

國立交通大學

電機學院通訊與網路科技產業研發碩士班

碩士論文

提升存活時間之適應性隨意行動無線網路擇路協定

Lifetime-Enhanced Adaptive Routing Protocol for Mobile Ad Hoc Network



研究生：范文炫

指導教授：方凱田 教授

中華民國九十七年二月

提升存活時間之適應性隨意行動無線網路擇路協定

Lifetime-Enhanced Adaptive Routing Protocol for Mobile Ad Hoc Network

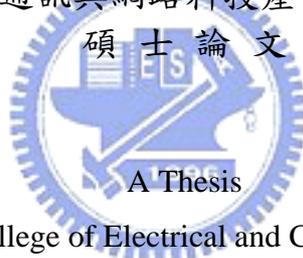
研究生：范文炫

Student : Wen-Hsuan Fan

指導教授：方凱田

Advisor : Kai-Ten Feng

國立交通大學
電機學院通訊與網路科技產業研發碩士班



Submitted to College of Electrical and Computer Engineering
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Industrial Technology R & D Master Program on
Communication Engineering

February 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年二月

提升存活時間之適應性隨意行動無線網路擇路協定

學生：范文炫

指導教授：方凱田

國立交通大學電機學院產業研發碩士班

摘 要

隨意行動無線網路是由一種不需要經由基礎建設的設備所形成的網路。在隨意無線網路中的行動節點可以在彼此之間做封包交換。由於目前的擇路演算法只考慮到中繼節點的數量多寡，而忽略到電池能量限制與移動模式等的影響。在考慮網路存活時間情況下，這篇論文將介紹提升存活時間之適應性擇路演算法(Lifetime-Enhanced Adaptive Routing Protocol, LEAR)，當擇路表單內有多條到目的節點的路徑時，資料的傳送可以選擇避開能量不足的中繼節點，提供單一多路徑資料傳輸的可靠性。當行動運算節點的能量低於設定臨界值時，會發送一個低能量通知的控制封包給上一個傳送資料的行動節點。當行動運算節的收到此控制封包，會更新擇路路表單的剩餘路徑能量，透過此機制以提升整個網路的存活時間及減少行動節點電池耗盡的數量。

Lifetime-Enhanced Adaptive Routing Protocol for Mobile Ad Hoc Network

student : Wen-Hsuan Fan

Advisors : Dr. Kai-Ten Feng

**Industrial Technology R & D Master Program of
Electrical and Computer Engineering College
National Chiao Tung University**

ABSTRACT

The mobile nodes (MNs) in the Mobile Ad hoc NETWORK (MANET) can exchange their data packets directly without the existence of network infrastructure. In recent research work, hop counts within the transmission route are considered for routing decision. However, the influence from the receiving power and remaining energy of the MNs are not fully investigated. This may results in degraded network lifetime within the MANET to successfully transmit the data packets. In this thesis, a Lifetime-Enhanced Adaptive Routing (LEAR) protocol is proposed for unicast multipath routing of data packets. As there are more than two routing paths existed within the routing table, the proposed LEAR algorithm will avoid adopting the low-energy intermediate node in the decision of the transmitting path. The reliability for packet transmission can therefore be preserved. When the remaining energy of the intermediate MN is less than a predetermined threshold, a control packet will be delivered to its previous hop node for noticing the status of low energy. The routing path will therefore be reselected based on the updated energy status. From the simulation results, it can be observed that the proposed LEAR protocol increases the network life-time by reducing the total number of death nodes within the network topology.

誌 謝

兩年的研究所生活隨著這本論文畫下句點。首先要感謝指導教授 方凱田老師於研究以及論文中提供許多詳實的指導與建議，老師嚴謹認真的研究態度以及實驗室定期報告訓練中，令學生受益匪淺。同時感謝交通大學電信系 王蒞君教授以及 廖維國教授，撥空來參加口試，並給予很多實質的指導與建議，使得論文更趨完整。

衷心的感謝實驗室的仲賢、建華、文俊、伯軒及裕彬學長，在我研究過程中遇到盲點時，能不厭其煩地給予我寶貴的意見及指導，讓我能很快完成程式的撰寫與模擬，也感謝實驗室的一起努力夥伴瑜智、信龍、伯泰及林志，還有碩一的學弟們俊傑、佳仕、佳偉及瑞廷，很高興認識你們，由於你們，讓我在這兩年的碩士生涯增添了許多快樂的時光，更有了難忘的回憶，謝謝你們

最後，要特別感謝我的父母以及家人，感謝父母的辛苦栽培以及無怨無悔的支持我，使我得以全心投入學業。在衷心的感謝以上諸位，由於妳們的幫忙，此論文得以完成，衷心感謝。

范文炫謹誌 于交通大學

2008 2月

目 錄

	頁次
摘 要	i
ABSTRACT	ii
誌 謝	iii
目 錄	iv
表 目 錄	vi
圖 目 錄	vii
一、 導論	1
1.1 概論	1
1.2 研究動機	3
1.3 論文架構	3
二、 背景知識及相關研究	5
2.1 媒體存取控制層 (Media Access Control Layer - MAC)	6
2.2 網路通訊層 (Network Layer)	9
2.2.1 單一 (Unicast) 擇路協定	10
2.2.2 單一多路徑擇路協定	16
2.2.3 能量相關的擇路演算法	20
三、 提升存活時間之適應性擇路演算法(Lifetime-Enhanced Adaptive Routing Protocol - LEAR)	24
3.1 設計考量	24
3.2 提升存活時間之適應性擇路演算法設計	25
3.2.1 LEAR 演算法擇路表單設計	26
3.2.2 LEAR 演算法的運作原理	26
3.2.3 資料傳送的過程	33
四、 模擬結果及分析	38
4.1 模擬平台-網路模擬器	38
4.2 模擬環境參數設定	39
4.3 數據分析	40
4.3.1 資料封包的到達率	40
4.3.2 資料封包平均延遲	40
4.3.3 網路負載	40
4.3.4 網路存活時間	40

4.3.5 行動節點電池耗盡的數量	41
4.4 模擬結果	41
五、 結論	49
參考文獻	50



表 目 錄

表 3.1 擇路表單與擇路列表.....	26
表 3.2 RREQ 的格式.....	26
表 3.3 RREP 的格式.....	29
表 4.1 模擬環境.....	39



圖 目 錄

圖 1.1 中央控制型網路.....	2
圖 1.2 隨意行動無線網路.....	2
圖 2.1 IEEE 802.11 競爭期間數值變化.....	7
圖 2.2 隱藏節點問題.....	8
圖 2.3 虛擬載波偵測機制.....	9
圖 2.4 AODV 找尋路徑過程.....	13
圖 2.5 路徑斷裂.....	15
圖 2.6 AOMDV 路徑建立過程.....	17
圖 2.7 AOMDV 路徑斷裂時處理.....	18
圖 2.8 考慮電池能量的路徑選擇方式.....	22
圖 3.1 建立逆向路徑程序.....	28
圖 3.2 RREP 建立正向路徑的程序.....	31
圖 3.3 資料封包選擇能量最大路徑過程.....	34
圖 3.4 行動節點 F 寄送低能量控制封包.....	35
圖 3.5 行動節點 B 及行動節點 C 寄送低能量控制封包.....	36
圖 4.1 根據不同連線數的資料封包的到達率 (平均速度=10m/s)	41
圖 4.2 根據不同平均速度的資料封包的到達率 (連線數=14)	42
圖 4.3 根據不同連線數的資料封包平均延遲 (平均速度=10m/s)	43
圖 4.4 根據不同平均速度的資料封包平均延遲 (連線數=14)	43
圖 4.5 根據不同連線數的網路負載 (平均速度=10m/s)	44
圖 4.6 根據不同平均速度的網路負載 (連線數=14)	45
圖 4.7 根據不同連線數的網路存活時間 (平均速度=10m/s)	46
圖 4.8 根據不同平均速度的網路存活時間 (連線數=14)	46
圖 4.9 根據不同連線數的行動節點電池耗盡的數量 (平均速度=10m/s)	47
圖 4.10 根據不同平均速度的行動節點電池耗盡的數量 (連線數=14) ...	48

一、導論

1.1 概論

隨意行動無線網路（Mobile Ad Hoc Networks，MANETs）為一種隨時隨地都可以建立，並對某一特定應用提供服務的網路。此種網路不需要基地台（Access Point，AP）負責管理整個網路的架構。IEEE 802.11 為無線區域網路規範中最普及的一個，主要是透過基地台（Access Point，AP）負責管理整個網路的通訊與傳輸（見圖 1.1），當網路中的行動節點 A 欲傳送資料給網路內另一個行動節點 B，則行動節點 A 須先將封包傳送給基地台，再由基地台再將封包傳送給行動節點 B。而這種網路架構必須增加架設基地台的成本，並不適用在某些環境上。例如：戰場上的大型的基地台很容易被發現且破壞、救難場所的救難人員之間的通訊，一旦行動節點處在沒有任何固定的基地台下，就無法透過無線通道作傳輸動作。相較於隨意行動無線網路（見圖 1.2），其特徵在於該網路中沒有基地台的存在，傳輸之工作均由網路各行動節點以一對一（Peer-to-Peer）方式進行資料傳送，因具有行動性佳，網路迅速成形的特性，故每個行動節點皆須扮演路由器的角色，進行網路架構的運作與維持。

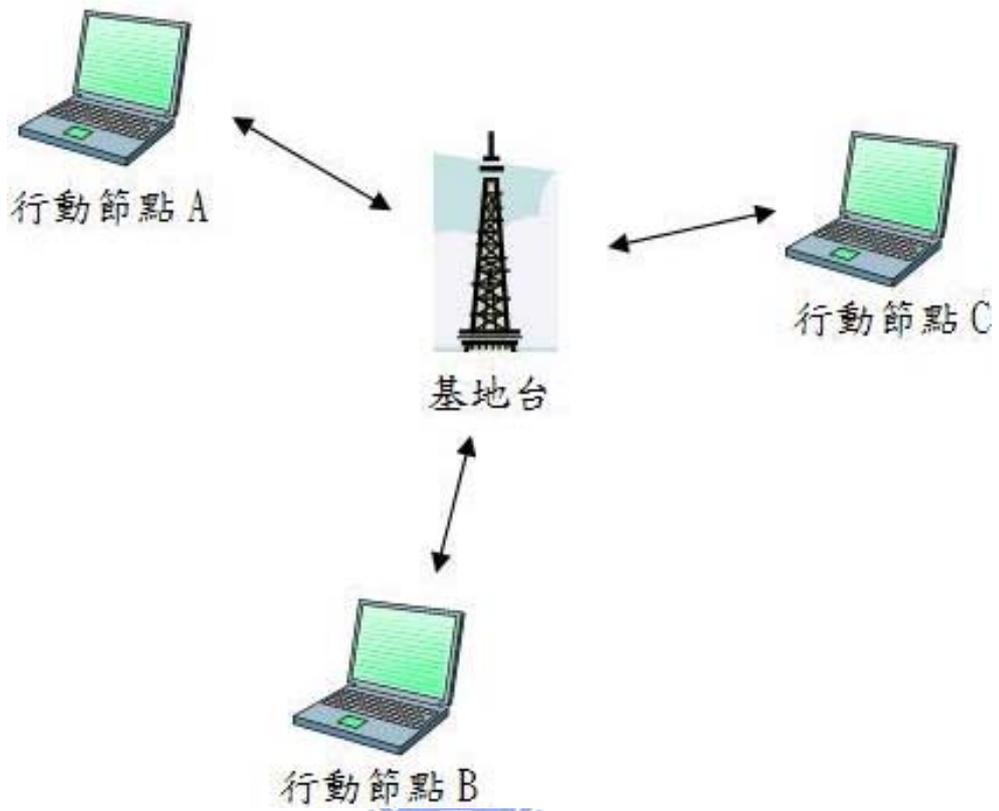


圖 1.1 中央控制型網路

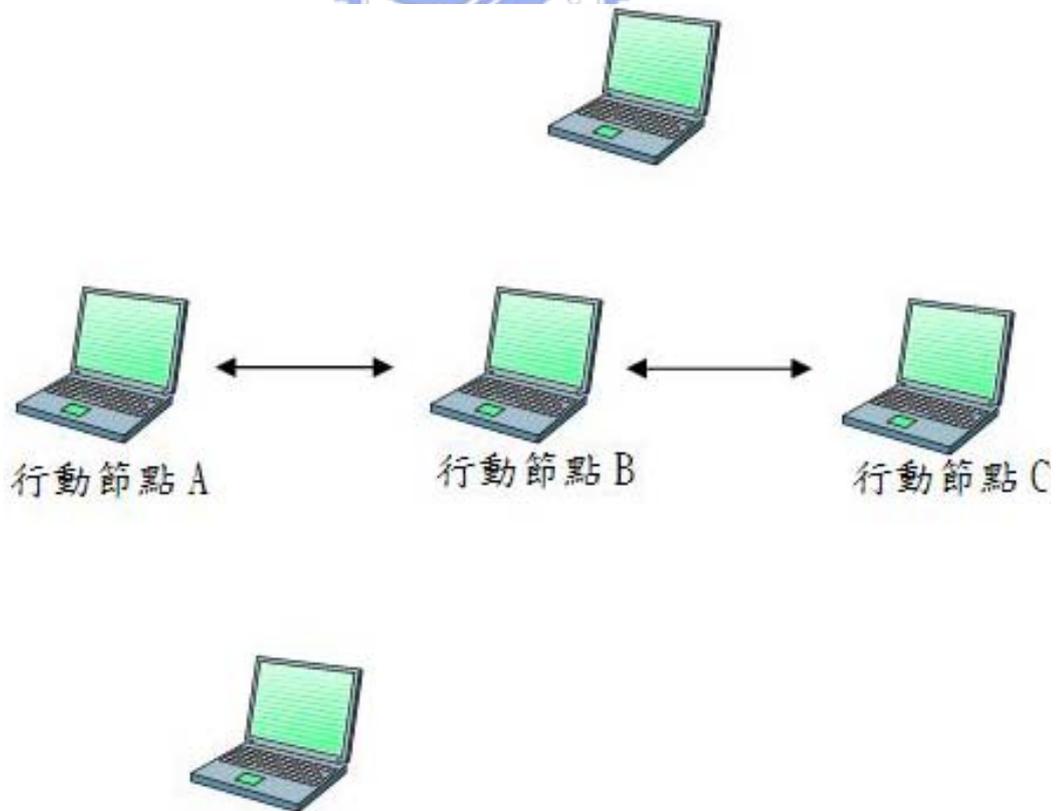


圖 1.2 隨意行動無線網路

1.2 研究動機

由於無線網路是一個快速變動位置且行動節點之間的路由很難長久維持，加上行動節點能量通常為電池，通常更換電池或電池充電比較困難，也因此行動無線網路上的演算法有一部分會以能量為第一優先。而本篇論文主要研究著重於 Routing 層，常見的擇路演算法，有表格驅動式擇路 (Table-driven Routing)[1][2]及需求式擇路(On-demand Routing)[3][4][5][6]的分別，都是在沒考慮到行動節點能量消耗的問題，若將行動節點的能量消耗與持續力列入設計考量，對於路徑及網路存活時間，都會有所改善，同時也增加資料傳遞的效能與網路的可靠性。本篇論文提出以能量為基礎的單一多路徑擇路演算法。經由路徑搜尋的過程，將多條路徑的能量資訊存進擇路表單，資料傳送將依據擇路表單的能量欄位選擇最大能量路徑做傳送，由能量不足發送控制封包來更新擇路表單的能量欄位，改善整體網路存活時間。

1.3 論文架構

以下是本論文的架構，第二章的背景知識及相關研究，先介紹媒體存取控制層在IEEE 802.11的運作方法及原理，接著再介紹目前相關的擇路演算法，包含單一路徑，單一多路徑擇路及能量相關的擇路演算法。第三章是本論文的主題，此章將詳細介紹我們的方法，從如何建立路徑，到資料傳送時，如何切換路徑，及達到提升網路存活時間的目的。第四章模擬結

果的部分將說明我們的模擬環境及設定等，並對模擬結果作分析及探討，
最後一章則是本論文的結論。



二、背景知識及相關研究

目前隨意無線網路相關研究主要由媒體存取控制層 (Medium Access Control Layer, MAC) 跟網路層 (Network Layer) 協定組成。根據國際標準組織 (International Standard Organization - ISO) 所制定的開放式網路架構 (Open System Interconnection - OSI)，媒體存取層屬於資料連結層 (Data Link Layer)，主要功能為建立一條可靠又有效率的連線，擇路演算法 (Routing Protocol) 屬於網路層 (Network Layer) 負責的是收、送兩行動節點之間整條路徑的通信行為，主要功能包括定址，路徑選擇和負責正確的資料流向，本篇論文主要是針對擇路演算法進行研究與設計，為了可以快速的瞭解這些研究與設計，在這一章會先針對媒體層進行相關研究的介紹，接著針對網路通訊層介紹一些擇路演算法做分析及介紹。

IEEE 802.11[7]為眾多無線區域網路 (Wireless Local Area Network WLAN) 規範中最普及的一個，其中定義了實體層 (Physical layer, PHY) 與媒體存取控制層 (Medium Access Control Layer, MAC) 協定，目前大部分的無線區域網路產品都支援 IEEE 802.11a/b/g [8][9][10]。其中 IEEE 802.11a 規範了正交分頻多工實體層 (Orthogonal Frequency Division Multiplexing, OFDM)，運作在 5GHz 的頻帶，傳輸速率為 54Mbps，IEEE 802.11b 規範了直接序列展頻實體層 (HR/DSSS)，運作於 2.4GHz 的 ISM 頻帶，傳輸數率為 11Mbps，IEEE 802.11g 使用 OFDM，運作於 2.4GHz 的

ISM 頻帶，傳輸速率為 54Mbps。未來的 802.11n[11]將加入 MIMO 或高傳輸量實體層，表示 IEEE 802.11 標準已經是一個相當成熟且可實現的規格，所以這篇論文將以這個標準為基礎進行研究，在 2.1 節將會介紹 IEEE 802.11 媒體存取控制層的功能與存取方式。

在網路通訊層部分，因目前並未有任何標準，但已存在很多有關隨意行動無線網路的擇路演算法研究。本篇論文主要探討單一及多重路徑與能量有關的研究。在 2.2 節將會介紹一些已存在的擇路演算法，並討論單一路徑、多路徑及考慮節點能量來傳送資料的擇路演算法。

2.1 媒體存取控制層 (Media Access Control Layer - MAC)



IEEE 802.11 使用碰撞避免 (Collision Avoidance) 機制。在 CSMA/CA 機制中，行動節點在傳送資料封包前，會先偵測頻道中的電磁波能量，若電磁波能量超過基準值，就代表頻道正被使用，因此要等待一段時間該行動節點才可以傳輸資料。分散式協調功能 (Distributed Coordination Function, DCF) 為標準 CSMA/CA 的存取機制，行動節點在傳送資料之前，會先檢查無線媒介的狀態。媒介仍為淨空，行動節點即可開始傳送資料。反之，若無線媒介為忙碌狀態時，行動節點除了延遲等待一段時間之外，還會隨機選取一個延遲 (Backoff) 時間，延遲時間為當媒介為淨空狀態時才會遞減，當時間遞減為零時，即可開始傳送資料。在 IEEE 802.11 中，工作站依

據下列公式來選定延遲時間。

$$\text{Backoff time} = \text{Random} () * \text{aSlotTime}$$

在此公式中，aSlotTime 為時槽時間，時槽時間會依據所使用的實體層而有所不同。Random () 為一個均勻分配的亂數值，其值的範圍介於 0 到 CW 之間，如圖 2.1 為例，CW 的初始值為 aCWmin=7，行動節點會在 0 到 7 之間隨機選出一數值並乘以 aSlotTime 做為延遲時間。當延遲時間倒數為 0 時，行動節點即可傳送資料。但若傳送失敗，則重送時，新的 CW 值會設定為原 CW 值加倍減一，直到 CW 值等於 aCWmax 為止。

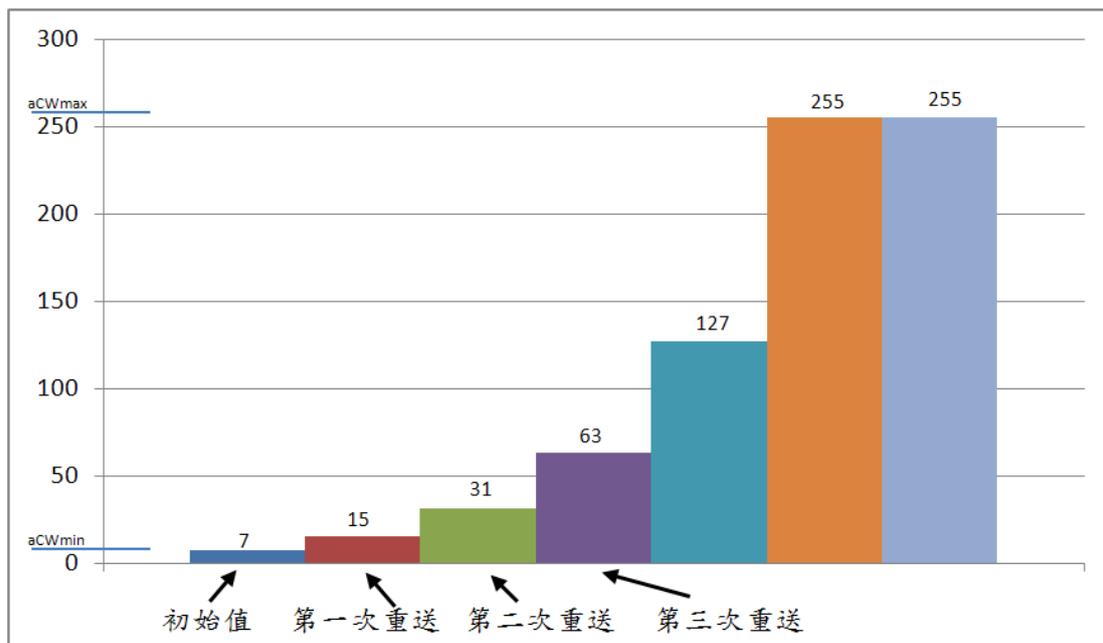


圖 2.1 IEEE 802.11 競爭期間數值變化

若直接使用 CSMA/CA 機制傳送資料封包，則會隱藏了一個嚴重的問題，導致封包碰撞情況發生。該問題被稱為隱藏節點問題 (Hidden terminals problem)。如圖 2.2，行動節點 B 位於行動節點 A 及行動節點 C 的通訊範

圍內，而行動節點 A 及行動節點 C 的傳輸範圍下，無法直接與彼此通訊。假設此時行動節點 A 欲傳送資料給行動節點 B，行動節點 A 等待媒介淨空後直接傳送訊框給行動節點 B，假設在此同時，行動節點 C 也有資料要送給行動節點 B 並且也發現媒介淨空，因此行動節點 C 也會直接傳送訊框給行動節點 B，此時訊框將會在行動節點 B 處發生碰撞。

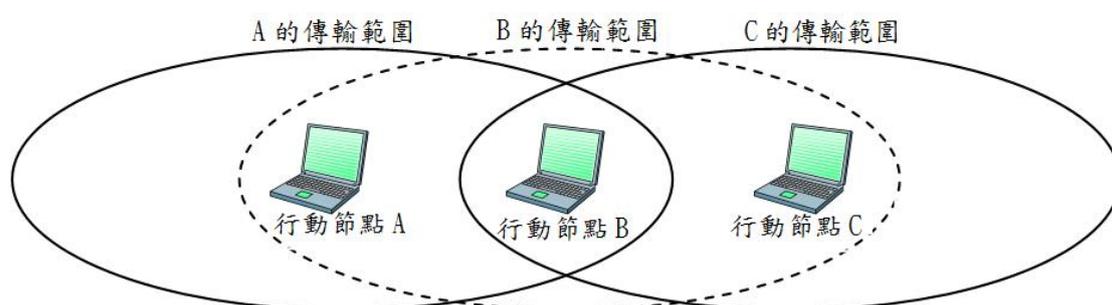


圖 2.2 隱藏節點問題

為了避免隱藏節點問題所造成的碰撞，IEEE 802.11 設計了四段交握（4-way handshake）的流程。當一行動節點要傳送訊框前先送出一個要求傳送（Request to Send, RTS）訊框，而接收端在收到 RTS 訊框後，會間隔一個短訊框間隔（Short Inter-Frame Space, SIFS）時間後送出允許傳送（Clear to Send, CTS）訊框，在完成此流程後，傳送端及接收端即可傳送資料訊框。如圖 2.3，行動節點 A 發送 RTS 給行動節點 B，且行動節點 B 亦回應 CTS 後，行動節點 C 即知道行動節點 B 正與其他行動節點交換訊息，所以行動節點 C 就不應該傳送封包給行動節點 B。至於行動節點 C 該等待多久時間才可以開始傳送資料給行動節點 B，在 IEEE 802.11 中定義了一網路分配向量（Network Allocation Vector, NAV）來記載行動節點將會

使用媒介多久，其他的行動節點會認為媒介在 NAV 資訊的例子。因此，在上一個例子中，行動節點 C 會收到來自行動節點 B 的 CTS，並且可以得知行動節點 B 會在哪時候結束使用媒介。使用網路配置向量概念的好處，在無線網路環境中，實體層使用載波偵測技術容易誤判媒介是否忙碌，而網路配置向量可以說是一種虛擬載波偵測（Virtual Carrier Sense）機制。

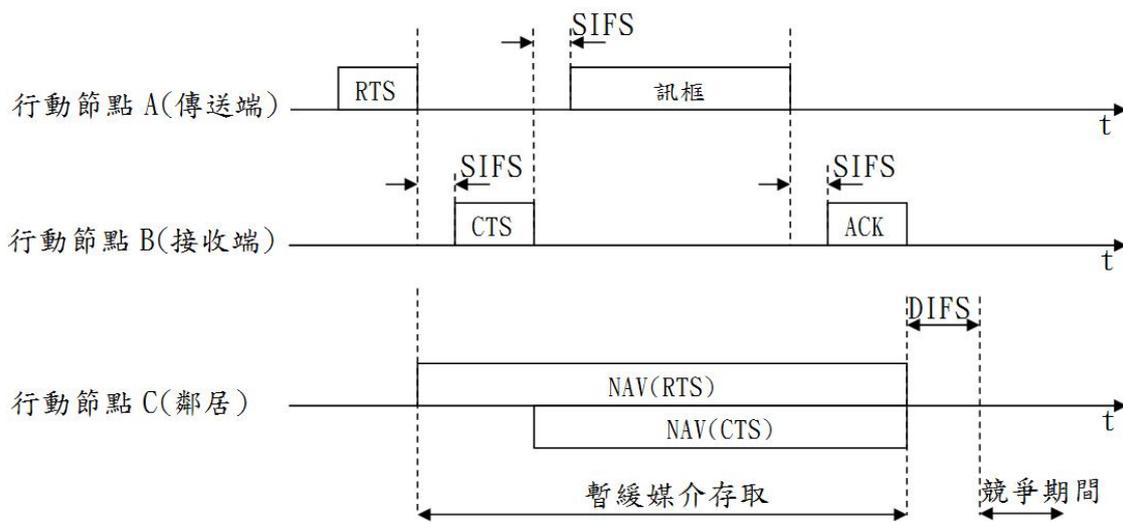


圖 2.3 虛擬載波偵測機制

2.2 網路通訊層 (Network Layer)

當我們要進行單一行動運算節點（來源節點）傳輸封包給在傳輸範圍之外的另一個行動節點（目標節點）時，藉由網路內其他的行動運算節點當作中繼節點幫忙傳遞來源節點的封包，間接地使封包可以成功的到達目標節點。從來源節點經過中繼點到目標節點之間所經過的行動運算節點就稱為路徑，而這些中繼節點就稱為擇路節點（Router）。在來源節點跟目標節點之間，通常會有很多條不同的路徑（所謂的“擇路”就是選擇最佳

路徑的意思)，並可以讓路徑上的行動運算節點都知道自己被當成是這一次傳輸的擇路節點。

一般來說，一個完整的擇路演算法會分成幾個部分，包括了擇路的搜尋與決定機制、擇路的維持機制與資料封包的傳輸，通常不同擇路演算法之間的差別都會在前二部分，所以之後的介紹也大都著重於這二個部分。

選擇最佳路徑的參數，也會隨著不同擇路演算法而不同，目前最常看到的考量參數就是路徑的跳躍數（從來源節點到目標節點的路徑上，傳遞同一個封包的次數），而利用這個考量參數所選出的路徑就稱為最短路徑（因為傳遞封包的擇路節點數最少）。由於目前的擇路演算法大都沒有針對特別的移動模式或傳輸環境模式作設計，而是假設所有的行動運算節點都在同一個固定的空間與不變的傳輸環境內自由移動，所以設計出來的演算法只考慮了路徑的跳躍數。在這篇論文中，不僅僅是考量的原本的路徑跳躍數，還另外利用多路徑方式考量每個行動運算節點能量大小的限制，針對多路徑能量的考量，使演算法的設計更加的貼切實際的狀況。

2.2.1 單一（Unicast）擇路協定

單一擇路協定歸納過去被提出的無線隨意網路擇路方法，可大致分成主動式擇路（Proactive Routing，或稱 Table-driven routing）跟被動式擇路（Reactive Routing，或稱 On-demand routing）。

主動式擇路（例如 DSDV[1]跟 WRP[2]）協定中，隨意無線網路下所

有行動節點都需收集其他行動節點的所在位址以及資料傳送過來的路徑資訊以建立擇路表單 (Routing Table)，並週期性的更新擇路表單裡的資訊以確保可以找到所要傳送目標節點。當有一行動節點想傳送資料封包時，此行動節點便可以依據擇路表單中所建立擇路路徑 (Routing Path) 來傳送資料封包給目標節點。

主動式擇路協定能保持一直找出最佳路徑，所有的行動節點在傳輸前已知道各行動節點的擇路路徑，因此不需要任何延遲就可以開始傳輸資料封包。但必須週期性的利用控制封包來維持擇路表單的正確性，故所需頻寬較大。

被動式擇路 (例如 AODV[3]、ABR[12]跟 SSA[13]) 協定中，隨意無線網路下所有行動節點不需要隨時的收集其他行動節點的所在位址，也不需要週期性的更新擇路表單，而是在有行動節點需要傳送資料封包時，將擇路請求 (Routing Request, RREQ) 的封包，以廣播方式傳給相鄰的行動節點，要求幫忙尋找目標節點，然後每個收到此封包的行動節點都會重複一樣的動作，直到行動節點收到或是尋找到目標節點才會停止。結束搜尋並決定路徑後，決定的路徑一樣會被存在擇路表單之內，然後才開始傳輸資料封包。

被動式擇路協定強調擇路的搜尋機制，只有當某行動節點想要傳送資料封包並發出擇路請求時，才會透過行動節點間的幫忙找出一條或是多條

可以連結來源節點與目標節點二點的傳遞路徑並開始傳遞資料封包。傳遞結束後，此擇路資訊也不會再被記錄儲存，更不會被維護。也因此只有在擇路請求發出時才會找尋擇路，其他的無線節點也是在收到擇路請求時發出廣播擇路資訊封包，因此所需頻寬較小。

有一些擇路演算法把主動式和被動式擇路整合成一混合式擇路(Hybrid Routing) 協定(例如 ZRP[14][15]、CGSR[16][17]跟 Cedar[18])，依照不同的設計觀念，適當的混合這二種擇路協定，其中 ZRP 對每個節點定義一個 Zone，即鄰近該節點 hop 數為 Zone 半徑範圍內的所有節點，而剛好距離為 Zone 半徑上的節點被稱為邊緣節點(Border node)，傳送封包時，在 Zone 內採用主動式擇路協定，Zone 外則由邊緣節點採用被動式擇路協定進行擇路探索程序，故 Zone 半徑的選取對其效率很重要，而 CGSR 則是使用叢集演算法，故叢集的複雜度(Complexity)會影響其效率，混和式擇路協定合成後的擇路協定可以同時保有各自的優點。目前也有很多研究專門比較上述這些擇路協定的優劣[19]-[22]。

由於全球衛星定位系統(GPS)[23]或其它間接的定位方法，使每個行動運算節點都能得知自己的座標。而目前也有很多擇路協定是假設知道自己的位置而進行擇路(例如 DREAM[24]、LAR[25]、GRA[26]、GPSR[27]、VAR[28]、PMLAR[29][30]跟 GLS[31][32])。此擇路協定不需要任何的擇路探索與維護，由於收集行動節點的位置資訊外，並不需要記錄目前的擇

路狀態以及互相交換擇路封包。因此，相對於其他擇路方法，利用位置資訊的擇路協定能更有效地適應頻繁的拓鋪改變，對於大型網路會有比較好的效能。

AODV (Ad Hoc On Demand Distance Vector Routing) 演算法是一種最常見的單一擇路演算法，擇路的流程也是最基本且簡單，所以常常被拿來當做研究的比較對象。因此在這一小節會介紹 AODV 演算法的基本擇路流程。AODV 屬於被動式協定，若有需要傳遞資料封包才會尋找擇路，並且在資料封包傳遞完成後結束一切擇路的維護，比較不同的是路徑維護的地方。在 AODV 裡，每個行動節點都需要維護一個類似傳統擇路表單，一條一條的路徑。而每個行動節點也清楚知道封包下一步該傳給哪個行動節點，並使用序列號碼 (sequence number) 來辨別此擇路資訊否為最新的擇路資訊。此外，每個行動節點也紀錄相鄰節點的位址。

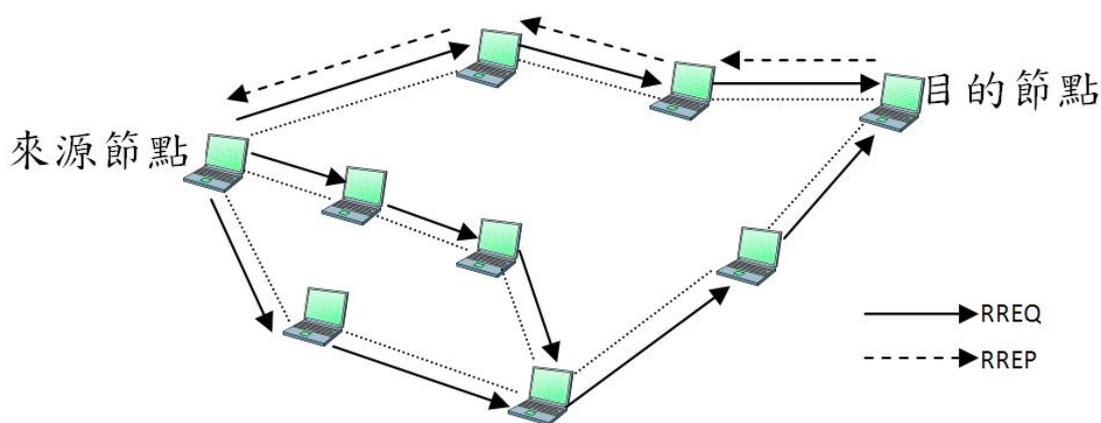


圖 2.4 AODV 找尋路徑過程

AODV 大致分為兩大部分：擇路探索 (Routing Discovery) 與擇路維護

(Routing Maintenance)。AODV 是藉由傳遞廣播擇路封包來達到擇路探索。當某個來源節點欲傳送資料至另一個目的節點，如果此來源節點在本身之擇路表單找不到可以到達該目的節點之路徑，或是此路徑資訊已經過期而被註記為無效時。這時來源節點將會廣播擇路請求(Routing Request, RREQ)封包給相鄰的行動節點(如圖 2.4)，當相鄰的行動節點收到 RREQ 封包時，會檢查目的節點是否是自己，如果不是，則在自己維護的擇路表單中建立一個逆向路徑 (Reverse Path) 的紀錄，指向上游裝置-也就是發送這個擇路請求封包的裝置，再繼續幫忙廣播此封包，直到目的節點收到此 RREQ 封包後才停止。而在 RREQ 封包方面，RREQ 封包並不會紀錄到達過的行動節點位址，而是只會紀錄到達過幾個行動節點，也就是記錄所謂的跳躍(Hop)數。

來源節點發出一個 RREQ 封包，都會有一個固定的廣播 ID，藉由這個廣播 ID，收到這個封包的行動節點可以知道之前是否有收過相同封包，若行動節點之前收到過此封包，也不會繼續廣播這個封包。當網路內的任何一個行動節點第一次收到 RREQ 封包，代表這個節點有路徑可以通往來源節點，當作之後回傳 RREP 封包的保留路徑。

當目的節點收到 RREQ 封包時，便會發送擇路回覆 (Routing Reply, RREP) 封包回來源節點，傳送的方法是利用逆向路徑紀錄所建立的路徑，而 RREP 封包包含整個路徑的跳躍數以及目的節點的序列號碼。當 RREP

封包回傳時，中間幫忙傳遞的行動節點便會在自己維護的擇路表單中建立或更新正向路徑（Forward Path）並紀錄目的節點的序列號碼，此正向路路徑即是來源節點的擇路路徑，其方向會剛好與逆向路徑相反。其他不在正向路徑上的節點，如果因為先前也收到路徑請求封包而建立了逆向路徑，則會因為在設定的等待時間內未收到擇路回覆封包而被節點清除。當來源節點收到目的節點所發送的擇路回覆封包後，便完成了擇路建立的程序。另外，因為來源節點與目的節點間可能存在多條擇路路徑，AODV 在擇路請求封中設置一個跳躍計數（Hop Count）欄位，此欄位值會隨者傳送次數的遞增，最後擇路回覆封包會將此值回傳給來源節點。當來源節點陸續收到不同路徑傳回的擇路回覆封包時，除了可儘快使用第一個回覆的擇路路徑先行傳送資料外，後續也可藉由其他擇路回覆封包上的結點計數值決定是否更新擇路路徑。

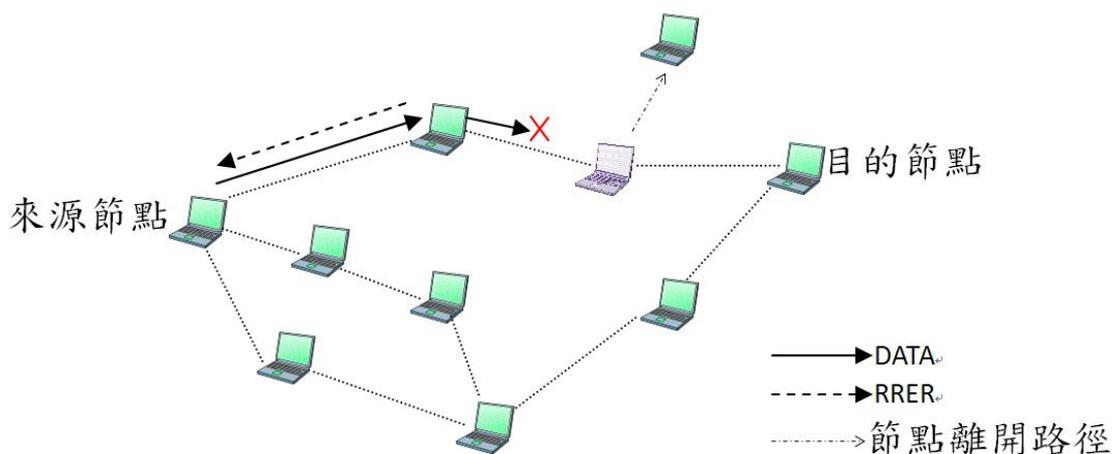


圖 2.5 路徑斷裂

而 AODV 在擇路維護（Routing Maintenance）的部分是經由發送週期

性的廣播 Hello 訊息給相鄰裝置來偵測此一連結是否失敗，或是藉由使用資料鏈節層的確認訊息（Data Link Layer Acknowledgement）來偵測下一節點是否失效。此時就會由該行動節點發出擇路錯誤（Routing Error, RERR）封包告知路徑中的所有行動節點以及來源節點此傳遞路徑已經失效，請將此失效擇路從擇路表單中刪除，並由來源端來決定是否要重新送出擇路請求封包來建立新的擇路（如圖 2.5）。

2.2.2 單一多路徑擇路協定

在 On-Demand 型式的繞徑演算法基礎下，多路徑或備份路徑可以使用多種不同方式來建立。目前現有多路徑演算法有 DSR[4]、TORA[5][6]、ROMA[33]、AODV-BR[34]及 AOMDV[35]。

DSR 是由來源節點進行主動尋路的擇路演算法，故任何一個來源節點皆可透過對自身擇路快取的維護，或是藉由擇路探知而動態產生路徑，並以此決定使用較合適之路徑，故任一節點能同時擁有多條路徑。TORA 為針對高度動態網路環境所發展出來的擇路演算法，本身為某一來源節點至某一目的節點提供多重路徑，路徑的建立是由來源節點在找尋路徑的過程中，快速建立一系列的多重路徑。ROMA 類似於 TORA 採用空間分隔多重存取技術（Space Division Multiple Access, SDMA），故只有一部分節點會收到欲建立路徑的控制封包，額外的負擔為兩點之間的參數必須密切調整，比較適合靜態或低移動性網路。AODV-BR（Backup Route）藉建置網狀

(Mesh) 的備用路徑來改善繞境協定的效能。此一演算法在無增加其它控制訊號負載的前提下，使用原本 AODV 當中的 RREP 控制訊號來建置網狀的多路徑。

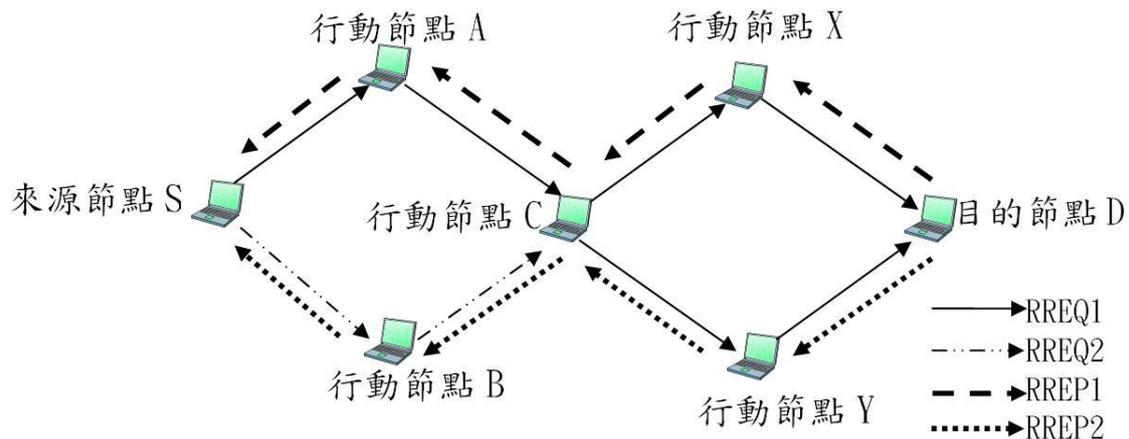


圖 2.6 AOMDV 路徑建立過程

在 AOMDV (Ad hoc On-Demand Multi-path Distance Vector) 中，利用了 AODV 協定中的 RREQs (Routing Requests) 控制訊號來建置多路徑 (如圖 2.6)。RREQs 是一種廣播型式的控制訊號，所以建置出來的多路徑將會離主要路徑較遠，其多路徑的建立問題會較難去處理，這將會增加管理這些多路徑的困難度。另外 RREQs 來建立多路徑或備份路徑需要一些機制去確保這些多路徑是連結分支 (Link-Disjoint) 以及不會產生回路的問題。

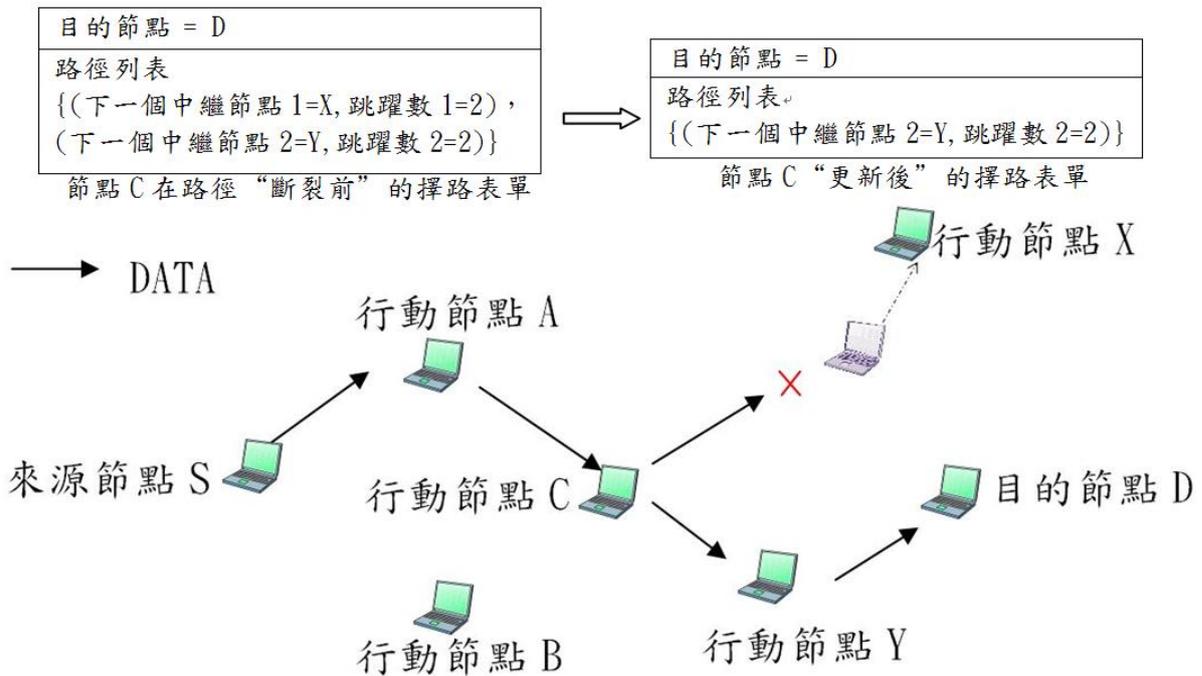


圖 2.7 AOMDV 路徑斷裂時處理

AOMDV 本身具備多路徑擇路的優點故當連線斷裂時（如圖 2.7），當一行動節點傳送資料的過程中，偵測到下一節點失效時，可以立即由擇路表單內看是否有較佳的備份路徑，若有則資料將走其較佳備份路徑，若無則廣播一個 RRER 至上一個行動節點，告知此路徑斷裂，並由擇路表單內尋找其較佳路徑，重複此方法，故 AOMDV 控制封包在建立的初期會比 AODV 還要多，但當在行動速度較快的網路環境下反而會產生較少的控制封包。

AOMDV 的擇路表單內的公佈跳躍數（Advertised Hop Count）取代 AODV 的跳躍數（Hop Count）欄位，擇路列表（Routing List）取代下一個中繼節點（Next Hop）欄位，用來建立多路徑。

公布跳躍數表示此中繼的行動節點到目的節點時，所能接受最大跳躍

數，當收到 RREQs 封包時會有兩種建立路徑情況發生，第一種若收到較大的序列號碼，會將公佈跳躍數變成無窮大，並且在擇路列表的路徑列表內插入一個欄位。第二種若收到相同的序列號碼，則判斷收到的 RREQ 欄位內的跳躍數是否小於等於擇路表單的公佈跳躍數，若是，則在擇路表單內擇路列表插入一個欄位。若是發生第一種情況且行動節點不是目的節點的情形，表示收到此行動節點為路徑上的中繼節點，廣播此 RREQs 時，則更新公佈跳躍數為路徑列表所有路徑的最大的跳躍數，而公佈跳躍數將可以保證陸續收到的 RREQs 不會有迴路發生的情況，這表示每個中繼的行動節點只發送一次 RREQ，並同時決定公布跳躍數。

AOMDV 允許中繼節點有兩種分支路徑的型態：節點分支 (Node-Disjoint) 及連結分支 (Link-Disjoint)，節點分支路徑表示除了來源節點與目的節點外，沒有任何共用行動結點，表示若有兩條資料傳送的路徑，除了來源節點及目的節點，兩條資料傳送路徑內的行動節點不可能相同。連節分支路徑沒有共用的連節，表示若有兩條資料傳送路徑，可能會有共同的行動節點但不會存在相同的邊線 (Edge)，邊線表示兩個行動節點之間的連線。然而節點分支路徑數目比連結分支路徑數目還要嚴格的情況才會產生且在實際情況下會有較少的分支路徑，所以 AOMDV 使用連結分支路徑型態建立多路徑。但若使用連結分支路徑，當目的節點回傳 RREP 到分支節點時，會無法判斷要走哪一個逆向路徑的問題，所以必須在

RREQ 欄位加上第一個跳躍點 (First hop) 欄位，每個行動節點需要加上第一個跳躍點列表 (Firsthop List)，用來決定連結分支路徑的建立的先後順序，當回傳 RREP 時，會依據跳躍點列表內先後順序回傳至來源節點。AOMDV 與 AODV 有著一樣的特性，AOMDV 路徑建立的優點在於當路徑的搜尋機制存在於當所有的路徑斷裂後才會重新搜尋。

2.2.3 能量相關的擇路演算法

無線隨意網路的行動節點在一個有限的電池能量下操作。有效地使用能量及降低功率的消耗將是非常重要的問題。為了能加大隨意網路的存活時間。經由減少傳送能量的消耗以延長每個個別行動節點的存活時間，將是必要的。從某些相關的研究發現，行動節點除了將部分的 power 消耗在資料傳送 (Transmit)、接收 (Receive) 以及待機狀態 (Standby) 外，對於執行這些動作的前置作業如 RTS/CTS 等動作，也會消耗掉一定程度的能量[36][37][38][39]。因此，有不少學者以減低功率的消耗以及增進能量的持續力為主要考量，而發展出能量方面 (功率方面) 的擇路協定；將擇路協定在資訊維護、資料傳送時的 Energy 列入設計之考量，以達到更好的能量使用率與持續力。以下我們簡介一些現有的能量方面擇路協定。由於電池的能量與功率有很大的關係，故以下將介紹幾個與能量或功率相關的演算法：MTPR (Minimum Total Transmission Power Routing) [40]，此方法是藉由 DSR[4]，增加對於功率部分的改良而來，在傳送資料前，計算路徑上每

一點所需消耗的功率量之總和；也就是說，MTPR 在找尋路徑時，發送 RREQ 後，中繼的行動節點在收到 RREQ 時，便計算自己傳送此資料所需消耗之功率，將此消耗的功率加到 RREQ 後再將 RREQ 廣播出去直到目的節點，目的節點會等待一段時間以接收數個來源節點經由不同中繼行動節點傳過來的 RREQ，接著再從這些 RREQ 中選出整條路徑所需消耗的功率最低者，回覆 RREP 來建立此路徑，完成路徑建立的動作。REDMER (Red Based Minimum Energy Routing) [41]，選擇路徑的方式考量最小能量消耗及 MAC buffer 長度，若網路中無任何資料傳送，則尋找最小路徑傳送，路徑中的行動節點能量不足時，停止接收資料，並重新找尋最短路徑（此時並不包含能量不足的節點），若網路中有多條資料流，則使用 MAC buffer 用來區分節點壅塞 (Congestion) 的狀況，壅塞的等級將依據已知的 AQM (Active Queue Management) 來判斷下個節點的傳送資料的情況，較大則表示有較多的資料傳送，盡可能走 MAC buffer 較短的路徑，但不一定是最短路徑，以平衡網路整個網路能量的消耗量。

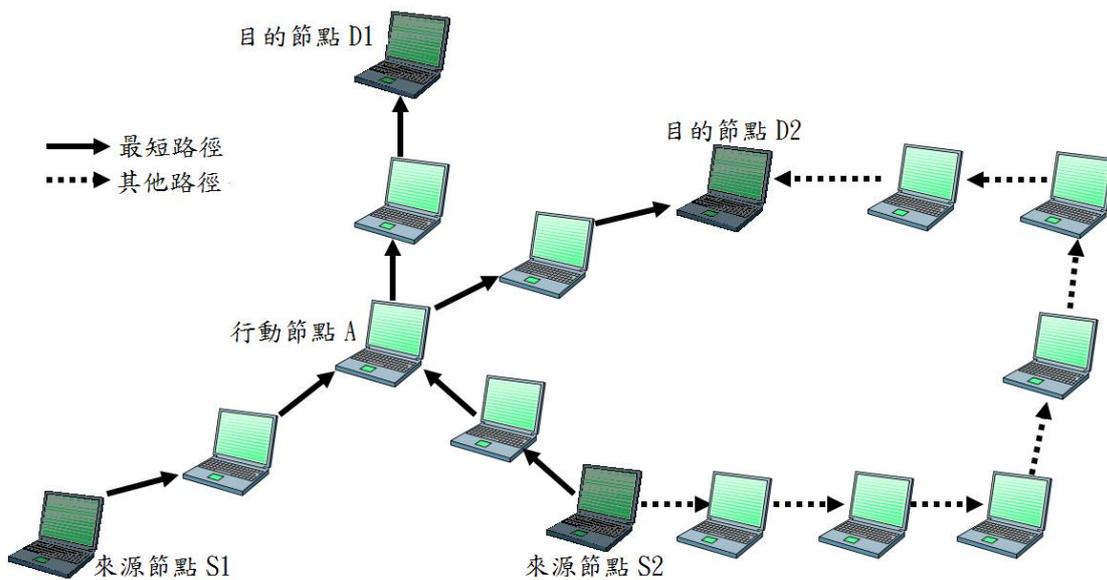


圖 2.8 考慮電池能量的路徑選擇方式

分散式擇路演算法 (Distributed Energy Routing Protocol, DER) [42], 以電池能量而設計的單一擇路演算法。其運作機制 (見圖 2.8), 行動節點 A 因為處在網路中的重要位置而容易擔任擇路節點的工作, 因此電池能量會消耗較快。假設來源節點 S1 正傳送資料封包給目的節點 D1。此時來源節點 S2 也想傳資料封包給目的節點 D2, 依照網路中所有行動運算節點佈點與只考慮跳躍數的單一擇路演算法, 來源節點 S2 會選擇最短路徑 (見圖 2.8 的實線箭頭), 而中繼節點 A、B、C、D 會同時間成為這條路徑與其他路徑的共用擇路節點, 必須承擔比較多次的封包傳遞動作, 很容易造成節點 A 的電池能量使用完而不能幫忙傳遞資料封包的工作, 因此嚴重影響封包傳遞成功率。若單一擇路演算法有考慮到電池能量的因素, 那來源節點 S2 就可能捨棄最短路徑而選擇電池能量較大的路徑 (見圖 2.8 的虛線箭頭) 將傳送資料封包給目的節點 D2, 將可避免節點 A 被重複使用, 造成電池能

量快速耗損。

因為 AODV 演算法只考慮跳躍數的情況，故在剩餘能量的限制下，效能將會大大的下降。DER 演算法則是以 AODV 演算法為基礎，加上剩餘能量的因素改良其演算法。

DER 演算法跟 AODV 演算法類似，故針對 AODV 演算法做了三點改良。第一、剩餘電池能量的考量方式。第二、最佳路徑的選擇。第三、反應計時器的設計。主要是在 RREQ 封包內，增加電池能量最小值的欄位。當 RREQ 傳送到中繼的行動節點後，會去比較電池能量是否小於 RREQ 的路徑電池能量，若是，則會把電池能量填入 RREQ 最小值欄位。當行動節點收到 RREQ 會去比較擇路表單的路徑剩餘能量，用此決定是否繼續廣播 RREQ，目的節點可以透過這個資訊來選擇最大的電池能量最小值的路徑，故需要反應計時器，去收集 RREQ 封包，以得到最佳路徑。

三、提升存活時間之適應性擇路演算法

(Lifetime-Enhanced Adaptive Routing Protocol - LEAR)

3.1 設計考量

正如第三章所提到，DER 使用路徑剩餘電池能量資訊設法讓每條路徑的選擇可以平均地分部在網路中，以避免重要行動運算節點不斷地被選擇當作擇路節點而造成電池能量快速地被使用完的問題，但必須在目的節點設定反應計時器，使目的節點可以得到每一條路徑的資訊並更新成最佳路徑，資料傳送的路徑也因為捨棄最短路徑而選擇擁有較大電池能量的路徑，雖然可以增加網路的存活時間及資料封包送達率提升，卻犧牲資料封包平均延遲加大及造成整個網路消耗的能量較大的缺點。DER 本身使用擇路表單內電池能量最小值的欄位來決定路徑的機制，行動運算節點本身具有移動性及電池能量減少的特性，很容易發生路徑的連節斷裂及行動節點電池能量用完的問題。AOMDV 演算法使用 RREQ 控制封包建立多路徑，公佈跳躍數來避免迴路及建立多條路徑到達目的節點，以最短路徑來傳送資料，但以不考慮能量的狀況下，很容易造成重要節點能量消耗完畢。若要延長 AOMDV 的存活時間 (Lifetime)，則必須考量到達目的節點的每條路徑上的剩餘能量，因此若備份路徑擁有較多的剩餘能量，則行動運算節點在傳

送資料的過程中可以選擇剩餘能量較多路徑做傳送，避免重新找尋路徑缺點。LEAR 演算法便是為了能延長 AOMDV 演算法存活時間中及改善 DER 缺點，考慮電池能量而設計的單一多路徑擇路演算法。

3.2 提升存活時間之適應性擇路演算法設計

因為 AOMDV 演算法假設電池能量無限制而只有考慮到跳躍數來建立多路徑，所以在剩餘電池能量限制下，重要的行動運算節點能量很快速的消耗完畢，導致網路節點存活時間較少。LEAR 演算法是以 AOMDV 演算法為基礎，及考量剩餘電池能量的因素改良演算法，達到 3.1 節敘述的概念。



LEAR 演算法跟 AOMDV 演算法類似，能量傳送的方式與 DER 演算法類似，將路徑電池能量最小值放入 RREQ 欄位，路徑的建立是使用接收到的要求封包(Route Request - RREQ)與擇路回應封包 (Route Reply - RREP) 的往返來完成，當行動運節點接收到兩個以上的 RREP，則表示此擇路表單到目的節點有兩個以上的路徑。當擇路表單擁有兩個以上的路徑可供選擇，行動運算節點傳遞資料的過程中，便可以走能量較大的路徑，當能量不足時，本身會發送一個控制封包將此節點的電池能量告知上一個行動運算節點。收到此控制封包的行動運算節點，知道此節點為表單上某路徑的下一個跳躍點，則更新此擇路表單上剩餘能量欄位的資訊，由於資料傳送的過程是選擇以能量較大的路徑傳送資料，避免能量不足的行動運算節點能量

快速消耗，將可延長整個網路的存活時間。

3.2.1 LEAR 演算法擇路表單設計

LEAR 演算法是根據 AOMDV 多路徑及路徑剩餘能量資訊做資料傳送，當資料封包欲傳送資料時，則會根據現有已建立的擇路表單內的擇路列表欄位的路徑能量（如表 3.1 所示）當做資料傳送，藉由接收到能量不足的行動節點更新此擇路表單的路徑剩餘能量。

擇路表單欄位架構

目的節點	序列號碼	通知跳躍數	擇路列表	終止時間
------	------	-------	------	------

擇路列表架構

跳躍數 1	下一個跳躍點 1	第一個跳躍點 1	路徑剩餘能量 1
跳躍數 2	下一個跳躍點 2	第一個跳躍點 2	路徑剩餘能量 2

表 3.1 擇路表單與擇路列表

3.2.2 LEAR 演算法的運作原理

LEAR 擇路演算法根據剩餘路徑能量方式在擇路表單內建立多路徑的過程，下表以路徑建立的過程來詳細描述擇路列表的資訊。

封包類型 Type	保留 Reversed	第一個跳躍 First hop	跳躍數 Hop count
擇路廣播 ID RREQ ID			
目的節點 IP 位址 Destination IP Address			
目的節點之序列號碼 Destination Sequence Number			
來源節點之 IP 位址 Originator IP Address			
來源節點之序列號碼 Originator Sequence Number			
路徑剩餘能量 Path energy remained			

表 3.2 RREQ 的格式

A、建立逆向路徑

當有一個欲傳送資料的來源節點需要一條路徑到目的節點且沒有路徑是可用的，這個來源節點會開始整個路徑搜尋的程序由廣播一個擇路封包 RREQ（見表 3.2），同時將來源節點的能量填入 RREQ 的路徑剩餘能量。當來源節點的任何鄰近的行動節點收到這個 RREQ，會優先將本身的位址填入 RREQ 的“第一個跳躍點（First Hop）”欄位，這個功能主要是為了在連結分支（Link-Disjoint）的行動節點建立一個逆向路徑，然後經由 RREQ 更新行動節點的擇路表單（見表 3.1）並形成一個逆向路徑。最後這個行動節點將繼續使用 RREQ，並做一些修改，例如增加 RREQ 的“跳躍數（Hop Count）”及廣播它。

一個中繼行動節點在相同的擇路表單只接收 RREQ 有不同的“第一個跳躍點”，及 RREQ 的“跳躍數”不能大於“公佈跳躍數（Advertised Hop Count）”並將 RREQ 的路徑剩餘能量填入擇路列表的路徑剩餘能量。這些中繼行動節點收到很多 RREQs，會根據上述規則，在“擇路列表”加入及建立有多條逆向路徑，但中繼行動節點只廣播第一個收到的 RREQ，當中繼行動節點廣播 RREQ 時，會將 RREQ 的路徑剩餘能量及此行動節點的能量兩個值比較，將較小的值填入 RREQ 的路徑剩餘能量。

當目的節點將接受所有的 RREQs，只會對有不同的“第一個跳躍點”建立逆向路徑，並將 RREQ 的路徑剩餘能量填入擇路列表的路徑剩餘能量

欄位。

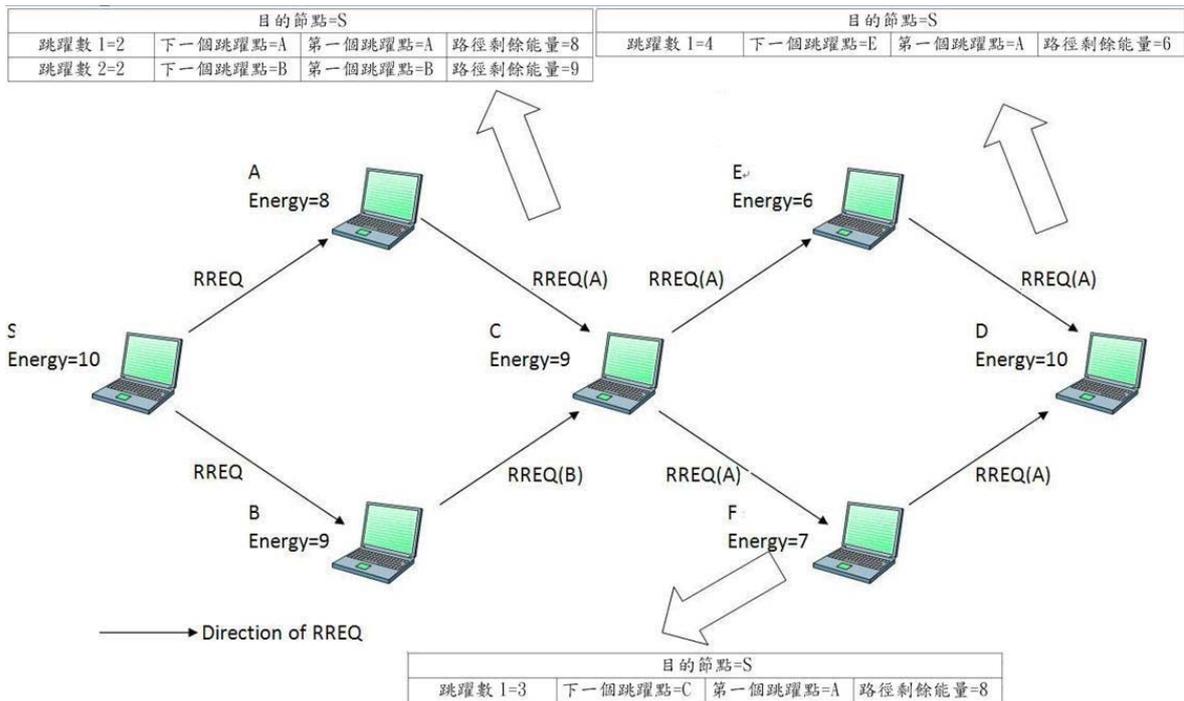


圖 3.1 建立逆向路徑程序

如圖 3.1，假如來源節點有需要將廣播一個在它的傳送範圍的 RREQ，並將來源節點的能量填入 RREQ 的路徑剩餘能量。當行動節點 A 及行動節點 B 收到這個 RREQ，它們優先加入它們各自的位址到 RREQs 當各自的第一個跳躍點。在這篇論文，RREQ 經由行動節點 A 被標記成 RREQ(A)，及這 RREQ 經由行動節點 B 被標記成 RREQ(B)，此標記的目的是個別表示第一個跳躍點。然後行動節點 A 及行動節點 B 更新各自的擇路表單的資訊去形成逆向路徑 (A-S 及 B-S)，並將 RREQ 欄位的路徑剩餘能量填入擇路列表的路徑剩餘能量，最後行動節點 A 廣播 RREQ(A) 及行動節點 B 廣播 RREQ(B)，在將 RREQ 的路徑剩餘能量及行動節點本身能量兩個值比較，將較小的能量值填入 RREQ 的路徑剩餘能量。行動節點 C 將形成兩

個連結分支返回路徑 (C-A-S) 及 (C-B-S)。若行動節點 C 有到達目的節點的可用路徑，行動節點 C 將回覆一個 RREP 到來源節點 S。若沒有，行動節點 C 將廣播第一個收到 RREQ(A) 到其他的行動節點 (假設第一個被行動節點 C 收到的控制封包是 RREQ (A))，同時將 RREQ 的路徑剩餘能量及行動節點 C 的能量兩個值比較，將較小的能量值填入 RREQ 的路徑剩餘能量。當行動節點 E 和行動節點 F 收到 RREQ (A)，它們將各自第形成返回路徑 (E-C-A-S 及 F-C-A-S)，並重複之前 RREQ (A) 將路徑剩餘能量填入擇路列表，及 RREQ(A) 與本身行動節點路徑剩餘能量的比較步驟。最後，目的節點 D 成功地收到 RREQ (A) 由行動節點 E 及行動節點 F。假如行動節點首先由行動節點 E 收到 RREQ(A)，一個返回路徑(D-E-C-A-S) 將被建立。當他收到由行動節點 F 來的第二個 RREQ (A)，其包含相同的“第一個跳躍點 (First hop)”，目的地節點將不會建立一個非分支返回路徑 (D-F-C-A-S)。

封包類型 Type	保留 Reversed	第一個跳躍 First hop	跳躍數 Hop count
目的節點 IP 位址		Destination IP Address	
目的節點序列號碼		Destination Sequence Number	
發出 RREP 節點 IP 位址		Originator IP Address	
路徑剩餘能量		Path energy remained	
存活時間		Lifetime	

表 3.3 RREP 的格式

B、建立正向路徑

當目的節點收到一個 RREQ，他將用一個擇路回覆封包 RREP (見表

3.3) ，即使先前收到相同“第一個跳躍點”，一樣回覆一個 RREP 封包，並把目的節點的電池能量填入 RREP 的路徑剩餘能量。當目的節點的鄰居收到 RREP 時，會先填入此行動節點的位址到 RREP 的第一個跳躍點欄位，這個功能是為了能在連結分支的行動節點建立一個正向路徑。然後此行動節點形成一個正向路徑到目的節點，接著此行動節點再傳送這個 RREP 到擇路表單的下一個跳躍點，當中繼行動節點傳送 RREP 時，會將 RREP 的路徑剩餘能量及行動節點能量兩個值比較，將較小的值填入 RREP 的路徑剩餘能量。在擇路表單的目的節點為 RREQ 的來源節點（或傳送資料的來源節點）。接著收到這個 RREP 的下一個跳躍點將重複前面的幾個建立正向路徑的步驟。當目的節點（或傳送資料的來源節點）收到 RREP 時，目的節點將用此 RREP 只建立正向路徑，若從其他節點收到的 RREP，則有另一個正向路徑的下一個跳躍點。

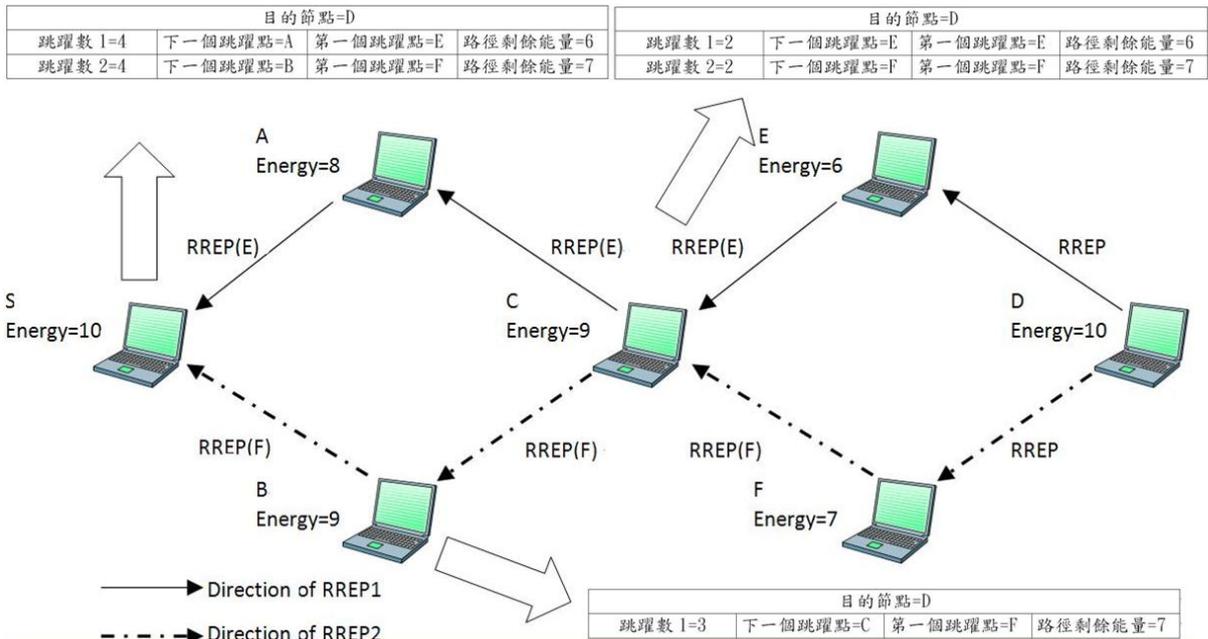


圖 3.2 RREP 建立正向路徑的程序

首先由目的節點 D 收到行動節點 E 傳送來的 RREQ(A) 後(見圖 3.2)，目的節點 D 則會回傳一個 RREP 給行動節點 E，同時目的節點 D 的電池能量會填入 RREP 的路徑剩餘能量。當行動節點 E 收到這個 RREP 時，會先填入本身位址到 RREP 的第一個跳躍點欄位，為了能在連結分支的行動節點建立正向路徑。在這篇論文，這個 RREP 經由行動節點 E 被標記成 RREP (E)。然後，行動節點 E 會形成一個正向路徑 (E-D)，接著此行動節點再傳送這個 RREP (E) 到擇路表單的“下一個跳躍點”，其中擇路表單的“目的節點”是來源節點 S。當一個經由行動節點 E 的逆向路徑被建立時，表示行動節點 E 知道下一個跳躍點是行動節點 C，將 RREP (E) 的路徑剩餘能量及行動節點 E 能量兩個值比較，將較小的值填入 RREP (E) 的路徑剩餘能量。那麼行動節點 E 將傳送 RREP (E) 到行動節點 C。行動節點 C

將形成一個正向路徑 (C-E-D)，接著再傳送 RREP (E) 到行動節點 C 的下一個跳躍點 (下一個跳躍點等於 A)，同時將 RREP (E) 的路徑剩餘能量及行動節點 C 能量兩個值比較，將較小的值填入 RREP (E) 的路徑剩餘能量。每一個逆向路徑的行動節點將收到 RREP (E) 將形成一個正向路徑。當來源節點 S 收到 RREP (E)，將形成一個正向路徑 (S-A-C-E-D)。這來源節點可以立即沿著正向路徑送資料封包，而不用等待其他 RREP 到來。

不久之後，目的節點 D 收到第二個經由行動節點 F 傳送過來的 RREQ (A) 後，行動節點 D 會回傳一個 RREP 給行動節點 F，同時目的節點 D 的電池能量會填入 RREP 的路徑剩餘能量。當行動節點 F 收到這個 RREP 時，會先填入本身的位址至 RREP 的第一個跳躍點欄位，這個 RREP 經由節點 F 被標記成 RREP (F)。然後，行動節點 F 會形成一個正向路徑 (F-D)，此行動節點再傳送這個 RREP (F) 到擇路表單的“下一個跳躍點”，行動節點 F 從擇路表單知道下一個跳躍點是 C，並將 RREP (E) 的路徑剩餘能量及行動節點 E 能量兩個值比較，將較小的值填入 RREP (E) 的路徑剩餘能量。當行動節點 C 收到 RREP (F)，將會形成一個正向路徑 (C-F-D) 接著尋找下一個跳躍點。行動節點 C 有兩個連結分支的逆向路徑 (C-A-S 及 C-B-S)，其中一條逆向路徑 (C-A-S)，因為傳送先前的 RREP (E) 而被用過，所以這裡只能使用另一條逆向路徑。行動節點的擇路表單的下一個跳躍點 2 是行動節點 B。接著行動節點 C 將傳送這個 RREP (F) 到 B，

並將 RREP (F) 的路徑剩餘能量及行動節點 C 能量兩個值比較，將較小的值填入 RREP (F) 的路徑剩餘能量。最後另一個正向路徑(S-B-C-F-D)將會被建立，當這 RREP (F) 的“第一個跳躍點 (First hop)”不同於之前 RREP (E)。注意兩個正向路徑是連結分支的。

3.2.3 資料傳送的過程

當來源節點收到第一個 RREP，直接傳送資料封包，資料封包會選擇路列表中的較大能量的路徑，資料傳送的過程中，若中繼節點或來源節點陸續收到第二個 RREP 封包後會增加擇路列表內的路徑數並且把路徑上的剩餘能量填入擇入列表內，當擇路列表路徑有一較大能量的值，資料封包會變更路徑。行動節點的電池能量在傳送資料的過程中會一直消耗電池能量，且行動節點在傳送資料時電池能量會消耗比較多，當行動節點能量消耗到剩下自己能量的一半時，LEAR 會設定一個電池能量的臨界值 (Energy threshold)，臨界值得初始值為總能量的一半，發送一個控制封包後，並修改此臨界值為原來的一半，通知正在傳送資料封包的上一個行動節點，此控制封包為一個跳躍數的 RREP 封包，控制封包內含有行動節點能量的資訊，正在傳送資料的上一個行動節點收到此控制封包後，會更新擇路列表內的路徑剩餘能量，由於變更後的能量會越來越小，資料封包就會選擇較大能量的路徑列表傳送。

當來源節點 S 收到第一個 RREP (E) 封包 (見圖 3.3) 後，會立即沿

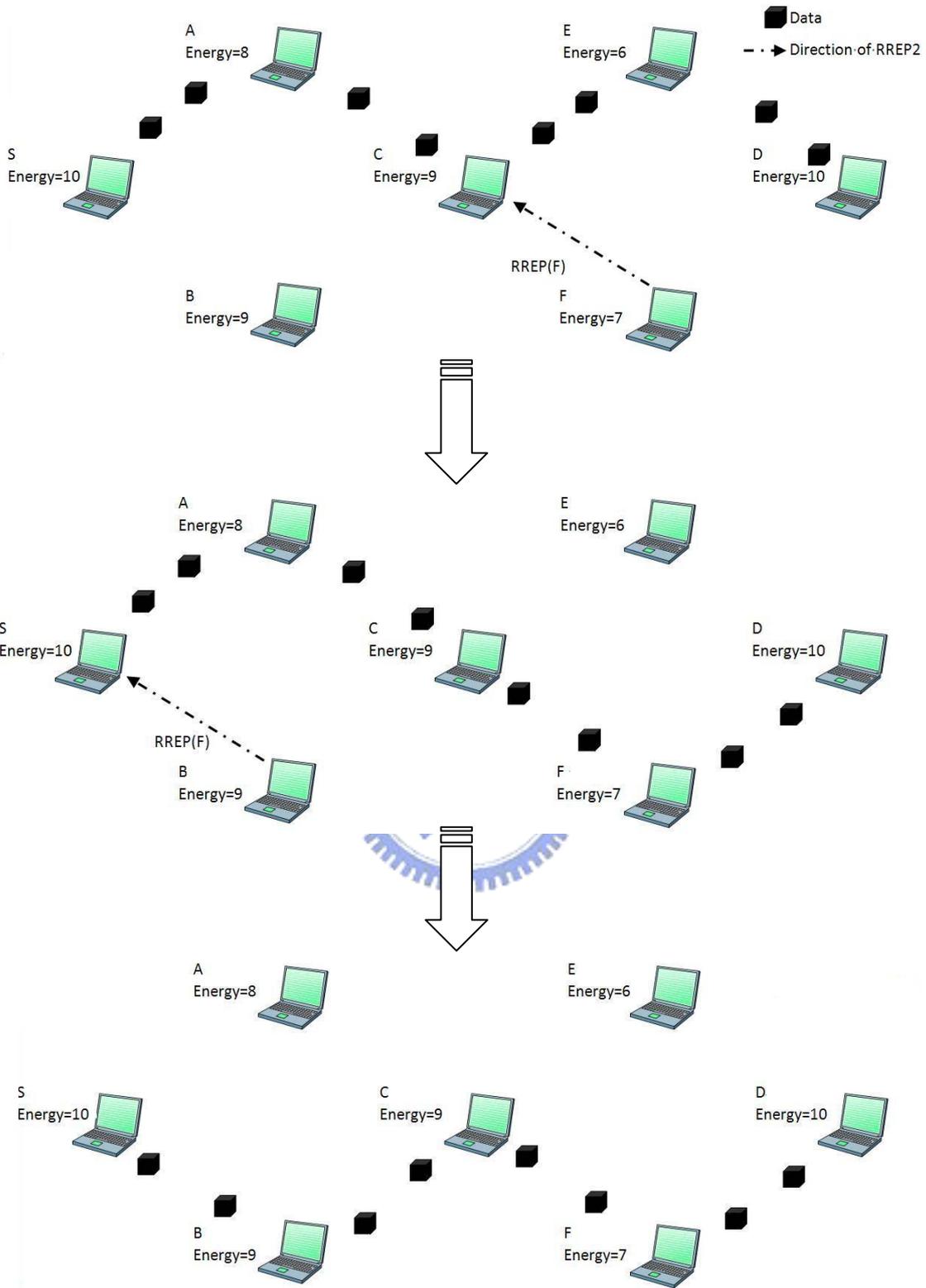


圖 3.3 資料封包選擇能量最大路徑過程

著決定好了路徑 (S-A-C-E-D) 傳送資料封包，當中繼的行動節點 C 收到行動節點 F 傳送過來的 RREP (F)，更新擇路列表內的路徑剩餘能量後，傳

送的資料發現有另一條較大的剩餘能量路徑（C-F-D），傳送的路徑（S-A-C-E-D）會更換成新的路徑（S-A-C-F-D），在傳送資料封包一段時間後，來源節點 S 又收到行動節點 B 傳送過來的 RREP（F），更新擇路表單後知道另一條較大的剩餘能量路徑，原本傳送的路徑（S-A-C-F-D）會變更成新的路徑（S-B-C-F-D）。

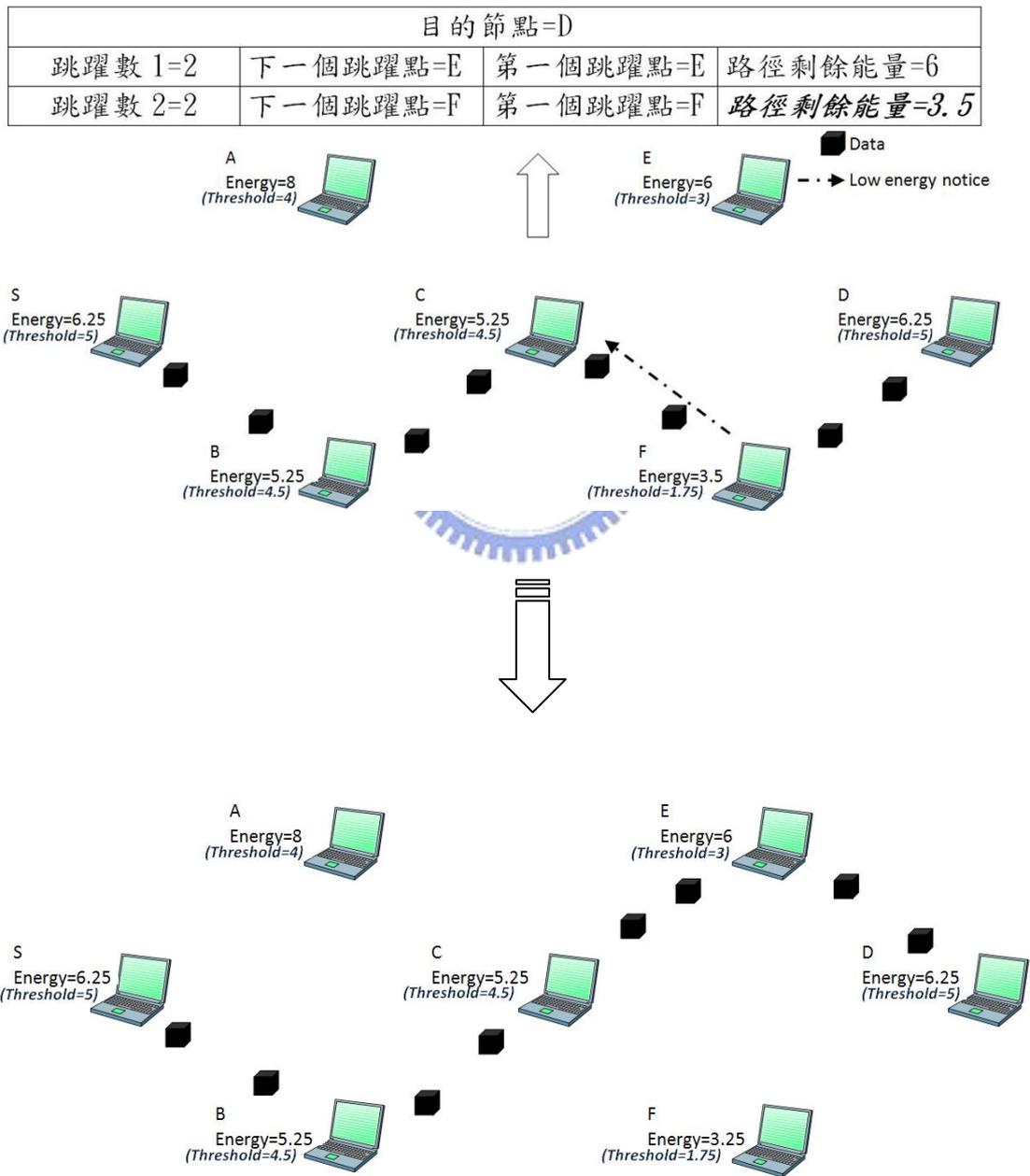


圖 3.4 行動節點 F 寄送低能量控制封包

資料傳送的過程中，行動節點的能量會一直消耗，故必須不斷的更新路徑剩餘能量欄位的資訊，才能延長行動節點在網路的存活時間，將表示行動節點剩餘能量不足時(見圖 3.4 及圖 3.5)，作為資料切換路徑的過程，

目的節點=D			
跳躍數 1=4	下一個跳躍點=A	第一個跳躍點=E	路徑剩餘能量=6
跳躍數 2=4	下一個跳躍點=B	第一個跳躍點=F	路徑剩餘能量=4.5

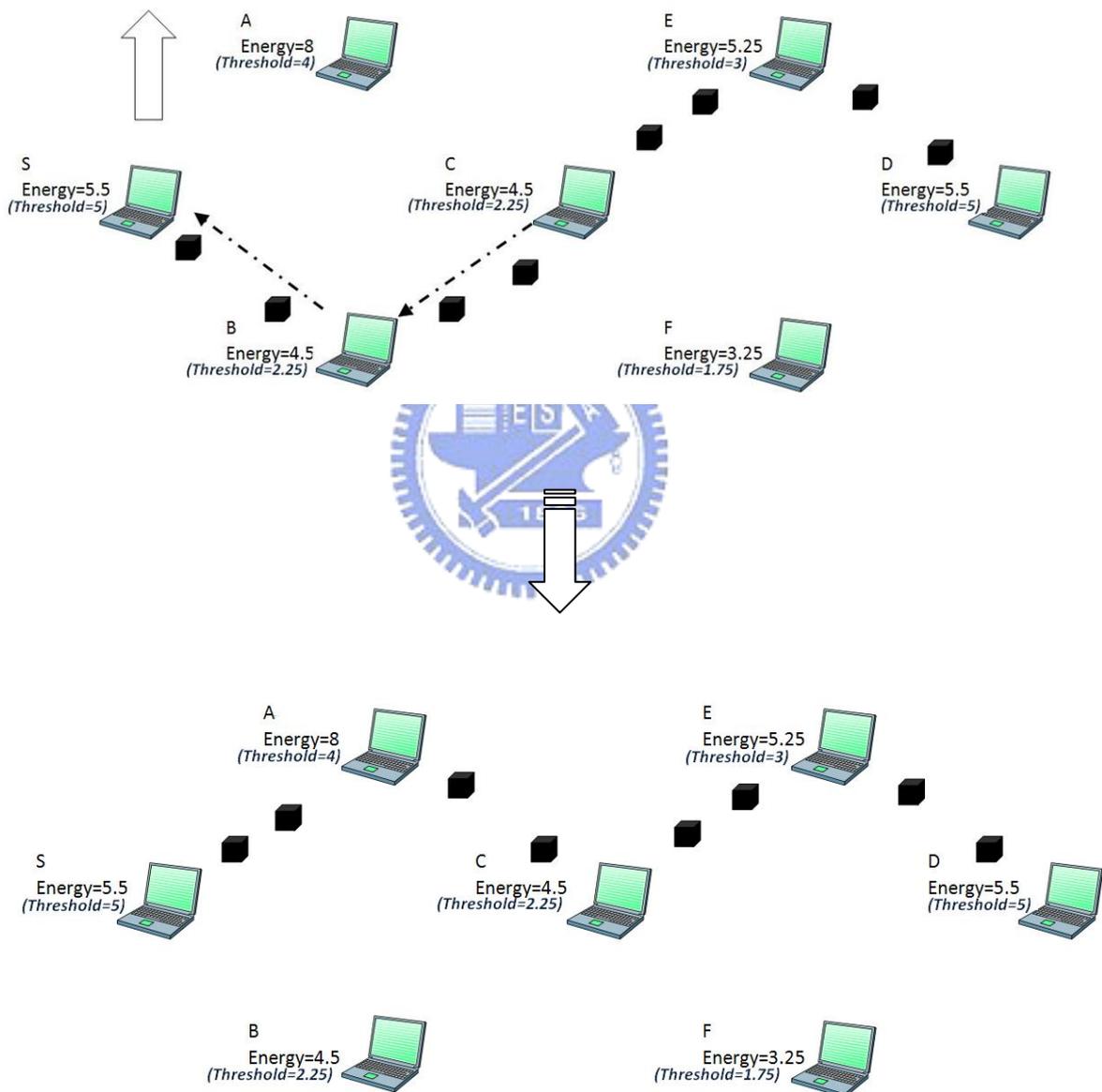


圖 3.5 行動節點 B 及行動節點 C 寄送低能量控制封包

假設現在資料傳送的路徑 (S-B-C-F-D)，為目前最佳路徑，資料傳輸的過

程中，整個網路的電池能量會下降，但若行動運算節點擔任資料傳輸的角色，則電池能量會消耗得更快，經過一段時間後，行動節點 F 的電池能量下降剩下 3.5 焦耳，此時 F 將發送一個控制封包，行動節點 C 收到此控制封包，更新路徑列表的剩餘能量欄位，發現有比原本電池剩餘能量還要大的路徑，此時資料傳送的路徑(S-B-C-F-D)會更換成一新路徑(S-B-C-E-D)，再經過一段時間後，行動節點 B 的能量降為原本的一半，此時行動節點便會在發送一個控制封包給行動節點 S，行動節點 S，更新擇路列表後，發現另一條路徑有較大的能量便更路徑(S-B-C-E-D)為一新路徑(S-A-C-E-D)，如此便可延長整體系統的存活時間。

若在一複雜網路，行動運算節點若只考慮跳躍數的情況下很容易產生資料壅塞 (Congestion) 及 RTS-CTS 競爭 (Contention) 所產生的延遲問題，藉由路徑能量的資訊，將資料封包選擇一條較大能量的路徑，同時表示此路徑有較少資料傳送。故 LEAR 除了能延長網路的存活時間，同時在複雜的網路環境對於資料封包的延遲也會有所改善。

四、模擬結果及分析

本章將對我們所提出的增加存活時間之適應性擇路演算法 (LEAR) , 與 Ad hoc On-demand Multipath Distance Vector (AOMDV) [35] , Distributed Energy Routing Protocol (DER) [42] , Ad hoc On-demand Distance Vector (AODV) [3] 做模擬的比較與分析。我們將針對以下方面作相關的模擬比較：資料封包的到達率 (Delivery Ratio of Data Packet) 、資料封包平均延遲 (Average Delay of Data Packet) 、網路負載 (Normalized Routing Load) 、網路存活時間 (Time of First Node Dead) 與行動節點電池耗盡的數量 (Number of Total Death Node) 。

4.1 模擬平台-網路模擬器



網路架構模擬器 (Network Simulator version 2 , ns-2) [43] 是一套物件導向的網路模擬器。它能模擬真實網路系統的架構和特性，在網路架構方面，有模擬路由器、資料連結層、媒體存取控制層、網路的行動節點等；在網路的特性方面又有封包的延遲 (Packet Delay) 或封包的丟棄 (Packet Drop) 等模擬。NS-2 是由 C++ 和 OTcl 做為開發的語言，使用 C++ 來編寫應用實例，在路由器可以使用 AODV、DSR 或自行開發的擇路演算法。OTcl 來操縱這些實例，NS2 是一個 OTcl 的腳本解譯器 (Script Interpreter) ，它把使用者所寫好的腳本進行解讀後，產生出模擬的結果，進而對結果進行分析，以對模擬的情境有更深入的瞭解，是一個相當方便的模擬平台。

4.2 模擬環境參數設定

隨意路徑移動模式 (Random Waypoint Model) [44]，表示網路上的每個行動節點先隨機選取一個目的地點，然後以預先設定的速度移動到該點，接著便休息一段時間 (Pause Time)，然後再隨機選取下一個目的地點，重複執行此移動模式。

我們藉由此模式下，針對不同的連線數 (Number of Source/Destination Pair) 及不同的平均速度 (Velocity) 為橫軸，探討演算法的效能分析。

下表為模擬環境所設定的參數：

參數名稱	參數數值	
模擬範圍	2400m*600m	
模擬時間	500seconds	
行動節點個數	100nodes	
傳輸頻寬	2Mbps	
傳輸半徑	250m	
擇路表單允許路徑數	3	
移動方式	random waypoint model, no pause time	
行動節點交通形式	固定位元速率 (Constant Bit Rate, CBR)	
資料封包速率	4packets/sec	
資料封包大小	512Bytes	
電池能量初值	70Joules	
傳送資料封包消耗能量	2W	
接收資料封包消耗能量	1W	
橫軸	針對不同連線數	針對不同移動速度
移動速度	10 m/sec	5、10、15、20 m/sec
連線數	10、12、14、16、18、20	14

表 4.1 模擬環境

4.3 數據分析

4.3.1 資料封包的到達率

測量資料封包的傳輸到達率，在模擬的環境下，以固定封包傳送，傳輸到達率等於正確送達之資料封包除以所有發送之資料封包。其數值越大越好，表示擇路演算法有較高的到達率。

4.3.2 資料封包平均延遲

資料封包平均延遲時間，封包從來源端透過網路傳輸到目的端所花費的時間為延遲時間，資料封包平均延遲時間等於所有封包延遲時間相加除以封包個數。其值越小越好，表示資料傳送的路徑有跳躍數或壅塞較少。

4.3.3 網路負載

網路的負載等於傳送的資料封包及傳送的控制封包相加，除以接收的資料封包。其值越小越好，表示較少的控制封包。

4.3.4 網路存活時間

網路存活時間，在模擬環境下，當網路上的行動節點開始傳送資料封包，某路徑上的中繼節點剛好把所有電池容量使用完，那就會造成路徑斷裂無法繼續做傳遞資料封包的工作，表示網路存活時間結束。網路存活時間越大越好，表示網路資料流量有較好的平衡。

4.3.5 行動節點電池耗盡的數量

行動節點電池耗盡的數量，模擬時間結束後，不能傳送資料的行動節點數量。其值越少越好，表示能夠減少重要行動節點電池消耗的能力。

4.4 模擬結果

我們可以看出 LEAR 在資料封包的到達率（見圖 4.1 及圖 4.2），封包的到達率由高到低依序為 LEAR、AOMDV、DER 及 AODV，由於 LEAR 與 AOMDV 路徑搜尋的機制相同，故封包到達率會很接近，DER 搜尋最大能量路徑，在低速時會有較大能量到達率，但高速及高連線數時，封包到達率會很接近 AODV。

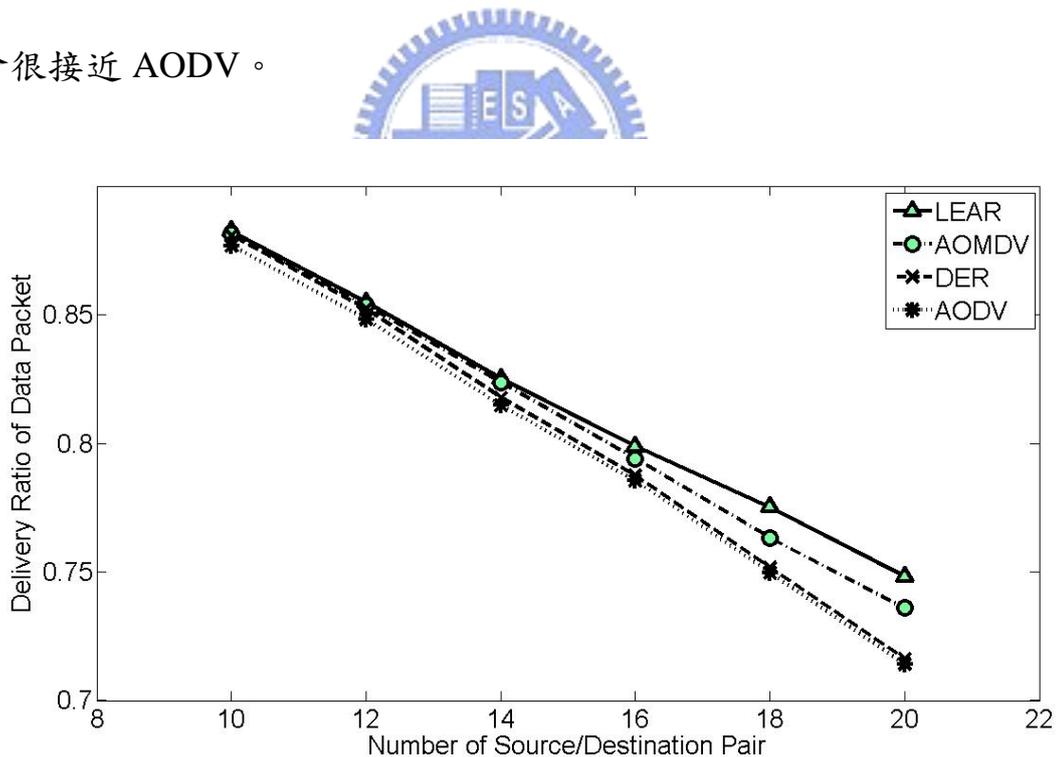


圖 4.1 根據不同連線數的資料封包的到達率（平均速度=10m/s）

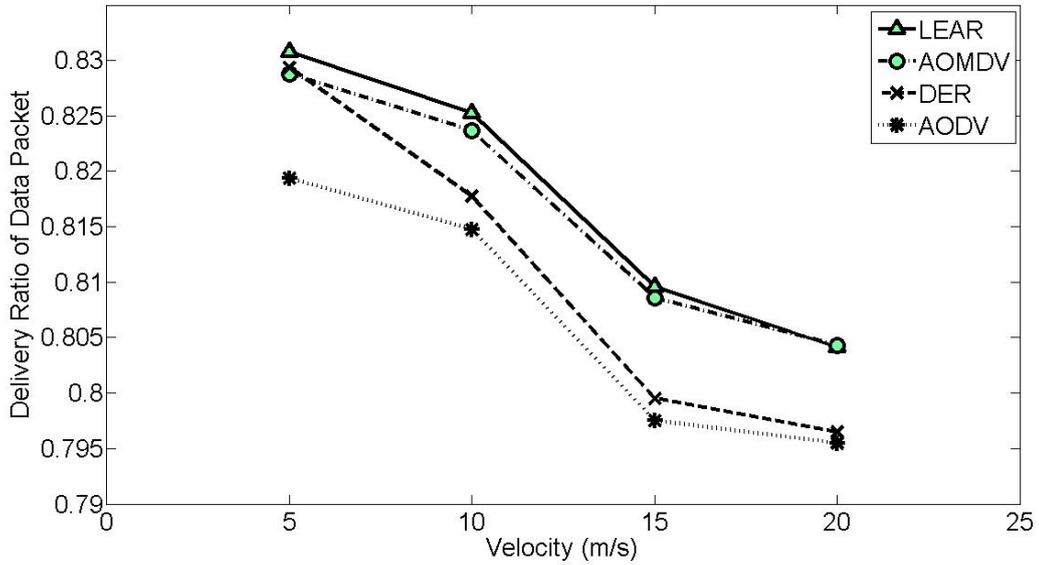


圖 4.2 根據不同平均速度的資料封包的到達率 (連線數=14)

平均延遲時間 (見圖 4.3 及圖 4.4)，時間越少越好，由低到高依序為 LEAR、AOMDV、AODV 及 DER，由圖中觀察，LEAR 多路徑的備份也是以最短路徑的方式 (跳躍數相同)，由於路徑能量較大的路徑通常有較少的資料在傳送，減少碰撞及壅塞 (congestion) 的發生，低速及連線數高的情況會比其它的擇路演算法好，而 DER 由於需傳送的跳躍點較多，故會有較多的平均延遲時間。

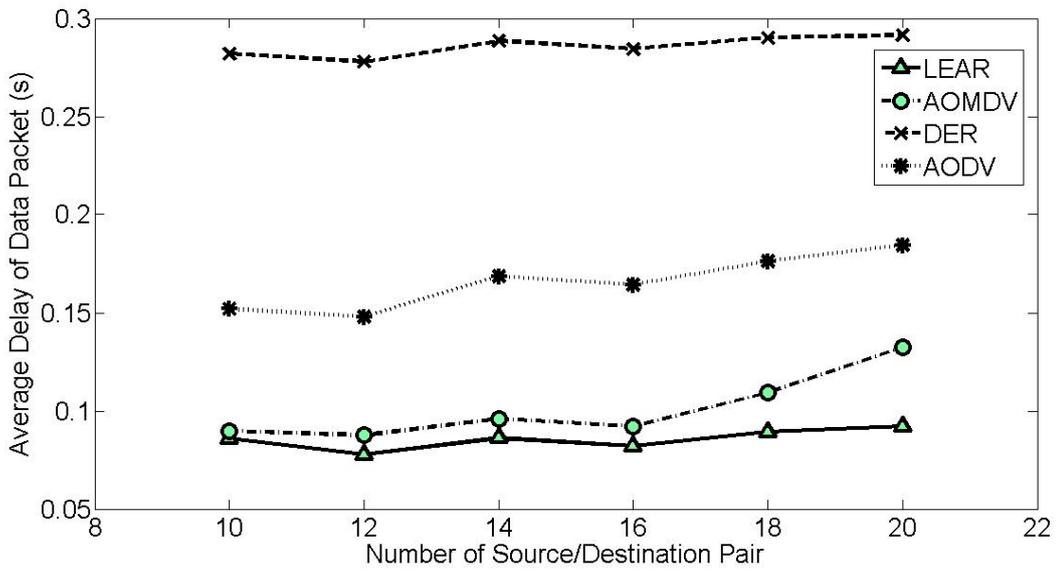


圖 4.3 根據不同連線數的資料封包平均延遲 (平均速度=10m/s)

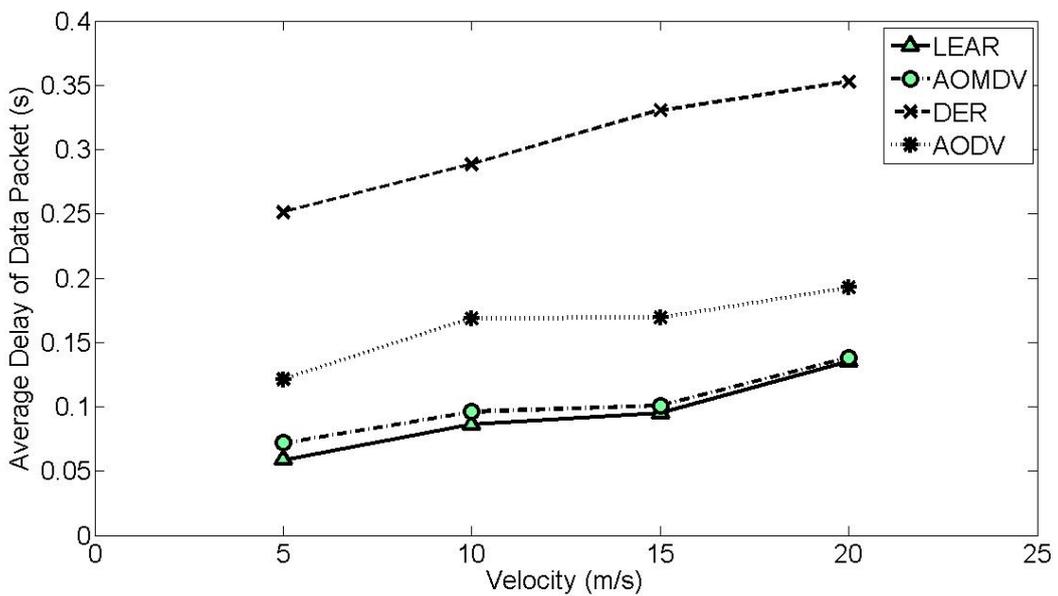


圖 4.4 根據不同平均速度的資料封包平均延遲 (連線數=14)

網路負載(見圖 4.5 及圖 4.6),值越低越好,由低到高依序為 AOMDV、LEAR、AODV 及 DER,而 LEAR 由於需要控制封包去通知上個節點擇路表單的路徑剩餘能量,以延長網路存活時間,比 AOMDV 有較多的控制封包,但 LEAR 擁有較多的備份路徑,所以比 AODV 及 DER 好 15~30%。DER 本身需較多控制封包去建立路徑,所以較差。

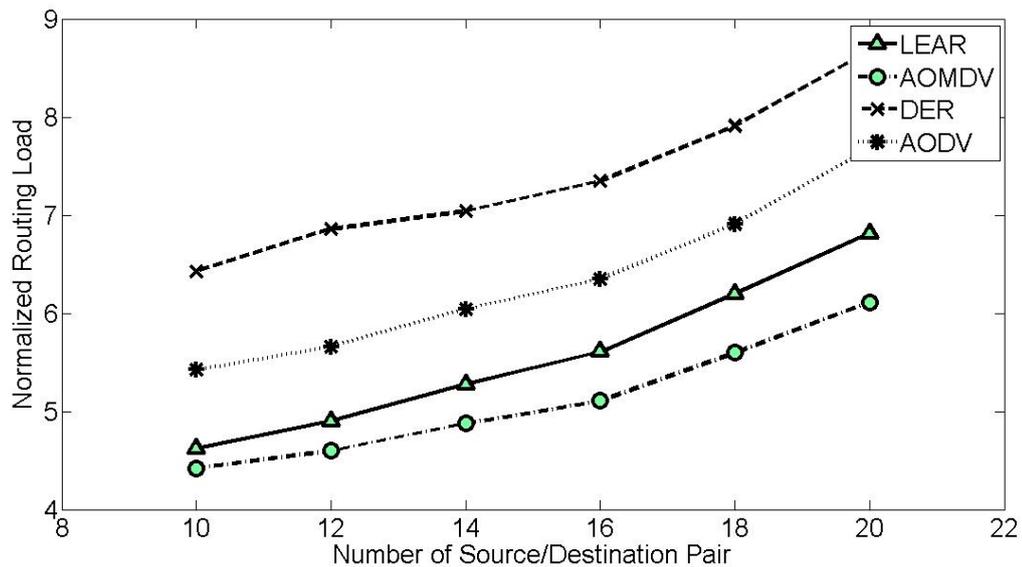


圖 4.5 根據不同連線數的網路負載 (平均速度=10m/s)

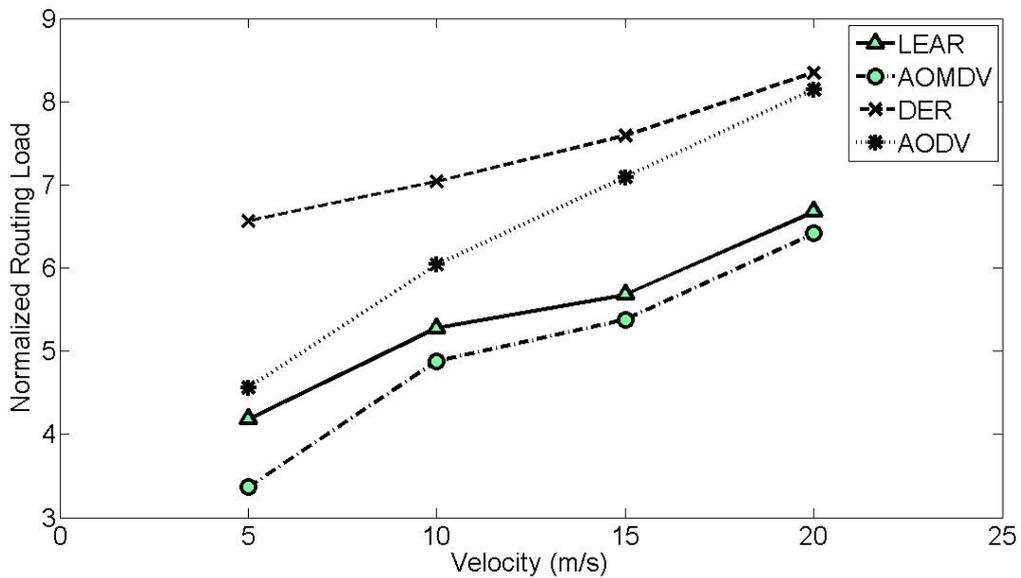


圖 4.6 根據不同平均速度的網路負載 (連線數=14)



網路的存活時間 (見圖 4.7 及圖 4.8) , 其值越高越好, 由高到低依序為 LEAR、AOMDV、DER 及 AODV, 由於 LEAR 及 DER 會選擇能量較大的路徑, 而 AOMDV 及 AODV 不會, 而 DER 走較多的跳躍點表示整個網路的能量消耗較多, 故會比 AOMDV 還差, 但在高連線數時, 則趨近於 AODV, 而 LEAR 的存活時間連線數 14 時, 會比其他的演算法大約好 10~35 秒。

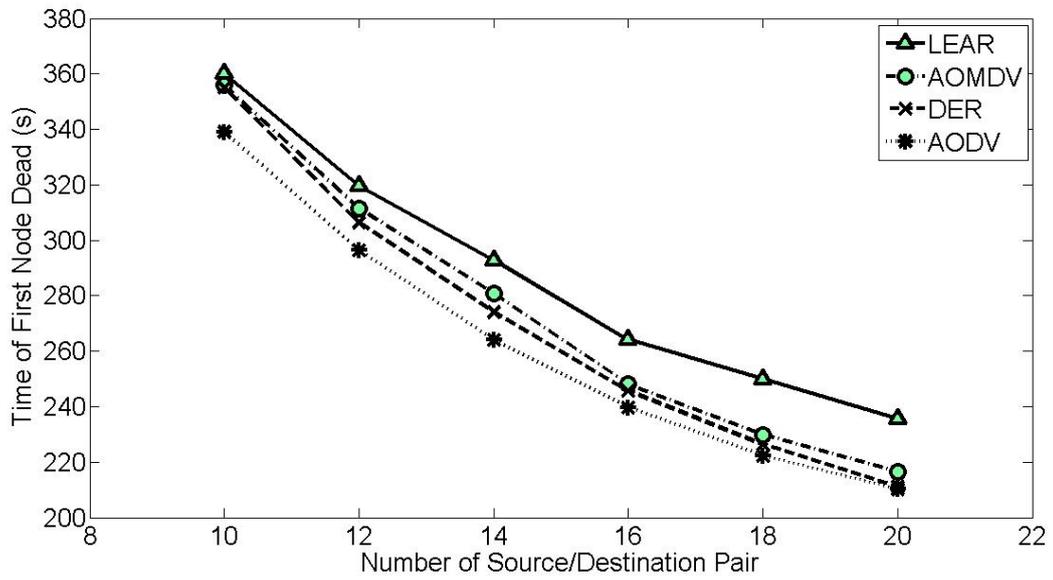


圖 4.7 根據不同連線數的網路存活時間 (平均速度=10m/s)

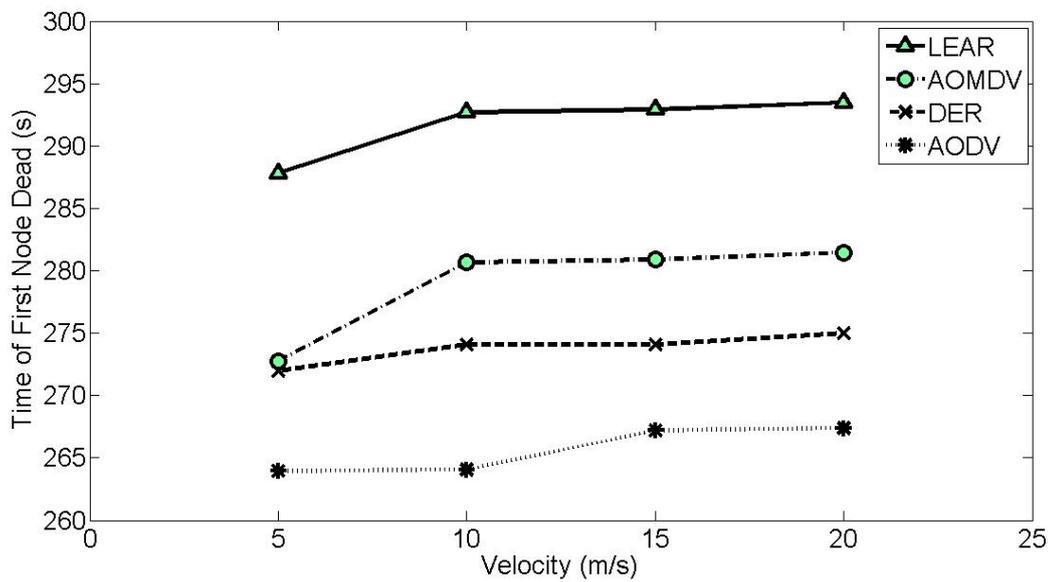


圖 4.8 根據不同平均速度的網路存活時間 (連線數=14)

行動節點電池耗盡的數量（見圖 4.9 及圖 4.10），其值越低越好，依序為 LEAR、AOMDV、DER、AODV。LEAR 在資料傳送的過程中會避開能量不足的行動節點去選擇一條電池能量較多的路徑，故優於其他的演算法，在連線數等於 18 時，會比其他演算法少 7~15 行動節點電池耗盡的數量。

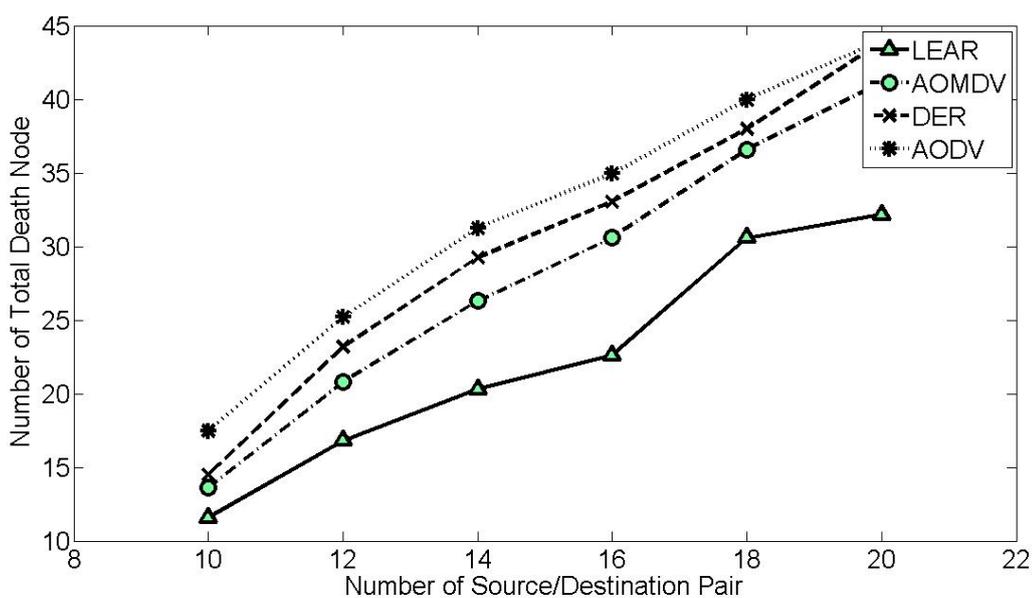


圖 4.9 根據不同連線數的行動節點電池耗盡的數量（平均速度=10m/s）

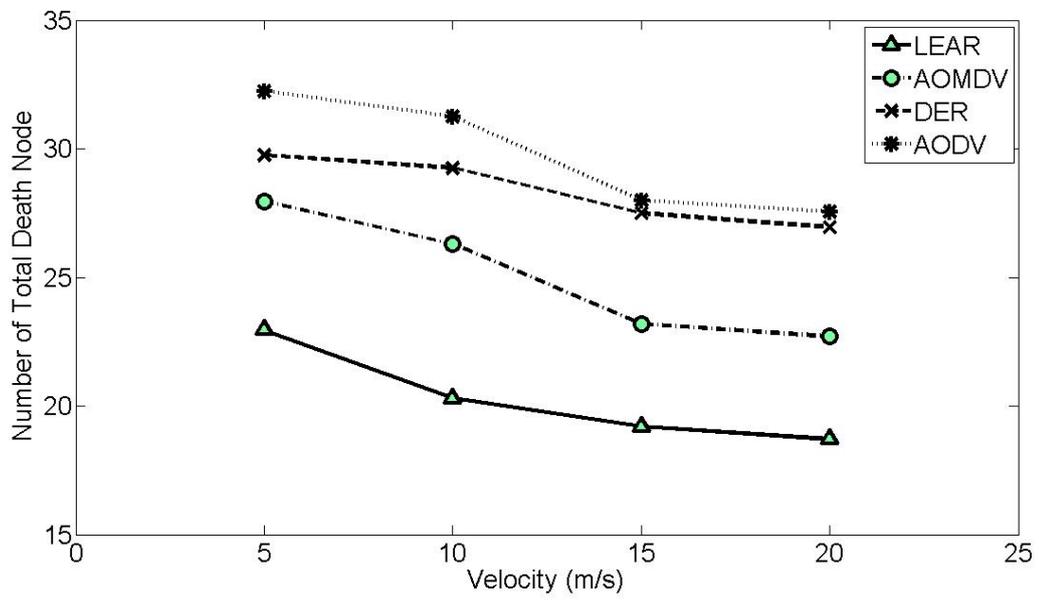


圖 4.10 根據不同平均速度的行動節點電池耗盡的數量 (連線數=14)



五、結論

目前網路發展的環境，無線隨意網路的應用很多，不論在軍事上，學術研究，和醫療用途上，大部分為了解決在網路不易架設或戶外的環境，有網路減少基地台的架設成本及較佳的移動性的優點，但由於受限於電池能量問題，產生整體資料傳送時間縮短問題，故本篇論文主要能有效利用多路徑的方式，延長各個行動節點的使用時間發展出來的演算法。

在這篇論文之中，為了延長行動節點使用時間，設計出 LEAR 演算法。讓資料的傳送可以經由擇路表點選擇最大剩餘能量的路徑傳送。傳送資料的過程中，當能量不足時，會通知上一個行動節點，並更新擇路表單內剩餘路徑能量欄位，由此資料便可選擇一條較多能量的路徑。經由模擬結果得知，LEAR 可以延長網路存活時間及減少整體行動運算節點電池耗盡的數量。

未來發展，目前電池發展的領域有一部是使用太陽能的方式來儲存電池能量，由於行動節點傳送資料時需消耗較多的網路資料，當行動節點能量不足時，會選擇較大能量路徑來傳送資料，但對於沒傳送資料的網路節點可以經由太陽能充電的方式儲存電能，如此對於整個網路的利用率將大大提升。

参考文献

- [1] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Proceedings of the ACM SIGCOMM '94 Conference*, Aug. 1994, pp. 234-244.
- [2] S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *ACM Mobile Networks and Appl. J., Special Issue on Routing in Mobile Communication Networks*, Oct. 1996, pp. 183-197.
- [3] C. E. Perkins and E. M. Royer, "Ad Hoc On-Demand Distance Vector Routing," *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Appl.*, Feb. 1999, pp. 90-100.
- [4] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks," *Ad Hoc Networking*, edited by C. E. Perkins, Addison-Wesley, 2001.
- [5] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *Proceedings of IEEE Infocom '97*, Apr. 1997, pp. 1405-1413.
- [6] V. D. Park, M. S. Corson, "Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification," *Internet-Draft, draft-ietfmanet-tora-spec-04.txt*, Jul. 2001.
- [7] IEEE Std 802.11-1997 Supplement to IEEE standard for information technology telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements-part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.
- [8] IEEE Std 802.11a-1999 Supplement to IEEE standard for information technology telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: high-speed physical layer in the 5 GHz band.

- [9] IEEE Std 802.11b-1999 Supplement to IEEE standard for information technology telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 2: higher-speed physical layer (PHY) extension in the 2.4 GHz band.
- [10] IEEE Std 802.11g-2003 Supplement to IEEE standard for information technology telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz band.
- [11] Draft Std 802.11n, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment: Enhancements for Higher Throughput, IEEE Draft Std P802.11n/D2.00, Feb. 2007.
- [12] C.-K. Toh, "A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing," *Proceedings of 15th IEEE Annual International Phoenix Conference Computers and Communications*, Mar. 1996, pp. 480-486.
- [13] R. Dube, C. D. Rais, K-Y. Wang, and S. K. Tripathi, "Signal Stability-Based Adaptive Routing (SSA) for Ad Hoc Mobile Networks," *IEEE Personal Communications*, Feb. 1997, pp. 36-45.
- [14] Z. J. Haas, M. R. Pearlman, "The Performance of Query Control Schemes for the Zone Routing Protocol," *IEEE/ACM Transactions on Networking (TON)*, vol. 9, no. 4, Aug. 2001, pp. 427-438.
- [15] P. Samar, M. R. Pearlman, Z. J. Haas, "Hybrid Routing: The Pursuit of an Adaptable and Scalable Routing Framework for Ad Hoc Networks," *The Handbook of Ad Hoc Wireless Networks*, Boca Raton, FL: CRC Press, 2003.
- [16] C.C. Chiang, H. K. Wu, W. Liu, and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," *Proceedings of IEEE SICON '97*, Apr. 1997, pp. 197-211.
- [17] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T.-W. Chen, "Scalable

Routing Strategies for Ad hoc Wireless Networks,” *IEEE Journal on Selected Areas in Communication*, vol. 17, no. 8, Aug. 1999, pp. 1369-1379.

[18] R. Sivahumar, P. Sinha, and V. Bharghavan, “CEDAR: a Core-Extraction Distributed Ad hoc Routing algorithm,” *IEEE Journal on Selected Areas in Communication*, vol. 17, no. 8, Aug. 1999, pp. 1454–1465.

[19] J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, “A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols,” *Proceeding of the ACM/IEEE International Conference on Mobile Computing and Networking*, Oct. 1998, pp. 25-30.

[20] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, “Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks,” *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, Aug. 1999, pp. 195-206.

[21] E. M. Royer and C.-K. Toh, “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks,” *IEEE Personal Communication Magazine*, Apr. 1999, pp. 46-55.

[22] S. R. Das, R. Castaneda, J. Yan, and R. Sengupta, “Comparative Performance Evaluation of Routing Protocols for Mobile, Ad hoc Networks,” *Proceedings of the International Conference on Computer Communications and Networks (ICCCN)*, Oct. 1998, pp. 153–161.

[23] B. W. Parkinson and S. W. Gilbert, “NAVSTAR: Global Positioning System - Ten Years Later,” *Proceedings of IEEE*, Oct. 1983, pp. 1177-1186.

[24] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, “A Distance Routing Effect Algorithm for Mobility (DREAM),” *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, Oct. 1998, pp. 76-84.

[25] Y.-B. Ko and N. H. Vaidya, “Location-Aided Routing (LAR) in mobile ad hoc networks,” *ACM Wireless Networks Journal*, vol. 6, no.4, 2000, pp. 307-321.

- [26] R. Jain, A. Puri, and R. Sengupta, "Geographical Routing Using Partial Information for Wireless Ad Hoc Networks," *IEEE Personal Communications*, vol. 8, no. 1, Feb. 2001, pp. 48-57.
- [27] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, Aug. 2000, pp. 243-254.
- [28] K.-T. Feng and T.-E. Lu, "Velocity and Location Aided Routing for Mobile Ad Hoc Networks," *Proceedings of IEEE Vehicular Technology Conference (VTC) Fall*, Sep. 2004, pp. 2789-2793.
- [29] T.-E. Lu. and K.-T Feng, "Predictive Mobility and Location-Aware Routing Protocol in Mobile Ad Hoc Networks," *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, Dec. 2005, pp. 899-903
- [30] K.-T. Feng, C.-H. Chen, and T.-E. Lu, "Velocity-Assisted Predictive Mobility and Location-Aware Protocols for Mobile Ad Hoc Networks," *accepted and to appear in IEEE Trans. on Vehicular Technology*, 2008.
- [31] J. Li, J. Jannotti, D. S. J. D. Couto, D. R. Karger, and R. Morris, "A Scalable Location Service for Geographic Ad Hoc Routing," *Proceedings of the ACM/IEEE Mobile Computing and Networking (MobiCom)*, Aug. 2000, pp. 120-130.
- [32] R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. Decouto, "CarNet: A Scalable Ad Hoc Wireless Network System," *Proceedings of the 9th ACM SIGOPS*, Sep. 2000, pp. 61-65.
- [33] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *T. Imielinski and H. Korth, Eds. Mobile Computing*, 1996, Ch. 5.
- [34] S. J. Lee and M. Gerla, "AODV-BR: Backup Routing in Ad hoc Networks," *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Sep. 2000, pages 1311-1316.
- [35] M. K. Marina and S. R. Das, "On-demand Multipath Distance Vector Routing in Ad Hoc Networks," *Proceedings of the 9th IEEE International Conference on Network Protocols*, Nov. 2001, pp. 14-23.

- [36] J.-C. Cano and D. Kim, "Investigating Performance of Power-aware Routing Protocols for Mobile Ad Hoc Networks," *Proceedings of the International Mobility and Wireless Access Workshop*, Oct. 2002, pp. 80-86.
- [37] J.-C. Cano and P. Manzoni, "A Performance Comparison of Energy Consumption for Mobile Ad Hoc Network Routing Protocols," *Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, Sep. 2000, pp. 57-64.
- [38] S. Singh, M. Woo, and C. S. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks," *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Oct. 1998, pp. 181-190.
- [39] M. Stemm and R. H. Katz, "Measuring and Reducing Energy Consumption of Network Interfaces in Hand-Held Devices," *IEICE Transactions on Communications*, vol. E80-B, no. 8, Aug. 1997, pp. 1125-1131.
- [40] C.-K. Toh, H. Cobb, and D. A. Scott, "Performance Evaluation of Battery-LifeAware Routing Schemes for Wireless Ad Hoc Networks," *IEEE International Conference on Communications*, vol. 9, 2001, pp. 2824-2829.
- [41] X. Jin, W. Cai, and Y. Zhang, "A RED Based Minimum Energy Routing Algorithm for Wireless Ad-Hoc Networks," *Wireless Communications, Networking and Mobile Computing*, vol. 2, 23-26, Sep. 2005, pp. 757-761.
- [42] W.-H. Cheng, "Energy-Aware Routing Protocols for Mobile Ad Hoc Network," *National Chiao Tung University, thesis*, Jun. 2006.
- [43] <http://www.isi.edu/nsnam/ns/ns-documentation.html>
- [44] T. Camp, J. Boleng, V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, 2002, pp. 483-502.