Evaluation of Distributed and Replicated HLR for Location Management in PCS Network

GUAN-CHI CHEN AND SUH-YIN LEE
Department of Computer Science and Information Engineering
National Chiao Tung University
Hsinchu, 300 Taiwan
E-mail: {gcchen, sylee}@csie.nctu.edu.tw

Personal Communication Service (PCS) network is the integration of cellular Network and conventional wired network. To support user mobility, the locations of the mobile stations (MSs) should be constantly tracked using a database. The widespread deployment of PCS will lead to a tremendous increase in the number of updates and queries to the location database. The centralized, stand-alone Home Location Register (HLR) in GSM can be a potential bottleneck. In case of the location database failure, incoming calls may be lost. According to the analysis of the load in HLR, we suggest replicating the HLR in order to increase its capacity and reliability. In this paper we evaluate the effects of the number of replicas in terms of query response time, misrouting probability of incoming calls and average cost per query. Another issue incurred in the replicated databases is data consistency. A concurrency control protocol called preemptive Read-One-Write-All (PROWA) protocol is also proposed.

Keywords: mobile computing, location management, HLR, replicated databases, PCS

1. INTRODUCTION

To support user mobility, the PCS networks have to store and maintain location information of mobile stations (MSs) so that an incoming call can be delivered to the target MS. The operations on location information consist of location updates and location queries. An update occurs when an MS changes location. A query occurs when an MS needs to be located, e.g., to deliver an incoming call to this MS. The widespread deployment of PCS will lead to a tremendous increase in the number of updates and queries to the location database. Thus, a key challenge to location management is to develop an efficient database architecture so that the location data can be readily available for signaling such as call setup and routing.

The current approach to support user mobility requires a two-level database [1]. This architecture consists of a home location database (or Home Location Register, HLR) and a visitor location database (or Visitor Location Register, VLR). This HLR-VLR architecture, has been established as the de facto, and is widely used in industry, such as Global System for Mobile Telecommunication (GSM) for Europe and the IS-41 recommendations for North America. We will describe location management in GSM and IS-41 in more detail in section 2. In such systems, HLR is centralized and stand-alone

Received September 28, 2000; revised April 18, 2001; accepted January 25, 2002.
Communicated by Ming-Syan Chen.

in the network. Some queries and updates have to travel long distances when the incoming calls or the MSs are far away from the HLR. Another drawback of conventional HLR-VRL architecture is that HLR tends to be the bottleneck. A typical GSM mobile customer traffic profile is [2]

- 9 – 12 min/day usage
- 60/40 split (60% of traffic is from wireless network to fixed network)
- average call duration approximately 50 sec.
- switching the phone on/off approximately 4 times per day

HLR implementations are commercially available to support approximately 300,000 customers. Each user with the above mentioned traffic profile will submit approximately 20 HLR operations per day (location updating, routing, authentication, network attachment). For a GSM with 300,000 subscribers, the load on HLR will thus be approximately 6,000,000 operations per day. From experience with SONOFON GSM, 12-13% of the operations are during busy hours, i.e. about 800,000 queries per hour (or 220 transaction per second). The peak value may be 50% higher [3]. Such a heavy load cannot be supported by standard relational databases even on the most powerful processing equipment available today. As a consequence, HLR has the potential to be a bottleneck and cannot guarantee the quality of service (e.g., call setup time).

Another class of mobility databases is based on the tree structures [4-6]. In such schemes, the user is assumed to be located at one of the leave nodes of a tree network. Some internal nodes in this tree may contain a database which maintains a list of all users currently located at nodes in its associated subtree. For a given user, a pointer is maintained at each database in the path from the root to the node in which the user is located. The lowest database in the path points to the cell in which the user is located. Any other database on the path points to the databases in the lower level on the path. When an MS moves to another cell, only the databases in the path between the two cells need to be updated. In this architecture, a call setup is by search upward to a common node which contains a location database and then follows the pointers downward to get the address of the cell where the target MS is located. This may involve many database queries when the path between the incoming call and the target MS is long. Therefore, the call setup time is relatively longer when compared with HLR-VLR architecture.

In addition to proposing new database architectures, some researchers try to reduce the location registration cost by dynamically adapting the paging area. These can be classified into three categories: distance-based [7, 8], movement-based [9, 10] and time-based [11]. The basic idea behind these schemes is that the user does not make any location update unless it has exceeded the boundary of a dynamically determined paging area. These schemes can efficiently reduce the number of location registrations. However, since the system has no exact location information for the mobile hosts, there will be some paging overhead and the call setup time will be longer than that in GSM and IS-41.

To guarantee the quality of service, we suggest adopting the HLR-VLR architecture. However, the problem of the overloaded HLR must be taken into account. From an analysis of the load source in HLR, we suggest replicating the HLR to avoid the performance bottleneck in HLR. A model to evaluate the effect of the number of replicas is developed in this paper. The performance metrics considered in this model are the average call setup time, the misrouting probability of incoming calls, and average cost per query. Another issue incurred in the replicated databases is data consistency. A modified Read-One-Write-All (ROWA) protocol referred to as Preemptive ROWA (PROWA) is also proposed.

The rest of this paper is organized as follows. In next section, we briefly describe the location management in GSM and IS-41. The effect of the number of HLR copies is evaluated in section 3. Experimental results and conclusions are presented in sections 4 and 5, respectively.

2. LOCATION MANAGEMENT IN GSM AND IS-41

In order to understand the location management in PCS network, we briefly describe the HLR-VLR architecture used in GSM and IS-41. The procedures of location update and call delivery are also described.

2.1 HLR-VLR Architecture

The communication area covered by a base station is called a cell. The location of an MS is thus the address of the cell in which this MS is located. The major task of mobility management is to update the location of an MS when it moves from one cell to another. The update procedure is referred to as registration. We use GSM as an example to illustrate the HLR-VLR architecture and the information stored in HLR and VLR. Fig. 1 shows the network architecture of mobility management in GSM [1].

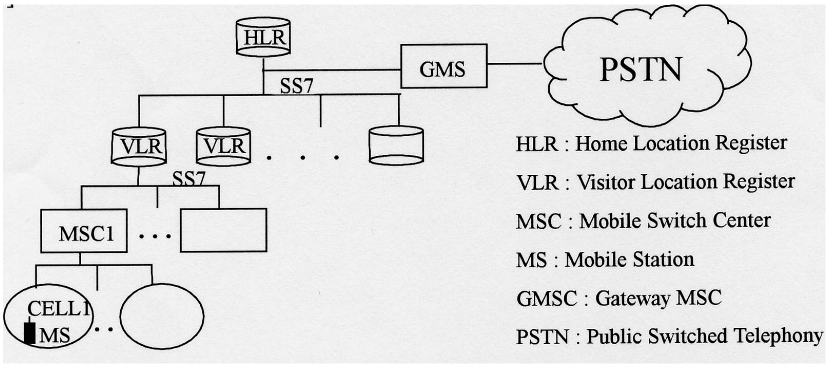
HLR
SS7
GMS
PSTN
VLR
VLR
SS7
MSC1
CELL1
MS
HLR : Home Location Register
VLR : Visitor Location Register
MSC : Mobile Switch Center
MS : Mobile Station
GMSC : Gateway MSC
PSTN : Public Switched Telephony

Fig. 1. The network architecture in GSM.

In this architecture, the base station in each cell is connected to a Mobile Switch Center (MSC), which is a telephone switch tailored for PCS network. Thus an MSC covers several cells. One (or several) MSC is connected to a VLR and exchanges the location information with the corresponding VLR through Signaling System Number 7 (SS7) network [13]. Similarly, the VLRs communicate with the HLR to exchange the location information using SS7 messages.

The location management in PCS network is achieved by the cooperation of HLR and VLR. The location information of an MS stored in HLR are the ISDN number (address) of the VLR visited by the MS and the ISDN number (address) of the MSC connected by the cell in which the MS is located. Similarly, the location data stored in VLR are the address of the corresponding MSC and the cell identity.

2.2 Registration and Call Delivery

In GSM the location update is referred to as registration and occurs when an MS moves from one cell to another. According to the connection relationship between these two cells, the location update occurs in the following three situations [1].

Case 1. Inter-Cell movement: These two cells are connected to the same MSC. Because the address of the cell in which the MS is located is stored in VLR, the registration message should be sent to VLR to update the cell identity. However, since the address of MSC and VLR do not change in this situation, no HLR update need be executed.

Case 2. Inter-MSC movement: These two cells are connected to different MSCs of the same VLR. Because the VLR stores the address of cells and MSC, the registration message should be sent to VLR to update the record to indicate the current location. In addition, the MSC address is also stored in HLR, the registration message should also be sent to the HLR to update the field of MSC address of this record.

Case 3. Inter-VLR movement: These two cells are connected to different MSCs and different VLRs. Since the MS moves to a different VLR, this VLR does not have a VLR record of the MS. The VLR creates a VLR record for the MS, and sends a registration message to update the HLR as described in Case 2 except that the VLR field is also updated. The record in the old VLR is deleted when the de-registration message from the new VLR arrived or when the old BS does not receive the response from the MS for a pre-defined time period.

To deliver an incoming call to the target MS, routing information (MSC address and cell address) must be obtained. When an incoming call arrives (from mobile or conventional PSTN phone), it is routed to a Gateway MSC (GMSC). The GMSC submits a query to the HLR to get the location of the target MS. When a query arrives, the HLR searches the database to get the address of the VLR in which the MS record is stored. After getting the VLR address, the HLR submits a query to the corresponding VLR to get the MSC address and cell address and then returns them to the Gateway MSC. Based on the MSC address and cell address, an incoming call can be correctly delivered to the target MS.

From the above description, we can find that when an incoming call arrives, both the HLR and the corresponding VLR will be queried. However, when an MS changes its location, the HLR will be updated only when Case 2 and Case 3 occur. Because an MSC covers several cells, the frequency of Case 1 occurring is far more than that of Case 2 and Case 3. From this observation, we have to take different considerations in the design of HLR and VLR.

3. REPLICATED HLR

The major tasks in HLR are

- location updating
- routing (directing incoming calls to the target MSs)
- authenticating (validating an MS)
- supplementary services

Except for location updating, most processing involves only a query operation in HLR. In other words, in most situations, the frequency of an HLR being queried is far greater than that of an HLR being updated. On the basis of this observation, we propose replicating the HLR (represented as RHLR) to resolve the problem of HLR being the performance bottleneck and to increase the reliability of HLR.

The most important issue in HLR replication is how many copies an HLR must be replicated and where to place them. Replicating an HLR and distributing the replicas in the network can increase the reliability of HLR and reduce the communication cost as well as call setup time. On the other hand, replicating an HLR will increase the cost to setup and maintain database copies. Also, the database load will increase due to the extra overhead of update operations, making the contents of the databases consistent. In this section, we evaluate the effects of the number of replicas in terms of query response time, misrouting probability of incoming calls, and average cost per query. Before we evaluate the effect of the number of replicas, we first describe the concurrency control protocol used in this model.

3.1 Concurrency Control

An important issue in replicated databases is to guarantee the consistency of data among all the replicates. If the location data in the replicas of HLR are not consistent, some queries may get obsolete data and cause the incoming calls to be misrouted or lost. Many concurrency control protocols have been proposed to guarantee the serializability in replicated databases [12]. However, the characteristics of location data stored in HLR are different from those of the data stored in conventional databases. First, the query and update in location management in PCS network often involve only one record. Second, the time to execute transactions in location management is different from that in conventional database service. In conventional database services, e.g., the banking system, the event in the real world can happen only when the corresponding transaction commits. For example, when one wants to withdrawal money from an account, one can get the money only when the withdraw transaction commits. If the withdrawal transaction fails, he/she must resubmit it or he/she cannot get the money. However, in PCS system, the update procedure is executed after an MS moves to a different cell. No matter whether or not an update transaction can commit, the movement of the MS has occurred. Furthermore, the MS has been in the new cell even during the processing of the update procedure. Some researchers proposed concurrency control strategies which guarantee the serializability of the execution of query and update operation in mobility databases. Some researchers proposed concurrency control strategies which guarantee the serializability of the execution of query and update operation in mobility databases. For example, in 1997 K. K. Leung proposed an update algorithm called Primary-Writer Protocol (PWP) using a multi-version approach [13]. In PWP each record is replicated and distributed to many sites, and one of the replicas is assigned to be the primary site (PS) of this record. Queries are routed to the PS or other replicas according to the call distribution algorithm. An update must be sent to the primary site, and then, if committed, sent to other replicas. PWP keep multiple versions for a record. A sequence of update operations is executed in the replicas according to the order of their version numbers in the PS. A query may need to read the old version of a record to prevent violating serializability. However, since a mobile unit has only one current location, it is meaningless to keep multiple versions of the location information and let a query read old versions of location data. Although the execution order of query and update operation in PWP guarantees serializability, the MS has already been in a new cell. As a consequence, the query will get an obsolete location data and cause the incoming call to be misrouted or lost. In this paper we present the observation that the update procedure in a location database is executed after the movement of an MS. No matter whether or not an update transaction can commit, the movement of the MS has occurred. Therefore, it is meaningless to let a query read old versions of location data. From this observation, we suggest aborting the queries on a record if an update request on the same record arrives. This is what we called "preemptive".

The Read-One-Write-All (ROWA) protocol is employed in many replicated database systems to guarantee consistent content [12]. In ROWA protocol, a query is routed to one of the replicas according to the routing function, and an update message is sent to all replicas to guarantee the consistency. In order to reduce the probability of an incoming call being misrouted or lost, we modify the Read-One-Write-All (ROWA) protocol and propose the Preemptive Read-One-Write-All (PROWA) protocol. The main idea of PROWA is based on the above observation. When an update operation arrives at the database, if there is a query operation executing on the same record, concurrency control agent will abort the query operation and restart it after a certain time (long enough to complete an update operation) in order to get the correct location data. fter the description of the concurrency control protocol, we can now present the proposed model to evaluate the effect of the number of replicas.

3.2 System Model

We partition the service area of the PCS network into several service regions (SRs). Each SR is covered by a GMSC. This partition reflects the situation that a query submitted to HLR is initiated from a GMSC; Table 1 lists the notations and symbols used. The loads incurred by authentication and supplementary service are not discussed here. The capacity of a database is defined as the number of queries that a database can process per unit time.

The main idea of the RHLR scheme is to replicate the HLR and distribute the replicas to the SRs. However, how many replicas an HLR should be replicated and where to locate the replicas are the main issues of the RHLR scheme. In the following sections, we describe a linear programming model to decide the placement of the replicas.

Table 1. Definitions of symbols used.

Symbol	Definition
S_i	a service region
n	number of service regions
λu_i	arrival rate of location updates submitted from S_i to the HLR
λq_i	arrival rate of queries submitted from S_i to the HLR
C_{ij}	communication cost to transfer an SS7 message between S_i and S_j
M	average cost to setup and maintain a database per unit time
P	capacity of a database (number of operations the database can execute per second)
B	bandwidth of a signaling link (bits per second)
α	load to process a query operation in HLR
β	load to process an update operation in HLR
σ	cost to process a query operation in HLR
ω	cost to process an update operation in HLR
$\wp$	length of an SS7 message for query and update (bytes)
K	number of replicas of an HLR
q	query-to-update ratio
D_i	number of HLR replicas located in region S_i
R_{ij}	ratio of queries submitted from S_i to the database located in S_j over all queries submitted from S_i

In addition, the relationships between the number of replicas and system performance (query response time, average cost per query and the probability of an incoming call is misrouted) are also discussed in this paper. The systems engineer can determine the number of replicas according to the quality-of-service the system should provide and the available budget.

3.3 Query Response Time

The time to set up an incoming call is affected by the time needed to get the location information of the target MS (by submitting a query to the HLR). The time to complete a query can be divided into three parts: the time to transfer SS7 messages between the GMSC and the HLR, the time to complete a query in the HLR and the time to get the MSC address and cell address from VLR. Since our main objective here is to gain a general understanding of the relative performance with respect to the number of replicas, we make the following assumptions. (1) Since the number of queries and updates submitted from an SR, S_i, is large enough, they can be approximated by a Poisson process with mean λq_i and λu_i respectively [1]. (2) The databases and signaling links are modeled as a single-server queue. Queries and updates are processed and transmitted on a First-Come-First-Serve (FCFS) basis. In order to maximize the utilization of each replica, we balance the load among all replicas. This can be achieved by defining a routing function for queries. We will discuss how to determine the routing function for queries in later discussion. The time to transfer an SS7 message between GMSC and HLR consists of message delay and propagation delay. When an HLR is replicated K times, the load in each replica is $(\alpha\sum_{i=1}^{n}\lambda_{qi})/K+\beta\sum_{i=1}^{n}\lambda_{ui}$. According to Little's theory [14], the message delay is

$$T_m=\frac{1}{B/(8\rho)-(\sum_{i=1}^{n}\lambda q_i)/K-\sum_{i=1}^{n}\lambda u_i}. \tag{1}$$

In Eq. (1), B is the bandwidth of the link between two adjacent SRs. Similarly, if there are K replicas of an HLR, the time to process a query in HLR is

$$Td=\frac{1}{P-(\alpha\sum_{i=1}^{n}\lambda q_i)/K-\beta\sum_{i=1}^{n}\lambda u_i}. \tag{2}$$

The average propagation delay is affected by the placement of the K replicas. The optimal value of the average propagation delay can be determined by linear programming. We describe the model to determine the optimal placement of the replicas as follows:

(A) Objective Function for Minimizing the Propagation Delay

Our objective is to minimize the average propagation delay of a query. We use R_{ij} to represent the ratio of queries which are submitted by S_i to S_j over all queries submitted by S_i. F_{ij} is the propagation delay of the link between S_i and S_j. The average propagation delay of a query is thus

$$Z_2=\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda q_i R_{ij}F_{ij}.$$

(B) Constraints

(i) Query Satisfaction Constraint

To ensure that every query will have a defined route, the following constraint must be satisfied.

$$\sum_{j=1}^{n}R_{ij}=1 \qquad \text{for all } i,\quad 1\le i\le n$$

(ii) Load Balancing Constraint

The load is balanced among all replicas. If there are D_j replicas allocated into S_j, the following constraint must be satisfied.

$$(\sum_{i=1}^{n} \lambda_{qi} R_{ij}) = ((\sum_{i=1}^{n} \lambda_{qi})/K)D_j, \qquad \text{for all } j,\ 1 \le j \le n.$$

(iii) Constraint on Number of Replicas

The sum of the number of replicas in all SRs must equal *K*.

$$\sum_{i=1}^{n} D_i = K$$

The resulting model is:

$$\text{Min} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{qi} R_{ij} F_{ij}$$

$$\text{s.t.} \sum_{j=1}^{n} R_{ij} = 1,$$

$$(\sum_{i=1}^{n} \lambda_{qi} R_{ij}) = ((\sum_{i=1}^{n} \lambda_{qi})/K)D_j, \quad \text{for all } j,\ 1 \le j \le n.$$

We do not modify the architecture of VLR in GSM and IS-41. The time to get the MSC address and cell address from VLR does not change when an HLR is replicated. For simplicity, we assume that the time to get the MSC and cell addresses from VLR is constant and is referred to as T_{qv}. As discussed in section 2.2, the queried result must be sent back to the Gateway MSC to route the incoming call to the target MS. The average time to complete a query when the number of replicas equals *K* is

$$(\sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{qi} R_{ij} (2F_{ij}) / \sum_{i=1}^{n} \lambda_{qi}) + T_d + T_m + T_{qv} \tag{3}$$

If the number of replicas, *K*, equals 1, the system is reduced to GSM. The average completion time of a query is determined by the network delay as well as the time to complete the query in HLR. In GSM, because there is only one database, all queries are routed to the same region. However, when the HLR is replicated and distributed in the network, the query can be routed to the database in a nearby region. In addition, the time to complete a query in HLR will decrease as the number of replicas increases. It is obvious that the completion time of a query is shorter than that of GSM in most situations.

3.4 Cost of Replicated HLR

The cost of Replicated HLR consists of the communication cost, operating cost and the cost to set up and maintain a database. We analysis the cost of RHLR as follows:

(A) Communication cost

The PROWA protocol is implemented in the replicated HLR. All updates must be sent to all the database copies to make the content consistent. The total communication cost for update and query operation per unit time is

Update communication cost:

$$Z_1 = \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{ui} D_j C_{ij}$$

Query communication cost:

$$Z_2 = \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{qi} R_{ij} C_{ij}$$

(B) Database cost

The cost to setup and maintain a database per time unit is denoted by *M*. The total cost to set up and maintain *K* replicas is

$Z_3 = MK$

(C) Cost of query and update operations

In order to maintain data consistency, in replicated HLR, some extra update operations need to be executed when compared with the centralized, stand-alone HLR. In the centralized, stand-alone HLR architecture, an update procedure needs to only update one database. However, in replicated HLR, the update procedure needs to update all replicas. The cost to process query and update operations is

$$Z_4 = \sigma(\sum_{i=1}^{n} \lambda qi) + \omega(\sum_{i=1}^{n} \lambda ui)(\sum_{i=1}^{n} D_i).$$

The overall cost is the sum of all the costs and is equal to

$$Z = Z_1 + Z_2 + Z_3 + Z_4 \tag{4}$$

The value of Eq. (4) varies depending on database cost, operation cost, as well as the communication cost. With advances in computer systems, the cost to set up a database will steadily decrease. However, the cost to transfer a message in the network is not expected to rapidly decrease in the near future.

3.5 Probability of Incoming Calls Being Misrouted or Lost

To evaluate the probability of an incoming call being misrouted or lost due to obsolete data from HLR, we define the vulnerable period to be the time interval during which, if a query completes, it may get an obsolete location data and thus cause a call to be misrouted or lost. We use the misrouting probability of incoming calls to represent the probability of an incoming call being misrouted or lost due to obsolete data from HLR. Fig. 2 illustrates the vulnerable period.

In Fig. 2, an MS movement of Case 2 or Case 3 (defined in section 2.1, that is, inter-MSC movement or inter-VLR movement) occurs (in S_i when an incoming call arrives in S_k). S_j is the SR with an HLR replica to which a query in S_k is routed. In PROWA, if a query is executed on the same record when an update arrives, since a location update arrived means that corresponding MS has already moved, the concurrency control agent will abort this query and restart it after a certain period of time. However, even in PROWA protocol, there will be a chance that an incoming call is misrouted or lost. An incoming call will be misrouted or lost if it receives the cell address; S_i in Fig. 2, after the MS's movement or the query arrives in S_j before the corresponding update arrives. This time interval is the vulnerable period as illustrated in Fig. 2.

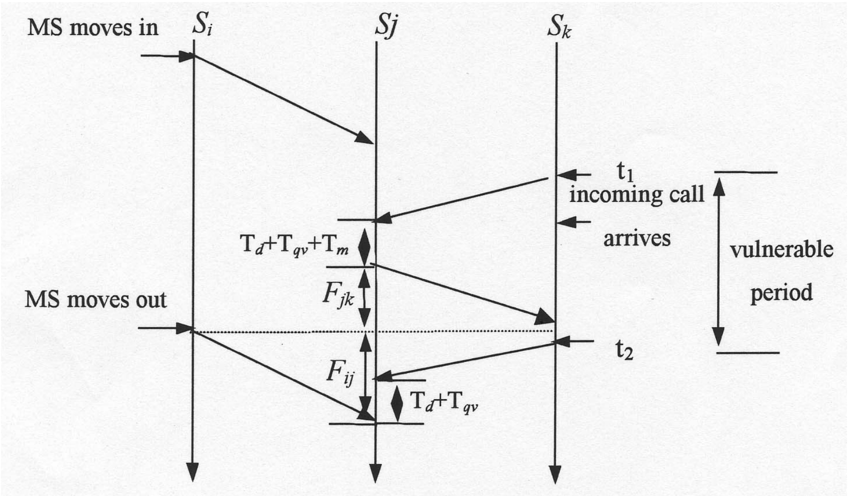

Fig. 2. Vulnerable period for incoming calls being misrouted or lost.

Now we evaluate the probability that an incoming call is misrouted or lost. Assume that the HLR update of an HLR is Possion distributed with mean μ. The average time interval between two updates is $1/\mu$. The probability of an incoming call to this MS being misrouted or lost is the vulnerable period over the average time interval between two updates. As shown in Fig. 2, the vulnerable period is ($t_2 - t_1$). Therefore, the probability of an incoming call to this MS being misrouted or lost is $(F_{ij}+F_{kj}+T_m)/(1/\mu)$, where F_{ij} and F_{kj} are the propagation delays of S_i-S_j and S_j-S_k, respectively, and T_m is the message delay of HLR.

4. EXPERIMENTAL RESULTS

The signaling network used in the simulation is a 5×5 mesh network, which has 25 SRs. The queries and updates submitted to HLR are uniformly distributed in each SR. The communication cost between the mobile or conventional PSTN phone and the Gateway MSC in the same SR is normalized to 1. The communication cost between two SRs is set to the length of the shortest path between them. Each database can support 220 queries per second. Each query costs 1 to be processed in the database. The cost of an update processed in database is set to 1.1. For demonstration purposes, the average cost, M, to set up and maintain a database is set to 1,000 in the experiments. The transmission speed of a link in each direction is set to 128k bits per second, which is the current ISDN standardized transmission rate. The propagation delay in the same SR is set to 5 msec. The propagation delay between two SRs is in proportion to the length of the shortest path between them. The delay, T_{qv}, caused by getting the result from VLR is set to 20 msec.

4.1 Load-Related Experiments

In these experiments we evaluate the effects of the number of replicas, K, in various loads. We vary the number of queries submitted from a service region from 100 to 600. A large number of queries submitted from a service region may represent the load at peak times. Fig. 3 shows the average completion time of a query. Let q represents the query-to-update ratio. In load-related experiments, q is set to 5. We will discuss the effects of the query-to-update ratio in the following sections. In Fig. 3 and the following figures, K represents the result obtained from the analytic model and SK represents the result obtained from the simulation model.

From Fig. 3, as expected, we find that the average completion time of a query is shorter when a large number of replicas is used. However, it will be infinitely long if the load in each replica exceeds the capacity of the database. The numbers on top of each curve in Fig. 3 give the maximum load that can be supported. Since the results obtained from the simulation and the analytic model are quite close, for clarity of illustration, the lines for the simulation results are not drawn in the figures in the following.

In addition to reducing the average completion time of a query, replicating an HLR can also reduce the misrouting probability of incoming calls. According to the discussion in section 3.5, the more copies an HLR replicates, the lower the misrouting probability of incoming calls is. The experimental result is shown in Fig. 4, where $P_{misrouted}$ represents the misrouting probability of incoming calls.

Although increasing the number of replicas can reduce both the average completion time and the misrouting probability of incoming calls, the cost will increase when the number of replicas increases. He result of cost-related experiments is shown in Fig.5. Here the average cost is the total cost (communication cost, database cost and cost of queries and updates) divided by the number of incoming calls.

There is no significant difference among the average costs of different numbers of replicas. The reason is as follows. When the number of replicas increases, the cost caused by the updates also increases. However, the communication cost of a query will decrease if the query can be routed to a nearer site.

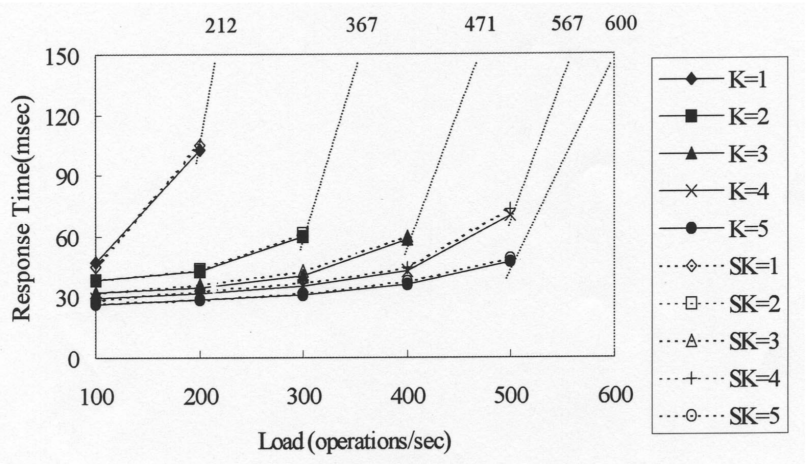

Fig. 3. Average query response time for query-to-update ratio *q* = 5.

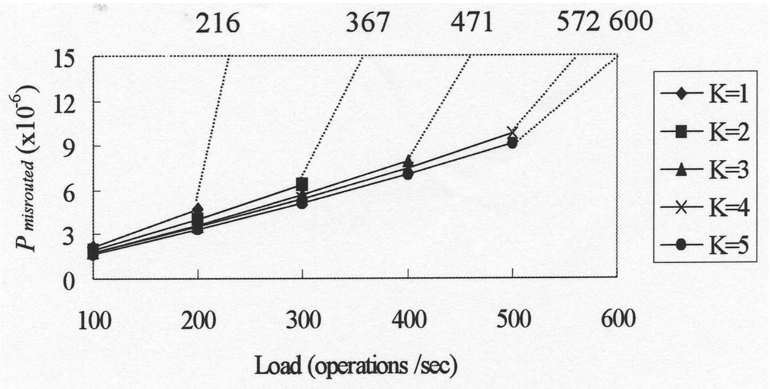

Fig. 4. Probability of call being misrouted for query-to-update ratio *q* = 5.

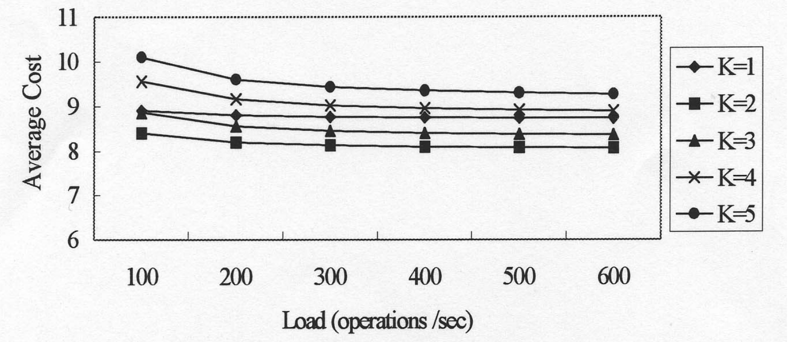

Fig. 5. Average costs for *q* = 5.

4.2 Access-Pattern-Related Experiments

The performance of RHLR will be affected by the ratio of updates over all operations. We use *q* to represent the query-to-update ratio which means that there are *q* queries arriving between two consecutive updates. If the total load is fixed, a large value of *q* means fewer updates and the load in each replica will be less. As a consequence, the time to process a query in HLR will be shorter. As we see in Figs. 6 and 7, when the value of *q* increases, the query response time and the misrouting probability of incoming calls decrease. However, the decrease in response time for a fix value of K is not significant. This is because that the load of an update in a database is set to 1.1. If the load to process an update were set to a larger value, the decrease would be greater.

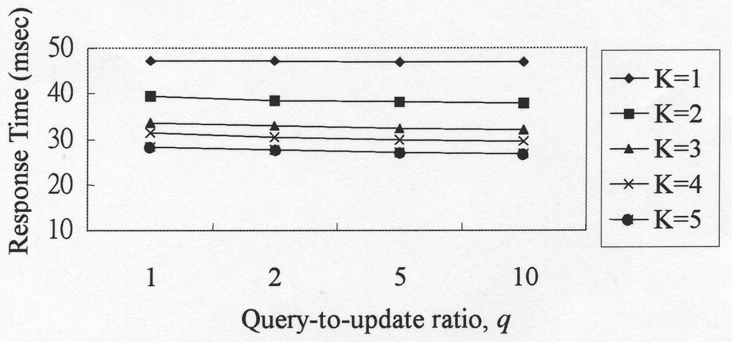

Fig. 6. Query response time for load =100.

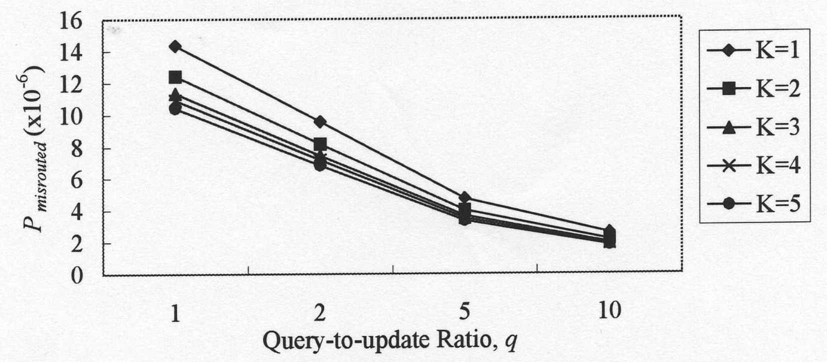

Fig. 7. Misrouting probability of incoming calls for load = 100.

Similar to the query response time, there are fewer updates when the query-to-update ratio is high, the cost resulting from updates is smaller than in low query-to-update ratios. The difference between the average cost with different numbers of replicas decreases as the value of *q* grows because there are fewer updates for larger *q*. Experimental results are shown in Fig. 8.

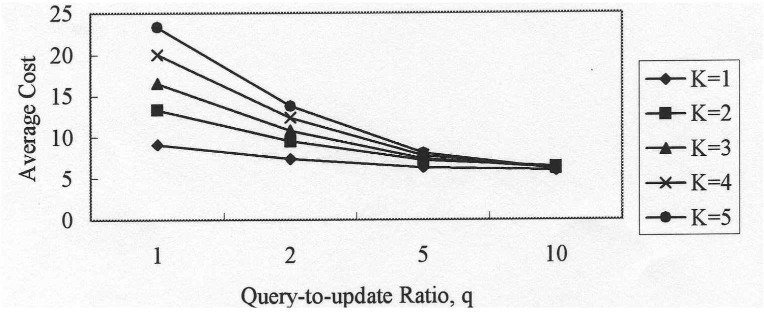

Fig. 8. Average cost for load = 100.

5. CONCLUSIONS

In this paper we present two observations in mobility management of PCS network and propose corresponding mechanisms to improve the system performance. The first observation is that the frequency with which an HLR is queried is greater than that of it being updated. Based on this observation, we propose replicating the HLR. The systems engineer can determine the number of replicas depending on the desired quality of and the available budget. The optimal placement of the replicas is determined using the linear programming method given in this paper. Another observation is that the update procedure in the location database is executed after the movement of an MS. From to this observation, we suggest aborting the queries in a record if an update arrives in the same record. This is what we called “preemptive”. The property “preemptive” can be applied to other concurrency control protocols. Since the read-one-write-all (ROWA) protocol is the most commonly used protocol in existing commercial replicated databases, we modify ROWA to be Preemptive ROWA (PROWA) in this paper. PROWA protocol can reduce the misrouting probability of incoming calls by aborting the query operations when an update operation arrives in the same record.

The cost of replicated HLR increases as the number of replicas increases. However, the call setup time is shorter and the misrouting probability of incoming calls is lower when we add more replicas. In other words, RHLR provides better quality of service than GSM by paying an extra cost. When the query-to-update ratio is high, the average cost per incoming call in RHLR is slightly higher than that in GSM and IS-41. However, the response time of RHLR is much shorter than in GSM and IS-41. In addition, RHLR can solve the problem that single HLR tends to be the performance bottleneck.

Updating all the replicated databases to maintain consistency of content incurs extra cost. In the future we will try to develop a new concurrency control protocol to reduce the cost of making the contents in the replicated HLR consistent.

REFERENCES

1. Y. -B. Lin and I. Chlamtac, *Wireless and Mobile Network Architectures,* Wiley, New York, 2001.
2. B. Gabelgaard, “The (GSM) HLR – advantages and challenges,” in *Proceedings of*

the 3rd IEEE International Conference on Universal Personal Communications, 1994, pp. 335-339.

3. S. Mohan and R. Jain, "Two user location strategies for personal communication services," *IEEE Personal Communications*, Vol. 1, 1994, pp. 42-50.
4. S. Dolev, D. K. Pradhan, and J. L. Welch, "Modified tree structure for location management in mobile environment," in *Proceedings of IEEE INFOCOM'95*, 1995, pp. 530-537.
5. A. Hac and C. Sheng, "User mobility management in PCS network: hierarchical databases and their placement," *in Proceedings of the 5th IEEE International Conference on Universal Personal Communications (ICUPC)*, 1996, pp. 847-851.
6. H. -C. Lin and S. L. Lee, "A presetting location strategy for personal communication using hierarchical location database," in *Proceedings of the 7th International Conference on Parallel and Distributed Systems*, 2000, pp. 349-354.
7. M. Verkama, "A simple implementation of distance-based location updates," in *Proceedings of the 6th IEEE International Conference on Universal Personal Communications (ICUPC)*, 1997, pp. 163-167.
8. J. H. Zhang and J. W. Mark, "A local VLR cluster approach to location management for PCS networks," in *Proceedings of 1999 IEEE Wireless Communications and Networking Conference*, 1999, pp. 311-315.
9. I. F. Akyildiz, J. S. M. Ho, and Y. -B. Lin, "Movement-based location update and selective paging for PCS networks," *IEEE/ACM Transactions on Networking*, Vol. 4, 1996, pp. 629-638.
10. V. Casares-Giner and J. Mataix-Oltra, "On movement-based mobility tracking strategy-an enhanced version," *IEEE Communication Letters*, Vol. 2, 1998, pp. 45-47.
11. S. H. Ryu, K. H. Lee, Y. Y. Oh, J. Y. Lee, and S. B. Lee, "Adaptive time-based location update scheme using fuzzy logic," in *Proceedings of 1999 International Conference on Consumer Electronics (ICCE)*, 1999, pp. 322-323.
12. A. K. Elmagarmid, *Database Transaction Models for Advanced Applications*, Morgan Kaufmann, 1992.
13. K. K. Leung, "An update algorithm for replicated signaling databases in wireless and advanced intelligent networks," *IEEE Transactions on Computers*, Vol. 46, 1997, pp. 362-367.
14. L. Kleinrock, *Queueing Systems*, Wiley, New York, 1975.

Guan-Chi Chen (陳冠棋) received the B.S. degree in Management Information System from National Central University, Taiwan, in 1993, the M.S. degree in Management Information System from National Taiwan Institute of Technology, Taiwan, in 1995, and the Ph.D. in Computer Science and Information Engineering from National Chiao Tung University, Taiwan, in 2001. Currently, he is a senior engineer at Ambit Microsystems Co., Taiwan. His research interests include wireless network, database systems, mobile computing and multimedia information systems.

Suh-Yin Lee (李素瑛) received the B.S. degree in electrical engineering from National Chiao Tung University, Taiwan, in 1972, the M.S. degree in computer science from University of Washington, U.S.A., in 1975, and the Ph.D. degree in computer science from Institute of Electronics, National Chiao Tung University. Her research interests include multimedia information system, computer network, mobile computing, data mining, and intelligent agent on web.