** **

**資訊工程研究所**

**碩士學位論文**

**再探資料廣播環境中關於道路網路上最近個鄰居搜尋的方法**

**Revisiting the NN Search Protocols on Road Networks in Data Broadcasting Environments**

**研究生：張中維**

**指導教授：劉傳銘 博士**

**中華民國一百零五年八月**

|  |  |
| --- | --- |
| **資訊工程系碩士班** | **國立臺北科技大學** |
|  |  |
| **編號：103598030** | **碩士論文** |
|  |  |
| **再探資料廣播環境中關於道路網路上最近個鄰居搜尋的方法** | |
|  | |
| **研究生：張中維** | |

摘要

論文名稱：再探資料廣播環境中關於道路網路上最近個鄰居搜尋的方法

頁數：四十五頁

校所別：國立臺北科技大學 資訊工程研究所

畢業時間：一百零四學年度 第二學期

學位：碩士

研究生：張中維

指導教授：劉傳銘 博士

關鍵詞：資料廣播、道路網路、最近個鄰居搜尋

在現今的資訊服務中，如何提供一個能同時處理大量查詢的適地性服務系統是日益重要的議題。由於利用資料廣播模式可以減少伺服端在頻寬、運算量的負擔以達成同時服務大量使用者的目的，基於這個理由，許多論文研究廣播環境中的適地性查詢，而在為數眾多適地性查詢裡，道路網路上的最近個鄰居搜尋因為能讓使用者快速獲取附近的真實景點資訊而被大量使用。因此，本篇論文探討在廣播環境中關於道路網路上的最近個鄰居查詢方法，並提出有效且快速的廣播協定來解決。在廣播伺服端方面，我們將道路網路資訊利用網格切割法分割成數個區塊，接著預先計算每個邊界點在區塊裡的最近景點距離後加入索引結構；在客戶端方面，我們提出相應的接收方式，利用所有跨區塊的路徑都必須經過邊界點的特性，精準過濾大量不需要的區塊資訊以有效減少客戶端的監聽時間與查詢經歷時間。為了證明方法的可行性，我們使用真實道路資料作為材料，實驗搜尋協定在不同參數背景中的效能，並與相同環境的其他論文進行比較。實驗結果顯示在較高的切割區塊數中，客戶端能更精準的排除不需要的區塊，因此監聽時間與查詢經歷時間都能有較好的表現。

ABSTARCT

Title：Revisiting the NN Search Protocols on Road Networks in Data Broadcasting Environments

Pages：45

School：National Taipei University of Technology

Department：Computer Science and Information Engineering

Time：August, 2016

Degree：Master

Researcher：Chung-Wei Chang

Advisor：Chuan-Ming Liu Ph.D.

Keywords：Data Broadcasting, Road Networks, NN Search

Providing a location-based system which can serve lots of query at once is a important issue in nowadays. Broadcast environment can greatly reduce server’s burden in bandwidth and computation. As a result, there are plenty of research discussing location-based system in broadcast environments. In the other hand, NN search in road networks is one of the most popular query in numerous of location-based service since NN search allows user to quick access nearby geographic information. Due to reasons above, this paper focus on discussing NN search in road networks on broadcast environments and propose a fast and effective broadcasting protocol to resolve. In the server side, we first use grid partition to divide the whole road network into few regions and precompute nearest neighbor of each border node in its located region then add the information into index; In the client side, we observed a property in road network that every path that across multiple regions must pass through border nodes, according to this observation, we proposed a corresponding protocol to receive index, which could precisely prune unnecessary regions, and processNN search efficiently. To prove the feasibility of the proposed method, we use real road network as material to experiment performance of our method and compare with other methods. Experiment result shows that our method will have better tuning time and latency in higher partitioning region number.

誌謝

時光飛逝，轉眼間碩士生活已經進入尾聲。在這兩年中，我能感受到自己的心態更加成熟，在專業技巧方面也有顯著的成長。首先必須感謝台北科技大學給我的栽培，學校提供了許多優良的課程供我挑選、學習，每個課程都需要大量的時間準備與消化，如此紮實的課程讓我學習了很多新知識，更磨練了自己解決問題的能力。除此之外，實驗室的同學：奕安、翔普、煒鵬、怡玫、宣芸可以與我在開心的時候開懷同樂，難過的時候一起解決遇到的關卡，有了他們，我的碩士生活才能如此的多采多姿。還需要感謝學長姐：傳淇、冠登、柏辰、育德、軒尉、玉玲的幫助，不吝嗇的將自己的經驗傳授給我，令我受益良多。而學弟妹：天駿、琳茹、靖翔在口試時提供的幫助。

我也必須要感謝指導教授劉傳銘老師給我的教導，老師總是以詼諧語幽默的方式，悉心指導與督促每一位實驗室的同學，在遇到困難時，老師也總是能一針見血的找到問題點，並提供我良好的建議與方向。也感謝口試委員王正豪老師、余征武老師在口試時給我的寶貴建議與一些完善論文的技巧，實在受用無窮。

最後，我必須感謝我的家人給予我的支持，在我迷茫的時候指引我，在我難過時陪伴我，讓我能夠順利的度過一次次的困難又一次次的成長。期許自己在未來的日子裡能更加茁壯，並將自己的所學所聞貢獻給所有需要幫助的人。

# 目錄

[摘要 i](#_Toc457834986)

[ABSTARCT ii](#_Toc457834987)

[誌謝 iv](#_Toc457834988)

[目錄 v](#_Toc457834989)

[圖目錄 vi](#_Toc457834990)

[第一章 緒論 1](#_Toc457834991)

[第二章 相關背景 4](#_Toc457834992)

[2.1資料廣播環境 4](#_Toc457834993)

[2.2歐基里得平面上的最近個鄰居查詢 6](#_Toc457834994)

[2.3道路網路的最近個鄰居查詢 9](#_Toc457834995)

[2.4在資料廣播環境中的適地性道路網路查詢 12](#_Toc457834996)

[第三章 Border-based Approximation 14](#_Toc457834997)

[3.1切割道路網路 14](#_Toc457834998)

[3.2事前運算與建立索引結構 16](#_Toc457834999)

[3.2.1實際距離與歐幾里得距離比例逼近法 18](#_Toc457835000)

[3.2.2邊界點最短路徑圖 21](#_Toc457835001)

[3.2.3邊界點最近鄰居距離表 22](#_Toc457835002)

[3.2.4 BA索引結構 24](#_Toc457835003)

[3.3 26](#_Toc457835004)

[第四章 模擬實驗 33](#_Toc457835005)

[第五章 結論 43](#_Toc457835006)

# 圖目錄

[圖1 道路網路上尋找最近兩個商店範例 1](#_Toc457835007)

[圖2 資料廣播模式示意圖 3](#_Toc457835008)

[圖3 (a)無索引輔助廣播模式， 5](#_Toc457835009)

[圖4 (1,*m*)索引廣播模式[4] 6](#_Toc457835010)

[圖5 分散式索引廣播模式(Distributed Indexing)[4] 6](#_Toc457835011)

[圖6 在歐基里得平面中，以點為查詢點的，其結果為 7](#_Toc457835012)

[圖7 Voronoi Diagram 8](#_Toc457835013)

[圖8 在道路網路中，以點為查詢點的，其結果為 10](#_Toc457835014)

[圖9 NVD圖，左圖為圖資物件與道路，右圖經虛線切割的區塊表示NVC範圍 12](#_Toc457835015)

[圖10 *Network Partition Index* (NPI)索引結構[17] 13](#_Toc457835016)

[圖11 切割道路網路成個區塊範例 15](#_Toc457835017)

[圖12 比例逼近法範例 (a)以為範例，，，(b)， 21](#_Toc457835018)

[圖13 邊界點最短路徑圖 22](#_Toc457835019)

[圖14 邊界點最近鄰居距離表範例 23](#_Toc457835020)

[圖15 索引結構 24](#_Toc457835021)

[圖16 參數名稱與參數意義表格 25](#_Toc457835022)

[圖17 計算查詢距離上限演算法 29](#_Toc457835023)

[圖18 計算每個區塊距離下限演算法 29](#_Toc457835024)

[圖19 BA搜尋協定演算法 30](#_Toc457835025)

[圖20 BA搜尋協定範例 (a)道路完整資訊 (b)索引結構 31](#_Toc457835026)

[圖21 BA範例資訊、、 32](#_Toc457835027)

[圖22 連江縣 33](#_Toc457835028)

[圖23連江縣在不同區塊數量下20個物件的最近5個鄰居搜索的廣播週期 35](#_Toc457835029)

[圖24連江縣在不同區塊數量下20個物件的最近5個鄰居搜索的監聽時間 35](#_Toc457835030)

[圖25 連江縣在不同區塊數量下20個物件的最近5個鄰居搜索的查詢經歷時間 35](#_Toc457835031)

[圖26連江縣在不同物件數量下切割64區塊的最近5個鄰居搜索的廣播週期 36](#_Toc457835032)

[圖27連江縣在不同物件數量下切割64區塊的最近5個鄰居搜索的監聽時間 37](#_Toc457835033)

[圖28連江縣在不同物件數量下切割64區塊的最近5個鄰居搜索的查詢經歷時間 37](#_Toc457835034)

[圖29連江縣在不同物件數量下切割16區塊的最近5個鄰居搜索的廣播週期 38](#_Toc457835035)

[圖30連江縣在不同物件數量下切割16區塊的最近5個鄰居搜索的監聽時間 38](#_Toc457835036)

[圖31連江縣在不同物件數量下切割16區塊的最近5個鄰居搜索的查詢經歷時間 39](#_Toc457835037)

[圖32 連江縣在不同下切割64區塊的最近5個鄰居搜索的廣播週期 39](#_Toc457835038)

[圖33 連江縣在不同下切割64區塊的最近5個鄰居搜索的監聽時間 40](#_Toc457835039)

[圖34 連江縣在不同下切割64區塊的最近5個鄰居搜索的查詢經歷時間 40](#_Toc457835040)

[圖35連江縣在不同下的最近5個鄰居搜索的廣播週期 41](#_Toc457835041)

[圖36連江縣在不同下的最近5個鄰居搜索的監聽時間 41](#_Toc457835042)

[圖37連江縣在不同下的最近5個鄰居搜索的查詢經歷時間 42](#_Toc457835043)

# 緒論

在現今的蓬勃發展資訊服務中，使用者常使用行動裝置配合全球定位系統(Global Position System, GPS)定位裝置的位置，查詢與使用者相關的地理資訊。由於這些資訊完整的反應真實的道路與地理資訊，使得道路網路上的適地性服務(Location-Based Service, LBS)[1]十分普及。

道路網路上的適地性服務其中一個重要功能是反應查詢點(Query Point)附近的空間資訊，因此能夠快速反應周遭景點的最近個鄰居搜尋(Nearest Neighbor Search, *k*NN Search)[2][3]常被使用。道路網路上的最近個鄰居搜尋可以讓使用者找出距離查詢點最近的附近個景點，其應用範圍極廣，舉例來說：找出距離現在位置最近的個商店或者地標。以圖1為例，每個點皆為道路網路上的商店，假設使用者在點的位置，要尋找距離點最近的兩間商店，利用最近2個鄰居搜尋，使用者很快就能知道距離自己最近的商店為與。

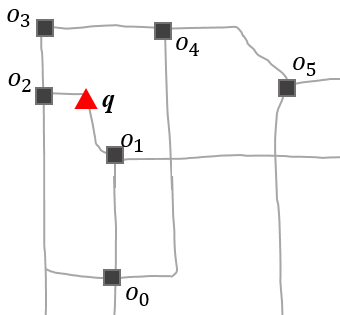


圖1 道路網路上尋找最近兩個商店範例

雖然道路網路上的適地性服務應用廣泛，但配合上傳統服務模式卻會產生許多瓶頸。傳統服務模式如下：由客戶端先發送一個查詢交予伺服端，伺服端接收了查詢之後再去計算查詢結果並傳回客戶端。這種一問一答的模式會造成伺服端要不斷處理重複的查詢，造成許多資源的浪費，除此之外，當大量查詢湧入時，伺服端常常無法負荷大量查詢產生的網路頻寬與計算量上的需求。突破這些瓶頸的方式往往是升級伺服端的設備，不但造成服務的成本提高，也無法根本性的解決問題。

採用資料廣播模式(Data Broadcasting)，如圖2所示，則可以有效的突破傳統服務模式的瓶頸。有別於傳統服務模式，資料廣播模式讓伺服端循環廣播所有客戶端可能需要的資訊，客戶端再從伺服端發出的廣播資料中擷取查詢所需的片段，並於自己的裝置上計算查詢答案，因此不會因使用者過多產生頻寬與計算量上無法負荷的問題。

為了同時處理大量的適地性查詢，本篇論文主要探討在廣播環境中的最近個鄰居搜尋，並提出一個有效的廣播協定來解決。在廣播環境中進行道路網路上的適地性查詢之難點在於伺服端無法事先知道查詢點的位置，而客戶端在下了查詢後也無法隨意取得自己需要的區塊資料。因此，索引結構須具備足夠的資訊讓客戶端精準篩選不需要的區塊，同時，為了降低監聽時間與查詢經歷時間，索引結構也不能佔據太多的廣播空間，而以下是我們提出的解決方案。在廣播伺服端方面，伺服端會先將道路網路利用網格切割法分割成數個區塊，接著預先計算所有邊界點對於相鄰區塊裡的最近景點距離以及每個區塊內部的最大距離後加入索引結構；在客戶端方面，客戶端以相應的方式接收索引，並利用索引內的資料計算出整個查詢的距離上限(Upper Bound)，與每個區塊與查詢點各自的距離下限(Upper Bound)，若區塊的距離下限大於整個查詢的距離上限則代表該區塊不可能成為答案，這些不可能成為答案的區塊就可排除不接收。當客戶端接收完所有可能的區塊後即可算出查詢的答案。



圖2 資料廣播模式示意圖

在模擬實驗中，我們證明了方法的可行性，並使用真實道路資料作為實驗材料，觀察搜尋協定在不同參數背景中的效能與正確性，並與相同議題的其他論文方法進行比較。實驗結果顯示，在較高的切割區塊數中，本篇論文提出的方法能更精準的排除不需要的區塊，因此在監聽時間上與查詢時間上都能有較好的表現。整體來說，本篇論文的貢獻可以分為以下幾點。

* 我們提出了Border Approximation Method(BA)有效解決廣播環境中道路網路上最近個鄰居搜尋之問題，並設計新穎的索引結構讓客戶端更準確地篩選出所需的區塊，節省客戶端接收與計算非必要區塊的時間。
* BA能夠使客戶端動態的更新查詢上限值與下線值，讓客戶端能夠隨著接收的區塊資料越多而有越精準的邊界值。
* 在模擬實驗中我們詳細的分析BA與其他方法的優缺點，從模擬實驗的結果中我們能看到BA在區塊分割數較高的情況下，監聽時間與查詢經歷時間接會有較好的表現。

本論文分為五個章節，從第二章節開始，我們會說明資料廣播環境、最近鄰居搜尋、道路網路模型等背景，並且介紹這些議題上的相關研究。第三章會介紹本篇論文所提出的Border Approximation Method之索引結構與搜尋協定，以及設計此方法的概念。在第四章，我們會利用模擬實驗來分析Border Approximation Method的優缺點，並與其他相關方法比較，而在第五章，我們會總結整篇論文。

# 相關背景

現在的適地型服務常與道路網路結合，讓使用者能夠了解真實道路的地理資訊。有別於歐幾里得距離(Euclidean Distance)以直線距離作為兩點之間的距離，真實道路資料上兩點之間的距離則為兩點間連通線段的長度總和而成，因此真實道路資料的適地性服務較為複雜。近年來，為了符合大量使用者對於道路網路上適地性服務的需求，已有研究專注於廣播環境中道路網路的適地性查詢。本章節中將介紹廣播環境、最近個鄰居查詢、道路網路以及相關研究的方法。

## **2.1**資料廣播環境

在資料廣播環境中，廣播伺服端(Broadcast Server)中包含一個或數個廣播頻道，每個廣播頻道是由多個資料片段所組成，並且以週期性的方式，由伺服端循環的播放給客戶端使用。客戶端再從廣播頻道中聽取所需的片段後進行運算。

不如傳統服務模式中伺服端可以直接回傳精確的答案給客戶端，客戶端要從廣播中自行抓取需要的資料。為了要讓客戶端知道自己所要的資料為何，廣播環境中往往會加入索引結構，這些索引結構通常是預先處理過的資料，讓客戶端能夠篩選可能需要的資料。在客戶端不需要的資料時正在播放時，客戶端裝置便可進入休眠模式直到廣播需要的檔案時再喚醒裝置已降低能耗。

索引結構的優劣十分重要，優良的索引結構可以在容量小的同時排除許多不需要的資訊，減少廣播週期的同時，也可以大幅降低監聽時間與查詢經歷時間。低監聽時間可以減少客戶端能耗，而低查詢經歷時間則可以提高系統的服務品質，而廣播週期的長短會大幅影響監聽時間與查詢經歷時間的好壞。

圖3比較了(a)無索引廣播模式與(b)有索引輔助廣播模式的區別，假設每個資料區塊接收時間都為1，而客戶端所需要的資料為。圖3(a)是一個無索引輔助廣播模式的例子。由於沒有索引的輔助，客戶端不知道自己需要的資料何時播出，因此要每個資料都依序聽取，直到自己需要的資料都拿到為止，因此無索引輔助廣播模式的查詢經歷時間會較高。圖3(b)則是有索引輔助廣播模式的例子。在有索引輔助下，客戶端可以知道自己需要的資料位置何時播出，因此查詢經歷時間較短，但由於在廣播週期其中增加了索引結構，有索引輔助廣播模式的監聽時間較高。

為了擴大索引輔助的優勢，讓客戶端能夠更快速的知道需要的資料位於何處，Imielinski et al.提出了(1,)索引廣播模式((1,) indexing)[4]來降低等待接收索引結構的時間。(1,)索引廣播模式的範例如圖4所示，伺服端將廣播切成段資料，每段資料前面皆放入一個完整的索引結構，也就是索引結構與資料會交錯放入廣播週期中。雖然(1,)索引廣播模式會造成廣播週期大量增加，但交錯放置索引可讓客戶端不需一定要從廣播的開頭聽取資料，有效減少接收索引的等待時間可降低客戶端的監聽時間。分散式索引廣播模式(Distributed Indexing)則是將一個完整的索引結構分成數分並分散放至於資料片段中，而客戶端仍可在任意點聽取資料如圖5所示。

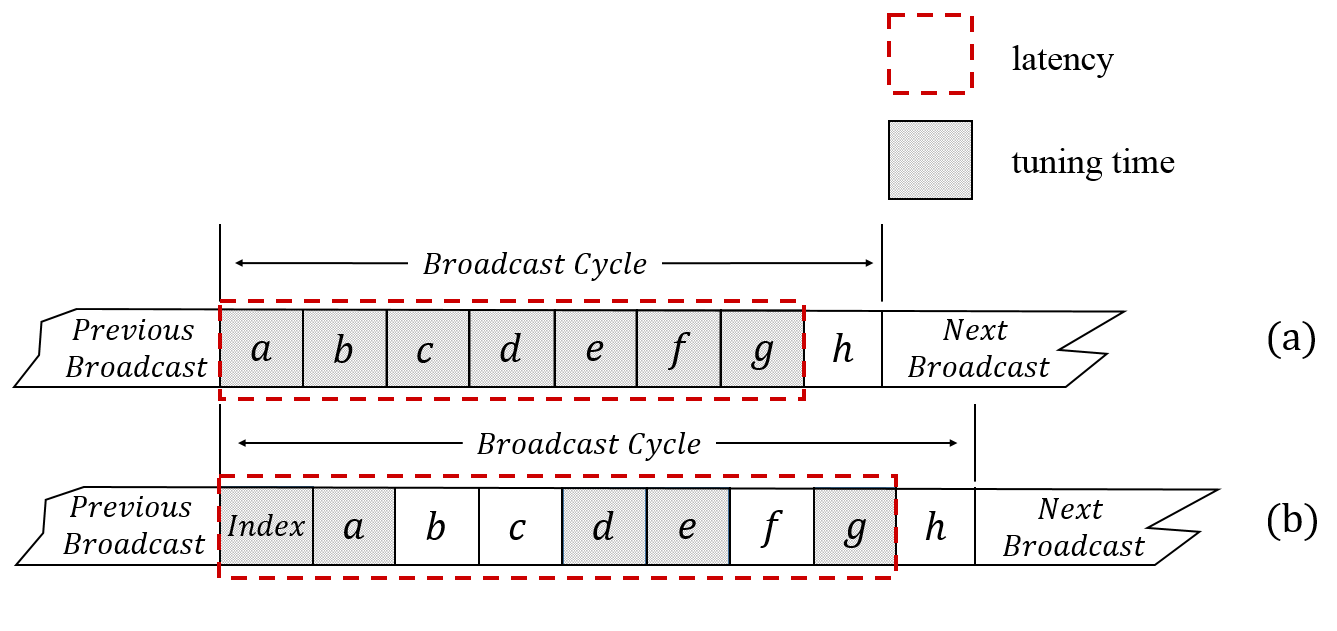


圖3 (a)無索引輔助廣播模式，

(b)有索引輔助廣播模式，

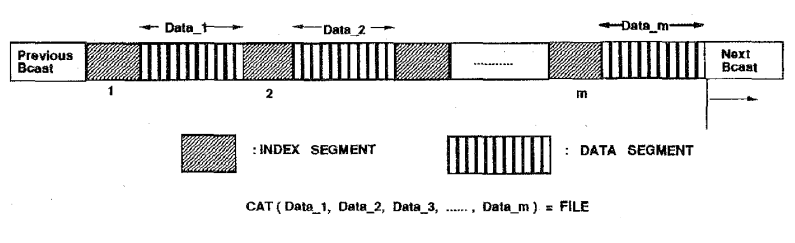


圖4 (1,*m*)索引廣播模式[4]

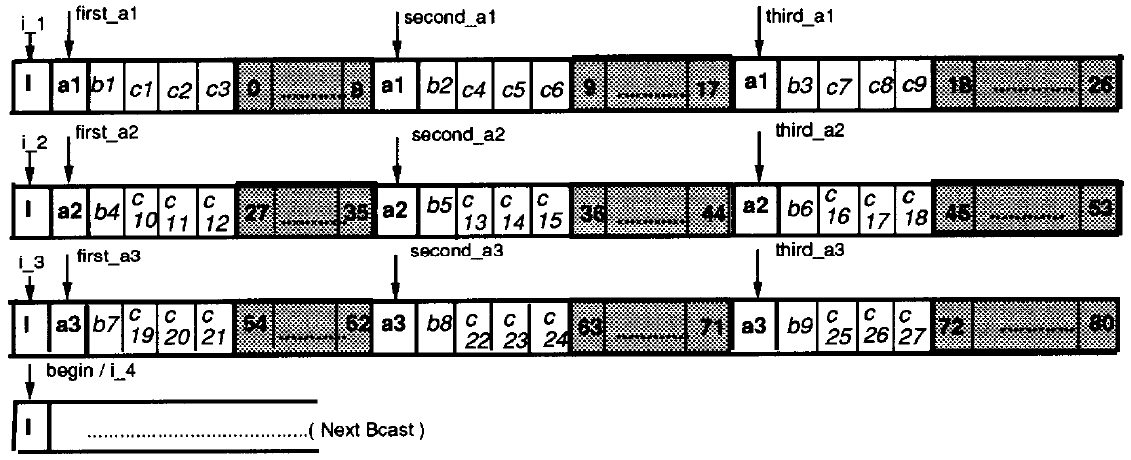


圖5 分散式索引廣播模式(Distributed Indexing)[4]

## **2.2**歐基里得平面上的最近個鄰居查詢

過往在探討最近個鄰居查詢問題時，都是以歐基里得距離為背景。歐基里得距離是[歐幾里得空間](https://zh.wikipedia.org/wiki/%E6%AC%A7%E5%87%A0%E9%87%8C%E5%BE%97%E7%A9%BA%E9%97%B4)中兩點之間的直線[距離](https://zh.wikipedia.org/wiki/%E8%B7%9D%E7%A6%BB)。由於只要有兩點的座標就可以計算出歐基里得距離，因此建立在這種背景之下的最近個鄰居查詢較為直覺。

假設一個歐基里得平面中有物件個數為的資料集每個資料物件有個維度的資訊，其中。兩物件在空間中的歐基里得距離以表示，定義如定義一。

定義一(歐基里得距離):

在二維平面上給定兩個物件和∊，坐標分別為和，則和之間的歐基里德距離

在最近個相鄰者的查詢中，結果取決於每個點的位置資訊。假設給定一個物件資料集以及一個查詢點，最近個相鄰者查詢結果即為資料集中距離查詢點最近的前個資料物件，最近個鄰居查詢問題如定義二。

定義二 (最近個鄰居查詢):

給定一個查詢點*q*及一個整數*k*，找一個集合且，使得任一資料物件，

圖6是一個歐基里得平面中的例子。以為查詢點，為資料物件，其中距離查詢點最近的是,與三個點，因此查詢點的結果為。

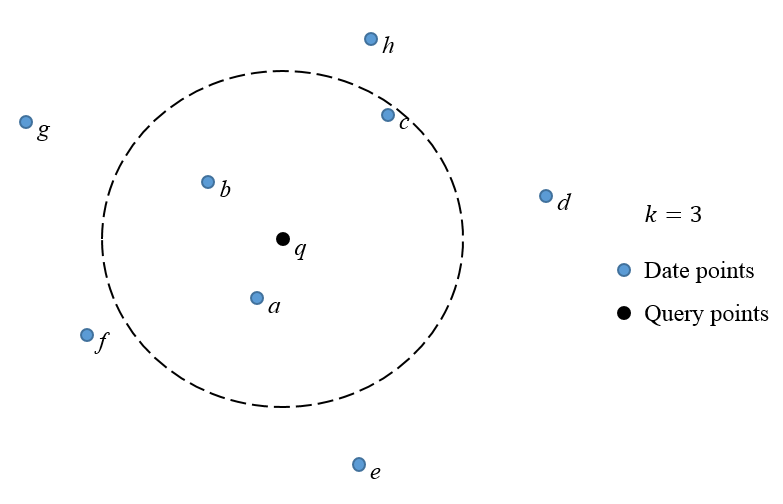


圖6 在歐基里得平面中，以點為查詢點的，其結果為

相關的最近個鄰居搜尋方法如下：[5]提出一個Branch and bound演算法在R-tree[6]上做最近個鄰居搜尋，利用到最小邊界矩形的概念與以下所定義的兩種查詢點到某一最小邊界矩形的距離概念來進行搜尋時的刪除節點，指的是從到的最小距離；指的是從到的每一邊最遠距離中的最小值。給定一個查詢點，演算法會從R-tree的根節點開始搜尋，利用上述的距離概念來刪除不必要搜尋之節點，以增進效能。此種最近相鄰者搜尋法可進一步推至最近*k*個相鄰者搜尋。在此後，利用R-tree做有關於最近相鄰者搜尋的演算法都依循此種設計模式，例如在[7]之中，作者提出用R-tree精準計算出最近個鄰居，並最佳化搜尋演算法，挑戰傳統的以深度優先(Depth-First)的排程方式，並探討不同排程方式的效能。

Voronoi Diagram (VD)是一個查詢歐幾里得平面上最近鄰居的有效的工具。假設在一歐幾里得平面上含有數個物件，則VD為兩兩物件連線之中垂線合併後形成的圖，如圖7所示，點集合為物件集合，每個物件與其他相鄰物件以中垂線劃分，單一物件以其周遭最接近的中垂線線段為邊界形成Cell (Voronoi Cell/VC)。任意點落在其中一個Cell，根據中垂線的特性，該點一定會距離那個Cell內的物件最近。而[8]提出一個有效率的最近個鄰居搜尋協定，在不使用索引結構的實況下，在資料中附加額外的VD鄰居資訊，在搜尋過程中，還能藉此判斷是否能提前結束聽取資料。

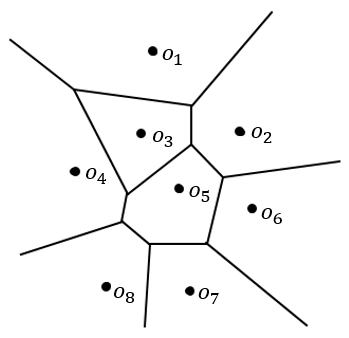
****

圖7 Voronoi Diagram

[9]首先考量在有障礙物的平面空間裡提出一個有效率的最近*k*個鄰居搜尋法，使用網格切割(Grid-Partition)建立索引結構，包含預先計算的障礙物Voronoi Diagram (Obstructed Voronoi Diagram)資訊，提出數個排除方法協助搜尋最近鄰居，再從最近鄰居搜尋延伸至最近*k*個鄰居搜尋。

## **2.3**道路網路的最近個鄰居查詢

道路網路是一個有加權值的有向圖，代表整張道路網路的道路以及節點。代表道路網路中的*節點*集合，節點可能是學校、餐廳、商店等地點。代表道路網路中的道路集合，道路是兩點之間連接的*邊*，每個邊都有自己的*權重*，這些權重可能是道路通行時間的長短也可能是道路的距離。在此篇論文中，*權重*以兩點之間的道路距離表示。

如同先前所介紹的歐基里得平面上的最近個相鄰者查詢問題概念相同，道路往路上的最近個相鄰者查詢以道路網路為基底。也就是在道路網路有向圖中找到距離查詢點最近的前個查詢點。不像歐基里得平面上以點之間的直線距離為依據，在道路網路中的最近個相鄰者查詢以點之間的權重做為標準，整個道路網路模型的定義如定義三。道路網路上的最近個相鄰者搜尋即是在加權有向圖中找出距離查詢點最近的個資料物件。

圖8是一個道路網路中的例子。以為查詢點，為資料物件，每個空心點代表一個頂點，每個資料物件與查詢點的最短距離以不同顏色的虛線表示。以查詢點與資料物件為例，綠色虛線代表與之間的最短距離，即。以此類推，，，因此查詢點的結果為。

定義三 (道路網路模型):

一個加權有向圖，，其中為頂點集合，代表邊集合。權重函數代表每邊的權重值，對一條路徑的權重值為𝑤(𝑝)=，其中*u*與*v*兩點間，有路徑存在，則最短路徑的權重值為，若路徑不存在則權重值為。

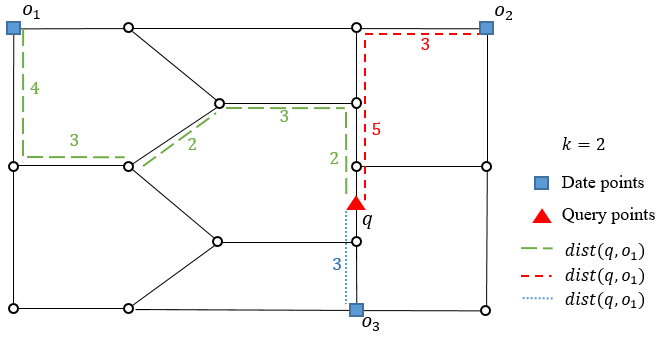


圖8 在道路網路中，以點為查詢點的，其結果為

比起歐基里得平面上的最近個鄰居查詢，道路網路的最近個鄰居查詢較為困難且不直覺。但因為貼近真實道路網路狀況，很多的論文也研究道路網路中的各種查詢。由於作為資料的道路網路往往非常大，為了要讓使用者能定位出查詢點所在的區域以縮小範圍，將道路網路切割成多個子區塊後，再利用區塊資訊建立索引可有效地達成目的。除此之外，藉由不同區塊間的距離關係，我們能夠鎖定查詢答案可能落在的區塊，並將不可能存在答案的區塊加以排除，以減少頻寬與計算量。在道路網路上有許多種類的切割方式[10]，常見的切割方式有利用網格、KD樹(KD)、R樹(Rtree)[8]、等方法。

網格切割法先以一個矩陣將整張道路網路包住，並將此矩陣切割成數個相同大小的子區塊，每個子區塊都代表一個道路網路的子圖。KD樹同樣是先將整個道路網路以一個矩陣包住，接著將矩陣內的資料物件利用垂直分割線分成兩等份的子區塊，再以水平分割線切割子區塊，依此不斷的輪流使用垂直與水平分割線，直到每個子區塊的資料物件數目都剩下一個點為止。

NVD是一種由Voronoi Diagram (VD)衍生而來的切割方式，VD中計算適地性服務的問題僅能使用在以兩點的直線距離作為實際距離的歐幾里得平面中使用，而NVD則包含確切的道路資訊。NVD邊界的計算方式為每一個物件同時使用Dijkstra演算法，在分別屬於二物件之線段交接處為此二物件的最短路徑中間點，稱為邊界點(Border Node)。如圖9所示，為道路上的物件，空心點表示路口，黑色點則為邊界點，圖中用虛線以不同的顏色將不同的NVC的邊界連成一塊區域，單一Cell (Network Voronoi Cell/NVC)的形成為物件到其鄰近物件之所有邊界的點之點與線段的集合。

[12]中將現存的道路網路上最近個鄰居搜尋分成兩大種類：混合索引(Blended Indexing)與去耦合索引(Decoupled Indexing)。混合索引切割道路網路並建立索引的方式與道路網路分布密切相關，NVD切割法便是一種建立混合索引的代表。NVD切割方法能讓使用者在定位查詢點的位置後，利用NVD相鄰Cell中必定包含最近資料物件的特性快速找到NN的答案，然而，因為NVD以每個資料物件為單位建立Cell，造成NVD索引的較大，並且NVD過大的計算量與每個資料物件耦合度極高的特性，使得NVD不易更新道路網路中的資料物件。去耦合索引以網格切割法、KD樹與R樹為代表。去耦合索引的切割方法並不與道路分佈緊密連結，因此每個子區塊中的道路不如NVD切割均勻，但在索引的大小、預先處理與更新道路網路的成本上皆優於混合索引。

[13]提出了IER(Incremental Euclidean Restriction)與INE(Incremental Network Expansion)這兩種無切割道路網路與索引的方式來解決道路網路上的NN問題。IER首先找出與查詢點的歐幾里得距離最近的個物件加入候選清單後，計算候選資料物件與查詢點的實際道路網路距離並排序，再以歐幾里得距離擴大查詢點的搜尋範圍，直到資料物件與查詢點的歐幾里得距離大於查詢點與第個候選物件的實際道路距離為止。INR則是以查詢點所在的路段為中心擴張尋找其他路段，若找到物件即加入候選清單，當找到個物件時完成查詢。

G-tree[14]是一個平衡的(Balanced)搜尋樹狀索引結構，由遞迴切割的次道路網路(sub-networks)所構成，每個節點對應到一個子道路網路。利用一個節點中的資料物件要到其他節點中的資料物件，其路徑都一定會經過邊界點而產生共同路徑(Common Sub-path)的特性，提出assembly-based方法以快速找到從G-tree的節點到查詢點位置的最短距離。

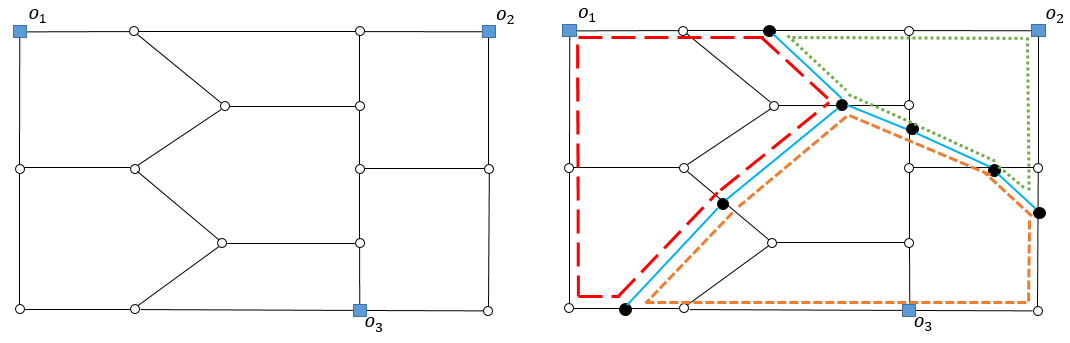


圖9 NVD圖，左圖為圖資物件與道路，右圖經虛線切割的區塊表示NVC範圍

## **2.4**在資料廣播環境中的適地性道路網路查詢

因為廣播模式可以有效地同時服務大量的查詢，近年來，有越來越多的論文探討如何在資料廣播環境中進行適地性道路網路查詢。與傳統服務模式不同，廣播模式中的客戶端並不預先擁有道路網路資料，無法隨時取得距離資訊，因此如何將足夠的篩選資訊加入有限大小的廣播索引結構，讓客戶端一接收到索引就可快速鎖定可能需要的區塊是一個重要的課題。

[15]提出了廣播環境中的道路網路上最短路徑查詢架構與*Elliptic Boundary* (EB)、*Next Region* (NR)兩種方法。利用KD樹將切割道路網路切割成數個區塊後後，EB預先計算每個區塊之間的最大與最小距離當作篩選資訊，並加入廣播索引結構中提供客戶端找出可能區塊。NR預先計算所有區塊間的最短路徑後，將最短路徑行經的區塊編號加入索引結構，以指示客戶端下一個需要聽取的區塊。

Index for spatial queries in wireless data environments (ISW)[16]的廣播索引架構可適用在最近*k*個鄰居查詢、反轉最近*k*個鄰居查詢(RNN Query)、範圍查詢(Range Query)等道路網路上適地性查詢。ISW的做法同樣是先以KD樹切割道路網路成數個區塊。接著計算每個區塊之間，所有節點與其他節點之實際距離與歐幾里得距離比例的最小值與最大值後加入索引。客戶端接收索引，找出查詢點所在的區塊，並計算查詢點與其他區塊的歐幾里得距離乘上得到，最後以區分需要的區域後算出答案。

*Network Partition Index* (NPI)[17]同樣也可進行最近個鄰居查詢以網格切割法切割整個道路網路，並在索引結構中加入道路網路邊界經緯度、網格分割數、每個區塊內的物件個數、每個區塊內的最大距離以及區塊之間的最大距離與最小距離。為了減少搜尋經歷等待時間，伺服端以(1, *m*)的方式將索引加入廣播頻道，如圖10所示，其中。客戶端拿到索引結構後把查詢區塊與候選區塊的最大距離加上候選區塊內的最大距離當作距離上限，以排除不可能存在答案的區塊。

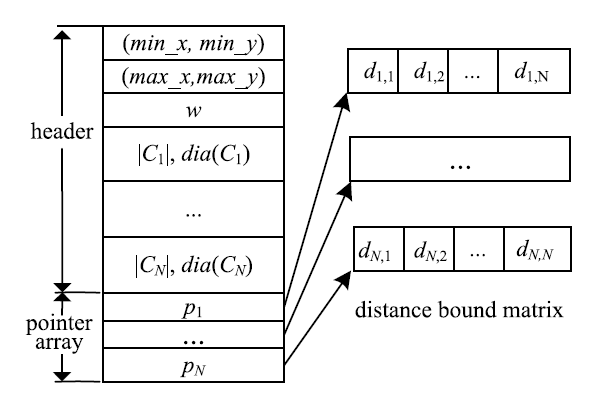


圖10 *Network Partition Index* (NPI)索引結構[17]

# 第三章 Border-based Approximation

當使用者進行適地性查詢最近個鄰居搜尋時，需要的物件往往都離查詢點的位置很近，所以在客戶端中，我們不需要儲存距離查詢點過遠的資訊，因此，將道路網路資料切割成適當數量的區塊並且提供客戶端有效的篩選資訊將不需要接收的區塊排除不聽可以大量減少客戶端在空間與時間上的浪費。在廣播環境中，篩選資訊會加入索引結構中並與道路網路資訊組成廣播週期循環播出，當篩選資訊過多的時候會造成索引結構肥大，若此時採用(1,*m*)索引廣播模式會讓廣播週期爆炸性的增加，客戶端的監聽時間與查詢經歷時間反而會不減反增。根據這些觀察，設計出一個篩選能力強大，同時容量小巧的索引結構是廣播環境中適地性查詢的一大重點。

鑒於以上的原因，本論文提出Border-based Approximation方法(BA)解決廣播環境中道路網路上的最近個鄰居問題，首先將道路網路利用網格切割法切割成數個區塊，索引結構中的內容則包含預先計算的每個區塊之最大距離、區塊邊界點的最短路徑圖以及本論文中提出的邊界最近鄰居距離估算法。利用這些索引內容，我們能精準的逼近實際道路距離，有效的排除大量不需要的區塊。在本章節中，3.1首先介紹網格切割法以及邊界點的定義，3.2會詳細的說明伺服端預先處理的步驟、索引結構的內容，最後在3.3說明客戶端的篩選機制。

## **3.1**切割道路網路

切割道路網路並利用篩選機制讓客戶端選擇要聽的區塊可以有效的縮小客戶端在空間與時間上的浪費。本篇論文中採用網格切割法來分割道路網路。網格切割法會將整個道路網路利用最小矩陣框起來，我們將此最小矩陣的最小值/最大值座標以()表示。接著，每個維度都將分割成*n*等份，因此道路網路在分割之後總共會有個區塊。圖11的例子將道路網路的每個維度都分割成兩等份，總共的區塊數量為個區塊。假設切割完的區塊集合為，若代表任意的子區塊，則。

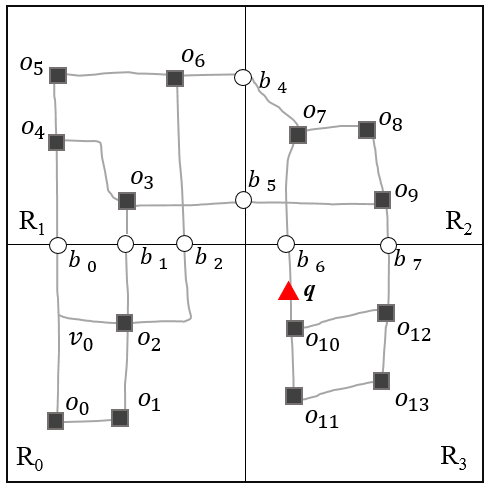


圖11 切割道路網路成個區塊範例

道路網路經過切割後，道路與切割線的交點會產生邊界點，邊界點的定義如下。

|  |
| --- |
| 定義四 (邊界點)：  一個切割為個區塊的集合，而區塊 。若有一條邊，，且，和切割線會產生交點*b*，稱此交點*b*為邊界點(border node)，其中 且 。 |

圖11中每個白色的點都代表一個道路與切割線產生的邊界點。以圖11中的為例，屬於，屬於，則與切割線相交會產生的邊界點，而同時屬於與兩個區塊。

## **3.2**事前運算與建立索引結構

為了要定位查詢點的位置，我們將3.1節提到的道路網路最小矩陣座標與分割道路網路的參數*n*加入索引結構中。然而，只有座標資訊與分割資訊只能算出查詢點所在的位置，仍無法讓客戶端鎖定查詢需要的區塊，因此我們還需要把每個區塊的物件個數與預先計算的距離加入索引結構，以提供客戶端計算整個查詢的距離上限與每個區塊的距離下限的必要資訊，來有效地篩選區塊。

使用區塊內最大距離來計算整個查詢的距離上限十分有效，代表區塊內的所有邊界點至最遠物件的最短路徑之最大值，因此可以保證此區塊內的任兩點距離必定比小。之定義如下。

|  |
| --- |
| 定義五 (區塊內距離的最大值)：  一個切割為個區塊的集合，則區塊，為區塊的物件集合，為區塊的邊界點集合，到之間的距離的最大值。 |

在查詢最近個鄰居搜尋中，伺服端必須提供每個區塊內的物件數量，讓客戶端知道接收到的區塊資料中的物件總數加起來是否大於等於，若預計收到的區塊中，其物件總數不足，則我們需要將更遠的區塊也納入考慮，直到預計的物件總數滿足為止。區塊內的物件數量定義如定義六。以圖11為例，圖中的黑色方塊代表物件，則中有物件集合，而。依此類推，我們可以得到，，。

|  |
| --- |
| 定義六 (區塊內的物件數量)：  一個切割為個區塊的集合，而區塊 ，為區塊的物件集合，為區塊的邊界點集合，則區塊內的物件數量為元素個數。 |

在製作其餘的索引內容之前，我們需要先了解客戶端如何判斷要接收哪些區塊的資料。若一物件屬於查詢點的最近個鄰居候選物件時，則客戶端就必須聽取此物件所在的區塊資料，以確定此物件與此區塊的其他物件是否為最近個鄰居的答案。因此，我們需要找到每個區塊中，離查詢點最近的物件，若查詢點與的距離之下限小於客戶端計算的距離上限，則所在的區塊就需要聽取，反之則可以排除此區塊。簡短來說，必須要找出查詢點與的距離之距離下限以及找出整個查詢的距離上限來篩選出需要聽取的區塊，這邊我們先介紹逼近的方法。

當一個區域中的任一點要抵達另一個區域中的任一點時，其路徑必定會經過與的邊界點。觀察到這個現象，我們提出定理一來說明只要能找出查詢點到邊界點的距離，邊界點到邊界點的距離、邊界點到物件的距離這三段路徑的距離近似值，就能夠逼近查詢點到物件的距離。我們以圖11中查詢點到的距離做為範例，查詢點到的距離可以被拆解為到，到以及到三段距離，只要我們能找出這三段距離，加總後我們就能得到。在本論文中，我們分別使用實際距離與歐幾里得距離比例逼近法、邊界點最短路徑圖、邊界點最近鄰居距離表這三項工具算出來的結果加入索引結構中，提供客戶端來逼近。以下將分為四個小節，3.2.1說明實際距離與歐幾里得距離比例逼近法；3.2.2說明邊界點最短路徑圖；3.2.3說明邊界點最近鄰居距離表；3.2.4說明BA方法完整的索引結構。

|  |
| --- |
| 定理一：  一個切割為個區塊的集合，區塊 ，為區塊的物件集合，為區塊的邊界點集合。若有一查詢點，一物件而到的最短路徑會經過兩個邊界點，其中，則查詢點到物件的距離。  證明：  當一區塊內的任意點要到達另一區塊內的任意點時，其路徑必定經過的邊界點與的邊界點，因此若要到達必定要經過的其中一個邊界點與的其中一個邊界點，所以到的路徑可以分成到、到與到，我們可以此推導出查詢點到物件的距離等於到的距離、到的距離與到的距離三段總和，亦即。 □ |

### **3.2.1**實際距離與歐幾里得距離比例逼近法

對於廣播環境中的適地性查詢來說，最大的考驗在於查詢點位置的不確定性以及無法預先取得道路網路資訊。因為查詢點可能落在任何位置，而客戶端下了查詢點之後，又無法利用道路網路資料知道查詢點的所在位置與周遭的邊界點與物件距離。此外，又因為，，這三段距離中，除了查詢點以外的元素都是可以固定且可以預先計算的，因此，這段距離只能取得近似值，並且近似值的精準程度會大大影響篩選的效果。

大多數的道路網路適地性查詢方法，例如EB與NPI，都以估算的距離上限。這個方法的好處在於只要使用很小的容量便可以逼近，然而，此方法的不夠精準，導致客戶端篩選候選區塊的效果不夠好，間接增加客戶端的監聽時間。而在本篇論文中，受到ISW的啟發，我們利用歐幾里得距離與實際距離的比例關係尋找的距離上限與下限，以取得比更接近的距離。

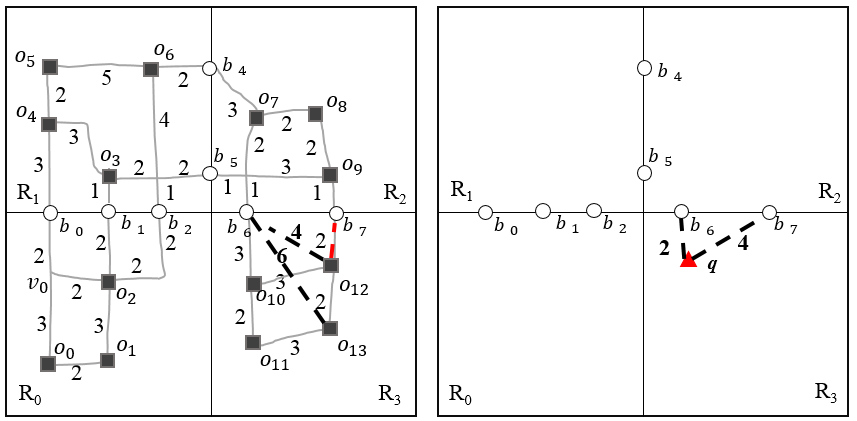
首先，我們介紹區塊內最小的實際距離與歐幾里得距離比值與 區塊內最大的實際距離與歐幾里得距離比值，與 分別代表一個區塊內，任一邊界點與任一物件實際距離和歐幾里得距離比值的最小值與最大值，其定義如下。

|  |
| --- |
| 定義七 (區塊內最小與最大的實際距離與歐幾里得距離比值)：  一個切割為個區塊的集合，而區塊 ，為區塊內的頂點集合，為區塊的邊界點集合。若代表中的任一頂點，代表中的任一邊界點，則區塊內最小與最大的實際距離與歐幾里得距離比值分別為  其中，以代表兩點之間的實際距離，代表兩點之間的歐幾里得距離。 |

伺服端會預先計算所有區塊的後加入索引結構中，除此之外，也會將所有邊界點的資訊放入索引結構。當客戶端下了查詢點時，由於此時客戶端擁有查詢點的座標位置，因此再將乘上便可以算出查詢點到其所在區塊內的每個邊界點的估計值範圍，在此篇論文中以查詢點至邊界點最小的估計距離與查詢點至邊界點最大的估計距離代表代表的估計範圍，詳細定義如定義八。

|  |
| --- |
| 定義八 (查詢點至邊界點最小與最大的估計距離)：  一個切割為個區塊的集合，而區塊 ，為區塊內的頂點集合，為區塊的邊界點集合，而代表中的任一邊界點。若有一查詢點，則區塊內最小的估計距離與區塊內最大的估計距離分別為 |

需要注意的是，此時算出的只代表查詢點到區塊中其中一個邊界點的估算範圍，但因為在找尋最近個鄰居時每個邊界點都可能落在查詢點到達其他物件的最短路徑上。若以代表查詢點所在的區域，我們會計算出一組的集合，其數量與的元素個數相同。以圖12的說明一個與的使用範例，在圖12(a)中，我們能看到的是與的實際距離與歐幾里得的比例；的是的際距離與歐幾里得的比例，因此。今客戶端一個查詢點落在中，如圖12(b)的所示，則客戶端會計算與，其中，而。



(a) (b)

圖12 比例逼近法範例 (a)以為範例，，，(b)，

### **3.2.2**邊界點最短路徑圖

我們使用邊界點最短路徑圖(Border Graph)來得到所有邊界點之間的正確距離。圖13是由圖11省略道路網路，只留下邊界點資訊的結果。邊界點之間的虛線代表兩點之間的最短路徑距離。邊界點最短路徑圖只需記錄區塊內的邊界點的距離，因為要經過一個區塊，必定要從該區塊中的其中一個邊界點進入，再從該區塊中的另一個邊界點離開。利用最短路徑圖，客戶端可以得到道路網路中所有邊界點之間的距離，因此客戶端算出的一定完全符合實際的道路距離。

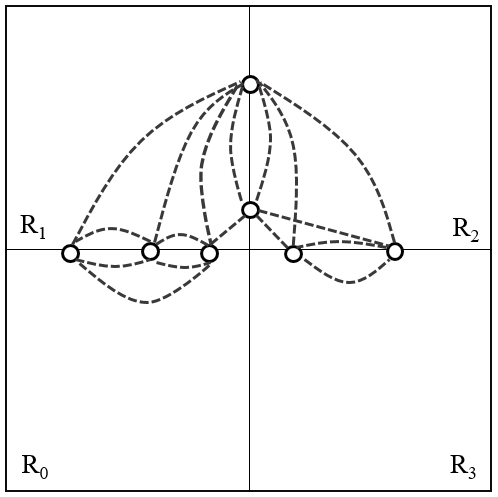


圖13 邊界點最短路徑圖

### **3.2.3**邊界點最近鄰居距離表

如本節前面所提，客戶端需要找到查詢點與每個區塊最近的物件之距離，若距離小於查詢的距離上限，客戶端就必須聽取此區塊。利用這個概念，我們提出了一個邊界點最近鄰居距離表(Border Nearest Neighbor distance table)，簡稱BNN Table，邊界點最近鄰居距離表是一個直覺且有效的索引內容，提供客戶端找出。以下用代表的邊界點最近鄰居距離表。

邊界點最近鄰居表紀錄一個區塊中的邊界點與此區塊中最近物件之距離。圖14是一個邊界點最近鄰居距離表的範例，以為例，我們可以看到的邊界點有與兩點，而的最近鄰居為，兩者間距離為；的最近鄰居為，兩者間距離為，因此的邊界點最近鄰居距離表。以這個方法我們可以推得，，。

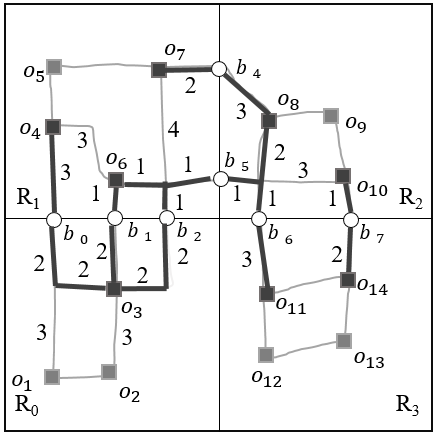


圖14 邊界點最近鄰居距離表範例

邊界點最近鄰居距離表可以提供有效的區塊篩選功能，然而，當切割的區塊數量較多時，在索引結構中放入所有區塊的邊界點最近鄰居距離表，並以(1,*m*)索引廣播會使重複的資訊過多，造成廣播週期與查詢經歷時間過長。為了避免這個問題，我們不將所有區塊的邊界點最近鄰居距離表放入每個索引結構中，取而代之的，我們只將目前廣播的區塊和與其鄰近的區塊之最近鄰居距離表加入索引結構。需要注意的是，若兩個區塊之間共有一個以上的邊界點，則此兩區塊才是鄰近的區塊。如圖14中，的鄰近區塊只有，因為與共有兩個邊界點與。雖然以切割線來看，緊鄰，但因在實際道路網路上與沒並沒有連通，也就是無共有邊界點，所以與並不是鄰近的區塊。因此的索引結構中只會加入與。

邊界點最近鄰居距離表可以得到道路網路中的實際距離，配合上3.2.2節所提的邊界點最短路徑圖，我們可以保證與實際道路距離完全一樣，亦即，只有這段距離會產生些許誤差。因此，我們此方法逼近的能力極為優秀。利用實際距離與歐幾里得距離比例逼近法、邊界點最短路徑圖與邊界點最近鄰居距離表，得到查詢點與每個區塊最近物件的距離後，客戶端就可以篩選出需要的區塊。

### **3.2.4** BA索引結構

在前面的章節中，我們介紹了事前運算以及這些事前運算的用處。在這邊我們會統整之前事前運算的結果，使讀者更了解BA的索引結構與排程。如圖15 所示，在索引結構中，我們首先放入網格切割法的資訊，包括包住整個道路網路的最小矩形的左下點、右上點與每個維度切割的次數。接著放入邊界點最短路徑圖以及區塊內物件數、區塊內最大值、。最後會放入目前廣播區塊以及其鄰近區塊的邊界點最近鄰居表。整個廣播結構使用索引廣播模式排程，在每個資料片段前放置各自的索引資料，而資料片段的內容放置的是目前播出區塊的完整道路與物件資訊。

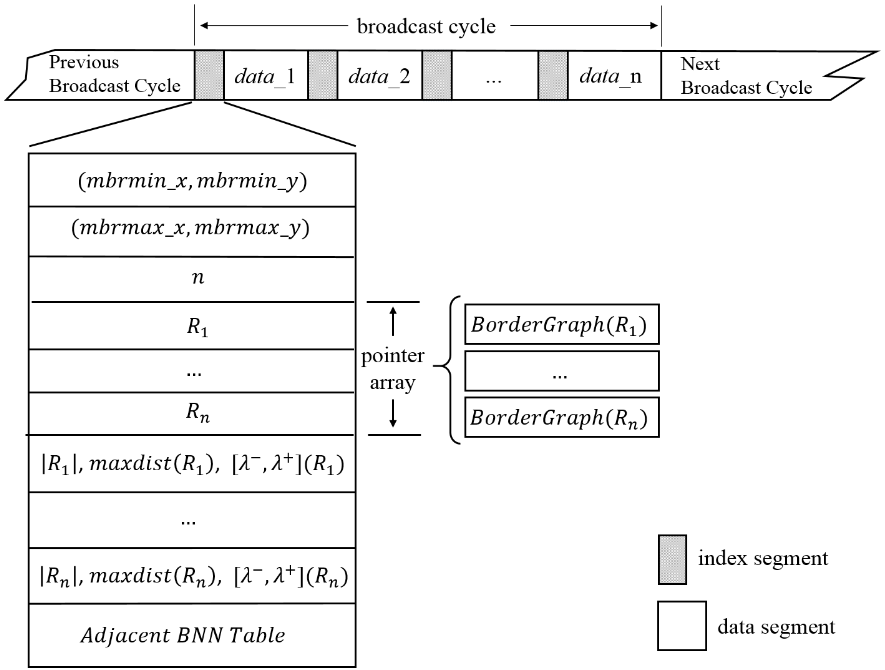


圖15 索引結構

本論文中使用了多種參數定義，為了避免參數的混淆，以下將把本論文中使用到的所有參數名稱整理成一個表格以提供對照使用。

|  |  |
| --- | --- |
| 參數名稱 | 參數意義 |
|  | 道路網路最小矩陣的左下點 |
|  | 道路網路最小矩陣的右上點 |
|  | 頂點 |
|  | 物件 |
|  | 區塊 |
|  | 邊界點 |
|  | 由頂點與頂點組成的邊 |
|  | 頂點到頂點的實際距離 |
|  | 頂點到頂點的歐幾里得距離 |
|  | 區塊內距離的最大值 |
|  | 區塊內最小與最大的實際距離與歐幾里得距離比值 |
|  | 查詢點至邊界點最小與最大的估計距離 |
|  | 區塊內的物件數量 |
|  | 的邊界點最近鄰居距離表 |
|  | 區塊到區塊距離的最小值 |
|  | 區塊到查詢點距離的最大值 |
|  | 區塊到查詢點距離的最小值 |

圖16 參數名稱與參數意義表格

3.3 **Border-based Approximation搜尋協定**

客戶端在下了查詢後，開始聽取廣播頻道，因為此時聆聽的資訊可能在廣播頻道的任意位置，客戶端會等待下一個索引結構聆聽。客戶端在收到索引結構後，首先會利用定位出查詢點所在的區塊，接著，客戶端先利用乘上到每個邊界點的歐幾里得距離計算出。需要注意的是，在最差的情況下，做為區塊內距離估算上限可能比區塊內距離最大值還要大，但因為區塊內的所有距離都不可能大於，因此若，則。

其後，我們會計算整個查詢的距離上限，演算法如圖17。計算整個查詢的距離上限需要先知道每個區塊到的最大距離，以表示，如定義九所示。可以由所有邊界點最大、邊界點到邊界點的最短距離、以及區塊內最大距離相加組成，詳細定理與證明如定理二， 定理二說明，因此我們可以把當作使用，而當時較為特別，因為就落在內部，因此其可用代表，其中代表任一區塊的邊界點到任一區塊的邊界點的最短距離，的定義如定義十所示。當客戶端計算出所有區塊的後，我們以由小到大排序每個區塊，並依這個順序依序累加區塊的物件個數，直到累積物件個數不小於值時，取目前區塊的當作整個查詢的距離上限。

|  |
| --- |
| 定義九 (區塊到查詢點距離的最大值)：  一個切割為個區塊的集合，區塊，為區塊的物件集合，為區塊的邊界點集合，則到查詢點的距離的最大值。 |

|  |
| --- |
| 定義十 (區塊到區塊距離的最小值)：  一個切割為個區塊的集合，區塊，，為區塊的物件集合，為區塊的邊界點集合，則到的距離的最大值。 |

|  |
| --- |
| 定理二：  一個切割為個區塊的集合，區塊 ，為區塊的物件集合，為區塊的邊界點集合。若有一查詢點，一物件。當時，則，當時，則。  證明：  假設會經過兩個邊界點與，其中，。由於道路都是連通的，我們能夠找到一條到的路徑，中途經過與。因此，這條路徑的距離可以表示為，而因為，，又，因此。當時，因為此時到的最短距離必定小於區域內最大距離，因此。 □ |

客戶端以區塊到的最小值當作此區塊的距離下限，並與距離上限比較，若則代表可排除不聽，反之為需聽區塊。可想成定理一裡的，因此可列出等式。然而，如3.2.3所提，在索引結構中只會放入目前播放的區塊以及其鄰近區塊的BNN表，因此，當客戶端尚未收到目前計算中區塊的時，以替代，若此時則仍將此區塊排除。當客戶端在之後接收到的索引結構包含此區塊的邊界點最近鄰居距離表並且此區塊沒有被排除時，客戶端會將更新為。計算每個區塊距離下限的演算法如圖18，而的定義如下。當我們取得了上限值與每個區塊的距離下限後，我們就可以算出所有的候選區塊，客戶端會接收這些候選區塊，若有收到之前尚未接收的BNN表則更新區塊下限值與候選區塊。當客戶端在接收完的情況下同時有接收到不小於值的物件個數，客戶端即會計算Dijkstra以找出第遠的物件距離作為新的上限值來篩選候選區塊。整個搜尋協定會在客戶端接收完所有利用Dijkstra得到最後的NN答案後結束。

|  |
| --- |
| 定義十一 (區塊到查詢點距離的最小值)：  一個切割為個區塊的集合，區塊，為區塊的物件集合，為區塊的邊界點集合，到查詢點的距離的最小職。 |

**Algorithm\_ComputeUpperBound**()

**Input**: *none*

**Output**: *none*

;

**for** each region**do**

**if** **then** ;

**else**

compute by *borderGraph*;

compute ;

**for** all regions ascending by **do**

;

;

**if** **then** break;

**End of ComputeUpperBound**

圖17 計算查詢距離上限演算法

**Algorithm\_ComputeEveryRegionLowerBound**()

**Input**: *none*

**Output**: *none*

**for** each region**do**

**if** BNN() is received **then** update , where ;

**else** compute ;

**End of ComputeEveryRegionLowerBound**

圖18 計算每個區塊距離下限演算法

**Algorithm\_BAQuery**(*queryPoint q*, *k*)

**Input**: *queryPoin*t *q*, *k*

**Output**: *k*NN of *queryPoin*t *q*

**Procedure**:

;

receiveIndex();

locate Region containing *q*;

**for** each border node in **do**

compute ;

**if** **then** ;

**ComputeUpperBound();**

**ComputeEveryRegionLowerBound();**

**for** each region**do**

**if**  **then** *candidateList*.enqueue();

**while** *candidateList* is not empty **do**

**if** now broadcast region *candidateList* **then**

;

receiveDataSegment();

**if** has been listened & received region object amount **then**

compute temporary KNN by Dijkstra;

update upperBound as the distance of q to the nearest neighbor;

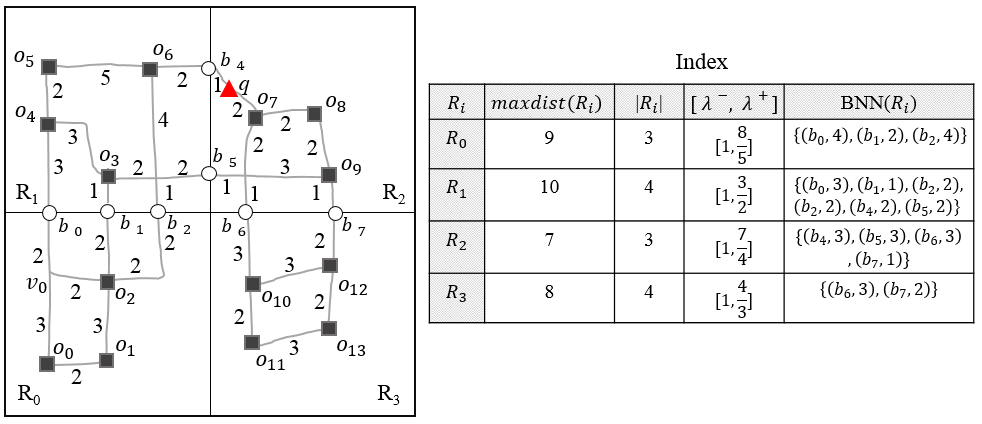
removes regions whose upperbound;

Compute KNN by Dijkstra;

**End of Algorithm\_BAQuery**

圖19 BA搜尋協定演算法

我們以圖20做為一個3NN的BA搜尋協定範例，假設客戶端從廣播的最開始進入拿到第一個索引結構，索引結構內容如圖20(b)。拿到索引結構後，利用、與定位出查詢點位於。接著，找出到每個邊界點的[]後，我們分別計算每個區塊的與。而，計算結果如圖21右上表，原本為，但，因此更新為。接著，將區塊以的大小，由小到大排序為、、、的順序。由於即符合的要求，因此整個查詢的距離上限為。而的計算結果則如圖21右下表，我們可以看到，，因為此時尚未拿到BNN()，因此，則因為包含，屬於必聽區塊，不用計算。由以上得結果得知當客戶端聽完的索引時，候選聽取區塊為、、。爾後，客戶端聽取的道路資訊，候選聽取區塊與原先相同，為、、。但在隨後播出的索引結構時，因為我們收到了BNN()，將更新為，由於大於距離上限值，將從候選聽取區塊排除。在接收完的道路資訊後，客戶端即拿到所有的候選區塊，因此利用Dijkstra算出3NN答案，其結果為。



1. (b)

圖20 BA搜尋協定範例 (a)道路完整資訊 (b)索引結構

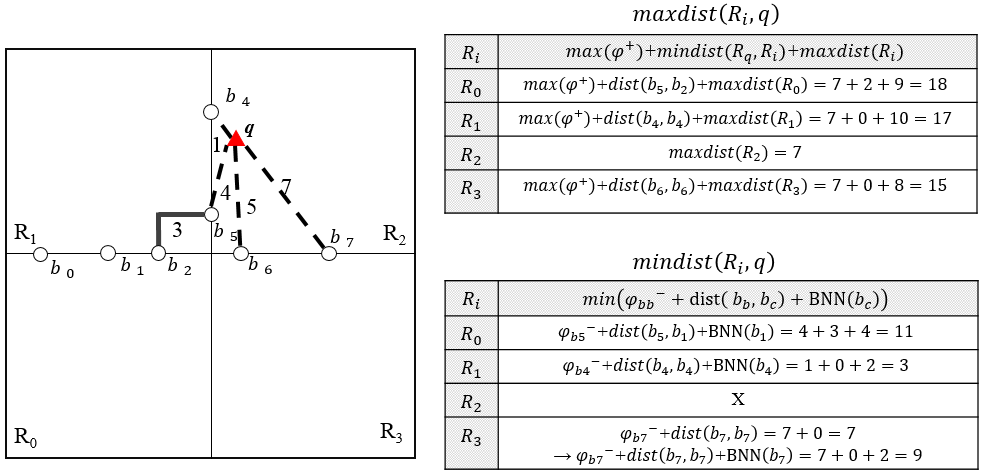


圖21 BA範例，包含資訊、、

# 第四章 模擬實驗

我們使用104年度交通路網數值圖(臺灣)進行模擬實驗，圖22為臺灣連江縣道路網路圖，表1為連江縣的道路、路段、路口的數量。由於連江縣的道路網路資訊較為鬆散，可以測試NPI與BA在這種較為分散的道路網路下的表現。程式語言方法以C++程式語言撰寫，運行環境為Windows 7作業系統，處理器為Intel® Core™ i7-3770 CPU @ 3.40Hz，記憶體容量為24G。

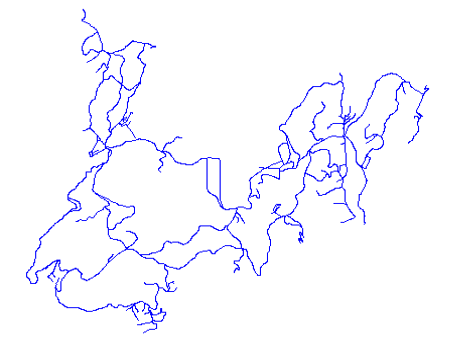


圖22 連江縣

|  |  |
| --- | --- |
|  | 連江縣 |
| 路段數量 | 20780 |
| 道路數量 | 231 |
| 路口數量 | 196 |

表1 縣市道路資訊

本章實驗中共比較兩種方法，分別為本篇論文提出的Border-based Approximation(BA)與Network Partition Index(NPI)。我們會比較不同的切割區塊數、不同的物件數量以及不同的以測量BA與NPI在不同參數下的表現，並分析兩種方法的優缺點。

首先，我們比較切割不同區塊對於兩種方法在監聽時間與查詢經歷時間的影響。區塊的大小會影響方法篩選區塊的能力，區塊切割的越多會讓篩選效果越精準，反之則篩選效果越差，然而，由於採用(1,*m*)索引廣播模式，廣播週期會因此增加。除此之外，每一種方法會因為切割區塊數增長而有不同的效能趨勢，有些方法會因道路網路分佈相關，因此不會隨切割區塊數增加而有太大區別；有些方法的索引結構則會計算兩兩區塊之間的關係，因此廣播週期會隨著切割區塊數增加而指數性的上升。在測試切割區塊數效果實驗中，因為我們認為一般使用者像要找的目的地數量並不會太多，因此我們採用的，而物件數量則是採用物件密度()約為0.1的20個物件。

圖23至圖25顯示，在區塊小的狀況下，NPI因為索引結構會較小，而BA的索引結構會因為邊界點最短路徑圖變得稍大，因此在低區塊的狀況下，監聽時間與查詢經歷時間上BA會略輸NPI。然而一旦區塊數變多，NPI因為需要記錄每個區塊間的最短距離與最長距離，造成索引結構大小快速上升，BA則因為區塊變多，使得BA篩選的能力更精準，同時BA的索引結構大小中，容量最大的邊界點最短路徑圖不會因為區塊數增加而大幅上升，因此在 切割數較大的狀況中，BA在監聽時間與查詢經歷時間皆勝過NPI。

圖23連江縣在不同區塊數量下20個物件的最近5個鄰居搜索的廣播週期

圖24連江縣在不同區塊數量下20個物件的最近5個鄰居搜索的監聽時間

圖25 連江縣在不同區塊數量下20個物件的最近5個鄰居搜索的查詢經歷時間

接下來測試物件數量對於效能的影響，此時依舊設定為5，在切割區塊數方面，由於我們認為一般道路網路道路網路只切割成四個區塊會無法顯著看出排除無用區塊的效果，因此我們在以下的實驗中會分別測試切割區塊數64區塊與16區塊下的狀況。在這個實驗中，我們能看到BA不論在監聽時間還是查詢經歷時間都優於NPI，BA物件個數大小會略為影響索引結構的大小，而NPI因為只記錄區塊間的最小與最大距離，因此NPI索引結構大小不會隨物件個數升高。切割成64個區塊下不同物件的實驗的結果如圖26至圖28。

在切割成16個區塊下不同物件的實驗中，BA的效能則較NPI差，會產生這個狀況也是因為BA索引結構比NPI大，使得查詢時間與監聽時間增加。切割成64個區塊下不同物件的實驗的結果如圖29至圖31。

圖26連江縣在不同物件數量下切割64區塊的最近5個鄰居搜索的廣播週期

圖27連江縣在不同物件數量下切割64區塊的最近5個鄰居搜索的監聽時間

圖28連江縣在不同物件數量下切割64區塊的最近5個鄰居搜索的查詢經歷時間

圖29連江縣在不同物件數量下切割16區塊的最近5個鄰居搜索的廣播週期

圖30連江縣在不同物件數量下切割16區塊的最近5個鄰居搜索的監聽時間

圖31連江縣在不同物件數量下切割16區塊的最近5個鄰居搜索的查詢經歷時間

圖32到圖34是我們測試，在切割成64個區塊下50個景點的實驗中，不同的*k*值對監聽時間與查詢經歷時間的影響，由於先前實驗中只有20個景點，測試較大的*k*值會較不客觀，因此在這裡採用50個景點作為實驗參數。在這個實驗中，我們可以看到BA由於嚴謹的篩選，使得*k*不會對監聽時間與查詢經歷時間有太大的影響，反之，NPI則因為篩選能力較差，因此監聽時間與查詢經歷時間隨值增加而上升。

圖32 連江縣在不同下切割64區塊的最近5個鄰居搜索的廣播週期

圖33 連江縣在不同下切割64區塊的最近5個鄰居搜索的監聽時間

圖34 連江縣在不同下切割64區塊的最近5個鄰居搜索的查詢經歷時間

我們還需要比較在連江縣這個道路網路中，不同的值對於不同切割區塊數大小的BA方法是否會有不一樣的影響。以下我們比較切割成16個區塊的BA方法(BA-16)與切割成64個區塊的BA方法(BA-64)在景點數50下的狀況，由於BA-64切割較多的區塊，且廣播模式是採用(1,*m*)廣播模式，因此監聽時間與查詢經歷時間都會比較高。以監聽時間與查詢經歷時間的趨勢來看，BA-16與BA-64增長的幅度都是些微上升。

圖35連江縣在不同下的最近5個鄰居搜索的廣播週期

圖36連江縣在不同下的最近5個鄰居搜索的監聽時間

圖37連江縣在不同下的最近5個鄰居搜索的查詢經歷時間

綜合以上的實驗結果，我們能知道影響BA與NPI最大的參數皆是切割區塊數量，因為此兩種方法都是採用(1,*m*)索引廣播模式，所以即便索引結構些許變大都會造成整個廣播週期大量增加。BA的優點與缺點都在於邊界點最短路徑圖，邊界點最短路徑圖在篩選區塊的能力上十分優秀，若區塊內道路資訊較大時，BA便可以用很少的監聽時間完成整個查詢，但邊界點最短路徑圖在邊界點太多時會則讓索引結構變大。

# 第五章 結論

本論文探討如何在廣播環境中有效的查詢道路網路上的最近個鄰居問題。我們分析出每個區塊中，只要找出與查詢點最近的物件距離不大於整個查詢的距離上限，則此區塊為候選聽取區塊。利用這個概念，提出了BA索引結構，將此距離以區塊間的邊界區分成三個子距離分別逼近，其逼近結果與實際道路資訊十分接近，因此BA有極為優秀的區塊的篩選能力。

BA的優點還有客戶端能根據接收到的資訊，動態的更新出更嚴謹的距離上限與距離下限並修正候選聽取區塊，換句話說，BA在聽取廣播資料的同時還能不斷的排除更多不需要的區塊，以有效的減少監聽時間。BA的缺點是索引結構中紀錄的邊界點最短路徑圖，雖然能夠提供完全正確的邊界點距離資訊以提高篩選區塊精度，然而卻會因此花費較多的空間，造成廣播週期與監聽時間較大。

實驗以連江縣為例，並與NPI方法比較優缺點。我們的方法在高切割區塊下會有很好的效果，因為高切割區塊能提供客戶端更精準的資訊來篩選不需要的區塊，也因此在監聽時間上有顯著的效果。但是，也因為使用邊界點最短路徑圖在精準的篩選區塊的同時也會造成索引結構資料量變多，所以搜尋經歷時間較長。

在未來的研究中，可以著重在如何簡化邊界點最短路徑圖，在可以接受的範圍中以降低些許篩選能力為代價，減少邊界點最短路徑圖的大小。

1. J. Spinney. A brief history of LBS and How OpenLS Fits Into the New Value Chain, *Location-Based Services ESRI*, 2003.
2. B. Zheng, J. Xu, W.-C. Lee, and D. L. Lee. Grid-partition Index: a Hybrid Method for Nearest-neighbor Queries in Wireless Location-based Services. *Very Large Data Bases Journal*, 15(1):21–39, 2006.
3. C.-M. Liu and S.-Y. Fu. Effective Protocols for kNN Search on Broadcast Multi-dimensional Index Trees. *Information Systems*, 33(1):18-35, 2008.
4. T. Imielinski, S. Viswanathan, and B. Badrinath, Data on air: Organization and access. IEEE Trans. Knowl. Data Eng., 9(3):353–372, 1997.
5. N. Roussopoulos, S. Kelley, and F. Vincent, Nearest Neighbor Queries. In *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, pages 71-79, 1995.
6. A. Guttman. R-trees: A Dynamic Index Structure for Spatial Searching. In *Proceedings of the 1984 ACM SIGMOD Conference on Management of Data*, pages 45-57, 1984.
7. B. Gedik, A. Singh, and L. Liu. Energy Efficient Exact *k*NN Search in Wireless Broadcast Environments. In *Proceedings of the 12th annual ACM International Workshop on Geographic Information Systems*, 2004.
8. W.-C. Yeh, C.-M. Liu, and K.-Y. Ho. A *k*NN Search Protocol Using a Voronoi Diagram in Wireless Broadcast Environments. In *Proceedings of the Second IEEE International Workshop on Wireless Network Algorithm and Theory*, 2009.
9. Y. Gu, G. Yu, X. Yu. An Efficient Method for k Nearest Neighbor Searching in Obstructed Spatial Databases. J. Inf. Sci. Eng. (JISE), 30(5):1569-1583, 2014.
10. R. Möhring, H. Schilling, B. Schütz, D. Wagner, and T. Willhalm,“Partitioning graphs to speed up dijkstras algorithm,”Exp Algorithms, pp. 273–283, 2005.
11. A. Okabe, T. Satoh, T. Furuta, A. Suzuki and K. Okano. Generalized network Voronoi diagrams: Concepts, computational methods, and applications. *In International Journal of Geographical Information Science Archive*, 22 (9), 2008.
12. T. Abeywickrama, M.-A. Cheema, and D. Taniar, “k-Nearest Neighbors on Road Networks: A Journey in Experimentation and In-Memory Implementation,” in *Proceedings of the VLDB*, 492-503, 2016
13. D. Papadias, J. Zhang, N. Mamoulis, and Y. Tao, “Query processing in spatial network databases,” in *Very Large Data Bases*, pp.802–813, 2003.
14. Ruicheng Zhong, Guoliang Li, Kian-Lee Tan, and Lizhu Zhou. G-Tree: An Efﬁcient Index for KNN Search on Road Networks. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, 39-48, 2013.
15. G. Kellaris and K. Mouratidis. Shortest Path Computation on Air Indexes. In *Proceedings of the VLDB Endowment*, 3(1-2):747-757, 2010.
16. Y. Wang, C. Xu, Y. Gu, M. Chen and G. Yu. Spatial query processing in road networks for wireless data broadcast. *Wireless Networks*, 19(4):477-494, 2013.
17. W. Sun, C. Chen, B. Zheng, C. Chen, and P. Liu. An Air Index for Spatial Query Processing in Road Networks. *Knowledge and Data Engineering, IEEE Transactions*, 27(2): 382-395, 2015.