Performance Analysis of a Fault-Tolerant Scheme For Location Management of Mobile Hosts*

Saad Biaz and Nitin H. Vaidya

Department of Computer Science
Texas A&M University
College Station, TX 77843-3112, USA
E-mail: {saadb,vaidya}@cs.tamu.edu

Abstract. We consider the problem of location management of mobile users. The objective is to keep track of the mobile users in order to deliver calls. The IS-41 is a simple strategy for location management used in North America. This strategy is not efficient for some movement and call patterns. Many strategies have been proposed to address efficiently given movement and call patterns.

For mobile users who move frequently but receive relatively rare calls, a forwarding scheme has been shown to outperform the normal IS-41 location management scheme. But the forwarding scheme is more vulnerable to intermediate Visitor Location Registers (VLRs) failure than the IS-41 scheme. On the other hand, the call set-up with forwarding pointers may be costly, as a list of pointers must be traversed. We propose a simple variation to the forwarding scheme to address the fault tolerance weakness and the call set-up cost. The idea is to maintain two paths from the home location server to the last VLR with each path being of half the length of the normal forwarding path. We study and compare the performance of the normal forwarding scheme and our scheme.

Key Words:
Wireless environment – Personal Communications Services – Location Management – Fault-Tolerance to Visitor Location Registers– IS-41 – Forwarding Scheme

1 Introduction

In Personal Communications Services, the user can receive calls at any location. The main problem is how to deliver calls since the user is mobile. This can be solved if the mobile terminal reports all moves to a well-known and fixed database on the wired network. Whenever a call has to be delivered to this mobile terminal, a query is sent to this well-known database on the network. In

* Research reported is supported in part by Texas Advanced Technology Program grant 009741-052-C and National Science Foundation grant MIP-9423735.

this scheme, a considerable traffic must be sustained to keep track of the mobile terminal location.

The rapid growth in Personal Communications Services (PCS) incurs increasing loads on the databases and the network signaling resources. Simple data management strategies would not handle these loads efficiently[11, 10]. In location management, all studies try to reduce these loads by exploiting specific user patterns of mobility and call arrival. In [9], Jain and Lin proposed an auxiliary strategy to the North American IS-41 standard to reduce these loads. This strategy would be used only for users who change PCS registration areas frequently but receive calls relatively infrequently. The IS-41 location strategy is based on a two-tier system consisting of a HLR (Home Location Register) and Visitor Location Registers (VLR) [2]. Each VLR serves a predefined area. Whenever a Mobile Terminal (MT) enters a new area served by a different VLR, the HLR must be updated with the new VLR address such that future calls to the mobile terminal can be correctly directed.

For users who move frequently and receive rare calls, a "forwarding" scheme avoids the update of the Home Location Register by setting a pointer from the previous VLR to the new VLR. This strategy reduces the load on the signaling network between the VLR and the HLR. It also avoids HLR database update. The price paid is, firstly, in terms of memory and CPU processing at the VLR and secondly by increasing the call set-up time (all pointers from the HLR to the last VLR must be traversed). These observations and limitations were clearly identified by Jain and Lin in [9]. The authors did not, however, pay attention to the vulnerability of the scheme in comparison to IS-41. In IS-41, the call delivery success requires the HLR and the VLR to be failure free : the HLR provides the current VLR address and the VLR will return a temporary number used by the switching system to deliver the call to the mobile terminal. In the forwarding scheme, the call success requires also on intermediate VLRs maintaining the pointers between the HLR and the last VLR to be failure free.

To decrease this vulnerability, the idea here is simply to maintain two independent paths between the HLR and the last VLR. Besides being fault tolerant, another benefit of our scheme is to cut in half the forwarding list of pointers during the call set up time. The price paid is additional pointers, i.e more memory and CPU processing at the VLR and network signaling. However, the signaling is very localized between the VLRs.

This paper is organized as follows : the acronyms used in this paper are summarized in the appendix. Section 2 presents the related work. Section 3 will present a broad architecture of the Personal Communications Services. In Section 4, we will describe the basic operations for IS-41, the simple forwarding scheme and then our enhanced scheme. Section 5 presents the performance analysis without taking the fault-tolerance into account. In Section 6, we analyze if the fault-tolerance introduced can balance the overhead due to our scheme. Section 7 presents the conclusions.

2 Related Work

There is limited work on the issue of fault tolerance in mobile systems. The most relevant work to this paper is by Lin [8] in which he studies HLR database restoration after a database crash. His approach is based on periodic checkpointing of location databases. Lin derives the optimal interval for checkpointing. Our analysis uses the results derived by Jain and Lin for the forwarding scheme [9]. The fault tolerance schemes presented here are also related to work on fault tolerant linked lists (for instance, [7, 5]). However, we investigate this problem in the context of location management of mobile hosts.

In [14], Rangarajan and Dahbura consider an architecture where each base station maintains the location directory of all mobiles in the network. They present a fault-tolerant protocol to maintain this directory despite base station failures and mobile disconnections. In their scheme, all base stations are updated after each mobile moves from one base station to another. The main problem addressed is avoiding old updates from overwriting current information. Alagar et al. [4] consider the problem of base station failures. They present two schemes to tolerate such failures which are based on secondary base stations. The idea is to use replication (optimistic or pessimistic) on the secondary base stations. Pradhan et al. [13] consider recovery from failures of the mobile host. They present different schemes for message logging and checkpointing. Acharya et al. [1] consider the checkpointing of distributed applications. They consider specifically schemes to tolerate the failure of a mobile host or its disconnection. No work, to our knowledge, considers delivery of calls despite a VLR failure. We did further work on this subject in [6]. We considered two fault-tolerant schemes : the scheme presented in this paper and another based on the knowledge of the VLR neighbors. We analyzed and compared these two schemes.

3 The PCS Architecture

The PCS architecture consists of two networks [2] : the Public Switched Telephone Network (PSTN) and a signaling network. The PSTN is the traditional telephone system carrying voice while the signaling network is used only for management purposes using the SS7 (Signaling System no 7 [12]). The mobile terminals get access to the PSTN through base stations using wireless links. A set of geographically close cells define a Registration Area (RA). A Visitor Location Register (VLR) covers a group of RA's. A VLR keeps track of all mobile terminals in its RAs. Each mobile terminal is registered permanently with a Home Location Register(HLR) which keeps the user profile and whereabouts. The HLR, VLRs and the PSTN switching elements communicate through the SS7 signaling network to keep track of the mobile terminal position. Messages on the SS7 network are routed through the Signal Transfer Points (STP) which are installed in pairs for reliability purpose. The Mobile Switching Center (MSC) provides normal switching functions and plays an active role in registration and call delivery. Each VLR can be associated with one or more MSCs. For simplicity, we assume that each VLR is associated with one MSC and that they are collocated. The communication between MSCs and VLRs are through the SS7 signaling network. We omit some elements in this architecture which are beyond the scope of this paper (please refer to [2] for a good description). Figure (1) gives a schematic view of the architecture.

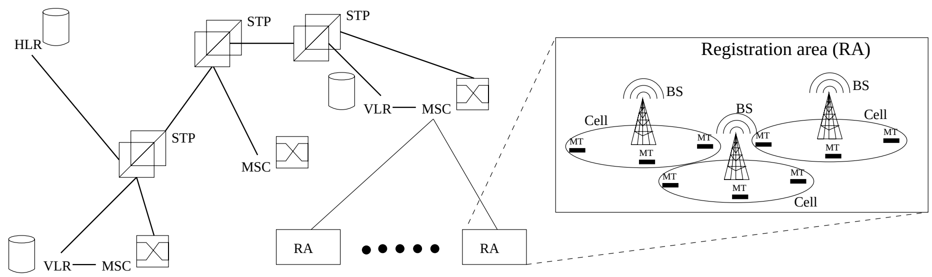

Fig. 1. Simplified PCS architecture

4 Location Management Schemes

Any location management scheme must provide two basic operations :

1: MOVE : registers the new location when a mobile terminal enters a new RA
2: FIND : provides the current location for a given user (to deliver a call)

We describe in the next subsections the IS-41 strategy, the per-user forwarding strategy and our enhanced per-user forwarding strategy. Jain and Lin showed in [9] that the forwarding scheme outperforms the basic IS-41 scheme when the user moves frequently and receives infrequent calls. In this paper, we consider forwarding schemes, therefore, we assume that the user moves frequently, compared to the rate at which it receives calls.

4.1 The IS-41 Location Management Scheme

The IS-41 strategy is based on the fact that, whenever a mobile terminal enters an RA covered by a different visitor location register VLR_{new}, the HLR is updated with the address of VLR_{new}. When a call is issued to the MT, the MT's HLR is contacted to obtain the address of the current VLR for the MT and the call is delivered.

We describe now in detail the $MOVE$ operation [9].

Let the mobile terminal MT be in a registration area (RA) covered by a visitor location register VLR_{old}. Whenever a mobile terminal enters an RA monitored by a visitor location register VLR_{new} different from VLR_{old}, the IS-41 basic procedure below is performed :

BasicMOVE() {

The mobile terminal (MT) sends a registration message to VLR_{new}
VLR_{new} sends a registration message to MT's HLR
HLR sends a registration cancellation to VLR_{old}
VLR_{old} sends a cancellation confirm message to HLR
HLR sends a registration acknowledgement message to VLR_{new}

}

Four messages involve the HLR. The load can be significant on the signaling network, specially if the HLR is "far away" from the VLRs. This MOVE operation implies also a database access and update on the HLR. These operations are just wasted if many moves occur without any call set-up.

When a call is issued for an MT, a FIND operation must be invoked. In normal IS-41, the HLR address can be directly extracted from the MT's number. Let a mobile terminal MT_a be calling a mobile terminal MT_b. The call will reach the mobile switching center (MSC_a) covering MT_a's registration area RA_a. MSC_a queries the visitor location register VLR_a covering RA_a. VLR_a determines if MT_b is in his coverage.

BasicFind() {

If MT_b is registered at the same visitor location register as MT_a, (i.e VLR_a) then RETURN
The Mobile Switching Center MSC_a queries MT_b's HLR, (i.e HLR_b)
HLR_b queries MT_b's current visitor location register VLR_b
VLR_b returns to HLR_b a temporary number to be used to reach MT_b
HLR_b forwards the temporary number to MSC_a

}

4.2 The Per-User Forwarding Strategy

This strategy proposed and analyzed by Jain and Lin in [9] modifies the basic IS-41 strategy in that it does not involve the HLR at each MOVE operation. Instead, at each move a pointer is established from the old visitor location register VLR_{old} to the new one VLR_{new}. This strategy clearly saves the HLR database update.

The price is paid when the mobile is called : a chain of VLRs must be queried before reaching the current VLR. To bound the call set up time, the chain length is limited to some value K. Let l be the current chain length. The procedures described by Jain and Lin [9] are as follows :

FwdMOVE() {

if (l < K-1) {
MT registers at VLR_{new} passing VLR_{old}'s address
VLR_{new} deregisters MT at VLR_{old}
VLR_{old} sends ACK and MT's profile to VLR_{new}
VLR_{old} sets a pointer pointing to VLR_{new}
l:=l+1
}

```
    else {
        BasicMOVE();
        l:= 0;
    }
}
```

For the FIND operation, the chain of VLRs will be queried until getting the current VLR covering the mobile terminal.

A mobile MT_a is calling the mobile MT_b. For other notation used here, please refer to the description of the BasicFIND described in the previous subsection (IS-41 scheme).

FwdFIND() {

if MT_b and MT_a are covered by the same VLR then RETURN
MSC_a queries MT_b's HLR (i.e HLR_b)
HLR_b responds with first VLR in chain (i.e VLR_0)
MSC_a queries VLR_0 about MT_b
while (MT_b's current VLR not reached)
  Queried VLR queries the VLR it points to.
l =0; { the path is compressed }
MT_b's current VLR VLR_b sends to HLR_b a temporary number to reach MT_b
HLR_b sends MT_b's temporary number to MSC_a
HLR_b updates database

}

Jain and Lin evaluated the overhead of their scheme for the FIND operation. But, if the scheme is used only for users moving frequently with infrequent incoming calls, the FIND overhead is rarely paid. However, the forwarding scheme is more vulnerable to VLRs failure as compared to the basic IS-41 scheme. In the basic scheme, the HLR and one VLR must be failure free for a call to succeed.

In the forwarding scheme, the HLR and a number of intermediate VLRs must be failure free. Figure 2 shows a situation where an MT has moved through the RAs of VLRs VLR_0 through VLR_3. If any VLR among VLR_0, VLR_1, VLR_2 fails, then the call could not be delivered. We propose an Enhanced Forwarding scheme to tolerate intermediate VLR failures and reduce the FIND operation cost. We describe this scheme in the next subsection.

4.3 The Enhanced Per-User Forwarding Strategy

The idea is to establish two independent paths from the MT's HLR to the current VLR. Figure 3 illustrates the idea. Let us assume that a mobile terminal MT traversed two registration areas covered by two different visitor location registers VLR_1 and VLR_2. Then MT enters a new RA covered by VLR_3 different from the two previous. Then MT will register with VLR_3 by passing the addresses of his two previous VLRs : VLR_1 and VLR_2. VLR_3 sends messages to VLR_2 and VLR_1 to set their pointers to VLR_3. The price paid is an additional VLR address

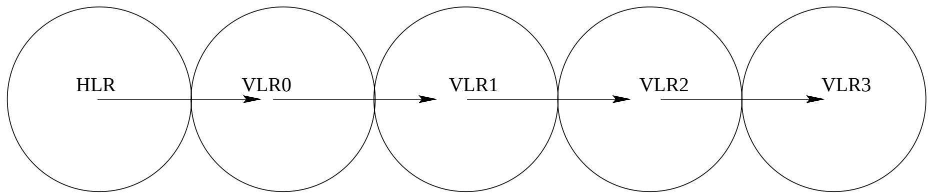

Fig. 2. Jain and Lin's forwarding scheme

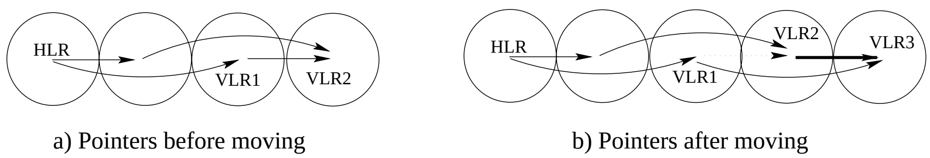

Fig. 3. MOVE operation and pointers update

on the HLR database for each MT entry, an increased load on the signaling network and more CPU processing on the VLRs.

We assume that the mobile terminal visited VLR_1 and VLR_2 sucessively and entered a new VLR's RA. Our enhanced MOVE procedure is as follows :

EnhanceFwdMOVE() {

if (l < K-1) { /* l is the chain's current length and K is the maximum length*/

MT registers at VLR_{new} passing VLR_1 and VLR_2's addresses
VLR_{new} deregisters MT at VLR_2
VLR_2 sets a pointer to VLR_{new}
VLR_{new} sends its address to VLR_1
VLR_1 sets a pointer to VLR_{new}
VLR_2 sends ACK and MT's profile to VLR_{new}
l:=l+1

}
else {

BasicMOVE();
l:= 0;

}

}

The HLR has two different pointers leading to the current VLR. The best path starts with the most recent pointer updated on HLR. To find the MT, the HLR will first follow this pointer. If the FIND operation fails, it will start again with the other pointer. The procedure presented by Jain and Lin [9] can be used for this purpose.

The performance analysis presented below has two objectives. First, we must evaluate how our scheme overhead will be balanced by the FIND reduction cost. Second, if there is still an overhead Δ_{cost}, how would it be balanced by our fault-tolerance scheme ? For the simple forwarding scheme, if one VLR fails, the only way to get the mobile terminal location is to page it. Paging is done by sending polling signals to the cells where the mobile terminal might be. These polling signals can be sent to all the cells in the PCS (too expensive) or selectively [3] based on the last known location. So, paging can be more or less expensive depending on how reliable we want it to be.

5 Performance Analysis without Fault-Tolerance

We will use similar notations as in [9], as follows :

- K is the largest length for the chain of pointers
- $1/\mu$ is the expected residual time of a user at an RA.
- f_m is the general density function of the residual time at an RA
- $f_m^*(s)$ is the Laplace transform of f_m
- λ is the call arrival rate (assumed a Poisson process)
- $p = \frac{\lambda}{\mu}$ is the call-to-mobility ratio
- M_{IS41} is the MOVE cost for basic IS-41 scheme
- S is the cost for setting a pointer between VLRs
- T is the cost for traversing a pointer
- M_f is the expected cost of the moves between two incoming calls with the simple forwarding scheme by Jain and Lin.
- F_f is the expected cost for one FIND operation. with the simple forwarding scheme by Jain and Lin.
- M'_f is the expected cost of the moves between two incoming calls with our enhanced scheme.
- OV_M is the overhead $M'_f - M_f$ of our scheme for the MOVE operations
- F'_f is the expected cost for one FIND operation with our scheme
- OV_f is the overhead $F_f - F'_f$ of the simple forwarding scheme

Jain and Lin derived the expected cost M_f of all MOVE operations occuring between two incoming calls to the MT:

$$M_f = \frac{\mu S}{\lambda} + \frac{\mu \left(1 - f_m^*\left(\lambda\right)\right)\left(f_m^*\left(\lambda\right)\right)^{K-1}\left(M_{IS41} - S\right)}{\lambda\left(1 - \left(f_m^*\left(\lambda\right)\right)^K\right)} \quad . \tag{1}$$

Observe that in the schemes considered here, the load on the wireless link is the same : one message from the MT to the VLR. Therefore, we consider only the load on the signaling network and on the databases.

In our scheme, we set two pointers at each move. Hence, the cost M'_f for our scheme is obtained by replacing S by $2.S$:

$$M'_f = \frac{\mu 2.S}{\lambda} + \frac{\mu \left(1 - f_m^*\left(\lambda\right)\right)\left(f_m^*\left(\lambda\right)\right)^{K-1}\left(M_{IS41} - 2.S\right)}{\lambda\left(1 - f_m^*\left(\lambda\right)^K\right)} \quad . \tag{2}$$

The overhead of our scheme $OV_M = M'_f - M_f$ is

$$OV_M = \frac{\mu S}{\lambda}\left(\frac{1 - f_m^*(\lambda)^{K-1}}{1 - f_m^*(\lambda)^K}\right) \quad . \tag{3}$$

From now, we assume that the RA residual time is exponentially distributed, then

$$f_m^*(\lambda) = \frac{\mu}{\lambda + \mu} = \frac{1}{1+p} \tag{4}$$

where $p = \lambda/\mu$ is the call-to-mobility ratio. p expresses the ratio of the average number of incoming calls to the average number of RAs visited.

The overhead OV_M becomes :

$$OV_M = S.\frac{1+p}{p}\left(\frac{(1+p)^{K-1} - 1}{(1+p)^K - 1}\right) \quad . \tag{5}$$

If we normalize the cost S to 1, we can sketch the shape of OV_M as in Figure 4. We observe that the overhead is increasing with the threshold K but is not

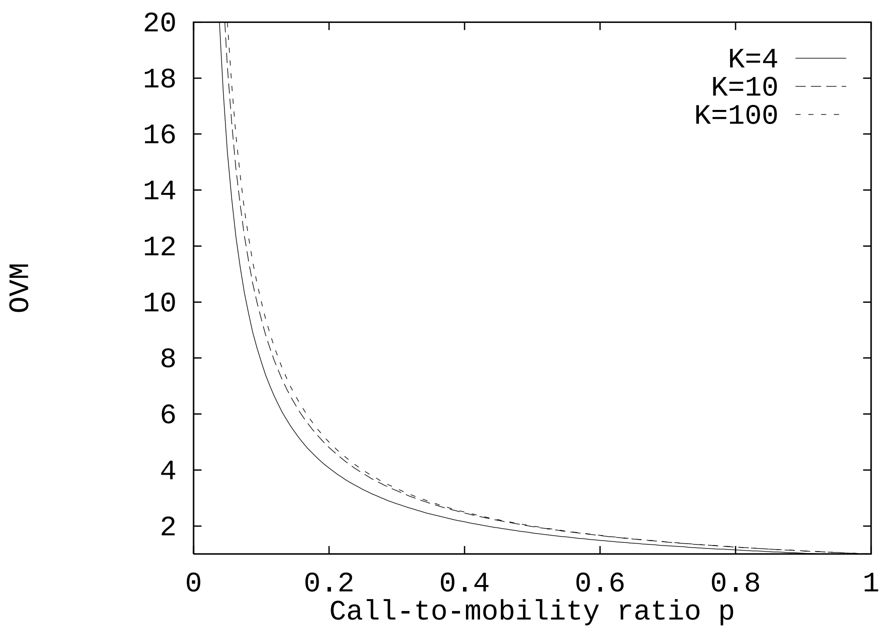

Fig. 4. Update overhead

very sensitive to it. With a call-to-mobility ratio less than 0.2, the overhead is considerable. Now, we consider the FIND cost F_f. If we assume that the user never revisits RAs, from Jain and Lin [9] :

$$F_f = \frac{\mu T}{\lambda} \left(\frac{1 - K f_m^*(\lambda)^{K-1} + (K-1) f_m^*(\lambda)^K}{1 - f_m^*(\lambda)^K} \right) + F \tag{6}$$

where

- T is the cost of traversing a pointer
- F is the cost for an IS-41 basic FIND

In our scheme, we traverse on the average half of the links. We can reasonably approximate the cost by replacing T by $\frac{T}{2}$ to compute the cost F_f' for the enhanced forward scheme

$$F_f' = \frac{\mu T}{2.\lambda} \left(\frac{1 - K f_m^*\left(\lambda\right)^{K-1} + \left(K-1\right) f_m^*\left(\lambda\right)^K}{1 - f_m^*\left(\lambda\right)^K} \right) + F \ . \tag{7}$$

The overhead $OV_F = F_f - F_f'$ for Jain and Lin's forwarding scheme is then

$$OV_f = \frac{\mu T}{2.\lambda} \left(\frac{1 - K f_m^*\left(\lambda\right)^{K-1} + \left(K-1\right) f_m^*\left(\lambda\right)^K}{1 - f_m^*\left(\lambda\right)^K} \right) \ . \tag{8}$$

Now, we want to know when the two overheads OV_M and OV_f can balance each other. We look then at the ratio $\frac{OV_M}{OV_f}$:

$$\frac{OV_M}{OV_f} = 2\frac{S}{T} \left(\frac{1 - f_m^*\left(\lambda\right)^{K-1}}{1 - K f_m^*\left(\lambda\right)^{K-1} + \left(K-1\right) f_m^*\left(\lambda\right)^K} \right) \ . \tag{9}$$

Let us compare the costs S and T. To set a pointer, the cost S consists of one message (VLR to VLR) and one database update. To traverse a pointer, the cost T consists of a retrieval database operation and one message. As messages size are comparable, we can reasonably consider S and T to be equal.

Replacing $f_m^*\left(\lambda\right)$ by $\frac{1}{1+p}$, the ratio of overheads becomes :

$$\frac{OV_M}{OV_f} = 2.\left(1+p\right) \left(\frac{\left(1+p\right)^{K-1} - 1}{\left(1+p\right)^K - K.p - 1} \right) \ . \tag{10}$$

The ratio above is plotted in Figure 5. It reveals that the decrease in the cost of FIND operation never balances the overhead introduced by our update.

The ratio is lower bounded by 2 for any value of p or K and is not upper bound. When p is less than 0.1, the ratio is large. This confirms the intuition that when call arrival is rare with respect to movement frequency, the gain in the FIND operation does not compensate the update overhead at each move. But, for the user, the call set up will take less time.

We study now if our scheme overhead can be balanced by the fault-tolerance introduced. We will try to establish the limits where our scheme can be beneficial.

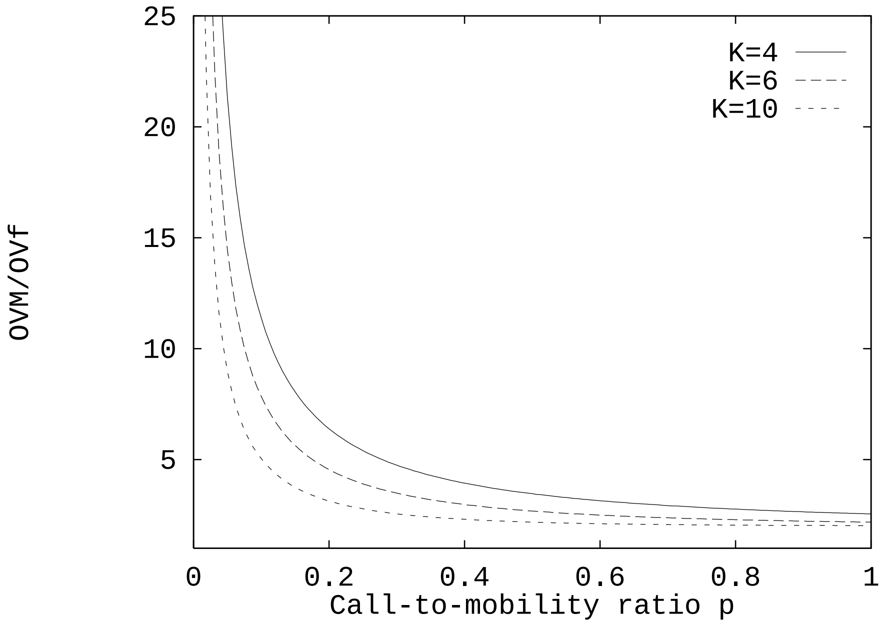

Fig. 5. Ratio of overheads $\frac{OV_M}{OV_f}$

6 Performance Analysis with Fault-Tolerance

With simple forwarding, if one intermediate VLR fails then the only way to find the mobile terminal is to page it. Let P_g be the cost of paging. We assume that paging will not fail. Let λ_f be the average failure rate of a VLR *experienced (observed) by a user*. The probability density function of time between two calls is f_f with Laplace transform f_f^*.

We will use in this section the following notations :

- λ_f is the failure rate of a VLR as observed by a user
- f_f is the probability density function of time between two calls.
- $f_f^*(s)$ is the Laplace transform of f_f
- $\beta(i)$ is the probability to have i calls between two failures
- P_g is the cost for paging a mobile terminal
- OV_M is the expected overhead of our scheme between two calls
- F_p is the expected cost of a FIND operation taking into account the possibility of VLR failures with the simple forwarding scheme
- F_p' is the expected cost of a FIND operation taking into account VLR failures with our scheme
- OV_f' is the expected overhead for the FIND operation with the simple forwarding scheme.

We can reasonably assume that the probability that two VLRs used by the same MT fail simultaneously is negligible. Let λ be the call arrival rate. We can derive the probability $\beta(i)$ to have i calls between two failures as follows :

$$\beta(i) = \frac{\lambda}{\lambda_f} . \left(1 - f_f^*(\lambda_f)\right)^2 \left(f_f^*(\lambda_f)\right)^{i-1} \quad . \tag{11}$$

Let us recall that OV_M is the expected overhead of our scheme between two calls. Now, let OV'_M be the expected overhead of location update between two failures :

$$OV'_M = \sum_{i=0}^{\infty} i . OV_M . \beta(i) \quad . \tag{12}$$

If we assume that the interarrival time of calls is negligible in comparison to the the time between failures, then we can consider overhead as constant and equal to the expected overhead computed above.

$$OV'_M = OV_M . \frac{\lambda}{\lambda_f} . \left(1 - f_f^*(\lambda_f)\right)^2 . \sum_{i=0}^{\infty} i . \left(f_f^*(\lambda_f)\right)^{i-1} \quad . \tag{13}$$

We get then

$$OV'_M = OV_M . \frac{\lambda}{\lambda_f} \quad . \tag{14}$$

Now, we compute F_p the FIND cost between two failures for the simple forwarding scheme with paging when a VLR fails,

$$F_p = P_g + \sum_{i=0}^{\infty} i . F_f . \beta(i) \tag{15}$$

which reduces to

$$F_p = P_g + \frac{\lambda}{\lambda_f} . F_f \quad . \tag{16}$$

The cost F'_p of our scheme will be

$$F'_p = F'_f . \left(\frac{\lambda}{\lambda_f} + 1\right) \quad . \tag{17}$$

This is because, when a VLR failure occurs, our scheme needs an additional FIND operation on the other path.

In this case, the simple forwarding scheme overhead $OV'_f = F_p - F'_p$ is

$$OV'_f = P_g - F'_f + OV_f . \frac{\lambda}{\lambda_f} \quad . \tag{18}$$

Now, if we want the overheads to balance each other, we must have $OV'_M = OV'_f$, i.e

$$P_g = F'_f + \frac{\lambda}{\lambda_f} . (OV_M - OV_f) \quad . \tag{19}$$

We must observe that the failure rate λ_f should be smaller when our scheme is used. We will try to integrate this fact in future work.

However, in general, $\lambda_f \ll \lambda$ by many orders of magnitude. So, the paging cost should be pretty high to compensate the overhead introduced by our update scheme. However, one VLR failure can affect many other users. This can lead to a brutal load increase on the SS7 network.

If we want to increase the performance of our scheme, we must strive to decrease the cost during the update operation. We must observe that this cost is higher than the simple forwarding scheme only in terms of messages. In term of time, our update does not cost more : the two messages to set the pointers are sent independently and the VLRs can set the pointers in parallel. To balance the load on the signaling network and on the VLRs, we can set the second pointer by sending a message with a very low priority. The first pointer is set normally. With this, the load can be staggered on the network and on the VLRs.

In [9], Jain and Lin showed that their forwarding scheme outperforms the basic IS-41 scheme when the Call-to-Mobility Ratio (CMR) is less than 0.5. If $CMR > 0.5$, the forwarding scheme performs poorly. From Figure (5), one may conclude that our scheme will then be justified when $0.2 < CMR < 0.5$.

7 Conclusion

We have described a new forwarding scheme. The scheme is fault-tolerant to intermediate visitor location registers failure and reduces the FIND operation cost. Hence, there is a clear overhead relative to the fault-tolerance. We showed that our scheme performs better if the intermediate location servers are not reliable or if the paging cost is very expensive. However, if the call-to-mobility ratio is less than 0.2, the simple forwarding scheme would perform much better. If the call-to-mobility ratio is larger than 0.5, the basic IS-41 scheme can at least be as good as the forwarding scheme (simple or enhanced). Hence, our enhanced scheme can be used for mobile terminals with a call-to-mobility ration between 0.2 and 0.5.

Also, our scheme does not introduce a time overhead. The overhead is only in terms of messages and VLR database update.

Further work on this subject has been done in [6]. We considered two fault-tolerant schemes : the scheme presented in this paper and another based on the knowledge of the VLR neighbors. We analyzed and compared these two schemes.

Appendix : Acronyms

CPU : Central Process Unit

HLR : Home Location Register

MSC : Mobile Switching Center

MT : Mobile Terminal

PCS : Personal Communications Services

PSTN : Public Switched Telephone Network
RA : Registration RA
SS7 : Signaling Systen no 7
STP : Signal Transfer Point
VLR : Visitor Location Register

References

1. A. Acharya and B. Badrinath, "Checkpointing distributed applications on mobile computers," in *IEEE Conf. on Parallel and Distributed Information Systems*, Oct. 1994.
2. I. F. Akyildiz and J. S. Ho, "On location management for personal communications networks," *IEEE Communications Magazine*, pp. 138–145, Sept. 1996.
3. I. F. Akyildiz, J. S. Ho, and Y.-B. Lin, "Movement-based location update and selective paging for PCS networks," *IEEE/ACM Transactions on Networking*, pp. 629–638, 1996.
4. S. Alagar, R. Rajagopalan, and S. Venkatesan, "Tolerating mobile support station failures," in *Proceedings of IEEE Conf. on Fault Tolerant Systems*, pp. 225–231, Dec. 1995.
5. N. M. Amato and M. C. Loui, "Checking linked data structures," in *International Symposium on Fault-Tolerant Computing FTCS-24*, pp. 164–173, June 1994.
6. S. Biaz and N. H. Vaidya, "Tolerating location register failures in mobile environments," Tech. Rep. 97-015, Department of Computer Science, Texas A&M University at College Station, Dec. 1997.
7. Q. Li and S. Vlaovic, "Redundant linked list based cache coherence protocol," in *1994 IEEE Workshop of Fault-Tolerant Parallel and Distributed Systems*, June 1994.
8. Y. Lin, "Failure restoration of mobility databases for personal communication networks," *ACM-Baltzer Wireless Networks*, pp. 365–372, 1995.
9. Y. Lin and R. Jain, "Performance modeling of an auxiliary user location strategy in a PCS network," *ACM-Baltzer Wireless Networks*, pp. 197–210, June 1995.
10. C. Lo and R. Wolff, "Estimated network database transaction volume to support wireless personal data communications applications," in *Proc. Conf. on Communications*, 1993.
11. K. Meier-Hellstern and E. Alonso, "The use of SS7 and GSM to support high density personal communications," in *Proceedings Int'l Conference on Communications*, 1992.
12. A. Modaressi and R. Skoog, "Signaling system no. 7," *IEEE Com. Mag.*, July 1990.
13. D. Pradhan, P. Krishna, and N. Vaidya, "Recovery in mobile environments: Design and trade-off analysis," in *26th Int'l Symposium on Fault-Tolerant Computing*, June 1996.
14. S. Rangarajan, K. Ratnam, and A. T. Dahbura, "A fault-tolerant protocol for location directory maintenance in mobile networks," in *25th Int'l Symposium on Fault-Tolerant Computing*, June 1995.