# 第一章

# 緒論

## 1.1 簡介

IEEE 標準委員會在 2004 年公佈 IEEE 802.16-2004(原名 IEEE 802.16d)的正式標準，IEEE 802.16-2004 是新一代都會型無線區域網路(Wireless Metropolitan Area Networks)的標準，做爲都會區無線寬頻接續系統(wireless broadband access system)的規格。2005 年，IEEE 進一步增加行動網路接續標準─802.16-2005(原名 802.16e)，提供行動通訊所需的訊號交換標準。爲使業界開發能有統一的標準與互通性，WiMAX Forum 由 IEEE 802.16 的多個標準中挑選出單一的實作版本，作爲生產設備之共同標準介面，因此也以 WiMAX (Worldwild Interoperability for Microwave Access)這個名字作爲此標準的另一個名稱。

在 IEEE 802.16-2004 標準中，對於固定式寬頻網路採用兩種不同的運作模式，一爲點對多點模式(Point-to-Multi-Point mode, PMP mode)，一爲網狀網路模式(Mesh mode)。點對多點模式是以基地台(base station, BS)爲中心，節點與 BS 間以無線電直接連線，並由 BS 進行允入控制(admission control)、媒體存取(media access)與資源分配(resource allocation)之工作。而網狀網路模式則是由多個網狀節點(mesh node)交互相連而成，各個網狀節點彼此溝通協調，並轉送(relay)資料到達 BS 或閘道器(gateway)，將資料傳入核心網路(core network)。兩種模式的 MAC 框架格式不同，故彼此之間並不互通；然而，由於地形與環境的影響，有

國立政治大學 National Chengchi University

時手持設備(handset)無法直接連上基地台，此時便可以進行軟體的轉換，使相同的機器設備轉換爲網狀網路模式，透過多個節點(Subscriber Station, SS)轉送的方式進行資料傳輸，以提高信號覆蓋率。另一方面，網狀網路模式的特性是可以自動進行資料轉送，因此在部署上容易許多，不僅不需要對每個 SS 鋪設固網線路，手持設備亦只需連上附近的其他 SS 即可，而無需直接與 BS 連線，可以有效節省手持設備的電力消耗。

網狀網路模式又可依據其資源分配的方式分爲集中排程(centralized scheduling)與分散排程(distributed scheduling)兩種，集中排程是 SS 將頻寬需求全部送交給 BS，BS 依照目前的可用頻寬及各 SS 提出的需求，進行集中式的資源分配。而分散排程環境則採取分散處理的方式，頻寬的分配是由通訊兩端的兩個 SS 自行協調，決定所要進行傳輸的 minislot。

然而在分散排程環境中，由於資料轉送必須透過多個 SS，而每兩個 SS 間又必須各自協調 minislot，因此每次 SS 在進行封包轉送時「協調 minislot 所需時間」都會增加封包的延遲，如何減少這些延遲，提高節點的封包轉送效率，將成爲改善網狀網路服務品質的重要課題。

本研究正是針對網狀網路中的分散排班(distributed scheduling)環境，探究該環境中造成傳輸延遲的產生原因，並建立動態調整的媒體存取機制，加快延遲敏感資料流(delay-sensitive traffic)的處理速度，以達到服務品質保證(quality of service)、提高頻寬使用的效益、降低網狀網路中的資料流延遲，使 delay-sensitive traffic 仍能在擁塞環境中維持一定的服務品質。

國立政治大學 National Chengchi University

## 1.2 研究動機與目的

本研究主要討論在網狀網路的分散排程環境中，節點在轉送封包的過程中會遭遇到的延遲現象，這些延遲會增加封包的傳送時間，容易造成封包比預定到達時間爲晚。對 delay-sensitive traffic 來說，封包若未能及時送達，此封包即屬於無效，故此種資料流對於網路環境有著較嚴格的要求。然而，在 IEEE 802.16-2004 的標準書中，僅規劃了點對多點模式的 QoS(quality of service)規格，網狀網路部分並未將 QoS 納入規範。另一方面，無線網路無法偵測封包碰撞(collision)，因此必須進行媒體存取控制；而網狀網路環境在進行通訊時，透過不斷競爭傳輸機會(Transmission opportunity, TO)，並與目的端進行三方交握(3-way handshake)，以取得媒體存取的權力。故參與競爭之節點數量增加時，TO 的競爭愈來愈激烈，三方交握將需要更久的時間才能成功，因此頻寬將趨於不穩定，容易造成傳輸延遲(transmission delay)以及抖動率(jitter)提高。因此，我們希望可以規劃出另一種具有動態調整功能的競爭機制，可以舒緩因節點增加導致的 TO 激烈競爭，並在三方交握階段加入「重要性因子(factor of importance)」的考量，使網狀網路在處於高負載情況下，仍能夠盡量使重要的資料流運作無礙。

因此，本篇論文的目的即是針對網狀網路中，對於媒體存取的競爭行爲進行調節，使轉送 delay-sensitive traffic 的節點具有較快速的訊息交換速度，並針對 request 的重要性進行評估，網狀網路節點在經過訊息交換後，可以決定自身鄰接區域(neighborhood)[1]的最低重要性門檻(threshold of importance)，以此過濾重要性不高的 request，使網狀網路即使在擁塞情況下，仍能將頻寬穩定安排給重要性較高的資料流，降低資料因等待 minislot 而產生的延遲。

[1] Neighborhood 的定義範圍是爲節點自身、及距離節點 one hop 的節點所共同構成的區域

## 1.3 IEEE 802.16 的分散排程

本節將介紹 IEEE 802.16 分散排程如何決定 MSH-DSCH 訊息的傳送時間，以及進行 3-way handshake 頻寬要求的步驟。在介紹之前，我們先介紹一些 WiMAX 設計上的一些新概念，以及與 IEEE 在 1997 年推出的第一套無線通訊標準 IEEE 802.11(Wireless Fidelity, WiFi)的差異。

WiMAX mesh mode 的設計概念是「SS 先競爭發送 MSH-DSCH 的 TO，在 TO 時間內進行 minislot 的預約」，亦即 WiMAX 將「傳輸資料」分成兩個步驟來分開處理，包括「建立控制信號的發送順序」以及「交換控制信號以預約資料傳輸時槽」。這樣的設計與過去發表的 WiFi 標準有些不同。首先，WiFi 採取的是隨機允入(random access)的方式，節點會先 listen 一段時間，若在時間內沒有其他節點進行傳輸，則節點即可進行媒體存取。然而，WiMAX 的設計是採時間排程，SS 彼此之間會進行時序同步，因此每個 SS 的 TO 都與鄰點共用，當 SS 有資料必須傳輸時，就必須透過競爭機制與鄰點競爭 TO 的使用權，勝出的 SS 即可在 TO 時間內發送 MSH-DSCH 信號，透過 MSH-DSCH 的信號進行三方交握，確認傳輸資料的 minislot 時間。

因此，WiMAX 的 MAC frame 被分爲 control subframe 與 data subframe 兩個部分，前端 control subframe 的長度可以變動，以系統中的 MSH-CTRL-LEN 變數定義之，control subframe 部分被劃分爲 MSH-CTRL-LEN 減一個 TO，提供給同一個 neighborhood 範圍內的 SS 競爭。而 data subframe 分爲 256 個 minislot，按照三方交握的結果分配。WiMAX MAC 的 frame 架構如 Figure 1 所示。

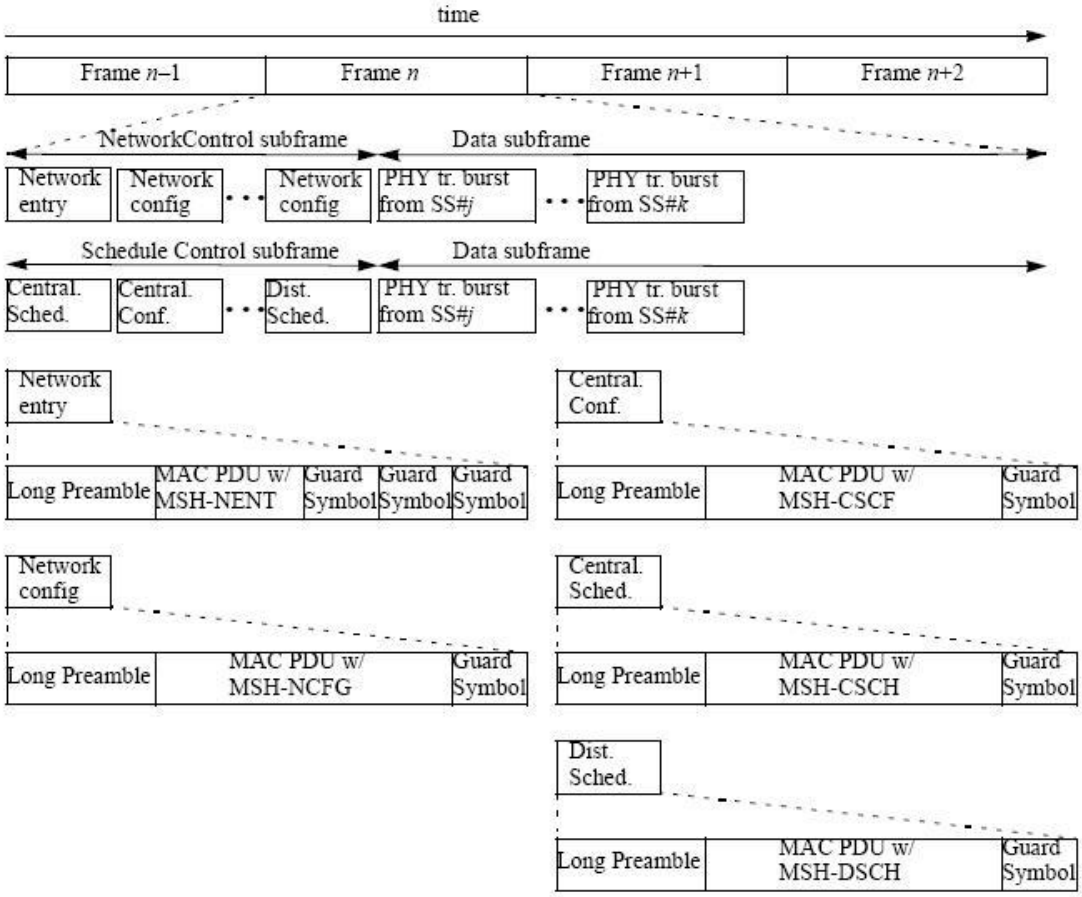


Figure 1: Mesh frame structure[2]

National Chengchi University

### 1.3.1 Competition of Transmission Opportunity

IEEE 802.16 標準定義了 TO 競爭的方式，競爭 TO 相當於在爭取媒體存取的權利，WiMAX 的每個 time frame 的前段都是 control subframe，並分爲 MSH-CTRL-LEN-1 個 TO，因此每個 time frame 就只有 MSH-CTRL-LEN 減一次的媒體存取機會，SS 必須透過一個虛擬的競爭程式來計算競爭的結果。我們將在以下詳述 SS 如何進行 TO 的競爭。

每個 SS 在加入該區網路後，會持續維護 holdoff exponent 與 mx 這兩個值；

[2] IEEE std 802.16-2004, pp.459

holdoff exponent 將影響 holdoff time 的長度，而 mx 則決定了 nxmt 區間(next transmission interval)的位置，這兩個值被用來定義 SS 的 nxmt 區間以及 holdoff time。Nxmt 區間表示 SS 將對此區間內的 TO 進行競爭，holdoff time 則表示 SS 在這段期間內不會參與競爭，而 Esxmt(Earliest sequence transmission time)則表示在 Nxmt 之後，節點最早重新參與競爭的區間，如 Figure 2 所示。

$$\textbf{Xmt Holdoff Time} = 2^{(\text{Xmt Holdoff Exponent}+4)} \tag{1}$$

$$2^{\text{Xmt Holdoff Exponent}} \cdot \text{Next Xmt Mx} < \text{Next Xmt Time} \le 2^{\text{Xmt Holdoff Exponent}} \cdot (\text{Next Xmt Mx}+1) \tag{2}$$ [3]

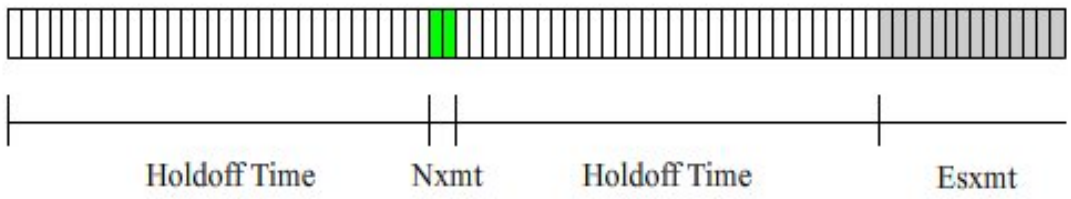


Figure 2: Holdoff time, Nxmt and Esxmt

Nxmt 區間表示節點會在這個區間內選一個 TO 傳送 MSH-DSCH，然而在一個 two-hop neighborhood 的範圍內，不能有兩個 SS 在同一個 TO 發送 MSH-DSCH，否則會造成信號干擾而不能解析。因此每個 SS 必須針對 nxmt 區間內，所有可能發送信號的鄰點進行虛擬競爭，決定 TO 最後應該屬於哪個 SS。由於 SS 在廣播 MSH-DSCH 時，會包含 mx 與 holdoff exponent 的數值，因此 SS 的鄰點都可以得知 SS 的這兩個值。反之，SS 亦可接收到所有鄰點的 mx 與 holdoff exponent，故每個 SS 都能夠推算其鄰點的 nxmt 區間。因此，SS 會從 nxmt 的第一個 TO 開始計算，觀察這個 TO 是否也同時屬於其他鄰點的 nxmt 或是 esxmt

[3] (1) (2) 二式來自 IEEE 802.16 標準書，pp.83

範圍內，藉以得知該 TO 的競爭者有哪些。然後 SS 使用 Mesh election algorithm 計算自己是否能贏得此 TO。若結果爲肯定的，則可用該 TO 發送 MSH-DSCH；若結果爲否定的，則用同樣的方法競爭下一個 TO，直到能得到一個 TO。若在 nxmt 區間結束時，還是沒有得到 TO，則必須停止競爭，進入 holdoff 區間，等待下一次的 nxmt 區間才能重新參與競爭。Figure 3 是一個四節點競爭 TO 的範例，說明如下：

Figure 3 中的綠色區間爲每個 SS 的 Nxmt，此時，假設 SS_1 試圖要取得第 34 個 TO，如圖中紅色線標示，則 SS_1 首先將 Temp Xmt Time 設定爲 34，並檢查其他鄰點是否有可能參與競爭。圖中可以看到 SS_2 的第 34 個 TO 落在綠色區域，表示 SS_2 有可能參與競爭；SS_3 落在白色區域，表示 SS_3 此時處於 holdoff time，不參與競爭；SS_4 落在灰色區域，表示處於 Esxmt 範圍內，有可能參與競爭。因此，SS_1 若想取得第 34 個 TO，其競爭者爲 SS_2 及 SS_4，這兩個節點被稱爲 eligible neighbor(s)。

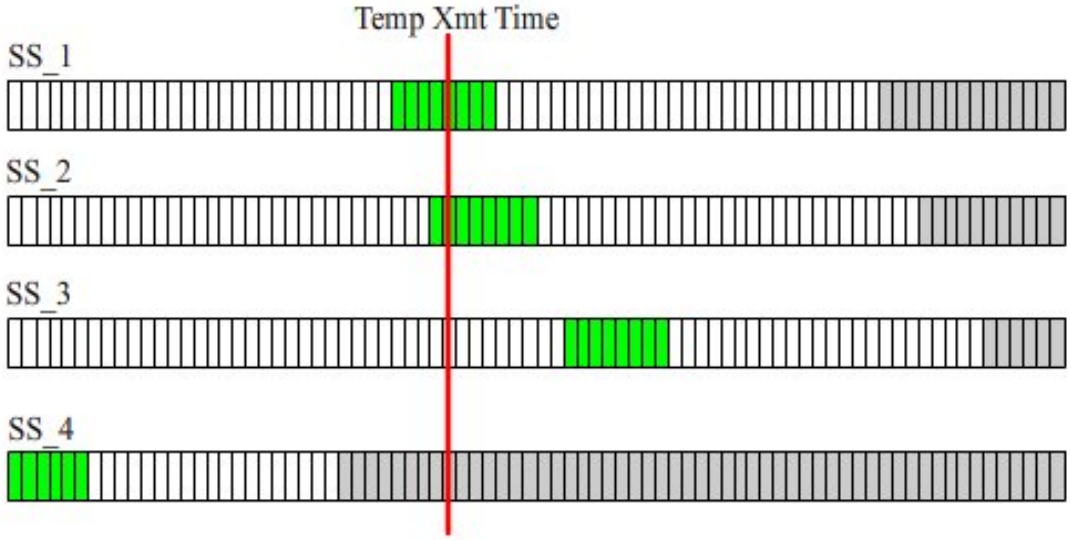


Figure 3: Deciding eligible neighbor

決定了 eligible neighbor(s)後，節點需透過 Figure 4 的流程來決定是否能得到 TO，若成功，則可得到該 TO；若失敗，則競爭下一個 TO，並重新進行一次 eligible neighbor 的選拔。以 Figure 3 的例子來說，若 SS_1 勝過 SS_2 及 SS_4，則得到第 34 個 TO；若失敗，則競爭第 35 個 TO，且需重新計算第 35 個 TO 的 elible neighbor 爲何。

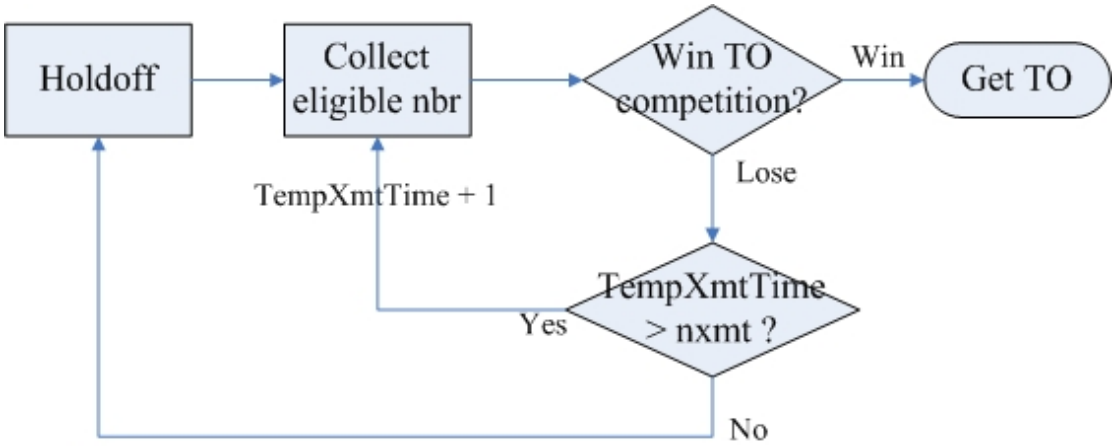


Figure 4: TO contention flow

Figure 4 中的「Win TO competitioin? 」的決策方塊是透過一個叫 Mesh election algorithm 的函數來計算，它是 IEEE 802.16 標準書中定義的一個雜湊涵數，其特性是各節點只需輸入自己的 node_id、其他參與競爭鄰點的 node_id、以及要競爭的 TO_id，就可計算出自己是否得勝，並且勝利者必恰爲一個。其原始碼如 Figure 5 所示：

```
boolean MeshElection (uint32 XmtTime,uint16 MyNodeID,uint16 NodeIDList [ ] )) {
    uint32 nbr_smear_val,smear_val1,smear_val2;
    smear_val1 =inline_smear(MyNodeID ^ XmtTime ));
    smear_val2 =inline_smear(MyNodeID +XmtTime );
    For each Node ID nbrsNodeID in NodeIDList Do {
        nbr_smear_val =inline_smear(nbrsNodeID ^ XmtTime ));
        if(nbr_smear_val >smear_val1 ) {
            return FALSE;//This node loses.
        }
        else if(nbr_smear_val ==smear_val1 ) {
            //1st tie-breaker.
            nbr_smear_val =inline_smear(nbrsNodeID +XmtTime );
            if(nbr_smear_val >smear_val2 ) {
                return FALSE;//This node loses.
            }
            else if(nbr_smear_val ==smear_val2 ) {
                //If we still collide at this point Break the tie based on MacAdr
                if ((XmtTime is even &&(nbrsNodeID >MyNodeID))||
                    (XmtTime is odd &&(nbrsNodeID <MyNodeID ))) {
                    return FALSE;//This node looses.
                }
            }
        }
        //This node won over this competing node
    }//End for all competing nodes
//This node is winner,it won over all competing nodes.
return TRUE;
}
```

generate 2 hash value of MyNodeID

generate 2 hash value of NBRNodeID

Competing with every Neighbor

If draw twice, Judge by XmtTime & NodeID

Figure 5: Mesh election algorithm

每個節點都會先以 inline_smear()這個雜湊函數產生兩個固定的雜湊值 smear_val1 及 smear_val2，並為自己所有的鄰點產生一個 nbr_smear_val 的雜湊值。首先比較 smear_val1 及 nbr_smear_val 的大小，若 nbr_smear_val 較大，則表示自己落敗；若 smear_val1 較大，則自己獲勝；若是平手，則進行第二輪的比較。第二輪需再產生一個新的 nbr_smear_val，比較自己的 smear_val2 及鄰點新的 nbr_smear_val。同樣的，若 nbr_smear_val 較大，則表示自己落敗；若 smear_val2 較大，則自己獲勝。若第二次仍是平手，則比較 nbrNodeID 與 MyNodeID 的大小，以 XmtTime 的奇偶決定取「大者為勝」或「小者為勝」。

### 1.3.2 Mesh mode signal

決定了 TO 的歸屬後，SS 即可利用 TO 的時間發送控制訊息(control

message)，在 Mesh mode 中比較重要的控制訊息是 MSH-NENT (mesh network entry)、MSH-NCFG (mesh network configuration)及 MSH-DSCH 三種。而 WiMAX 的控制訊息還包含許多的資訊元素(information element, IE)，這些 IE 的功能各自獨立，如 Entry Open IE、Schedule IE、Request IE 等，因此 SS 發送控制信號時，能夠一次送出多個 IE，同時間之內即可與多個 neighbors 溝通。

MSH-NENT 信號是 Mesh network entry 的縮寫，主要用途是讓一個新的 SS 要加入區域網路時，能通知區域網路內的 neighbor 新增此一 SS 的信號。MSH-NENT 信號又可包含 Network Entry Open IE、Network Entry Reject IE、Neighbor Link Establishment IE、Request IE 等。

MSH-NCFG 信號是 Mesh network configuration 的縮寫，主要包含該區網路的組態設定資訊，如無線電發送功率、天線型態、BS 資訊、同步時戳等，在分散排程機制中，是由該網路中的 SS 輪值發送。其中還可以包含 Nbr-Physical IE、Nbr-Logical IE、Network Descriptor IE 及 Channel IE，主要用途是發佈網路基本組態設定，使網路得以正常運作。此外， MSH-CTRL-LEN 及 MSH-DSCH-NUM 這兩個重要參數被放置在 Network Descriptor IE 中，這兩個參數會影響 Control subframe 的長度及 MSH-CSCH 與 MSH-DSCH 的數量。

MSH-DSCH 信號是 Mesh Distributed Scheduling 的縮寫，主要用途是使兩個 SS 透過此信號進行 minislot 的預約。MSH-DSCH 會包含 Request IE、Availability IE、Grant IE 三種資訊元素，這三種元素被用來進行 3-way handshake，使兩個 SS 得以預約 minislot，同時周圍的 neighbor 也會得知該 minislot 已被使用，而不會同時進行傳輸。

Request IE 中包含的資訊是 Demand Level 及 Demand Persistence。Demand

國立政治大學 National Chengchi University

Level 表示在每個 frame 裡要使用多少個 minislot，Demand Persistence 則表示這樣的需求必須持續多少個 frame。而 Availability IE 是 SS 的 available minislot 配置圖，包括了 available frame 的起始位置(Start Frame number)與長度(Persistence)、Minislot 起始位置(minislot start)與長度(minislot range)、以及邏輯頻道(channel)等資訊。Grant IE 的結構類似 Availability IE，也有 Start Frame number、Persistence、minislot start、minislot range 四個參數，只是 Grant IE 表示的是傳輸預定使用的 minislot。此外，Grant IE 另有一個 Direction 欄位，是用來分別 Grant IE 所表示的為「Grant」或「Confirm」，這在三方交握階段的意義不同，將在下一小節詳述。

### 1.3.3 3-way handshake

MSH-DSCH 的發送時間固定後，節點會透過 MSH-DSCH 來傳遞 schedule 以及進行 handshake。三方交握的目的是為使通訊的兩端彼此知道資料會被放在 data subframe 的哪一個 minislot，以便進行傳送與接收；三方交握是透過 Request IE、Availability IE 及 Grant IE 三個資訊元素完成，其流程如下：

(1) **Request** : $SS_a$ 發送 Request IE 及 Availability IE 給 $SS_b$。

(2) **Grant** : $SS_b$ 從 Request IE 中可得知 $SS_a$ 所需要的 Demand Level 及 Demand Persistence，並從 $SS_a$ 的 Availability IE 中得知 $SS_a$ 目前有哪些 frame 及 slot 屬於 available 的狀態，將 $SS_a$ 的 Availability IE 與自己的 Available minislot 取交集，從中挑選符合 Request IE 的 frame 及 minislot，再以 Grant IE 回覆給 $SS_a$。

(3) **Confirm** : $SS_a$ 收到 $SS_b$ 送來的 Grant_IE 後，再重新確認一次 Available minislot 是否都屬於 available。若是，則將 $SS_b$ 送來的 Grant IE 重覆發送

國立政治大學 National Chengchi University

一次，作爲向 $SS_b$ 確認的動作；同時，在 $SS_a$、$SS_b$ 周圍的其他 neighbor 也會按照這兩次 Grant_IE 的資訊，將 $SS_a$、$SS_b$ 所使用的 minislot 標記爲 non-available，以避免同時使用該 minislot。

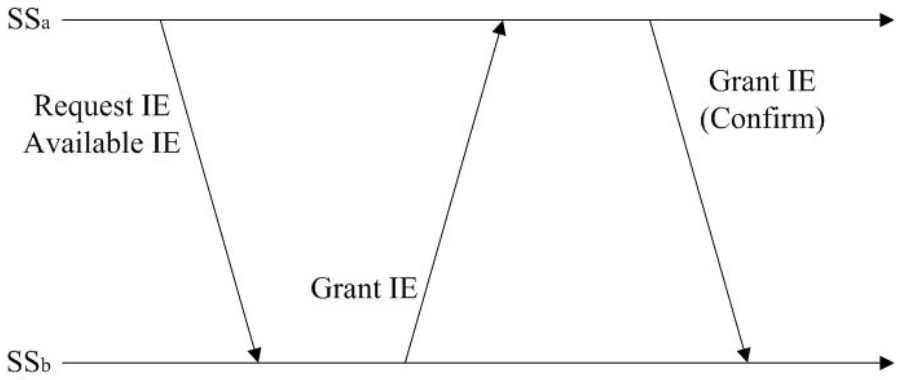


Figure 6:Transmit information element in 3-way handshake

經過了三方交握之後，傳送端的 SS 可以與接收端的 SS 確認傳輸用的 minislot 爲何，因此 Data subframe 的分配方式取決於三方交握的結果，而 TO 的競爭則決定了 Control subframe 中 SS 發送控制訊號的權力。在資料流傳輸過程中，兩個 SS 必須時常進行三方交握，以維持 minislot 的用量。整個 WiMAX 在頻寬分配的流程如 Figure 7 所示：

國立政治大學 National Chengchi University

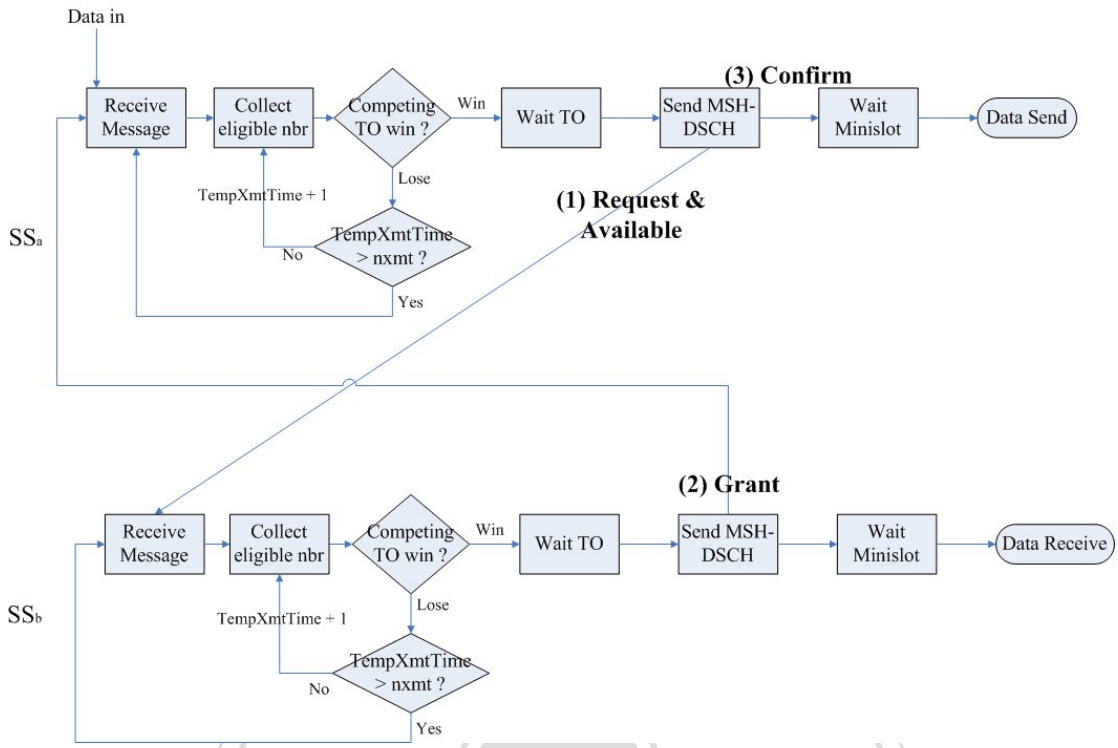


Figure 7: The process from bandwidth request to minislot allocation

從 Figure 7 可知，節點要進行傳輸前，需最少先進行一次 3-way handshake，而每次 3-way handshake 的過程共包含三次「Wait TO」以及一次「Wait Minislot」；本研究就是希望降低這兩個階段的等待時間，使整體網路效能有更大的提升。

國立政治大學 National Chengchi University