

大規模動態物聯網之路由演算法設計

Routing Protocol of the Dynamic Large-scale Internet of Things

鄭元博

郭慧冰

Yuan-Po Cheng, Hui-Ping Kuo

中文摘要

如何讓大規模動態物聯網中的裝置可以透過網路將資料傳到其它裝置，是物聯網研究上一個很重要的議題。若同時要達到避免廣播風暴、適應動態環境、並具備低耗能等特性，目前還沒有可以實際應用到現實環境中的路由協定。本文介紹的LF-GFG路由協定是第一個可以滿足上述性質的大規模動態物聯網路由協定，可望解決目前困境並加速物聯網實現的腳步。

Abstract

Routing protocol of the dynamic large-scale internet of things is an important issue. The proposed location-free greedy-face-greedy (LF-GFG) routing protocol is the first delivery-guaranteed location-free routing protocol which can be used in the dynamic large-scale internet of things. We introduce LF-GFG in this paper to help people solving the problem of developing the IoT.

關鍵詞 (Key Words)

物聯網 (Internet of Things ; IoT)

路由協定 (Routing Protocol)

面路由協定 (Face Routing)

1 · 前言

物聯網 (Internet of Things ; IoT) 將各種裝置互相連接，形成一個網路，讓不同裝置之間能夠互相溝通、協調、控制、以及蒐集資料，減少中間需要人為判讀操作的部分，可以讓生活增加很多的便利性，並有很廣泛的應用。

如何讓物聯網中的裝置可以透過網路將資料傳到目的地，是物聯網研究上一個很重要的議題。在物聯網中，裝置一般是透過無線網路傳輸，裝置可能會移動，並且可能是使用電池作為電源。因此，物聯網的路由協定必須要具備低耗能的特性，不適合使用 3 G、4 G 等大範圍無線網路，但是又必須能適應動態網路環

境的特性，一般會使用多跳無線隨意網路 (Multi-hop Ad-hoc Networks) 的方式傳輸。

目前最常見的路由協定有 AODV[1] 與 DSR[2] 等，這些路由協定會先透過一個或多個洪泛廣播 (flooding)，得到將封包送達目的地的傳輸路徑，再將資料沿著這條路徑傳到目的地。隨著物聯網的規模逐漸擴大，這些洪泛廣播會造成大量的廣播風暴 (Broadcast Storm) [3]，導致網路效能低落，所以有人提出了位置路由協定 (Geographic Routing Protocol) 的概念，希望能改成透過目的地的位置資訊來取得所需要的傳輸路徑，以避免洪泛廣播所遇到的問題。

2 · 虛擬平面化網路

在最早的位置路由協定研究中，研究者會先將問題簡化，假設(1)每個裝置都有一樣的傳輸半徑，並且假設(2)每個裝置都能透過一些定位的方法取得自己的精確位置。早期代表性的研究有GPSR[4]、GFG[5]、GOAFR+[6]等，這些方法在適當的假設下，可以不需要任何額外的封包，就將網路轉換為一個平面化的圖形(planar graph)，然後透過右手定則(Right-Hand Rule)取得所需的傳輸路徑。這些方法所需的封包非常少，可以大幅度增加網路中可容納的裝置數目，同時大幅降低傳輸的耗能，非常適合大規模動態物聯網使用；然而，假設(1)和假設(2)都不容易在真實世界中達成，所以研究者提出了各種方法來克服這兩個假設。

其中最成功的方法是提出虛擬座標的概念，如VRR[7]、VPC[8]、和ABVCap[9]。這些方法會在網路剛建立的時候進行分配虛擬座標的動作，當每個裝置都得到了虛擬座標之後，這些裝置就能夠透過這些具有特殊涵義的虛擬座標取得所需的傳輸路徑。這些方法完全解決了假設的問題，可以實際應用到真實的環境中，但是為了要保持虛擬座標所具備的特殊性質，反而讓它們失去了適應動態網路的能力。當物聯網內的裝置有移動或是開關，這些方法的傳輸能力就會開始下降並且不容易修復。

這篇文章中要介紹的Location-Free Greedy-Face-Greedy (LF-GFG)[10]，是第一個可以適用於大規模動態物聯網的位置路由協定。在LF-GFG中，網路剛建立的時候依樣會分配虛擬座標給每個裝置，但是這些虛擬座標只代表了一個二維平面上的點，並不具備複雜的特殊意義。LF-GFG中提出了一個方法，可以不需要額外的封包就將整個網路轉換成一個平面化的圖形。因此，透過GFG所使用的右手定則，LF-GFG可以取得所需的傳輸路徑，完美地保留GFG的所有優點並克服了實際環境應用的問題。

接下來我們會在段落2介紹LF-GFG的概念，段落3描述LF-GFG實作的細節，在段落4簡述實驗的結果，並且在段落5提供一個簡單的結論。

2.1 右手定則

GFG路由協定中提出的右手定則是早期所有位置路由協定的基礎。在右手定則中，將圖上的邊想像成牆壁，將右手放在牆上往前走(走在牆的左側)，若這是一個平面化圖形，則右手定則保證可以越走越靠近目的地。換句話說，若能將網路連線轉化成一個平面化圖形，重複地使用右手定則來決定路由路徑，可以在理論上保證將封包送達目的地。

隨著各種的後續研究推出，使用右手定則可以達到不用任何額外的封包，只透過無線網路傳輸時的RTS、CTS封包，就直接將資料傳到傳輸路徑上的下一個點[11]。換句話說，使用右手定則的傳輸率和傳輸消耗，都幾乎達到了理論上的最佳化。因此，若能克服右手定則實作時所遇到的問題，將會成為物聯網路由協定發展的一個重要里程碑。

GFG路由協定的右手定則要能順利運作，有兩個需求：

- (1) 每個裝置都有一個在二維平面上的座標。
- (2) 整個網路可以轉換成一個平面化的圖形。

因此，如何達到這兩個需求就是LF-GFG的目標。

2.2 平面化圖形

在研究多跳無線隨意網路路由協定時，我們最常使用的方法是將每個裝置表示成一個二維平面上的節點，若兩個裝置間可以互相傳輸資料，我們就在這兩個節點間加上一條邊。這樣一來，網路就可以表示成一個圖形(graph)，而路由協定的問題就能轉化成一個圖形學(Graph Theory)的問題了。

在一個平面化圖形中，所有的邊都不會交叉，如何將一個圖形平面化，是一個很複雜的問題。在物聯網中，為了減少網路的負擔，每個節點是不可能擁有完整的網路架構資訊，只會擁有和其鄰近節點資訊而已。因此，如何讓每個節點，利用鄰近節點的資訊，就能分散式地將一個圖形平面化，就是想要使用右手定則所

要面臨的問題。

傳統的做法是透過讓每個節點分散式地滿足 Gabriel Graph (GG) [12] 或 Relative Neighborhood Graph (RNG) [13] 等圖形特性，來取得平面化圖形，但是這個做法很難在實際的環境中實作出來。另一個方法是透過 Voronoi Diagram 來取得，而這也是 LF-GFG 的靈感來源。

2.3 概念簡介

LF-GFG 延續虛擬座標的概念，但是為了能夠適應動態網路環境，我們想要讓 LF-GFG 所使用的虛擬座標，包含越少的特殊資訊越好。也就是說，希望它只是一個滿足右手定則使用條件的虛擬座標：

- (1) 是一個二維座標。
- (2) 透過這個座標可以將網路轉成平面化圖形。

其中較困難的是第二個性質，為了讓我們可以分散式的取得平面化圖形，如何設計這個二維虛擬座標就成了這篇論文的關鍵。

我們的想法來自於 Voronoi Diagram。在 Voronoi Diagram 中，每個節點會被劃分一個區域。若將區域相鄰的節點連起來，就會得到一個平面化圖形。因此我們想到，若是我們所設計的二維虛擬座標，都可以轉化成為一個區域，然後透過判斷這些區域是否相連，可能就能夠順利地平面化圖形了。舉例來說，在圖 1 中，有 A 到 L 共 12 個節點，兩個節點之間若有連線則表示這兩個裝置可以互相通訊。每個節點我們都劃分了一個區域給它之後，只要判斷兩個區域是否相鄰，只留下相鄰區域間的連線，就能夠得到一個平面化的圖形了，如圖中粗線所示。

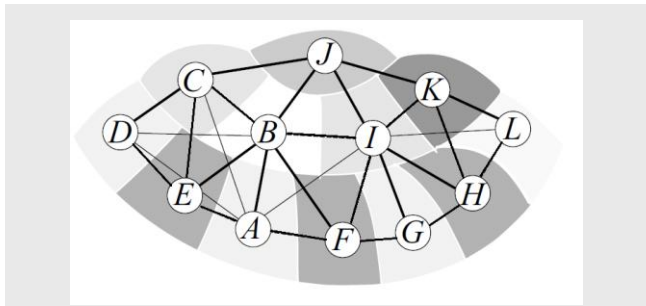


圖 1 LF-GFG 的概念示意圖

3 · 實作細節

3.1 分配虛擬座標

在這節中，我們會用一個範例來說明實作的細節。假設現在有一個物聯網網路如圖 2 所示。在網路中，共有 A 到 K 共 11 個裝置，彼此間可以互相通訊的裝置就會有一條邊將它們連起來。這些裝置並不知道它們自己的位置，只知道它附近有哪些其它節點。

首先，LF-GFG 會從一個預先設定好的起始點開始分配虛擬座標的工作。在我們的範例

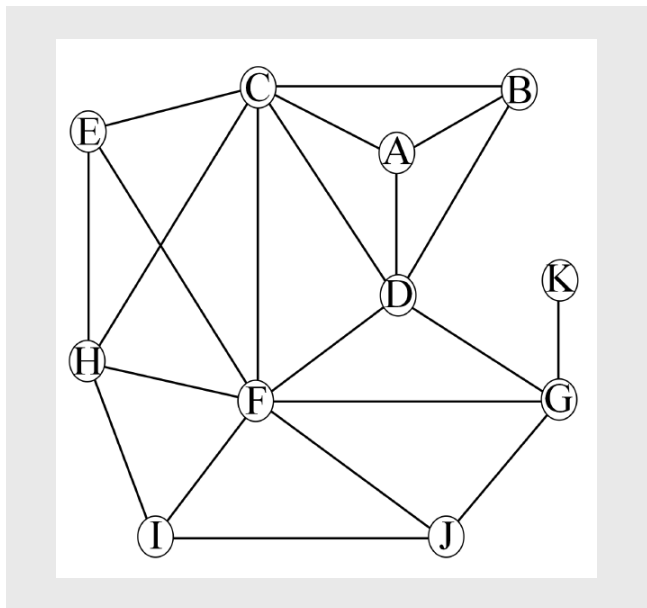


圖 2 實際網路連線情形

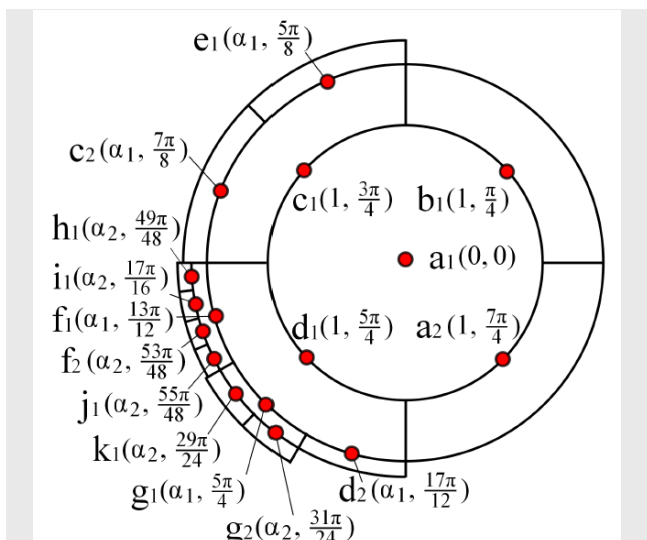


圖 3 分配虛擬座標，其中 $\alpha_1 = 1/\cos \frac{\pi}{4} + \delta$ ， $\alpha_2 = \alpha_1/\cos \frac{\pi}{12} + \delta$ 。

中，A點是這個起始點。A點會先被分配到一個虛擬座標 a_1 ， $(0,0)$ ，注意這裡的座標是用極座標 (polar coordinate) 表示。同時A點會為這個虛擬座標劃分出一個區域 $R(a_1)$ 。我們把 $R(a_1)$ 用四個欄位 $(0,1,0,2\pi)$ 來記錄，分別代表這個區域的內半徑、外半徑、起始角度、結束角度，如圖3中間的圓形所示。接下來，A會發出一個封包，通知它周圍的其它節點B、C、D，它剛剛被分配的虛擬座標與區域資訊。這個控制封包內還會包含一個串列 $FL(A)$ ，它包含了深度優先搜尋(Depth-First Search; DFS)演算法在由A、B、C、D所組成的子圖(subgraph)中間節點造訪的順序，也就是 $FL(A) = (B,C,D,A)$ 。收到這個封包的節點就會開始分配虛擬座標。分配的方法如下：

(1)看 $FL(A)$ 內有幾個節點，就將封包內存的角度區間分成幾等分，依序分給每個節點。而每個節點的角度座標就設為分得角度區間的正中間。

(2)距離座標則是封包內存區域的外半徑。這樣一來，如圖3所示，A、B、C、D四個點都各被分配到一個虛擬座標 a_2 、 b_1 、 c_1 、 d_1 。 a_2 、 b_1 、 c_1 、 d_1 也會被分配到相對應的區域，其中角度區間就是前面透過 $FL(A)$ 分得的角度區間，而內半徑就是 a_1 的外半徑，新的外半徑則會被設成比內半徑的 $1/\cos\theta$ 大的值，其中 θ 是所分配到的角度區間。例如 c_1 分配到的區域 $R(c_1)$ 就會被記錄成 $(1, 1/\cos\frac{\pi}{4} + \delta, \frac{\pi}{2}, \pi)$ ，其中 δ 是一個很小的常數。演算法的細節可以從[10]中找到。

在A、B、C、D分配完虛擬座標後，會檢查是否有相鄰的節點還沒有被分配到虛擬座標，然後重複之前的動作，將座標分配給還沒有座標的節點。其中因為A和B的所有相鄰節點都有至少一個虛擬座標了，所以A和B並不會再發送分配座標的指令封包。而假如D先發送了分配座標的指令封包，裡面包含的 $FL(D)$ 會是 (F,G,D) 。此時C要發送分配座標的指令封包時，它周圍就只剩下E還沒有拿到任何的虛擬座標，所以 $FL(C) = (E,C)$ 。每個分配到虛擬座標的節點都重複這個動作，最後整個網路的節點就會都分配到至少一個虛擬座標了，如圖3所示。

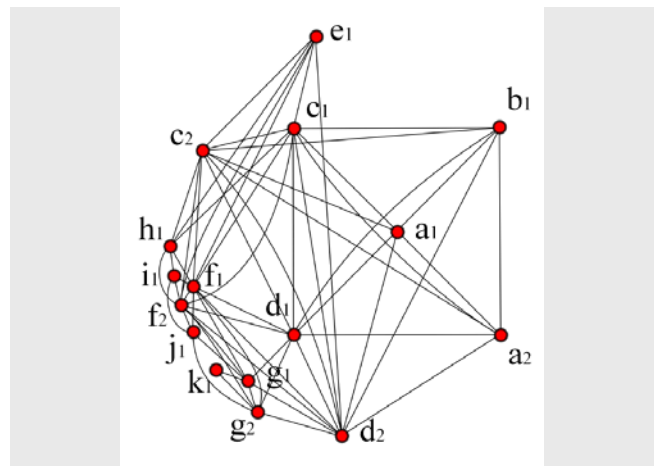


圖 4 建立虛擬網路圖

3.2 建立平面化圖形

在每個節點都有虛擬座標之後，我們要先將代表網路的圖2轉化成一張虛擬的二維網路圖。由於每個節點可能會分配到超過一個虛擬座標，所以每個節點可能會對應到超過一個虛擬節點。屬於同一個節點的虛擬節點之間當然可以互相溝通，所以我們會在它們之間加上虛擬的邊。而對照圖2，若兩個虛擬節點屬於可以互相溝通的兩個節點，則我們也在它們之間加上虛擬的邊。最後的虛擬網路圖如圖4所示。

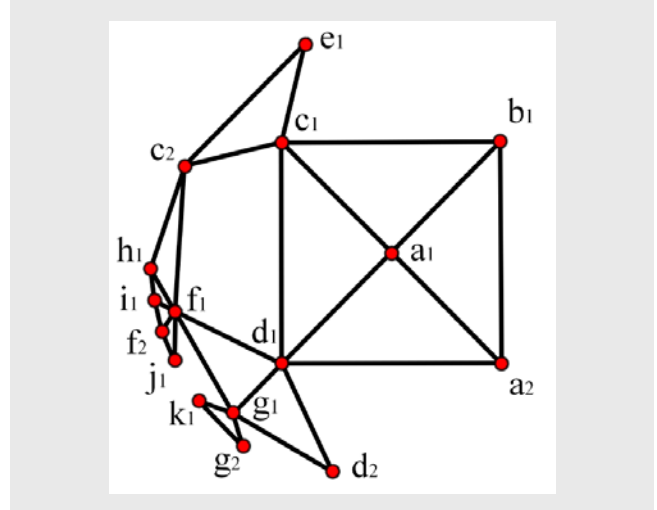


圖 5 取得虛擬平面化結果

這時，我們就能開始取得平面化的虛擬圖形，取得方法如下：若兩個虛擬節點所分配到區域是相鄰的，而且它們之間有一條邊相連，那麼我們就把這條邊加入平面化的虛擬圖形中。例如 c_2 和 h_1 之間有一條邊而且 $R(c_2)$ 和 $R(h_1)$ 相連(可參考圖4和圖3)，所以 c_2 和 h_1

之間的邊會被加入平面化的虛擬圖形中，如圖5所示。

是否保證取得平面化的虛擬圖形和分配虛擬座標的方法有很大關聯，並不是隨意分配的虛擬座標和區域都能用這個方法取得平面化的圖形，而其中的關鍵在於分配區域的內外半徑差距。從圖3我們可以看出來，其實這些分配的區域會形成一個的類似同心圓的架構。每一層同心圓的厚度如果不足，就不能夠保證這個方法取得的圖形是一個平面化的圖形。而我們在分配虛擬座標時所設定的外半徑需大於內半徑的 $1/\cos\theta$ 倍，就是滿足平面化的最低半徑要求，我們可以設定任何大於 $1/\cos\theta$ 倍內半徑的值，都能夠保證讓我們得到平面化的圖形。詳細的證明請參考[10]的定理一。

3.3 決定路由路徑

在取得平面化圖形之後，可以開始運行GFG的右手定則。在GFG中，先使用貪婪演算法 (Greedy Algorithm) 盡量地將封包送到更靠近目的地的地方。但是貪婪演算法並不能保證將封包送達，有些時候，封包會卡在某個節點，而相鄰的節點都比這個節點離目的地更遠。這個時候，就需要靠右手定則來幫我們解決這個問題。

當有一個封包要從K點送往E點時，我們會先找到K點的一個虛擬座標 k_1 ，然後將目的地設定成E點的一個虛擬座標 e_1 。我們會發現沒有其它與 k_1 相連的虛擬節點比 k_1 更靠近 e_1 ，所以開始使用右手定則。在使用右手定則的時候，我們只看平面化圖形，想像人站在 k_1 的位置，將右手放在 k_1 上沿著平面化圖形走，這樣我們就遇到了下一個虛擬節點 g_4 ，而G就被我們選擇為路由路徑上的下一個節點，並將封包傳送給G。

在節點G，我們會進行同樣的動作，先檢查是否可以使用貪婪演算法決定下一個節點，這時我們就發現有與 g_4 相連的其它節點比 g_4 更靠近 e_1 ，而我們透過貪婪演算法選擇了其中最靠近 e_1 的 f_1 ，並將封包傳給 f_1 由擁有者F。最後封包就成功地被送往了E。

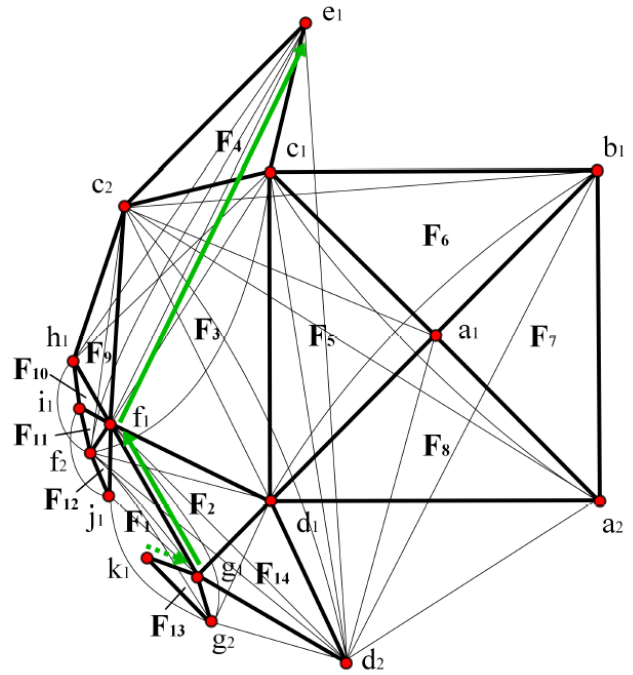


圖 6 決定路由路徑

3.4 適應動態網路

最後我們要介紹，在動態網路中，要怎麼迅速的維護虛擬座標，讓網路維持高效能不受影響。動態網路的各種行為，可以簡化為節點離開與節點加入這兩種，隨著節點從甲地離開到乙地加入則可以組成各種動態網路的移動行為。因此，我們在此用兩個範例說明LF-GFG在有節點離開或加入時，會如何處理。

如圖7所示，假設D節點離開了網路，網路的連線情形從圖2變成了圖7，這時候首先與節點D相連的節點會發現節點D已經離開了，然後

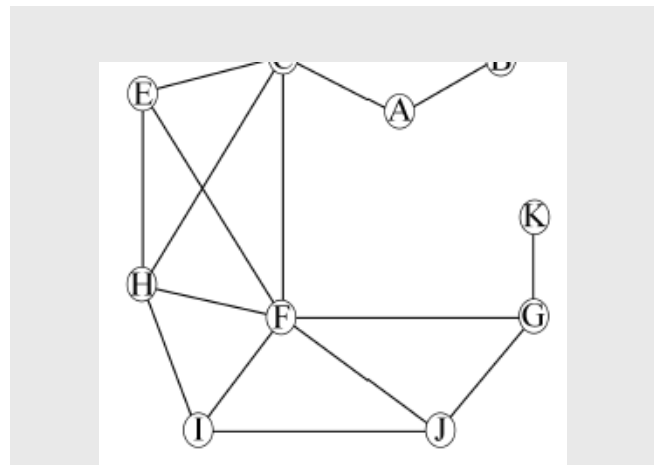


圖 7 D節點離開後實際網路連線情形

屬於節點D的虛擬節點會被移除，與這些虛擬節點相連的虛擬邊也會被移除。最後的虛擬座標分配情形如圖8所示。

當有一個新的節點L加入這個網路時，與L相鄰的節點會發現L還沒有虛擬節點，並發送一個分配虛擬節點的控制封包給L。假設是由節點B先發現L的加入，並且要發送分配虛擬節點的控制封包，這個控制封包內會包含B的一個最外層的虛擬節點資訊，還有 $FL(B) = (L, B)$ 。這個例子中因為B只有 b_1 一個虛擬節點，所以會將 b_1 的資訊傳給L。當L收到這個控制封包後，B和L都會再分配到新的虛擬節點，如圖10中的 l_1 和 b_2 所示。

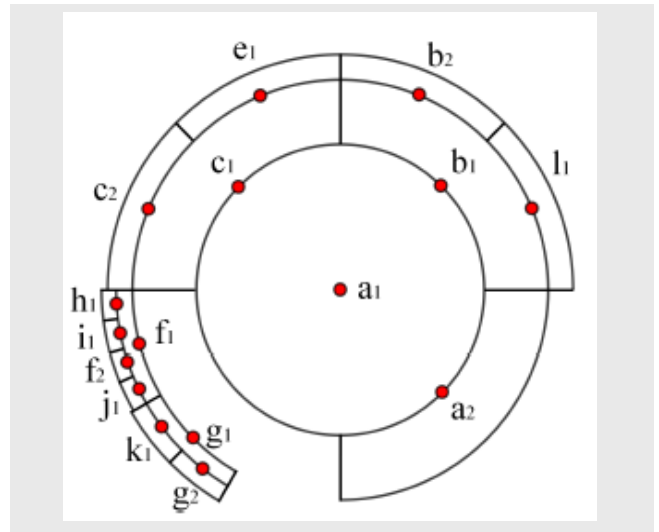


圖 10 L節點加入後虛擬座標分配情形

在上述的範例中，我們了解到為什麼 $FL(X)$ 這個序列裡面需要包含送出控制封包的節點在裡面。這是為了讓每個節點都至少有一個屬於最外層的虛擬節點。也就是說，這個虛擬節點的區域外相鄰的區域，還沒有分配給其它的虛擬節點。如此一來，當有新的節點加入時，我們就保證可以找到這種虛擬節點，發送控制封包分配新的虛擬節點給這些新加入網路的節點。

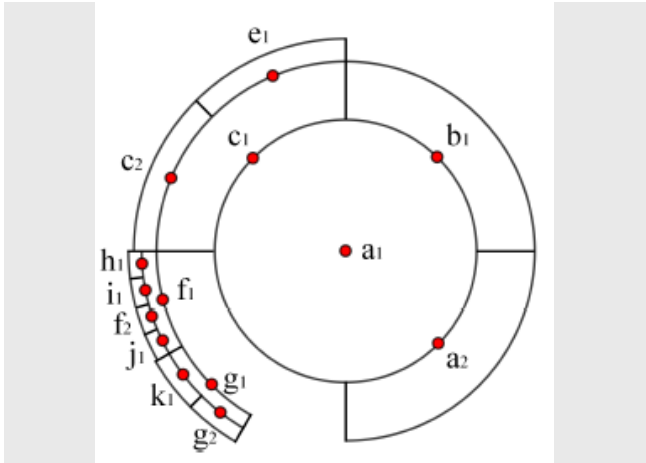


圖 8 D節點離開後虛擬座標分配情形

4. 實驗結果

我們透過了網路模擬器 NS-2 來進行實驗。在實驗中，我們實作了 APIT+GFG (黑色) [14]、ABVCap (綠色)、VPC (藍色)、和 LF-GFG (紅色) 四種路由協定。我們隨機建立了 100 張網路圖，每張網路圖都有 450 個物聯網的裝置。實驗持續了 500 個時間單位，在第 50 個時間單位的時候，我們讓網路發生隨機的 1 個 (粗線)、5% 個相連 (虛線)、5% 個分散 (細線) 的節點故障，觀察節點故障對網路效能的影響。每 0.1 個時間單位，網路中就有隨機一個裝置送出一個封包到另一個隨機選擇的裝置，我們將 10 秒內 100 個網路圖的所有封包傳輸結果平均之後，得到了下面的實驗結果。

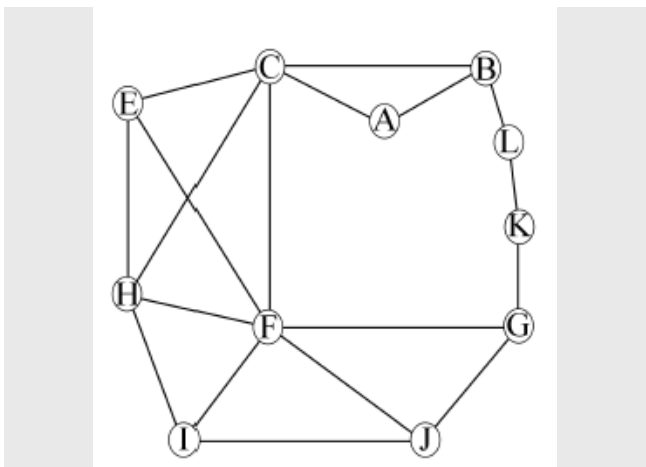


圖 9 L節點加入後實際網路連線情形

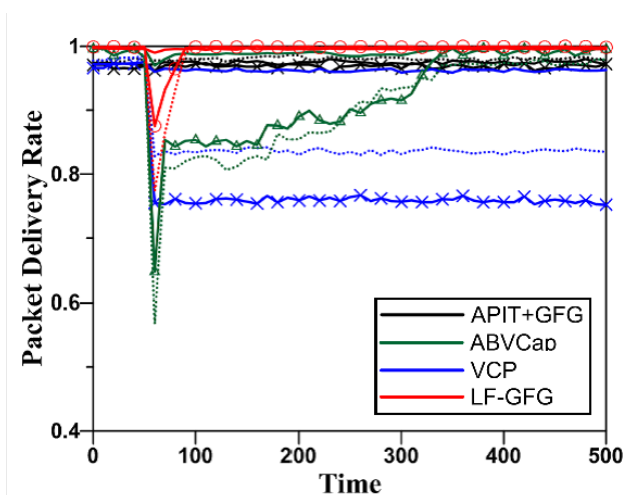


圖 11 封包傳達率比較圖

圖11顯示了封包傳達率的實驗結果，我們可以發現，在網路內有450個節點時，雖然這四個方法在理論上都有百分之百的封包傳達率，但是實際實驗卻會有封包傳輸失敗的情形發生。其中APIT+GFG會傳輸失敗是因為它需要假設每個節點的傳輸半徑都是一個完美的圓，但是電腦程式會有小數點精確度的限制，無法達到這個假設。當第50個時間單位發生節點故障時，VPC因為沒有維護的機制，會導致封包傳達率大幅下降且難以回復；APIT+GFG受影響比較小是因為它假設每個節點都是被散布在一個二維平面且都有精確的座標，在這個情況下，GFG本身就能理論上有百分之百的封包傳達率。LF-GFG在節點壞掉時受影響較小也是因為它使用了GFG的右手定則的關係。ABVCap和LF-GFG都有維護的機制，所以隨著維護機制的啟動，我們會發現封包傳達率降低後會慢慢的回升，其中我們發現，由於LF-GFG的維護機制十分簡單迅速，所以封包傳達率回升的速度比ABVCap高出很多。

圖12顯示了封包傳輸時間的比較圖，封包傳輸的時間除了受到路徑長度的影響之外，還會受到網路壅塞程度影響。從圖中我們會發現，雖然LF-GFG較能夠適應動態的網路環境，但是由於它的虛擬座標分配並沒有真的平均分散在二維平面上，當使用GFG的右手定則時會容易產生一些熱點，讓這些熱點的負擔很大，造成網路壅塞，拉長傳輸時間。當有節點壞掉LF-GFG進行維護之後，因為維護時分配的虛擬

座標，會在較外層的位置，所以反而讓虛擬座標的分配較為分散，降低了網路壅塞的情形。

在實驗中我們還比較了LF-GFG和ABVCap在維護動態網路時所需要的封包量。同樣是5%相連節點壞掉時，在LF-GFG中平均只需要使用9.3個封包就能夠使網路效能恢復，而ABVCap則平均需要112.9個封包。當只有1個節點壞掉時的數據則是1.3與3.7個封包，從這個數據我們可以發現LF-GFG有較好的容錯效能。實際上，我們發現假如只有1個節點壞掉，有60%的時候是不會影響網路效能的。

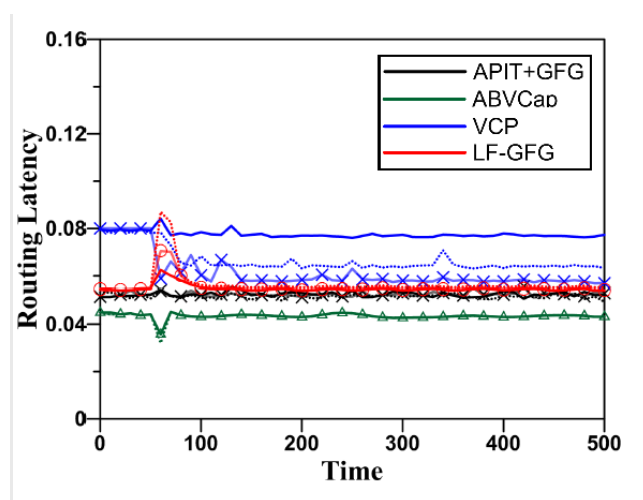


圖 12 封包傳輸時間比較圖

5 · 結論

這篇文章中我們介紹了一個適用於大規模動態物聯網的路由協定。從NS-2實驗結果我們可以發現，LF-GFG在大規模動態物聯網中，的確有較好的表現。相較於其它的路由協定，它有較好的容錯率，較低的維護成本，和較快的維護速度。更多的實驗結果和分析證明，可以在[10]中找到。

LF-GFG提出了一個新的路由概念，結合了傳統位置路由協定所使用的右手定則和虛擬座標系統的易實現特性，達到適應動態環境的目的。但是LF-GFG在很多細節的地方其實並沒有十分計較它的效能，其實還有很多可以改善的地方。舉例來說，LF-GFG分配虛擬座標時，假如可以依照目前的節點密度，來決定要使用的區域外半徑，就能夠大幅度的改善效能。LF-GFG分配出來的虛擬節點也有其連結特

性，若能利用這個特性動態交替使用右手定則和左手定則，也能夠大幅縮短傳輸路徑的長度。希望LF-GFG的提出與後續的改良，能夠讓大規模的動態物聯網更快在我們的日常生活中實現。

參考文獻

- [1] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing", RFC3561, 2003.
- [2] David B. Johnson and David A. Maltz, "Dynamic source routing in ad hoc wireless networks", Mobile Computing, 1996, pp. 153-181.
- [3] Y. Tseng, S. Ni, E. Shih, "Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc network," IEEE Transactions on computers, vol. 52, 2003, pp. 545-557.
- [4] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in ACM MobiCom, 2000.
- [5] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with guarantee delivery in ad hoc networks," ACM Wireless Networks, vol. 7, 2001, pp. 609-616.
- [6] F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger, "Geometric ad-hoc routing: Of theory and practice," in ACM PODC, 2003.
- [7] M. Caesar, M. Castro, E. B. Nightingale, G. O'Shea, and A. Rowstron, "Virtual ring routing: network routing inspired by DHTs," in ACM SIGCOMM, 2006.
- [8] A. Awad, R. German, and F. Dressler, "Exploiting virtual coordinates for improved routing performance in sensor networks," IEEE Transactions on Mobile Computing, vol. 10, 2011, pp. 1214-1226.
- [9] M.-J. Tsai, H.-Y. Yang, B.-H. Liu, and W.-Q. Huang, "Virtual coordinate-based delivery-guaranteed routing protocols in wireless sensor networks," IEEE/ACM Transactions on Networking, vol. 17, 2009, pp.1228-1241.
- [10] Y. Cheng; Y. Tang; M. Tsai, "LF-GFG: Location-Free Greedy-Face-Greedy Routing With Guaranteed Delivery and Lightweight Maintenance Cost in a Wireless Sensor Network With Changing Topology," IEEE Transactions on Wireless Communications, vol.13, 2014, pp.7025-7036.
- [11] S. Ruehrup and I. Stojmenovic, "Optimizing communication overhead while reducing path length in beaconless georouting with guaranteed delivery for wireless sensor networks," IEEE Transactions on Computers, vol. 62, 2013, pp. 2440-2453.
- [12] K. R. Gabriel and R. R. Sokal, "A new statistical approach to geographic variation analysis," Systematic Zoology, vol. 18, 1969, pp. 259-287.
- [13] K. J. Supowit, "The relative neighborhood graph, with an application to minimum spanning trees," Journal of the ACM, vol. 30, 1983, pp. 428-448.
- [14] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in ACM MobiCom, 2003.

作者簡介

鄭元博



現任工研院資通所網路視訊系統部工程師，清華大學資訊工程研究所博士。

E-mail: goldmoun@itri.org.tw

郭慧冰



現任工研院資通所網路視訊系統部工程師。交通大學資訊工程研究所碩士，專長於視訊壓縮、數位影像處理、及視訊串流。

[E-mail: joe.hpkuo@itri.org.tw](mailto:joe.hpkuo@itri.org.tw)