

考慮間歇性之存取廣播資料

Retrieving Broadcast Data Considering Intermittent Disconnection

林聯發

Lien-Fa Lin

高苑科技大學資訊傳播系,成功大學資工系博士班

Department of Data Information Communication

Kao Yuan University of Technology, Kaohsiung, Taiwan, R.O.C.

摘要

在無線資訊系統中，廣播磁碟技術已經變成一個熱門的資料散播的方式。然而，當有些因素，如週期性的斷線發生時，廣播的策略就需要告訴行動通訊使用者一個適合的存取廣播資料的方法。包含以上技術的應用程式可以廣泛地應用在我們日常生活中，例如，新聞發佈、地方旅遊資訊的發送、進階交通資訊系統、在泛歐式行動電話環境下使用蜂巢式的廣播等等的應用。

本篇論文中，我們提出了一個完整的機制來解決行動通訊使用者在行動環境中如何有效率的存取廣播的資料。我們的目標是發展一種利益導向(benefit-oriented)的資料擷取協定，讓行動通訊使用者能夠在消耗少量的電源的情況下，取得對行動使用者而言更多有價值的資料。

關鍵詞：行動資料管理，行動計算，資料庫管理系統，資料廣播，無線資訊系統。

Abstract

Broadcast disk technology has become a popular method for data dissemination in wireless information systems. However, when the Factors such as intentional periodic disconnection are involved, a broadcast strategy needs to actively tell the mobile host (MH) a suitable access mode for retrieving the broadcast data.

Applications including the above techniques can be widely used in our life, e.g., news delivery, local travel information delivery, advanced traffic information systems, cell broadcast under GSM environment, ..., etc.

In this paper, we propose a more complete mechanism for mobile clients to efficiently and effectively access broadcast information. Our goal is to develop a retrieval protocol so that a mobile host can retrieve more valuable broadcast information by consuming less power of a mobile device.

Keywords : mobile data management, mobile computing, database management systems, data broadcast, wireless information systems.

壹. 緒論

在無線傳輸的環境下，因為硬體設備的物理限制，這些無線傳輸系統有一個特性：下載的頻寬比上傳頻寬更多。我們稱這個特性為非對稱的特性 (Imielinski et al. 1994)。許多研究資料管理的學者發現，在這個非對稱特性環境下，資料的傳輸如果使用主從架構的傳輸方式，系統的效率會十分差。一個解決這種窘境的方法就是使用資料廣播 (data broadcast) 的方法來克服這種特殊的環境。所謂的資料廣播的方法就是，當行動客戶端 (mobile client, MC) 需要資料的時候，不要馬上透過上傳頻道把要求傳送給伺服器，而是行動客戶端要先到廣播頻道中監聽廣播伺服器廣播的資料。若是有聽到想要的資料，那麼行動客戶端就少發出一次要求，所以減少上傳頻道的使用量。因此，資料廣播變成在無線通訊網路的環境下一種有效而且重要的手段來服務行動使用者 (mobile user, MU) (Barbara 1999; Datta et al. 1999; Traintafillou et al. 2001; Zheng & Lee 2005)。

在深入研究各式的廣播應用 (broadcast applications) 後，我們發現針對存取行為的電源使用 (power consumption of access behavior) 在許多相關的文獻中完

全沒有被提出來討論。然而，這在許多應用程式中是非常重要的。例如，在漫長的旅途之後，一下飛機，旅客第一件在機場想要做的事情就是透過廣播頻道 (broadcast channel) 讀取地方新聞以及地方性的旅遊資訊。這有助於旅客感覺就像在他們原來所生活的地方一樣。但是要注意的是，在大多數的情形下，旅客的行動裝置所剩下的電力可能很少，因為現在許多飛機都有提供行動裝置的使用環境。所以，旅客在機場時，就會面對如何使用有限的電力去存取廣播頻道中大量有用的新聞或是旅遊資訊。本研究的目標是在資料廣播的環境下設計一種機制，可以提供行動使用者一個使用少量電力來存取最多有價值資料的高效率方法。

一個有效率的存取方法會指示行動裝置何時連線，何時斷線來存取大量的資料。現存的資料廣播有一種方法也可以使用少量的電量存取資料，那就是使用索引 (*index*) 去指示 MU 何時該覺醒而且接收所需資料。但是它沒有解決上面所提及之問題。主要原因就是索引方法 (Datta et al. 1999; Hu et al. 1999, 2000, 2001a, 2001b) 只能提供給那些精確地知道該抓取什麼資料的使用者 (或應用程式)。然後藉由索引資訊的幫助引導他們到所需資料所在的廣播位置。但是在資料廣播的環境下，使用

者可能剛進入廣播頻道而不能明確地說出他們所需要的資料名字，亦就是行動使用者無法預先知道廣播伺服器廣播的內容為何。在這情況下，索引沒辦法幫助去減輕這問題。另外，若是對於一些聰明的應用程式（例如有預測機制），由於不需要有使用者明確的指定去抓特定的資料（可是應用程式又必須抓一些資料來進行預測處理來達到一些讓使用者覺得方便的功能），因此索引在這時候也不能夠提供一個有效率存取方法。

無線資料廣播的服務已經被使用在商用產品有多年了，例如 StarBand (<http://www.starband.com/>) 與 Hughes Network Systems (<http://www.direcway.com/>) 即是此類著名的商品。不管在研究上或是在工業上，資料廣播的應用程式都是廣泛被注意的焦點。在工業界，我們可以找到許多真實的應用，包括：在 GSM 環境的蜂巢式廣播系統(Cell Broadcast System - <http://www.cellbroadcastforum.org/>)，進階式的交通資訊系統(Advanced Traffic Information System (ATIS))(Shekhar et al. 1996)，無線教室(Wireless Classrooms - Katz 1994)，無線環境新聞散播(News Delivery in Wireless) (Gifford 1985)，波士頓社區資訊系統 (Boston Community Information System)(Gifford 1990)，空中媒體生活網際網路廣播網 (AirMedia Live Internet Broadcast NetWorks)(<http://www.airmedia.com/>)，以及 Bellcore 的週期式資料專案 (Data-Cycle Project at Bellcore) (Bowen et al. 1992)都是具有這種傳輸型態。然而，在開發初期，這些系統的效能並不十分好。這原因是早

期系統製作廣播節目的時候並沒有考慮資料在廣播節目的組織以及排程。Swarup Acharya 等人在 1995 年針對這個缺點，提出一個廣播磁碟(Broadcast Disk) (Acharya et al.1995,1996a) 的方法，廣播磁碟的基本想法就是將使用者常需要的資料多廣播幾次。這樣的好處可以讓一些人的存取時間 (access time) 降低。因為廣播磁碟在存取時間的顯著改進，所以它在這個領域中是一個重要的成果。幾乎上述所有的應用可以使用這個方法來減少存取時間，尤其是適合是公共資訊型態的資料，e.g., ATIS (Shekhar et al. 1996)。除了工業上的應用，很多研究人員也延伸廣播磁碟系統，使它具備更先進的功能。舉來來說，Swarup Acharya 等人根據廣播磁碟技術建立了散播式的資訊系統工具(Dissemination-Based Information Systems (DBIS) toolkit)(Acharya et al.1996a)。Evaggelia Pitoura(1999,2002) 等人討論在廣播磁碟系統下研究增加交易的平行性的問題。Jeong-Hyon Hwang(2001) 等人提出改善廣播磁碟效率的方法。從上面我們舉的應用範例中可以發現，廣播磁碟確實在真實世界的廣播驅動(broadcast driven)的應用程式中扮演重要的角色。而 Microsoft 在 2003 年在國際消費電子展中宣佈的聰明的個人物件技術(Smart Personal Objects Technology(SPOT)) 的 DirectBand Network (<http://www.microsoft.com/spot/direct.msp>) 更進一步確定企業界對於能便利的利用無線廣播來提供無處不在隨時隨地的資訊服務深感興趣。

無線資訊系統由於環境與行動裝置本身的特性，其資訊擷取有以下的特性，

(1)行動裝置儲存空間有限，沒有辦法儲存太多資料。(2)MU 有移動性的特性，MU 經過一段時間的移動，可能就會切換到另一基地台(base station)的服務範圍，新的基地台服務的資料可能有所不同或者新舊基地台廣播節目的排程時序不一致，這時 MU 將無法抓取所需資料，故 MU 必須在限定的時間完成資料的存取。(3)行動裝置電力有限且連線費用通常是較昂貴，不宜也沒辦法一直連線抓取資料。因此無線環境設計一個高效率資料廣播系統有別於傳統的設計方式，必須就節省電源與 MU 能在有限的時間存取資料的特性加以考量。

目前在資料廣播環境下已經有許多論文討論資料的組織以及索引的論文 (Huang & Chen 2003; Hsu et al. 2002; Hameed & Vaidya 1999; Sanieev & Shiyu 1998; Kenyon & Schabanel 1999; Khanna & Zhou 1998; Zheng et al. 2004)，這些成果對於欲抓取的資料為單一明確的資料提供了有效率的存取方式。本論文，更進一步的，從過去討論單一筆明確資料的存取延伸到行動使用者無法預先知道廣播伺服器廣播的資料為何以及行動使用者必須在一個有限的回復時間(turnaround time) T 內以更省電的方式存取整個所需要資料的行為。這個研究的成果不但拓展了過去研究的角度，而且也更提供一個更真實且完整的無線應用程式發展的模型。本論文的技術貢獻可以歸類成下面四點：

- (1) 提出無線應用程式的新的存取模型。
- (2) 設計支援有效率的存取機制。
- (3) 設計接近最佳化的計算及最有效率存取模式的演算法。

(4) 詳細而且完整的實驗結果支持提出的機制及演算法。

總結來說，本研究的成果可以很容易的應用在許多目前現有的應用程式的資料存取機制中，例如旅遊導覽系統，新聞廣播系統，交通狀況通知系統等等。而使用這個存取機制的應用程式在幾乎不影響執行結果的情況下將會比現在論文中的方法，延長行動裝置的工作時間(約 2 倍的時間)，使得可以完成更多的工作。

本篇論文的其餘架構如下：第貳節為探討與本研究相關的研究。第參節則談論背景知識與支援有效率廣播的資料結構設計。第肆節提出評估模式與演算法設計。第伍節說明模擬實驗的模式與實驗的結果。第陸節是本論文的總結與未來展望。

貳.相關研究

本論文相關的研究，大致上可分成兩類：一類是資料廣播(data broadcast)的研究；另一類是斷線處理的研究。以下，我們分別就這兩類描述目前研究的成果以及進展。

一. 資料廣播 (Data Broadcast) 的相關研究進展

Weiser (1993a,1993b)提出許多行動計算環境的特性，以及許多資料庫領域相關的議題。其中資料傳遞 (data dissemination) 是一個十分新穎且在傳統環境下很少被深入討論。Tomasz Imielinski(1994)等人指出在無線網路中，由於硬體的限制而造成下傳頻寬遠比上傳頻寬大。因此，在這個環境中，客戶

端最好不要常常送出要求而造成上傳頻寬的浪費。爲了達到這個目的，專家們建議用廣播的方式傳送資料。也就是伺服器不停的利用大量的下傳頻寬把資料在廣播頻道中播出，當客戶端要求資料的時候，客戶端並不馬上送出要求，而是先到廣播頻道中監聽是否有自己想要的資料。等確定自己想要的資料不在廣播中再對伺服器送出要求。利用這種廣播的方式，許多使用者就可以直接從廣播頻寬中拿到自己想要的資料而不浪費任何上傳頻寬。上傳頻寬因此可以大量被節省。

但是，廣播的方式可能會造成使用者的存取時間 (access time) 大量增加。爲了減緩這個問題，Swarup Acharya(1995,1996)等人提出廣播磁碟的方法有效的降低存取時間。這個方法的主要想法就是提高熱門資料的廣播次數，而降低冷門資料的廣播次數。廣播磁碟的方法十分簡單而且有效率，之後的研究有許多都是根據這個方法做延伸。在這些延伸的研究中大致可以分成兩類：(一)在客戶端增加機制再降低存取時間，(二)降低客戶端電源的使用。茲分述如下：

(一) 在客戶端增加機制再降低存取時間。

這一類的方法主要是靠快取 (caching)(Chen et al.2005; Xu et al.2004; Hu et al.2005)，預先抓取 (prefetching)(Acharya et al.1996b)，資料壓縮(Data Compression)(Chen 1999; Chen et al.2003)或是平衡推與拉之間的頻寬 (push and pull bandwidth)(Acharya et

al.1996b)。Swarup Acharya 等人提出一個稱爲 *PLX* 的快取方式來降低存取的時間。*PLX* 會根據資料的存取機率以及資料廣播的頻率在快取中挑選一個最少會被用或是即使很常用但很快會再被播出的資料當犧牲者從快取中移除，而把空出來的位置來改善存取效率的資料 (Acharya et al.1996b)。Swarup Acharya et al.(1997) 也提出利用備援頻道 (backchannel)來平衡推(push channels)與拉(pull channels)的頻道的使用量。Quinglong Hu et al.(1999)則是整合快取以及拉式(pull-based)點對點資料傳輸來達到降低存取時間的目的。

(二) 降低客戶端電源的使用

這一類的方法是在廣播節目中使用索引技巧以提供行動客戶端可以用比較低的電源讀取資料。主要是在減少行動客戶端讀取資料時所花費的連線聽取資料的時間 (tuning time) (Hu et al.1999,2001a,2001b; Lo & Chen 2001)。這類的研究主要是在讓行動裝置在廣播不是自己感興趣的資料的時候進入睡眠模式(doze mode)，等到自己感興趣的資料廣播時才切入廣播頻道連線以行動模式 (active mode) 抓取資料。這些研究成果雖然也是在節省電力的前提下進行研究，但這些相關研究跟本論文的重點有幾點不同，1. 索引技術沒考慮針對個人最佳化，索引的技術強調在降低平均的連線聽取資料的時間(tuning time)，並不針對個別使用者進行最佳化，不同的使用者進入廣播頻道的時間有所不同，不同的進入時間點，等待抓取索引資料的時間亦不同，也導致資料存取的效能有較大

的差異。2. tuning time 不能真正反應電能的消耗，雖然行動模式所消耗的電力遠大於睡眠模式所消耗的電力，約 20 倍 (Viredaz et al.2003)但若只為了節省連線聽取資料的時間(tuning time)而導致太長的等待時間(waiting time)，則總消耗電量將可能不會是最少，3.兩者的應用環境不同。上面 index 相關文獻是在處理如何有效率的抓特定一個 data，我們的研究的重點並不是在處理如何存取特定的資料，而是在一個固定的回復時間的時間內讓客戶端可以儘量省電的方式抓到足夠的重要資料。

二. 斷線行動 (Disconnected Operations)

在行動計算環境中，行動客戶端可能會經常的斷線 (disconnection)。造成斷線的原因主要是無線通訊媒體或是行動客戶端的移動性造成。斷線可以分成可以預期或是不可預期斷線兩種 (Kistler & Satyanarayanan 1992)。不可預期的斷線通常很難預防。為了克服可預期的斷線，讓使用者在斷線的時間仍然可以繼續存活工

圖 1是描述一個資料廣播系統的架構。在這個架構中主要包含了三個部分：廣播伺服器，廣播頻道，以及行動客戶端。第一個部分，也就是廣播伺服器，主要的功能就是蒐集使用者需要的資料來產生廣播節目。產生的廣播節目會存在資料庫中。等節目預定的時間一到，廣播伺服器就把相關的節目內容送到廣播頻道中讓行動客戶端存取。一般而言，廣播節目通常是根據客戶端的需求來組成。需求率越高，資料就越有可能被廣播；反之，需求率越低，資料可能被廣播的機率就越低。Hu & Chen (2002)

作，目前的做法是在斷線前將行動客戶端事先將所需的資料下載到行動裝置上。這種動作稱為儲藏(hoarding) (Kuenning & Popek 1997)。由於行動裝置會一直有新的工作而且這些工作可能都是具有不同的存取模式，如何讓儲藏的動作正確與有效是一重要的研究課題。目前解決這個問題的相關論文主要集中在如何支援斷線運算。重要的成果大致有下面幾篇。Shirish Hemant Phatak et al.(2002)等人依據儲藏屬性(hoard attribute)適當的分割資料(partition data)以改善儲藏(hoarding)的效能。另外，Ing-Ray Chen et al.(2002)提出一個演算法來找出適當的斷線時間，目的在於減少資料更改的代價。

參.背景知識與支援有效率廣播的資料結構設計

一. 系統架構

提供了一些可行的方法來得到資料的需求率。

目前製作廣播節目的方法有很多 (Acharya et al.1995; Triantifillou et al.2001)，每一種製作的演算法提供不同的功能。我們舉一個最常被使用的製作廣播節目演算法，廣播磁碟，為例子來說明如何製作廣播節目。我們以下並且會用這個例子來說明整個研究的主要想法。

圖 2是演算法執行過程的說明圖。我們假設要建構的是一個 3-Disk的廣播節目。假設 disk D_1 的資料廣播次數是disk D_2 的兩倍，而且是disk D_3 的四倍。資料庫中總共有 11 個pages (廣播頁)，而且被分成三個群 (分別是 1 個 page，2 個 pages，和 8 個pages) 而且每個群就是一

圖 2中的排程(Program)。

廣播系統的第二個部分就是廣播頻道，它的特點是它具備大量的下傳頻寬，但是只有少量的上傳頻寬。跟其他相關論文一樣 (Acharya et al.1995; Trianfillou et al.2001)，在我們的研究中，我們把廣播頻道視為一個大家都可以存取의公共頻道。

廣播系統的第三個部分就是行動客戶端。行動客戶端的設備可能會有很多種，例如，Notebook，行動電話，PDA，PC 等等。在這些設備上配備有無線網路

的接收器。除此之外，每個行動設備上也會有軟體來監控廣播頻寬中目前的廣播資料是不是自己想要的。以目前的發展進度，這些監控廣播頻寬的軟體十分聰明。他們可以配合廣播節目上的資訊 (例如 index) 來事先知道自己要的資料會在何時廣播出來。因此，他們可以在其他時間把無線網路的接收器關閉或是進入睡眠模式以節省電力 (Imielinski et al.1994,1997)。

個disk。從廣播次數的比例，每個disk可以分成數個chunks。最後，排程式靠交錯每個disk的chunks來產生廣播節目，每個 disk 抓取一個 chunk 稱為小週期 (minor cycle)，等所有 disk 的 chunks 皆播完一次稱為大週期(major cycle)結果如

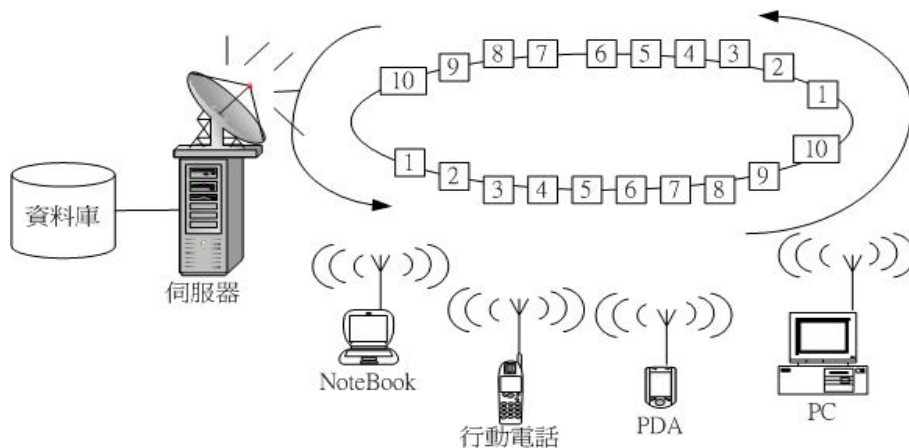


圖 1. 無線資料廣播架構

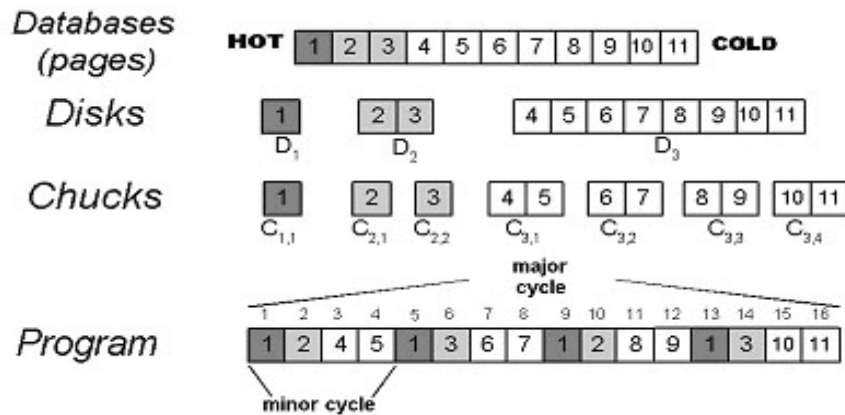


圖 2. 一個 3-Disk broadcast program 的製作範例

二. 存取模式指示器 (AMI) 的設計

在資料廣播的應用程式中，當行動客戶端有要求發生時，行動客戶端不直接向伺服器端發出要求，而是直接從廣播節目中監聽是否有自己需要的資料。而行動客戶端往往無法知道所有的廣播節目內容，因此客戶端是無法自己決定一個有效率的存取模式。為了克服這個問題，我們想到一個解決的方法，就是要求廣播伺服器預先計算行動客戶端可能進入的時間以及可能監聽的時間長度，然後在廣播節目中加入一些訊息來引導客戶端應該如何存取才会有好的效率。為了達到這個目的，我們需要在每頁廣播資料中加入一些資訊，我們稱這個資訊為存取模式指示器 (Access Mode Indicator, AMI)。我們把 AMI 的資料結構畫在圖 3。AMI 由兩個屬性 (attributes) 組成。這第一個屬性是回復時間 T ；第二個屬性是以符合最佳存取模式 (access mode) 的形式 (t_s, t_w) 來表達。當一個 MU 開始去接收廣播資料 (在 t_s 時)，MU 在第一次擷取廣播資料頁時會去查詢 AMI

以獲得一個建議存取方式。然後 MU 在整個 T 期間去擷取廣播資料都會跟隨這個建議的存取方式。讓我們用一些例子來說明 AMI 的使用以及帶來的好處以及更深入的特性探討。例如，MU 有 8 單位的回復時間 (亦即 $T = 8$)，MU 的 $t_s = 1$ 。AMI 建議 (2, 3) 是最佳的存取模式。因此，若 MU 以行動模式聽取 2 單位的時間，然後進入睡眠模式休息 3 個單位時間，重複上述的存取模式直到回復時間 T 結束，MU 將得到最大的效益 (benefit)。假如 MU 開始進入廣播頻道監聽資料的時間是在 page 8 播放的時間，也就是不同的 t_s 。那麼 AMI 建議 (2, 4) 是最佳的存取模式。MU 不同的開始進入聽取資料的時間 t_s 不同就會有不同的存取模式。故如何找出最佳的存取模式是很複雜很耗時間的計算。為了解決此一問題，我們之前的研究 (Chen et al. 2004) 提出了一些探索式演算法 (Heuristic Algorithms)，可以大量減少計算成本 (computing cost) 而能得到接近最佳 (near optimal) 的結果，不過該研究為了簡

化問題,方便該研究的進行,設定存取模式是固定的,也就是每次 MU 以行動模式抓取資料的時間 t_a 固定,同樣 MU 進入睡眠模式以節省電源消耗略過較冷門資料的時間 t_d 也固定。這種固定存取模式的作法有一個缺點,就是當 MU 設

定的工作時間較長時,固定的存取模式將會抓取到重複的資料,導致 AMI 存取資料的效能遞減。這樣的問題在於存取模式是固定,故 MU 沒有辦法以較聰明的方式略過重複的資料不必抓取。

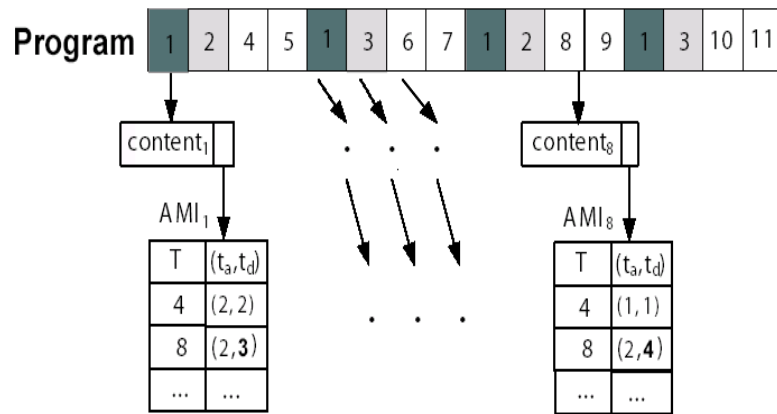


圖 3. Single access mode indicators 的資料結構

此篇論文,我們放寬存取模式必須是固定的限制,而是讓 MU 可以有多組的存取模式(multiple access modes) 如圖 4 所示我們稱為 Flexible access modes indicators 簡稱 FAMI, 這樣 MU 的行為能依據廣播排程播放資料的情況重新調整,如此可更進一步避免 MU 抓取到重複的資料,將有限的電力用在更有效率的資訊存取。為了支援 MU 多組存取模式,AMI table 作了一些必要的修改,詳細結構如圖 4 所示,存取模式改為多組的 (t_a, t_d, cnt) , 代表 MU 連續進行 cnt 次的 (t_a, t_d) 後效能遞減,改成另一組新的 (t_a, t_d, cnt) , 如此持續的變換存取模式直到工作時間 T 結束。以圖 4 為例, 若 MU 進入廣播頻道

開始監聽資料的時間 t_s 為 8, $T = 4$, 則 MU 從抓取到的資料頁查詢 AMI table 的 T 欄位, 可以得到 MU 行為模式的建議值為多個存取模式的集合, 分別為 $(1, 1, 1)$, $(2, 3, 1)$, $(1, 0, 1)$ 。因此 MU 將採取 $(1, 1)$ 的存取模式 1 次, 接下來 $(2, 3)$ 的存取模式一次, 最後 $(1, 0)$ 的存取模式一次, 就能以最省電的方式抓取較熱門的資料。在 AMI 表格裡, T 可以不必徹底的列出所有的可能情況以節省 AMI 的容量。而這個節省 AMI 容量的方法所要付出的代價就是行動客戶端可能無法拿到最有效率的 (t_a, t_d) 。如果一個 MU 的 T 沒有列在 AMI 裡, 則在 AM 中最接近給定的 T 就會被選擇。

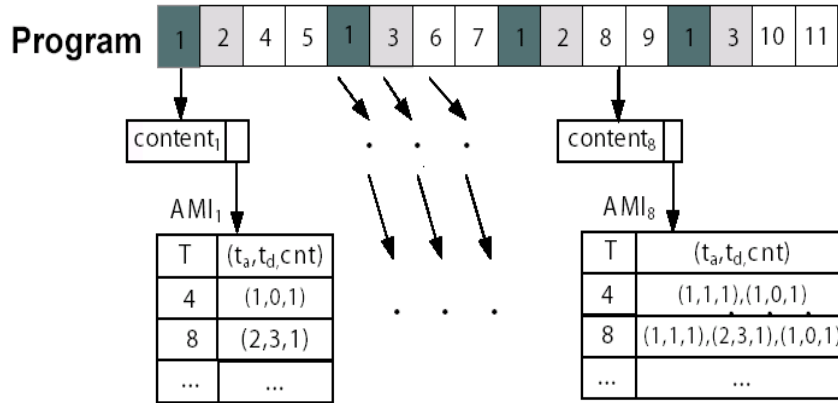


圖 4.Flexible access modes indicators (FAMI) 的資料結構

三. AMI 儲存空間分析

因為 AMI 是我們新增的資料結構，因此我們必須估計它的大小以評估此結構是否可行。如果它的容量太大，整個廣播節目的長度就會明顯增加。結果會造成行動客戶端要花費更長的時間來存取等量的資訊。以下，我們提出一個估計 AMI 容量的方法並且說明 AMI 並不是一個負擔很重的資料結構。我們假設伺服器收集從使用者到備援頻道的往返時間，並且將它們分成 n 個叢集。比起其他叢集的往返時間，每個叢集中的往返時間彼此較接近。在每個叢集裡，伺服器計算其平均往返時間以及那些往返時間出現的平均頻率。接著，伺服器決定提供在 AMI 中， ℓ 個較常被使用的往返時間。

這決定了 AMI table 的長度。在這裡，使用者所需要的 T 是有可能不在 AMI 所提供的列表上的。此情況下，使用者可以選擇最接近 T 的那一個。當然，越大的 ℓ 或許較能符合使用者的需求。在我

們的設計中， ℓ 的值會根據使用者的需求來調整。假設最大平均往返時間為 T_{max} 。因此， T 將最多有 $\lceil \log_2 T_{max} \rceil$ bits。在最糟的情況下， t_a 和 t_d 各自佔用了 $\lceil \log_2 T_{max} \rceil$ bits。然後，我們可以得到，AMI 中一行的空間大小有 $3 \times \lceil \log_2 T_{max} \rceil$ bits。因此，在一個廣播頁中，AMI 的大小(記成 Γ_{page})能用下面的方程式計算。

$$\Gamma_{page} = (3 \times \lceil \log_2 T_{max} \rceil) \times \ell \times \frac{1}{8} \text{ bytes}$$

，在一個廣播排程中，整個 AMI 的大小(記成 $\Gamma_{program}$)為所有 Γ_{page} 的總和，也就是

$$\begin{aligned} \Gamma_{program} &= \Gamma_{page} \times |B| \\ &= (3 \times \lceil \log_2 T_{max} \rceil) \times \ell \times \frac{1}{8} \times |B| \text{ bytes} \end{aligned}$$

其中 $|B|$ 是廣播排程(broadcast program)的所有資料頁，故額外的儲存負擔(storage overhead)可以表達成 AMI 的大小(size)除以原始的廣播排程的大小，也就是

$$\frac{\ell \times \lceil \log_2 T_{max} \rceil \times |B|}{|B| \times page_size} = \frac{\ell \times \lceil \log_2 T_{max} \rceil}{page_size}$$

以下我們用一個例子來說明 AMI 並不是一個重負載的資料結構。假設 $|B|=2048$, $\ell=10$, $T_{max}=1024$ 。每個廣播頁的 AMI 大小 (Γ_{page}) 只有 $((3 \times 10) \times 10) / 8$ 約 37 bytes, 對一個實際的應用來說算是相當小的了。即使是整個廣播排程的所有 AMI 大小也只用了 2048×37 約 75 KB。如果一個廣播頁的大小大約 4 KB, 那麼廣播排程中 AMI 所造成的額外負擔也只有 $\frac{2048 \times 37 \text{ bytes}}{2048 \times 4 \text{ KB}} < 1\%$

換句話說, 在每個資料頁中加入輔助資料 AMI, 對原本的廣播排程, 會帶來少於 1% 的額外負擔。這樣小的負擔在實際的系統中, 是可以忽略且肯定能被大多數使用者接受。即使在其他合理的設定下, 我們可以得到 storage overhead 不會很大而且可以被接受, 也因為如此,

我們可以假設資料內容跟 AMI 可以在一個廣播時間內被成功播出。至於可以支援多組使用者存取模式的 FAMI 所佔的額外的儲存負擔 (storage overhead) 比起 AMI, 不超過系統使用的廣播磁碟個數乘 2, 以圖 2, disk 數是 3 的話, 則 FAMI storage overhead 不超過 6%, 由於篇幅的關係, 在此我們不說明整個推導過程, 有興趣的讀者可參考 (Lin et.al.2005)。

肆. 評估模式與演算法設計

一. 評估模型 (Evaluation Model)

從行動使用者的角度來看, 具有最大利益的存取模式是不管使用者從什麼時候切入資料廣播系統, 都可以使用最少量的連線時間在監聽廣播節目並且得到最多最有價值的資料。因此, 我們可以將這個問題正式的描述如下:

Problem Formulation:

Given a broadcast disk program B , MU's turnaround time T , and MU's start time t_s .

Find the $(t_a, t_d)^{opt}$ pairs such that the *Benefit* is maximized.

Benefit 是一個用來評量一個 access mode 有多好的評量標準。基本上, *Benefit* 越大, 這個存取模式越好。從這個觀點, 我們可以發現兩個事實:

事實 1: 全部的斷線時間愈久, *Benefit* 越大。也就是 *Benefit* 跟斷線時間成正比。

事實 2: 存取到的資料頁愈重要, *Benefit* 越大。也就是 *Benefit* 跟存取到的

資料頁的廣播頻率的總和成正比。在此我們只考慮抓取到的所有相異資料, 因為重複的資料並不會提供更多的資訊。故我們將 *Benefit* 定義成:

$$\text{Benefit} = (\text{total doze mode time}) \times (\text{sum of the frequency of distinct pages})$$

。我們將根據這些觀察到的事實 (facts), 來設計怎樣能達到對客戶端最有利益 (benefit)。

二. 多重存取模式的動機

我們之前的研究首先提存取模式指引器的方法(AMI approach)(Chen et al.2004)，利用 AMI 來提供 MUs 最佳 的行為模式。然而 AMI 的方法法，存取模式是固定不能調整。當 MU 經過幾次的存取模式以後，將造成 MU 浪費一部分的連線時間抓取重複的資料，不但消耗 MU 的電力，同時重複資料沒帶來任何新的資訊對滿足 MU 的需求亦沒有幫助。因此對 MU 而言，一個好的存取行為，應該是每次的連線抓取皆能帶來新資訊，不能造成資料的重複抓取。MU 應該在適當地方(如抓取的資料重要性遞減或抓取到重複資料)適當調整存取行為，以確保該存取行為的有效性。以圖 2 為例，假設 $T = 15$ ， $t_s = 1$ 若存取模式 固定不變則 (2, 4) 是最佳的存取模式，得到 $benefit = 90$ ，若採用多重存取模式將存取模式改成 (2, 4), (2, 5)，則 $benefit = 99$ ，此結果說明 多重存取模式帶來的 $benefit$ 很明顯優於單一存取模式。

這種可隨時根據存取行為的效能調整成另一組不同的存取行為的方法，我們稱為 FAMI。廣播伺服器則利用 FAMI 提供 MUs 多重存取模式的建議值。就 FAMI 而言，一個最直覺(naïve)的方法就是測試所有的存取模式，找出多組存取模式的集合，使得 MU 可以，以最少的電力在有限時間抓取更多、更重要資料。然而，找出最佳多組存取模式的窮舉查詢(exhaustive search)的方式比起 AMI 方法所需的計算成本更高。因為要找出多組存取模式比起 AMI 方法更加的複雜，FAMI 沒辦法事先知道到底存取模式該變化兩次或是三次，甚至更多次最好，因此必須測試所有可能的存取模式的組合(2 組, 3 組, ...)，又每一組 存取模式(t_a, t_b) 的值到底為何也沒辦法事先得知。因此計算成本遠遠超過 AMI approach，就多組存取模式而言，窮舉查詢的方法是不可行的。甚至連任一選取(random)的方式也不行。就 AMI 方法而言，隨機選取的方法，是任選一組 (t_a, t_b) 計算 $benefit$ ，重覆此行為直到統計上具意義

實驗次數為止(Ross 1987)。然而 FAMI 不只一組存取模式，而是多組存取模式。因此一個較長的(t_a, t_b)的選取範圍被分割成數份，若不同的存取模式其隨機選取的(t_a, t_b)的有效範圍也將不同，造成隨機方法很難實行。故如何找出最佳化的 FAMI 相當困難。在此研究裡面，我們從 $Benefit$ 的定義找出簡易的規則，依此規則利用 Greedy Approach，設計出簡單且有效率的演算法。詳細的演算法將在下一節介紹。

三. 多重存取模式演算法

我們的目標是要找出 $benefit$ 最佳者，故簡易的規則就是當 MU 抓取此資料頁 data page 造成新的 $benefit$ (new benefit)大於尚未抓取此頁時的 $benefit$ ，則抓取此頁明顯會為 MU 帶來更多 $benefit$ ，故 MU 必需抓取此頁。為了讓整個存取模式的組合能得到最佳 $benefit$ ，我們的作法是從帶來最多 $benefit$ 的資料先抓，也就是先抓取高廣播頻率的資料。因此我們將資料頁按照廣播頻率由大到小排序(其實廣播磁碟本身已經排好， $disk1 > disk2 > disk3$)，然後依序抓取。故抓取的順序就是 $disk1$ 先抓取，當 $disk1$ 所有 pages 抓完再抓取 $disk2$ ，最後再抓取 $disk3$ 。抓取的過程若不能增加 $benefit$ 就停止。故最佳的抓取行為可看成 $disk1, disk2, disk3$ 各該抓多少頁。如此只要 T 給定後我們就能得到最佳抓取資料的組合為必須從各個 $disk$ 分別抓取多少個資料頁，再根據 MU 的 t_s 將上面的資訊轉成相對的存取模式。

底下我們利用圖 2 以實例的方式，說明我們的演算法如何找出 maximal $benefit$ 。首先，將所有廣播資料頁依照廣播頻率排序，故資料抓取的順序將為 $disk1, disk2, disk3 \cdot T$ 等於 total active time 加上 total doze mode time，故計算最佳 $benefit$ 的方式，可以利用窮舉法(exhaust search)的方式列舉所有 total active time 與 total doze mode time 的組合直到抓取的資料不會再增加 $benefit$ 為止。因為我們抓取資料的順序是由廣播頻率高的資料開始，若開始時的 $benefit$ 是增加的，代表消耗一

單位的電力來抓取一頁是值得的，但當 benefit 不再增加時代表以後再抓取一頁不可能帶來更大的 benefit，因為再抓取一頁 total disconnect 將減少，不利於 benefit，同樣的再抓取一頁該頁的廣播頻

率亦低於前次抓取的頁，亦不利於 benefit 故以後將不會出現更大的 benefit.

假設 $T = 16$, $t_s = 1$ ，則找出最佳存取行為的過程將如下：

Run1：先抓取 disk1 第 1 個 page，則 $\text{benefit} = (1 \times 4) \times (15) = 60$

Run2：disk1 只有一個廣播頁，故 disk1 資料已抓取完畢，開始抓取 disk2 第 1 個 page，則 $\text{benefit} = (1 \times 4 + 1 \times 2) \times (14) = 84$

Run3：繼續抓取 disk2 第 2 個 page，則 $\text{benefit} = (1 \times 4 + 2 \times 2) \times (13) = 104$ Run4：disk2 有 2 個廣播頁，故 disk2 資料亦已抓取完畢，接下去繼續抓取 disk3 第 1 個 page， $\text{benefit} = (1 \times 4 + 2 \times 2 + 1 \times 1) \times (12) = 108$

Run5：繼續抓取 disk3 第 2 個 page， $\text{benefit} = (1 \times 4 + 2 \times 2 + 2 \times 1) \times (11) = 110$

Run6：繼續抓取 disk3 第 3 個 page， $\text{benefit} = (1 \times 4 + 2 \times 2 + 3 \times 1) \times (10) = 110$

到第 5 個 Run，也就是花 5 單位的 active 時間，抓取 1 個 disk 1 的資料頁，2 個 disk 2 的資料頁，以及 2 個 disk 3 的 data pages 得到的 benefit 與第 6 個 Run 的結果相同，亦即 benefit 已經不再增加，因此 Run5 抓取資料的方式即為最佳的存取行為。這樣的抓取資料行為的存取行為可用 $\text{DiskAccessMode}(\text{num}_1, \text{num}_2, \dots, \text{num}_n)$ 表示，這代表若要產生對使用者最大的 benefit 則必需從 disk1 抓取 num_1 個 pages，從 disk2 抓取 num_2 個 pages， \dots ，以及從 disk n 抓取 num_n 個 pages。因為我們抓取資料的方式是一個 page 一個

page 抓取，因此每經一個 Run，雖然 total frequency 將增加，但相對 total doze mode time 會遞減，一開始抓取的資料頁有較大廣播頻率故 benefit 會跟著增加，但慢慢的廣播頻率增加的幅度將減少但 total doze mode 亦減少，最後將達到 benefit 不再增加。故最後得到最大 benefit 的方式是 MU 必需分別從 Disk1, Disk2, Disk3 抓取(1, 2, 2) 頁的組合，最後根據 MU's 的 t_s 將(1, 2, 2) 組合換算成(4, 1, 1)，(1, 10, 1) 的存取模式。詳細的轉換存取模式的演算法如演算法 1。

```

INPUT:  $\mathcal{B}$ ,  $t_s$ , DiskAccessMode[num1, num2, ..., numk]LIST;
OUTPUT: AccessModes[ta, td, cnt] LIST;
1: initial StartTime= $t_s$ , num=0,  $t_a=0$ ,  $t_d=0$ , cnt=1;
2: initial boolean access array AccessArray[] and set to zero;
3: Get the data begin at time slot  $t_s$ ;
4: while One of element of the DiskAccessMode[ ] !=0 do
5:   Compute the Disk Numer  $i$  of the page at time slot  $t_s$ ;
6:   if  $num_i > 0$  then
7:     AccessArray[ $t_s$ ]=1;
8:      $num_i - -$ ;
9:   end if
10:   $t_s++$ ;
11: end while
12: for  $j=StartTime$  to  $t_s$  do
13:   if AccessArray[ $j$ ] = 1 then
14:     if AccessModeFlag Is False then
15:        $t_a++$ ;
16:     else
17:       if [ $t_a, t_d$ ] access mode pair can be found in AccessMode[ $t_a, t_d, cnt$ ] LIST then
18:         ADD one to the cnt of the AccessMode[ $t_a, t_d, cnt$ ];
19:       else
20:         OUTPUT an new AccessMode[ $t_a, t_d, 1$ ];
21:       end if
22:        $t_a=0, t_d=0, AccessModeFalg = False$ ;
23:        $t_a++$ ;
24:     end if
25:   else
26:      $t_d++$ ;
27:     AccessModeFlag = True;
28:   end if
29: end for

```

演算法 1. Transfer DiskAccessMode 成爲 access modes.

伍. 模擬與實驗結果

一. 實驗設計

圖 5 描述我們效能研究(performance

study)的模擬模式(simulation model)。行動資訊系統由廣播伺服器與行動客戶端兩個部份構成，詳細的運作情形說明如下。

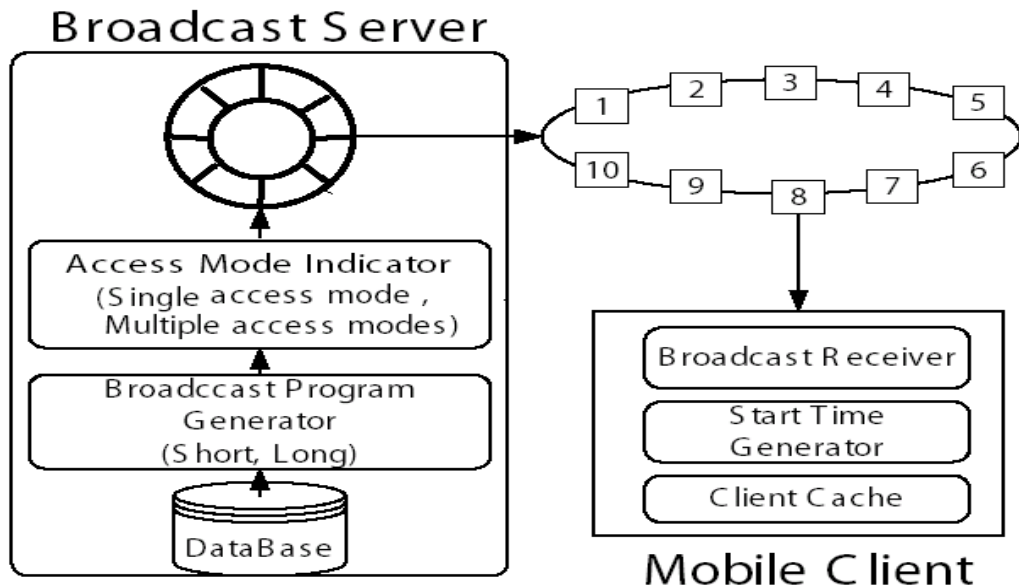


圖 5. Simulation Model

(一) 廣播伺服器(Broadcast server)

在圖 5 中 廣播伺服器 下方是一包含所有要廣播資料的資料庫。這些資料頁將被利用來產生我們實驗中所需的廣播排程。廣播排程產生器產生 Short 與 Long 兩種型態的 廣播排程。在 Short Program 我們以縮寫 *SP* 來代表，這類型排程其廣播磁碟之間的廣播頻率是一 simple ratio $x^n : x^{n-1} : \dots : x:1$ (Acharya et al.1995, 1996a)。若廣播磁碟之間的資料頁不是採取 simple ration 的方式則廣播排程將會比較長造成 longer broadcast program, 在此我們稱為 Long Program, 以縮寫 *LP*表示。在我們的 performance study 中, *LP*的長度約為 *SP*長度的兩倍。在實驗中使用兩種型態的 broadcast programs 可以使我們的實驗結果更具一般性與更具代表性。廣播排程產生器產生的 廣播排程 被送到 AMI generator。AMI generator 在每一個資料頁加入 AMI table。加入 AMI 的方式分成 single access mode 與 multiple access modes 兩種方式。

表格 1 是廣播伺服器相關參數表格，說明有關我們效能評估中所使用的參數與其代表的意義，相關的參數設定是參考有關廣播磁碟技術方面的重要論文(Katz 1994; Gifford 1985; Acharya 1997; Barbara 1999)。NumDisks 代表廣播伺服器中廣播磁碟的個數，設定值為 5。DiskSize_i 代表每一個 disk 的資料頁數。不管 *SP* 或 *LP* 為了方便比較其差異故全部的頁數設定為相等約 1600 頁。至於每一個 disk 包含的頁數如表 1 所示，其基本原則是存取機率愈 hot 的 disk 其頁數愈少，存取機率愈 cold 的 disk 其頁數愈多。至於如何計算出每一 disk 所含的 page 數則是根據每一 disk 的 DiskFreq_i 按照產生廣播磁碟排程的程式產生，詳細的演算法可參考(Acharya et al.1995, 1996a)。

表格 1：廣播伺服器 的參數與設定.

參數	描述	設定
$NumDisks$	廣播磁碟的磁碟個數	5
$DiskSize_i$	每個磁碟的資料頁數， $1 \leq i \leq NumDisks$	$SP: 10, 40, 140, 360, 1040$ $LP: 35, 90, 315, 420, 730$
$DiskFreq_i$	每個磁碟的廣播頻率， $1 \leq i \leq NumDisks$	$SP: 16 : 8 : 4 : 2 : 1$ $LP: 9 : 7 : 5 : 3 : 1$

(二) 行動客戶端(Mobile Client)

圖 5 的 broadcast receiver 是 client 用來從無線頻道(wireless channel) 抓取廣播資料頁。這些被抓取的資料被使用端快取(client cache)管理與儲存。start time generator 用來產生不同的 start time (t_s) 與 turnaround time(T)。表格 2 列出實驗中 client 端所用到的參數與預設值。turnaround time T 跟據不同的 program topology 而變化。這樣的設計是為了能儘

可能符合多數 MUs 的需求，可以有組 working time T 來滿足 MUs。基本上 start time t_s 可以是 major cycle 的任一時間點，這將會造成 t_s 有太多的選擇，將造成實驗太複雜。為了簡化此一工作，我們只考慮 MU 只從 hot, medium, cold 這 3 個時間點進入，也就是 MU 只會從 broadcast disk 1, 3, or 5 開始抓取資料。

表格 2：MU 的參數與設定

參數	設定
T	$SP: 從 600 變動到 2800$ $LP: 從 600 變動到 4800$
t_s	(hot, medium, cold)

二. 比較效能的方法

關於實驗的效能評估，我們提出的多組存取模式只與單一存取模式中最佳的方法比較，因為其它 Heuristic Algorithms 只能得到接近最佳化的結果，若多組存取模式方法獲得的 benefit 大於單一存取模

式的結果，則說明多組存取模式方法會優於所有單一存取模式的演算法。

三. 實驗的分類

為了實驗的完整性，我們實驗的設計同時考慮 Server 與 行動客戶端 的因素，針對 Program Topology 與 MU 的 Start time 兩個維度加以分類，在 Program

Topology 則根據播放 Disks 的 frequency 是否成比例關係分成 Short Program 與 Long Program，而 MU' s start time 的維度

又根據進入點抓到的 data page 的受歡迎程度分成 Hot，Medium，Cold 三個等級。詳細分類如圖 6 所示。

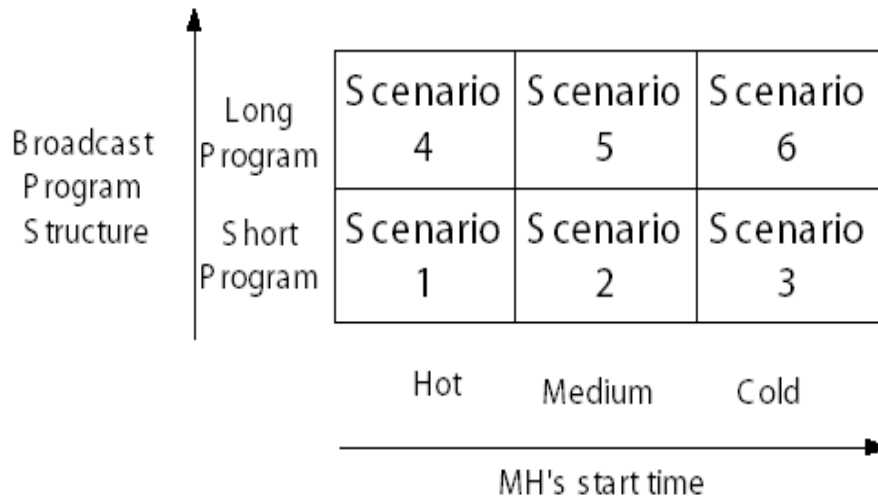


圖 6.實驗分類

四. 各演算法的效能

這個實驗主要在比較 AMI 方法採取的單一存取模式的方法與 FAMI 方法採取的多組存取模式方法在效能上的差異，對於單一存取模式我們只取最佳的結果加以比較，因為 FAMI 方法的多組存取模式的方法若能優於 AMI 方法的最佳結果，則 FAMI 方法必優於任何 AMI 方法的方法。為了達到統計上足夠代表性的實驗資料我們的模擬次數進行了 30 次(Ross 87)。另外為了突顯 turnaround time T 的長短對不同方法帶來的影響，我們依據不同的 program topology SP 與 LP 各分成兩組來討論 Short-1，Short-2 與 Long-1，Long-2。Short-1 與 Long-1 代表 turnaround time T 的設定值較小。反之 Short-2 與 Long-2 則 turnaround time T 的設定值較大。

實驗結果如圖 7 所示。圖 7(a)，圖 7(b) 與圖 7(c)，圖 7(d) 分別是 Short Program (SP) 與 Long Program(LP) 的實驗結果。X 軸代表 total turnaround time T，Y 軸代表 benefit，由結果可很明顯看出不管 SP 或 LP，多組存取模式皆優於單一存取模式。

當 T 愈大，效果愈明顯(如圖 7(b)、圖 7(d))。這是因為當 T 夠大時，單一存取模式因 access mode 是固定的，無法完全避免抓取重複無用的資料頁，重複的頁不但沒有帶來任何新資訊反而還浪費電力，造成 benefit 不升反降。而多組存取模式卻能完全避免抓取重複的資料頁，而能得到最佳的 benefits。

除此之外，我們看到一個現象是多組存取模式不管 MU 開始進入廣播頻道監聽資料的時間區域是 hot，medium 或 cold，多組存取模式總是能達到最佳的 benefit 不會受 MU' s 進入廣播頻道不同時間點的影響。反之，單一存取模式則會受到不同時間進入點的影響，大至上 MU 的進入點是 hot area 會有較佳的 benefit 表現。故多組存取模式針對不同 MU' s 進入點帶來對 benefit 的影響，比起單一存取模式更穩定(stable)，這符合一個優良演算法的準則。這也更證明我們所提出的 FAMI 方法更具一般性，就廣播伺服器方面可適合不同的 Program Topology(SP or LP)。就 MU 端而言則不管使用者進入公共頻道聽取廣播資料的時

間點爲何，FAMI 方法皆能根據使用者的狀況來調整使用者的資料存取模式，讓使

用者能得到最大 *benefit*。

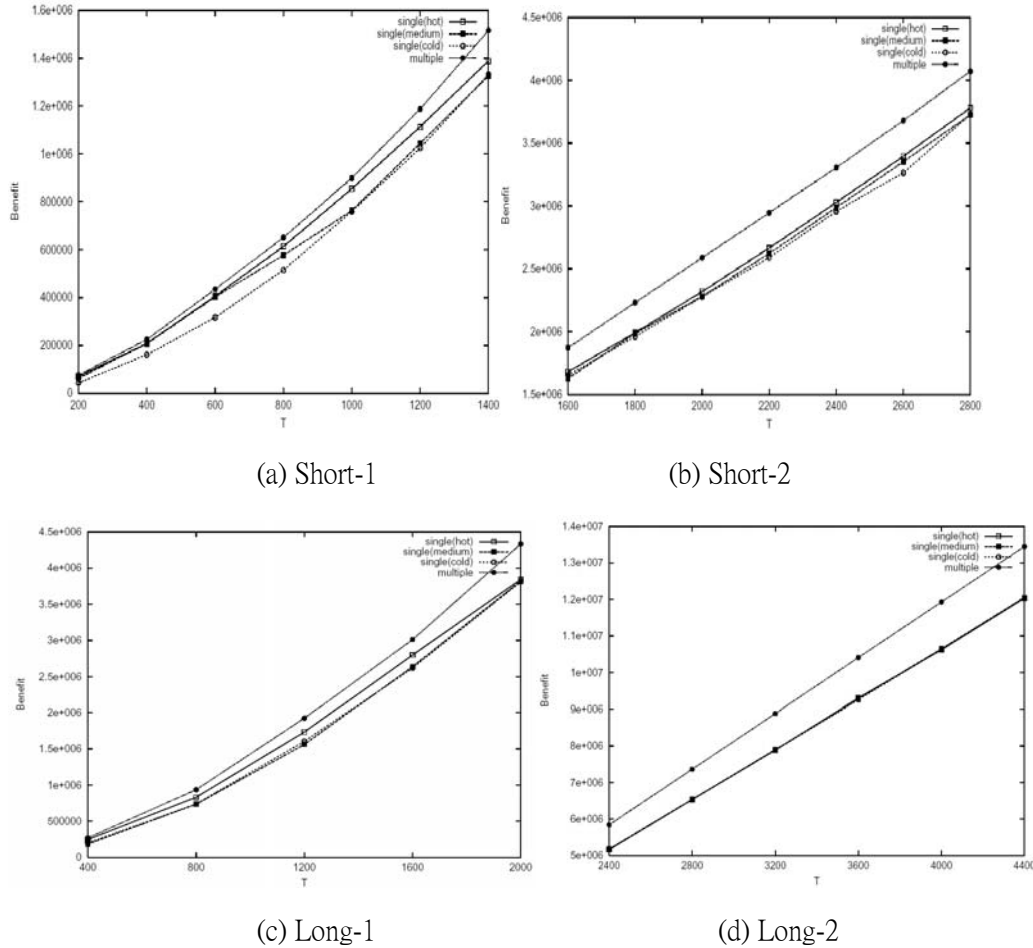


圖 7. AMI 與 FAMI 的 benefit 比較

陸. 結論與未來展望

無線通訊環境下，無線的特性提供我們可在任何時間任何地點抓取資料進行我們的工作，帶給我們相當大的便利性。然而無線的環境下經常性預期與非預期的斷線以及行動裝置的電力資源有限的情況下，在設計無線資訊系統時考慮與適當的解決這些問題是很重要的。這篇論文中，我們研究如何安排廣播排程來讓廣

播系統可以主動的提供 MH 適當的存取模式。MH 只要依據這樣的存取模式就能以最符合使用利益(benefit)的方式抓取資料。在此我們改良單一存取模式的 AMI 方法成爲多組存取模式，經由實驗證明，多組存取模式確實優於單一存取模式。目前我們只處理單一廣播頻道的問題，未來我們希望將此問題擴大到多個廣播頻道上。

柒. 參考文獻

S. Acharya, R. Alonso, M. Franklin, and S. Zdonik, "Broadcast Disks: Data Management for Asymmetric Communications Environments", in *Proceedings of the ACM SIGMOD Conference*, May 1995, San Jose, CA, USA, pp.199-210.

S. Acharya, M. Franklin, and S. Zdonik, "Dissemination-Based Data Delivery Using Broadcast Disks", *IEEE Personal Communications*, 2(6), 1995, pp. 50-60.

S. Acharya, M. Franklin, and S. Zdonik, "Dissemination Updates on Broadcast Disks", in *Proceedings of the 22nd VLDB Conference*, September 1996, Mumbai(Bombay), India, pp. 354-365.

S. Acharya, M. Franklin, and S. Zdonik, "Prefetching from a Broadcast Disk", in *Proceedings of the 12th International Conference on Data Engineering*, February 1996, New Orleans, Louisiana, pp. 276-285.

S. Acharya, M. Franklin, and S. Zdonik, "Balancing Push and Pull for Data Broadcast", in *Proceedings of ACM SIGMOD Conference*, May 1997, Tuscon, Arizona, p.p. 183-194.

AirMedia Live Internet Broadcast Networks, <http://www.airmedia.com/>,

1997.

D. Barbara, "Mobile Computing and Databases - a Survey", *IEEE Transactions on Knowledge and Data Engineering*, 11(1), 1999, pp. 108-117.

C. Baquero, V. Fonte, F. Moura, and R. Oliveira, "MobiScape: WWW Browsing under Disconnected and Semi-Connected Operation", in *Proceeding of the First Portuguese WWW National Conference*, Braga, Portugal, July 1995.

T. F. Bowen, G. Gopal, G. Herman, T. Hickey, K. C. Lee, W. H. Mansfield, J. Raitz, and A. Weinrib, "The Datacycle Architecture", *Communications of the ACM*, vol. 35, no. 12, December 1992, pp. 71-81.

Cell Broadcast Forum, "<http://www.cellbroadcastforum.org/>", 2001.

S. Chang and D. Curtis, "An Approach to Disconnected Operation in an Object-Oriented Database", in *Proceedings of the Second International Conference on Mobile Data Management (MDM02)*, Singapore, January 8-11, 2002, p.p. 19-26.

C. C. Chen, "Compression-Based Broadcast Data for Reducing Access

- Time in Wireless Environment", in *Proceeding of 1999 National Computer Symposium (NCS'99)*, Taipei, Taiwan, R. O. C., December 1999, pp. C539-C546.
- C. C. Chen, C. Lee, and C. H. Ke, "Compress-Based Broadcast Strategies in Wireless Information Systems", in *Proceedings of the International Conference on Advanced Information Networking and Applications (AINA' 2003)*, March 2003, Xi'an China.
- C. C. Chen, L. F. Lin, and C. Lee, "Supporting Benefit-Oriented Retrieval for Data on Air", in *Proceedings of the 9th International Conference on Database Systems for Advanced Applications (DASFAA04)*, March 2004, p. p. 915-921.
- I. R. Chen, N. A. P., and I. L. Yen, "Algorithms for Supporting Disconnected Write Operations for Wireless Web Access in Mobile Client-Server Environments", *IEEE Transactions on Mobile Computing*, Vol. 1, No. 1, 2002, p. p. 46-58.
- A. Datta, D.E. Vandermeer, A. Celik, and V. Kumar, "Broadcast Protocols to Support Efficient Retrieval from Database by Mobile Users", *ACM Transactions on Database Systems*, Vol. 24, No. 1, March 1999, p. p. 1-79.
- Microsoft Corporation. What is directband network? "<http://www.microsoft.com/spot/direct.msp>", 2003.
- David K. Gifford, "The Application of Digital Broadcast Communication to Large Scale Information Systems", *IEEE Journal on selected areas in communications*, vol. 3, no. 3, May 1985, pp. 457-467.
- David K. Gifford, "Polychannel Systems for Mass Digital Communications", *Communications of the ACM*, vol. 23, no. 2, February 1990, pp. 141-151.
- J. L. Huang and M. S. Chen, "Adaptive Balanced Hybrid Data Delivery for Multi-Channel Data Broadcast", in *Proceedings of IEEE ICC' 2002*, April 2002.
- C. L. Hu, M. S. Chen, "Dynamic Data Broadcasting with Traffic Awareness", in *Proceedings of ICDCS 2002*, pp. 112-119.
- J. H. Hwang, S. Cho, and C. S. Hwang, "Optimized Scheduling on Broadcast Disks", in *Proceedings of the Second International Conference on Mobile Data Management (MDM01)*, Hong Kong, China, January 8-10, 2001, p. p. 91-104.

- Q. Hu, W. C. Lee, and D. L. Lee, "Indexing Techniques for Wireless Data Broadcast under Data Clustering and Scheduling", in *Proceedings of the 1999 ACM CIKM International Conference on Information and Knowledge Management (CIKM99)*, Kansas City, Missouri, USA, November 2-6, 1999, p.p. 351-358.
- Q. Hu, W. C. Lee, and D. L. Lee, "Power Conservative Multi-Attribute Queries on Data Broadcast", in *Proceedings of the 16th International Conference on Data Engineering (ICDE' 2000)*, San Diego, California, USA, February 2000, pp. 157-166
- Q. Hu, W. C. Lee, and D. L. Lee, "A Hybrid Index Technique for Power Efficient Data Broadcast", *Distributed and Parallel Database*, Vol. 9, No. 2, 2001, p.p. 151-177.
- Q. Hu, W. C. Lee, and D. L. Lee, "Index Techniques for Power Management in Multi-Attribute Data Broadcast", *Mobile Networks and Applications (MONET)*, Vol. 6, No. 2, 2001, p.p. 185-197.
- S. Hameed and N. H. Vaidya, "Efficient Algorithms for Scheduling Data Broadcast", *ACM/Baltzer Wireless Networks*, 5(3):183-193, 1999.
- Q. Hu, Xu, J., Wong, W.S., Zheng, B., D. L. Lee, and W. C. Lee, "Proactive Caching for Spatial Queries in Mobile Environments", *Proceedings of the International Conference on Data Engineering (ICDE' 05)*, Tokyo, Japan, 2005. (to appear)
- T. Imielinski, S. Viswanathan, and B. R. Badrinath, "Energy Efficient Indexing on Air", in *Proceedings of the ACM SIGMOD Conference*, May 1994, Minneapolis, Minnesota, USA, pp. 25-36.
- T. Imielinski, S. Viswanathan, and B. R. Badrinath, "Data on Air: Organization and Access", *IEEE Transactions on Knowledge and Data Engineering*, 9(3), 1997, pp. 353-372.
- Randy H. Katz, "Adaption and Mobility in Wireless Information Systems", *IEEE Personal Communications*, vol. 1, no. 1, 1994, pp. 6-17.
- Geoffrey H. Kuenning and Gerald J. Popek, "Automated Hoarding for Mobile Computers", in *Proceedings of the Sixteenth ACM Symposium on Operating Systems Principles*, 1997, p.p. 264 - 275.
- James J. Kistler and M. Satyanarayanan, "Disconnected Operation in the Coda File System", *ACM Transactions on*

- Computer Systems*, Vol. 10, No. 1, February 1992, p.p. 3-25.
- S. Khanna and S. Zhou, "On Indexed Data Broadcast", in *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, May 1998, p.p. 463-472.
- S. C. Lo and L. P. Chen, "Optimal Index and Data Allocation in Multiple Broadcast Channels", in *Proceedings of the 16th International Conference on Data Engineering*, February 2000, San Diego, California, pp. 293-302.
- L. F. Lin, C. C. Chen, and C. Lee, "Optimal Data Retrieval in Data Broadcast Environment", Technical Report 95-10-12, Department of Computer Science and Information Engineering, National Cheng-Kung University, Taiwan, 2005.
- M. A. Viredaz, L. S. Brakmo, and W. R. Hamburger, "Energy management on handheld devices", *ACM Queue*, 1(7), October 2003, pp. 44-52.
- S. H. Phatak and B. R. Badrinath, "Data Partitioning for Disconnected Client Server Databases", in *Proceedings of the ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE99)*, 1999, p.p. 102-109.
- W. C. Peng and M. S. Chen, "Dynamic Generation of Data Broadcast Programs for a Broadcast Disk Array in a Mobile Computing Environment", in *Proceedings of the ACM 9th International Conference On Information and Knowledge Management*, November 2000, McLean Virginia, USA, pp. 6-11.
- Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", *John Wiley & Son Inc.*, 1987.
- S. Shekhar, A. Fetterer and D. R. Liu, Genesis: An Approach to Data Dissemination in Advanced Traveler Information Systems, *Data Engineering Bulletin*, vol. 19, No. 3, 1996, pp. 40-47.
- StarBand. <http://www.starband.com/>.
- N. Shivakumar and S. Venkatasubramanian, "Energy-Efficient Indexing for Information Dissemination in Wireless Systems", *ACM-Baltzer Journal of Mobile Networks and Nomadic Applications (NOMAD)*, 1996.
- P. Triantafillou, R. Harpantidou and M. Paterakis, "High Performance Data Broadcasting: a Comprehensive System's Perspective", in

Proceedings of the Second International Conference on Mobile Data Management (MDM01), Hong Kong, China, January 8-10, 2001, p. p. 79-90.

Mark Weiser, "Some Computer Science Problems in Ubiquitous Computing", *Communications of the ACM*, July 1993

M. Weiser, "Hot Topics: Ubiquitous Computing", *IEEE Computer*, October 1993.

J. Xu, Q. L. Hu, W.-C. Lee, and D. L. Lee, "Performance evaluation of an optimal cache replacement policy for wireless data dissemination", *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 16(1), January 2004, pp. 125-139.

J. Xu, W. C. Lee, and X. Tan, "Exponential Index: A Parameterized Distributed Indexing Scheme for Data on Air", in *Proceedings of the Second International Conference on Mobile Systems, Applications, and Services (MobiSys04)*, June 2004, Boston, MA, p. p. 153-164.

J. Xu, B. Zheng, M. Zhu, and D. L. Lee, "Research Challenges in Information Access and Dissemination in a Mobile Environment", in *Proceedings of the Pan-Yellow-Sea International*

Workshop on Information Technologies for Network Era (PYIWIT'02), Saga, Japan, March, 2002, pp. 1-8 (Invited Paper).

E. Yzjima, T. Hara, M. Tsukamoto, and S. Nishio, "Scheduling and Caching Strategies for Broadcasting Corrected Data", in *Proceedings of the 16th ACM Symposium on Applied Computing*, pages 504-509, March 2002.

B. Zheng, and D. L. Lee, "Information Dissemination via Wireless Broadcast", *Communication of ACM (CACM)*, 48(5), May 2005, pp.105-110.

B. Zheng, J. Xu, W. C. Lee and D. L. Lee, "Energy Conserving Air Indexes for Nearest Neighbor Search", *Proceedings of the 9th International Conference on Extending Database Technology*, Heraklion, Crete, Greece, March 2004, pp. 48-66.

作者簡介



林聯發

國立成功大學資訊工程所博士班，現任教於高苑科技大學資訊傳播學系，研究興趣領域為資料庫系統、行動資料管理以及 Web 應用系統設計與開發