

# 無線網路隨意路由(Ad Hoc Routing)協定概論

◎周育德 黃召翔

## 前言

在今日這資訊發達的世界中，無線網路(Wireless Network)已經變得越來越重要了，如何才能使想要傳遞的訊息，又快又準確到達指定的位置，以下介紹路由的相關資訊。無線ad hoc network是以無線通訊來構成一個區域網路的架構，而這項系統的難題大多在ad hoc network要如何去選擇傳送的路徑，因為ad hoc network內的每個用戶都是可動式的，也就是說每個節點都可以任意進入或離開這個network，因此在一個不斷改變的network中去尋找一個溝通的路徑及適當的Routing Protocol是很重要的。

在隨意網路中，因為各節點可以任意行動，因此在封包傳送過程中可能發生接收節點已經離開傳送節點的傳輸範圍，因而導致封包在傳遞當中遺失。當此種情況發生時，要偵測出封包遺失需要等待一段長時間，才能重新開始封包傳遞的作業。近年來衛星通訊協定蓬勃發展，在本篇文章中，將介紹數種可行之路由協定，其中包含利用GPS的裝置，從接收到全球定位系統資料中可推測出各節點的位置及速度，藉由這些資訊，來推測兩節點何時會離開彼此的傳送範圍，做為路徑選擇時的參考依據。但所推測出來路徑何時會失效，如何在路徑失效之前找到新的路徑，讓封包傳輸的過程不致於中斷，可以有效地降低封包的平均傳輸延遲及封包遺失率，是目前研究無線網路路由最重要的課題。

## 介紹

路由器廣義的來說就是我們常聽到的Hub或是IP分享器，在網路上傳送或者接收到的封包(訊息)，可能都不是從發送端直接送到接收端的，而是經過無數個路由器，將封包切割成一定的長度，並且配合網路位址(IP address)，再經過幾次的轉傳才到達我們的眼前，這就好像古時候傳遞訊息的“驛站”，須經過幾次的傳遞才能到達指定的地點。然而，現今的傳播顯然與以前的方式大有不同，在透過路由器就可以收到我們想要的資訊消息等，而且路由器也會自行找出，最合適和快速的路徑來傳送訊息。

路由器是位於OSI中的第三層(網路層)，主要就是將網路上兩台電腦，甚至是多台電腦想要傳送的訊息連接在一起，所以我們才能夠做出傳遞消息、資料以及網路分享的原因。在了解路由器所扮演的角色之後，在現今的網路世界中，並不是只有透過網路連接的方式，因為無線網路的發展迅速，現在有不少的人在研究無線網路的路由方法，就路由的原理來說其連線的方法相同，可透過路由器來傳送和接收，但就字面意義來說的確是很容易，可是要如何做到卻是十分困難的，因為有線連接方法，只須透過連線以及IP位置，就能穩定的接收和傳送，反觀無線網路在這方面反而成了致命傷，雖然能減少線材佈置的問題，可是卻犧牲掉傳輸品質。因此，無線網路的技術在現階段只能算熱門並不能說成熟，所以才會有人仍然在做研究。



其中無線網路的架構又分為：Ad Hoc Mode和Infrastructure Mode兩種，前者是透過主機(傳送)和主機(接收)以點對點的方式傳送和接收，不須經過基地台和AP(Access Point)，而電腦週邊機器只需要無線網路卡即可做到，但後者則必須經過無線基地台，並且必須加上AP才能連上無線網路的環境，因此在學校以及企業較常使用這種方式。本文中，將以LABAR (Location Area Base Ad Hoc Routing) 和ABRP (Ad Hoc Backup Node Setup Routing Protocol) 兩種Routing Protocol做介紹及比較。

### Ad hoc的基本架構

Ad hoc的Routing Protocol可分為三大類，如下表所示。

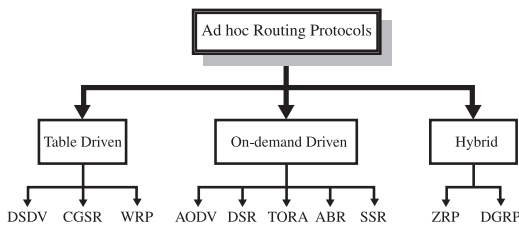


表1. Ad hoc Routing Protocol

### 1. Proactive Protocol

Proactive Protocol (也稱為 Table driven protocol)，每個節點會週期性地傳送或交換 routing information，並藉由收到的 routing information 來更新 routing table，當要傳送資料時節點會從routing table找出一條傳送路徑。Destination Sequenced Distance Vector Routing (DSDV)、Cluster-Head Gateway Switch Routing (CGSR) 和 Wireless Routing Protocol (WRP) 屬於這種 Protocol。因為當節點有封包要傳送時，可以從routing table得知要到達目的地所要走的路徑，可

省去搜尋路徑的時間，但因為每個節點會週期性的廣播交換訊息，以來確認routing table的正確性，相對而言會浪費網路的頻寬和電力。如果只是為了要拉長交換訊息的週期，這又會使得 routing table 的正確性下降，所以 Proactive Protocol 的重要課題，是如何確保 routing table 的正確性。

### 2. Reactive Protocol

Reactive protocol (也稱為 source initiation on-demand protocol)，當節點要傳送封包時才會開始去尋找到達目的地的路徑。節點會先不斷的廣播route discovery的訊息，一直到找到一條或找到可以用的route後才會停止廣播的動作。Ad hoc On-Demand Distance Vector Routing (AODV)、Dynamic Source Routing (DSR)、Temporally Ordered Routing Algorithm (TORA)、Associativity Based Routing (ABR)、Single Stability Routing (SSR) 就屬於這種 Protocol。Reactive Protocol 其優點就是佔用的頻寬資源很小，只有當行節點要送封包時，卻找不到路徑的時候才開始，但要克服的難題是如何減少搜尋路徑的時間。

### 3. Hybrid Protocol

Hybrid Protocol是結合了Proactive Protocol和Reactive Protocol優點的混合式Protocol。Zone Routing Protocol (ZRP)、Zone-Based Hierarchical Link State Routing、Dynamic Group Routing Protocol (DGRP) 就屬於這種Protocol。所有的節點會形成一個區域，而區域的大小和節點數與設定的區域半徑有關，所以在區域內是使用 Proactive Protocol，而區域外是使用 Reactive Protocol。但是

Hybird Protocol 也會面臨著很多困難，例如區域的選擇和維護、以及 Proactive Protocol 和Reactive Protocol 的合理選擇等問題。

### LABAR的架構

在LABAR裡，節點會被GPS系統歸類為G-nodes，G-nodes的功能是在 virtual backbone 中，可以互相連絡和提昇在區域網路內的有效率，並藉由交換訊息來建立 IP 地址清單。因此 Virtual backbone routing protocol 的優點，在 mobile network 內只管理某一個區域的集合，而這些 network 可集合組成一個樹狀結構。並藉由 Virtual backbone 來傳遞，因此 routing information 可以不必將大量的資料傳送到 network 上，而是只有在 backbone 的連結中傳遞。因此 LABAR 是結合了 proactive protocol 與 reactive protocol，而 Virtual backbone 提供封包使用 directional routing 方式傳送到 directional zone 的目的地時，可使用 G-nodes proactive 的方式傳遞訊息和更新 location information。

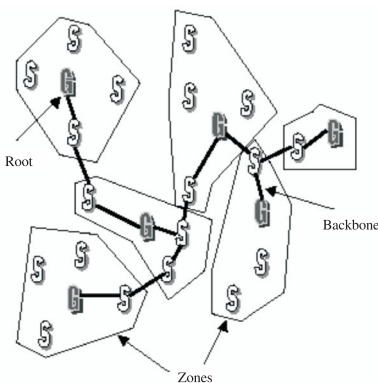


圖 1. LABAR的Virtual backbone架構

### LABAR的路由架構

第一步LABAR和zone會決定S-node屬於那個G-node，G-node在建立zone時，會去搜尋還沒被其他zone使用的S-node來加入自己的zone，如果S-node已經有所屬的zone時，S-node會拒絕其他zone的邀請。在zone裡的S-node會告知附近還沒有zone的S-node，主動找一個zone。因此zone可由數個S-node和一個G-node來構成，而G-node會知道zone內S-node的ID。

第二步是建立一個方便管理的Virtual backbone傳遞節點的位置訊息。G-node在Virtual backbone扮演的角色，是處理區域內的IP address。當zone連接上Virtual backbone時，G-node會從backbone叫出root，而root會傳送連結訊息給附近的zone。如果有zone還未加入其他系統時，則會主動邀請這個zone加入Virtual backbone的系統。如果zone已加入其他的backbone，則會忽略這個訊息。最後的Virtual backbone系統，使G-node再交換control information時比較容易管理。

在direction routing方面，封包內含有目的地節點的zone，而IP地址的區域location清單是由G-node所提供。來源端的節點會詢問Source G-node來找到目的地的位置，G-node也會決定到目的地的location位置，而路徑的決定是由附近的zone找到適合的傳送路徑。在決定zone之後，當封包傳到中繼zone時，節點會詢問G-node來找出附近適合的zone，將封包傳送到下一個中繼zone(或是最後一個zone)。當傳送路徑中有一個節點無法work時，來源端會得到這一個錯誤訊息，此時來源端會找出其他適合的Virtual backbone，繼續把封包傳送到目的地。

在ad hoc routing中，路徑的選擇是重要的一環，使用Monte-Carlo法則來找來源端到目的地的最短距離。LABAR會以routing的平均距離，和最理想的捷短路徑作比較，找出最適合的路徑來傳送資料。

### ABRP的架構

ABRP是以Dynamic Source Routing (DSR)為基礎，ABRP的no-demand routing protocol是以DSR為基礎，DSR不會要求Routing table和來源端到目的地的路由清單，但ABRP允許中繼的節點從來源端重複地接收傳送request封包，與不斷建立backup nodes的資料。ABRP分成三個部分，Routing Discover、Backup Node Setup及Route Maintenance，另外還有三個緩衝區分別為Fresh Routes cache、RD-request cache 和 Backup Routes (圖3)。

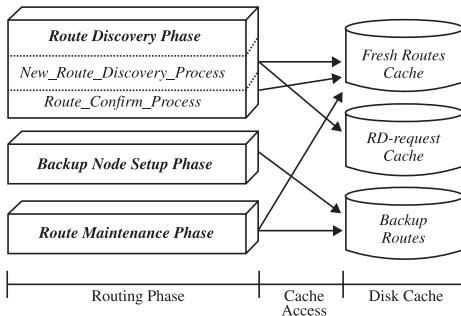


圖3. ABRP的結構

### ABRP的路由方法

第一步來源端由S送資料到目的地端D（圖四），在S會進入Routing Discover從Fresh Routes cache尋找，是否有可以到目的地端D的路徑。如果有找到S便會進入Route Confirm Process。反之，沒有找到S將會進入New Routing Discover Process狀態，來搜尋找到D路徑。

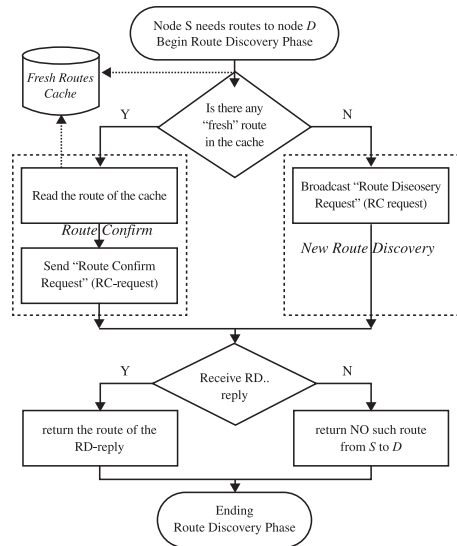


圖4. ABRP 路由流程圖

在New Routing Discover Process中，S會以廣播RD-request的訊息給附近的節點尋找路徑。在RD-request裡會以sequence number來區分Routing Discover Process，其所送來的資訊（S到D的路徑）並會被存入route content，而節點也會把RD-request的位置存入route content，並以廣播RD-request的sequence number和位置給附近的節點。當節點收到由X傳來的同一個sequence number，節點會先檢查route content內是否有這個位置，如果有的話會清除這個RD-request。反之，沒有時會把這位置存入RD-request內。接著節點會檢查X是否有在RD-request，有的話會把X從RD-request cache清除，並再一次傳送RD-request給附近節點的RD request cache，沒有X時會傳送RD-request給附近節點的RD-request cache。

然而ABRP和DSR是有所不同，在採用ABRP的方式時，route content在有節點的位置，與沒有節點的位置可以傳的兩種情況下，都會先把重覆RD-request清

除，因此ABRP可以防止永無止境的傳送。在目的地端D接收到RD-request時，會回覆第一個RD-request給來源端S，然後在tolerable time內等待RD-request，然後進入Backup Node Setup。如果從Fresh Routes Cache可以找到目的地端D的路徑，則S將會把這個路徑加到RC-request中，然後由這個路徑傳送RC-request。在傳送RC-request時所經過的節點會檢查，從Fresh Routes Cache到D的路徑是否比這個路徑好，如果自己的路徑比較好，則用自己的路徑傳；如果對方的路徑比較好，則會把這個路徑存入RC-request，並以這個路徑傳下去。當D收到RC-request時，RD-reply將送回訊息來確定是S要建立這個路徑。

在找到幾個路徑後，將把這些路徑作比較找出相同的部分，而這節點將被設定為Backup Node（在S到D的路徑中），這些節點也會被歸入BS-Packet中，目的地端D可以用BS-Packet來組合出幾條個別的路徑（Backup Route Cache）。當連結發生錯誤，在路徑中有個節點不能傳送，而這個節點也會傳送錯誤訊息給前一個節點，當backup node得知有連結錯誤時，會從Backup Route Cache找出其他的替代的路徑，並通知S更換傳送路徑。Backup Route Cache會將這個出錯的節點從backup node刪除，如果無法從Backup Route Cache找到可替代的路徑，則S也會收到連結失敗的訊息，並重新回到Route Discovery來建立新的路徑。

所以說S無法傳資料到D時，S會把先前所建立的有用路徑存入Fresh Routes cache，然後送出RE-request的訊息，當backup node收到這個訊息後，會Backup Route Cache內的路徑存入Fresh Routes cache。

#### LABAR和ABRP的差異性

LABAR是以proactive protocol和reactive protocol的方式來建立路徑，而ABRP是以reactive protocol的方式來建立路徑，所以proactive protocol的優點是會週期性的更新routing information，因此當有某個節點無法工作時，proactive protocol可以快速的得知並更新路徑，雖然ABRP有Backup Route的功能，但如果沒有其他可以替代的路徑時，必須重新尋找路徑而浪費了不少時間。因此proactive protocol必須不斷的更新，所以當網路的規模變大時，相對的proactive所需的頻寬也會變大。而LABAR為了解決這個問題，所使用Hybrid和geographical routing protocol，但Hybrid protocol是結合proactive protocol和reactive protocol，proactive protocol則是在區域網路，因為區域網路的規模不會很大所以解決了頻寬的問題，reactive protocol是在遠端的傳送。

然而geographical routing protocol主要是用來找節點的路徑或傳送訊息，但geographical routing protocol必須靠GPS來實現，而GPS的價格昂貴和耗電大。如果網路規模不大和經濟的考量下，ABRP會比LABAR適合，如果要解決ABRP找不到路徑的問題，可以在區域間建立幾個固定的節點，而且ABRP有Backup的功能，如果至少有一條固定且故障率低的路徑，則可以減少找不到路徑的風險。

#### 結論

在本文開始提到了關於有線和無線網路的架構和路由方法。在有關Ad Hoc Routing Protocol的兩種介紹和比較，以及其優缺點的說明。在本文中不斷強調的就是如何快速且正確的接收傳送訊息，除了基本的路由條件和架構上的問題，還有必需要克服網路管理，防毒及網路安全性等的幾項問題。

## 名詞說明

1.S-node：在 adhoc中節點分為G-node和S-node兩種，S-node和G-node的差異為G-node把知道的location提供S-node知道，也就是說S-node必須從G-node取得地址清單。

2.Zone：每個S-node都有一個所屬的location area或在G-node的zone內，zone是由一個G-node和數個S-node所組成的。

3.adjacent zone：zone藉由S-node或G-node連結到附近的zone，adjacent zone是由所有G-node所組成的zone之清單，adjacent zone也包括location。

4.N、ni、zj：N是指用戶端的節點，ni是表示i的節點，zj是表示j的節點。

5.Source G-node：Source G-node是指G-node所在的來源端，zone formation process的區域是由G-node和S-node所構成的。

6.Route Discovery Request (RD-request)：從來源端到目的地尋找路徑並記錄在route content。

7.Route Confirm Request (RC-request)：由 Fresh Routes cache 重新建立來源端到目的地的路徑。

8.Route Discovery Reply (RD-reply)：由目的地端回覆route content給來源端。

9.backup setup Packet (BS- Packet)：目的地端傳送backup的訊息來備分節點以建立backup routes。

10.Route Erasure Request (RE-request)：由來源端來決定清除沒有用的路由訊息。

## 參考文獻

1. E.M. Royer, and C-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," IEEE Personal Communications, pp. 46-55, April 1999.

2. X. Cheng, and D-Z. Du, "Virtual Backbone-Based Routing in Multihop Ad Hoc Wireless," submitted to IEEE Transactions on Parallel and Distributed Systems, 2002.

3. C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," Comp. Commun. Rev., Oct. 1994, pp. 234-44.

4. S. Basagni, I. Chlamtac, and V.R. Syrotiuk, "A Distance routing Effect Algorithm for Mobility (DREAM)," ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98), pp. 76-84, October 1998.

5.T. Camp, J. Boleng, B. Williams, L. Wilcox, and W. Navidi, "Performance Comparison of Two Location based Routing Protocols for Ad hoc Networks," Proceedings of INFOCOM 2002, pp. 1678-1687, June 2002.

6.B. Liang, and Z.J. Haas, "Virtual Backbone Generation and Maintenance in Ad Hoc Network Mobility Management," Proceedings of Infocom 2000, pp. 1293-1302, March 2000.