

在隨意無線通訊網路上利用最小干擾路徑以提高群播繞徑效率之研究

王三元¹、林冠佑²、黃蓮池³

義守大學 資訊工程系^{1,2} 義守大學 電機工程系³

E-mail: (sywang,lain)^{1,3}@isu.edu.tw joe2@mail2000.com.tw²

摘要

群播(Multicasting)是一個效率高的一對多通訊的方法，通常是以建立一個群播樹(multicasting tree)來施行。因為在Ad hoc 無線網路中侷限的電池能量與傳輸頻寬的限制，群播可以顯著地增進此種網路的性能績效。由於Ad hoc 網路的拓樸改變頻繁與難以預測，群播樹的維護來確保它的可利用性，將是一份困難的工作。

本論文在探討依連結狀態利用最小干擾路徑演算法的概念來增進隨意無線網路上的群播效率，以群播路徑協定為基礎實行群播繞送，提出了一個針對目的端作最小干擾的路徑維護方法，就其所作出的群播樹在封包傳送中，有一些封包競爭問題值得探討，並提出一個改善的方法，以提高資料封包傳送率，降低封包延遲。

本研究將以QualNet 模擬器來做實驗，也將開發設計一個發展QualNet 模擬數據的使用者介面，來自動產生大量的模擬數據來彰顯模擬的成效並同時做有效的分析；此一使用者介面大大地節省了需要產生大量的樣本數據的時間，並能夠快速有效的統計出圖表結果。

模擬實做將以ODMRP 演算法為基礎，分別針對不同的模擬環境進行模擬，產生的模擬結果做統計與分析。並與改善後的演算法針對來源端與接收端間的Total Packets Received 與Throughput 做比較分析與探討；模擬結果顯示，改善後的演算法無論在靜止中或是移動中的環境下，在Total Packets Received 與Throughput 的表現上有出現比較好的結果，且在multicast group 與size 較多的情況下愈趨明顯。

1 概論

隨意無線網路是由一群行動路由主機所組成的，他們能夠經由無線連結與任意地自由的移動來完成互相連結，對於使用在多點分散式的即時應用設備，其需要涵蓋非常寬的應用範圍包括複製資料庫的更新與影音視訊會議上。

群播(Multicasting)是一個效率高的一對多通訊的方法，通常是以建立一個群播樹(multicasting tree)來施行。因為在Ad hoc 無線網路中侷限的電池能量與傳輸頻寬的限制，群播可以顯著地增進此種網路的性能績效。

目前在ad hoc 網路上許多主要的群播演算法被提出，如ODMRP[4]，MAODV，CAMP等，

這些協定當中我們可以粗略分類為兩類：shared-based schemes (如shared tree)和source-based schemes (如per-source tree)，針對於移動性以及訊號頻寬的觀點來看，shared-based schemes 要比source-based schemes 來的好。在拓樸架構上大致分有Tree、Mesh Schemes 等形式，在傳統有線網路中，tree 架構是經常被使用於群播繞路上，其拓樸架構之特性是來源端到接收端間只有一條路徑。因此當路徑發生斷裂時，便需要重新建立路徑，如MAODV 和CBT。Mesh 拓樸結構也是許多群播繞路協定使用的拓樸結構，它有許多備份的路徑可以到達相關的目的點。Mesh 拓樸結構可以被用來改善傳輸到達率，但是其群播效率就較不好，因為當點數越多時，所造成的負擔就會越大。如ODMRP 和CAMP。

有關Ad Hoc Network 相關的研究已然成為目前網際網路發展趨勢最值得探討的課題，以下為幾篇有關在隨意無線網路上群播繞徑的研究。Alternate path routing in mobile ad hoc networks (APR)[3]，Reliable Multicast of the On-Demand Multicast Routing Protocol[2]，Routing and Multicast in Multihop Mobile Wireless Networks[1]，Independent-Tree Ad hoc Multicast Routing (ITAMAR)[5]。以上無論是Tree 或Mesh Schemes 形式，所產生的Tree 或Mesh 當在以群播方式發送封包時，將會有大量的碰撞產生而發生競爭的現象使得封包傳送率效能降低，尤其在越接近Source node 的節點競爭現象所產生的影響，更是明顯。

本論文將探討依連結狀態利用最小干擾路徑演算法的概念來增進隨意無線網路上的群播效率，以群播路徑協定為基礎實行群播繞送，提出了一個針對目的端作最小干擾的路徑維護方法，就其所作出的群播樹在封包傳送中，有一些封包競爭問題值得探討，並提出一個改善的方法，以提高資料封包傳送率，降低封包延遲。

本文其餘內容之架構如下，第二章介紹針對利用群播繞徑法則來增進隨意無線網路上的群播效率的相關研究，第三章將說明本文所提出以群播繞徑協定為基礎，針對目的端作最小干擾的路徑繞徑方法，就目前已提出的群播繞送方法做一系列的研究與分析改善。第四章介紹一個以VB 程式語言設計的發展QualNet[8]模擬數據的使用者介面，以QualNet 模擬器為基礎，動態產生模擬環境參數並統計分析製圖的模擬應用程式，第五章模擬結果的部分將說明我們

的模擬環境及設定，並以 QualNet 模擬器作效能模擬，進而對模擬結果作分析及探討；最後一章則是本論文的結論及未來展望。

2 相關研究

本研究的目的著重在以有效的利用最小干擾路徑法則的群播繞徑機制來增進 Ad Hoc 網路上的群播績效(Multicasting performance)，雖然群播繞徑在網路上是一個非常重要的議題，然而，因其在 ad hoc 無線網路上較具困難度，目前所有的研究結果並不算太多，本研究將以 ODMRP (On Demand Multicast Routing Protocol) [4]與 Independent-Tree Ad hoc Multicast Routing (ITAMAR)[5] 做探討。

2-1 On Demand Multicast Routing Protocol (ODMRP)

ODMRP 是屬於 reactive 群播繞路協定。其用 Forwarding Group 的概念建立起 source 至所有 receivers 的一個 mesh 拓樸結構，以便用來傳送所要群播的封包。由於它提供了路徑的備份機制，因此較 tree 更為牢固，不過，它的 overhead 也相對的高。Mesh 拓樸結構可以被用來改善傳輸到達率，但是其群播效率就較不好，因為當點數越多時，所造成的負擔就會越大。特別是當有多個 sources 時，針對每個 source 若都得建立一個 mesh 架構，那網路恐怕會很快癱瘓。

2-2 Independent-Tree Ad hoc Multicast Routing (ITAMAR)

在隨意無線網路所需要的是我們利用非常少的時間來計算樹(tree computation)出來，因此，對於演算法的低複雜度(low complexity)是非常重要的。ITAMAR 的研究中針對時間複雜度(time complexity)、通訊複雜度(communication complexity)的考量下，提出一個有效率的機制來找出備援路徑樹，一個較實用的方法就是計算出足夠獨立的樹，它盡量避免與其他的樹有共同邊(edges)與節點(nodes)，以供資料傳輸失敗時，在最短的時間內達到群播繞徑(multicast routing)的恢復。

對於此研究設計以 Alternate Path Routing 概念提出一個以一組持續保持的 Mulicasting trees 的設計方案，來使得從當一棵樹(Tree)失敗時到群播繞徑(multicast routing)恢復的這段時間，能達到最小。Independent Path Algorithm 在計算樹的路徑時，它是找出對每一個接收點(Receiver)的有效路徑且此路徑盡量避免與之前的樹有共同邊(edges)與節點(nodes)。第一棵樹採用 Dijkstra algorithm 最短距離演算法來計算，對於每一個接收點，運用修改 Cost function 的方式來避免重複使用已使用過的節點(Nodes)和邊(Edges)。

Independent Path Algorithm

T1=Dijkstra_Algorithm(G, Cost, Source, Receivers)

Initialize Cost1 to equal Cost

For receiver node k

{

For link i in Path (in T1) node k

{Cost1i= Costi+ LinkWeight }

For each node in Path (in T1) node k

{

For each link in G incident on this node in T1

{Cost1i= Costi + NodeWeight }

}

Backup Path to k=Dijkstra_Algorithm(G, Cost1, Source, k)

}

Dijkstra's algorithm 是用來找出一個圖形 (graph) 中某點到其它各點間最短路徑的演算法。ITAMAR 用此演算法來找出各 router 之間的最短路徑。利用 Dijkstra ' s algorithm 想要求出圖形中某一點 R 到其它各點的最短路徑，其計算步驟如下：

(一) 開始建立一最短路徑樹 (Shortest Path Tree)，以 R 當作此樹的根點 (root)，並將此樹根當作目前的處理點。

(二) 加入目前處理點的所有相鄰點加到樹中，並且計算出由 root 到這些新加入點的距離。

(三) 由目前已形成的樹中找出最短的一條路徑。若此路徑的最末端點為 L，則此一路徑代表由 R 到 L 的最短路徑。所以刪除在路徑樹中其它末端為 L 的點，不必再找。

(四) 以 L 當作目前的處理點，重覆步驟二、三、四。如果遇到已被找出的最短路徑的點就不必將此點再加入樹中。

3 最小干擾路徑群播繞徑協定 (MIPMRP)

本章將說明本文提出的利用最小干擾路徑以達群播繞徑之研究，說明運用最小干擾路徑演算法作為群播繞徑法則的探討與改善方法。

3-1 群播繞徑演算法的探討與改善方法

上一章節介紹的群播繞徑協定是運用 Mesh 或 Tree Scheme 來達到群播繞徑的目的，但在這演算過程當中有一個盲點產生，所產生的 Tree 或 Mesh 當在以群播方式發送封包時，將會有大量的碰撞產生而發生競爭的現象使得封包傳送率效能降低，尤其在越接近 Source node 的節點競爭現象所產生的影響，更是明顯。我們用圖示來說明問題所在，圖 1 表示整個網路拓譜結構(G)，A 為來源端(Source)，L、K、L 分別為接收端(Receivers)，當在建立第一棵樹時，是用 Dijkstra algorithm 最短距離演算法來計算如圖 2，整個群播樹的形成以紅色線代表。在此例中我們看到當封包開始從 Source 端向 Receiver 端傳送時，會依循樹的路徑廣播出去，Node A 到 Node B、C、D，然後接著尋徑再廣播出去。

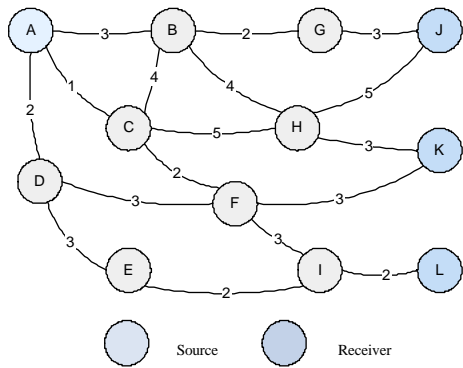


圖 1 MANET 網路拓譜結構

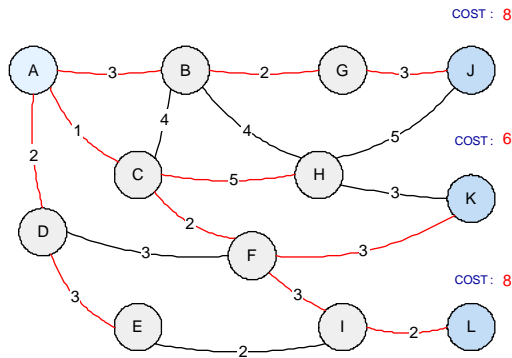


圖 2 利用 Dijkstra algorithm 最短距離演算法，建立第一棵樹

此時，當 Node B 與 Node C 同時要傳送封包時，將會產生競爭，而導致兩個 Nodes 在分食頻寬下封包傳送率降低，往下的 node 直到目的端皆受到影響，如圖 3。

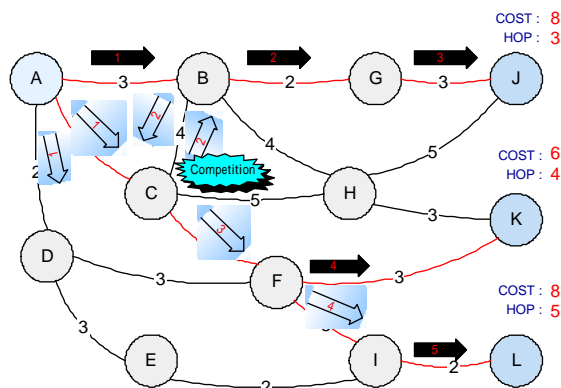


圖 3 群播封包時，產生的競爭現象

因此，當在建立群播樹時，如何避免此一碰撞發生，將是一個可以探討的問題，提出一個

有效的方法，盡量避免有相鄰的枝葉(Leave)和相鄰的 Node 出現，同樣的方法也運用在 T2 備援路徑上，以增加傳輸效能。

以局部修改演算法 Dijkstra algorithm 或運用修改 Cost function 的方式來避免有相鄰的節點 (Nodes) 和邊 (Edges) 的 Node 出現，預期得到的結果如圖 4。在建立群播樹時，以取道節點 D 代替節點 C 來避免與其他節點競爭的可能。

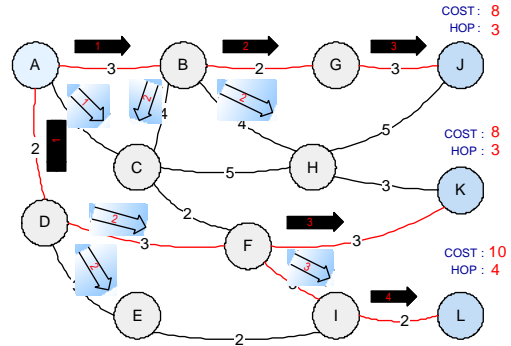


圖 4 局部修改演算法，得到較少干擾、競爭的 Multicasting Tree

為了達到在傳送封包時避免產生競爭的現象，我們提出一個改善並盡量避免有競爭現象產生的演算法，得到較小干擾競爭的路徑，使得在封包傳送的過程當中，避免封包傳送率效能降低，以增加產能。

3-2 較小干擾路徑群播繞徑演算法 (MIPMRP)

在 ODMRP 演算法的改善方面，我們利用在發送 Join Query 封包尋徑時，進行局部的修改演算法以達到在傳送封包時避免產生競爭的現象，得到最小干擾的路徑；以圖 5 的流程圖來說明此改善後的演算法原理：

Step 1:

當 Node 收到 JOIN_QUERY 的封包時，將進入 Handle Join Query 的程序，首先，會檢查自己本身的 Route Table，察看是否來源目的位址是否等於 Next Node 的位址，如果是的話表示此 Node 為第一個 hop，並且曾經收到 JOIN_QUERY 的封包，避免競爭現象產生我們將 Next Node 的位址指定為最後傳送封包的 Node address，如此在 JOIN_REPLY 封包尋徑時就會走另一路徑；如果否的話，將進入下一個步驟。

Step 2:

接下來檢查 Message Cache 是否曾經收到重複相同 JOIN_QUERY 的封包，如果是的話表示已經曾經 forward package 了，就直接離開不處理；如果否的話，就加入此筆紀錄到 Message Cache 與更新 Route Table 等程序，再繼續以群播的方式傳送封包到鄰近的 Node。

Step 3:

在 Reply Phase 時，JOIN_REPLY 封包尋徑時就會以 Route Table 完成一條 source 到 destination 的封包傳遞路徑。(如圖 6)

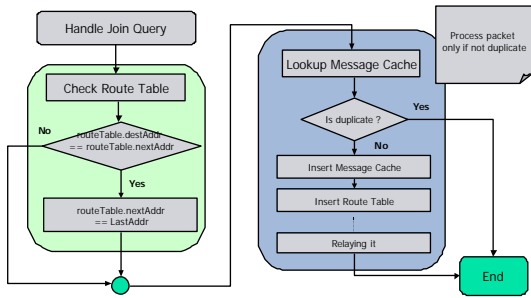


圖 5 MIPMRP 改善 ODMRP 演算法的流程圖

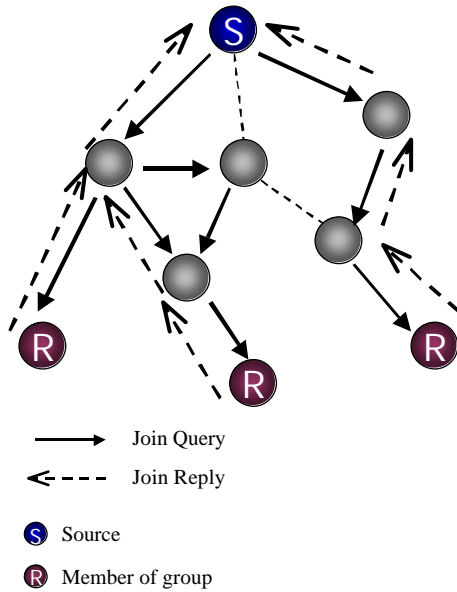


圖 6 MIPMRP 示意圖

4 發展 QualNet 模擬數據的使用者介面

本研究將以 QualNet 模擬器來做實驗,由於需要大量的模擬數據來彰顯模擬的成效並同時做有效的分析,是故先行研究開發一套發展 QualNet 模擬數據的使用者介面,以增進產生快速而可靠模擬演算法的結果。本應用程式除了主要基本功能有載入檔案、顯示檔案、列表瀏覽、搜尋功能外,並有統計分析製圖功能、資料庫的運用處理、動態新增模擬的參數與執行 Qual Net 並產生模擬數據等功能。

4-1 基本功能架構

應用程式基礎平台整體上以簡單的按鈕圖示來執行各項功能,基本功能包含有離開主程式、清除檔案、載入檔案、顯示檔案、列表瀏覽、搜尋等功能。如下圖。

使用者首先必須載入欲作用的檔案 (.stat 或.config),當系統載入檔案內容後會經過整理分析,分別以 Node,IP,Protocol 等有系統的呈現在畫面上,讓使用者能夠依不同的需求點選不同的 Node,察看模擬過程當中各層的狀況與資

料數據,並且可以運用搜尋的功能,針對特定的 Node 或關鍵字快速的找尋所在位置,以利對部分演算法的成效做事後效能判斷上的參考。

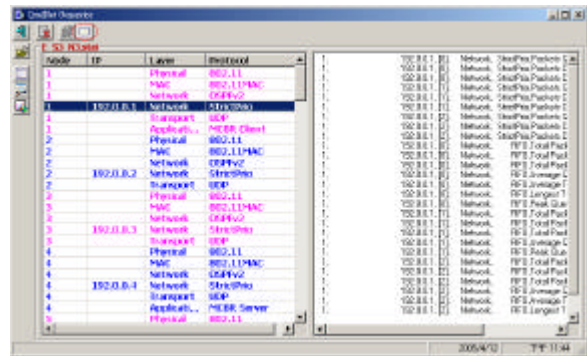


圖 7 顯示檔案列表瀏覽畫面

4-2 動態新增模擬的參數 Parameter Generator

此功能主要的目的是由系統依設定的參數,進行相關的設定檔的產生,並能快速有效的產生模擬數據。在相關的設定檔的命名方面,系統將統一檔案命名規則,避免覆蓋先前模擬的數據並有效管理參數與模擬數據檔案。如圖 8。

在 General Setting 的區塊中,可設定一般參數如工作路徑、實驗名稱、模擬時間、應用層的協定、群播群組位址等。Group Size, Group Count 也依模擬的需要填入適當的數值,當要由系統自動產生模擬數據時,系統將參考此參數設定的內容來進行相關的設定檔的產生。

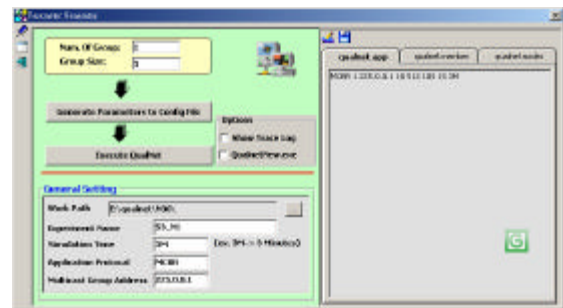


圖 8 模擬參數產生主畫面

此系統提供使用者可直接執行 QualNet 模擬產生數據,按 "Execute QualNet" 按鈕後,系統會依照所鍵入的相關設定檔,直接呼叫 QualNet 主程式並執行,其產生的模擬結果數據檔案將儲存在指定的位置上。

與在 QualNet Developer 下執行做比較,直接執行 QualNet 模擬的優點:

- (1) 方便性提高。
- (2) 產生數據速度快。
- (3) 可觀察模擬運算細節內容。
- (4) 可指定.stat 模擬數據檔儲存位置。

4-3 數據統計分析製圖功能

主要功能為將已完成模擬的數據檔 (.stat) 做必要篩選與統計,並儲存至資料庫,針對特定的參數如 EndToEnd Delay,Throughput 做整合分

析，進而繪製出圖表顯示在畫面上。

4-3-1 Transfer- 資料統計與轉換

此功能運用了資料庫來儲存模擬後產生的結果數據 (.stat 檔)，系統將會從選取的數據檔案中，擷取必要的資訊存至資料表 (StatData)，此資料表紀錄著每個樣本 (sample) 的結果數據，諸如 End2EndDelay、Total Bytes Received、Total Packets Received 與 Throughput 等，最後分別針對 End2EndDelay 與 Throughput 兩項數據做資料統計與分析。

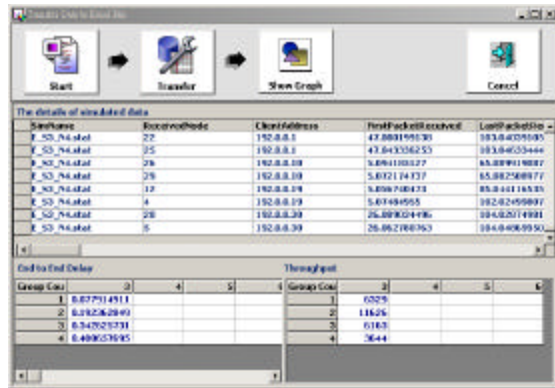


圖 9 模擬參數產生主畫面

4-3-2 Show Graph- 統計數據圖表分析

系統會將 EndToEnd Delay, Throughput 數據資料，以圖形的方式表現出來，在此功能中提供使用者能夠自行調整圖形的呈現方式，可經由左列的功能鍵做變化調整，有圖形旋轉功能、2D/3D 曲線呈現方式、2D/3D 立體圖呈現方式、縱軸與橫軸換位顯示功能，並可以將現有的圖表儲存成影像檔。

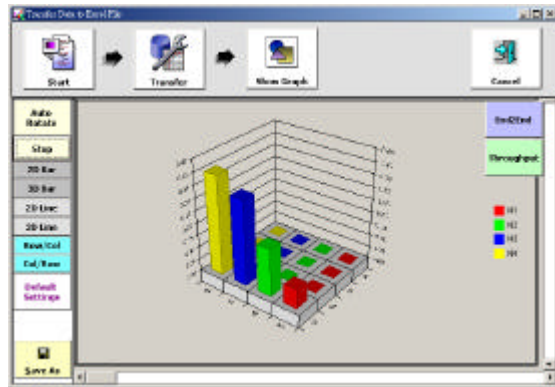


圖 10 模擬參數產生主畫面

5 模擬結果及分析

我們以 QualNet [7] 模擬器做實驗來研究以 ODMRP 演算法做為群播繞徑的行為與效率，並同時做有效的分析。

5-1 模擬環境

在模擬過程當中，我們使用 QualNet 所提供既有的參數與功能，在來源端以 MCBR (Multicast Constant Bit Rate) 帶有 UDP

(User Datagram Protocol) 的機制產生資料封包，並以群播方式傳送資料到接收端。

在模擬實作當中，分別以 ODMRP 與 MIPMRP (Enhanced ODMRP) 為模擬的對象，將從來源端以每個封包量為 512 bytes 的大小向接收端傳送資料封包。有關設定如下：

- (1) 以 30, 50, 70 個 Nodes 在 1500m x 1500m 的區域中隨機分佈
- (2) UDP traffic, 每一個 Package 為 512 Bytes.
- (3) 使用 ODMRP 的繞徑協定, 每 20 秒發送 Join Query 封包。
- (4) The MAC protocol considered is the IEEE 802.11.
- (5) 模擬時間為 3 分鐘, Source node 在第一秒時開始傳送封包, 間隔一秒發送一個資料封包, 總共發送 100 個封包。
- (6) 分別以 group size 為 3, 4, 5, 6 與 group 的個數為 1, 2, 3, 4 的變動為模擬環境。每間隔一秒 Receiver node 依序加入 multicast group.
- (7) 每個模擬條件均以 20 個樣本取平均值
- (8) 節點移動速率: 0, 10, 20, 30 km/hr.

5-2 模擬結果

我們將以自行開發設計的發展 QualNet 模擬數據的使用者介面，來產生模擬結果數據，以增進產生快速而可靠模擬演算法的結果。在模擬過程當中，我們預計針對 group size 為 3, 4, 5, 6 與 group 的個數為 1, 2, 3, 4 的組合為模擬環境，換言之，需要實做有 4x4 16 種的模擬環境來做模擬，每個環境要取樣 20 個，即 16x20=320 需要做 320 個 samples，以產生結果數據來做統計與分析。若以原有的 QualNet 模擬器實做來產生數據的話，每個模擬週期大概需要花費 1 至 5 分鐘，這對於需要產生大量的模擬數據是一大隱憂。至於使用模擬數據的使用者介面來產生數據，整個情況就改觀了，每個 sample 模擬週期大概只需要花費 2 至 7 秒的時間，且能夠自動產生 multicast group 的 nodes，取樣快速，大大地增進了整個模擬的時間。

再者，需要兩個演算法的模擬數據，取樣加倍就要實做 640 個 samples，以單一個 sample 會產生 5 個檔案來看，總共會產生達 3200 個檔案，數目實在驚人，並且更要能夠整合分析處理這些數據且製成圖表。是故，此發展 QualNet 模擬數據的使用者介面對於模擬的結果將是一舉足輕重的角色。

以下將以單一特定樣本與樣本整合統計結果做進一步的探討分析。

• 單一特定樣本

圖 11 為 group size 為 3 與 group 的個數為 1 的模擬環境所產生的結果。由圖表可知，在與原協定 average end-to-end delays 接近的情況下，改善後的演算法 (MIPMRP) 在 Throughput 的表現上有出現比較好的結果。

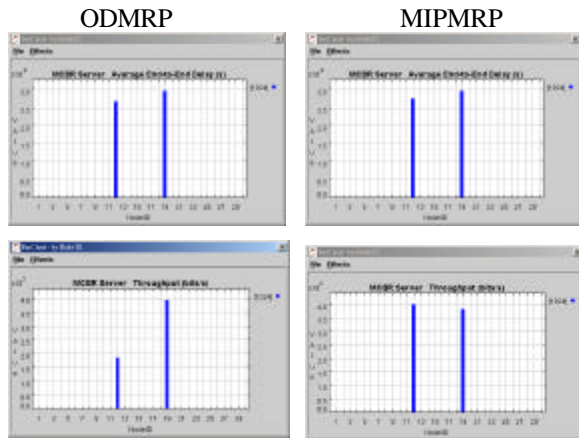


圖 11 單一特定樣本模擬的結果

• 樣本整合統計

圖 12 為 50Nodes 速度 10 km/hr 分別以 group Size 為 3,4,5,6 與 Number of group 為 1,2,3,4 的變動為模擬環境所產生的結果。由圖表可知，當 multicast group 的數目越多時，Total Packets Received 與 Throughput 越少，此因為 multicast group 的增加會造成彼此間競爭碰撞機率增大，封包的流失率也就增加，從 source node 到 destination node 之間的封包傳遞將減弱而導致 Total Packets Received 減少，Throughput 也為之減少。

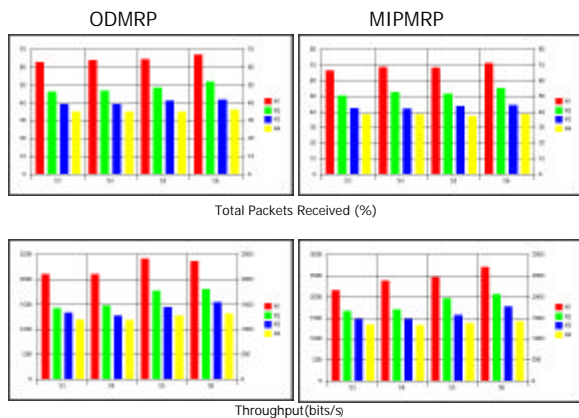


圖 12 樣本整合統計模擬的結果

在與 ODMRP 協定比較模擬結果，改善後的演算法 (MIPMRP) 在 Throughput 的表現上同樣地有出現比較好的結果，且在 multicast group 與 size 較多的情況下越明顯。(表 1、2)

Num. Of Group	Group size			
	3	4	5	6
1	2105.375	2107.534	2408.113	2361.53
2	1420.95	1476.491	1775.878	1807.155
3	1336.441	1277.188	1449.33	1540.725
4	1191.978	1189.742	1272.714	1317.425

表 1 ODMRP 的產能
Throughput (bits/s)

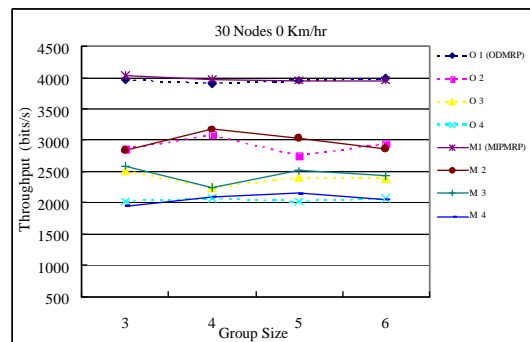
表 2 MIPMRP 的產能

Num. Of Group	Group size			
	3	4	5	6
1	2149.125	2376.3	2460.45	2707.19
2	1661.15	1695.659	1966.178	2060.625
3	1465.457	1468.355	1566.277	1771.896
4	1330.485	1317.962	1361.702	1412.973

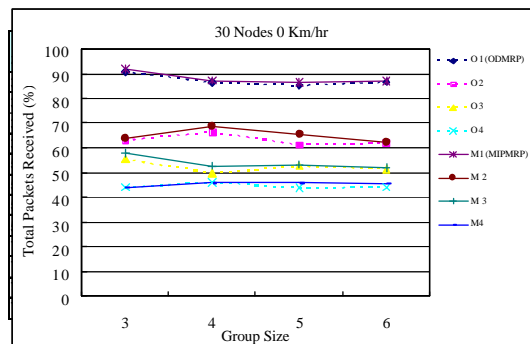
以下為以

- 靜止中密度的變化：分別以 30, 50, 70 Nodes 的模擬環境。
 - 移動中速率的變化：分別以速率為 10, 20, 30km/s 的模擬環境。
 - Traffic Load 的變化：分別以 Number of Group 為 1,2,3 與 Group size 為 3,4,5 的模擬環境。
- 整合統計的結果。

表 3 MIPMRP 對於 ODMRP 產能增加的比率



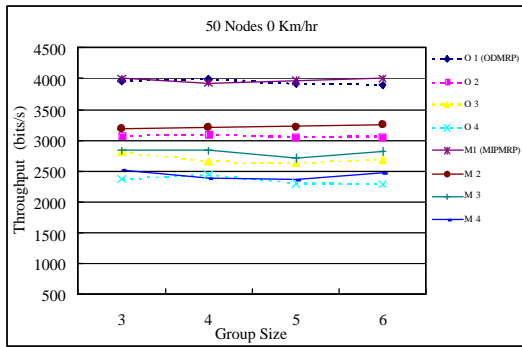
統計表



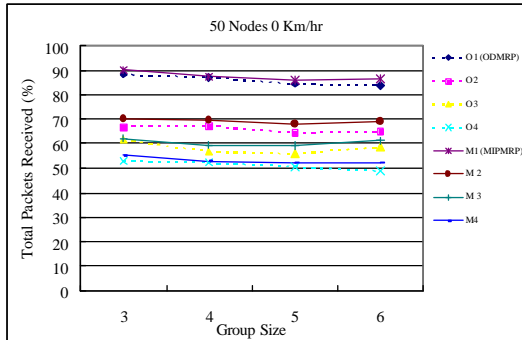
(a) Throughput

(b) Total Packets Received

圖 13: 靜止中 30Nodes 以 group 的個數 1,2,3 和 4 整合模擬的結果

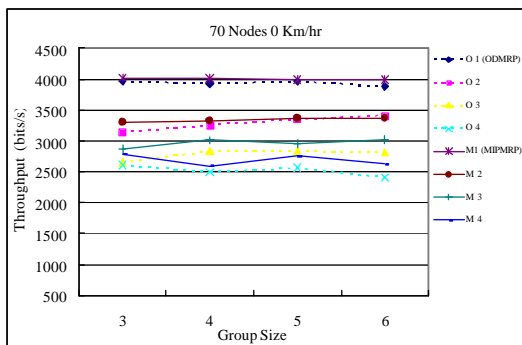


(a) Throughput

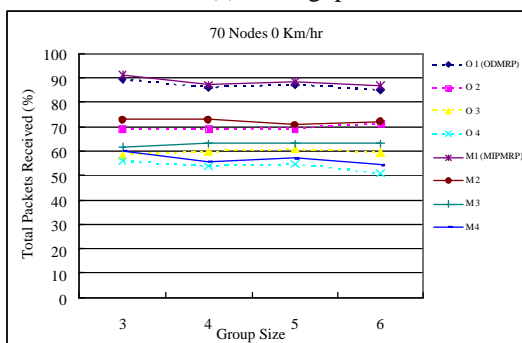


(b) Total Packets Received

圖 14: 靜止中 50Nodes 以 group 的個數 1,2,3 和 4 整合模擬的結果



(a) Throughput



(b) Total Packets Received

圖 15: 靜止中 70Nodes 以 group 的個數 1,2,3 和 4 整合模擬的結果

由實做模擬的結果得知，有以下幾項將影響模擬效能的因素：

- 拓樸結構的密度的影響

結構的緊密與寬鬆將影響尋徑的效能。在模擬結果發現 MIPMRP 演算法與原來的 ODMRP 比較，以 70 nodes 的環境下(圖 15)throughput 明顯的比 30 nodes(圖 13)的來得好，亦即結構越緊密，MIPMRP 演算法 throughput 表現得比原來的 ODMRP 越佳，靜止時 30 nodes 只增加 1.44% 而 70 nodes 增加了 3.7 % (表 3)。

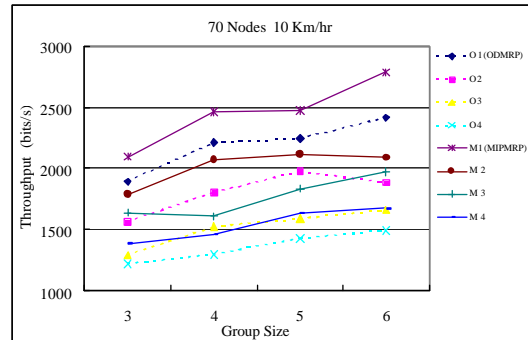


圖 16 移動中以 70 Nodes 速率為 10km/hr

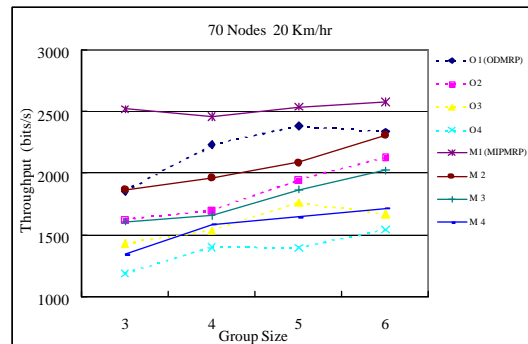


圖 17 移動中以 70 Nodes 速率為 20km/hr

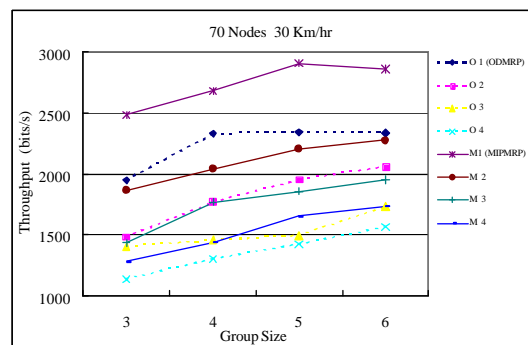


圖 18 移動中以 70 Nodes 速率為 30km/hr

- Multicast Member 的相關位置
 - 模擬時 Multicast Member 是以程式動態產生的，每一個 node 之間的相關位置有可能影響模擬的結果，會在 group member 均勻的分佈在網路各地時，模擬效能會比原有的來得好。
- 靜止與移動中速率的變化的影響
 - 實驗結果顯示，在移動中的環境其

Throughput 比靜止時的來得少(如圖 15、圖 16、圖 17、圖 18)，而在移動中速率的變化上，平均在 1000-2500 bps 的範圍內差異不大，但在 MIPMRP 演算法與原來的 ODMRP 比較上，當速率越大時 MIPMRP 演算法與原來的 ODMRP 兩者 Throughput 的差距明顯的拉大，修改過的演算法表現比較佳。以 70Nodes 為例，當靜止時 Throughput 只增加 3.7%，但是當速率為 10,20,30 時其增加的幅度最多可達 16.89%(如表 3)。

- Multicast Member 的相關位置

模擬時 Multicast Member 是以程式動態產生的，每一個 node 之間的相關位置有可能影響模擬的結果，會在 group member 均勻的分佈在網路各地時，模擬效能會比原有的來得好。

- Traffic Load 的影響

Multicast group 的個數與每個 group 的大小也會影響效能，multicast group 的數目越多時因競爭與碰撞加大 Throughput 會越減少，然而由實驗結果發現 MIPMRP 演算法比原來的 ODMRP，在 Throughput 的表現上也出現比較好的結果。由表 3 整合所有統計結果圖表中，其每一個模擬環境所產生的折線圖皆以 group size 與 group 的個數為基礎分別呈現每一組的效能，也同時可輕易的觀察出 traffic load 的影響，無論在靜止或移動中，當 group size 為 3、group 的個數為 1 時，multicast member 並不多，此時 MIPMRP 與 ODMRP 在 Throughput 上相差不大，一旦 group size 與 group 的數目越多時，Throughput 會越減少，但 MIPMRP 會比 ODMRP 的來得好。但是，當 group size 乘上 group 個數的乘積越接近 Node 的總數時 ODMRP 在 Throughput 表現上較好，如圖 13，30 nodes 在 group size 6 與 group 個數 3,4 時 MIPMRP 的 Throughput 比較差。

6 結論

在隨意無線網路上做群播(Multicasting)比起網際網路(internet)上來得更多的挑戰，因為需要同時對一些資源做最有效的運用。本篇論文使用 ODMRP 協定做為群播繞徑演算法，並運用了一個發展 QualNet 模擬數據的使用者介面，來自動產生大量的模擬數據來彰顯模擬的成效並同時做有效的分析；此一使用者介面大大地節省了需要產生大量的樣本數據的時間，並能夠快速有效的統計出圖表結果。

在利用最小干擾路徑演算法繞徑改善方面，實驗結果顯示與原本協定，在 Total Packets Received 與 Throughput 的表現上有出現較有改善的結果；有如拓樸結構的密度與 Multicast Member 的相關位置等將會是影響模擬效能的因素，在某特定的分佈方式下，模擬效能會比原有的來的好，且在 multicast group 與 size 較多的情況下越趨明顯。

由實驗結果得知，我們可以總結以下影響模擬效能的因素：

- (1) 拓樸結構密度的影響- 結構的緊密與寬鬆

將影響尋徑的效能，結構越緊密，MIPMRP 演算法 throughput 表現得比原來的 ODMRP 越佳。

- (2) 移動中速率的變化的影響- 實驗結果顯示，在移動中的環境其 Throughput 比靜止中的來得少，而在移動中速率的變化上，當速率越大時 MIPMRP 演算法與原來的 ODMRP 兩者 Throughput 的差距明顯的拉大，修改過的演算法表現比較佳。

- (3) Traffic Load 的影響- Multicast group 的個數與每個 group 的大小也會影響效能，multicast group 的數目越多時 Throughput 會越減少，MIPMRP 演算法比原來的 ODMRP，在 Throughput 的表現上也出現比較好的結果；但是，當 group size 乘上 group 個數的乘積越接近 Node 的總數時 MIPMRP 在 Throughput 表現上就較差。

未來，我們希望改良其他 Tree scheme 的群播繞路協定，同樣的以最小干擾路徑的機制，來做進一步的探討，並對 Total Packets Received 與 throughput 等作更詳盡的實驗與分析。

References

- [1] Ching-Chuan Chiang and Mario Gerla, "Routing and Multicast in Multihop, MobileWireless Networks", In Proceedings of ICUPC '97.
- [2] Ken Tang and Mario Gerla, "Reliable Multicast of the On-Demand Multicast Routing Protocol", Proceedings of SCI 2001, Orlando, Florida, July 2001.
- [3] M. Pearlman, Z. Haas, P. Sholander and S.S. Tabrizi, "Alternate path routing in mobile ad hoc networks", MILCOM 2000. 21st Century Military Communications Conference Proceedings 10/22/2000 -10/25/2000, 2000.
- [4] S.-J.Lee, M. Gerla and C.-C. Chiang, "On-Demand Multicast Routing Protocol", Proceedings of IEEEWCNC'99, New Orleans, LA, Sep. 1999, pp. 1298-1302.
- [5] S. Sajama and Zygmunt J. Haas, "Independent-Tree Ad hoc Multicast Routing (ITAMAR)", Mobile Networks and Applications, Vol. 8, No. 5, October 2003.
- [6] UCLA. Glomosim: A scalable simulation environment for wireless and wired network systems.
- [7] Vincent D. Park and M. Scott Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. April 09 - 11, 1997.