

在分波多工網路上符合光功率預算的群播演算法之設計

蔡財量

永達技術學院 電子工程系
國立高雄第一科技大學 工程科技研究所
u9215902@ccms.nkfust.edu.tw

吳介騫

國立高雄第一科技大學
電腦與通訊工程系
jcwu@ccms.nkfust.edu.tw

摘要

群播是目前網際網路上一項應用廣泛的服務型態，當考慮要在分波多工網路施行群播傳輸時，利用光分裂技術即可達成群播傳輸之目的，且可節省大量的網路頻寬，但光分裂將會造成光功率的嚴重衰減。目前雖有文獻提出減少光功率衰減的方法，但都只是針對提高光功率的平均值，並不注意變異數的大小，這種做法並不能符合 WDM 網路實務上之所需。

因此，本文提出一個快速計算動態群播樹最小功率的方法及藉以修正既有群播演算法的策略，修正後的演算法可以確保傳送至各目的節點的光功率不小於最低的光功率臨界值，以符合 WDM 網路實務上之所需。

關鍵字：分波多工，群播。

Abstract

The multicasting is one of the important applications in recently Internet networks. Using the technology of light splitting can achieve the purpose of multicasting and save a large amount of the bandwidth of networks. However, light splitting will result in series power loss. Although several literatures of reducing the optical power loss have been proposed, which are always aimed to increase the average value of the optical power but not care about the variance. Such methods are not satisfying the requirement in WDM networks in practice.

In this paper, we propose an effective algorithm which not only can resolve the multicasting routing problem in Sparse WDM networks but also can ensure the minimum of optical power, which is transmitted to destination nodes, within the power budget and is satisfied the requirement in WDM networks in practice.

Keywords: WDM, Multicast

1. 簡介

在目前網際網路上，需要群播(Multicasting)型態服務之應用越來越多，所謂群播是將單一來源訊息

同時分送到數個目的地，可使用在視訊會議(video conferencing)、遠距教學(distance learning)及網際網路廣播(webcasting)等應用之中。若是以 IP 封包來傳送廣播訊息，必需在群播樹的每個分歧節點執行封包複製，當目的節點很多且分散時，大量的複製封包將降低網路效能；若應用光路徑(lightpath)技術於群播服務時，來源節點必須分別發送不同波長的雷射光到每一個目的節點，當目的節點很多時，網路頻寬將被嚴重耗盡而變得難以接受[1][4]。

在文獻[3]中，提出光樹(light-tree)的觀念可達到節省 WDM 網路頻寬的目的，所謂光樹是在 WDM 網路中的每一個路由節點，都支援以光功率分裂(light split)的方式來達到群播的目的，於是傳送到許多目的節點的群播資料，可由於共享了靠近群播樹來源端的鏈路而節省了網路頻寬。一般而言，在 WDM 層支援群播比在 IP 層支援群播更有效率，同時可提供光網路上訊息編碼的一致性(consistent of coding format)與傳送速率的通透性(bit-rate transparency)。

在文獻[5]中，考慮到從目前的網路過渡到全光網路(All-Optical Network)，可能只有部分節點才具有光分裂能力，而部分節點則沒有光分裂能力，此稱為稀疏分波多工網路(Sparse WDM Network)，顯然地此項限制在實務上是非常重要的。稀疏網路意謂著在網路拓撲上，各節點的特性並非一致的，這使得群播路由演算法(routing algorithm)的設計倍加困難，單一光樹可能無法將群播訊息繞送到所有目的節點，因此可能必須建立若干棵有同樣來源節點的光樹，以涵蓋所有目的節點。這些根源於同一來源節點的光樹，便合稱為光森林(light-forest)。文獻[5]中並提出了四種在稀疏分波多工網路上繞送群播通量的演算法，分別是 Re-route-to-Source、Re-route-to-Any、Member-First 及 Member-Only 演算法，並且比較這些演算法的效能(performance)，其中以 Member-Only 演算法的效能最佳。

雖然光樹或光森林可以以光分裂的方式，有效率地傳送群播通量，但也會導致雷射光功率嚴重衰減的後果，分裂次數越多光功率衰減越是嚴重，雖然目前有諸如 EDFA(Erbium-Doped Fiber Amplifier)的光放大器以補償光功率的損失，但此等元件的價位仍

高，可放大的頻寬也不足。文獻[2]中提出了集中分光(Centralized-Splitting)的概念以減少雷射光功率的損失，此方法是先以 Member-Only 演算法來建構群播的光樹或光森林，再重新調整光樹或光森林的架構，使得光樹盡可能只做一次的分裂就傳送到所有的目的節點。在性能上，此演算法平均可減少光功率損失 7%，最多可減少 17%。不過，此演算法需先執行“Member-Only”演算法之後，再調整光傳播樹的樹狀結構，有較高的複雜度。

文獻[6]中提出藉由調整鏈路的比重，使鏈路成本隨著節點的跳躍數增加而遞減，如此一來，光樹的展開，便自然而然地選擇離來源節點較遠、成本較低的鏈路。如此則愈是遠離來源節點的鏈路成本會愈低，也就愈有機會被選機會被選為光樹展開的路徑。比起文獻[2]的演算法，此方法可使得所建立的群播樹平均最小功率(average minimum power)最多高出 46%，平均跳躍數也最多可減少 38%，程式平均執行時間也降低了 75%。

不過，文獻[2]及[6]在評估效能時，都是採計各項效能的「平均值」，例如：平均波長使用數、平均跳躍數等。但是在實際情況下，效能採計平均值在實務上是不合適的，例如群播樹的平均最小功率就是這樣，即使某種演算法構成的群播樹平均最小功率提高了，但變異數太大的話，此表示仍然有一些群播樹末端的最小功率太低，以致目的節點的接收器因訊號雜訊比太小而無法接收。是故，改善效能「平均值」的演算法並不能符合 WDM 網路實務上之所需。

因此，本文的研究的目標是提出一個修正群播演算法的策略，此修正後的演算法必須可以在稀疏分波多工網路上建構群播樹以繞送群播通量的演算法，並確保傳送至各目的節點的光功率滿足最低的功率臨界值，以符合 WDM 網路實務上的需求。

本文其餘部份安排如下：首先於第 2 節分析造成光功率衰減的因素，並推演計算靜態群播樹傳送至目的節點光功率的法則；第 3 節則提出一種有效率地計算動態群播樹的最小功率的方法；應用這種方法，第 4 節則提出修正既有群播演算法的策略，以確保傳送至各目的節點的光功率；第 5 節則呈現修正後演算法的效能分析；並在第 6 節做出結論。

2. 群播樹的功率分析

2-1. 由下往上(bottom-up)法則

如圖 1 所示，我們可以觀察到在群播樹中，任意節點 x 之功率衰減是由於來源(source)節點 s 至該節點多次分裂的相乘效應所造成。若我們假設饋入

(incoming)任意節點的功率，經分裂為 l 個分支(branch)之後，各分支所得之光功率為 $1/l$ 。即各分支之功率分配為均勻分佈，且在有分光能力的節點中，無功率損失。則任意節點 x 之功率計算，可以延著由 x 回溯到來源節點 s 的路徑，將各父節點(parent node)的分支數(或子節點數)的倒數相乘，即可得 x 之功率 $P_{wr}(x)$ 。

若我們定義 $D(x)$ 為節點 x 的分支數， $H(x)$ 為節點 x 在該群播樹的深度， $P^n(x)$ 為節點 x 之第 n 層父節點，則節點 x 之功率 $P_{wr}(x)$ 表為下式：

$$P_{wr}(x) = \prod_{n=1}^{H(x)} \frac{1}{D(P^n(x))} = \frac{1}{D(p(x))} \times \frac{1}{D(p^2(x))} \times \dots \times \frac{1}{D(P^{H(x)}(x))}$$

例如在圖 1 中， $D(u)=4$ ， $D(t)=2$ ， $D(s)=3$

$$\therefore P_{wr}(x) = \frac{1}{4} \times \frac{1}{2} \times \frac{1}{3}$$

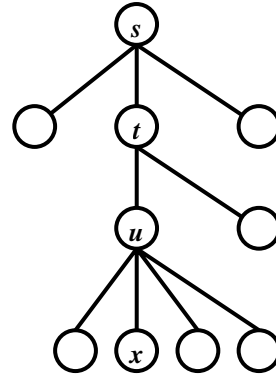


圖 1. 以節點的分支數 $D(x)$ 計算靜態群播樹的最小功率，其中 $D(u)=4$ ， $D(t)=2$ ， $D(s)=3$

2-2 由上往下(top-down)法則

再度觀察圖 1 之群播樹，我們也可以用由上往下的手法來計算節點之光功率，假設來源節點 s 之光功率為 1，由於 s 有 3 個分支，則 s 以下深度為 1 之各節點功率各為 $1/3$ ，記錄此值於節點 t ，再往下一層探索，由於 t 節點有 2 個分支，所以第二層各節點之功率為 $(1/3) \times (1/2) = 1/6$ ，同理，記錄 $1/6$ 於節點 u ，則 x 節點(及所有 u 之子節點)功率為 $(1/6) \times (1/4) = 1/24$ 。此法最大的好處是可寫為遞迴(recursive)程式(如附錄 1)。

不論用 2-1 或 2-2 節的方法，計算出所有目的節點之功率後，還須經逐一比較，方可找到群播樹之最小功率 P_{min} ，即 $P_{min} = \text{Min} [P_{wr}(x_i)]$ 。

3. 符合功率預算的群播樹

如上節對群播樹功率的計算，並不能保證長出符合光功率預算的群播樹，因為上節的計算是基於群播樹為已知的前提之下，若群播樹已經建構完成後才發現最小功率 P_{min} 低於臨界值 P_{th} ，則只好放棄此群播樹，或如文獻[2]提出的方法以繁複的方式重新調整群播樹，所以最好是以探索式(Heuristic)演算法逐步擴張群播樹的過程中，隨時監視群播樹的最小功率 P_{min} 是否低於最小光功率臨界值 P_{th} ，因此在我們這個研究中，如何快速又有效率地計算出 P_{min} 就成為一項重要的課題。

當然，我們也可以採用上一節分析靜態群播樹功率的方法，每長出一個新的分支就計算一次，但顯然太沒有效率。我們也不能只計算接到剛剛長出來新分支上節點的功率，因為一棵群播樹的最小功率並不一定發生在新長出來的分支上。

例如圖 2-(a) 中，若 (s, x) 是新長出來的分支，則 x 節點的功率為 $1/2$ ，並不是最小功率， u 或 v 節點在 x 節點加入後，功率由 $1/2$ 降為 $1/4$ ，才是群播樹的最小功率。這個例子給我們一個很好的提示，那就是對於動態加入分支的群播樹，我們必須隨時維持一個資料結構，來記錄群播樹的動態分裂狀況，才能有效率地計算加入新分支之後，對最小功率產生的影響。

計算動態群播樹最小功率的方法

對於計算動態擴張之群播樹的最小功率，我們提出的方法是對每一個節點，除了記錄該節點的子節點數 $D(x)$ 之外，還要維持一個足以顯示其子樹分裂狀況的 B 參數， B 參數的定義是：

$$B(x) = \begin{cases} 1 & , \text{ while } x \text{ is a leaf node} \\ \text{Max}[B(x_i)] \times D(x) & , \text{ otherwise} \end{cases}$$

則我們可以迅速地計算出群播樹之最小功率為來源節點 s 之 B 參數的倒數，即 $P_{min} = 1/B(s)$ 。

例如：在圖 2-(a)中，尚未加入 (s, x) 分支之前，

$$\begin{aligned} B(u) &= B(v) = 1, \quad B(t) = 1 \times 2 = 2, \\ B(s) &= 2 \times 1 = 2, \quad \text{故 } P_{min} = 1/B(s) = 1/2 \end{aligned}$$

若在此群播樹中，加入 (s, x) 分支，則由於 x 是樹葉節點，故 $B(x) = 1$ ，我們沿著 x 追溯到 s 的路徑，一路更新所遇到之節點的 B 參數，在此例中，只有 s 節點的 B 參數需更新。由於 $D(s) = 2$ ，則 $B(s) = B(t) \times D(s) = 2 \times 2 = 4$ ，故 $P_{min} = 1/B(s) = 1/4$ 。

以同樣的方式，在圖 2-(b)中， x 節點加在 t 節點上，則需更新的是 $B(t)$ 及 $B(s)$ ，更新後 $B(s) = 3$ ，故 $P_{min} = 1/3$ 。在圖 2-(c)中， x 節點加在 v 節點之後，則更新後 $B(v) = 1$ ， $B(t) = B(s) = 2$ ，故 $P_{min} = 1/2$ 。

計算出 P_{min} 之後，若低於最低功率臨界值 P_{th} ，則放棄這個分支，尋求以其他的分支取代之。

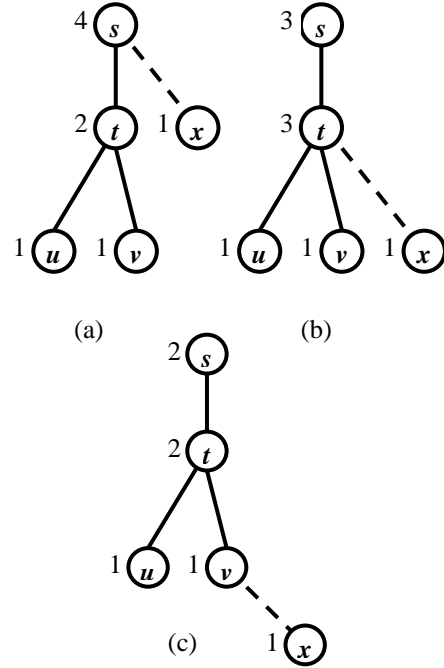


圖 2. 以 B 參數動態計算群播樹的最小功率，節點旁所標示數字為該節點的 B 參數

4. 符合光功率預算的群播演算法

由於我們的演算法是改進 Member-Only 演算而來 [5]，故我們先回顧 Member-only 演算法。再提出我們的修改意見。詳細的演算法則列於附錄 2 之中。

4-1. Member-Only Algorithm

Member-Only 演算法維持如下的動態資料結構：

V_t ：記錄所有可繼續延伸之節點，例如有分裂光能力之節點(MC節點)，及任何一個樹葉節點。

V'_t ：記錄不可能再繼續延伸或擴展之節點，例如沒有分裂光能力之節點(MI節點)。

D^* ：記錄尚未包含在群播樹中之節點。

T ：記錄 current tree。

F ：記錄 current forest。

Member-Only 演算法建構群播樹的方式是一次加入一條連接到目的節點的光徑，距來源端最近或最低成本者優先，但只允許由可分光節點或樹葉節點延伸至下一個目的節點。依此規則即可建構出在稀疏網路的群播樹。

演算法如附錄 2 所示，第(1)步驟是先起始一個光森林 F ，接著第(2)步驟是起始一棵群播樹，並將來源節點 s 加入 V_i 之中。第(3)步驟是尋找由 V_i 中的節點到 D^* 中目的節點的所有可能的路徑，並試著去選出一條最短路徑，若此最短路徑可找到則執行第(4)步驟。第(4)步驟是將此最短路徑加入current tree T 中，並且將此路徑上之所有節點，依可否繼續延伸之特性，分別歸入 V_i 或 V_i' 中，新加入群播樹之目的節點則自 D^* 中刪除，(別忘了剛加入之路徑是由某一在 V_i 中之可延伸節點到 D^* 中之某一目的節點)；接著檢查 D^* ，若已經空了，則停止這個演算法並將 F 輸出；若 D^* 尚有元素，則表示尚有目的節點尚未包含在群播樹中之，故回到步驟(3)去找另一個最短路徑。若 D^* 尚有元素，但目前的群播樹 T 已找不到恰當的路徑來包含這些剩餘的目的節點，則將 T 加入 F 中，並回到第(2)步驟去起始一棵新的群播樹。

4-2. Modified Member-Only Algorithm

我們所提出的修正方式只需在原Member-Only演算法中加入最小功率的限制即可。當第(3)步驟找到一條最短路徑時，先用上一節我們所提出的方法檢驗群播樹的最小功率(minimum power, P_{min})，若 P_{min} 小於最低功率臨界值(power threshold, P_{th})，則放棄此一路徑，改以成本次低之路徑取代，即只有符合最低功率限制的路徑才有可能被選為最短路徑。加上這項限制之後，即可確保群播樹的最小功率不會小於最低功率臨界值。

其實我們以上所提出的修正方式適用在任何基於以探索方式(Heuristic)逐步擴張群播樹的演算法，此處以修正 Member-Only 演算法為例，是因為在文獻[5]中 Member-Only 演算法是效能最佳的一個。

詳細的演算法則列於附錄 2，修正的部份以黑體字加在 Member-Only 演算法中。

5.效能分析(Performance Evaluation)

5-1.模擬方法說明：

為了研究修正後的“Member-Only”演算法的效能，並與原版“Member-Only”及“Central-Splitting”演算法做比較，我們採用 MATLAB 軟體進行數值模擬。

此處所考慮網路拓模是一個如圖 3 所示有 11 個

節點的 NSFNET 網路[5]。為了形成稀疏網路，我們對每一個節點以機率 S 隨機決定是否該節點具有光分裂能力，即若 S 定為 0.8，則平均將有八成的節點具有光分裂能力。接著我們每次隨機產生一個群播通量，來源節點是在所有的節點當中隨機選取，其他的節點則以機率 G 隨機決定是否該節點是否成為目的節點，即若 G 定為 0.6，則平均將有 60%的節點成為目的節點。機率 S 與 G 都是均勻分佈於 0 與 1 之間且彼此互相獨立的參數。

群播通量產生後，我們便以選用的演算法來建構群播光樹，並將產生的光樹計算其效能值(performance)，此處我們定義兩個在本文中檢視的群播樹效能：

- (1) **最小功率(minimum power)**：光源經繞路分裂之後，群播樹中的每個目的節點所接收光功率各有不同，這些目的節點所獲得的光功率中最小的一個，即是群播樹的最小功率。
- (2) **平均跳躍數(average numbers of hop count)**：由來源節點到目的節點的最短光路徑，需經過的鏈路數目便稱為是該目的節點的跳躍數。將光森林中的每個目的節點的跳躍數的和，除以總目的節點的數目，即是光森林的平均跳躍數。

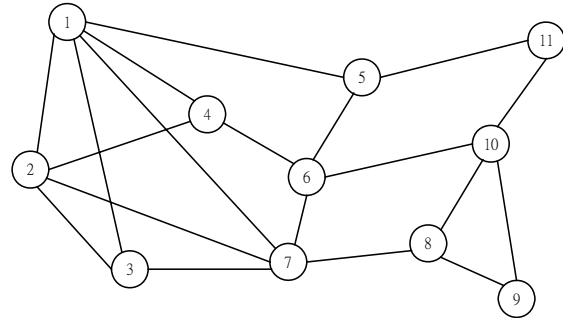


圖 3. NSFNET 測試網路(testbed)拓模圖

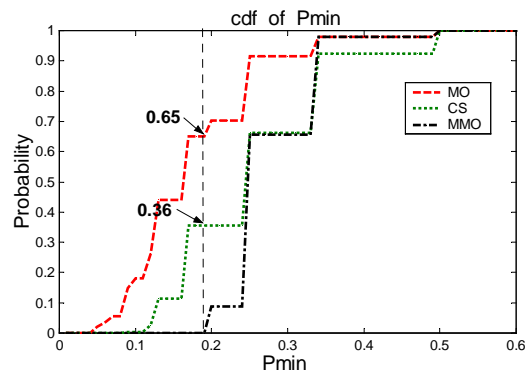


圖 4. 在 S 為 0.8, G 也為 0.8 的假設條件之下，比較 MO, CS 及 MMO 三種演算法的最小功率 P_{min} 之累積分佈函數 cdf。

5.2. 演算法效能之比較

如圖 4 所示，我們先假定有光分裂能力節點之機率 S 為 0.8，產生目的節點之機率 G 也假定為 0.8，以此等條件產生一萬次的群播通量(multicast traffic)；然後以三種演算法 Member-Only (MO), Centralized-Splitting(CS), 及 Modified Member-Only (MMO) 分別建構群播樹，並分別統計各群播樹的最小功率 P_{min} 之累積分佈函數(cumulative distribution function, cdf)，若假設群播光樹的來源節點之發射功率為 1 時，最低功率臨界值 P_{th} 定為 0.2 (即假設當 P_{min} 大或等於 P_{th} 時，目的節點的接收器可正常接收)，則我們可由圖中觀察到 MO 將有 65% 的失敗率，CS 將有 36% 的失敗率，MMO 則可以完全分佈在最低功率臨界值 P_{th} 之上。

MMO 演算法可以保障傳送至各目的節點的最低功率，所付出的代價是跳躍數的增加。如圖 5 所示，在 S 為 0.8， G 也為 0.8 的假設條件之下，我們可觀察到以 MMO 演算法所建構的群播樹之平均跳躍數將隨著 P_{th} 之增加而增加(注意此圖是以 $1/P_{th}$ 為橫軸)。對比於 CS 演算法平均跳躍數固定為 3.22，MMO 演算法平均跳躍數都是較低的；對比於 MO 演算法平均跳躍數固定為 2.15，MMO 演算法在 P_{th} 為 0.2 時($1/P_{th}$ 為 5)，其平均跳躍數 2.29，只比 MO 演算法高 6.5%；當 P_{th} 小於 1/8 時，其平均跳躍數與 MO 幾乎相同。

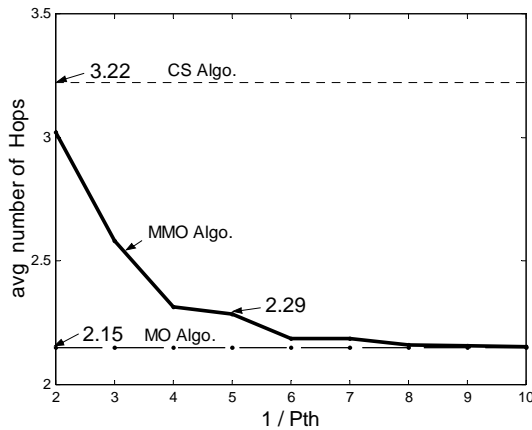


圖 5. 在 S 為 0.8， G 也為 0.8 的假設條件之下，觀察以 MMO 演算法所建構的群播樹之平均跳躍數(Hops)隨著 P_{th} 之變化，注意此圖是以 $1/P_{th}$ 為橫軸。

為了比較各演算法的執行時間，如圖 6 所示，在 S 為 0.8，最低功率臨界值 P_{th} 定為 0.2 的假設條件之下，我們以三種演算法 MO、CS 及 MM 分別建構群播

樹，並分別統計各演算法累積執行 1000 次所需之時間。我們可觀察到各演算法所需之執行時間皆隨著目的節點之增加而增加，其中未考慮群播樹光功率衰減的 MO 演算法所需時間最低，而在考慮群播樹光功率衰減的兩種演算法中，本文所提出的 MMO 演算法則顯然比 CS 演算法低了許多，平均約為 CS 演算法的 24%。

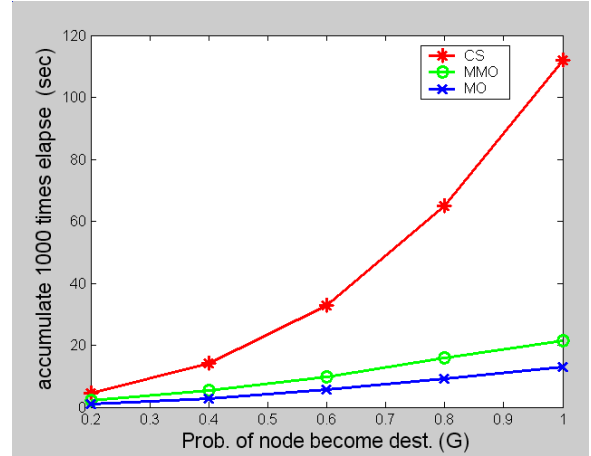


圖 6. 在 S 為 0.8， P_{th} 為 0.2 的假設條件之下，比較 CS, MO 及 MMO 三種演算法各累積執行 1000 次所需之時間。

6. 結論

在稀疏分波多工網路上以光功率分裂的方式來實現群播時，顯然比在 IP 層來施行群播更有效率，缺點則是會導致雷射光功率嚴重衰減的後果。為了研究改善這個缺點，本文首先分析造成光功率衰減的因素及推演計算靜態群播樹最小功率的法則，並提出一種有效率地計算動態群播樹的最小功率的方法，應用這種方法，本文提出修正 Member-Only 演算法的策略，修正後的演算法可確保傳送至各目的節點的雷射光不低於於最低功率臨界值。

我們的數值模擬顯示，改進後的 Member-Only 演算法確實可以使群播樹的最小功率完全分佈在最低功率臨界值 P_{th} 之上；且其平均跳躍數比 CS 演算法的還要低，且在 P_{th} 為 0.2 時，平均跳躍數只比 MO 演算法增加 6.5%；而在 S 為 0.8 的假設條件之下，程式執行所需時間則平均約為 MO 演算法的 24%。

本文中的模擬場景選擇以 NSFNET 為例，這是為了方便與現有的文獻[2][5]作比較，事實上本文所提出的方法可適用在任意的網路拓樸上。NSFNET 是一個只有 11 個節點的網路，但當網路越趨於大型時，任何演算法建構的群播樹分叉必然越多，目的

節點的雷射光功率衰減的問題必然越趨嚴重，則本文所提出的觀點就應該會更加適合被考慮引用之。

7. 致謝

本研究承蒙永達技術學院研發處提供經費支援，計畫編號為 YTIT(93)-ES-11，謹此致謝。

8. 參考資料

- [1]. B. Mukherjee et al., "Some principles for designing a wide-area optical network." *IEEE/ACM Trans. Networking*, vol.4, pp. 684-96, Oct. 1996.
- [2]. K. D. Wu, J. C. Wu, and C. S. Yang, "Multicast Routing with Power Consideration in Sparse WDM Networks," *IEEE ICC2001*, pp. 513-517, Helsinki, Finland, June 2001.
- [3]. Laxman H. Sahasrabudde and Biswanath Mukherjee, "Light-Trees : Optical Multicasting for Improved Performance in Wavelength-Routed Networks," *IEEE Communications Magazine*, pp. 67-73, Feb. 1999.
- [4]. V. Kompella, J. Pasquale, and G. Polyzos, "Multicasting for Multimedia Applications," *IEEE INFOCOM*, pp. 2078-2085, May 1992.
- [5]. Xijun Zhang, John Y. Wei, and Chunming Qiao, "Constrained Multicasting Routing in WDM Networks with Sparse Light Splitting," *Journal of Lightwave Technology*, vol. 18, pp. 1917 -1927, Dec. 2000.
- [6]. 蔡財量，吳介騫 "在分波多工網路上降低光功率損耗的群播演算法之設計"，2002 年全國電信研討會，南投縣，國立暨南國際大學，2002 年 12 月。

9. 附錄

附錄 1. 計算任意節點光功率之遞迴程式

```

w=1
  Evaluate( $P_{wr}, s, w, T$ )
End;

Procedure Evaluate( $P_{wr}, x, w, T$ )
 $P_{wr}(x)=w$ ; (* $P_{wr}$  is the power to be returned)
If  $D(x)<0$ 
  For each child of  $x$ 
    Evaluate( $P_{wr}, \text{child}(x), w/D(x), T$ );
  Else return;

```

附錄 2. 原版及修正後之 Member-Only Algorithm，修正的部份以黑體字表示之。

Modified Member-Only Algorithm

- (1) $F(s,D)=\phi, D^*=D$;
- (2) $V_T=s, V_T'=\phi$, and $T=\phi$;
- (3) try to find the shortest path $P(v, u)$ **under the constraint $P_{min} < P_{th}$** , where $v \in V_T, u \in D^*$, which does not involve any node in V_T' ;
- (4) if such a path is found
 - {
 - add every link $e \in P(v, u)$ to the multicast tree T ;
 - if v (which just became a nonleaf node) is MI, move v from V_T to V_T' ;
 - for any node y on $P(v, u)$, where $y \neq u$ and $y \neq v$ (y is a nonleaf node)
 - if y is MI, move y to V_T' ;
 - otherwise, move y to V_T ;
 - move u from D^* to V_T ;
 - if $D^*=\phi$, stop; otherwise, go back to step (3);
 - }
- else (i.e., no such a path can be found and $D^* \neq \phi$)
 - move the branches in T to $F(s, D)$ and go to step (2) to construct another tree;