

移動隨意網路下植基於基因演算法之能源考量 QoS 群播選徑

顏雲生 詹益恭 廖玉卿⁺ 趙涵捷⁺

國立東華大學電機工程研究所,⁺國立宜蘭大學電子工程系

{d9223008, m9223511}@em92.ndhu.edu.tw, {ycliaw, hcc}@niu.edu.tw

摘要

求解兩個或多個限制參數 (Multi-constraints) 的 QoS 群播路由問題, 已被實證是屬於 NP-complete (Non-deterministic Polynomial) 問題, 無法在多項式時間內求得最佳解。本論文提出在移動隨意網路下植基於基因演算法之能源考量 QoS 群播選徑 (Energy-Aware QoS Multicast Routing Protocol, EAQMRP) 機制來解決此問題, 同時考量傳播延遲和節點電量作為 QoS 群播選徑問題中目標函數之參數, 並且提出一種新式的 ST (Sequence and Topology) 編碼方式, 使基因演算法的操作可以快速收斂選出既省電又有效率和強韌性的群播繞送路徑。

關鍵詞: 移動隨意網路 (MANET), 群播選徑 (Multicast Routing), 服務品質 (QoS), 基因演算法 (Genetic Algorithm), ST 編碼。

1. 前言

移動隨意網路 (Mobile Ad Hoc Networks, MANETs) 是由一群可隨意移動的主機自我組織起來構成的無線網路。MANETs 並不需要預先建立基地台 (Base Station) 或接取點 (Access Points) 等基礎設施 (infrastructures), 而且 MANETs 的每一個可隨意移動的主機都具有路由功能, 它的通訊方式是藉由中間節點轉遞封包, 兩個移動式主機若在彼此的傳送範圍內, 則直接的通訊, 否則必須透過其它主機來轉遞封包, 以多點跳躍的方式來完成封包傳送。MANETs 有不受到移動性限制和佈署容易的特性, 使其日漸受到歡迎。

一般而言, 設計多重限制參數的低成本 QoS 群播路由通訊協定時, 常常沒有將節點處理資料所需消耗的電力列入考量, 使得整個網路的運作極可能因為路徑上節點電量消耗不平均, 造成某些節點的電量被過度的消耗, 一旦這些節點的電力被消耗殆盡, 傳送封包的鏈路便會出現中斷的情況, 當被選為傳遞封包的路徑中斷情況增多時, 群播網路的生命週期 (lifetime) 將無法持續到封包傳遞完成。

在群播繞送路徑通訊環境中, 許多曾經被提出的相關文獻 [1], 常常只考慮一般的 QoS 參數 (Metrics), 例如頻寬、封包漏失率和傳播延遲等等。

我們的研究先以延遲限制建立群播樹, 在最小延遲的 QoS 下, 嘗試以節點的剩餘電量當作 QoS 參數, 將群播系統的總消耗電力, 由電量較多的節點平均分攤, 使中間節點在轉遞封包的過程中不會

因電量不足而中斷, 可以維持最長的群播系統週期。基因演算法雖然可以在多項式時間內求得近似最佳解, 但是編碼方式的良窳與否, 卻會影響到基因操作、收斂速度, 所以我們提出一種由 ST 編碼 [10] 改良而來的新式 ST 編碼, 可以控制節點的分支度數, 使基因操作更加簡易、有效率。

2. 背景及相關研究

2.1 背景

2.1.1 群播路由 (Multicast Routing) 簡介

群播路由演算法, 是將信號從源節點 (source node) 同時傳送到多個目的節點 (destination node), 良好的群播路由演算法能夠控制資料傳送的延遲時間與頻寬使用率, 甚至能源之利用率。

由目前已經提出的多種 MANET 群播路由協定, 按照支持群組成員關係的架構, 可分為四大類 [11]: 基於樹的 (tree-based) 群播路由協定、基於網格的 (mesh-based) 群播路由協定、Stateless 群播路由協定和 Hybrid approaches 群播路由協定。

由於我們只強調群播樹建構出來的形狀是屬於那一類型, 所以我們的分類方式與其他文獻有點不同, 也就是說, 當網路建好群播路由時, 判斷群播路由是屬於那一類型, 而非從“如何建立維、護群播路徑”的觀點去分類 [11]。例如: (AMRoute) [12] 若是以另一種分類方式 (以 Multicast Protocol State Maintenance 的分類方式) 來區分會屬於所謂的“Overlay Multicast Routing”, 在這裡我們將它歸屬於基於樹的群播路由協定。

A. 基於樹的群播路由協定:

基於樹的群播路由協定沿用傳統群播路由的演算法, 在所有群組成員之間建立一個共享群播樹。

1.) 利用遞增序號的 Ad Hoc 群播路由協定 (AMRIS) [13]: AMRIS 是一種 on-demand、source-initiated、基於共享樹的群播路由協定。

2.) Ad Hoc 群播路由協定 (AMRoute) [12]: 因為基於樹的群播路由協定可以提供較好的封包轉遞效率, 但是所建構的群播路由強韌性較差; 基於網格的群播路由協定可提供較佳的強韌性路由, 但是轉遞封包的費用較高。所以後來的學者提出了另一種混合型的群播路由協定 AMRoute, 可以同時獲得基於樹與基於網格的好處。這個協定的兩個主要階段是建立網格鏈路與建立共享群播樹。AMRoute 的特點是, 只要樹成員之間的網格鏈路仍然存在, 當網路拓撲架構改變時, 群播樹不需重新調整, 因為利用虛擬網格鏈路建立一個雙向的共享群播樹, 只有群組成員才會在所屬的群播樹中傳遞封包, 所以也將之歸類為基於樹的群播路由協定。

B. 基於網格的群播協定：

基於網格的群播協定是建立一個共享的節點網格，讓群組成員在這個網格中傳輸。這類協定優於基於樹協定的一個特點就是提供了路徑的冗餘性，在移動性情況下增強協定的可靠性，缺點是可能產生迴路。由於要維護網格架構，基於網格的協定需要傳輸的路由訊息要比基於樹的協定要多。

1.) On-Demand 群播路由協定 (ODMRP) [14]：ODMRP 是一種 On-Demand 協定，不需要永久性維護路由訊息。它採用一種軟狀態方法維護群組訊息，群組成員的關係和群播路由由源節點 On-Demand 建立、更新。

2.) 核心輔助網格協定 (CAMP) [3]：將基於樹的群播概念轉換到具有比樹連通性更強的群播網格。節點使用 receiver-initiated 的方式加入群組，以共享網格架構支援動態的 Ad Hoc 網路。

2.1.2 服務品質保證 (Quality of Service)

QoS 量測項目 (Metrics) 可分為可用頻寬、端點到端點延遲、封包遺失率、抖動、和花費，不同的量測項目具有不同的性質，這些量測項目依據性質來分類為可加性、可乘性和最小性等三種，量測項目分別描述如下：

1). 可加性：

$$Add_Metric = \sum_{Tree} Add_Metric_i,$$

例如延遲、抖動、花費和 Hop 數。

2). 可乘性：

$$Mul_Metric = \prod_{Tree} Mul_Metric_i,$$

例如封包遺失率。

3). 最小性：

$$Min_Metric = Min\{Metric_i\},$$

例如鏈路可用頻寬，也就是路徑上瓶頸鏈路之頻寬。

2.1.3 基因演算法 (Genetic Algorithm)

基因演算法是一種啟發性求解法則，藉由生物基因工程的概念，解決組合最佳化問題。

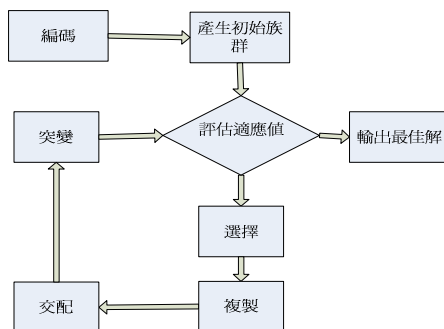


圖1：基因演算法的流程圖

基因演算法首先必須將相對應的問題加以編碼

對應到一組字串 (Strings) 之集合，這些字串就是染色體 (Chromosomes)，每一個染色體代表一個可能的解 (Feasible Solution)，會設定適應函數 (Fitness Function)，染色體的適應函數值愈大者，愈有機會得到生存與複製下一代的機會。基因演算法操作流程，如圖1所示。

基因演算法的各項函數定義如下：

A. 適應函數 (Fitness Function)：

一般基因演算法的文獻大多以下面的公式表示適應函數。 $Fit(T) = \frac{\rho \cdot \Omega(T)}{Cost(T)}$ 此處， ρ 是正實數系數，

$$\rho: \rho \rightarrow R^+.$$

$$\Omega(T) = \sum_{l \in P(s, v)} \Phi(D(l) - D_{max}), \Omega(T)$$

是群播樹的處罰函數， $\Phi(\epsilon) = \begin{cases} 1, \epsilon < 0 \\ \lambda, \epsilon > 0 \end{cases}$ ， $\Phi(\epsilon)$ 是每

個鏈路 l 的處罰函數， $p(s, v)$ 為由源節點 s 到目的節點 v 的路徑， λ 是處罰程度， $0.01 < \lambda < 0.5$ 。

B. 修補函數 (Repair Function) [8]：處理非可行解有兩種方式，一種是修補方式，另一種是設定處罰值。我們選擇使用修補函數 f_{repair} ，限制節點分支度數，以較多電量的節點替換電量較少的節點。

C. 處罰函數 (Penalty Function)：主要目的是避免最佳解落入非可行解的區域 (infeasible solution region)。處罰函數與適應函數 (目標函數) 的組合有兩種型式：

1). 相加型： $Eval(C) = Fit(C) + \Omega(C)$ ，求最大值問題時，若染色體 C 落入可行解區域， $\Omega(C) = 0$ ；若染色體 C 落入非可行解區域， $\Omega(C) < 0$ 。 $Fit(C)$ 是適應函數， $\Omega(C)$ 是處罰函數。

2). 相乘型： $Eval(C) = Fit(C) * \Omega(C)$ ，求最大值問題時，若染色體 C 落入可行解區域， $\Omega(C) = 1$ ；若染色體 C 落入非可行解區域， $0 \leq \Omega(C) < 1$ 。

2.2 相關研究

為了有效的基因參數搜尋空間，首先要確認每個參數的搜尋範圍，會將每個參數以固定長度的字串加以編碼。一般較常被使用的編碼方式是二進位編碼。搜尋精度取決於編碼的長度，愈長的編碼位元，精確度愈高，相對地，要有較大記憶體儲存空間與犧牲運算速度的代價。

樹的編碼可以分成三大類 [7]：(1) 邊的編碼 (Edge Encoding)，一般而言，邊的編碼方式不適合用在擴張樹；(2) 頂點編碼 (Vertex Encoding)，頂點編碼方式較多人使用，典型的編碼方式有 Prüfer number 編碼 [4]、ST 編碼 [10]；(3) 邊與頂點的編碼 (Edge and Vertex Encoding)。

2.2.1 Prüfer number

Prüfer number 節點編碼方法 [4] 是 Cayley 教授所提出的，此方法的編碼及解碼過程分述如下。

A. 編碼(Encoding) :

步驟 1: 在 n 個節點的擴張樹 T 中, 分別對每一個節點標示識別碼; 步驟 2: 選取其中識別碼最小的葉片節點 u ; 步驟 3: 找出與節點 u 唯一連接的節點 v , 將節點 v 當做 Prüfer number 編碼最左邊的第一位元; 步驟 4: 刪去節點 u 及節點 u 與 v 之間的鏈路, 此時, 擴張樹 T 剩下 $n-1$ 節點數; 步驟 5: 重複操作步驟 2、步驟 3、步驟 4, 由左至右依序得到 Prüfer number 各位元, 直到剩下一個邊, 可獲得 Prüfer number 編碼字串 P_n ; 步驟 6: 將沒有出現在 P_n 中的其餘節點識別碼, 由小至大排序得到葉片節點字串 Pl 。 u : 代表在整個樹狀拓樸中, 位於最末端, 亦即此節點之後沒有再連接另外節點, 且為識別碼最小之節點。 v : 代表與節點 u 相連之非葉片節點。鏈路: 代表兩個節點相連接之部份。我們以圖 2 的擴張樹 MTree 1 為例, 可以得到 Prüfer number $P_n = [2, 1, 7, 1, 8, 2, 7]$, $Pl = [3, 4, 5, 6, 9]$ 。

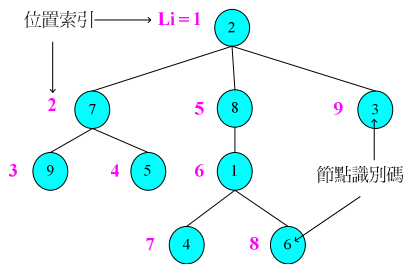


圖 2: 群播樹 MTree 1

B. 解碼(Decoding) :

步驟 1: 已知一 Prüfer number 編碼字串 P_n 及其葉片節點字串 Pl ; 步驟 2: 令 x 為葉片字串 Pl 中最小編號的位元, 而 y 為 P_n 中最左邊的位元, 在節點 x 與 y 之間連接一個邊, 接著從 Pl 中移除 x 與 P_n 中移除 y ; 步驟 3: 檢查 P_n 中是否還有 y 存在, 如果 y 沒有在 P_n 中, 則將 y 加入 Pl 中, 並對 Pl 做由小至大的排序; 重複上述步驟直到字串 P_n 中無任何數字位元為止; 步驟 4: 當字串 P_n 中無任何數字位元時, 字串 Pl 中應剩下節點 r 及 s , 將此兩節點連接起來而形成一個樹狀拓樸。 x : 代表已知 Pl 中編號最小的位元。 y : 代表已知 P_n 中最左邊的位元。 r, s : 代表在解碼過程最後階段時, 當 P_n 中無任何數字位元時, 在 Pl 中所剩下之兩個位元。

Prüfer number 編碼之優點為當總節點數為 k 時, 編碼長度僅需 $k-2$ 個位元長度, 比其他編碼方法節省記憶體儲存空間。放射狀樹的拓樸圖, 經過基因操作交配與突變後, 仍然能夠保持樹狀的合理拓樸。

Prüfer number 編碼之缺點為每一次基因操作交

配與突變後, 會繁衍出與母代相似程度只有一點點的子代。也就是說, 當 Prüfer number 稍為微改變時, 會造成網路拓樸中許多節點之間的鏈路改變, 破壞整棵樹的結構, 無法直接求得區域性可行解。

3. 移動隨意網路下植基於基因演算法之能源考量 QoS 群播選徑(EAQMRP)

3.1 群播路由網路模型:

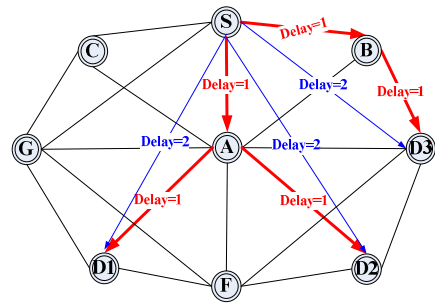


圖 3: 網路拓樸圖

由圖形理論與相關文獻[9]得知, 一個網路可以權重圖 $G(T) = (V(T), E(T))$ 表示, $V(T)$ 為群播樹中節點之集合, $E(T)$ 為群播樹中節點到節點之鏈路集合, $k = \|V(T)\|$ 為群播樹中所有節點之數量, $\|E(T)\|$ 為群播樹中所有節點到節點之鏈路數量, 令 $s \in V(T)$ 為群播樹的源節點, $M \subseteq \{V(T) - \{s\}\}$ 為群播樹的目的節點, 我們先定義 QoS 量測項目: 對節點到節點鏈路 $l \in E(T)$, 延遲函數為 $Delay(l): E(T) \rightarrow R^+$, 此處, R^+ 為正的鏈路權重值; 對任意節點 $v \in V(T)$, 節點電量成本函數為 $Cost(v): V(T) \rightarrow R^+$ 。我們以 $T(s, M)$ 代表群播樹, 與群播樹相關的延遲函數、成本函數如下所述:

$$Delay(T(s, M)) = \sum_{l \in T(s, M)} Delay(l),$$

$$Cost(T(s, M)) = \sum_{v \in T(s, M)} Cost(v).$$

在圖 3 的網路拓樸圖中, 假設每個節點有一定的發射功率, 任一節點皆為可傳送範圍。當源節點 S 要傳送資料給群組組員 $D1$ 、 $D2$ 、 $D3$ 時, 如果直接由源節點 S 傳送到目的節點, 源節點 S 的分支度數等於 3, 路由層、MAC 層則需消耗三個單位的封包處理電力。直接由源節點 S 傳送到目的節

點之群播路由為: $\left\{ \begin{array}{l} \rightarrow D1 \\ S \rightarrow D2 \\ \rightarrow D3 \end{array} \right\}$ 。經過 EAQMRP 演算

法, 可以兼顧 QoS, 同時節省源節點 S 三分之一的電力消耗。改良後的群播路由如下:

$$\{S \rightarrow B \rightarrow D3\} \text{ 與 } \left\{ \begin{array}{l} S \rightarrow A \rightarrow \rightarrow D1 \\ \rightarrow D2 \end{array} \right\}.$$

3.2 新式 ST 編碼與解碼

我們局部修改STMP編碼[10]，提出適合基因交配與突變，操作容易的新式ST編碼。

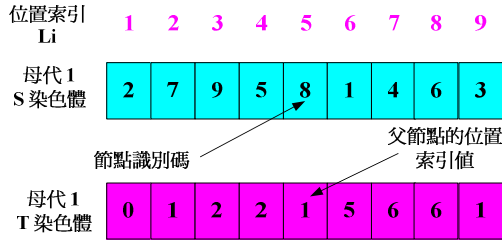


圖 4：轉換群播樹 MTree 1 之新 ST 編碼

由圖 2 之群播樹 MTree 1 中，每個節點給定唯一的識別碼，以 Pre-order 的方式由上而下、由左至右走訪樹，令起始位置索引 Li 為整數值 1，然後依序遞增索引值，同時將節點所在的識別碼填入代表基因序列的 S 陣列中，一直持續到最後節點。代表基因拓撲的 T 陣列則依序紀錄父節點的位置索引值。經過編碼後得到的 S 陣列、T 陣列，就是相對應的 S 染色體與 T 染色體。例如，圖 2 之群播樹 MTree 1 經過編碼轉換之後的 ST 編碼，如圖 4。同理，圖 5 之群播樹 MTree 2 經過編碼之後可以得到另外一對 S 染色體與 T 染色體，如圖 6。由以上得知，一個母代是由一對 S 染色體、T 染色體組成。

群播樹轉換為 ST 碼的虛擬碼如下：

```

/* 群播樹轉換為ST碼的虛擬碼 */
Begin
  Given: 長度為k的S陣列, T陣列, 整數值的
         位置索引Li.
  令S[Li <- 1] 為根節點;
  令T[Li] <- 0;
  For Li=2 to k {
    S[Li] <- 位置索引Li的節點識別碼;
    T[Li] <- 位置索引Li的父節點的位置索引值;
  }
End

```

ST 碼轉換為群播樹之虛擬碼如下：

```

/* ST碼轉換為群播樹的虛擬碼 */
Begin
  Given: 長度為k的S陣列, T陣列, 整數值的
         位置索引Li.
  令S[Li <- 1] 為根節點;
  令T[Li] <- 0;
  For Li=2 to k {
    連線S[Li] 到 S[T[Li]];
  }
End

```

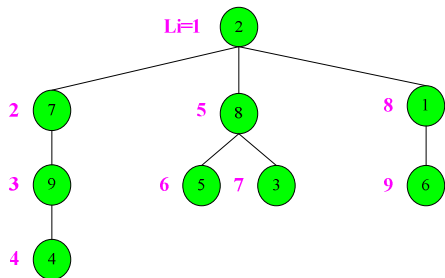


圖 5：群播樹 MTree 2

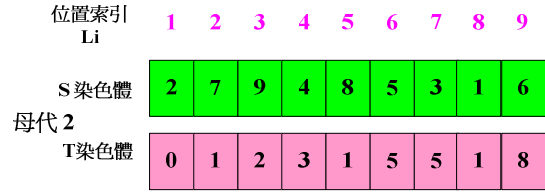


圖 6：轉換群播樹 MTree 2 之新 ST 編碼

3.3 能源考量之基因演算法：

EAQMRP 演算法藉由計算群播樹中所有節點的總剩餘電量當作成本函數，透過操作基因突變修補函數將群播系統的生命週期延長到最長。能源考量之基因演算法虛擬碼如下：

```

/* 能源考量之基因演算法虛擬碼 */
The Energy_Aware GA() procedure:
Begin
  Initialization ();
  Do {
    /* choose parents from population */
    Selection ();
    /* generate offspring by combining parents */
    Crossover ();
    /* optimized offspring */
    Mutation_Replace ();
    /* replace worst parents by better offspring */
    Evaluation ();
  } while (criterion is true)
End

```

由 2.1.3 節得知，我們採用相乘型的適應函數：

$$Fit(T(s, M)) = (\alpha \cdot f_{delay} \cdot \Omega) \times (\beta \cdot f_{repair})$$

， α 、 β 為正實數的權重係數， Ω 為處罰函數。傳播延遲的成本函數：

$$f_{delay} = c \cdot \frac{1}{Delay(T(s, v))}, \quad c \text{ 為正實數的係數。}$$

突變修補函數： $f_{repair} = Tot_bty(T(s, M))$ 。

為了適合我們假設的條件，我們稍為修改[2]的成本函數公式，定義群播樹中所有節點的總剩餘電量如下： $Tot_bty(T(s, M)) = \sum_{v \in T(s, M)} (Bt_v(t))$

此處， $Bt_v(t) = \zeta_v \cdot \left(\frac{Res_Bt_v(t)}{Full_Bt_v} \right)^\pi$ ； $Bt_v(t)$ ：節點 v 在時間 t 的電量百分比； $Res_Bt_v(t)$ ：節點 v 在時間 t 的剩餘電量值； $Full_Bt_v$ ：節點 v 的滿電量值； ζ_v ：節點 v 的發射功率，因為我們先前已假設所有節點發射功率為一定值，所以令 $\zeta_v = 1$ ； π ：正實數權重值，與節點分支度數成反比。

很顯然地，群播樹的最大生命週期是由剩餘電量百分比最小的一個節點 v 控制，因為當節點 v 在時間 t 的電量百分比最小時，群播路徑即將會在這

很顯然地，群播樹的最大生命週期是由剩餘電量百分比最小的一個節點 v 控制，因為當節點 v 在時間 t 的電量百分比最小時，群播路徑即將會在這

個節點發生中斷的情況。所以，我們定義群播樹的最大生命週期時間為：

$$Max_Life(T(s, M)) = \tau \cdot \underset{v \in T(s, M)}{Min} \{Bt_v(t)\}$$

， τ 是時間修正係數。

我們為了延長群播樹的最大生命週期，採取如何平衡整個群播樹的電量消耗為策略。在基因操作過程中，我們設定 $Bt_v(t)$ 為基因操作的一個突變因子。因此，低電量而且出現在T染色體中的核心節點，將會被高電量而且非出現在T染色體中的葉節點所取代。T染色體之基因交配方式參考[6]，針對被選取的兩個母代染色體採隨機兩點式基因交配 (two-crossover)，交配的過程中，由母代1 中選取兩個交配點間的基因遺傳至子代，子代其餘的基因則遺傳自母代2，交配過程如圖7，T染色體交配虛擬碼如下：

```

/* T 染色體交配虛擬碼 */
The Crossover_T() procedure:
Begin
  Void crossover_T(Cr){
    seed = mod(rand()/k);
    If (seed <= Cr){
      For Li = 1 to k
        exchange Li_th gene of Parent 1' T
        with (k - Li+1)_th gene of Parent 2' T
      }
    }
  End

```

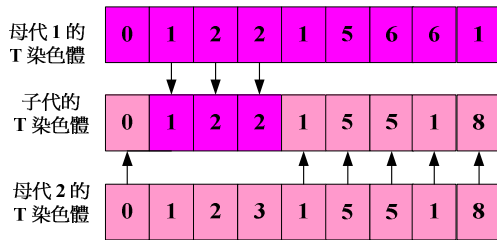


圖 7：T 染色體之交配過程

由於S染色體的基因具有唯一且不可重複的特性，我們局部修改[5]之交配策略，令母代1 的S 染色體具有較高的基因遺傳優先權，先由母代1 中選取兩個交配點間的基因遺傳至子代，子代的其餘基因則由母代2 的基因依序遺傳至子代，交配過程如圖 8，S 染色體交配虛擬碼如下所示。

```

/* S 染色體交配虛擬碼 */
The Crossover_S() procedure:
Begin
  Void crossover_S(Cr){
    r = rand();
    If (r <= Cr){
      For Li = 1 to k
        crossover Li_th chromosome with (k - Li+1)_th chromosome
      }
    }
  End

```

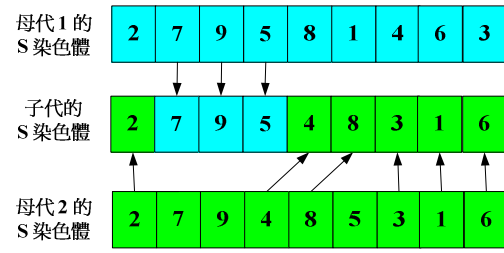


圖 8：S 染色體之交配過程

令 $deg(v)$ 為節點之分支度數。由於分支度數 $deg(v)$ 較多之節點會快速耗掉較多之電量，我們提出的基因演算法，以限制 $deg(v)$ 優先，同時考量如何平均地分配能源消耗問題。我們使用突變修補函數 $Mutation_Replace()$ ，以電量當作一個突變因子去控制基因突變，基因突變操作概念參考[7]。

由如圖7的子代T染色體得知， $|T[Li]=1|=3$ 、

$|T[Li]=2|=2$ 、 $|T[Li]=8|=1$ 和 $|T[Li]=5|=2$ ，T染色體中相同基因的數目代表

父節點下游 (downstream) 的 $deg(v)$ ，我們令 $deg(v) \geq 1$ 的節點為核心節點 (Core Nodes)，其位置索引為 $\{CN_Li\}$ ； $deg(v) = 0$ 的節點稱為葉節點 (Leaf Nodes)，其位置索引為 $\{LN_Li\}$ 。舉例說明如下：觀察交配後的子代T染色體、子代S染色體如圖7、圖8，T染色體的基因 $\{1, 2, 5, 8\}$ 為父節點位置索引值 $\{CN_Li\} = \{1, 2, 5, 8\}$ ，我們稱 $\{CN_Li\}$ 相對應的節點 $\{S[CN_Li]\} = \{2, 7, 4, 1\}$ 為核心節點。

不在T染色體中的基因為葉節點位置索引值 $\{LN_Li\} = \{3, 4, 6, 7, 9\}$ ，我們稱 $\{LN_Li\}$ 相對應的節點 $\{S[LN_Li]\} = \{9, 5, 8, 3, 6\}$ 為葉節點。

操作S染色體突變可以使電量低之核心節點被電量高的葉節點替換。我們由葉節點 $\{S[LN_Li]\} = \{9, 5, 8, 3, 6\}$ 中，挑選一個電量最高之基因 $\{8\}$ ，替換核心節點 $\{S[CN_Li]\} = \{2, 7, 4, 1\}$ 中最低電量之基因 $\{7\}$ ，操作過如圖9。

```

/* S 染色體突變虛擬碼 */
The Mutation_Replace() procedure:
Given: a set of core node: Cn
      a set of leaf node: Cn'
      and a fixed battery_threshold%: BT%
Begin /* exchange core node with leaf node */
  For each core node {
    If (battery_max% < BT%){
      exchange Cn with Cn' ;
    }
  }
End

```

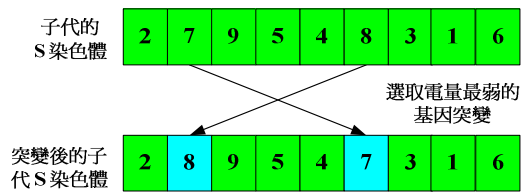


圖 9：S 染色體之突變過程

4. 結論

一般研究群播路徑的文獻，針對能源使用效率的探討，通常只強調如何選取電力消耗最少的群播路徑，這種機制，會造成一種不良的現象，可能因為某些節點的條件較好而經常被選取，在過度使用後將使其電量快速消耗殆盡，進而縮短了群播路徑的生命週期，使群播服務系統提早結束。本論文，我們採取如何平衡整個群播樹的電量消耗為策略，可以延長群播樹的最大生命週期。

EAQMRP演算法在基因演算法之精神下，改良交配機制和導入以能源考量的突變修補函數，可以控制節點的分支度數，也可以電量較多的節點替換電量較少的節點，將群播系統的總消耗電力，由其他電量較多的節點平均分攤。同時，操作簡易的新式ST編碼，可以直接在兩個陣列中計算群播樹的成本，不必另外重新建構新的群播樹。

突變修補函數，僅有交換葉節點與核心節點，其操作的時間複雜度為 $O(k - Ln)$ ， k 為節點的數目， Ln 為葉節點的數目。由上述的操作步驟及時間複雜度分析顯示，我們提出的基因演算法，每一代繁殖只做基因交配、突變操作，不但可以達到節省電能與快速收斂的目標，而且可以讓群播系統維持最長久的生命週期，使基因演算法適用於移動性隨意網路。

參考文獻

[1] Li Layuan, Li Chunlin, "A QoS Multicast Routing Protocol for Mobile Ad-Hoc Networks", IEEE, Information Technology: Coding and Computing, 2005. International Conference on Volume 2, 2005 Page(s):609 – 614.

[2] Morteza Maleki, Massoud Pedram, "Lifetime-aware multicast routing in wireless ad hoc networks", WCNC 2004 - IEEE Wireless Communications and Networking Conference, vol. 5, no. 1, March 2004 pp. 1305-1311.

[3] J.J. Garcia-Luna-Aceves and E.L. Madruga, "The Core Assisted Mesh Protocol (CAMP)", IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, Vol. 17, No. 8, pp. 1380-1394, August 1999.

[4] A. T. Haghghat, Karim Faez, M. Dehghan, A. Mowlaei and Y. Ghahremani, "a genetic algorithm for steiner tree optimization with multiple using Prüfer number", Proceedings of the First EurAsian Conference on Information and Communication Technology table of contents

Pages: 272 - 280, 2002.

[5] Gengui Zhou; Yihua Zhu; Xiaobin Weng; Feng Ye; "The generalized approaches of genetic algorithms on constrained minimum spanning tree problems" Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on Volume 3, 15-19 June 2004 Page(s):2141 - 2145 Vol.3

[6] M. Gen, K. Ida and J. KIM, "A spanning tree-based genetic algorithms for bicriteria topological network design," In Proceedings of IEEE International conference on Evolutionary Computation, 1998, pp. 15-20.

[7] G. Zhou, M. Gen and T. Wu, "A new approach to the degree-constrained minimum spanning tree problem using genetic algorithm," in Proceedings of IEEE International Conference on System, Man, and Cybernetics, vol. 4, pp.2683-2688, 1996.

[8] Devaraj, D., Veerakumar, S., "Data communication network design using genetic algorithm", Intelligent Systems., Proceedings. 2004 2nd International IEEE Conference Vol. 2, 22-24, Vol.2, Page(s):612 – 616, June 2004.

[9] Li Layuan, Li Chunlin, "Genetic Algorithm-Based QoS Multicast Routing for Uncertainty in Network Parameters". In Web Technologies and Applications: 5th Asia-Pacific Web Conference, APWeb 2003, Xian, China, Pages 430-441, April 23-25, 2003.

[10] Tzu-Chiang Chiang, Yueh-Min Huang, "A sequence and topology encoding for multicast protocol (STMP) in wireless ad hoc networks", In proceedings of the fourth international conference on parallel and distributed computing, applications and technologies, Chengdu China, Pages 351-355, 2003.

[11] C. Cordeiro, H. Gossain, and D. Agrawal, "Multicast over Wireless Mobile Ad Hoc Networks: Present and Future Directions," IEEE Network Magazine, vol. 17, no. 1, pp. 52-59, January/February 2003.

[12] E. Bommaiah, M. Lui, A. McAuley, R. Talpade, "AMRoute: Adhoc Multicast Routing Protocol," Internet draft, IETF, Aug. 1998.

[13] C. W. Wu and Y. C. Tay, "AMRIS: A Multicast Protocol for Ad hoc Wireless Networks," Proc. IEEE MILCOM '99, Atlantic City, NJ, Nov. 1999.

[14] S.J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks," Mobile Networks and Applications, ACM/Kluwer, vol. 7, no. 6, 2002, pp. 441-453.