

行政院國家科學委員會專題研究計畫 期中進度報告

行動網際網路環境中同儕即時服務之研究(1/2)

計畫類別：個別型計畫

計畫編號：NSC94-2213-E-009-108-

執行期間：94年08月01日至95年07月31日

執行單位：國立交通大學資訊工程學系(所)

計畫主持人：曹孝櫟

計畫參與人員：鄭建明、謝凱翔、周金璋、楊雅筑、黃名杰

報告類型：精簡報告

處理方式：本計畫可公開查詢

中 華 民 國 95 年 5 月 30 日

中文摘要

隨著個人電腦計算能力的提升與寬頻網路的快速發展，網際網路應用服務技術逐漸從主從式架構演進到同儕計算 (Peer-to-Peer、P2P Computing)，P2P 技術無論在學術研究上及應用方面都受到相當的重視。然而考慮同儕計算將伺服器工作分擔至客戶端的特性，必將增加終端裝置對記憶體、網路頻寬與計算能力的需求。本計畫主要在研究同儕行動服務之即時網路通訊技術，考慮在不影響即時網路服務品質的情況下，降低行動裝置對計算、網路頻寬和記憶體等的需求。計畫的第一年著重於研究在同儕行動環境 (Peer-to-Peer Mobile Computing Environment) 中進行資源搜尋的效能評估，並分析不同的節點識別對應方式對搜尋過程所造成的影響，最後推導出數學公式並用模擬方式來驗證其正確性。

關鍵詞：同儕計算、即時服務、行動通訊與無線網路

Abstract

The advances in PC and broadband networking technologies certainly lead to peer-to-peer (p2p) services and applications over Internet. Peer-to-peer computing is not only a hot academic research topic but also an attractive technique applied to Internet applications. However, p2p technology that distributes server loads to clients has increased the memory, network bandwidth and computation needs of mobile devices. This project aims to study p2p real-time networking technologies in a mobile Internet environment to reduce resource usages and requirements of mobile devices running p2p real-time applications. In the first year of this project, we focus on study the performance analysis of resource discovery in peer-to-peer mobile computing environment. We also discuss the effects of host identifier mapping on resource lookup. Finally, we use simulation to verify the correctness of our analysis.

Keywords: Peer-to-Peer Computing, Real-Time Services, Mobile Communication and Wireless Networking

一、前言

早期網際網路應用程式的發展因終端設備及網路頻寬的能力限制，所以採取主從式

(Client-Server) 架構為開發主流，但近幾年來科技的進步使得一般個人電腦的運算能力及頻寬比以前伺服器的能力可說是有過之而無不及，網路程式的開發也漸漸由具有高度可擴充能力及可快速部建的 P2P 架構所取代。

在此同時無線網路的終端設備不管在無線頻寬及計算能力皆大大提昇，使得有越來越多的網際網路應用程式從有線網路的終端設備移植到無線網路的終端設備，因此我們不難預見到在未來世界裡，將會有 P2P 應用程式移植到手持型設備並在行動網路中出現。

檔案分享 (File-Sharing) 是近年來盛行的 P2P 應用，依照其系統運作方式，可以分為三種類型：集中式 (Centralized)、分散式且非結構化 (Unstructured) 以及分散式且結構化 (Structured) 架構。P2P 的運作方式有別於早期主從式架構，檔案等資源是分散在各個節點 (Node 或 Peer) 中，每個節點同時扮演客戶端及伺服器的角色，並且可以不透過特定伺服器而直接進行內容交換及資源分享，因此如何在沒有中央伺服器的幫助下找尋資源已是一項難題。在同儕行動環境中，如果同時再考慮節點於無線環境下的可移動性，勢必會增加 P2P 資源搜尋的困難度，是一個相當重要且富挑戰性的研究課題。

在此研究中我們考慮具有高度負載平衡 (Load Balance) 及延展性 (Scalability) 等優點的結構化 P2P 架構，並以 Chord 為例，做為我們研究的目標。在 Chord 系統中的每個節點都必需維護一個記錄其他節點位置的狀態表，稱為 Finger Table，此狀態表會因為節點的加入或離開而改變。當 Finger Table 所記錄的資訊無法及時反應節點的加入或離開時，在查詢過程中會造成失敗或者增加回應時間。

二、研究目的

本研究的主要目的在於探討如何在無線網路的環境下，利用同儕計算技術提供行動裝置上的即時通訊服務，如網際網路電話或以視訊隨選服務觀看影片等。在同儕行動環境中建立點對點的通訊連線以前，必須先知道另一通訊方或需求資源的位置，亦即 P2P 網路之搜尋、位址探索或資源探索等研究課題，目標是希望能夠降低搜尋所需要的時間並且有效地找到目標。

目前常見的 P2P 搜尋技術大致可分為三種：集中式 P2P 架構由中央伺服器或伺服器叢集 (Cluster) 負責維護集中化索引，優點是實作方式簡單和搜尋效率高，但是存在單一失敗點 (Single Point of Failure) 的缺點，一旦伺服器發生故障，將使整個服務癱瘓；非結構化的 P2P 架構不需要中央伺服器，節點和檔案的分佈是毫無關連地隨意散佈，這種架構通

常採用氾濫(Flooding)網路的方式來進行搜尋，對整個網路造成極大的負擔且不具延展性；結構化的 P2P 則是根據分散式雜湊表 (Distributed Hash Table, 簡稱 DHT) 而組成一個邏輯性的架構，除了可以保證在 $\log(N)$ 的詢問次數內搜尋到所需要的物件 (N 為節點總數)，同時具有延展性，因此 P2P 搜尋技術之發展有逐漸偏重此類型的趨勢，其中又以 Chord 系統較常被提及與探討，我們也選擇其作為研究目標。

計畫的第一年著重於研究移動特性對於結構化 P2P 架構的 Chord 系統所造成之影響，根據不同的節點識別對應方式而推導出數學公式以精確衡量移動設備對 P2P 資源搜尋的影響，並用模擬方式來驗證前述式子的正確性。

三、文獻探討

同儕計算技術已被廣泛地使用在網際網路上的音樂檔案交流及檔案分享的系統 [1][2]，例如 Napster[3]、Gnutella[4]和 Kazaa[5]等，其中 CAN[6]、Chord[7]和 Tapestry[8]等系統則是採用 DHT 的方式來組成 P2P 網路。早期的許多研究只分析因為節點的加入、離開或失效 (此現象稱為 churn) 對 P2P 系統所造成的影響，最早可溯及 Liben-Nowell 等人[9]的研究，在該研究中證明了於 churn 的情況下，一個 DHT 系統最少要花多少成本來維持網路上的連結，才可以確保網路的連結還能被使用。而 Aspnes 等人[10]在研究中更證實當節點或節點間的連結可能消失或者壞掉時，找尋一個檔案或節點所需的訊息個數的上限及下限將會是多少。[11][12][13]文獻中為了更深入的了解 churn 對於 DHT 的影響，就以程式撰寫出模擬環境，並觀察模擬程式的結果來加以分析。Sameh El-Ansary 等人[14]更進一步地除了推導出數學公式來呈現 churn 對 DHT 搜尋效能的影響，並用模擬程式加以佐證。但是，前述研究大部分還是針對固定節點。近年來，因為無線網路的普及，導致愈來愈多研究開始致力於分析行動裝置對於 DHT 系統的影響，[15]即是採用模擬程式的方式來衡量 DHT 系統在行動環境下的效能。

四、研究方法

在選定 Chord 做為我們的研究目標後，首先必須瞭解其運作方式及搜尋過程中所受影響的因素。Chord 是一個有結構性的 P2P 網路，節點和提供分享的物件於 P2P 網路的分佈位置被 DHT 嚴密地控制，每個 Chord 節點和物件都有一個由 DHT 演算法運算而得到的鍵值 (ID 或 Key)。根據節點的 ID，整個 P2P 網路會被切割為數個區域，每個區域皆有負責

節點。根據物件的 Key，相關資訊會儲存於 Key 所屬區域的負責節點上。除了儲存責任區內的物件資訊，每個節點還必須維護一個 Finger Table，這個表格記錄了和節點本身有結構關係的其他節點資訊，亦即這些節點在 P2P 網路中將會互相聯繫以協助搜尋。當搜尋進行時，如果被尋找的物件不在節點的責任區內，即根據 Finger Table 及需求物件之 Key 來決定將詢問的下一個節點，直至詢問到負責儲存需求物件資訊的節點。

Finger Table 的資訊會因為所記錄的節點之加入或離開而改變，所以每個節點必須定期地偵測 P2P 網路的組織狀態以更新 Finger Table。當發生節點加入或離開時，會受影響的節點若無法及時更新其 Finger Table，將會造成經由此節點的查詢失敗或增加查詢時間。

每個 Chord 節點在 P2P 網路中都有一個代表的 ID，通常是由節點的網路位址運算而獲得。節點的離開或加入可以視為一個 ID 的消失或新 ID 的出現，而節點在每次移動後若取得不同的網路位址，也將產生不同的 ID。根據節點可能取得的網路位址型態和 ID 的產生方式，可以整理出如下表的三種類型：

表一 節點 ID 的產生方式

	不同 ID	相同 ID
變動的網路位址	(I) IP	(II) E-Mail
固定的網路位址		(III) Mobile IP

類型 (I) 是一般常見的情況，當節點 ID 是由 IP 位址所產生時，節點在每次移動後取得不同的 IP 位址，造成節點 ID 也會隨之改變，某些節點的 Finger Table 進而需要更新。我們可以把這種情形看成在 Chord 中發生一次節點離開並接著是一次節點加入。

當節點 ID 經由非 IP 位址的固定來源所產生，例如 E-Mail，雖然節點移動時會造成其網路位址的改變，但是並不會改變節點 ID，我們稱此為類型 (II)。每個節點除了要維持原有的 Finger Table，可能還必須另外儲存兩個對照表，一個是關連到本身的其他節點資訊，可稱為 Backward-subscriber Table，另一個則是自己所儲存之節點 ID 所使用的網路位址。

最後一種情況是節點 ID 同樣由 IP 位址所產生，但是使用固定的 IP 位址做為產生 ID 的來源，例如 Mobile IP 的 Home Address，不論如何移動，此 IP 位址都不會改變，因此節點 ID 即不變，也對 Chord 中所有節點的 Finger Table 不會有任何影響。

經過上述的分類，我們將建立數學公式來計算在不同情況下所需的搜尋時間，並用程式模擬方式來驗證其正確性。

五、結果與討論

在類型 (I) 中，節點的移動可以視為在 Chord 中發生一次節點離開並接著是一次節點加入，這樣會增加 Leave Rate，而 Leave Rate 的增加將導致 Finger Table 錯誤率的增加。我們參考[14]的研究，在沒有行動節點組成的 Chord 系統中，一個節點找尋距離其為 $\xi+z$ 的 Key 所需時間期望值 $C_{\xi+z}$ 如下：

$$\begin{aligned}
 C_{\xi+z} = & [1-a(z)] * C_{\xi} \\
 & + (1-f_k) * a(z) * [1 + \sum_{i=0}^{m-1} [bc(i,m)C_{m-i}]] \\
 & + f_k * a(z) * [1 + \sum_{i=1}^{k-1} h_k(i) \sum_{l=0}^{\xi/2^i-1} bc(l, \xi/2^i)(1+(i-1) + C_{\xi-l+m}) + O(h_k(k))]
 \end{aligned}$$

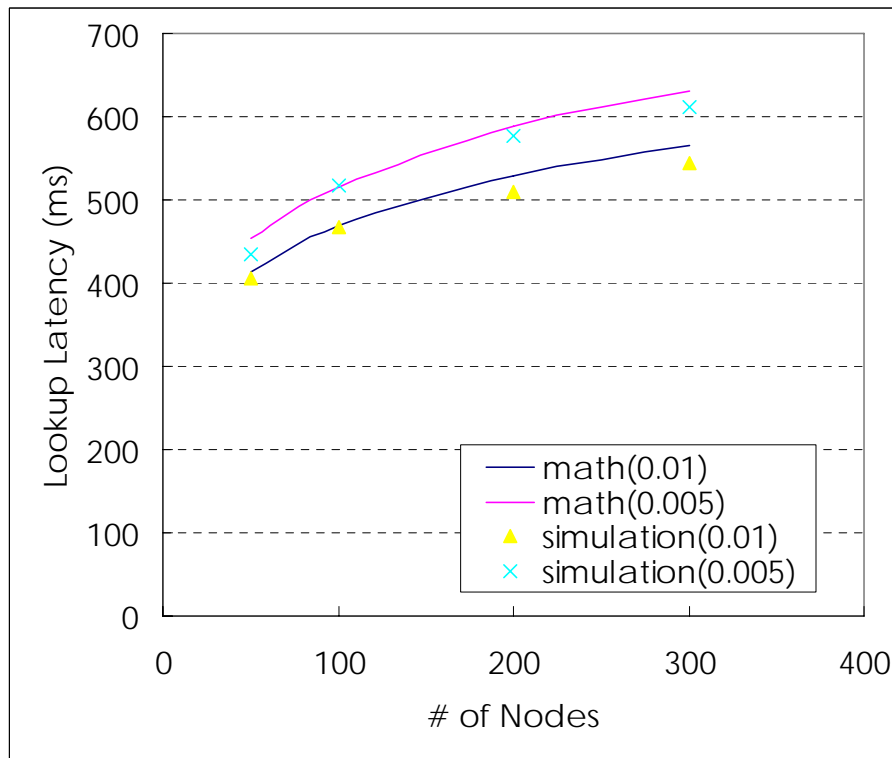
其中的 f_k 代表第 k 項 Finger Table 紀錄是指向錯誤節點的情況在全部節點中所佔的比例，其和節點離開、節點加入與維持系統狀態穩定 (Stabilization) 的程序有關。

考慮在同儕行動環境中，假設 Chord 中存在比例為 P_m 的固定節點，此節點在系統中不會移動位置並且其 Leave Rate 為 λ_f ，另一種類型則是行動節點，其 Move Rate 為 λ_{fm} 。因為節點 ID 是由 IP 位址所產生，IP 位址在節點移動後將會變動，而 IP 位址的變動造成其 ID 也會隨著改變，我們可以把這種情形看成在 Chord 中發生一次節點離開後馬上再發生一次節點加入，因此行動節點在 Chord 中的移動可以看成節點的離開，而此節點的 Leave Rate 即等於其 Move Rate。同時，一個新加入的節點複製其 Successor 節點的 Finger Table 來建立自己的 Finger Table，如果 Successor 節點的第 k 項 Finger Table 紀錄在複製時是儲存著錯誤的資訊，勢必會造成 f_k 的增加。此外，節點會定期針對 Finger Table 的紀錄執行穩定程序，錯誤的節點資訊經過穩定程序後可以被修正，將會造成 f_k 的減少。我們根據前述分析即可重新推導出其中的 f_k 如下：

$$f_k = \frac{\left[(2 * \text{Pr } ep(k) + 2) * \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right) - \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right) * P_{join}(k) + \frac{r(1-\alpha)}{M} \right]}{(2 * \text{Pr } ep(k) + 2) * \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right)}$$

$$\sqrt{\frac{\left((2 * \text{Pr } ep(k) + 2) * \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right) - P_{join}(k) * \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right) - 4 * \left((2 * \text{Pr } ep(k) + 2) * \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right) \right) \right)^2}{(2 * \text{Pr } ep(k) + 2) * \left((1 - P_m) * 1 + P_m \frac{\lambda_{fm}}{\lambda_f} \right)}}$$

根據前述式子的計算與模擬程式執行結果之比較如下圖所示：

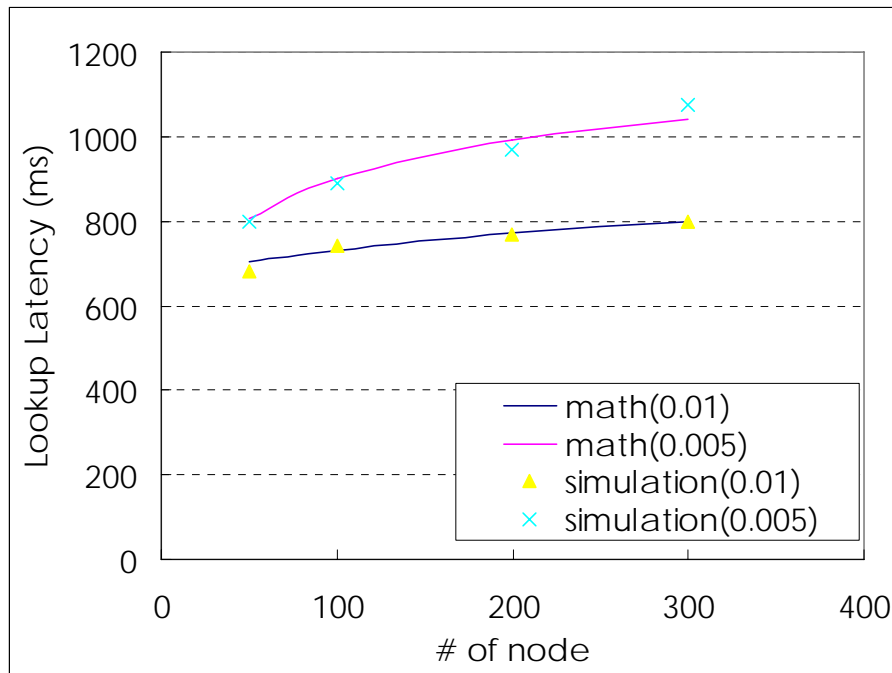


圖一 類型(I)的理論值與模擬結果之比較

考慮一個行動節點 B 的 ID 是由 IP 位址所產生時，並且採用 Mobile IP 架構，當行動節點移動時，其 IP 位址並不會變動，因此 ID 也就不會改變。節點 A 在查詢過程中如果需要經過這類型的行動節點，即使 B 於查詢進行時正在移動，只要能等待 B 移動後重新連上 Access Point 並且向 Home Agent 更新其位置的這段時間，A 就可以再經由 B 來幫忙轉送查詢訊息，因此行動節點並不會影響 Finger Table 的正確性。在此類型 (III) 中，我們推導出一個節點找尋距離其為 $\xi + z$ 的 Key 所需時間期望值如下：

$$\begin{aligned}
C_{\xi+z} &= [1 - a(z)] * C_{\xi} \\
&+ (1 - f_k) * a(z) * P_m \{x1 + x2 + x3 + \sum_{i=0}^{m-1} bc(i, m)C_{m-i}\} \\
&+ (1 - f_k) * a(z) * (1 - P_m) \{x3 + \sum_{i=0}^{m-1} bc(i, m)C_{m-i}\} \\
&+ f_k * a(z) * [\sum_{i=1}^{k-1} h_k(i) * \sum_{l=0}^{\xi/2^i-1} bc(l, \xi/2^i) * ((x1 + x2 + x3) * i + C_{\xi-l+m}) + O(h_k(k))]
\end{aligned}$$

根據上述式子的計算與模擬程式執行結果之比較如下圖所示：



圖二 類型(III)的理論值與模擬結果之比較

從模擬程式執行結果與由數學公式計算求出值的關係，可以驗證式子的正確性。

最後，在類型（II）中，當節點 ID 是經由 E-mail 這種非 IP 位址所產生，行動節點更換網路位址並不會造成 ID 變動，因此在 Chord 中只需通知 Backward-subscriber Table 中所有節點更新其 ID-IP 對照表，如果移動節點在找尋過程中尚未完全通知存於 Backward-subscriber Table 中的所有節點，將會造成經由此移動節點的查詢在一段時間內無法聯繫此移動節點。根據模擬結果，此種類型和類型（III）極為相似。

六、參考文獻

- [1] Dejan S. Milojevic, Vana Kalogeraki, Rajan Lukose, Kiran Nagaraja1, Jim Pruyne, Bruno Richard, Sami Rollins, and Zhichen Xu, "Peer-to-Peer Computing," Technical Report HPL-2002-57, HP Laboratories Palo Alto, 2002.
- [2] Stefan Saroiu, P. Krishna Gummadi, and Steven D. Gribble, "A measurement study of peer-to-peer file sharing systems," in ACM/SPIE Multimedia Computing and Networking, San Jose, CA, USA, January 2002.
- [3] Napster.com, <http://www.napster.com/>
- [4] The Gnutella protocol specification v0.4, 2000, http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf
- [5] Kazaa Media Desktop, <http://www.kazaa.com/>
- [6] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Shenker, "A scalable content addressable network," in ACM SIGCOMM, pp. 161-172, San Diego, CA, August 2001.
- [7] Ion Stoica, Robert Morris, David Liben-Nowell, David R. Karger, M. Frans Kaashoek, Frank Dabek, and Hari Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications," IEEE/ACM Transactions on Networking, Vol. 11, No. 1, pp. 17-32, Feb. 2003.
- [8] Ben Y. Zhao, John D. Kubiatowicz, and Anthony D. Joseph, "Tapestry: An infrastructure for fault-tolerant wide-area location and routing Berkeley," CA: Univ. California, Tech. Rep. CSD-01-1141, Apr. 2001.
- [9] David Liben-Nowell, Hari Balakrishnan, and David Karger, "Analysis of the evolution of peer-to-peer systems," in ACM Conf. on Principles of Distributed Computing (PODC) (Monterey, CA), July 2002.
- [10] James Aspnes, Z. Diamadi, and Gauri Shah, "Fault-tolerant routing in peer-to-peer systems," in Proceedings of the twenty-first annual symposium on Principles of distributed computing, pp. 223–232, 2002.
- [11] Jinyang Li, Jeremy Stribling, ThomerM. Gil, RobertMorris, and Frans Kaashoek, "Comparing the performance of distributed hash tables under churn," in the 3rd International Workshop on Peer-to-Peer Systems (IPTPS'02) (San Diego, CA), Feb 2004.
- [12] Sean Rhea, Dennis Geels, Timothy Roscoe, and John Kubiatowicz, "Handling churn in a DHT," in Proceedings of the 2004 USENIX Annual Technical Conference(USENIX '04) (Boston, Massachusetts, USA), June 2004.
- [13] Miguel Castro,Manuel Costa, and Antony Rowstron, "Performance and dependability of structured peer-to-peer overlays," in Proceedings of the 2004 International Conference on Dependable Systems and Networks (DSN'04), IEEE Computer Society, 2004.
- [14] Sameh El-Ansary, Supriya Krishnamurthy, Erik Aurell, and Seif Haridi, "An analytical study of consistency and performance of DHTs under churn," Tech. Report TR-2004, Swedish Institute of Computer Science, October 2004.

[15] Hung-Chang Hsiao and Chung-Ta King, "Mobility churn in DHTs," in 25th IEEE International Conference on Distributed Computing Systems Workshops, pp. 799- 805, 6-10 June 2005.

七、計畫成果自評

在這一年的時間裡，我們除了深入瞭解 P2P 網路的架構與運作，並分析行動環境對 P2P 搜尋效能的影響，這對後續的研究提供了不少的助益。目前正整理相關研究成果，預計將本年度有關行動環境對 P2P 搜尋效能的影響投稿國際研討會與學術期刊。