WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2003; **3**:311–327 (DOI: 10.1002/wcm.90)

A batch-update strategy for the distributed HLRs architecture in PCS networks

Hang-Wen Hwang, Chien-Chao Tseng*,†
and Ming-Feng Chang

Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu, Taiwan, Republic of China

Summary

Owing to the increasing population of mobile subscribers, the rapidly expanding signaling traffic has become a challenge to the mobility management in PCS networks. Multiple database schemes to reduce signal traffic and to solve the bottleneck problem of the single *home location register* (HLR) architecture have been proposed by many researchers. However, in most of the multiple location databases or HLR systems, extra signaling is required for the multiple database updates. We propose a batch-update strategy, instead of the immediate update method, for the location-tracking schemes with replication to reduce the signaling overhead. In this paper, we first introduce a distributed HLRs architecture in which each HLR is associated with a localized set of VLRs and the location registrations and queries are processed locally. Then we propose our batch-update strategy and present two pointing schemes for inter-HLR call deliveries. The numerical result shows that our approach can effectively decrease the signaling cost of location registration and call delivery compared with the IS-41 standard. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS
distributed HLRs
mobility management
location registration
call delivery
batch update
mobile terminal

Published online: 3 September 2002

*Correspondence to: Professor Chien-Chao Tseng, Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu, Taiwan 300, Republic of China.
†E-mail: cctseng@csie.nctu.edu.tw

1. Introduction

The personal communication service (PCS) is a system that provides wireless telecommunication to the subscribers with mobile terminals (MTs). To enable the mobile subscribers to communicate with a remote terminal (static or mobile) regardless of its current location, the PCS network must implement the mobility management function to keep track of the MTs and locate the called MT when a call is initiated. The two most commonly used mobility management standards are EIA/TIA IS-41 [1] and ETSI GSM MAP [2].

The PCS mobility management has two basic operations, *location registration* and *location tracking*. The former is the process that an MT informs the network of the location changes when it moves and the latter is required when the network attempts to deliver a call to a target MT. Both IS-41 and GSM standards employ a two-level database architecture, which consists of the *home location register* (HLR) and the *visitor location registers* (VLRs). The HLR contains the profiles of the users who have subscribed to the services. When the MT moves into a new registration area (RA), a temporary record is created in the visited VLR. The VLR then sends a message to the HLR for location registration. As described in Reference [3], in IS-41 Revision B, the registration message is first forwarded to a signal transfer point (STP) through Signaling System no. 7 (SS7) for global title translation (GTT). After the HLR address of the MT is found via the GTT procedure, the message is forwarded to the HLR. Similarly, if the network wants to deliver a call to an MT from a VLR or originating switch (initiated by a static terminal), a GTT procedure is required to access the HLR. Then the HLR sends a query to the serving VLR of the MT. The serving VLR returns a routable address called a temporary-location directory number (TLDN) to the calling VLR or originating switch through the HLR. On the basis of the TLDN, a trunk is established from the calling switch to the called MT.

Because of the heavy traffic generated by PCS location registration and tracking, the HLR may become bottlenecked. To reduce the traffic to an HLR, one natural solution is to distribute the HLR function in several locations. Under the distributed architecture, we assume that each distributed HLR (DHLR) is nearby or is collocated with the GTT STP, so that the signaling for GTT can be neglected or avoided [3]. However, for consistency of the location information among HLRs, extra traffic is generated for multiple HLR updates. Therefore, in this paper we propose a batch-update strategy to reduce the overhead for the multiple HLR updates. This batch-update strategy could make the DHLR architecture more feasible.

2. Related Research

Owing to the increasing population of mobile subscribers, the signaling traffic of the PCS network is expanding rapidly. So the mobility management for a huge number of the MTs becomes more important. In addition to the studies [4–9] for reducing the paging delay, many schemes have been proposed to improve the location management [3,10–18]. These schemes aim to reduce the signaling traffic generated by location registration and/or to make the call delivery more efficient.

A fully distributed strategy for location registration is proposed in Reference [18]. Under this scheme, all the location databases are organized as a tree structure. Each database (node of the tree) contains the location information of the MTs that reside in its subtree area. So the processing of the location registration and tracking is effectively localized. However, an MT's movement or call between large subtree areas will cause location updates or queries in many databases.

A *location forwarding* strategy is first proposed in Reference [14] to reduce the signaling cost for location registration. When an MT moves to a new registration area, a pointer is set up from the previous VLR and the location registration to HLR is no longer needed. As a call is generated, the HLR can find the current VLR of the called MT by following the chain of forwarding pointers. In Reference [3], a DHLR architecture is further suggested to solve the bottleneck problem of the single HLR. The incoming calls are distributed to their closest HLRs for location queries. As a result, the calls can be delivered efficiently. In Reference [15], the authors proposed a modification to limit the forwarding pointers to one step at the most. However, both the above location forwarding schemes pose reliability problem since a failure in one VLR can result in the loss of the track of all MTs that are currently visiting other VLRs but their forwarding pointer chains pass across the malfunctioned VLR.

In Reference [17], the authors propose a *dynamically hierarchical database* architecture. A level of database, called *directory register* (DR), is implemented between the HLR and VLR. The DRs determine the distribution strategy of location information for each of their associated MTs based on their mobility and call arrival parameters. Basically, the location registrations of all MTs are distributed to the DRs and the DRs could deliver a call with the already set up location pointer. This implementation effectively reduces the signal traffic in most cases. However, extra signaling is needed for multiple DR updates. Besides, a call delivery initiated by a static terminal does not benefit from this architecture but spends time in routing through the DR.

For the data consistency of the MT's location information in distributed databases, the scheme in Reference [17] normally sends messages to update the location information in the associated DRs as soon as the MT's location changes. In this paper, we propose a batch-update method that can significantly reduce the signaling traffic for location management by allowing temporary inconsistency of the MT's location information among DHLRs. Our analysis shows that the impact of this temporary inconsistency of MT's location is negligible on the call delivery in our architecture. In Section 3 we describe how the batch-update strategy is applied in the DHLR architecture, and then evaluate its performance and present the numerical results in Section 4.

3. Distributed HLRs Architecture

In this section, we describe the DHLR/VLRs architecture and the proposed scheme in detail. Figure 1 shows that several HLRs are distributed in the PCS network and that each DHLR serves a number of VLRs. The DHLRs communicate with each other and with their associated VLRs by SS7 network, which is responsible for routing of signaling messages based on their destination addresses. The area covered by a VLR is called the registration area (RA) and the area covered by a DHLR is called the distributed area (DA).

Under the DHLR structure, the location registration for each MT is performed in its local DHLR. The signaling messages for the registration process are no longer sent to the unique HLR located far away. Thus, the traffic on the network will be effectively reduced. However, for the registration of an MT, it is required to update all the DHLRs. Extra signaling messages are needed to complete the updating. We observe that among the DHLRs there exist many movements of the MTs. Each movement requires inter-DHLR messages to complete the registration process. If we exploit these inter-DHLR messages to convey the updating information by batch, the extra update messages are no longer needed. In Section 3.2 an example will be presented to describe the detailed operation of the batch-update strategy. As for call delivery, the location query for a call is always served at the local DHLR. The DHLR can lead the call to the target MT by using a well-updated location pointer. Obviously, this localization of the service for a call delivery lowers the searching cost and also shortens the setup delay for a connection.

As shown in Figure 2(a), an MT currently resides under VLR1, which is managed by DHLR1. DHLR1 locates the MT with a *local pointer*. Meanwhile, in DHLR2 (a remote DHLR) there is a *direct remote*

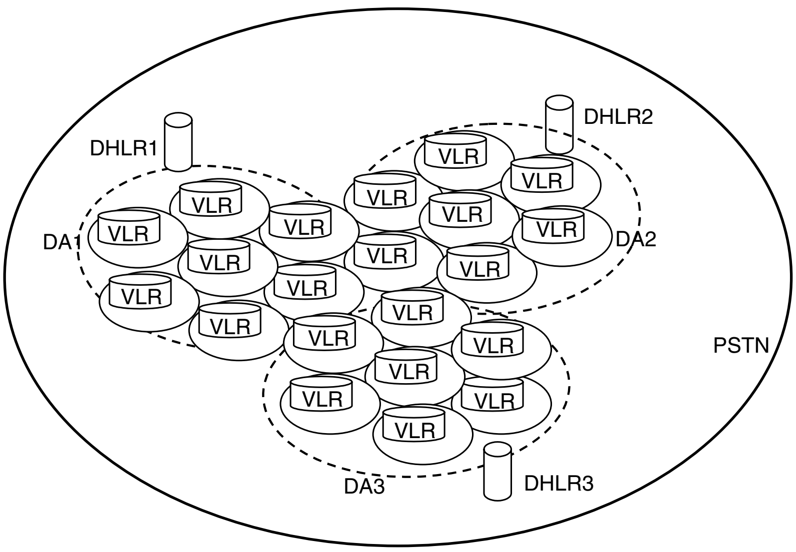


Fig. 1. Distributed HLRs architecture.

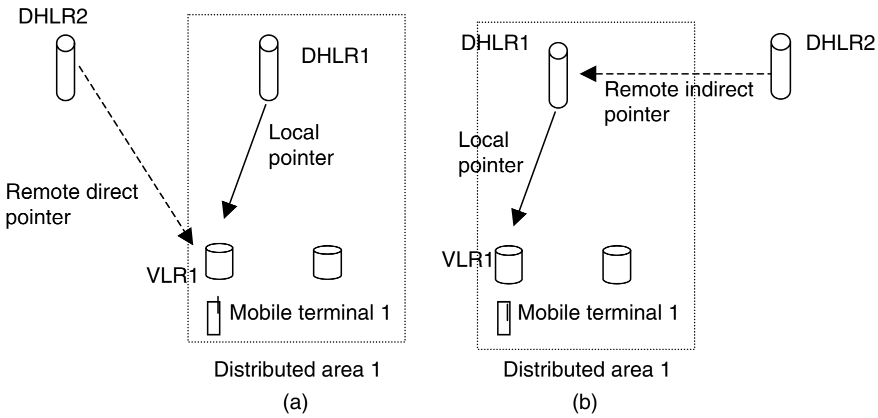


Fig. 2. (a) Configuration under the direct remote pointing scheme and (b) Configuration under the indirect remote pointing scheme.

pointer used to lead to the serving VLR of the MT. As for Figure 2(b), the direct remote pointer in DHLR2 is replaced with an *indirect remote pointer*. By using the indirect remote pointer, DHLR2 locates the currently serving DHLR (DHLR1) of the MT. Then DHLR1 finds the MT with the local pointer. Although we cannot directly locate the VLR in which the target MT currently resides, we need not update the location changes happening within a DA under the indirect remote pointing scheme. This is a trade-off between the direct and indirect remote pointing schemes. Later, we will analyze the performance of each scheme.

3.1. Location Registration

When an MT is initialized in some RA, an initialization message is sent to its local DHLR to create a local pointer that leads to the serving VLR of the MT. At the same time, the local DHLR informs each of the remote DHLRs of this initiation to create a remote pointer for the MT.

As soon as an MT moves to another RA, the MT registers with the DHLR. The registration procedures for an intra-DA movement and an inter-DA movement are depicted in Figures 3 and 4, respectively. The steps shown in Figures 3 and 4 are described as follows:

Step 1. The MT sends a location registration message to the new VLR.

Step 2. The new VLR records the MT and forwards the location registration message to update the local pointer in the associated DHLR. Then the DHLR sends an acknowledgement message back to the VLR.

Step 3. The new VLR sends a successful registration message to the MT.

Step 4. If the movement is an intra-DA movement, we have the following:

- The DHLR sends a location cancellation message to the old VLR to cancel the record of the MT.
- The old VLR sends an acknowledgement message back to the DHLR.

Otherwise, if the movement is an inter-DA movement, we have the following:

- The new DHLR sends a location update message to the old DHLR.
- The old DHLR replaces the invalid local pointer by a new remote pointer and sends an acknowledgement message back to the new DHLR.

Step 5. If the movement is an inter-DA movement, we have the following:

- The old DHLR sends a location cancellation message to the old VLR to cancel the record of the MT.
- The VLR sends an acknowledgement message back to the old DHLR

3.2. Batch-update Strategy for Remote Pointers

When an MT moves to a new RA, the local pointer and the remote pointers for that MT must be updated. The local pointer will be updated naturally during the location registration. However, the updating of the

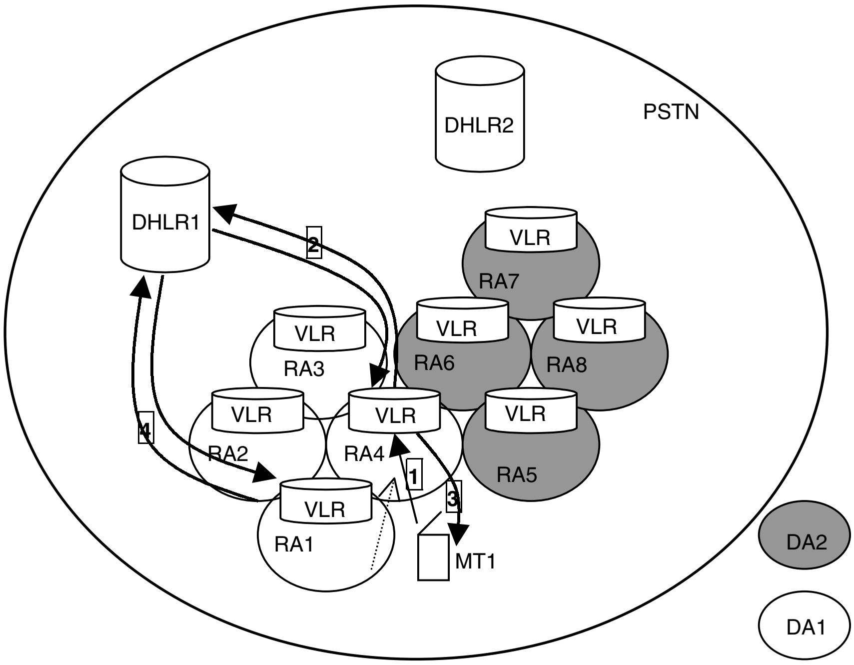


Fig. 3. Registration for an intra-DA movement.

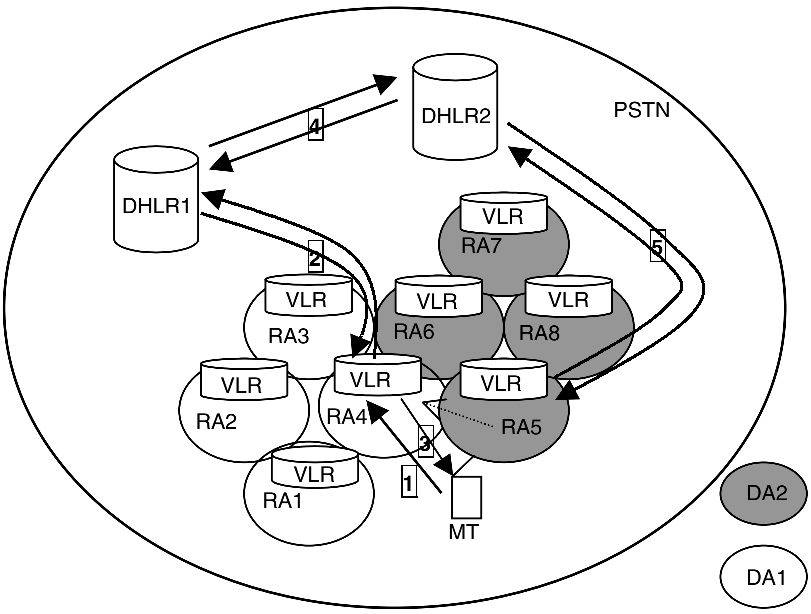


Fig. 4. Registration for an inter-DA movement.

remote pointers must be done explicitly. Therefore, we propose a batch-update method to reduce the cost of updating the remote pointers. In our method, remote pointers are always updated in a batch fashion only when an inter-DA movement occurs. Figure 5 shows an example of how the DHLRs update the remote pointers for each other. We assume that two MTs, MT2 and MT3, have changed their locations in DA1 and MT5 has moved to DA1 since the last inter-DA movement between DHLR1 and DHLR2. When MT1 moves to DA1, MT1 initiates an inter-DA registration procedure. In Step 4 of the procedure, in addition to the location of MT1, the new locations of MT2, MT3 and MT5 are sent in the update message

For the *direct* remote pointing scheme, the contents in the messages are

Update message :

HEADER	MT1 : VLR4	MT2 : VLR2	MT3 : VLR3	MT5: VLR5

Acknowledgment message:

HEADER	MT4: VLR13

For the *indirect* remote pointing scheme, the contents in the messages are

Update message :

HEADER	MT1 : DHLR1	MT5 : DHLR1

Acknowledgment message:

HEADER

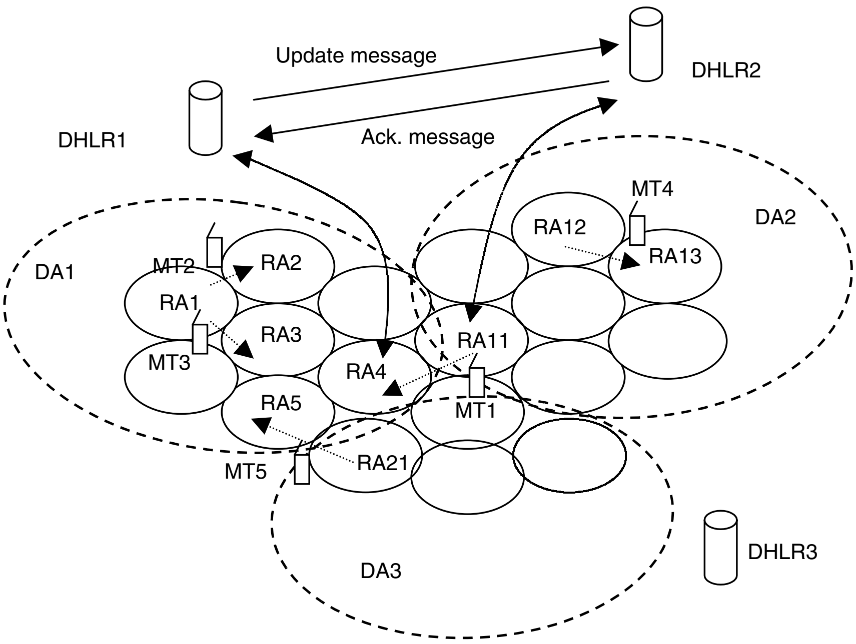


Fig. 5. Update strategy for remote pointers (through an inter-DA location registration).

to DHLR2. DHLR2 will change the remote pointers for MT2, MT3 and MT5 as well as the one for MT1. Similarly, the location change of MT4 will be added to the corresponding acknowledgement message to update the remote pointer for MT4 in DHLR1. As for the indirect remote pointing scheme, the operation for updating indirect remote pointers is similar to that for the direct remote pointing scheme, except for the MT's location information. However, the update or acknowledgement messages must contain the location pointers to the new serving DHLRs, instead of the serving VLRs, of the MTs.

The remote pointers may be temporarily obsolete in our batch-update strategy. If an MT's location is queried when the corresponding remote pointer is obsolete, we call it a *query miss*; otherwise it is a *query hit*. However, by following these obsolete remote pointers, we still can find out the correct location of the target MT. In general, the number of MTs is so huge that the inter-DA movement could be frequent, and almost all the remote pointers could be updated in time for the location queries for the mobile terminating calls. In the next section, we evaluate the average update rate of the remote pointers and show that a very high hit ratio of a location query can be achieved with our proposed update strategy. This effectively reduces the setup time for call delivery without extra update messages.

In order to append the location changes to the update message as soon as possible, each DHLR maintains a table to record the location changes of the MTs. Table I, for example, is the update table for the DHLR2. DHLR1 and DHLR3 are two neighboring DHLRs of DHLR2. For each neighboring DHLR,

Table I. The update table for the remote pointers.

DHLR1	MT5	MT6	...	...
	VLR10	VLR12	...	...
DHLR3	MT1	MT5	MT6	...
	VLR11	VLR10	VLR12	...
...	...	...	...	...

Table II. The relay table for the remote pointers.

From DHLR1 to DHLR3	MT10	MT3	...
	VLR16	VLR18	...
From DHLR3 to DHLR1	MT2	MT7	...
	VLR13	VLR14	...

there is an entry in the location changes table. An entry will be sent to its corresponding DHLR during the next location registration for an inter-DA movement between these two DHLRs. As soon as it is sent out, the entry has to be cleared for collecting the next batch of the changed locations of the MTs.

For those nonneighboring pairs of DHLRs, we can choose a unique relay DHLR between them to relay the update of the remote pointers in both directions. As shown in Figure 6, DHLR2 can relay the location changes for the nonneighboring pair (DHLR1, DHLR3). Whenever DHLR2 receives the location changes of the MTs from either DHLR1 or DHLR3, it will record the location changes in a relay table for each as shown in Table II.

3.2.1. Hit ratio for a query of the remote pointer

As described above, a remote pointer for a target MT is updated by a batch-update method instead of an immediate-update method. As a consequence, a remote pointer may be obsolete when it is accessed for a call connection. Therefore, we would like to evaluate the probability that a remote pointer is always updated in time for a location query initiated by a call.

In general, only a few HLRs are enough to achieve a good performance under the two-level (HLR/VLR) database architecture with multiple HLRs. Therefore, we assume that there are a few DHLRs deployed and their covering areas are fully neighbored in the following analysis. Under such an assumption, a DHLR can directly use our batch-update method to update the remote pointers in every DHLR.

We define the following parameters used for analysis:

D the number of DHLRs

N the average number of MTs within a DA

$1/\lambda_a$ the average RA residence time of an MT

q the probability that the movement of an MT is an inter-DA movement

Furthermore, we assume the network as a homogeneous system. Under this assumption, we can focus our analysis on a single DHLR. In unit time, an MT in a DHLR will take $\lambda_a\, q/(D-1)$ movements into one of the neighboring DAs. For a DHLR with N MTs, there will be $N\lambda_a\, q/(D-1)$ movements into a neighboring DA. Similarly, there will be $N\lambda_a\, q/(D-1)$ incoming movements from a particular neighboring DA. As mentioned above, a DHLR can update the remote pointers in a neighboring DHLR by the location registration procedure stimulated by an MT arriving or leaving the neighboring DA. So we can calculate the average update rate for a remote pointer as follows:

$$R = \frac{2N\lambda_a q}{D-1}$$

A location query for an inter-DA call delivery will be missed if the remote pointer for the target MT cannot be updated in time as shown in Figure 7(a). To evaluate the probability of a query miss, p_m, we first assume the movements for a target MT as a Poisson

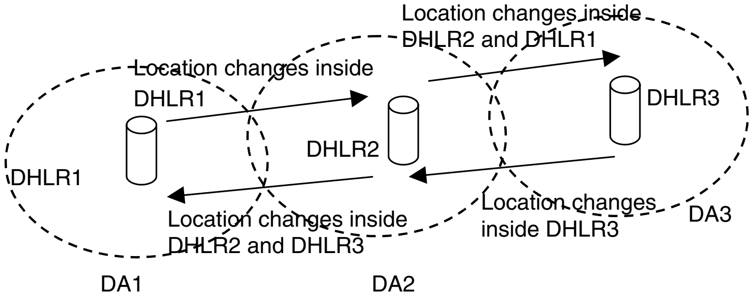


Fig. 6. Relaying the remote pointers for a nonneighboring pair of DHLRs.

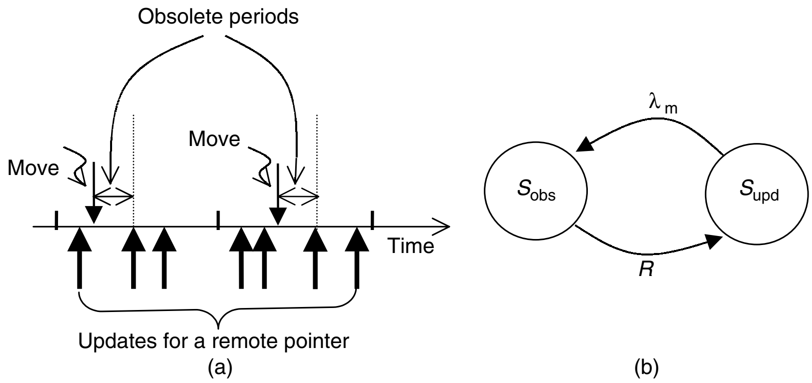


Fig. 7. (a) Illustration for the update timing of a remote pointer and (b) validity state diagram of a remote pointer.

process with rate λ_m, while the updates for a remote pointer of the target MT are also a Poisson process with rate R since an MT's residence time could also be assumed to be an exponential distribution with rate λ_a, and then use a simple Markov chain as shown in Figure 7(b) to model the validity status of a remote pointer. In the Markov chain, the S_{obs} and S_{upd}, respectively, represent the remote pointer for an MT that is obsolete and updated. The steady state probabilities of S_{obs} and S_{upd}, which stand for the probabilities of a query miss and a query hit (p_m and p_h), can be obtained as follows:

$$p_m = \pi_{obs} = \frac{\lambda_m}{\lambda_m + R}, \quad p_h = \pi_{upd} = \frac{R}{\lambda_m + R}$$

We can expect a high probability of p_h since the average number of MTs, N, is normally large.

3.3. Call Delivery

Similar to location registration, call delivery can be classified into intra-DA and inter-DA types. On receiving a call, the DHLR uses the local pointer to deliver the call to the serving VLR of the MT for the former type and uses the remote pointer for the latter. Figure 8 shows the normal operations of call delivery. The steps, as shown in Figure 8, are described as follows:

Step 1. A call is initiated by an MT (or a Fixed Terminal) and forwarded to the VLR (or the Public switched telephone network (PSTN) switch).

Step 2. The VLR (or the PSTN switch) sends a location request message to the local DHLR.

Step 3. When the callee and caller are within the same DA, we have the following:

(a) The DHLR sends a location request to the local VLR that serves the called MT.

(b) The VLR assigns a TLDN to the called MT and sends this TLDN to the DHLR.

If the callee and caller are not in the same DA, as mentioned above, we have two cases of remote pointing.

A: direct remote pointing scheme

(c) The DHLR sends a location request to the remote VLR that serves the called MT.

(d) The VLR assigns a TLDN to the called MT and sends this TLDN to the calling DHLR.

B: indirect remote pointing scheme

(e) The local DHLR sends a location request to the DHLR that serves the called MT.

(f) The called DHLR forwards the location request message to the VLR that serves the called MT.

(g) The VLR assigns a TLDN to the called MT and sends this TLDN to the calling DHLR.

Step 4. The calling DHLR forwards the TLDN to the calling VLR.

Step 5. The calling mobile services switching center (MSC) (or the calling PSTN switch) sets up a connection to the called MSC using this TLDN.

Because the remote pointers are not updated immediately after location changes of the corresponding

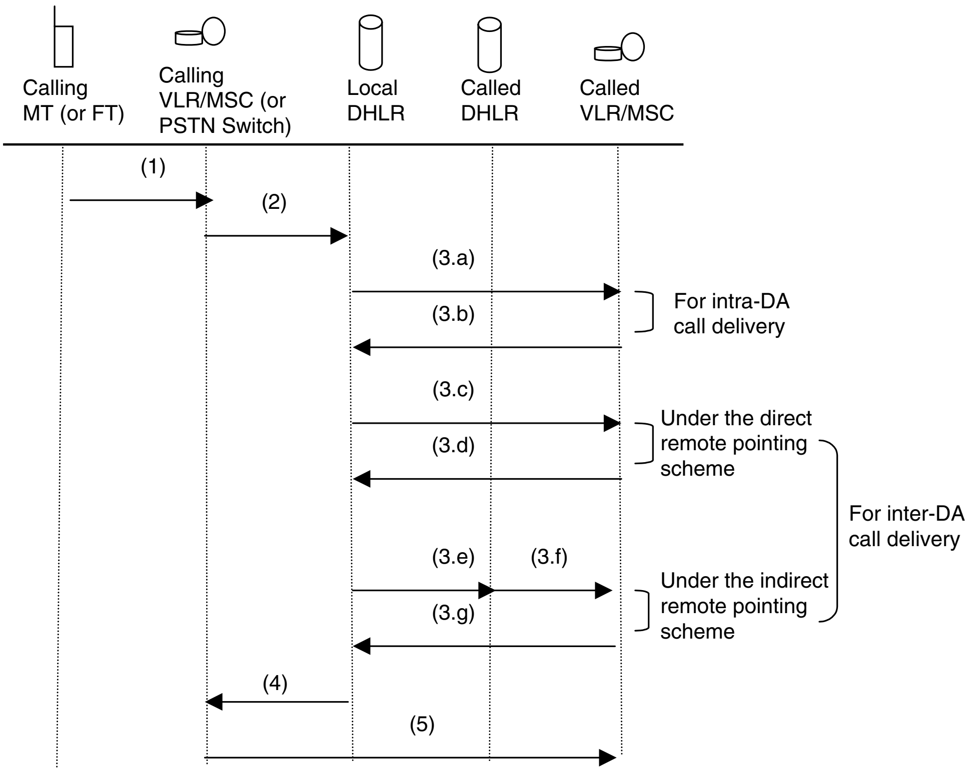


Fig. 8. Normal operations of call delivery.

MT, it is possible that a remote pointer fails to point directly to the right serving VLR or DHLR of the target MT. However, by following the obsolete remote pointers, we can still find out the correct location of the target MT. Figure 9 sketches some worse cases for an inter-DA call delivery under the direct remote pointing scheme. In Figure 9(a), for the called MT, DHLR1 has a pointer that leads to VLR5. After failing to find the MT in the VLR5 pointed out by the obsolete remote pointer, VLR5 forwards the location query to DHLR2. The DHLR then uses the local pointer to find out the serving VLR of the called MT. As described in Section 3.2, under the present scheme the miss ratio for location queries is very low. So the case with two consecutive failures in querying the target MT's location would take place with an extremely low probability. In the following section we neglect the call delivery cases with more than one query of the target MT's location in our analysis.

4. Performance Analysis

Our proposed architecture aims to reduce the signaling cost for location registration and call delivery. However, it should be noted that, our scheme will not incur extra database access compared with the single HLR architecture. Therefore, we just evaluate the performance of our scheme in terms of the signaling cost. The signaling cost could be measured by the bandwidth required to complete the signal transmission. In the following analysis, we use an embedded Markov chain [19] to model the behavior of the MT's movement with respect to its incoming calls from each individual DHLR. The signaling cost for location registration and call delivery will be evaluated separately.

4.1. The Analytical Model

As mentioned in Section 3.2.1, we assume that all DAs are fully neighbored. For a particular DHLR, the location pointer to a target MT may be local or remote. If the pointer is a remote one, a movement of the MT will make it temporarily obsolete under the batch-update strategy. In order to represent different conditions of the location pointers, we model the activity of a particular MT using an embedded Markov chain in which the states, with respect to a call originated from DA_i, are defined as follows:

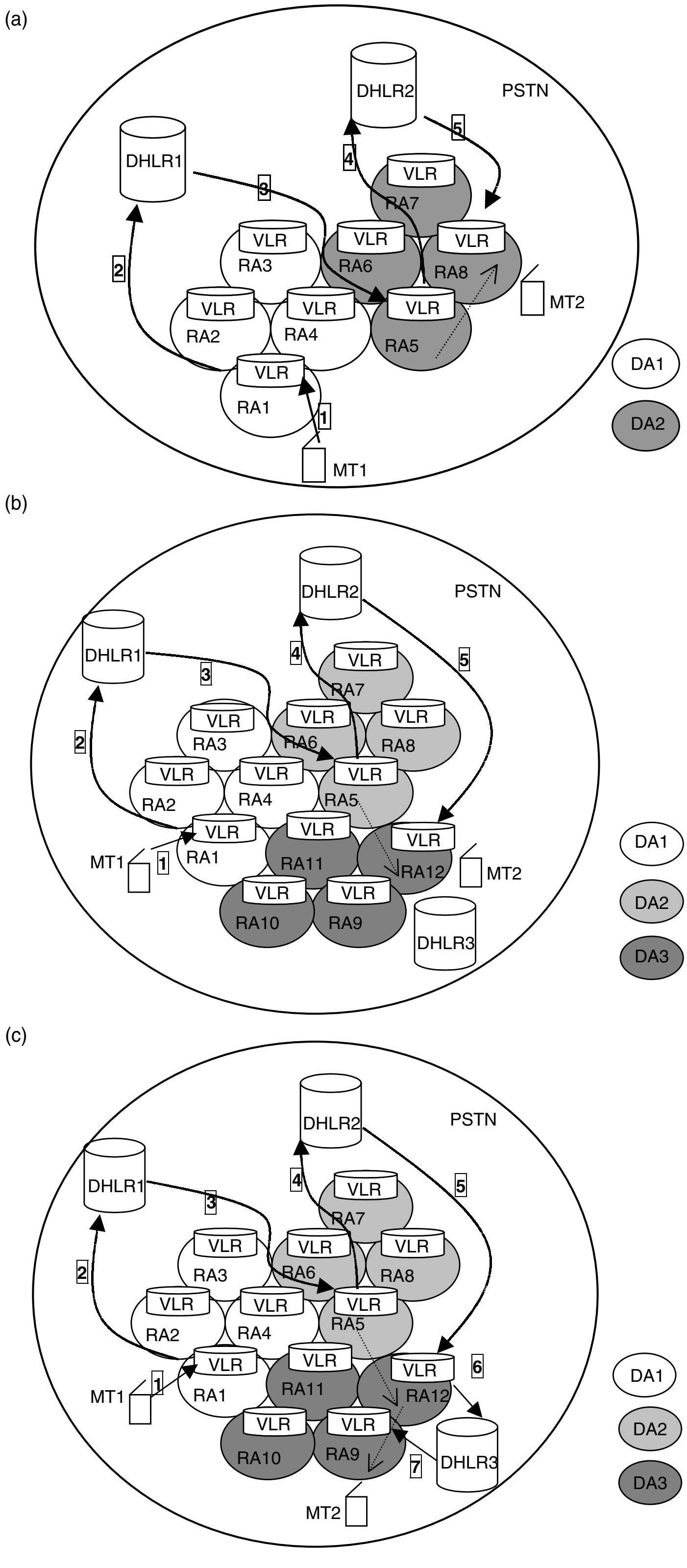


Fig. 9. (a) A remote call delivery with one remote and one local pointer tracing; (b) A call delivery with two remote pointers tracing; and (c) A remote call delivery with two remote and one local pointer tracing.

S^i_{LC} the call is a local one to the MT.

S^i_{RC} the MT has made an inter-DA movement from DA_i but its location information in DHLR_i, a remote DHLR, is still valid. Therefore, a call to the MT from DA_i is a remote one and will be delivered in exactly one step from DHLR_i. If a remote call originates from DA_i, in which the location pointer is obsolete, the delivery will be completed with one remote and one local pointer tracing.

S^i_{RLC} the MT has further made an intra-DA movement after leaving DA_i and its location information in DHLR_i, a remote DHLR, may be obsolete.

S^i_{RrplusC} the MT has made more than two inter-DA movements to a new DA, which is different from DA_i. Similar to S^i_{RLC}, the MT's location information in DHLR_i, a remote DHLR, may be obsolete. If a remote call originates from DA_i, in which the location pointer is obsolete, the delivery will be completed with not less than two remote pointers tracing.

We also assume, for a target MT, that the inter-RA movements are a Poisson process with rate λ_{m}, while the arrivals of its incoming calls from DA_i are a Poisson process with rate λ_{i}. State transitions of the embedded Markov chain occur right after the movement of the MT to one of the adjacent RAs or the arrival of a call from DA_i. We assume that a target MT starts moving in its local DA, DA_i. For each movement, the MT moves out of DA_i with the probability q_i. Once the MT leaves DA_i, it moves back to its local DA with the probability $\mu_i k_i$ in each movement. The parameter k_i is the inter-DA movement probability for the MT outside DA_i, and μ_i is the move–back probability that the MT moves back to DA_i in each inter-DA movement. A high value of μ_i means a high locality of the MT to DA_i. The state transition diagrams for the embedded Markov chain are given in Figure 10.

Let π^i_{LC}, π^i_{RC}, π^i_{RLC} and π^i_{RRplusC} denote the steady state probabilities of being S^i_{LC}, S^i_{RC}, S^i_{RLC} and S^i_{RrplusC}, respectively. We further assume that from each adjacent DA of the DA_i an MT moves to DA_i with equal probability. Therefore, we can derive the following equations:

$$\sum_{i=1, i\neq j}^{D} \mu_i k_i = q_j, \text{ for } 1 \leq j \leq D$$

Then, we have the balance equations as follows:

$$\pi^i_{\mathrm{LC}} + \pi^i_{\mathrm{RC}} + \pi^i_{\mathrm{RLC}} + \pi^i_{\mathrm{RRplusC}} = 1 \quad (1)$$

$$\pi^i_{\mathrm{RC}}\lambda_{\mathrm{m}} = \pi^i_{\mathrm{LC}} q_i \lambda_{\mathrm{m}} + (\pi^i_{\mathrm{RLC}} + \pi^i_{\mathrm{RRplusC}})\lambda_{\mathrm{i}} \quad (2)$$

$$\pi^i_{\mathrm{LC}} q_i \lambda_{\mathrm{m}} = \mu_i k_i \lambda_{\mathrm{m}} (\pi^i_{\mathrm{RC}} + \pi^i_{\mathrm{RLC}} + \pi^i_{\mathrm{RRplusC}}) \quad (3)$$

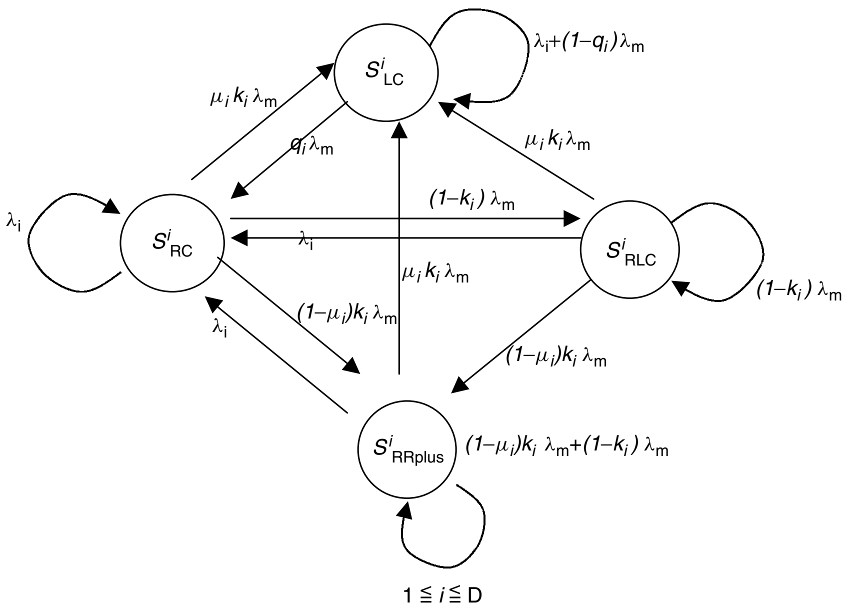


Fig. 10. State transition diagrams for the embedded Markov chain.

$$\pi^i_{\mathrm{RLC}}(k_i\lambda_\mathrm{m}+\lambda_\mathrm{i}) = \pi^i_{\mathrm{RC}}(1-k_i)\lambda_\mathrm{m} \quad (4)$$

$$\pi^i_{\mathrm{RRplusC}}(\mu_i k_i\lambda_\mathrm{m}+\lambda_\mathrm{i}) = (1-\mu_i)k_i\lambda_\mathrm{m}(\pi^i_{\mathrm{RC}}+\pi^i_{\mathrm{RLC}}) \quad (5)$$

After simple derivation, we have

$$\pi^i_{\mathrm{LC}} = \frac{\mu_i k_i}{q_i+\mu_i k_i} \quad (6)$$

$$\pi^i_{\mathrm{RC}} = \frac{q_i(\mu_i k_i\lambda_\mathrm{m}+\lambda_\mathrm{i})}{(q_i+\mu_i k_i)(\lambda_\mathrm{m}+\lambda_\mathrm{i})} \quad (7)$$

$$\pi^i_{\mathrm{RLC}} = \frac{(1-k_i)q_i\lambda_\mathrm{m}(\mu_i k_i\lambda_\mathrm{m}+\lambda_\mathrm{i})}{(k_i\lambda_\mathrm{m}+\lambda_\mathrm{i})(q_i+\mu_i k_i)(\lambda_\mathrm{m}+\lambda_\mathrm{i})} \quad (8)$$

$$\pi^i_{\mathrm{RRplusC}} = \frac{(1-\mu_i)k_i q_i\lambda_\mathrm{m}}{(q_i+\mu_i k_i)(k_i\lambda_\mathrm{m}+\lambda_\mathrm{i})} \quad (9)$$

The average inter-DA movement probability, q, for the MT can be calculated as follows:

$$q = \sum_{i=1}^{D} q_i\pi^i_{\mathrm{LC}}$$

In the next subsection, we use the above steady state probabilities and average inter-DA movement probability to evaluate the signaling costs.

4.2. Cost Evaluation

Under our architecture, the signaling cost comes from three parts, location registration, batch-update of remote pointers and call delivery. The costs for a registration after an inter-RA movement and an inter-DA movement are denoted as c_{ir} and c_{id}, respectively. The cost for a call delivery depends on where the call is from and whether the location pointer is up-to-date or not. Table III groups the call delivery costs under various situations. In the table, the lowercase subscripts 'r' and 'l', respectively, indicate 'remote' and 'local' with respect to the called MT. For example, $\mathrm{DHLR_l}$ indicates the local DHLR for a target MT, while $\mathrm{DHLR_{r1}}$ and $\mathrm{DHLR_{r2}}$ are two remote DHLRs.

Location registration cost. As assumed above, an MT makes λ_m movements per unit time, and each movement is of inter-DA type with probability q. Let α denote the average registration cost for an MT, then

$$\alpha = (1-q)\lambda_\mathrm{m}c_{\mathrm{ir}} + q\lambda_\mathrm{m}c_{\mathrm{id}} \quad (10)$$

Update cost. Because two types of remote pointing schemes are proposed, the update cost for different schemes will be evaluated, respectively.

1) Direct remote pointing scheme: Under this scheme, for each intra-DA movement of a particular MT, we need to update $D-1$ remote pointers in the $D-1$ neighboring DHLRs. However, for each inter-DA movement we only need to update $D-2$ remote pointers since the location pointers of the MT in the other DHLRs are naturally updated in the registration process. The number of location updates for an MT per unit time could be $[(1-q)(D-1)+q(D-2)]\lambda_\mathrm{m}$. The total update cost for an MT could be normalized, with respect to the update cost in the immediate update scheme, as

$$\beta_1 = [(1-q)(D-1)+q(D-2)]\lambda_\mathrm{m}c_{\mathrm{D2D}}\frac{l}{L} \quad (11)$$

where l is the length of location information data for an MT in our scheme and L is the average length of an update message in the immediate update scheme. Besides, c_{D2D} is the average signaling cost for transmitting an update message sent from one DHLR to the other.

2) Indirect remote pointing scheme: As mentioned earlier, an indirect remote pointer is used to point to the serving DHLR of a target MT instead of the serving VLR. The remote pointers are updated only when an inter-DA movement occurs. So we can easily obtain the update cost, β_2, as follows.

$$\beta_2 = q(D-2)\lambda_\mathrm{m}c_{\mathrm{D2D}}\frac{l}{L} \quad (12)$$

Call delivery cost. The call delivery costs for the two proposed schemes are as follows:

Table III. The various costs for a call delivery.

Cost parameter	Finding operation
c_{lc}	$\mathrm{MT} \rightarrow \mathrm{VLR_{calling}} \rightarrow \mathrm{DHLR_l} \rightarrow \mathrm{VLR_{called}}$
c_{rc1}	$\mathrm{MT} \rightarrow \mathrm{VLR_{calling}} \rightarrow \mathrm{DHLR_r} \rightarrow \mathrm{VLR_{called}}$
c_{rc2}	$\mathrm{MT} \rightarrow \mathrm{VLR_{calling}} \rightarrow \mathrm{DHLR_r} \rightarrow \mathrm{VLR} \rightarrow \mathrm{DHLR_l} \rightarrow \mathrm{VLR_{called}}$
c_{rc3}	$\mathrm{MT} \rightarrow \mathrm{VLR_{calling}} \rightarrow \mathrm{DHLR_{r1}} \rightarrow \mathrm{VLR} \rightarrow \mathrm{DHLR_{r2}} \rightarrow \mathrm{VLR_{called}}$
$c_{\mathrm{rc1'}}$	$\mathrm{MT} \rightarrow \mathrm{VLR_{calling}} \rightarrow \mathrm{DHLR_r} \rightarrow \mathrm{DHLR_l} \rightarrow \mathrm{VLR_{called}}$
$c_{\mathrm{rc2'}}$	$\mathrm{MT} \rightarrow \mathrm{VLR_{calling}} \rightarrow \mathrm{DHLR_r} \rightarrow \mathrm{DHLR_r} \rightarrow \mathrm{DHLR_l} \rightarrow \mathrm{VLR_{called}}$

1) Direct remote pointing scheme: For a target MT, a call from DA_i may be classified into four cases with the following probabilities:

- LC_i — the probability that a call originated from DA_i under the state S^i_{LC}.
- RC_i — the probability that a remote call originated from DA_i under the state S^i_{RC}.
- RLC_i — the probability that a remote call originated from DA_i under the state S^i_{RLC}.
- $RRplusC_i$ — the probability that a remote call originated from DA_i under the state $S^i_{RRplusC}$.

Then, the total call delivery cost due to the calls from all DAs is computed as

$$\begin{aligned}\gamma_1 = \sum_{i=1}^{D} \lambda_i [&LC_i c_{lc} + RC_i c_{rc1} \\ &+ RLC_i(c_{rc1} p_h + c_{rc2} p_m p_h) \\ &+ RRplusC_i(c_{rc1} p_h + c_{rc3} p_m p_h \\ &+ f(p_m^2 p_h, p_m^3 p_h, \ldots))] \end{aligned} \quad (13)$$

where $f(p_m^2 p_h, p_m^3 p_h, \ldots)$ represents the summation of all other terms with a parameter p_m and the order of p_m is not less than 2. In the steady state of our model, we have

$$\lim_{t\to\infty} LC_i = \pi^i_{LC}, \lim_{t\to\infty} RC_i = \pi^i_{RC},$$
$$\lim_{t\to\infty} RLC_i = \pi^i_{RLC} \text{ and } \lim_{t\to\infty} RRplusC_i = \pi^i_{RRplusC}$$

Because the cost function $f(p_m^2 p_h, p_m^3 p_h, \ldots)$ with parameters of high order p_m is much smaller than the former term, we can neglect it to obtain a simple form in terms of c_{rc1}, c_{rc2} and c_{rc3}, as shown in Table III.

2) Indirect remote pointing scheme: Following a similar approach, we can derive the total delivery cost of indirect remote pointing scheme as

$$\begin{aligned}\gamma_2 = \sum_{i=1}^{D} \lambda_i [&LC_i c_{lc} + (RC_i + RLC_i) c_{rc1'} \\ &+ RRplusC_i(c_{rc1'} p_h + c_{rc2'} p_m p_h \\ &+ f'(p_m^2 p_h, p_m^3 p_h, \ldots))] \end{aligned} \quad (14)$$

The intra-DA movement of the MT does not change the cost of searching the target MT. Therefore, the cost of the remote call delivery under the states S^i_{RC} and S^i_{RLC} are the same, that is, $c_{rc1'}$. Similarly, the cost function, $f'(p_m^2 p_h, p_m^3 p_h, \ldots)$, is too small to be taken into account. The costs $c_{rc1'}$ and $c_{rc2'}$ are also shown in Table III.

Total cost. Summing the Equations (10), (11) and (13), we obtain the total cost under the direct remote pointing scheme $C_1 = \alpha + \beta_1 + \gamma_1$. Similarly, we obtain the cost under the indirect remote pointing scheme $C_2 = \alpha + \beta_2 + \gamma_2$. Besides, we also derive the total cost under the immediate update method that is given by

$$\begin{aligned} C_3 = \alpha &+ [(1-q)(D-1) + q(D-2)]\lambda_m c_{D2D} \\ &+ \sum_{i=1}^{D} \lambda_i [LC_i c_{lc} + (1 - LC_i) c_{rc1}] \end{aligned} \quad (15)$$

4.3. Numerical Results

In this section, we present the numerical results based on our analytical model. The signaling costs are constituted from the following elementary cost parameters and can be calculated from the expression shown in Table IV.

- c_{LD2V} — the average signaling cost of a message transmission between a VLR and its local DHLR
- c_{RD2V} — the average signaling cost of a message transmission between a VLR and a remote DHLR
- c_{D2D} — the average signaling cost of a message transmission between two DHLRs
- c_{SH2V} — the average signaling cost of a message transmission between a VLR and the single HLR under IS-41 standard
- c_{V2G} — the average signaling cost of a message transmission between a VLR and the *GTT* STP under IS-41 standard
- c_{G2SH} — the average signaling cost of a message transmission between the *GTT* STP and the single HLR under IS-41 standard

Besides, we assume that the signaling cost is proportional to the routing distance and the message load transmitted between the two communicating sites, obtaining the following relation:

$$c_{V2G} \risingdotseq c_{G2SH} \risingdotseq c_{SH2V} \geq c_{RD2V} \geq c_{D2D} \geq c_{LD2V}$$

The cost values we use in the performance evaluation are shown in Table V.

Table IV. Cost expressions for the location registration and call delivery operations.

Cost parameter	Expression
c_{ir}	$4c_{\mathrm{LD2V}}$
c_{id}	$4c_{\mathrm{LD2V}} + 2c_{\mathrm{D2D}}$
c_{lc}	$4c_{\mathrm{LD2V}}$
c_{rc1}	$2c_{\mathrm{LD2V}} + 2c_{\mathrm{RD2V}}$
c_{rc2}	$4c_{\mathrm{LD2V}} + 2c_{\mathrm{RD2V}}$
c_{rc3}	$3c_{\mathrm{LD2V}} + 3c_{\mathrm{RD2V}}$
$c_{\mathrm{rc1'}}$	$3c_{\mathrm{LD2V}} + c_{\mathrm{D2D}} + c_{\mathrm{RD2V}}$
$c_{\mathrm{rc2'}}$	$3c_{\mathrm{LD2V}} + 2c_{\mathrm{D2D}} + c_{\mathrm{RD2V}}$

Table V. Cost sets.

Set	c_{LD2V}	c_{D2D}	c_{RD2V}	$c_{\mathrm{V2G}} = c_{\mathrm{G2SH}} = c_{\mathrm{SH2V}}$
1	1	1.5	1.5	1.5
2	1	2.5	2.5	2.5
3	1	5	5	5

The total cost of the proposed location management scheme is measured as the ratio of the total cost per unit time for the proposed scheme to that of the IS-41 standard, $C/C_{\mathrm{IS\text{-}41}}$. We can express the total cost of IS-41 standard, $C_{\mathrm{IS\text{-}41}}$, as

$$C_{\mathrm{IS\text{-}41}} = (\lambda_{\mathrm{m}} + \lambda_{\mathrm{c}})(3c_{\mathrm{SH2V}} + c_{\mathrm{V2G}} + c_{\mathrm{G2SH}}) \quad (16)$$

where λ_{c} is the total call arrival rate to the target MT. An extra visit to the GTT STP costs c_{V2G} and c_{G2SH}. In the registration operation, we need c_{SH2V} to send an acknowledgment message to the new VLR, and $2c_{\mathrm{SH2V}}$ are to inform the old VLR of canceling the obsolete record. As for the call delivery operation, $3c_{\mathrm{SH2V}}$ is needed to complete the necessary signaling. In our analysis, we assume that there are four DHLRs ($D = 4$) deployed and each DHLR serves an average of one hundred thousand of users ($N = 100\,000$), while the data length ratio, l/L, in Equations (11) and (12) is set to be 0.1. The comparisons of numerical results are discussed as follows:

Figure 11 shows the relative cost for the direct and indirect remote pointing schemes under the batch-update strategy. The vertical axis represents the relative cost $C/C_{\mathrm{IS\text{-}41}}$ and the horizontal axis is the ratio of the call arrival rate to the mobility rate (CMR), $\lambda_{\mathrm{c}}/\lambda_{\mathrm{m}}$, varies from 0.01 to 100. The numerical result for the immediate update strategy is also provided. As mentioned before, the immediate update strategy needs a lot of signaling to complete the multiple HLR updates. When users move frequently, it is not worthy to adopt the distributed HLRs architecture if the immediate update strategy is used. Figure 11 also illustrates that the batch-update strategy reduces the update overhead when the CMR is low. Since the *indirect* pointing scheme only needs to update the remote pointer in each DHLR due to the target MT's inter-DA movement, this further results in significant cost savings for the indirect pointing scheme. However, when the CMR is high, the call delivery cost dominates. The extra cost for indirect pointing obviously degrades the performance. Therefore, the best policy is to adopt the direct remote pointing scheme for the MT with a high CMR and adopt the other one for the MT with a low CMR. It should be noted that if each signaling to a VLR is always routed through the STP to which the local DHLR is connected, the indirect remote pointing scheme is undoubtedly the best choice.

Figure 12(a) and (b) show the effect of the inter-DA movement probability, q, on the direct and indirect remote pointing schemes, respectively. Because an inter-DA registration process needs more signals than an intra-DA registration one, the smaller the inter-DA movement probability, the lower the relative cost. In general, we can increase the size of the DA to obtain a lower inter-DA movement probability [17]. However, the enlarged DA lengthens the routing distance between the local DHLR and VLR, and thus the signaling cost of the local transmission also increases. Consequently, the relative cost is unavoidably increased.

From Equation (13), the total local call arrival rate for a target MT is determined by the summation term, $\sum_{i=1}^{D} \lambda_i LC_i$. The LC_i is further affected by the locality parameter, μ_i. A high value of μ_i implies that the target MT tends to move back to DA_i once it moves. Therefore, a high value of λ_i and a high probability of μ_i represent a high total local call

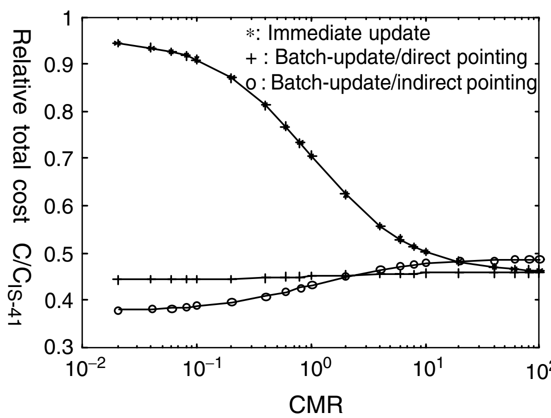


Fig. 11. The relative total costs for the batch-update and immediate update schemes with Cost Set 2.

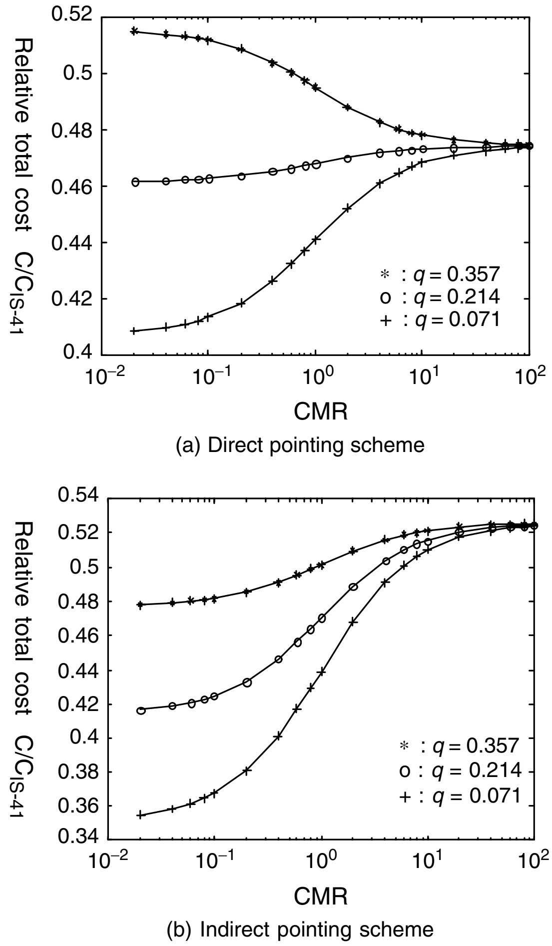


Fig. 12. Comparison of different inter-DA movement probabilities.

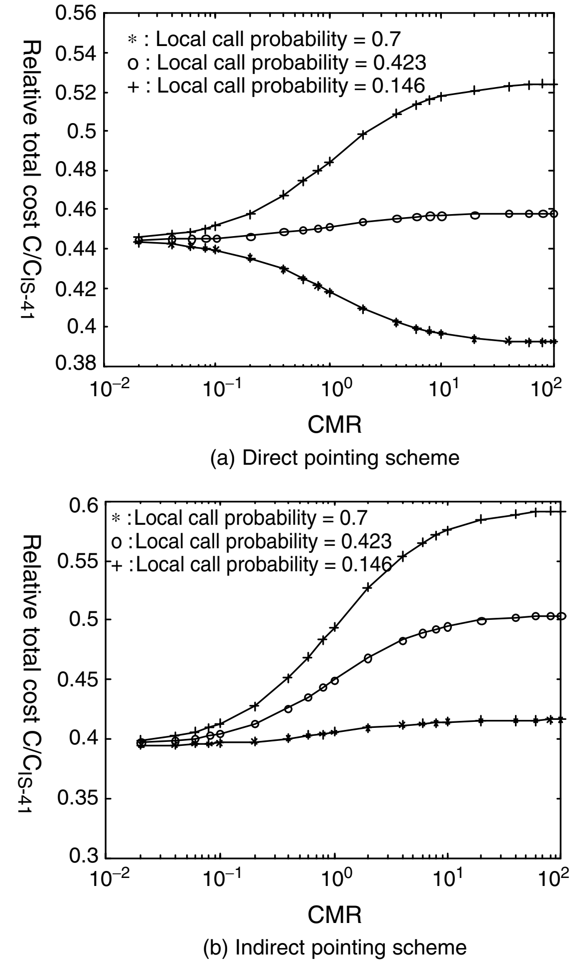


Fig. 13. Comparison of the costs among various local call arrival rates.

arrival rate. The higher the total local call arrival rate is, the less the call delivery cost will be. Figure 13(a) and (b) obviously shows this phenomenon.

Figure 14 shows the comparison of the improvements under different sets of the elementary costs as shown in Table V. We can expect that the more localized the processing of the network is, the lower the ratio of the local signaling cost to the remote signaling cost (LRCR), $c_{\mathrm{LD2V}}/c_{\mathrm{RD2V}}$, will be. Therefore, the performance improves proportionally as the LRCR decreases. More improvement can be achieved when both the LRCR and the CMR are low. This phenomenon implies that we can benefit more from the property of frequent movements in a local area by our approach compared with the single HLR one.

Finally, Figure 15 shows the comparison of the call delivery performance. As we analyzed before, the batch-update strategy for the direct remote pointing scheme does not incur significant impact on signaling. By using our approach, a call delivery can be completed in the direct remote pointing scheme as well as in the immediate update scheme.

5. Conclusion

In this paper, we introduce a distributed HLRs architecture. Under this architecture, each MT only makes location registration to its serving DHLR, and each DHLR is responsible for updating the MT's location change in other DHLRs. A batch-update strategy is proposed to significantly reduce the heavy traffic that the immediate update method in most replication systems could generate. In our batch-update strategy, the remote pointers (pointing to the MTs residing in other DHLRs) are always updated in a batch fashion only when an inter-DA registration process occurs. So the signaling overhead for updating the remote pointers

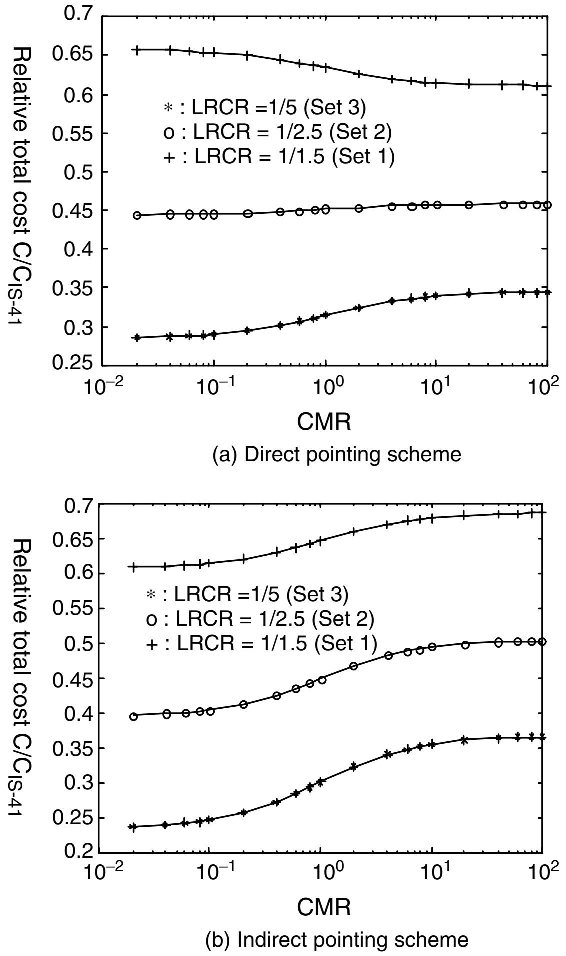


Fig. 14. Comparison of the improvements among different sets of the elementary costs.

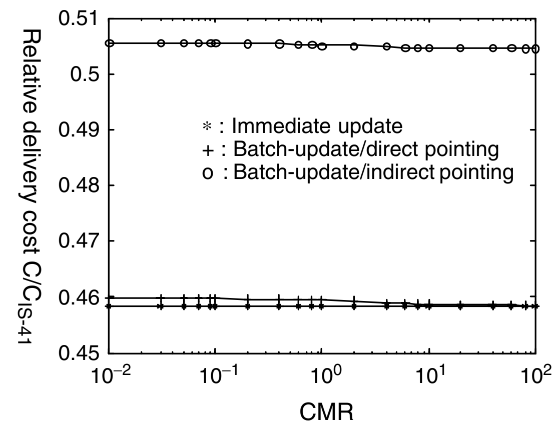


Fig. 15. Comparison of the call performance.

can be avoided. Our analysis shows that most of the location changes of the MTs will be updated in time and that the delivery of the remote calls could be performed as well as the replication systems do. In PCS networks, distributed mobility schemes seem to be a trend to reduce the signaling overhead caused by the increasing number of subscribers. Our approach makes the multiple HLRs architecture feasible.

We also analyze the properties of both the direct and indirect remote pointing schemes used in the DHLR network. There exists a trade-off between them. However, on the basis of the MT's mobility and call arrival patterns, we can dynamically switch the pointing method between these two schemes to obtain a better performance. Therefore, developing an adaptive algorithm to get an optimal result will be our future work.

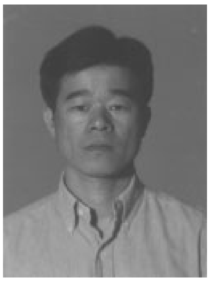
References

1. EIA/TIA. Cellular Radio-Telecommunications Intersystem Operations. EIA/TIA Tech. Rep. IS-41 Revision D, April 2002.
2. Mouly M, Pautet M-B. *The GSM system for Communications*. Palaiseau: France, 1992.
3. Lin Y-B, Tsai W-N. Location tracking with distributed HLRs and pointer forwarding. *IEEE Transactions on Vehicular Technology* 1998; **47**(1): 58–64.
4. Jeong DG, Jeong WS. Probabilistic location update for advanced cellular mobile networks. *IEEE Communications Letters* 1998; **2**(1): 8–10.
5. Akyildiz IF, Ho JSM. Dynamic mobile user location update for wireless PCS networks. *ACM-Baltzer Wireless Networks* 1995; **1**(2): 187–196.
6. Ho JSM, Akyildiz IF. Mobile user location update and paging under delay constraints. *ACM-Baltzer Wireless Networks* 1995; **1**(4): 413–425.
7. Akyildiz IF, Ho JSM, Lin Y-B. Movement-based location update and selective paging for PCS networks. *IEEE/ACM Transactions on Network* 1996; **4**(4): 629–638.
8. Pollini GP, Tabbane S. The intelligent network signaling and switching cost of an alternate location strategy using memory. In *Proc. IEEE VTC*, 1993; pp. 931–934.
9. Xie H, Tabbane S, Goodman JD. Dynamic location area management and performance analysis. In *Proc. IEEE VTC*, 1993; pp. 536–539.
10. Jain R, Lin Y-B, Lo CN, Mohan S. A caching strategy to reduce the network impact of PCS. *IEEE Journal on Selected Areas in Communications* 1994; **12**(8): 1434–1445.
11. Lin Y-B. Determining the user location for personal communications networks. *IEEE Transactions on Vehicular Technology* 1994; **43**(3): 466–473.
12. Shivakumar N, Widom J. User profile replication for faster location lookup in mobile environments. In *Proc. ACM MOBICOM '95*, November 1995; pp. 161–169.
13. Yi S-Y, Shin H. A new locating scheme for mobile objects based on mobile characteristics and replication of location information. In *Proceedings of the 2nd International Mobile Computing Conference*, March 1996; pp. 123–130.
14. Jain R, Lin Y-B. Performance modeling of an auxiliary user location strategy in a PCS network. *ACM-Baltzer Wireless Networks* 1995; **1**(2): 197–210.
15. Sue K-L, Tseng C-C. One-step pointer forwarding strategy for location tracking in distributed HLR environment. *IEEE Journal on Selected Areas in Communications* 1997; **15**(8): 1455–1466.

16. Lin Y-B. Reducing location update cost in a PCS network. *IEEE/ACM Transactions on Network* 1997; **5**(1): 25–33.
17. Ho JSM, Akyildiz IF. Dynamic hierarchical database architecture for location management in PCS networks. *IEEE/ACM Transactions on Network* 1997; **5**(5): 646–660.
18. Wang JZ. A fully distributed location registration strategy for universal personal communication systems. *IEEE Journal on Selected Areas in Communications* 1993; **11**(6): 850–860.
19. Harrison PG, Patel NM. *Performance Modeling of Communication Networks and Computer Architectures*. Addison-Wesley: Reading, MA, 1993.

Authors' Biographies

Hang-Wen Hwang received the M.S. degree in Control Engineering from National Chiao-Tung University, Hsin-Chu, Taiwan, in 1987. He is currently a specialist in the Chung Shan Institute of Science and Technology, Taiwan, and working toward Ph.D. degree with the Department of Computer Science and Information Engineering at National Chiao-Tung University. His research interests include Personal Communication Systems and Mobile Computing.



Chien-Chao Tseng is currently a professor in the Department of Computer Science and Information Engineering at National Chiao-Tung University, Hsin-Chu, Taiwan. He received his B.S. degree in Industrial Engineering from National Tsing-Hua University, Hsin-Chu, Taiwan, in 1981; M.S. and Ph.D. degrees in Computer Science from the Southern Methodist University, Dallas, Texas, USA, in 1986 and 1989, respectively. His research interests include Mobile Computing, and Wireless Internet.

Ming-Feng Chang received the Ph.D. degree in computer science from the University of Illinois at Urbana-Champaign in 1991. He is currently a Professor in the Department of Computer Science and Information Engineering, Chiao-Tung University, Taiwan, R.O.C. His research interests include Internet Communication, Mobile Computing and VLSI system design. Current research projects include VoIP for wireless networks, internetworking of VOIP protocols and cache model for WAP applications.