

IPv6-SIP-VoIP 在 TWAREN 網路上的 跨校整合測試 (2005)

國立中山大學

電機系 許蒼嶺副教授

電機系碩士班研究生 陳佳駿

電機系碩士班研究生 杜家銘

電機系碩士班研究生 黃啟任

協助完成：國立成功大學 黃悅民 教授

及研究生

章節目錄

第一章	Pv6-SIP-VoIP 測試架構.....	1
1.1	SIP over wireless Infrastructure mode 的網路架構.....	1
1.1.1	peer-to-peer mode.....	1
1.1.1.1	測試目的.....	1
1.1.1.2	測試方法步驟與說明.....	2
1.1.2	SIP proxy mode.....	4
1.1.2.1	測試目的.....	4
1.1.2.2	測試方法步驟與說明.....	6
1.2	SIP over wireless Ad Hoc mode 的網路架構.....	6
1.2.1	測試目的.....	6
1.2.2	建置無線 Ad Hoc 網路平台.....	7
1.2.3	測試方法步驟與說明.....	8
第二章	IPv6-SIP-VoIP 的測試結果.....	10
2.1	Infrastructure mode 的測試結果.....	11
2.1.1	peer-to-peer mode.....	11
2.1.2	SIP proxy mode.....	12
2.2	在無線 Ad Hoc mode 的測試結果.....	15
2.2.1	Single stream.....	15
2.2.2	Multiple streams.....	16

第三章 未來測試工作與遭遇的問題.....	18
3.1 整合多個大學校園的 SIP 測試.....	18
3.2 設定路由器的 Priority Queues.....	18
3.3 Handoff 過程並非 smooth handoff.....	18
3.4 解決 multiple streams 的數量有一定的限制.....	18
3.5 Ad Hoc 部份目前僅能使用 static route.....	18
3.6 實現多台 SIP Proxy Server 的互連.....	19

圖表目錄

圖一	peer-to-peer mode(無 handoff) 測試架構圖.....	1
圖二	peer-to-peer mode(有 handoff) 測試架構圖.....	2
圖三	Windows XP 無線網路連線圖.....	3
圖四	Windows XP 無線慣用網路設定圖.....	4
圖五	SIP proxy mode(無 handoff)測試架構圖.....	5
圖六	SIP proxy mode(有 handoff) 測試架構圖.....	5
圖七	在 Ad Hoc mode 下的 single stream 測試架構圖.....	6
圖八	在 Ad Hoc mode 下的 multiple streams 測試架構圖.....	7
圖九	ICMP PING Request/Reply 的程序.....	7
圖十	SIPv6 Analyzer	10
圖十一	EtherPeek VX	11
圖十二	透過有線與無線 AP 及發生 handoff 在 peer-to-peer mode 下 audio 的 jitter 分布.....	11
圖十三	透過有線與無線 AP 及發生 handoff 在 peer-to-peer mode 下 video 的 jitter 分布.....	12
圖十四	SIP Session 建立方式.....	13
圖十五	透過 SIP proxy server 及 peer-to-peer mode 的連線流程比較.....	13
圖十六	透過有線與無線 AP 及發生 handoff 在 SIP proxy server 連線下 audio 的 jitter 分布.....	14
圖十七	透過有線與無線 AP 及發生 handoff 在 SIP proxy server 連線下 video 的 jitter 分布.....	14
圖十八	透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 audio 的 jitter 分布.....	15
圖十九	透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 video 的 jitter	

分布.....	15
圖二十 透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 audio 的 jitter	
分布.....	16
圖二十一 透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 video 的 jitter	
分布.....	16
表一 Ad-Hoc 網路平台的規格.....	8

第一章 IPv6-SIP-VoIP 測試架構

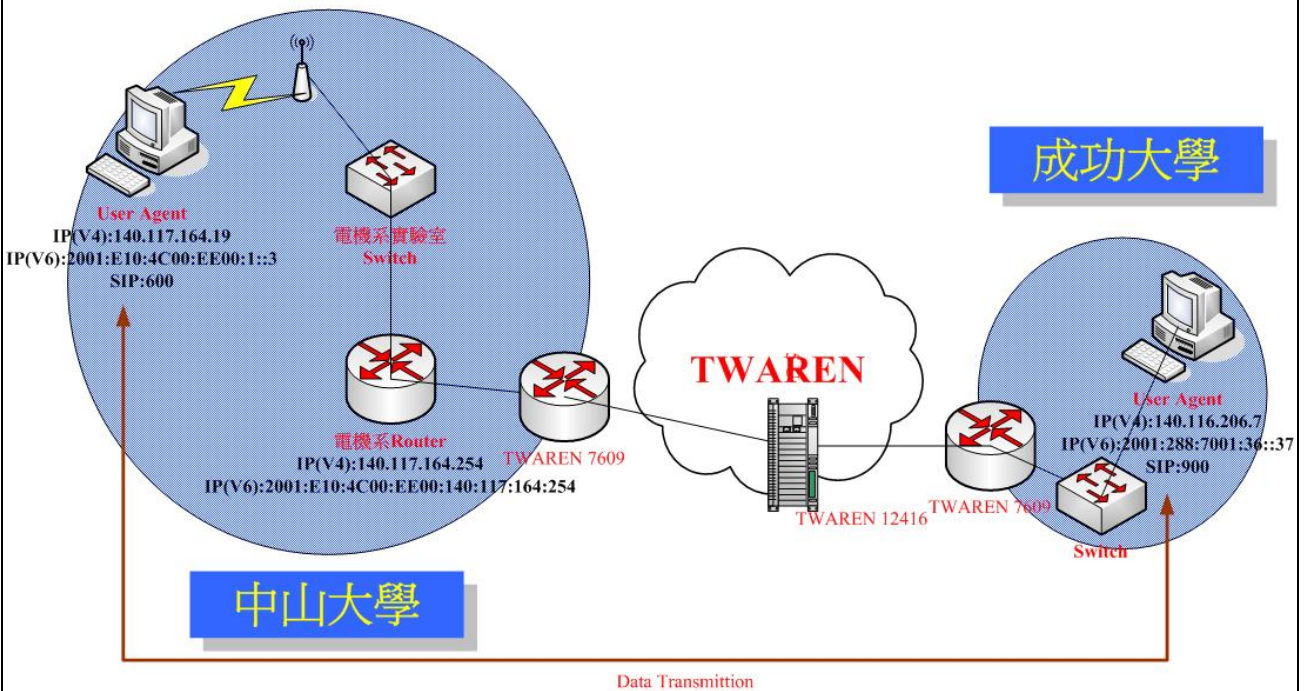
1.1 SIP over wireless Infrastructure mode 的網路架構

1.1.1 peer-to-peer mode

1.1.1.1 測試目的

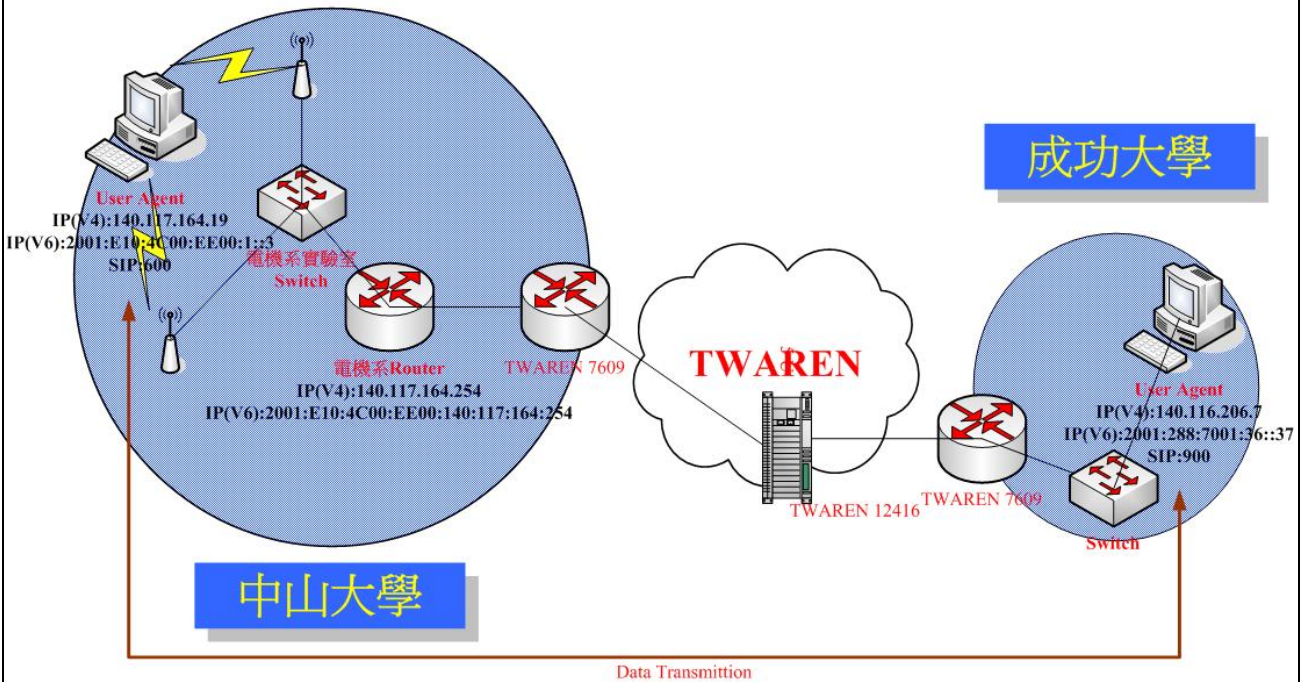
本測試架構目的為在 IPv6 跨校區網路環境下，經由 peer-to-peer mode 建立 SIP-VoIP 連線，其中一端 User Agent 位於無線 802.11 的 Infrastructure Mode 下透過 AP 與另一端連接在有線網路的 User Agent 通訊，量測並觀察連線情形。實驗中，區分為無 handoff 的測試架構<圖一> 及通訊中在兩台 AP 間發生一次 Handoff 的測試架構<圖二>。

End user經由無線AP連線通訊之拓撲圖



圖一、peer-to-peer mode(無 handoff) 測試架構圖

End user經由無線AP連線且在通訊中發生Handoff之拓撲圖



圖二、peer-to-peer mode(有 handoff) 測試架構圖

1.1.1.2 測試方法步驟與說明：

我們將一台安裝無線網卡的 PC 作為中山大學的 User Agent，另外放置兩台 AP 在中山大學校園網路中的電機大樓內，User Agent 經由無線網卡連線至 AP，再經由 switch 連線至中山大學校園網路，接著透過校園內的骨幹路由器 Juniper M160 走 IPv6 的骨幹網路，連線至成功大學計網中心內的 User Agent。中山大學與成功大學的 User Agent 都使用 PCA softphone 軟體來做雙向視訊通話。

我們在 AP 上的主要測試項目分為兩項：第一項測試是在中山大學無線網路環境下的 User Agent 透過一台 AP 連上網路；第二項測試則是將 AP 增加為兩台，先由其中一台 AP 連上網路，接著再 handoff 至另一台 AP。測試在此兩種情形下，SIP 使用 peer-to-peer 連線的結果。

在測試軟體方面，我們將測試軟體 SIPv6 Analyzer 安裝在中山大學無線網路環境下的 User Agent，以便量測在無線網路上的情形。每次連線測試時間為 30 秒，在每次的量測中做五次的連線測試，最後將五次連線測試的數據作平均。為使視訊有變化，因此在連線時，將視訊裝置做搖晃動作。

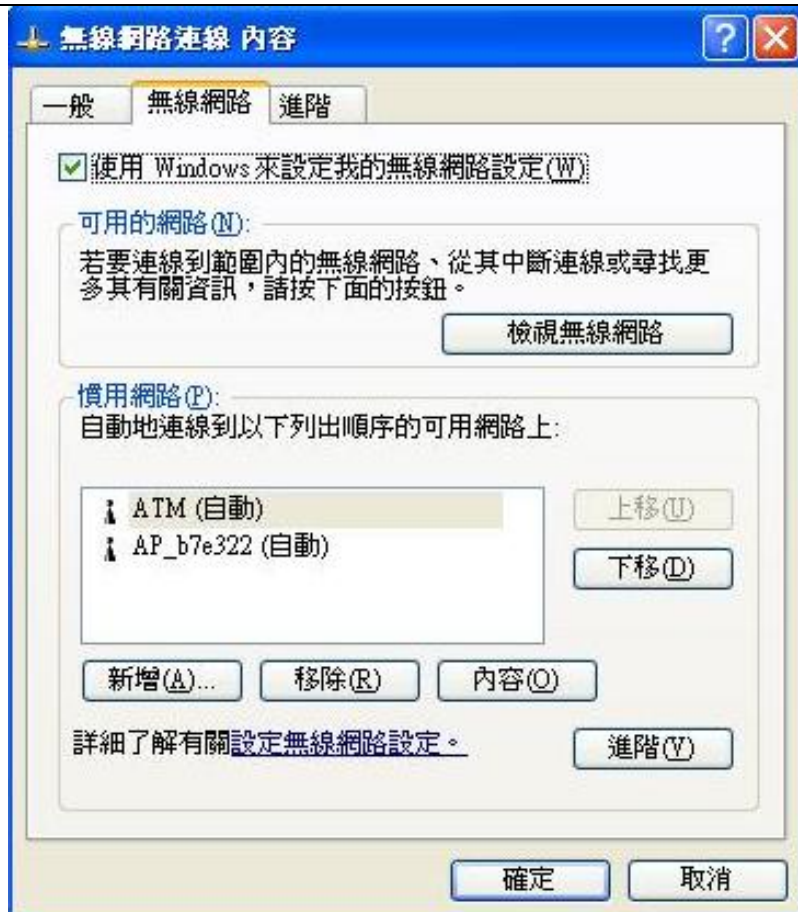
由於測試軟體對於記憶體的需求高，無法在 notebook 上使用，此外透過移動方式的 handoff，常常無法成功而顯示斷線。因此我們使用桌上型電腦，搭配無線 PCI 網卡。在 handoff 部分，我們則是強制將連線中的 AP 斷電，使電腦強制 handoff 至另一 AP 連上網路。hand off 連線測試中，在連線 10 秒時，強制將連線中的 AP 斷電，約略在 4 至 7 秒後，桌上型電腦顯示連線至另一台 AP。

為了能夠順利的在兩個不同的 AP 間 handoff，因此需要在 Windows XP 下做設定。點選開始->連線到->無線網路連線，接著出現下列視窗<圖三>。此視窗顯現出偵測到不同 ESSID 的無線網路。



圖三 Windows XP 無線網路連線圖

點選上列視窗中左列相關的工作欄中的變更喜好網路的順序，出現下列視窗<圖四>。可透過下列的視窗來新增想要自動連上無線網路的 ESSID。系統將會由上至下的順序來自動連線。當設定好所欲連線 AP 的 ESSID 後，便可以開始進行透過單一 AP 與由一 AP handoff 至另一 AP 的測試。



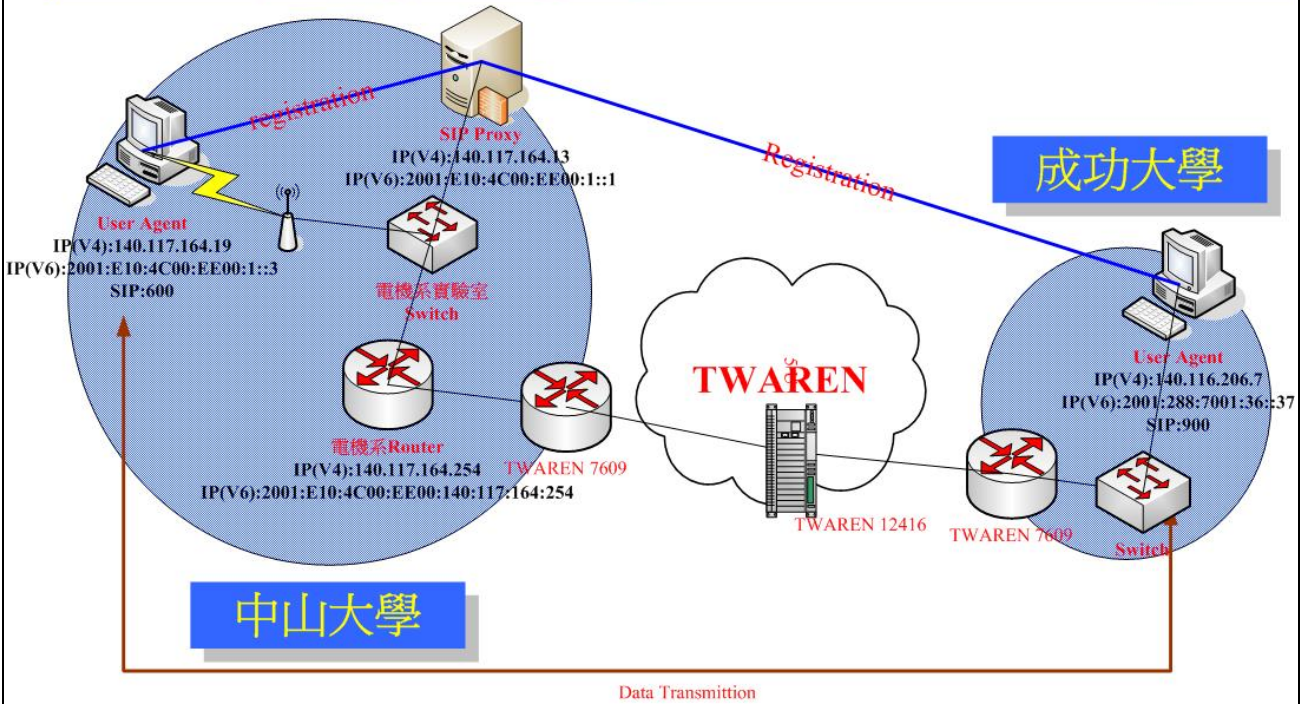
圖四 Windows XP 無線慣用網路設定圖

1.1.2 SIP proxy mode

1.1.2.1 測試目的

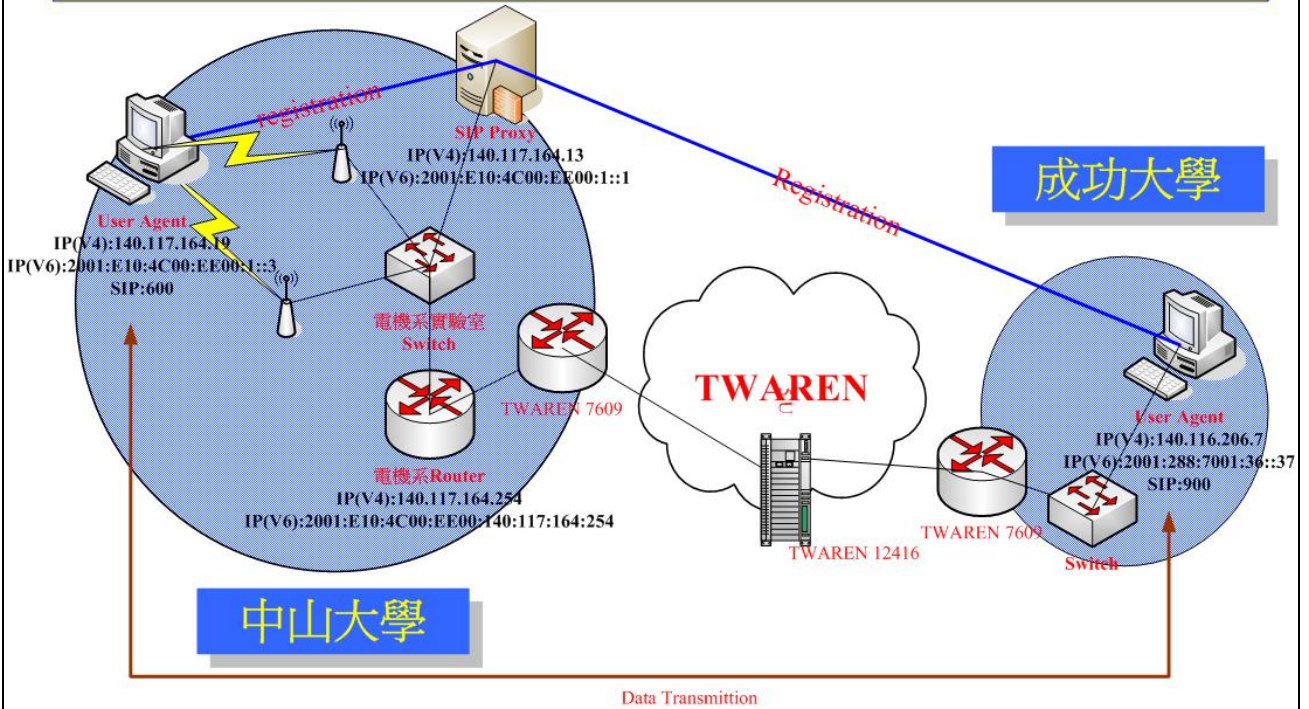
本測試架構目的為在 IPv6 跨校區網路環境下，不同的 User Agents 透過 SIP proxy 註冊並建立 SIP-VoIP 連線，其中一端 User Agent 位於無線 802.11 的 Infrastructure Mode 下透過 AP 與另一端連接在有線網路的 User Agent 通訊，量測並觀察連線情形。實驗中，區分為無 handoff 的測試架構〈圖五〉及通訊中在兩台 AP 間發生一次 Handoff 測試架構〈圖六〉。

End user經由無線AP連線並使用 SIP PROXY通訊之拓撲圖



圖五、SIP proxy mode(無 handoff)測試架構圖

End user經由無線AP連線且在通訊中發生Handoff並使用 SIP PROXY之拓撲圖



圖六、SIP proxy mode(有 handoff) 測試架構圖

1.1.2.2 測試方法步驟與說明：

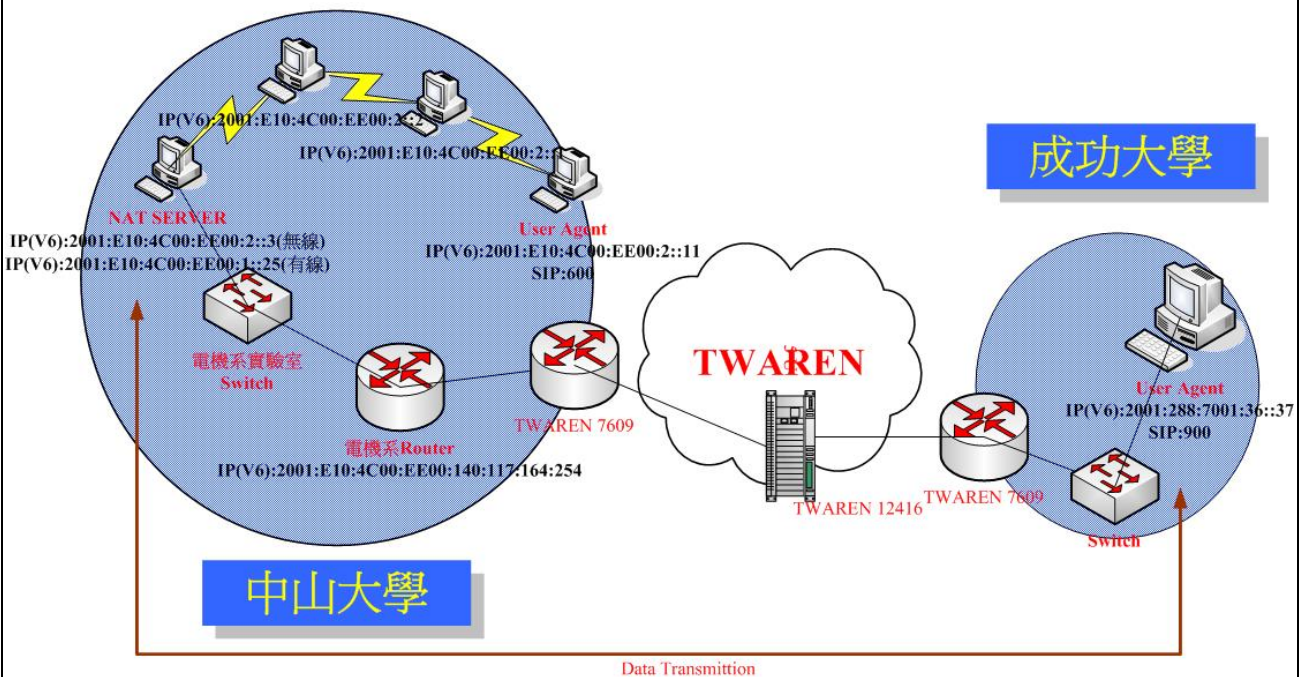
在測試透過SIP proxy server方面，首先要在PCA softphone內設定SIP proxy server的IPv6位置，接著PCA softphone便會透過AP的連線，向設定內所指定到的SIP proxy server註冊。接著鍵入對方所註冊的名稱，便可開始連線。透過SIP proxy server與peer-to-peer的差別，主要在於連線部分，在peer to peer的連線方式，需由使用者鍵入對方的完整IP位置，而透過SIP proxy server的連線方式，則是由對方的註冊名稱，向SIP proxy server取得對方IP位置。

1.2 SIP over wireless Ad Hoc mode 的網路架構

1.2.1 測試目的

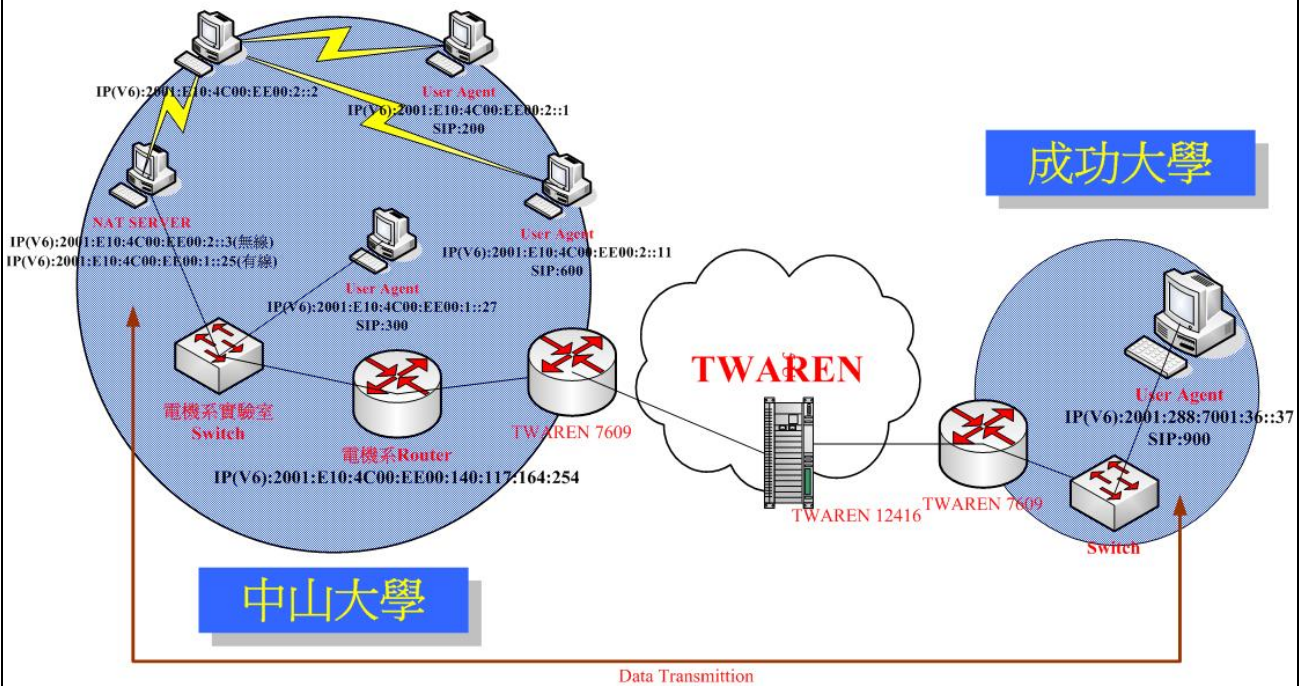
本測試架構目的為在 IPv6 跨校區網路環境下，一端 User Agent 位於無線網路 Ad Hoc 模式下與另一端連接有線網路的 User Agent 通訊，量測並觀察此時的連線情形。實驗中，區分為 single stream 測試架構<如圖七>及 multiple streams 測試架構<如圖八>。

End user在Ad Hoc網路下single stream通訊之拓撲圖



圖七、在 Ad Hoc mode 下的 single stream 測試架構圖

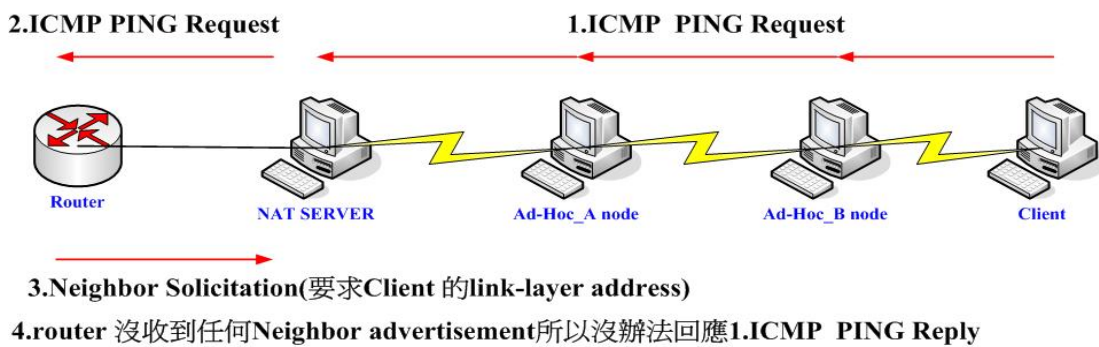
End user在Ad Hoc網路下multiple streams通訊之拓撲圖



圖八、在 Ad Hoc mode 下的 multiple streams 測試架構圖

1.2.2 建置無線 Ad Hoc 網路平台

在建立 SIP 連線之前，我們使用 ICMP PING 來確認通訊，但發現沒有回應，檢查之後得知問題出在於 NAT server 和 router 之間，ICMP Echo Request/Reply 程序如〈圖九〉。



圖九、ICMP PING Request/Reply 的程序

在 Ad-Hoc 部分，因為 routing tables 已經設定好的關係，ICMP PING Request 會送到 Next hop，因此發出的 Neighbor Solicitation 是向 Next hop 要求 link-layer address，Next hop 也會給予回應，等到 router 收到 ICMP PING Request 時，router 的 routing tables 中沒有 client 或 Ad-Hoc 中任何一台電腦的資訊，因此會將 client

當作是鄰近的點，發出 Neighbor Solicitation 要求 link-layer address，所以 router 收不到 Neighbor Advertisement，也就不會回應 client 的任何訊息。

我們採取的解決方式是在連接 Ethernet 和 Ad-hoc 的電腦上，加上 NAT 的功能，也就成了〈圖九〉中的 NAT server，由於目前沒有支援 IPv6 的 NAT 套件，因此我們修改 linux kernel 原始碼，寫出簡易的 NAT 功能，對外一律轉換成 NAT server 的 Ethernet 卡的 IP，等對方有回應時，再根據 port 對應回原本 IP，做到兩邊的通訊。

加上 NAT 功能後，會遭遇與 IPv4 環境下在 NAT 後端使用 SIP 相同的問題，由於 SIP 建立連線時，是根據 SIP Message 裡面的 Contact 和 Via 記載的資訊，來建立聲音及視訊的 RTP/RTCP session，而正常 NAT 最高只看到 TCP/UDP 層，對於更上層的 SIP Message，不會去做處理修正，所以產生的問題是只有單方向有聲音影像，或者甚至連線無法建立，因此我們的 NAT server 也加上對 SIP Message 處理，方式與我們的簡易 NAT 相同，將 Contact 和 Via 換成 NAT server 的 IP，以 NAT server 的 IP 來註冊與建立連線，比較需要注意的是，在連線建立之前，必須先在 port 對應的 tables 中，加入即將建立的 audio、video session 使用的 port，之後再將 NAT server 會轉換成對外使用的 port，修改到 SDP Message m=audio 和 m=video 中描述使用 port 的部分，這樣才能做到 multiple streams。

在無線 Ad-Hoc 網路實驗平台中，NAT Server、Ad-Hoc_A、Ad-Hoc_B、Client 各主機的規格詳列如〈表一〉；

表一 Ad-Hoc 網路平台的規格

	NAT Server	Ad-Hoc_A	Ad-Hoc_B	Client
CPU	AMD K8 2800+ 1.8 GHz	Pentium III 750 MHz	AMD K8 2800+ 1.8 GHz	Pentium III 933 MHz
RAM	1 GB	256 MB	512 MB	256 MB
O. S.	RedHat Fedora Core 2 Kernel-2.6.10- 771	Ubuntu 5.04 Kernel-2.6.10	RedHat Fedora Core 2 Kernel-2.6.10- 771	Windows XP Professional SP2
無線網卡	SMC 2082w 802.11 b/g 54 Mbps	SMC 2082w 802.11 b/g 54 Mbps	SMC 2082w 802.11 b/g 54 Mbps	Lantech WL54G-PCI 802.11 b/g 54 Mbps

1.2.3 測試方法步驟與說明：

先將電腦的無線網卡模式設定成 Ad-Hoc mode，再給定相同的 ESSID 以及 channel，最後再設定 IPv6 address，便可以互相通訊，由於使用的是桌上型電腦，無法任意移動，彼此之間距離太近，因此選定一台當 User Agent 後，其餘電腦放置

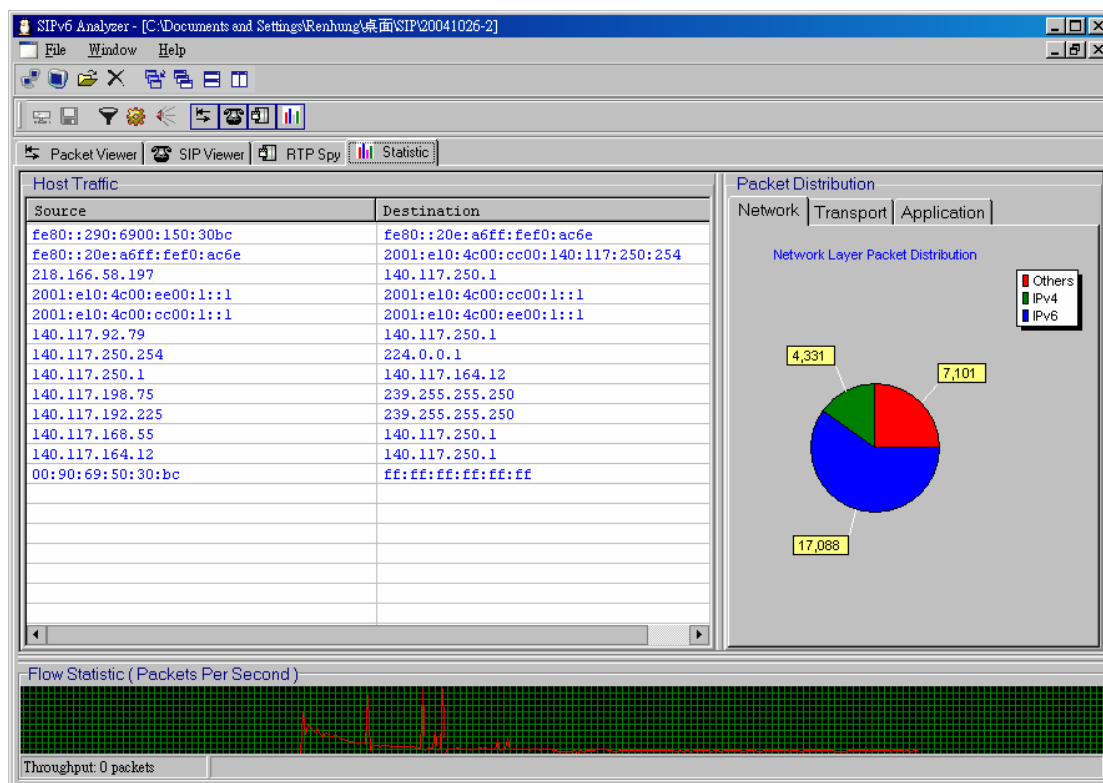
在中間，設定固定的 routing tables，來達到 multi-hop 的狀態，最後再將所有電腦的 IPv6 forwarding 功能開啟，即可讓 Ad-Hoc 下的電腦連上 Ethernet，Ad-Hoc 內的 User Agent 便可經由 NAT server 再透過 switch 向 SIP Proxy 註冊，之後再和成功大學的 USER Agent 建立 SIP 連線，測量數據，測量時間為 30 秒。

為了測試兩條 streams，我們從〈圖七〉做了一些修改，在 Ad-Hoc 內部放置兩個 User Agent，這兩個 User Agent 以及其中一個中間節點是可以相互通訊的，並且透過兩個 hop 後就可以透過 NAT server 來跟 Ethernet 部分的電腦通訊，另外再增加一個 User Agent 接到 switch 上，之後讓 Ad-Hoc 中隨意一個 User Agent 跟成功大學建立連線，而 Ad-Hoc 中的另一個 User Agent 再和新增的電腦連線，測量時間一樣為 30 秒，多次量測後，再比較兩條 streams 的數據。

第二章 IPv6-SIP-VoIP 的測試結果

我們在初期先使用交通大學的SIPv6 Analyzer，然後再使用廣聯科技代理的EtherPeek VX測試軟體。茲將兩項工具說明如下：

(1)交通大學資訊工程系 Lab117 與 Lab610 所發展的 SIPv6 Analyzer，包含主要功能有：Packet Viewer、SIP Viewer、RTP Spy、Statistic。



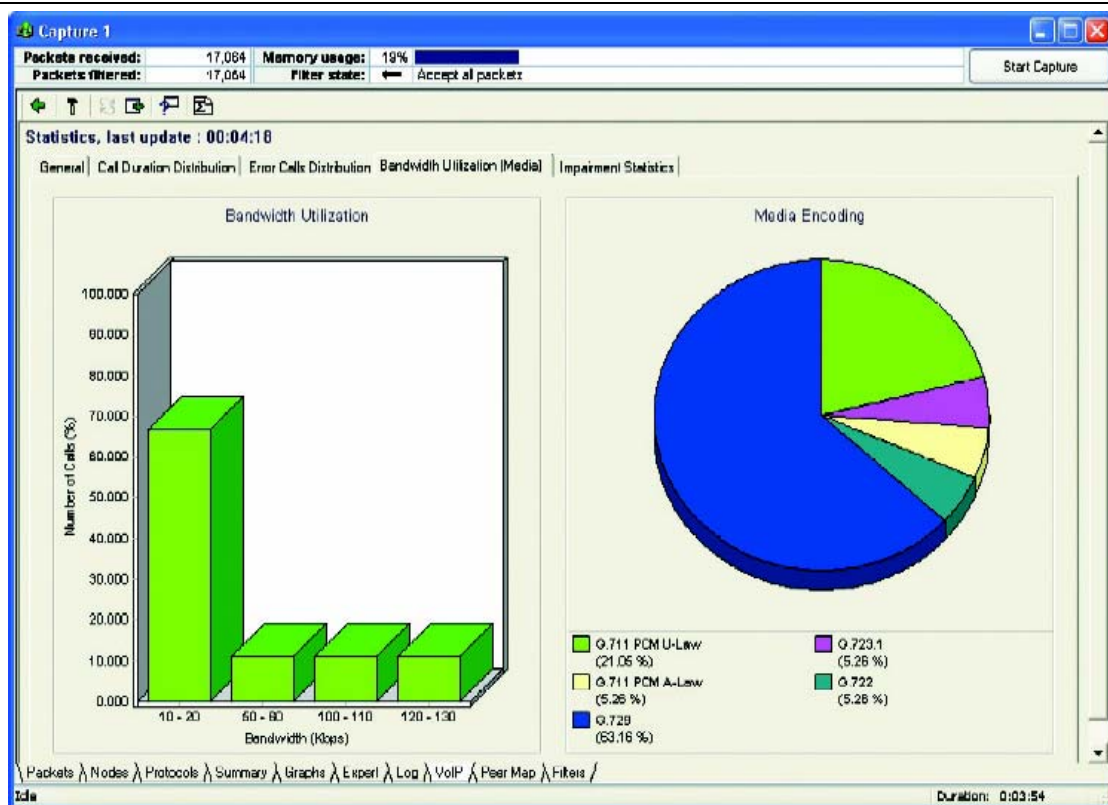
圖十、SIPv6 Analyzer

此工具可分析 User Agent 端接收到的 SIP 以及 RTP 封包，並且由這些 Offline 封包運算出 VoIP traffic 在固定的 Jitter 值下的 Packet Drop 值。

網址為 http://www.csie.nctu.edu.tw/~yhsung/SIPv6_analyzer/

(2)廣聯科技代理的 EtherPeek VX 軟體，包含主要功能有：Packet Generator(並可修改傳送之 packet 內容)、Online VoIP QoS Monitor、Average Packet Delay、Jitter、Average Packet Loss。

但與廣聯科技聯繫後，獲知 EtherPeek VX 軟體目前尚未支援在 IPV6 環境下分析 VoIP 的功能，故無法使用此軟體做 IPv6-SIP-VoIP 的量測與數據統計分析。

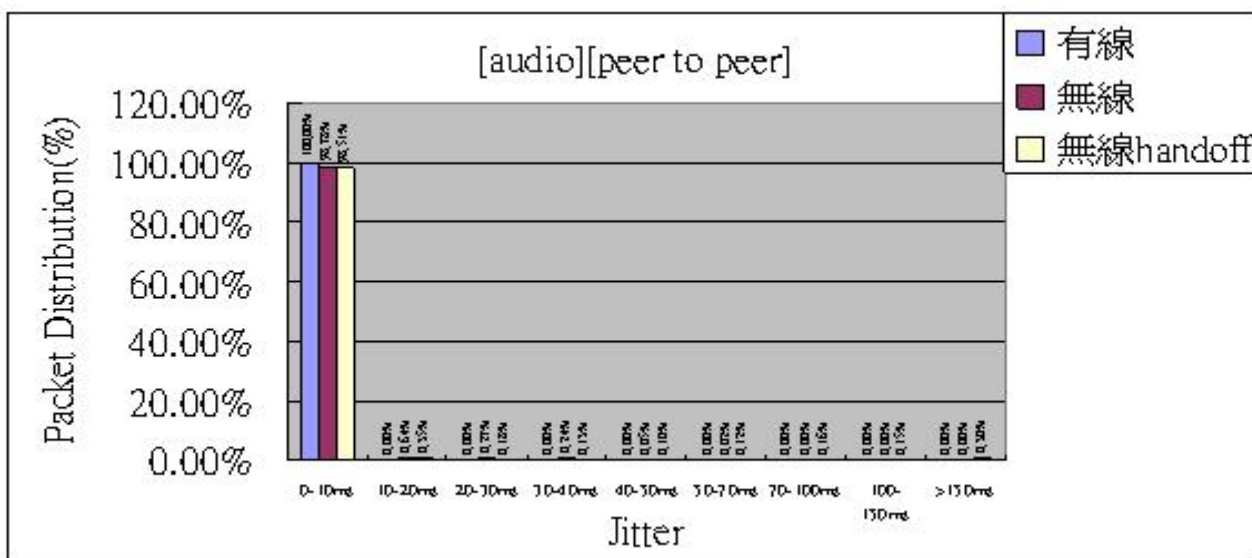


圖十一、EtherPeek VX

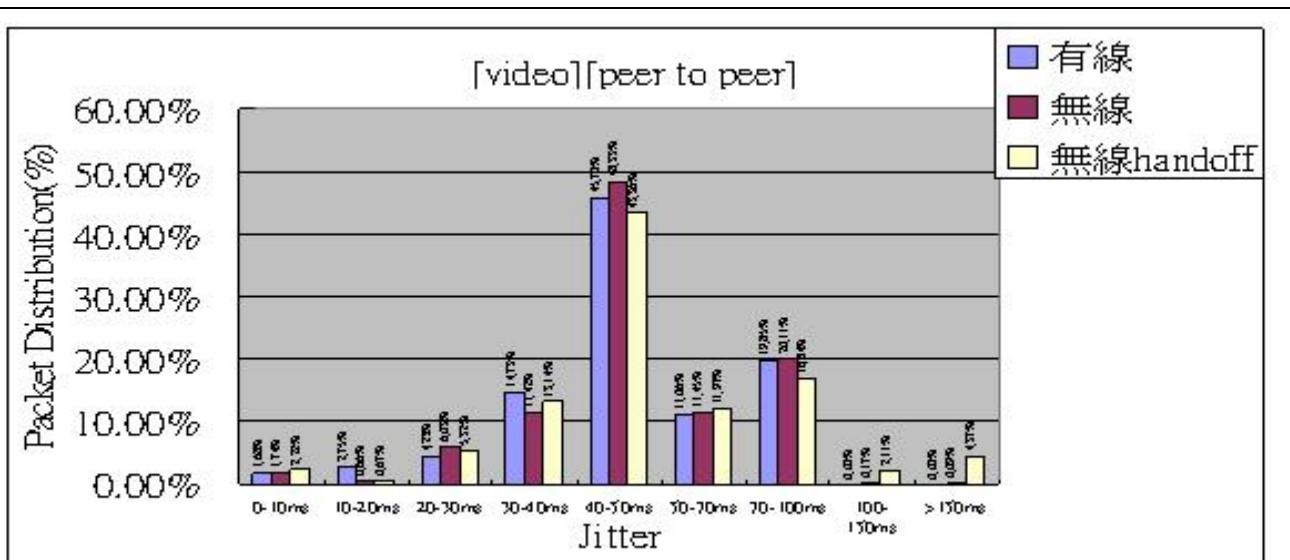
網址為 http://www.paralink.com.tw/product/wildpacket/pdf/epvx_datasheet.pdf

2.1 Infrastructure mode 的測試結果

2.1.1 peer-to-peer mode



圖十二、透過有線與無線 AP 及發生 handoff 在 peer-to-peer mode 下 audio 的 jitter 分布



圖十三、透過有線與無線 AP 及發生 handoff 在 peer-to-peer mode 下 video 的 jitter 分布

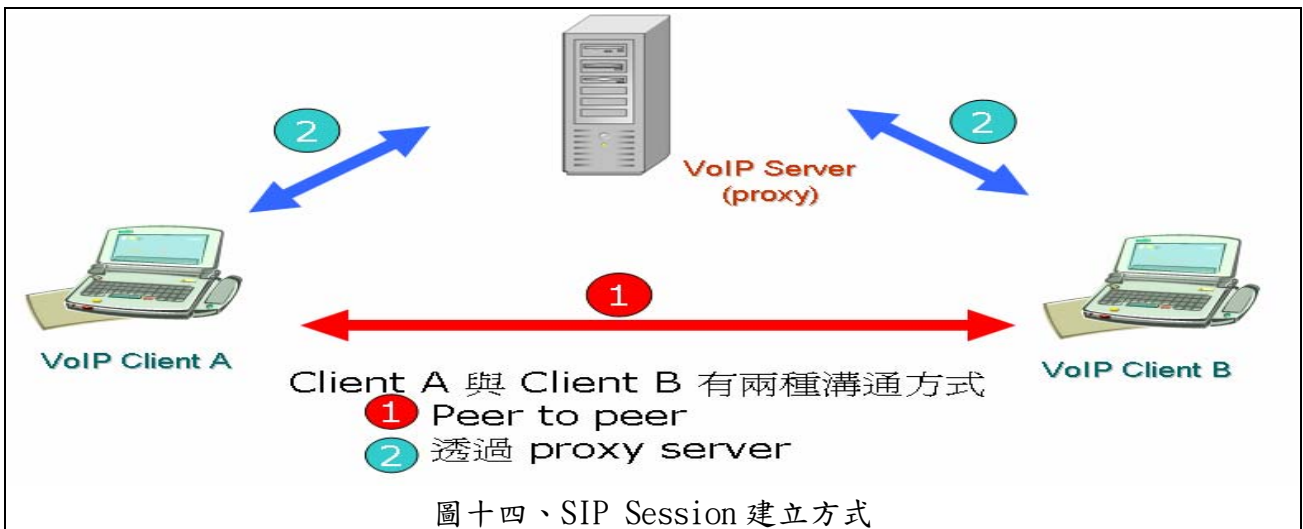
我們使用交通大學資訊工程系 Lab117 與 Lab610 所發展的 SIPv6 Analyzer，主要能分析 SIP 連線時，jitter 的分布情形，分成 video 及 audio 兩項來討論。首先我們在此 SIP 的測試中發現，audio 的 jitter 主要分佈在 10ms 內，然而 video 的 jitter 卻主要分佈在 40ms-50ms，可見 audio 在網路上的傳送比 video 的傳送來的平順，這是由於 audio 有較小的 packet size 所導致的結果。

在透過 AP 的無線網路上所造成的 jitter，相較於單純在有線網路上的 jitter 並不如想像中的嚴重。這主要是由於在無線網路上競爭的 node 個數並不是很多，因此觀察 SIP 透過無線 AP 連上網路的數據，我們發現在 audio 上的 jitter 只在 10ms-40ms 的區間有些微的增加，而 video 上的 jitter 也在 40ms-100ms 的區間增加 1 至 3 個百分點。

接著我們比較有興趣的是在於強制性的 handoff，對於整個 SIP 效能所造成的影響有多大呢？很明顯的在 handoff 造成的 4 至 7 秒無連上 AP 的狀況下，造成不小的影響，尤其是在 video 方面。在 audio 的 jitter 分佈圖上，有些甚至大於 150ms，這對於 audio 而言，是非常的嚴重。在 video 的 jitter 分佈圖上，當發生在兩台 AP 間 handoff 時，在 jitter 較大的區間，比例有明顯的上昇。以上這情形主要的原因在於當 handoff 剛發生時，無線上傳送的 packet，由原本較流暢的傳送，接著就大量的暫存於 buffer 中，當連上另一台 AP 後，原本暫存於 buffer 中的 packet 開始傳送。在這兩種情形時，delay 的變化都較大，因此產生了比較大 jitter 的可能。

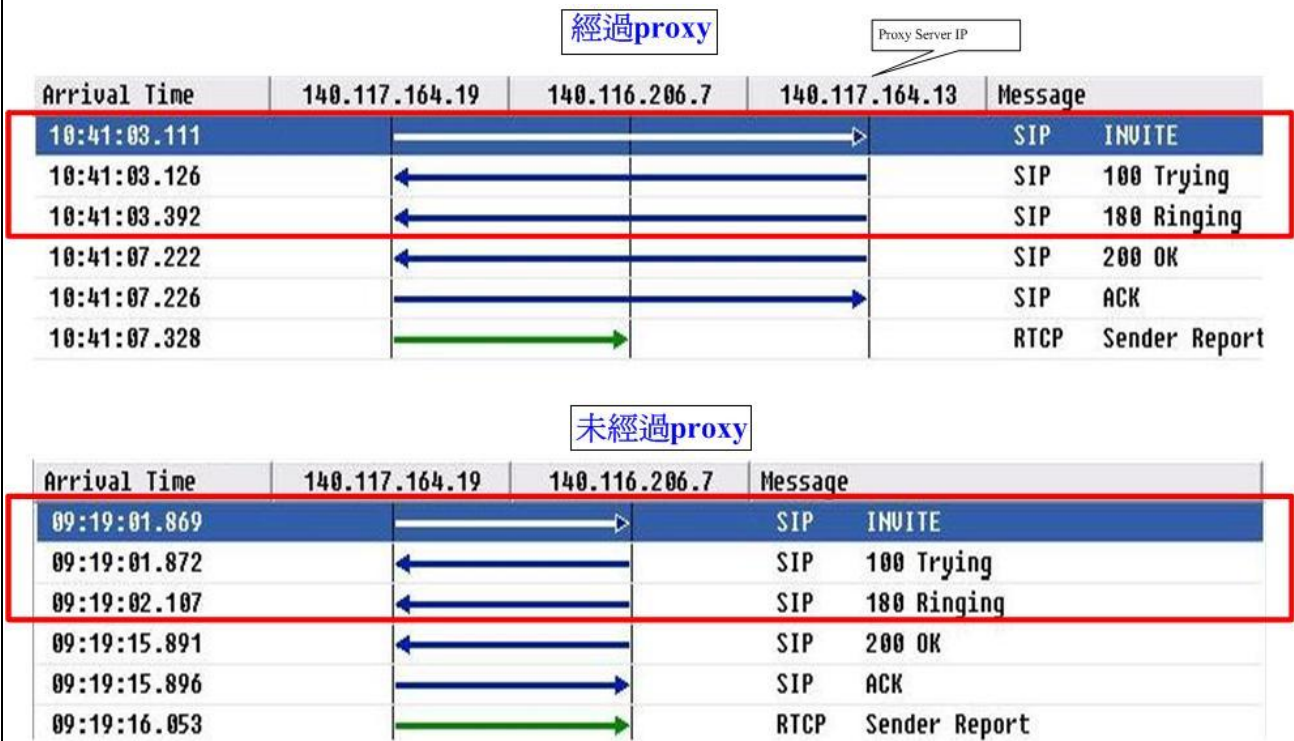
2.1.2 SIP proxy mode

在 SIP VoIP 連線建立中，有<圖十四>中的兩種方式：

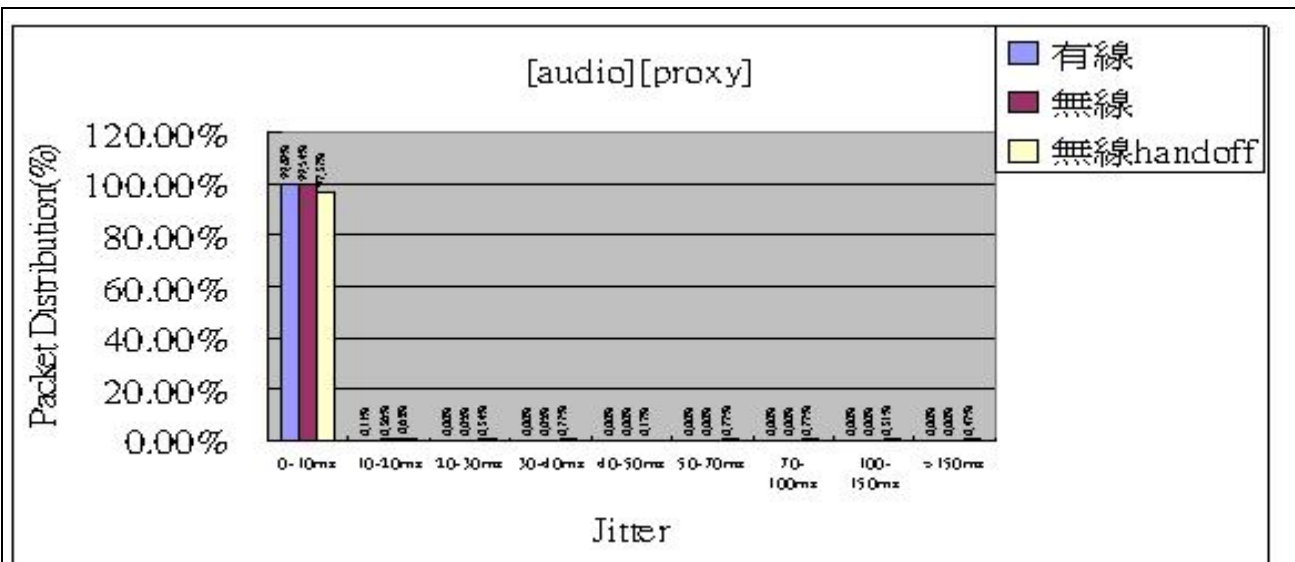


因為 SIPv6 Analyzer 不能顯示 IPv6 網路連線建立流程的時間，但為了量測 peer-to-peer mode 與透過 SIP proxy server 通訊的時間差值，所以使用廣聯科技 EtherPeek VX 軟體量測在 IPv4 網路環境下，兩種不同建立連線方式的時間差。

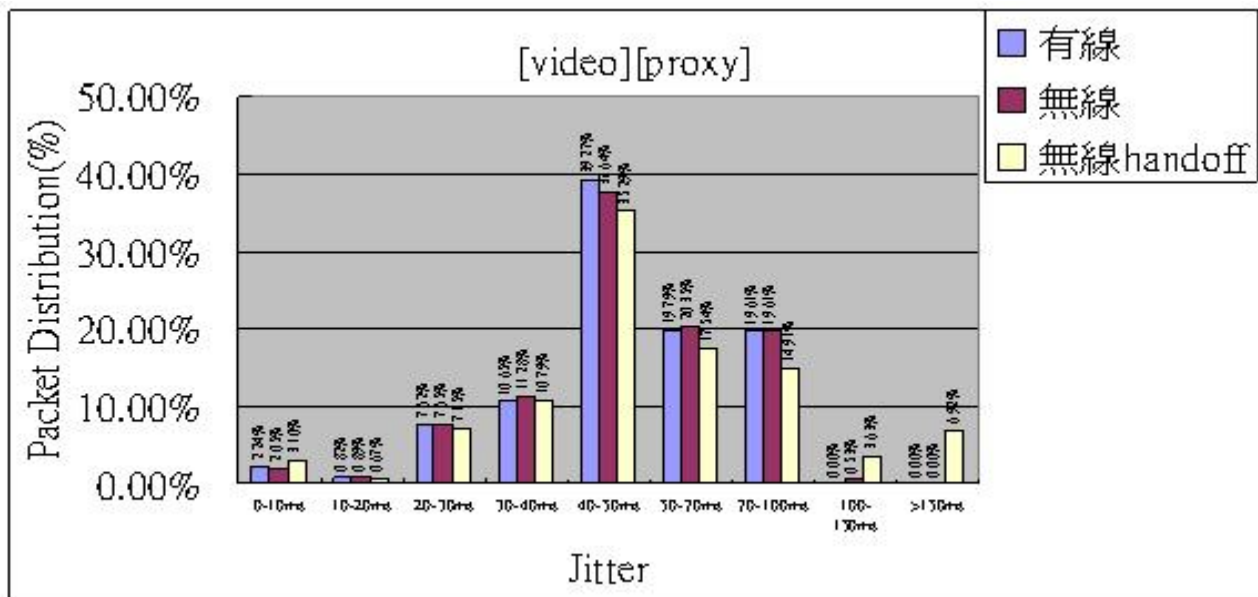
在<圖十五>中可看到有經過 proxy 的圖中多出一個 Proxy Server IP 欄位，計算的時間為紅色框框內 Ringing 的 Arrival Time 減去 Invite 的 Arrival Time，得到的數值即為撥號開始至對方響鈴的時間，經過多次實驗數據統計，在此環境下所量測的結果為透過 SIP proxy server 註冊連線模式比 peer to peer 模式約多出 40ms。



圖十五、透過 SIP proxy server 及 peer-to-peer mode 的連線流程比較



圖十六、透過有線與無線 AP 及發生 handoff 在 SIP proxy server 連線下 audio 的 jitter 分布

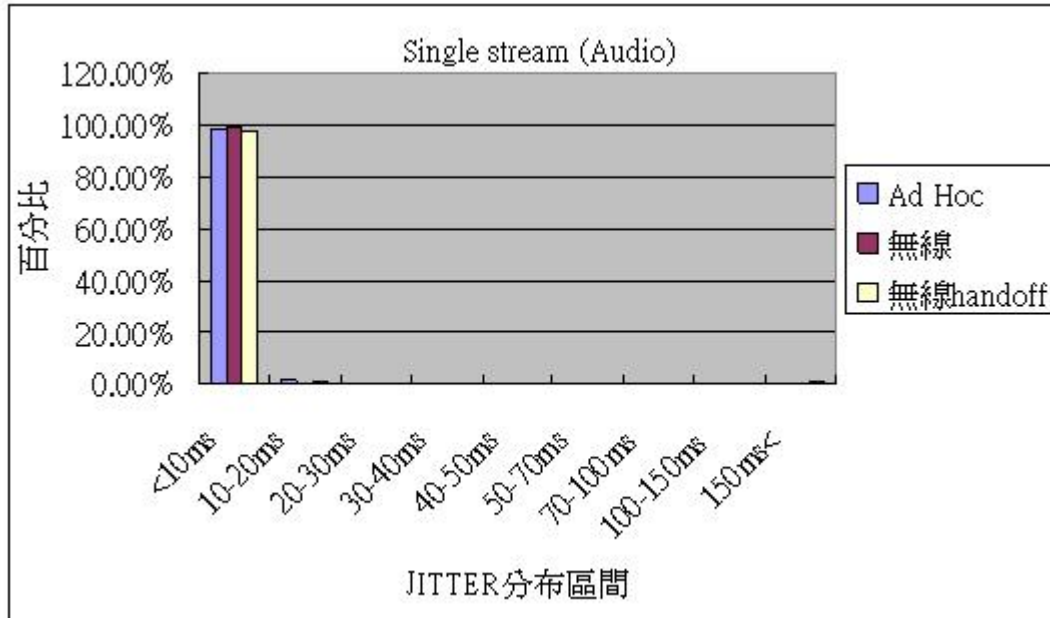


圖十七、透過有線與無線 AP 及發生 handoff 在 SIP proxy server 連線下 video 的 jitter 分布

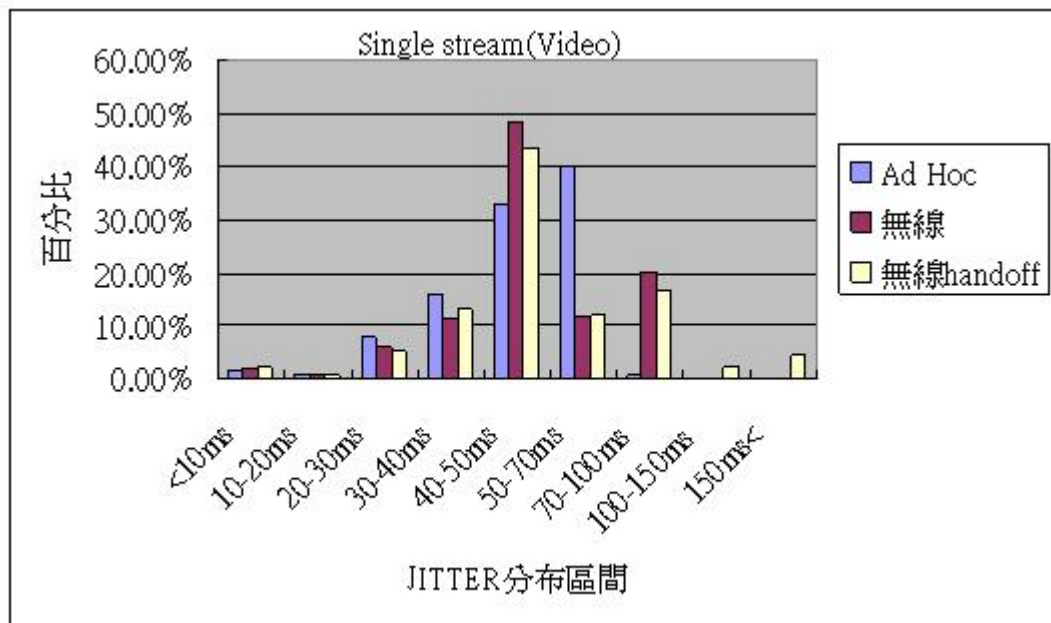
圖十六圖十七為實際在 IPv6 網路透過 proxy 的測試結果，在與 peer-to-peer 的結果比較下，在有 handoff 的情形下，在 jitter 較大的區間，百分比都有增加的趨勢。但在透過 proxy 的情形下，主要的差別還是在於連線的部份，至於傳送 audio 與 video 時應該就如同 peer-to-peer 的方式一樣，而在 handoff 期間，PCA softphone 仍未顯示斷線，因此提出個假設，是否在 handoff 時由於過久時間未收到 packet，因此會有再次詢問 proxy 的可能，而導致 jitter 的增加。

2.2 在無線 Ad Hoc mode 的測試結果

2.2.1 Single stream



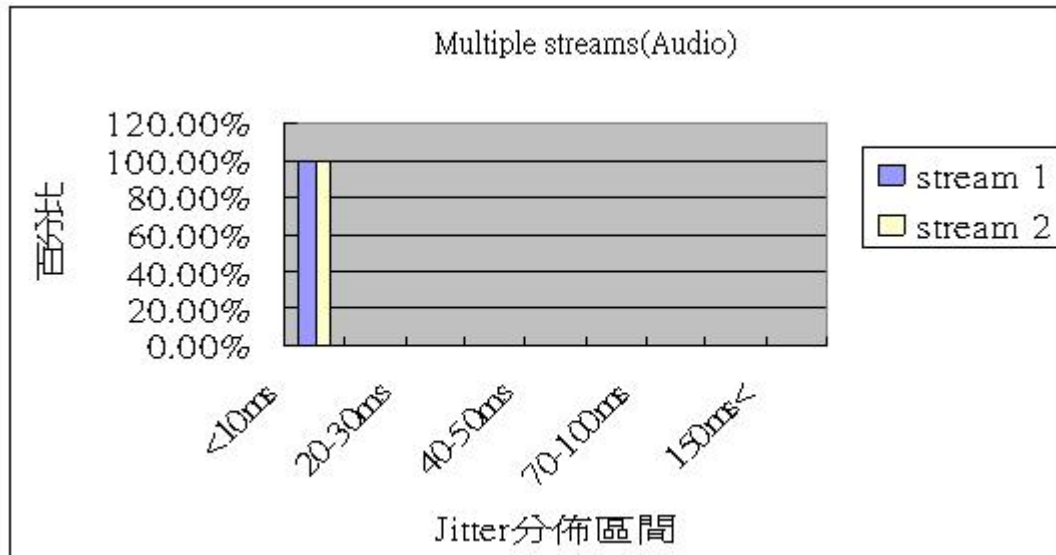
圖十八、透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 audio 的 jitter 分布



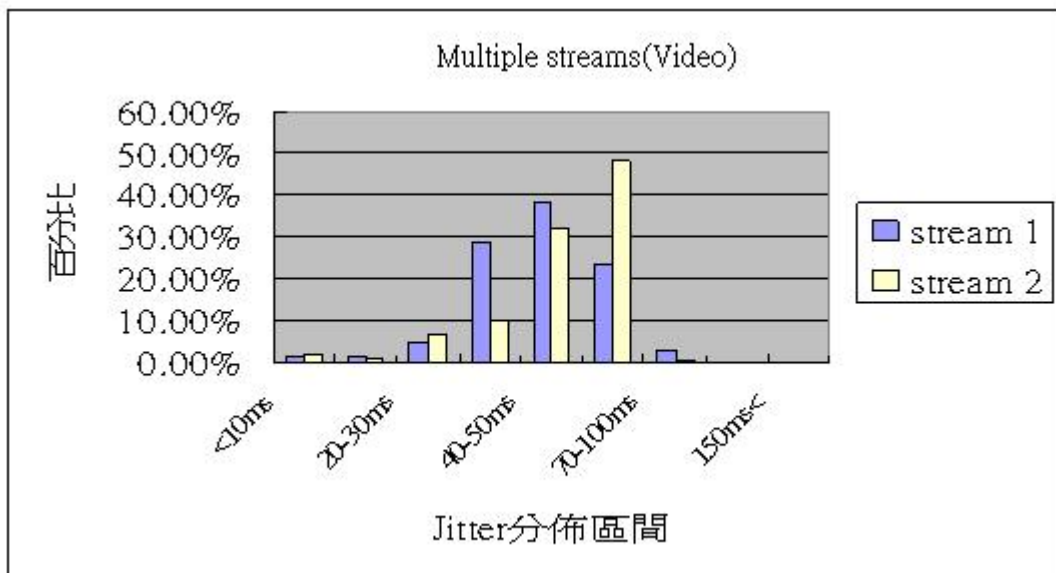
圖十九、透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 video 的 jitter 分布

我們將結果跟使用 AP 來比較，因為在 Ad-Hoc 裡連線的路徑比使用 AP 長，封包的傳輸時間變長，並且當資料是由 Ethernet 進 Ad-Hoc 時，傳輸的速率降低，Queue 比較容易有較多的封包，所以 jitter 也就容易變大，而在 handoff 時會花比較長的時間，因此會有一些封包量出來的 jitter 比 Ad-Hoc 測量出來的大很多。

2.2.2 Multiple streams



圖二十、透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 audio 的 jitter 分布



圖二十一、透過有線與無線 AP 及發生 handoff 在 ad hoc mode 下 video 的 jitter 分布

stream 1 是由跟 NAT server 接在相同 switch 的 User Agent 與 Ad-Hoc 網路裡的 User Agent 所建立的連線，而 stream 2 是成功大學與 Ad-Hoc 裡網路的 User Agent

所建立的連線，兩條 stream 測出來的結果，在 video 上比較看得出差異，stream 2 看的出來比較集中在 40~100ms 之間，這是因為我們每次的實驗都是讓 stream 1 先建立，過了 3~5 秒後，才建立起 stream 2，因此 stream 1 有一小段時間只有一條 stream 在網路上跑，並且 stream 1 的路徑比較短，所以 stream 1 的 jitter 會比 stream 2 來的低。

第三章 未來測試工作與遭遇的問題

3.1 整合多個大學校園的 SIP 測試

未來我們將再嘗試與成功大學連線作測試，並進一步再與交通大學、台灣大學等校連線，測試中山大學的無線網路與 Ad-Hoc 網路與各校之間的 SIP-IPv6-VoIP 透過 TWAREN 骨幹網路是否可以正常運作的。

3.2 設定路由器的 Priority Queues

我們將使用 EtherPeek VX 產生不同 priority 的 traffic，並且讓不同的 traffic 對應到路由器中不同的 queue，觀測當網路流量較大時，high priority 與 low priority traffic 在傳送品質上的差異。

3.3 Handoff 過程並非 smooth handoff

在實作 handoff 過程中，曾嘗試過 smooth handoff，但是因為環境限制且 AP 皆須接上電源插座，必須先找到讓兩台 AP 相互重疊區域，然後讓筆記型電腦從一台 AP 區域移動至重疊區域再移至另一台 AP 區域，觀察是否能自動切換 AP，發現因為電腦接收訊號並不穩定，所以經常在移出重疊區域時就發生斷線。未來希望能找到穩定在不同 AP 間 handoff 的方法，來進行 smooth handoff 的測試。

3.4 解決 multiple streams 的數量有一定的限制

由於目前撰寫出來的僅是簡易的 NAT，因此未來可能參考 IPv4 NAT 的作法，來補強目前 NAT 功能的不完整，希望能做到由使用者以指令模式設定 NAT，而不是目前設定 NAT 時，必須重新編譯 kernel。

3.5 Ad Hoc 部份目前僅能使用 static route

由於使用的是桌上型電腦，無法移動，因此我們採用設定 static route 的方式來做到 multihop，未來希望能使用多部筆記型電腦，並且嘗試利用 Dynamic routing，做出較為實際的 Ad-Hoc 網路，另外需考慮到有線與無線網路上不同的 routing protocol 能否相互支援配合，當解決 routing 問題後，可以實作測試在無線網路路徑切換時對 SIP-VoIP 的影響。

3.6 實現多台 SIP Proxy Server 的互連

目前我們只使用一台 SIP Proxy Server 作為註冊及建立連線的主機，但是實際網路上會有多台 SIP Proxy Servers，當架設多台 SIP Proxy Servers 就必須交換 User Agent 的註冊資料，且須維持帳號資料的單一性。

未來我們將嘗試解決不同學校 SIP Proxy 之間互相溝通的問題，並使用 SIP Proxy 來建立連線。