A Contention-Free Mobility Management Scheme Based on Probabilistic Paging

Wing Ho A. Yuen and Wing Shing Wong, *Senior Member, IEEE*

***Abstract*—In forthcoming personal communication systems (PCSs), small cells are deployed to achieve high spectral efficiency. This has significant impacts on location tracking of mobile users. The increase in location update (LU) load leads to more contention on the reverse control channel. Thus, many algorithms are designed to distribute the LU load to a larger number of cells. This avoids the inefficiency of random accessing due to high offered load. In an alternative approach [3], a contention-free LU algorithm is proposed. Two or more mobile units are permitted to register with a base station simultaneously without contention. A probabilistic paging mechanism called Bloom filtering is used to select cells to be paged. Since there is no contention in LU, inefficiencies due to random accessing are bypassed. In this paper, we present another contention-free LU algorithm. It is hybrid in the sense that LUs are temporally or geographically triggered. The use of hybrid LU alleviates inefficiencies inherent to temporal triggered LU in [3]. Three selective paging schemes are considered in this paper. Tradeoff between paging delay and paging bandwidth is addressed. The performance of this algorithm is compared to [3] and other conventional strategies. Numerical results shows that the new algorithm compares favorably with previous proposed strategies.**

I. Introduction

In personal communication systems (PCSs), microcells and picocells are extensively deployed. This allows power savings in base-station transceivers and mobile handsets. More important, spectral efficiency is enhanced through denser spatial reuse of frequency. However, the use of small cells poses a severe burden on the common air interface and the signalling network. On the reverse control channel, higher contention of location update (LU) messages is expected. Retransmission of LU messages is needed. This leads to a large increase in signalling traffic on the radio link.

The geographic-based strategy is currently adopted by the wireless standards [GSM mobile application part (MAP) and IS-54 [4]]. The whole network is partitioned into nonoverlapping location areas. Each mobile unit (MU) monitors the forward control channel of the local cell for its cell identifier, which is broadcasted periodically. When it detects that the local cell has fallen out of the original location area, an LU is triggered. Thus, when a call is terminated to an MU, the network deterministically retrieves the location of the called MU at a resolution of a single location area.

The geographic-based strategy performs well in second-generation cellular networks. However, the efficiency of geographic LU deteriorates dramatically when small cells are deployed. This is because the LU load is shared by the boundary cells of a location area only. The high offered load in each cell adversely affects the probability of a successful transmission. Thus the geographic-based strategy is not an ideal candidate for forthcoming systems.

Many strategies have been proposed in literature to tackle the problem. A convenient way of classifying these strategies is outlined below.

1) For systems employing a *static* strategy, the location areas are globally defined for all MUs. Each MU has the same set of location areas irrespective of its mobility and call arrival behavior. The geographic-based strategy is an example of a static strategy.
2) In *semistatic* strategies [10], [12], location areas are also predefined. The system consists of overlapping layers of location areas. When an MU exits a location area of the present layer, it switches to another layer that matches to its mobility profile. Note that the strategy proposed in [8] is classified as a static strategy under this definition, for the reason that overlapping layers of location areas are used to provide hysteresis against frequent switching between location areas, rather than match to specific mobility profiles of MUs.
3) In *dynamic* strategies, each MU has a location area tailored for its mobility profile. No permanent location area boundaries are defined for any MU. The well-known distance-based strategy [5] and its variants [11], [14] belong to this category. Since location area sizes for dynamic strategies are tailored for individual users, these algorithms in general outperform static strategies. However, as noted in [15], these strategies are amenable to implementation only when the size of the location area is small.

In literature, location tracking algorithms are engineered such that the combined LU and paging bandwidth is minimal. This is often accomplished through personalized assignment of location areas. Nevertheless, these algorithms still experience the same problem as the geographic-based strategy. The inefficiency associated with random access is only partially solved by distributing the LU load to a larger number of cells. An alternative approach to the problem is proposed in [3]. Instead of adopting random accessing on the reverse control channel, a contention-free algorithm for LU is proposed. Each cell maintains *cell vectors*, which summarize the location information of local mobile users. When a call is terminated to a mobile user, relevant location information in each cell is retrieved. A

Manuscript received May 5, 1999; revised April 25, 2000. This work was supported by the Hong Kong Research Grants Council under a Grant.

W. H. A. Yuen was with the Chinese University of Hong Kong. He is now with Wireless Information Network Laboratory (WINLAB), Rutgers—The State University of New Jersey, Piscataway, NJ 08854-8060 USA (e-mail: andyyuen@winlab.rutgers.edu).

W. S. Wong is with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: wswong@ie.cuhk.edu.hk).

Publisher Item Identifier S 0018-9545(01)01222-1.

0018–9545/01$10.00 © 2001 IEEE

probabilistic paging mechanism called Bloom filtering is used to select cells to be paged. Since there is no contention in LU, inefficiencies due to random accessing are bypassed.

In this paper, we propose another contention-free LU algorithm. It is hybrid in the sense that LUs are temporally or geographically triggered. The use of hybrid LU alleviates inefficiencies inherent to temporal triggered LU in [3]. With our LU scheme, paging (Bloom filtering) is more accurate. Performance of our algorithm compares favorably with [3] and conventional strategies.

The rest of this paper is organized as follows. In Section II, the system model is described. A contention-free LU scheme and the probabilistic paging mechanism are outlined in Section III. We refer to the algorithm as the time-based Bloom filter (TBBF) algorithm. Section IV describes our hybrid Bloom filter algorithm (HBF), which is analyzed in Section V. A numerical study is given in Section VI to compare the HBF with [3] and other conventional strategies. This is followed by a discussion in Section VII. For completeness, a simulation study is reported in the Appendix to verify the numerical results we obtained.

II. System Model

We consider a cellular system consisting of M cells and N MUs. The number of users in a cell is denoted by k. In steady state, the average value of k is assumed to be N/M. We assume that each cell is identically shaped. The actual shape could be arbitrary as long as the cells are packed regularly. The dwell time in a cell of each MU is assumed to be exponentially distributed with mean $1/\mu$. Individual call interarrival time is also exponentially distributed with mean $1/\lambda$. Since the interarrival time of calls is generally much larger than the period τ_g between two LUs, we assume that there can be at most one incoming call during one update cycle. By the memoryless property of exponentially distributed random variable, the call arrival time is uniformly distributed in the update cycle.

Every MU has a unique mobile unit identity vector (MUID) for unique identification during paging. It is also assigned an n-bit Bloom filter identity vector (BFID), which is used for LU and will be described in detail in Section III. Each bit of an arbitrary BFID is independent and is a "1" with probability p. Note that we do not require the BFIDs to be unique. We also assume that the maximum propagation delay of pulses from the MUs to a base station is much smaller than the bit time of an LU message. That is, there is no bit synchronization problem when LU is performed.

III. Contention-Free Location Update Algorithm

In [3], the time-based Bloom filter algorithm is presented. It belongs to a category of LU algorithms characterized by contention-free access of the reverse control channel during LU. Collision of LU messages is prevented.

Periodically, each MU performs an LU by sending its n-bit BFID to the local cell. When the corresponding bit in the BFID is a "1," a pulse is sent. Otherwise, the MU sends nothing. A base station receives the superimposition of pulses from all MUs in the local cell and store it as a *cell vector*. If the base station detects one or more pulses in the ith bit interval, it infers that at least one MU in the cell contains a "1" in the ith bit. On the other hand, if the base station detects no pulse in the ith bit, it concludes that all MUs inside the cell have BFIDs that are zeros at the ith bit. The zero in the ith bit in the cell vector conveys significant information. One could infer that a MU with an ID of "1" in the ith bit is not in the cell.

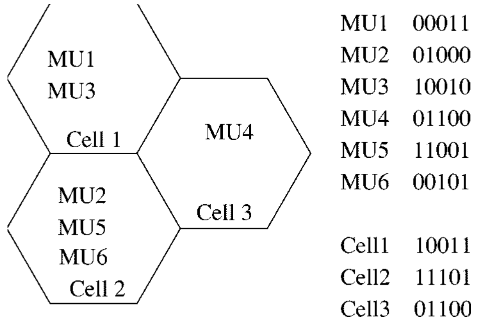

Fig. 1. Illustration of contention-free LU and Bloom filtering.

Consider a system consisting of three cells and six MUs as depicted in Fig. 1. During the most recent LU cycle, the cell vectors of cell 1 to cell 3 are 10011, 11101, and 01 100, respectively. Suppose a call is terminated to MU 6. We compare the BFID of MU 6 with the cell vectors. Since the third and the last bit of MU 6 are one, we should send a paging message to the cell only if the corresponding bits of the cell vectors are also "1." It turns out that only cell 2 should receive the paging message.

Mathematically, we denote l_i, $i \in [1, M]$, as the cell vector obtained in the most recent LU cycle in cell i from a system of M cells. When a call arrives for the MU with ID BFID, cell j is paged if

$$\text{BFID} \otimes l_j = \text{BFID}$$

where $\otimes$ denotes bit by bit multiplication. This operation is called Bloom filtering [6]. Only the cells whose cell vectors match to the targeted MU will be paged. After the network "filter" out cells to be paged, the MUID of the called MU is then broadcasted on the paging channel of the matched cells.

Suppose there are k MUs in a cell. The probability that an arbitrary bit in the cell vector is "0" is $q = (1-p)^k$. The cell is not paged if the corresponding bit in the BFID is a "1." This occurs with probability pq. Thus, on average, a cell is paged with probability

$$(1-pq)^n. \tag{1}$$

It is possible that the targeted MU is not located in the paged cells. This happens when the targeted MU leaves a cell between the call arrival time and the most recent LU cycle. Denote the probability of successful and unsuccessful paging as $P_{C,L}$ and $P_{L,C}$, respectively. Denote the period of an LU cycle by τ_g. Assume that the dwell time of an MU and the call interarrival time are exponentially distributed with mean $1/\mu$ and $1/\lambda$, respectively. Since $\tau_g \ll 1/\lambda$ in general, we further assume that at most one call arrives between two update cycles. By the memoryless property of an exponentially distributed random variable,

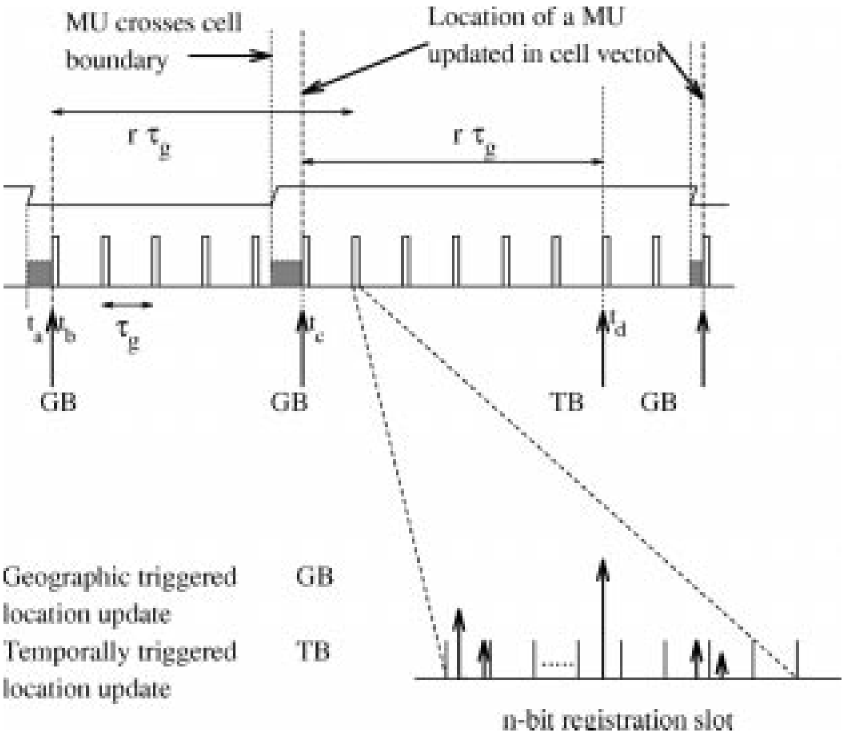
MU crosses cell boundary
Location of a MU updated in cell vector
r τg
r τg
ta
tb
τg
tc
td
GB
GB
TB
GB
Geographic triggered location update
GB
Temporally triggered location update
TB
.....
n-bit registration slot

Fig. 2. Illustration of LU on uplink control channel for an arbitrary MU.

the call arrival time is uniformly distributed in the time interval of τ_g. It is straightforward to show that

$$P_{C,L} = \frac{1 - e^{-\mu\tau_g}}{\mu\tau_g} \tag{2}$$

$$P_{L,C} = 1 - P_{C,L}. \tag{3}$$

In the TBBF algorithm, each MU periodically reports its BFID to the local cell. The large number of superimposed LUs a cell receives increases the probability that this cell will be paged when a call arrives. This in turn increases the system paging cost. It is desirable to reduce the LU rate of each MU promote more accurate paging. With this insight, we propose a hybrid LU strategy in which LU could be temporally or geographically triggered. We limit the rate of updates when an MU is less mobile. This promotes efficient paging and leads to lower control overhead ultimately in an optimized system.

IV. Hybrid Bloom Filter Algorithm

In this section, we present a new, hybrid location update protocol for MUs and a choice of algorithms that can be used by the fixed network in tracking the MUs.

A. Location Update Protocol

Under the new protocol, the reverse control channel periodically allocates *registration slots* with period τ_g. Only LU messages are allowed in these slots. All other control messages such as call setup negotiations can contend for the channel in other unreserved time slots. When an MU crosses a cell boundary at time t_a, it attempts to register to the new cell. However, it can do so only at the next registration slot. In the meantime, the system loses track of the MU temporarily, as shown by the shaded area in Fig. 2. At time t_b, the MU sends its BFID to the reverse control channel. If more than one MU enters the given cell since the last registration slot, the received cell vector is a superimposition of several BFIDs.

The registration information is held at the base station for a period of $r\tau_g$. If an MU remains in the same cell at the end of this period, it is preprogrammed to perform a temporally triggered LU reregister with the cell. If the cell does not receive any reregistration from an MU that registered at a time $r\tau_g$ before, the system infers that the MU has left the cell and deregisters it. Viewed from the base station, a bit in the cell vector will be reset if no pulse is received in the corresponding bit time for the r most recent registration slots.

We need to fix some notation to help our subsequent discussion. At any time t, label the beginning of the most recent registration slot by T_r, label the end of the immediately preceding registration slot by T_{r-1}, and so on. The cell vector at time T_i is denoted by S_i. We label the time interval $[T_1, T_r]$ as the *registration memory window* at time t.

B. Paging Protocol

When an MU is called, the system retrieves the most recent r cell vectors from each cell. These cell vectors are used to locate an MU via Bloom filtering. There is a finite probability $P_{L,C}$ that the targeted MU has just left the cell when the call arrives. In this casen the MU is lost and flooding is needed to locate it. With probability $P_{C,L} = 1 - P_{L,C}$n the targeted MU is still in the registered cell when the call arrives. This MU is registered in one of the r most recent cell vectors stored at the corresponding base station.

By the nature of the LU algorithm, any one of the r cell vectors provides usable information for the Bloom filtering operation. However, based on our mobility assumption, the probability of successfully locating the targeted MU is higher for more recent cell vectors (see Lemma 1). To exploit this property, we institute up to r *matching cycles* in paging. In each matching cycle, a group of consecutive cell vectors are used to decide whether the targeted MU is in the cell under consideration. The first matching cycle uses the most recently received group of cell vectors. The second matching cycle uses the next most recently received group, and so on. If the targeted MU is found, the matching cycle stops immediately. If the targeted MU cannot be identified with this process, the system will flood all yet unpaged cells in the system to locate it.

Three types of matching cycle groupings are considered here for comparison. In paging scheme 1, there is only one matching cycle. That is, all r cell vectors in each cell are used to match the BFID of the incoming MU. In paging scheme 2, there are two matching cycles. In the first pass, the w most recent cell vectors from each cell are used. In the second pass, the remaining $r-w$ vectors are used. In paging scheme 3, there are r matching cycles consisting of a single cell vector from each cell. The cycle starts with the most recent cell vector. In all three cases, if the targeted MU cannot be located after all the matching cycles, the remaining cells will be flooded.

V. Performance Evaluation

In order to compare different location update schemes, a common cost function needs to be identified. Since the radio bandwidth is a more critical resource than the bandwidth of a fixed network, we define the cost function to be a weighted sum of location update and paging data rate per MU per hour on the radio link. The expected value of the cost function for the proposed algorithm is derived in Theorem 1.

Assume a call arrives at time t and the targeted MU is located at cell 0. Consider the registration memory window at time t. Assume the targeted MU has not changed cell during the time interval $[T_r, t]$. Define $Z = t - T_r$. It follows from the basic assumption of the system model that Z is uniformly distributed between $[0, \tau_g]$. The targeted MU must register at least once with cell 0 during the registration memory window at time t. Let I be the most recent registration slot in the window when the targeted MU updates at cell 0. Denote the probability $I = i$ by P_i.

Let X_j be the time instants at which the targeted MU changes cells. Let t fall in the interval defined by X_j and X_{j+1}. By renewal theory, the *age* of the arrival defined by $T = t - X_j$ is known to be exponential distributed with mean μ [1]. With this assumption, P_i is derived in the following lemma.

Lemma 1:

$$P_i = P_{C,L}\,\frac{e^{-\mu(r-i)\tau_g} - e^{-\mu(r-i+1)\tau_g}}{1 - e^{-\mu r\tau_g}} \tag{4}$$

where

$$P_{C,L} = \frac{1 - e^{-\mu\tau_g}}{\mu\tau_g}. \tag{5}$$

Proof: The event $I = i$ is equivalent to having a geographically or temporally triggered registration at T_i. This in turn is equivalent to the age T falling in the intervals $[lr-i,\, lr-i+1]\tau_g$ for some $l = 1, 2, \ldots$ (refer to Fig. 3). Note that if $l = 1$, the registration is geographically triggered. It is temporally triggered if $l = 2, 3, \ldots$. Since T is exponential distributed with mean $1/\mu$, we have

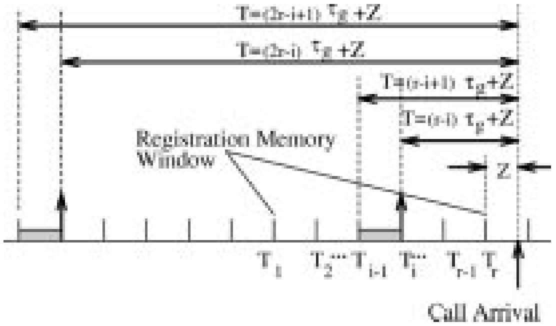

Fig. 3. When an MU is registered in slot i, the dwell time up to the call arrival may take the value in the interval $[(lr-i)\tau_g + Z,\, (lr-i+1)\tau_g + Z]$ for $l = 1, 2, \ldots$.

$$P_i = \sum_{l=1}^{\infty} P[(lr-i+1)\tau_g + Z > T > (lr-i)\tau_g + Z] \tag{6}$$

$$= \sum_{l=1}^{\infty} P[T > (lr-i)\tau_g + Z] - P[T > (lr-i+1)\tau_g + Z] \tag{7}$$

$$= \sum_{l=1}^{\infty} \int_0^{\tau_g} P[T > (lr-i)\tau_g + z]\frac{1}{\tau_g}\,dz - \int_0^{\tau_g} P[T > (lr-i+1)\tau_g + z]\frac{1}{\tau_g}\,dz \tag{8}$$

$$= \frac{1 - e^{-\mu\tau_g}}{\mu\tau_g}\sum_{l=1}^{\infty}\left\{e^{-\mu(lr-i)\tau_g} - e^{-\mu(lr-i+1)\tau_g}\right\} \tag{9}$$

$$= P_{C,L}\frac{e^{-\mu(r-i)\tau_g} - e^{-\mu(r-i+1)\tau_g}}{1 - e^{-\mu r\tau_g}}. \tag{10}$$

□

For any cell, define $P_l(i)$ as the probability that the ith cell vector matches to the BFID of the targeted MU. Define $P[\text{page}]_j$ to be the probability that a cell is paged when the paging scheme j is used.

Theorem 1: The combined paging and update radio bandwidth usage per MU per hour is given by the following equation:

$$C_{\text{HBF}} = \frac{c_1 nM}{\tau_g N} + c_2\lambda\log_2 N[P_{C,L}(P[\text{page}]M) + P_{L,C}M] \tag{11}$$

where $P[\text{page}]_j$ for the paging schemes are given by

$$P[\text{page}]_1 = 1 - \prod_{i=1}^{r}(1 - P_l(i)) \tag{12}$$

$$P[\text{page}]_2 = \left[1 - \prod_{i=r-w+1}^{r}(1 - P_l(i))\right] + \sum_{i=1}^{r-w} P_i \cdot \left[1 - \prod_{i=1}^{r-w}(1 - P_l(i))\right] \tag{13}$$

$$P[\text{page}]_3 = \sum_{i=1}^{r}\sum_{j=i}^{r} P_i P_l(j) \tag{14}$$

and

$$P_l(i) = (1 - p(1-p)^{v_i})^n$$
$$P_{C,L} = \frac{1 - e^{-\mu\tau_g}}{\mu\tau_g} \quad P_{L,C} = 1 - P_{C,L}. \tag{15}$$

Proof: Define v_i to be the number of MUs sending updates in registration slot i in cell 0. Since the BFID bits are generated independently, the probability that there is a match in a given bit between the BFID and the ith cell vector is $p(1-p)^{v_i}$. Therefore, the probability that a match is identified using the ith cell vector is given by $(1 - p(1-p)^{v_i})^n$.

In paging scheme 1, for any cell, all r cell vectors, S_i, $i \in [1, r]$, are used. The cell will be paged if there is a match with any one of the r cell vectors. Therefore

$$\begin{aligned} P[\text{page}]_1 &= P[\text{at least one cell vector match}] \\ &= 1 - \prod_{i=1}^{r}(1 - P_l(i)). \end{aligned} \tag{16}$$

In paging scheme 2, cell vectors from slot S_i, $i \in [r - w + 1, r]$ are used for matching in the first pass. With probability $\sum_{i=1}^{r-w} P_i$, the MU is not located in first matching cycle. Therefore, $P[page]$ is given by

$$\begin{aligned} P[page]_2 = &\left[1 - \prod_{i=r-w+1}^{r}(1 - P_l(i))\right] + P[\text{no match}] \\ &\cdot \left[1 - \prod_{i=1}^{r-w} 1 - (1 - P_l(i))\right] \end{aligned} \tag{17}$$

$$\begin{aligned} = &\left[1 - \prod_{i=r-w+1}^{r}(1 - P_l(i))\right] + \sum_{i=1}^{r-w} P_i \\ &\cdot \left[1 - \prod_{i=1}^{r-w}(1 - P_l(i))\right]. \end{aligned} \tag{18}$$

In paging scheme 3, a cell vector is used only if no match has been found with succeeding cell vectors. Therefore

$$\begin{aligned} P[\text{page}]_3 = &P_r P_l(r) + P_{r-1}(P_l(r) + P_l(r-1)) \\ &+ P_{r-2}(P_l(r) + P_l(r-1) + P_l(r-2)) + \cdots \end{aligned} \tag{19}$$

$$= \sum_{i=1}^{r}\sum_{j=i}^{r} P_i P_l(j). \tag{20}$$

□

It is obvious that $P[\text{page}]_1 > P[\text{page}]_2 > P[\text{page}]_3$ by our paging construction. Thus, paging bandwidth is conserved by trading off paging delay.

Since v_i, $i \in [1, r]$ are intricately interdependent random variables, it is intractable to compute cost C_{HBF} exactly. For the purpose of comparison with other algorithms in our numerical study section, we have chosen to use average values of v_i for the computation of C_{HBF}.

In steady state, all $\overline{v}_i$s are identical. We denote the average as $\overline{v}$. $\overline{v}$ is equal to sum of the mean number of temporally triggered registrations and the mean number of geographically triggered registrations. By symmetry, the mean number of incoming MUs in a cell during an update cycle is equal to the mean number of departures of MUs in the same cycle. Thus, the mean number of geographically triggered registrations is equal to

TABLE I
SYSTEM PARAMETERS ADOPTED IN THE NUMERICAL STUDY

System Parameter	Numerical value
Area of each cell	$0.25km^2$
Number of cells in the system	$M = 900$
Number of users in the system	$N = 2^{15}$
Avg. cell crossing rate per MU	$\mu = 50$ cells/hr
Avg. call arrival rate per MU	$\lambda = 1$ calls/hr
weight factor for LU cost	$c_1 = 5$
weight factor for paging cost	$c_2 = 1$

$$k\,\Pr[T < \tau_g] = k(1 - e^{-\mu\tau_g}). \tag{21}$$

The mean number of temporally triggered updates is given by

$$\overline{v}\,\Pr[T > r\tau_g] = \overline{v}e^{-\mu r\tau_g}. \tag{22}$$

Hence

$$\overline{v}e^{-\mu r\tau_g} + k(1 - e^{-\mu\tau_g}) = \overline{v}$$

or

$$\overline{v} = \frac{k(1 - e^{-\mu\tau_g})}{1 - e^{-\mu r\tau_g}}. \tag{23}$$

Substitute all v_i for $\overline{v}$. It follows that $P_l(i) = P_l$ for all i. The average probability of paging a cell for the proposed paging schemes are, respectively

$$P[\text{page}]_1 = 1 - (1 - P_l)^r \tag{24}$$

$$P[\text{page}]_2 = [1 - (1 - P_l)^w] + \sum_{i=1}^{r-w} P_i[1 - (1 - P_l)^{r-w}] \tag{25}$$

$$P[\text{page}]_3 = \sum_{i=1}^{r} P_i(r + 1 - i)P_l. \tag{26}$$

Suppose the length of a BFID is n, the period of a registration slot is τ_g, and p is the same for both hybrid (HBF) and time-based Bloom filter (TBBF) schemes. Note that when $r = 1$, the hybrid algorithm degenerates to the time-based Bloom filter algorithm. This is obvious since every MU must register in every registration slot. The number of update messages expected in each registration slot v equals the number of MUs in each cell k. The probability of paging a cell $P[\text{page}]_1$ when $r = 1$ is thus given by

$$\begin{aligned} 1 - (1 - P_l)^1 &= P_l \\ &= [1 - p(1-p)^v]^n \\ &= [1 - p(1-p)^k]^n \end{aligned} \tag{27}$$

which agrees to (1).

TABLE II
OPTIMIZED PARAMETERS VALUES FOR VARIOUS STRATEGIES

Strategy	τ	n	r	w	m	p
TB	∞	-	-	-	-	-
GB	-	-	-	-	$\boxed{\left(\frac{c_1 c_3 \mu}{2 c_2 \lambda}\right)^{\frac{1}{3}}}$	-
TBBF	0.01190701	208	-	-	-	$\frac{1}{1+k}$
HBF paging scheme 1	0.004227	112	13	-	-	$\frac{1}{1+v}$
HBF paging scheme 2	0.00413976	98	15	5	-	$\frac{1}{1+v}$
HBF paging scheme 3	0.00367644	80	31	-	-	$\frac{1}{1+v}$

Assume the mobile population consists of low-mobility MUs, that is, $\mu\tau_g \ll 1$. In the case of a hybrid update scheme, each MU performs mostly temporally triggered LUs. That means that on average, an inactive MU will need to register only once every r registration slots. Battery power in the mobile handset can be saved.

Since the rate of LU decreases, the average number of LUs in a registration slot $\overline{v}$ also decreases. The number of expected updates per registration slot v is given by

$$\begin{aligned}\overline{v} &= \frac{k(1-e^{-\mu\tau_g})}{1-e^{-\mu r\tau_g}}\\ &\approx \frac{k(1-1+\mu\tau_g)}{1-1+r\mu\tau_g}\\ &= \frac{k}{r}. \end{aligned} \tag{28}$$

Thus the expected number of LUs received per registration slot is r times smaller than the time-based Bloom filter algorithm. As the number of MUs required to register decreases, there are more zero bits in cell vectors. This decreases the probability of a match during the Bloom filtering operation, which in turn improves the accuracy in paging. For this reason, the hybrid algorithm outperforms the time-based counterpart when the average mobility μ is low.

VI. NUMERICAL STUDIES

In this section, we present a numerical study to compare the performance of four location tracking strategies. We consider the time-based (TB) and geographic-based (GB) strategies, the hybrid Bloom filter algorithm with its three paging schemes, and the time-based Bloom filter strategy. The system parameters adopted in our study are shown in Table I.

In practice, in order to conserve power usage in the MUs, preference is given to minimizing LU traffic. The weighted cost should bias heavily against LU. As a rough estimate, the weight factors c_1 and c_2 in the cost function are chosen to be five and one, respectively, in this study.

For the GB strategy, we add an extra penalty factor c_3 in the update component of the cost function. This account for conflicts due to random uplink access, packet overheads, and switching instability due to fading phenomenon. Since the throughput of random access is limited to at most 1/2e, the LU bandwidth is weighted by a factor of 2e. The exact value of packet overhead varies from applications to applications. Based on the GSM specifications [2], it is justified to assume that packet overhead is the same size as the LU payload. This adds another factor of two in the location update cost. Finally, we also assume that there are undesirable switching of location areas due to fading phenomenon. We assume this occurs with probability 0.05. Combining all effects, a factor of $c_3 = 2(2e)/0.95$ is weighted to the LU cost for the geographic based strategy.

In the case of TB strategy, there are also collisions due to random access. However, there are no synchronization overheads in this case since each MU updates periodically. Also, there are no extraneous LUs due to switching instability. Thus, a factor $c_4 = 2e$ is weighted to the LU cost for the time-based strategy.

In the case of Bloom filter algorithms (TBBF and HBF), no inefficiency arises from contention or switching instability. There are also no packet overheads once synchronization is acquired. Thus, there is no penalty factor in the LU component.

The cost functions for GB, TB, and TBBF algorithms are, respectively, given in (29)–(31). Note that αm^2 is the number of cells in a location area. The value of α depends on the cell shape. For illustration purpose, we assume the network consists of square cells in this study. That is, α is set to one. It is straightforward to derive these cost functions. The left terms represent LU data rate and the right terms represent paging rate

$$C_{\mathrm{GB}} = \frac{c_1 c_3 \mu \log_2(N)}{m} + c_2 \lambda \alpha m^2 \log_2(N) \tag{29}$$

$$C_{\mathrm{TB}} = \frac{c_1 c_4 \log_2(N)}{\tau_t} + c_2 \lambda \log_2(N)[P_{L,C} M + P_{C,L}] \tag{30}$$

$$\begin{aligned} C_{\mathrm{TBBF}} = \frac{c_1 n M}{\tau_t N} &+ c_2 \lambda \log_2(N)\{P_{L,C} M + P_{C,L} \\ &\cdot \lceil M(1-p(1-p)^k)^n \rceil\}. \end{aligned} \tag{31}$$

The performance of the optimized protocols are compared. The parameters values for various strategies are shown on Table II.

Referring to Fig. 4, the costs of the optimized algorithms are plotted against cell dwell time $1/\mu$, where μ is the average cell crossing rate of the MUs. All cost functions are optimized with the assumption $\mu = 50$ and $\lambda = 1$. Referring to Fig. 4, if the MUs match this profile, i.e., $1/\mu = 0.02$ and $\lambda = 1$, all three paging schemes of the hybrid algorithm have lower costs than the geographic-based strategy. For all values of μ, paging scheme 3 gives the lowest cost, as expected. Paging scheme 1

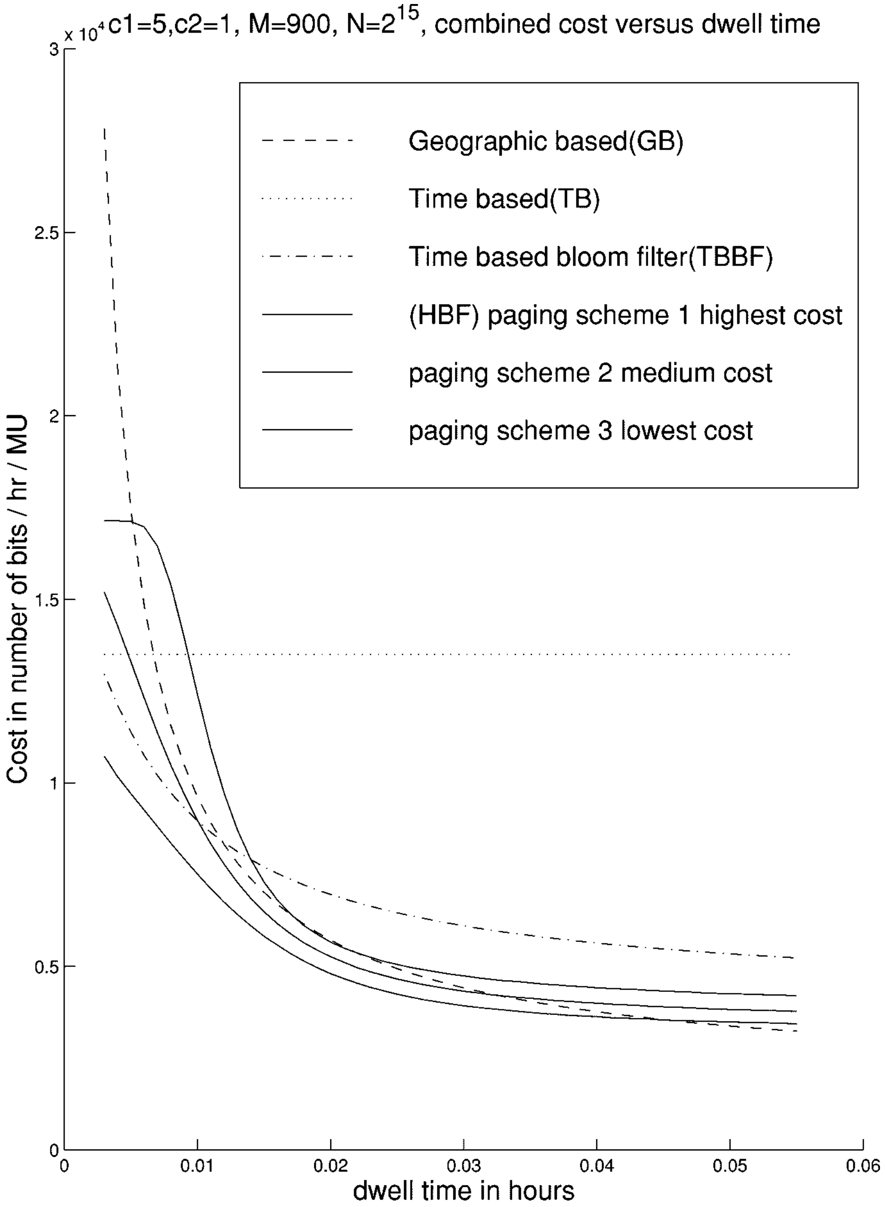

Fig. 4. Combined costs versus average cell dwell time.

gives the highest cost. The cost of paging scheme 2 approximately lies between paging scheme 1 and 3. Thus paging bandwidth can be conserved without a large sacrifice of polling delay.

The performances of these three paging schemes are close at low mobility scenario. At low mobility, $P_i \rightarrow 1/r$ for all i. Thus the relative advantage of using selective paging schemes diminishes. On the other hand, we observe that $P_i < P_j$ for all $i < j$ in the high-mobility scenario. This property could be exploited using selective paging schemes to achieve better performance. Therefore, the differences in cost of the paging schemes increase as μ increases.

The optimized period for LU in the time-based strategy is ∞. That is, no LU is ever performed. When a call arrives, the targeted MU is located by flooding. It shows that the location information provided by LU is inaccurate for paging a single cell. The network has to be flooded frequently.

The cost of the hybrid algorithms is lower than the time-based Bloom filter algorithm when average mobility is low. As shown in Section V, slowly moving MUs perform temporally triggered reregistrations most of the time. The number of LUs in each registration slot is reduced by at most r times. This improves the accuracy of the Bloom filtering paging mechanism. Unless the MUs have very high mobility, the hybrid algorithm outperforms all other time-based schemes. When paging scheme 3 is used, the HBF algorithm outperforms other time-based algorithms even in the high-mobility scenario. This is due to the fact that for a highly mobile MU, the probability of locating it in the first few matching cycles is high. Compared with the geographic-based strategy, the HBF algorithm always has lower cost, except when mobility is low. In a low-mobility scenario, most registration slots will be wasted, whereas there is no LU overhead for the geographic-based strategy.

We observe that the geographic-based strategy is robust in the low-mobility scenario but not in the high-mobility scenario. On the other hand, the time-based strategies are robust in the high-mobility scenario only. The HBF algorithm is superior in

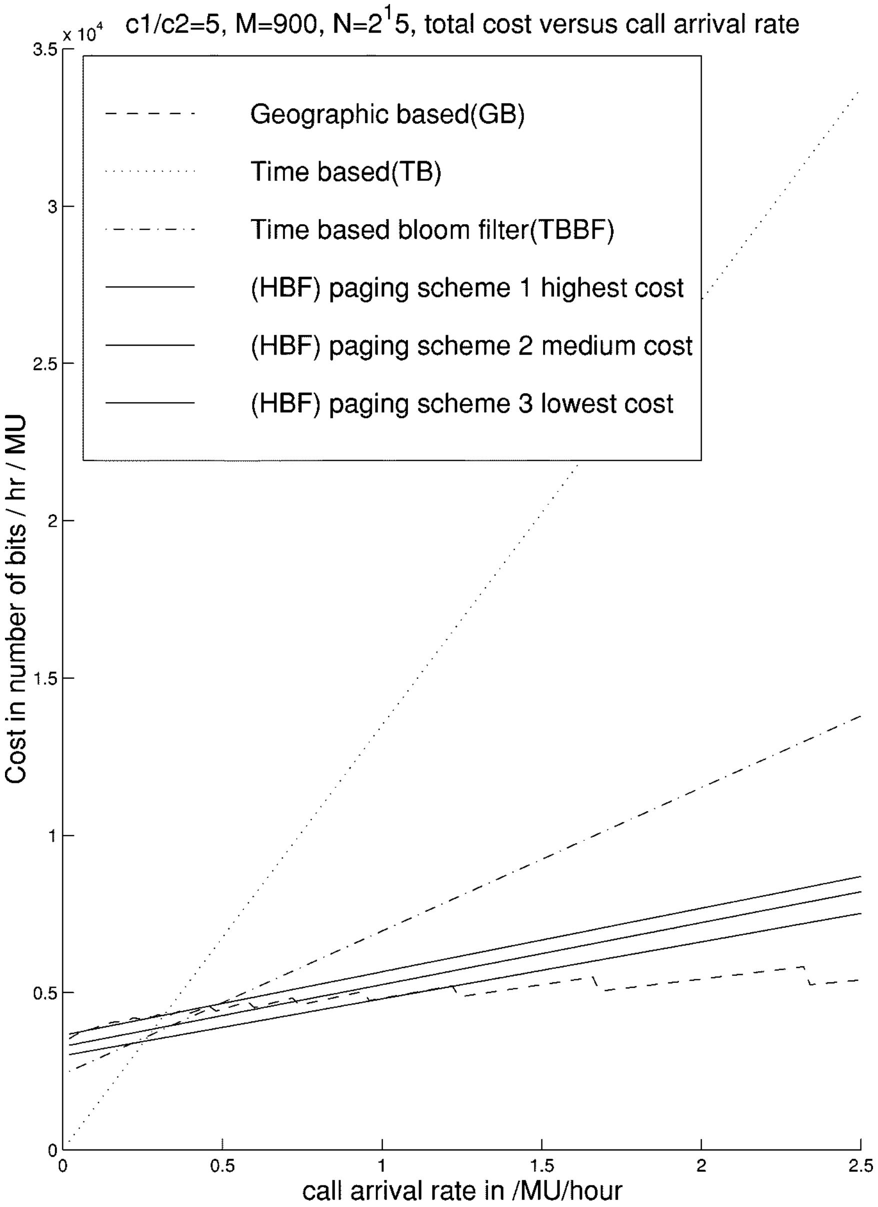

Fig. 5. Combined costs versus average call arrival rate.

the sense that it is robust in both the high- and low-mobility scenarios. Good performance could be achieved when the mobility μ is different from the assumed value used in optimization.

Referring to Fig. 5, costs are plotted against average call arrival rate λ. When λ increases, paging cost also increases. The slope for the hybrid algorithms is less steep than that of the time-based Bloom filter algorithm. This is because the Bloom filter has higher accuracy in locating a targeted MU for the hybrid algorithms. Fewer cells are paged on average. Note that if the average call arrival rate of the MUs is near the nominal value $\lambda = 1$, the cost of the hybrid algorithms is close to that of the geographic-based strategy.

VII. Implementation of the HBD in the Fixed Network

In this section, we investigate the performance of the HBF algorithm in the fixed backbone network. GSM is used as an example for comparison.

In GSM, a two-level data hierarchy is maintained on the fixed network [9]. The operator keeps a *home location register* (HLR) in which information of all mobile subscribers is stored. There is one *visitor location register* (VLR) collocated with the mobile switching center (MSC) in each location area. The update procedure is initiated when an MU enters a new location area. After receiving a request from the MU, the corresponding base station informs the MSC to update the location database in the new VLR. For security reasons, the MU does not send its international mobile subscriber identity (IMSI) through the air interface. A temporary mobile subscriber identity (TMSI) is sent instead. Thus, the new VLR must retrieve the IMSI of the MU from the old VLR through an authentication procedure. After the IMSI is obtained, the new VLR registers to the HLR and deregister the MU from the old VLR. The sequence of signaling operations is illustrated in Fig. 6.

It is of interest to compute the total signaling overhead generated by the MUs in the whole network. We observe that

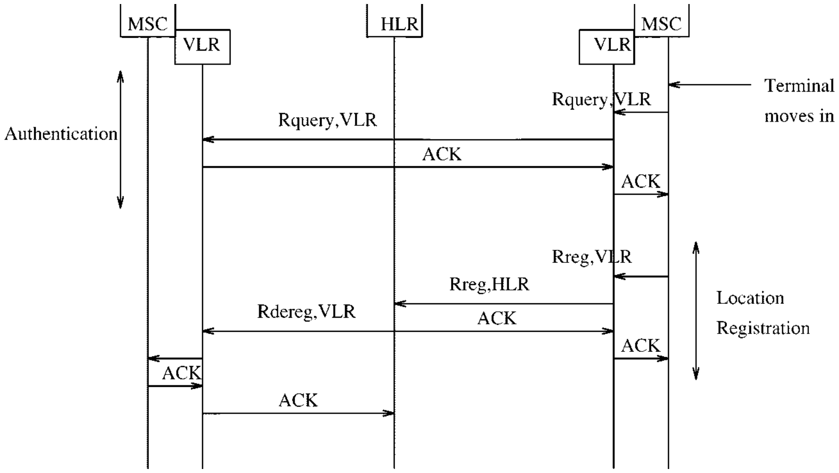

Fig. 6. Network signaling flow for terminal registration.

the total VLR registration rate $R_{\mathrm{reg,\,VLR}}$, the HLR registration rate $R_{\mathrm{reg,\,HLR}}$, and the VLR deregistration rate $R_{\mathrm{dereg,\,VLR}}$ are equal. Whereas the VLR query rate $R_{\mathrm{query,\,VLR}}$ is twice the VLR registration rate $R_{\mathrm{reg,\,VLR}}$. Each signaling message should be $\log_2 N$ long for unique identification of MUs. Thus we have

$$C_{\mathrm{GSM}} = \log_2 N(R_{\mathrm{reg,\,VLR}} + R_{\mathrm{dereg,\,VLR}} + R_{\mathrm{reg,\,HLR}} + R_{\mathrm{query,\,VLR}}).$$

The term $R_{\mathrm{reg,\,VLR}}$ is given by the location area crossing rate of all MUs. Plugging in the parameters in our numerical study, we estimate the overall signaling cost for GSM as

$$C_{\mathrm{GSM}} = \log_2 N \frac{5N\mu}{m} = 11.17 \text{ Mbits/h.}$$

For our HBF algorithm, the signaling flow is simpler. First, an MU does not send its MUID over the air interface. The security problem is automatically solved. There is no need for handshaking between the new and old serving MSC in authentication. Second, the use of a timeout in location registration exploits the advantage of soft state. No deregistration message is necessary when an MU moves from one cell to another. Third, no VLR is needed in the MSC. The location vector in each cell is periodically sent to the HLR

$$C_{\mathrm{HBF}} = \frac{M}{\tau} = 23.84 \text{ Mbits/h.}$$

In this particular example, the cost of the HBF is twice that of GSM. This is due to the frequent temporally triggered registration for HBF. However, the signaling overhead of the GSM is not scalable to large population. Consider a more realistic system with ten-fold more subscribers. Then, C_{GSM} will be 136.45 Mbits/s, whereas the C_{HBF} remains insensitive to changes in N.

VIII. Discussion

Our proposed algorithm gives comparable performance to the geographic-based strategy for a wide range of mobility. The inefficiency of random accessing in the geographic-based strategy is avoided through a contention-free LU mechanism. Nevertheless, there are two factors that limit the performance of our family of algorithms. This includes the need for system flooding and unnecessary LUs. The first factor is dominant in high-mobility scenarios, whereas the second factor is dominant in low-mobility scenarios.

At high mobility, there is an increased chance that a targeted MU is not located by probabilistic paging. We assume that network flooding is used subsequently. This flooding operation is costly and significantly increases the paging bandwidth usage. In practice, the lost MU is most likely be found in the surrounding of those paged cells. A better algorithm would be paging the surrounding cells first before resorting to flooding. Thus we could reduce the paging cost further without adding much paging delay.

At low mobility, on the other hand, a temporal-based LU scheme is less efficient. Each MU has to periodically register with the local base station even if it is stationary. There are many unnecessary LUs, which give no additional location information to the network. Our HBF algorithm is a compromise between the geographic-based strategy and the TBBF algorithm by cutting down the rate of updates. At low mobility, the LU rate for each MU is $1/r$ times the maximum value. Thus, the HBF scheme attain robustness in the high-mobility scenario compared with the geographic-based strategy while minimizing the amount of unnecessary LUs.

In the literature, most work is aimed at bandwidth minimization on the radio link, which is decoupled from the problem of bandwidth optimization in the fixed network. Often, radio bandwidth is conserved by the used of more complex protocols. This incurs an extraneous cost on the fixed network due to more complex signaling procedures. The HBF is superior to other strategies in the sense that it reduces control overhead on the radio link without a corresponding increase of the fixed network overhead. There is no need to perform authentication and deregistration explicitly with the previous location area. Network traffic is reduced considerably. Moreover, the signaling overhead does not depend on the number of mobile subscribers. Thus the HBF algorithm is scalable to large mobile population. There is also no need to maintain a VLR in each MSC. Each base station only

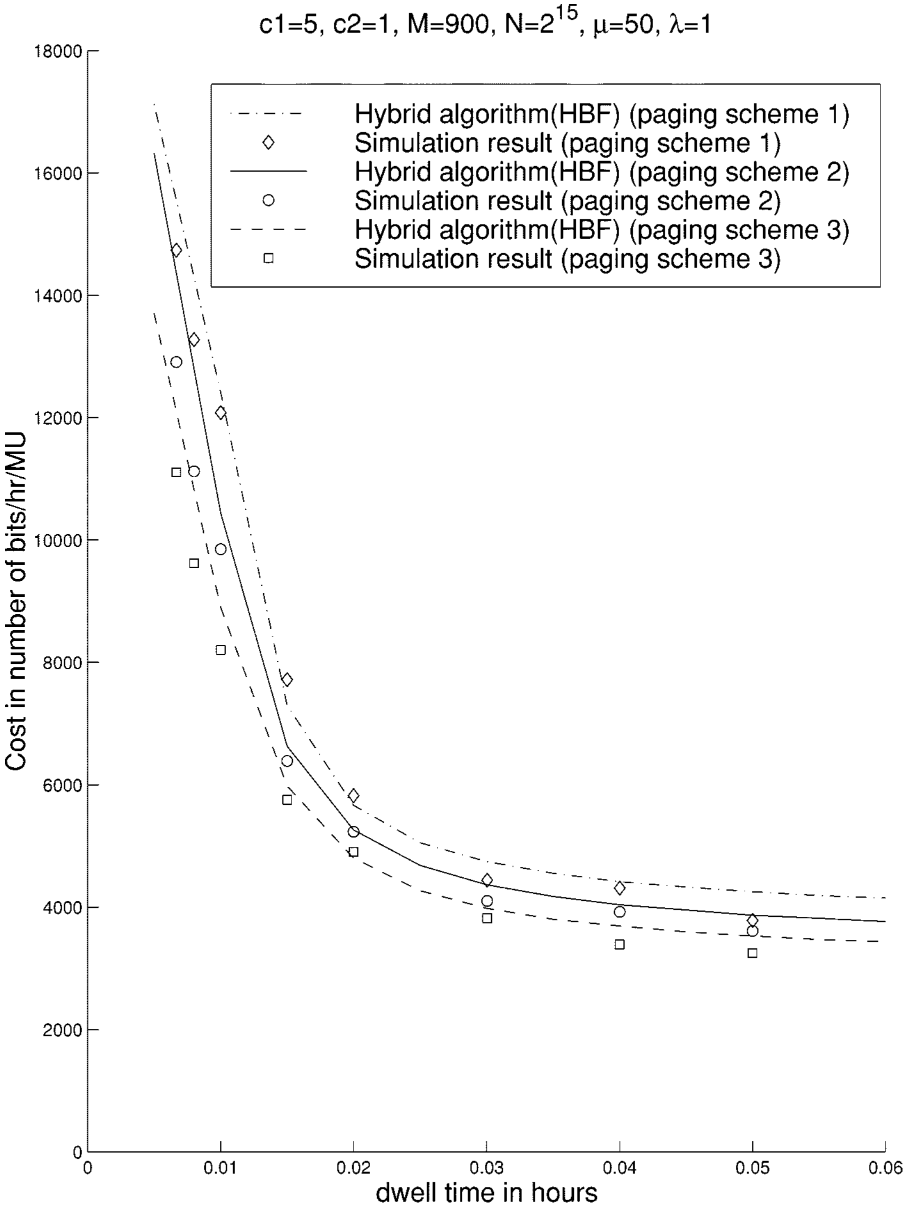

Fig. 7. Simulation results of the hybrid algorithm (paging scheme 1).

needs to maintain a cell vector and relay the information to the HLR periodically.

We hope this paper provides a new perspective on location-tracking strategies. In contrast to traditional random access and deterministic paging, the contention-free LU and probabilistic paging schemes offer an alternative paradigm on location tracking. More sophisticated protocols based on [3] should be designed to jointly optimize radio and network resources usage.

Appendix

In this Appendix, a simulation study on the HBF algorithm is reported. The probabilities of paging a cell in each proposed paging scheme are given in (24)–(26), respectively. They are obtained with the assumption that the number of registrations in each slot v_i is equal to the average value $\overline{v}$ given in (23). However, the actual number of registrations in the slots are intricately interdependent random variables. A simulation study is performed to determine whether the simplified analytical result still gives an accurate prediction to the actual performance of the algorithm. All three paging schemes are investigated in our study. For each paging scheme, the system parameters (n, τ_g, r) are optimized with the assumption $\mu = 50$. We run the simulation for eight different mobility scenarios by varying the mean dwell time of the mobile users. This is done for all three paging schemes. The simulation is written in Matlab and run for a simulation time of 300 h. This generates enough calls to average out variations of the probabilistic paging schemes.

As shown in Fig. 7, the simulation results for all paging schemes are plotted with the analytical results. We observe that the simulation results are very close to analytical results. Thus, our analytical results are justified. We conjecture that the analytical result is a conservative estimate of the actual cost, especially in a high-mobility scenario. At high mobility, a large proportion of cells are paged. If the targeted MU just departs a cell when a call arrives, it may happen that the new cell is still one of the paged cells. This decreases the need for flooding and hence the cost.

REFERENCES

[1] L. Kleinrock, *Queueing Systems*. New York: Wiley, 1975, vol. I.

[2] M. Mouly and M. Pautet, "GSM system for mobile communications,", 1992.

[3] W. S. Wong, "A new registration scheme and its communication complexity," in *Proc. Symp. Personal Communication Services*, Hong Kong, 1995, pp. 77–81.

[4] R. Jain and S. Mohan, "Two user location strategies for personal communications services," *IEEE Personal Commun. Mag.*, pp. 42–50, 1994.

[5] A. Bar-Noy and I. Kessler, "Mobile users: To update or not to update," *ACM-Baltzer J. Wireless Networks*, pp. 175–186, July 1995.

[6] B. H. Bloom, "Space/time tradeoffs in hash coding with allowable errors," *Commun. ACM*, pp. 422–426, 1970.

[7] W. S. Wong, "Control and estimation problems in mobile communication systems," in *Proc. IEEE 34th Conf. Decision and Control*, 1994.

[8] S. Okasaka, S. Onoe, S. Yasuda, and A. Maebara, "A new location updating method for digital cellular systems," in *Proc. 42nd IEE Veh. Technol. Conf.*, Denver, CO, 1992, pp. 345–350.

[9] S. Mohan and R. Jain, "Two user location strategies for personal communications services," *IEEE Personal Commun.*, 1994.

[10] H. Xie, S. Tabbane, and D. Goodman, "A dynamic location area management and performance analysis," in *Proc. IEEE '93 Veh. Technol. Conf.*, 1993, pp. 536–539.

[11] J. S. M. Ho and I. F. Akyildiz, "Mobile user location update and paging under delay constraints," *ACM-Baltzer J. Wireless Networks*, vol. 1, no. 4, pp. 413–425, Dec. 1995.

[12] K. L. Yeung and T.-S. P. Yum, "Comparative study on location tracking strategies in cellular mobile radio systems," in *Proc. IEEE '95 Globecom*, 1995, pp. 22–28.

[13] W. H. A. Yuen and W. S. Wong, "A hybrid bloom filter location update algorithm for wireless cellular systems," in *Proc. IEEE ICC '97*, June 1997.

[14] ——, "A dynamic location area assignment algorithm for mobile cellular systems," in *Proc. IEEE ICC '98*, June 1998.

[15] M. Verkama, "A simple implementation of distance-based location updates," in *Proc. IEEE '97 ICUPC*, 1997, pp. 163–167.

Wing Ho A. Yuen received the degree in electrical and electronic engineering from Hong Kong University of Science and Technology, Hong Kong, in 1995 and the M.Phil. degree in information engineering from the Chinese University of Hong Kong in 1997. He is currently pursuing the Ph.D. degree in electrical engineering at Rutgers—The State University, New Brunswick, NJ.

His research interests include mobility management, power control, rate management for cellular, and *ad hoc* networks.

Wing Shing Wong (M'81–SM'90) received the M.A.B.A. degree *(summa cum laude)* from Yale University, New Haven, CT, in 1976 and the M.S. and Ph.D. degrees from Harvard University, Cambridge, MA, in 1978 and 1980, respectively.

He joined AT&T Bell Laboratories in 1982. From 1987 to 1992, he managed a group of technical staff working on a number of research and development projects and consulting activities. He joined the Chinese University of Hong Kong in 1992 and is now Professor of information engineering. He has been the Chairman of the Information Engineering Department since 1995. He is actively involved in a variety of R&D projects, including topics such as mobile communication systems, Web technology development, and information issues in estimation and control. He has published more than 85 refereed journal and conference papers and is the recipient of multiple competitive R&D grants from the Hong Kong Research Grant Council and the Industry Support Fund. He leads a project on a Chinese search engine, Jasmine & ANSeRS, which has been licensed to several high-tech companies. He is one of the initiators of the Information Technology Entrepreneur Programme hosted by the Information Technology Entrepreneurs Association. He is a Discipline Representative for Information Engineering in HKIE and a Group Editor for the *HKIE Transactions*. He is the Coeditor-in-Chief of *Communications in Information and Systems.*

Prof. Wong is a Fellow of the Hong Kong Institution of Engineers. He was an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL for four years.