

MIP6-MANET: 實作一整合式行動 IPv6 和隨意行動網路

MIP6-MANET: The Integration and Implementation of Mobile IPv6 and Mobile Ad Hoc Networks

陳裕賢 楊允軒 黃仁竑
Yuh-Shyan Chen, Yun-Hsuan Yang, Ren-Hung Hwang

Department of Computer Science and Information Engineering
National Chung Cheng University
Chiayi, Taiwan, R.O.C.

Email: {yschen, yyh91, rhhwang}@cs.ccu.edu.tw

摘要

在本篇論文中，我們主要說明如何實作一個整合式行動 IPv6 和隨意行動網路(MIP6-MANET)系統，此系統將整合無線 Ad hoc 網路和 Mobile IPv6 技術，我們將實做出隨意行動網路(MIP6-MANET)此系統，以便提供使用者一無拘無束的行動上網空間，由於 Mobile IPv6 技術僅適用於聯繫 single hop 內的 AP，無法直接和相鄰的 Mobile Node 做直接的通訊，相反的、ad hoc 模式雖然可以經由多個 hop 連結到較遠的電腦，但是確有無法連結至 Internet 的問題，此篇論文主要說明我們如何實作整合式行動 IPv6 和隨意行動網路系統，讓每一個 Mobile Node (MN) 可以先透過 Ad-Hoc 的方式連結至最近的 AP，再連接到 Internet，並進而整合在 Linux 下的 MIPV6 套件，以實踐了在 IPv6 下整合 Mobile IPv6 與 MANET 的無線網路環境，並整合 MIPV6 套件下所提供移動性管理機制，如此一來便能將 Mobile IPv6 完全整合在無線 MANET 的無線網路環境中。

1. 簡介

隨著無線網路目前蓬勃的發展，有越來越多的人投入無線網路的研究，隨著最近幾年的演進，目前無線網路主要可以分為 Ad hoc 無線網路環境架構，以及 Infrastructure 環境架構，在 Infrastructure 環境架構中主要強調的是使用者可以自由自在的移動，隨時的透過 AP 連上 Internet，但是其條件是必須是要能在一個 Hop 就到他溝通的 AP，所有的通訊都必須透過 AP 來轉送至 Internet，縱使有兩台 Mobile Node 在附近，他們也必須要透過 AP 來做通訊，並無法直接的通訊，相反的、Ad hoc 無線網路環境主要是在強調使用者可以隨時隨地的建立出一

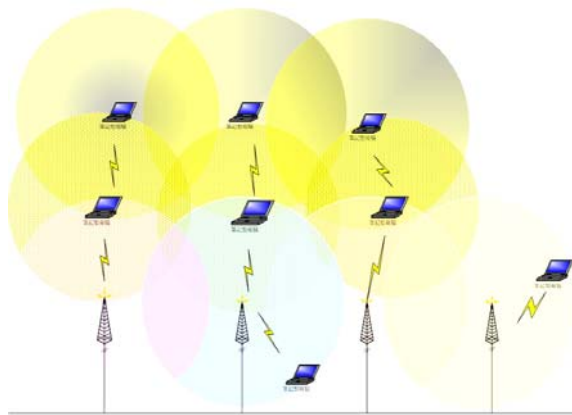
封閉的私人網路，而不需要透過 AP 來做通訊，而使用者與使用者之間只要透過相同的 Ad hoc routing 即可相互交換資料，此種通訊的方法很適用於一些需要立即建立通訊的地方，如戰場在對戰前不太可能預先架好 AP，因此 Ad hoc 的技術就可以提供一個隨建及連的網路通訊環境。

而隨著目前 IP 漸漸不夠用，以及 IPv4 早期的一些缺點和不良處所演發出來的 IPv6 [22, 23] 中也有支援無線網路的移動性，以及自動取得 IP [6, 7] 的功能，因此我們就以現今的 Mobile IPv6 協定 [1, 2, 9, 15] 以及早期 Ad hoc 的協定做整合，使得每一台電腦不但可以相互的以 Ad hoc 連結，並且也可以透過 Mobile IPv6 的協定，使得 Mobile Node 也可以有移動和漫遊的能力。

此篇論文主要是透過 IPv6 [22, 23] 來將 Mobile IPv6 的功能以及 Ad Hoc 的功能實做整合在一起，如圖一所示，在此系統中我們的作業系統都是採用 Linux 作業系統 [19, 20, 21]，且每一台 MN 都有配置一張無線網卡，但是只有和外部相連的那台電腦有連接有線網路線，將扮演 AP 的角色，此外，為了能夠達成 Ad Hoc 網路之間相互交換訊息，我們所採用的 routing protocol 為 DSDV，此外、為了能使 ICMPv6 封包 [1, 3, 4] 也可以經由 multi-hop 來傳送，我們必須實做一支轉送以及模擬 mobile IPv6 的程式，詳細的流程我們將在之後討論。

而我們此篇論文的整體架構如下：我們將會在第二段中將先對 MIP6-MANET 此系統的整體架構做一個初步的介紹，接著我們將在第三段中說明我們的 MIP6-MANET 系統的實做環境，而在第四段將說明我們 MIP6-MANET 系統整個運作的細部流程以及相關的系統細部設定，而在最後為這篇論文

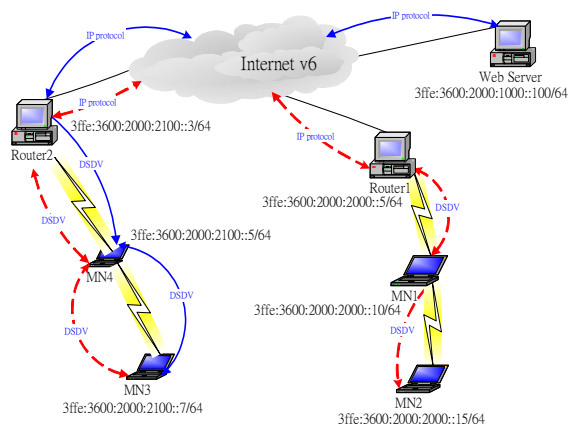
的結論、未來可能的繼續發展、致謝以及此篇論文相關的參考文獻。



圖一、MIPv6-MANET 系統整合示意圖

2. 系統架構介紹

此部分主要是要使Ad hoc 無線網路環境能透過我們所設的router來將封包轉送至Internet，由於Ad hoc 無線網路環境提供電腦與電腦間的通訊可以直接由網路卡來溝通，並不需要經過Access Point，透過此特性，我們將所有無線網卡都設為Ad hoc 模式，如圖二所示。



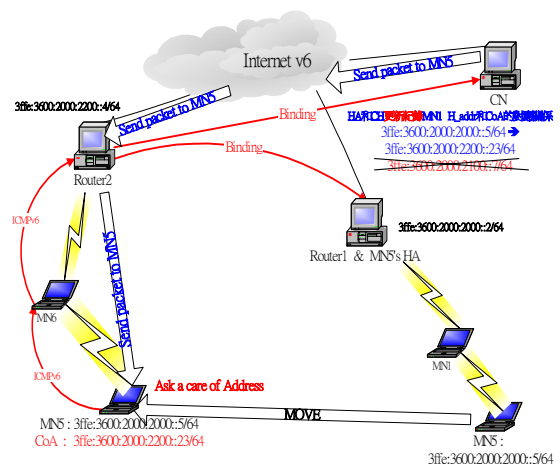
圖二、Multi-Hop情境圖

所有MN以及Router的無線網路卡都設為Ad hoc 模式，而Router1和Router2是裝有Linux作業系統的筆記型電腦，其主要功能為扮演這 Access Point 的角色，所以其forwarding的功能需要開啟，Routing Table也必須設定將無線網卡的封包重新轉移至有線網卡上，並將其送至 IPv6 網際網路上，此外每一個 Mobile Node (MN) 裡面還需要有一個程式定期去告知其周遭鄰居其IP，以便建立出到 Router (AP) 的路徑，且每一台MN的forwarding功能也必須都開啟，如此在中間的 Mobile Node才可以幫助其他Mobile

Node轉送封包至Router或是由Router再將封包轉送回Mobile Node。

在圖二主要是在說明我們在建立IPv6-based multi-hop routing protocol 時的情境圖，我們會建立兩個 IPv6 的網段 [5, 6, 7]，分別是 3ffe:3600:2000:2100::/64 和 3ffe:3600:2000:2000::/64 兩網段，首先MN3會將封包以紅色虛線斷的路徑先將封包轉送至Router2的無線網路卡，再透過Routing Table的設定[22]，再將封包轉送至有線網卡，並將其送至Router1之後再將封包轉送至MN2。另一情境為MN3先將封包依藍色線轉送至Router2，之後再透過Router2將封包轉送至MN3欲連結的web 網站，並由此網站下載所需軟體回MN3。

此部分主要是提供MN一個移動的環境[11, 12, 15, 17]，使得MN可以由原先的subnet做通訊時，移動到另一個subnet，而原先和MN做通訊的電腦CN，並不需要改變傳送的IP Address，其運作流程如圖三所示：



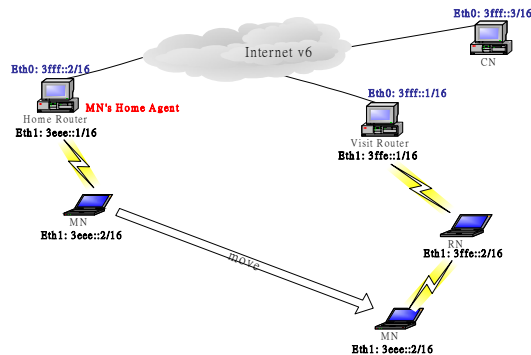
圖三、Mobile IPv6運作圖

Step1 Mobile Node 獲取一新的IP: 當MN5由3ffe:3600:2000:2000::/64 網段移動到3ffe:3600:2000:2200::/64網段時，此時router2 會先發送一 CoA 給 MN5 [1, 2, 6, 7]。

Step 2 Mobile Node 做Binding [1, 2, 12, 16, 19] 動作: 之後MN5傳送一binding information 給他subnet下的HA以及之前有和他連線的電腦，告訴他們他目前的IP Address，使得他們可以繼續傳送封包。

Step 3 和CN 做Binding: 此外，如果CN之前有在和MN5連線，此時MN5由某一-subnet移到另一-subnet，在CN收到binding後，他會知道MN5目前的IP，因此就不需做叫沒效率的三角路徑[2, 15]，因此也可以改善，封包傳送的效率。

在最後一部分中我們主要任務是將之前兩部分做整體的測試，並將其做整合，在此最後的整合情境如圖四所示。



圖四、IPv6-based Multi-hop routing protocol和Mobile IPv6整合環境圖

我們將網路分成三段，分別為3eee::/16、3fff::/16以及3ffe::/16三個區段[5]，而MN在此情境中扮演著移動的節點，他會由3eee::/16網段移動到3ffe::/16網段下，且MN的距離無法直接收到visit router的訊號，因此封包必須透過RN來轉送，如同之前Multi-hop所描述的。

在此情境中RN會不斷的發送ICMPv6的訊息[1, 5]，此訊息中會帶有此subnet的prefix訊息(3ffe::/16)，因此當MN移動到此subnet下時就會收到RN發來的ICMPv6封包，因此就透過此prefix在加上他的MAC Address來形成一CoA[2, 15, 16, 17]，同時會透過DSDV得知連到Internet的routing路徑訊息，因此MN會先修改其routing table，使其可以透過Multi-hop連至visit router，透過visit router來將封包轉送至internet，之後MN會透過此路徑來對其Home router做binding的動作，此binding message就會透過RN以及visit router來轉送至MN的home router。

當Home router收到此binding[2]之後就會修改其binding table以便可以記錄MN目前的CoA所對應其原先的Ipv6 Address，之後會修改其routing table[21, 22]使得封包可以順利的轉送至MN目前的subnet下，使得CN可以將其封包轉送至MN。

當MN收到CN傳的封包之後會再對CN做binding的動作，使得CN也可以得知MN目前的CoA，如此一來CN便可以透過此CoA來和MN做直接的連結，而不需透過Home router來轉送，如此一來可以解決原先的三角路徑問題[2, 15, 16]，如此一來可以提升封包傳送的效率。

3. MIP6-MANET 實作環境說明

在本段落中我們將介紹 MIP6-MANET 的實做環境，首先我們先建制一個以 IPv6 為通訊協定的

multi-hop 環境，使的每一個 MN 都可以相互的以 IPv6 協定做通訊，之後每一 MN 再透過 DSDV 做封包轉送的協定。

接著我們會介紹如何對 Linux 做相關的 Routing table 設定，使得每一個 Mn 可以透過 multi-hop 上網連出 Internet 環境可以實現出來此外別人也可以傳送封包透過 multi-hop 的機制將封包轉送給你，並且我們也將介紹其各細部的每一運作流程圖。

而在最後一個段落中我們將介紹 Mobile IPv6 的相關系統設定，使得每一 MN 不但可以透過 multi-hop 上網並且還可以有移動漫遊的特性，使得每一個使用者的通訊範圍將不再限制在一個 Hop 至 AP 的通訊範圍裡，而是可以透過其鄰居轉送封包至最近的 AP，之後再透過此 AP 上網，此外我們也將介紹其細部的運作流程圖。

A. 區域IPv6-based multi-hop routing的建制

在此項目中每一個MN會定期的交換Routing Path的訊息，以便每一台MN可以知道如何到達他subnet下的Router，其作法主要是透過一DSDV程式來相互溝通其可行走的路徑，此外為了使每一 Mobile Node在不是收到是要給他的封包時會再將此封包依據之前DSDV所建立出的Routing table來將封包轉送至目的地，我們必須將每一台MN Linux系統中的forwarding選項給打開，之後程式會將得知的訊息真正的寫到Linux的Routing Table[21, 22]中，之後MN就可以正常的發送封包給他鄰近的電腦，如此一來即可達到MN做區域性的Multi-hop。

B. Router (Access Point) 的相關設定

為了能使內部的Mobile Node可以連至外界，在完成以上子項目後，我們已經可以做到在一個subnet做multi-hop，如圖二所示，MN3已經可以將封包傳送至router2，但是還出不了Router2，由於router2有兩張網卡，一邊連結無線網路端，另一張為連接有線網路端，由於兩張網卡的subnet不同，如果沒有經過特別的設定，當MN欲傳送至subnet以外的電腦時，將會因為不同的subnet，因而封包會被檔下，因此我們還要針對封包要去的地方做Routing table的設定，在Routing table裡，我們必須加入欲連結到區域(Router1)的IP Address，對方也必須加入我們的IP address，如此一來封包才可以順利的由某一subnet，經由router轉送至另一subnet，詳細流程請參見圖五，我們將整個過程分為八個步驟來討論，以下將會一一來介紹：

Step 1 獲取 routing path 訊息: 當 MN1 和 MN2 分別

開機或是進入到此 subnet 時，會定期發送 multicast 封包告知周遭的鄰居 route path 相關訊息，以便同 subnet 下的 Mobile Node 可以知道如果封包不是要給他的話，他要如何將封包轉送至 Router 端，而我們實做的 routing 方式為 DSDV。

Step 2 查詢自己的 routing table: 當 route 的路徑已由先前的 DSDV routing protocol 決定時此時 MN3 欲傳送封包給 CN，此時 MN3 會先去查詢他自己系統目前的 routing table 而得知他目前周遭有和他相鄰的 Mobile node 有那些，以便透過他們來將封包轉送至 CN。

Step 3 幫忙轉送鄰居的封包包: 此時 MN4 會聽到 MN3 傳來的封包，由於每一台 MN 的 forwarding 機制都有打開，所以此時 MN4 就會如同一台 router 會將他所聽到不是給他的封包，經由查詢他自己的 routing table 而將此封包轉送至下一站。

Step 4 router2 轉送封包至不同網卡: 最後封包會被 router2 的無線網路卡 eth1 所接到，此時這台由 PC 扮演的 router2 會將封包轉送至有線網路卡 eth0 上，之後封包就會經由網際網路傳送至 CN，此傳送過程則是以我們目前 IPv4 的 protocol 來傳送。

Step 5 CN 收到封包: CN 收到 MN3 所傳來的封包，之後再回送封包回 MN3。

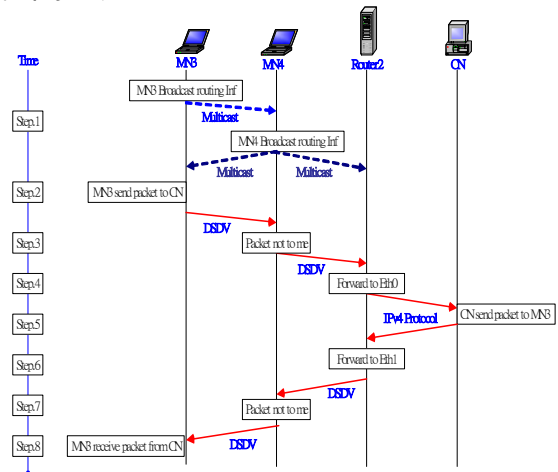
Step 6 在將封包轉回無線網卡: router2 由有線網路卡 eth0 接收到 CN 所傳來的封包，先去察看 routing table 發現此目的地是在無線網路端那 subnet，所以在將封包轉送至無線網路卡 eth1 上並將封包由 eth1 送出。

Step 7 幫忙轉送鄰居的封包包: MN4 聽到 router2 所傳來的封包，他會知道此封包不是要給他的，於是會先去查詢他的 routing table 看看目的地需要經由哪一個 MN 來幫他 forward，之後再將此封包繼續經由無線網卡傳送至下一個 mobile node。

Step 8 MN3 收到 CN 的封包: 最後 MN3 會收到 CN 送出經由 MN4 轉送的封包。

我們可以由圖四來說明我們最終實做的情境圖，Mobile Node 所有進出的封包都必須透過 RN 來做轉送其通訊範圍並沒有辦法在一部內到達他範圍下的 AP，而 Mobile Node 以及 visit router 之所以可以知道要將封包傳送至 RN 就是透過圖五所示的 DSDV 協定來達成，每一節點平時都會不斷的以 DSDV 協定來建立他們的 routing table 資訊，以便之後各節點

要傳送封包時可已有一個依循的根據來將封包做轉送或是收下。



圖五、資料傳送示意圖

C. Mobile IPv6 的系統環境設定

● MIPL Mobile IPv6 for Linux

1. Kernel Space :

MIPL [19, 21, 22] 必須先作 Linux kernel patch 的動作，因為需要更改 ipv6 kernel stack 的部分，所以也需要重新編譯 kernel。

在 HA 與 Mobile Node 的選項也有些許的不同，一個 Linux kernel 只能擇一編譯成 HA (Home Agent) 或是 MN (Mobile Node)，不可同時編譯成 HA 及 MN。以下解說 patch 的方法。

Step1: 由 <http://www.mipl.mediapoli.com/> 下載 mipv6-0.9.5.1-v2.4.20 至 /usr/src。

Step2: 在 /usr/src 目錄中解開檔案後，進入 kernel source 的目錄，進行 patch

```
# patch -p1 < /usr/src/mipv6-0.9.5.1-v2.4.20/mipv6-0.9.5.1-v2.4.20.patch.
```

Step3: 執行 make menuconfig，設定 kernel 為 HA 或是 MN #makemenuconfig，設定完畢後，.config 檔案中 Mobile IPv6 部分的選項如下所示：

```
CONFIG_IPV6_MOBILITY_HA = y (for Home Agent)
或是 CONFIG_IPV6_MOBILITY_MN = y (for Mobile Node).
```

2. User Space :

MIPL 有提供一個 user space 的工具，執行命令為 mipdiag，也必須先編譯它。

```
# cd /usr/src/ mipv6-0.9.5.1-v2.4.20
```

```
# ./configure.
```

```
# make ; make install.
```

3. 增加 MIPv6 Device

```
# mknod /dev/mipv6_dev c 0xf9 0.
```

4. 修改在 HA 中，MIPL 的設定檔 /etc/network-mip6.conf

```
#Home Agent configuration file
```

```
FUNCTIONALITY=ha
```

```
DEBUGLEVEL=1
```

```
TUNNEL_SITELOCAL=yes
```

5. 修改在 MN 中，MIPL 的設定檔 /etc/network-mip6.conf

```
#Mobile Node configuration file
```

```
FUNCTIONALITY=mn
```

```
DEBUGLEVEL=1
```

```
MIN_TUNNEL_NR=1
```

```
MAX_TUNNEL_NR=3
```

```
HOMEDEV=eth1
```

```
HOMEADDRESS=3ffe:3600:7:1000::6/64
```

```
HOMEAGENT=3ffe:3600:7:1000::1/64
```

● RADVD(Router ADvertisement Daemon)

RADVD[20, 21, 22]是 IPv6 的 router advertisement daemon，它能聆聽 router solicitations 與發送 router advertisements (Neighbor Discovery for IPv6, RFC 2461)。在 Debian GNU/Linux[22]上，網路安裝 radvd 的方式為 # apt-get install radvd

1. 修改在 HA1 中，RADVD 的設定檔

```
/etc/radvd.conf
```

```
interface eth1
```

```
{
```

```
    AdvSendAdvert on;
```

```
    MinRtrAdvInterval 3;
```

```
    MaxRtrAdvInterval 10;
```

```
    AdvHomeAgentFlag off;
```

```
    prefix 3ffe:3600:7:1000::/64
```

```
    {
```

```
        AdvOnLink on;
```

```
        AdvAutonomous on;
```

```
        AdvRouterAddr on;
```

```
    }
```

```
}
```

2. 修改在 HA2 中，RADVD 的設定檔

```
/etc/radvd.conf 與 HA1 相同，僅 prefix 不同，HA2
```

```
為 prefix 3ffe:3600:7:2000::/64。
```

4. MIP6-MANET 系統運作流程

最後我們將把上述兩項功能來做合併[9, 13, 14]，其情境如圖四所示，我們將網路分為三個網段，期中 3eee::/16 網段為原先 MN 的無線網路網段，3ffe::/16 為 MN 將移動到的無線網路網段，而 3fff::/16 為有線網路相連的網段。

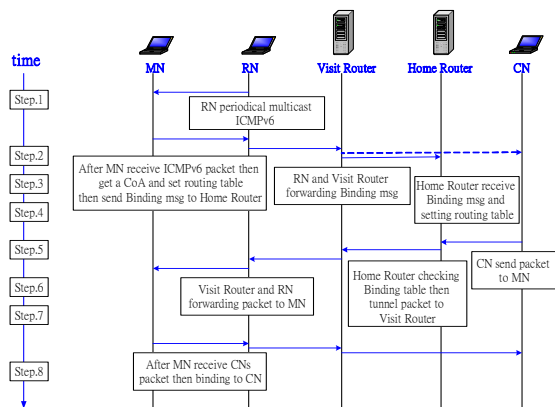
而整體流程為 MN 從原先的網段移動到某一網段下，其通訊的範圍無法直接和他的 visit router 做通訊，因此必須透過 RN 來做封包的轉送，而其運作的詳細細節我們將在圖六中做細部的解說，以及 MN 從他所拜訪的網段下在移回原先的 home network 時將會做的細部過程將會在圖七中做各部細節的解說。

A. MN 由 home network 移動到 visit network 下的流程圖

Step 1 RN 不斷的發送 ICMPv6 訊息: 在一開始時 RN 會以 ICMPv6 協定，定期不斷的發送含有他網域資訊下的 prefix[2, 5, 6, 7]，以便可以使得進入到此網域的 Mobile node 可以得知此網域下的 prefix 值是多少。

Step 2 Mobile Node 獲得一 CoA[2, 15, 16, 17] 之後對其 routing table 做設定: 當 MN 由他原先的 Home network 移動到 visit network 時，便會收到由 RN 發送出來的 ICMPv6 封包[1, 2] (此時 MN 的距離無法直接與 visit router 做通訊，需透過 RN 來轉送)，因此 MN 變會以收到的 prefix 在加上他的 MAC address 來產生一個 Care of address[2, 15, 16, 17]，之後便會設定其 routing table[21, 22] 使得他的封包可以透過 RN 來轉送至 visit router，之後便可以發送 Binding message 給他的 Home router，來達到 Binding 的動作。

Step 3 RN、visit router 幫忙轉送封包: 做上述 step 2 中，Mobile Node 所發送的封包接會被 RN 以及 visit router 來做轉送，而在圖六中虛線的部分為：假如 Mobile Node 之前有和某一 CN 做通訊的話，當他在做 Binding 的動作時也會同時和此 CN 做 binding 的動作[2]，使得 CN 也可以在第一時間得知 Mobile Node 最新的 Care of address 是多少，而之後及可以透過此一新的 CoA 來和 Mobile Node 做通訊。



圖六、MN 移動到 Visit Router 網路下的流程圖

Step 4 Home router 收到 binding 資訊後做相關設定: 當 Mobile Node 的 Binding message 透過許多的電腦轉送至 Home router 之後，Home router 就會將此資訊記錄至 binding table 中，因此便可以追蹤 Mobile Node 目前的 CoA 與他原先的 Home IP 作對應，此外 home router 也會修改其 routing table[21, 22]使得欲傳送至 Mobile Node 的封包可以轉送至 Mobile Node 目前所在的 visit router 下，並透過 RN 再轉送至 Mobile Node。

Step 5 CN 傳送封包給 Mobile Node: 有了以上的先前設定之後，當有某一使用者 CN 想要傳送封包給 Mobile Node 時，此封包便會先傳送至 Home router(Mobile Node 原先所在的網域下)。

Step 6 Home router 檢查其 binding table 而得知 MN 目前所在地: 當 Home router 收到 CN 傳來的封包時，會檢查他的 Binding table 得知 MN 目前不在他的網域下，此時 Home router 變會透過他先前所設定的資訊，將封包封裝[2, 3]便轉送至 MN 目前所在的 visit router 網域下。

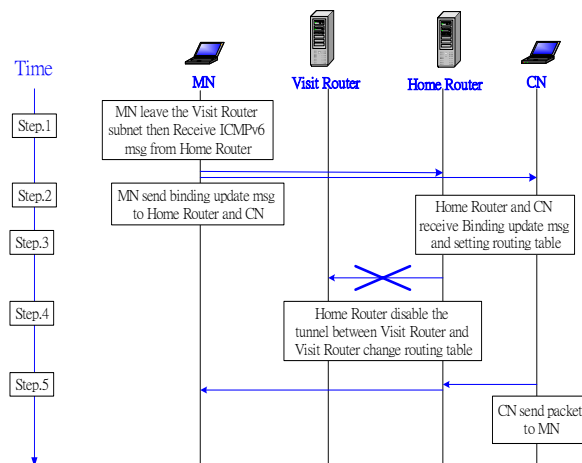
Step 7 RN、visit router 幫忙轉送封包: 當 visit router 收到由 Mobile Node 的 Home router 所轉送過來的封包之後，他會先將此封包做解封裝[2, 3]的動作，並且在察看其 routing table 並得知此封包要透過哪一台電腦 RN 來轉送至 Mobile node。

Step 8 Mobile Node 再對 CN 做 Binding 的動作: 最後當 MN 收到 RN 所轉送過來的封包後，MN 便會再對此 CN 做 Binding 的動作，並再次透過 RN 以及 visit router 的轉送至 CN，使得 CN 可以得知 Mobile Node 目前的 CoA，以便 CN 可以直接的透過此 CoA 來將封包直接的傳送至 MN，而不需先傳送至 MN 的 home router 再轉送給 visit router，因此便可

以解決此三角路徑[2, 15, 16]的問題，並可提升封包傳送的效率。

B. Mobile Node 由目前的 visit router 移回原先的 home router 網域下流程圖

接著將介紹 MN 由目前的 visit router 再移回他原先的 home router 網域下的細部流程圖，其流程圖如圖七所示，一共可以分成五個步驟來探討。



圖七、MN 移回 Home Router 網路下的流程圖

Step 1 Mobile Node 回到其原先 Home router 的網域下: 當 Mobile Node 由目前的 visit router 網域移回到他原先的 home router 網域時，便會收到 home router 所發送出來的 ICMPv6 訊息，之後會比較此訊息裡的 prefix [2, 5, 6, 7] 是否為他的 home network 的 prefix，如果不是他便會在做我們之前所述的一到四步驟，如果所收到的 prefix 為他的 home network 的 prefix 時，他便會對其 home router 做 binding update 的動作。

Step 2 Mobile Node 傳送 Binding update: 當 Mobile Node 檢查此 ICMPv6 封包裡的 prefix 訊息為他的 home network 時，他便會對其 home router 做 binding update [2] 的動作，如果之前有 CN 和他通訊，他也會在此時和之前的 CN 做 binding update 的動作，告知他目前已經回到 home network。

Step 3 Home router 以及 CN 收到 Binding update message: 當 Home router 以及 CN 收到 Mobile Node 傳來的 Binding update message 之後，home router 將會先修改其 binding table 以及 routing table，使得封包不在轉送至 visit router 下，而是直接的轉送至 home router 他目前的網域下，而 CN 收到 Mobile Node 傳來的

Binding update message [2]後也會對其 routing table 做修改，使其封包可以正確的傳送至目前 Mobile Node 所在的 home router 網域下。

Step 4 Home router 改變其系統相關設定: 此外 home router 還會取消之前和 visit router 的 tunnel 動作，使得可以將別人(CN) 傳來的封包直接傳送至 home router 他本身的網域下，而不需再轉送至 visit router 的網域下。

Step 5 CN 傳送封包給 Mobile Node: 當完成上述設定之後，此時如果有某使用者 CN 這時想要傳送封包給 Mobile Node 時，他就可以直接的將封包傳送至 home router，再透過 home router 直間轉送至 Mobile Node。



圖八、實體測試畫面

圖八為我們實機測試的情境圖，在圖中左側和右側兩台電腦分別是 Home Network 和 Foreign Network 的 router，可將其功能視為一台 AP，他們分別以圖中黃色的有線網路相接，此外、此兩台筆記型電腦因為需扮演 AP 的角色，因此也分別有安裝無線網卡與他們個別的 subnet 下的 Mobile Node 做溝通，而中間兩台電腦即為 Mobile Node，其中一台預設會是在左邊的 subnet 而另一台會在右邊的 subnet，而這兩台 Mobile node 皆只有安裝無線網卡，並無有線網路的溝通能力，且其無線網路通訊的模式接設為 Ad Hoc 的模式，而實際運作流程將會如同前述所說的方式，已開始時會有某一 Mobile node 由他原先的 Home network 移動到另一 Foreign 的 subnet 下，並且會先透過 ICMPv6 自動取得一 IPv6 address，並透過 multi-hop 以及 Mobile IPv6 的結合來達成相互通訊的能力。

5. 結論

在此篇論文中我們成功的將現有的無線網路通訊的範圍大大的提升，使得每一個 Mobile Node 和

AP 通訊的範圍可以更加的延伸，透過中間許多的 Mobile Node 來幫我們轉送封包至我們 subnet 下的 router，使得傳送距離不再是像以往只限制在 AP 電波可以傳送到的距離，更增加了 Mobile Node 的移動性，使得 Mobile Node 可以邊上網邊移動到另一個 subnet，使得網路的連線範圍可以在做更大的延伸，至於下一個目標，我們將試著將 GPRS 模組[8, 10, 13, 14, 17]加入到我們目前的系統中，使得此系統在可以使用 WLAN 上網時，將會盡可能的採用 WLAN 上網，當訊號 WLAN 強度下降時，此時系統將切換至 GPRS 的頻道下，使得 Mobile Node 可以透過我們目前電信業者連上網路，如此一來可以使得我們系統上網的範圍更加的完整無縫隙[13, 14, 17, 18]，之後也將試著改進 Handoff 的效率，使得 Mobile Node 在移動的同時也可以擁有較好的通訊品質，以及網路存取安全方面的機制，使得整個系統更加的完整。



圖九、Mobile Node 移至 VR 下 CN ping MN 的畫面

從圖九中看到，當 MN 移動至 visit router 下時，CN 以 ping 的方式來和 Mobile Node 做溝通的畫面，由於封包必須經過多次轉送之後才到達 Mobile Node，因此我們可以看到其反應時間會拉的比一般還要常，從圖九中可以看到其平均直約為 4~5ms，此直會比 Mobile Node 回到其原先的 subnet 下時反應的時間還要來的長一些，那是由於 Mobile Node 移回 home router 後，其通訊是直接和他的 home router 做通訊，不需要經過多餘的轉送，因此當 CN ping Mobile Node 時，封包是直接送至 home router 就到達 Mobile Node，因此其反應時間大約就只需要 2ms，比起 Mobile Node 移動到 visit router 下時的反應時間短些。

6. Acknowledgement

The research is supported by Taiwan NICI IPv6 Steering Committee, R&D Division. Under contract number R-0300.

7. 參考文獻

- [1] Conta, A. and S. Deering, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", *IETF RFC 1885*, December 1995.
- [2] David B. Johnson, Jari Arkko and Charlie Perkins, "draft-ietf-mobileip-ipv6-21.txt", *INTERNET-DRAFT*, November 2002.
- [3] W. Stevens, M. Thomas, E. Nordmark, T. Jinmei, "Advanced Sockets Application Program Interface (API) for IPv6", *IETF RFC 3542*, May 2003.
- [4] Robert E. Gilligan, Robert E. Gilligan, Jim Bound, W. Richard Stevens, "Basic Socket Interface Extensions for IPv6", *IETF RFC 2553*, March 1999.
- [5] S. Deering, R. Hinden, Internet Protocol, "IP Version 6 (IPv6) Specification", *RFC 2460*, December 1998.
- [6] T. Narten, E. Nordmark, W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", *RFC 2461*, December 1998.
- [7] S. Thomson, T. Narten, "IPv6 Stateless Address Autoconfiguration", *RFC 2462*, December 1998.
- [8] Haiyun Luo, Ramachandran Ramjeeey, Prasun Sinhas, Li Liy, Songwu Lu, "UCAN: A Unified Cellular and Ad-Hoc Network Architecture", *ACM MOBICOM*, 2003.
- [9] Yu-Chee Tseng, Chia-Ching Shen, and Wen-Tsuen Chen, "Mobile IP and Ad Hoc Networks An Integration and Implementation Experience", *IEEE INFOCOM 2002 Conference*, 2002.
- [10] Kilian Weniger, Martina Zitterbart, Institute of Telematics University of Karlsruhe, 76128 Karlsruhe, Germany, "IPv6 Autoconfiguration in Large Scale Mobile Ad-Hoc Network", *European Wireless*, Florence Italy 26-28, Feb 2002.
- [11] Rajiv Chakravorty, Pablo Vidales, Leo Patanapongpibul, Ian Pratt, Jon Crowcroft, "Inter-network Mobility with Mobile IPv6", *IEEE Communications Magazine*, 2003.
- [12] Yu Zhang, Ziqiang Hou, "The Improvements on MIPv6 Based on NAI Name Resolution and Home Agent Discover", *IEEE ICCCN'03*, 2003.
- [13] Milind Buddhikot, et al., "Integration of 802.11 and Third-Generation Wireless Data Networks", *In Proceedings of IEEE INFOCOM 2003*, Volume: 1, Page(s): 503 -512.
- [14] Salkintzis A.K, Fors C. Pazhyannur R, "WLAN-GPRS integration for next-generation mobile data networks" *IEEE Wireless Communications*, Volume: 9, Issue: 5, Page(s): 112 -124, Oct. 2002.
- [15] Vaughan-Nichols, S.J., "Mobile IPv6 and the future of wireless internet access", *IEEE International Conference 2002*, Volume: 36, Issue: 2, Page(s): 18 -20.
- [16] Perkins, C.E, "Mobile networking through Mobile IP", *IEEE Internet Computing 1998*, Volume: 2, Issue: 1, Page(s): 58 -69.
- [17] Kempf, J, Wood, J, Guangrui Fu, "Fast mobile IPv6 handover packet loss performance", *IEEE Wireless Communications and Networking (WCNC) 2003*, Volume: 2, Page(s): 1230 -1235.
- [18] Cornall, T, Pentland, B, Khee Pang, "Improved handover performance in wireless mobile IPv6", *IEEE ICCS*, Nov. 2002, Volume: 2, Page(s): 857 -861.
- [19] "Mobile IP for Linux (MIPL)" Implementation by HUT Telecommunications and Multimedia Lab", <http://www.mipl.mediapoli.com>.
- [20] "Linux IPv6 Router Advertisement Daemon (radvd)", <http://v6web.litech.org/radvd/>.
- [21] Tcpdump reference (<http://www.tcpdump.org>) , and tcptrace reference (<http://www.tcptrace.org>), tcptrace+ (<http://www.cl.cam.ac.uk/users/rc277/soft.html>).
- [22] S. Deering [Cisco], R. Hinden [Nokia], "Internet Protocol, Version 6 (IPv6) Specification ", *IETF RFC 2460*, December 1998.
- [23] R. Hinden [Nokia], S. Deering [Cisco Systems], "IP Version 6 Addressing Architecture", *IETF RFC 2373*, Jul 1998.