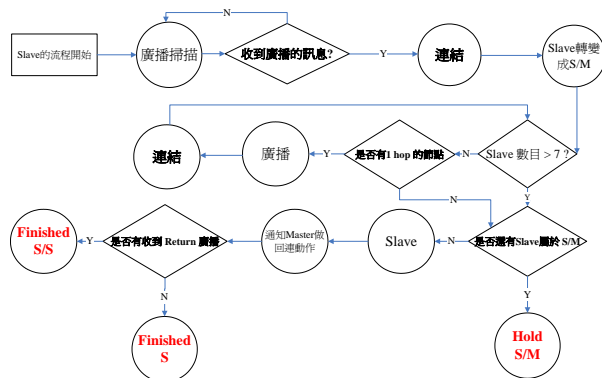


圖 10 BlueTree Slave 連結流程圖

一開始 Slave 先進入廣播掃描(Page Scan)的狀態，若收到附近的 Master 所發出的廣播訊息(Page)，則 Slave 就會跟此 Master 做連結。若沒收到，就繼續等待 Page 的訊息，因為我們假設沒有孤立的節點，所以所有的 Slave 都會接收到 Page 的訊息。

連結建立完成之後，Slave 開始作角色轉換的動作，由 Slave 角色轉換成 S/M 角色，這個節點同時扮演著 Slave 與 Master 兩種角色，S/M 會先檢查微網內的 Slave 個數是否大於七？如果沒有，則去檢查是否有一個 hop 的 slave 節點？若有，則 Page 此節點，並且和此節點連結，則此節點就成為這個 S/M 的 Slave。

若 S/M 的 Slave 個數已經到達七個，或是已經沒有其它一個 hop 可達的 Slave 節點可以連結時，則要判斷是否有 Slave 屬於 S/M？，如果有則這個 S/M 的連線動作完成，以 S/M 角色進入保留(Hold)狀態。

若完全沒有 Slave，則它會開始轉換角色，從 S/M 轉換成一般的 Slave，此時 Slave 會通知所屬的 Master 進行回連的動作，如果 Slave 收到了 Return Page 的訊息，則與 Return Master 做連接，成為中繼 Relay (S/S)，進入完成狀態，若沒收到 Return Page 的訊息，此時角色就是一般的 Slave。如圖 10

## 4. 模擬環境與效能

### 4.1 模擬環境

在這個章節中，為去評估這個 BlueTree 與 BRC 的效能，我們使用 Matlab R2008a 版來當作我們模擬的工具，首先節點個數分別從 40，60，80，100，120，140，160，隨機分布在一個 40 \*40 平方公尺的矩形區域大小裡，且節點位置不會重複，然後每個節點都有一個識別碼(id)，在從節點中，隨機選出一個節點當 BlueTree 開始進行連接，而電波可傳送距離是 10 公尺。

圖 11、圖 12 簡化的模擬示意圖，讓大家了解這兩種演算法的連結情況。

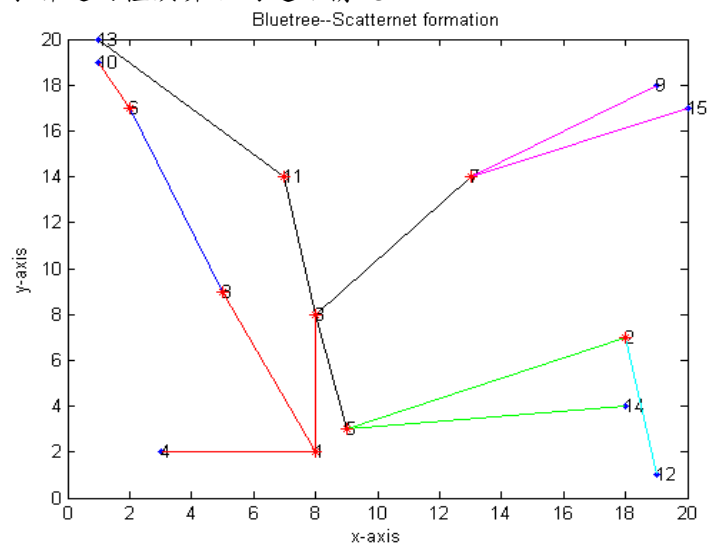


圖 11 BlueTree 的模擬示意圖

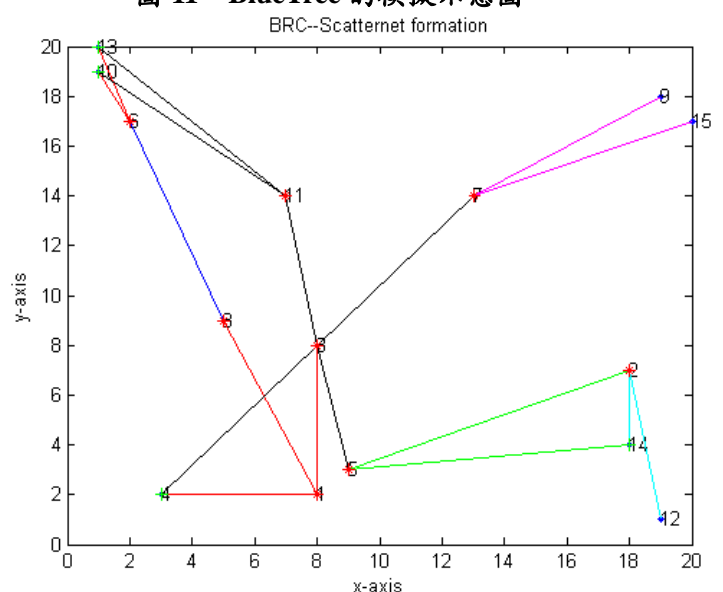


圖 12 BRC 的模擬示意圖



上面兩圖是簡化版 Bluetree 與 BRC 的模擬示意圖，模擬區域大小是 20 \* 20 平方公尺，連線距離是 10 公尺，節點個數是 15，其中星號 "\*" 是代表 Master，點 "." 代表 Slave。十字號 "+" 是代表 Relay。

## 4.2 效能模擬

最後我們將進行比較下列五種參數，分別是 piconet\_number(微網路個數)、piconet\_eff(微網路效率)、avg\_hop\_length(平均路徑長)、avg\_role\_num(平均角色扮演)、time(連結時間)。

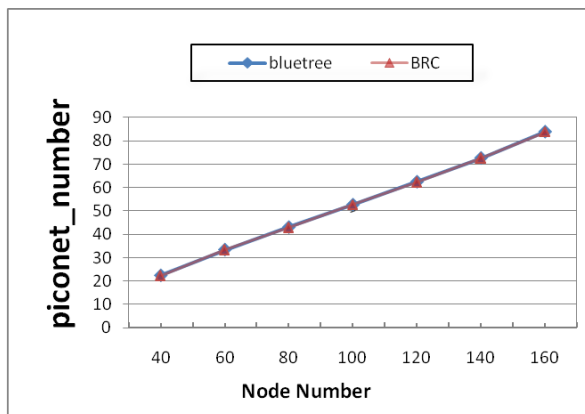


圖 13 微網路個數分布圖

微網路個數定義，是指在相同的節點分布，所形成的微網路個數。由圖 13 表示，發現兩條線是重疊，因此可以看出 BlueTree 與 BRC 的微網個數是一樣的，那是因為 BRC 的連結方式主要是以 BlueTree 為基礎，只是在末端的 Slave 的節點，會再進行回連的機制，只有增加連 Slave 的數目，並沒有影響到微網個數。

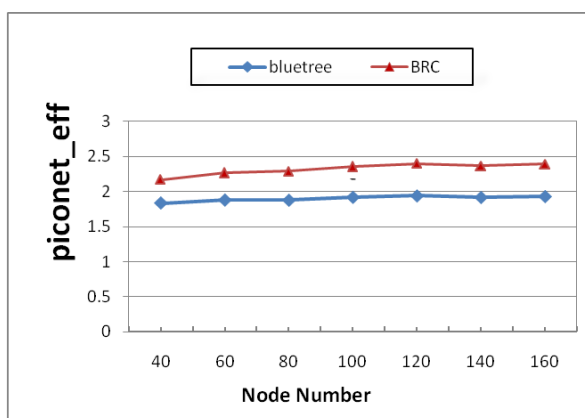


圖 14 微網路效率分布圖

微網路效率定義，是將節點個數除以微網路數目，得到平均每個微網路中有幾個節點。

如圖 14 所示，BRC 平均高於 BlueTree 大約有 0.5 個單位，因為 BRC 具有回連機制，使 Master 可以再次連其它 Slave，以增加微網內的數目，使其效率提升。

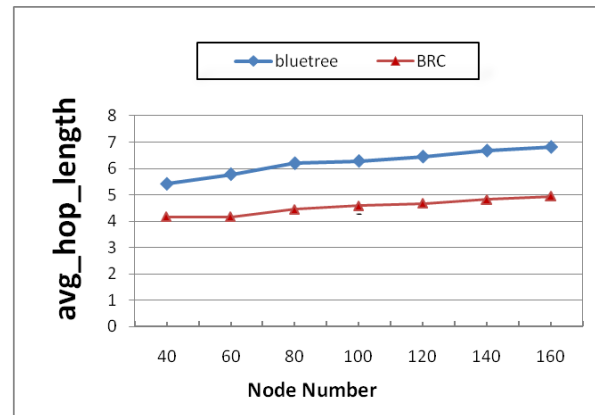


圖 15 平均路徑長度分布圖

接下來要看平均路徑長，這裡我們定義兩個節點之間的連線路徑長算 1 個單位，由一個節點到任一個節點，平均所要走的路徑長，在這裡的路徑長是指節點間的最短路徑，然而由圖 15，可以看出，BRC 低於 Bluetree 平均有 1.6 個單位，因為 BRC 有回連機制，所以連線的路徑會多於 Bluetree，因此 BRC 的平均路徑長會比 Bluetree 小。

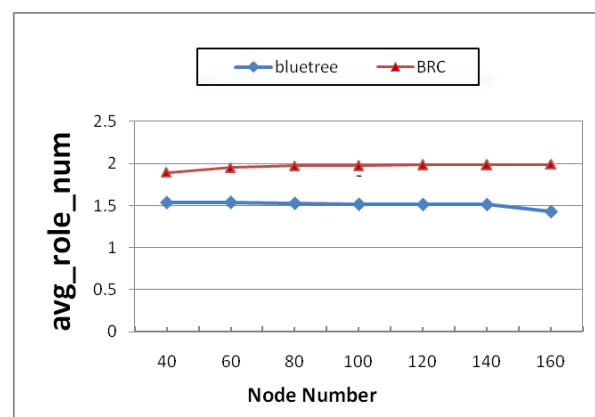


圖 16 平均角色扮演分布圖

平均角色扮演定義，是指在區域內每個節點平均所扮演的角色數目，我們將所有的角色 (Master 算 1, Slave 算 1, Relay 算 2) 加總除以節點個數。由圖 16 中可以看出，BlueTree 的增加斜率幾乎維持 1.5 個單位左右，而 BRC 因為有回連的機制，所以末端的節點會多扮演一次 Relay 的角色，因此平均會高於 BlueTree 0.4 個單位。

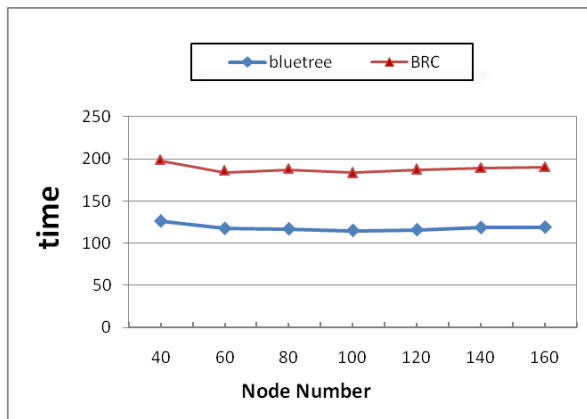


圖 17 連結時間分布圖

連結時間是指系統模擬時間也就是網路形成時間，在這裡我們只有考慮 Page 到 connection 的時間，並沒有包含 Inquiry·Inquiry Scan 的這段時間。由圖 17 可明顯知道，BlueTree 是低於 BRC 平均有 70 個單位時間 (1/1600 秒)，因為 BRC 有回連的機制，所以需要一些時間進行回連的動作，因此大概會 BlueTree 多出一些時間，而連結時間因為並行連結，故不會隨著節點的個數多寡而有太大的起伏。

## 5. 結論

本篇論文主要探討 BlueTree 加上回連機制後，網路形成的 BRC 架構，然後進行模擬與效能比較，模擬比較出微網路個數、微網路效率、平均路徑長、平均角色扮演、連結時間等等五張模擬圖。而根據五張模擬圖所呈現的數據，我們可以得知，BRC 主要在微網路效率與平均路徑長比 BlueTree 有更好的效能，但在連結時間方面，卻比 BlueTree 多了一些時間，而對於藍芽連結方面影響不大，而本篇論文所以提出的回連機制，有效縮短 BlueTree 節點間的平均路徑長度，可以減少 Blueroot 的訊務負載量和增加網路容量。

## 參考文獻

- [1] Bluetooth SIG, "Specification of the Bluetooth System, Version 1.1", <http://www.bluetooth.com/>
- [2] Gergely V.Zaruba, Stefano Basagni, and Imrich Chlamtac. "Bluetrees—Scatternet Formation to Enable Bluetooth-Based Ad Hoc Networks", IEEE International Conference, 2001, vol.1, pp. 273-277.

- [3] Ting-Yu Lin, Yu-Chee Tseng, Keng-Ming Chang and Chun-Liang Tu. "Formation, Routing, and Maintenance Protocols for the BlueRing Scatternet of Bluetooths", IEEE Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) 2002
- [4] Zhifang Wang, Robert J.Thomas and Zygmunt Hass. "Bluenet—a New Scatternet Formation Scheme". IEEE Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS~35'02) 2002.
- [5] Stefano Basagni and Chiara Petrioli. "Configuring BlueStars : Multihop Scatternet Formation for Bluetooth Networks", IEEE Transactions on Computers, Vol.52, No.6, June 2003