

行動資訊系統之位置分割式資料預取法

Partition-based Data Prefetch for Mobile Information System

段裘慶
Chiu-Ching Tuan

李秩維
Zhu-Wei Lee

楊鍵樵
Chen-Chau Yang

國立臺灣科技大學 電子工程系
Department of Electronics Engineering
National Taiwan University of Science and Technology
cctuan@en.ntut.edu.tw, weilee@selab3.et.ntust.edu.tw, ccyang@et.ntust.edu.tw

摘要

在行動式位置知覺資訊系統中，系統會隨著行動用戶端位置的改變而更新查詢結果，以符合用戶端目前的位置關聯。但由於無線式資料傳輸的頻寬較固接式低，所以用戶端等待資料擷取的時間相對較長，這可能導致快速移動的用戶端所取得的查詢結果無法滿足位置關聯的限定。

本論文提出位置分割式資料預取法以減低位置關聯資料被擷取時的等待延遲和同時提升預取資料的使用效率，以使得查詢結果吻合位置關聯的需求。主要方法係藉由分析行動用戶端過去的移動與存取記錄，以推導出使用者經常漫遊的位置區域供作為資料預取的範圍。

經由模擬實驗結果得知，在承認人類具備高度的行為慣性之前提下，分割式預取法的平均資料擷取時間與平均資料下載量均為最低且是最優，故以分割式法來預取資料是一項可獲益的良好機制。

關鍵字：位置分割、資料預取、行動用戶端、位置關聯資料、行動計算。

Abstract

For a wireless mobile information system, however, its data communication bandwidth is lower than the fixed one; therefore, the corresponding data retrieval latency perceived by clients will be longer than the fixed one. As a result, the mobile client with high mobility may have a chance to receive an incorrect result, if the response sent from the server received too late.

The purpose of this paper is to propose a partition-based data prefetch (PDP) mechanism for the system to effectively reduce the data retrieval latency. The principle method for PDP to prefetch is to parse the profile of client's movement paths and access rates, and then organize them into the hot partitions used as the range to prefetch data.

From the comparison of simulation results, we found that the PDP's average data retrieval time (ADRT) and average data fetching (AFD) is lower and is better than other twos. Therefore, the PDP used to prefetch data is a profitable mechanism if we consider the human being is with high regularity of motion in the daily life.

Keywords: location partition, data prefetch, mobile client, location-context data, mobile computing.

1. 緒論

隨著無線通訊網路與電腦科技的蓬勃發展，經由掌上型或攜帶型電腦的使用人們已可以突破時空的藩籬透過無線網路的傳輸來擷取存放在遠端伺服器內的資訊，由此造就了行動計算環境(Mobile Computing Environment)並進而衍生出行動式位置關聯資訊系統。於近年來，這已是一項廣受注目的研究議題。

行動計算的發展使得用戶端具有可移動性(Mobility)與可攜性(Portability)，從此改變了以往使用者為不可移動的限制。所以行動式資訊系統的設計就必須因應使用者的移動所引發的通訊變動、環境變動與位置變動等項的影響。此類系統稱之為行動關聯知覺應用(Mobile Context Aware Application)，故在系統的規劃上就必須要能滿足位置透明(Location Transparency)與位置知覺(Location Awareness)的抽象要求[7]。

1.1 位置知覺運算

在行動計算環境下的查詢運算除了要反應用戶端位置更動的事實外，同時也要因應無線網路的低傳輸頻寬、高延遲，以及常會發生的弱連線與斷線等特有的通訊特性，藉以維護位置知覺查詢結果的正確性。

主動式查詢(Active Query)會因用戶端移入新的位置關聯地點時而自動地激發另一個查詢，並回應新的查詢結果給使用者。但是在無線網路環境下，如果以抵達新細胞區域後才請求新資料的擷取，如此作法在較長的連線延遲、擷取延遲、弱連線甚至於斷線等不良現象下，恐怕會無法完整地下傳新的查詢結果。

由 Padmanabhan[8]統計得知現行網路連線的建立時間 $T_{\text{setup}}=1.13$ 秒，遠大於每位元資料的傳輸時間 $T_{\text{Byte}}=53.6 \times 10^{-6}$ 秒/位元。由於 $T_{\text{setup}} \gg T_{\text{Byte}}$ 之故，所以若能在連線建立之初就將全部資料抓取完畢，則可以減少後續資料抓取時的等待時間，這就是資料預取的概念。

1.2 相關研究工作

由哥倫比亞大學的 Bill Schilit [10]所提出的行動關聯知覺的概念是指應用程式本身會因應外在環境的變化而自動地發出不同的資料請求，甚至可參照使用者以往於不同位置關聯下所反應的行為範例，用來預測使用者未來的行進而主動給予協助。

CyberGuide[1]是由喬治亞理工學院(GIT)所發展的電子地圖旅遊導覽系統，提供使用者在漫遊時可查詢與環境四周相關的地理資訊。另由華盛頓大學所發展的 Mobisaic 則是應用於全球資訊網(WWW)上的行動資訊系統[12]，該系統係利用

標準 Unix 中動態環境變數的觀念來表現出位置關聯的變化。例如 URL $\text{Http://www.ntust.edu.tw/lab}/\(wei.location) 的網址內容是會隨著用戶端動態位置($\$(\text{wei.location})$)的變化而直接反應在 URL 表示式上。

Padmanabhan 與 Schechter 則分別針對網際網路之網頁存取設立預取的機制[8]，方法是將使用者過去瀏覽網頁的存取記錄作統計分析，然後分離出具有高存取率的網頁以設定為預取的對象。此外，Carl Tait[5]中論述的透通式監聽分解法(Transparent Analytical Spying)係依據於檔案存取期間所自動建立的檔案存取樹，用以達成可在行動環境中漫遊時作檔案的預取。

1.3 本研究的目的與方法

顯然地，若於資料使用之前就將它預取至用戶端中，則可以有效地縮減等候資料查詢處理與資料描顯的時間。特別是在行動運算環境中，其低頻寬資料傳輸甚至於是弱連線或離線情況下更可顯著預取的效果，惟用戶端需要備有足夠的信意空間以儲存所預取的資料。

在行動位置知覺資訊系統中，為了減少因關聯變化而需等待新資料下載所須的時間及減少關聯應用程式的反應時間，本研究提出位置分割式資料預取法(Partition-based Data Prefetch)來預取與位置相關的資料。其要點在於計算位置分割，即分析行動用戶端過去的移動與存取記錄，先行劃分出用戶端經常拜訪的熱門區域，然後再由此區域中分離出具有高存取率的細胞群並將它們設定為資料預取的對象。

Person[9]距離式資料預取法(Distance-based Data Prefetch)則以接近半徑距離內的細胞群作為資料預取的範圍，但 Person 法顯然地忽略了用戶端於實際的移動行為中是會與其個人以往所慣有的行動路徑有關，而並非是完全的隨機移動模式(Random walk model)。故分割式資料預取法有別於距離式法，特別考究用戶端過去的行動路徑上所反應出的移動慣性，以此為基礎來萃取出所謂的熱門分割以作為資料預取範疇的依據。

1.4 本文架構

本論文章節之主要架構如后，第二節將說明行動位置知覺與資料預取法，第三節詳述位置分割演算法的內容，第四節討論行動主機資料預取的模擬與效能分析，最後一節為結論並探討未來的研究方向。

2. 行動位置知覺與資料預取法

在本節中，將就行動位置知覺之系統架構、用戶端代理者結構與資料預取法於行動資訊系統中的優點等項議題作探討。

2.1 行動位置知覺之系統架構

行動位置知覺資訊系統的架構如圖 1 所示，主要部份有：(1) 位置資訊伺服器(Location Information Server, LIS)用以儲存與地理位置有相關聯的資訊，於該資訊中含有細胞代碼及地區名稱以表現出資料的位置屬性；(2) 位置搜尋伺服器(Location Search Server, LSS)負責搜尋在指定區域內可移動物件的所在精確位置；(3) 行動支援基地站(Mobile Support Station, MSS)提供無線通訊服務讓行動用戶端可經由無線網路連接上固接式網際網路；(4) 用戶端代理者(Client Side Agent, CSA)位於行動用戶端負責執行行動位置知覺的查詢服務及負責維護視界(View)的正確性。

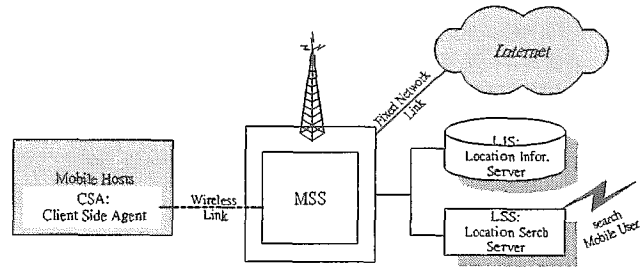


圖 1. 行動位置知覺資訊系統架構圖

於行動式資訊系統中有關行動位置管理問題，Sistla 提出移動向量的觀念來管理與記載系統中的移動物件因位置快速變化而得同步更新資料的需求[11]，然而實際上要取得精確的移動向量並不容易，其應變的作法可分為主動式回報通知以及被動式廣播搜尋兩種。

Alonso[3]就位置關聯的查詢問題提出近似查詢回覆的概念，簡言之，當位置搜索伺服器欲知可移動物件的精確位置時，可依該物件於先前所回報的大略位置內廣播搜尋即可。目前的 PCS(Personal Communications System)的位置管理方式[2]與 Imielinski[4]論述中的區域位置管理都採用近似於此的作法，其優點在於可以有效地降低過多的位置更新命令以及過多的位置查詢需求。

2.2 用戶端代理者(CSA)

行動用戶端經由代理者 CSA 作位置關聯查詢時可以滿足位置知覺與位置透通的需求[13]，並且可由用戶端的移動日誌中所推算出的細胞造訪率與存取率兩項來判決是否執行資料預取的動作，其內部架構詳見圖 2 所示。

用戶端代理者結構中包含有：(1) 位置感知器：執行位置偵測與容量管理的功能；(2) 路徑日誌資料庫(Path Log Database, PLD)：記載與管理用戶端的行動路徑與存取行為次數；(3) 位置分割管理：以分析日誌資料庫中的行動記錄及完成位置分割工作為主。

位置感知器可經由通訊偵測技術得知行動用戶端目前所在的細胞區域，且若檢測出用戶端有跨越(handoff)細胞的行為時(例如， $e_j: \text{Cell}_i \rightarrow \text{Cell}_j$)，便會於日誌資料庫中對此路徑(e_j)所記錄的參訪次數累加一次。另外，用戶端漫遊於新細胞區域期間若有查詢資料的行為發生，則對應此移入路徑記錄中的存取行為次數便累增一次。

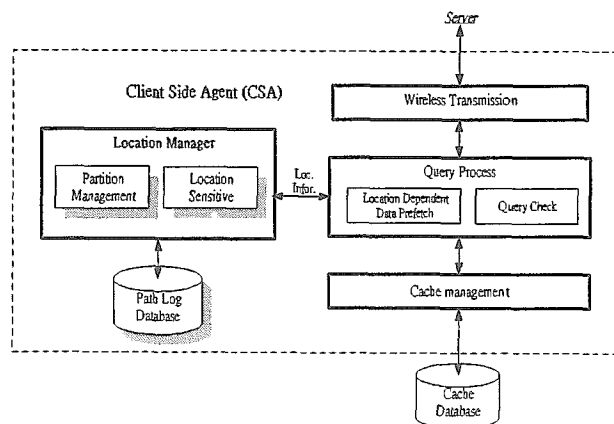


圖 2. 用戶端代理者之架構

但路徑日誌資料庫受制於其有限的儲存空間，所以必須將可記錄的移動路徑資料之筆數設定一上限，令最多存PLDsize筆且當其中可記錄的空間用盡時則以先進先出(FIFO)的方式刪出較舊的移動記錄，藉以儲存新的移動記錄以逐步地反應出行動慣性所呈現的趨勢。

位置分割管理基於節省行動主機於執行分割計算時所需花費的計算成本與電池電能的理由，因此於演算法中定義參數 T_{move} 用以限定重組位置分割的時機。即當使用者離上一次(重)計算位置分割之後，又產生了 T_{move} 次的跨細胞移動後，CSA才令位置分割管理再次地計算新的位置分割。

當 T_{move} 設為較小值時，計算所得的位置分割之時效性預期會較佳，惟它需要花費較多次的分割計算成本。反之，當 T_{move} 設為較大值時，則時效性雖會稍差但可降低行動主機的計算成本與節省電池電能。況且於近期內所產生的新移動路徑若為數不多，通常對位置分割重組的結果不至於會產生明顯的改變。

2.3 資料預取法

由於擬預取的資料屬性與地理位置有關，所以距離式資料預取法[9]以行動用戶端所在的細胞位置為中心點，再取半徑距離(D)所涵蓋的區域作為預取資料的範圍。

在圖3(a)所示範例中的半徑距離 $D=3$ ，區域內總計有25個鄰近細胞，所以除非用戶端離開此區域否則不需再向LIS伺服器端提出新的查詢請求。基本上，此法雖能達成資料預取的目的，惟應就下述的問題再作進一步的探究。

此距離式預取法，若所設定的半徑距離越大($D=n$ ，細胞總數 $=2n(n+1)+1$)則查詢結果直接由暫存資料命中的機率就可相對地提升，但是得需付出較大的下載成本因每次所需要預取的頁資料量(pages)會隨 $O(n^2)$ 同步地激增。且由於此一近似圓狀輻射的預取區域與個別使用者之行動慣性所形成之區域於兩者間的交集程度恐怕偏低，所以易導致所預取的頁資料量過多同時也無可避免地會降低預取資料的使用率。另就維持暫存資料一致性的觀點而言，該項工作負荷也會增加使得驗證資料所需的傳輸成本也會上升。

由於距離式法的缺點在於其假設由半徑 D 所涵蓋的區域內，每一個細胞會被使用者拜訪的可能性均為相等。為修正此項缺點，本研究的位置分割式資料預取法中就以考量細胞參訪的機率與資料存取率為重點，目的是為了改善過量預取資料所造成的高傳輸成本與低資料使用率的問題，以改善資料預取機制的有效性。

分割式資料預取的作法：(一)先依據用戶端的行動日誌分析出

較常造訪的區域以形成所謂的熱門位置分割，(二)再就此一熱門分割內的全體細胞中析離出具備高存取率的細胞群，(三)最後便以此細胞群作為爾後資料預取的對象範圍。如圖3(b)所示之熱門位置分割，假設係由具高拜訪率與高存取率的8個相鄰細胞所組成。相較於距離式預取法，此舉將可大幅降低預取的資料量並提升資料的使用率。

故當主動式的位置關聯查詢被提出後，有鑑於使用者移動路徑的可預測性，所以可以預先將因應位置關連變更而必須回應的資料於事前預取到行動用戶端中。屆時若行動用戶端移進所預期的新細胞中時，CSA即不需要再一次地經由MSS向LIS提出新的資料查詢，藉此以降低使用者等待查詢處理與資料下載的時間。

2.4 位置關聯查詢語式

行動位置關聯查詢語式是以SQL為基礎再增入位置條件語句與查詢存活的控制等兩項。首先，位置條件語句：NearMe、WithMe(distance)，可簡速地於查詢中限定位置物件(例如：櫃銀機、巡邏警車)的搜尋範圍。同時，為了能夠有效地控制主動式查詢的活動區間，也定義了查詢存活的控制語句：(1)Begin(time1), End(time2)設定該單一查詢存活的時段(例如：0810,1030)、(2)Inside(AreaName)設定該查詢存活的區域範圍(例如：台灣科大)，以讓主動式查詢的處理更加的有效率。定義 2.1 為行動位置知覺查詢語法的完整格式[13]：

[定義 2.1] 位置關聯查詢語句：

```
SELECT <Attribute List>
FROM <Table List>
WHERE <Condition> | [NearMe(distance)] | [WithMe] |
[Begin(time1), End(time2)] | [Inside('AreaName')]
```

3. 位置分割演算法

本節將詳述如何引用行動用戶端的路徑日誌資料庫來分割出使用者經常漫遊的細胞區域。

George Liu認為一個使用者的移動型態，可以分為習慣性移動與隨機式移動，同時也提出了行動預測演算法(Motion Prediction Algorithm, MPA) [6]。George Liu所稱的習慣性移動為日常作息慣性會走過的路線是為可預測的行動，而隨機式移動則為屬於隨興的或突發性的移動是為不可預測的。

分割式預取法中亦將使用者的移動區隔為習慣性與隨機式兩類與MPA相仿，且設習慣性移動所經過的細胞為熱門細胞，

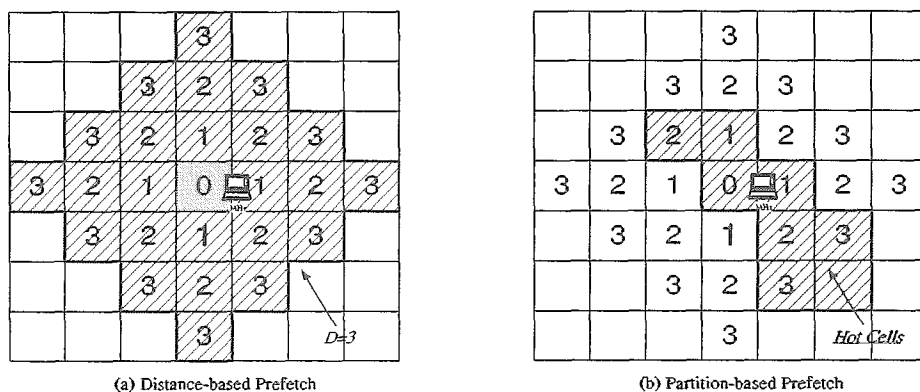


圖 3. 資料預取區域：(a)距離式 $D=3$ ；(b)位置分割式

反之隨機式者視之為冷門細胞。在位置分割與群組的過程中，所要推演的是究竟哪些細胞區域所集合而成的位置分割是最能代表使用者經常漫遊的地點。基本上，判定兩個相鄰細胞區域Cell_i與Cell_j是否可歸屬為同一個位置分割的準則是依照通過這兩個細胞的路徑次數之高低來決定。

基於產生位置分割的概念是以路徑造訪次數高低為分割的基礎，所以拜訪次數較高的熱門路徑予以保留，相對地可把次數較低的冷門路徑暫時先去除，經此程序之後由相鄰接的熱門路徑所組成的每一個連通成份(Connected Component)即成為一個位置分割。

3.1 相關定義

由PLD中的移動路徑 $\langle c_i, c_j \rangle$ 與相關資料可以建構一個加權有向圖 $G=(C, E, W_v, W_a)$ 。其中， $C(G)$ 為在PLD中全體細胞區域代碼 c_i 的集合，對映於有向圖中的頂點； $E(G)$ 則為全部移動路徑 $\langle c_i \rightarrow c_j \rangle$ 的集合，對映於圖中的有向邊 e_{ij} 。

[定義 3.1] 權重函數 $W_v(e_{ij}), W_a(e_{ij})$ ：

1. $W_v(e_{ij})$ 表有向移動路徑 $\langle c_i, c_j \rangle$ 被參訪過的累積次數。
2. $W_a(e_{ij})$ 表為查詢行為的累積次數，即當MH由 c_i 進入 c_j 後之期間，若曾發生查詢行為，則令 $W_a(e_{ij})=W_a(e_{ij})+1$ ，由於此並非查詢次數之累計，故 $W_a(e_{ij})=W_v(e_{ij})$ 。
3. $W_v(e_{ij}), W_a(e_{ij})$ 兩權重值範圍均為含零的正整數，即 $W_v(e_{ij}), W_a(e_{ij}) : E \rightarrow Z^+ \cup \{0\}$ ；並且滿足 $0 = W_a(e_{ij}) / W_v(e_{ij}) = 1$ 。

[定義 3.2] 位置分割 P_i ：

1. 令 $G(C, E)$ 為原始位置分割圖， $P_i(C_i, E_i)$ 為新位置分割，則 $P_i(C_i, E_i) \subseteq G(C, E)$ ，且 P_i 仍為一個連通圖。
 $C(P_i)$ 與 $|C(P_i)|$ 分別表為位置分割 P_i 內的細胞集合及細胞個數。
2. 若 P_i 與 P_j 各為兩個不同的位置分割，其應滿足 $C(P_i) \cap C(P_j) = \emptyset$ 。
3. 對於全體所有的位置分割 P_i ，應滿足下列聯集運算式子：

$$\bigcup_{i=1}^n C(P_i) = C(G)$$

[定義 3.3] 內圍邊(IPE)與外圍邊(OPE)：

1. $IPE_i = \{ \langle c_i, c_j \rangle \mid c_i \in C(P_i) \text{ and } c_j \in C(P_i) \}$ ，
 IPE_i (Inner Partition Edge)為在位置分割 P_i 中屬於內圍邊的集合， $IPE_i(k)$ 表示在 IPE_i 中第 k 個邊。
2. $OPE_i = \{ \langle c_i, c_j \rangle \mid c_i \in C(P_i) \text{ and } c_j \notin C(P_i) \}$ ，
 OPE_i (Outside Partition Edge)為在位置分割 P_i 中屬於外圍邊的集合，其跨接於兩個不同的位置分割， $OPE_i(k)$ 表在 OPE_i 中第 k 個邊。
3. $|IPE_i|$ 與 $|OPE_i|$ 各表示為位置分割 P_i 中內圍邊與外圍邊的總個數。

[定義 3.4] 熱門閥值(Hot Threshold) HOTP：

為計算位置分割 P_i 中所有內圍邊的平均權重值。

$$HOTP_i = \left(\frac{1}{|IPE_i|} \right) \sum_{k=1}^{|IPE_i|} W_v(IPE_i(k))$$

當內圍邊的權重 $W_v(e_{ij})$ 大於HOTP時，認定 e_{ij} 為熱門路徑，反

之視為冷門路徑。引用函數CutColdEdge(P_i)可將冷門路徑移除而保留熱門路徑(詳如演算法 3.2)。由於熱門閥值為IPE的權重平均值，故每次計算時至少可以移除一個冷門路徑因為 $\text{Min}(W_v(IPE_i(k))) = \text{HOTP}_i = \text{Max}(W_v(IPE_i))$ 。故若每個權重值 $W_v(IPE_i(k))$ 均為相等時，必然可得 $W_v(IPE_i(k)) = \text{HOTP}_i$ ，則無從區分出是熱門路徑或者是冷門，故將所有的IPE都視為冷門路徑先加以移除，但可待後續再行逐步合併之。

3.2 位置分割計算

為避免因位置分割過大而造成預取的頁資料量有偏多且使用率偏低之現象，所以定義了參數 MaxPartitionSize 為最大位置分割細胞數之上限，當一個位置分割內的細胞總數超過 MaxPartitionSize時，得將此位置分割再行細分為數個更小的位置分割，使得每一個位置分割內的細胞總數都滿足小於等於 MaxPartitionSize之限制。

同時定義了參數 MiniPartitionSize 為最小位置分割細胞數之下限，以去除因位置分割過小而使得預取的頁資料量過少進而造成暫存資料命中偏低之現象。故當位置分割之細胞總數低於MiniPartitionSize時，將由此位置分割的四周逐步地找尋最佳臨近位置分割，於不違反MaxPartitionSize之前提下可將這些位置分割合併成一個較大的分割區域。演算法 3.1所列為完整之位置分割步驟。

[演算法 3.1] 位置分割：

```

Partition( P0 ) Begin
    Pnew = P0;
    Do // splitting cycle
        If there exists a connected component Pi ∈ Pnew And
            |C(Pi)| > MaxPartitionSize
        Then CutColdEdge( Pi );
            Find all new partitions: { Px, Py, ... } ∈ Pi;
            Pnew = Pnew + { Px, Py, ... };
            Remove Pi;
        EndIf
    Until all Pi ∈ Pnew satisfy |C(Pi)| ≤ MaxPartitionSize;
    K = 1;
    Do // merging cycle
        For every Pi that |C(Pi)| = K Begin
            Neighborhood( Pi );
        End
        K = K + 1;
    Until K ≥ MinPartitonSize;
End

```

End

[演算法 3.2] 冷門路徑刪除：

```

CutColdEdge( Partition Pi ) Begin
    Compute HOTPi for Pi;
    For all eij ∈ E(Pi) Begin
        If Wv(eij) ≤ HOTPi
        Then Remove eij; // a cold path
        EndIf
    End
End

```

3.3 相鄰近位置分割之合併

當新位置分割的細胞總數小於MiniPartitionSize時，需尋找相鄰近的位置分割來予以合併。首先將尋求合併機會的位置分割 P_i 之外緣邊(OPE)由移除過的冷門路徑中給予還原，再引用函數Neighborhood(P_i)找出最佳的臨近分割，最後將之合併以擴大位置分割所包含的區域(詳見演算法 3.3)。

[定義 3.5] 最佳外緣邊(BestOPE)：

$$\text{BestOPE}(P_i) = \text{Max}(\{W_v(e_{ij}) \mid e_{ij} \in \text{OPE}_i\}) \text{ 並且} \\ |C(P_i)| + |C(P_j)| = \text{MaxPartitionSize}。$$

欲成爲位置分割 P_i 最佳的相鄰近位置分割之要件有：(1)該相鄰近位置分割(P_j)與原位置分割(P_i)兩者於合併後的細胞總個數須不超過 MaxPartitionSize；(2)同時外緣邊之權重 $W_v(e_{ij})$ 是滿足前項要件中具最大值的一個邊(e_{ij})。

[演算法 3.3] 鄰近位置分割合併：

```
Neighborhood( Partition  $P_i$ ) Begin
  Recover OPEi for  $P_i$ ; // from removed cold edges
   $e_{ij} = \text{BestOPE}( P_i )$ ; //  $e_{ij}$  exits between inter-partition
   $P_i = P_i + P_j$ ; // merging two partitions into new one
  Remove  $P_j$ ;
End
```

3.4 高存取率細胞群

由於細胞被造訪次數的高低，並不意謂著它就有較高或較低的資料存取行爲，故須由熱門分割內的全體細胞再行個別地求出存取率詳見定義 3.6)，以便挑選出具備高存取率的細胞群，以作爲資料預取的對象範圍，詳如演算法 3.4所示。

[定義 3.6] 存取率(Access Rate) $\text{AccRate}(C_j)$ ：

爲計算單一細胞之存取行爲次數與造訪次數之比值。

$$\text{AccRate}(c_j) = \frac{\sum_i W_a(e_{ij})}{\sum_i W_v(e_{ij})}$$

[演算法 3.4] 高存取率細胞群：

```
CellGroup( Partition  $P_h$ ) Begin
  Group( $P_h$ ) =  $\emptyset$ ;
  For all  $c_j \in P_h$  Begin
    Compute  $\text{AccRate}(c_j)$ ;
```

```
If  $\text{AccRate}(c_j) \geq \text{HighAccessRate}$ ;
  Then Group( $P_h$ ) = Group( $P_h$ ) +  $\{c_j\}$ ;
End
```

3.5 位置分割演算圖例

以圖4爲例來說明區域位置經由位置分割演算法於執行分割之前後的效果。且令PLD中的初始路徑資料所組成的等效向圖如圖4(a)所示。

最後所得出的位置分割有三個： $P_1 = \{1, 2, 3, 8\}$ 、 $P_2 = \{6, 7\}$ 、 $P_3 = \{4, 5\}$ ，均滿足大於等於MiniPartitionSize(=2)及小於等於MaxPartitionSize(=4)。此範例與經由位置分割模擬程式執行後所得的分割結果完全吻合。

下節以模擬實驗方法對不同的資料(存)預取方式作效能比較，以資料描顯延遲與預取資料使用率等兩項數據作評估。

4. 行動主機預取模擬

本節所欲模擬與比較分析的資料(存)預取方法計有三種：(1)簡單式資料抓取法，假設行動用戶端無預取與暫存資料的機制，而每次取用資料都是直接地向伺服器端請求；(2)距離式資料預取法；(3)位置分割式資料預取法。

4.1 模擬實驗架構

模擬的架構中主要有兩個部分：(1)移動範例(Movement Pattern)產生器：依模擬系統所行路徑的參數來產生行動主機移動的範例；(2)行動主機模擬器：主機的移動路線係遵照移動範例來逐步地前進；並設定用戶端代理器每次發出資料請求的時機僅發生在主機有跨越移動的時候；且令該項查詢類型是屬於主動式查詢語句。

移動範例產生器的設計概念係參仿[4]的模擬實驗，爲行動用戶端設立一 Profile 用以假定該用戶端於習慣上會因漫遊而進入與其相鄰接細胞區域的機率，並限定用戶端的可移動方向爲：上(北)、下(南)、左(西)、右(東)。有別於[4]的作法，在此模擬程序中定義了隨機因子(Random Factor, RanFactor)，並依此參數值(0~1)來決定試驗中的用戶端於每次移動時選擇方向的隨機程度，這可使得模擬實驗更具客觀

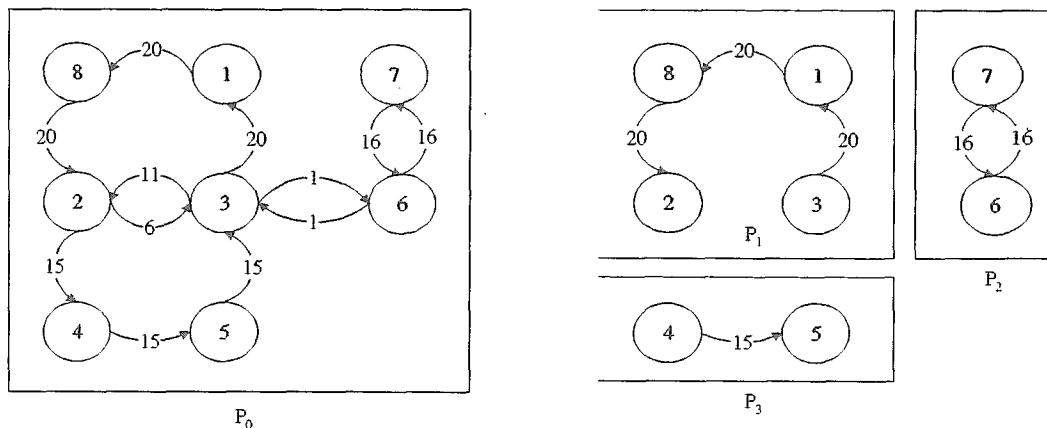


圖 4. 位置分割演算圖例：(a)分割前；(b)分割後

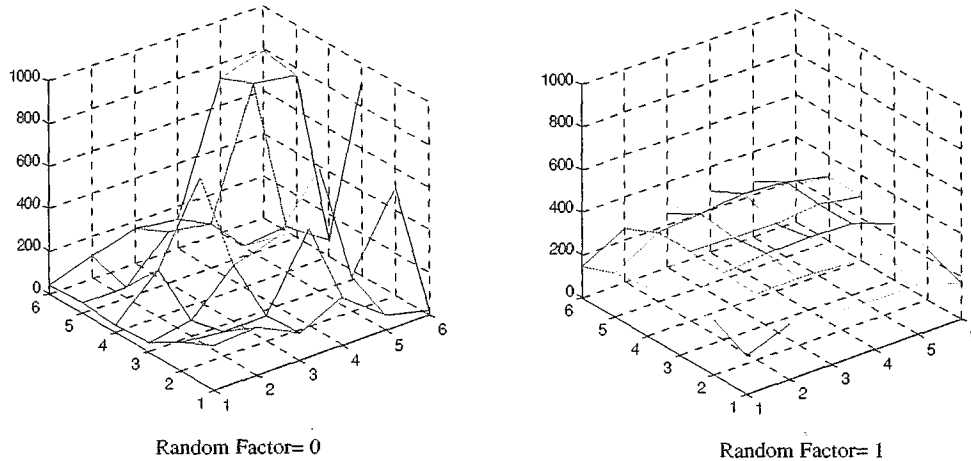


圖 5. 細胞拜訪次數參考圖：(a)RanFactor=0；(b)RanFactor=1

性。

當 RanFactor=0 時表示為絕對慣性，即行動用戶端的移出方向完全遵照 Profile 中所預定的機率值來決定之。反之，當 RanFactor=1 時為絕對隨機，即可移出方向的機率值均為相等而不受用戶端過去偏好的影響，是為一完全隨機移動模式。

上述之兩極模式是模擬時的特例行為，故當 $0 < \text{RanFactor} < 1$ 時表為局部慣性亦為局部隨機的混合式行為。故用戶端移出方向的決定可依 RanFactor 的百分比值來作選擇，以 RanFactor=0.2 為例，即 80% 機會為依慣性由 Profile 預設值來決定，而另 20% 機會則為依隨機方式決定。

在 6×6 的細胞空間中行走 10000 次之後，由圖 5 所示結果證實每一細胞被造訪次數的多寡明顯地受到 RanFactor 參數與 Profile 機率表等設定值的高低而有所變化。其中當隨機因子越大時(如圖 5(b))，行動用戶端造訪每一個細胞的次數越趨平均，反之隨機因子越小時，則表行為慣性越強於是熱門與冷門細胞的區隔性就越趨顯著(如圖 5(a))。

4.2 效能參數 ADRT / ADF

為了強調不同(存)預取機制的效能及簡化模擬實驗程序，故於模擬的程序中限定暫存資料庫僅只儲存當次所(存)預取的資料為主，而將上一次所預(存)取的資料全部移除之。同時也假設每個分割區域內全部都是高存取率的細胞。

有關效能的分析將採用平均資料擷取時間(Average Data Retrieval Time, ADRT)與平均資料下載量(Average Data Fetching, ADF)兩參數值作為評估對象，其計量方式定義如下。

[定義 4.1] 平均資料擷取時間(ADRT)：

$$\text{ADRT} = P_{\text{miss}} \times T_{\text{scr}} + (1 - P_{\text{miss}}) \times T_{\text{cl}}$$

其中，令 P_{miss} 表示為系統的快取誤失值，即查詢結果無法由預取暫存資料庫直接命中取得的比率值。 T_{cl} 為查詢資料由暫存資料庫命中所需的回應時間； T_{scr} 則為查詢資料因快取誤失而必須從伺服器端下載時必要的回應時間，並且設 T_{scr} 與 T_{cl} 為均常數值。而 T_{scr} 可設為常數係假設系統以背景預取的方式來完成所有資料預取的工作，故 T_{scr} 不受預取行為的影響。

[定義 4.2] 平均資料下載量(ADF)：

$$\text{ADF} = \text{Page} / \text{Query} \quad (\text{頁}/\text{查詢})。$$

Page 表為所下載的頁資料總量；Query 則為所發出的查詢總次數。同時為簡化模擬，故假設每次由單一細胞區域所(存)預取的資料量皆為一個頁資料單位。

由定義知，ADRT 是系統於執行(存)預取功能時資料下載的平均反應時間，所以 ADRT 值越小越好；而 ADF 則為資料下載的網路頻寬成本，故 ADF 值越小則網路的使用成本越低。

在簡單式資料抓取法中，由於假設系統無預取機制且僅能儲存一筆頁資料量，所以 $P_{\text{miss}}=1$ ， $\text{ADRT}=T_{\text{scr}}$ ，及 $\text{ADF}=1$ (頁/查詢)，於效能作比較時以上的數值可當作是數據參照組。

4.3 效能模擬結果與討論

模擬系統的參數值設定詳如表 1，另於模擬過程中以 0.05 為間距來改變隨機因子的設定值，藉以探討 RanFactor 對系統效能所造成的影響。

表 1. 模擬系統的參數值

參數類型	參數名稱	參數值(單位)
系統基本屬性	Walk(行走總數)	1000(次)
	T_{cl}	10(msec)
	T_{scr}	5(sec)
	RanFactor	0 ~ 1
距離式資料預取法	D(半徑距離)	1(cell)
位置分割式資料預取法	MaxPartitionSize	4(cell)
	MiniPartitionSize	2(cell)
	PLDSize	300(path)
	T_{move}	100(movement)
	W_v	1

模擬結果如圖 6 所示，該比較圖之 Y 軸各為(a)ADRT 與 (b)ADF，而 X 軸為 RanFactor 以利觀察其對於 ADRT 與 ADF 的影響。以下就模擬結果作要點分析與討論：

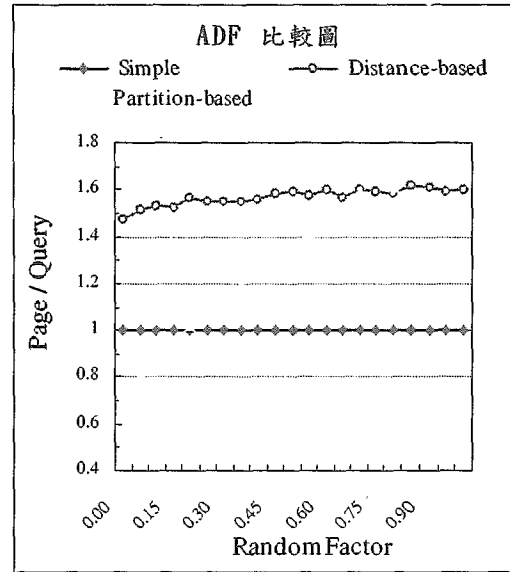
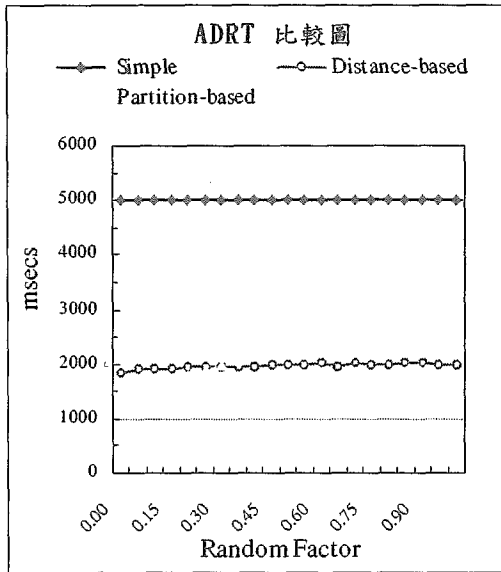


圖 6. 資料預(存)取法之模擬結果: (a)ADRT 比較圖; (b)ADF 比較圖

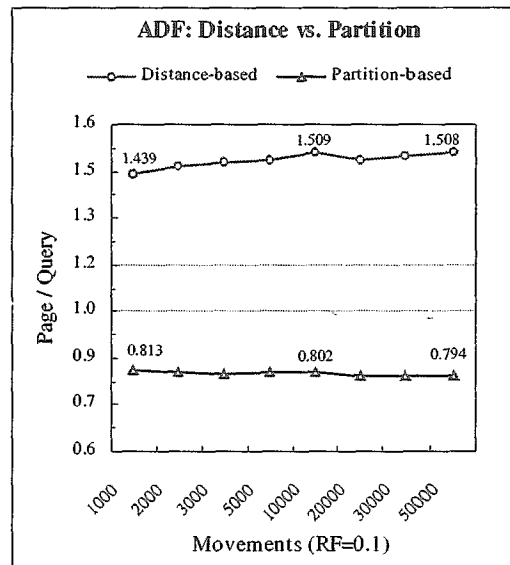
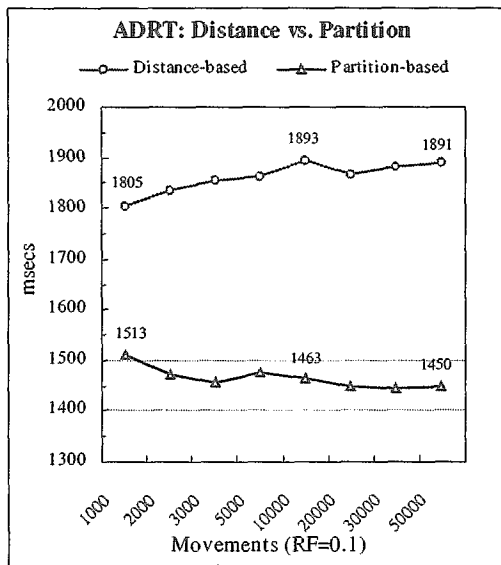


圖 7. 分割式法 vs. 距離式法: (a)ADRT 比較圖; (b)ADF 比較圖

1. 由 ADRT 比較圖可知，具有資料預取機制的分割式及距離式系統其 ADRT 值顯著地比無預取行為的簡單式存取法來的低，其幅度約可減少 40%~70% 之間。其中距離式法的 ADRT 值約 2000(ms) 且該值與 RanFactor 的大小較無顯著關係。
2. 分割式預取法在約 RanFactor < 0.35 之前的 ADRT 值約為 1200~2000(ms) 顯然地較距離式為低且是三者中最優的。另由 ADF 比較圖得知在約 RanFactor < 0.2 之前分割式法的 ADF 值約為 0.7~1 (頁/查詢)，亦是三者中最低最優的。

由此點可證實在接受人類具備高度的行為慣性之前提下，以分割式法所得的熱門區域內來執行資料預取是一項可獲益的機制。

3. 資料預取機制雖然可以明顯地降低 ADRT 值，但其必需付出較高的網路成本用以下載較多的資料需求量。就距離式法而言，其 ADF=1.5 (頁/查詢) 較簡單式抓取法平均多了 0.5 (頁/查詢)，顯示出距離式法之資料平均使用率約為 65% (=1/ADF)，有明顯偏低的現象。

但分割式法在 RanFactor < 0.2 之前，尚可維持 ADF=1 之優勢，即此預取法的資料使用率可高於 100%。惟當使用者移動模式越趨向於隨機行為時 (RanFactor > 0.35)，由於每個細胞受參訪的機率逐步地趨於相近以致於位置分割效果難以凸顯，因此 ADRT 與 ADF 便緩緩地往上升，故在隨機性偏高的情況下分割式法就不太適用。

4. 單就分割式與距離式兩預取法比較：(1) 在 RanFactor < 0.35 之前為高度慣性行動，分割式法顯著優於距離式法；(2) 在 0.35 < RanFactor < 0.75 之間時，分割式法 ADRT

值雖然較距離式法為高但是其 ADF 則較低；(3)在 $RanFactor > 0.75$ 之後為高度隨機行動，距離式法確認優於分割式法。

由圖 7 所示之另一模擬結果係分割式法及距離式法於 $RanFactor = 0.1$ 之前提下，各行走 50000 次之效能分析，藉以探討行走次數對效能變化的影響。其中分割式法的 ADRT 與 ADF 均隨著行走次數增多而呈現微幅下降趨勢，但距離式法則有微幅上升的現象。

顯見地，分割式法除了以數值低於距離式法之優勢外，另由曲線之長期走向觀察得知，分割式法的緩下降趨勢也意味著其系統使用之穩定性依舊優於後者。

5. 結論與未來研究方向

於行動位置關聯資訊系統中，如何減少主動式資料查詢的擷取時間是項重要議題，本論文所提的位置分割式資料預取的機制旨在於達成此目的。主要作法是借助於分析行動用戶端的移動日誌來推導出熱門位置分割，並就此分割範圍再選擇高存取率的細胞群作為資料預取的依據。

此外，分割式預取機制對無線網路頻寬與行動主機電池電能等資源的需求也可以有效地降低，以及支援用戶端斷線與弱連線的運算模式。

由模擬實驗結果可歸納出如后的結論：由 ADRT 比較可知，具備資料預取機制的系統優於無資料預取行為的系統，其 ADRT 降幅約為 40%~70% 之間。而分割式法的 ADRT 在約 $RanFactor < 0.35$ 之前；以及 ADF 在約 $RanFactor < 0.2$ 之前，兩數值均明顯地比距離式法為佳且亦是三者之中最優的。

再由圖 7(a)ADRT 與(b)ADF 觀察得知，分割式法的曲線值具有長期微幅下降之趨勢，故顯示出其仍舊優於距離式法。由此模擬結果可驗證得知，在承認人類行為具備高度慣性之前提下，以分割式預取法去執行資料預取是一項可獲益的良好機制。

本研究主要的焦點在於位置異動後對於行動用戶端存取資料的影響，但除此之外會影響行動用戶端的因素尚有以下諸項可供作為未來的研究方向：

1. 依照行動用戶端移動的速度而適時地調變預取資料的範圍，既當用戶端作較快速行動度時，相對也加快位置的變化，故預取的資料範圍應適切地加大，反之應縮減預取的範圍，以控制所擬預取的資料總量。
2. 處在不同的移動速度下，所須要回應的資料顆粒大小應有所不同，故將移動速度反應在資料綱要上，可以提高資料的參考價值。
3. 行動主機於移動方向上所呈現的移動慣性，可以進一步將之引為位置分割與資料預取時的考量因素。
4. 在伺服器端諮詢所有使用者對資料存取的行為模式，以建立全體用戶端共同的存取範例，則可透過此共用範例的引用來提高預取資料的使用率。
5. 探討系統所配置之快取記憶體的大小對不同(存)預取機制於效能上的影響。

參考文獻

- [1] G D. Abowd, C. G. Atkeson, J. Hong, S. Long, R. Kooper and M. Pinkerton, "Cyberguide: A Mobile Context-aware Tour Guide," *Wireless network*, Vol. 3, No. 5, pp. 421-433,

Oct. 1997.

- [2] L. F. Alkyildiz and J. S. M. Ho, "On Location Management for Personal Communications Networks," *IEEE Communications Magazine*, pp. 138-145, Sep. 1996.
- [3] R. Alonso and H. F. Korth, "Database System Issues in Nomadic Computing," *ACM SIGMOD RECORD*, Vol. 22, pp. 388-392, May 1993.
- [4] T. Imielinski and B. R. Badrinath, "Querying in Highly Mobile Distributed Environments," *18th Very Large DataBases*, pp. 41-52, Aug. 1992.
- [5] H. Lei and D. Duchamp, "An Analytical Approach to File Prefetching," *1997 USENIX Annual Technical Conference*, January 1997.
- [6] G. Liu and G. Q. Maguire, "A Class of Mobile Motion Prediction Algorithms for Wireless Mobile Computing and Communications," *ACM Mobile Networks and Applications*, Vol. 1, pp. 113-121, 1996.
- [7] H. Maass, "Location-aware Mobile Applications Based on Directory Services," *ACM Mobile Networks and Applications*, Vol. 3, pp. 157-173, 1998.
- [8] V. N. Padmanabhan and J. C. Mogul, "Using Predictive Prefetching to Improve World Wide Web Latency," *ACM SIGCOM Computer Communications Review*, Jul. 1996.
- [9] V. D. N. Person, V. Grassi and A. Morlupi, "Modeling and Evaluation of Prefetching Policies for Context-aware Information Services," *The 4th Annual ACM/IEEE Int'l Conf. on Mobile Computing and Networking*, pp.55-65, 1998.
- [10] B. N. Schilit, "A Context-aware System Architecture for Mobile Distributed Computing," *Ph.D. Thesis*, Columbia University, May 1995.
- [11] A. P. Sistla, O. Wolfson, S. Chamberlain and S. Dao, "Modeling Querying Moving Object," *IEEE Data Engineering*, 1996.
- [12] G. M. Voelker and B. N. Bershad, "Mobisaic: An Information System for a Mobile Wireless Computing Environment," *Proc. Workshop on Mobile Computer Systems and Applications*, Dec. 1997.
- [13] 李秩維, "行動位置知覺應用與位置分割資料預取", 台灣科技大學電子工程研究所 碩士論文, Jul. 1999.