國 立 交 通 大 學

資訊管理研究所

碩 士 論 文

一個基於資訊家電的無線隨意區域網路分群機制

A Study on Wireless Ad Hoc Network Clustering Mechanism for Information Appliance

研 究 生：李 銘 家

指導教授：羅 濟 群 博士

中華民國 九十八 年 六 月

一個基於資訊家電無線隨意區域網路分群機制

A Study on Wireless Ad Hoc Network Clustering Mechanism for Information Appliance

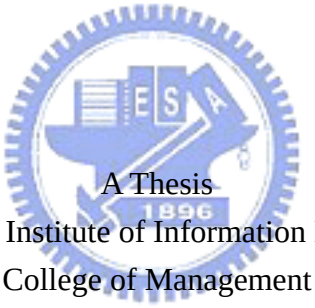
研 究 生：李 銘 家　　Student: Ming-Chia Lee

指導教授：羅 濟 群　　Advisor: Chi-Chun Lo

國立交通大學

資訊管理研究所

碩士論文



A Thesis
Submitted to Institute of Information Management
College of Management
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master of Science
in
Information Management
June 2009
Hsinchu, Taiwan, the Republic of China

中華民國 九十八 年 六 月

一個基於資訊家電的無線隨意區域網路分群機制

研究生：李銘家　　指導教授：羅濟群 教授

國立交通大學資訊管理研究所

摘要

資訊家電(Information Appliance, IA)是一種易於使用、具備網路通訊、資訊存取功能的智慧型家電。當 IA 的普及化時代逐漸來臨，IA 之間的通訊將可不受限於無線 AP 等基礎建設中樞裝置，而是以隨意網路的形式建立自適化網路拓撲，以增加網路的彈性與延展性。然而，隨意網路架構會衍生出基礎建設型網路不需考慮的安全性問題；同時，目前的研究尚未針對 IA 的特性設計適當之隨意網路拓樸分群演算法。

本論文提出一個基於 IA 的隨意網路分群機制，其中包含一個針對 IA 特性設計的隨意網路拓樸分群演算法：IA based Ad hoc Network Clustering Algorithm (IAdNCA)。IA 隨意網路分群機制導入 Wi-Fi 聯盟所提出的 Wi-Fi Protected Setup (WPS)，讓使用者可簡易地對資訊家電進行安全性設定與認證，以達到 IA 隨意網路的安全性與設定上的方便性。而 IAdNCA 將 IA 隨意網路拓撲分成數個群集，各群集中將有一個領導者節點(Leader Node)負責 IA 的註冊、認證與路由資訊的維護。透過安全性分析可證明此 IA 隨意網路分群機制的運作符合安全需求；而經模擬實驗可發現，IAdNCA 所構築的分群式 IA 隨意網路拓撲，其平均有效拓樸維持時間與現行演算法相比有 20%以上的增幅，說明 IAdNCA 可選出較適當的 IA 節點作為 Leader Node，使拓撲運作時的穩定性、壽命均高於現行演算法，由此證明了 IA 隨意網路分群機制的穩定性和實用性。

關鍵字：無線網路、隨意網路、資訊家電、WPS、分群拓撲、WCA

A Study on Wireless Ad Hoc Network Clustering Mechanism for Information Appliance

Student: Ming-Chia Lee Advisor: Dr. Chi-Chun Lo

Institute of Information Management

National Chiao Tung University

Abstract

Information Appliances (IAs) are devices which not only have the capability to communicate with other wireless-enabled facilities but provide user-friendly operating interface. With the pervasion of IAs, the communication among IAs may be no more restricted by infrastructure devices. They may establish self-organized wireless Ad Hoc network topology to increase scalability. However, there's still no adequate algorithm to help IAs establish effective Ad Hoc Network topology.

Our research provides an Ad Hoc Network Clustering Mechanism for IA. This mechanism comprises a clustering algorithm which is called IAdNCA. IAdNCA divides IAs into clusters, and a Leader Node is elected for each cluster to be in charge of cluster maintenance and authentication. Besides the functionality mentiond above, our mechanism as well adopts Wi-Fi Protected Setup (WPS) to make users easily configure security settings of IAs

Through the security analysis, our clustering mechanism is proved to fulfill the security requirement. In addition, through the stability analysis, IadNCA makes average effevtive topology maintenance time 20% better than other Ad Hoc clustering algorithms. Based on the results of analysis, our mechanism manifests its own stability and security.

誌謝

原本以為研究所兩年求學時光還有得我熬，不料光陰似箭，轉眼間我就是站在台上進行撥穗儀式的畢業生了。回顧在風城的這兩年，泰半都是在實驗室中度過，我在實驗室中得到了 know-how、友情、願景，以及做人處事的道理。

在這兩年之中，我有太多太多需要感謝的人。首先，感謝我的指導教授羅濟群老師，老師自由的教學風格使我能依照自己的興趣廣泛學習；而老師清晰的邏輯思維則使我學會既正確又嚴謹的研究方法及邏輯性的思考。再來，感謝實驗室的學長姐：強哥、建全、朝尉、阿吉、小牛、邱大人、栩嘉、鼎元和邦曄，教導我技術、知識、態度和人生的哲學。當然，也要感謝和我一同打拼的同學：志華、大蓉、小榕、昌民、家偉，沒有你們的支持與陪伴，我鐵定無法熬到畢業的這一天(特別是志華，我和你相處的時間都要超過我的女朋友了..)。然後，我也必須感謝碩一的學弟妹們：Popular、大頭、死人、小米、老健、一姐、鴻鈞、Vivian(偽碩一)，感謝你們和我一同度過作案子、耍白爛、打屁的歡樂與痛苦時光。又，我自認不是個忘本的人，因此大學時重要的麻吉：Alex, 阿寬, 杏仁茶, 料吃吃, Ghome, Oneway，你們是我人生中最無可取代的朋友，因為你們的友情 support，我才能順利完成畢業論文，我衷心感謝你們。最後，感謝我的家人：媽媽、爺爺、奶奶、外公、外婆...等，因為你們的栽培和教養，我才能有今日小成。最後的最後，我必須感謝我的女朋友：盈初，這兩年來委屈你了，必須因為我課業的忙碌而遷就於我；不但如此，還不斷鼓勵我、安慰我，以度過無數求學路途中的難關，千言萬語，我化為一句發自內心的「謝謝妳」！

此篇論文，獻給所有幫助過我的好友、家人，以及默默守護著全交通大學的土地公神明，想當然爾，致謝文的最後，就是「謝天」。

目次

圖目次

表目次

第一章 緒論

本章為緒論，主要說明本論文研究背景與動機、研究目的和研究流程，最後簡單的介紹後續章節的內容。

1.1 研究背景與動機

由於無線網路技術的成熟以及蓬勃發展，加上無線網路設備製造成本的降低，利用無線網路技術進行通訊已然成為生活中不可或缺的一部分。藉由無線網路利用射頻當作傳輸媒介的特性，人們不再受限於連接線路，而可以帶著個人行動運算設備進行移動，諸如筆記型電腦、手機或是個人數位助理，一般認為，在二十一世紀的今天，整個世界逐漸走向行動化[2][21]。

無線通訊技術的快速發展，加上資訊產業的推波助瀾，造成後 PC 時代(Post-PC Era)的來臨。依據[2]，所謂「後 PC 時代」代表使用者得以使用各種型態的裝置，包括計次付費的智慧型公共零售服務亭(kiosks)、膝上型電腦、PDA、手機等去存取網路服務。根據[21]，在後 PC 時代中，資訊電腦設備將往兩個方向發展：一是強調多功能、高規格及運算能力強的個人電腦、筆記型電腦、工作站或是伺服器，另一則是強調單一功能、用途簡單及價格低廉的資訊家電產品，因為資訊家電產品價格較低、操作容易並且多為網際網路之互動而設計，因此資訊家電可能是未來市場的主流。

對於資訊家電之定義，[3][21]認定：資訊家電(Information Appliance, IA)為設計於從事特定活動，且具備資訊專門功能的個人化裝置。[16]將資訊家電類別分為智慧型裝置、家庭娛樂、行動電話、網際網路裝置、個人計算機裝置等五項。至於資訊家電的主要特色則如下列所述[3]：

- 被設計且預先設定組態用於單一、專門功能應用
- 對於未受訓練的使用者而言易於使用，讓使用者忽略其存在

- 能夠與其他資訊家電進行資訊分享

IA 未來發展的願景固然令人期待，然而針對後 PC 時代仍有許多必須重要的議題需要加以考量，諸如資通存取的安全性與多重裝置的相互連結[21]，其中，因應 IA 通訊安全性議題，Wi-Fi 聯盟於 2006 年底推出 Wi-Fi Protected Setup (WPS) 標準[23]，以讓未諳資訊安全知識的 IA 消費者能夠透過標準化且簡易、方便的方式對無線區域網路(AP 與無線裝置客戶端)進行安全化設定。

當 IA 的普及率與通訊功能均到達一定程度時，無線網路拓樸將可能朝分散式的隨意網路[2]發展，以擺脫無線 AP 等中樞傳輸設備對於網路拓墣地理性的限制。但目前 WPS 標準主要是對應家庭網路、辦公室網路等小型基礎建設(Infrastructure Mode)無線網路環境而設計，在其標準文件中明確提到 WPS 不支援無線隨意網路(Wireless Ad Hoc Network)拓樸架構[23]；由此不難看出，IA-based 無線隨意網路的發展，在安全性機制方面仍有進步的空間。

1.2 研究目的

本論文目的在於提出一個利用資訊家電(IA)相互連結的二階層式無線安全隨意網路分群機制，在克服基礎建設(Infrastructure Mode)無線網路拓墣模式無法動態彈性擴充網路拓墣的缺點之前提下、同時保證隨意網路構成程中各個環節的安全性。此機制之運作流程包含兼顧安全性的群集領導節點(Leader Node)競選與自適化分群演算法；另外，在合法資訊家電的註冊程序方面，採用 Wi-Fi 聯盟所提出的 WPS(Wi-Fi Protected Setup)協議，以不修改固有 WPS 核心流程的前提下，讓拓樸群集中的 Leader Node 兼任群集管理者及 WPS 裝置註冊者角色，達成資訊家電使用者在隨意網路環境中進行 easy secure configuration 的目標；至於 IA 節點的通訊機制，則是採用 Cluster-Based routing protocol，讓 Leader Node 負責旗下成員節點的通訊。

總體而言，本論文目標列舉如下：

- IA 隨意網路領導節點競選與自適化分群方法設計
- 融合 WPS 安全性設定流程的隨意網路認證機制
- 提升 IA 隨意網路傳輸穩定性、延長 IA 隨意網路拓墣的有效運作時間

1.3 論文架構

本論文論文共分為五章，茲分別敘述如表 1 所示：

表 1 論文架構簡表

章節	**內容概述**
第一章	緒論，對研究背景、動機及整體研究方向做一概略的描述
第二章	文獻與相關研究探討，首先對於資訊家電(IA)的定義作相關文獻探討，然後針對目前的隨意網路分群演算法進行研究，並對具有相關研究方向的文獻進行介紹。
第三章	首先提出無線 IA 隨意網路的構築情境，然後提出實行 IA 隨意拓墣構築的解決方案
第四章	安全性分析與實驗模擬；對第三章提出的流程架構進行安全性分析、對於提出的分群演算法則以軟體模擬測試穩定性
第五章	總結與未來發展，總結本論文之成果，並提出未來可能的研究方向。
參考文獻	列舉本論文所參考之各項文獻。

第二章 文獻探討

在本章節裡，主要介紹及說明與本論文主題相關的一些研究內容，包括資訊家電的定義與發展、WPS 無線網路裝置註冊程序、以及分群式無線隨意網路的拓墣構成演算法。

2.1 資訊家電

科技日新月異，不論是硬體的運算能力或是無線通訊傳輸技術都不斷在進步中。PC 變成辦公室或家庭中不可或缺的設備之一，而其發展也逐漸朝向微型化、低價、具有高速運算能力的方向前進。然而，除了 PC 產品之外，結合了嵌入式系統(Embedded System)、寬頻網路、Internet、以及傳統家電功能特色的資訊家電(Intelligent Appliance, 以下簡稱 IA)在後 PC 時代的電器產品市場中逐漸站有一席之地[5]。

目前在學術領域中，尚未對 IA 做出統一的定義，但基本上，各方對於 IA 的認定與評估均不脫出幾個要點[3][5][8][16][21]：

- 低價位、且易於使用、具備連網、資訊存取功能之裝置
- 通常具有相當程度運算能力的嵌入式系統
- 專精於某方面傳統家電的功能
- IA 即將繼 PC 成為明日之星，PC 也將朝 IA 之路而趨近

基於各方對於 IA 的認定，不難發現 IA 所涵蓋的層面廣大、且沒有統一硬體型態與規格，而[16]對於資訊家電的種類，則提出如圖 1 的分類方式，將資訊家電分為智慧型裝置、家庭娛樂、行動電話、網際網路裝置、個人計算機裝置等五項；至於 IA 相關的研究主題，主要圍繞在結合 IA 和資訊網路的機制架構、及如何令 IA 彼此通訊、合作，以產生單一傳統家電無法提供的服務和功能[5]。

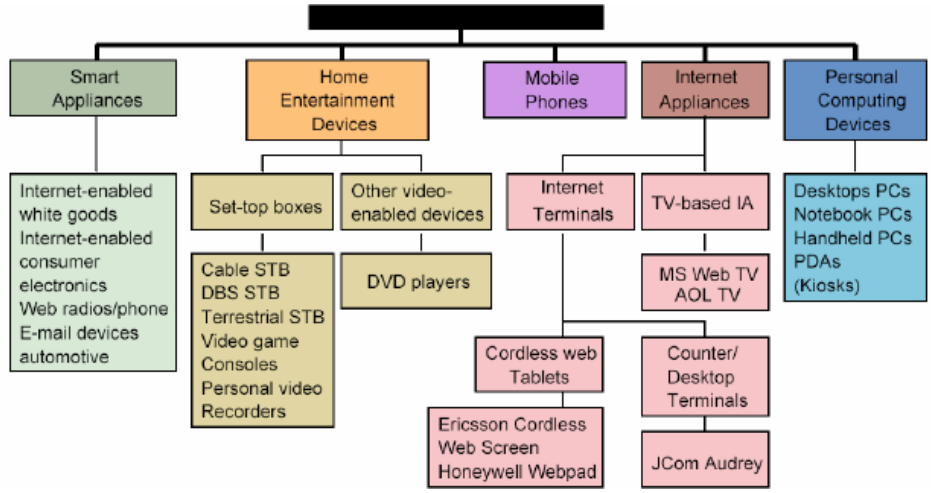


圖 1　Information Appliance classification[16]

2.1.1 Smart Home

在 IA 的概念與各類 IA 產品問世後，更多樣化的電腦應用出現在日常生活，人機互動不再局限於 PC 前。由於 IA 的使用範圍絕大多數還是在家庭裡，因此串連各設備的家庭網路(Home Network)的概念與發展也越來越受重視[16]；借助 IA 功能的家庭網路期望達到"Smart Home"的願景，所謂"Smart Home"涵蓋以下三點特色[22]：

- 智慧型網路
- 智慧型控制
- 家庭自動化

Smart Home Network 的構築主要取決於 IA 本身以及 IA 間的資料傳輸。IA 設備包含硬體架構與系統軟體二部分，硬體相對於一般的個人電腦擁有較少的運算能力與儲存空間，可執行家電之間的溝通協調和少量資料儲存即可，系統軟體則用於輔助軟硬體之間的溝通協調[8]。在傳輸部分，IA 間的資料傳輸可分別利用有線介質如電源線(Electric Power Line)、雙絞線(Twisted Pairs)、光纖(Optical Fiber)、同軸電纜(Coaxial Cable)等或是無線介質如紅外線(Infrared)、無線射頻(Radio Frquency)來達成。

2.1.2 Wi-Fi Protected Setup

IA的發展使得智慧型家庭網路擁有更多拓展應用的潛力，然而相對地，網路安全的重要性也變得越來越不可忽視。一般而言，目前的IA使用者普遍未體認到網路安全之於IA的重要性、且尚未擁有足夠的網路安全知識。為了能補足IA消費者在這方面的不足，WPS(Wi-Fi Protected Setup)應運而生。WPS為2007年由Wi-Fi 聯盟發布的協定標準[23]，目的是讓消費者不需資訊安全的知識背景，透過更簡單的方式即可註冊無線網路裝置，讓無線網路中的認證者(Authenticator，通常為無線AP的角色)和合法裝置擁有相同的安全性證書(security credential，通常意指pre-shared key)，註冊設定過後，合法裝置可利用安全性證書通過無線網路的認證以使用網路；相反的，沒有安全性證書的節點則無法通過認證進入無線網路，藉此保證無線網路的安全性。

WPS 在使用上提供用戶容易操作的步驟，而在無線安全方面的強度則支援WPA 至WPA2等級的加密方式。在WPS中，Pin code、SSID 和加密金鑰profile是在此協定上的主要傳輸資料，所有的資料都是先經過加密、雜湊後再傳送到無線網路中，接收者收到資料後，再轉換成原始的內容，安全性較佳。

2.1.2.1 WPS Registration Protocol

在WPS的註冊程序中，主要由三種角色所組成，分別是Enrollee、Registrar和AP，三者的關係如圖2所示。使用者將Enrollee經由註冊程序透過E介面註冊至Registrar，Registrar登陸Enrollee後產生安全性證書再透過E介面回傳，並將安全性證書利用M介面傳至AP以供認證之用，註冊程序完成後，Enrollee透過A介面利用先前產生的安全性證書進行認證，認證成功後得以存取網路。在實務上，支持WPS的Registrar產品通常也兼具無線AP的功能，因此Registrar和AP將合併為同一個角色。

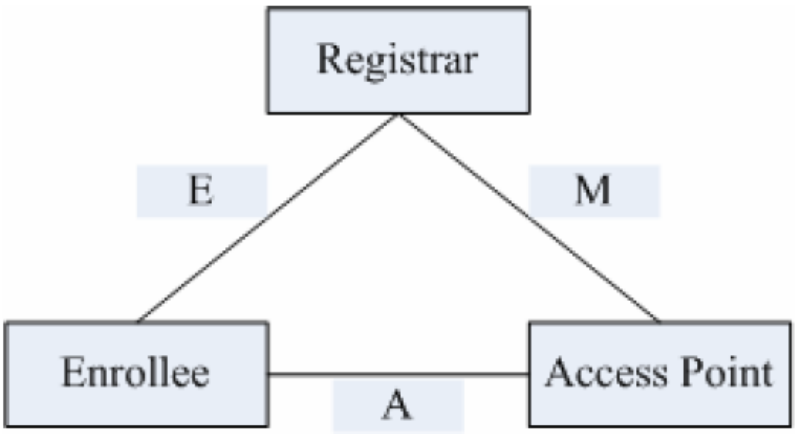


圖 2　WPS 元件角色關係圖

WPS註冊程序細節主要由M1~M8等八個訊息交換所完成，分成眾多訊息交換的理由在於確保裝置註冊流程的安全性，M1~M8的訊息交換如圖3，根據WPS規格中的定義，主要允許支援WPS標準的無線網路產品提供兩種讓使用者進行WPS註冊程序的方式：

1. Pin Input Config (PIN)：

- 首先，使用者啟動Registrar的註冊功能，Registrar將提示要求輸入Enrollee的pin code，然後使用者啟動Enrollee的註冊功能，此時Enrollee會傳送M1訊息告知Registrar自己的Diffie-Helman公鑰PK_R和隨機數N1，並生成的Pin code，使用者須在timeout時間內將Pin code輸入Registrar。
- Registrar得到Pin code之後，會利用PK_R和自己Diffie-Helman公鑰PK_E的生成Diffie-Helman私鑰p，然後利用Enrollee MAC Address、p、N1與隨機數N2生成AuthKey、KeyWrapKey與EMSK三把key，再發送M2訊息給Registrar，M2裡包含Registrar的Diffie-Helman公鑰、N2與經過HMAC雜湊後的M1與M2訊息，雜湊金鑰使用AuthKey。
- Enrollee收到M2後，同樣生成AuthKey並確認HMAC中的雜湊值無誤，然後產生臨時秘密參數E-S1與E-S2，利用Pin code的前半段MAC值、E-S1、PK_R與PK_E生成E-Hash1、利用Pin code的後半段MAC值、E-S2、PK_R與PK_E生成

E-Hash2，再將N2、E-Hash1、E-Hash2與經過AuthKey加密後的M2、M3訊息附在M3當中傳送給Registrar。

- Registrar收到M3後，產生臨時秘密參數R-S1與R-S2，使用規格中定義的ENC函式和KeyWrapKey將R-S1加密，並利用Pin code的前半段MAC值、R-S1、PK_R與PK_E生成R-Hash1，利用Pin code的後半段MAC值、R-S2、PK_R與PK_E生成R-Hash2，再將N1、R-Hash1、R-Hash2、加密後的R-S1以及經過AuthKey加密後的M3、M4訊息附在M4，傳送給Enrollee。
- Enrollee收到M4後，使用規格中定義的ENC函式和KeyWrapKey將E-S1加密，將N2、加密後的E-S1、經過AuthKey加密後的M4、M5訊息附在M5中傳送給Registrar。
- Registrar收到M5後，使用規格中定義的ENC函式和KeyWrapKey將R-S2加密，將N1、加密後的R-S2、經過AuthKey加密後的M5、M6訊息附在M6中傳送給Enrollee。
- Enrollee收到M6後，使用規格中定義的ENC函式和KeyWrapKey將E-S2加密，將N2、加密後的E-S2、經過AuthKey加密後的M6、M7訊息附在M7中傳送給Registrar。
- 經過M2至M7訊息交換後，Enrollee利用得到的R-S1、R-S2重新計算R-Hash1與R-Hash2，比對過後確認Registrar的合法性；而Registrar利用得到的E-S1、E-S2重新計算E-Hash1與E-Hash2，比對過後確認Enrollee的合法性，至此雙方均確立了對方的身分，Registrar使用規格中定義的ENC函式和KeyWrapKey將安全性證書加密，伴隨N1與經過AuthKey加密後的M7、M8訊息，附在M8中傳送給Enrollee，結束註冊流程。

2. Push Button Configuration (PBC)：

- 首先，使用者按下Enrollee的PBC鈕，Enrollee將發送probe request搜尋Registrar，並在Walktime時間內(120秒內)接受來自Registrar的PBC訊息。

- 使用者在Walktime時間內按下Registrar的PBC紐，Registrar發送Probe告知Enrollee PBC Active state啟動。
- 若Registrar偵測到來自一個以上Enrollee所發送的probe request，將停止PBC method。
- Enrollee發送隨機生成的Pin Code給Registrar，並開始M1至M8的訊息交換，交換過程如同PIN method，故在此不再贅述。

Enrollee → Registrar: $M_1 =$ Version || N1 || Description || PK_E

Enrollee ← Registrar: $M_2 =$ Version || N1 || N2 || Description || PK_R [|| ConfigData] || $HMAC_{AuthKey}(M_1 \| M_2^*)$

Enrollee → Registrar: $M_3 =$ Version || N2 || E-Hash1 || E-Hash2 || $HMAC_{AuthKey}(M_2 \| M_3^*)$

Enrollee ← Registrar: $M_4 =$ Version || N1 || R-Hash1 || R-Hash2 || $ENC_{KeyWrapKey}$(R-S1) || $HMAC_{AuthKey}(M_3 \| M_4^*)$

Enrollee → Registrar: $M_5 =$ Version || N2 || $ENC_{KeyWrapKey}$(E-S1) || $HMAC_{AuthKey}(M_4 \| M_5^*)$

Enrollee ← Registrar: $M_6 =$ Version || N1 || $ENC_{KeyWrapKey}$(R-S2) || $HMAC_{AuthKey}(M_5 \| M_6^*)$

Enrollee → Registrar: $M_7 =$ Version || N2|| $ENC_{KeyWrapKey}$(E-S2 [||ConfigData]) || $HMAC_{AuthKey}(M_6 \| M_7^*)$

Enrollee ← Registrar: $M_8 =$ Version || N1 || [$ENC_{KeyWrapKey}$(ConfigData)] || $HMAC_{AuthKey}(M_7 \| M_8^*)$

圖 3 WPS Registration Protocol Process

2.2 分群隨意網路架構

依照 IA 發展與普及的速度，可以預見的是將來環境中會充斥著大量具有計算能力和網路連線能力的 IA 設備，現行 IA 的網路拓撲運行模式主要採用具基礎架構的無線網路(Infrastructure Wireless Network)。在此模式中，網路拓撲的建構

必須受限於基地台或交換機等封包轉送管理設備而降低拓撲的延展性和彈性。對於小規模的 IA 隨意網路應用而言(如 IA 構成的智慧型家庭)，基礎建設的限制與不便尚在可容許的範圍內，一旦 IA 隨意網路的規模範圍擴大，大量基礎建設的佈署與設定將提高網路拓撲構築的成本。為了能在中型至大型 IA 無線網路中強化拓撲彈性和降低基礎建設佈署成本，本論文導入隨意網路架構(Ad Hoc Network)的觀念，本節將著重於介紹隨意網路的定義、結構，以及分群式隨意網路架構(Cluster-Based Ad Hoc Network)概念。

2.2.1 隨意網路定義

IEEE 802.11在定義無線網路時將無線網路分成兩種：具基礎建設的無線網路(Infrastructure Wireless Network)及無基礎建設的隨意無線網路(Ad Hoc Network)，以目前無線網路的發展情況而言，較多數的無線網路架構屬於前者。

具基礎架構的無線網路是讓無線裝置(mobile device) 經由基地台(Access Point, AP) 來連上網路，並透過其他設備，如交換機(switch)、路由器(router) 等裝置來幫助無線裝置上網，與其他使用者進行通訊，並使用網路服務。顯而易見地，在此架構中，所有設備節點均須仰賴AP來和其他節點溝通、或是連到Internet，這表示網路架構將被AP的傳輸訊號所束縛且缺乏拓樸彈性、且若AP因故失去功能，該AP所管轄的所有裝置節點將完全失去通訊功能。

為了解決Infrastructure Wireless Network的問題和限制，Ad Hoc Network，又被稱做Independent Network[10]應運而生。ACM (Association for Computing Machinery) 曾經對ad hoc無線網路做以下的定義[1]：A“mobile ad hoc network”(MANET) can be defined as a collection of nodes(or routers) equipped with wireless receiver/transmitters which are free to move about arbitrarily。這句話定義ad hoc無線網路為一群備有無線傳送/接收器且能任意自由移動的點或是路由器設備所組成，且彼此不需要AP等中樞裝置也可以運行，網路內部的無線裝置若在彼此訊號範圍內，則可做直接的通訊；各個節點也能隨意移動，並繼續保持與其

他無線裝置之間連線的狀態，如圖4所示。Ad Hoc Network的特性如下所述[10][27]：

1. 動態拓樸(dynamic topology)：Ad Hoc 網路內，各個無線裝置可以任意移動位置，而還能繼續和其他無線裝置做溝通。
2. 容易且可迅速佈建：這對於可能聚集少量電腦群組，且不需要存取其他網路的地方十分有幫助。
3. 具有自我組織(self-organization)能力，在沒有AP等中樞裝置的情況下、即使節點處於動態狀況下也可以進行通訊連結和網路管理，做理想的資源有效運用。

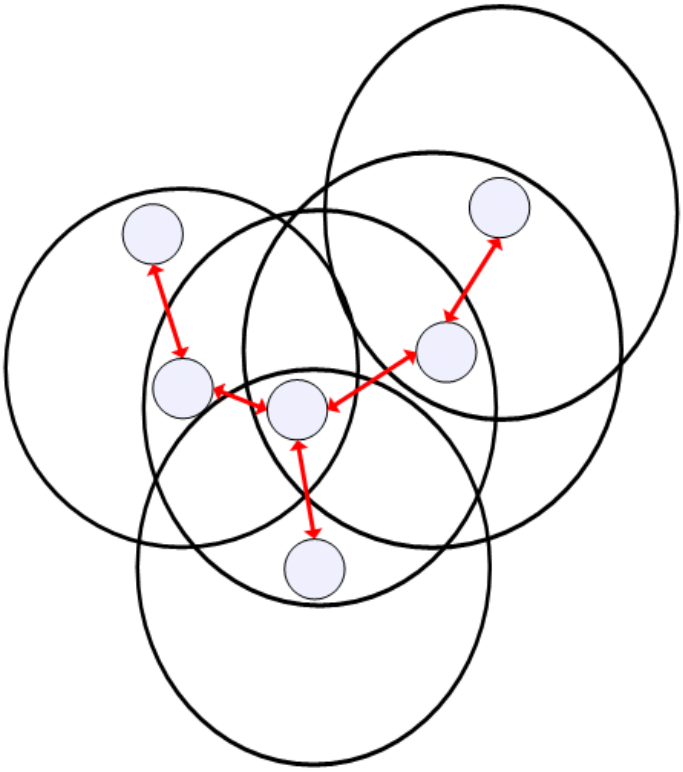


圖 4　Ad Hoc Network 架構

Ad Hoc無線網路是由許多無線裝置所構成，無線裝置之間要如何傳送訊息、溝通，是一個重要的議題。以OSI模型Layer2的觀點來檢視Ad Hoc Network運作的原理，由於節點之間的通訊乃藉由單一的共享頻道，通常會使用Multiple Access

with Collision Avoidance(MACA)協定，藉Request To Send/Clear To Send(RTS/CTS)交握(Handshake)動作以讓各節點在同一頻道進行非同步通訊時避免Collision的和Retransmission的發生[1]。若以OSI模型Layer3的觀點來檢視Ad Hoc Network之運作，有許多學者提出應用於Ad Hoc Network上的無線裝置邏輯拓樸架構[14][17]，分類過後大致為以下三種：

1. 由一個單一的憑證管理中心(Certificate Authority, CA) 來提供憑證，並進行認證的動作。憑證管理中心分派憑證給使用者，讓使用者之間能夠透過憑證來驗證彼此身份合法性，並安全的傳送訊息。
2. 將整個網路分成數個子網路(subnetwork)，組合成一個階層式的架構，每個子網路也有各自的子網路，分別形成一樹狀的架構，並互相合作來提供安全的服務，這樣的架構較適合提供給中型至大型的無線網路使用。
3. 將網路中的節點分成數個群集(cluster)的分群架構，此種架構又可分為兩種類型： without clusterhead 以及with clusterhead[6]。前者個個群集中沒有負責管理的特殊節點角色，每個節點相互協同以保持網路運作；後者每個群集中有各自的群集頭(cluster head) 及數個群集成員(cluster member)[18][22]，由群集頭來負責群集內部成員的管理，及訊息傳送與轉送等動作。由於網路在分群架構被分成多個小叢集，因此管理網路成員將更方便；而with clusterhead架構中成員的認證是由群集頭與每個成員個別來進行，不需透過multihop 傳輸，減少認證過程中遭受攻擊的可能性，

在未分群的Ad Hoc Network中，各個節點基本上必須記錄整個網路的資訊和到其他節點的路由資訊，因而造成負擔。這項問題在Cluster-Based Ad Hoc Network with Cluster Head架構中可被減輕：各群集中被推舉出來的cluster head可負責成員節點管理、路由資訊收集和安全認證議題[1][10][18][22]，當member 節點有特殊需求時(如路由或認證)，就直接向所屬的叢集管理者進行溝通，此法可以得到不錯的效率且更易於管理[27]。

基於Cluster-Based Ad Hoc Network with Cluster Head所能提供的管理面、功能面優勢，本論文將採用Cluster-Based Ad Hoc Network with Cluster Head作為分群機制的目標拓墣型態。以下將針對Cluster-Based Ad Hoc Network with Cluster Head相關文獻進行更進一步的探討。

2.2.2 隨意網路分群演算法

如何將整個隨意網路網路適當地分群與cluster head的選擇是cluster-based with cluster head的Ad Hoc Network架構兩項值得關注的議題，因為適當的分群架構將會使得路徑繞送的效能提昇、群集維持時間延長、節點間連通率的提高[14]。當網路拓樸有為小的改變時，分群架構只需要做區域性的拓樸資訊更新，如此一來更增進了網路的可掛載性 (scalablility) [6][20]。然而，如何行成最佳的分群架構是一個不易解決的NP-hard 問題[17][28]。

以下介紹幾種著名的 Ad Hoc Network 分群演算法，各種演算法均有其優點與缺陷，也可能依照不同的考量適用於不同的環境中。

2.2.2.1 Highest-Connectivity Cluster Algorithm

HCA(Highest-Connectivity Cluster Algorithm)，或稱作 Highest-Connectivity Heuristic[17]，主要利用節點的鄰居個數(Node Degree)大小來決定 cluster head 角色由何者擔任，其演算法步驟如下[15][19][28]：

- 每個節點廣播 beacon 封包讓周圍的鄰居節點接收
- 節點接收到鄰居的 beacon 封包後計算自身的 degree 值
- 經過一段時間後，所有節點將自己的 degree 值紀錄在 beacon 封包中，並廣播封包資料
- 擁有最大 degree 值的節點將被選為 cluster head(若 degree 值，則以最低 ID 優先當選)

- cluster head 周圍的鄰居將加入該 cluster head，且不得再參加 cluster head 的競選
- 前述步驟將反覆進行，直到所以節點都被分群完成，分群流程才算結束。
- 由於節點拓樸的移動性，每過一段時間需重新進行 HCA 流程，以更新節點資訊與選擇適當的 cluster head

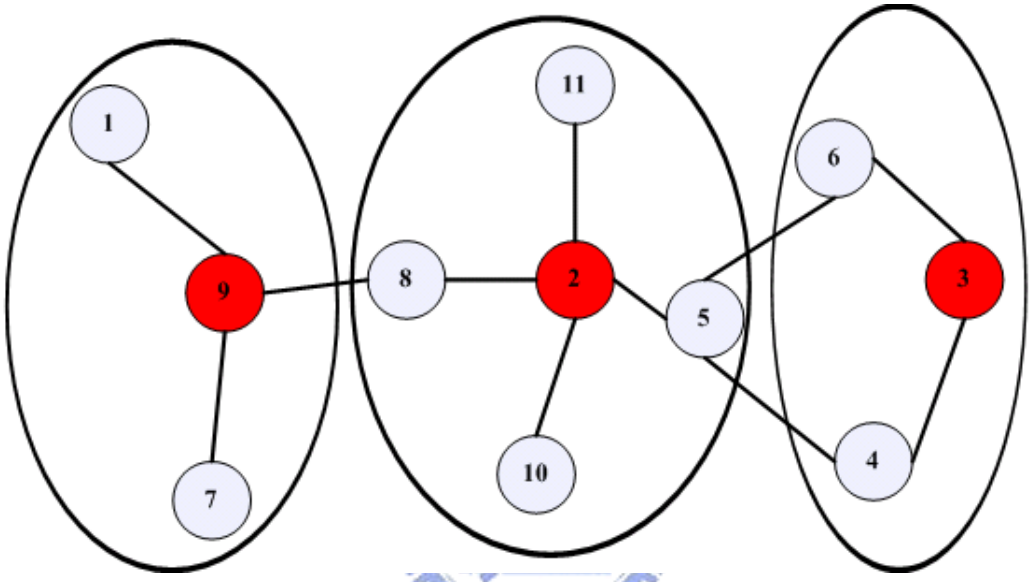


圖 5 HCA Ad Hoc Network

根據[13][19]，實驗數據顯示 HCA 分群結果下的 cluster head 更動率可有效降低，但系統整體的封包吞吐量(throughput)降低。一般而言，每個群集都會被配置有限的資源以供群集內部節點以 TDMA 的方式分時存取，若群集內部成員數增加，無疑會導致系統效能減低，HCA 以 degree 數來當作 cluster head 遴選標準的作法將更突顯此問題而造成分群功用不如預期。

2.2.2.2 Lowest-ID Cluster Algorithm

LCA(Lowest-ID Cluster Algorithm)，或稱作 Lowest-ID Heuristic[17]，主要利用代表節點的獨特 ID 編號大小來決定 cluster head 角色由何者擔任，其演算法步驟如下[17][19][28]：

- 每個節點廣播 beacon 封包讓周圍的鄰居節點接收，而 beacon 封包中含有該節點的 unique ID
- 經過一段時間後，各個節點比對所有鄰居的 ID，選出 ID 最小的鄰居當作 cluster head，並加入所選的 cluster head；若自己擁有最小 ID，則宣告自身為 cluster head
- 已加入 cluster 的節點將不得再參加競選
- 前述步驟將反覆進行，直到所有節點都被分群完成，分群流程才算結束。
- 由於節點拓樸的移動性，每過一段時間需重新進行 LCA 流程，以更新節點資訊與選擇適當的 cluster head

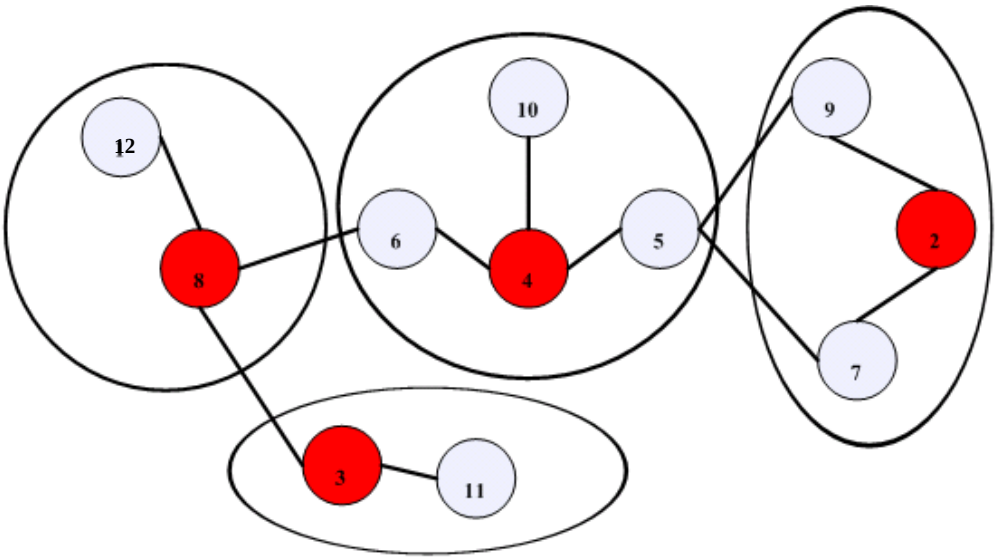


圖 6　LCA Ad Hoc Network

根據[19][28]，LCA分群程序選出的cluster head因ID-based的競選標準簡單又快速，且得以解決HCA帶來的系統效能降低問題。但cluster head本質上不具任何優勢(處理能力、剩餘電力、位置)，加上劃分出來的群集相當零亂，沒有規則性可言，還可能產生出較多的叢集個數[15]，由此可見LCA也帶來許多尚待解決的問題。

2.2.2.3 Battery-Energy Based Clustering Algorithm

BEBCA(Battery-Energy Based Clustering Algorithm)[9]，主要依據節點剩餘電量為主要分群考量，BEBCA 定義剩餘電能小於門檻值的節點為 bottleneck，為了不讓這些 bottleneck 成為 cluster head，節點競選 cluster head 的規則為彼此比較 one-hop 內的 bottleneck 節點數，bottleneck 鄰居節點數量越高者，越有機會成 cluster head，這種設計可有效避免網路上過多的 bottleneck 被選為 cluster head，造成網路拓撲的不穩定。如圖 7，節點 4, 6, 7 藉由鄰居的 beacon 資訊得知自己周遭的 bottleneck(灰色鄰居節點)較其他節點多，因此宣告自己為 cluster head，讓周圍的節點加入。然而，此演算法設計可能因為節點佈署位置而產生負面影響，如節點 4 附近擁有為數眾多 bottleneck，即使節點 4 本身能力不佳(節點 4 也為 bottleneck)，也將被選為 cluster head。以節點 4 為首的群集，其穩定性自然大打折扣。

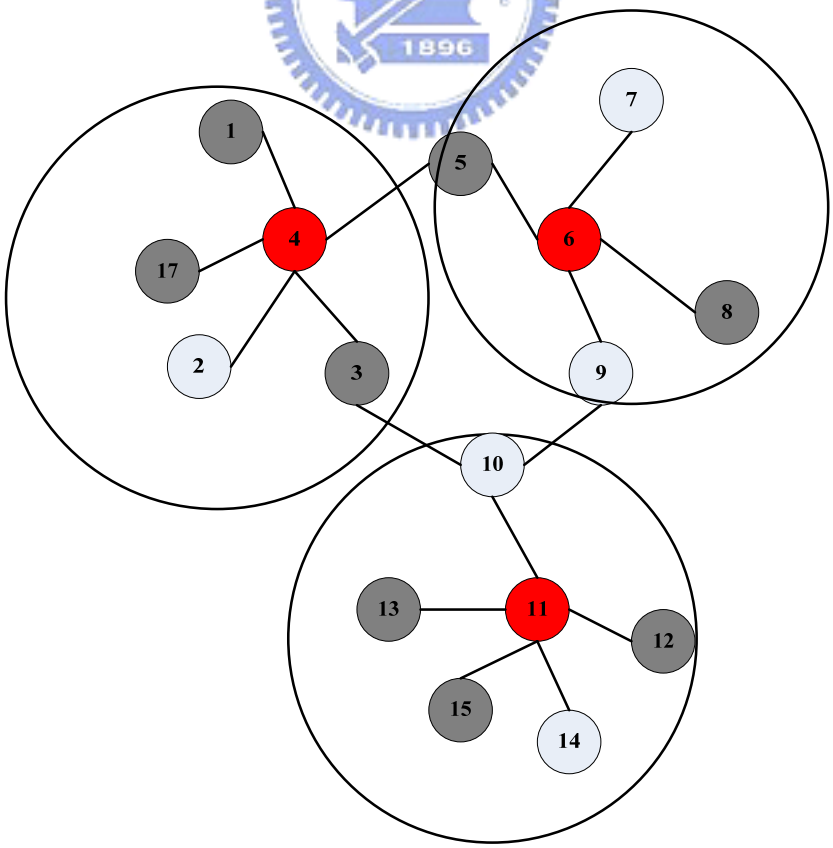


圖 7　BEBCA Ad Hoc Network

2.2.2.4 Weighted Cluster Algorithm

LCA 與 HCA 分別以 Node ID 及 Node degree 當作遴選 cluster head 之標準。WCA(Weighted Cluster Algorithm)，或稱作 Weighted Heuristic[17]，乃是將決定 cluster head 的標準改為加權分數的方式，得到最高加權分數的節點取得 cluster head 的資格。加權分數的計算方式通常會將多種系統變因加以考量[17][19][28]，比如 node degree、節點的剩餘能量、節點移動速度...等，每種變因被賦予一個權重，來表示該變因對於節點分群的重要性，而權重設定也可視環境的不同加以改變。

表 2 將針對 LCA、HCA、BEBCA、WCA 等四種 1-hop with cluster head 的分群方式作比較：

表 2 1-hop with cluster head 分群演算法比較表

	LCA	**HCA**	**BEBCA**	**WCA**
cluster head 遴選依據	id 大小	Degree 大小	鄰居 bottleneck 個數	加權分數高低
優勢	分群程序容易	cluster head 更動率較 LCA 為低	有效阻止 bottleneck 成為 cluster head	1. 可納入多項系統變因 2. 可調權重決定系統變因重要性
劣勢	cluster head 的選擇原理相當於隨機選取，無意義可言	cluster head 負擔較重	bottleneck 節點可能被選為 cluster head	加權分數計算較複雜，系統變因的選擇影響分群的結果好壞

節點自身能力考量	無	無	無	依照採用的系統變因而定

2.2.2.5 K-Hop Cluster Algorithm

前文所提到的HCA、LCA與WCA，在想法提出的當時，所建構出的叢集架構都是屬於1-hop 的，換言之，就是叢集管理者只負責管轄一步跳躍範圍的鄰居節點，如圖8所示。但是，如果在規模較大的網路狀況下，叢集的個數會因此變多，反而無法表現出叢集的優點，所以就有部分學者提出K-hop 的叢集架構演算法，用以滿足規模較大的網路需求。

Z. J. Hass 提出一種K-hop的ZRP(Zone Routing Protocol)[29]，此法把每一個節點都視為叢集管理者，各個節點都必須去紀錄其K-hop 範圍內的資訊，因此所紀錄的叢集資訊重疊性相當高，難免浪費太多的網路頻寬在維護叢集和繞送路徑資訊上。

在D. C. Su 等人所提出的叢集演算法[11]中，作者設計出一種計分機制去計算每個節點的權重大小，並依照此權重去選出較合適的節點當叢集管理者，這種方法可以讓每個叢集分布較均勻且有較好的效能，且消除ZRP重疊性帶來的問題，但是網路上的繞送卻較耗費時間。

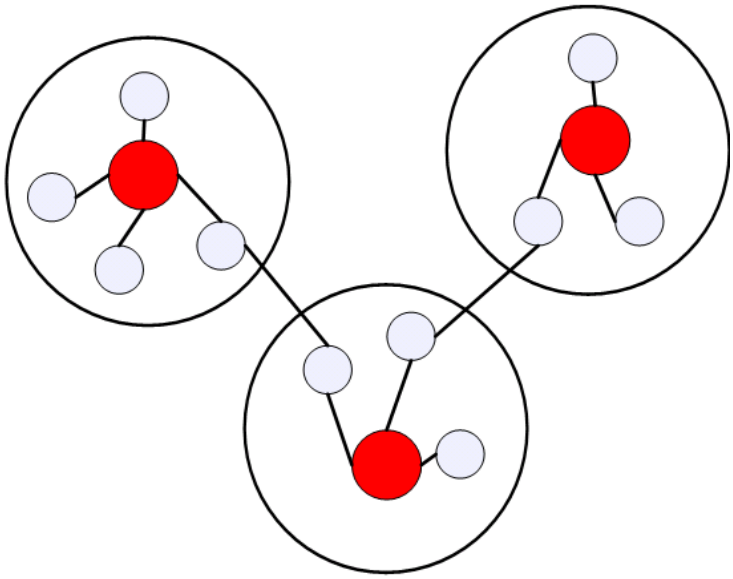


圖 8　1-hop Clustered Ad Hoc Network

2.3 無線感知網路技術

隨著時代的演進，科技技術不斷更新。微型製造、通訊及電池技術的翻新，促使小型感測器（Sensor）具有感應、無線通訊及處理資訊的能力。因此不但能夠感應及偵測環境的目標物及改變，並且可處理收集到的數據，並將處理過後的資料以無線傳輸的方式送到資料收集中心或基地台。

此類感測器多為微小及便宜的裝置，由於單一裝置的成本較小，因此可大量放置於環境中形成一個巨大而稠密的感測器網路以對環境進行即時目標偵測。一般而言，感測器通常為低能量，且電源不具補充性，所以當感測器的內部能源耗盡時，就必須予以拋棄。

WSN 並不界定網路拓撲型態，也就是可以為 star、mesh、P2P 或綜合以上型態的網路，但都一定具備下列的功能[24][28]：

1. Sensors/microcontroller：偵測、蒐集以及處理環境中的資料，例如偵測溫度或溼度。
2. Radio frequency：節點或 gateway 用以收發資料。
3. Software：包含在節點端的嵌入式系統以及使用者端的管理程式，軟體保

證資料感測的功能運行正常以及提供容易閱讀的介面。

為了節省傳輸時的能量消耗，無線感知網路中的裝置進行資料傳輸時，當基地台距離感測器太遠時，感測器需要利用網路路由 （routing）的方法將資料經由多個感測器組成的路徑傳回基地台，而不同路由方式與拓樸型態都將對於網路整體壽命將會有直接的影響。故目前無線感知網路的研究而言，多數有關路由或拓樸形成的研究都擺在延長網路整體壽命之議題。

2.3.1.1 ZigBee

2004 年底，Zigbee聯盟(Zigbee alliance)發佈了一種應用於無線感測網路的短距離傳輸的新標準。Zigbee 擁有低能源消耗、低資料傳輸率、低成本等特性，因此適合用於智慧型家庭網路、智慧型辦公室，並可應用於大樓自動化、醫療看護、能源控制和自動量測等，如圖9所示[8]。

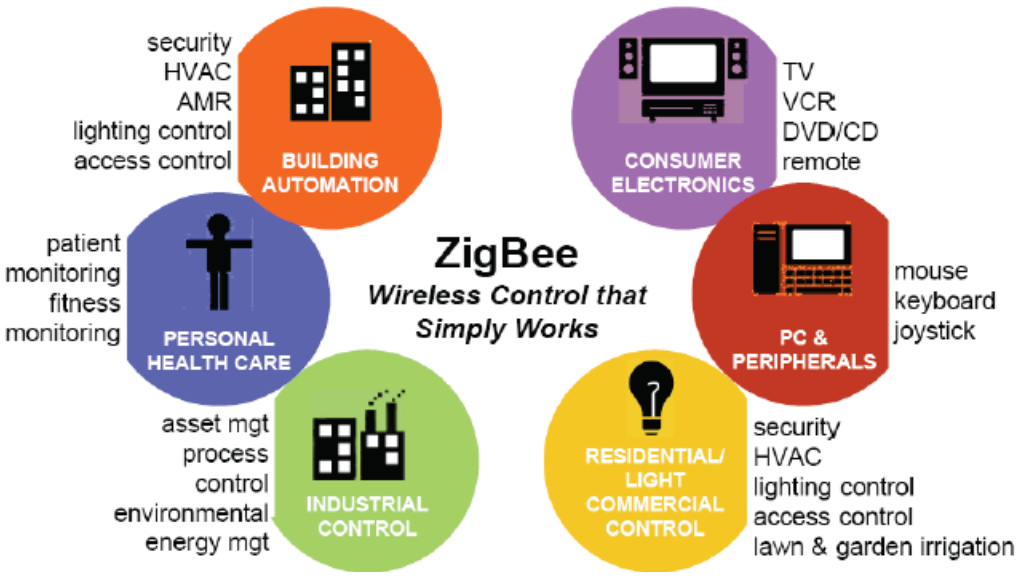


圖 9　ZigBee 應用層面類別[8]

在Zigbee的的規範中，設備依功能多寡可分為完整功能設備(Full Function Device)與部分功能設備(Reduced Function Device)二種。而依種類來分，可分為以下三種：

1. Coordinator：每一個Zigbee 網路必定要有且是唯一的coordinator，屬於

完整功能設備，負責初始化與管理整個網路，同時擁有Router 的功能。

2. Router：可能直接連接於Coordinator 或其他的Router，屬於完整功能設備，在網路中擔任Router 的角色，負責遞送資料與交換訊息。

3. End Device：可能直接連接於Coordinator 或Router，但不負責遞送資料與交換訊息，其設備可能附有感測裝置。

目前，ZigBee 技術在無線感知網路上的應用得到了較廣泛的支持，因此相較於其他技術規格開發陣營，ZigBee 可說是站穩了腳步。

2.3.1.2 Ultra Low Power Wi-Fi

縱使 ZigBee 技術已經在無線感知網路領域得到了較為廣泛的應用。但從長遠角度來看，Zigbee 在安全性、技術成熟度、節點管理、QoS 特性、設備互通性等方面都不夠成熟，而這些都是商業顧客群最需要的，因此難以運用在商業用途。但以上所提到的要素在 Wi-Fi 規格均可以得到良好的支持。

總體而言，在無線通訊規格的領域中，Wi-Fi 擁有許多 ZigBee 陣營所無法比擬的優勢，如基礎建設的普及率、QoS、安全性架構解決方案…等。然 Wi-Fi 旗下的 802.11 系列技術規格要進入無線感知網路市場最大的阻礙在於發送資料時的功耗過大，因此主導 Wi-Fi 的 Intel 公司在 2006 年將新業務開發部門中的 WSN 研發小組加以切割成立一家名為 GainSpan 的初創公司。經過 3 年的研究，GainSpan 在 2009 初發表”Ultra Low Power Wi-Fi”計畫所研發出的 SoC 晶片模組與配套的 Development Tool Kit[13]，此模組支持 802.11b/g，並成功降低發送數據時的功耗，而且提供 WPA 等級的網路安全性功能。

圖 10　Ultra Low Power Wi-Fi 晶片套件

把 Wi-Fi 用於 WSN 這類微型裝置互聯網路無疑是個吸引人的方式，相較於 ZigBee，Ultra Low Power Wi-Fi 能享受到正在被大規模部署的 Wi-Fi 網路所帶來的成熟的技術、各類層出不窮的 Wi-Fi 設備、和既有的網路設施與架構支援，對於 WSN 與一般網路的界接，Ultra Low Power Wi-Fi 的地位將可能超越 ZigBee。

第三章 一個基於資訊家電的無線隨意區域網路分群機制

首先，本章先對 IA 所組成無線隨意網路(以下簡稱 IA 隨意網路)進行定義，接著描述 IA 隨意網路的構築與使用情境以及定義本論文對於 IA 隨意網路主要探究的問題，然後對於 IA 隨意網路分群機制運行的各個流程環節作詳細介紹。

3.1 IA 隨意網路定義

若把無線隨意網路(Wireless Ad Hoc Network)與無線感知網路(Wireless Sensor Network)互相進行比較，兩者有許多相似之處：首先，無線隨意網路和無線感知網路都是由為數眾多的節點構成，而節點通訊時均是採用無線界面，此外，商雙方節點之間的溝通可採 multi-hop 的方式進行，每個節點距有轉發封包的功能，而不需依賴特定路由器角色來幫忙轉發。

雖然有以上的相似點，但是無線隨意網路和無線感知網路的用途和硬體能力有所不同：無線隨意網路中節點的硬體能力與製造成本都遠高於無線感知網路中任一感測器節點；反觀無線感知網路的節點，往往是由微小且便宜的裝置，只具備簡單的資料傳送能力與有限的計算能力，而且為了壓低單一感測器的製造成本，感測器的內部電源供應採用不可更換的鋰電池，所以當感測器的電源耗盡就須丟棄。

當 IA 的普及率在後 PC 時代大幅提升後，若能將廣大範圍內的眾多 IA 套用隨意網路和無線感知網路的概念，勢必可形成一個大規模的 IA-Based 隨意透通性互連架構，此網路拓撲架構的特性與前文介紹的無線隨意網路和無線感知網路相比較，可發現具有一定程度的差別：

1. **硬體功能：**

IA 的運算模組和網路傳輸模組，雖不能與 PC 或膝上型電腦相比，但是比起成本和體積都壓在底限的無線感知網路裝置節點來說，功能的延展性要提高許多。

2. **異質性：**

如同前文所述，IA 的規格未有統一的型態，且依照[16]，IA 可被分類為許多種類，不同種類的 IA 其功能性與成本相差甚遠，在體積大或單價高的 IA 設備上，資料處理能力和網路傳輸功能就可加以強化；但在微小或低成本的 IA 設備上，資料處理能力和網路傳輸功能很有可能受限於體積或成本因素。因此相較於無線感知網路中節點具有高同質性，IA 隨意網路中的節點彼此功能性的差異極大。

3. **移動性：**

在無線隨意網路的環境中，網路節點多數為移動式設備，拓墣在每一個時間點的變化性極大；但 IA 設備通常不會隨意移動、即使重新配置 IA 位置，時移動性相對於無線隨意網路較低，因此在 IA 隨意網路中，移動性議題不會是阻礙的因素。

4. **識別性：**

通常在無線感知網路環境中，拓樸中的節點數量是無線隨意網路數百至數千倍，節點將 sensor 所感應到的資料後送至 sink 節點，再由 sink 節點傳送至後台資料統合中心，在此種運作模式下，同質的 sensor 節點不重視彼此之間的溝通，只須將資料送至中心即可；但在 IA 隨意網路中，IA 設備並非重複執行一成不變的簡單任務，考量到日後基於 IA 隨意網路拓樸的各項應用服務，IA 隨意網路須具有點對點溝通傳輸的能力，為達到此點，IA 節點彼此間在網路層須要有如同 IP 地址的共同識別證，以辨認網路中其他各別節點的身分，以及儲存與其溝通的路由資訊。

5. 安全性：

現今無線感知網路與無線隨意網路的運用範疇絕大部分僅限於工業或研究用途，成功的商業化應用服務屈指可數，追究原因，安全性便是主因之一。商業顧客無法忍受網路技術背後所隱藏的網路安全危機；因此，若想提高 IA 隨意路拓樸概念在現實上的可行性，就必須兼顧網路安全要素。

表 3 為無線感知網路、無線隨意網路和 IA 隨意網路的比較整理：

表 3 無線感知網路、無線隨意網路、IA 隨意網路比較表

	無線感知網路	**無線隨意網路**	**IA 隨意網路**
節點數量	大	小	中
裝置成本	低	高	中~高
節點裝置異質性	小	中	大
硬體功能	弱	強	中
節點移動性	低	中~高	低
拓樸改變性	小	大	小
節點識別	不需要	需要	需要
持續電源	無	不定	不定

IA 隨意網路的議題在目前可說是一個嶄新的概念。它將原先小規模的智慧型家庭、智慧型辦公室功能性繼續向前延伸，讓後 PC 時代中的產品與無線網路真正產生結合的契機。

3.2 IA 隨意網路使用情境

IA 設備普及的一個前提下，一個組織機構的佔地範圍環境中座落各式各樣的 IA，而 IA 設備均採用功率可調式資料傳輸模組、並且被賦予獨特不重複的 ID 編號。構築 IA 隨意網路拓樸之前，機構內所有 IA 裝置須先向該機構 WPS Server

進行 WPS 註冊流程，並在 WPS 註冊程序完成後得到分群用的安全密鑰資訊，然後將 IA 裝置放在任意處。

當 IA 設備準備就緒，網路管理者啟動 WPS Server 的分群功能，WPS Server 發出分群指令封包，讓已註冊過的 IA 設備開始分群動作，一但分群動作啟動，IA 設備利用 beacon 封包通訊的方式，執行本論文所提出的節點分群演算法，並為每一個群集選出適當的領導者節點(Leader Node)，最終構築完成一個 one-hop with-cluster head 的大型安全性無線隨意網路架構，如圖 11。

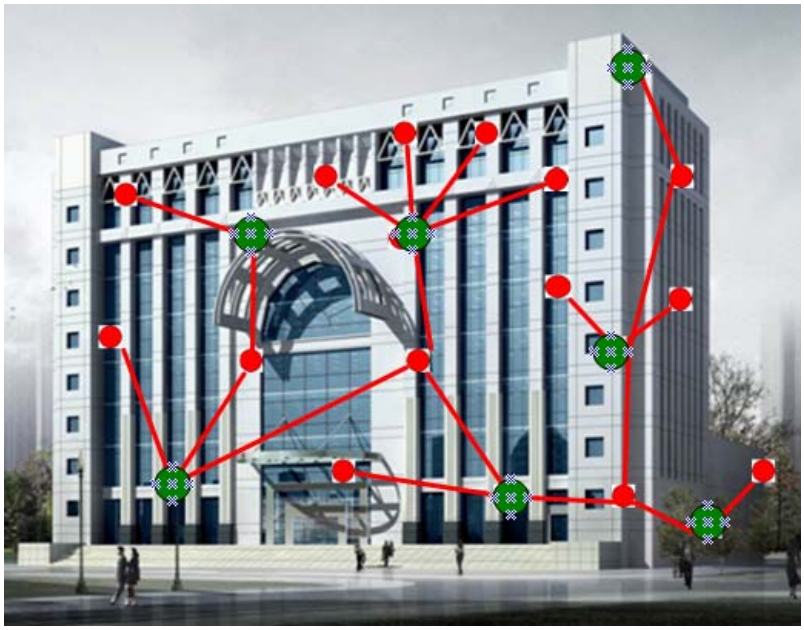


圖 11　組織機構內的大型 IA 隨意網路

應用 one-hop with-cluster head 分群架構於 IA 無線隨意網路情境的原因在於不同 IA 裝置其硬體運算能力或內藏電能並不相等，所以讓能力較強或是電能較可以持久的 IA 來擔任群集管理者(Leader Node)，如此能力或電能不足的 IA 裝置就不需負擔過多的工作。另外，因為 Leader Node 所管轄的 IA 裝置都在 1-hop 的範圍內，Leader Node 在執行管理工作時將比較容易。

Leader Node 所擔負的責任較一般節點來的重大，主要有下面幾項功能：

1. 擔任跨群集溝通的協調者角色：以群集中其他角色的角度來看，Leader Node 等同於執行一般網路上 router 的工作；以整個拓樸中所有 Leader Node 的角度

來看，它們互相串連成為無線隨意網路中的路由虛擬骨幹(Virtual Backbone)。

2. 擔任群集的 Authenticator：當有新節點欲加入該 Leader Node 控管的群集時，Leader Node 必須對此節點進行認證的動作，以為網路安全進行把關。新節點的認證程序通過之後，才得以加入群集，成為群集中的成員節點，此時，Leader Node 將替新成員節點進行 Key Management 的程序。
3. 其他管理任務：如群集中節點傳輸在 MAC 層的 TDMA Scheduling、群集節點 QoS 等、上層服務應用權限管理，此類議題不在本論文的討論範圍之內。

IA 隨意網路應用的情境中，常常發生點對點溝通的情境，舉例來說，停車場的大門偵測到公司於員工準備進入建築，就與特定房間內的冷氣設備溝通，調整溫度。考量到以上訴求，網路拓樸成型後，IA 節點可憑藉 ID 尋求特定 destination 之路由資訊，而不是像無線感知網路拓撲中的感測裝置，只能將資訊回傳給特定的節點；在安全性議題方面，IA 隨意網路的安全議題遠比工業或研究用途的無線感知網路重要，試想前述停車場大門與冷氣節點的情景，若有不法節點侵入 IA 隨意網路，並假冒合法節點的名義對其他 IA 下達指令，很可能對使用者造成負面的影響。然而，現實中 IA 使用者可能不具備資訊安全與電腦知識，加上多數 IA 節點無法擁有如個人電腦或膝上型電腦般完整的操作與顯示介面，使得 IA 裝置安全性設定在一些使用者眼中變成燙手山芋。為解決安全性問題， IA 隨意網路架構導入 WPS 裝置註冊流程，令 IA 裝置進行安全性組態設定變得自動化且便捷；如果使用者添購新的 IA 設備，只要向最接近的 Leader Node 進行 WPS 註冊流程，獲得認證金鑰後，就可將新設備加進 IA 隨意網路中。

與以往智慧型家庭、智慧型辦公室網路相比，IA 隨意網路可建立具規模性的拓樸、且不受限於無線 AP 或路由器的優勢；與無線感知網路相比，IA 隨意網路則增強安全性與點對點互通性。

3.3 問題定義

IA 隨意網路拓樸的特性與以往的 multi-hop 無線網路有所不同，故以往研究所提出的網路拓樸構築流程不適合直接套用於 IA 隨意網路。本論文的主要目的在於設計以 IA 裝置特性為考量，並兼具安全性、穩定性與方便性的無線隨意網路分群機制，而其中包括一個有效的分群演算法。演算法在網路中選出擔任 Local Authenticator 和 Manager 角色的 Leader Node，而 Leader Node 身旁的 IA 節點將進行加入動作而形成群集。此流程在設計上主要有三個考量：在安全性方面，須盡可能保證分群流程不受惡意攻擊影響，在穩定性方面，拓樸構築流程中 Leader Node 選擇演算法須針對 IA 隨意網路的用途與特性選擇電源充足的節點為適當的 Leader Node，以維持網路的穩定運作；在方便性方面，架構採用 WPS(Wi-Fi Protected Setup)標準讓使用者輕易將新 IA 設備做好安全性設定，並註冊至網路中，以期增加本論文提出之架構實務上的可行性。本論文所提出的方法適用於中型至大型的 IA 隨意網路區域，如企業機構佔地或是整體校園網路。

3.4 基於資訊家電的無線隨意區域網路分群機制

本論文所提出的 IA 隨意網路分群機制，其內容將涵蓋從初始 IA 設備的註冊直至拓樸網路構築完成為止，主要可分為三個階段，如圖 12 所示：

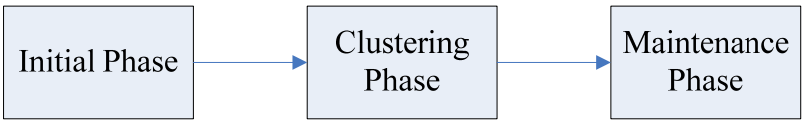


圖 12　分群架構主要流程

以下的章節將介紹本論文所提出之 IA 隨意區域網路分群架構各階段的流程細節：

3.4.1 Initial Phase

欲進行IA隨意區域網路分群架機制的機構須先備有一台WPS Root Server，在初始階段，WPS Root Server 擔任整個網路中最初的 WPS Registrar 角色，故又稱做 super registrar。在建構分群隨意網路拓墣之前，機構中所有的 IA 利用 WPS Root Server 進行初始註冊，初始註冊時 IA 提供 WPS 程序所需的 Pin code，以宣告自己的合法身份。WPS 註冊程序結束後，IA 取得藉由 M8 訊息取得 Credential，Credential 中包含 secret 和 g_session_key 待之後的分群程序使用。

Initial Phase 的資料交換流程相對單純，交換過程所用到的參數說明如表 4 所述：

表 4 Initial Phase 參數表

參數符號	**說明**
||	參數值作 concatenation 動作
RS	WPS Root Server
IA_n	任一 IA 設備
PIN	IA 隨機產生的 Pin code
g_session_key	跨群組的 global 金鑰
certificate	WPA 認證金鑰
secret	暫時性秘密參數值
WPS(M1,M7)	WPS Registration Protcol 中的 M1 至 M7 訊息
Credential{ secret||g_session_key||certificate}	M8 訊息中附的安全性證書，證書中包含 secret、g_session_key

Initial Phase 的訊息交換流程如圖 13：

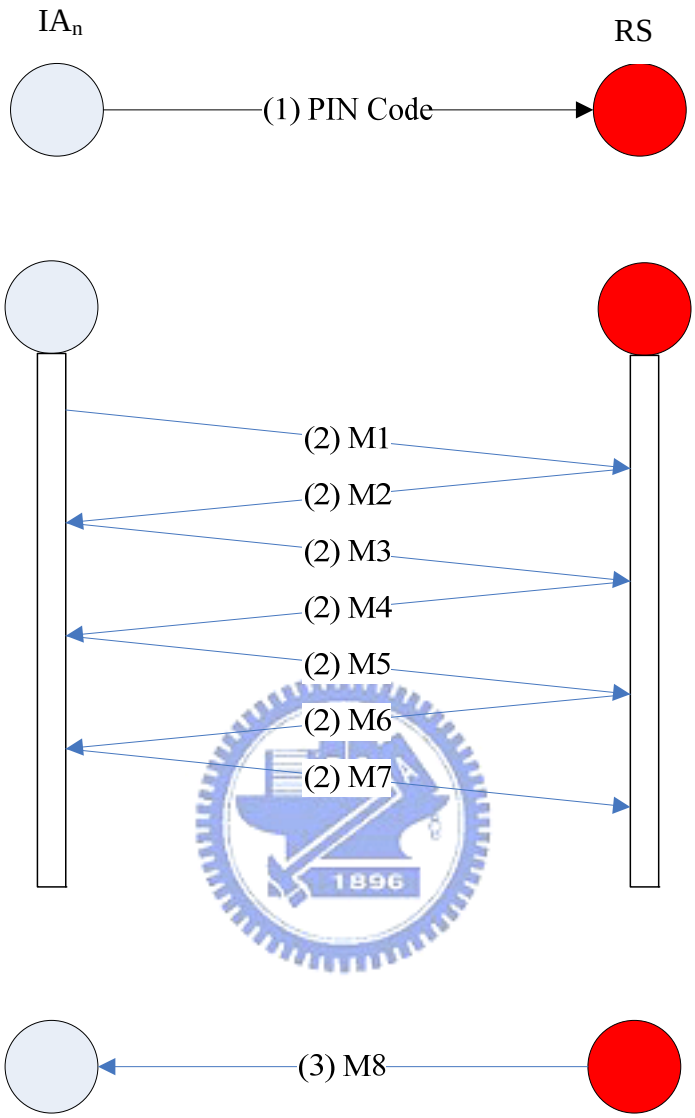


圖 13　Initial Phase 訊息交換

IA_n -> RS: PIN Code	(1)

(1)：使用者將 IA 的 PIN code 輸入 WPS Root Server

IA_n <-> RS: WPS(M1,M7)	(2)

(2)：IA 與 WPS Root Server 進行 WPS Registration Protcol 中 M1 至 M8 訊息交換，WPS Root Server 在(2)中驗明 IA 設備合法性

RS -> IA_n: M8{ Credential{ secret|| g_session_key||certificate }}	(3)

(3)：WPS 註冊程序結束後，WPS Root Server 確認 IA 的合法性，在 M8 訊息中授予 IA 分群程序必須的秘值 secret 和 g_session_key，certificate 則是 IA 節點以後存取網路的 WPA 認證金鑰

3.4.2 Clustering Phase

所有 IA 在 Initial Phase 進行完註冊流程後，使用者將其配置在任一地點。之後，網路管理者啟動 WPS Root Server 的分群功能，WPS Root Server 向全體 IA 進行廣播，宣告進入 Clustering Phase。Clustering Phase 當中又可細分為兩個步驟：

1. **Authentication:** 在此步驟中，各 IA 節點須確認身旁的鄰居節點完成 Initial Phase 程序，以確認其合法性。
2. **Election:** 在此步驟中，各 IA 節點利用鄰居發出的 beacon 訊息，進行本究所提出的 MyAgorithm，計算本身的加權分數 Election Score，以 Election Score 判斷自己在所處的區域內是否適合擔任 Leader Node。

Clustering Phase 的訊息交換流程所用到的參數說明如表 5 所述：

表 5 Clustering Phase 參數表

參數符號	**說明**
||	參數值作 concatenation 動作
N{}	網路中所有 IA 節點的集合
N_i, N_j	網路中任二 IA 節點 i, j
RSID	WPS Root Server 的 ID
NID_i, NID_j	網路中任二 IA 節點的 ID

$Neighbor_i\{\}$, $Neighbor_j\{\}$	網路中任二 IA 節點 i, j 的鄰居節點集合
ID_{WPS}	WPS Root Server 的 ID
Timestamp	不重覆的時間戳記值
GLOBAL_IAM	GLOBAL Initial Authentication Message 全域初始認證訊息
IAM1	Initial Authentication Message 1 初始認證訊息 1
IAM2	Initial Authentication Message 2 初始認證訊息 2
g_i, n_i, g_j, n_j	網路中任二 IA 節點 i, j 的 Diffie-Hellman 金鑰協議參數
$session\_key_{ij}$	網路中任二 IA 節點 i, j 之間溝通的金鑰
X_i, X_j	網路中任二 IA 節點 i, j 所選定的大數值，用於進行 Diffie-Hellman 運算，產生 $session\_key_{ij}$

3.4.2.1 Authentication

Authentication 步驟的訊息交換流程如圖 14：

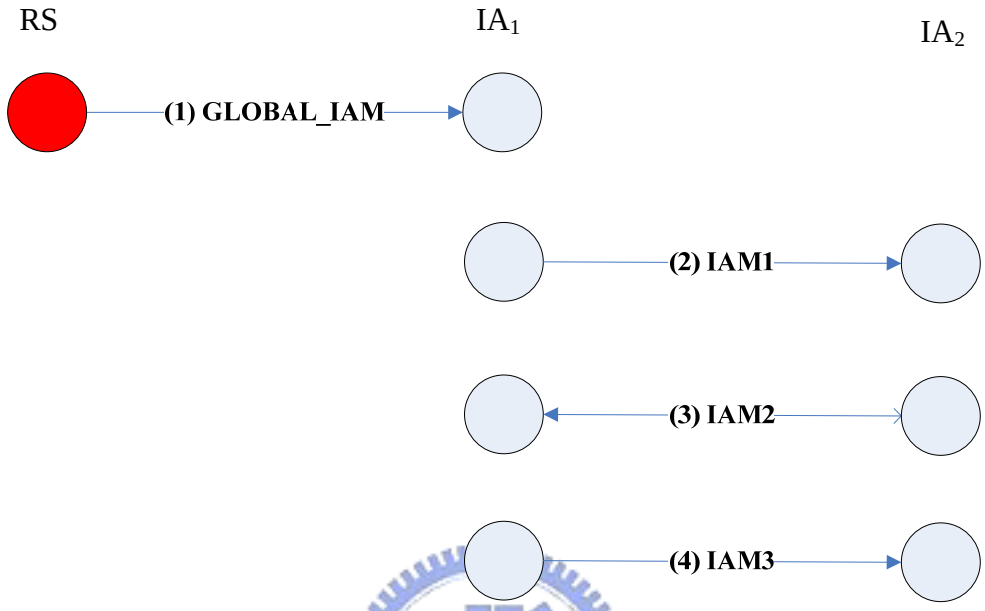


圖 14　Authentication 訊息交換

RS->N{}: GLOBAL_IAM{RSID||g_session_key(RSID||Secret||Timestamp)||Timestamp}	(1)

(1)：當網管人員啟動 WPS Root Server 的分群功能，Server 將發出 GLOBAL_IAM 給周遭的 IA 節點(如圖 13 中 RS 發送給 IA_1)。其中，Secret 和發送 GLOBAL_IAM 時的時間戳記 Timestamp 將被 g_session_key 所加密。當周遭的節點接收到 GLOBAL_IAM 後，利用 g_session_key 解開加密的欄位，確認分群指令無誤，再將 GLOBAL_IAM 訊息傳播下去，直至整個網路中的 IA 節點都收到。

N_i -> $Neighbor_i${}:

IAM1{ NID_i|| g_i||n_i || X_i ||$HMAC_{g\_session\_key}$(NID_i||g_i||n_i ||X_i ||$Timestamp_i$)||$Timestamp_i$} (2)

(2)：分群作業開始之後，每個節點向周圍鄰居發送 IAM1(如圖 13 中 IA_1 發送給 IA_2)，IAM1 的欄位包括發送者 ID、Diffie-Hellman 金鑰協定所需用到的參數，以及發送時的時間戳記，上述資料將被 g_session_key 進行 HMAC 雜湊。

N_j -> N_i: IAM2{ NID_j ||N_j ||$HMAC_{session\_keyij}$(NID_j|| X_j||$Timestamp_j$)||$Timestamp_j$} (3)

(3)：當鄰居節點 N_j 收到來自 N_i 的 IAM1，先用 g_session_key 驗證 HMAC 無誤，然後同樣選定 Diffie-Hellman 金鑰協定參數，連同自己的 ID 和當時的時間戳記進行 HMAC 雜湊，附在 IAM2 中回傳給 IAM1(如圖 13 中 IA_2 發送給 IA_1)。

N_i -> N_j:

IAM3{NID_i|| $session\_key_{ij}$ (NID_i||Secret|| $Timestamp_i$)||

$HMAC_{session\_keyij}$ (NID_i||Secret|| $Timestamp_i$)} (4)

(4)：N_i 收到 IAM2 後，一樣用 g_session_key 驗證 HMAC 無誤，然後利用雙方的 Diffie-Hellman 金鑰協定參數產生彼此的 Session Key；而後，N_i 將 Session Key 記在 Key Table 中，然後回復確認訊息 IAM3 給 N_j，訊息中包含 N_i 的 ID、Secret、時間戳記，以及三者的 HMAC(如圖 13 中 IA_1 發送給 IA_2)。

Authentication 的動作結束以後，網路上每一個節點都應維護一個 key table，table 中記錄所有鄰居和自己共同生成的 Diffie-Hellman Session Key。

表 6 Key Table

NID	1	5	57	32	7	9
Key	Key1	Key5	Key57	Key32	Key7	Key9

Authentication 整體的流程如圖 15 所示。

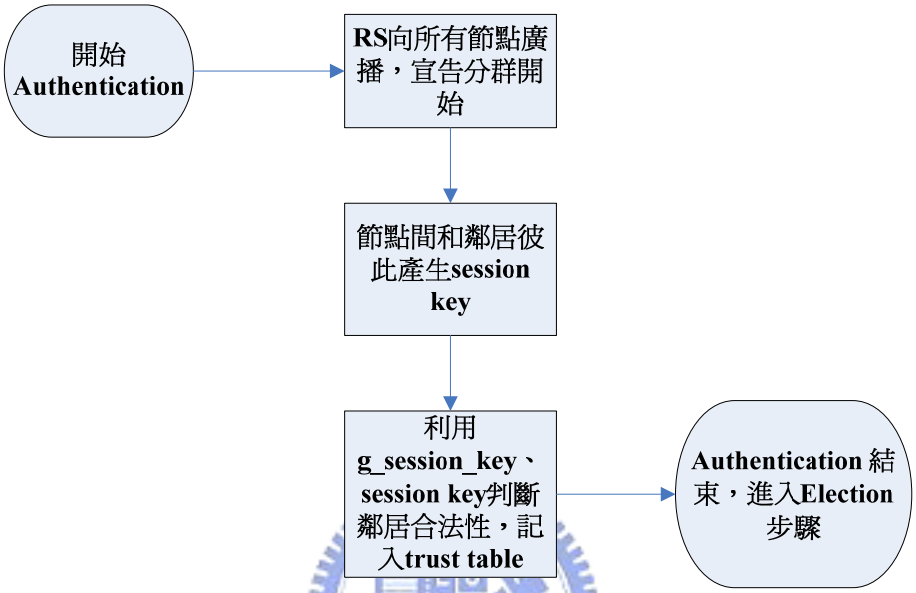


圖 15 Authentication 流程圖

3.4.2.2 Election

Authentication 作業結束之後，網路中的所有 IA 節點此時已確立鄰居節點的合法性，而進行 Election 步驟時，IA 節點要根據本論文所提出的考量因子為依據，選擇出適當的 Leader Node。

Leader Node 的角色定位，除了須擔負一般分群隨意網路群集中的 Cluster Head 的責任外，同時也擔任地區性的 WPS Registrar。日後若有新的 IA 節點欲加入此 Leader Node 所負責的網群集，則必須向該 Leader Node 進行 WPS 註冊程序並進行認證，認證程序成功後才得以存取網路。

在 IA 隨意網路中，IA 的電力供給方式因設備種類的不同而異，具有插座持續電源供給的 IA，其通訊傳輸模組若能從插座持續獲得電源，則該 IA 節點就不

需擔心因過度傳輸資料而造成電源耗盡，被迫離開隨意網路；相反的，若 IA 設備受限於體積、功能性等因素，使通訊傳輸模組所能獲得的電源有限(如安裝鋰電池)，則該節點對於資料傳輸的使用率上就必須加以斟酌。以 1-hop with-cluster head 的隨意網路拓撲來說，可明顯看出 Leader Node 與一般的節點所負擔的責任並非對等，Leader Node 必須花費更多的運算功能與封包傳輸以維持群集的運行。因此，功能性不足或是電力無法持續供給的 IA 節點，相較於功能性強大或電源充足的節點來說，較不適合擔任 Leader Node 的角色。

基於各個 IA 節點功能性與電源供給條件不相等的情境下，本論文提出名為 IA based Ad hoc Network Clustering Algorithm (IAdNCA)的分群演算法，根據 IA 隨意網路的特性來選出適當的 IA 節點當作區域群集的 Leader Node。IAdNCA 之遴選方式以各個節點的 Election Score 作為選擇標準，Election Score 越高，代表該節點各方面因素的綜合考量越具優勢；反之則代表該節點的綜合考量不適合擔任 Leader Node。

3.4.2.2.1 Seed Leader Node Election

首先，環境中所有節點都會持續收到鄰居節點所發出的 beacon 封包，beacon 的格式如(1)

beacon{ NID||LNID||Seed|| $Flag_{Energy}$ ||Energy || ElectionScore||Timestamp}	(1)

NID 代表發送 beacon 之裝置的 ID 編號，此 ID 必須是唯一不可重複；LNID 代表該 IA 節點的 Leader Node 編號，若 LNID 為 0 則代表此節點目前處於未分群(unclustered)狀態；Seed 為一布林值，如圖 16，若 Seed 為 1 代表此 IA 節點評估自己有能力擔任 Leader Node，推舉自己成為種子領導者節點(Seed Leader Node)，

若為 0 則代表 IA 節點並未做出推舉自己的動作，必須參加之後的選舉競爭 Leader Node 資格，至於 IA 節點是否具有 Seed Leader Node 能力，可由 IA 製造商在產品出場前設定、或由使用者自行設定。$Flag_{Energy}$ 亦為一布林值，若為 1，代表發送 beacon 的節點擁有持續電源供應(如插座)、若為 0，則代表電源有限；Energy 欄位代表節點的剩餘能源量；ElectionScore 為 Normal Leader Node Election 執行時發送 beacon 的節點所得到的加權分數，預設值為 0；Timestamp 為發出 beacon 封包的時間戳記。

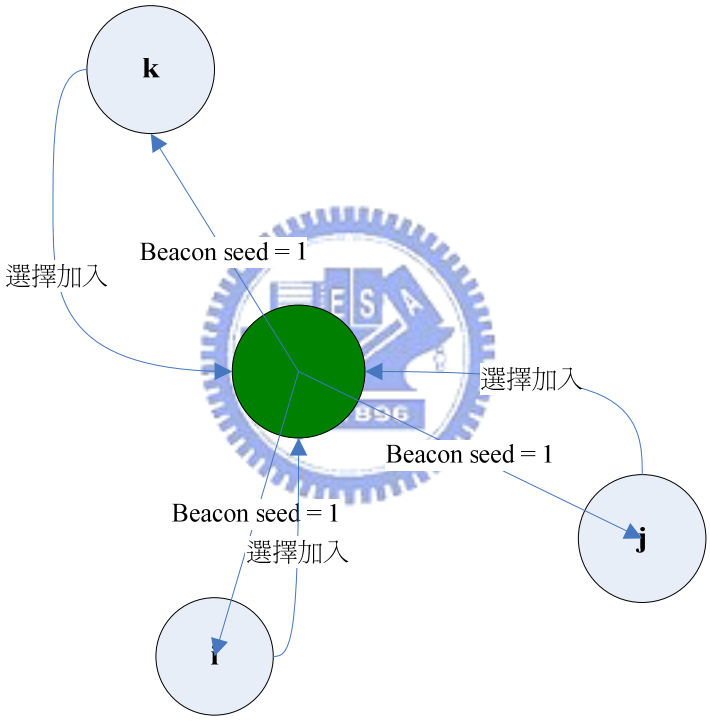


圖 16　Seed Leader Node

Seed Leader Node 的周圍鄰居收到 Seed 為 1 的 beacon 後，發送 JOIN 封包表示加入該節點所開創的群集的意願(如圖 17)，JOIN 封包格式如(2)：

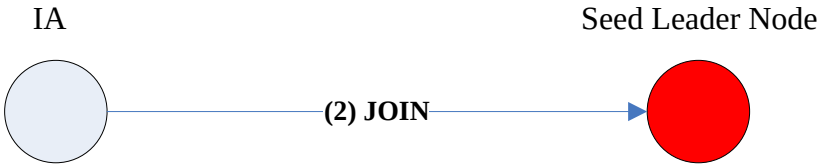


圖 17　JOIN 封包發送

JOIN{ NID||LNID||session_key(NONCE||Timestamp||Key_{WPA})||$HMAC_{session\_key}$(NID|| Key_{WPA} ||Secret||Timestamp)||Timestamp}	(2)

JOIN 封包格式中，NID 代表發出 JOIN 封包節點之 ID、LNID 代表欲加入的 Seed Leader Node ID、Nonce 為隨機亂數值、而 Timestamp 為 JOIN 發送時的時間戳記，Key_{WPA} 為 WPS 註冊程序後從 WPS Root Server 得到之 Credential 中的 WPA Pre-Share Key，Nonce、Key_{WPA} 與 Timestamp 將被發送者與 Seed Leader Node 共有的 session_key 加密而附在 JOIN 封包中，HMAC 同樣以 session_key 產生以維護完整性。若某節點收到多於一個 Seed 值為 1 的 beacon(圖 22 節點 21)，該節點將選擇 becon 訊號強度較強者，也就是距離較近者進行 JOIN 程序(此例中節點 22 將選擇節點 10)。

發出 Seed 為 1 之 IA 節點收到某鄰居節點 i 的 JOIN 封包後，使用 Authentication 步驟中和該鄰居共同產生的 Diffie-Hellman $session\_key_{i,SeedLeader}$，取得被加密的隨機 Nonce 值，然後向該節點 i 發送 GRANT 封包以代表同意節點 i 的加入(如圖 18)，GRANT 封包格式如(3)：

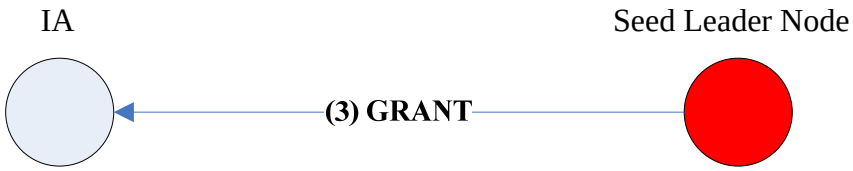


圖 18　GRANT 封包發送

GRANT{LNID|| $session\_key_{i,SeedLeader}$(NONCE|| Timestamp)||Timestamp}	(3)

LNID 代表 Seed Leader Node 之 ID、NONCE 為前次從節點 i 的 JOIN 封包取得之隨機亂數值、Timestamp 為發送 GRANT 封包時的時間戳記，NONCE 和 Timestamp 將被 $session\_key_{i,SeedLeader}$ 加密，只有同樣握有 $session\_key_{i,SeedLeader}$

節點 i 收到後可加以解開，解開之後取得 NONCE，若與當初發送的值一樣，代表此 Seed Leader Node 為合法節點，節點 i 更改自身的 LNID 為 Seed Leader Node 的 ID，宣告已加入別的 Leader Node 群集。

若 Seed Leader Node 的地點分布不均勻，造成兩個 Seed Leader Node 彼此在通訊範圍之內，如同圖 22 中的節點 10 與節點 11，則有一方必須放棄 Seed Leader Node 身份，成為普通節點。節點 10 和 11 將互相朝對方發送 CONLICT 封包(如圖 19)，封包格式如(4)：

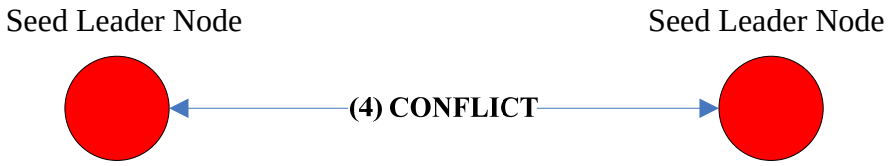


圖 19 CONFLICT 封包發送

CONFLICT{NID||Degree||$HMAC_{session\_key}$(NID||Degree||Timestamp)|| Timestamp}	(4)

NID 為發送 CONFLICT 封包的節點 ID、Degree 為發送節點的鄰居聯結支度、Timestamp 為時間戳記，這三者將用發送方與接收方(此例中為節點 10 與 11)共有的 Diffie-Hellman 金鑰作 HMAC 雜湊。

當雙方收到 CONFLICT 封包後，比對封包中的 Degree 和自己的 Degree，若自己的 Degree 較高，則發出 KEEP 封包表示欲保持 Seed Leader Node 的地位，若 Degree 數相同則以 ID 較小者維持 Seed Leader Node 地位。KEEP 封包傳送如圖 20，其格式如(5)

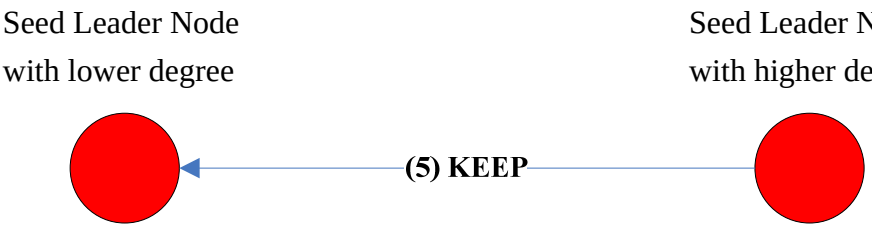


圖 20　KEEP 封包發送

KEEP(NID|| $HMAC_{session\_key}$(NID|| Timestamp)||Timestamp)	(5)

KEEP 封包中 NID 與 Timestamp 的意義與 CONFLICT 封包相同，故在此不再贅述。圖 23 中，節點 10 的 Degree 為 3，而節點 11 的 Degree 為 2，因此節點 10 會發出 KEEP 封包告知節點 11；若在 CONFLICT 的比較程序結果出來之前，已經先有普通節點加入 Degree 較少的 Seed Leader Node(如圖 23 節點 9 先加入節點 11)，則 CONLICT 競爭中落敗的 Seed Leader Node(如圖 23 節點 11)必須發送 CANCEL 封包告知加入的節點，令其回復到 unclustered 狀態(如圖 21)，CANCEL 封包格式如(6)：

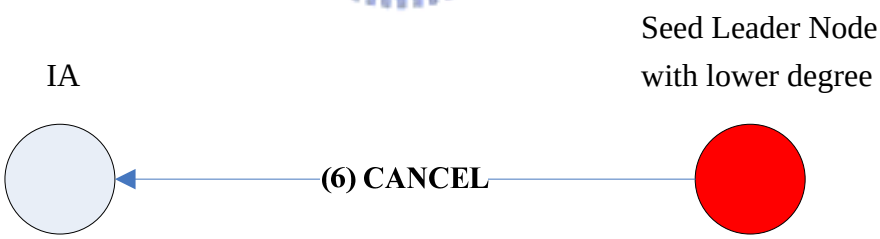


圖 21　CANCEL 封包發送

CANCEL(NID|| $HMAC_{session\_key}$(NID|| Timestamp)||Timestamp)	(6)

CANCEL 封包裡的欄位屬性 NID、Timestamp 之意義與前面的 KEEP 封包相同，而 HMAC 用的金鑰則是 Seed Leader Node 和先加入的節點之共有金鑰。

最後，落敗而回復成普通節點的 Seed Leader Node(圖 23 節點 11)，因為和發

送 KEEP 的 Seed Leader Node(圖 23 節點 10)為鄰居關係，所以按照一般程序加入 Seed Leader Node 10。

Election 步驟執行至此，拓撲的雛形已然浮現，如圖 24，Seed Leader Node 的鄰居節點紛紛加入最接近的 Seed Leader Node。原先的 Seed Leader Node 節點 10 因 CONFLICT 程序中落敗而轉為一般節點，之後加入 Seed Leader Node 11；節點 15 加入最接近的 Seed Leader Node 節點 14，但其同時也是 Seed Leader Node 節點 13 的鄰居(節點 13 可以接收到節點 15 的 beacon 封包)，因此名義上節點 15 將 beacon 封包上的 LNID 改為 14，但 Seed Leader Node 節點 13 與 Seed Leader Node 節點 14 將同時記錄節點 15 的存在，此種特殊的節點稱為 Gateway Node，其功用為達成跨群集的通訊。一但節點成為 Gateway Node，將發送 GW 封包給鄰近 Seed 的 Leader Node(如圖 22)，GW 封包格式如下：

圖 22　GW 封包發送

$$GW(NID||LNID||\ HMAC_{session\_key}(NID||LNID||Timestamp)||Timestamp) \quad (7)$$

GW 封包裡的欄位屬性 NID、Timestamp 和之前的封包作用相同，LNID 代表該節點所能感應到的 Leader Node 集合，用來保證完整性與合法性的 HMAC 用的金鑰則是採用 GateWay Node 和 Seed Leader Node 共有的 session key。

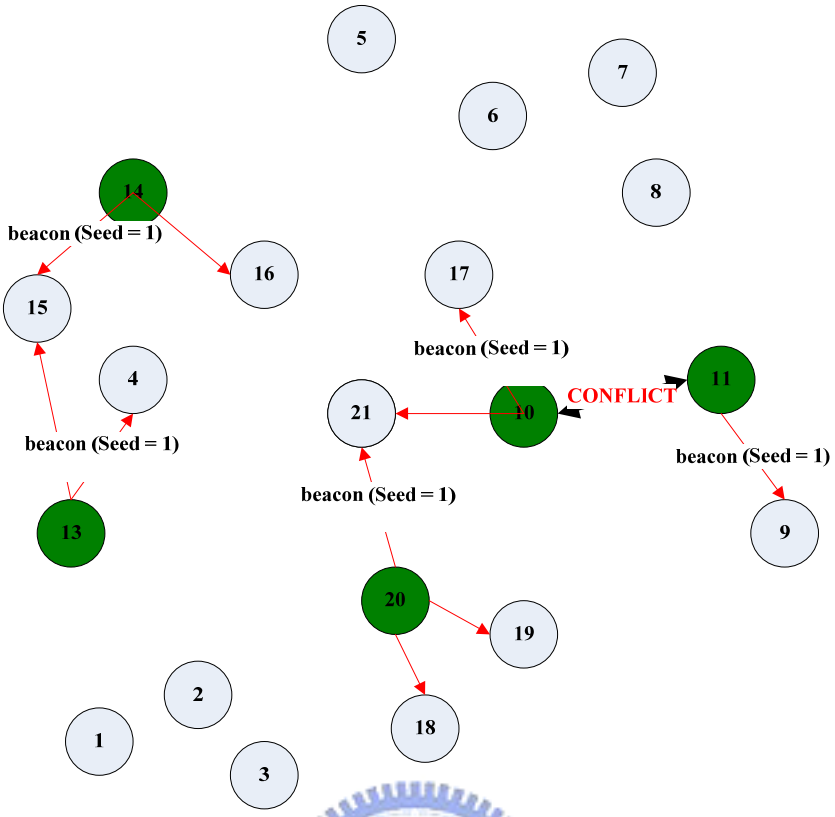


圖 23　Seed Leader Node Conflict

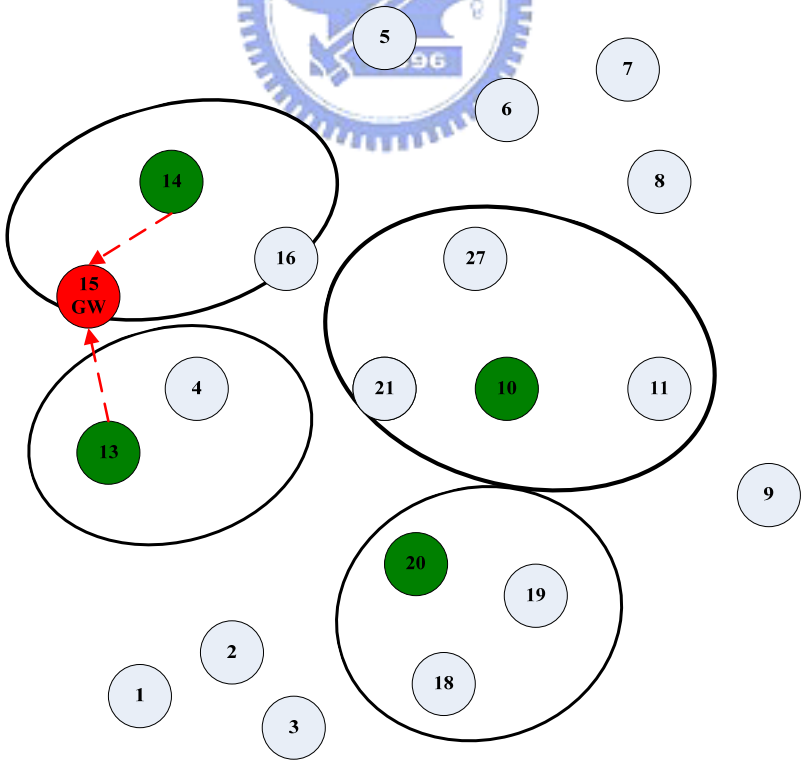


圖 24　Gateway Node

3.4.2.2.2 Normal Leader Node Election

經過 Seed Leader Node Election，網路中已有群集的產生，但未座落在 Leader Node 附近的節點仍然處於 unclustered 狀態，故須進行 Normal Leader Node Election 程序來替所有 unclusterd 節點找出適合的 Leader Node，並組成 cluster。Normal Leader Node Election 程序的核心即為本論文所提出的 IA based Ad hoc Network Clustering Algorithm (IAdNCA)分群演算法，IAdNCA 在分群時主要考量四個因素，分別為節點 Degree 數、節點間距離、節點剩餘電能、以及 Potential Strong Gateway Node 數量。

1. 節點 Degree 數

Degree 數代表節點的鄰居數量，此數值直接反映某節點成為 Leader Node 後是否能服務夠多的鄰居節點，若讓 Degree 數低的節點成為 Leader Node，其他非鄰居節點無法加入該群集，只好繼續執行分群流程以找出更多 Leader Node。假使拓樸中的 Leader Node 太多，顯示網路中出現許多過小的群集，由於跨群集通訊需要複數個 Leader Node 的溝通，所耗成本比群集內通訊的成本要大，大量的小群集將阻礙網路拓墣的效率和運作。

2. 節點間距離

本論文假定所有 IA 裝置進行資料傳輸時，訊號發射功率為可調整的。若節點之間的距離較近，可調降其發射功率，使節點間的傳輸耗電量減小，撇除擁有持續電源供應的 IA 節點，此點對於受限於電能條件的小型 IA 節點而言，可提升在網路中的穩定存活時間。今假使某節點處於相對中心位置，且和其他鄰居節點的距離都較短，由其來擔任 Leader Node，和群集內節點的溝通所耗電能將可降低。

3. 節點剩餘電能

此因素的考量點與距離相同，若節點擁有較大的剩餘電能，作為 Leader Node，能維持群集的時間就越長；若群集成立後短時間內就崩解(Leader Node

電能耗盡)，重新執行分群流程將耗用更多的成本，且為整體拓墣提高不穩定因素。

4. Potential Strong Gateway Node

Normal Leader Node Election 程序中，每一個 IA 節點持續接收鄰居節點的 beacon 封包，如果接收到 LNID 不為 0 之 beacon 封包，可得知該鄰居節點已經加入別的群集，如圖 25 所示。

在圖 25 中，假設節點 1 為當作 Leader Node 的適合人選，一旦節點 1 當選 Leader Node，節點 2 勢必成為節點 1 和節點 3 的 Gateway Node，從網路拓樸的角度來看，Gateway Node 擔任跨群集溝通的橋樑，倘若一個群集內的 Gateway Node 全部失去作用而無法進行傳輸功能，則此群集將被「孤立」(isolated)而無法對外聯絡，由此可看出 Gateway Node 角色的重要性。從這個觀念我們不難發現，某個節點擔任 leader Node 的拓樸中，若旗下有越多 Gateway Node，則群集越不容易發生孤立的現象，另外，若群集內的 Gateway Node 越「健壯」，也就是生存壽命越長(剩餘電能越大)，則代表群集通往外部的橋樑是相對穩固的；綜合以上敘述，如果一個節點成為 Leader Node 時擁有多於其他節點的長壽命 Gateway Node，其將比別人更適合擔任 Leader Node。

本論文定義，在 Election 步驟執行期間，若一處於 unclustered 狀態的節點 n_i 接收到鄰居節點 n_j 所發 beacon，而 beacon 中的 LNID 欄位不為 0(即該鄰居節點已加入別的群集)，則 n_j 為 n_i 的 Potential Gateway Node，如圖 24 中節點 2 為節點 1 的 Potential Gateway Node；另外，假設節點 n_j 為 n_i 的 Potential Gateway Node，而 n_j 擁有持續電能供應或是 n_j 的剩餘電能超過預先定義的門檻值，則 n_j 為 n_i 的 Strong Potential Gateway Node。

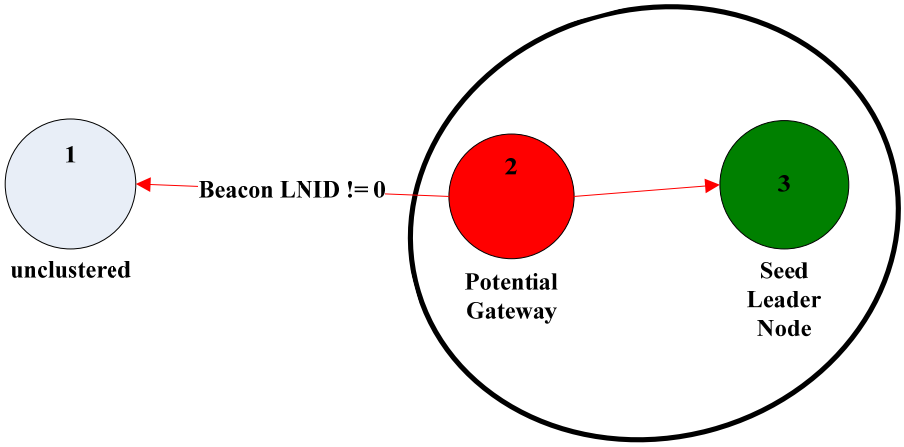


圖 25　Potential Gateway Node

IAdNCA 執行期間，網路上任一節點可藉鄰居 beacon 封包的收集得知自己的 Degree 值；從 beacon 的訊號強度進行鄰居節點距離的估測；利用 beacon 封包中的 LNID、Energy 及 $Flag_{Energy}$ 欄位可得知周遭存在多少 Strong Potential Gateway Node，取得這四樣要素後，採用以下的公式計算 Election Score，然後進行比較

$$ElectionScore = w_1 \times Degree + w_2 \times Degree_e + w_3 \times GWDegree_{strong}$$

$$\sum_{i=1}^{3} w_i = 1$$

w1、w2 和 w3 為 0 到 1 之間的權重參數，$Degree_e$的計算方式如公式(1)：

$$Degree_e = Degree - Degree_{far} \quad (1)$$

Degree 代表節點所偵測到的連結支度，而$Degree_{far}$則代表「考量到節點目前剩餘電量狀態」距離過於遙遠的鄰居節點。$Degree_{far}$的判斷由公式(2)進行判斷：

$$\text{Distance}_{\text{far}} = \text{EnergyRatio}^{\text{L}} \times \text{MAX\_Range} \quad (2)$$

EnergyRatio為節點的剩餘電量，簡化成以百分比表示，百分比越高，代表節點的剩餘電量越充足；L 為懲罰係數(Penalty Coefficient)，L 與EnergyRatio之關聯性如公式(3)：

$$\text{L} = \begin{cases} 1, \text{EnergyRatio} \geq 0.5 \\ 2, \text{EnergyRatio} < 0.5 \end{cases} \quad (3)$$

計算出$\text{Distance}_{\text{far}}$的數值後，任何鄰居節點的距離大於$\text{Distance}_{\text{far}}$，將被算入$\text{Degree}_{\text{far}}$，如圖 26 所示，節點 7 原本的 Degree=3，但經過計算，節點 7 與節點 8 之間的距離超過$\text{Distance}_{\text{far}}$，故節點 8 將被歸為$\text{Degree}_{\text{far}}$中的一員，節點 7 的$\text{Degree}_{\text{e}} = (3-1) = 2$。此種算法的好處在於：假設某節點 n_i 身處的位置可偵測到多數的鄰居節點，但 n_i 本身的剩餘電能所剩無幾，藉計算Degree_{e}的機制，即使 n_i 在 Degree 項目取得很好的加權分數，然低剩餘電能百分比造成$\text{Distance}_{\text{far}}$的限制距離極短，多數鄰居節點都將被歸類在$\text{Degree}_{\text{far}}$，最終 n_i 於Degree_{e}取得的加權分數會遜於剩餘電力充足的節點，降低成為 Leader Node 的機率；另外一個極端的案例為 n_i 本身的剩餘電力百分比很低，但是多數鄰居節點的位置均離其非常近，如圖 27 所示，就算節點 5 的剩餘能量不多，造成$\text{Distance}_{\text{far}}$的限制，但仍然無法將任何一個鄰居歸類在$\text{Degree}_{\text{far}}$。為防止此類案例，懲罰係數 L 的設計可以讓剩餘電量百分比低於門檻值(0.5)的節點，其$\text{Distance}_{\text{far}}$大幅增加，該節點終無法在$\text{Degree}_{\text{e}}$項目中拿到好的加權分數。

最後，$\text{GWDegree}_{\text{strong}}$代表 Strong Potential Gateway Node 的數量，但計算方式如同Degree_{e}，若節點 n_i 和鄰居相距超過$\text{Distance}_{\text{far}}$，該鄰居並不能算做是 n_i 的 Strong Potential Gateway Node，理由同上。

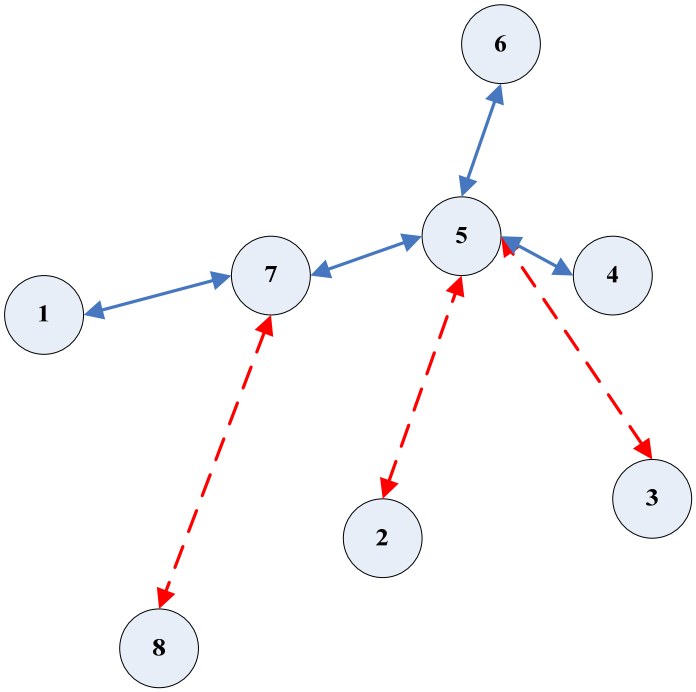


圖 26　$Degree_{far}$ 判定

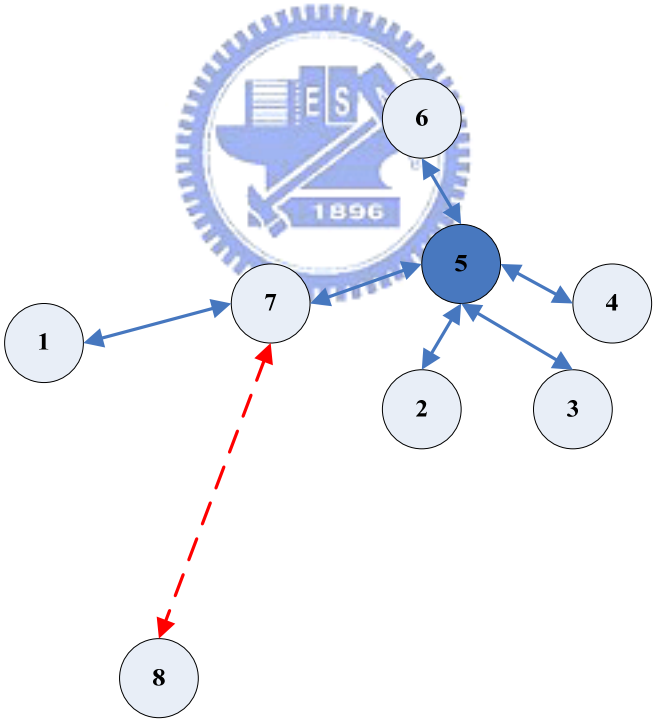


圖 27　懲罰係數 L 使用情境

計算完三個要素的加權分數後，分數總和即為ElectionScore，各節點將計算完畢的Election Score記錄在 beacon 封包的ElectionScore欄位中，而後傳輸給所有鄰居。若某一節點發現自己的Election Score高於所有身旁的鄰居，且自己有能

力擔任 Leader Node，則發送 ANNOUNCE 封包，宣告自己成為 Leader Node(如圖 28)，ANNOUNCE 封包格式如(1)：

圖 28　ANNOUNCE 封包發送

ANNOUNCE{NID|| $HMAC_{g\_session\_key}$(NID||Secret||Timestamp)|| Timestamp }	(1)

NID 為發送 ANNOUNCE 封包之節點 ID、Secret 為 Initial Phase 中向 WPS Root Server 取得的秘值、Timestamp 為 ANNOUNCE 封包發送時的時間戳記。接收到 ANNOUNCE 的鄰居節點利用訊號強度判斷句離，並發送 JOIN 封包給距離自己最接近的 Leader Node。

IAdNCA 反覆執行，陸續有節點成為 Leader Node 或是加入群集，每當有節點加入群集，意味著剩餘的鄰近 unclustered 節點可能得到新的 Strong Potential Gateway Node，這可從持續接收的 beacon 封包察覺；一但偵測到新的 Strong Potential Gateway Node，unclustered 必須重新計算ElectionScore，再以新的 Election Score 去和其他 unclustered 鄰居節點做比對，直到所有節點不是成為 Leader Node 去開創一個群集，就是加入某個 Cluster。等到所有人的 beacon 封包中 LNID 均不為 0 時，整體 IA 隨意網路分群程序結束，一個自適性的 one-hop with-cluster 分群拓樸便產生。如圖 29，深藍同心圓代表 Leader Node、淺藍圓圈代表一般成員節點，若成員節點在一個以上 Leader Node 的訊號範圍內，即為 Gateway Node。

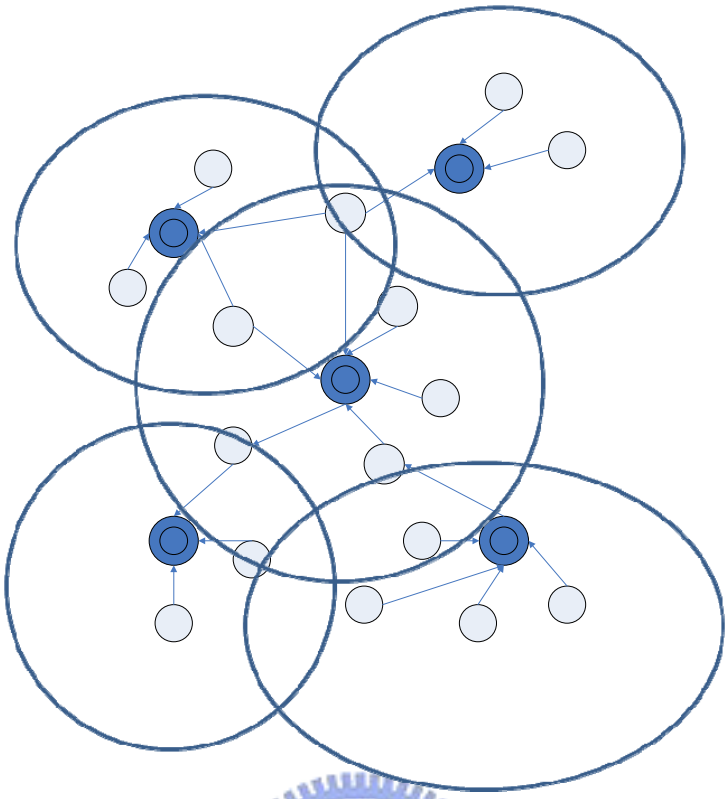


圖 29　隨意網路拓撲完成圖

整個 Election 的執行流程如圖 30 所示：

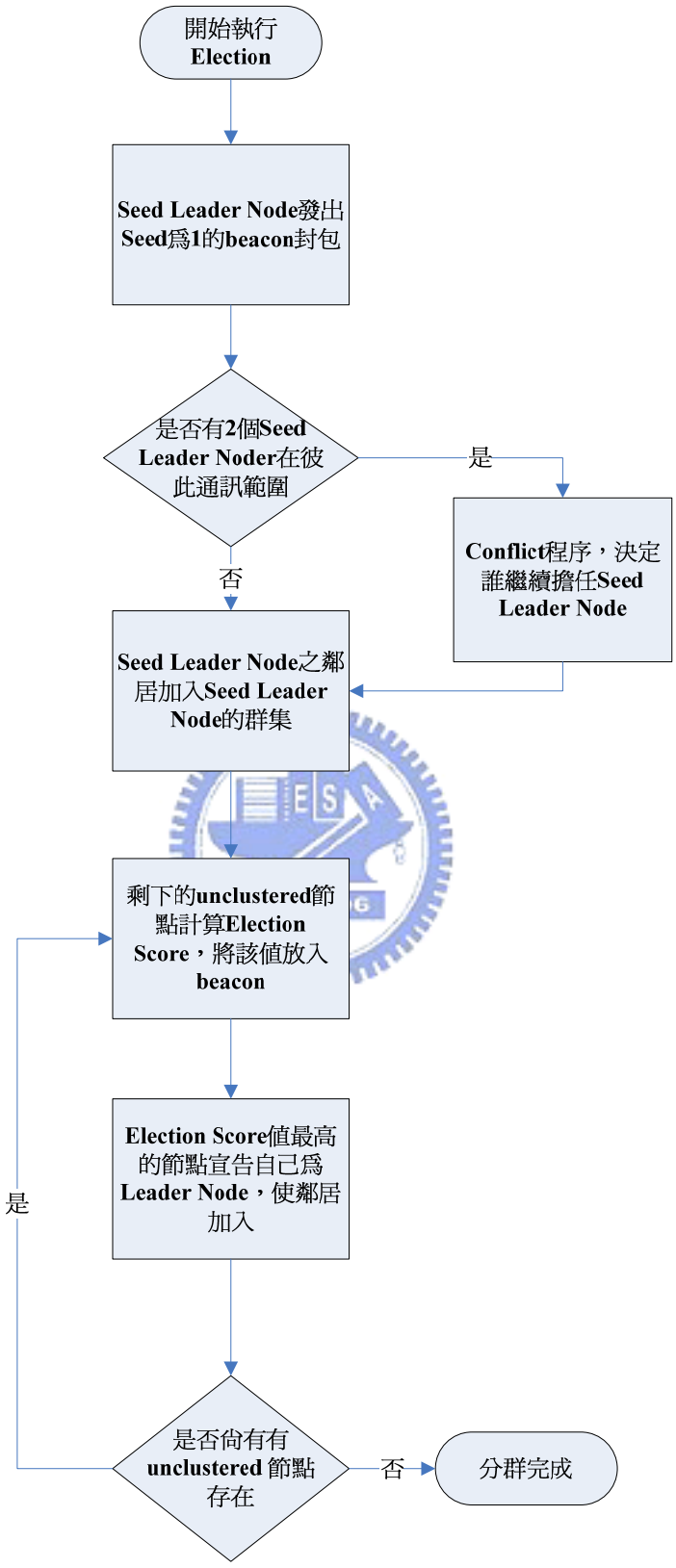


圖 30　Election 流程圖

3.4.3 Maintenance Phase

本章將敘述經 IAdNCA 演算法產生的網路拓撲如何維持各群集運作，而當新節點加入或是 Leader Node 退出網路時又將如何做出微調。

3.4.3.1 New Node Joining Event

當 Clustering Phase 執行完畢，IA 隨意網路拓撲理應構築完成，如同之前圖 29 中的拓樸形態。依照前文所描述 IA 隨意網路的特性，由於 IA 隨意網路中節點的移動性小，除非意外發生、或是節點電能耗盡而無法正常行使通訊功能，否則拓撲中的 Leader Node 將持續擔任此一職務。

至於若有新的 IA 節點 n 在 IA 隨意網路拓撲完成建構後才意圖加入，首先使用者須向最靠近 n 的 Leader Node 輸入 n 的 PIN Code(無論使用 PIN Method 或是 PBC Method)，然後讓 Leader Node 和 n 進行 WPS 註冊程序，程序完成後，Leader Node 將 certificate (Key_{WPA})、Secret 和 g_session_key 附在 M8 訊息中傳給 n，n 收到這些密值後先和各個鄰居進行 Authentication 步驟中 IAM1 至 IAM3 的訊息交流，以建立和各個鄰居之間的 session key(如圖 32)，再把這些 key 保存在 key table 內；然後取隨機值開始倒數，倒數完成後發送 KEY 訊息給 Leader Node，讓 Leader Node(如圖 31)，取得加密後 n 的 certificate (Key_{WPA})。

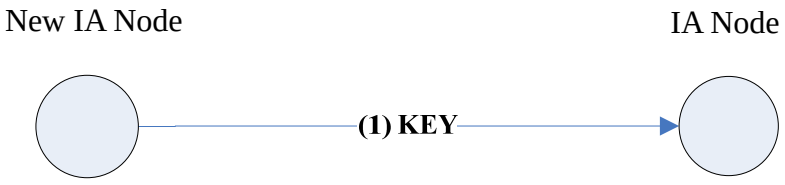


圖 31　KEY 封包發送

KEY{ NID||LNID||session_key(NONCE||Timestamp||Key_{WPA})|| $HMAC_{session\_key}$(NID|| Key_{WPA} ||Secret||Timestamp)||Timestamp }	(1)

KEY 封包結構中，NID 為發送者 ID、Key_{WPA} 為發送者在 WPS 註冊程序中獲得的 pre-shared key、NONCE 為隨機亂數、Timestamp 為時間戳記，Key_{WPA}、NONCE、NONCE 將被發送者與 Leader Node 的 session key 所加密，而 HMAC 同樣利用 session key 產生以保持完整性。

經過以上程序，各個群集的 Leader Node 都握有成員節點的 certificate(WPA pre-shared key)，以後若有 IA 節點離開網路後又要重新加入，只需用 certificate(WPA pre-shared key)和 Leader Node 進行認證即可重回網路。

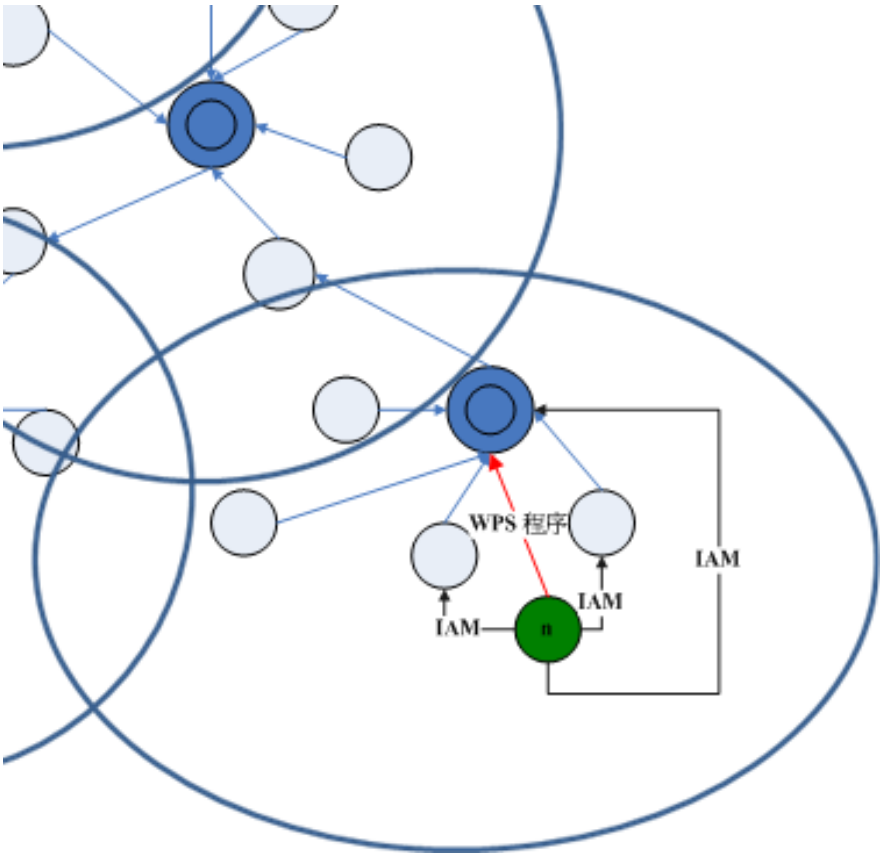


圖 32　新節點加入群集

3.4.3.2 Cluster Reconstruction

當拓撲中某個群其中的成員節點 n 因故退出網路時，將會停止 beacon 封包的發送，負責該成員的 Leader Node 過了一陣子都無法感應到 n 的 beacon 封包，

因而確定 n 失效，把 n 從成員的集合中移除。

當某個群集的 Leader Node 因電源耗盡或故障而退出網路時，將會停止 beacon 封包的發送，原群集中的成員節點過一陣子都沒有感應到 Leader Node 的 beacon 封包，因而確定 Leader Node 失效，全部轉換成 unclustered 狀態，如圖 33;此時,這些 unclustered 節點將重新進行 Election,計算 Election Score，選出新的 Leader Node，並且加入其中，最後，取隨機值開始倒數，倒數完成後發送 KEY 訊息給新 Leader Node，讓 Leader Node 取得的 certificate，重新分群的拓撲如圖 34 所示。

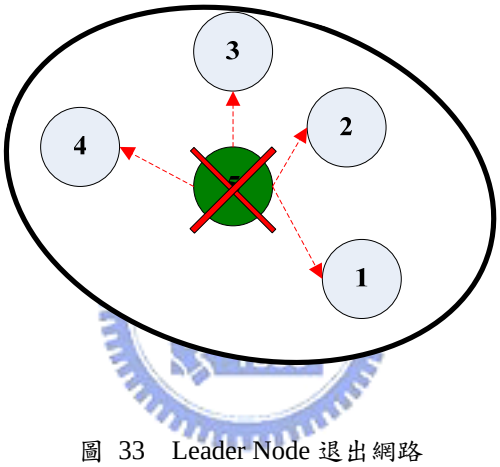


圖 33　Leader Node 退出網路

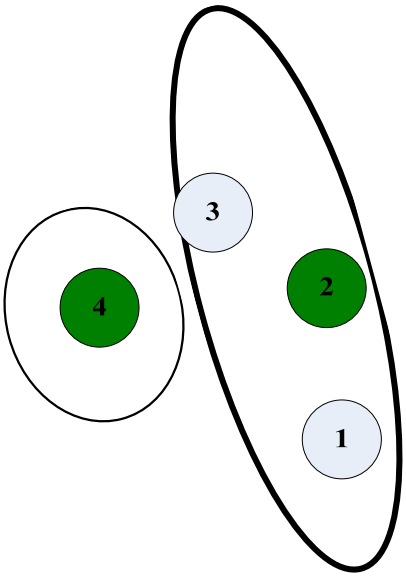


圖 34　重新進行分群程序

第四章 安全性與穩定性分析

在此章中，本論文將利用安全性分析來探討 IA 隨意網路架構的安全防護；另外，亦撰寫程式來模擬 IAdNCA 與 LCA、HCA、BEBCA 的分群流程，以比較產出之分群拓墣的穩定性。

4.1. 安全性分析

本節中，將針對 IA 無線隨意網路分群流程的各部份進行分析，以證明此架構在安全性上的可靠度。

4.1.1. 重送攻擊

所謂重送攻擊(Replay Attack)，意指攻擊者將網路上合法的封包側錄下來，而後將側錄的封包傳輸至系統，進行欺騙與擾亂行為。本論文將針對分群流程中的各部份，針對重送攻擊進行分析

然而本論文所設計的封包格式,都會在加密內容或 HMAC 內容中加上 Timestamp，即使亂封包被加以側錄，惡意者進行重送攻擊時往往已過了封包時效期限，收到重送封包之節點自然會將該封包丟棄，而不予其發生作用。

4.1.2. 身分偽造

在 Initial Phase 時，唯有經過 WPS 註冊程序的 IA 設備才能拿到分群所需的秘值 Secret 和 g_sesion_key。而 g_sesion_key 又是合法 IA 節點與周遭合法鄰居產生 session key 時的負責進行 HMAC 雜湊的必須金鑰；缺少 g_sesion_key 的身分偽造者無法和別的節點順利產生 session key，因此周遭節點將不會把身分偽造者視為合法節點，而將其從 trust table 中刪除。

在 Election 步驟執行時，即使身分偽造者打算冒用合法節點的名義發送

JOIN、CANCEL 或是 ANNOUNCE 封包，以期誤導周遭節點加入不適當的節點，然身分偽造者卻無法得知所冒用之節點與其鄰居的 session key，因此無法產生合法的 HMAC，周遭節點取得偽造的 JOIN、CANCEL 或是 ANNOUNCE 封包後，即得知該封包是有問題的。

假使身分偽造者扮演架空的節點身份並持續發送 beacon 封包，打算成為非法的 Leader Node，但周遭的節點可利用 trust table 檢查鄰居身份的合法性，當他們發現架空節點的 ID 不在 trust table 內，將不予理會身分偽造者所發出的 beacon 封包，如圖 35。

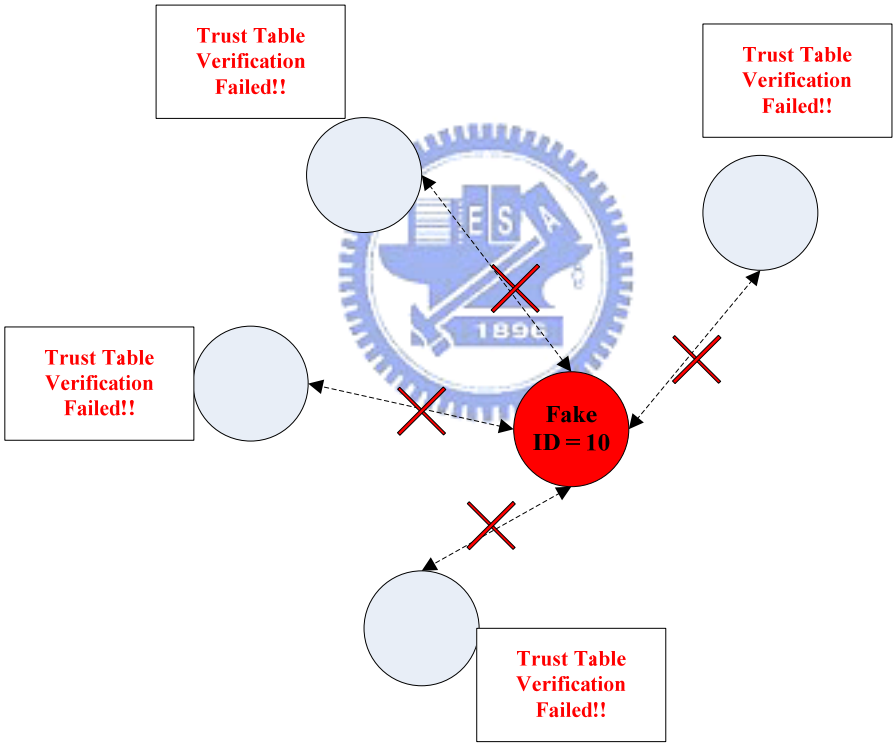


圖 35　Trust Table 防範身分偽造

4.1.3. 竊聽

在 Initial Phase 階段，WPS Root Server 於 M8 訊息中交付給 IA 節點的 credential 會利用前面步驟所生成的 AuthKey 進行加密，以保證其隱密性；而合

法IA節點與其鄰居之間可用彼此共同產生的Diffie-Hellman session key對資料進行加密，以保證隱密性。

4.2 穩定性分析

為了評估分群流程的核心演算法 IAdNCA 所產生的網路拓樸是否真的能達到高度穩定性，本論文採用 c++實作分群模擬程式，讓 IAdNCA 分群演算法和LCA、HCA、BEBCA 產生出來的網路拓撲進行比較，以證明 IAdNCA 的效用。

4.2.1 實驗參數設定

本論文將以多組實驗參數來測試 IAdNCA 產生的網路拓撲是否具有較高穩定性，實驗平台如表所示：

表 7 實驗平台

硬體環境	Intel(R) Core(TM)2 Duo CPU T8100 2.10 GHz 3.0 GB RAM
軟體環境	Microsoft Windows XP sp3 Eclipse CDT, MinGW

實驗相關參數如下表所示：

表 8 實驗參數表

參數	**數值**
場地尺寸大小	200*200, 250*250, 300*300, 350*350,400*400 (公尺)
節點個數	400
節點訊號傳輸最大距離	70
持續電源節點供應比例	5%
w_1, w_2, w_3	(0.3333, 0.333, 0.333)

個別實驗重複次數	500 次

實驗的場地大小分為5種，任一節點的初始電能百分比採均勻分配，隨機賦予1%至100%的初始剩餘電能，分群後所有IA節點呈非移動狀態。5種矩形場地的實驗將各作500次取平均值，實驗結果以三種穩定性指標來進行分析。分群實驗過程將忽略關於無線傳輸所受到的雜訊及干擾、多重路徑衰落、接收資料錯誤、隱藏節點問題。

4.2.2 穩定性指標

為了評估演算法的功能，本論文提出 3 項穩定性指標，利用這 3 種指標對 HCA, LCA, BEBCA 以及 IAdNCA 進行比較。

1. 選出的 Leader Node 平均剩餘電量

分群拓撲產生之後，計算各群集 Leader Node 的平均剩餘電量，如果選出來的 Leader Node 平均剩餘電量較高，群集長時間維持的可能性將大大敵高，對於隨意拓撲的穩定性有明顯的正面影響。

2. 平均群集內 Strong Gateway Node 比例

群集和群集間溝通的橋樑即為 Gateway Node，雖然 Gateway Node 所負擔的責任並不會比 Leader Node 來的大，但若 Gateway Node 因電能耗盡而無法正常執行，Gateway Node 所屬的群集和外界(其他群集)的聯繫就會減弱。當群集監所有的 Gateway Node 均死亡，該群集將被孤立，群集內即使 Leader Node 正常運行， 群集內的成員依舊無法正常發揮通訊功能；因此，在群集中若有較高比例的 Strong Gateway Node (Strong Gateway 個數 / 總 Gateway 個數)，將可對網路拓撲有運作有正面的影響。

3. 平均有效拓樸維持時間

一般的研究裡，典型的分群式架構通訊機制大多採用群集式繞路法(Cluster-Based Routing Protocol, CBRP)，CBRP 的繞路型態分成兩種：

(1) 群集內繞路：

節點 i 欲與節點 j 溝通，若兩個節點身處相同的群集之中，則節點 i 將先與 Leader Node 通訊，經過 Leader Node 在轉至目標節點 j。

(2) 跨群集繞路：

節點 i 欲與節點 j 溝通，若兩個節點身處不同的群集之中，則節點 i 將先與 Leader Node 通訊，Leader Node 找到通往目標節點的路由資訊與對應的 Gateway Node，將封包傳送至 Gateway Node，由 Gateway Node，繼續轉發至目標節點所在群集；最後，目標節點所在群集的 Leader Node 接收封包後，再轉傳給目標節點。

以上兩種路由模式，將消耗 Leader Node 與 Gateway Node 的剩餘電能，當群集的 Leader Node 電能耗盡而死亡，群集中所有成員節點將被迫回復成 unclustered 狀態，重新進行分群流程來尋找新 Leader Node；另外，如果群集內所有通往其他群集的 Gateway Node 電能耗盡死亡時，即使 Leader Node 依然在運作，該群集仍會變為孤立狀態而與外界斷訊。

由於 Leader Node 死亡後將會重啟分群流程，以選出新世代的 Leader Node，所以隨意網路在一定時間內得以繼續運作而不會降低連通性，但隨著各節點的電能漸漸耗盡，網路拓墣的機能將開始降低。

本論文定義平均有效拓墣維持時間的定義為計算拓樸運作時間持續到存活節點數目低於 30%為止(擁有持續電能供應的 Seed Leader Node 將不在考量範圍之內)。若某群集發生孤立的情況，視作與外界通訊不能，故判定該群集所有節點死亡。實驗後，平均拓墣維持時間越長者，代表穩定性越高。

4.2.3 實驗結果

實驗1. Leader Node 平均剩餘電量模擬實驗

對於 5 種不同大小的場地進行 4 種演算法分群模擬，分群完畢後網絡拓樸中所有 Leader Node 剩餘電能百分比的實驗統計結果圖 36：

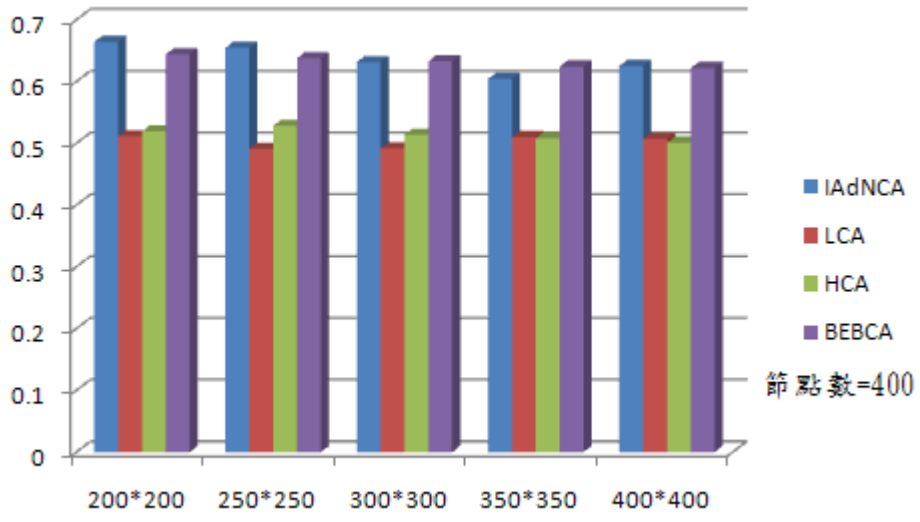


圖 36 Leader Node 平均剩餘電能比較圖

由比較圖表可看出，LCA 與 HCA 分群並未針對節點剩餘電能進行考量，而 BEBCA 和 IAdNCA 不論在哪一種場地尺寸中，Leader Node 剩餘電量幾乎都可維持在 60%以上，高出其他兩者 10%左右，這代表 IAdNCA 和 BEBCA 遴選出來的 Leader Node 在自身能力的條件上更為適任。

實驗2. 平均群集內 Strong Gateway Node 比例模擬實驗

同樣的，對於 5 種場地各作 500 次實驗，每種演算法所產生的網路拓撲中平均 Strong Gateway 總數之實驗統計如圖 37：

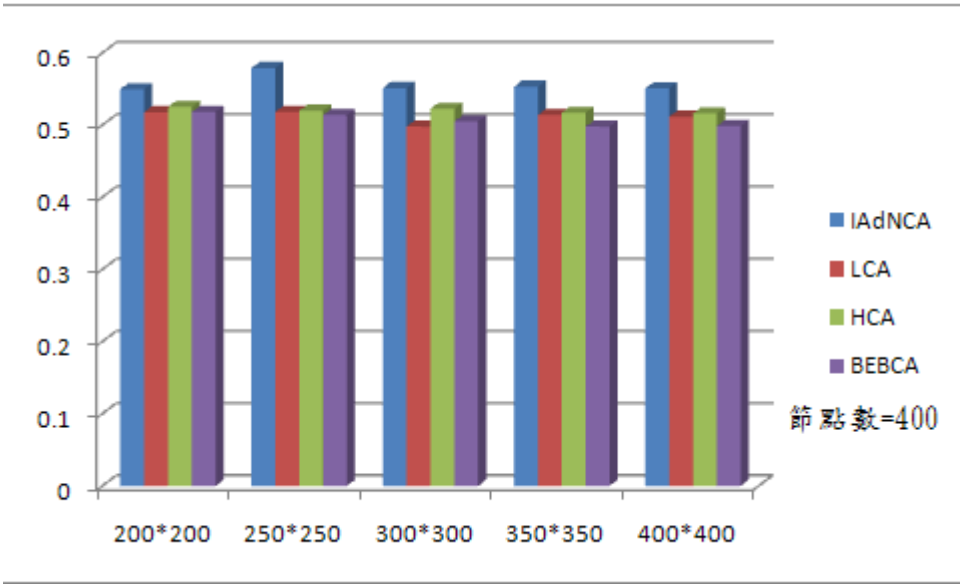


圖 37　Strong Gateway 平均個數比較圖

由比較表中可看出 IAdNCA 執行後的分群拓撲，在 5 種場地中 Strong Gateway Node 總數都比其他 3 種演算法高出 4 個百分點以上，和前一個指標的實驗合起來審視，可判定經由 IAdNCA 所執行產出的群集在拓樸開始運作的一段時間內，相較於其他演算法，能夠保持一定程度的通訊穩定性。

實驗3. 平均有效拓樸維持時間模擬實驗

進行平均拓撲維持時間模擬實驗時，模擬程式所使用的能耗模型為First Order Radio Model[24][28]，此模型可用來計算muilti-hop環境中節點經過k-hop到達距離為d之目的地接收端之能耗。在First Order Radio Model當中，傳送資料所需要消耗的總能量設為E_{TX}，接收資料所需要消耗的總能量設為E_{RX}。在傳送資料時，除了需要考慮傳送資料所需要消耗的能量（ETX_elec），還要考慮放大機(Amplifier）擴大訊號所消耗的能量（ETX_amp），傳送每單位的資料所需要消耗的能量設為Eelec，每單位的資料擴大訊號所需要消耗的能量設為E_{amp}，而E_{amp} 和傳送距離 d 的n 次方（d^n）成正比，一般環境中，n<4。綜合以上變數，資料發送節點所要消耗的總能量如(1)：

$$E_{TX}(k,d) = E_{elec} * k + E_{amp} * k * d^{n} \quad (1)$$

資料接收節點所要消耗的總能量如(2)：

$$E_{RX}(k) = E_{elec} * k \quad (2)$$

因此資料發送端和資料接收端的能耗總和如(3)：

$$C_{ni,\,nj}(k) = 2\,E_{elec}*k + 2\,E_{amp} * k * d^{n} \quad (3)$$

實驗進行的方式為讓拓樸中所有的節點隨機尋找destination並進行封包傳輸，傳輸持續時間為1至10秒之間，通訊過程採用CBRP進行繞路；為求縮短實驗時間，給與每個節點微小的初始電能。

表9為進行平均拓撲維持時間模擬實驗關於能耗模型及網路傳輸設定的各項參數：

表 9 First Order Radio Model 參數設定表

參數	**數值**
E_{elec}	50nJ
E_{amp}	100pJ
n	3
節點剩餘能量最大值(100%)	50J (扣除持續電源供應節點)
單一封包大小	1024 (byte)
封包傳送速率	10 packet (per second)

圖 38 為各個尺寸的場地進行 500 次反覆實驗的平均結果：

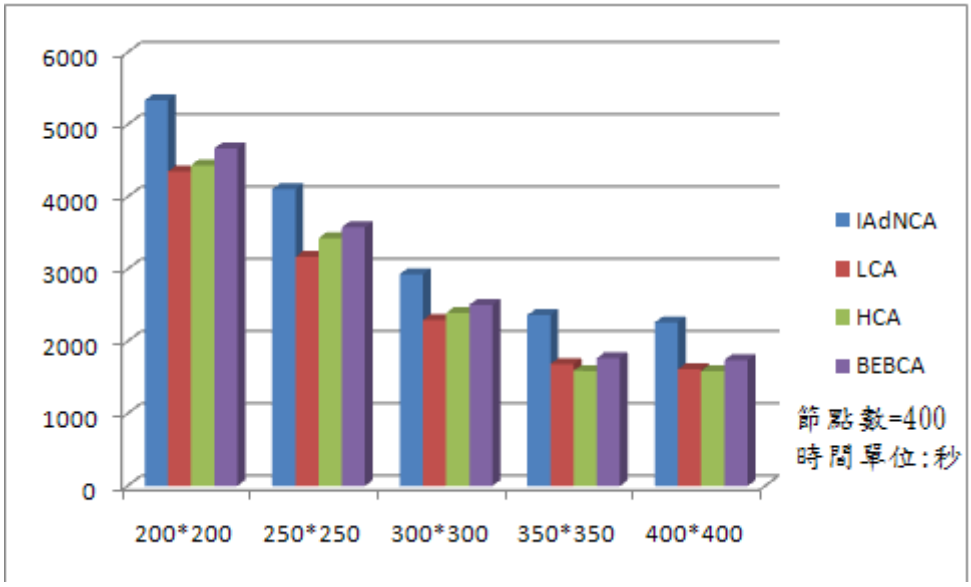


圖 38 有效拓撲維持時間比較圖

由圖 38 可看出，IAdNCA 分群產生的拓樸，其平均有效拓撲維持時間在各個尺寸場地的表現均優於其他三種演算法超過 600 秒，BEBCA 所產生的拓樸雖然在 Leader Node 平均剩餘電能的數據項目表現良好，但未考量其他因素，IAdNCA 在同時考量電能、距離、Degree 數、Strong Gateway Node 的情形下，建構出來的網路拓撲具有更久的有效拓撲維持時間。至於場地大小對於拓撲節點的影響，從圖 38 中可看出各演算法在大型場地的表現皆不如小尺寸場地理想。由於在本實驗中節點個數於各個尺寸場地都維持 400 個，相較於小尺寸場地，大尺寸場地中節點散布於各處，節點密度小同時意味著節點間彼此的平均距離拉大，根據 First Order Radio Model，距離加大發送封包所耗的電能也會跟著提高，最終導致有效拓撲維持時間明顯縮減。

4.2.4 小結

從模擬實驗 1 可得知，在相同的條件之下，IAdNCA 所選出的 Leader Node 之平均剩餘電能和 BEBCA 所選出之 Leader Node 相當，從電能維持的角度來看，IAdNCA 和 BEBCA 所選出的 Leader Node 其壽命明顯高於 HCA 和 LCA。又，網路拓撲運作的穩定性不只依賴 Leader Node 的電能壽命，也須考量跨群集溝通的協調者 Gateway Node 的電能壽命；我們從模擬實驗 2 得知 IAdNCA 演算法成功地讓選出之 Leader Node，其身旁的 Strong Gateway Node(剩餘電量百分比超過 50%)比例超過 HCA、LCA 與 BEBCA，以維持各群集間溝通的通道。最後，模擬實驗 3 的結果顯示 IAdNCA 所產生的拓撲，其平均有效拓樸維持時間明顯高於其他三種既有演算法，證明了較高 Leader Node 剩餘電能百分比、較高 Strong Gateway Node 比例、以及 IAdNCA 所考量的距離、Degree 等因素總體發揮正面的影響，延長了拓撲整體運作的壽命。

第五章 總結與未來展望

5.1 總結

本論文提出一個兼具穩定性與安全性的 IA 隨意網路分群架構，其貢獻主要包含了安全性、穩定性與方便性三方面。其安全性貢獻如下：

(1) 利用 WPS Root Server 的設計和節點初始註冊動作，確保惡意/非法節點無法輕易偽裝成拓撲建立前的初始合法節點，以在分群隨意拓撲成型前混入網路中。

(2) 分群時節點間相互產生的 Diffie-Hellman session key 使惡意/非法節點無法偽造身分加入群集、或非法創立群集。

(3) HMAC 和 Timestamp 的設計確保各類封包的認證、時效性與完整性

(4) 導入 WPS 註冊程序確保 IA 節點註冊時 Credential 發放的隱密性

穩定性的貢獻則下所述：

(1) 分群式隨意拓撲架構符合 IA 設備能力不對等的情境，讓能力較強的適當節點擔任群集領導者(Leader Node)，並管理附近的節點，此乃較有效率的作法。

(2) Seed Leader Node 機制設計既符合 IA 的特性，又能保證 Leader Node 電供給，保證隨意拓撲部分群集的永續運作。

(3) IAdNCA 演算法考量到距離、節點剩餘電能、Degree 等因素，選擇出來的 Leader Node 所組成的拓撲可延長運作時間，增加拓撲壽命。

(4) Strong Gateway Node 評斷機制強化跨群集通訊的穩定性，降低群集被孤立的可能性。

最後，方便性的貢獻為導入 WPS 註冊流程，令不具備良好 UI 的 IA 設備，能以幾乎自動的方式加入拓撲網路並得到 Credential，讓使用者在 IA 設定上的困難降到最低。

雖然本論文提出之分群架構具有以上貢獻，卻仍有限制及待改進之處，列舉如下：

(1) 礙於 IA 設備的效能考量，Initial Phase 中 WPS 註冊程序所發配給初始合法節點之 Credential 內含相同的對稱式金鑰(g_session_key, secret)，而非安全性更強的非對稱金鑰組合。因此，Authentication 步驟執行上的安全性將會降低。

(2) 分群流程中對 L2 封包進行修改，造成一般具有 Wi-Fi 802.11 連線能力之設備和 IA 間的溝通仍須經過閘道角色轉換；除非另外定義 802.11 based 設備和 IA 之間的溝通流程，否則直接的通訊將發生困難。

(3) IAdNCA 未定義 Leader Node 所管理之成員數的最大值，若在 IA 高密度充斥的環境中，可能發生 Leader Node 旗下的所屬成員過多，反而快速耗費 Leader Node 的電能，降低拓撲壽命與通訊穩定性。

5.2 未來展望

本論文的研究重心主要擺在網路層穩定隨意拓撲的建立，然而對於 IA 無線隨意網路的應用面卻未多加著墨。因此，未來研究的發展方向可朝向上層服務應用與下層分群式網路拓撲的整合界接，以發揮後 PC 時代中 IA 產品的潛在價值。另外，在分群演算法的設計部分，本論文提出的 IAdNCA 並未考量到 IA 節點訊號發送功率與節點密度的因素，若在一個 IA 擺設密度極高的環境中，節點依然維持較高的 beacon 發送功率，則單一群集將會有大量的成員節點加入其下，造成 Leader Node 負擔過重，因此，本論文另外一個未來方向可朝 IA 群集分裂或群集內多管理者的方向努力，以期達到更為優化的拓撲結構。

參考文獻

[1] A.D. Amis, R.Prakash, T.H.P. Vuong, D.T. Huynh, “Max-Min D-Cluster Formation in Wireless Ad Hoc Networks”, in Proceedings of IEEE INFOCOM , pp. 32-41, 2000.

[2] A.Pirhonen , H. Isomäki , C. Roast, P. Saariluoma, “Future Interaction Design” Springer. pp. 129, 2008.

[3] B. David, T. Phil, T. Susan, “Designing Interactive Systems: People, Activities, Contexts, Technologies”, Addison-Wesley, 2005.

[4] C.H. Huang, “Cluster-Based Routing with Backup Route in Wireless Ad Hoc Networks”, Master Thesis, Department of Electrical Engineering, National Sun Yat-Sen University, 2006.

[5] C.H. Leon Lee, W.S. Tseng, C.M. Chen, A. Liu, S.M. Huang, H.B. Huang, “Designing an Intelligent Agent for Information Appliances”, Journal of Computers , Vol.18, pp.61-70, 2007.

[6] C.R. Lin, M. Gerla, “Adaptive clustering for mobile wireless networks,” IEEE J. Select. Areas Commun., pp. 1265-1275, 1997.

[7] C.R. Lin, M.Gerla, “Adaptive clustering for mobile wireless networks,” IEEE J. Select. Areas Commun., pp. 1265-1275, 1997.

[8] C.W. Tseng , “Algorithms for the Tree-based Home Network Design Problem”, Master Thesis, Institute of Information Management, National Chiao Tung University, 2008.

[9] C.W. Wang, “A Study on Stable Clustering for Mobile Ad Hoc Networks”, Master Thesis, Department of Electrical Engineering, National Yunlin University of Science and Technology, 2002.

[10] C.Y. Lee, “A Novel Authentication Protocol for Ad Hoc Networks”, Master

Thesis, Institute of Information Management, National Chiao Tung University, 2005.

[11] D.C. Su, S.F. Hwang, C.R. Dow, Y. W. Wang, “An efficient k-hop clustering routing scheme for ad-hoc wireless networks”, Journal of the Internet Technology, Vol.3, No.2, Apr. 2002, pp.139-146, 2002.

[12] Directorate-General of Budget, Accounting and Statistic, Executive Yuan, ROC (Taiwan), http://www.dgbas.gov.tw

[13] Gainspan, http://www.gainspan.com/

[14] H. Luo, S. Lu, “Ubiquitous and Robust Authentication Services for Ad Hoc Wireless Networks.” Technical Report TR-200030, Dept. of Computer Science, UCLA, 2000

[15] H.C. Ke, “A Double-Manager K-hop Clustering Algorithm in Mobile Ad Hoc Networks”, CIT The Fourth International Conference on Computer and Information Technology, 2004.

[16] Institute for Information Industry Market Intelligence Center, Industry & Technology Intelligence Service Report, 2000

[17] J. Kong, P. Zerfos, H. Luo, S. Lu, L. Zhang, “Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks”, IEEE 9th International Conference on Network Protocols (ICNP'01), 2001

[18] L. Venkatraman, D. Agrawal, “A novel authentication scheme networks” in IEEE Wireless Communications and Networking Conference (WCNC 2000), vol. 3, pp. 1268--1273, 2000.

[19] M. Chatterjee, S.K. Das, D. Turgut, “An On-Demand Weighted Clustering Algorithm (WCA) for Ad hoc Networks”, Global Telecommunications Conference, 2000.

[20] M. Gerla, J. Tsai, “Multicluster, mobile, multimedia radio network”,

ACM/Baltzer Journal of Wireless Networks, vol. 1, pp. 225-238, 1995.

[21] T.J. Ross, J.L. Hill, M.Y. Chen, A.D. Joseph, D.E. Culler, E.A. Brewer, “A Composable Framework for Secure Multi-Modal Access to Internet Services from Post-PC Devices”, Third IEEE Workshop on Mobile Computing Systems and Applications, pp.171-182, 2000.

[22] V. Varadharajan, R. Shankaran, M. Hitchens. “Security for cluster based ad hoc networks.” Computer Communications, vol. 27, pp.488-501, 2004

[23] Wi-Fi Alliance, “Wi-Fi Protected Setup Specification”, version 1.0h, 2006.

[24] X. Chen, H.Y. Tang, W.J. Chen, S.L. Tu, Z.L. Chen, “Energy-based Multi-hop Clustering Algorithm for WSN”, Computer Engineering, Vol. 34, 2008

[25] Y.G. Sun, X.G. Wu, Y. Liu, “Cluster Header Algorithm for Wireless Sensor Network”, Computer Engineering, Vol. 34, 2008

[26] Y.H. Chien , “Locality-Based Trust Group Authentication Services in Mobile Ad Hoc Networks”, Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University, 2007.

[27] Y.Y. Su, S.F. Hwang, C.R. Dow, “A Cluster-Based Routing Algorithm in Ad Hoc Networks with Unidirectional Links”, Master Thesis, Department of Information Engineering and Computer science, Feng Chia University, 2002.

[28] Z. Fan, H. Zhou, “A Distributed Weight-based Clustering Algorithm for WSNs”, WiCOM International Conference on Wireless Communications, Networking and Mobile Computing, 2006.

[29] Z.J. Haas, “A new routing protocol for the reconfigurable wireless networks”, Proceedings of IEEE ICUPC’97, pp.562-566, 1997.