

# 無線多躍程中繼網路提供服務品質路由機制及 連線排程之研究

胡碩誠 李效謙

世新大學資訊管理學系

11604 台北市文山區木柵路 1 段 17 巷 1 號

## 摘 要

無線多躍程中繼網路是由基地台 (base station, BS) 及中繼站 (relay station, RS) 所組成，藉由設置中繼站來增加無線網路涵蓋範圍具有建置成本低與容易建置的優點。中繼站將來自於使用者的資料封包傳送給連接有線網路的基地台 (可能通過其他的中繼點)。兩個中繼點間不允許直接傳遞封包。一般認為這樣的架構較其他無線網路架構 (如無線網狀網路) 更適合真實的佈建。

為了滿足越來越多的即時性應用需求，如何在無線網路上提供服務品質保證一直是網路研究的重要課題。本研究藉由計算路徑干擾值的方式在無線多躍程中繼網路上求得以基地台為樹根的樹狀路由，目標在於降低整體網路的干擾程度，提高連線的服務品質。在頻寬管理方面，我們提出了一個集中式連線排程以及一個階層式的連線排程機制。在階層式的連線排程機制中樹狀路由的每個節點 (中繼站) 可將來自子節點頻寬需求做可能的整合排程後，再回傳給基地台。如此，除了可以減輕使用集中式連線排程所帶給基地台的負荷外，中繼站也可以做到更有彈性的頻寬配置。依據實驗結果，本研究所提出的方法確實可以有效的增加網路傳輸量，提高頻寬的重複利用率。

**關鍵詞：**無線多躍程網路，服務品質，干擾，路由，連線排程

## QoS Routing and Link Scheduling in Wireless Multi-Hop Relay Networks

SHUO-CHENG HU and HSIAO-CHIEN LI

*Department of Information Management, Shih-Hsin University*

*#1 Lane 17, Sec. 1, Mu-Cha Rd., Taipei 11604, Taiwan, R.O.C*

### ABSTRACT

A wireless multi-hop relay network consists of a base station plus various relay stations. The introduction of relay stations greatly extends the network coverage, to the benefit of users, and provides the salient advantages of lower implementation cost and easier system deployment. Relay stations transmit packets from (to) users toward (from) the base station, which is connected to a wired

network. Unlike mesh networks, however, direct transmission between relay stations is not allowed. Therefore, wireless network architecture is more feasible for deployment in the real world.

For real-time applications, much research has been devoted to providing QoS guarantees in wireless networks. Accordingly, this report first proposes an interference-aware routing algorithm to construct a tree-based routing path rooted at the base station. The algorithm explores the least-interfering path to improve routing efficiency. For bandwidth management, we propose both centralized and hierarchical link scheduling mechanisms. In the latter, each node in the routing tree transmits its bandwidth request to the root, where potentially concurrent transmissions of descendents are combined to optimize data delivery. As a result, not only is the workload at the base station in the centralized link scheduling reduced considerably but the relay stations can also manage bandwidth in a flexible manner. Furthermore, the simulation results show that the proposed strategies can effectively improve overall network throughput and enhance frequency reuse. Thus, the performance evaluation indicates the usefulness of our development in actual practice.

**Key Words:** multi-hop relay network, QoS, interference, routing, link scheduling

## 一、研究動機

隨著可攜性裝置的普及，新一代的無線網路技術被期待能提供更高的資料傳輸率與更廣的訊號涵蓋範圍，以提供越來越多要求服務品質（quality of service, QoS）的即時性應用需求（如 VoIP、線上會議等），現今無線網路之架構主要由眾多的基地台（base station, BS）與用戶端組成，藉由事先網路規畫將基地台佈置於適當的地點以確保在有限傳輸功率下可使訊號完整的覆蓋特定區域，然而此種網路架構會產生基地台彼此間的干擾與多徑傳播，使得無線訊號衰退，導致有效資料傳輸速率降低的問題，如 WiMAX 系統（worldwide interoperability for microwave access）雖然宣稱可提供較第三代行動通訊系統（3rd-generation, 3G）更高的傳輸速率與更廣的訊號涵蓋範圍，但根據實際網路規劃後得到的結果，WiMAX 基地台在市區內合理的涵蓋半徑僅為數公里，且有效的資料傳輸速率會隨使用者與基地台間的距離增加而遞減 [16]。

在 TDMA-based 模式（time division multiple access）下，使用者需在實際傳送資料前，取得一條到基地台的傳輸路徑，以及能滿足其 QoS 需求的時槽（time slot）數量，因此除了改善網路架構外，路由機制與連線排程方式的好壞，對傳輸的效能與品質有直接的影響。本研究目的即在於建置出一個考慮整體網路干擾程度網路架構，並配合所設計的路由及排程機制以增進網路的傳輸效率與提高頻寬的重複利用率。本研究第二節將介紹無線多躍程網路的架構及其他學者提出的研究成果，並說明各自面臨的一些可能潛在性問題。

第三節討論本研究所提出的方法。第四節為說明電腦模擬網路架構並對實驗結果加以分析、討論。第五節則是結論與未來展望。

## 二、網路架構及相關研究

### （一）網路架構

在無線多躍程網路架構下，來源端與目的端的資料傳輸可透過中間節點將資料轉送至目的端，因此，在擴展無線網路的服務範圍上，可使用設置小型的無線設備的方式來代替傳統基地台的佈建，如圖 1。無線多躍程中繼網路是由基地台（base station, BS）及中繼站（relay station, RS）所組成 [1, 8]，藉由設置中繼站來增加無線網路涵蓋範圍具有建置成本低與容易建置的優點。中繼站將來自於使用者的資料封包傳送給連接有線網路的基地台（可能通過其他的中繼站）。與無線網狀網路 [2] 不同的是，兩個中繼站間不允許直接傳遞資料，因此，RS 只需具備轉傳訊號的功能，在設計上較無線網狀網路路由器（mesh router）來的簡單，在實際運用上也可以減少網路中控制封包的數量以提高網路的傳輸效能 [10]。

### （二）提供服務品質路由機制

由於及時性應用需求的增加，如何找出一條符合連線（connection）需求的傳輸路徑成為無線網路領域中一個重要的研究方向，當網路上的某個連結（link）正在傳送或接收資料時，若受到其鄰近節節的干擾，則容易造成資料遺失等問題，而無法保證該連結所要求的 QoS，因此當連結正在

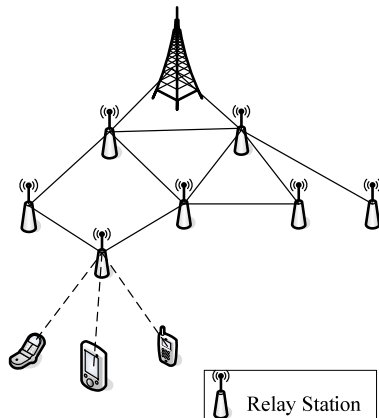


圖 1. 無線中繼網路

傳送或接收資料時，被該連結所干擾（interference）的其他連結則不能同時運作，如圖 2，當連結 AB 在運作時，連結 CD 與連結 EF 處於被干擾的情況而無法運作，而當使用者提出傳輸需求時，傳送端與目的端需預先決定傳輸路徑，因此，若在決定傳送路徑時選擇干擾值最小的路徑，便可以減少在傳送資料時影響其他連結的數量，提高能同時運作的連結數，增加網路傳輸量。

### 1. 單通道路由機制

單通道路由機制著重於找出一條符合連線需求且路徑干擾值為最小的路徑。有研究者提出，每個點各自紀錄本身干擾值，當有連線需求被提出時，路由機制將從所有可行路徑中選擇干擾總和最小的路徑作為傳輸路徑 [6]。另外也有研究假設網路中，節點是一個一個依序的加入，因此，在新節點要加入網路時，會從所有已加入網路的可連結點中選擇一個從連結有線網路的基地台至該點的路徑干擾總和最小的節點來連結，如此，在整個網路拓樸建置完成同時，也決定了各節點的路由 [14]。然而，作者在文中並未明確地提

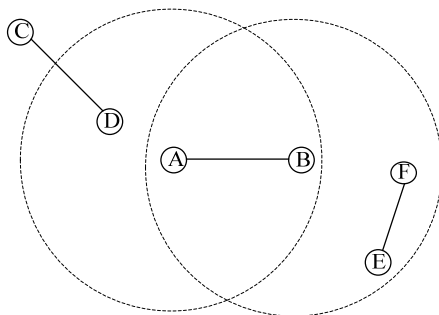


圖 2. 當連結 AB 在運作時，連結 CD、EF 無法同時運作

及實際的建置方法。

### 2. 多通道路由機制

為了降低單一通道（single channel）所造成的干擾，有學者提出使用多個彼此不互相干擾的通道來建構網路拓樸。有研究者提出一個集中控制式的無線通道配置與 QoS 路由的機制 [13]。也有研究使用預期的連結負擔程度來作為通道分配的依據 [11]，在一開始分配好通道之後，接著使用最短路徑演算法以預期網路中各連結的負擔程度，若網路中每個連結的負擔程度均小於其所能提供的頻寬，則使用此通道分配模式，否則便重新作通道分配，直到找出一個合適的路由拓樸為止。其它研究者也提出了各自的作法 [3, 7]。雖然多通道分配方式的確可以降低整體網路的干擾程度，但是除了在實體設備上需支援多通道功能外，在建置路由拓樸時勢必要花費更多的時間，這種問題在網路密集度越大的情況下將越加明顯。

### 3. 連線排程機制

在 multi-hop 網路架構下，依傳輸方式不同可分為集中式 [9, 12] 與分散式 [4] 兩種排程方式，集中式排程分配是中繼站將傳送需求往上传給基地台，當基地台收集完所有的頻寬需求後，由基地台決定網路中所有中繼站的傳送排程，之後基地台再將分配結果往下傳送讓所有的中繼站知道明確的排程資訊。分散式分配則是各個中繼站間可自行交換本身的排程訊息以決定彼此間的資源排程，然而，標準裡雖提及此兩種排程模式，卻未明確說明運作方式，且分散式排程分配方式在中繼站數量多的時候會過於複雜，因此多數的學者提出集中式分配的排程方式來作為 multi-hop 網路架構下的排程方式，而在使用分時雙工模式（time-division duplex, TDD）下，有研究者提出當基地台收到所有中繼站所提出的傳輸要求後，利用貪婪式（greedy）演算法來作為分配時槽的方式 [4]。Wei 等人 [14] 則提出一個時槽再利用（slot reuse）的排程方法使得同一個 slot 可以讓多個不互相干擾的連結同時傳輸資料。

## 三、研究方法

### （一）基本概念

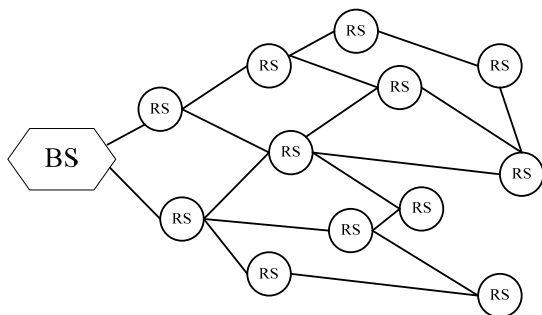
為了建置一個可涵蓋廣域範圍且能有效提供具 QoS 服務保證的無線網路，網路的拓樸架構、路由機制與連線排程機制都是必須考慮的因素。由於在無線多躍程中繼網路中，中繼站彼此間過多的連結會使得傳輸路徑受到干擾而導致

傳輸效能降低。因此，本研究藉由計算路徑干擾值的方式在無線多躍程中繼網路上求得以基地台為樹根的樹狀路由，如圖 3 所示。目標在於降低整體網路的干擾程度，提高連線的服務品質。在頻寬管理方面，我們使用階層式的連線排程取代先前提出的集中式排程分配。樹狀路由中的每個節點（即中繼站）可將來自子節點的頻寬需求做可能的整合排程後，再回傳給基地台。如此，除了可以減輕使用集中式連線排程所帶給基地台的負荷外，中繼站也可以做到更有彈性的頻寬配置。我們將在下面做更詳細的敘述。

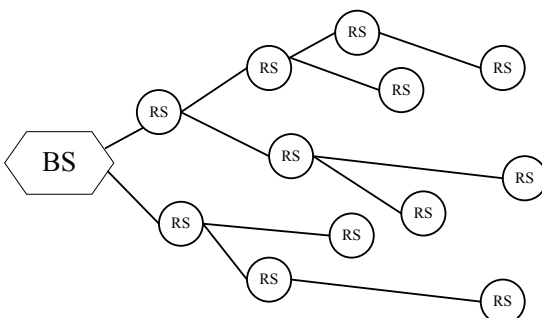
本研究提出的網路架構下，傳輸通道依方向不同可分為上行通道（uplink）—由 RS 傳向 BS，以及下行通道（downlink）—由 BS 傳向 RS。在分時多工的系統中，傳輸框架（frame）可分為上行通道子框架（uplink subframe）與下行通道子框架（downlink subframe），各包含若干個時槽，由 BS 負責進行動態的配置 [10, 15]。由於下行通道傳輸時，BS 是唯一有傳輸需求的節點，因此排程機制比較容易。所以本研究將把焦點放在上行通道傳輸的排程上。

## （二）樹狀路由拓樸之建立

在無線多躍程中繼網路中，中繼站彼此間過多的連結會使得傳輸路徑受到來自其他連結的干擾而導致傳輸效能下



原始無線中繼網路拓樸架構



本研究提出的拓樸架構

圖 3. 原始無線中繼網路拓樸架構與本研究提出的拓樸架構

降，而無線多躍程中繼網路的資料傳遞模式中，由於 RS 只負責將資料透過與其他 RS 間的轉傳，將資料一路傳遞至基地台，因此，為了減少網路中非必要的連結，以提高傳輸效能，本研究將無線多躍程中繼網路設計為樹狀拓樸，在樹狀拓樸中，每個 RS 決定一條路徑干擾值最小的路由做為該點至 BS 的傳輸路徑，而因為 RS 是靜止不動的設備，所以當節點決定好了一條樹狀路由後便不再更改，除非節點離開網路（如故障或關機）或一段時間後有新節點加入網路，才會改變已經配置好的路由，在選擇父節點方面，本研究以整條路徑所可能造成的干擾值來作為新節點選擇父節點的基準。某個節點的初始路徑干擾值表示在其傳送範圍內的其它節點數。每個已加入網路中的節點紀錄從父節點收到的路徑干擾值加上本身初始干擾值，形成該節點之路徑干擾值。新加入節點從可連結的候選父節點中選擇路徑干擾值最小的節點作為父節點，如此不僅可以確保每個節點在加入時能選擇傳輸效能最好的路由，也可以降低整體網路的干擾程度。詳細步驟如下：

- (1) 在整個系統開始之初，所有欲加入網路的節點在加入前須先廣播探測封包給其傳輸範圍內的其他節點，藉由有多少個節點回應此探測封包的訊息，來決定節點本身的初始路徑干擾值與干擾點的集合。
- (2) 在建立路由階段，首先，由 BS 廣播包含路徑干擾值（起始值為 0）的連線訊息給傳輸範圍內的其他節點，收到此訊息的節點，以 BS 做為父節點記錄由 BS 至該點的路徑干擾值，計算方式為父節點干擾值+節點本身的初始路徑干擾值。
- (3) 當節點加入網路後，廣播包含路徑干擾值的連線訊息，給傳輸範圍內的其他節點，當有尚未加入網路的節點收到此訊息時，等待一段時間來接收其他已加入網路的節點所傳來的連線訊息，若收到來自兩個以上的節點所傳來的連線訊息時，則選擇路徑干擾值最小的節點做為父節點，並計算本身的路徑干擾值。
- (4) 重覆步驟 3，直至所有節點都已加入網路為止。

舉例如下：圖 4(a) 中節點 D 傳送探測封包後，收到節點 A、B、C 的回應，因此節點 D 的初始路徑干擾值為 3，干擾集合為 {A、B、C}。而在圖 4(b) 中，節點 A、B 收到來自 BS 的連線訊息，因此選擇 BS 為父節點，節點 A 的路徑干擾值為 2(父節點干擾值 0 + 節點 A 本身的初始路徑干擾值 2)，節點 B 的路徑干擾值為 3。之後，節點 D 收到來

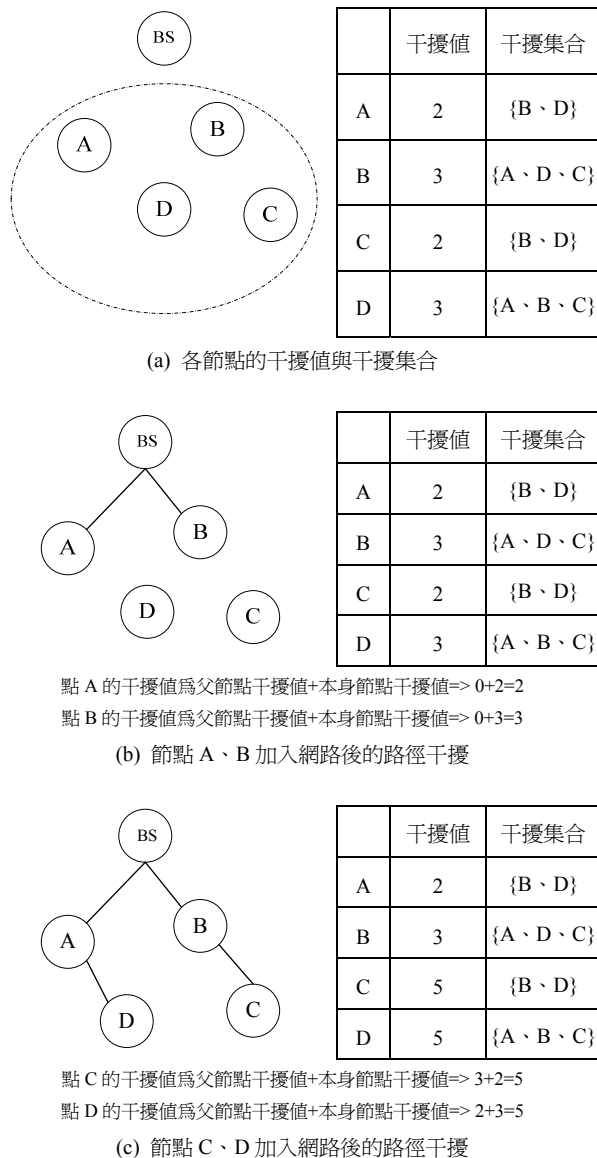


圖 4. 建立樹狀路由拓樸

自節點 A、B 的連線訊息，選擇節點 A 做為父節點，路徑干擾值為 2+3=5，節點 C 因只收到來自節點 B 的訊息，所以選擇節點 B 做為父節點，路徑干擾值為 3+2=5，如圖 4(c) 所示。

### (三) 傳輸集合的選擇

在 TDMA 模式下，時槽的重複利用率是影響傳輸效能的關鍵，時槽的重複利用率越高則代表網路中可同時傳輸的節點越多因此整體網路的傳輸效能越好，所以，如何判斷哪些不互相干擾的節點可放入相同的時槽中是值得探討的一個方向，而在我們所設計的樹狀拓樸中，因為 RS 是靜止不動的設備，且拓樸完成後，除非節點離開網路（如故障或關

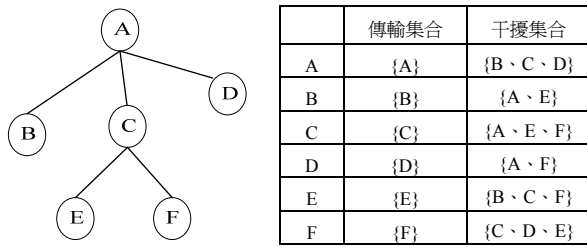
機）或一段時間後有新節點加入網路，才會修改拓樸架構，因此，當所有節點都已加入網路後，我們提出傳輸集合的概念來判斷網路上可同時傳輸的節點，若網路上節點屬於同個傳輸集合，則代表節點間可以同時傳輸而不會互相干擾。本研究從樹狀路由最下端的節點開始，向父節點傳送包含節點本身與該子樹的集合資訊，此集合資訊中，包含節點本身的傳輸集合及與節點互相干擾的干擾集合資訊兩部分，當父節點收到來自子節點的集合資訊後，將該子樹內可以同時傳輸的節點作傳輸集合的合併，之後再繼續往上層傳送包含節點本身與該子樹的集合資訊，如此，當傳輸集合選擇結束後，網路上所有節點均有一份記錄其子樹內所有節點的傳輸集合資訊，方法說明如下：

- (1) 節點收到所有子節點所傳送的傳輸集合 ( $T\_set$ ) 與干擾集合 ( $I\_set$ ) 資訊。
- (2) 當節點  $i$  收到來自子節點 ( $child(i)$ ) 的集合資訊後，依照子節點的干擾集合  $I\_set_{child(i)} = \{I\_set_i \cup I\_set_{child(i)}\} - \{child(i)\}$ ，修改每個子節點本身的干擾集合，並記錄於節點  $i$  的集合資訊內。
- (3) 在節點所收到的集合資訊中，若存在  $k$  個傳輸集合 ( $T\_set_1, T\_set_2, \dots, T\_set_k$ )， $k \geq 2$ ，與其所對應的干擾集合 ( $I\_set_1, I\_set_2, \dots, I\_set_k$ ) 滿足  $(T\_set_1 \cup T\_set_2 \cup \dots \cup T\_set_k) \cap (I\_set_1 \cup I\_set_2 \cup \dots \cup I\_set_k) = \emptyset$ ，表示這  $k$  個傳輸集合中的節點可以同時傳輸而不會有互相干擾的情況發生。接著則將此  $k$  個傳輸集合及干擾集合以其聯集的結果取代。
- (4) 節點將本身所記錄的集合資訊傳送給其所屬的父節點。

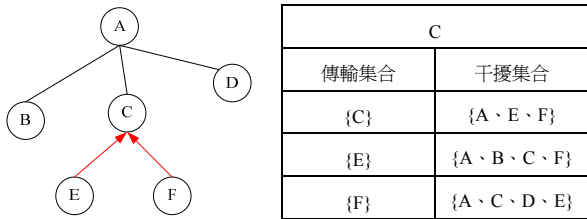
如圖 5(a)，網路拓樸與各節點一開始的傳輸集合與干擾集合資訊。一開始節點 E、F 分別送出本身的集合資訊給節點 C，節點 C 收到 E 與 F 的集合資訊後修改 E、F 的干擾集合，並將之紀錄於本身的集合資訊中如圖 5(b)。在圖 5(c) 中，節點 A 收到來自 B、C、D 的集合資訊後，分別將 B、F 與 D、E 做合併，因此，最後節點 A 往上層傳送的集合資訊為圖 5(d)。

### (四) 連線排程機制

在連線排程機制中，集中式排程分配在多躍程 (multi-hop) 網路架構中會有資料到達順序的問題，因此單純以連結所提出的傳輸要求量來做連線排程是不甚合理的，如圖 6 所示，假設各節點本身要求的傳送量均為 5 個時槽，由於上層的節點需承接從下層來的資料量，因此節點 A

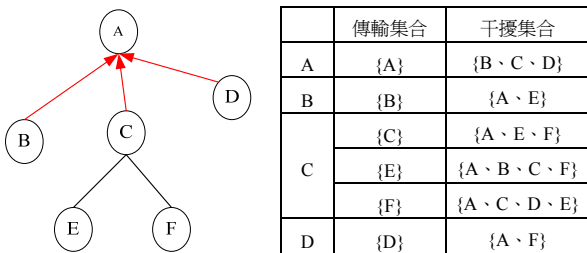


(a) 各節點初始所記錄的傳輸集合與干擾集合資訊

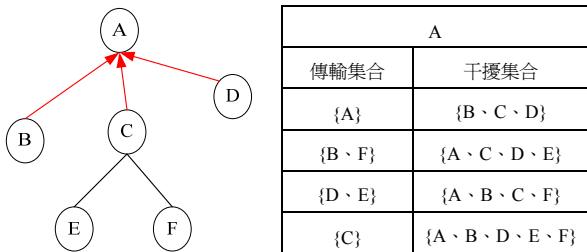


節點 C 收到 E 與 F 的集合資訊後，修改 E、F 的干擾集合  
 E 的干擾集合為 {A、E、F} U {B、C、F} - E = {A、B、C、F}  
 F 的干擾集合為 {A、E、F} U {C、D、E} - F = {A、C、D、E}

(b) 點 C 所記錄的集合資訊



(c) 點 A 所收到的各點集合資訊，其中 B、F 及 D、E 是可合併的



(d) 節點 A 所記錄的集合資訊

圖 5. 節點中傳輸集合與干擾集合之建立

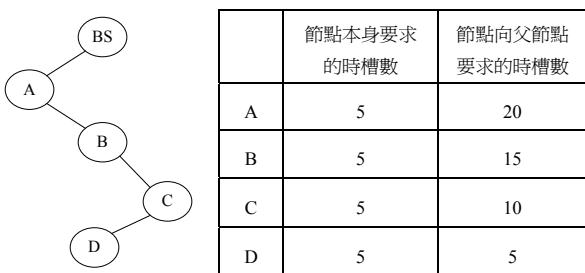


圖 6. 集中式排程問題

會向 BS 要求 20 個時槽來傳送資料，Wei 等人 [14] 提出的排程演算法是選擇連線需求量最大的連結加入排程，因此一開始 BS 便會先分配 20 個時槽給予節點 A，然而，此時節點 A 所能傳送的資料量只有 5 個時槽，因此，有 15 個時槽是浪費的。下面我們除了將上述的集中式排程方式做修改，使其可以符合本研究所提出的樹狀網路拓樸架構，避免產生上述的問題之外，我們也設計一個階層式連線排程機制減輕使用集中式連線排程所帶給基地台的負荷。在第四節的實驗中分析中，我們將以此兩種排程方式來做比較。

1. 修改之集中式連線排程機制

有別於多躍程網路中，節點所提出的傳輸要求量中包含節點本身的傳輸量及從下層承接上來的傳輸量，我們設計當節點收到下層子節點所傳來的傳輸要求量後只負責將此要求訊息轉傳至 BS 中，而不會將子節點的傳輸量加入本身的傳輸要求量中，所以，最後 BS 會有網路中所有節點本身所要求的傳輸量資訊，且因為本研究的網路架構為樹狀拓樸，因此，BS 可以紀錄網路中所有節點的父節點資訊如圖 7(a) 中 BS 所收到的傳輸要求資訊，與父節點資訊。接下來，使用 Wei 等人[14] 的排程方式，不同的是當某一節點被選擇進入排程後，BS 會對照該節點的父節點資訊，將該節點的傳輸量加入其父節點的傳輸要求量中，如圖 7(b)，點 D 進入排程後，BS 修改點 A 的傳輸量，之後再從已更新過的傳輸資訊表中，依照 Wei 等人[14] 的方法選擇其他節點進入排程，如此直至網路上所有點的傳輸要求都被完成為止，我們以這種即時反映各節點當前可傳送的資料量方式來避免因資料到達順序不同所造成的時槽浪費。表 1 為修改之集中式排程演算法。

2. 階層式連線排程機制

在目前的連線排程機制中，節點負責傳遞連線需求給連接有線網路的基地台，由基地台配置完網路中所有節點的傳輸時槽之後，再將排程通知網路中所有節點，這種集中式排程的作法對於 BS 的負擔程度會隨著網路節點增多而加重，因此我們設計一個階層式的排程方法，將一部分的排程分配交由 RS 負責，以降低 BS 的負擔，針對樹狀網路拓樸，本研究所設計的連線排程機制原則如下：當與 RS 相連的用戶端設備對 RS 提出頻寬的需求（包括欲新建立或現存的連線），RS 在收集了這些頻寬需求之後向上送出需求 (request) 給父節點，父節點收集了所有子節點的頻寬需求之後依照傳輸集合資訊將其子樹的頻寬需求做時槽的分配及合併可共

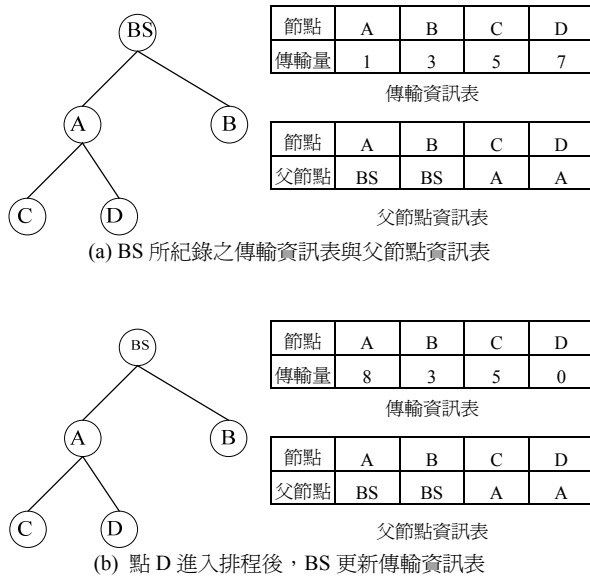


圖 7. 修改之集中式排程演算法中傳輸資訊表之更新

表 1. 修改之集中式排程演算法

```

R(i): 節點 i 所提出之傳輸時槽需求
par(i): 節點 i 的父節點
ActiveNodes( $t_s, t_e$ ): 排程于第  $t_s$  至  $t_e$  時槽間的節點所成的集合
 $t_s \leftarrow 1$ 
while exist any  $R(i) > 0$  for any node i
   $k \leftarrow \underset{\forall i}{\operatorname{argmax}} R(i)$ 
   $I\_set \leftarrow \emptyset$  //在此 iteration 所記錄之干擾集合
   $A\_set \leftarrow \emptyset$  //在此 iteration 所記錄可傳輸節點的集合
   $t_e = t_s + R(k) - 1$ ;
  while  $k \neq \text{NULL}$ 
     $R(\text{par}(k)) = R(\text{par}(k)) + R(k)$ 
     $R(k) = 0$ 
    add k to  $A\_set$ 
     $I\_set = I\_set \cup I\_set_k$ 
     $k \leftarrow \underset{j \in A\_set \cup I\_set}{\operatorname{argmax}} R(j)$ 
    where  $R(j) > 0$ 
  end while
  ActiveNodes( $t_s, t_e$ )  $\leftarrow A\_set$ 
   $t_s = t_e + 1$ ;
end while

```

用時槽的節點，之後繼續往上一層的父節點傳送包含此排程結果的需求訊息，如此一層層往上處理，最後 BS 只需合併不同子樹間可共用的時槽數後，再將最後的排程分配結果向下傳給所有的 RS 即可，而不需全權由 BS 決定所有節點的時槽分配。

為了實作此排程機制，我們除了需要節點所記錄的傳輸集合資訊外，還需要在各 RS 提出的需求訊息中，加入一項

資訊跳躍節點數 (hop-count)，用來判別排程中的資料量到達順序，跳躍節點數為 0 代表此資料量可立即被服務，跳躍節點數為 1 則代表此排程需要等所有跳躍節點數為 0 的排程都傳送完後才能被服務，依此類推，跳躍節點數越大代表資料需要等待時間越久。詳細的階層式連線排程演算法如表 2 所示。圖 8 為網路拓撲與各節點所記錄的集合資訊及各節點本身的 time slot 需求量。圖 9 和圖 10 則為階層式連線排程演算法的例子，並說明如下：

- (1) 節點 E、F 向節點 C 提出頻寬要求並將本身排程資訊告知節點 C，如圖 9(a)。
- (2) 節點 C 收到 E、F 的排程資訊後，將自己的頻寬需求加入排程，接著將子節點 E、F 本身要傳送的資料量加入排程，改其所屬集合為 C 的所屬集合，將 hop-count 加 1，之後將 E、F 的排程加入節點 C 的排

表 2. 階層式連線排程演算法

```

假設節點 i 擁有  $c_1, c_2, \dots, c_k$ ，共 k 個子節點
R(i): 節點 i 所提出的節點 i 所提出之傳輸時槽需求
Sch_Tab[n]: 節點 i 所記錄之包含 ni 個元素的排程表，表中每個元素包含
  nodes: 節點集合
  slots: 分配的傳輸時槽數量
  hop_count: 表示資料到達順序
當節點 i 收到來自 k 個子節點的傳輸時槽需求與排程表
index  $\leftarrow 1$ 
Sch_Tab[index].nodes  $\leftarrow i$  //將節點 i 的需求加入排程表
Sch_Tab[index].slots  $\leftarrow R(i)$ 
Sch_Tab[index].hop_count  $\leftarrow 0$ 
index  $\leftarrow index + 1$ 
for j=1 to k do
  Sch_Tab[index].nodes  $\leftarrow i$  //將節點 cj 的需求加入排程表
  Sch_Tab[index].slots  $\leftarrow R(c_j)$ 
  Sch_Tab[index].hop_count  $\leftarrow 1$ 
  index  $\leftarrow index + 1$ 
end for
//在 Sch_Tabc1, Sch_Tabc2, ..., Sch_Tabck 的元素中，若其 nodes 欄位資料在同一個傳輸集合，而且 hop_count 欄位的資料相等，則為可合併的元素 //
while Sch_Tabc1, Sch_Tabc2, ..., Sch_Tabck 中存在可合併的元素
  Sch_Tab[index].nodes  $\leftarrow$  可合併節點的聯集
  Sch_Tab[index].slots  $\leftarrow \text{MAX}$ (可合併節點的傳輸時槽需求)
  Sch_Tab[index].hop_count  $\leftarrow$  可合併元素的 hop_count
  index  $\leftarrow index + 1$ 
end while
while Sch_Tabc1, Sch_Tabc2, ..., Sch_Tabck 中存在其它不可合併的元素，假設為 Sch_Tabc[k]
  Sch_Tab[index].nodes  $\leftarrow$  Sch_Tabc[k].nodes
  Sch_Tab[index].slots  $\leftarrow$  Sch_Tabc[k].slots
  Sch_Tab[index].hop_count  $\leftarrow$  Sch_Tabc[k].hop_count
  index  $\leftarrow index + 1$ 
end while
temp  $\leftarrow index$ 
for p=1 to index-1 do
  if Sch_Tab[p].nodes  $\neq \{c_1, c_2, \dots, c_k\}$  and Sch_Tab[p].hop_count  $\neq 0$  then
    Sch_Tab[temp].nodes  $\leftarrow i$ 
    Sch_Tab[temp].slots  $\leftarrow$  Sch_Tab[p].slots
    Sch_Tab[temp].hop_count  $\leftarrow$  Sch_Tab[p].hop_count + 1
    temp  $\leftarrow temp + 1$ ;
  end for
index  $\leftarrow temp - 1$ 
將 Sch_Tab[ ] 中的元素依照 hop_count 欄位之值由小至大排序

```



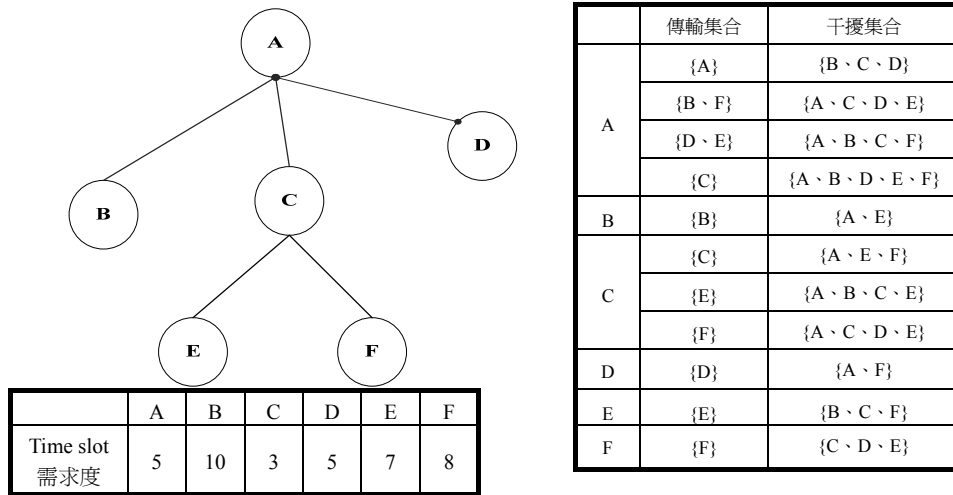


圖 8. 各節點的集合資訊與傳輸時槽需求

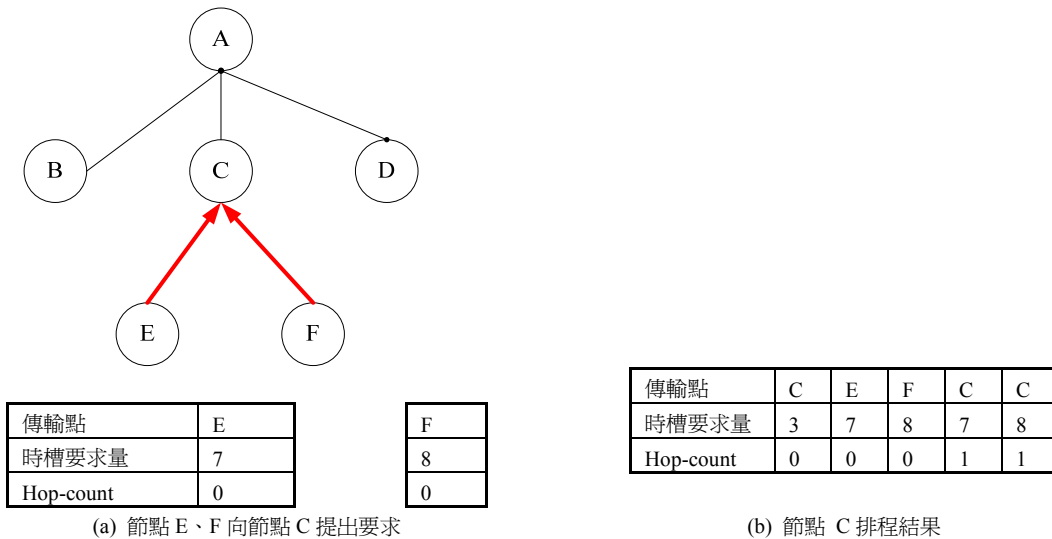


圖 9. 節點 C 收到節點 E、F 的要求後建立本身的排程

程。依 Hop-count 由小到大排序後即完成節點 C 的排程。圖 9(b) 為節點 C 之排程。

- (3) 節點 A 受到來自節點 B、C、D 的排程後，將自己的頻寬需求加入排程，接著將子節點 B、C、D 本身要傳送的資料量加入排程，改其所屬集合為 A 的所屬集合，將跳躍節點數加 1，之後節點 A 依照本身的集合資訊，將 A、B、C、D 的傳輸集合做合併，合併原則為排程中任兩個可合併的傳輸集合且跳躍節點數相同的時槽可同時傳輸，將 A、B、C、D 合併完後的排程加入節點 A 的排程中，最後將排程中，屬於子節點集合且跳躍節點數不等於 0 的時槽分配複製一份，改成

節點 A 所屬集合且跳躍節點數+1，加入節點 A 排程，依跳躍節點數由小到大排序後即完成節點 A 的排程，圖 10 為節點 A 之排程流程。

#### 四、實驗分析

在本節中，我們將透過程式模擬的方式，來證明本研究提出的方法可以有效的增加網路傳輸量，提高頻寬的重複利用率。

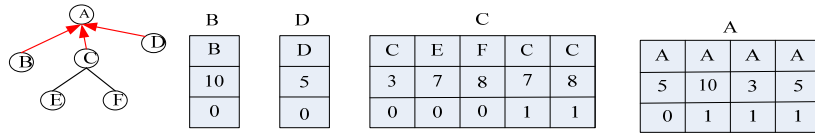
##### (一) 建立樹狀拓模與傳輸集合選擇

如圖 11(a) 所示，實驗模擬時所考慮之網路傳輸環境近似曼哈頓 (Manhattan-like) 地區，具有建築物密集且街道



Step 1

節點A受到來自節點B、C、D的排程後，將自己的頻寬需求加入排程，接著將子節點B、C、D本身要傳送的資料量加入排程，改其所屬集合為A的所屬集合，hop-count+1



Step 2

由節點A的傳輸集合資訊可知節點B與F及節點D與E是可同時傳輸的，因此依合併原則將A B C D四個排程做合併。此時節點A的排程如下

A	A	A	A	B、F	D、E	C	C	C
5	10	3	5	10	7	3	7	8
0	1	1	1	0	0	0	1	1

Step 3

在節點A目前的排程中，若有其子節點所屬的傳輸集合且hop-count不為0的時槽分配，將此分配複製一份，改成節點A所屬集合且hop-count+1加入節點A的排程，此時節點A的排程如下

A	A	A	A	B、F	D、E	C	C	C	A	A
5	10	3	5	10	7	3	7	8	7	8
0	1	1	1	0	0	0	1	1	2	2

Step 4

將節點A的排程依hop-count的大小順序作排序後即完成節點A的排程步驟

A	B、F	D、E	C	C	C	A	A	A	A	A
5	10	7	3	7	8	10	5	3	7	8
0	0	0	0	1	1	1	1	1	2	2

圖 10. 節點 A 之排程步驟

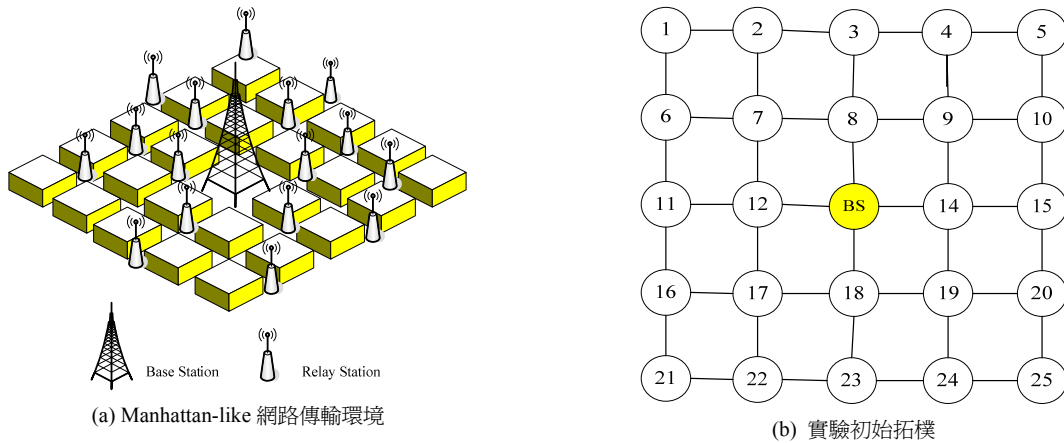
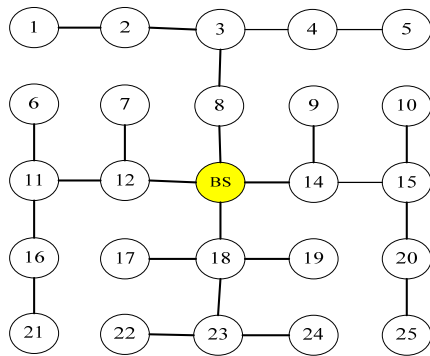


圖 11. 實驗中網路的架構

交錯之都市特徵 [4]，而中繼站則建置於街道交叉路口。本研究即在此種網路環境情況下模擬 24 個中繼站與一個 BS 所構成的樹狀網路拓樸，圖 11(b) 為實驗初始的網路拓樸，圖 12 則為經由本研究提出的方法所建置出的樹狀網路拓樸。圖 12(a) 中，我們可以看出樹狀拓樸略成對稱狀，這是因為在建置網路的方式是由 BS 所發起，再逐一的向外通知

其他尚未加入網路的中繼站加入網路，因此，當新節點收到兩個以上路徑干擾值相同的中繼站所送出的連線訊息時，新節點會選擇最先收到連線訊息的節點作為父節點，所以樹狀拓樸不會是完全的對稱性拓樸。而實驗結果我們也發現此種樹狀網路拓樸的建置方式能有效避免節點選擇過長的路徑做連結，提高路由的服務品質。



(a) 本研究建置出之樹狀拓模

傳輸集合
1、4、11、15、23
2、5、12、19、22
8、10、25
7、16、18
9、17、20、24
3、6、14、21

(b) BS 所記錄的傳輸集合表

圖 12. 樹狀拓模與 BS 所記錄的傳輸集合表

在傳輸集合選擇方面，圖 12(b) 為 BS 最後整合所有子節點的集合資訊後，所得到之各節點所屬的傳輸集合表，在此集合表中，我們發現在各集合中，可傳輸的節點數量約成均勻的分布，而不會有某節點獨自形成一個傳輸集合的情形，因此我們相信本研究在頻寬的重複利用率上，能獲得不錯的效果。

(二) 連線排程實驗分析

在實驗環境中，一開始每個中繼站會產生本身所要求的資料傳輸量，之後依中繼網路 (relay network) 運作的方式，將所有節點的傳輸要求一路傳遞至 BS，再由 BS 決定好所有連線的時槽排程方式，在排程方面，實驗中將以本研究提出的階層式排程方式與修改之集中式排程分配方式所呈現的實驗結果來分析並且與每個時槽僅有一個節點傳送資料的方法 (圖中稱為非合併法) 做對照。圖 13 為網路中提出要求的節點分布範圍不同的情況下所得的實驗結果。由圖 13 中可知，當提出要求的節點分布集中時，傳送完所有資料量所需的時槽數會高於節點分散時所使用的時槽數，這是因為當有傳輸需求的節點分布集中時，其節點間的干擾程度會使得可合併的時槽數減少，所以使用時槽數較高，然而不管是修改之集中式排程或是階層式排程，都能有效的減少使用時槽數。

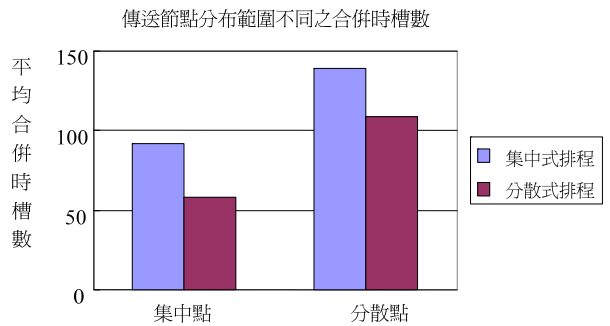
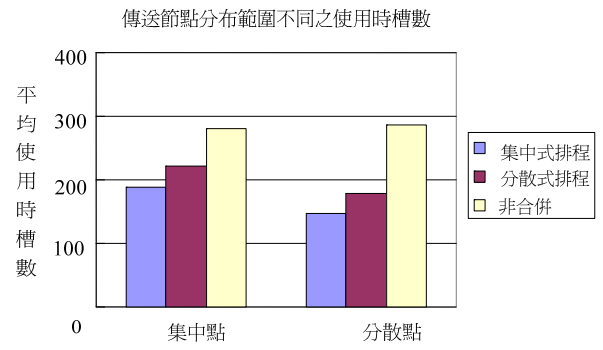


圖 13. 不同分布範圍內的使用時槽數與所合併的時槽數

接下來，在實驗中，我們讓各節點隨機產生介於我們限定範圍內的資料量，以比較在資料量變動情況下修改之集中式排程與階層式排程所得的結果，圖 14 為網路中各節點初始資料量變動下的實驗結果，我們發現，雖然修改之集中式排程與階層式排程均能有效的降低使用時槽數，提高傳輸效能，然而隨著資料量變動範圍的增加，修改之集中式排程與階層式排程的效能差距也會隨之增加，這是因為在連線排程的設計上，階層式排程在事先便已經決定好可同時傳輸的節點集合，且為了避免承接下層資料量時所產生的問題，階層式排程限制 hop-count 數相同的時槽才能做合併，由於多了此兩項限制，所以使得修改之集中式排程分配在實驗中所產生的結果會優於階層式排程，圖 15 為此實驗中，本研究所提出的修改之集中式與階層式排程分配對於網路效能提升率的比較。

五、結論與未來展望

為了拓展無線網路的服務範圍，現今的無線接取網路的架構採用佈建基地台的方式來獲得更廣的訊號涵蓋範圍，然而基地台的建置成本太高，且隨著基地台數量的增多，網路上的干擾程度也隨之增加，影響傳輸效能，本研究藉由使用

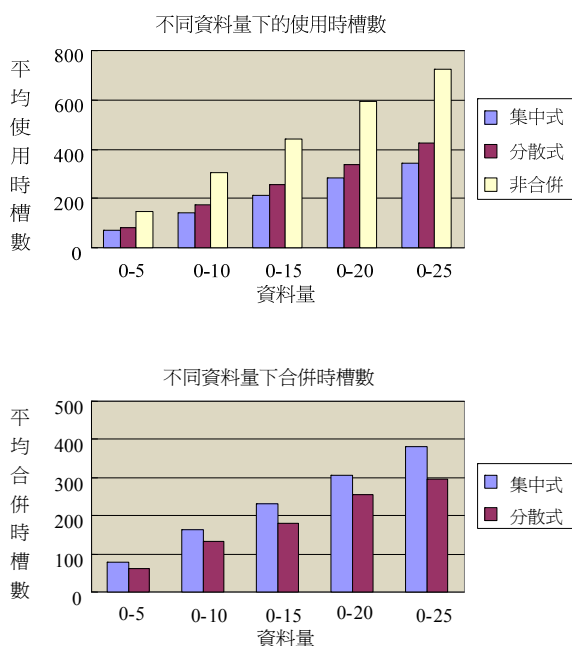


圖 14. 不同資料量下所使用的時槽數與所合併的時槽數

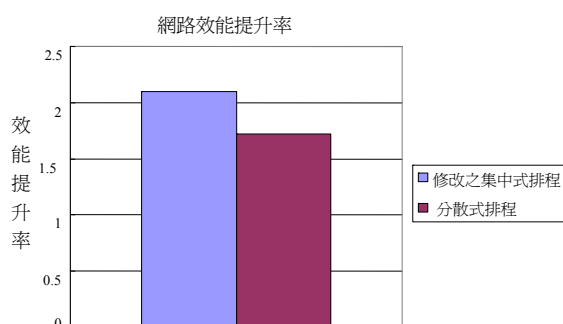


圖 15. 網路效能提升率比較圖

無線多躍程中繼網路的網路架構，以考慮干擾值的方式將網狀拓模化為樹狀拓模，以降低網路的干擾程度，並提出階層式的連線排程方式以提高網路的傳輸效能，並證明其確實能有效的提高網路傳輸量，改善網路的干擾程度，而在未來發展上，除了繼續研究改善階層式連線排程的方式來趨近集中式排程分配的傳輸效能外，也將設計一個合適的允入控制機制，以達到在有限的頻寬資源內，能滿足多數使用者的傳輸需求，提高服務品質。

## 誌謝

本研究承蒙世新大學學術研究專案補助（編號：P9510）始得以順利完成。

## 參考文獻

- Asa, M., R. Peterson, S. Ramachandran, D. T. Chen and N. Natarajan (2005) Recommendations for the Scope and Purpose of the Mobile Multihop Relay Study Group. Document IEEE C802.16mmr-05/032.
- Akyildiz, I. F., X. Wang and W. Wang (2005) Wireless mesh networks: A survey. *Computer Networks*, 47, 445-487.
- Alicherry, M., R. Bhatia and L. Li (2005) Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. ACM International Conference on Mobile Computing and Networking, Cologne, Germany.
- Cheng, S. M., D. W. Huang, P. Lin and S. R. Yang (2006) A study on distribute/centralized scheduling for wireless mesh network. IEEE The International Wireless Communications and Mobile Computing Conference, Vancouver, Canada.
- Chen, D., I. K. Fu, M. Hart and W. Wong (2006) Channel models and performance metrics for IEEE 802.16j Relay Task Group. Contribution of IEEE 802.16 Broadband Wireless Access Working Group: IEEE C80216j-06/020.
- Draves, R., J. Padhye and B. Zill (2004) Routing in multi-radio, multi-hop wireless mesh networks. ACM International Conference on Mobile Computing and Networking, Philadelphia, PA.
- Fu, L., Z. Cao and P. Fan (2005) Spatial reuse in IEEE802.16 based wireless mesh networks. IEEE International Symposium on Communications and Information Technologies, Beijing, China.
- IEEE Std. 802.16j-2009 (2009) *IEEE standard for local and metropolitan area networks Part 16: Air interface for broadband wireless access systems, amendment 1: multiple relay specification*, The Institute of Electrical and Electronics Engineers, NY.
- Nahle, S. and N. Malouch (2008) Joint routing and scheduling for maximizing fair throughput in WiMAX mesh network. IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, Cannes, France.
- Peters, S. W. and R. W. Heath (2009) The future of WiMAX: Multihop relaying with IEEE 802.16j. *IEEE Communications Magazine*, 47, 104-111.

- 
11. Raniwala, A., K. Gopalan and T. Chiueh (2004) Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM Mobile computing and communication Review*, 8, 50-65.
  12. Shetyia, H. and V. Sharma (2005) Algorithms for routing and centralized scheduling to provide QoS in IEEE 802.16 Mesh Networks. The 8th ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Montreal, Canada.
  13. Tang, J., G. Xue and W. Zhang (2005) Interference-aware topology control and QoS routing in multi-channel wireless mesh networks. ACM International Symposium on Mobile Ad Hoc Networking and Computing, Urbana-Champaign, IL.
  14. Wei, H., S. Ganguly, R. Izmailov and Z. J. Haas (2005) Interference aware IEEE 802.16 WiMax mesh networks. IEEE Vehicular Technology Conference, Stockholm, Sweden.
  15. Wongthavarawat, K. and A. Ganz (2005) Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems. *International Journal of Communication Systems*, 1(16), 81-96.
  16. WiMAX Forum (2006) Mobile WiMax part I: A technical overview and performance evaluation. Retrieved March 20, 2006, from [http://www.wimaxforum.org/technology/downloads/Mobile\\_WiMAX\\_Part1\\_Overview\\_and\\_Performance.pdf](http://www.wimaxforum.org/technology/downloads/Mobile_WiMAX_Part1_Overview_and_Performance.pdf).

收件：98.04.11 修正：98.09.07 接受：98.12.30