Wireless Networks 1 (1995) 197–210

An auxiliary user location strategy employing forwarding pointers to reduce network impacts of PCS

Ravi Jain and Yi-Bing Lin

Applied Research, Bellcore, 445 South St, Morristown, NJ 07960, USA

Received December 1994

Abstract. We propose an auxiliary strategy, called *per-user forwarding*, for locating users who move from place to place while using Personal Communications Services (PCS). The forwarding strategy augments the basic location strategy proposed in existing standards such as GSM and IS-41, with the objective of reducing network signalling and database loads in exchange for increased CPU processing and memory costs. The key observation behind forwarding is that if users change PCS registration areas frequently but receive calls relatively infrequently, it should be possible to avoid registrations at the Home Location Register (HLR) database, by simply setting up a forwarding pointer from the previous Visitor Location Register (VLR). Calls to a given user will first query the user's HLR to determine the first VLR which the user was registered at, and then follow a chain of forwarding pointers to the user's current VLR. We use a reference PCS architecture and the notion of a user's *call-to-mobility ratio* (CMR) to quantify the costs and benefits of using forwarding and classes of users for whom it would be beneficial. We show that under a variety of assumptions forwarding is likely to yield significant net benefits in terms of reduced signalling network traffic and database loads for certain classes of users. For instance, under certain cost assumptions, for users with $CMR < 0.5$, forwarding can result in 20–60% savings over the basic strategy. This net benefit is due to the significant saving in location update compared to a penalty of moderately increased call setup times for the infrequent occasions when these users do receive calls.

1. Introduction

We consider the problem of locating users who move from place to place while using Personal Communications Services (PCS). Previous studies [10,14,11,12] have shown that, with predicted levels of PCS users, there will be significant loads upon the signalling network and network databases, and that these loads are dependent upon the data management strategies adopted. We present a user location strategy which has the potential to reduce these loads significantly. The strategy we discuss here is an *auxiliary* strategy, in that it augments the *basic* user location strategies proposed in standards such as the North American IS-41 cellular standard [6] and the European GSM standard for mobile communications [13,17]. For a survey of various user location strategies in PCS systems, see [9] and [16].

The strategy we present is the use of *per-user forwarding*. This strategy, like other auxiliary strategies [9], attempts to reduce the network signalling and database loads of the basic strategies in exchange for increased CPU processing and memory costs. Since technology trends are driving the latter costs down, deploying the forwarding strategy on a system-wide basis will become increasingly attractive. Once deployed, whether the forwarding strategy should be invoked for a particular user is a function of the user's mobility and communications patterns, as discussed below.

The *basic* user location strategies proposed in the IS-41 [6] and GSM [13,17] standards are *two-level* strategies in that they use a two-tier system of Home Location Register (HLR) and Visitor Location Register (VLR) databases (see also [16]). In two-level strategies, as in most user location strategies, the user performs registration at the HLR every time the user changes registration areas, and deregisters at the previous VLR. The key observation we make is that, in many cases, it should be possible to avoid these registrations at the HLR, by simply setting up a forwarding pointer from the previous VLR. Calls to a given user will first query the user's HLR to determine the first VLR which the user was registered at, and then follow a chain of forwarding pointers to the user's current VLR. This observation results in a strategy which will be useful for those users who receive calls infrequently relative to the rate at which they change registration areas. This idea attempts to exploit patterns in the call reception and mobility of PCS users, and is called *per-user forwarding*. We have previously designed an auxiliary strategy, called *per-user caching*, which attempts to exploit the calling and mobility patterns of a different population of users, namely those who receive calls frequently relative to the rate at which they change registration areas [19,21].

Per-user forwarding aims to reduce the total network costs of supporting users who travel frequently (hence creating registration and deregistration traffic) but do not receive calls frequently. This situation arises, for example, in a scenario where the user commutes a fairly long distance to work. At work the user may receive numerous calls (some possibly directed to a fixed telephone in order to avoid wireless communication costs) but does not change registration areas frequently. When the user is commuting, however, the user may not receive many calls.[1] We will see that if forwarding is applied to a rapidly moving user who receives calls infrequently, and the user does in fact receive a call, the setup time for the call is likely to be somewhat longer than it would have been otherwise. (The per-user nature of forwarding means that the penalty in mean call setup time is paid only by such users.) Thus the overall network cost for supporting the user is reduced, at the expense of increased call setup time on the infrequent occasions when the user does receive a call.

The outline of this paper is as follows. (Some of this background material in this paper is repeated from our earlier paper [19] in order to keep this paper self-contained.) In section 2, we describe the PCS network architecture that we assume for the presentation and analysis of the basic and forwarding location strategies. In section 3, we describe the forwarding strategy. A feature of the forwarding location strategy is that it is useful only for certain classes of PCS users, those meeting certain call and mobility criteria. We encapsulate this notion in the definition of the user's call-to-mobility ratio (CMR) in section 4 and use it to develop a performance model of per-user forwarding parameterized for different classes of users.[2] We observe that the benefits of forwarding are highly dependent upon the cost of setting up forwarding pointers relative to the cost of updating the HLR. Thus in section 5, we use our PCS network reference architecture to quantify the costs and benefits of forwarding and the network elements for which forwarding is beneficial. In section 6, we briefly discuss alternative architectures and implementation issues of the strategy proposed, and briefly discuss related work. Section 7 provides some conclusions.

It should be stressed at the outset that the assumptions we have used in this paper constitute one reference set of assumptions. A number of variations in the assumptions could be considered, including variations in the network architecture and in the auxiliary location strategy. In most cases these variations may alter our analysis in relatively minor ways and may not significantly affect the qualitative conclusions we draw. The intent of this paper is to present the key ideas behind the forwarding auxiliary strategy and to develop a method for quantifying its costs and benefits. This method can then be applied to specific architectures and deployment scenarios as needed.

2. PCS network architecture

PCS users receive calls via either wireless or wire-line access. In general, calls may deliver voice, data, text, facsimile or video information. For our purposes, we define the location of a PCS user, as known by the wire-line network, as the *registration area* (RA) in which the user is located. For users attached directly to a wire-line network, the RA is defined as the point of attachment. For users attached via wireless links, the situation is described as follows. In order to deliver calls by wireless links, the geographical region covered by a PCS network is divided into radio port coverage areas, or *cells*. Each cell is primarily served by one radio *base station*, although a base station may serve one or more cells. The base station locates a user, and delivers calls to and from the user, by means of paging within the cell(s) it serves. Base stations are connected to the rest of the wire-line network by wire-line links. An RA consists of an aggregation of cells, forming a contiguous geographical region.

We assume that a signalling network is used to set up calls which is distinct from the network used to actually transport the information contents of the calls. Specifically, we assume a Common Channel Signalling (CCS) network is used to set up calls which uses the Signaling System No. 7 (SS7) protocols (see [15] for a tutorial).

Fig. 1 illustrates the reference signalling network assumed in this study. This architecture is meant as a reference architecture for a hypothetical geographical region, and is not necessarily the architecture corresponding to any particular implementation. The cells of the geographical region are served by base stations and are aggregated into RAs. The base stations of an RA are connected via a wire-line network to an end-office switch, or Service Switching Point (SSP). Each SSP serves a single RA. SSPs of different RAs are in turn connected to a two-level hierarchy of Signalling Transfer Points (STPs), comprised of a Regional STP (RSTP)

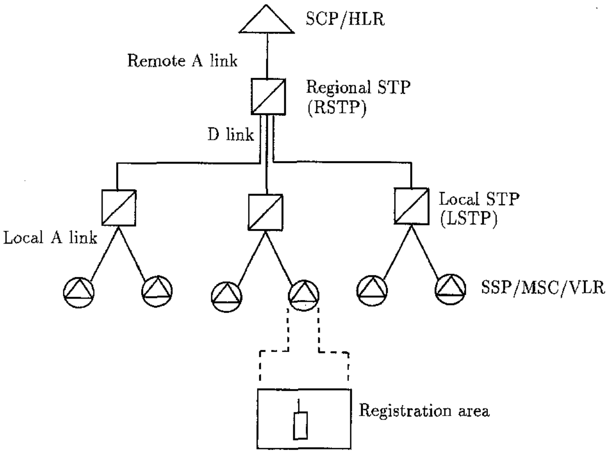


Fig. 1. Reference CCS network architecture.

---

[1] Note that current billing policies for cellular phone usage, where the subscriber is billed for received calls, also tend to discourage subscribers from giving out their cellular number widely and hence from receiving calls. (It is not clear if similar billing policies will apply to future PCS networks.)

[2] As we will see in later sections, the user does not have the burden of informing the network about its CMR; this information is maintained by the network.

connected to all Local STPs (LSTPs) in the region, which perform message routing and other SS7 functions. (In practice each STP actually consists of two STPs in a mated-pair configuration for redundancy [15]; for simplicity Fig. 1 only shows one of the two STPs of each mated pair). The RSTP is also connected to a Service Control Point (SCP), which is assumed to contain the functionality associated with a Home Location Register (HLR) database.

For simplicity, we assume that in functional terms the Mobile Switching Center (MSC) is collocated with the SSP. In addition, a distinct Visitor Location Register (VLR) database is associated with each MSC. These assumptions are not unreasonable. It is already anticipated that the VLR will be collocated with the MSC, and the MSC/VLR combination is likely to evolve to be SS7 compatible. (In the rest of this paper, the terms switch or SSP will be used interchangeably, depending on the context). Each switch is assumed to serve exactly one RA, which, in turn, may be comprised of one or more cells. This assumption is used to simplify the ensuing analysis, although in practice, each VLR may serve a number of RAs.

In this paper we do not address issues relating to the content of messages and other information transfer (e.g., for billing, etc.) which may occur during a call. For simplicity it is assumed that message sizes are equal for different types of transactions (e.g., location request, registration and deregistration), for both query and update invocations as well as their associated response messages. Since we only perform a comparative analysis of the basic strategy with and without the auxiliary forwarding strategy, the conclusions will not be affected by this simplification.

3. Per-user forwarding

The basic idea behind per-user forwarding is to avoid unnecessary CCS message traffic, and updates of a user's HLR, if the user moves across RAs relatively frequently but receives calls relatively infrequently. Two operations which incur signalling are defined (we use the terminology of [19,1]):

1. *MOVE*, where the PCS user moves from one RA to another, and
2. *FIND*, where the RA in which the PCS user is currently located is to be determined.

A. Basic user location procedures

We first present the *MOVE* and *FIND* procedures used in current PCS standards proposals such as IS-41 [6]; we call these the *BasicMOVE* and *BasicFIND* procedures. (Note that the *BasicMOVE* and *BasicFIND* procedures we present are simplifications of those in the standards, and only attempt to capture the major interactions between the HLR and VLR databases relevant to our purposes. The reader is referred to [6;17] for their detailed specification.) We use the following pseudocode to describe the *BasicFIND( )* and *BasicMOVE( )* procedures.

```
BasicMOVE( ) {
    The mobile terminal detects that it is
        in a new registration area;
    The mobile terminal sends a registration
        message to the new VLR;
    The new VLR sends a registration message
        to the user's HLR;
    The HLR sends a registration cancellation
        message to the old VLR;
    The old VLR sends a cancellation confirm
        message to the HLR;
    The HLR sends a registration confirm
        message to the new VLR;
}
```

Notice that the *BasicMove( )* procedure involves notifying the HLR every time a user enters a new registration area. The registration information is used to find the user when a call is to be delivered to that user.

```
BasicFIND( ) {
    Call to PCS user is detected at local switch;
    if called party is in same RA then return;
    Switch queries called party's HLR;
    HLR queries called party's current VLR, V;
    VLR V returns called party's location to HLR;
    HLR returns location to calling switch;
}
```

B. Forwarding user location procedures

The forwarding procedures modify the basic procedures as follows. When a user moves from one RA to another it informs the switch (and VLR) at the new RA of the old RA from which it arrived. The switch at the new RA determines whether to invoke the basic *MOVE* or the forwarding *MOVE* strategy.

In *FwdMOVE ( )*, the new VLR exchanges messages with the old VLR to set up a forwarding pointer from the old to the new VLR, but does not involve the user's HLR. See Fig. 2. The message labeled *REG(RAa)* in Fig. 2 is a registration message from the user terminal indicating that it has moved from the previous RA, *RAa*, to the current RA. The messages *REG(RAb)*... *REG(RAx)* are similar. The message *REGPTR* is a message from the new VLR to the old VLR specifying that a forwarding pointer is to be set up; message *regptr* is the confirmation from the old VLR that this has been done.

A subsequent call to the user from some other switch will invoke the *FwdFIND( )* procedure, which queries

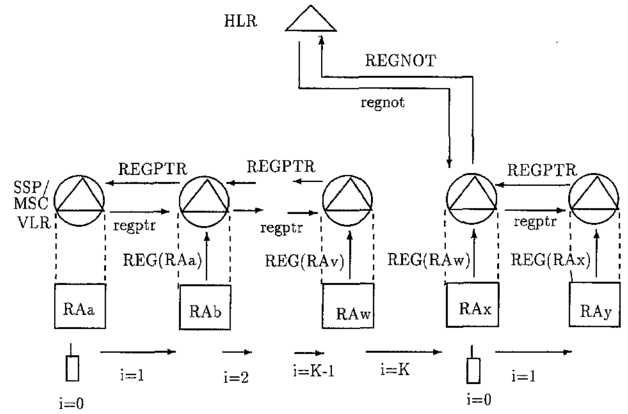


Fig. 2. The *FwdMOVE( )* procedure.

the user's HLR as in the basic strategy, and obtains an outdated pointer to the old VLR. The pointer from the old VLR is then followed to the new VLR, in order to determine the user's correct current location. (See Fig. 3). To ensure that the time taken by *FwdFIND( )*, and hence the call setup time, is bounded to a reasonable value, the length of the chain of forwarding pointers must be limited. This is accomplished by allowing the chain to grow to at most K pointers during the *FwdMOVE( )* process.

We describe the forwarding procedures using pseudocode as follows. (We use the shared global variable i in the pseudocode).

```
FwdMOVE( ) {
  /*Initially, $i$ is 0 */
  if ($i<K-1$) {
      User registers at new RA/VLR, passing
          id of former RA/VLR;
      New VLR deregisters user at old VLR;
      Old VLR sends ACK and user's service
          profile to new VLR;
      $i := i+1$;
  }
  else {
      BasicMOVE();
      $i := 0$;
  }
}
```

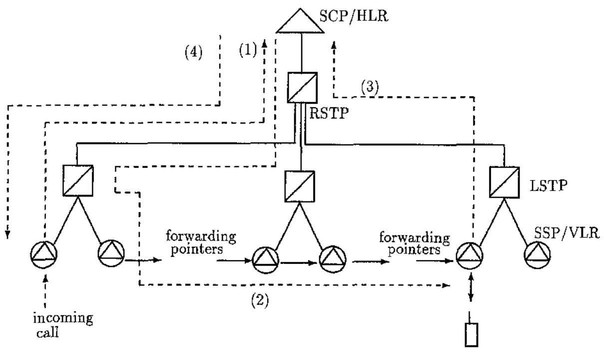


Fig. 3. The *FwdFIND( )* procedure.

```
FwdFIND( ) {
  Call to PCS user is detected at local switch
  if called is in same RA then return
  Switch queries called's HLR
  HLR responds to caller's switch with $V_0$
  Caller's switch queries $V_0$
  while (Queried VLR is not called's current VLR)
      VLR queries next VLR in pointer chain
  /* Now called's actual VLR has been found */
  $i := 0$;
  Called's current VLR sends user location to HLR
  HLR sends user location to caller's switch
}
```

4. Performance modeling of per-user forwarding

In this section we develop an analytic model to study the performance of per-user forwarding parameterized for different classes of users.

We characterize classes of users by their *call-to-mobility ratio* (CMR). The CMR of a user is defined as the expected number of calls to a user during the period that the user visits an RA. (Note that the CMR is defined here in terms of the calls *received* by a particular user, not calls originated by the user). If calls are *received* by the user at a mean rate λ and the time the user resides in a given RA has a mean $1/\mu$, then the CMR, denoted p, is

$$p = \lambda/\mu . \tag{1}$$

As we will see later, forwarding will only be applied to classes of users whose CMR is below a certain CMR threshold.

Suppose a user crosses several RAs between two consecutive phone calls. If the basic user location strategy is used, the *BasicMOVE* procedure is called every time the user moves to a new RA, in order to update the user's location at the HLR. If the forwarding strategy is used, the HLR is updated only every Kth move, with forwarding pointers being set up for all other moves.

We calculate C_B and C_F, the total costs of maintaining user information and locating the user between two consecutive calls for the basic and forwarding strategies, respectively. Let m denote the cost of a single invocation of *BasicMOVE*, M be the total cost of all the *BasicMOVEs* between two calls, and F be the cost of a single *BasicFIND*. Then,

$$C_B = M + F = m/p + F . \tag{2}$$

Similarly, let M' be the expected cost of all *FwdMOVEs* between two consecutive calls, and F' the average cost of the second *FwdFIND*. Then

$$C_F = M' + F' . \tag{3}$$

We are interested in two measures of performance: the ratio F'/F, which reflects the additional cost paid during call setup for those users to whom call forwarding is applied, and the ratio C_F/C_B, which reflects the net effect of applying the forwarding scheme. We will also examine the ratio M'/M which reflects the benefit of forwarding in terms of reduced *MOVE* costs, in order to get better understanding of the performance criteria.

We denote costs for various operations used in the forwarding strategy as follows.

S = *Cost of setting up a forwarding pointer between VLRs during a FwdMOVE.*

T = *Cost of traversing a forwarding pointer between VLRs during a FwdFIND*

We now derive formulas for M' and F'. The cost M' is derived as follows. Let $\alpha(i)$ be the probability that there are i RA crossings between two call arrivals. Suppose that a user crosses i RA boundaries between two call arrivals. Then there are $i - \lfloor i/K \rfloor$ pointer creations (every K moves require $K-1$ pointer creations). The HLR is updated is $\lfloor i/K \rfloor$ times (with the forwarding strategy, the HLR is updated every Kth move). Thus,

$$M' = \sum_{i=0}^{\infty}\left[\left(i - \left\lfloor \frac{i}{K} \right\rfloor\right)S + \left\lfloor \frac{i}{K} \right\rfloor m\right]\alpha(i) . \qquad (4)$$

The cost F' is derived as follows. After the last *Basic-Move* operation (if any), the user crosses $n = i - K\lfloor \frac{i}{K} \rfloor$ RA boundaries. Thus,

$$F' = F + \sum_{i=0}^{\infty}\left(i - K\left\lfloor \frac{i}{K} \right\rfloor\right)T\alpha(i) . \qquad (5)$$

In order to evaluate $\alpha(i)$, the following assumptions are made.

1. The call arrivals to a user are a Poisson process with the arrival rate λ.
2. The residence time of a user at a registration area is a random variable with a general density function f_m and the Laplace transform

$$f^*_{\mathrm{m}}(s) = \int_{t=0}^{\infty} f_m(t)e^{-st}dt .$$

The expected residence time of a user at an RA is $1/\mu$.

Observe that $p = \lambda/\mu$ as before. We will use the abbreviation $g = f^*_{\mathrm{m}}(\lambda)$ for convenience in the following. With these assumptions, it can be shown that (see Appendix A for the derivation):

$$M' = \frac{S}{p} + \frac{(1-g)g^{K-1}(m-S)}{p(1-g^K)} , \qquad (6)$$

$$F' = F + \frac{[1 - Kg^{K-1} + (K-1)g^K]T}{p(1-g^K)} . \qquad (7)$$

For demonstration purposes, we assume that the RA residence time of a user is a Gamma distribution with mean $1/\mu$. The Gamma distribution [18] is selected for its flexibility. By selecting the proper values for the parameters, a Gamma distribution can model an Exponential, an Erlang or an Chi-square distribution. A Gamma distribution can also be used to represent the distribution for a set of measured data. The Laplace transform of a Gamma distribution is

$$f^*_{\mathrm{m}}(s) = \left(\frac{\gamma\mu}{s+\gamma\mu}\right)^{\gamma} ,$$

so that, in our case,

$$g = f^*_{\mathrm{m}}(\lambda) = \left(\frac{\gamma\mu}{\lambda+\gamma\mu}\right)^{\gamma} = \left(\frac{\gamma}{p+\gamma}\right)^{\gamma} . \qquad (8)$$

In particular, choosing $\gamma = 1$ corresponds to an exponential distribution of residence time.

4.1. Performance analysis for exponential residence time

We first consider the situation when the residence time is exponentially distributed. Then setting $\gamma = 1$ in Eq. (8), we have

$$g = \frac{1}{1+p}$$

and Eqs. (6) and (7) can be written as

$$M' = \frac{S}{p} + \frac{m-S}{(1+p)^K - 1} ,$$

$$F' = F + \frac{T}{p} - \frac{KT}{(1+p)^K - 1} . \qquad (9)$$

From Eqs. (3) and (9),

$$C_F = F + \frac{T+S}{p} + \frac{m-S-KT}{(1+p)^K - 1} . \qquad (10)$$

The next section will give a detailed analysis of the costs m, F, S and T based on the proposed architecture in Fig. 1. As a first approximation, this section considers the values of the operation costs as follows. We observe that updating the HLR and performing a *BasicFIND* involve the same number of messages between HLR and VLR databases, so we set $m = F$. We also assume that the cost of setting up a forwarding pointer is about twice the cost of traversing it, since twice as many messages are involved, i.e., we set $S = 2T$. Finally, we normalize $m = 1$, and $S = \delta$, with $\delta < 1$. Then, from Eqs. (2), (9) and (10)

$$C_B = 1 + \frac{1}{p} ,$$

$$\frac{M'}{M} = \delta + \frac{p(1-\delta)}{(1+p)^K - 1} , \qquad (11)$$

$$\frac{F'}{F} = 1 + \frac{\delta}{2p} - \frac{\delta K}{2[(1+p)^K - 1]} , \qquad (12)$$

$$\frac{C_F}{C_B}=\frac{p}{1+p}\left\{1+\frac{3\delta}{2p}+\frac{2-(K+2)\delta}{2[(1+p)^K-1]}\right\}. \qquad (13)$$

In Fig. 4 we plot the equations above as functions of the CMR for various values of K and δ. In the rest of this section we will use the plots to discuss the behavior of the forwarding scheme. We make a quick observation as follows. Fig. 4(a) shows that under certain assumed conditions ($\delta=0.3$, $K\leq 4$, $p\leq 2$), forwarding can result in 35%–65% reductions in location update cost compared to the basic strategy. However, Fig. 4(b) indicates that the improvement comes at a price, namely increased mean call setup time; however, for the same assumed conditions, we see that this penalty is at most 25%. (The call setup time penalty will also be discussed briefly in section 6.4.) The net effect shown in Fig. 4(c) indicates that forwarding can result in 20%–60% reductions in the total network cost. (The dashed line in Fig. 4(c) shows the value $C_F/C_B=1$.)

We now discuss the plots of Fig. 4 in more detail. Fig. 4(a) plots M'/M against the CMR, p, using Eq. (11), for the case $\delta=0.3$. Fig. 4(b) plots F'/F against the CMR using Eq. (12), for the case $\delta=0.3$. It is apparent that $F'\geq F$ for all p values. F'/F is a decreasing function of p. Figs. 4(c) and 4(f) plot C_F/C_B against the CMR p, using Eq. (13), for various values of K and the cases $\delta=0.3$ and $\delta=0.7$.

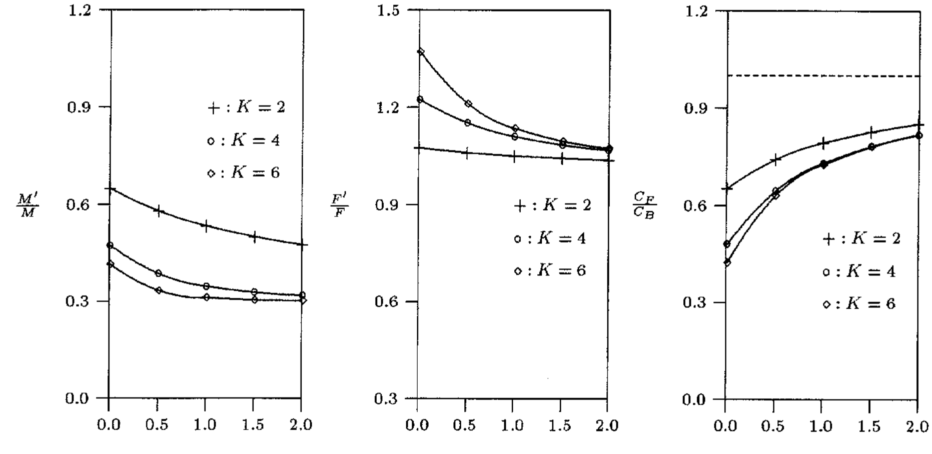
Consider the behavior of F'/F and C_F/C_B as p decreases from 2 to 0; we see that while the improvement in total cost increases so does the penalty in setup time. This is because when p is small, the improvement in total cost is high because most *MOVEs* do not result in an HLR update but a pointer creation. However, if a call does occur, a lengthy chain of pointers will have to be traversed, leading to the high setup time penalty.

To get further understanding of the behavior of the forwarding scheme, we analyze upper and lower bounds for the performance measures C_F/C_B and F'/F. It can be shown (see Appendix B for the derivation) that

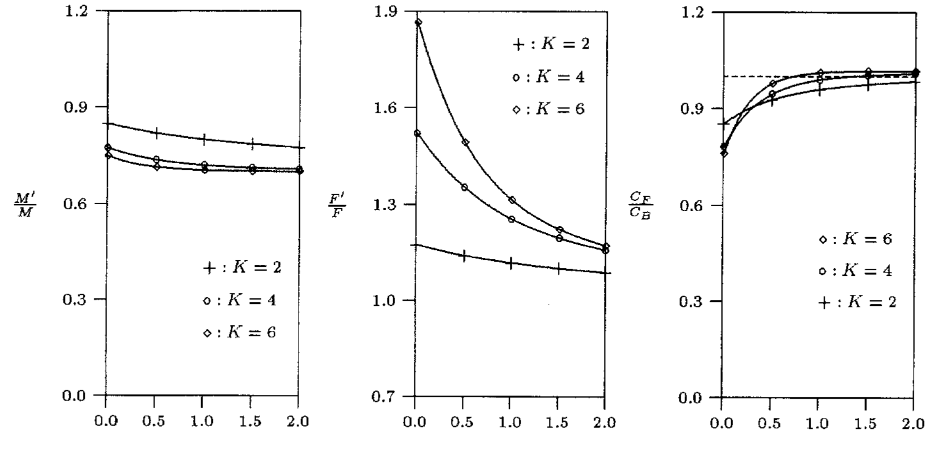
$$1\leq\frac{F'}{F}\leq 1+\frac{T(K-1)}{2F}, \qquad (14)$$

$$\frac{1}{K}+\frac{(K-1)S}{Km}\leq\frac{C_F}{C_B}\leq\max\left(1,\frac{T+S}{m}\right). \qquad (15)$$

Thus, setting $m=F=1$ and $S=2T=\delta$, we have



(a) The *MOVE* cost ($\delta=0.3$) (b) The *FIND* cost ($\delta=0.3$) (c) The net cost ($\delta=0.3$)



(d) The *MOVE* cost ($\delta=0.7$) (e) The *FIND* cost ($\delta=0.7$) (f) The net cost ($\delta=0.7$)

Fig. 4. Relative *MOVE* and *FIND* costs of forwarding.

$$1 \le \frac{F'}{F} \le 1 + \frac{(K-1)\delta}{4}, \qquad (16)$$

$$\frac{1}{K} + \frac{(K-1)\delta}{K} \le \frac{C_F}{C_B} \le \max(1, 1.5\delta). \qquad (17)$$

The network designer can use the inequalities above for a quick first evaluation of the costs and benefits of the forwarding scheme. For example, if $\delta = 0.5$ and $K = 2$, then forwarding can result in upto 25% reductions in total network cost over the basic scheme. Similarly, if $T + S \le m$, the total network cost of the forwarding scheme cannot exceed the basic scheme.

As shown in Appendix B, the lower and upper bounds in Eq. (16) occur as $p \rightarrow \infty$ and $p \rightarrow 0$, respectively (see Fig. 4(b)). Intuitively, this can be explained as follows. As $p \rightarrow \infty$, the user never moves out of an RA, so the HLR points to the VLR where the user actually resides, and the cost of *FwdFIND* is the same as *BasicFIND*. On the other hand, as $p \rightarrow 0$, the user is very likely to cross many RA's (much larger than K) between consecutive calls, so that when a call does occur, $(K-1)/2$ pointers have to be traversed to locate the user, on average; in that case, the cost of locating the user (compared to the basic scheme) increases with $T/F = \delta/2$. Putting the two observations together, we see why the curves in Fig. 4(b) and (e) will diverge in proportion to K near $p = 0$ and converge towards 1 as p gets large.

The lower bound for C_F/C_B in Eq. (17) occurs for $p \rightarrow 0$ (see Fig. 4(c) and 4(f)). In this case the net costs of the two schemes are dominated by the *MOVE* operations, and the *FwdMOVE* operations are less costly than the *BasicMOVE* operations. Note that the improvement in C_F over C_B thus arises largely because only every Kth *FwdMOVE* has the same cost as a *BasicMOVE*, i.e., the improvement is dependent on the fraction $(K-1)/K$. Since this fraction decreases relatively slowly for $K > 5$, this explains why increasing K further results in diminishing improvements in C_F/C_B for small p. This suggests that the value of K should be chosen carefully and not set too large.

As expected, increasing δ from 0.3 (in Fig. 4(a)–(c)) to 0.7 (in Fig. 4(d)–(f)) increases the cost of *FwdMOVE* and hence diminishes the improvement in total cost.

The behavior of C_F/C_B as p increases is more complex. First note that the advantages provided to the forwarding scheme by the *FwdMOVE* operation are offset by the disadvantages due to *FwdFIND*. In addition, the net costs of the two schemes are dominated by the *MOVE* operations when p is small, and by the *FIND* operations when p is large. The bounds analysis in Appendix B shows that as p increases, C_F/C_B has an upper bound of $\max(1, (T+S)/m) = \max(1, 1.5\delta)$. Thus, for $\delta < 0.67$, as $p \rightarrow \infty$, the ratio C_F/C_B is dominated by F'/F, which, as discussed above, approaches unity. (See Fig. 4(c)). For $\delta > 0.67$, as p increases, C_F/C_B first reaches a maximum before converging to unity; this can be observed (although not very clearly) for the case $K = 6$, $\delta = 0.7$ in Fig. 4(f).

Consider now the behavior of C_F/C_B for large p as K and δ are varied. For large p, the pointer chain rarely becomes as long as K, so increasing K has little effect on F'/F and hence C_F/C_B. Similarly, the cost of setting up and traversing pointers is less often incurred than for small p, so increasing δ has less effect on F'/F and hence C_F/C_B. Finally, observe that Fig. 4(c) indicates that for small δ, increasing K reduces the cost of the forwarding scheme (because the pointer operations are cheap). As δ increases, the performance of the forwarding scheme with a large K degrades faster than the scheme with a small K (see Fig. 4(f)). The phenomenon is due to the fact that more pointer operations are performed in the scheme with a larger K value, and thus it is more sensitive to the costs of the pointer operations δ.

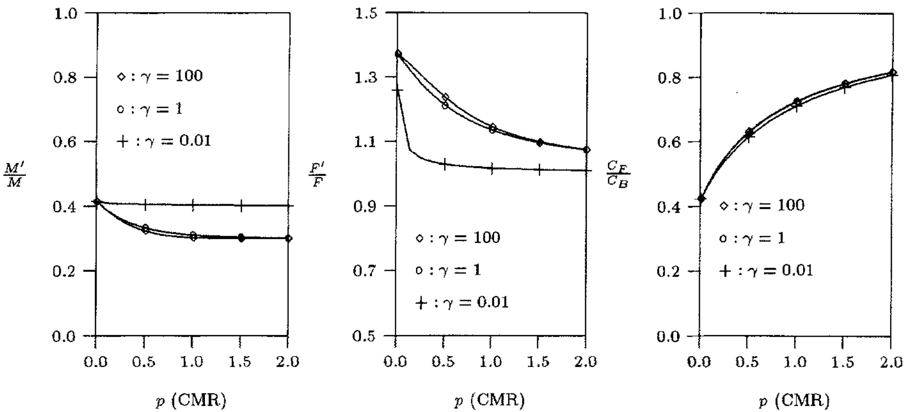
4.2. *Sensitivity of performance to residence time variance*

We now investigate the sensitivity of the performance costs and benefits of the forwarding scheme to the variance in the user's mobility patterns. We do this by keeping call receptions a Poisson process, but considering the effects when the residence time has a Gamma distribution.

For a Gamma distribution, the variance is $V = \frac{\mu^2}{\gamma}$. That is, a large γ value implies a small variance. Fig. 5 shows the effect of γ on M'/M, F'/F and C_F/C_B. In Fig. 5 we see increasing the variance of residence time (smaller γ) increases M'/M but decreases F'/F; the net effect is a small increase in C_F/C_B. Thus as variance in residence time increases a small decrease in the call setup time penalty occurs in exchange for a small reduction in the improvement in total network cost.

We define a *cycle* as the time between two consecutive phone calls to the user. A cycle is called *dense* if it contains many RA crossings, and *sparse* if it contains few RA crossings, compared to $1/p$. For a given p value, a larger variance (smaller γ) in residence time implies more variance in the number of RA crossings per cycle. Thus, for instance, compared to a Poisson residence time distribution ($\gamma = 1$), a distribution with $\gamma < 1$ implies more dense cycles as well as more sparse cycles. A distribution with $\gamma > 1$ implies that the number of MOVEs in a cycle is likely to be close to $1/p$.

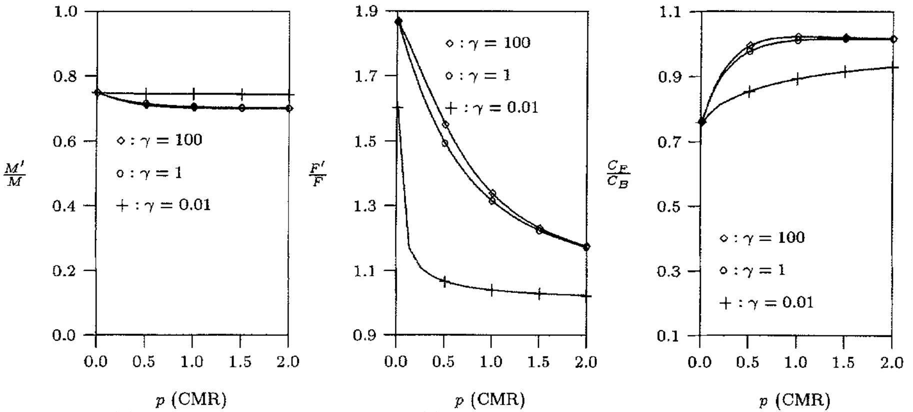
Consider M'/M for $\gamma < 1$ compared to $\gamma = 1$, for any given value of $p > 0$ (see Fig. 5(a) and (d)). The dense cycles will tend to increase M' while the sparse cycles will not. This is because dense cycles will tend to create longer pointer chains; after a chain contains more than $K-1$ pointers, it will be truncated and the HLR will be updated, resulting in increasing M' by an amount $m = 1$. On the other hand, the sparse cycles will only result in shorter pointer chains, saving the pointer setup



(a) The *MOVE* cost $K = 6, \delta = 0.3$

(b) the *FIND* cost $K = 6, \delta = 0.3$

(c) The net cost $K = 6, \delta = 0.3$



(d) The *MOVE* cost $K = 6, \delta = 0.7$

(e) the *FIND* cost $K = 6, \delta = 0.7$

(f) The net cost $K = 6, \delta = 0.7$

Fig. 5. The effect of variance in residence time (γ).

cost (which is $S/m = \delta < 1$ for each pointer which is created). The net effect is an increase in M'.

Now consider F'/F for any given value of $p > 0$, say $p = 1$ (see Fig. 5(b)). For low variance, most cycles will have a single *MOVE*, so that a *FIND* will result in traversal of a single pointer. On the other hand, for high variance for the same value of p, there will be more cycles with zero *MOVE*s, reducing F'. While these sparse cycles will be partially counterbalanced by dense cycles, note that dense cycles with K or $K+1$ *MOVE*s will have their pointer chains truncated so that these cycles cannot increase F'; so will all dense cycles with an integer multiple of K or $K+1$ *MOVE*s. The net effect is a significant improvement in F'/F.

The net effect of the variance of residence time on M'/M and F'/F for $\delta = 0.3$ is a negligible decrease in the total cost ratio C_F/C_B (see Fig. 5(c)). It is interesting to see that this effect is very dependent on δ, as shown in Fig. 5(d)–(f). *Thus increased variance in residence time distribution can mitigate the negative effects of a large value of δ.* Once again, it is instructive that the costs and benefits of forwarding depend significantly upon the costs of setting up and traversing pointers relative to the costs of updating the HLR; this will motivate our more detailed evaluation in section 5.

5. Forwarding threshold analysis

We have seen in the previous section that the costs and benefits of the forwarding strategy are strongly dependent upon the costs of setting up an traversing pointers compared to the costs of updating the HLR, i.e., the value δ. For this reason, the strategy is useful only for users who are below a certain CMR threshold.

In this section we compare the costs of the forwarding strategy with that of the basic strategy for a specific cost model based upon the reference architecture of Fig. 1. As seen from the analysis of the previous section, the benefits of forwarding depend on p (the user's CMR), and K (how long the chain of forwarding pointers is allowed to get). When using the specific reference architecture of Fig. 1, we see that it also depends upon the probability that a given move takes the user outside a region served by an LSTP.

We will focus on finding the situations for which forwarding is beneficial, i.e., $C_F < C_B$. From Eqs. (2) and (3) in the previous section, we find this holds iff

$$\frac{T+S}{p} + \frac{(m-S-KT)}{X} < \frac{m}{p}, \tag{18}$$

where $X = (1+p)^K - 1$. To quantify this further, we estimate S, T and m by assuming costs for traversing various network elements as follows:

$A_l =$ Cost of transmitting a location request or response message on A-link between SSP and LSTP.

$D =$ Cost of transmitting a location request or response message on D-link.

$A_r =$ Cost of transmitting a location request or response message on A-link between RSTP and SCP.

$L =$ Cost of processing and routing a location request or response message by LSTP.

$R =$ Cost of processing and routing a location request response message by RSTP.

$H_U =$ Cost of an update to the HLR.

$V_Q =$ Cost of a query to the VLR to obtain the routing address.

$V_U =$ Cost of an update to the VLR.

From Fig. 1, if the called party moves outside its LSTP area frequently, searching for its current location will also involve following forwarding pointers via the RSTP frequently. To quantify this behavior we introduce some additional parameters. Later we will comment on how to estimate them. Let

$s =$ *Prob (any given* MOVE *by the user is within its current LSTP area)*

We can examine the *FwdMOVE* algorithm line by line and write down a cost for setting up a forwarding pointer as follows:

$$S = s(V_U + A_l + L + A_l + V_U + A_l + L + A_l) + (1 - s)(V_U + A_l + L + D + R + D + L + A_l + V_U + A_l + L + D + R + D + L + A_l)$$

$$= 4A_l + 2(2 - s)L + 4(1 - s)D + 2(1 - s)R + 2V_U. \qquad (19)$$

Similarly, from *FwdFIND*, we can express T, and from *BasicMOVE* we can express m.

$$T = \frac{S}{2} - 2V_U + V_Q, \qquad (20)$$

$$m = 4(A_l + L + D + R + A_r) + H_U + V_U. \qquad (21)$$

Collecting terms and substituting Eq. 19, 20 and 21 into Eq. 18, we have $C_F < C_B$ iff $\Delta C > 0$, where

$$\Delta C = (4Y + 4Z)A_l + [2Y(2 - s) + 4Z]L + [4Y(1 - s) + 4Z]D + [2Y(1 - s) + 4Z]R + [2Y + Z - 2(pK - X)]V_U + (pK - X)V_Q + 4ZA_r + ZH_u \qquad (22)$$

and $Y = p + pK/2 - 3X/2$, $Z = X - p$. In order to get an intuitive understanding for this formula, we consider situations for which only one of the cost terms above dominates, and calculate the conditions when $C_F < C_B$ (See Table 1.)

In three of the situations in Table 1, it is not clear if forwarding is worthwhile or not. To investigate this, we attempt to quantify s. To estimate s, we need a model of the called user's mobility across LSTP areas. We use the following simple model. Assume that an LSTP area consists of $x \times x$ RA's arranged in a square, and each RA is itself a square. Users are assumed to be uniformly distributed throughout the LSTP area. Furthermore, each time a user leaves an RA, one of the four sides is crossed with equal probability. Then,

Table 1
Total cost / benefit evaluation of forwarding for various dominant cost terms.

Dominant term	Condition when $C_F < C_B$	Comments
Signaling links:		
A_l	$p < 0$	Never use forwarding
D	$(3s - 1)[(1 + p)^K - 1] + (1 - s)pK - 2sp > 0$	See following table
A_r	$p > 0$	Always use forwarding
Switch processing:		
L	$(3s - 2)[(1 + p)^K - 1] + (2 - s)pK - 2sp > 0$	See following table
R	$(3s + 1)[(1 + p)^K - 1] + (1 - s)pK - 2(1 + s)p > 0$	See following table
Database costs:		
H_U	$p > 0$	Always use forwarding
V_U	$K < 1$	Never use forwarding
V_Q	$p < 0$	Never use forwarding

Table 2
Total cost / benefit evaluation of forwarding with simplifying assumptions.

Dominant term	Condition when $C_F < C_B$	Comments
D	$(1+p)^K - 1 + (0.081K - 1.081)p > 0$	Always use forwarding
L	$(1+p)^K - 1 + (1.85K - 2.85)p > 0$	Always use forwarding
R	$(1+p)^K - 1 + (0.036K - 1.036)p > 0$	Always use forwarding

Prob[user crosses an LSTP area boundary]
= Prob[user is in border RA of LSTP area]
× Prob[user movement crosses LSTP area boundary]
$= 1/x$.

Thus, under these assumptions, $s = 1 - 1/x$. We now try to estimate a value for s for a typical regional telephone company region. There are 160 Local Access Transport Area (LATA) over seven regional telephone company regions, and typically there are 1250 SSP per region [3]. Assuming one LSTP per LATA, the number of SSPs per LSTP is $\frac{1250 \times 7}{160}$. Now, assuming each SSP corresponds to a single RA, $x = \sqrt{1250 \times 7/160} \approx 7.4$, and therefore $s \approx 0.87$. Table 2 shows the evaluation of forwarding using these estimates.

We see from Tables 1 and 2 that under the assumptions of our analysis, forwarding results in reduced mean loads for all network elements, except for the local A-link and the VLR. Informally, the explanation for this is that with the forwarding strategy the reduced load on the HLR is obtained in exchange for increased load on the VLRs, and reduced traffic on the D-links and RSTP is obtained in exchange for increased traffic at lower levels of the CCS network. Thus the forwarding strategy serves to redistribute the load to portions of the network where excess capacity may be more likely to exist (e.g. SSP and A-links).

Observe that when the pointer chain is truncated, the previous pointers may be deleted to save memory at the VLRs. Doing this will only modify the cost associated with *FwdMOVE( )*, but does not affect our performance analysis.

6. Discussion

Here we only sketch some of the issues involved in the use of the forwarding strategy.

6.1. Alternative network architectures

The reference architecture we have assumed (Fig. 1) is only one of several possible architectures. It is possible to consider variations in the placement of the HLR and VLR functionality, (e.g., placing the VLR at a Local SCP associated with the LSTP instead of at the SSP), the number of SSPs served by an LSTP, the number of HLRs deployed, etc. Rather than consider many possible variations of the architecture, we have selected a reference architecture to illustrate the new auxiliary strategy and our method of calculating their costs and benefits. Changes in the architecture may result in minor variations in our analysis but may not significantly affect our qualitative conclusions.

6.2. User terminal capabilities for forwarding

Note that the *FwdMOVE* procedure requires that the user's equipment inform the new VLR of the registration area from which the user just arrived. We stress three points related to this process:

1. The user terminal equipment must have the capability of storing this information. We observe that such a capability is implicitly required in the terminal registration procedure specified in the PCS Network and Operations Plan [2], explicitly mentioned in the Wireless Access Communications Systems (WACS) Technical Advisory [4] as well as required in the European GSM standard [17].
2. The user terminal equipment need not use SS7 address information for this purpose. We observe that typically the radio port periodically broadcasts a registration area identifier which can be used for this purpose.
3. Current air interface protocols do not support the relaying of this information to the network. Thus new protocols (new messages or message elements) would have to be designed to support the relaying of this information.

The user equipment also needs to inform the new VLR of the number of RA crossings it has performed, which uses this to determine whether a basic move is required. (Note that in IS-41 this information is provided to the VLR for security purposes anyway [6].)

6.3. CMR and its estimation

It is important to note that the CMR as defined in this paper is only an average measure. In fact, good performance can be expected with a large average CMR in many realistic situations. For example, consider a commuter who commutes from his home, through several towns, to his work place in a different town. These towns may be covered by different RAs. Most calls occur in the working area (which is covered by an RA). Suppose a user crosses 9 RAs (without receiving any calls) to the working area and receives 50 calls in that area. With $K = 4$, 7*BasicMOVE* operations are avoided, and only the first call traverses a pointer. In this case, the CMR of the user before the first call is zero, but averaged over all 50 calls, CMR = 5.5. However, forwarding has still resulted in substantial benefits.

We sketch some methods of estimating the CMR. A simple and attractive policy is to not estimate CMR on a per-user basis at all. For instance, if it is known that the average CMR over all users in a PCS system is low enough, then forwarding could be used at every SSP to yield net system-wide reductions in total network costs.

One possibility for deciding about forwarding on a per-user basis is to calculate running estimates of users' CMRs at each SSP. We have discussed two schemes for doing so in [19]. Another possibility is to maintain information about a user's calling and mobility pattern at the HLR, and download it periodically to selected SSPs during off-peak hours. It is easy to envision numerous variations on this idea. In general, schemes for estimating the CMR range from static to dynamic, and distributed to centralized. Evaluating their effectiveness depends upon numerous criteria outside the scope of this paper.

6.4. Implicit compression of pointer chains

A realistic implementation of the forwarding scheme will take into account the possibility that loops may form as the user visits several RAs in succession. Thus if a user revisits an RA, and a forwarding pointer for that user is found in the VLR at that RA, then it is possible to delete the old forwarding pointer so as to avoid a *FwdFIND* operation needlessly having to traverse a loop. We call this "implicit pointer compression", to distinguish it from the explicit pointer chain truncation which is performed when there are more than $K - 1$ pointers in the chain.

For example, consider a user whose movements are restricted to an 8×8 grid of RAs as shown in Fig. 6. Let q be the number of *MOVE*s between two consecutive phone calls, and $n = q \,\mathbf{mod}\, K$ be the number of pointers which would be traversed by a *FwdFIND* operation if loops in the movement pattern were ignored. Then let $\Theta(n)$ denote the number of pointers actually traced in the *FwdFIND* operation. The figure illustrates a move sequence ($q = 18$ moves) from $R0$ to $R1$. If $K = 10$ then $n = 8$, $\Theta(8) = 2$ for the move sequence in Fig. 6. With the function Θ, we can write an expression for F' similar to that of Eq. (5):

$$F' = \sum_{i=0}^{\infty} \Theta\left(i - K\left\lfloor \frac{i}{K} \right\rfloor\right) T\alpha(i) + F. \qquad (23)$$

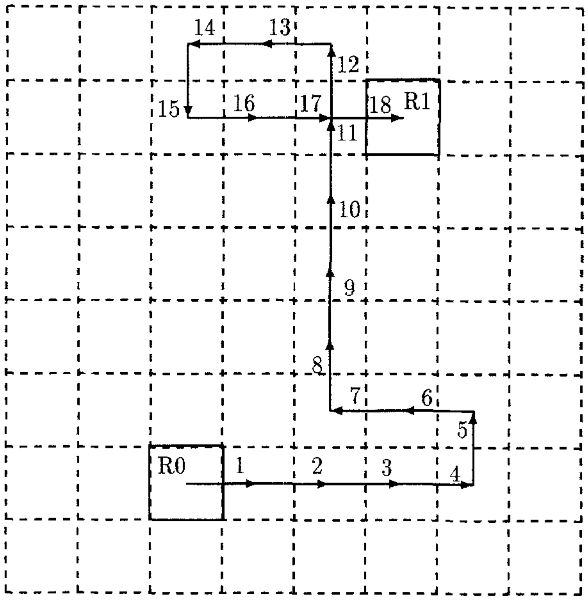


Fig. 6. A move sequence.

Thus Eq. (5) is a special case of Eq. (23), for the case where the user never revisits RAs, i.e., $\Theta(n) = n$. For a situation in which the user's movements are restricted (i.e., greater spatial locality of movement), we may expect to see some reduction in F'/F due to implicit pointer compression. Further investigation of this is outside the scope of the present paper.

6.5. Previous related work

As with the case of the per-user caching scheme presented in [19], forwarding is an example of a well-known technique used successfully in computer systems (e.g. [7]) applied to telecommunications networks.

For mobile communication networks in particular, forwarding pointers have been used in the work of Ioannidis et al. [8] and Awerbuch and Peleg [1]. The work by Ioannidis et al. (and succeeding work e.g. [5]) is focused on IP-based protocols and has different architectural assumptions. Awerbuch and Peleg present a theoretical model for online tracking of mobile hosts in which user locations are stored in a hierarchy of sets of storage locations, called matching directories, which are augmented by the use of forwarding pointers used to track the mobile host. The paper mainly considers the scalability of the technique in terms of the communication complexity. The work presented in [1] also differs from ours in terms of the architectural assumptions. Our forwarding protocol is a modification of those presented in the IS-41 and GSM standard protocols. Our analysis considers the probability distributions of call arrivals and user movements and presents detailed cost and benefit comparisons, rather than a scalability evaluation.

7. Conclusions

The forwarding strategy serves to redistribute the total costs of user location from the HLR and higher levels of the network to the VLR and lower portions of the network, where capacity may be more likely to exist (e.g. SSP and A-links). The forwarding strategy is very promising when the HLR update load, or the remote A-link, is the performance bottleneck (see Table 1). In terms of total cost, when the D-link, RSTP, and LSTP are the performance bottlenecks, the benefits of the forwarding strategy increase with s, the probability that a given *MOVE* does not lead the user outside an LSTP area; note that s is increased when there is locality in the user's mobility pattern and the number of registration areas served by an LSTP is fairly large. For our model of LSTP areas, the forwarding strategy results in net reductions of total costs for the D-link, RSTP and LSTP also (see Table 2).

Depending upon the costs of traversing and setting up forwarding pointers relative to those of *MOVE* and *FIND* in the basic strategy, if pointer chains are kept short ($K < 5$), forwarding for users with low CMR ($CMR < 0.5$) can result in substantial reductions in total user location costs. The total cost reduction is due to significant saving in location update and the low penalty of call setup for infrequent incoming calls. Note that the *per-user* nature of this forwarding scheme means that the maximum penalty in mean call setup time is not only paid infrequently, but only by users who receive calls infrequently; this tradeoff may be an acceptable price to pay for overall reductions in the network loads incurred in serving such users.

Acknowledgement

We thank Charles Lo for many enlightening and stimulating discussions. We also thank S. Mohan, A. Knapp, S. Levenson, J. Rizzo, and R. Wolff for their helpful comments.

Appendixes

A. Derivation of expressions for M' and F'

In this Appendix we derive Eqs. (6) and (7). With the assumptions in section 6, Eq. (4) can be written as

$$M' = \sum_{i=0}^{\infty} \left[\left(i - \left\lfloor \frac{i}{K} \right\rfloor \right) S + \left\lfloor \frac{i}{K} \right\rfloor m \right] \alpha(i) = X + Y \,, \tag{24}$$

where

$$X = \sum_{i=0}^{\infty} iS\alpha(i) \tag{25}$$

and

$$Y = \sum_{i=0}^{\infty} \left\lfloor \frac{i}{K} \right\rfloor (m - S)\alpha(i) \,. \tag{26}$$

Now Eq. (25) can be simplified from the definition of $\alpha(i)$:

$$X = S \sum_{i=0}^{\infty} i\alpha(i) = \frac{S}{p} \,. \tag{27}$$

To simplify Eq. (26), we need an expression for $\alpha(i)$. The probability $\alpha(i)$ is expressed as (see [20] for the detailed derivation)

$$\alpha(i) = \frac{(1 - g^2) g^{i-1}}{p} \,. \tag{28}$$

Let $i = jK + k$; Eq. (26) is re-written as

$$Y = \sum_{j=0}^{\infty} \sum_{k=0}^{K-1} j(m - S)\alpha(jK + k) \,. \tag{29}$$

From Eq. (28), we have

$$\alpha(jK + k) = yz^j x^k \,, \tag{30}$$

where

$$y = \frac{(1 - g)^2}{pg} \,, \quad z = g^K \,, \quad \text{and} \quad x = g \,.$$

Replacing α by Eq. (30), Eq. (29) is re-written as

$$\begin{aligned} Y &= \sum_{j=0}^{\infty} \sum_{k=0}^{K-1} j(m - S) yz^j x^k \\ &= \frac{y(m - S)(1 - x^K)}{1 - x} \left(\sum_{j=0}^{\infty} jz^j \right) \\ &= \frac{yz(1 - x^K)(m - S)}{(1 - x)(1 - z)^2} \\ &= \frac{(1 - g) g^{K-1} (m - S)}{p(1 - g^K)} \,. \end{aligned} \tag{31}$$

From Eqs. (27) and (31), Eq. (24) is re-written as

$$M' = \frac{S}{p} + \frac{(1 - g) g^{K-1} (m - S)}{p(1 - g^K)} \,. \tag{32}$$

Similarly, we now derive Eq. (7), by starting with Eq. (5).

$$\begin{aligned} F' &= F + \sum_{i=0}^{\infty} \left(i - K \left\lfloor \frac{i}{K} \right\rfloor \right) T\alpha(i) \\ &= F + \sum_{j=0}^{\infty} \sum_{k=0}^{K-1} kT\alpha(jK + k) \\ &= F + \sum_{j=0}^{\infty} Tyz^j \left(\sum_{k=0}^{K-1} kx^k \right) \end{aligned}$$

$$= F + \frac{(1-g^2)T}{pg(1-g^K)}\left(\sum_{k=0}^{K-1} kg^k\right)$$

$$= F + \frac{[1 - Kg^{K-1} + (K-1)g^K]T}{p(1-g^K)} . \qquad (33)$$

B. Derivation of upper and lower bounds

In this appendix we derive upper and lower bounds for F'/F and C_F/C_B when the residence time is exponentially distributed, i.e., we derive Eq. (14) and (15). Assuming a Poisson residence time distribution, from Eq. (10) and Eq. (2),

$$\frac{C_F}{C_B} = \frac{F + \frac{T+S}{p} + \frac{m-S-KT}{(1+p)^K-1}}{F + \frac{m}{p}}$$

$$= \frac{pF + T + S + \frac{p(m-S-KT)}{\sum_{i=0}^{K}\binom{K}{i}p^i-1}}{pF+m}$$

$$= \frac{pF + T + S + \frac{m-S-KT}{\sum_{i=1}^{K}\binom{K}{i}p^{i-1}}}{pF+m} . \qquad (34)$$

Consider the case when the CMR p approaches 0. From Eq. (34),

$$\lim_{p\to 0}\frac{C_F}{C_B} = \frac{T + S + \frac{m-S-KT}{K}}{m}$$

$$= \frac{1}{K} + \frac{(K-1)S}{mK} . \qquad (35)$$

When $p = 0$, the *FIND* operation is never performed. The costs of the two schemes are for the *MOVE* operations. Thus, we have the intuitive result that Eq. (35) is equal to M'/M.

For the upper bound, we first consider the case when p approaches ∞. We have, from Eq. (34),

$$\lim_{p\to\infty}\frac{C_F}{C_B} = 1 .$$

In this case, the user never leaves an RA (there is no *MOVE* operation), and the *FIND* operations for the both schemes are the same.

For values of p less than infinity, we see that two cases arise. If $m > S + KT$, then

$$\frac{m-S-KT}{\sum_{i=1}^{K}\binom{K}{i}p^{i-1}} \le \frac{m-S-KT}{K}$$

and Eq. (34) implies that

$$\frac{C_F}{C_B} \le \frac{pF + T + S + \frac{m-S-KT}{K}}{pF+m}$$

$$= 1 + \frac{(K-1)(S-m)}{K(pF+m)}$$

$$\le 1 . \qquad (36)$$

On the other hand, if $m \le S + KT$, then

$$\frac{m-S-KT}{\sum_{i=1}^{K}\binom{K}{i}p^{i-1}} \le 0$$

and Eq. (34) implies that

$$\frac{C_F}{C_B} \le \frac{pF+T+S}{pF+m}$$

$$= 1 + \frac{T+S-m}{pF+m} . \qquad (37)$$

If $T + S \le m$ then Eq. (37) implies that

$$\frac{C_F}{C_B} \le 1 . \qquad (38)$$

If $T + S > m$ then since $pF > 0$, Eq. (37) implies that

$$\frac{C_F}{C_B} \le 1 + \frac{T+S-m}{m} = \frac{T+S}{m} . \qquad (39)$$

From Eqs. (35), (36), (38) and (39), we have for all p values

$$\frac{1}{K} + \frac{(K-1)S}{Km} \le \frac{C_F}{C_B} \le \max\left(1, \frac{T+S}{m}\right) .$$

References

[1] B. Awerbuch and D. Peleg, Concurrent online tracking of mobile users, *Proc. SIGCOMM Symp. Comm. Arch. Prot.* (1991).

[2] Bellcore, Network and Operations Plan for access services to Personal Communications Services (PCS) providers, Special Report SR-TSV-002402, Bellcore (1992).

[3] Bellcore, Switching system requirements for interexchange carrier interconnection using the integrated services digital network user part (IS-DNUP), Technical Reference TR-NWT-000394, Bellcore (1992).

[4] Bellcore, Generic criteria for version 0.1 wireless access communications systems (WACS) and supplement, Technical Report TR-INS-001313, Issue 1, Bellcore (1993).

[5] P. Bhagwat and C. Perkins, A mobile networking system based on Internet Protocol (IP), *Proc. USENIX Symp. Mobile and Location-Independent Comput*, USENIX Assoc. (1993).

[6] EIA/TIA, Cellular radiotelecommunications intersystem operations, Technical Report IS-41 (Revision B), EIA/TIA (1991).

[7] R.J. Fowler, The complexity of using forwarding address for decentralized object finding, *Proc. SIGCOMM Symp. Comm. Arch. Prot.* (1986).

[8] J. Ioannidis, D. Duchamp and G.Q. Maguire Jr., IP-based protocols for mobile internetworking, *Proc. SIGCOMM Symp. Comm. Arch. Prot.* (1991) p. 234–245.

[9] R. Jain, A classification scheme for user location strategies in personal communications services systems, submitted for publication.

[10] C.N. Lo, R.S. Wolff and R.C. Bernhardt, An estimate of network database transaction volume to support personal communications services, *Proc. Int. Conf. Univ. Pers. Comm.* (1992).

[11] C. Lo, S. Mohan and R. Wolff, Performance modeling and simulation of data management for personal communications applications, *Proc. 2nd Bellcore Symp. Perf. Modeling* (1992).

[12] C. Lo and R. Wolff, Estimated network database transaction volume to support wireless personal data communications applications, *Proc. Int. Conf. Comm.* (1993).

[13] E. Lycksell, GSM system overview. Technical Report, Swedish Telecom. Admin. (1991).

[14] K. Meier-Hellstern and E. Alonso, The use of SS7 and GSM to support high density personal communications, *Proc. Int. Conf. Comm.* (1992).

[15] A.R. Modaressi and R.A. Skoog, Signalling System No. 7: A tutorial, IEEE Commun. Mag. (July 1990) 19–35.

[16] S. Mohan and R. Jain, Two user location strategies for PCS, IEEE Pers. Commun. Mag. 1 (1994) 42–50.

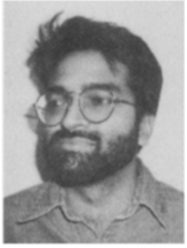
[17] M. Mouly and M.B. Pautet, *The GSM System for Mobile Communications*, M. Mouly, 49 rue Louise Bruneau, Palaiseau, France (1992).

[18] N.L. Johnson and S. Kotz, *Continuous Univariate Distributions-1* (Wiley, 1970).

[19] R. Jain, Y.-B. Lin, C. Lo and S. Mohan, A caching strategy to reduce network impacts of PCS, IEEE J. Select. Areas Commun. 12 (1994) 1434–1444.

[20] Y.-B. Lin, S. Mohan and A. Noerpel, Queueing priority channel assignment strategies for handoff and initial access for a PCS network, IEEE Trans. Veh. Tech. 43 (1994) 704–712.

[21] H. Harjono, R. Jain and S. Mohan, Simulation modeling of a caching strategy for locating mobile PCS users, Int. J. Comp. Simulation (1996), to appear.



Ravi Jain received the Ph. D. in computer science from the University of Texas at Austin in 1992. Prior to his Ph. D. degree he worked at Syntrex Inc., SES Inc. and the Schlumberger Laboratory for Computer Science on developing communications and systems software, performance modeling, and parallel programming. In 1992 he joined Bellcore, where his research interests include design and analysis of algorithms and techniques for efficient resource management, focusing on reducing network impacts of supporting PCS, mobile information access, and Intelligent Vehicle-Highway Systems (IVHS) applications such as the SCOUT traveler information system. His activities also include research in mobile and parallel computing. He is a guest editor for a special issue of *Distributed and Parallel Databases* on Databases and Mobile Computing, and he is co-chair of the annual Workshop on Input/Output in Parallel and Distributed Systems held in conjunction with the International Parallel Processing Symposium. Jain has several publications in the area of communications and parallel computing and is a member of the Upsilon Pi Epsilon and Phi Kappa Phi honorary societies, as well as ACM and IEEE.

Yi-Bing Lin received his BSEE degree from National Cheng Kung University in 1983, and his Ph.D. degree in computer science from the University of Washington in 1990. Since then, he has been with the Applied Research Area at Bell Communications Research (Bellcore), Morristown, NJ. His current research interests include design and analysis of personal communications services network, distributed simulation, and performance modeling. He is a subject area editor of the Journal of Parallel and Distributed Computing, an associate editor of the International Journal in Computer Simulation, an associate editor of SIMULATION magazine, a member of the editorial board of International Journal of Communications, a member of the editorial board of Computer Simulation Modeling and Analysis, Program Chair for the 8th Workshop on Distributed and Parallel Simulation, and General Chair for the 9th Workshop on Distributed and Parallel Simulation. Dr. Lin is an Adjunct Research Fellow at the Center for Telecommunications Research, National Chiao-Tung University, Taiwan, R. O. C.