

Designing Cost-Efficient Wireless Mesh Networks

無線網狀網路之成本最佳化設計

洪啟堯¹

許俊彥²

陳金蓮¹

國立台灣科技大學電子工程系¹

國立台灣科技大學資訊工程系²

A9302033@mail.ntust.edu.tw

yen@nlhyper.et.ntust.edu.tw

jcw@et.ntust.edu.tw

摘要

目前無線都會網路主要用途定位於取代有線網路之“最後一哩”，可作為連接終端設備與有線骨幹網路之中繼媒介，然而無線骨幹網路頻寬受限於無線技術及傳輸距離等因素影響，拓樸安排方式成為整體網路建置成本之關鍵，但目前尚未有一套方法能解決此問題。

本論文主旨在於無線網狀網路之拓樸設計，提出預先定義開道集合演算法(pre-defined gateway set algorithm, PGSA)。PGSA 採用繞徑方式有效安排網路流量路徑，並以基因演算法為基礎，試圖求得最小建置成本之網路拓樸，可有效解決上述議題。在繞徑演算法中，我們考慮使用最短路徑演算法及最小成本演算法。此演算法具有不同的搜尋特性，且皆能在一定時間內求得無線都會骨幹網路拓樸中各個節點所需花費之天線數，及各節點其是否需花費成本佈置有線網路至外部網路，將有助於提昇網路規劃之有效性。

本文描述預先定義開道集合演算法之設計概念與適用環境，並在各種網路限制與流量需求下比較兩種繞徑演算法所得之結果。

關鍵詞：無線都會網路、無線網狀網路、網路拓樸、預先定義開道集合演算法、基因演算法

Abstract

Mesh topology provides a robust structure for wireless metropolitan area backbone networks. Wireless mesh network allows for a rapid and cost-effective deployment of large numbers of clients throughout a community, without the need for hard-wired backhaul. In this paper, the pre-defined gateway set algorithm is proposed to solve the topology arrangement problem in the above environment. The proposed mechanism adopts the genetic algorithm and a Dijkstra-based routing algorithm to obtain good results effectively. Two routing strategies, namely, shortest path and

minimum cost, are used. We evaluate the proposed algorithm by simulation. The result shows that using minimum cost routing strategy outperforms using the shortest path one and ensures that network construction cost is kept significantly in most cases.

Keywords: wireless metropolitan area network, wireless mesh network, network topology, pre-defined gateway set algorithm, genetic algorithm

一、前言

由於無線網路應用服務之需求快速成長，各種無線網路的通訊協定亦隨而蓬勃發展，近年來以無線都會網路(metropolitan area network, MAN)取代有線網路作為中繼傳輸媒介之相關議題受到相當大的注目。IEEE 自 1999 年 7 月迄今制定了無線寬頻網路 IEEE 802.16 標準 [1][2]，又名無線都會網路(Wireless MAN)，其具高頻寬、良好的空中長距離傳輸能力、及保障服務品質(Quality of Service, QoS)等特性，預計可以其低成本的優勢取代傳統有線連結方式。

現今無線都會網路之寬頻無線存取技術(broadband wireless access)正引起國際上高度重視與關切，特別是針對網狀(mesh)架構的支援，如 2005 年 5 月於義大利舉行之會議中，日本 KDDI R&D Labs 提出 802.16 之網狀網路相關問題，並向 IEEE 802.16 工作群組建議成立一新的研究小組 [3]，特別針對無線寬頻網路之網狀架構議題作更深入之探討。此外，國內如工研院電通所等機構目前亦積極爭取參與相關之標準制定。

本文選擇以 IEEE 802.16 之無線網狀網路(wireless mesh network, WMN)為基本架構，圖 1 為一無線網狀網路之基本架構，無線骨幹網路(wireless backbone network)之節點(node)稱為無線路由器(wireless router)，其中有一台以上之無線路由器需作為開道器(gateway)，以有線方式連結至外部網路，處理此大都會區域之進出口流量。然而，此段對外的有線寬頻線路，往往因其高頻寬之需求特性而需埋設光纖及加裝光轉接器等昂貴設備，以

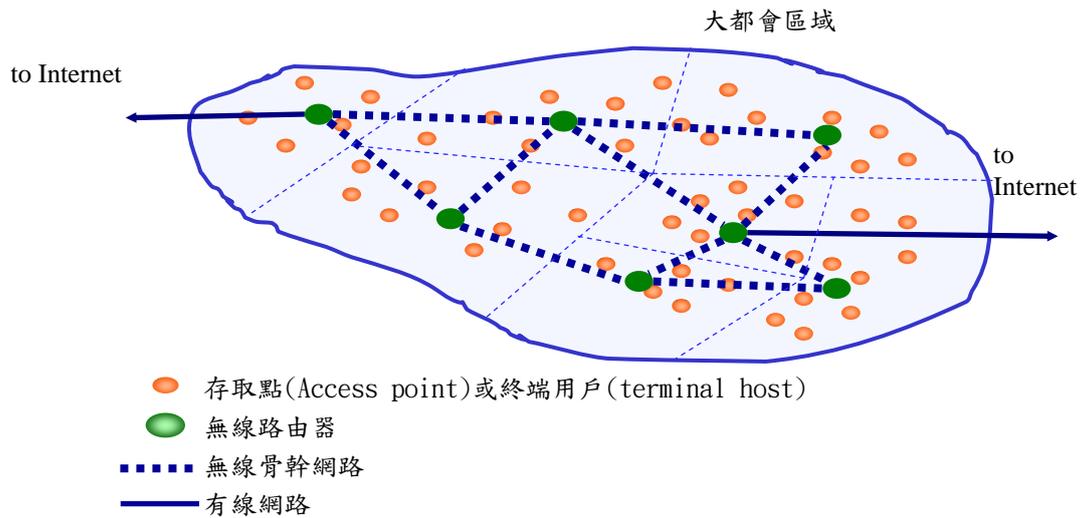


圖 1 無線網狀網路實體架構圖

提供如 T3 或 OC-1 等級之高頻寬服務，因而增加網路建設成本。故本文提出一套預先定義閘道集合演算法(pre-defined gateway set algorithm, PGSA)解決此問題，其首要目標即為在該無線存取系統之限制環境下，判斷出此網狀網路所需之最小閘道器數量，次要目標則為在此閘道器數量限制下，應當選擇哪些無線路由器作為閘道器，建置對外有線網路，以降低各節點之天線使用總數為較佳解。此外，天線可分為指向性(directional)天線及全向性(omni-directional)天線，由於指向性天線之方向增益較高、傳輸距離較遠、適用於點對點(point-to-point)直視(line-of-sight, LOS)傳輸等特性符合本架構需求，故我們選擇以指向性天線作為本系統之基本傳輸設備。

此類多目標函數之拓樸設計問題屬於一種 NP-complete 問題，我們可使用啟發式的方法(heuristic techniques)來解決，以降低此問題之計算時間複雜度。PGSA 之搜尋階段亦包含搜尋最佳解部分，本文係採用基因演算法(genetic algorithm, GA)搜尋最佳拓樸。PGSA 另一搜尋階段為搜尋閘道器總數部分，以給定之節點優先順序分別使用兩種繞徑演算法作為基因演算法之合適度函數(fitness function)計算依據。此兩種演算法皆以 Dijkstra 最短路徑演算法為基本架構修改而成，主要為符合針對此特定架構之需求與限制條件。第一種演算法稱為最短路徑演算法(Shortest Path algorithm, SPA)，以各節點距離閘道器之 hop 數作為繞徑優先考量，而第二種演算法為最小成本演算法(minimum cost algorithm, MCA)，則優先考慮繞行較大頻寬之路徑。我們模擬實際設計情形，並評估 PGSA 演算法所求得之拓樸成本比較。

以下第二節為相關理論，將簡短地回顧傳統基因演算法與各項基因演算法改良機制，再下節則將詳細說明本文所提出之 PGSA 演算法，並說明此演算法使用之基因演算法與傳統基因演算法之差異所在。第四節說明模擬結果及其比較，第五節為結

論與未來之展望。

二、相關理論

本節將介紹相關理論之運作方式，第一小節介紹簡單基因演算法(Simple Genetic Algorithm, SGA)運作流程，第二小節則說明各項改良型基因演算法之基本概念。

2.1 簡單基因演算法

傳統 SGA 之基本理論由 Holland 於 1975 年所提出[4]，其仿效生物界物競天擇、適者生存之演化方式，是種基於自然演進之最佳化問題隨機搜尋機制。但 SGA 不同於一般隨機搜尋方式，其引入了生物演化概念，可適當地調整搜尋方向與區域，使整體搜尋較不易陷入局部最佳解，並能在合理的有限時間內，求得符合目標函數及限制式之較佳解，故此機制廣泛的被應用在搜尋資料量龐大之最佳化問題。

SGA 之基本架構係由一群族群(population)進行世代(generation)演化，族群由多個個體(individual)所組成，每個個體有獨立的染色體(chromosome)，代表此個體的特性，亦可當成一個搜尋點，以一個二進制位元字串表示，其中各個字元為'1'或'0'分別表示個體之對應的遺傳基因(gene)為顯性(dominant)或隱性(recessive)。每一世代演進過程包含合適度計算、複製、交配及突變等過程，如圖 2 所示，為 SGA 運作流程圖。

合適度計算(fitness calculation)需定義合適度函數(fitness function)，此函數將依其目標函數及限制式來定義，合適度愈高之個體競爭力越強，存活至下一世代之機率相對較高。

複製(reproduction)過程採用輪盤選擇法(Roulette Wheel Method)來挑選進入交配池(mating

pool)之個體[5]，如式 1 所示，為任一個體被複製到進入交配池之機率。其中 f_i 為第 i 個節點之合適度。

$$P_i = f_i / \sum_{i=1}^n f_i \quad (1)$$

單點交配(one-point crossover)為隨機於交配池內取得個體並兩兩配對，配成一對的個體稱為雙親，再以交配率 P_c 決定是否要交換基因，要交換基因之組合則需再隨機於雙親染色體之基因中任意挑選一點，作為交配點(crossover point)，最後將交配點右側的雙親基因互換，如此生成兩個新的子代(child)。

突變(mutation)則以預設之突變率 P_m 來決定子代是否要進行突變，做法為隨機把基因子代的某一基因之位元反轉(0 變成 1、1 變成 0)。

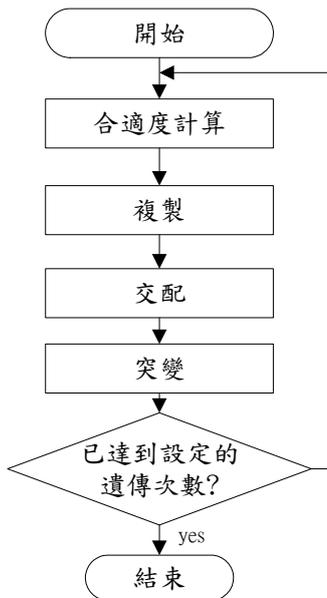


圖 2 簡單基因演算法架構流程圖

2.2 各項基因演算法改良機制

許多研究顯示，傳統 SGA 容易陷入不成熟的收斂(premature convergence)而得到局部最佳解，近年來有許多用於改良 SGA 的新機制被提出，然其大部分做法仍保留原有 SGA 運作架構，僅在舊有架構間加入新的機制，如合適度調整、均勻交配、菁英政策、汰新政策等機制[6][7]。

合適度調整(fitness scaling)係指族群中所有個體合適度若過於相近或差異過大，易造成收斂過慢或不成熟收斂現象，此方法可依來源資料差異程度作適度的調整，使個體間合適度較接近於平均分布。

均勻交配(uniform crossover)做法為隨機產生一串與雙親長度相同的樣本字串(template string)，在該樣本字串中所有字元為'1'之位置，雙親需在這些相對應的位置互相交換字元。

菁英政策(elitist strategy)選定某一比例之個體數目作為菁英個體，菁英在這一世代將不需再進行複製、交配、突變等過程，並保證這些菁英能存活至下一世代。

汰新政策(extinction and immigration strategy)可避免所有個體之染色體基因過於相近，而侷限於局部搜尋之問題，當所有個體之染色體基因皆相同時，則將族群中一半的個體以新的個體取代；當連續數個世代，族群中最大合適度皆相同時，則將現今族群中小於族群合適度平均值之個體置換成新的個體。

三、預先定義閘道集合演算法

本文提出一套在 IEEE 802.16 之無線網狀網路拓樸安排方法，如圖 3 所示，為 PGSA 流程圖。首先需給定相關參數，演算法根據給定之參數建立對應的拓樸，此為拓樸初始化階段。建立拓樸後，進入第一階段搜尋，此階段為測試此網路限制條件下所需建設之閘道器最小數量；第二階段搜尋則採用基因演算法搜尋最佳解，以第一階段所找出之閘道器總數為基準，搜尋整體網路使用最少天線數量之拓樸型式。最後結果將輸出顯示及記錄。

本文以下定義 PGSA 所使用之資料結構及圖 3 中各階段之細部運作說明。

(一) 染色體架構

本文定義染色體字串由兩個部分所組成，此兩部份之字串長度皆與網路中節點總數 n 相同。如圖 4 所示，第一部份每個字元分別對應至網路中各個節點，若該節點是閘道器，則以'1'表示，反之

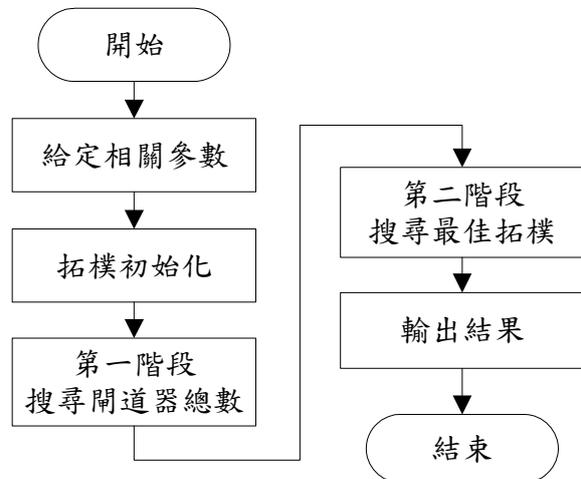


圖 3 PGSA 流程圖

則以'0'表示。第二部分用於表示每個節點繞徑處理之優先順序，故每個字元亦分別對應至網路中各個節點，以不重複的'0'至'n-1'表示，數字越小表示優先順序越高。第二部分定義與一般基因演算法差異頗大，一般基因演算法之染色體編碼僅限使用'0'或'1'。如圖 4 所示，為此型式下的染色體字串範例，其中 $n=4$ 、且僅有一台閘道器。

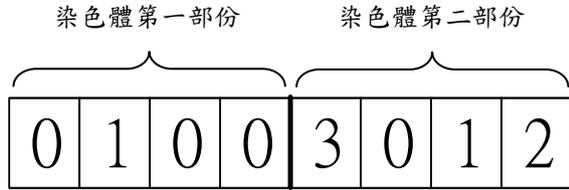


圖 4 染色體字串結構範例

(二) 拓樸初始化

我們假設網路中各節點之位置及其負載已知，則各節點間距離亦可求得，並考慮採用 IEEE 802.16 支援之正交頻率多工(orthogonal frequency division multiplexing, OFDM)系統，其各種調變方式對應之傳輸頻寬與接收端所要求之最低訊號雜訊比(signal-to-noise ratio, SNR)如表 1 所示。

有鑑於無線骨幹網路之高頻寬需求，我們選擇使用增益較高之指向型天線，如此可大幅減少相互間之干擾，惟若有二組以上天線過於接近，干擾仍可能發生，此時各組天線可使用不同頻率，以避免兩組有向性天線之夾角過於接近所造成之干擾問題。在上述前提下，本文考慮適用於指向性天線之直視長距離傳輸模型 Friis free space 方程式[8]：

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \quad (3)$$

其中 P_r 為接收端接收功率、 P_t 為傳送端發送功率、 d 為傳送端與接收端之間距離、 G_t 為傳送端天線增益、 G_r 為接收端天線增益，而 λ 則為波長。且

$$SNR = 10 \log \left(\frac{P_r}{Noise} \right) \quad (4)$$

由(3)、(4)式考慮距離 d 與 SNR 關係時即可得：

$$d \propto 10^{-SNR/20} \quad (5)$$

由式5之結果，可求得在各種調變方式下相對應的天線涵蓋距離索引，記於表1末欄。為求簡化模擬，我們忽略 P_t 、 G_t 、 G_r 及 $Noise$ 影響，以涵蓋範圍索引(Range index)作為各調變方式之相對傳輸距離，在實際評估時則可依據所選用之實體參數帶入，求得各調變方式所對應之實際涵蓋距離。

拓樸初始化時，本演算法即查詢此表，以紀錄各條連線可建立之最大頻寬。

表 1 802.16之調變方式比較表[9]

Modulation	Coding rate	Peak data rate in 5MHz(Mb/s)	Receiver SNR	Range index
BPSK	1/2	1.89	6.4 dB	478.6
QPSK	1/2	3.95	9.4 dB	338.8
	3/4	6.00	11.2 dB	275.4
16-QAM	1/2	8.06	16.4 dB	151.3
	3/4	12.18	18.2 dB	123.0
64-QAM	2/3	16.30	22.7 dB	73.2
	3/4	18.36	24.4 dB	60.3

(三) 第一階段搜尋—搜尋閘道器總數

本階段之主要目標在於測試此網路限制條件下所需使用之所需閘道器最小數量，如圖 5 所示，為本階段流程圖。本階段中採用一個變數 g ，以表示此網路之閘道器數量測試值。首先，將此變數設為最大值，即等於網路總節點數 n 減去 1，並依設定之族群大小產生相對應之染色體數量，分別以繞徑演算法為基準，來進行流量安排¹。若任一個體有解，則將 g 變數遞減，並重新初始化族群基因，再次進行測試，直至發生所有個體皆無解時，再將 g 值增加 1，則此拓樸之最小閘道器總數即為此值，並記錄此值，傳遞給第二階段搜尋。

(四) 第二階段搜尋—搜尋最佳拓樸

第二階段搜尋採用基因演算法搜尋最佳拓樸，為符合第一階段所求得之閘道器總數條件，我們限制基因演算法之染色體字串正位元數，將搜尋範圍限定於前一階段所傳入之網路閘道器總數。圖 6 為第二階段搜尋流程圖，經流量安排取得最佳解，由合適度計算求得各個體對應之合適度，再經由菁英政策、複製、均勻交配、突變、汰新政策等機制後，即完成一個世代的搜尋。反覆搜尋直至發現連續五十個世代所得解品質皆相同時，則認定所得解已收斂並結束第二階段搜尋。為符合第一階段搜尋之閘道器總數限制，在第二階段搜尋中，我們固定基因演算法之染色體字串正位元數，SGA 之交配、複製、汰新政策等步驟亦有所修改，其餘步驟則與傳統 SGA 相同，圖六中以*標註者，表示該步驟與 SGA 做法相異，以下各小節將詳述其觀念與差異所在。

(a) 合適度計算

若某個體繞徑失敗，則設定其合適度為一微小值 ϵ ，以降低此個體被複製的機率；若該基因繞徑成功，則以其成本之倒數作為合適度函數，即：

¹ 流量安排與二種繞徑演算法將分別說明於附錄

$$fitness = 1/cost \quad (6)$$

而個體之成本可以下式表示：

$$cost = \sum_{i=1}^n \sum_{j=1}^n A_{ij} \quad (7)$$

其中 $A_{ij} = \begin{cases} 1, & \text{if the link } i \text{ to } j \text{ is established} \\ 0, & \text{otherwise} \end{cases}$ 。在第一階段搜尋中，我們已確定拓樸需花費的開道器最少數量，因此在第二階段搜尋中，所有個體求

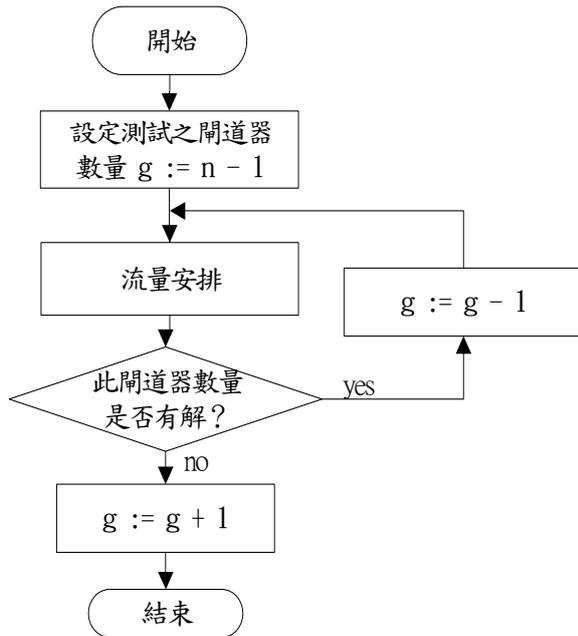


圖 5 第一階段搜尋流程圖

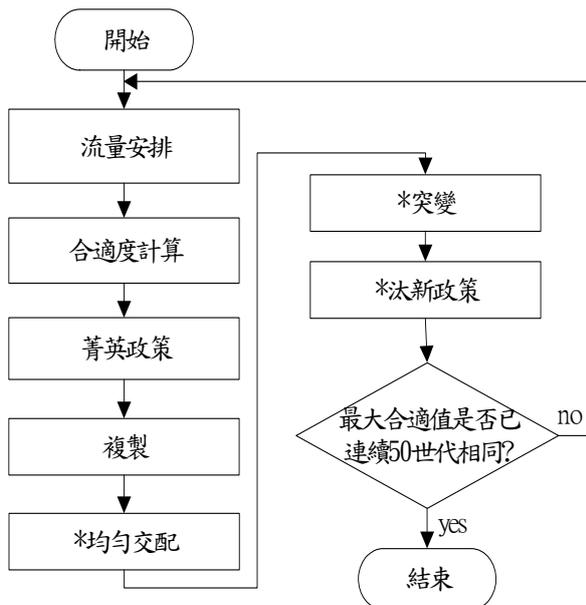


圖 6 第二階段搜尋流程圖

得之解花費的開道器總數必相同，故成本函數以個體花費之天線數為基準，並以成本之倒數作為合適度。換言之，越少天線數之個體將可獲得較高的合適度。

(b) 均勻交配

前之研究報告中顯示，均勻交配優於 SGA 之單點交配[10]，因此本文修改均勻交配機制，採用固定正位元數染色體架構之均勻交配方式，以維持網路開道器總數不變。在個體之染色體第一部份中，由於須維持子代的顯性基因數與親代相同，因此當雙親某一基因相異時，我們以隨機方式選擇子代相對應位置的基因，並於每一對親代交配完成後，檢查子代染色體第一部份之顯性基因總數是否與親代相符，若不符合則重試。在染色體第二部份中，由於交配後所產生之子代可能發生兩個節點有相同優先處理次序的問題，因此在交配染色體第二部分時，進行交配的基因將以亂數加減一固定偏移值。此偏移值範圍需大於 0 且小於 1，且均勻交配後將染色體重新排序，如此可避免均勻交配產生相同優先順序之衝突問題。

(c) 突變

傳統的突變機制在突變之後可能造成開道器總數變化，因此我們使用基因交換的方式作為突變機制的替代方案，即當染色體中某個基因突變時，此基因將隨機選擇染色體中同一部份之任一基因進行交換，如此運作既能維持本文所定義之染色體限制，也保留原先突變程序之意涵。

(d) 汰新政策

世代最大合適度表示在某一世代中之所有個體合適度最高者。當連續二十個世代之世代最大合適度皆相同時，在此世代暫時大幅調高突變率，以汰換目前個體之大部分基因，避免搜尋點過於雷同使族群多樣性較差，而無法搜尋到最佳解之問題。且由於菁英政策的保護，目前搜尋到的優秀個體將有一定的比例不受汰新政策影響。在汰新政策執行後的下次世代，突變率將恢復正常。

由於基因演算法需設定世代數目，當到達指定的世代時，從族群中挑出最優良的個體，作為基因演算法之解。若世代數目設得太少，有可能在解還沒收斂時就結束搜尋；若世代數目設得太多，可能會浪費搜尋時間。因此，基於汰新政策等機制支援下，我們選擇當連續五十個世代，其最大合適度皆相同時，才認同目前之解已收斂並結束搜尋。

四、模擬結果

本文之模擬環境以表 1 中各種調變方式所對應之天線涵蓋範圍索引為基準，設定一長與寬為 2000 相對距離之區域。在此區域隨機安置 50 個節點後，以兩種拓樸安排演算法來求得理想的拓樸安排，並在有限計算時間內，得知此無線都會網路之最低成本要求。我們分別測試各項參數之設定值對結果之影響，如交配率 P_c 、突變率 P_m ，菁英比率 P_e 等參數後，並決定以表 2 所列之參數進行模擬。

表 2 基因演算法參數

族群大小 m	100、300 及 500
交配率 P_c	0.8
突變率 P_m	0.001
汰新突變率 P_{im}	0.2
菁英比率 P_e	0.02
節點最大天線 上限數	6

此外，族群大小的選擇將影響結果的品質好壞，一般而言，族群越大所花費的計算量越大，即所需計算時間愈長，但能得到較佳解。本文所考量的應用環境為規劃階段，不需即時性的判斷求解，求解品質之重要性亦高於花費的計算時間，但仍需在合理的有限時間內求得理想之拓樸安排方式。由於本研究提出之機制適用於網路拓樸之事先規

劃階段，因此忽略網路在實際運作之流量變化情形，而以一預估最差情況之網路流量最大值作為各節點之流量需求(flow demand)。本文定義四種流量需求等級，如表 3 所列，在所屬情形下，以均勻分布隨機設定各節點之流量需求。

表 3 流量需求等級及需求對應表

等級	流量需求範圍(Mbps)
Light Load	0-3
Medium Load	2-5
Heavy Load	4-7
Extra Heavy Load	6-9

此外，我們重新設定整體網路成本函數，與式 7 不同之處在於加入考量閘道器數量之影響，如式 8 表示：

$$\text{cost} = \sum_{i=1}^n (\alpha G_i + \sum_{j=1}^n A_{j,i}) \quad (8)$$

n 代表網路節點數， $G_i = \begin{cases} 1, & \text{if node } i \text{ is a gateway} \\ 0, & \text{otherwise} \end{cases}$ 。由於閘道器成本為第一優先考量因素，因此在(8)式中，設定 $\alpha = n \times k$ ，其中 k 為任一節點之天線數上限值，表示一個閘道器成本為一只天線的 $n \times k$ 倍，換言之當 $\alpha \geq n \times k$ 時，無論天線數使用量多寡，使用較少閘道器之成本必小於使用較多閘道器。

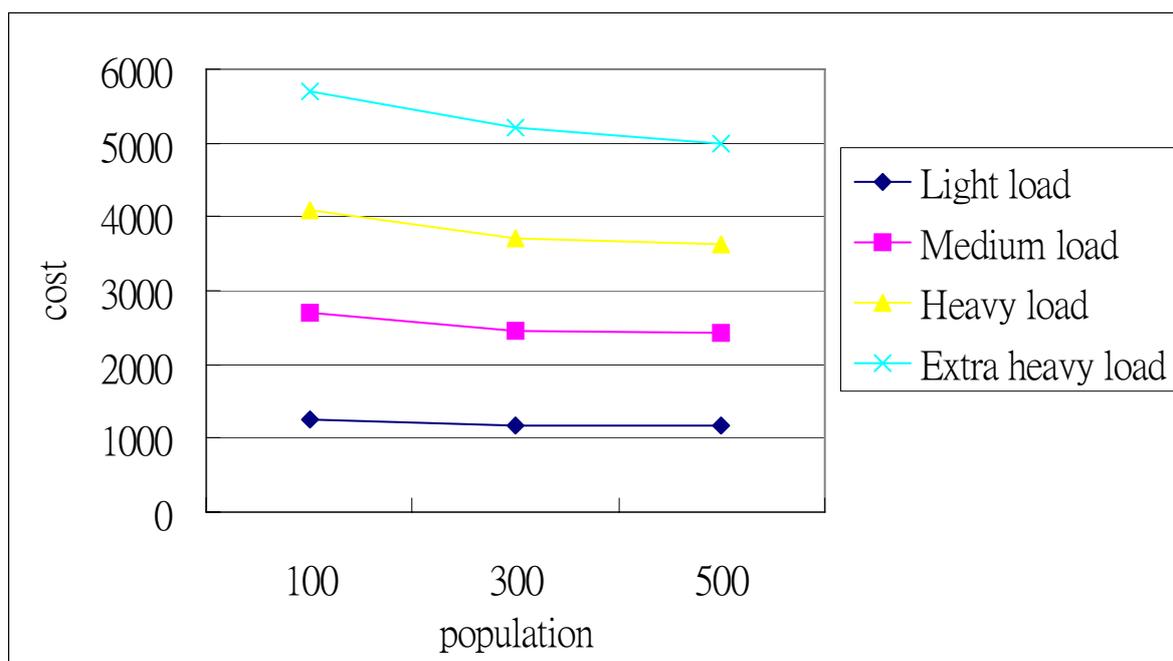


圖 7 族群大小與成本比較圖

本文以不同族群大小來比較成本差異，如圖 7 所示，為採用 SPA 時各種流量需求與族群大小不同時，所求得之整體網路成本的差異比較，其中並考慮 60 種不同的初始節點位置，此做法之目的在於比較二種演算法的一般表現。圖 7 中各點，為此 60 種網路所得之解的平均值。當族群大小由 100 增加至 300 時，發現成本略有改善，但當族群大小由 300 增加至 500 時，成本相差無幾，皆可獲得相當好的結果。為獲得較佳結果，本模擬選擇族群大小

為 500。

我們考慮 SPA 與 MCA 所得解與需花費的世代數之差異。如圖 8 所示，可發現 MCA 在各種流量需求下皆優於 SPA，主要原因在於 SPA 優先考慮最短 hop 數之路徑，其平均每條路徑之頻寬小於 MCA，需使用較多路徑以滿足各節點之流量需求，故 MCA 在各種流量需求下皆優於 SPA。

MCA 花費世代數亦少於 SPA，如圖 9 所示，

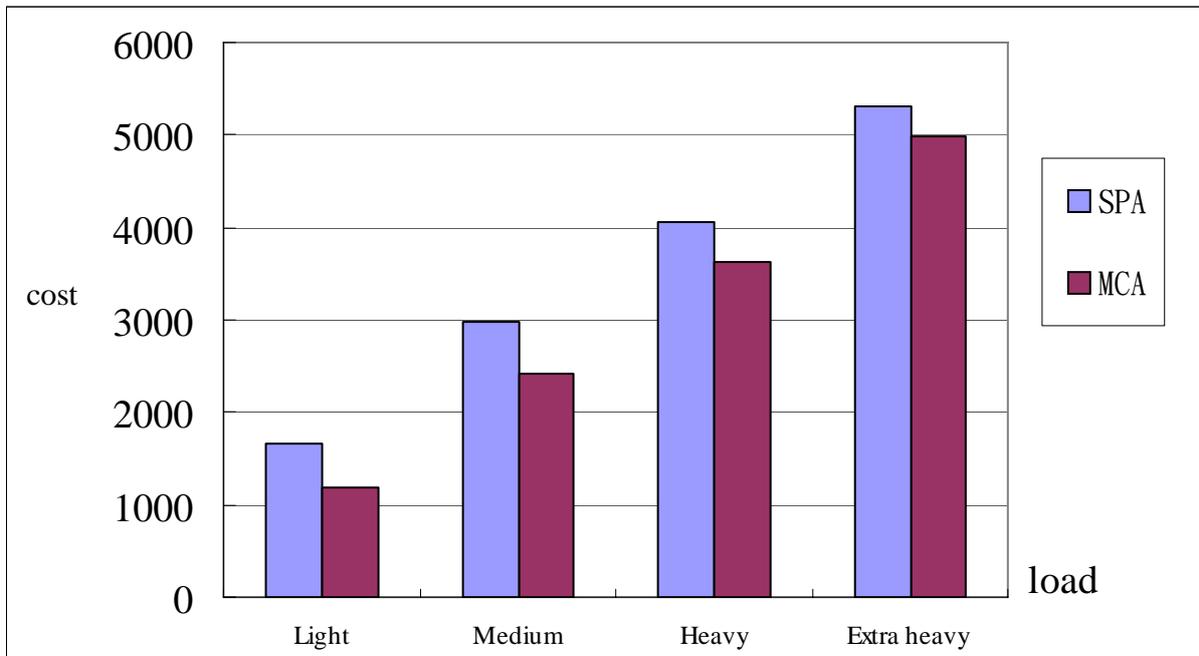


圖 8 SPA 與 MCA 之成本比較圖

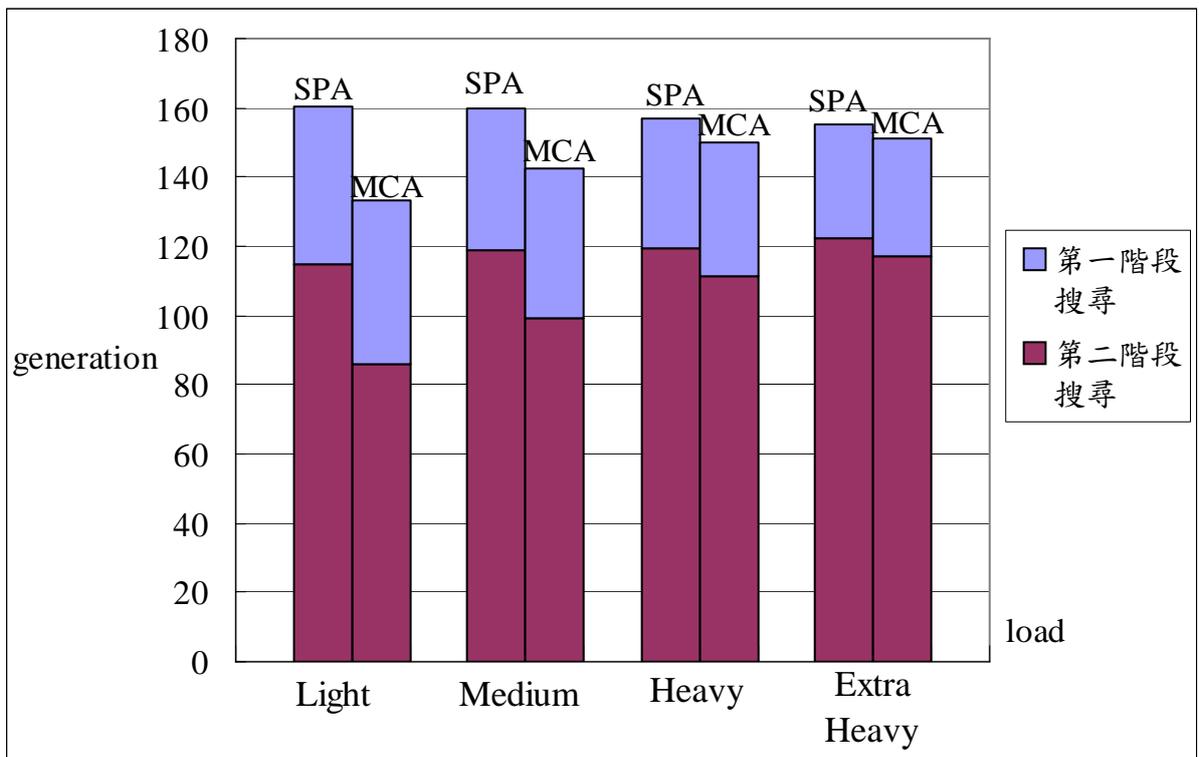


圖 9 SPA 與 MCA 之世代數比較圖

為兩種繞徑方法在不同流量需求下所花費之世代數比較。圖中每段數值之上半段與下半段分別表示為第一階段搜尋及第二階段搜尋所花費的世代數。由於 MCA 以最大頻寬路徑為優先考量，節點處理順序不同對最終結果的影響較小，故往往因收斂較快而節省世代數。SPA 中所有節點皆以最短路徑為優先，因此當染色體不同使得節點處理先後不同時，所得之路徑安排差異較大，因此具有收斂較慢的特性。

此外，如圖 9 所示，SPA 在較高的流量需求下，花費的總世代數略為降低，乃因較高的流量需求所使用之閘道器總數較多，在第一階段測試閘道器總數之最小值時，能較快找到解，而花費較少的世代數。而 MCA 世代數卻反隨著流量需求上升而增加，主要可歸因於流量需求越高時，拓樸中較大頻寬路徑所能滿足之節點數越少，故 MCA 之節點亦需使用多條路徑才能滿足其流量需求，故 MCA 對於節點處理順序不同對最終結果的影響較小之優勢將隨著流量需求的增加而逐漸被抵銷，在極高負載下，SPA 與 MCA 之世代數十分接近。

五、結論與討論

本研究提出 PGSA 解決 IEEE 802.16 之無線網狀網路拓樸設計問題，使用預先定義閘道器集合方法，能有效判斷無線網狀網路所需之最小閘道器數量，並降低各節點之天線使用數，進而得到較佳成本，可作為實際建置無線都會網路拓樸之依據。

PGSA 之流量安排使用繞徑方法實現，我們測試兩種繞徑演算法，並比較在各種流量需求下所求得之成本與花費世代數。PGSA 亦採用基因演算法搜尋最佳成本，並限制其染色體正位元數，符合此拓樸設計需求。

正逢此無線都會網路標準化制定日益成熟之際，本文所提出之無線網路拓樸設計方式，在實用層面充分展現其價值與必要性，其設計理念不僅可適用於本文所描述之背景環境，未來更可擴充運用於其他具有相關特性之廣泛議題

誌謝

本研究係在國科會專題研究計畫 NSC-94-2815-C-011-013-E 贊助下進行。

六、附錄

(一) 流量安排

圖 10 為流量安排流程圖，每次考慮拓樸中某一節點之流量需求，並採用繞徑演算法選擇路徑，若繞徑失敗，回報流量安排失敗；若繞徑成功，則將此路徑中各段連線剩餘頻寬最小者視為此路徑

之可用頻寬，且若目前考慮的節點之流量需求小於此路徑頻寬，表示不需再另尋路徑，則更新拓樸之各段剩餘頻寬及視需求扣除路徑上各節點可用之天線量，並回到開頭處理下一節點流量。若雖繞徑成功，但此路徑頻寬無法完全滿足該節點之流量需求，則需再為此節點另尋路徑，故將該節點之流量需求扣除目前路徑之頻寬後，再次使用繞徑演算法搜尋新的路徑，直至所有節點之流量需求皆處理完畢，才回報繞徑成功。

(二) SPA/MCA

本文所採用之繞徑演算法皆以 Dijkstra 最短路徑演算法為基準，加入考慮網路中單一節點有天線數之限制，可作為兩種不同優先考量的繞徑方法。SPA 以最少 hop 作為繞徑基準，MCA 則優先考量最大頻寬路徑。圖 11 為此兩種演算法之虛擬碼。其中若兩節點之距離符合最低通訊品質之條件，則此段鏈結稱為 potential link，如圖 11 中， i 節點與 j 節點間之鏈結成本 $L_{i,j}$ 在非 potential link 時以無限大表示，而在 potential link 時，SPA 設定 $L_{i,j} = 1$ ，而 MCA 則設定 $L_{i,j} = 1/B_{i,j}$ ，其中 $B_{i,j}$ 為 i 節點至 j 節點之最大頻寬。此外，此繞徑演算法與 Dijkstra's 最短路徑演算法尚有如下幾點差異：

- 考慮頻寬問題，頻寬已用盡之 link，其 L_{ij} 必為 Infinity，即繞徑過程中不考慮此路徑。
- 考慮單一節點最大天線使用數(maximum antenna degree)問題，如圖 12 所示，在考慮加入節點時於 F 集合時，額外考量可用天線數問題。
- 當 F 集合中有任一節點為閘道器則停止繞徑，並回報成功。
- 圖 12 中，當選擇下一節點之 pick() 發現無任何節點可選擇時，則回報失敗。
- 若起始節點為閘道器，則不需繞徑，直接回報成功。

七、參考文獻

- [1] IEEE 802.16-2001, "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems", Dec. 2001.
- [2] IEEE 802.16.2-2004, "IEEE Recommended Practice for Local and metropolitan area networks Coexistence of Fixed Broadband Wireless Access Systems", Oct. 2004.
- [3] The IEEE 802.16 Working Group on Broadband Wireless Access Standards, <http://www.ieee802.org/16/>
- [4] J.H. Holland, "Adaptation in Natural and Artificial Systems", Ann Arbor: Univ. of Michigan Press,

- 1975.
- [5] K. F. Man, K. S. Tang, and S. Kwong, "Genetic Algorithms: Concepts and Applications", IEEE Transaction on industrial electronics, Vol. 43, no. 5, pp. 519-534, Oct. 1996.
 - [6] M. Gen and R. Cheng, "Genetic algorithms and engineering design", New York : Wiley, 1997.
 - [7] R. L. Haupt and S. E. Haupt, "Practical Genetic Algorithms", N.J.: Wiley, 2004.
 - [8] T. S. Rappaport, "Wireless Communications: Principles and Practice", N.J.: Prentice Hall, 2002.
 - [9] 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 Communications Magazine, Vol. 43, pp. 129-136, Feb. 2005.
 - [10] G. Syswerda, "Uniform crossover in genetic algorithms", in Proc. 3rd Int. Conf Genetic Algorithms, pp. 2-9, Dec. 1989, George Mason Univ., United States.

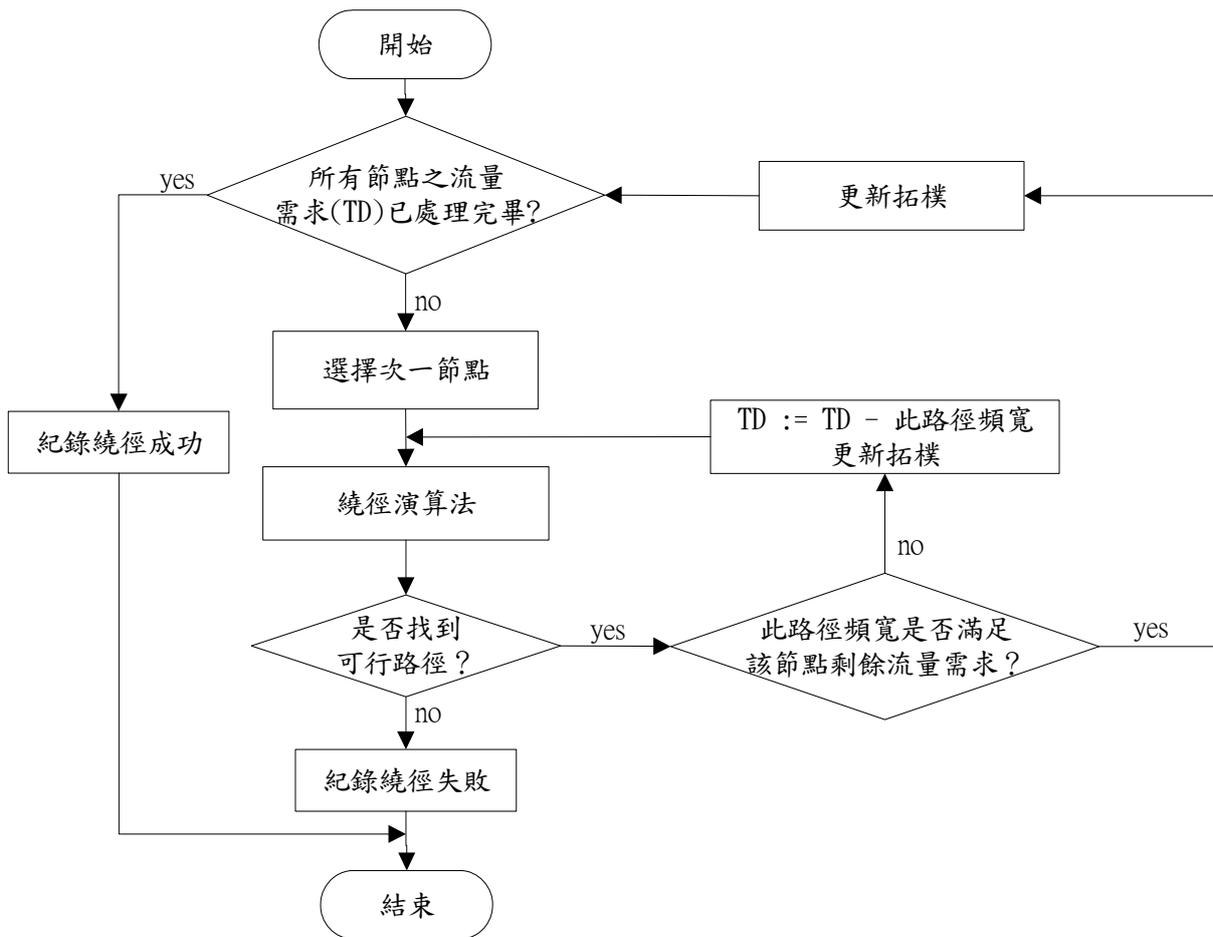


圖 10 流量安排流程圖

/* 給定拓撲 G ，其中節點集合為 V ，及任兩集合所形成之邊集合 E ，以 $(v, v2)$ 表示。起始節點以 S 表示。

D_i : 起點至 i 點之 hop counts/cost。

$L_{i,j}$: i 點至 j 點若為 potential link，以之間的 hop counts/cost 表示。否則設為 Infinity。

G_i : $\begin{cases} 1, & \text{if node } i \text{ is a gateway} \\ 0, & \text{otherwise} \end{cases}$ 。

π_i : i 節點之 predecessor。

current_node : 最新加入 F 集合之節點

F : 已完成搜尋之節點集合

U : 未完成搜尋之節點集合

G : 閘道器之節點集合

finish : 是否已找尋到閘道器，以 1 表示已搜尋到閘道器，0 表示尚未解決 */

Algorithm:Routing()

```

    if( $G_s = 1$ )
        finish = 1
    end if
    /* 初始化設定所有節點與起點間 hop counts/cost 為無窮大，predecessor 為 NULL */
    for  $i = 0$  to  $|V| - 1$ 
         $D_i = \text{Infinity}$ 
         $\pi_i = \text{NULL}$ 
    end for
    /* 起始節點到起始節點之 hop counts/cost 為 0 */
     $D_s = 0$ 
    current_node =  $s$ 
    /* while 迴圈每次選擇一個最小  $D_i$  之節點，將其由  $U$  集合移入
     $F$  集合，並更新各節點之 predecessor 與 hop counts/cost。 */
    while(finish != 1)
         $v = \text{pick}(U)$  /* 選擇下一節點，參考圖 12 */
        add  $v$  to  $F$  /* 將  $v$  節點納入考慮 */
        if( $v = \text{Failure}$ ) /* 若無解 */
            return Failure
        elseif( $G_v = 1$ ) /* 若已找尋到閘道器 */
            finish = 1 /* 不必再選擇下一節點 */
        for each neighbor of  $v$ ,  $v2$ 
            /* 更新 dist table 與 predecessor */
            if( $D_v + L_{v,v2} < D_{v2}$ )
                 $D_{v2} = D_v + L_{v,v2}$ 
                 $\pi_{v2} = v$ 
            end if
        end for
        current_node =  $v$  /* 更新目前節點 */
    end while
    return Success
end.
```

圖 11 繞徑演算法之虛擬碼

/* $A_{ji} : \begin{cases} 1, & \text{if link } i \text{ to } j \text{ is established connection.} \\ 0, & \text{otherwise} \end{cases}$.

a_i : i 節點所剩餘之天線個數。

U : 未完成搜尋之節點集合。 */

/* 在 antenna degree constraint 下選擇下一節點 */

pick(U)

Find the minimum hop counts/cost node i that is in U and satisfies either of following conditions:

(1) $A_{i,j} = 1$

(2) (($current_node = s$) or ($A_{\pi_{current_node}, current_node} = 1$))
and ($a_{current_node} > 0$ and $a_i > 0$)

(3) ($a_{current_node} > 1$ and $a_i > 0$)

If node i exists

return i

else

return Failure

end if

end.

圖 12 在節點之天線使用數限制下選擇次一節點之虛擬碼