

國立交通大學

資訊科學與工程研究所

碩 士 論 文



研 究 生：莫尚儒

指 導 教 授：趙禧綠 教授

中 華 民 國 101 年 9 月

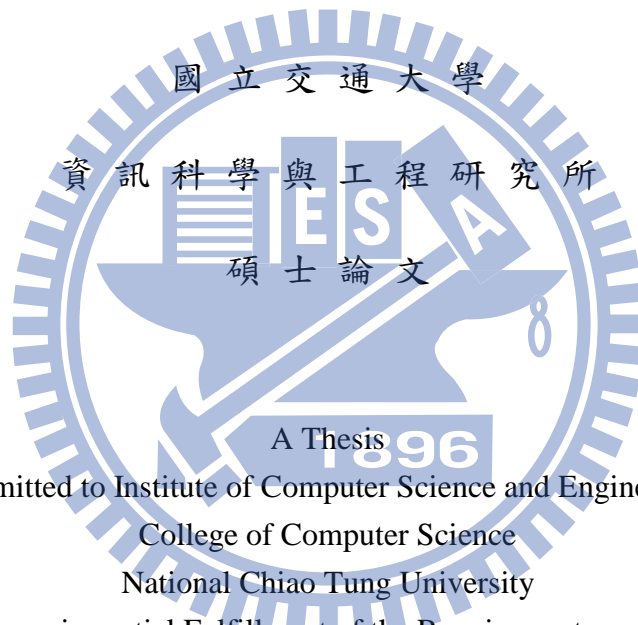
無線感知雲端網路資源分配平行化與可擴性之研究
The Study of Parallelization and Scalability of Resource
Allocation in Cognitive Radio Cloud Networks

研究生：莫尚儒

Student : Shang-Ru Mo

指導教授：趙禧綠

Advisor : Hsi-Lu Chao



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

September 2012

Hsinchu, Taiwan, Republic of China

中華民國 101 年 9 月

中文摘要

在資訊爆炸的年代，隨著網路中資料量不斷的增加，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。

近年來隨著無線感知技術的崛起，各式各樣的無線感知網路架構被提出包含了分散式無線感知網路、集中式無線感知網路，許多架構甚至採用了雲端來滿足大量的運算需求。

在這篇論文中，我們著重於大尺度的頻譜資源分配問題，為了達到排程的時間限制需求，我們將問題區分為兩個子問題：分群問題與頻譜資源分配問題，針對兩個不同的問題我們分別提出最佳化演算法以及時間複雜度較低的雲端可擴演算法，分群問題的目標為，在給定伺服器負載的限制下對無線感知網路基地台做分群，每一群交由一台伺服器處理。另一方面，資源分配問題則是著重在邊界無線感知網路基地台的資源分配，期望對頻譜做最佳化的利用。

在這篇論文中，我們不單單闡述問題並且設計解決問題的方式，更進一步的我們提出一些理論及定理，並且為所設計的方法分析其時間複雜度，最後，經由模擬結果以實際的數據分析並驗證其效能及可能性。

ABSTRACT

With the network in the era of information explosion, the amount of data size continuously increases and thus gradually causes the insufficiency of bandwidth resources. Cognitive Radio technique has been proposed as a promising solution to improve the spectrum utilization efficiency in wireless communications.

There have been lots of researches and development in this area over the past decade. Specifically, several cognitive radio network architectures are proposed and they could be classified into distributed cognitive radio networks or centralized cognitive radio networks. Some architecture even utilizes cloud to support the massive computing demands.

In this thesis, we tackle the problem of large-scale resource allocation. To meet the requirement of scheduling response time, this problem is divided into two subproblems: clustering and resource allocation. We first formulate the problem and develop an approach to have the optimal solution. Further, we propose a low-complexity scalable cloud-based resource allocation algorithm. The objective of clustering is, based on the predefined computation load constraint of each virtual machine (VM), to partition all access points (APs) into several clusters, and thus each cluster is served by a VM. On the other hand, the designed channel assignment algorithm is only performed for all boundary APs to achieve maximum channel reusability.

We not only formulate the problem and design feasible mechanisms, but also propose some lemmas and theorems, and analyze the time complexity for all proposed methods. Through simulations we validate and evaluate our methods and show the effectiveness and feasibility of our approaches.

誌 謝

在攻讀碩士的路上，我體驗了各種不同的感受

在面對問題時的艱鉅苦澀、緊張徬徨

在解決問題時的共同辯論、深入探討

在實現成果時的互助合作、交互溝通

在展現成果時的無比踏實、信心驕傲

在攻讀碩士的生涯中，所獲得的經歷及收穫無法用言語比擬

如今，我已到了能夠收拾成果的一刻，我感到十分感動及滿足。

首先，感謝在這一路上默默支持我的父母，沒有父母的支持，我不可能完成碩士學業，謝謝爸媽的鼓勵，讓我在碩士這條路上能屹立不搖的走下去。

第二，感謝我的指導老師趙老師以及伍紹勳老師，謝謝老師在碩士生涯中的諄諄教誨，謝謝老師的磨練，謝謝老師的各種容忍耐心與體諒，教導了我在面對任何事物，皆要有著實事求是、堅持到底的精神，我終生受用。

第三，謝謝我的朋友及戰友們，Bibi、冠勳、奕霖、彥翰、皓鈞、寰融、冠廷、Hank、阿愷、趴特、青菜、喬峰、佳賢、等等等等等...，有了你們一路上的陪伴，讓我在面對困難時能宣洩、面對喜悅時能分享，謝謝你們。

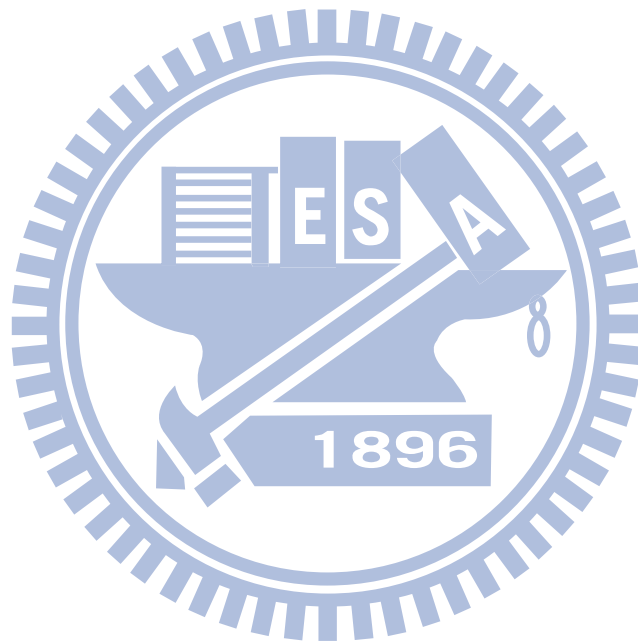
最後，感謝上天，讓我能有這個機會經歷各種試驗，無論是酸、甜、苦、澀，都讓尚儒的生命中更豐富、更美好。

目 錄

中文摘要	I
英文摘要	II
誌謝	III
目錄	IV
表目錄	V
圖目錄	VI
第一章	前言.....	1
1.1	無線感知網路(Cognitive radio networks).....	1
1.1.1	頻譜感測(Spectrum sensing).....	3
1.1.2	頻譜管理(Spectrum management).....	3
1.2	無線感知雲端網路(Cognitive radio cloud networks).....	3
1.2.1	無線感知雲端網路架構.....	6
1.2.2	元件運作流程介紹.....	9
1.2.3	統整與問題動機.....	10
第二章	系統環境介紹與問題定義描述.....	14
2.1	分群問題.....	14
2.2	邊界點的頻譜資源分配問題.....	16
第三章	提出方法與時間複雜度分析.....	18
3.1	分群.....	18
3.1.1	最佳化分群法介紹.....	18
3.1.2	啟發式分群法介紹.....	24
3.1.3	時間複雜度分析.....	30
3.2	邊界點的頻譜資源分配.....	31
3.2.1	最佳化頻譜資源分配法介紹.....	31
3.2.2	啟發式頻譜資源分配法介紹.....	34
3.2.3	時間複雜度分析.....	38
第四章	模擬數據結果與分析.....	39
第五章	結論與未來發展目標.....	44
參考文獻	45
附錄一	工作量產生方式>Loading Generation).....	47

表目錄

表一	分群法時間複雜度比較.....	31
表二	資源分配演算法時間複雜度分析.....	38
表三	含最佳化的比較數據一覽.....	40
表四	不含最佳化的比較數據一覽.....	43



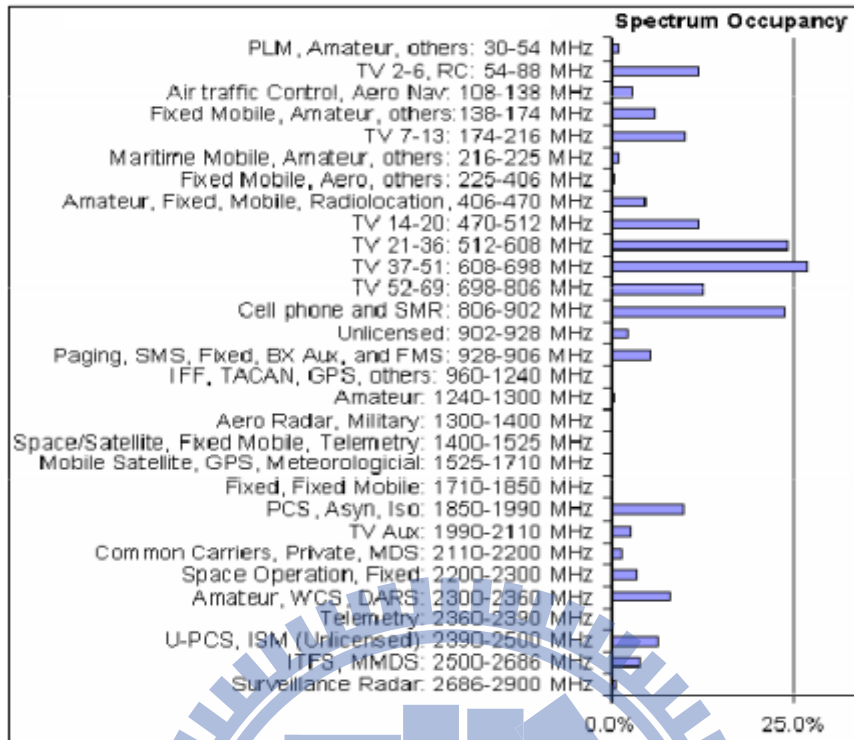
圖目錄

圖一	頻譜使用率.....	2
圖二	無線感知雲端網路意向圖.....	5
圖三	無線感知雲端網路環境示意圖.....	6
圖四	無線感知雲端網路架構圖.....	7
圖五	雲端內部系統架構圖.....	9
圖六	雲端架構圖.....	11
圖七	不同伺服器之間因擁有資訊不足，可能導致干擾發生.....	12
圖八	三角形佈件與方形佈件.....	14
圖九	分群環境轉換示意圖.....	15
圖十	邊界無線感知網路基地台轉換圖.....	17
圖十一	最佳化分群演算法解析.....	19
圖十二	隨機分群法執行流程解析.....	25
圖十三	最大代價分群法起始選點解析.....	27
圖十四	負載平衡分群法分群示意圖.....	28
圖十五	單群產生所有可能性示意圖.....	30
圖十六	最佳化頻帶資源分配法流程解析.....	32
圖十七	干擾值定義.....	35
圖十八	啟發式頻譜分配演算法環境示意圖.....	36
圖十九	分群法執行時間比較圖.....	41
圖二十	頻譜資源分配法執行時間比較圖.....	42

第一章 前言

1.1 無線感知網路(Cognitive radio networks)

在資訊爆炸的時代，越來越多的人們使用行動通訊裝置(例如手機、平板電腦等)藉由無線網路的媒介來分享其文章、照片、甚至是影片。隨著科技的進步發展，人們所要求的品質越高，在無線網路中所傳遞的資料量更是呈現爆炸性的成長。然而，頻譜資源並非能無限擴充使用，在頻譜資源有限的情況下頻譜顯的十分珍貴，要藉由拓展新頻譜的方式來增加傳輸品質將是十分困難且不切實際的。幸運的是，在 2002 年，美國聯邦通信委員會(Federal Communications Commission, FCC)發表了一項調查結果，此項調查結果指出頻譜的使用率隨著時間與地點的改變會有著極大的差異性(15%~85%)[1]，美國國防部先進研究計畫局(Defense Advanced Research Projects Agency, DARPA)也指出在某些離峰時間點，某些頻譜的使用率甚至只有 2% (如圖一)。總結以上觀察，感知無線電技術 (Cognitive radio technology) 成為了一提升頻譜使用率進而解決頻譜資源不足問題的可行方法。



圖一、 頻譜使用率，取自M. McHenry in “NSF Spectrum Occupancy Measurements Project Summary”, Aug 15, 2005

感知無線電技術的概念最早由 Joseph Mitola III 提出 [2]，感知無線電技術期望著藉由改善頻譜使用率來解決大都市中頻譜資源短缺的問題，感知無線電技術的主要想法是，非擁有頻譜執照的使用者(Unlicensed user/Secondary user, SU)能夠在擁有頻譜執照使用者(Licensed user/Primary user, PU)離開時，去借用其頻譜使用，以”借用”的概念期望提升整體頻譜的使用率。在感知無線電技術網路的領域中存在著兩個主要的挑戰，頻譜感測(Spectrum sensing)與頻譜管理(Spectrum management)，在以下將一一介紹。

1.1.1 頻譜感測 (Spectrum sensing)

感知無線電技術的主要概念為，當擁有頻譜執照使用者不在網路環境中時，非擁有頻譜執照使用者藉由存取擁有頻譜執照使用者所屬頻譜來增加頻譜使用率。非擁有頻譜執照使用者是否能夠存取頻譜，全藉由感測的結果，因此若是感測頻譜機制失準，將會因干擾產生而嚴重影響擁有頻譜執照使用者使用品質，將是不被容許的，因此，如何設計一套有效率且準確的頻譜感測方法是被廣泛討論的議題。然而，就單一非擁有頻譜執照使用者而言，有時會因為低的信噪比(signal-to-noise ratio, SNR)或是一些環境的不利因素而產生頻譜感測的誤判，因此近年來，協同式感測(cooperative spectrum sensing, CSS)的方式也被廣泛的提起討論[3]-[5]，憑藉著多筆的頻譜感測資訊，協同式感測能夠篩選出最正確的頻譜資訊結果，而在我們的雲端環境架構中也是採用協同式感測來進行。

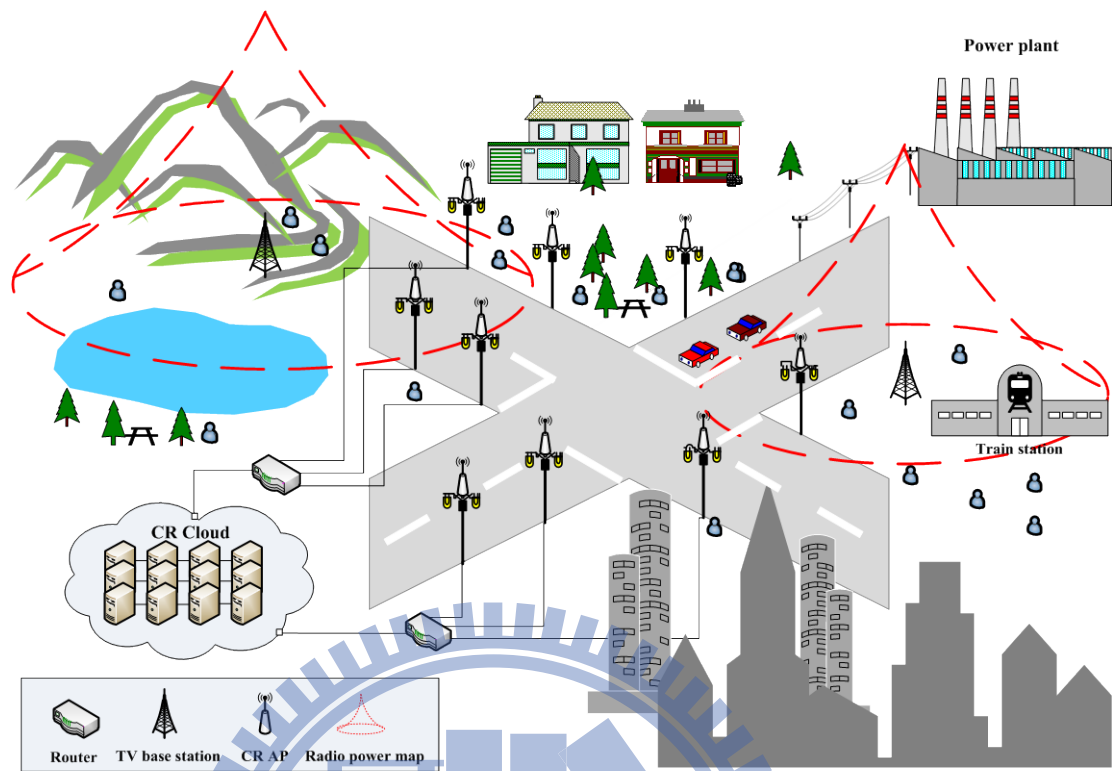
1.1.2 頻譜管理 (Spectrum management)

頻譜管理包含了頻譜確認(spectrum decision)，與頻譜資源分配(spectrum allocation)，其中頻譜確認是依據網路環境以及使用者的需求來決定每個不同使用者所能夠使用的頻譜集合。而頻譜資源的分配，則是在盡可能避免干擾的情形下，將頻譜資源做最有效的分配。頻譜資源的管理在整體無線感知網路中扮演著非常重要的角色，若能將整體資源做最大化的分配，將使整體網路頻譜使用率最佳化，對頻譜資源做最妥善的使用。

1.2 無線感知雲端網路(Cognitive radio cloud networks)

[6][7]兩篇論文提出了嶄新的概念，藉由著整合雲端的方式推出了無線感知雲端網路的架構。他們想像著，若有了無線感知雲端網路，感知無線電的概念將不在是小範圍的通訊方法，而是能成為或取代類似 3G、4G 的

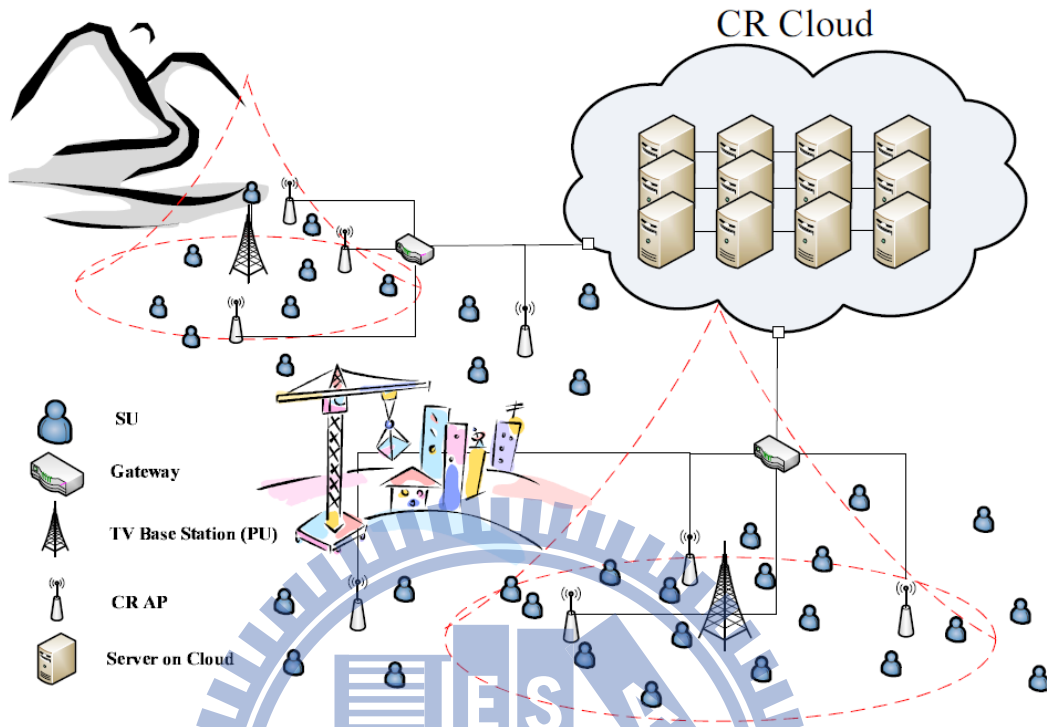
大規模通訊方式。為了在大尺度的環境中達成以上目標，我們需要一個強大的管理者來處理頻譜估測、頻譜管理、甚至於針對使用者的活動管理換手(hand-off)等等。為了整合以上目標，雲端就是我們最好的選擇，他們將雲端比喻成發電廠，使得人們能行走在有路燈照明的路上(如圖二)。這張圖表示未來若無線感知雲端網路普及，人們將可在路上享受著無縫式(seamless)的通訊品質且完全不用付費，最棒的是，使用者不用為了擔心影響 PU，而在本地端做複雜的估測動作。使用雲端帶給了整體感知無線電網路許多優點，第一、像是藉由雲端”租用”伺服器的概念，我們可以隨著網路中使用者分佈、活動的情形不同來動態的增加或減少所租用的伺服器個數，一方面在離峰時段能減少租用伺服器節省成本，另一方面也能在尖峰時段增加伺服器來負擔部分過載的伺服器的工作量，第二、雲端的主要概念，便是有著強大的資料處理能力與儲存空間，藉由雲端，我們可以有效的管理數以千萬計的頻譜感測資訊，並且能有效並且快速的運用這些資訊來估測頻譜當前狀況，第三、藉由著雲端的出現，它對未來的通訊方式提供了一些想像與彈性，日本情報通信研究機構(National Institute of Information and Communications Technology, NICT)在[8]提出了感知無線雲端(Cognitive wireless cloud)的概念，就是藉由著雲端的力量能夠管理並整合多種通訊方式，為異質網路架構提供嶄新的可能性。



圖二、無線感知雲端網路意向圖，取自[7] A Cognitive Radio Cloud

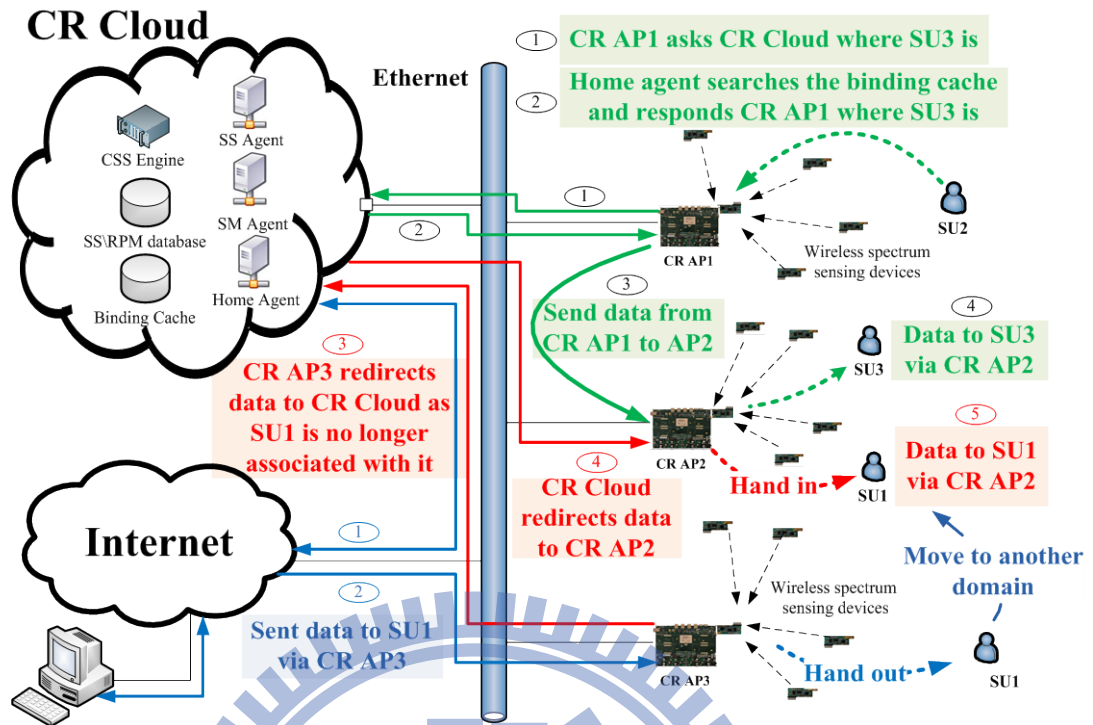
Networking (CRCN) service model in TV White Space

1.2.1 無線感知雲端網路架構



圖三、無線感知雲端網路環境示意圖取自[6] A conceptual Cognitive Radio Cloud network (CRCN) model in TV White Space.

無線感知雲端網路的架構(如圖三)，在環境中使用者(SU)分別由不同的無線感知網路基地台(CR access point, CR AP)管理，無線感知網路基地台又藉由網際網路來向雲端溝通，在[7]中提出了無線感知雲端網路上的架構，其在雲端上將協同式頻譜感測(CSS)、頻譜管理(spectrum management)、以及使用者的資訊活動皆由不同的伺服器作管理，在雲端架構中，主要有三個介面管理者(Agent)、一個運算單元(engine)以及兩個資料儲存單元(如圖四)，以下我們將一一介紹。



圖四、無線感知雲端網路架構圖，取自 [6] “A conceptual model and prototype of Cognitive Radio Cloud Networks in TV White Spaces,”

(A) 頻譜感測介面管理者 (Spectrum sensing agent, SS Agent)

頻譜感測介面管理者負責處理來自無線感知網路基地台回報的感測資訊，將來自不同地理位置的感測資料做處理分類並且將資訊(包含無線感知網路基地台 ID、座標、時間、RSSI 值)儲存至不同的頻譜感測資料庫(SS database)中，藉由這個簡單的分類動作，將能加速協同式頻譜感測演算法的速度。一旦將資料寫入頻譜感測資料庫後，協同式頻譜感測運算單元將立即自資料庫中取出資訊進行頻譜估測。

(B) 協同式頻譜感測運算單元 (Cooperative spectrum sensing engine, CSS Engine)

協同式頻譜感測運算單元負責針對 PU 所屬頻譜建立無線功率圖 (radio power map, RPM)，一旦頻譜感測介面管理者將頻譜感測資訊寫入頻譜感測

資料庫後，協同式頻譜感測運算單元將自資料庫中取出頻譜感測資訊並且執行基於貝式稀疏演算法(sparse Bayesian learning, SBL)所改良的協同式感測演算法[9]來估測 PU 所發出的訊號，並且將頻估結果寫入無線功率圖資料庫(RPM database)。

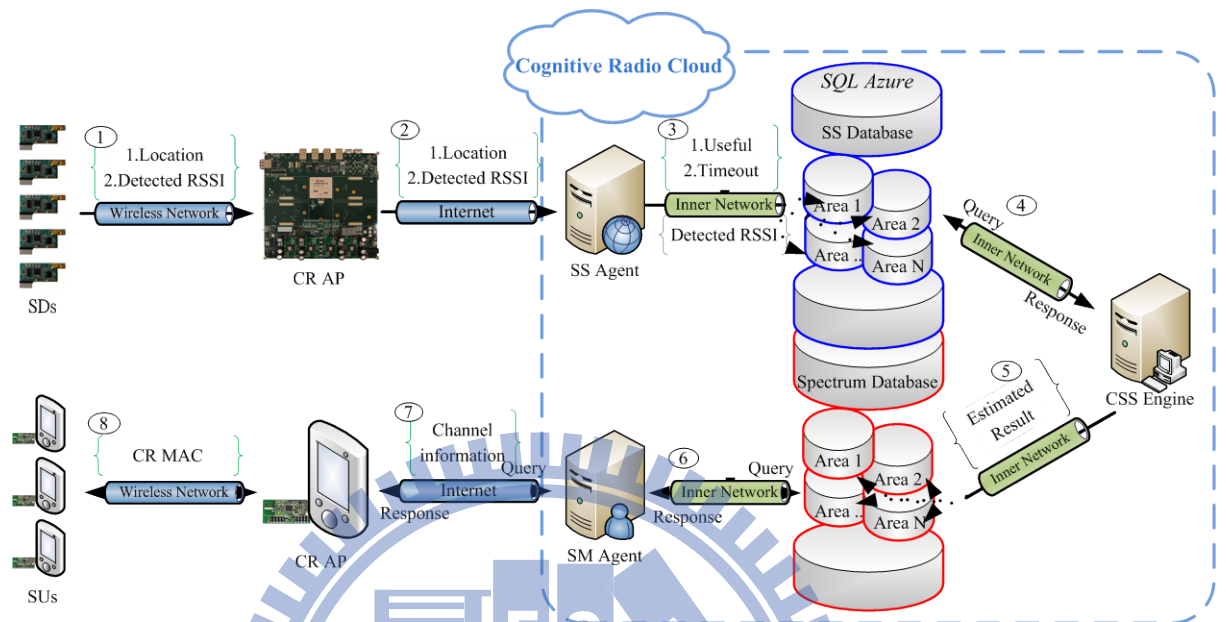
(C) 頻譜管理介面管理者(Spectrum management agent)

頻譜管理介面管理者負責分配各個無線感知網路基地台所能使用的頻譜資源，頻譜管理介面管理者會依據無線感知網路基地台的所屬位置，查詢所對應地區的無線功率圖資料庫來得之頻譜當前使用情況，執行頻譜資源排程後將能用頻譜分配給各個無線感知網路基地台。

(D) 行動資訊控管元件(Home Agent)

行動資訊控管元件負責管理各個不同使用者以及無線感知網路基地台的資訊，為了達到無縫換手的目標，行動資訊控管元件提供了暫存資料的服務，舉例來說，若使用者 1 將資訊傳給使用者 2，此時使用者 2 正在執行換手而不屬於任何一台無線感知網路基地台時，使用者 1 所傳輸的資訊將會儲存於行動資訊控管元件，直到使用者 2 正確無誤的換手完畢後，行動資訊控管元件將會查詢使用者 2 所屬無線感知網路基地台，並且將暫存資訊轉送至該基地台中，用以完成無縫換手的目的。藉由行動資訊控管元件，我們將能正確無誤的紀錄各個使用者及基地台的資訊用以提供各式各樣的應用服務。

1.2.2 元件運作流程介紹



圖五、雲端內部系統架構圖，取自[7]“A cloud model and concept prototype for cognitive radio networks,”

這一小節將完整介紹在雲端上完整的通訊運作流程。在[6][7]所提出的無線感知網路架構(如圖五)，在環境中存在著多個頻譜感測元件(sensing devices, SDs)依據其所定義的無線感知網路媒介存取控制(Media Access Control, MAC)負責週期性的感測頻譜資料，無線感知網路基地台會藉由輪詢(polling)的方式要求頻譜管理介面管理者回報頻譜感測結果。在其設計的無線感知網路媒介存取控制中，無線感知網路基地台必須在每個超級訊框(superframe)的開頭，週期性的回報其收集的感測資訊至雲端，藉由週期性的更新動作，雲端能迅速掌握當下頻譜資訊狀況。雲端一旦接收到最新的頻譜感測資訊，頻譜感測介面管理者將做相對應動作如之前所述，連帶著帶動協同式頻譜感測運算單元、頻譜管理介面管理者做一連串的頻譜資

訊處理，隨時更新保持雲端上面頻譜資訊的正確性。

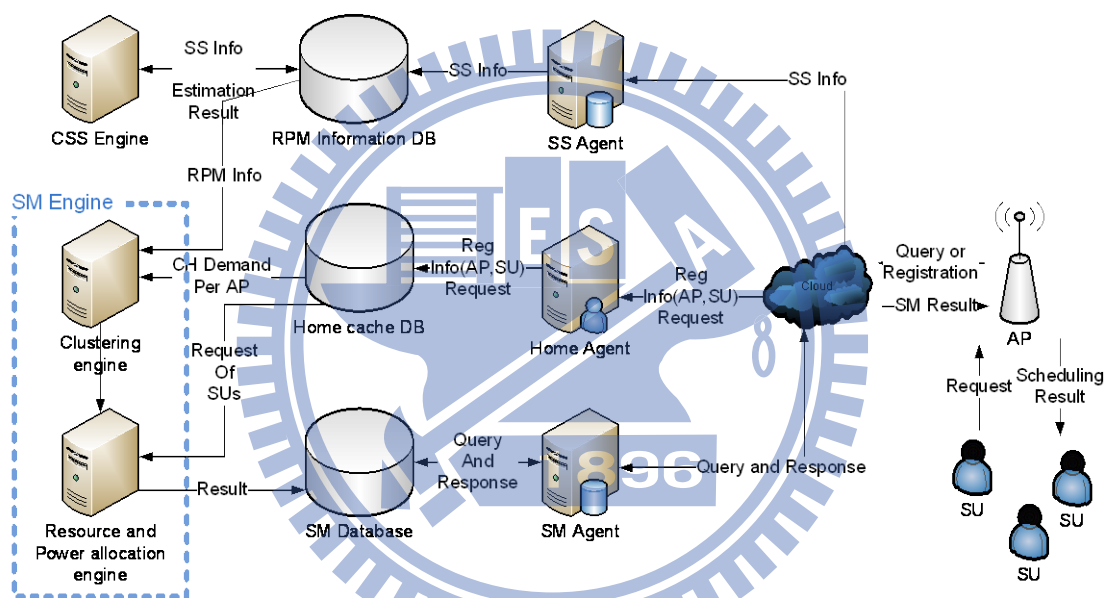
一旦使用者(SU)想要使用溝通服務時，使用者將發送頻譜使用需求給其所屬無線感知網路基地台(無線感知網路基地台會藉由週期性(每個超級訊框)的詢問雲端來得知該地區所能使用的頻譜資訊)，在無線感知網路基地台對該地區的使用者進行頻譜資源排程後，將會將頻譜分配結果夾帶在信標(Beacon)中廣播出去，假如該使用者獲得頻譜使用，使用者將可使用所被分配到的頻譜進行通訊，若該使用者未得到任何的頻譜資源，則該使用者必須在下一個超級訊框中再做一次需求的發送。

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數以萬計的無線感知網路基地台。我們觀察到現今存在的各種資源配置方式不適用於我們所面對的環境中，依據我們所量測的結果發現，時間複雜度約在 $O(N^2)$ 左右的演算法，在 N 超過 10000 時，必須花費大於兩秒的運算時間(Intel Core i5,3.1GHz)，然而這個運算時間是我們所不能忍受的。為了符合FCC對時間的規範(對PU的干擾時間必須小於 2 秒)，考慮到時間限制因素，我們嘗試著去改進雲端的架構。

我們最主要的想法為，首先將大尺度的環境進行切割，區分為多個子環境，接著將每個子環境交付由單一伺服器管理，藉由雲端伺服器平行化的特性，平行做資源與功率分配(Resource and power allocation)，藉由平行化的計算方式加速來達到FCC的時間規範。在實做中，第一步我們將頻譜管理介面管理者內部添加一頻譜管理運算單元(Spectrum management engine, SM Engine)包含了分群處理單元(Clustering engine)以及資源與功率分配單元(Resource and power allocation engine)(如圖六)，在頻譜管理運算單元內將對各個無線感知網路基地台執行分群、資源分配與功率分配，而

頻譜管理介面管理者則扮演著雲端與無線感知網路基地台間傳遞頻譜資訊的介面。分群處理單元內部主要負責兩件事、將大環境進行切割至多個小環境以及對¹邊界的無線感知網路基地台進行初步的頻譜資源分配。在分群處理單元完成工作後，整體大環境已被區分為多個子群，我們將每個子群交付給不同的資源與功率分配單元執行資源分配與功率分配。然而，為何分群處理單元必須先對邊界的無線感知網路基地台進行初步的頻譜資源分配呢?原因在於我們必須避免邊界無線感知網路基地台的互相干擾產生。

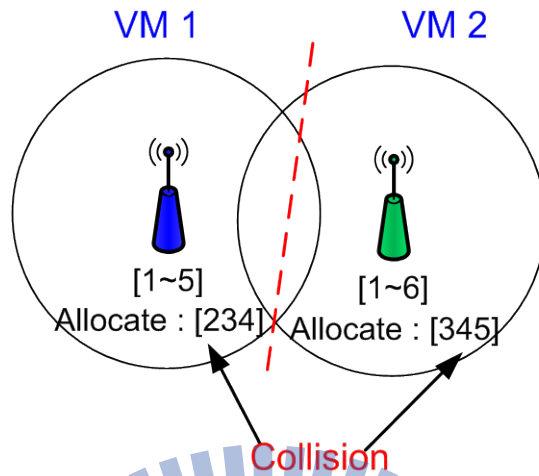


圖六、雲端架構圖

分群處理單元一旦將子群交付給不同的資源與功率分配單元進行處理後，各個資源與功率分配單元所考慮的環境將不再是大環境而是被分配過後的子群，因此若不先對邊界的無線感知網路基地台進行初步的頻譜資源分配，將可能導致干擾產生。舉例來說(如圖七)，在圖中存在著兩個邊界無線感知網路基地台分屬於不同的群由不同伺服器管理，若是未將其做初步的頻譜資源分配，資源與功率分配單元將因看不到相鄰但卻不同群的邊

¹ 邊界的無線感知網路基地台: 兩個基地台相鄰但卻被分配為不同群組，此兩個基地台就被稱為邊界的無線感知網路基地台

界無線感知網路基地台，而有可能賦予其相同頻譜導致干擾產生(例子中頻譜 3,4 重覆配置導致干擾產生)。



圖七、不同伺服器之間因擁有資訊不足，可能導致干擾發生。

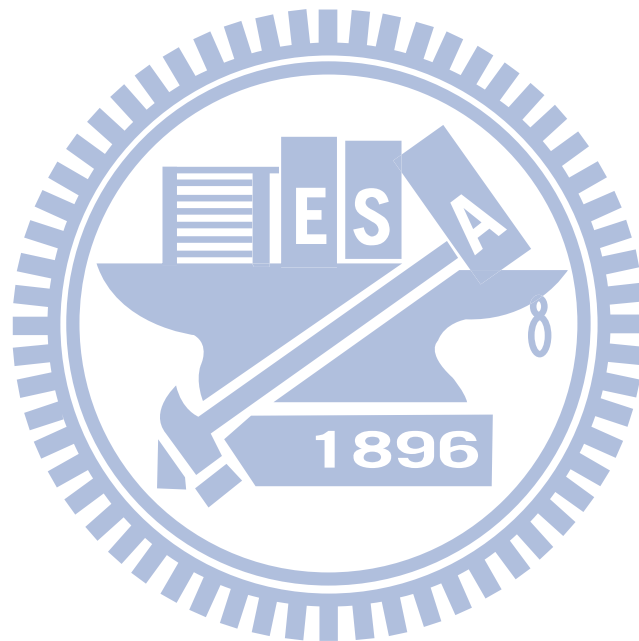
在這篇論文中，我們強調在分群處理單元上，主要負責對環境進行分群以及對邊界的無線感知網路基地台進行初步的頻譜資源分配。在本篇論文中我們針對這兩個問題分別提出了最佳化的演算法，並且做理論上的證明與時間複雜度的分析，然而對應在實作上，執行最佳化演算法將是十分耗時且不切實際的，因此我們嘗試著針對目標設計好的啟發式演算法並且分析其時間複雜度以及以數據模擬的方式比較其與最佳化的差距。

分群問題在現今學術界中是一個已被廣泛討論的問題，至今已有許多啟發性的演算法被提出 [10][11]，近年來更進一步提出多層 (Multilevel)[12-14] 架構有效且有彈性的減低時間複雜度。然而，這些被提出的分群方法皆沒有考慮到地區性的連通必要性，換句話說，這些方法皆有可能導致多個完全分散的子圖被分配到同一群，考慮到雲端上地區性管理的角度，這些方式皆不適用於我們的雲端架構中。

邊界的無線感知網路基地台的頻譜資源分配問題，相當於是 vertex listed multi-coloring problem，在[15]已被提出並證明是 NP-hard 的問題，在

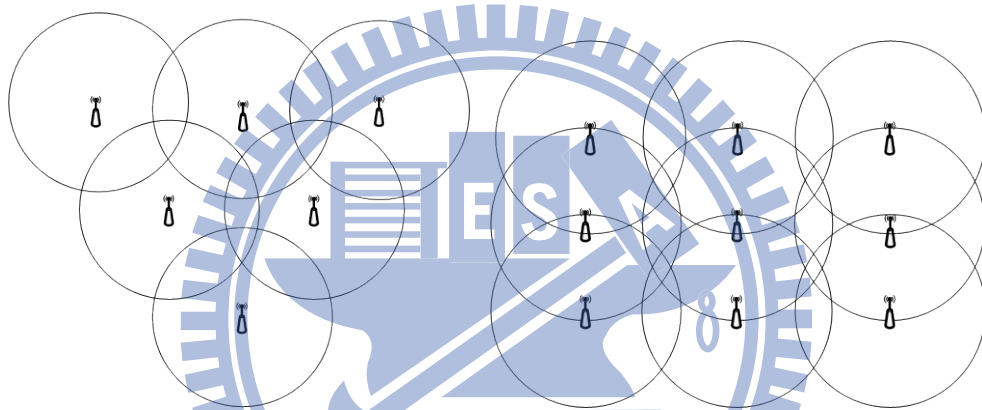
我們的研究討論中，我們提出最佳化的解決方式，但如上述所說，受限於時間因素我們也提出時間複雜度相對低但能有不錯效能的啟發性演算法。

在第二章我們將介紹整體考量的系統環境，並且在第三張對分群與資源分配問題做數學定義及清楚的問題描述，第四章將對所提出的方法做介紹、理論證明、以及時間複雜度分析，而第五章將模擬並分析比較數據結果最後在第六章將對本篇貢獻及未來目標做闡述。



第二章 系統環境介紹與問題定義描述

在無線感知雲端網路中，我們考量的是一個大尺度的環境，網路中存在著數以萬計的無線感知網路基地台被配置在網路環境當中。在網路環境中我們考量兩種佈件方式，三角形以及方形(如圖八)。無線感知網路使用者在無線感知網路基地台的覆蓋範圍下皆呈現均勻分布(uniformly distribution)，我們假設每個無線感知網路基地台所打的最大功率皆相同，為了簡化問題，我們考慮每條頻譜都擁有相同的資料傳輸率(data rate)，每個無線感知網路基地台的可用頻譜也設為已知。



圖八、三角形佈件與方形佈件

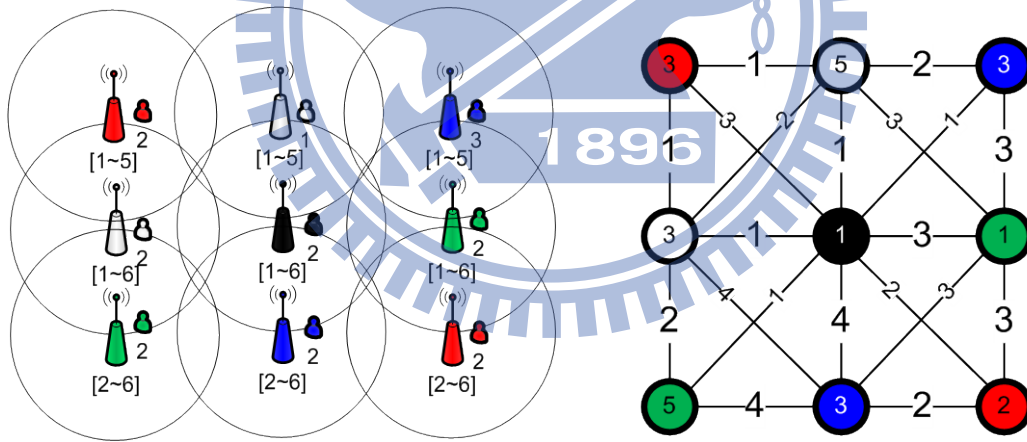
2.1 分群問題

在環境網路中，我們將無線感知網路基地台的拓樸(topology)轉變為一簡單圖 $G = (V, E)$ (如圖九)。每個點表示無線感知網路基地台，並且擁有其可用頻譜集合以及所需求的頻譜數目，點跟點之間存在邊表示其基地台間功率範圍有互相覆蓋，每個點上都存在一個值”工作量”代表其所需要的工作運算單元量(請參考附錄一)，每個邊上也存在一值”代價”，代表其必定產生的不被滿足需求量，其中代價值 $p_{i,j}$ 定義如下

$p_{i,j}$

$$\begin{aligned}
 & \text{average requested channels in the intersection of nodes } i \text{ and } j && \text{No of uncommon channels of node } i \\
 & \frac{q_i A_{inj}}{A} && - (n_i - n_{inj}) \\
 = & \text{ceiling}(\max\{ \underbrace{\max[\frac{q_i A_{inj}}{A} - (n_i - n_{inj}), 0]}_{\text{the number of residual requests after allocating uncommon channels of node } i} + \max[\frac{q_j A_{inj}}{A} - (n_j - n_{inj}), 0] - n_{inj}, 0\}) \\
 & && \underbrace{\hspace{10em}}_{\text{the number of residual requests after further allocating common channels of nodes } i \text{ and } j}
 \end{aligned}$$

其中 q_i 代表基地台 i 的頻帶需求數， A_{inj} 代表基地台 i 與基地台 j 最大功率覆蓋交集面積大小， A 代表基地台最大功率覆蓋面積， n_i 代表基地台 i 的可用頻帶個數， n_{inj} 代表基地台 i 與基地台 j 相同的可用頻帶個數，因此 $p_{i,j}$ 表示為給定最多能使用的資源前提下必定不滿足的需求數。



圖九、分群環境轉換示意圖

給一圖 $G = (V, E)$ ，工作負載量 $L(v) \forall v \in V$ ，對於所有邊 $\bar{e}_{i,j}$ 存在代價 $P_{i,j}$ ，每個群的最大工作負載量 μ ，我們將問題轉化為下面最佳化式子，我們期望能夠在滿足最大工作負載量的條件下最小化分群所產生的總代價。

Primal :

$$\min \sum_{(i,j) \in E} P_{i,j} * X_{i,j}$$

Subject to

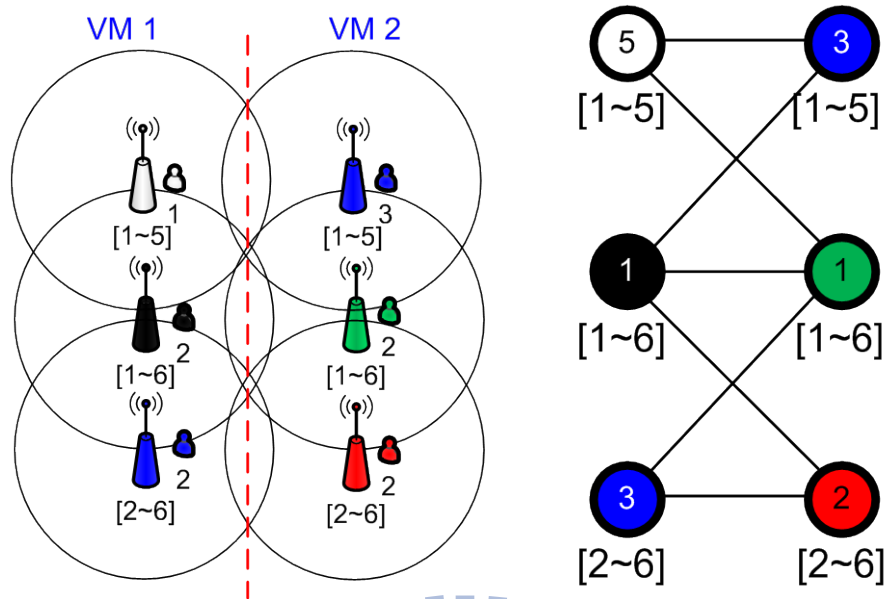
$$Q_i \leq \mu$$

$$\forall v_a, v_b \in S_i, \exists 1 \text{ path in } S_i : v_a \text{ connect } v_b$$

我們假設M為子群總數， $S = \{S_1, S_2, \dots, S_M\}$ ，S為群集合，其中 $X_{i,j} = 1$ 假如 $\bar{e}_{i,j}$ 是邊界邊 ($v_i \in S_a, v_j \in S_b, S_a \neq S_b$)，否則 $X_{i,j} = 0$ ， $Q_i = \sum_{v_j \in S_i} \sum L(v_j)$ ，代表子群 S_i 的工作量。

2.2 邊界點的頻譜資源分配問題

在分群演算法執行過後，我們將擁有邊界無線感知網路基地台的資訊，接續分群後的結果，我們將其拓撲轉變成 $G = (V, E)$ ，其中 V 代表邊界無線感知網路基地台的集合，兩點之間存在邊代表其所打的最大功率範圍將會互相覆蓋並且不屬於同一群(如圖十)，換句話說，我們不考量同群之間所產生的干擾，而只考慮不同群之間，原因在於同群之間的問題將在下一層”資源與功率分配單元”中被處理。我們將問題之數學定義如下，我們期望在分配出去頻譜數量最大化的前提下找尋最公平的結果。



圖十、邊界無線感知網路基地台轉換圖

給一圖 $G = (V, E)$ ，對於所有點存在可用頻譜集合 R_v ，無線感知網路基地台的頻譜需求量 q_v ，我們定義最佳化問題如下

Primal :

$$\text{Maximum } (\sum_v p_v) - \sigma$$

Subject to.

$$p_v \leq q_v$$

$$K_u \cap K_v = \emptyset, \text{ for all } \overline{e_{u,v}} = 1$$

$$K_v \subseteq R_v, \forall v$$

Where

p_v is number of allocated channel of node v , $p_v = |K_v|$

K_v is the allocated channel set of node v

σ^2 : the satisfaction ratio variance

$$\sigma^2 = \frac{1}{|V|} \sum_v (F_v - F_{\text{avg}})^2$$

F_v : the satisfaction ratio of node v , define $F_v = \frac{p_v}{q_v}$

F_{avg} : the average satisfaction ratio

第三章 提出方法與時間複雜度分析

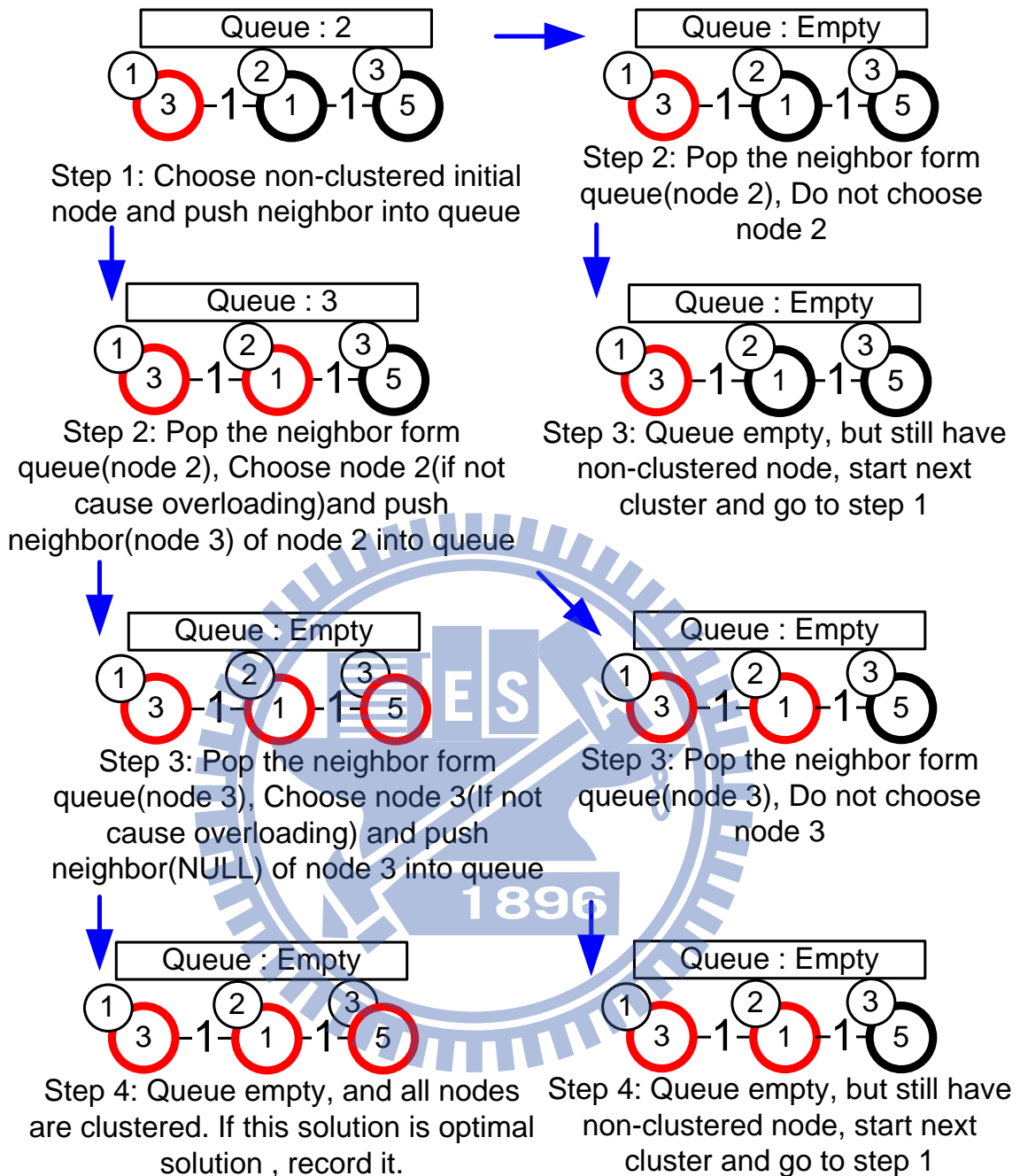
3.1 分群

在這一小節中，我們將針對分群問題介紹所提出的最佳化與啟發式解決方式，我們將提出並證明一些理論與定理，並且比較其時間複雜度。

3.1.1 最佳化分群法介紹

此最佳化演算法想法啟發自廣度搜尋法(breadth-first search, BFS)，我們使用廣度搜尋法來探索子圖，一但探索到其中一點將存在著兩種可能性，”選”或”不選”，如果選擇了此點我們將此點的鄰居放入序列(Queue)中並且繼續最佳化演算法，若不選，我們將完全捨棄此點並繼續直到序列中全空為止。在選擇此點之前我們還須考量是否符合工作量負載的限制，假如選擇此點將導致子群過載，我們將捨棄此點(如圖十一)，整體詳細的虛擬程式碼(pseudo code)如下所示。





圖十一、最佳化分群演算法解析

BFS_Optimization(StartNodeNumber N, AdjacencyQueue Q , clusternum C)

If (Q is empty)

// this cluster's process finish

- If(there is no non-clustered node)
 - Check this solution is better than optimal or not
 - If yes renew the solution
- Else
 - Set all the non-clustered node to white color
 - Find the not clustered node N'
 - Push N' to Q
 - Start next cluster, BFS_Optimization(N', Q , C+1)

Else

// Find all BFS sub-graphs

- TargetNode T = Q's front node, and pop this node from Q
- Color this node to black
- Record the condition of Queue and Color
- If(Select the node T will not cause overloading)
 - Choose this node T , Set the cluster number of T = C
 - Push all the neighbors of T into Q which are not already in Q and it has white color
 - BFS_Optimization(N, Q , C)
- If(T != N)
 - Recovery the condition of Queue and color and set the cluster number of T = non-clustered
 - Not choose this node T
 - BFS_Optimization(N, Q , C)

Main()

Push initial node N into Q;

BFS_Optimization(StartNodeNumber N, AdjacencyQueue Q , clusternum C);

3.1.1.1 理論與證明

- 理論 1. 給一 N 個點的簡單圖 (simple graph) $G=(V,E)$ ，給任意起始輸入點 \bar{v} ，此最佳化演算法可以找尋所有包含點 \bar{v} 的連通子圖。

證明:

在這邊我們使用數學歸納法證明。

當 $N=1$ 時，只有一種簡單圖的可能性: ①

給定任意起始點為點 1

我們的演算法必定會選擇第一點 (因為我們不考慮空群)



可以，此演算法可以跑過所有包含點 1 的可能連通子圖當 $N=1$ 時。

當 $N=2$ 時有以下幾種可能的簡單圖: ① ② ①—②

我們不失一般性的假設任意起始輸入點為點 1

我們的演算法必定會選擇第一個點 1，並且循序的去探索序列中的鄰居並跑兩種可能”選”或”不選”

種類一:

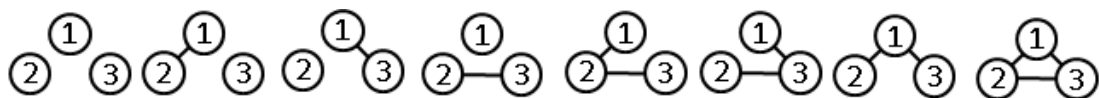


種類二:



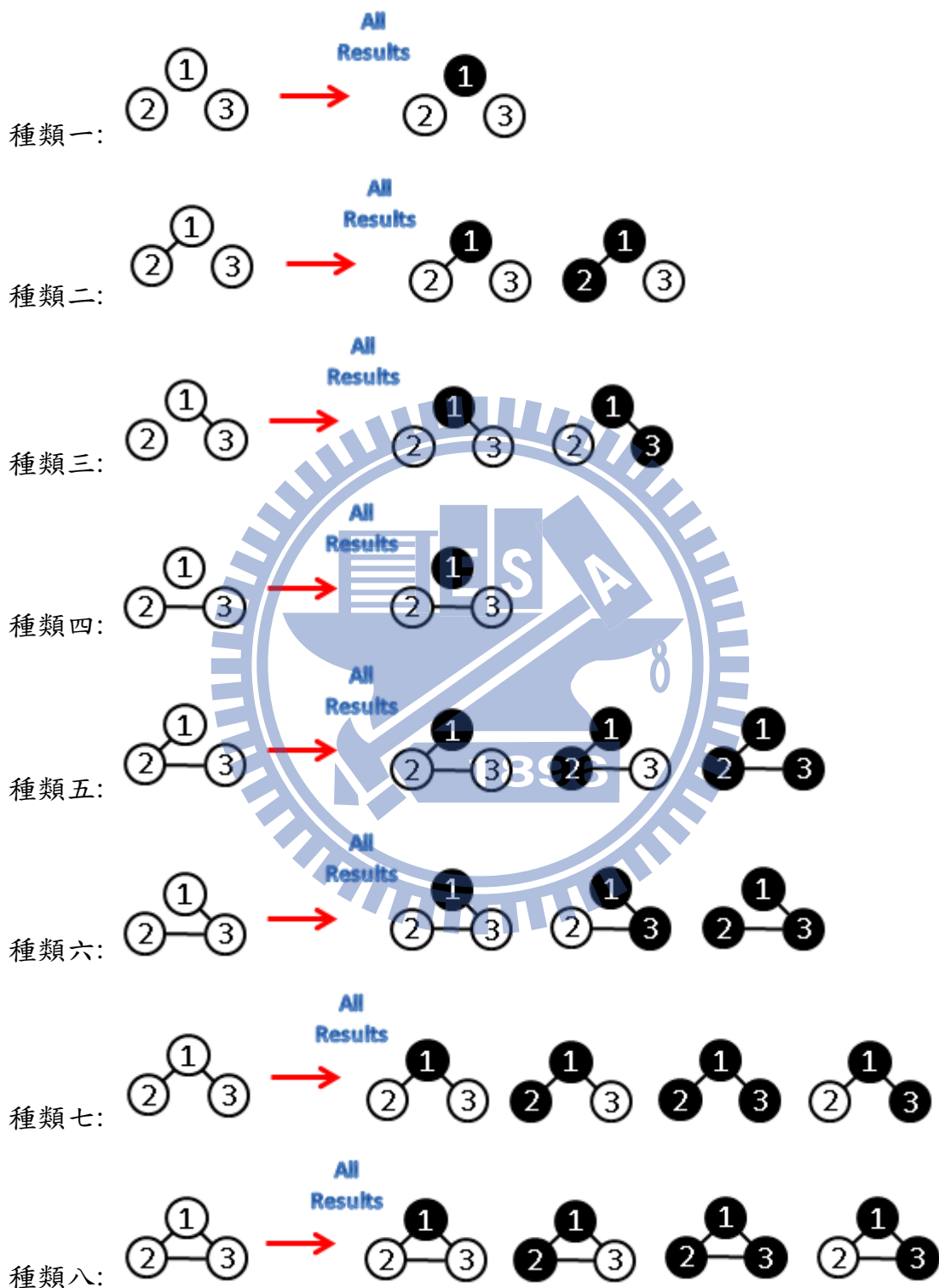
可以，此演算法可以跑過所有包含點 1 的可能連通子圖當 $N=2$ 時。

當 $N=3$ 時有以下幾種可能的簡單圖:



我們不失一般性的假設任意起始輸入點為點 1

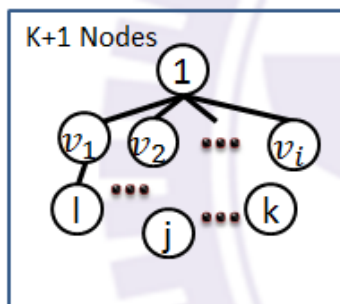
我們的演算法必定會選擇第一個點 1，並且循序的去探索序列中的鄰居並跑兩種可能”選”或”不選”



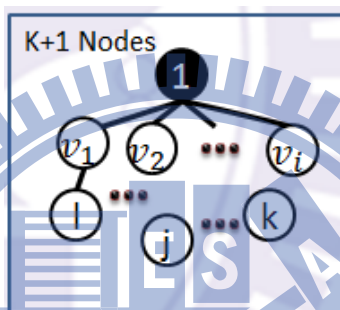
可以，此演算法可以跑過所有包含點 1 的可能連通子圖當 $N=3$ 時。

我們現在假設當 $N=k$ 時可以，我們考慮當 $N=k+1$ 時

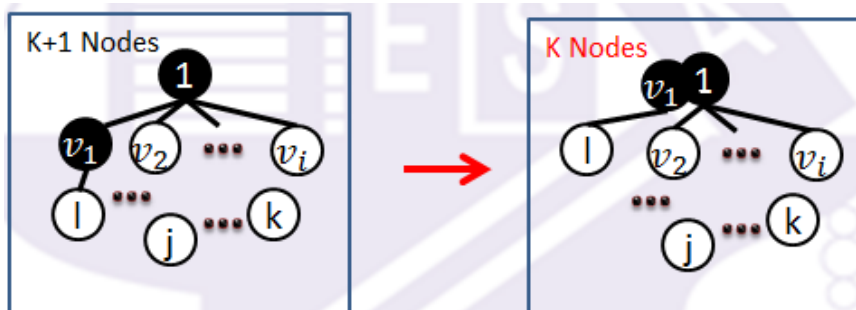
給一簡單圖包含 $k+1$ 個點，不失一般性的我們假設給任意起始點點 1，並且假設點 1 存在鄰居 $v_1, v_2, \dots, v_i, 0 \leq i \leq k$



我們的演算法必定會選擇第一個點 1



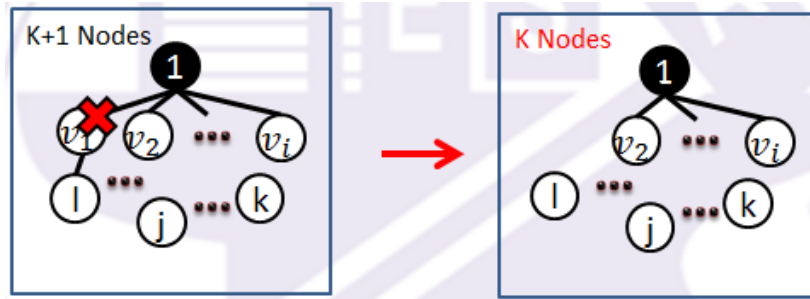
假如點 1 不存在鄰居，我們已經跑過所有可能，可以否則，對於所有點 1 的鄰居我們將循序的去探索序列中的鄰居並跑兩種可能”選”或”不選”，在這邊不失一般性的假設 v_1 是我們第一個探索的點若選擇 v_1 ，我們將會將 v_1 的鄰居放入序列中，此情況相當於我們將點 1 及點 v_1 兩點整合，並且將其當作起始點在只有 k 點的環境下進行我們的演算法如下圖所示



依照先前在 k 點環境成立的假設，此情況成立。

反之若不選 v_1 ，此情形相當於我們直接將 v_1 從圖中刪除，並且把點一當作

起始點在只有 k 點的環境下進行我們的演算法如下圖所示



依照先前在 k 點環境成立的假設，此情況成立。

在這裡我們證明了當 $N=k+1$ 時，此演算法可以跑過所有包含點 1 的可能連通子圖。

- 定理 1. 給一 N 個點的簡單圖 (simple graph) $G=(V,E)$ ，此最佳化演算法可以找尋所有可能子群，其中子群必為完全連通圖。

證明:

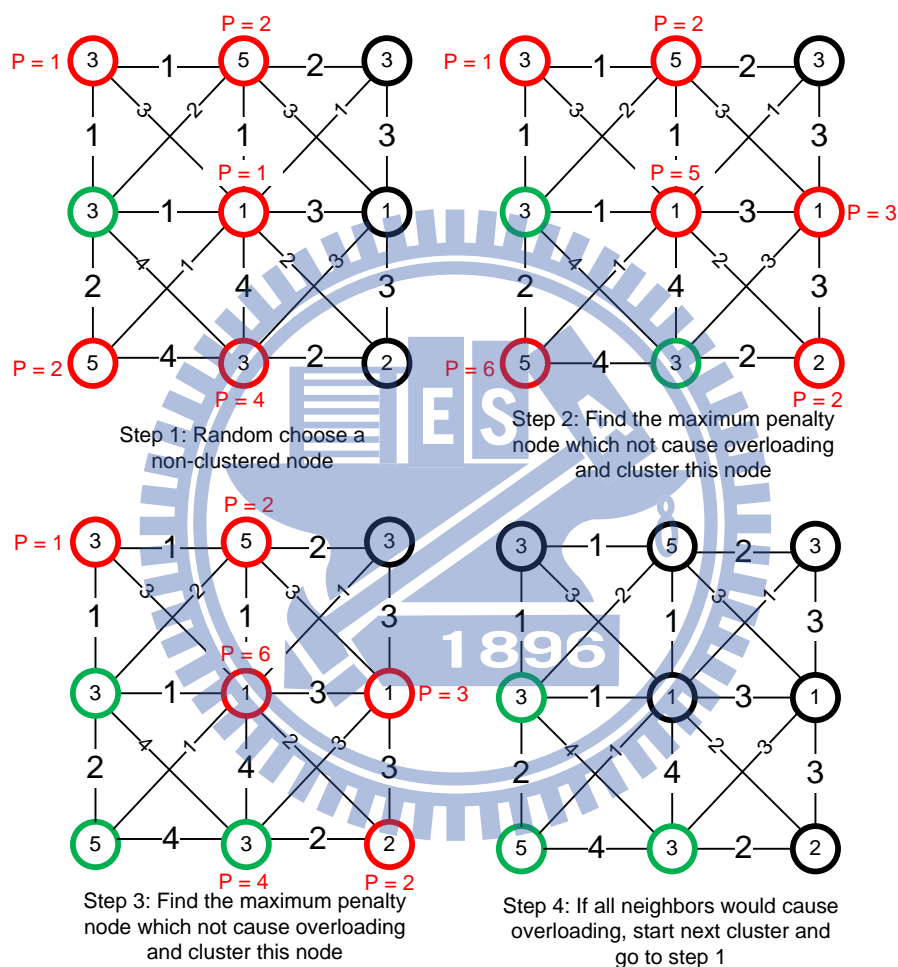
由理論 1 我們可以知道，給任意起始點 \bar{v} ，最佳化演算法將會搜尋包含點 \bar{v} 的所有可能連通子圖，在演算法第一層將會定義被選擇到點的群集編號為 $C1$ ，並且繼續找尋剩餘尚未被分群的點進行第二次分群，重覆著相同動作並定義第二層所選點的群集編號為 $C2$ ，依此類推...直到所有點皆已被分群。每一次做最佳化演算法皆是根據上一層所產生的分群的结果為先前條件而繼續分群，然而每一層皆會跑過所有可能結果，因此我們的最佳化演算法能夠找出所有可能子群，且其子群必為完全連通圖。

3.1.2 啟發式分群法介紹

在本篇論文中，我們提出三種啟發性的分群解法其中包含隨機分群法、最大代價分群法以及負載平衡分群法，以子群內必須連通的本質為基準，依據不同角度切入解決問題，以下將分別介紹。

(A) 隨機分群法

如圖十二，隨機分群法主要想法為在環境中任意選擇子群起始點(綠色點)，依據子群內成員必須連通為原則，循序找尋其族群成員鄰居代價最高者指派為同一群成員，直到子群無法在抓取其他族群成員鄰居為止(工作量滿載)，其虛擬程式碼如下。

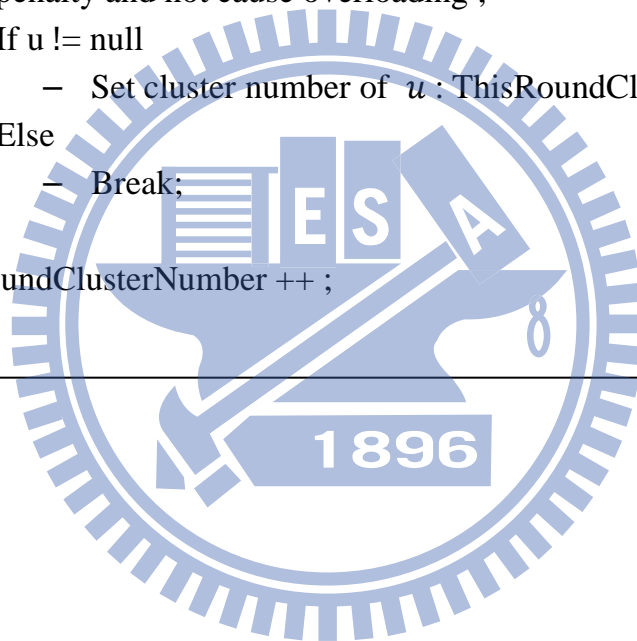


圖十二、隨機分群法執行流程解析，假設工作量負載限制為 11

ThisRoundClusterNumber = Initial Cluster Number

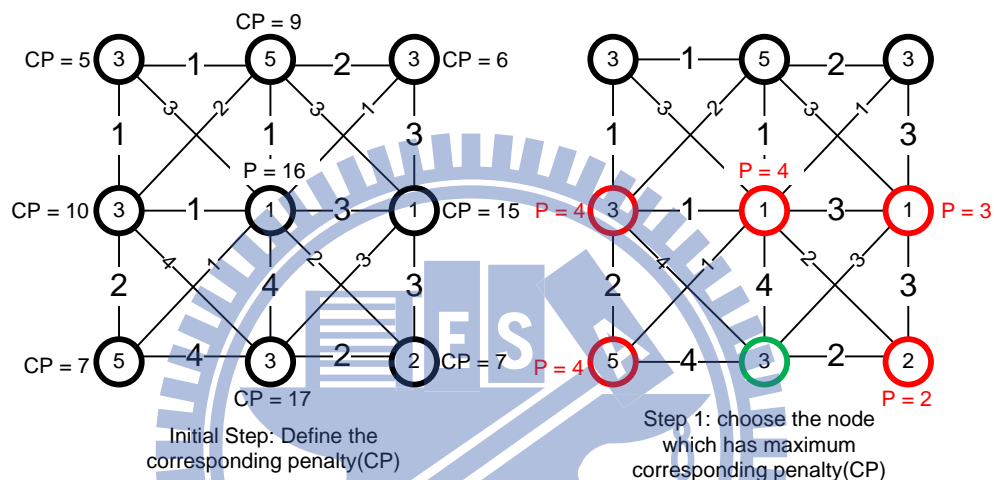
While (all the nodes are clustered)

```
{  
  - Random choose a node  $v$  ;  
  - Set the cluster number of  $v$  : ThisRoundClusterNumber;  
  - While (true)  
  - {  
    • Choose the neighbor node of clustered node  $u$  which has max  
      penalty and not cause overloading ;  
    • If  $u \neq \text{null}$   
      - Set cluster number of  $u$  : ThisRoundClusterNumber;  
    • Else  
      - Break;  
  - }  
  - ThisRoundClusterNumber ++ ;  
}
```



(B) 最大代價分群法

最大代價分群法主要想法為在環境中選取最有影響力的點為子群起始點(最有影響力定義為選取該點對周遭造成的代價最高)(如圖十三),依據子群內成員必須連通為原則,循序找尋其族群成員鄰居代價最高者指派為同一群成員,直到子群無法在抓取其他族群成員鄰居為止(工作量滿載),其接續選點法如同隨機分群法所設計,其虛擬程式碼如下。



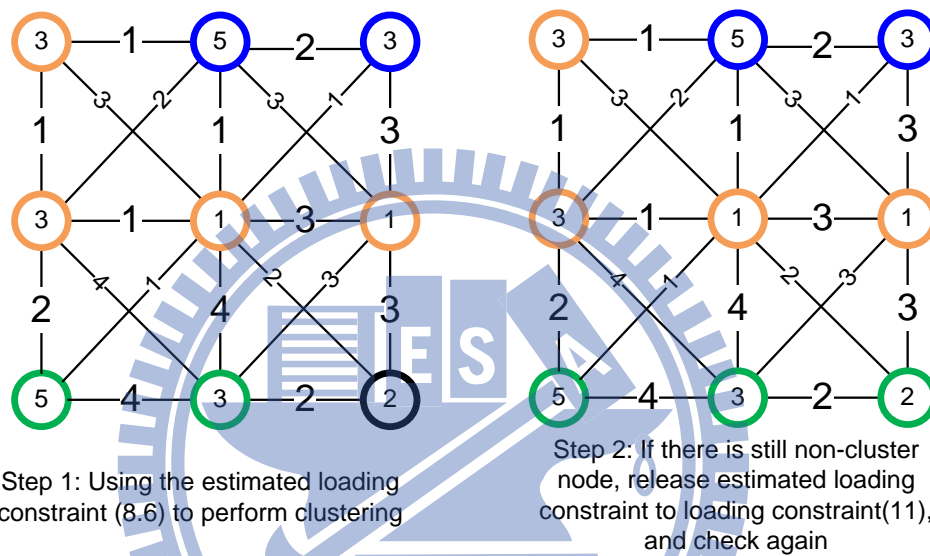
圖十三、最大代價分群法起始選點解析

```

ThisRoundClusterNumber = Initial Cluster Number
While (all the nodes are clustered)
{
    - Choose the node  $v$  which has maximum  $\sum_j P_{v,j}$  ;
    - Set the cluster number of  $v$  : ThisRoundClusterNumber;
    - While (true)
    - {
        • Choose the neighbor node of clustered node  $u$  which has max
          penalty and not cause overloading ;
        • If  $u \neq \text{null}$ 
            - Set cluster number of  $u$  : ThisRoundClusterNumber;
        • Else
            - Break;
    - }
    - ThisRoundClusterNumber ++ ;
}
    
```

(C) 負載平衡分群法

負載平衡分群法的主要想法為，希望藉由估測平衡的負載量將伺服器做最平衡的負載配置，其分群方式我們採用隨機分群法，差別在於，我們在負載平衡分群法中先做第一階段的子群分群，但其負載限制設定為所估測最平衡的負載值進行分群，若環境中尚有未被分群到的成員，在將負載限制放寬，進行第二階段分群(如圖十四)，其虛擬程式碼如下。



圖十四、負載平衡分群法分群示意圖，假設附載限制為11，整理環境總工作量為26 因此所需三台伺服器，因此每台伺服器平均負載為8.6

```

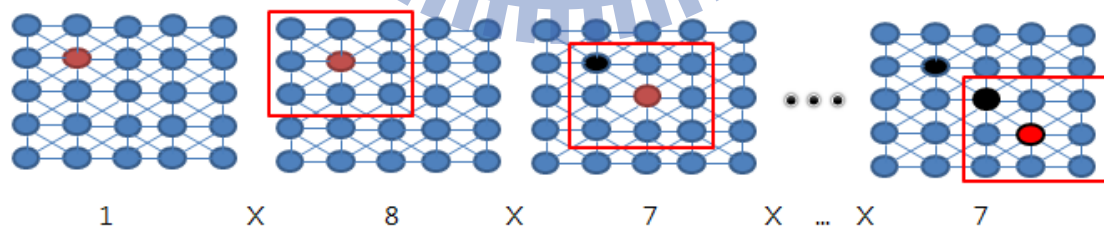
While (all the nodes are clustered)
{
  • If(First Round Clustering)
    – Set loading constraint = best balancing loading;
  • Else
    – Release loading constraint to max loading constraint  $\mu$ ;
    – Estimated number of VM = infinity;
  • ThisRoundClusterNumber = Initial Cluster Number;
  • While(true)
  • {
    – If ThisRoundClusterNumber > Estimated number of VM;
      • Break;
    – Random choose a node  $v$  ;
    – Set the cluster number of  $v$  : ThisRoundClusterNumber;
    – While (true)
    – {
      • Choose the neighbor node of clustered node  $u$  which has max
        penalty and not cause overloading ;
      • If  $u \neq \text{null}$ 
        – Set cluster number of  $u$  : ThisRoundClusterNumber;
      • Else
        – Break;
    – }
    – ThisRoundClusterNumber ++ ;
  • }
}

```

3.1.3 時間複雜度分析

首先我們分析最佳化分群演算法的時間複雜度，最佳化分群演算法為窮舉演算法將會掃描所有分群可能，我們以下嘗試著為最佳化演算法找出上界(Upper bound)。

我們定義符號 C_i 代表群集 i 內成員個數， N 代表所有成員總數，首先我們假設每個群集內成員數固定，觀察在成員數固定的情況下有幾種存在的可能。我們先觀察單一群集，群集 1 存在 C_1 個點，那其所有子圖的可能數必小於 $1*8*7*...*7 \cong 7^{C_1-1}$ 種可能(如圖十五)， i 個群集種共有 $7^{C_1-1}*7^{C_2-1}*...*7^{C_i-1}$ 種可能，以上考慮為 C_1, \dots, C_i 成員數固定的情況下，接下來我們將考慮 C_1, \dots, C_i 的成員數所有組合數，恰等於 $C_1 + C_2 + \dots + C_i = N, C_1, \dots, C_i \geq 1$ 中求所有 C_1, \dots, C_i 的組合的可能數，其解為 $\frac{(N-1)!}{(N-i)!(i-1)!}$ 。在群集數為 i 時，可能數必小於 $7^{C_1-1} * 7^{C_2-1} * \dots * 7^{C_i-1} * \frac{(N-1)!}{(N-i)!(i-1)!}$ ，所以所有可能數必小於 $\sum_{i=1}^N 7^{C_1-1} * 7^{C_2-1} * \dots * 7^{C_i-1} * \frac{(N-1)!}{(N-i)!(i-1)!} \cong \sum_{i=1}^N 7^{N-i} \frac{(N-1)!}{(N-i)!(i-1)!}$ ，因此最佳化分群演算法的時間複雜度為 $O(\sum_{i=1}^N 7^{N-i} \frac{(N-1)!}{(N-i)!(i-1)!})$



圖十五、單群產生所有可能性示意圖

而啟發式演算法因每抓取一點皆搜尋本回合群集鄰居，鄰居不多於 N 個，共抓取 N 點，因此時間複雜度為 $O(N^2)$ ，時間複雜度比較表如下圖。

表一、分群法時間複雜度比較

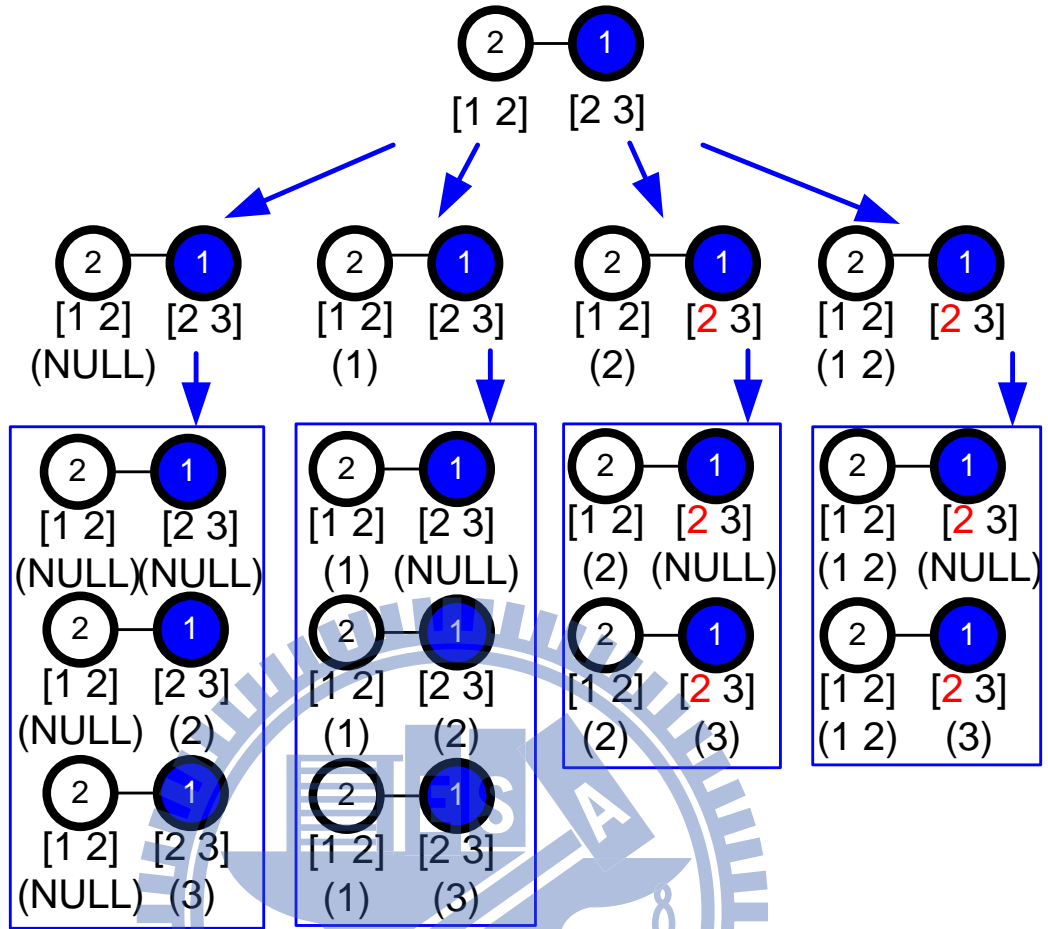
Random choose	$O(N^2)$
Max Penalty	$O(N^2)$
Release Loading Constraint	$O(N^2)$
Optimal	$O(\sum_{i=1}^N 7^{N-i} \frac{(N-1)!}{(N-i)!(i-1)!})$

3.2 邊界點的頻譜資源分配

在這一小節中，我們將針對分群問題介紹所提出的最佳化與啟發式解決方式，我們將證明一些理論與定理，並且比較其時間複雜度。

3.2.1 最佳化頻譜資源分配法介紹

此最佳化頻譜資源分配法想法於嘗試找尋所有頻譜分配可能，並由所有可能中找尋最佳的分配方式，虛擬程式碼如下，其概念為針對每個環境中的邊界無線感知網路基地台，我們找尋其所有小於需求數的頻譜配置可能，針對每種配置方式，我們限制其鄰居不可配與相同頻譜，並繼續搜尋，重覆已找尋所有可能(如圖十六)，其最佳化頻譜資源分配法的正確性將在下一小節說明證明。



圖十六、最佳化頻帶資源分配法流程解析，點中的數字代表頻帶需求
量，而[]中數字代表可用頻帶，()中數字代表配置頻帶。

```
Resource_Allocation_Optimization(Node N)
```

```
{
```

```
  If(N == total number of nodes)
```

```
  {
```

```
    if this solution is better than optimal solution
```

```
      replace this solution to optimal solution;
```

```
  }
```

```
  Else
```

```
  {
```

```
    Find the power set of available channel of N;
```

```
    For each set S of power set, which  $|S| < q_N$ 
```

```
      Record the neighbor's available channel set condition;
```

```
      Disable the neighbor's intersection available channel;
```

```
      Resource_Allocation_Optimization(N+1);
```

```
      Recovery the neighbor's available channel set condition;
```

```
  }
```

```
}
```

```
Main()
```

```
{
```

```
  Resource_Allocation_Optimization(Node 0);
```

```
}
```

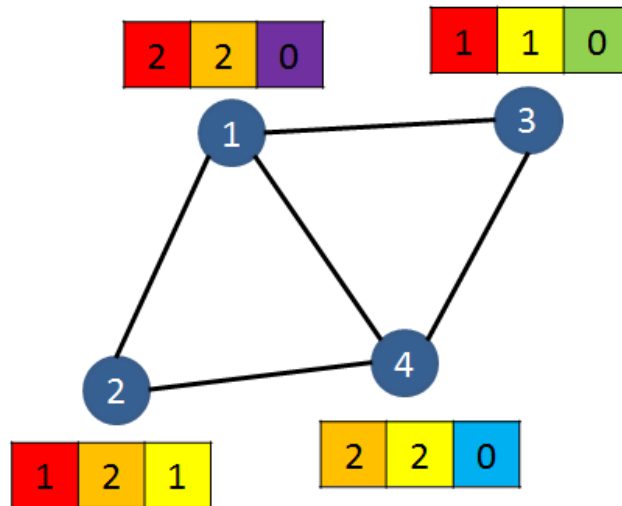
3.2.1.1 理論與證明

- 理論 2. 最佳化頻譜資源分配法能搜尋所有可能的頻譜資源分配方式
證明:

最佳化頻譜資源分配法首先處理點 0，最佳化頻譜資源分配法基於點 0 可用頻譜集合找尋所有點 0 小於其需求數的可能頻譜分配情形，對於每種分配情形我們都相先禁止鄰居被賦予相同頻譜(直接修改鄰居的可用頻譜集合)而繼續，對於每種點 0 的可能分配情形我們都將接續著處理點 1，依此類推...直到點 N(假設 N 為所有點數)，因此，每一點都是基於其上一點跑出的可能性修改頻譜可用集合來進行接下來所有可能性的搜尋，然而上一點也找尋了所有可能分配情形，因此，最佳化頻譜資源分配法搜尋了所有可能的頻譜資源分配方式。

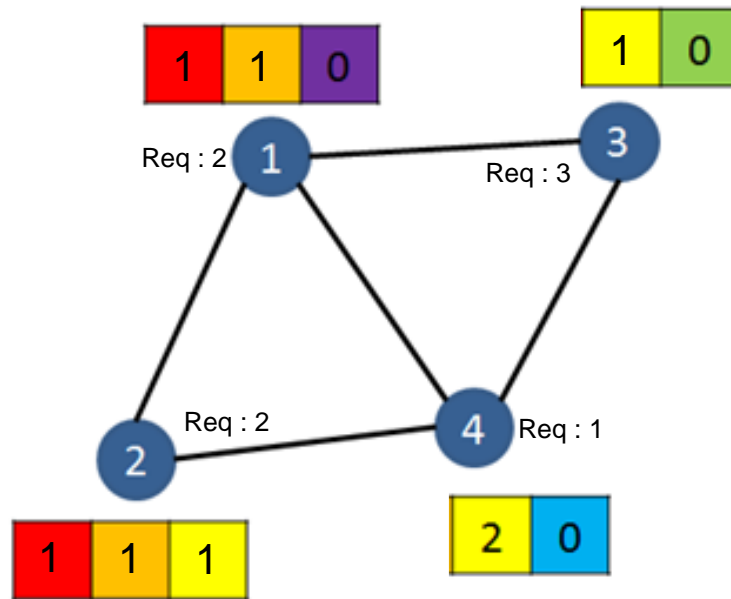
3.2.2 啟發式頻譜資源分配法介紹

我們針對我們的頻譜資源分配目標設計了啟發性的頻譜資源分配法，其概念為在盡可能的滿足頻譜需求數的前提之下找尋公平的分配頻譜資源方式，其運作流程包含了選點及配置頻譜兩步驟。在這邊我們提出了許多選點的頻估的方式，第一個為”最小干擾值”(Minimum interference value)，干擾值定義為頻譜影響的鄰居總數，而最小干擾值為該點所擁有的最小干擾值，舉例來說(如圖十七)，圖十七點 1 的紅色頻譜若配與將影響點 2 及點 3，其干擾值為 2，而最小干擾值為 0 是因為其紫色頻譜若配與將不會影響到任何人因此點 1、2、3、4 的最小干擾值分別為 0、1、0、0，我們希望在環境中優先被置擁有最小干擾值的頻譜。



圖十七、干擾值定義

若存在多個具備有最小干擾值的點我們將進一步比較”頻譜飽和狀況”(saturation degree)，頻譜飽和狀況定義為所剩能被配與的可用頻譜總數，我們期望給擁有較少可用頻譜的使用者優先配置的權力。若有存在多個點具備相同的頻譜飽和狀況我們將比較誰能帶給網路最公平的結果(也就是導致公平值(satisfaction ratio)的變異數最低)，若是前述條件都相同，我們最後將隨機取出一點配置，舉例來說，在圖十八的環境中首先選擇尚未被著成黑色的點，由圖可知有 1、2、3、4 四點，接著再由四點中首先選擇擁有最小干擾值的點，為 1、3、4 三點擁有最小干擾值 0，而其中 3、4 有較小的頻譜飽和狀況(剩下兩條可用頻帶)，我們接著觀察 3、4 兩點的公平性，發現選擇點 3 可使整體公平性上升(公平值變異值最低)，我們將選擇點 3 配置頻譜，而配置頻譜的部分，我們將配置擁有最小干擾值的頻譜(延續上述例子我們將配與點 3 綠色)，並且在配置頻譜過後進行鄰居的干擾值更新，詳細的步驟請參考虛擬程式碼。



圖十八、啟發式頻譜分配演算法環境示意圖



Input: $G = (V, E)$, $R(v)$ number of request

OurAlgorithm

```
{
    // Initialization
    Color all nodes white color;
    Set interference value;
    // Afterward Step
    do{
        choose the node has (0) white color (1) min interference value (2) min saturation
            degree (3) cause min satisfaction ratio variance (4) random choosed ;
        AllocateResource(v);
    }until the colors of all nodes are black;
}
```

AllocateResource(v)

```
{
    choose the active channel form available channel set of  $v$  which has min
    interference value;
    if there's no channel , color the node black color and return;
    allocate the active channel "CH" to  $v$ ;
    disable the "CH" of the neighbor( $v$ ) and renew the interference value of
    neighbor(neighbor( $v$ ));
    if( $R(v) ==$  number of allocated channel)
        disable the rest channel of  $v$ ;
        renew the interference value of neighbor( $v$ );
        color the node black color and return;
}
```

3.2.3 時間複雜度分析

這一小節，我們比較最佳化頻譜資源分配法以及啟發式頻譜資源分配法的時間複雜度，首先我們觀察最佳化頻譜資源分配法。

我們假設點 1 至點 i 的可用頻譜數分別為 C_1, \dots, C_i ，其需求頻譜數為 q_1, \dots, q_i ，對於每個點我們都將搜尋其可用頻譜所有可能性，因此其時間複雜度將不大於

$$2^{\min(C_1, q_1)} * 2^{\min(C_2, q_2)} * \dots * 2^{\min(C_i, q_i)} = 2^{\min(C_1, q_1) + \dots + \min(C_i, q_i)}$$

因此最佳化頻譜資源分配法的時間複雜度為 $O(2^{\min(C_1, q_1) + \dots + \min(C_i, q_i)})$

而啟發式演算法的時間複雜度為 $O(Q*(N+C))$ ，其中 Q 為環境中的需求總數， N 為點數， C 為可用頻譜字集合的頻譜個數，針對每個需求我們都將選點、選色，因此為 $Q*(N+C)$ 。

表二、資源分配演算法時間複雜度分析

Heuristic method	$O(Q*(N+C))$
Optimal method	$O(2^{\min(C_1, q_1) + \dots + \min(C_i, q_i)})$

第四章 模擬數據結果與分析

本章節中，我們比較各種分群以及資源分配方式，以 C++ 開發模擬程式，我們假設環境中每個無線感知網路基地台皆存在相同的 3 條可用頻譜集合，每個基地台的頻譜需求為 1~3 呈現 Uniform distribution，而啟發式分群法及啟發式頻譜資源分配法數據結果皆是執行 100 次取平均。以下我們考量不同點數時方行佈件圖的模擬結果，我們觀測並分析以下數據。

1. # of utilized VMs，不同分群方法所使用的伺服器個數。
2. Handled loading (mean, standard deviation)，不同分群法中伺服器所處理的平均工作負載量及其變標準差。
3. Total penalty，不同分群法中邊界無線感知網路基地台所產生的代價值總合。
4. Average of sum penalty，不同分群法中平均每一群所產出的代價值。
5. Number of non-satisfied requests，不滿足的頻譜需求總數。
6. Satisfaction ratio variance，資源分配後的公平值變異值。
7. Fairness index，公平指數，定義為 $\frac{(\sum_{i=1}^q \frac{b_i}{a_i})^2}{q[\sum_{i=1}^q (\frac{b_i}{a_i})^2]}$ 其中 q 為總點數 a_i 及 b_i 分別為點 i 的需求數及被配置頻譜數。

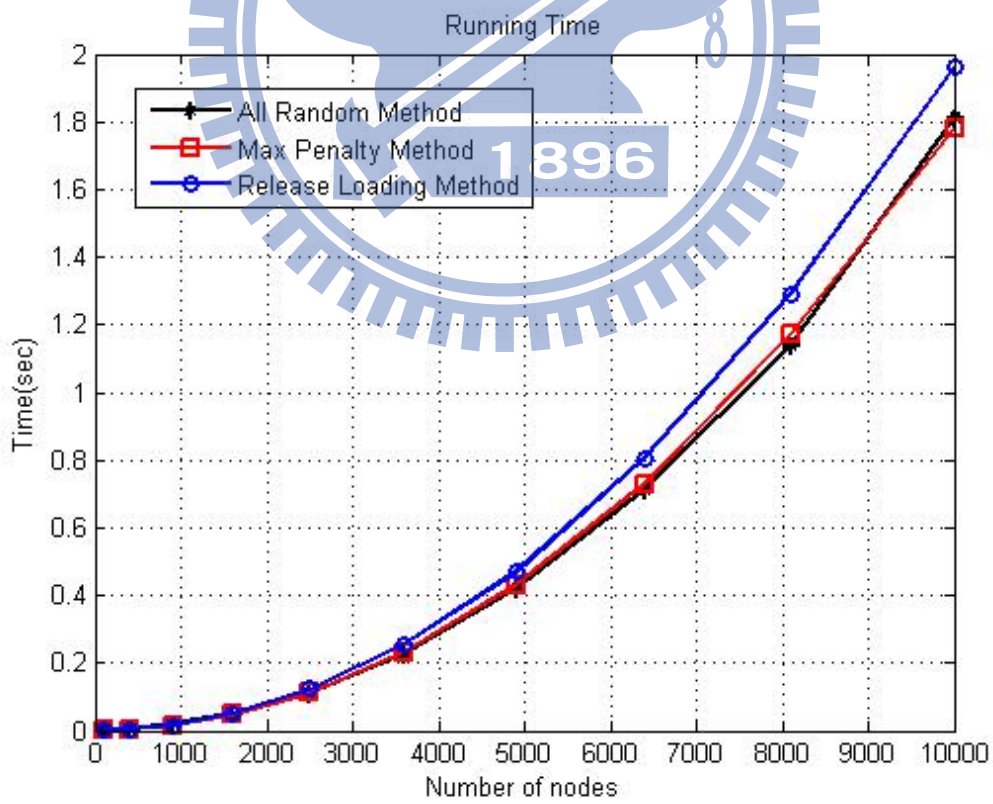
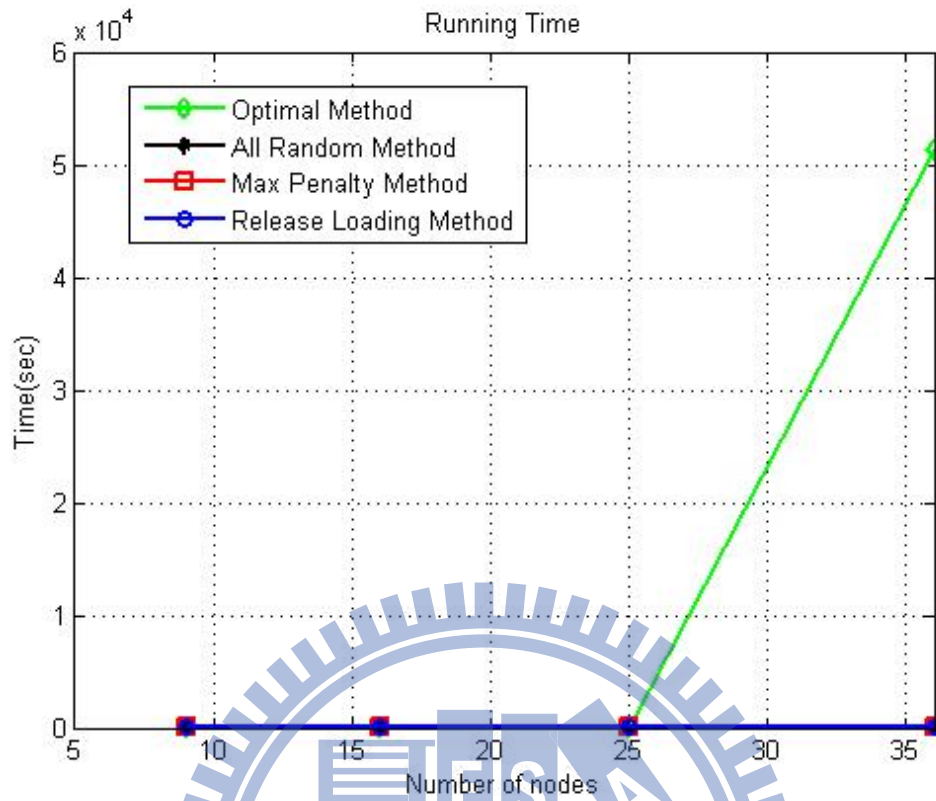
表三為我們考慮 9 個點與 16 個點數時的方行佈件環境分別跑 100 種可能性取平均的結果，我們可以發現最佳化的分群方式在 9 個點與 16 個點數的前提下提供了最少的使用伺服器數，最佳的總代價、平均代價產出，但在工作量負載變異值上在 9 個點時卻略遜於負載平衡分群法，其原因在於最佳化分群法目標於最佳化代價總產出的前提下最佳化工作量負載變異值，因此相較於著重在平衡負載的負載平衡分群法，最佳化分群法有時將產生較大的工作量差異，我們進一步觀察執行時間的變化，在圖十九上

面的圖中我們可以觀察道最佳化演算法在點數大於 25 後執行時間呈指數型成長，在 36 個點的環境下已經需要將近半天的處理時間，進一步看圖十九下面的圖(因為時間因素在這張圖中不比較最佳化分群法)，我們可以看到，當點數逼近 10000 時，我們所提出的啟發式演算法在實做上皆能在 2 秒內完成，這也說明了在實做上，我們所提出的啟發式演算法是相當適用的。

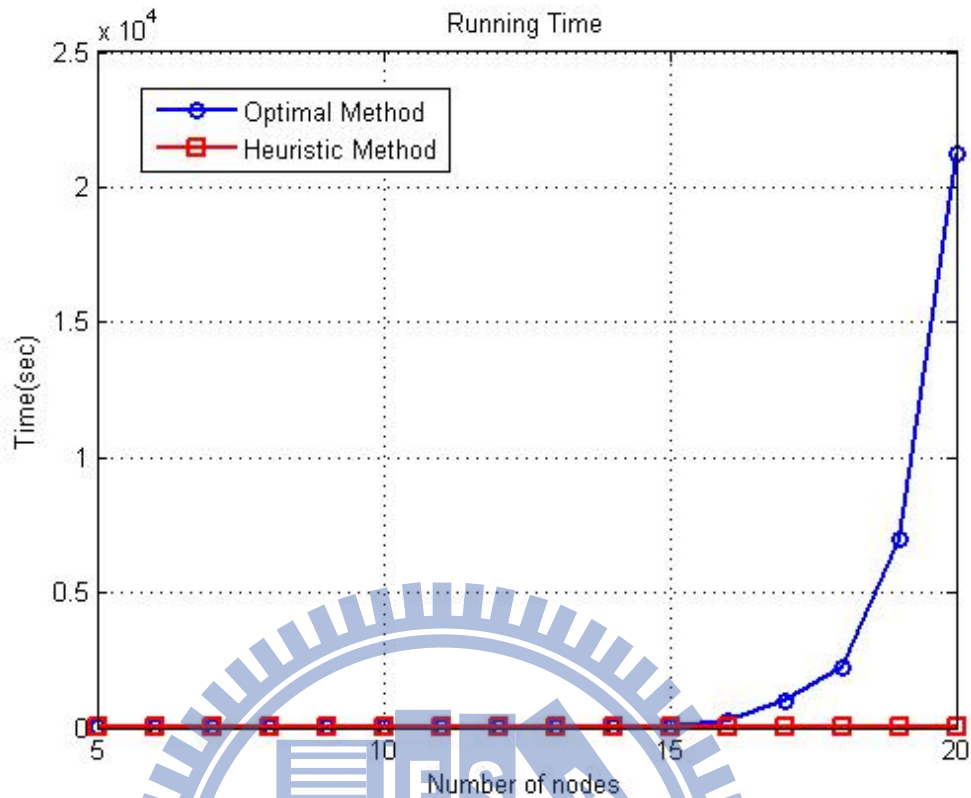
接著看到表三資源分配演算法的部分，我們可以看到最佳化資源分配法在平均頻譜利用次數、公平性變異值、公平指數皆有最佳的表現，然而我們也觀察到，啟發式頻譜分配法所產生的效能離最佳化只有些微的差距，但是在運算時間上卻相差甚遠(如圖二十)，最佳化資源分配演算法的執行時間在點數大於 15 時便超過 FCC 所設定的時間限制(2 秒)並且所花費的時間隨著點數的增加呈現指數型的成長。

表三、含最佳化的比較數據一覽

9 Nodes										
Scenario 1: max VM loading is 35, (mean, standard deviation) of CR AP (100graph)average. loading is (5.93, 2.28)										
	# of utilized VMs	Handled loading (mean, standard deviation)	Total Penalty	Average of sum penalty	# of non-satisfied requests		Satisfaction ratio variance		Fairness index	
					OPT	Heuristic	OPT	Heuristic	OPT	Heuristic
Optimal	2.2	(25.3375,3.97074)	3	1.40267	7.15	7.2919	0.171069	0.241628	0.941916	0.886179
Random	2.31	(24.0225,7.92598)	4.08	1.74833	7.67	7.8423	0.185321	0.282276	0.935635	0.85
Max Penalty	2.34	(23.7925,8.07833)	4.05	1.71	7.47	7.6486	0.176932	0.267223	0.939108	0.859613
Release Loading	2.07	(26.125,2.56758)	4.67	2.23	7.67	7.8674	0.195488	0.305509	0.932609	0.842392
16 Nodes										
Scenario 1: max VM loading is 35, (mean, standard deviation) of CR AP (100graph)average. loading is (6, 2.22)										
	# of utilized VMs	Handled loading (mean, standard deviation)	Total Penalty	Average of sum penalty	# of non-satisfied requests		Variance		Fairness index	
					OPT	Heuristic	OPT	Heuristic	OPT	Heuristic
Optimal	3.2	(29.75,1.74853)	5.4	1.66667	9	9.49	0.19145	0.379159	0.947551	0.799302
Random	4.2	(23.25,9.56668)	7.4	1.73667	7.2	7.904	0.20711	0.330161	0.934111	0.829911
Max Penalty	3.6	(27.29,6.42855)	7.2	2.02333	8.4	8.984	0.197745	0.387266	0.94437	0.782208
Release Loading	3.2	(29.75,3.37637)	9	2.78333	7.6	8.194	0.202722	0.317974	0.942297	0.856788



圖十九、分群法執行時間比較圖(上包含最佳化，下不包含最佳化)



圖二十、頻譜資源分配法執行時間比較圖

因為時間複雜度的因素，我們在接下來的數據中拿掉最佳化演算法部分，我們單純比較啟發式演算法在點數增多時各種數據的表現狀況，以下我們觀察由 25 個點到 100 個點的環境，同樣的每個啟發式演算法我們皆跑 100 次取其平均值。如表四，負載平衡分群法不意外的不論是在伺服器使用量或是在工作負載變異值上都有較佳的數據表現，我們觀察到最大代價演算法在所有觀察環境中皆提供了最好的總代價及平均代價產出但在使用伺服器數量上大於負載平衡分群法，而隨機分群法的數據表現則趨於中間。看到啟發性的頻譜資源分配法，在點數上升時平均頻譜利用次數也跟著上升，有趣的是公平性變異值與公平指數卻不隨點數變化，有著穩定的表現。

表四、不含最佳化的比較數據一覽

25 nodes				
Scenario 1: max VM loading is 35, (mean, standard deviation) of CR AP (100graph)average. loading is (6.06, 2.37)				
	# of utilized VMs	Handled loading (mean, standard deviation)	Total Penalty	Average of sum penalty
Random	6.15	(24.6407,10.1919)	17.1	2.79936
Max Penalty	6.07	(24.9857,10.1574)	16.57	2.73388
Release Loading	5.41	(27.8438,6.3583)	18.68	3.40401

64 nodes				
Scenario 1: max VM loading is 35, (mean, standard deviation) of CR AP (100graph)average. loading is (6, 2.43)				
	# of utilized VMs	Handled loading (mean, standard deviation)	Total Penalty	Average of sum penalty
Random	15.74	(24.314,11.5046)	53.75	3.41747
Max Penalty	15.35	(25.0252,11.167)	51.48	3.3528
Release Loading	14.01	(27.3056,9.22717)	55.55	3.95487

100 nodes				
Scenario 1: max VM loading is 35, (mean, standard deviation) of CR AP (100graph)average. loading is (5.98, 2.44)				
	# of utilized VMs	Handled loading (mean, standard deviation)	Total Penalty	Average of sum penalty
Random	24.13	(24.9444,11.2077)	92.16	3.8188
Max Penalty	23.97	(25.1428,11.4176)	87.16	3.63982
Release Loading	22.52	(26.7216,10.58)	92.5	4.10234

第五章 結論與未來發展目標

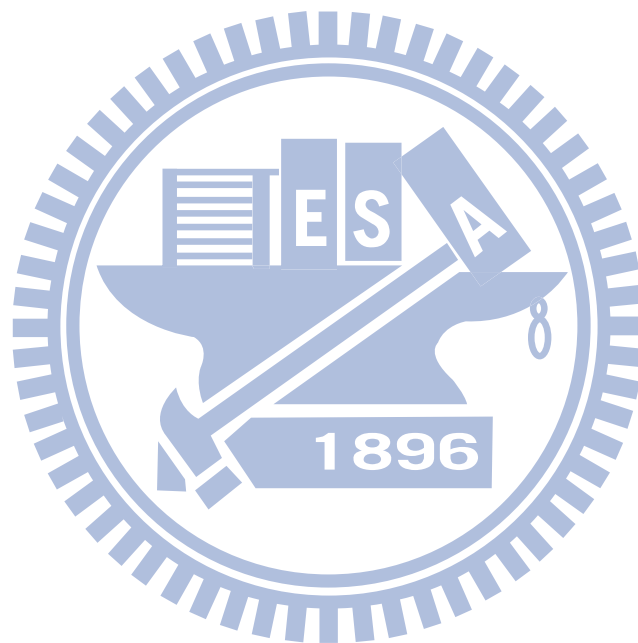
在這篇論文中，我們介紹無線感知雲端網路的概念，並且對其發展可能性及架構做了初步發想與介紹，其中我們針對在無線感知雲端上頻譜管理進行研究，希望藉由地區的平行化，加速整體在雲端上的運作速度，我們在本篇論文中，探討各種分群的可能(最佳化分群、最大代價分群及平衡負載分群)，提出了理論證明、時間複雜度分析及模擬數據結果。對於分群過後因為資訊缺乏的原因，可能導致邊界無線感知網路基地台的彼此干擾的問題，我們進一步的提出解決方案，將邊界無線感知網路基地台做初步的頻譜資源分配，本篇論文提出兩種頻譜資源分配方法(最佳化資源分配，啟發性的資源分配)以理論證明其最佳性，以時間複雜度及模擬數據結果分析並驗證其可行性。

在未來，我們將著手於實做無線感知雲端網路，將分群法及邊界的資源分配演算法實做於真實的雲端平台，並結合功率配置來完整實做頻譜管理運算單元於雲端上，更進一步的在雲端上結合頻譜感測及換手管理，期望將無線感知雲端網路的概念實做於生活之中，為未來的生活型態提出新的可能。

參考文獻

- [1] FCC, Spectrum Policy Task Force Report, ET Docket No. 02-155, Nov. 2002.
- [2] J. Mitola and G. Q. Maguire, "Cognitive radios: making software radios more personal," IEEE Personal Communication Magazine, Vol. 6, pp. 13-18, Aug. 1999.
- [3] E. Visotsky, S. Kuffner, and R. Peterson, "On collaborative detection of TV transmissions in support of dynamic spectrum sensing" Proceedings of IEEE Symposium on Dynamic Spectrum Access Networks (DySpan) 2005, pp. 131-136, Nov. 2005.
- [4] A. K. Kattapur, A. T. Hoang, Y.-C. Liang, and M. J. Er, "Data and decision fusion for distributed spectrum sensing in cognitive radio networks," Proceedings of IEEE International Conference on Information and Communications Security (ICICS) 2007, pp. 1-5, Oct. 2007.
- [5] Viktoria Fodor, Ioannis Glaropoulos and Loreto Pescosolido, "Detecting low-power primary signals via distributed sensing to support opportunistic spectrum access," Proceedings of IEEE International Conference on Communications (ICC) 2009, pp. 2929-2934, June 2009.
- [6] S.-H. Wu, H.-L. Chao, C.-T. Jiang, S.-R. Mo, C.-H. Ko, T.-L. Li, C.-F. Liang, and C.-C. Cheng, "A conceptual model and prototype of cognitive radio cloud networks in TV white spaces," Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), Wireless Cloud and White Space Oriented Networks workshop, pp. 425-430, Apr. 2012.
- [7] S.-H. Wu, H.-L. Chao, C.-H. Ko, S.-R. Mo, C.-T. Jiang, T.-L. Li, C.-F. Liang, and C.-C. Cheng, "A cloud model and concept prototype for cognitive radio networks," IEEE Wireless Communications Magazine, vol. 19, pp. 49-58, 2012.
- [8] H. Harada, H. Murakami, K. Ishizu, S. Filin, Y. Saito, H. N. Tran, G. Miyamoto, M. Hasegawa, Y. Murata and S. Kato, "A software defined cognitive radio system: cognitive wireless cloud", Proceedings of IEEE Global Telecommunications Conference (Globecom) 2007, pp. 294-299, Nov. 2007.
- [9] D-H. Huang, S.-H. Wu, and P.-H. Wang, "Cooperative spectrum sensing and locationing: A Sparse Bayesian Learning Approach," Proceedings of IEEE Global Telecommunications Conference (Globecom) 2010, pp. 1-5, Dec. 2010.
- [10] B.W. Kernighan, S. Lin, "An efficient heuristic procedure for partitioning graphs," Bell System Tech. Journal, vol. 49, pp.291-307, Feb. 1970.
- [11] C. M. Fiduccia and R. M. Mattheyses. "A linear-time heuristic for improving network partitions", Proceedings of IEEE Design Automation Conference (DAC) 1982, pp 174-181, June 1982.

- [12] S. T. Barnard, H. D. Simon, “A fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems,” Proceedings of SIAM Conference on Parallel Processing for Scientific Computing, pp. 711-718, 1993.
- [13] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar, “Multilevel hypergraph partitioning: application in VLSI domain,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol 7, No. 1, pp 69-79, 1999.
- [14] P. Chardaire, M. Barake, and G. P. McKeown, “A PROBE-based heuristic for graph partitioning,” IEEE Transactions on Computing., Vol. 56, No. 12, pp. 1707–1720, Dec. 2007.
- [15] Daniel Marx, “Graph coloring with local and actions global constraints,” PhD thesis, Department of Economics, Budapest University of Technology, 2004.



附錄一、工作量產生方式>Loading Generation)

Power allocation engine loading :

M : Number of channels

L : Number of APs

k : Number of users

(A) Optimal power allocation :

If $k < ML$, $f(1)=1$, $f(2)=2$

$$f(k) = 1 + C_2^k f(k-2) + C_3^k f(k-3) + \dots + C_{k-1}^k f(1) + 1$$

Computation complexity = $f(k) * O(L^3)$

(B) Suboptimal power allocation :

If $k \leq ML$,

$$\text{Computation complexity} = (ML + (k-1)(ML)^2 - \frac{1}{2}ML(k^2 - k)) * O(L^3)$$

詳細內容請參閱 C.-C. Cheng, "Feasibility Assessment for A Cloud-Based Cognitive Radio Mobile Network - A Power Perspective," Master thesis, Institute of Communications Engineering, National Chaio Tung University, 2012.