

# 國立交通大學

## 資訊科學與工程研究所

### 碩士論文

自組織雲端網路中

雲端的可擴充性與資源分配之研究

The Study of Cloud Scalability and Resource Allocation in

Cloud-Based Self-Organizing Networks

1896

研究生：黃世鑫

指導教授：趙禧綠 教授

中華民國 102 年 11 月

自組織雲端網路中雲端的可擴充性與資源分配之研究

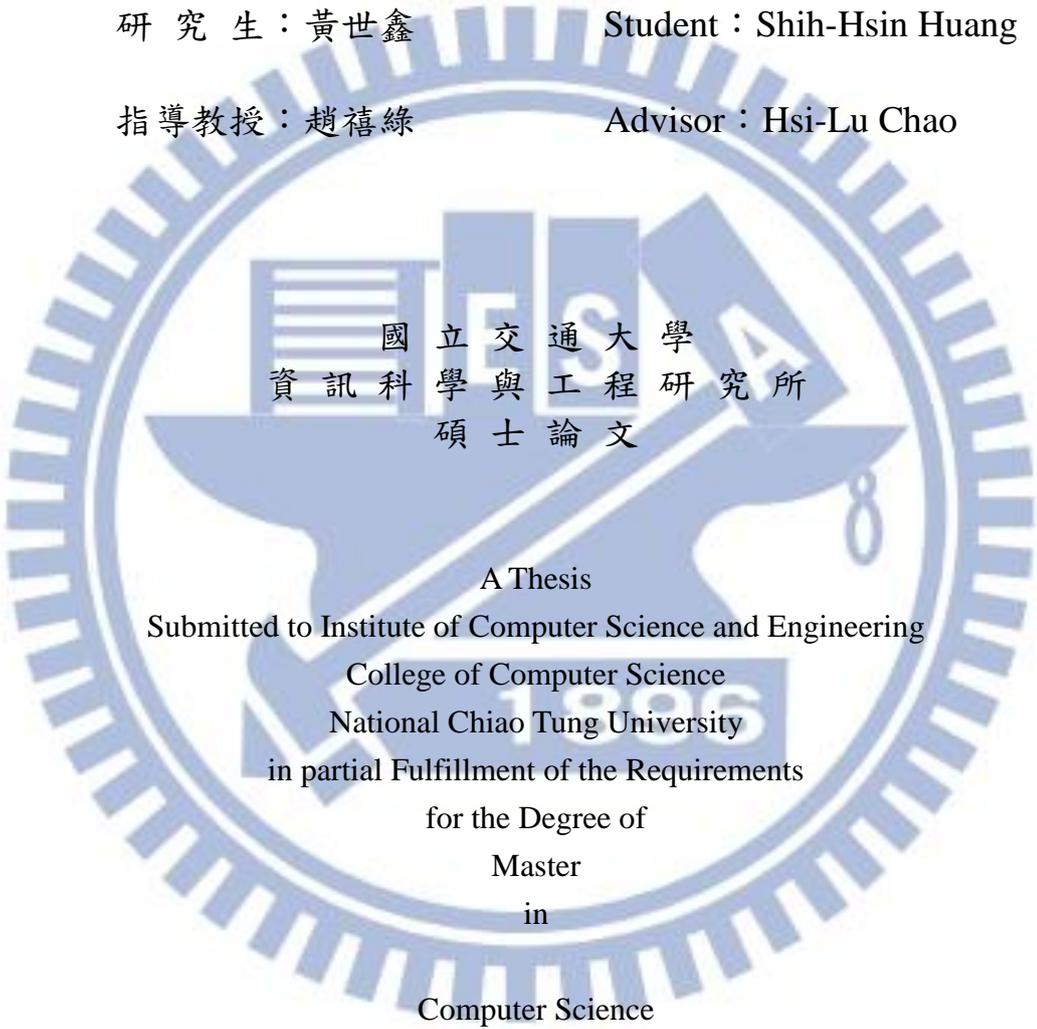
The Study of Cloud Scalability and Resource Allocation in Cloud-Based  
Self-Organizing Networks

研究生：黃世鑫

Student : Shih-Hsin Huang

指導教授：趙禧綠

Advisor : Hsi-Lu Chao



國立交通大學  
資訊科學與工程研究所  
碩士論文

A Thesis  
Submitted to Institute of Computer Science and Engineering  
College of Computer Science  
National Chiao Tung University  
in partial Fulfillment of the Requirements  
for the Degree of  
Master  
in  
Computer Science

June 2012

Hsinchu, Taiwan, Republic of China

中華民國 102 年 11 月

## 摘要

近年來，隨者網路使用者不斷的增加以及使用者的上網習慣也在逐漸改變，越來越多使用者利用網路來觀看影音服務，導致頻譜資源逐漸供不應求。而無線感知技術可以用來改善頻譜資源不足的問題。

但是如果想要服務數以萬計的使用者，那麼勢必需要一控管中心，在我們的架構中我們以雲端的方式去集中管理我們的無線感知網路基地台以及使用者。在這篇論文中我們著重於兩個部份，第一是對於雲端系統架構的討論，要如何才能使我們的系統更好，像是如何針對雲端中各個元件去做擴充，以及對於雲端中的資料傳遞。第二，考慮一大環境，我們為了支援即時的通訊服務，我們提出分群演算法將一個大型環境切割成數個子群，而每個子群都要滿足所限制的工作負載量，並且將每個子群分給一台虛擬機器負責分配頻譜資源，如此一來就可以平行的分配各個子群的頻譜資源。而當我們分出子群之後，對於子群與子群間邊界上的無線感知網路基地台會有干擾情況發生，而不同子群為不同虛擬機器所管理無法立即得知彼此對於頻譜的分配情況，所以我們提出邊界頻譜資源分配演算法對邊界上的無線網路基地台頻譜先做一個初步的頻譜資源分配。

對於分群與邊界頻譜資源分配問題，我們分別提出的最佳化與啟發式演算法並做時間複雜度與模擬的比較，並且會針對一大環境做模擬來驗證我們的雲端系統的可行性。

# Abstract

In recent years, Internet users continue to increase as well as the user's surfing habits are gradually changing,, more and more users use the Internet to watch video services, cause the shortage of spectrum resources. The cognitive radio technology can be used to improve the spectral problem of insufficient resources.

But if we want to serve tens of thousands of users, it is need a control center, in our architecture we use cloud to manage our cognitive radio network base stations and users. In this paper we focus on two parts , the first is for the discussion of cloud systems architecture , how to make our system better , like how to scale-out each component of the cloud , as well as in the cloud data transfer . Second, consider a large-scale environment, we have to support real-time communication services, we propose a clustering algorithm will be cut several cluster in a large-scale environments, and each cluster must meet the work load is limited and will be allocated to each cluster a virtual machine is responsible for the allocation of spectrum resources, as a result, we can parallel the distribution of the spectrum of each cluster. But when we separate the cluster, the boundary base station between cluster and cluster will cause interference, and different cluster management by different virtual machines can not immediately know each other for spectrum allocation situation, so we propose a algorithm for wireless networks for boundary base stations do a preliminary allocation of spectrum resources.

For clustering and boundary base stations spectrum resource allocation problem, we were presented optimization and heuristic algorithms and do the time complexity compared with the simulation, and will make for a large-scale emulations to verify the feasibility of our cloud systems.

## 致謝

從進入研究所到現在已經經過了兩年多，在這兩年多來歷經了許多的事情並且學習到許多的東西。

首先，感謝家人與親戚對我的支持與照顧，總是常常關心以及鼓勵我，讓我能專心讀書，因為有他們的一路的支持以及經濟上的支援，我才能完成我的碩士學業。

接著，我想感謝我的指導老師趙禧綠教授以及伍紹勳老師，謝謝老師教導了我許多做事的方法與態度，並且總是有耐心的與我討論問題幫助我解決問題。謝謝老師在我們參加比賽的那段時間每天都抽出時間幫助我們解決碰到的問題，以及提供許多協助，讓我們能完全專心的準備比賽。

最後，我也要感謝學長姐、同儕及學弟們。謝謝俊先學長每當我遇到問題時總是很耐心的與我一起討論以及提出建議。謝謝參加比賽時的戰友俊先、鈺涵、尚晟、庭瑋、昀庭，在比賽期間一路的陪伴，讓我可以度過那段最艱難的日子謝謝你們。謝謝同儕們在課業上總是提供我許多幫助，也謝謝學弟們時常為實驗室帶來歡樂。

感謝交通大學提供許多的資源給我，也感謝一路上所遇到的老師們在課堂上，在課堂上總是可以學到許多的知識，雖然即將畢業，但我會一直記得在交大生活的這兩年。

## 目錄

|  |     |
|--|-----|
| 摘要 .....   | I   |
| Abstract.....  | II  |
| 致謝 .....   | III |
| 表目錄 .....  | V   |
| 圖目錄 .....  | VI  |
| 第一章 前言 .....   | 1   |
| 1.1 無線感知網路(cognitive radio network) .....                      | 1   |
| 1.2 無線感知雲端網路(Cloud-Based Cognitive Radio Access Network) ..... | 2   |
| 1.2.1 無線感知雲端網路架構 .....   | 4   |
| 1.2.2 元件運作流程介紹 .....   | 6   |
| 1.2.3 統整與問題動機 .....  | 7   |
| 第二章 系統環境介紹與問題描述 .....  | 9   |
| 2.1 雲端架構問題 .....   | 10  |
| 2.1.1 HBase 架構 .....   | 11  |
| 2.1.2 問題討論 .....   | 12  |
| 2.2 分群與邊界點頻譜資源分配問題 .....                                       | 13  |
| 2.2.1 分群問題(Clustering) .....                                   | 13  |
| 2.2.2 邊界點頻譜資源分配問題(Coloring) .....                              | 15  |
| 第三章 提出方法與討論 .....  | 17  |
| 3.1 雲端架構 .....   | 17  |
| 3.1.1 資料庫表格優化 .....  | 17  |
| 3.1.2 行動資訊控管元件擴充條件 .....                                       | 20  |
| 3.2 分群與邊界點頻譜資源分配 .....   | 21  |
| 3.2.1 分群 .....   | 21  |
| 3.2.2 邊界點頻譜資源分配 .....  | 29  |
| 第四章 模擬數據結果與分析 .....  | 36  |
| 4.1 演算法比較 .....  | 36  |
| 4.2 大尺度系統測試 .....  | 38  |
| 4.2.1 參數及環境設定 .....  | 38  |
| 4.2.2 大規模模擬結果 .....  | 43  |
| 結論 .....   | 46  |
| 參考文獻 .....   | 47  |

## 表目錄

|    |   |    |
|----|---|----|
| 表一 | 表格設計(粗體為鍵值).....                        | 18 |
| 表二 | 分群演算法時間複雜度比較.....                       | 28 |
| 表三 | 頻譜資源分配演算法時間複雜度比較.....                   | 35 |
| 表四 | 最佳化與啟發式演算法比較.....                       | 37 |
| 表五 | 各地區模擬數值.....                            | 39 |
| 表六 | 台北市的移動通訊使用者(Mobile Users, MUs)數量估測..... | 40 |
| 表七 | 各類 MU 使用的資料流量以及需求速率.....                | 41 |
| 表八 | 台北市頻譜資源分享結果以及所需虛擬機器數.....               | 44 |
| 表九 | 和 3.5G 系統的比較.....                       | 44 |

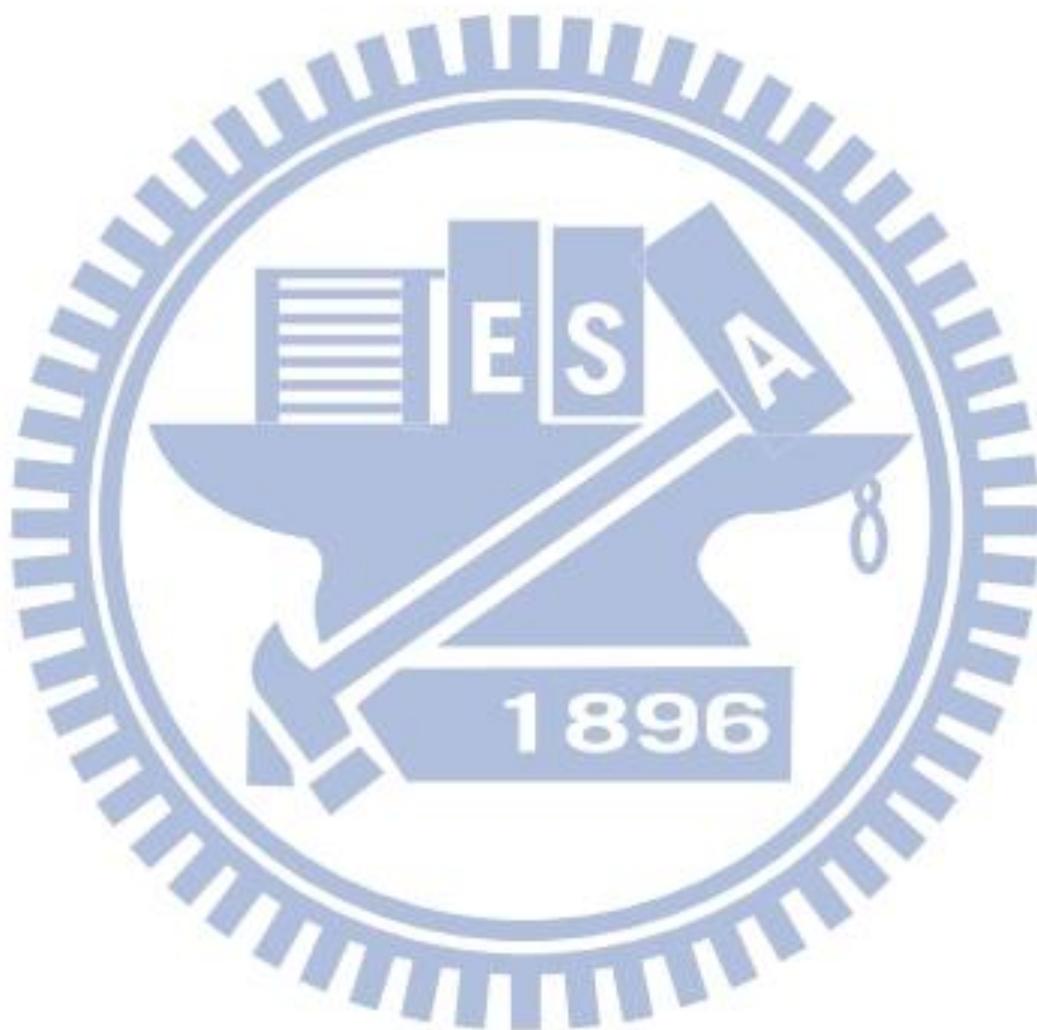


## 圖目錄

|      |                           |    |
|------|---------------------------|----|
| 圖一   | Cisco 所估測的未來網路趨勢.....     | 1  |
| 圖二   | 無線感知雲端網路示意圖.....          | 3  |
| 圖三   | 無線感知雲端網路架構圖.....          | 4  |
| 圖四   | 環境示意圖.....                | 6  |
| 圖五   | 不同虛擬機間因資訊不足,可能導致干擾發生..... | 8  |
| 圖六   | 可擴充性之無線感知雲端網路架構.....      | 9  |
| 圖七   | HBase 架構.....             | 11 |
| 圖八   | 分群問題拓樸環境轉換圖.....          | 13 |
| 圖九   | 邊界點頻譜資源分配問題拓樸環境轉換圖.....   | 15 |
| 圖十   | 資料取得方式比較.....             | 17 |
| 圖十一  | 無線感知雲端網路資料流向圖.....        | 19 |
| 圖十二  | 行動資訊控管元件測試.....           | 20 |
| 圖十三  | 最佳化分群演算法實例.....           | 22 |
| 圖十四  | 啟發式分群演算法選點(起始).....       | 25 |
| 圖十五  | 啟發式分群演算法選點(Step1).....    | 26 |
| 圖十六  | 啟發式分群演算法選點(Step2).....    | 26 |
| 圖十七  | 最佳化頻譜資源分配實例.....          | 29 |
| 圖十八  | 干擾值定義.....                | 31 |
| 圖十九  | 餘頻譜數實例.....               | 32 |
| 圖二十  | 邊界點頻譜資源分配流程圖.....         | 32 |
| 圖二十一 | 模擬的分區圖.....               | 38 |
| 圖二十二 | 在一天中不同時刻的同時上網人數.....      | 39 |
| 圖二十三 | Cisco NPI.....            | 41 |

圖二十四 每一個 AP 的實體通道(SU)需求數量.....42

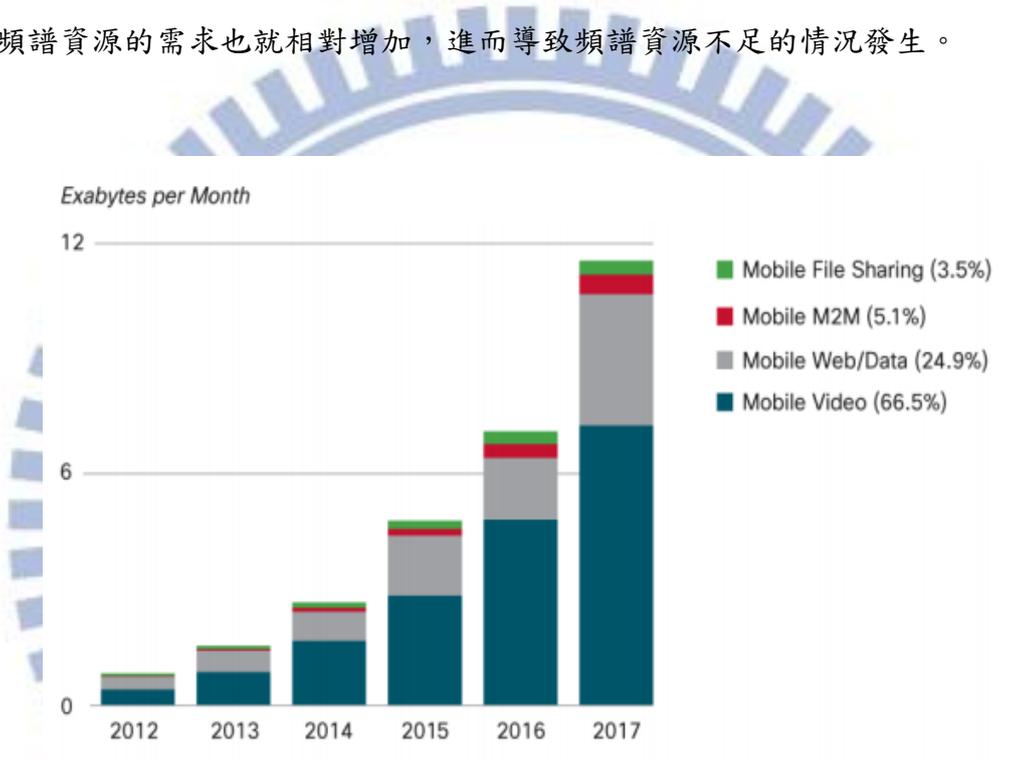
圖二十五 雲端整體系統執行時間.....45



# 第一章 前言

## 1.1 無線感知網路(cognitive radio network)

隨著科技快速的進步與發展，使用行動通訊裝置(例如：手機、平板電腦等)通過無線網路來上網的人快速增加，由(圖一)可以看到為思科(Cisco)[1]是對於未來網路使用的預估，可以看到對於通訊量的需求為指數性的成長，尤其在影音方面的需求更占多數，對於頻譜資源的需求也就相對增加，進而導致頻譜資源不足的情況發生。



圖一、Cisco 所估測的未來網路趨勢[1]

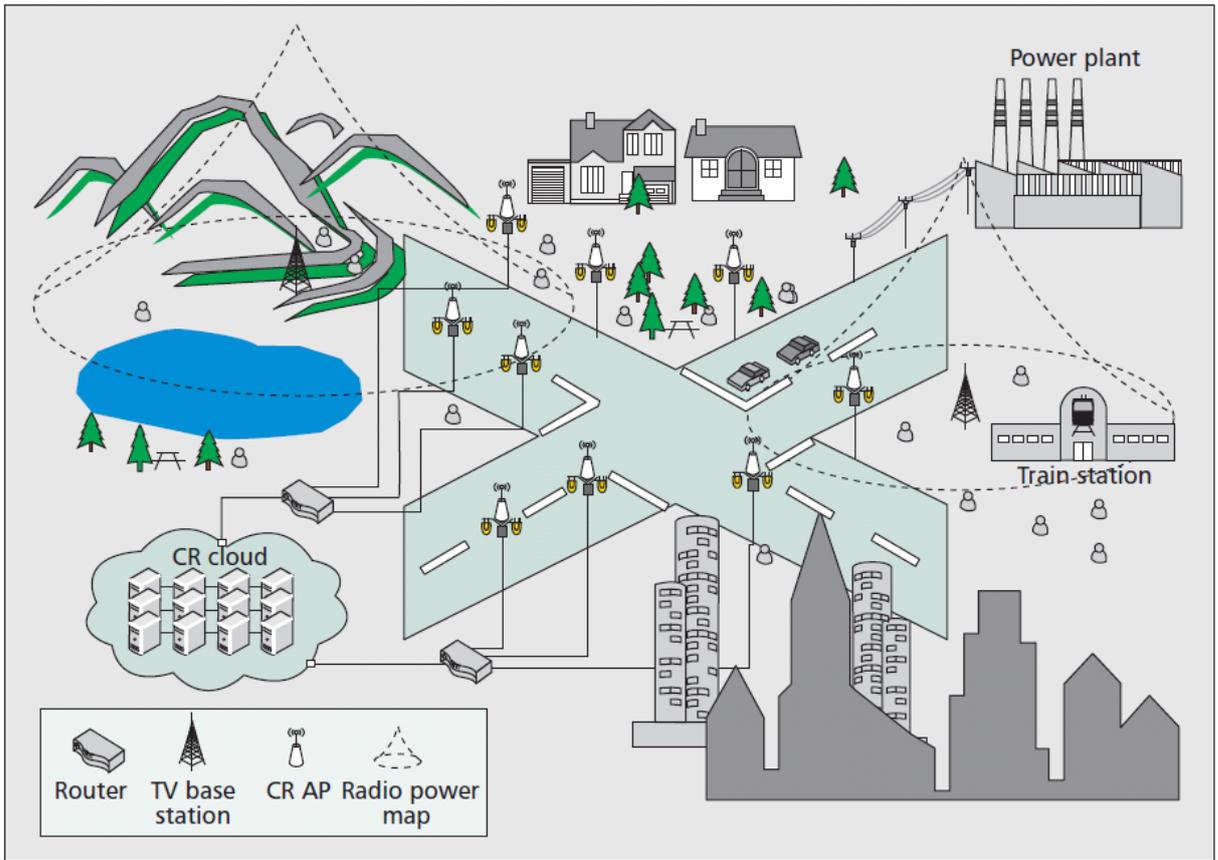
根據美國聯邦通訊委員會(Federal Communications Commission, FCC)在 2002 發表的量測報告[2]，根據量測報告結果指出頻譜資源的使用隨著空間與時間上的不同使用效率可從 15% 變化至 85%，並且多數頻譜的使用率均未達 25%，為了讓頻譜使用率提高，感知無線電技術(Cognitive radio)成為進一步提升頻譜使用率解決頻譜資源不足的問題。

感知無線電技術最早由 Joseph Mitola 所提出[3]，希望用來解決頻譜資源短缺的問題，主要想法為，分擁有頻譜執照的使用者(Unlicensed user/Secondary user, SU)能夠在擁

有頻譜執照的使用者(Licensed user/Primary user, PU)離開時，去借用頻譜使用，在不影響到擁有頻譜執照的使用者的情況下，以”借用”的概念提升整體頻譜的使用率。

## 1.2 無線感知雲端網路(Cloud-Based Cognitive Radio Access Network)

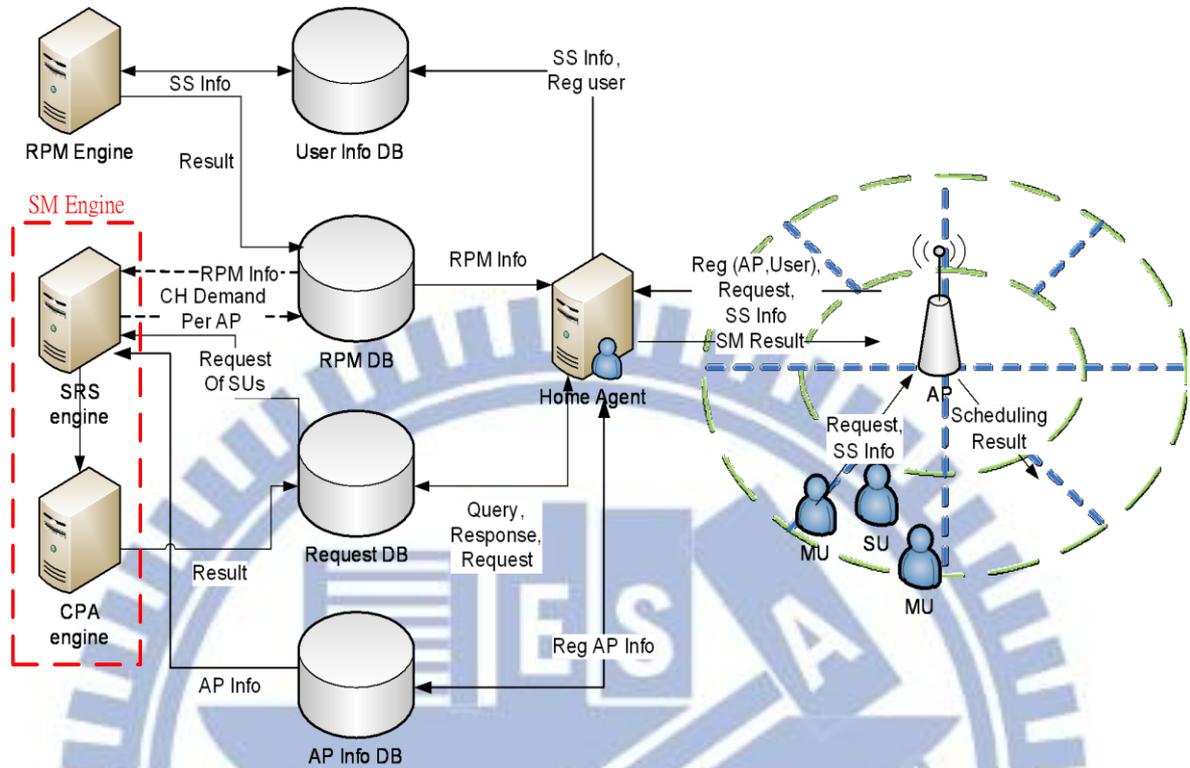
傳統上，為了提供較高的傳輸速率，需要增加使用者所接收到的訊號能量。這樣的需求，能夠藉由增加基地台的傳送能量，或是減少使用者至訊號源的距離來達成，由於能量在空氣介質中成指數衰減，加上手持裝置的上傳傳輸功率之限制，遍佈式的低功率基地台將會是較有效率的做法。因此在我們的系統中會布建大量的低功率的無線感知網路基地台(CR access point, CR AP)，但是考慮到在一大環境中會布建數千多個的無線感知網路基地台，要如何建立一個網路控管中心，管理如此巨量的無線感知網路基地台將是一個困難的問題，在[4][5]我們提出了無線感知網路與雲端的結合。藉由與雲端的結合可以將頻譜估測、頻譜管理、甚至於針對使用者的換手(hand-off)等等，(如圖二)[5]將雲端比喻成發電廠，路燈比喻成無線感知網路基地台，使用者可以在任何地方使用免費的無線網路並且可以無縫的換手就像夜晚走在城市中到處都有路燈的照明。而整合在雲端上並可帶來許多的好處，第一、因為是中央式的控管方法，所以我們可以根據目前系統中的使用者數量來增減雲端中的資源，一來在尖峰的時段可以開啟較多的資源來負擔過大的伺服器工作量，在另一方面離峰的時段可以關閉一些資源達到節省能源的效果，第二、雲端主要優勢在於有著強大的運算能力與儲存空間，藉由雲端，我們可以有效的管理成千上萬的頻譜感測數據，並且有效的估測出目前頻譜的狀況。第三、雲端所在之資料中心位於骨幹網路之節點，可以提供高速網路接取，這樣一來每個無線感知網路基地台都可以快速的將資訊傳至雲端中心。



圖二、無線感知雲端網路示意圖[5]



## 1.2.1 無線感知雲端網路架構



圖三、無線感知雲端網路架構圖

(圖三)為無線感知雲端網路的架構圖，在環境中無線感知網路基地台會蒐集 MU(Mobile User)產生的需求並且將這些需求群組為 SU(Super User,在雲端分配資源皆以 SU 為單位)即時的上傳到雲端，其在雲端上的功能可以分成下列幾項，合作式頻譜估測 (CSS)、頻譜資源管理(SM)、資料庫(DB)以及行動資訊控管元件(Home Agent)，而各個功能皆運作在不同的機器上，而其功能將在下面一一介紹。

## 合作式頻譜估測(cooperative spectrum sensing, CSS)

在我們的系統裡面，為了符合感知無線電的功能，保護優先權使用者通訊品質，必須要有一套機制來管理 SU 對於頻帶的存取，需要一套頻譜估測演算法決定 PU 的功率覆蓋範圍。在 FCC 規定中，SU 必須存取無線功率圖譜資料庫，該資料庫必須包含優先權使用者之位置、傳輸功率、頻帶等資訊。

為了達成 FCC 的規範，並且對抗無線通道中的遮蔽效應(shadowing effect)或是隱藏點問題(hidden node problem)，我們提出一合作式頻譜估測演算法[15]，藉由蒐集各不同位置的感測結果，透過演算法估測 PU 的位置。我們在不干擾優先權使用者的前提下，將可以根據無線感知網路基地台的地理位置給予不同無線感知網路基地台動態的通訊資源，增加頻譜的使用效率，並能夠避開不同通訊系統之間的相互干擾。

## 頻譜資源管理(Spectrum management, SM)

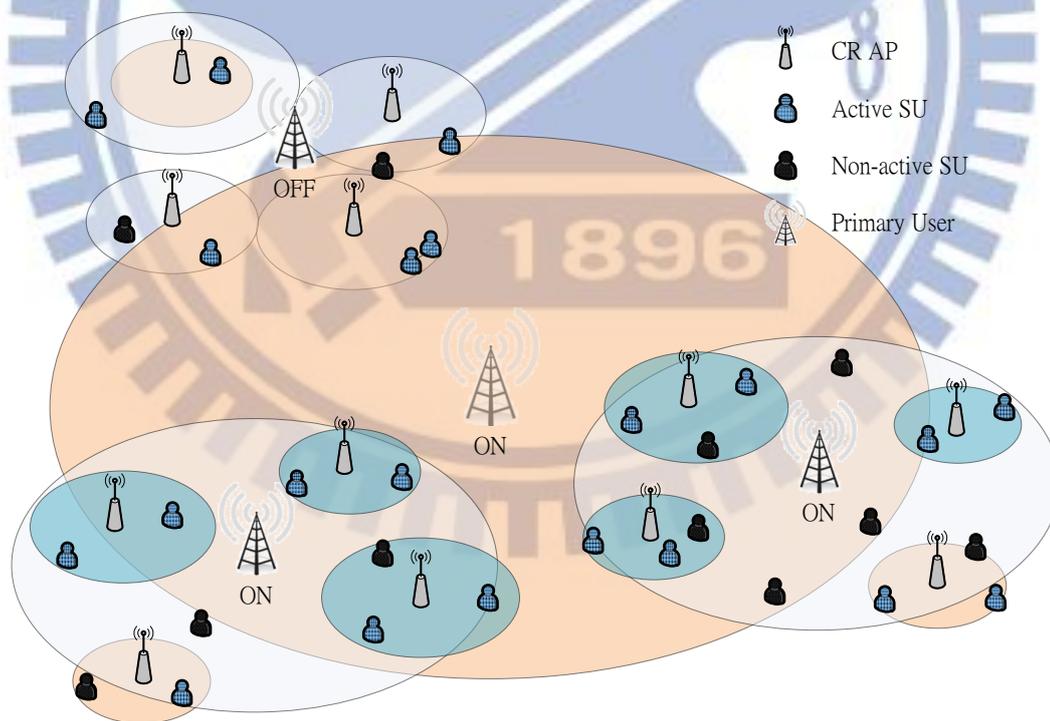
頻譜資源管理包含了頻譜資源分享(Spectrum Resource Sharing, SRS)以及頻道配置與功率配置(channel and power allocation, CPA)，頻譜資源管理負責將一大區域分成數個子群為了讓頻道配置與功率配置能夠即時的運算出分配結果，頻譜資源分享主要是根據一大區域每小時的統計值為依據將環境做切割，因為在短時間中，環境內的 MU 集中區域並不會有嚴重的改變。而頻道配置與功率配置主要負責將沒有 PU 存在的頻譜資源與功率，配置給各個無線感知網路基地台，由無線感知網路基地台再分給底下的 SU。

## 行動資訊控管元件(Home Agent)

行動資訊控管元件主要負責雲端對無線感知網路基地台的連絡窗口，會接收下層無線感知網路基地台所收集到的 SU 資訊，例如：SU 加入、需求、頻譜狀況等等，並且由行動資訊控管元件將這些資訊存到資料庫(DB)中，以及將頻譜資源管理的結果從資料庫中取出告知無線感知網路基地台。

## 1.2.2 元件運作流程介紹

這一小節將簡單介紹整個通訊的流程，(如圖四)環境中有三個小區域而每個小區域內有各有 4 個無線感知網路基地台，並且用不同顏色代表不同的頻譜，在其中一個頻譜上有一全域 PU 並且三個小區域內各自存在一個小範圍的 PU，Active SU 表示此 SU 正在通訊，而 non-active user 則反之。首先環境中的每個 MU 會將感測到的頻譜資訊以及通訊需求送至無線感知網路基地台，無線感知網路基地台收集這些訊息後，發送至雲端上的行動資訊控管元件，行動資訊控管元件將這些資訊存到資料庫中後，合作式頻譜估測會根據 MU 的感測報告將環境內的 PU 偵測出來，而頻譜資源管理會將 PU 沒在使用的頻譜分配給無線感知網路基地台使用以及調配其功率大小，過一小段時間後之後再由無線感知網路基地台向詢問行動資訊控管元件分配結果，無線感知網路基地台根據拿到的結果分配給其底下的 SU。

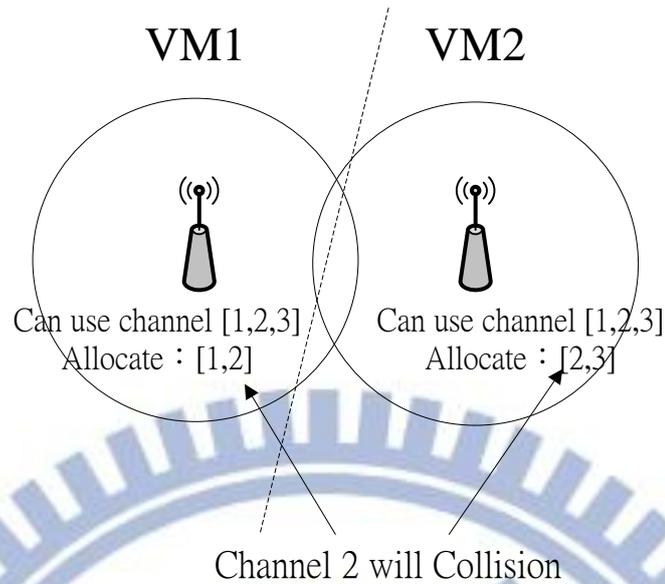


圖四、環境示意圖

### 1.2.3 統整與問題動機

在無線感知網路雲端中，我們考慮的環境是管理成千上萬的無線感知網路基地台。在這樣大的環境之下，不論是頻道配置與功率配置以及行動資訊控管元件都不可能單靠一台的虛擬機器就可以即時的服務使用者，考慮到通訊中即時性的需求，我們嘗試去改進我們雲端中的架構。

在這篇論文中我們主要會針對我們雲端中二個部份作探討。第一、對於雲端系統架構的討論如何讓我們雲端中各個部份都能支援擴充，而對於資料庫的存取對於時間也是很急迫的，要如何設計出合適的表格(Table)去存放各個資訊，讓提取與存取資料時能省下時間也是一重要的問題。第二、對於如此大的環境去做頻道配置與功率配置必定無法即時的反應，那麼要如何適當的將這一大尺度的環境做切割，區分為多個子群，將這些子群分別放到不同的虛擬機器上去執行頻道配置與功率配置以及因為若將這些子群分派到不同的虛擬機器管理，對於子群間邊界上的無線感知網路基地台彼此間可能會產生干擾，舉例來說(圖五)圖中兩個無線感知網路基地台分屬於不同子群各有 3 條頻譜可以使用(頻譜 1、頻譜 2、頻譜 3)，若沒有先對其做初步的頻譜資源分配，在頻道配置與功率配置時因為看不到相鄰子群的無線感知網路基地台頻譜配置情況，而有可能會分配相同頻譜進而導致互相干擾的情況發生。在分群問題上已有許多演算法被提出[6][7]，可是這些分群方法沒有考慮分出來子群的連通性，使得分出來的子群可能由多個不連通的子群組成，在我們的系統中必須考慮到地區性的管理，這些方法並不是用於我們的系統，而邊界點頻譜資源分配問題，如同多著色問題，在[8]已經被提出並且證明是 NP-Hard 的問題，在[9]也有類似的問題，但須要花費過多時間，不適合我們系統。

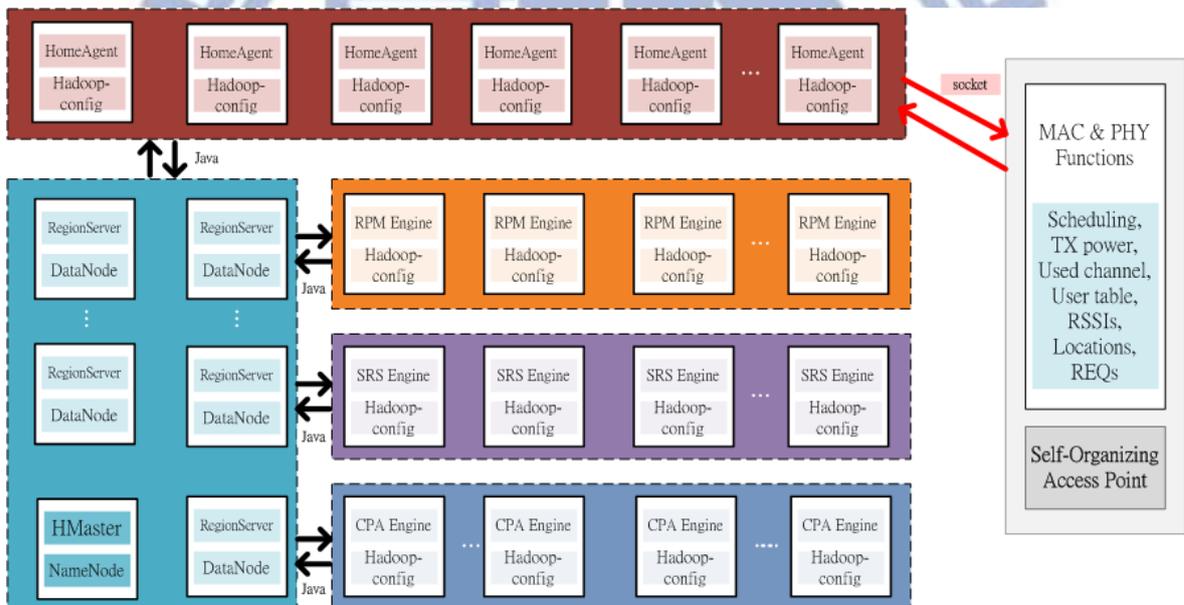


圖五、不同虛擬機間因資訊不足，可能導致干擾發生

在這篇論文中，我們會針對上述三個問題分別去做探討，對於第一與第二個問題我們會以模擬測試的方式去對其作討論，對於問題三我們會對其的分群、子群間邊界點頻譜資源分配分別提出最佳化演算法，並且做理論上的證明與時間複雜度的分析，但是最佳化演算法非常耗時，因此我們常是針對目標設計好的啟發式演算法，並且分析其時間複雜度以及用數據模擬與最佳化比較，最後我們會針對一個大尺度的環境在雲端上真實實作並配合模擬產生 SU 需求和位置並模擬無線感知網路基地台的行為將資料上傳到我們的系統，來驗證我們系統是真實可行的。

## 第二章 系統環境介紹與問題描述

在我們的無線感知雲端網路中，我們考慮的是一個大尺度的環境，我們的雲端必須能夠服務數以萬台的無線感知網路基地台，並且能夠即時的配置資源，若(圖三)中的每個元件皆只以一台虛擬機去做運算是無法即時的去分配出資源，所以我們雲端架構必須能夠平行運算，如(圖六)我們將每個元件獨立開來看並且每個部份都有各自的擴充條件，而如果要支援平行運算與追求即時的服務那麼我們將會碰到許多問題將在下面一一說明。



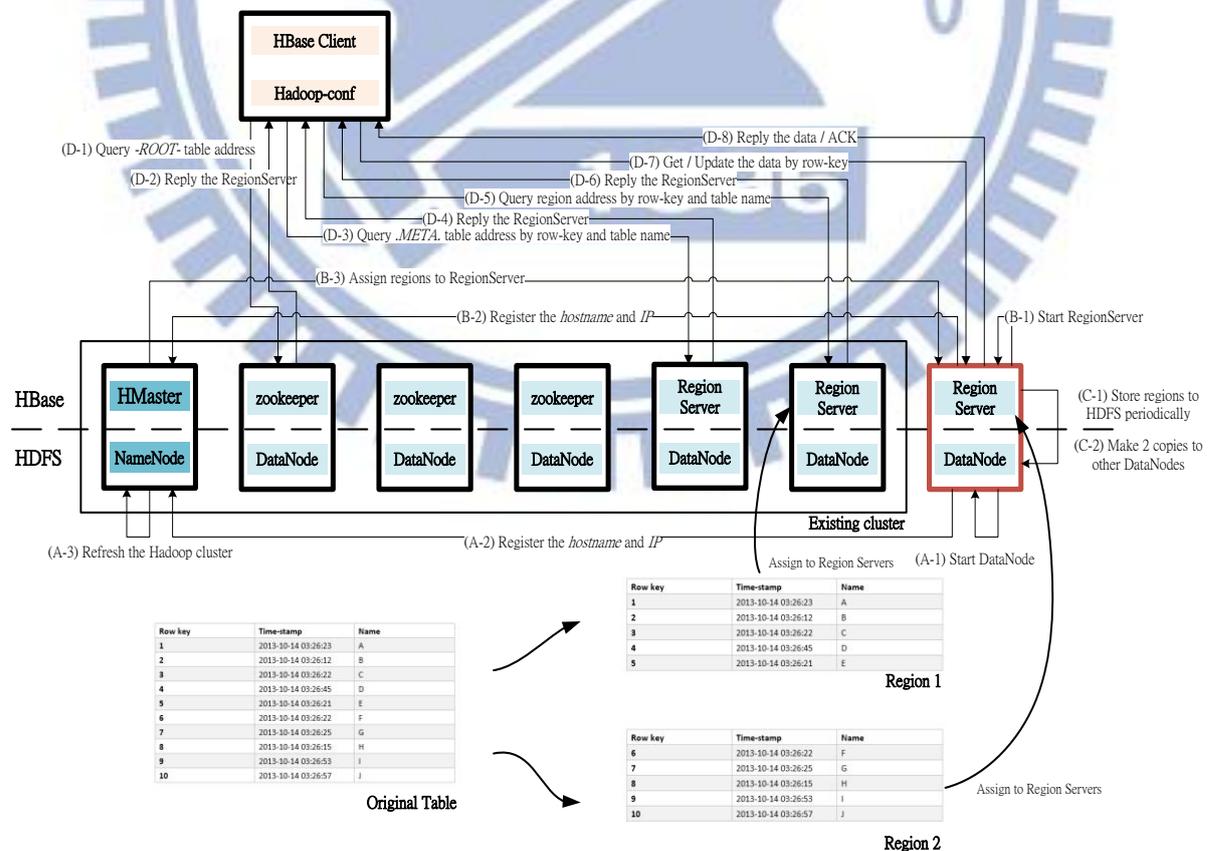
圖六、可擴充性之無線感知雲端網路架構

## 2.1 雲端架構問題

在無線感知雲端網路系統中我們選擇 HBase 作為我們的資料庫，因為 HBase 有許多的優點，第一、HBase 為一開源的系統，便於我們可以根據我們的需求對系統做修改，把其運作變得更適合我們的系統，第二、HBase 起初的設計目的就是為了使用在大量資料的儲存上，像是 facebook 也使用 HBase Hive 作為其資料庫，而我們的無線感知雲端網路系統為了管理數以萬計的無線感知網路基地台與使用者，有著大量的資料也是適用於 HBase 的，第三、在無線感知雲端網路系統中我們追求即時性的服務，所以我們希望資料庫的存取時間能夠越快越好，而 HBase 並不同於關聯性資料庫。傳統的關聯性資料庫，每次存取時都以一整張表格(Table)為單位，存取速度較慢，而 HBase 則將資料按其 row-key 和 column 的數值分開儲存，以利快速存取，根據[11]的測試，在讀寫比例為 1:1 的時候關聯性資料庫的讀寫時間都高於 HBase，考慮我們的應用，寫入與讀取的需求比例相同 HBase 較為合適，在另一方面 HBase 可以根據儲存的資料量，動態調整所使用的資源，因此表格資料量大時，能夠簡單的調整(scale-out)所使用的叢集，並能夠調整每個節點負載，相對而言關聯性資料庫則難以動態的調整資源或是分配負載於各節點。

## 2.1.1 HBase 架構

HBase 架構如(圖七)所示，在 HBase 中基本的儲存單位為 Region，用來儲存使用者定義的表格，當使用者想要讀取或存取資料時，將透過 Region Server 幫忙從 Region 將資料寫入 Region 或是取出，每個 Region Server 將會管理數個 Region(s)，Region Server 提供寫入與讀取的服務，並且定時將 Region 的資料存入到 hdfs 中，hdfs 負責資料的備分與分散式儲存(圖七-C)。而在 HBase 中的表格在一開始將被 HMaster 預先指定給 Region Server 存在其所管理的 Region 中，當表格中資料增長並超出 Region 所容量大小時，將根據 row-key 數值被切割成兩個 Region 儲存此表格，如(圖七)中起始表格存放 row-key 1~10 當超出 Region 大小時將 row-key 1~5 資料存至一個 Region 而 row-key 6~10 存放至另一個 Region，並存放於原本的 Region Server 中，在一段時間之後由 HMaster 負責平衡 Region Server 所負責的 Region。



圖七、HBase 架構

Region 和 Region Server 的關係被記錄在.META.的表中，.META.表也將被分成多個 Regions 存放於不同 Region Server，其對應關係被記錄在唯一的-ROOT-表中，由一群同步的 zookeeper 所管理，當一個使用者進行查詢時，將先根據表單名稱以及 row-key 向 zookeeper 查詢對應的.META.表所在位置，並進一步得到，所查詢目標所在的 Region Server，之後直接和 Region Server 溝通(圖七-D)。

## 2.1.2 問題討論

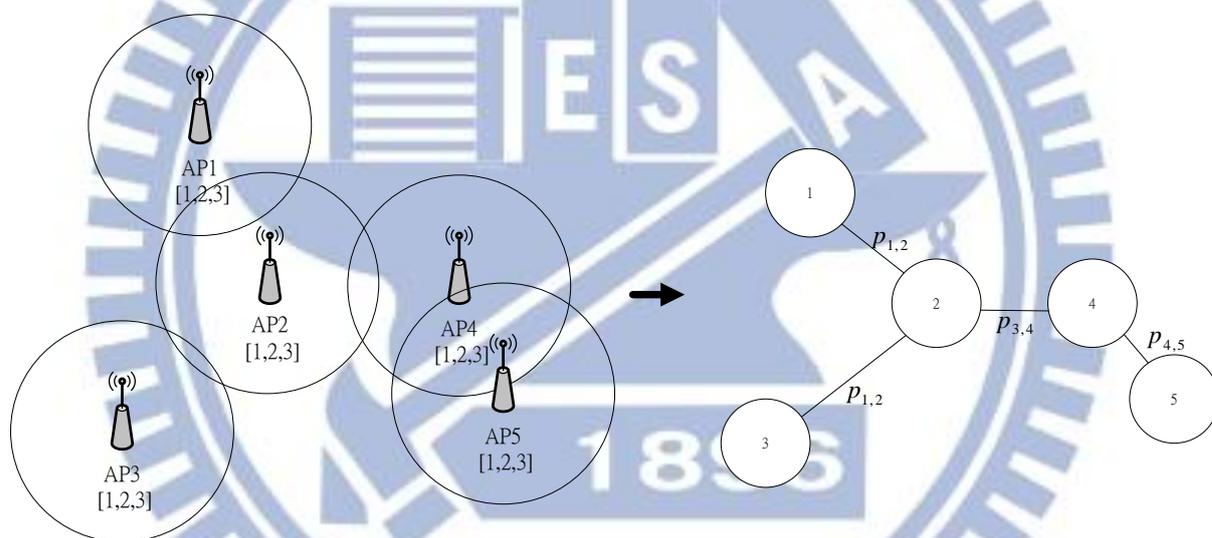
雖然使用 HBase 可以為我們系統帶來許多好處，但是要如何讓我們存取 HBase 可以更為快速，這就關係到我們對於 HBase 中表格的設計以及取得資料的方法，在後面的章節中，我們會根據不同的表格以及不同的資料量與讀取方式去做測試，藉此找到適用於我們系統的表格架構，並且根據一些已經有的 HBase 測試結果文章來對我們的 HBase 做一些校調，希望對於 HBase 存取時間能對我們的系統造成最小的影響。

再另一方面，我們的系統想要支援大量的無線感知基地台與使用者，是必須要讓各個元件有擴充的能力，像是行動資訊控管元件若在大量無線感知網路基地台的環境下，不可能只靠一個行動資訊控管元件就能即時的服務，所以勢必得對其做擴充的功能，在後面的文章中我們會依據模擬測試來討論擴充的條件。

## 2.2 分群與邊界點頻譜資源分配問題

為了要讓頻道配置與功率配置可以即時的完成，我們要對一大環境做切割，而對於我們的環境我們考慮無線感知網路基地台是均勻分布的，而 MU 為均勻分布，並且我們假設每個無線感知基地台所打的最大功率皆相同，並且假設頻譜都擁有相同的資料傳輸率(data rate)，而無線感知基地台可用頻譜則來自合作式頻譜估測的估測結果。

### 2.2.1 分群問題(Clustering)



圖八、分群問題拓樸環境轉換圖

在環境中我們將無線感知網路基地台的拓樸(topology)轉變成一無向圖  $G=(V,E)$ (圖八)， $V$  代表圖中的點， $E$  則代表邊，圖中每個點表示無線感知網路基地台，而無線感知網路基地台之間若功率覆蓋範圍有互相重疊到那麼在我們的無向圖中兩個無線感知網路基地台間就存在邊(edge)  $e_{i,j}$ 。並且在邊上有著一值”penalty”，其表示意義為必定不被滿足的需求量，兩個無線感知網路基地台  $i,j$  之間的 penalty 值  $p_{i,j}$  定義如下

$$p_{i,j} = \max\{0, [(Y_i + Y_j) * \frac{A_{i \cap j}}{2A - A_{i \cap j}}] - (|M_i \cap M_j| * \frac{A_{i \cap j}}{2A - A_{i \cap j}})\}$$

Where

$Y_i$ : the request of AP I

$M$ : channel set

$M_i$ : channel set of AP i

$A$ : the AP coverage area

$A_{i \cap j}$ : intersection area of AP i and AP j

前半部  $((Y_i + Y_j) * \frac{A_{i \cap j}}{2A - A_{i \cap j}})$  為兩個無線感知網路基地台互相交集區域所占的需求比例，

後半部  $(|M_i \cap M_j| * \frac{A_{i \cap j}}{2A - A_{i \cap j}})$  則代表兩個無線感知網路基地台互相交集區域所占的頻譜

資源比例，因此  $p_{i,j}$  表示超出現有資源而必不能被滿足的需求數量。

給一圖  $G=(V,E)$  我們將我們的最佳化問題定義如下

$C_k$ : set of APs in cluster k

$X_{i,j}^{b-edge}$ : boundary node edge indicator  $\begin{matrix} \text{if } e_{i,j} = 1, & i \in C_k & j \in C_{k'} & \text{then } X_{i,j}^{b-edge} = 1 \\ \text{otherwise } & & & X_{i,j}^{b-edge} = 0 \end{matrix}$

Object :

$$\min \sum_{\substack{i,j \in V \\ e_{i,j} \in E}} X_{i,j}^{b-edge} p_{i,j}$$

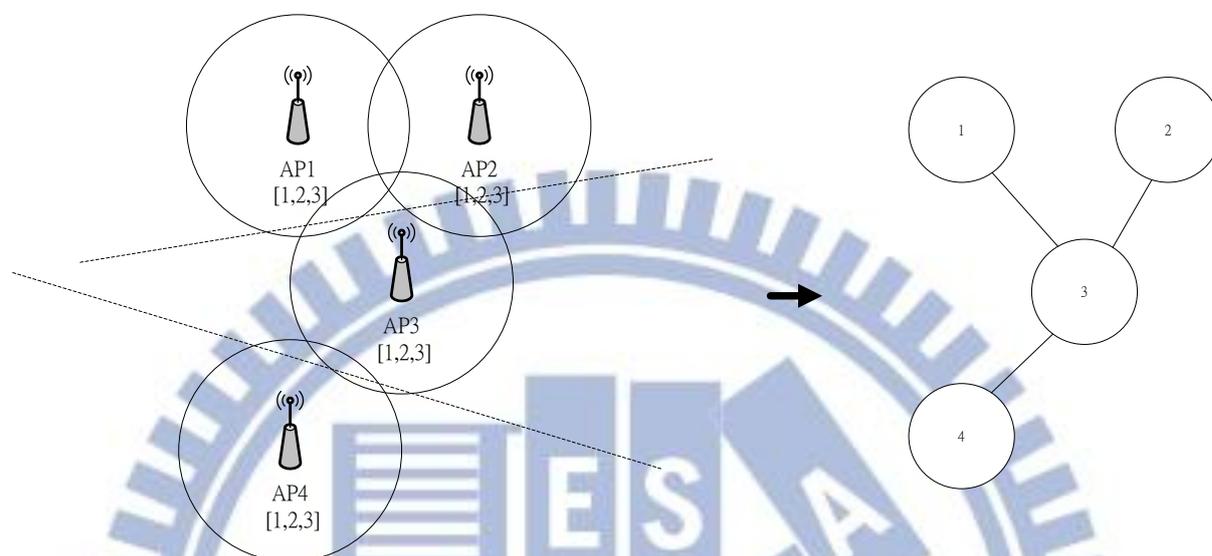
Subject to :

$$\bigcup_k C_k = V, C_k \cap C_{k'} = \phi, \forall k, k', k \neq k'$$

$$Path(i, j) = 1, \forall (i, j) \in C_k, \forall k$$

$$Loading(C_k) \leq W, \forall k$$

## 2.2.2 邊界點頻譜資源分配問題(Coloring)



圖九、邊界點頻譜資源分配問題拓模環境轉換圖

在分群演算法執行過後，若兩個無線感知網路基地台之間存在邊，並且分屬於兩個不同的子群並且不可使用相同頻譜，那麼我們將這兩個無線感知網路基地台都定義為邊界無線感知網路基地台，而我們也會將邊界無線感知網路基地台拓模轉換為一無向圖  $G=(V,E)$ (圖九)，並且只有不同子群間的邊界無線感知網路基地台存在邊，因為在這只考慮不同子群之間的頻譜資源分配，而子群內的資源分配將在下一個元件頻道配置與功率配置中執行。

在這部份的問題中，我們希望能夠將邊界無線感知網路基地台的頻譜資源分出去最多，我們假設在分群結束後有  $B$  個邊界無線感知網路基地台並且用  $I_B$  代表邊界無線感知網路基地台集合。給一圖  $G=(V,E)$ ，對於點  $i$  可用的頻譜資源為  $M_i$ ，我們將我們的最佳化問題定義如下

Object :

$$\max \sum_i^B |\overline{M}_i|$$

Subject to :

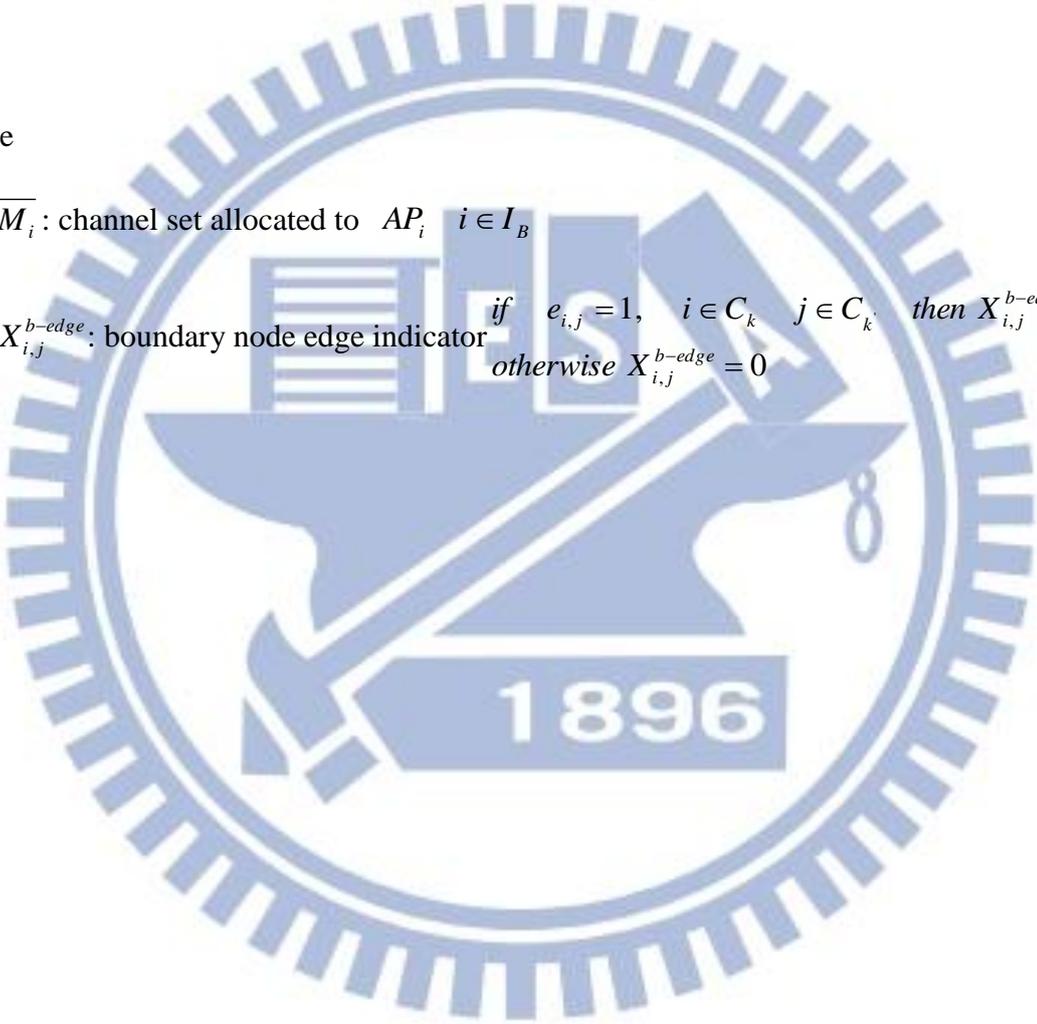
$$\overline{M}_i \cap \overline{M}_j = \phi, \quad \forall X_{i,j}^{b-edge} = 1, \quad i \neq j$$

$$\overline{M}_i \subset M_i, \quad \forall i \in I_B$$

Where

$\overline{M}_i$  : channel set allocated to  $AP_i$   $i \in I_B$

$X_{i,j}^{b-edge}$  : boundary node edge indicator *if*  $e_{i,j} = 1, \quad i \in C_k, \quad j \in C_k$  *then*  $X_{i,j}^{b-edge} = 1$   
*otherwise*  $X_{i,j}^{b-edge} = 0$



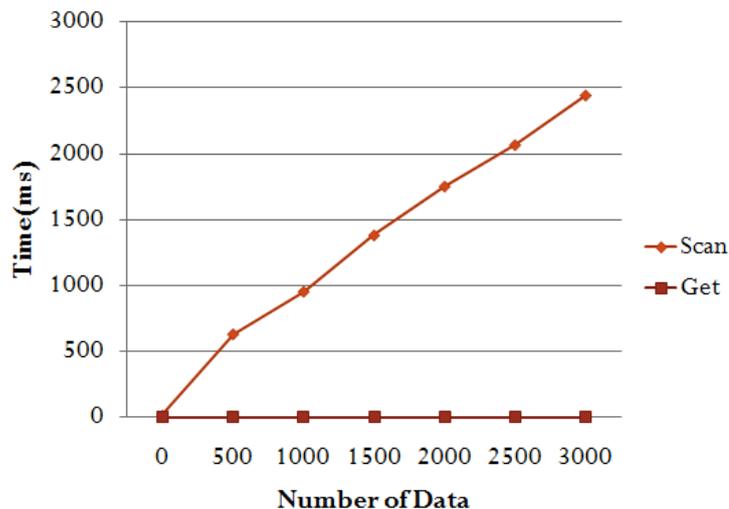
## 第三章 提出方法與討論

### 3.1 雲端架構

在這一章節我們將會對於我們的系統設計不同的表格，並且以模擬的方式來驗證如此設計是比較好的，並且根據一些其他文章的測試結果，來對我們的HBase做一些校調，使得效能可以變好。

#### 3.1.1 資料庫表格優化

在這裡我們希望設計出好的資料表格架構，來讓我們能夠花費很少的時間來取得我們要的資料，在我們系統中，我們會去記錄MU、SU、無線感知網路基地台的各種資訊以及各個演算法的執行結果，其中頻道配置與功率配置須要即時的將資源配置給各個無線感知網路基地台，要如何快速找到每個無線網路基地台分屬於哪個子群為表格設計的重點，我們先對於HBase兩種讀表格的方法分別為Scan與Get做比較，Scan為從Hbase中取得整張表格，而Get為從表格中取得一筆資料，我們表格內存放不同資料筆數對兩種方使做比較(圖十)，可以看到隨著資料量越大Scan越花費時間，因此我們應該多使用get來取得我們的資料。



圖十、資料取得方式比較

基於上面的測試結果我們提出 (表一)來做為我們給頻道配置與功率配置快速查詢用的表格，頻道配置與功率配置可以先藉由 SRS Table 中 Get 所負責子群底下有哪些無線感知網路基地台，再從 AP Info Table Get 出先前查到的無線感知網路基地台底下 SU 提出的要求的訊息，。

AP Info Table

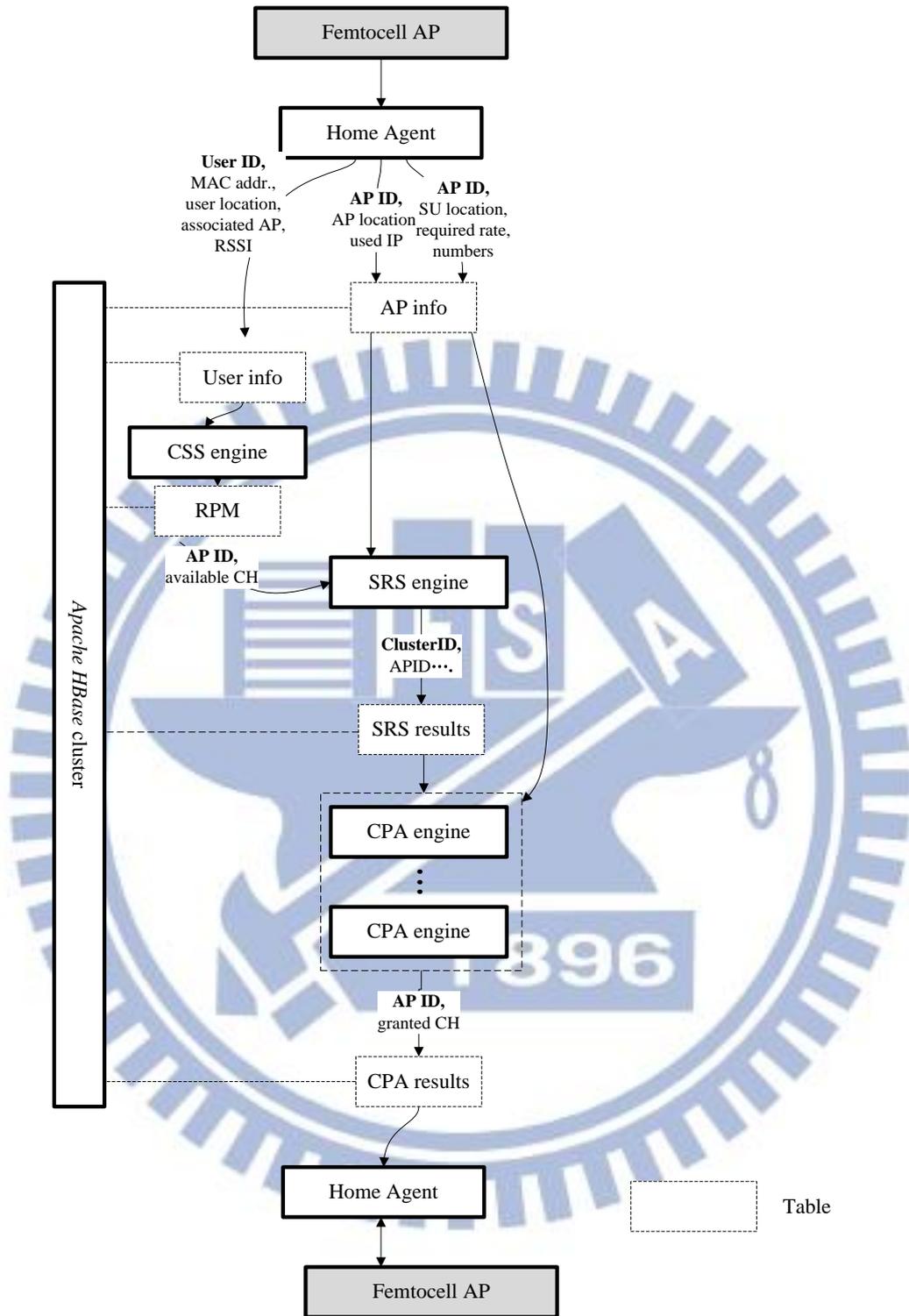
|              |       |      |      |            |              |            |
|--------------|-------|------|------|------------|--------------|------------|
| <b>AP_ID</b> | AP_IP | AP_X | AP_Y | AP_Channel | Request Info | ColoringCH |
|--------------|-------|------|------|------------|--------------|------------|

SRS Result Table

|                   |           |
|-------------------|-----------|
| <b>Cluster_ID</b> | AP_ID.... |
|-------------------|-----------|

表一、表格設計(粗體為鍵值)

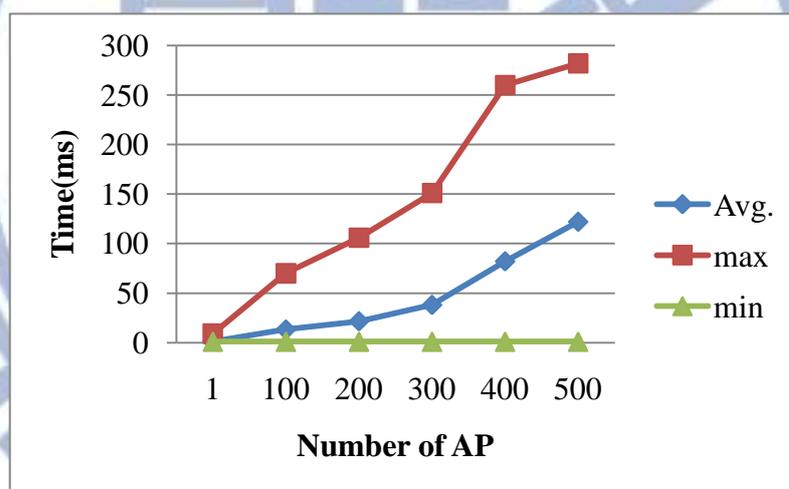
因此對於我們系統中的資料流向為(圖十一)，一開始的資料由無線感知網路基地台送入行動資訊控管元件，行動資訊控管元件根據資料種類分散儲存到不同的表格當中，MU 的資訊存放置 User Info 表格中，無線感知網路基地台以及 SU 的資訊則儲存在 AP Info 表格，合作式頻譜估測元件拿取出 User Info 表格訊息後判斷哪些頻譜可以使用存放到 RPM 表格，而頻譜資源分享元件根據 RPM 表格與 AP Info 表格中資料切割出多個子群，將資料存入 SRS Result 表格中，最後由頻道配置與功率配置元件根據 AP Info 表格與 SRS Result 表格資訊分配出所負責子群的頻譜資源將結果放入 CPA Result 表格中，由無線感知網路基地台詢問行動資訊控管元件之後取得。



圖十一、無線感知雲端網路資料流向圖

### 3.1.2 行動資訊控管元件擴充條件

在這裡我們以模擬測試的方式對行動資訊控管元件的擴充做討論，我們想要測試在同一時間內行動資訊控管元件能夠負載多少的無線感知網路基地台連線進來透過行動資訊控管元件向 HBase 取得頻道配置與功率配置的結果。我們以一台 1 個核心 2G 記憶體的虛擬機器做為行動資訊控管元件，並且透過其他虛擬機器模擬無線感知網路基地台，模擬結果(圖十二)顯示出，我們的行動資訊控管元件服務到約 400 台無線感知網路基地台時平均服務時間落在 82(ms)，而到了 500 台無線感知網路基地台時平均服務時間落在 120(ms)，我們這段時間花費太久，所以將我們擴充界限定在 400 個連線術時進行擴充，也就是一台行動資訊控管元件可以服務 400 台左右的無線感知網路基地台。



圖十二、行動資訊控管元件測試

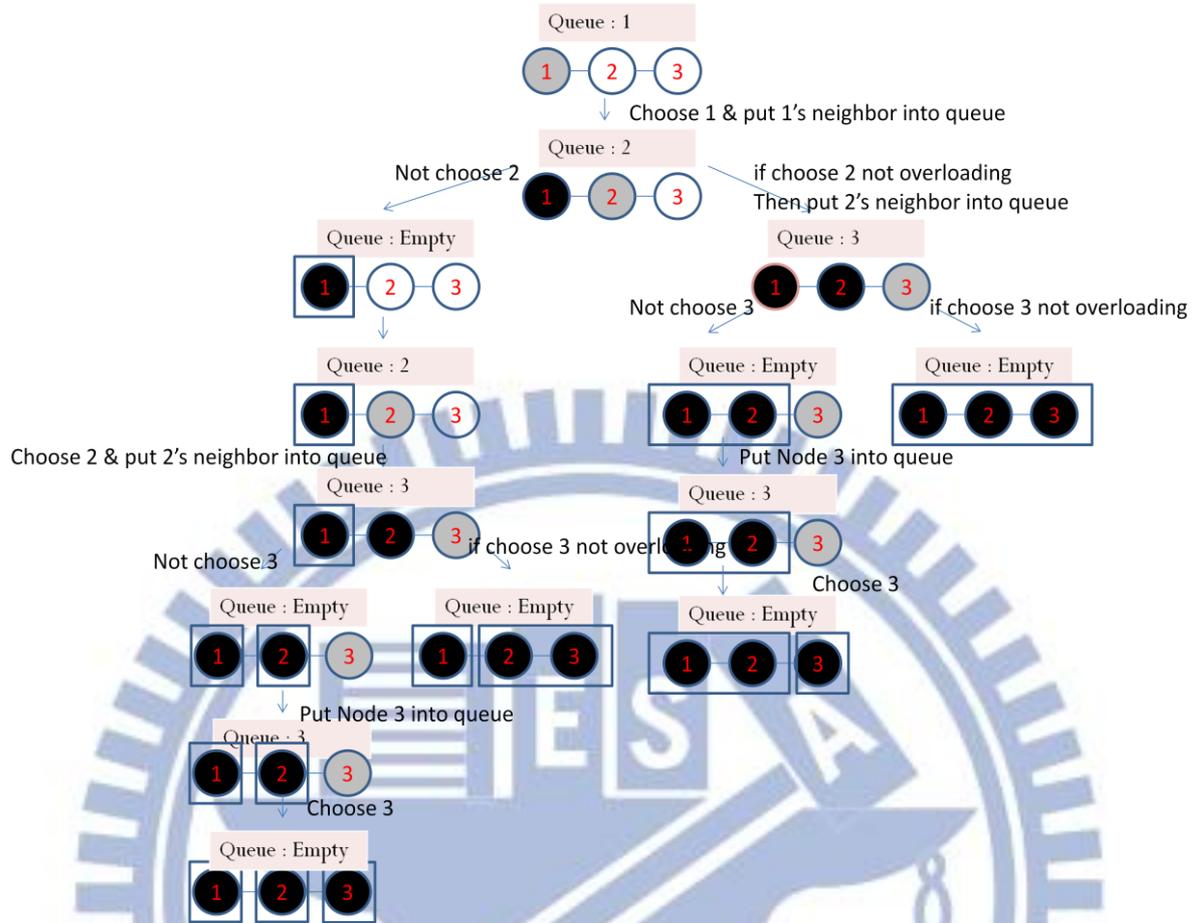
## 3.2 分群與邊界點頻譜資源分配

在這一章節我們將會對於分群以及邊界點頻譜資源分配的問題分別提出最佳化演算法以及因為最佳化演算法所花費的時間複雜度太高，為了因應我們系統的需求所以提出一啟發式的演算法降低複雜度，並且在最後分別對於兩個問題去分析最佳化與啟發式演算法的時間複雜度。

### 3.2.1 分群

#### 3.2.1.1 最佳化分群介紹

此最佳化演算法主要藉由找到所有滿足限制下分群的可能性，並且從這些可能中找出最佳解，我們使用到廣度搜尋法(Breadth-first search, BFS)的概念來以遞迴方式搜尋所有可能，在一開始我們任取一點放進隊列(queue)中，(圖十三)中為點 1 作為這一個子群的起始點，並且對於每個子群的起始點我們只考慮選的情況，再來會將此點的鄰居放入對列中，下一步從對列中取出一點，並且對此點在不超過工作負載量的前提下”選”或”不選”兩種可能皆會考慮，若當對列為空時則開始下一個子群的選點，當找出滿足限制下所有分群可能性時，演算法才會結束，詳細過程可參照(圖十三)以及 pseudo code。



圖十三、最佳化分群演算法實例

**Parameter :**

p : temp Penalty

C : cluster number

Node : AP's ID

Node.CN : Cluster's Number of Node

Node.color : AP's status

- White : 1.Non-clusterd
- Black : 2.Clustered
- Gray : 3.Not be clustered but be checked

**Initial :**

Penalty p =  $\infty$

all Node.CN = -1

all Node.color = white

C = 0

**Main()**

Push Node 0 into Queue

BFS\_OPT(Node 0, Queue Q, ClusterNum 0)

## **BFS\_OPT(Node N, Queue Q, ClusterNum C)**

if(Q is not empty)

// find all BFS sub-graphs

- tempNode T = pop node from Q
- Color T to Gray
- Record the Q and color
- If(Select node T will not cause overloading) //select this node
  - Color T to Black
  - Choose node T , set node T.ClusterNumber = C
  - Push all neighbors of T into Q which are not already in Q and its color is white
  - BFS\_OPT(N , Q , C)
- If(T != N) //avoid first round not choose node
  - Recover the Q and color and set the cluster number of T.ClusterNumber = -1 // case for not choose Node T
  - Not choose this Node T
  - BFS\_OPT(N , Q , C)

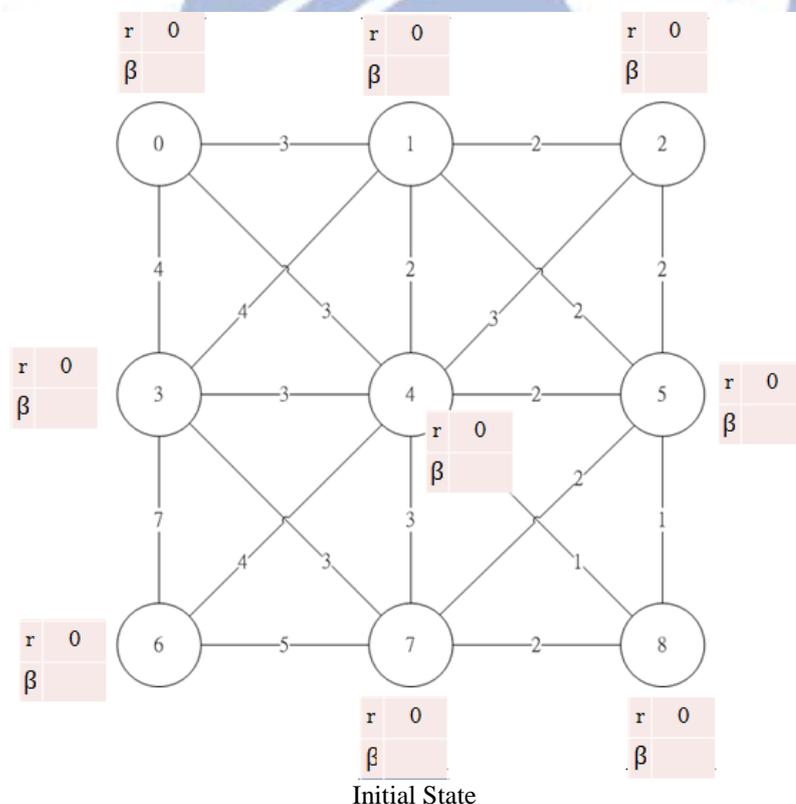
Else

// this cluster's process finish

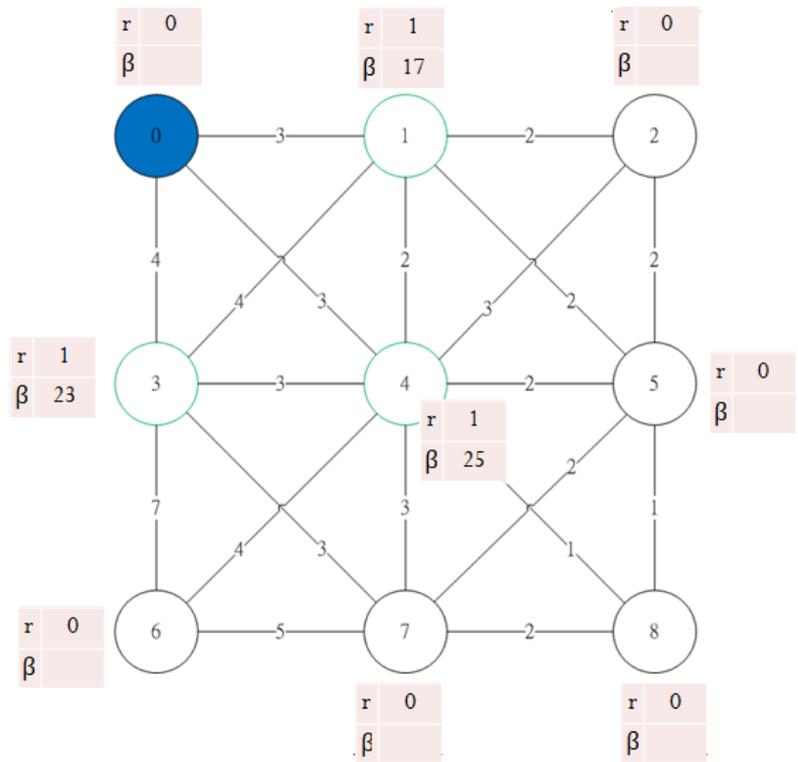
- if(all node be clustered)
  - if (this solution penalty < p )  
p = this solution penalty
- else
  - for all Node if(Node.CN == -1)  
set Node.color = White // recover
  - Find the not clustered node N' //initial new cluster
  - Push N' to Q
  - Start next cluster, BFS\_OPT(N' , Q , C+1)

### 3.2.1.2 啟發式分群介紹

在啟發式演算法中每個子群的起始點我們找一度數(degree)最低的點當做起始點，為了防止分群到最後的零碎點太多，而在選點加入子群的部份我們想盡可能降低邊界點間  $X_{i,j}^{b-edge}$  的數量，我們的想法是若讓邊界點的數量降低，那麼就可以使  $X_{i,j}^{b-edge}$  的數量降低，所以我們定了一個值  $r_i$  用來表示點  $i$  與目前正在圈的子集之間的關聯度當選擇某一點進入目前子群時會將與此點之間存在邊的其他點的  $r_i$  值加 1，關聯度越高將此點納入此子群我們就可以使得  $X_{i,j}^{b-edge}$  的數量越少，除此之外在此關聯度  $r_i$  相同的情況下我們再去考慮若將某點選進此子群後此子群對外的 penalty 總和，我們定義此值為  $\beta_i$ ，當  $r_i$  相同同時我們就會選擇  $\beta_i$  較低的點進入目前子群，如圖十四~圖十六在一開始選擇點 0(藍色點)之後會將與其相鄰的點 1、3、4 的  $r$  值加 1，因為這三點  $r$  相同所以去計算其  $\beta$  值，並且選  $\beta$  值最小的點進入這個子群。其例子與 pseudo code 如下。

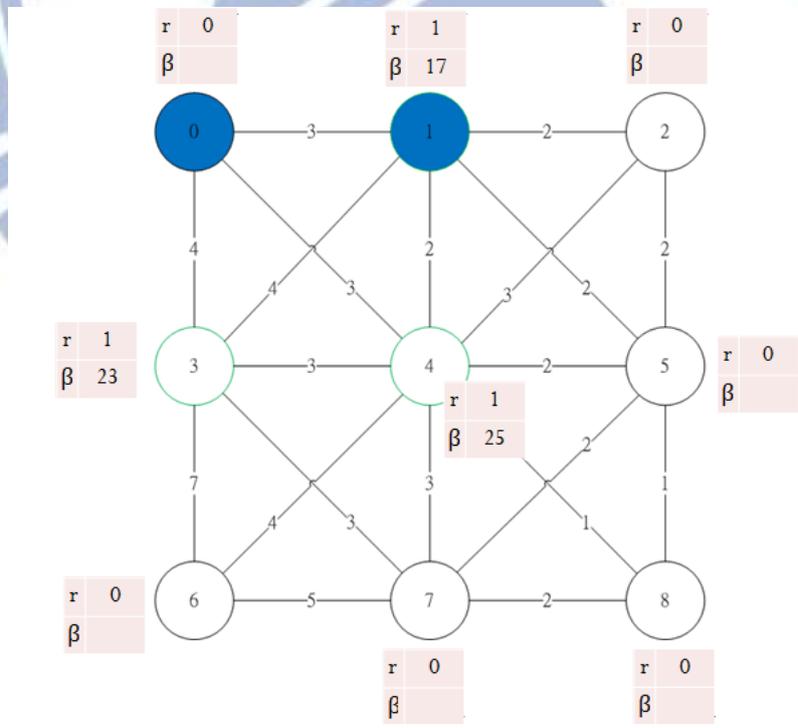


圖十四、啟發式分群演算法選點(起始)



Step1: choose minimum degree node as initial node in this cluster

圖十五、啟發式分群演算法選點(Step1)



Step2: choose minimum  $\beta$  add into this cluster

圖十六、啟發式分群演算法選點(Step2)

**Parameter :**

N : number of node

**Initial :**

C = 0

$r_i = 0 \quad \forall i \in V$

**While(All node be Clustered)**

- Choose Node  $v$  have Min Degree;
- Set the cluster number of  $v = C$
- Mark Boundary Node of this cluster;
- While(this Cluster's loading < Loading Constrain)
  - For(1;N)
    - Choose Boundary Node  $v$  have Max  $\gamma$
    - if equal
      - Choose Node  $v$  have Min  $\beta$
  - Set the cluster number of Node  $v = C$
  - Mark Boundary Node of this cluster;
- C+1
- Recovery Not be selected Boundary Node

### 3.2.1.3 時間複雜度分析

針對我們的最佳化分群演算法的時間複雜度做討論，我們的最佳化演算法為列出所有可能，從其中去找出最佳解，所以我們考慮最差情況下的所有可能數量。

我們定義  $N$  為點的數量，我們考慮在無線感知網路基地台隨機布建的最差情況，即為全部的無線感知網路基地台間皆會互相干擾(彼此間皆有邊存在)，那麼在這種情況下我們可以將問題轉變為將  $N$  個點圈成  $K$  個子群並且允許子群為空，那麼其可能數量即可表示為  $\frac{(N+K-1)!}{(K-1)!}$  又  $N$  個點最多就只能分成  $N$  個子群所以可以將其可能數量轉變為

$$\frac{(N+N-1)!}{(N-1)!} = \frac{(2N-1)!}{(N-1)!} \text{ 種可能，因此最佳化分群演算法的時間複雜度為 } O\left(\frac{(2N-1)!}{(N-1)!}\right)。$$

而啟發式演算法因為每次皆選與當下子群有相連的一個點，與其相連的點必不超過  $N$  個點，並運算其  $\beta$  值，共抓取  $N$  個點因此複雜度為  $O(N^3)$ 。

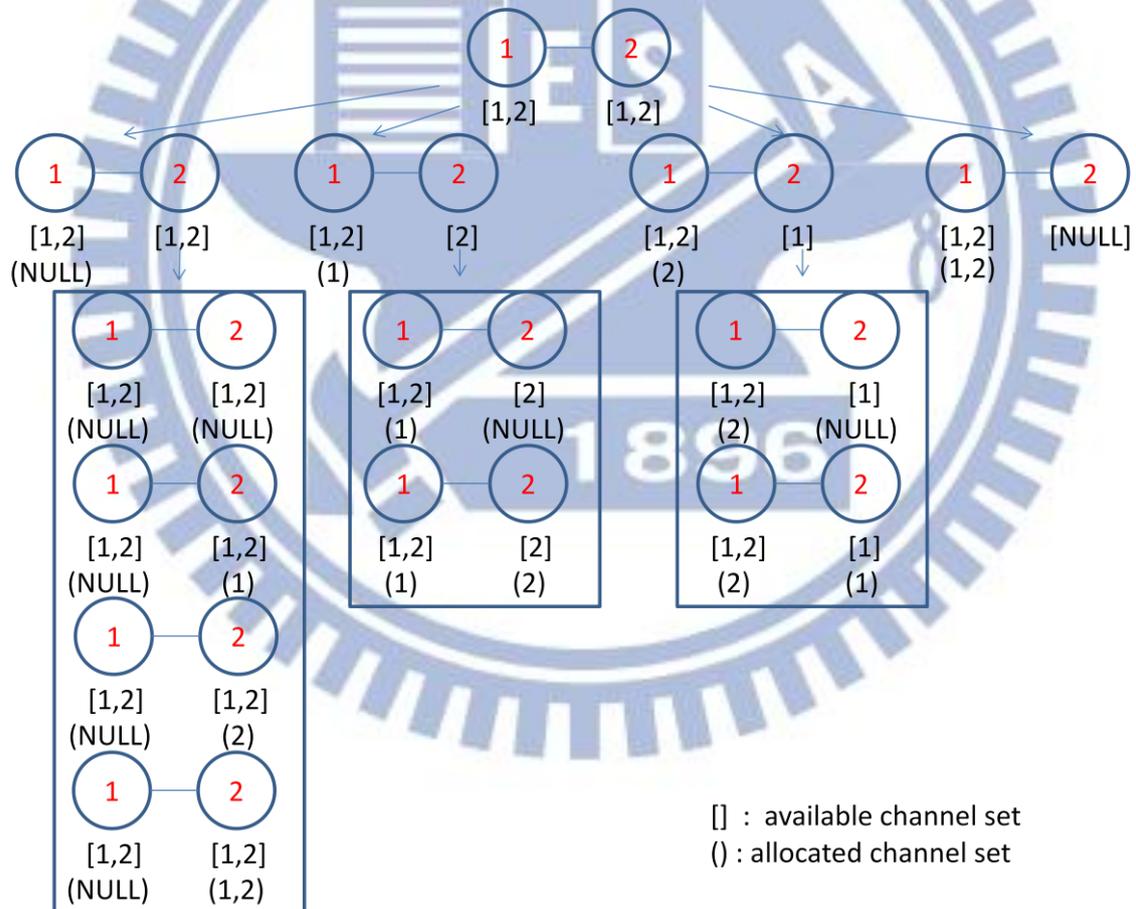
|        |  |
|--------|--|
| 啟發式演算法 | $O(N^3)$ 。                             |
| 最佳化演算法 | $O\left(\frac{(2N-1)!}{(N-1)!}\right)$ |

表二、分群演算法時間複雜度比較

### 3.2.2 邊界點頻譜資源分配

#### 3.2.2.1 最佳化頻譜資源分配介紹

在最佳化頻譜資源分配演算法中，我們考慮其所有頻譜分配的可能，並從其中找出最好的分配方式，我們會針對每一點的每個頻譜去考慮”配置”或”不配置”，如圖十七，我們會先從點 1 開始考慮，考慮點 1 頻譜所有的分配可能，並且限制點 1 的鄰居不可以配與相同的頻譜，接著再針對每一種可能分別去考慮點 2 可以分配的可能詳細的實例以及 pseudo code 如下。



圖十七、最佳化頻譜資源分配實例。

**Parameter :**

N : total number of AP

temp solution : temp total allocated channel number for optimal

**Initial:**

temp solution = -1

**Main()**

Coloring\_OPT(Node 0)

**Coloring\_OPT(Node  $i$ )**

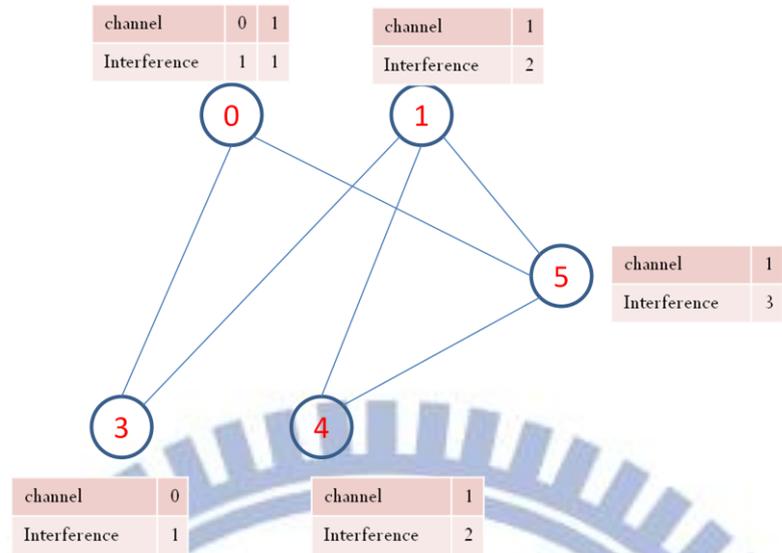
- If(Node  $i <$  total number of nodes)
  - Find the power set S of Node  $i$   
 $S = \{ S_1, S_2, \dots \}$
  - For each set  $S_i$ 
    - Record the all Node's channel
    - Set  $S_i$  be allocated
    - Disable  $S_i \cap neighbor's\ channel$
    - Coloring\_OPT( $i+1$ )
    - Recovery the *neighbor's channel*
- Else
  - If(this solution better than temp solution)
    - temp solution = this solution

### 3.2.2.2 啟發式邊界點頻譜資源分配介紹

我們對於我們的邊界點頻譜資源分配設計了一啟發式邊界點頻譜資源分配，我們會根據三個數值來決定要將頻譜配置給哪一個點，第一為干擾值(interference)，干擾值表示使用相同頻譜的鄰居數量，如圖十八點 3 的鄰居為點 0 與點 2，而這兩點皆可以使用頻譜 0，所以對於點 3 頻譜 0 的干擾值即為 2，我們一開始會選擇最小頻譜干擾值的點去配置頻譜給他。

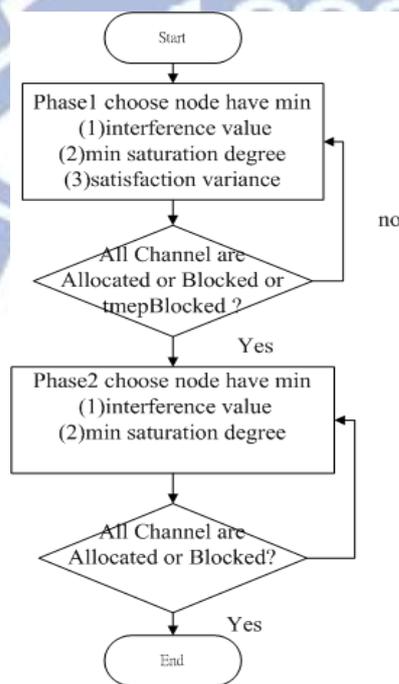


而當干擾值相同的情況我們會去考慮剩餘頻譜數(saturation degree)，如圖十九點 0 與點 3 的頻譜 0 干擾值皆為 1 此情況之下，就會選擇剩餘頻譜數較少的點 3 進行分配，我們希望能夠讓頻譜資源快沒的點能夠先拿到頻譜但又不會導致過多的鄰居不能使用同一頻譜。



圖十九、餘頻譜數實例

而當剩餘頻譜數也相同時我們將去考慮公平性，我們定義一公平值(satisfaction ratio)為無線感知網路基地台被分配的頻譜資源數去除以無線感知網路基地台底下使用者對於頻譜之需求數的變異數，我們考慮若將某頻譜資源分給某點能使此公平值最低則分配出此頻譜，最後若上述三個條件皆相同，我們則會隨機選一點的頻譜資源分配。



圖二十、邊界點頻譜資源分配流程圖

我們會將上述的分配頻譜分配方法做兩個階段圖二十，在第一個階段我們會依據每點上的需求數做為其分配資源的上限，若其被分配的頻譜資源已經與其所提出需求數相同那麼在第一個階段我們將不會再分配其他資源給此點，第一階段會在可不超出需求數之前提下可分配頻譜資源皆被分配出去後結束需求數被滿足的點會先將其先暫時 block，接著第二階段會將暫時被 block 的頻譜恢復並將這些剩餘的頻譜資源依據干擾值與剩餘頻譜數兩個來分配第一階段結束後剩下的頻譜資源，會這樣做的原因在於我們希望將頻譜資源優先分配給有需求的點，接著再將有剩餘的頻譜資源盡量分配出去，下面為我們啟發式邊界點頻譜資源分配的 pseudo code。

**Parameter :**

$R(v)$ :number of request in node  $v$

**Main:**

Phase1 algorithm

Recovery tmpBlock Node

Phase2 algorithm

### Phase1 algorithm

While(can allocate channel)

- Choose Node  $v$  have min interference value
- if(min interference equal)
  - Choose Node  $v$  have min saturation degree
  - if(min saturation degree equal)
    - Choose Node  $v$  cause min satisfaction ratio variance
    - if(min satisfaction ratio variance equal)
      - Random choose Node  $v$
- AllocateResource Phase1 (Node  $v$ )

AllocateResourcePhase1(Node  $v$ )

- Choose available channel CH of  $v$  which has min interference
- Allocate CH to Node  $v$
- If( $R(v)$ ==number of allocated channel of  $v$ )
  - tempBlock Node  $v$
- For(all Node) // Block neighbor's CH
- if(Node  $u$  are Node  $v$  neighbor)
  - block Node  $u$  CH

### Phase2 algorithm

While(can allocate channel)

- Choose Node  $v$  have min interference value
- if(min interference equal)
  - Choose Node  $v$  have min saturation degree
  - if(min saturation degree equal)
    - Random choose Node  $v$
- AllocateResource Phase2 (Node  $v$ )

AllocateResourcePhase2(Node  $v$ )

- Choose available channel CH of  $v$  which has min interference
- Allocate CH to Node  $v$
- For(all Node) // Block neighbor's CH
- if(Node  $u$  are Node  $v$  neighbor)
  - block Node  $u$  CH

### 3.2.2.3 時間複雜度分析

對於我們的邊界頻譜資源分配最佳化演算法，因為我們考慮其所有可能性，會找出所有可能的頻譜資源分配方法，所以我們以下列方式來找出時間複雜度。首先假設我們有  $N$  個點對於每點的可用頻譜數為  $C_1, C_2 \dots C_N$ ，對於每個點我們都考慮可用頻譜的所有可能性也就是其 power set 對於點 1 來說其可能性即有  $2^{C_1}$  種可能，因此時間複雜度將不大於  $2^{C_1} * 2^{C_2} * \dots * 2^{C_N} = 2^{\sum_{i=1}^N C_i}$  因此複雜度為： $O(2^{\sum_{i=1}^N C_i})$ 。

而啟發式演算法因為我們每次選擇一個點的某一條頻譜資源作分配，最鑿情況則必須挑選  $(\sum_{i=1}^N C_i) * N$  次，即代表對於每個點的每條頻譜資源都要挑選，因此複雜度為  $O((\sum_{i=1}^N C_i) * N)$ 。

|        |                             |
|--------|-----------------------------|
| 啟發式演算法 | $O((\sum_{i=1}^N C_i) * N)$ |
| 最佳化演算法 | $O(2^{\sum_{i=1}^N C_i})$   |

表三、頻譜資源分配演算法時間複雜度比較

## 第四章 模擬數據結果與分析

在這一章節中我們將分成兩個部份呈現我們的模擬，在第一部分會模擬分群演算法與邊界點頻譜資源分配演算法的最佳化演算法與啟發式演算法的比較，在第二部分模擬我們會對於一大尺度環境針對台北市做模擬，並使用其中的一個區域對我們的系統做測試。

### 4.1 演算法比較

我們假設環境中每個無線感知網路基地台皆有 4 條頻譜資源可以使用，而每個無線網路基地台底下的需求數隨機介於 1~4 個，表四為我們考慮 9 個點時方形布建環境分別執行 100 次取平均所得到的結果，而各個數據所代表意義如下。

1. VM#，分群演算法所分出的子群數量一個子群配給一台虛擬機器。
2. Handled loading(mean, standard deviation)，不同分群法中虛擬機器所處理的平均工作負載量及其標準差。
3. Total Penalty，不同子群間的 penalty 值。
4. Avg. Penalty，平均每個子群占多少 penalty。
5. Avg. Allocated Channel(Boundary AP)，對於邊界點執行完演算法後平均每個邊界點被分配到的頻譜資源數量。

6. Fairness index，定義為 
$$\frac{(\sum_{i=1}^B \min(\frac{|M_i|}{r_i}, 1))^2}{B \sum_{i=1}^B (\min(\frac{|M_i|}{r_i}, 1))^2}$$

| AP:9                                      |                   |  |                  |                 |  |                       |                |                       |
|---|-------------------|--|------------------|-----------------|--|-----------------------|----------------|-----------------------|
| Max VM loading is 50(loading constrain) , |                   |  |                  |                 |  |                       |                |                       |
|   | VM#<br>(cluster#) | Handled<br>loading(mean,<br>standard<br>deviation) | Total<br>Penalty | Avg.<br>Penalty | Avg. Allocated<br>Channel<br>(Boundary AP) |                       | Fairness index |                       |
|   |                   |  |                  |                 | OPT  | Heuristic<br>coloring | OPT            | Heuristic<br>coloring |
| OPT                                       | 2.52              | (32.83 , 5.999)                                    | 11.15            | 4.42            | 2.41                                       | 2.278                 | 0.784          | 0.935                 |
| Heuristic<br>clustering                   | 2.23              | (33.21 ,<br>12.578)                                | 12.74            | 5.71            | 2.36                                       | 2.233                 | 0.666          | 0.907                 |

| AP:16                                     |                   |  |                  |                 |  |                       |                |                       |
|---|-------------------|--|------------------|-----------------|--|-----------------------|----------------|-----------------------|
| Max VM loading is 50(loading constrain) , |                   |  |                  |                 |  |                       |                |                       |
|   | VM#<br>(cluster#) | Handled<br>loading(mean,<br>standard<br>deviation) | Total<br>Penalty | Avg.<br>Penalty | Avg. Allocated<br>Channel<br>(Boundary AP) |                       | Fairness index |                       |
|   |                   |  |                  |                 | OPT  | Heuristic<br>coloring | OPT            | Heuristic<br>coloring |
| OPT                                       | 3.48              | (41.28 ,<br>5.2426)                                | 16.67            | 4.41            | 1.89                                       | 1.712                 | 0.608          | 0.796                 |
| Heuristic<br>clustering                   | 3.94              | (33.17 ,<br>13.106)                                | 23.51            | 5.96            | 1.73                                       | 1.644                 | 0.583          | 0.741                 |

表四、最佳化與啟發式演算法比較

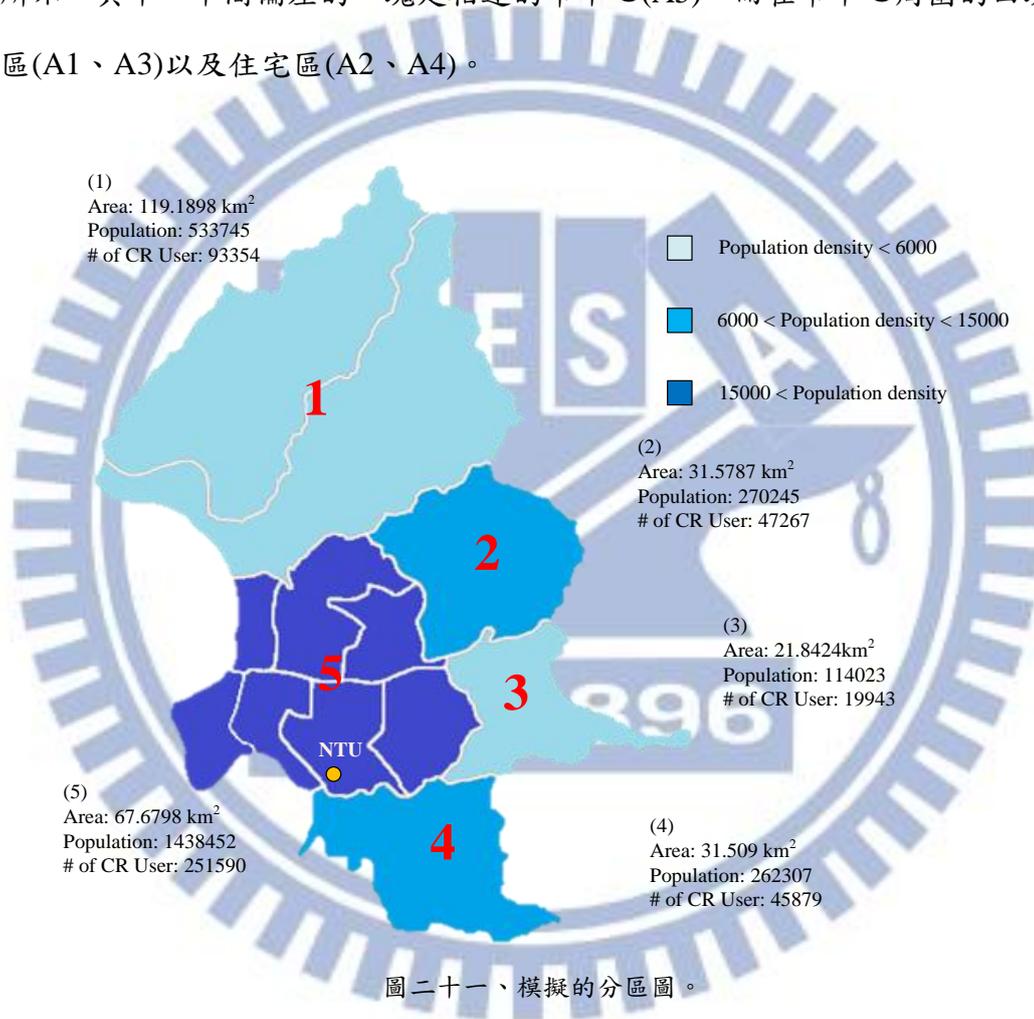
由表四可以看出，在分群演算法中最佳化演算法在 9 個點以及 16 個點時總 penalty 與平均 penalty 都為最小，優於啟發式演算法，但考慮到實作層面最佳化演算法的複雜度太高執行時間太長，無法用來實做在大尺度的環境當中。

在邊界點頻譜資源分配的比較中，可以看到最佳化邊界點資源分配的方法在針對邊界點的資源分配多於啟發式演算法，但是在公平性方面，就比啟發式演算法來的差，這是因為，我們在啟發式演算法中考慮了兩階段的頻譜分配，在第一個階段對於需求滿足的點我們就暫時先不將資源配置給他，所以雖然分出去的頻譜資源較少，但是容易發生頻譜資源都集中在某一點的情況。

## 4.2 大尺度系統測試

### 4.2.1 參數及環境設定

我們考慮台北市的行政區劃分以及人口密度，我們將台北市切割成 5 個區域，如圖十八所示，其中，中間偏左的一塊是相連的市中心(A5)，而在市中心周圍的四塊則為鄰近的郊區(A1、A3)以及住宅區(A2、A4)。



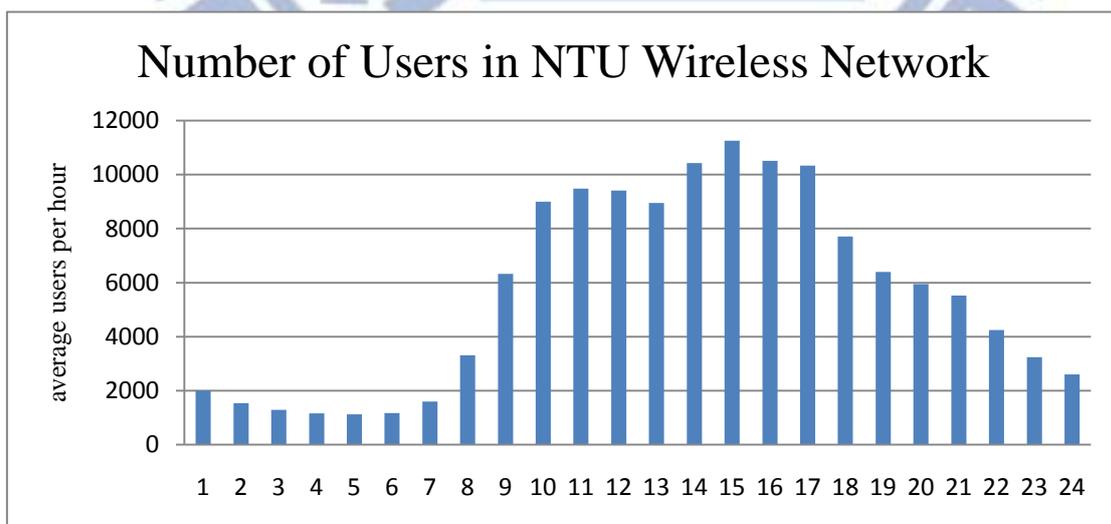
由於在不同的區域內的人口密度並不相同，所產生的資料總數也不相同。我們在不同區域內依據使用者密度布建不同個數的無線感知網路基地台，並考慮空間上的重複利用性，也就是說，在不同樓層布建無線感知網路基地台。並假設在不同高度的無線感知網路基地台不會互相干擾。在市中心的環境中，由於 MU 的密度較高，在我們的模擬中設定在市中心此重複利用的係數為 4。換句話說，在此地區內，每個無線感知網路基地

台所鏈結的 MU 的數目為原有的四分之一。在住宅區的環境中，設定在市中心此重複利用的係數為 2。經過此設計後，在 busy hour，每個感知網路基地台所要服務的使用者個數將落在 42 至 56 的區間(如表五)，大致平衡每一個感知網路基地台的負載。我們將 A1 至 A5 依據其面積產生正方形區域，按照相對地理位置相鄰排列，並在每平方公里內建立 16 個無線感知網路基地台，隨機的決定無線感知網路基地台的位置。

|    | Area     | # of APs | # of MU/AP |
|----|----------|----------|------------|
| A1 | 119.1898 | 1908     | 46.47      |
| A2 | 31.5787  | 506*2    | 44.46*2    |
| A3 | 21.8424  | 350      | 54.07      |
| A4 | 31.509   | 505*2    | 42.57*2    |
| A5 | 67.6798  | 1083*4   | 56.0775*4  |

表五、各地區模擬數值。

台北市的面積一共為 271.8 平方公里，平均人口密度為 9635(人/平方公里)，根據台北市政府主計處[12]的統計，61.2%的人口曾使用行動上網，而每日平均上網時間為 216 分鐘。為了考慮在一天中不同時刻上網人口的分布，我們參考台灣大學計算機及網路資源中心[13]的統計資料，並觀察連續四天的同時上網人數，繪製出圖二十二。



圖二十二、在一天中不同時刻的同時上網人數

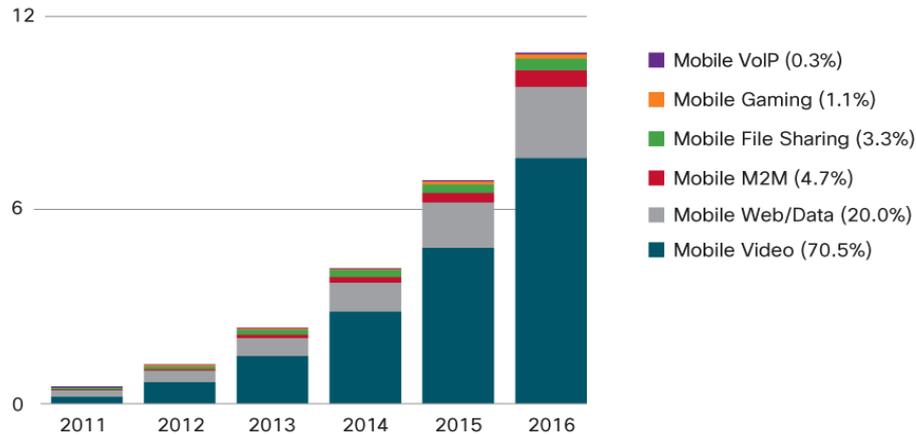
在圖二十二中，我們可以觀察到，在台大的校園中，下午三點為使用人數最多的時刻，早上5點則較少使用。在下述的案例分析中，將舉下午三點為尖峰時刻的範例，而早上五點為離峰時刻的範例。根據台大校園的統計資訊，以及台北市主計處的資訊，我們可以知道在一日中各時段的行動人口密度，顯示如下表六。

|     | 土地面積     | 人口      | MUs(離峰)     | MUs(尖峰)     |
|-----|----------|---------|-------------|-------------|
| 台北市 | 271.7997 | 2618772 | 45588.34914 | 458032.0375 |
| 松山區 | 9.2878   | 208434  | 3628.480053 | 36455.80818 |
| 信義區 | 11.2077  | 225092  | 3918.467391 | 39369.34845 |
| 大安區 | 11.3614  | 311565  | 5423.814674 | 54493.76722 |
| 中山區 | 13.6821  | 220126  | 3832.01781  | 38500.77834 |
| 中正區 | 7.6071   | 159536  | 2777.249363 | 27903.38339 |
| 大同區 | 5.6815   | 124600  | 2169.073254 | 21792.9594  |
| 萬華區 | 8.8522   | 189099  | 3291.890716 | 33074.0516  |
| 文山區 | 31.509   | 262307  | 4566.316999 | 45878.37722 |
| 南港區 | 21.8424  | 114023  | 1984.945743 | 19943.0065  |
| 內湖區 | 31.5787  | 270245  | 4704.504025 | 47266.76014 |
| 士林區 | 62.3682  | 284539  | 4953.338159 | 49766.82885 |
| 北投區 | 56.8216  | 249206  | 4338.250958 | 43586.96822 |

表六、台北市的移動通訊使用者(Mobile Users, MUs)數量估測。

在模擬中，我們根據該地區的人口統計資訊，乘上上網比例(0.612)，乘上每人使用的時間(3.416 hour/day)，所得到的數值(user\*hour/day)依據不同時段的上網人口所佔總體人口的比例(peak hour: 0.836, off-peak hour: 0.0832)，計算出該地區的 MU 數量。

我們先根據通訊服務的即時性以及其需求的資料量分成四類，並依據Cisco NPI所估測2016年的網路流量分析[14](圖二十三)作為我們估測服務的依據，並將估測的結果放在表七。在進行分類的過程中，Mobile Web/Data的類別難以分配到我們定義的類別，所以把一半的資料流量分配至Data Transmission，一半分配到Web Browsing。



Figures in legend refer to traffic share in 2016.  
Source: Cisco VNI Mobile, 2012

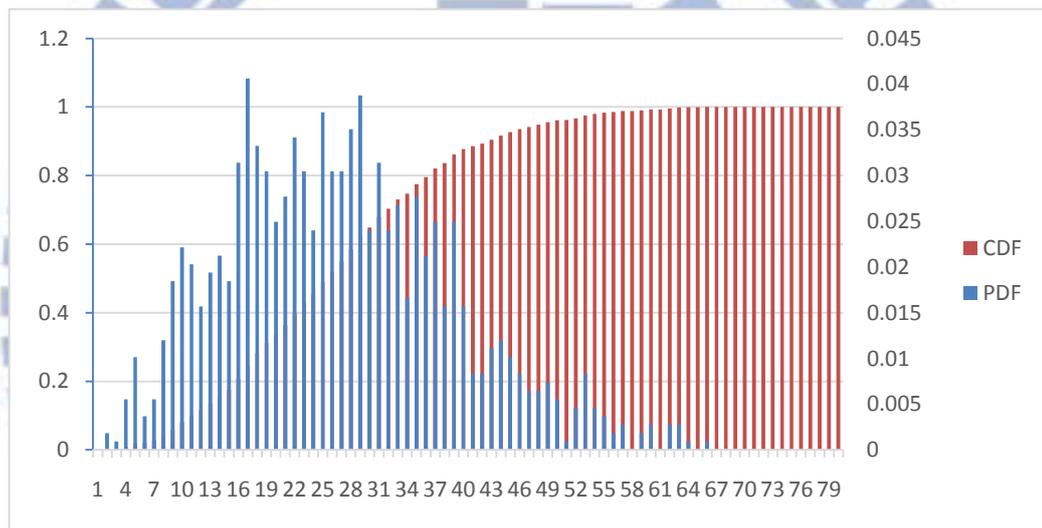
圖二十三、 Cisco NPI

我們依據每一區的使用者(MU)數量，隨機了決定所服務的使用者位置，使用者在產生後將加入距離最近的無線感知網路基地台，並根據所要求的傳送速率以及所在的 Sector。對於每一個無線感知網路基地台，將得到所服務的 MU 以及其需求，並在每一個 sector 中計算該 sector 中所有 MU 所需求的下載傳送資料量(request rate \* number of request\* 0.4s, 0.4s 為一個 subframe 的時間)，將上述資料量分配到實體通道(physical channel or SU, {3,9,18,27}Mbps)，其中，由於 MAC 層的設計，每個實體通道，都只有 150ms 作為下載傳輸，因此每一實體通道在一個 subframe 中，能負擔的資料量為{0.45, 1.35, 2.7, 4.05}Mb。

| Defined service                    | Web Application            | % of data | Required rate | % of MUs |
|------------------------------------|----------------------------|-----------|---------------|----------|
| Video Streaming (RT, High Rate)    | Video                      | 69.1%     | 4500Kbps      | 34.07%   |
| Data Transmission (nRT, High Rate) | Data, File Sharing         | 13.1%     | 2000Kbps      | 14.53%   |
| Web Browsing (nRT, Low Rate)       | Audio Service, M2M, Gaming | 15.5%     | 1000Kbps      | 34.39%   |
| VoIP、RT Service (RT, Low Rate)     | VoIP<br>Voice Call         | 2.3%      | 300Kbps       | 17.01%   |

表七、各類 MU 使用的資料流量以及需求速率

我們將原有的服務速率對應道實體通道，先計算該服務(MU)所產生的資料量加上來自於更高速服務所剩餘的需求，除以對應的實體通道速率，四捨五入後得到對於該實體通道(SU)的需求。圖(二十一)為所每個無線感知網路基地台所要服務的需求(SU)數量，根據此統計數量，我們為了使百分之八十的感知網路基地台獲得足夠的資源，我們假設每一個無線感知網路基地台擁有一 Control Channel，並可以管理 36 個不同的 Data Channel(6 layers, 6 channels per layer)。考量頻譜資源分享和頻道配置與功率配置演算法的複雜度，我們以 6 條 Data Channels 作為頻譜資源分享和頻道配置與功率配置的運算單位，稱作為一個 layer，獨立進行演算法。



圖二十四、每一個 AP 的實體通道(SU)需求數量

若是按照我們的計算，在不計入 Control channel 所佔用的時間以及無線感知網路基地台間的干擾，並假設所有分配的速率皆為最高的通道下，則此通訊系統在台北市建立 8612 個無線感知網路基地台，最多能夠提供 8367Gpbs $((4351+2093+1082*2)*36*27\text{Mbps})$  的通道容量。

## 4.2.2 大規模模擬結果

根據頻譜資源分享結果以及對於行動資訊控管元件承載測試做為預估表八，在尖峰時段一共需要 1821 台虛擬機器，而離峰時段僅需要 340 台虛擬機器其中頻譜資源分享元件是依據我們在台北市對於空間上布了 4 層的無線感知網路基地台每層各使用一個頻譜資源分享元件。在另一方面對於每個 layer 的無線感知網路基地台各別有 6 條頻譜在經過頻譜資源分享之後在尖峰以及離峰分別可以分到 3.41 與 3.84 條頻譜資源，因為對於子群邊界上的無線感知網路基地台之間所分配出去的頻譜集合必為互斥，如此一來對於邊界上所分配的資源會較少。

接著考慮到我們所實作的系統與現存系統的比較，我們以 HSPA(High Speed Packet Access)作為我們的比較系統，對於 HSPA 的系統設定，我們參考了 Nokia Siemens 的白皮書” Mobile broadband with HSPA and LTE – capacity and cost aspects”。根據白皮書中的設定與試算，對於一個 HSPA 的通訊系統，一個基地台有三個 sector，每個 sector 擁有三根天線，每根天線服務 5MHz 的頻寬，並能夠提供 6MHz 的 throughput。同時，我們根據 UMTS 的資料，知道在 3G (WCDMA)的系統架構下，一個基地台的覆蓋範圍約為 2.3 平方公里。

為了評估兩個系統所消耗的功率，我們以 Nokia Siemens 的 Universal Base Station (UBS)作為評估的依據，在無線感知網路基地台的消耗功率上，我們使用現有室外型的 wifi 基地台作為無線感知網路基地台基礎功率的參考基準，並加上在雲端 CPU 運算所需求的功率，估測整體功率消耗。詳細的數據以及比較列在表九。

對於整個台北市而言，一共有 170MHz 的頻譜資源用於 3G 網路系統，也就是說可以提供 452,880Mbps 的資料量，頻譜的使用效率為 2.664。而我們所提出的系統，則可以提供最多 758,754Mbps 的傳輸量，頻譜使用效率為 3.512，約為 3.5G 系統的 1.5 倍。

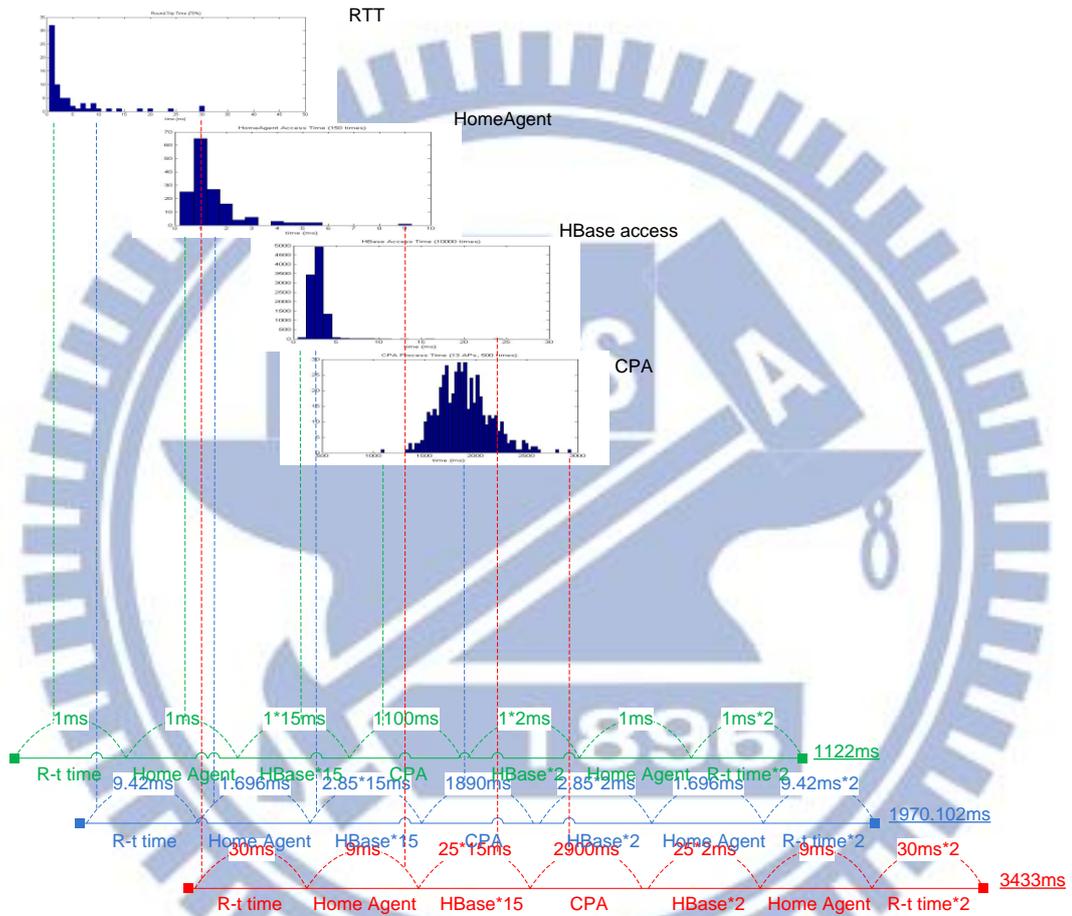
|                                      | Peak | Off-Peak |
|--------------------------------------|------|----------|
| <b># of Channel per AP after SRS</b> | 3.41 | 3.84     |
| <b>#HomaAgent</b>                    | 22   | 22       |
| <b>#HregionServer</b>                | 16   | 16       |
| <b>#SRS</b>                          | 4    | 4        |
| <b>#CPA</b>                          | 2091 | 383      |

表八、台北市頻譜資源分享結果以及所需虛擬機器數

|                                     | 3.5 G (HSPA)                   | CRCN                         |
|-------------------------------------|--------------------------------|------------------------------|
| <b># of AP(BS) in Taipei City</b>   | 148 BTSs                       | 8612 CR APs                  |
| <b># of VM</b>                      | 0                              | 2133/425 (Peak/Off)          |
| <b>Channel bandwidth</b>            | 5 MHz                          | 6 MHz                        |
| <b>Used spectrum bandwidth</b>      | 15 MHz per BTS                 | 180 MHz per AP (max)         |
| <b>throughput (Peak time)</b>       | 8 Gbps (15 MHz)                | 758.8 Gbps (216 MHz)         |
| <b>throughput / BTS (Peak time)</b> | 54 Mbps<br>(6*3*3Mbps)         | 88.1 Mbps                    |
| <b>Max Bit/s</b>                    | 21.1 Mbps (R.7)<br>4.22 bps/Hz | 27 Mbps<br>4.5 bps/Hz        |
| <b>Transmission power</b>           | 180 (W)                        | 0.97 (27mW*36) (W)           |
| <b>Coverage range</b>               | 2.3 (Km <sup>2</sup> )         | 0.12 (Km <sup>2</sup> )      |
| <b>Computing power</b>              | 6310W per BTS                  | 10W per core<br>28.5W per AP |

表九、和 3.5G 系統的比較

最後我們在交通大學的 OpenStack 架設我們的系統並且對於我們系統的整體時間做一個測試，圖二十四，對於頻譜資源分想因為為長時間才做一次，所以不列入測量，由圖中可以看到在平均時間上我們可以在 2 秒內將資源分出去給無線感知網路基地台，但在最糟情況下，必須要到 3 秒多才可以完成資源的分配。



圖二十五、雲端整體系統執行時間

## 結論

在這篇論文中，我們介紹我們的無線感知雲端網路的架構，並且為了可以服務大量的使用者，對其各個元件討論擴充的規則，像是對於行動資訊控管元件與頻道配置與功率配置的擴充，對於行動資訊控管元件我們以模擬測試的方式去設定擴充條件，而頻道配置與功率配置的擴充我們提出分群演算法，將大區域切割成多個子群並且提出邊界點頻譜分配，對於邊界點先做一初步的頻譜資源分配，讓頻道配置與功率配置可以平行化的運算，希望能藉由將各個元件擴充可以加速整體在雲端上的運作時間，以追求即時性的通訊服務，我們也對 HBase 中我們的表格做設計，呈現出在雲端中我們對於資料的流向，希望能藉由這樣的設計降低因為對於資料庫存取所造成的時間消耗。

最後我們以模擬的方式，對我們所提出的分群與邊界點頻譜資源分配的最佳化演算法與啟發式演算法來做比較，並且模擬台北市與現有 3.5G 系統做比較，並且取台北市中的一個區域來實際模擬我們的系統，從無線感知網路基地台發送訊息至行動資訊控管元件存入 HBase 再由各個工作元件取出資訊做運算，再將結果送回無線感知網路基地台做排程藉此來驗證我們系統的可行性，並且呈現出我們的整體系統時間。

## 參考文獻

- [1] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017, February 6, 2013
- [2] Federal Communications Commission Spectrum Policy Task Force, ET Docket No. 02-155, Nov. 2002.
- [3] J. Mitola and G. Q. Maguire, “Cognitive Radio : Making Software Radios More Personal.” IEEE Personal Communication Magazine, Vol. 6, pp. 13-18, Aug. 1999.
- [4] Sau-Hsuan Wu, Hsi-Lu Chao, Chun-Hsien Ko, Shang-Ru Mo, Chiau-Feng Liang, and Chung-Chieh Cheng, “Green Spectrum Sharing in a Cloud-Based Cognitive Radio Access Network,” IEEE GreenCom 2013
- [5] Sau-Hsuan Wu, Hsi-Lu Chao, Chung-Ting Jiang, Shang-Ru Mo, Chun-Hsien Ko, and Tzung-Lin Li, “A Conceptual Model and Prototype of Cognitive Radio Cloud Networks in TV White Space,” IEEE WCNC 2012, Workshop Wireless Cloud and White Space Oriented Networks.
- [6] E. Rolland, H. Pirkul, and F. Glover, “Tabu Search for GraphPartitioning,” Annals of Operational Research, vol. 63, pp. 209-232, 1996
- [7] R. Battiti and A. Bertossi, “Greedy, Prohibition and Reactive Heuristics for Graph Partitioning,” IEEE Trans. Computers, vol. 48, pp. 361-385, 1999.
- [8] Daniel Marx, “Graph coloring with local and actions global constraints,” PhD thesis, Department of Economics, Budapest University of Technology, 2004
- [9] Jian Tang, Roberto Hincapié, Guoliang Xue, Senior Member, Weiyi Zhang, and Roberto Bustamante “Fair Bandwidth Allocation in Wireless Mesh Networks With Cognitive Radios” IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 59, NO. 3, MARCH 2010
- [10] Brian F. Cooper, Adam Silberstein, Erwin Tam, Raghu Ramakrishnan, Russell Sears, “Benchmarking Cloud Serving Systems with YCSB.” Yahoo! Research

- [11] Lizhi Cai, Shidong Huang, Leilei Chen, Yang Zheng, "Performance Analysis and Testing of Hbase Based on Its Architecture." Computer and Information Science (ICIS), 2013 IEEE/ACIS 12th International Conference on, pp. 353 – 358
- [12] Ravi Kumar and Andrew Tomkins, "A Characterization of Online Browsing Behavior," WWW '10 Proceedings of the 19th international conference on World wide web, Pages 561-570.
- [13] Saikat Guha and Neil Daswani , "An Experimental Study of the Skype Peer-to-Peer VoIP System," IPTPS'06 - Proceedings of The 5th International Workshop on Peer-to-Peer Systems, 2006
- [14] Cisco Visual Networking Index, "Global Mobile Data Traffic Forecast Update, 2011–2016" [Online]. Available: [http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white\\_paper\\_c11-520862.html](http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html)
- [15] Ko, Chun-hsien; Din Hwa Huang; Wu, Sau-hsuan, "Cooperative spectrum sensing in TV White Spaces: When Cognitive Radio meets Cloud," *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on* , vol., no., pp.672,677, 10-15 April 2011