JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 23, 421-439 (2007)

Mobility Management Using P2P Techniques in Wireless Networks[*]

SHOU-CHIH LO
Department of Computer Science and Information Engineering
National Dong Hwa University
Hualien, 974 Taiwan
E-mail: sclo@mail.ndhu.edu.tw

Mobility management is an important task in wireless networks. The Mobile IP protocol provides a basic solution to mobility management in future all-IP wireless network environments. However, Mobile IP suffers from several problems such as triangular routing, long distant and frequent registration update. In this paper, we propose to use the *Peer-to-Peer* (P2P) network technology to improve Mobile IP. We organize home agents into P2P networks, and take addressing binding data as shard data. A user's address binding will be hashed into the P2P network and can be queried using P2P lookup mechanisms. We can use a key value (*e.g.*, email or IP address, telephone number) in the P2P lookup, to uniquely locate a user's address binding. The potential characteristics of load balancing and fault tolerance in a P2P network make our approach more scalable and robust than the standard Mobile IP. To reduce the registration update cost, we chose a dynamic home agent for each mobile user that is close to the user's location. We study the performance of our proposed approach both qualitatively and quantitatively.

Keywords: wireless networks, peer-to-peer networks, mobility management, mobile IP, dynamic home agent assignment

1. INTRODUCTION

There has been an evolutionary trend in wireless networks toward the next-generation ones. The next-generation wireless networks would be heterogeneous, which include different access networks [19]. For instance, the network services are provided by the cellular radio systems for outdoor environments and by the wireless local area network systems for indoor ones. One promising feature of next-generation wireless networks is to provide an all-IP (Internet Protocol) architecture and the connectivity to anywhere at anytime.

Mobility management in wireless networks is an important task in order to keep connectivity with roaming users at anytime. Mobile IP [10], which is a standard proposed by the Internet Engineering Task Force (IETF), becomes the potential solution for mobility management in the future all-IP wireless networks [9]. Mobile IP uses the home agent (HA) and foreign agent (FA) to maintain the mobility of a mobile node (MN). The HA maintains the address binding of an MN, which is a mapping between the permanent

---

Received February 1, 2005; revised April 25, 2005; accepted May 23, 2005.
Communicated by Ten-Hwang Lai.

[*] The preliminary version of this paper was presented in the 6th IFIP IEEE International Conference on Mobile and Wireless Communication Networks, Paris, France, October 25-27, 2004.

home address to the care-of address (CoA) temporally borrowed from an FA. However, Mobile IP suffers from several problems such as the triangular routing, frequent and long distant registration updates, and single-point of failures [2, 4, 5, 13, 26-28].

In this paper, we propose some mechanisms to solve the problems experienced in Mobile IP, and most of importance, we introduce the emerging technique of *Peer-to-Peer* (P2P) networks into Mobile IP. P2P networks are overlay networks whose topologies are fully independent of physical networks. P2P networks are mostly designed for data sharing applications such as the existing systems: Napster [20], Gnutella [21], FreeNet [22], KaZaA [23]. One user can publish its shared data items such as songs or pictures into the P2P network. The developed P2P lookup mechanisms enable one to efficiently locate its desired data item by using the file name as a key value. Some P2P system protocols like Pastry [11], CAN [12], Chord [15], and Tapestry [25] can support an efficient search of logarithmic time. Also, P2P networks with self-organizing and self-configuring features can provide the benefits of load balancing and fault tolerance.

We organize all HAs installed in the network into a P2P network such that these HAs will work together and share overall workload. We then take the address bindings as shared data and distribute them into the constructed P2P network. Any binding update will be performed into the P2P network. We can identify these address bindings by different key values like IP addresses, e-mail addresses and E.164 telephone numbers. One can locate the address binding of an MN by issuing a P2P lookup request, which carries the MN's key value, into the P2P network.

Our proposed system has the following prominent features: local update, shared workload, and fast lookup. We dynamically select an HA near to the MN in the P2P network to be the MN's temporary HA. The temporary HA will maintain the binding update of the MN during the association time. This can save update cost particularly for high-mobility users. Besides, we consider the workload of an HA. If heavily loaded, we will seek another HA to share the workload. The connection setup time is a concern in mobility management. We provide fast lookup by locally caching some HAs' addresses one user frequently accessed to. This can shorten the time for locating an address binding when a connection is going to be established.

The Voice-over-IP (VoIP) applications have greatly attracted Internet users' attention and are increasingly popular. One commercial system called Skype [24] which uses the P2P technology enables users to make Internet phone calls. One user can attach its computer to a so-called super node. The super nodes will communicate with each other such that a user can know who present in the network from its attached super node. Our proposed system can be developed to provide the VoIP applications especially for mobile users. We can provide seamless roaming for users during voice sessions. By contrast, Skype cannot support this kind of mobile services unless the Mobile IP protocol is used. However, a simple combination of Mobile IP and Skype will incur poor mobility management for mobile users due to the intrinsic problems of Mobile IP.

The rest of this paper is organized as follows. In section 2, we give a brief survey on the mobility management using Mobile IP. In section 3, we present the design of our proposed approach. Section 4 compares the difference and performance of a variety of approaches. Finally, we give a conclusion in section 5.

2. RELATED WORK

2.1 Mobile IP Basis

Mobile IP specified a mechanism to enable an MN to change its point of attachment without changing its IP address. Both Mobile IPv4 and Mobile IPv6 are discussed in the IETF (Internet Engineering Task Force). In this paper, we explain our main idea based on Mobile IPv4, and the same idea can be deployed in the IPv6 framework. In Mobile IP, an MN is assigned with a permanent home address in its home network, and will borrow a temporary care-of address (CoA) in any foreign network. The HA in the MN's home network will maintain the address binding from the home address to the CoA. The CoA is the IP address assigned by the FA in the currently visited foreign network or can be acquired from the local address pool using protocols such as Dynamic Host Configuration Protocol (DHCP). The former case of getting the CoA is used throughout the paper.

When an MN visits a new foreign network, the MN learns a new FA by listening to the agent advertisement broadcasted from the local FA. The MN sends a registration update request to the new FA which in turn forwards the message to the MN's HA. Then the HA updates the new address binding and sends a registration reply to the MN. Usually, the MN will periodically register to the HA for binding data renewal.

Packets which are sent from a corresponding node (CN) in the Internet and are destined to an MN are first intercepted by the MN's HA, and then tunneled to the current serving FA using the MN's CoA. The FA then decapsulates the tunneled packets and forwards them to the MN. This routing way is known as the triangular routing.

2.2 Mobile IP Enhancement

Mobile IP suffers from several problems on performance. First, the triangular routing will increase the packet delivery time. Second, as an MN moves far away from its permanent HA, the registration update cost to the HA is high. Third, frequent registration updates may happen for an MN with high mobility. Finally, the HA or FA is sensitive to the single-point of failure. Several solutions have been proposed in the past years; however, these solutions have the drawbacks of high maintenance cost and inability to solve all the problems simultaneously. By contrast, we provide a uniform platform (based on the P2P network) with the benefits of easy deployment and low maintenance cost to cope with all the problems of Mobile IP. In [29], the P2P network is also used to support user mobility, but the proposed solution is primitive and cannot efficiently solve the problem of long and frequent registration update. Below, we discuss some typical solutions to these problems.

1. *Triangular Routing*: This problem can be solved by directly establishing a connection from a CN to the currently visited FA of an MN. Here, we need a database to store all users' address binding data and provide a lookup service to query the address binding of any MN. The approaches [1, 3, 14, 18] proposed to use the *Domain Name System* (DNS). The address binding data are viewed as resource records in DNS and are maintained using the dynamic update protocol [16]. When a CN wishes to communicate with an MN, it first queries the DNS using the domain name of an MN. Then the DNS

returns the currently associated FA of the MN to the CN. As a result, the consequent data delivery can be directed to the FA without the intervention of the HA. The drawback of this approach is high maintenance cost in DNS.

Furthermore, there are two design issues that should be concerned when using DNS:

- Cached Address: The CN might cache the query result from DNS and will use the old address binding to connect to an MN. The MN will become unreachable if it has attached to another foreign network. To avoid this cache problem, the lifetime of any cached address should be set to zero.
- Seamless Handoff: The DNS enables a CN to acquire the current location of an MN before any connection establishment. However, the CN would never know any subsequent location change as the MN moves during the connection session. We need a mechanism to support seamless handoff. Two kinds of approaches can be adopted: CN-aware and CN-unaware. In the CN-aware approach, a CN is always notified with the new location of an MN and can re-configure the underlying connection to the MN. The TCP migration technique is used in [14] to support this function. In the CN-unaware approach, a CN which is unaware of an MN's movement simply sends data packets to the initially acquired location. Redirection techniques are used in [1, 3, 18] to forward packets from the old location to the new location.

2. *Long Distant Registration*: Since an MN is permanently associated with an HA, a high registration cost is incurred as the MN moves far away from its HA. The dynamic HA assignment by which a nearby HA to an MN is used becomes a potential solution. In [1, 18], a domain-level agent (called Gateway FA, GFA) whose service range the MN is currently in is selected as a temporary HA of the MN. The association of a temporary HA to an MN is registered in DNS. Considerable update traffic to DNS will degrade system performance. In [7], an FA can select a nearby and lightly loaded HA to be a temporary HA for an MN. Then a redirection link is created between the permanent and temporary HAs. However, how to select a proper HA for an MN is not discussed.
3. *Frequent Registration*: The regional registration [2, 5, 13, 17, 28] is commonly used to reduce the registration cost due to high mobility. The regional registration which is based on hierarchical mobility management makes most of registration updates hidden from an HA. When an MN moves within the same domain (or region), all registration updates are locally handled by a domain-level agent (*i.e.*, GFA). The optimal setting of a domain size was discussed in [17]. At the HA, the address of the current serving GFA of the MN will be recorded instead. The packet delivery from the GFA to the currently visited FA of the MN can be done by a host-specific routing [2, 13], tunneling [5], or forwarding chain [28].
4. *Single-Point of Failure*: The fault-tolerant issue becomes important in Mobile IP. Approaches [26, 27] introduce the agent redundancy by having each HA or FA a backup set. Once an HA or FA is failed, another one from the backup set will be selected. However, these approaches did not consider the load balancing issue and the situation that an HA or FA will occasionally become overloaded. In this paper, we only consider the failure problem of HA, since in most commercial products the FA may not exist and can be replaced by a DHCP server.

3. PEER-TO-PEER BASED MANAGEMENT

In this section, we introduce our proposed mobility management. First, we give a snapshot of the system architecture. Then we discuss each component involved in our system. Finally, we show the processing steps of registration update and packet delivery.

3.1 System Overview

We use the subnet-level granularity to explain the basic operations of our mobility management. With the subnet-level granularity, the coverage area of a subnet, which may contain several base stations, is viewed as a foreign network. Suppose that each subnet is associated with an FA. Several subnets would constitute a domain which is associated with a GFA. First of all, we assume that there is only one service operator providing a P2P network for global usage. This is called a single-operator environment. The functional overview of our proposed architecture is depicted in Fig. 1.

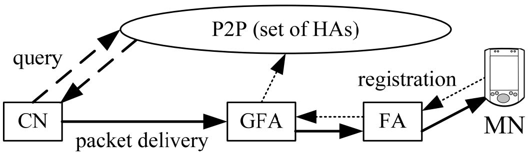


Fig. 1. P2P-based mobile IP in single-operator environments.

All the existing HAs have been organized into a P2P network. An MN is assigned with a permanent HA that is randomly selected (Indeed by a hash function) from the P2P network. Moreover, we will dynamically select an HA that is close to the MN in terms of network proximity to be a temporary HA. The FAs are also organized into a two-level hierarchy for the purpose of regional registration.

When an MN moves within the same domain, the registration update is locally performed to the GFA. Therefore, the GFA can be always known with the FA that currently serves the MN. Only whenever an MN moves to another domain, the registration update to the HA is performed. Here the HA is a temporary one, so the update cost is low. However, when the MN is far away from the temporary HA, we need to select another proper temporary HA and inform this information to the permanent HA.

When a CN would like to connect to an MN, it should first query the MN's address binding by issuing a lookup request into the P2P network. The peers (*i.e.*, HAs in our system) will route this request to the MN's permanent HA which in turn forwards the request to the MN's temporary HA. The temporary HA then returns the CoA that is used by the MN in the current domain. As a result, the CN directly sends packets to the GFA the MN is currently located in. The GFA then forwards these packets to the FA which in turn forwards them to the MN.

It is hard to provide a single P2P network for worldwide usage. One feasible way is to allow different service operators to construct their own P2P networks. We call this as a multi-operator environment. An MN should subscribe to one P2P network (called the home P2P network) and should register to a permanent HA in that network. When the

MN moves to other areas where the P2P networks are possibly provided by other service operators, a temporary HA of the MN is selected from these foreign P2P networks.

To provide worldwide communications, we need another directory to provide the query about where the home P2P network of an MN is. This directory can be constructed into a proprietary one or be provided using the existing DNS. In this paper, we use the DNS for the reason that we can use the same domain name as in DNS to query other information about a user. The overall system architecture is shown in Fig. 2. The DNS mainly provides the lookup of entry points (*i.e.*, the peers' addresses) to different P2P networks. A CN will first query the DNS which returns the address of one of peers within the MN's home P2P network. Then the CN issues a P2P lookup request to this peer and follows the same process as in a single-operator environment.

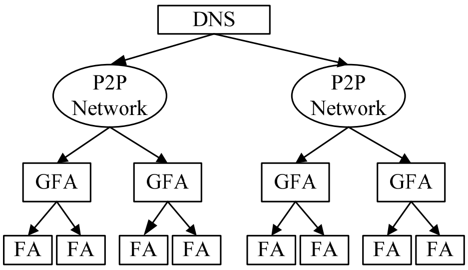


Fig. 2. P2P-based mobile IP in multi-operator environments.

We claim that this architecture can have the following advantages:

- *Update locality*. The frequent registration updates due to the MN's movement in its proximity will be locally performed using the regional registration technique. Moreover, any registration update to a temporary HA is cost saving.
- *Load balancing*. In the P2P networks, each HA has the equal probability to be a permanent HA of an MN. Therefore, the workload is balanced among these HAs when being permanent HAs. The only concerned issue is that when a temporary HA becomes overloaded due to crowded MNs in a certain domain. In this case, we will attempt to find another lightly loaded HA nearby to share the workload of a highly loaded HA.
- *Self-administration*. Each service operator can host its own P2P network, which avoids revealing subscribers related information to other service operators. Also, a service operator can freely increase or decrease the number of HAs, depending on the number of users that are served.

3.2 DNS Structure

The DNS in our proposed architecture acts as a common gateway to the multi-operator environment. Typically, the DNS performs the name resolution of translating a domain name of a host into an equivalent IP address. Here we need the DNS to return information about an MN when given the MN's domain name. This information is the address of one peer in the MN's home P2P network.

The MN's domain name is a design issue. We hope that the domain name is not associated to a physical device that the user equips but the MN itself. This can facilitate an MN to be reached in any number of ways by means of the same domain name. In [8], the telephone number is suggested to be a uniform identifier for a user both in the telephone and the Internet world. IETF Request for Comments (RFC) 2916 [6] states a solution by converting a typical E.164 telephone number into a domain name. One can specify the preferred means to be reached at any particular time into a resource record in DNS. Consequently, one can use a domain name of telephone number style to connect to another user.

Assume each MN is associated with and identified by a unique telephone number (called ENUM [8] throughout the paper). As illustrated in the IETF RFC 2916, a typical E.164 telephone number like +886-3-8634029 can be transformed to a domain name with format "9.2.0.4.3.6.8.3.6.8.8.e164.arpa". The transformation is performed by following the procedure below:

1. Remove all characters with the exception of the digits.
2. Put dots (".") between each digit.
3. Reverse the order of the digits.
4. Append the string ".e164.arpa" to the end.

e164.arpa is the suggested root of the ENUM domain names. A possible portion of the ENUM domain name space is shown in Fig. 3. The second-level domains include one entry for every country code and the third-level domains include one entry for every area code and every operator code.

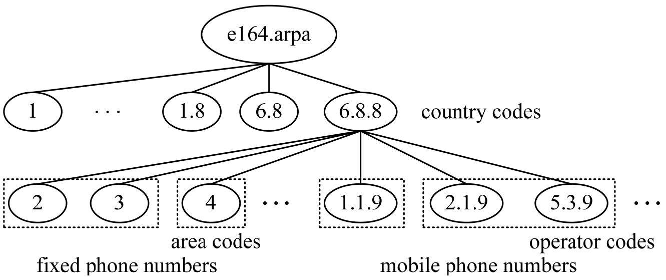


Fig. 3. ENUM domain name space.

Assume that the ENUM domain name space is divided into non-overlapping zones for the administration reason. We indicate each zone in Fig. 3 by using a dotted rectangle. Within the geographical service area of each zone, an individual P2P network is constructed to incorporate all the HAs in that service area. Each zone will have one primary name server and several secondary name servers for the fault-tolerant reason.

Each service operator has the responsibility to register some peers' addresses (as entry points) of each constructed P2P network to the name server of the corresponding zone. The name server when asked with an MN's home P2P network will return the address of one peer randomly selected from these registered peers. Therefore, each name server can perform the following additional function:

- get_peer_address($ENUM\_NAME_{MN}$): This function returns the address of one peer in the home P2P network of the MN with ENUM domain name $ENUM\_NAME_{MN}$.

3.3 P2P Structure

The emerging P2P networks have potential to support large data sharing applications. Some system protocols like Pastry [11], CAN [12], Chord [15], and Tapestry [25] have been proposed for building large P2P networks. These protocols are based on a *distributed hash table* (DHT) which allows shared data items to be uniformly distributed into peers in the P2P network. One user can issue a data lookup request to any peer in the P2P network, and from there the request would be subsequently forwarded by referring to local index information at most log N times before reaching the target peer containing the desired data item.

These protocols can also efficiently support a peer's join and leave with $\log^2 N$ message exchanges. When a new peer inserts into the P2P network, only the shared data items located in the neighboring peers of this peer will be redistributed. The users should periodically re-hash their shared data items into the P2P network to cope with any dynamic network change due to a peer's leave.

3.3.1 P2P network construction

The P2P networks in our proposed architecture mainly connect all the HAs. These HAs are not necessary to be associated with network routers as opposed to the deployment in Mobile IP, and can be placed anywhere. Moreover, an HA can freely join (when newly installed) and leave (when failed) the P2P network. The above two features can facilitate the deployments of Mobile IP related applications.

Any MN would have two HAs in operation: permanent HA and temporary HA (denoted by pHA and tHA, respectively in the following discussion). The pHA is the peer in the MN's home P2P network that is hashed into using the ENUM domain name of the MN. Therefore, the pHA of an MN will not be changed unless the network topology is changed. The tHA is selected from the peers of the visited foreign P2P network of the MN, and this selection has to satisfy the nearness criterion. Therefore, the tHA of an MN will be dynamically changed during the MN's movements. Also, we should notice that a tHA and the corresponding pHA may not belong to the same P2P network in the multi-operator environment.

The distinction between pHA and tHA is as follows. The tHA is a place where the actual address binding is stored. Hence, the MN's registration update is performed to the tHA. The pHA acts as a fixed anchor point to trace the MN's location, so the pHA always records the current address of tHA. Therefore, when an MN associates to a new tHA, the new tHA will register its address to the MN's pHA. The pHA of an MN can be located through the traditional P2P lookup mechanism. The tHA would store the address of pHA when known.

Consider a scenario example shown in Fig. 4. Suppose HA_1 is the pHA of the considered MN. Initially, the MN is located in domain 1, and HA_4 (the nearest HA from GFA_1) is selected as the tHA. Conceptually, we have HA_1 pointing to HA_4 and HA_4 pointing to GFA_1. When the MN moves into domain 2, the HA nearest to GFA_2 is still

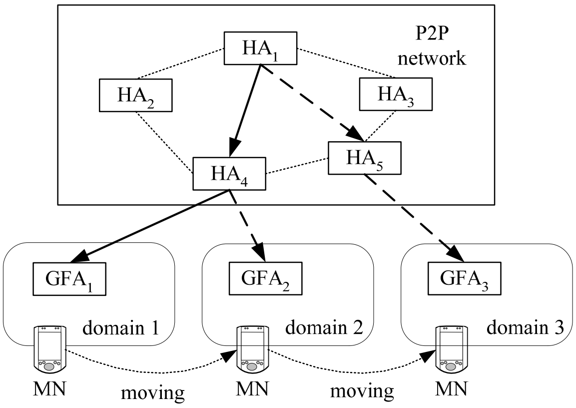


Fig. 4. ENUM domain name space.

HA_4, so the tHA remains unchanged. Continuously, the MN moves into domain 3 and finds that the HA nearest to GFA_3 is HA_5, so the tHA is changed to HA_5. Also, the pHA is updated to point to HA_5.

Each HA in our P2P networks can perform the following functions:

- query_pHA($ENUM\_NAME_{MN}$): This function will return the address of the pHA of the MN with ENUM domain name $ENUM\_NAME_{MN}$. This is performed by the traditional P2P lookup operation.
- query_binding($ENUM\_NAME_{MN}$): This function will return both the address of the pHA and the binding data of the MN. This is performed by first locating the MN's pHA via the traditional P2P lookup operation, and then by following the pointer information of pHA to locate the tHA of the MN. Finally, the tHA returns the address of the recorded pHA and the binding data of the MN to the caller of this function.
- access_binding($ENUM\_NAME_{MN}$, IP_{pHA}): This function will return the binding data of an MN, given the address of the MN's pHA. This is performed by directly connecting to the pHA with IP address IP_{pHA}, and then by following the pointer information to retrieve the binding data in the tHA.

A CN can get the address binding of an MN by either calling function query_binding to one entry point of the MN's home P2P network or calling function access_binding directly to the pHA of the MN. The latter call is performed when the CN has already known the address of the MN's pHA through the previous call of function query_binding. That is, the CN will store the returned pHA's address after calling query_binding in the so-called hot list. The hot list keeps the addresses of pHAs for those MNs to which the CN frequently connects. Each entry in the hot list has the three-tuple value ($ENUM_{MN}$, pHA's address, entry_peer's address), where the entry_peer's address is the retuned value of function get_peer_address. By using the hot list, we can reduce the search time of a pHA in a P2P network. The maintenance of a hot list is similar to traditional cache management and is beyond our discussion.

3.3.2 P2P network maintenance

Now we consider the impact of a peer's join and leave on our proposed architecture. As mentioned before, some peers' data that are hashed to them will be redistributed into some other peers when a new peer joins the P2P network. This will cause the pHAs of certain MNs to change places. When a CN uses the old address of a pHA in its hot list in order to establish a connection with an MN, the call to function access_binding will return a missing error. In this case, the CN has to call function query_binding to refresh the new address into the hot list. When a tHA performs registration to the old address of a pHA, a missing error occurs too. In this case, the tHA will call function get_peer_address to get one peer of the MN's home P2P network, and the call function query_pHA to this peer for getting the new address.

When a peer fails, some MNs will lose contact with their pHAs or tHAs. We follow the soft-state maintenance of Mobile IP to cope with this error. In Mobile IP, an MN would periodically send a registration update request to the currently serving FA and the HA, and this is called binding renewal. The binding renewal is an indication of whether the MN is still active or not. We perform the binding renewal between MN and tHA, and between tHA and pHA, each with a different renewal interval. When an MN fails to connect with a tHA during the binding renewal, the MN will re-select an HA nearby to be a new tHA. Similarly, when a tHA fails to connect with a pHA during the binding renewal, the tHA calls functions get_peer_adress and query_pHA to relocate the pHA.

Finally, we should notice that the peers (consisted of HAs) in our P2P networks are not autonomic ones, and will not frequently join and leave the networks. The maintenance cost is expected to be low as opposed to the pure P2P networks.

3.4 Region Structure

The region structure mainly provides the regional registration. We construct a GFA in each domain, and this GFA can provide a global CoA (GCoA) to a registered MN under its domain. By contrast, an FA can provide a local CoA (LCoA) to a registered MN under its subnet. The tHA and the GFA will record the currently used GCoA and LCoA of the MN, respectively. Therefore, an MN will register its new LCoA to the GFA when changing subnets, and will register its new GCoA to the tHA when changing domains.

The GFA has the responsibility to select an HA in the P2P network that is near to the MN to be a tHA. We perform this function by enabling a GFA to maintain a list of nearby HAs. The GFA can for example broadcast a neighbor discovery within its proximity to get this information. When an MN underlying the domain issues a registration update to the HA, the GFA will randomly select an HA from the list to be the MN's tHA. The selected HA can indeed reject to be an MN's tHA due to current heavy workload. In this case, the GFA will re-select another HA from the list till the successful registration.

After a CN gets the address binding (Indeed, the GCoA) of an MN, it tunnels all packets to the GFA where the GCoA is provided. Then the GFA tunnels these packets to the FA using the recorded LCoA. As can be seen, the CN will directly deliver packets to the GFA after knowing the address binding. One question is that when the MN makes a movement, can we guarantee that the packet delivery is still reachable to the MN? If the movement is within the same domain (see Fig. 5), the new FA will perform a registration

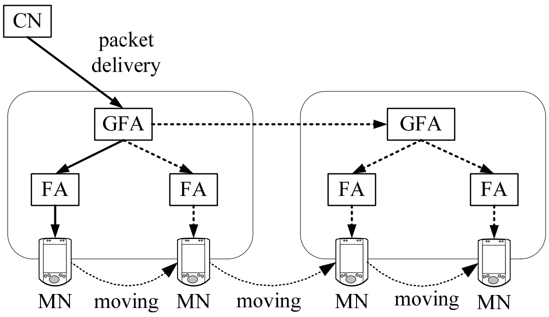


Fig. 5. Packet redirection.

update to the GFA. Hence, the packets will be tunneled from the GFA to the new FA without any problem. If the movement is across different domains, the old GFA will lose the current location of the MN and this will cause a problem on packet delivery. One solution is to construct a redirection path between the old GFA and the new GFA after the MN has registered to the new GFA.

3.5 System Operations

3.5.1 Registration update

In Fig. 6, we depict the signaling flow during registration update. Whenever an MN changes subnets within the same domain, the MN obtains a new LCoA by performing a subnet-specific registration update to the new FA, and then communicates this new LCoA to the serving GFA. Whenever an MN changes domains, it first obtains an LCoA from the serving FA. Then the MN performs a domain-specific registration update by communicating its current LCoA to the GFA of the current domain. The GFA replies to the registration with a GCoA and the address of a selected tHA. Then the MN performs a home registration update by communicating its current GCoA to this selected tHA. If the tHA is different from the previously registered one, a request to establish a redirection link to point to the current tHA is sent to the pHA. If the MN has an active session, a redirection path between the new and old GFAs has to be created as discussed in section 3.4.

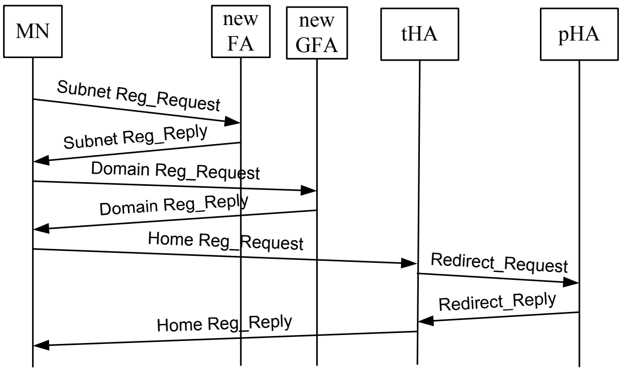


Fig. 6. Registration flow during inter-domain movement.

3.5.2 Packet delivery

In Fig. 7, we depict the signaling flow during packet delivery of a CN which can not find the MN's pHA in the hot list. The CN calls function get_peer_address($ENUM\_NAME_{MN}$) to DNS. The DNS returns the address of one peer (called initiated HA and denoted by iHA) to be the entry point of the MN's P2P network. Then the MN calls function query_binding($ENUM\_NAME_{MN}$) to iHA which in turn performs the traditional P2P lookup operation to locate the MN's pHA. The found pHA forwards the query request to the tHA which in turn returns the binding data to the CN. As a result, the CN can directly tunnel packets to the GFA, and these packets are further tunneled to the FA, and to the MN.

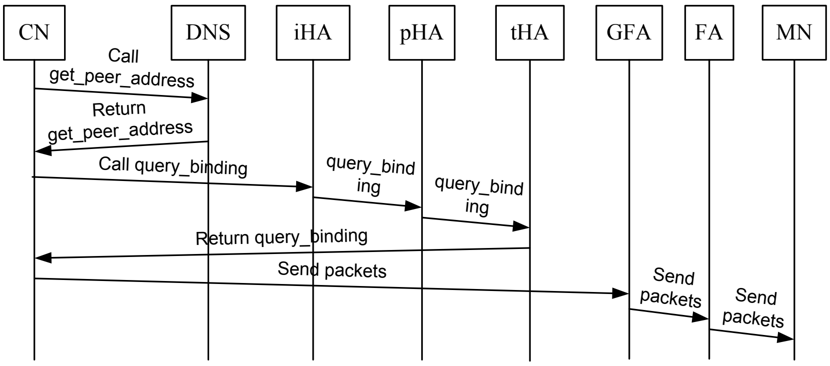


Fig. 7. Packet delivery flow with a miss in the hot list.

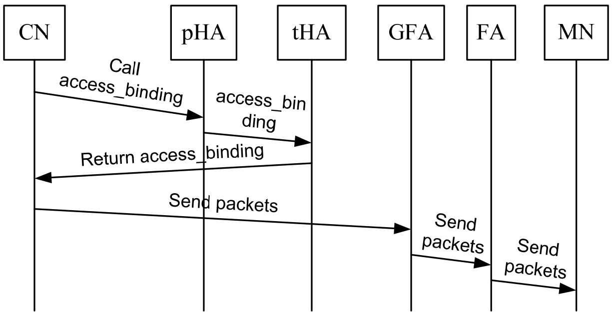


Fig. 8. Packet delivery flow with a hit in the hot list.

If a CN can find the MN's pHA in the hot list, the CN calls function access_binding ($ENUM\_NAME_{MN}$, IP_{pHA}) to the pHA. If the pHA is the correct one, this pHA will forward the request to the tHA which in turn returns the binding data to the CN. Otherwise (in case of the missing of pHA), the CN will call function query_binding again to retrieve the address binding of the MN and will refresh its hot list.

4. PERFORMANCE EVALUATION

We have mentioned three techniques to improve Mobile IP by using DNS, regional registration and P2P technology. Our design architecture makes use of all these techniques. In the following, we evaluate the system performance both qualitatively and quantitatively. We compare our approach with the standard Mobile IP, the pure DNS, and the pure region based Mobile IP.

4.1 Qualitative Analysis

In Table 1, we summarize the advantages and disadvantages of different approaches. Our proposed approach essentially inherits the advantages of DNS and region based ones, so we have no triangular routing (due to DNS) and frequent registration update (due to regions) problems. The self-configuring characteristic of P2P networks makes our approach having no single-point of failure on the HA.

Table 1. Qualitative comparison.

	Standard	Pure Region	Pure DNS	Our Approach
Triangular Routing	Yes	Yes	No	No
Frequent Registration	Yes	No	Yes	No
Single-Point of Failure	Yes	Yes	Yes	No
Load Balancing	No	No	No	Yes
Update Locality	No	Yes	No	Yes

Moreover, our proposed approach has good load balancing due to the following reasons:

- In the DNS: The domain name hierarchy of DNS can naturally distribute the workload to different name servers. Moreover, the random selection of an entry point to a P2P network from a name server can distribute the P2P lookup overhead to different peers.
- In the P2P network: The operation of a P2P lookup query is naturally distributed to the peers involved. When a peer acts as a tHA and becomes heavily loaded, this peer can reject to serve more number of MNs.
- In the GFA: The tHA is randomly selected from a list maintained by the GFA, which can avoid one HA to become heavily loaded.

4.2 Quantitative Analysis

Next, we analyze the registration update, connection setup, and packet delivery costs in terms of hop distances for these approaches. The following parameters will be used in the analysis.

$C_u(P)$: total registration update cost for approach P.
$C_s(P)$: connection setup cost for approach P.
$C_d(P)$: packet delivery cost for approach P.

m: average number of movements during the whole connection.
r: percentage of number of movements that are crossing regions.
p: probability that the tHA is unchanged when the MN is across regions.
h: probability that there has a hit in the hot list when establishing a connection.
H_{x_y}: average number of hops between x and y.
t: average number of iterations in processing a DNS query.
N: number of HAs in a P2P network.

The registration update cost cumulates each delivery cost of a pair of registration update request and reply for each movement. The connection setup cost is for a TCP/IP connection which is set up between a CN and an MN. The packet delivery cost is for a packet conveyed from a CN to an MN. We make the following assumptions in our analysis:

- An MN has an active connection during the whole period, and an MN is changing to a different subnet on each movement.
- The protocol design is based on an MN-initiated control scheme in which all subnet, domain, and home level registration updates are initiated from an MN.
- For simplicity, if the region structure is used, the redirection path is established between the new GFA and the GFA to which the MN first registers in inter-domain movement and between the FA and GFA in intra-domain movement. If the region structure is not used, the redirection path is established between the new FA and the FA to which the MN first registers.
- The influence of a peer's join and leave is neglected, so the recorded pHA's address in the hot list is always correct.

The standard Mobile IP follows the path MN-FA-HA during the registration update. During the connection setup and packet delivery, the path CN-HA-FA-MN is followed. Hence, we have

$$C_u(\text{Standard}) = 2(H_{MN\_FA} + H_{MN\_HA}) \times m$$
$$C_s(\text{Standard}) = 2(H_{CN\_HA} + H_{HA\_FA} + H_{FA\_MN})$$
$$C_p(\text{Standard}) = H_{CN\_HA} + H_{HA\_FA} + H_{FA\_MN}.$$

In the pure region based approach, an inter-domain registration update follows the path MN-FA-GFA-HA; while an intra-domain registration update follows the path MN-FA-GFA. The connection setup and packet delivery follow the path CN-HA-GFA-FA-MN. Hence, we have

$$C_u(\text{Pure Region}) = 2(H_{MN\_FA} + H_{MN\_GFA}) \times (1 - r)m + 2(H_{MN\_FA} + H_{MN\_GFA} + H_{MN\_HA}) \times rm$$
$$C_s(\text{Pure Region}) = 2(H_{CN\_HA} + H_{HA\_GFA} + H_{GFA\_FA} + H_{FA\_MN})$$
$$C_p(\text{Pure Region}) = H_{CN\_HA} + H_{HA\_GFA} + H_{GFA\_FA} + H_{FA\_MN}.$$

In the pure DNS based approach, the path MN-FA-DNS is followed during a registration update. Besides, a redirection path is created between two FAs. The path CN-DNS-CN-FA-MN is followed during a connection setup. The path segment CN-DNS-CN is to iteratively locate the proper name server in the domain name hierarchy. A packet delivery follows the path CN-FA-FA-MN, where the path segment FA-FA is the redirection path. We have

$$C_u(\text{Pure DNS}) = 2(H_{MN\_FA} + H_{MN\_DNS} + H_{FA\_FA}) \times m$$
$$C_s(\text{Pure DNS}) = 2(t \times H_{CN\_DNS} + H_{CN\_FA} + H_{FA\_MN})$$
$$C_p(\text{Pure DNS}) = H_{CN\_FA} + H_{FA\_FA} + H_{FA\_MN}.$$

In our approach, the path MN-FA-GFA-tHA-pHA is followed during an inter-domain registration update. If the tHA remains unchanged, the update to the pHA is needless. By contrast, the path MN-FA-GFA is followed during an intra-domain registration update. The cost of establishing the redirection path should be additionally counted. The path CN-DNS-CN-iHA-pHA-tHA-CN-GFA-FA-MN is followed during a connection setup if the MN is not recorded in the hot list; otherwise, the path CN-pHA-tHA-CN-GFA-FA-MN is followed. A packet delivery follows the path CN-GFA-GFA-FA-MN. We have

$$C_u(\text{Our approach}) = 2(H_{MN\_FA} + H_{MN\_GFA}) \times (1 - r)m + 2(H_{MN\_FA} + H_{MN\_GFA} + H_{MN\_tHA} + (1 - p) \times H_{HA\_HA} + H_{GFA\_GFA}) \times rm$$
$$C_s(\text{Our approach}) = (1 - h) \times C_s(\text{Our approach with a miss}) + h \times C_s(\text{Our approach with a hit})$$
$$C_s(\text{Our approach with a miss}) = 2t \times H_{CN\_DNS} + H_{CN\_iHA} + (\log N + 1)H_{HA\_HA} + H_{CN\_HA} + 2(H_{CN\_GFA} + H_{GFA\_FA} + H_{FA\_MN})$$
$$C_s(\text{Our approach with a hit}) = 2H_{CN\_HA} + H_{HA\_HA} + 2(H_{CN\_GFA} + H_{GFA\_FA} + H_{FA\_MN})$$
$$C_p(\text{Our approach}) = H_{CN\_GFA} + H_{GFA\_GFA} + H_{GFA\_FA} + H_{FA\_MN}.$$

4.3 Simulation Results

The parameter settings in our experiments are listed in Table 2. The settings of the H_{x_y} values are represented by Fig. 9. For example, the hop distance is 4 between an MN and an HA, and is 2 between two HAs in the P2P network.

Table 2. Parameter settings.

Parameter	Value
m	5~14 (default 8)
r	0%~80% (default 40%)
p	0.4
h	0~1
t	2
N	20

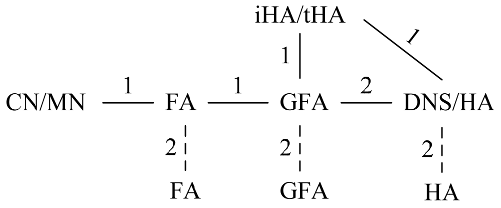


Fig. 9. Relative distances in hops in the simulated network.

In Fig. 10, we show the total registration cost during the whole connection as the number of movements is getting large. The DNS based approach performs worse due to long distant registration updates to DNS and the redirection path updates. Our approach (denoted as P2P in the figure) suffers little overhead from the P2P network and performs

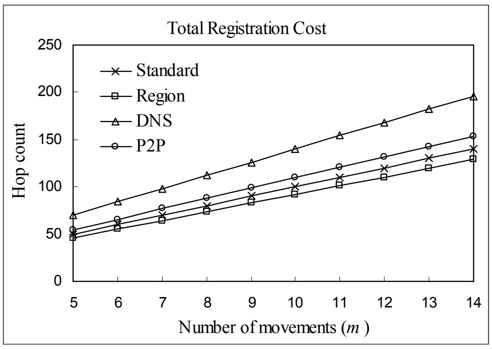


Fig. 10. Effect of parameter *m*.

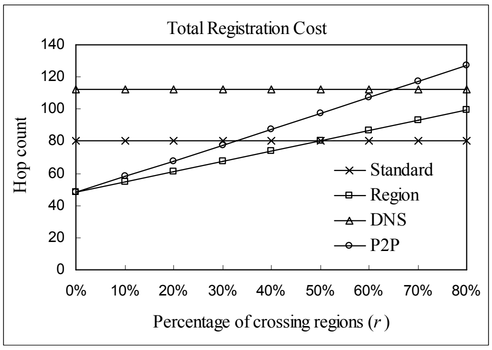


Fig. 11. Effect of parameter *r*.

worse than the standard one. That is, we get the advantages of load balancing and fault tolerance at the expense of maintenance cost. The region based approach has good performance on the registration update due to the user's movement of locality.

In Fig. 11, we show the total registration cost as the percentage of movements across regions is getting large. Those non-region based approaches (the standard and DNS based ones) are invariant with the parameter *r*. Those region based approaches (including ours) perform well particularly when *r* is small (*i.e.*, movement with high locality). For instance, our approach will outperform the standard one when *r* is less than 30%.

We show the connection setup and packet delivery costs of different approaches in Fig. 12. Our approach has the highest connection setup cost when there has a miss in the hot list (*h* = 0), and has significant improvement when there has a hit in the hot list (*h* = 1). One possible further reduction on the connection setup time is by locally checking the binding data in the FA and the GFA. For example, when a CN and an MN are all located in the same domain, the query about the address binding of the MN sent by the CN can be locally answered by the FA or the GFA. The iterative query on DNS brings overhead on the connection setup. The DNS based and our approaches can have good performance on packet delivery after a connection has been set up. Other approaches (the standard and region based ones) suffering from the triangle routing problem have bad performance on packet delivery.

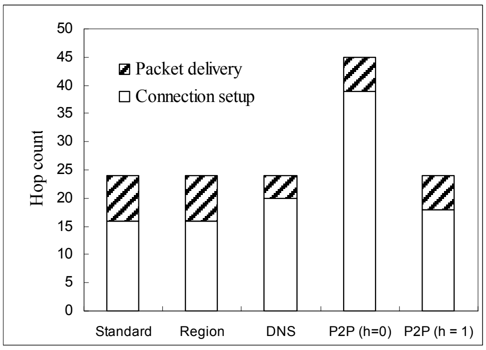


Fig. 12. Basic operations cost.

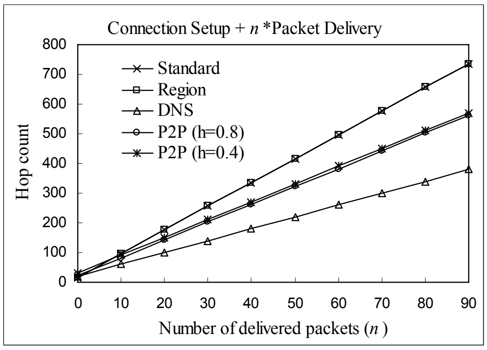


Fig. 13. Connection and delivery cost.

We use the expression $C_s(P) + n \times C_d(P)$, where n is the number of packets delivered, to show the overall performance of packet transmission for a certain approach. In Fig. 13, we found that the standard and region based approaches perform equally. Our approach outperforms the previous two especially when n is large. This is due to the low packet delivery cost of our approach. A high hit ratio in the hot list can slightly improve the performance. The DNS based approach performs best among others due to the lowest packet delivery cost, but this approach has a high registration cost and the problems of scalability and load balancing.

5. CONCLUSION

In this paper, we introduce the emerging P2P network technology into Mobile IP to efficiently support mobility management. In our proposed architecture, we provide dynamic HA assignment and the capability of load balancing and fault tolerance on all the HAs. We overcome the problems of Mobile IP such as triangular routing, long distant and frequent registration update, at the expense of the delay on connection setup. We consider both single and multiple operators' environments, and use the DNS to integrate different operator's P2P networks. Actually, the DNS can be replaced by a lightly weighted directory server or an AAA (Authentication, Authorization, and Accounting) server. The AAA server can additionally provide the security and billing capacity.

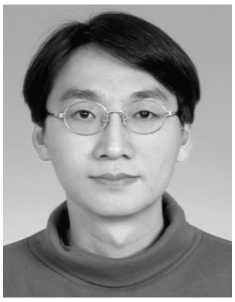
Using the P2P network has another advantage in group communications. Each user can subscribe its interests on certain topics and can submit this subscription into the P2P network. Those subscriptions with the same or similar interests will be hashed into the same peer or nearby peers. We can then use the similarity search in the P2P network to find one's friends with the common interest. A mobile multicasting protocol in the P2P network is needed to support the group communications. We will further consider this issue. The proposed concept on mobility management can be used not only in Mobile IP but also other location protocols such as HLR/VLR (Home Location Register/Visitor Location Register) in cellular communication systems and SIP (Session Initiation Protocol) in the Internet. We will extend our work to these fields in the future.

REFERENCES

1. Y. Chen and T. Boult, "Dynamic home agent reassignment in mobile IP," *IEEE Wireless Communications and Networking Conference*, 2002, pp. 44-48.
2. A. Campbell, J. Gomez, S. Kim, Z. Turanyi, C. Y. Wan, and A. Valko, "Design, implementation and evaluation of cellular IP," *IEEE Personal Communications Magazine*, Vol. 7, 2000, pp. 42-49.
3. M. Conti, E. Gregori, and S. Martelli, "DNS-based architectures for an efficient management of mobile users in Internet," in *Proceedings of the 15th International Symposium on Parallel and Distributed Processing*, 2001, pp. 1957-1964.
4. R. Caceres and V. N. Padmanabhan, "Fast and scalable wireless handoffs in support of mobile Internet audio," *ACM/Baltzer Journal of Mobile Networks and Applications*, Vol. 3, 1998, pp. 351-363.

5. S. Das, A. Mcauley, A. Dutta, A. Misra, K. Chakraborty, and S. K. Das, "IDMP: an intradomain mobility management protocol for next-generation wireless networks," *IEEE Wireless Communications*, Vol. 9, 2002, pp. 38-45.
6. P. Faltstrom, "E.164 number and DNS," IETF RFC 2916, 2000.
7. M. Kulkami, A. Patel, and K. Leung, "Mobile IPv4 dynamic home agent (HA) assignment," IETF RCF 4433, 2004.
8. C. Mctaggart, "Telephone numbers, domain names, and ENUMbers," *IEEE Communications Magazine*, Vol. 40, 2002, pp. 26.
9. L. Morand and S. Tessier, "Global mobility approach with mobile IP in all IP networks," in *Proceedings of the IEEE International Conference on Communications*, Vol. 4, 2002, pp. 2075-2079.
10. C. Perkins, "IP mobility support for IPv4," IETF RFC 3344, 2002.
11. A. Rowstron and P. Druschel, "Pastry: scalable distributed object location and routing for large-scale peer-to-peer systems," in *Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms* (*Middleware*), 2001, pp. 329-350.
12. S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content-addressable network," in *Proceedings of ACM SIGCOMM*, 2001, pp. 161-172.
13. R. Ramjee, K. Varadhan, L. Salgarelli, S. R. Thuel, S. Y. Wang, and T. L. Porta, "HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks," *IEEE/ACM Transactions on Networking*, Vol. 10, 2002, pp. 396-410.
14. C. Snoeren and H. Balakrishnan, "An end-to-end approach to host mobility," in *Proceedings of the 6th International Conference on Mobile Computing and Networking* (*MOBICOM*), 2000, pp. 155-166.
15. I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: a scalable peer-to-peer lookup service for internet applications," *IEEE/ACM Transactions on Networking*, Vol. 11, 2003, pp. 17-32.
16. P. Vixie, S. Thomson, Y. Rekhter, and J. Bound, "Dynamic updates in the domain name system (DNS UPDATE)," IETF RFC 2136, 1997.
17. J. Xie and I. F. Akyildiz, "A distributed dynamic regional location management scheme for mobile IP," in *Proceedings of IEEE INFOCOM*, Vol. 2, 2002, pp. 1069-1078.
18. R. Zheng, Y. Ge, J. C. Hou, and S. R. Thuel, "A case for mobility support with temporary home agents," *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol. 6, 2002, pp. 32-46.
19. L. Becchetti, F. D. Priscoli, T. Inzerilli, and P. Mahonen, "Enhancing IP service provision over heterogeneous wireless networks: a path toward 4G," *IEEE Communications Magazine*, Vol. 39, 2001, pp. 74-81.
20. Napster, http://www.napster.com.
21. Gnutella, http://gnutella.wego.com.
22. FreeNet, http://freenet.sourceforge.net.
23. KaZaAe, http://www.kazaa.com.
24. Skype, http://www.skype.com.
25. Y. Zhao, H. Ling, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz, "Tapestry: a resilient global-scale overlay for service deployment," *IEEE Journal on Selected Areas in Communications*, Vol. 22, 2004, pp. 41-53.

26. J. W. Lin and J. Arul, "An efficient fault-tolerant approach for mobile IP in wireless systems," *IEEE Transactions on Mobile Computing*, Vol. 2, 2003, pp. 207-220.
27. J. H. Ahn and C. S. Hwang, "Efficient fault-tolerant protocol for mobility agents in mobile IP," in *Proceedings of the 15th International Symposium on Parallel and Distributed Processing*, 2001, pp. 1273-1280.
28. S. C. Lo, G. Lee, W. T. Chen, and J. C. Liu, "Architecture for QoS and mobility support in IP-based wireless networks," *IEEE Journal on Selected Areas in Communications*, Vol. 22, 2004, pp. 691-705.
29. I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana, "Internet indirection infrastructure,“ in *Proceedings of ACM SIGCOMM*, 2002, pp.73-86.



Shou-Chih Lo (羅壽之) received the B.S. degree in Computer Science from National Chiao Tung University, Taiwan, in 1993, and the Ph.D. degree in Computer Science from National Tsing Hua University, Taiwan, in 2000. He joined the Computer and Communication Research Center at National Tsing Hua University in 2000 as a Postdoctoral Fellow. He has been with National Dong Hwa University, Taiwan, since 2004 and is now an Assistant Professor in the Department of Computer Science and Information Engineering. His current research interests are in the area of mobile and wireless Internet with emphasis on mobility management and MAC protocols with QoS guarantee. He also works on problems related to index and data allocation on broadcast channels. Dr. Lo received the Best Ph.D. Thesis Award from the Chinese Institute of Information & Computer Machinery in 2000.