

# A Novel P2P Mobile Proxy Caching Scheme

James Z. Wang                      Zhidian Du

*Department of Computer Science  
Clemson University, Box 340974  
Clemson, SC 29634-0974, USA  
+1-864-656-7678*

*{jzwang, zdu}@cs.clemson.edu*

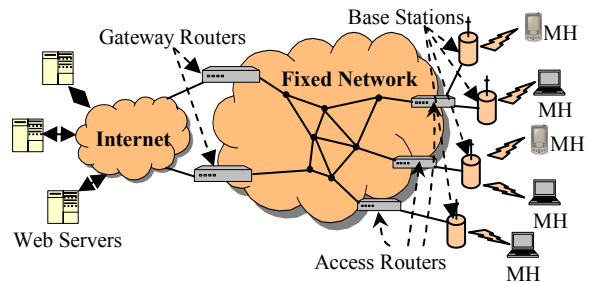
## Abstract

*In this paper, a novel P2P mobile caching scheme for caching web documents in mobile base stations is proposed. In this scheme, a mobile cache model is used to facilitate the data cache and data replication. A self-configured and self-managed proxy overlay network is employed to fulfill the P2P data search and proxy server cooperation. Using the proposed caching scheme, the individual proxy in the base station makes cache decisions based solely on its local knowledge of the global cache state so that the entire wireless proxy cache system can be efficiently managed without centralized control. Furthermore, the proposed P2P mobile caching scheme can take advantage of the mobile host movement to improve the proxy caching performance. It is proven by formal analysis and simulation studies that the proposed P2P mobile caching scheme outperforms other caching schemes in terms of different performance metrics.*

## 1. Introduction

Recent advances in mobile networks have made the mobile internet applications more popular and sophisticated. Many interesting web applications, such as multimedia streaming, have flourished with the growth of Internet and WWW. Due to strong demand of bringing these internet applications into wireless environments, much effort has been made to consolidate the WWW with wireless networks [1, 2]. Such integration is also referred to as W4 – World Wide Web for Wireless. Figure 1 depicts a typical W4 architecture, in which mobile hosts access the mobile network through base stations, which are interconnected by access routers to form wireless LANs and, in turn, are connected to the internet through gateway routers.

Among the numerous studies for enhancing the mobile internet performance, caching popular web data at locations close to the mobile clients is an effective solution to improving the quality of mobile internet applications. Although caching can be implemented in various points in a wireless network, proxy caching in mobile base stations [3, 4] is the most efficient caching solution. Caching popular web documents in mobile base stations can ease network traffic between base stations and web servers, and reduce user web request latencies. Providing caching mechanisms in mobile base stations can also reduce the connection time between the mobile host and the base station due to the reduced waiting periods experienced when fetching remote web documents, thus saving the limited wireless bandwidth. Especially, constant movements of the mobile hosts among the base stations require the caching proxies to cooperate for improving the performance of mobile internet applications.



**Figure 1: W4 network architecture**

It is challenging to design a cooperative proxy caching scheme suitable for wireless environments. Due to the non-hierarchical nature of the distributed mobile base stations, adopting a distributed cooperative proxy caching scheme proposed in a wired network environment seems to be a viable solution.

However, the distributed cooperative proxy cache systems [5, 6, 7, 8] developed in wired network environments employ sophisticated caching and searching schemes, such as centralized or distributed directory lookup, distributed hashing, multicasting and routing, to distribute and search the cached web documents. Using sophisticated caching scheme increases the complexity of base station functions. On the other hand, the objective of caching web documents in mobile base stations is to reduce the user request latencies so that the wireless network bandwidth and mobile battery power can be saved. Yet most of these existing caching schemes often concentrate on increasing the cache hit ratio while ignoring the cost of retrieving the web document from the server where it is cached [9, 10]. A cooperative proxy caching scheme for mobile base stations should emphasize on moving the cached web documents closer to interested clients through data replication.

Besides the aforementioned problems, there are other challenges for web caching in distributed mobile base stations. First, the mobile hosts in wireless environments constantly move among the base stations. Moving mobile hosts cause tougher challenges in caching strategy and increase complexity of the cooperative proxy cache system. Second, reducing user request latency is more critical in wireless environments since wireless links have limited bandwidth and mobile hosts have limited battery power. Finally, in wired networks, a caching proxy usually serves clients within an organization, in which clients share common interests. However, in wireless environments, mobile clients connecting to the same base station might come from different regions and have different backgrounds, hence having different interests. The diversity of client interests directly affects the cache strategies and replacement policies.

To address these challenges in proxy caching for mobile base stations, we propose a novel P2P mobile caching scheme for the distributed base stations. The rest of the paper is organized as follows. In section 2, we discuss the existing solutions to web caching in wireless environments. In section 3, we propose the P2P mobile caching scheme, in which a mobile cache model is employed to facilitate the distributed web document cache, and a proxy overlay network is used to assist P2P data search, data cache and data replication. In section 4, we analyze, using probability theory, the performance advantages of the proposed mobile caching scheme, compared to other caching schemes. In section 5, we use simulation to quantitatively evaluate the performance of different caching schemes. Finally we give our conclusion and discuss future studies in section 6.

## 2. Related works

Constant mobile host handoffs and geographically distributed base stations exacerbate such problems as the limited battery power and wireless bandwidth of mobile hosts in wireless environments. To address the problems, Hadjiefthymiades and Merakos [4] proposed a proxy cache relocation scheme to accelerate the web browsing. This scheme tries to predict the next base station a mobile host will move to and copies the cached web documents in the dedicated cache space for this mobile host to the predicted base station before the handoff is actually taken place. Due to uncertainty of the predictions, besides copying 100% of cached data to their best predicted neighbor proxy, they also copy 70% of the cached data to the second best predicted neighbor proxies and 30% to the rest neighbor proxies.

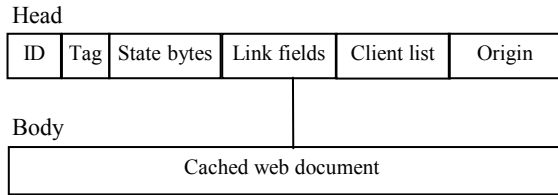
There are three problems in this proxy cache relocation scheme. First, their caching scheme assigns each mobile host a dedicated cache space in the proxy server. Recent studies found that user access to web documents follows a Zipf-like distribution with high Zipf factors [11, 12]. It means that user access to web documents is highly skewed to a few web documents. Although assigning each mobile host a dedicated cache space simplifies the proxy cache management, it wastes the limited proxy cache space when multiple clients replicate the same web document in the same proxy cache. Second, inaccuracy of the predictions causes unnecessary data copies among the base station proxies. Third, copying web documents based on the prediction of the mobile host movement may introduce data oscillation in the wireless network. These problems hinder the performance of the mobile proxy cache system.

## 3. A P2P mobile caching scheme

To simplify the cooperation among the caching proxies in the distributed base stations while avoiding the problems in dedicated cache approach, we propose a P2P mobile caching scheme in this section.

### 3.1 P2P mobile cache model

Designing a cache model has to be based on the characteristics of the cached data and the specific caching environment. The cached data in the distributed mobile base stations are web documents that may be migrated to one proxy from other caching proxies instead of directly being fetched from the original web servers. To cope with the characteristics of cached web data and wireless network environments, we design a mobile cache line as depicted in Figure 2 to facilitate a shared cache model, avoiding the problems in the dedicated cache model.

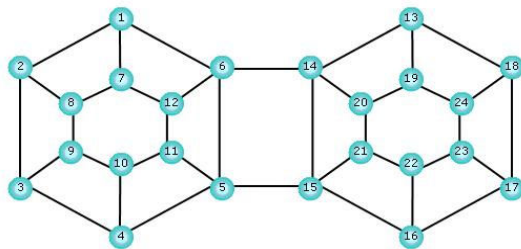


**Figure 2: A mobile cache line**

Besides having the cached web document as its body portion, the proposed mobile cache line contains a head portion which includes *ID*, *Tag*, *State bytes*, *Link fields*, *Client list* and *Origin*. The *ID* field contains a *UUID* which the URI of the cached web document will be hashed to. The *Tag* is the name of the cached web document. *State bytes* are used to store information for cache coherency protocol as well as some other statistic data, such as number of cached replicas for a web document that the current proxy is locally aware of. Because the cached web document might be replicated from other caching proxies in the wireless network, link fields are needed to provide the links to the neighbor proxies from whom web replicas can be found. A link field is organized as a pair of integer  $\langle NID, Dist \rangle$ . *NID* is an integer used to lookup the neighbor table for a neighbor's IP address. *Dist* is round trip distance to reach the cached web document through the neighbor specified by *NID*. *Client List* contains IDs of the clients that are currently accessing the cached web document. *Origin* indicates whether this current web replica is fetched from the original web server or is migrated from another proxy in the P2P mobile proxy cache system.

### 3.2 Proxy overlay network

Since the proposed mobile cache lines use links to group the cached web replicas, it is necessary to avoid one caching proxy having to connect to too many other proxies. Thus, we use a proxy overlay network (PON) to facilitate the data linkage and data exchange among the caching proxies. Figure 3 demonstrates a mobile proxy overlay network with 24 nodes.



**Figure 3: A mobile proxy overlay**

A mobile PON forms the foundation for data search, data cache and data replication in the P2P

mobile proxy cache system. A proxy server will only exchange data and information with its neighbors in the PON. There is no centralized controller for building the PON in our P2P mobile proxy cache system. Instead, the PON is automatically configured by individual caching proxies in base stations using simple protocols. The individual caching proxies can also self-manage the PON based on dynamic data flow in caching proxies. The PON self-configuration and self-management include the following two phases:

**Self-configuration phase:** A proxy initially broadcasts a join-request message to all nodes in the network, asking for joining the proxy cache system. After receiving responses from the existing caching proxies, the joining proxy selects a set of existing caching proxies, which are close to this proxy and have extra connection capacities, as its neighbors in the PON. A neighbor table will be built in this proxy to facilitate the neighbor selection. An entry in the neighbor table includes an ID which is usually a small integer and the IP address of the neighbor.

**Self-management phase:** After joining the PON, proxy servers enter the self-management phase. In this phase, a proxy continuously monitors its data and information exchanges with its neighbors. It may remove a link in the PON to a neighbor if there is a substantial reduction in data exchange with that neighbor for a certain period of time. It may add a link to a node in the PON if a significant amount of the data it received is originated from that node. This dynamic self-management process makes sure that the PON is always effective for data search, data cache and data replication. The PON self-management is very important in wireless environments because changes of the wireless environment, such as new construction or road blocking, may alter the data flow among the mobile base stations.

### 3.3 P2P data search and data cache

When a new web request reaches a certain proxy server from its client, the proxy server takes actions based on three different situations as follows:

- **The entire mobile cache line is cached:** If the entire mobile cache line is cached, the web request is satisfied.
- **Only the head is cached:** If only the head of the mobile cache line is cached in this proxy server, the current proxy will check the head of the mobile cache line for the shortest distance to the cached web replicas. If the distance to the nearest web replica is larger than its expected response time by directly requesting from the original web server, a query is sent to the original web server.

Otherwise, a query message will be created, and propagated, along the links in the heads of mobile cache lines, to the proxy server that holds the nearest web replica.

- **Nothing is cached:** if nothing is cached at this proxy server, the proxy server first sends a query message to all its neighbor proxies and waits for the responses from them. The query message will be disseminated to other peer proxies through peer relay until the message lifetime is expired or a matching mobile cache line (maybe head only) is reached. The proxy that has the matching mobile cache line responds the query. The responding message is routed back to the requesting proxy along the query path. The proxies on the routing path create or update the appropriate mobile cache line heads based on the responding message to provide links to the cached web replica. Once the responding messages reach the requesting proxy and related mobile cache line head is created on that proxy, the rest of the search process would be same as in case 2.

Our P2P mobile proxy caching scheme can take advantage of the cooperation among base station proxies to increase the cache hit ratio and reduce the web request latency. When the mobile host moves from one base station to its neighbor base station, it is not necessary to move its cached web documents to the new base station prior to or at the handoff. Our P2P data search scheme can quickly find the cached web documents from the proxy to whom the mobile host previously connected if the cached web documents have not been removed by cache replacement, because these two base station proxies are likely neighbors in the PON. The request latencies of querying web documents from neighbor proxies are much smaller than that of fetching them from the original web servers. Furthermore, we can exploit the mobile host movement by moving the heads of the previously accessed mobile cache lines to the new base station during the handoff. Using this simple method, we can effectively reduce the cost of searching the web documents because the links in mobile cache line heads can directly lead to the previously cached web replica in the neighbor caching proxy.

#### 4. Performance analysis

To assess the performance of the proposed P2P mobile caching scheme, we compare its cache hit ratio with some other caching schemes. In this study, the following 5 different proxy caching schemes are evaluated:

**Cache Relocation (CR):** The proxy cache relocation scheme proposed in [4] is studied with the

assumption of 100% accuracy in prediction of the mobile host movement, although it is unlikely for any prediction to be 100% accurate.

**Non-cooperative Caching (NC):** In this scheme, each individual proxy acts independently and there is no cooperation among the caching proxies. This scheme can be used to evaluate how proxy cooperation impacts the performance of the mobile proxy cache system.

**Multicast-based Cooperative Caching (MCC):** MCC was originally designed for distributed wired networks, it is one of the existing proxy caching schemes that can be easily adapted into the distributed mobile base stations.

**P2P Mobile Caching (PMC):** The basic P2P data caching, searching and replicating methods based on the proposed mobile cache model and the self-configured PON are implemented.

**PMC with Head Move (PMC-HM):** To take advantage of the mobile host movement, besides implementing the basic PCC scheme, this scheme moves the heads of the mobile cache lines with the moving mobile hosts to the new base stations when handoffs occur.

To simplify the analysis, we assume all schemes use the same PON, and each proxy has the same proxy cache space, and all schemes are compared under the same cache space. We also assume that a caching proxy waits for its neighbors to respond its queries for a time  $t_{\max}$  before it sends query directly to the original web server. A TTL (Time-To-Live) value is used to control the number of hops that a query message will travel in the network before it is expired. To ensure a fair comparison, we also set the same TTL value for query messages in our P2P proxy caching schemes, so that the proxy servers will not forward the query message to their neighbors if the query message's life time is expired.

Since mobile cache line heads are used to link the cached web documents together through data search, data cache and data replication, we may assume that a requested web document can be always reached by following the links in the mobile cache line heads. If we assume the probability of finding a cached web document locally is  $p_{loc}$ , then local cache hit ratio is  $p_{loc}$ . Thus the probability of finding a cached web document in a single non-cooperative cache is  $p_{nc} = p_{loc}$ .

In CR scheme, each mobile host is allocated an independent cache space in the caching proxy. Assume the probability of finding a cached web document using CR scheme is  $p_{cr}$ . If there are  $N$  mobile hosts, then the total cache space is partitioned

into  $N$  partitions ( $N > 0$ ). Since all  $N$  partitions in CR scheme are independent, looking for a document in one partition obeys the Bernoulli trail. The distribution of not finding a document in these  $N$  partitions obeys the Binomial distribution where the probability of finding the document in a trail is  $p_{cr}$ , i.e.,

$$P(X = 0 | N, p_{cr}) = \binom{N}{0} p_{cr}^0 (1 - p_{cr})^N = (1 - p_{cr})^N.$$

So the probability of finding the cached web document in at least one of the  $N$  partitions,  $p_{cr}^*$ , is as follows:

$$p_{cr}^* = 1 - (1 - p_{cr})^N \quad (1)$$

Thus we have,

$$p_{cr}^* - p_{cr} = (1 - p_{cr}) - (1 - p_{cr})^N \geq 0$$

Since  $N > 0$ , we have  $p_{cr}^* \leq p_{loc} = p_{nc}$ . So,

$$p_{cr} \leq p_{cr}^* \leq p_{loc} = p_{nc}$$

It means that the NC proxy caching scheme results in higher cache hit ratio than CR scheme does.

For the MCC scheme, if there is a miss in local cache, the probability of finding a cached web document from its neighbors during  $t_{\max}$  is:

$$p_{mcc} = p_{loc} + (1 - p_{loc}) \cdot p_{bc\_mcc} \quad (2)$$

where  $p_{bc\_mcc}$  is the probability of finding the cached web document by broadcasting query messages to neighbors. Obviously we have  $p_{mcc} \geq p_{loc} = p_{nc}$ .

In basic PMC scheme, if there is a cache miss locally, the caching proxy sends query messages to their neighbors. We assume that the probability of finding a mobile cache line head within time  $t_{\max}$  is  $p_{hd}$ , and the probability of finding a cached web document through broadcasting within time  $t_{\max}$  is  $p_{bc\_pmc}$ . Then we can calculate the probability of finding a cached web document using the basic PMC scheme is,

$$p_{pmc} = p_{loc} + (1 - p_{loc}) \cdot (p_{hd} + (1 - p_{hd})p_{bc\_pmc}) \quad (3)$$

We assume  $p_{bc\_pmc} = p_{bc\_mcc}$  because they are essentially using the same scheme to probe the neighbor proxies. Comparing  $p_{pmc}$  and  $p_{mcc}$ , we have,

$$\begin{aligned} p_{pmc} - p_{mcc} &= (1 - p_{loc})(p_{hd} + (1 - p_{hd})p_{bc\_pmc} - p_{bc\_mcc}) \\ &= (1 - p_{loc})(p_{hd} + (1 - p_{hd})p_{bc\_mcc} - p_{bc\_mcc}) \\ &= (1 - p_{loc})(1 - p_{bc\_mcc})p_{hd} \geq 0 \end{aligned}$$

Thus, we have  $p_{pmc} \geq p_{mcc}$ .

Now we study the impact of moving the mobile cache line heads along with mobile hosts on the proxy cache system performance. Assume the probability of finding a mobile cache line head in a proxy cache is now  $p_{hdmv}$ . We have  $p_{hdmv} \geq p_{hd}$  due to the added mobile cache line heads moved in with mobile hosts

during handoffs. Thus, when mobile cache line heads can be migrated with mobile hosts, the probability of finding a cached web document will be,

$$p_{pmchm} = p_{loc} + (1 - p_{loc}) \cdot (p_{hdmv} + (1 - p_{hdmv})p_{bc\_pmchm}) \quad (4)$$

Because a mobile cache line head is very small compared to the cached web documents, it does not take much proxy cache space. We can reasonably assume that  $p_{loc}$  in PMC-HM scheme will still be the same as in the basic PMC scheme, and  $p_{bc\_pmchm} = p_{bc\_pmc}$ . Thus we have,

$$p_{pmchm} - p_{pmc} = (1 - p_{loc})(p_{hdmv} - p_{hd})(1 - p_{bc\_pmchm}) \geq 0.$$

That is, we have  $p_{pmchm} \geq p_{pmc}$ .

Our analysis reveals that the proxy cache system using CR scheme has the least cache hit ratio while using PMC-HM scheme results in the best performance. The links among the cached web documents in PMC proxy caching scheme help it outperform the MCC scheme. As we expected, non-cooperative proxy caching scheme is worse than any of the cooperative proxy caching schemes in terms of cache hit ratio.

## 5. Simulation study

Although our analysis has proven the superiority of the proposed P2P mobile caching scheme compared with the other schemes, it is still necessary to quantitatively evaluate how these caching schemes perform under various system conditions. We use simulation to assess the performance of different proxy caching schemes in terms of various system metrics.

### 5.1 Simulation model

We assume that there are 100 base stations forming a 10x10 mesh-like grid in the wireless network. We also assume that the automatically configured PON has the same structure in which all base station proxies have 4 neighbors, excepting those base station proxies on the edges and the corners, which have 3 and 2 neighbors respectively. We further assume that the PON structure will not change during the entire simulation. To evaluate the impact of the mobile host movement, we use a variant random waypoint movement pattern [13] to simulate the movement of the mobile hosts. The events of the mobile host movement are independent of the events of web requests. After a mobile host stays in the coverage area of one base station for some time, it may move to one of its neighbor base stations, or jump to a non-neighbor base station (this is the case when the user turns off the mobile device and turns on again after driving for a while), or simply choose to stay with the same base station. The mobile host has equal probability to make any of these moves. The time for

a mobile host to stay with one base station follows the Poisson distribution with a mean duration of 180 seconds. This time is based on the assumption that the distance between two adjacent base stations is 2 miles, the average speed of vehicles is 40 miles/hour, and handoff happens at center point of two adjacent base stations.

We assume that the user request latency is 100 ms ignoring the other overheads if the requested web document is cached in the proxy server that the mobile host is currently connected to. The roundtrip network distance between a pair of neighbor proxies is assumed to be 100 ms. So if a requested web document is found in a neighbor proxy server, the user request latency is assumed to be 200 ms, ignoring other overheads. If a requested web document is two hops away from the current proxy server, the user request latency will be 300 ms. The average user request latency for a cache miss is 2000 ms, ignoring all other overheads. These assumptions are based on the data collected by running the benchmark Polygraph [14] on a web caching appliance developed by Swell Technology [15].

We also assume that there are 10,000 distinct web documents on the internet and the average web document size is 60K. 35% of the web documents have size less than 10K. 60% of them have size in the range from 10KB to 100KB. The sizes for the rest web documents are in the range of 100KB to 1MB. The assumption on web document sizes is based on the latest web access statistics of several different web servers [16, 17]. It is assumed in [18] that internet user clicks a web link once every 12.5 seconds. Thus in our simulation, we assume the inter-arrival time of web requests generated by a mobile host follows a Poisson distribution and the average inter-arrival time equals to 12.5 seconds.

The access frequencies to the web documents follow a Zipf-like distribution [11, 12]. The access frequency for web document  $i$  is determined as follows:

$$f_i = \frac{1}{i^z \cdot \sum_{j=1}^m 1/j^z} \quad (5)$$

where  $m$  is the number of the web documents in the system, and  $0 \leq z \leq 1$  is the Zipf factor [11]. A larger  $z$  value corresponds to a more skewed data access pattern, i.e., some objects are accessed considerably more frequently than other objects. When  $z = 0$ , the distribution is uniform, i.e., all the objects have the same access frequency. Total of 400,000 web requests are issued to the cache system simulator.

To avoid the inaccurate statistics due to the startup of simulation, we start to collect data after the first

20,000 web requests are processed. A simple LRU cache policy is used for every tested proxy caching schemes in our simulation. We list most of our default simulation parameters in Table 1. We will use these parameters throughout the different simulations unless being explicitly stated otherwise.

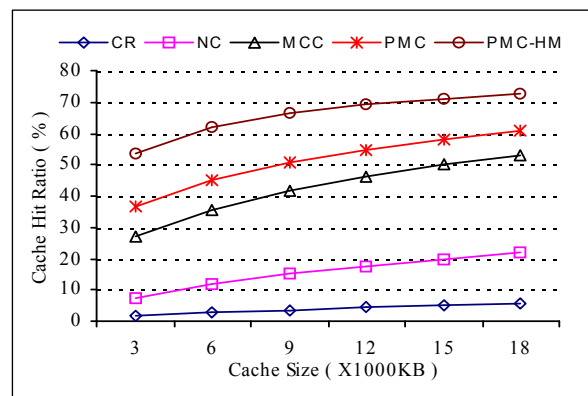
**Table 1: Simulation Parameters.**

Proxy cache space	6000 KB
Number of distinct web pages	10000
Total number of web requests	400,000
Zipf Factor	0.75
Mean request interval time	12.5 seconds
Mean handoff interval time	180 seconds
Number of base stations	100
Number of mobile hosts	500
Average size of the web page	60 KB
Average size of the query/reply message	0.1 KB
Average size of the cache line head	0.1 KB

## 5.2 Simulation results

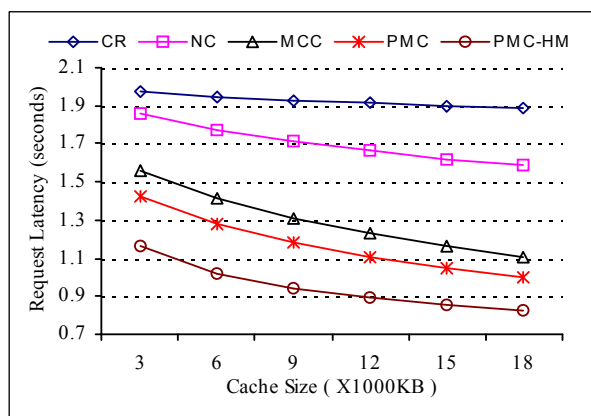
We vary the cache size in each proxy server from 3000 KB to 18000 KB, and observe the cache hit ratios and average user request latencies under various cache sizes. We also monitor the data exchange between a pair of proxy servers and record the average data exchange per query. The simulation results are depicted in Figure 4, Figure 5 and Figure 6.

As we expected, increasing the cache size increases the chance of a web page being cached in the proxy servers, hence increasing the cache hit ratio and decreasing the average user request latency. Among the tested proxy caching schemes, the cache relocation scheme performs the worst due to its dedicated cache model. Compared to non-cooperative proxy caching approach, proxy cooperation improves the system performance significantly.

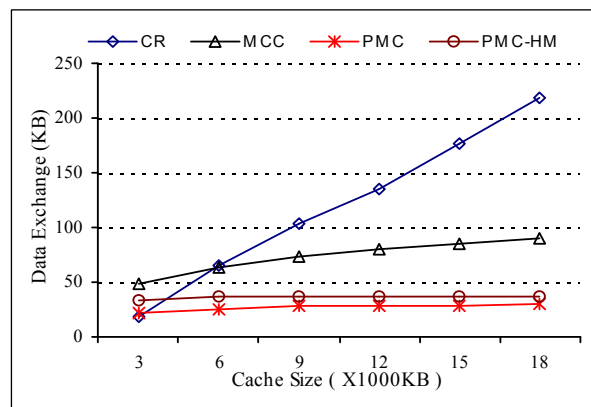


**Figure 4: Cache hit ratio under various cache sizes**

For instance, when cache size is 9000KB, MCC scheme and PMC scheme outperform NC scheme by 175% and 235% respectively in terms of cache hit ratio. In the meantime, the average user request latency using NC scheme is 31% and 45% higher than that using MCC and PMC schemes respectively. Among the cooperative proxy caching schemes, our proposed P2P mobile caching schemes demonstrate better system performance in terms of cache hit ratio and user request latency. For instance, when cache size is 9000KB, the cache hit ratio using PMC scheme is 22% higher than that using MCC scheme. On the other hand, the average user request latency using MCC scheme is 11% higher than that using PMC scheme.



**Figure 5: Average request latency under various cache sizes.**



**Figure 6: Data exchange under various cache sizes**

Most importantly, our PMC scheme, compared to MCC scheme, requires much less network bandwidth within the wireless network while achieving better performance in terms of cache hit ratio and average user request latency. On average, as shown in Figure 6, the data exchange between a pair of proxy servers for each web request using MCC scheme is much

higher than that using our proposed mobile proxy caching schemes. For instance, when cache size is 9000KB, the average data exchange using MCC scheme is 167% higher than that using PMC scheme. The superiority of our proposed PMC scheme is due to its excellent P2P data search, data cache and data replication strategies, which not only create links among nearby cached web replicas but also move the cached web documents closer to interested mobile clients. Thus, the number of messages required for searching the cached web documents is massively reduced. Although moving and replicating mobile cache line heads add extra network traffic, the reduced query message exchange due to the improved data search greatly outweighs the extra data traffic caused by replicating the cache line heads.

Another advantage of our proposed P2P mobile proxy caching scheme is its natural adaptivity to wireless network environments. This P2P mobile caching scheme can exploit the mobile host movement to improve the proxy caching performance. As shown in Figure 4 and Figure 5, our PMC-HM scheme outperforms the other schemes by very large margins in terms of cache hit ratio and user request latency. For instance, when cache size is 9000KB, the cache hit ratio using PMC-HM scheme is 59% better than that using MCC scheme, and the average user request latency using MCC scheme is 39% worse than that using PMC-HM scheme.

Although moving mobile cache line heads with the mobile hosts in PMC-HM scheme causes slightly more data exchange between proxy servers compared to the basic PMC scheme, the cost is justified considering the PMC scheme is only 22% better than MCC scheme in terms of cache hit ratio while PMC-HM is 59% better. Most impressively, PMC-HM scheme significantly outperforms the MCC scheme in terms of data exchange between the proxy servers, even if it requires extra data exchange compared to the basic PMC scheme. As shown in Figure 6, when cache size is 9000KB, MCC scheme generates 2 times of data traffic in wireless network compared to PMC-HM scheme. Cache relocation scheme shows the worst performance in terms of data exchange between proxy servers. We must note here that the non-cooperative caching approach (NC) does not need extra data exchange among proxy servers. Thus, it is not shown in Figure 6.

## 6. Concluding remarks and future studies

In this paper, we propose a novel P2P mobile proxy caching scheme, which uses a mobile cache model to facilitate the proxy caching cooperation and uses a self-configured and self-managed proxy overlay

network to fulfill the P2P data search, data cache and data replication in the P2P proxy cache system. Using our P2P mobile proxy caching scheme, the individual proxy server makes cache decisions based on its local knowledge of the global cache state so that the entire wireless proxy cache system can be efficiently managed without centralized control. The aggregate effect of data caching, searching and replicating actions by individual proxies automatically distributes cached web documents closer to the interested mobile clients. Furthermore we flow and replicate the heads of mobile cache lines associated with a moving mobile host to the new base station during the mobile host handoff. These replicated cache line heads provide direct links to the cached web documents previously accessed by the moving mobile host in neighbor proxies, thus improving the mobile web caching performance. Our performance study has shown that our proposed P2P mobile proxy caching schemes outperform the other proxy caching schemes in terms of different performance metrics.

In our simulation studies, we used the LRU algorithm for cache replacement. Other factors, such as document size, data replication rate, and mobile host handoff also impact the performance of the proxy cache system. It is necessary to design a cache replacement algorithm specifically for our P2P mobile proxy cache system, taking consideration of all these factors. To further improve the system performance, we are also investigating how to accurately predict the mobile host movement and integrate the prediction with our proposed P2P mobile caching scheme.

## 7. Reference

- [1] C. Baquero, V. Fonte, F. Moura, and R. Oliveira. MobiScape: WWW Browsing under Disconnected and Semi-Connected Operation, in *proceedings of First Portuguese WWW National Conference*, Braga, Portugal, July 1995.
- [2] Barron C. Housel and David B. Lindquist. WebExpress: A system for optimizing Web browsing in a wireless environment. In *Proceedings of the Second Annual International Conference on Mobile Computing and Networking*, pages 108-116, November 1996.
- [3] Z. Jiang, L. F. Chang, B. J. Kim and K. K. Leung, Incorporating Proxy Services into Wide Area Cellular IP Networks, In *Proceedings of the IEEE WCNC*, 2000.
- [4] Stathis Hadjiefthymiades and Lazaros Merakos, Using Proxy Cache Relocation to Accelerate Web Browsing in Wireless/Mobile Communications, pages 26-35, *WWW10*, May 1-5, 2001, Hong Kong.
- [5] R. Tewari, M. Dahlin, H. Vin and J. Kay. Beyond Hierarchies: Design Consideration for Distributed Caching on the Internet. *Technical Report: TR98-04, Department of Computer Science, University of Texas at Austin*, February 1998.
- [6] Joe Touch. The LSAM Proxy Cache – a Multicast Distributed Virtual Cache. *the 3<sup>rd</sup> International WWW Caching Workshop*, Manchester, England, June 1998.
- [7] L. Fan, P. Cao, J. Almeida and A. Broder. Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol. *IEEE/ACM Transactions on Networking*, 8(3):281-293, 2000.
- [8] Alex Rousskov and Duane Wessels, Cache Digests. *Computer Networks and ISDN Systems*, 30(22-23):2155-2168, June 1998.
- [9] M. Rabinovich, J. Chase and S. Gadde. Not All Hits Are Created Equal: Cooperative Proxy Cache Over a Wide-Area Network. *Computer Networks and ISDN Systems*, 30(22-23):2253-2259, November 1998.
- [10] A. Wolman, G. M. Voelker, N. Sharma, N. Cardwell, A. Karlin, and H. M. Levy. On the scale and performance of cooperative web proxy caching. *Proc. of 17th ACM Symposium on Operating Systems Principles (SOSP)*, 1999.
- [11] L. Breslau, P. Cao, L. Fan, G. Phillips and S. Shenker. Web Caching and Zipf-like Distributions: Evidence and Implications. in *proceedings of IEEE INFOCOM'99*, pp. 126-134, March 1999.
- [12] James Z. Wang and Ratan K. Guha. "Proxy Ecology - Cooperative Proxies with Artificial Life." In *Proceedings of IEEE/WIC IAT-2003*, Halifax, Canada. 2003.
- [13] W. Su, S.-J. Lee, and M. Gerla, Mobility Prediction in Wireless Networks, *Proceedings of the IEEE Military Communications Conference (MILCOM)*, Los Angeles, CA, October 2000.
- [14] The Measurement Factory, <http://www.measurement-factory.com/>
- [15] Tsunami, <http://www.swelltech.com/products/caching.html>
- [16] Web server statistics, <http://www.lescroupiersrunningclub.org.uk/ace/logfile.php>
- [17] Web Server Statistics for VSOHP, <http://satobs.org/ps.html>
- [18] Microsoft Internet Security & Acceleration server: ISA Server Scales Out to Meet Enterprise-Class Caching Demands, <http://www.microsoft.com/isaserver/evaluation/competitive/scaleout.asp>.