Wireless Networks 9, 637–646, 2003

Tracking Mobile Users with Uncertain Parameters

ZOHAR NAOR
Department of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel

Abstract. A method of reducing the wireless cost of tracking mobile users with uncertain parameters is developed in this paper. Such uncertainty arises naturally in wireless networks, since an efficient user tracking is based on a *prediction* of its future call and mobility parameters. The conventional approach based on dynamic tracking is not reliable in the sense that inaccurate prediction of the user mobility parameters may significantly reduce the tracking efficiency. Unfortunately, such uncertainty is unavoidable for mobile users, especially for a bursty mobility pattern. The two main contributions of this study are a novel method for topology-independent distance tracking, and a combination of a distance-based tracking with a distance-sensitive timer that guarantees both efficiency and robustness. The expected wireless cost of tracking under the proposed method is significantly reduced, in comparison to the existing methods currently used in cellular networks. Furthermore, as opposed to other tracking methods, the worst case tracking cost is bounded from above and governed by the system, such that it outperforms the existing methods. The proposed strategy can be easily implemented, and it does not require a significant computational power from the user.

Keywords: PCS, mobile, user tracking

1. Introduction

Tracking mobile users is a key issue in wireless networks. Contrary to wired networks in which the user location is fixed, in wireless networks a user can potentially be located anywhere within the system service area. Whenever there is a need to route an incoming call directed to the user, the network must find the user location in order to set up the call. For this reason, tracking mobile users has a major importance in wireless networks. Due to the growing demand for Personal Communication Services (PCS), future wireless networks will have to provide efficient and low cost services to a large population of users. As the number of users keeps increasing, the amount of signaling traffic associated with the need to locate mobile users keeps growing. The problem of reducing the tracking cost has two aspects: First, the utilization of the wired links of the backbone network, for accessing data bases in which the user location information is stored. Second, the utilization of the wireless resources associated with tracking. In this study, our concern is in the utilization of the wireless links associated with tracking. The importance of this issue comes from the fact that the signaling traffic associated with tracking is transmitted through the control channel. Due to the limited bandwidth available for wireless communication, the control channel is the most precious resource in a wireless network.

User tracking is based on two elementary operations: *Registration* (location update) – a message sent by the user that informs the network about its location, and *paging* (or search)– a search conducted by the network, to find the user location. Since our interest is in the utilization of the network wireless resources, we measure the paging cost by the number of wireless search operations, and ignore the cost of searching the relevant data bases. There is a clear trade-off between the rate of registration messages and the cost of paging the user when a connection to the user (for an incoming call) is requested. Hence, the problem of tracking mobile users is an optimization problem: The goal is to minimize the *combined cost* of registration and search operations.

Existing tracking schemes and previous studies addressing this issue are based either on *user operated* algorithms which are *pure distributed* in nature, or on *network operated* algorithms which are *centralized* in nature. Unfortunately, due to the complex nature of the tracking problem, efficient user tracking cannot be done by the user only nor by the network only. Due to the huge number of mobile users, any efficient tracking scheme that consider the individual user call and mobility parameters, must be distributed among the users. On the other hand, an efficient tracking scheme must use network-dependent information, such as the system topology and the activity of other users. Such information is not generally available to the user. Furthermore, the requirement of real-time service to millions of users implies that no sophisticated algorithm is indeed feasible, if it is to be installed *solely* on user equipment. For these reasons, a pure distributed tracking scheme is incomplete and unreliable under realistic conditions.

Recognizing these drawbacks, this work proposes a novel approach to the problem of tracking mobile users. The proposed method is neither centralized nor distributed. Rather, it is a combined scheme that incorporates a distributed scheme with a centralized scheme. We refer to this scheme as an *interactive scheme*. The basic idea is to use the broadcast mechanism that already exists in wireless networks in order to: (a) Provide users with the information essential for efficient tracking, that is not generally available to them, and (b) to govern the registration activity and reduce the contention among users wishing to register. The basic concept of the proposed method is to leave user specific decisions in the user equipment while moving network general decisions to the network. Consequently, user tracking is neither done by the user only nor by the network only. Rather, a combined scheme that utilizes the strengths of both entities is used. Guided by this idea, we construct a scheme based on *interaction* between the network and the users, yielding a very efficient tracking algorithm. The tracking scheme proposed in this study can significantly reduce the wireless cost of tracking mobile users, in comparison to existing methods and previous studies.

1.1. Related works

The problem of reducing the wireless cost of tracking mobile users has been addressed by many studies [1,2,4,5,10,13,16]. Existing cellular networks use the following tracking strategy,known as the geographic-based (GB) strategy: The geographic area is partitioned into zones, known as *location areas* (LA) in GSM systems. The IS-41 standard [6] refers to the *location area* as the *registration area*. The system partitioning into *location areas* is static, based on the commercial licenses granted to the operating companies. Each LA consists of a number of cells. The user registers whenever it enters a new LA. Within the LA the user never register in the GSM system. Thus, when there is an incoming call directed to a user, all the cells within its current LA are paged. Since the number of cells within a typical LA is very large, the tracking cost associated with the GB strategy is very high. Many systems incorporate the IS-41 standard with a fixed timer as follows: The user registers periodically every T time units, where T is a fixed, pre-defined parameter. Since no knowledge about the user call and mobility parameters is assumed, the same timer T is used by *all* the users, independent of the dynamic behavior of each user. Clearly, this strategy is far from being optimal.

Many location management strategies, aiming to reduce the wireless cost of tracking mobile users were proposed [1,2,4,5,10,13,16]. The basic idea shared by these studies is that upon location change, the user may or may not update its location. The criterion for user registration may be a function of time [16], distance from last known location [4,10], or number of movements between cells [1,4]. An on-line learning algorithm, used to built the user profile, was suggested in [5], in order to reduce the cost of location update. The strategies proposed in these studies are pure distributed. As such, they have many drawbacks:

1. An accurate prediction of the user future behavior is required. Such a prediction is feasible when a fixed mobility pattern is assumed (e.g., random walk, a diffusion process). However, in reality, user mobility, as well as its call pattern, are in many cases bursty. Their prediction is therefore not a simple task. Unfortunately, the optimal registration rate is very sensitive to the user mobility pattern, as well as to other parameters. The value of the registration parameter (e.g., distance, time, etc.) which is optimal for one mobility pattern (e.g., random walk) may not be optimal for another mobility pattern. Even if the mobility pattern can be accurately estimated, it is not clear that past mobility parameters (e.g., movement probability) are to be expected in the future too.
2. The computational power imposed on the user is very high, and a dynamic programming method is often required. Even if an accurate prediction of the user mobility pattern and call profile is assumed, the evaluation of the exact parameter used for optimal registration (e.g., distance, time, movements) is not a simple task. It often requires a dynamic programming method and a significant computing power. It is not clear if the computational power imposed on the user by some of these methods is indeed feasible.
3. The behavior of other users is ignored. Since the basic idea underlying these methods is to use registration messages in order to reduce future paging cost, the cost of registration must be taken into account. Moreover, a pure distributed registration strategy is based on the assumption that user movements are independent from one user to another. If this assumption does not hold, the likelihood of collisions between registration messages may be significant. In extreme situations, such as sport events, or a train, it may even block the control channel.
4. The information required by some of these methods, such as the distance in terms of cells traveled by the user, is not generally available to the user.

To overcome the third drawback, a load-sensitive approach, in which the tracking activity can be adjusted to both user and system activity was suggested in [9,13]. The methods proposed in these studies can be combined with any one of the other methods described above (i.e. the timer-based, movement-based, or distance-based method). In [14], the use of *Cell Identification Codes* (CIC) was proposed, in order to provide the users the location information required for efficient tracking.

For each one of the tracking strategies mentioned above, the value of the registration parameter (timer, distance, or number of movements), is very sensitive to the user call and mobility parameters. Consequently, in most cases the performance of these strategies exceeds that of the existing strategy (e.g., the GSM system) only under certain mobility and call parameters. For different range of parameters the performance comparison to the existing method is not clear. Unfortunately, an accurate and efficient *prediction* of the user call and mobility parameters is still an open problem. Even if more accurate estimation of the user behavior is provided in the future, a certain uncertainty, which is inherent to the bursty nature of user mobility, is yet to be expected.

Hence, the tracking strategies mentioned above are not reliable in the sense that inaccurate estimation of the user parameters may significantly reduce their efficiency. Unfortunately, such uncertainty is unavoidable for mobile users, especially for a bursty mobility pattern. For this reason, our goal is to construct a tracking scheme that is not sensitive to the user mobility pattern. The uncertainty in the location of mobile users has been discussed in many studies (see, for example, [15,18]). In this study, the term *uncertainty* has a different meaning: our concern is in the uncertainty in the traffic model.

2. The contribution of this work

The goal of this work is to propose a tracking strategy, which is efficient yet reliable. To achieve this goal, we propose a strategy that is neither centralized nor distributed. Rather, it is a combined scheme that incorporates a distributed scheme with a centralized scheme. We refer to this scheme as an *interactive scheme*.

Previous studies addressing the issue of tracking mobile users aimed to reduce the tracking cost under certain assumptions about the user behavior (e.g., its mobility pattern). However, if these assumptions are violated, the performance of these strategies may be very poor. For this reason, many of these strategies are considered as unreliable for realistic systems. In this study our goal is to reduce both the expected tracking cost, as well as the worst case tracking cost.

Using a simple user–network interaction, the rate of location update is based both on the dynamic activity of the individual user, as well as on the (local) system activity, and takes into consideration the limitations of predicting a bursty behavior. The tracking strategy incorporates a distance-based strategy with a timer, that counts the time duration since the last location update. The system-dependent information required for efficient registration, such as the system load and location information, is provided via *Dynamic Cell Identification Code* (DCIC): Each cell periodically broadcasts a short message which provides the information required for efficient registration. The Dynamic Cell Identification Code of each cell depends on the cell load. Hence, it changes dynamically, as a function of the statistical usage of the local up link control channel. The tracking method proposed in this paper can be implemented independently of the network topology and the user mobility pattern. As opposed to other methods, the computational power imposed on the user is relatively low, and the user is not required neither to implement a complex algorithm, nor to use a dynamic programming. The proposed method is shown to outperform both the geographic-based strategy, used by current systems, and the timer-based method, as used by many systems who incorporate the IS-41 standard with a fixed (static) timer. We show that the registration rate under the proposed method is not larger than the actual registration cost of these methods, while the paging cost under the proposed method is significantly lower than the paging cost under the existing methods. The main results of this study are:

1. A topology-independent distance tracking algorithm in which the user, while roaming between the cells, learns about the system topology, using location information announced by the system as a broadcast message.
2. The usage of a distance-sensitive timer, to accommodate users with uncertain mobility pattern.
3. The use of an interactive scheme, that yields a tracking method that is both efficient and robust.

The expected tracking cost under the proposed method is significantly reduced, in comparison to the existing methods used in current networks. Moreover, the worst case performance is bounded from above, and governed by the system such that it outperforms the existing methods.

The structure of the rest of this paper is as follows: Model and notation are described in section 2. The tracking strategy is described in section 3. In section 4 we provide a performance comparison to other tracking strategies, including the strategies used in current systems. Summary and concluding remarks are given in section 5.

3. Model and notation

We consider a wireless network partitioned into cells. The user location is understood as an identifier of the cell in which the user is currently residing. Two cells are called neighboring cells (or: nearest neighbors) if a user can move from one to the other, without crossing any other cell. To model user movement in the network we assume that time is slotted, and that a user can make at most one cell transition during a single time slot. It is assumed that the movement of the user is done just at the beginning of time slot, such that it precedes any other event, such as paging event. The user *roaming interval* is defined as in [16], as the time interval since the last contact of the user with the system, and the next paging event to this user. Location management messages are transmitted through the up-link and down-link signaling channels. Since the bandwidth of these channels is fixed, our goal is the utilization of the signaling channels. It is assumed that the system has no knowledge about the user mobility pattern. Hence, the cost of searching the user depends only on its distance, in terms of number of cells, from its last location known to the system.

The *mobility graph* is defined as an undirected graph, in which each cell in the network is represented by a vertex. An (undirected) edge exists between two vertices in the *mobility graph*, say $x1$ and $x2$, if and only if, the cells $x1$ and $x2$ are nearest neighbors cells.

The *distance* between two cells, say $x1$ and $x2$, denoted by $d(x1, x2)$, is defined as the minimal number of cell boundary crossing upon traveling from $x1$ to $x2$. Hence, the distance $d(x1, x2)$ is the length of the shortest path between $x1$ and $x2$ on the *mobility graph*, measured in number of edges on the *mobility graph*. That implies that for any cell x, $d(x, x) = 0$, and that $d(x1, x2) = 1$ if and only if, $x1$ is a nearest neighbor of $x2$.

4. The interactive tracking scheme

Below we describe the proposed tracking strategy. The basic idea is to incorporate a distance-based strategy with a distance-sensitive timer, using the following mechanism:

- The user tracks the distance, in terms of cells, traveled from its last known location.
- In addition, the user counts the time duration since its last location update.
- The correspondence between the distance counter and the time counter is given by a pre-defined look-up table, in which for each distance d_i, $d_i = 1, 2, \ldots$, there is a corresponding pre-defined parameter T_i. Whenever the user travels a distance d_i *and* the time duration since its last location update is at least T_i, the user transmits a registration message. The condition for registration is that *both* conditions are satisfied. If the user travels a distance d_i at less than T_i time units, the user has to wait until the time duration since its last location update is at least T_i.
- The sequence d_i, $i = 1, 2, \ldots$, is in increasing order, and the sequence T_i, $i = 1, 2, \ldots$, is in non-increasing order i.e. if $i > j$, then $d_i \geqslant d_j$, and $T_i \leqslant T_j$. In order to make the look-up table as small as possible, starting from a threshold index i_t, we get that $T_i = T_j$, $\forall i, j \geqslant i_t$, implying that all distances greater or equal to a pre-defined threshold have the same timer.
- The look-up table that defines the correspondence between the distance counter and the time counter can be load-dependent. The registration rate can (and should) increase at lightly loaded cells, implying that the time counter corresponding to each distance counter can be load-dependent. For example, the registration rate used at urban area may differ from that used in a highway. Consequently, different look-up tables may be used at the same location, at different load conditions. The network determines which look-up table is to be used at each cell, at any given moment, and announces it, as a broadcast (short) message, to the users. A detailed description of such mechanism can be found in [13]. Nevertheless, in order to simplify the analysis, it is assumed from now on that the user holds only one look-up table.

Since the registration time threshold decreases with the distance traveled by the user, while the maximal distance the user can travel increases with time, both the registration rate and the paging cost are bounded from above. To realize this mechanism, let $i_{\max}$ be the maximal index (entrance) in the look-up table. The registration rate is bounded from above by $1/T_{i_{\max}}$. In addition, let $v_{\max}$ be the maximal user speed, in terms of number of cell boundary crossing per time unit. The maximal distance, in terms of cells, that the user can travel before transmitting a registration message is the minimal value of i that satisfies the condition: $d_i/v_{\max} \geqslant T_i$. *Hence, the combination of a distance-based strategy with a distance-sensitive timer that decreases with the distance forms a mechanism that bounds from above both the registration rate and the paging cost.*

The implementation of the proposed strategy is very simple: Each cell periodically broadcasts a short code which identifies the look-up table to be used. The wireless bandwidth required to transmit the code is relatively small. In order to support M different look-up tables, only $\log_2 M$ bits suffice.

4.1. The paging strategy

The proposed registration strategy determines an upper bound on the uncertainty in user location. This upper bound is used for efficient paging strategy.

Let τ be the length of the user *roaming interval*, defined as the time duration since the last location update and the next paging event [16].

Lemma 4.1. The distance d between the user location at the time of paging event and its last known location is bounded from above as follows:

- the condition $\tau \geqslant T_1$ implies that $d < d_1$,
- if $T_1 > \tau \geqslant T_2$ then $d < d_2$,
- if $T_2 > \tau \geqslant T_3$ then $d < d_3$,

$\vdots$

- $T_{i-1} > \tau \geqslant T_i \Longrightarrow d < d_i$.

Proof. The proof follows directly from the correspondence between the distance counter and the time counter. Given that $\tau \geqslant T_1$, the user must transmit a registration message whenever its distance from its last known location is greater than or equal to d_1. Since no registration message was sent, it follows that $d < d_1$. For example, if $d_1 = 1$ then $d = 0$. Similarly, if $T_{i-1} > \tau \geqslant T_i$ then the distance d traveled by the user must be in the range $0 \leqslant d < d_i$. Otherwise, if $d \geqslant d_i$ the user would register. □

Remark 4.1. Substitute $d_i = i$, $\forall i$ we get that the condition $T_{i-1} > \tau \geqslant T_i$ implies that $d < i$.

Lemma 4.1 implies that the longer is its roaming interval, the less is the uncertainty in the user location. Hence, the probability to find the user at paging event at a distance d_i from its last known location is the probability of traveling a distance d_i ($i = 1, 2, \ldots$) during *less than* T_i time units, multiplied by the probability of paging event during *less than* T_i time units. Taking into consideration that the probability to travel a long distance *increases* with the user's roaming interval, the proposed combination of the registration rate with the distance traveled forms a mechanism which *bounds both the expected paging cost and the registration rate.*

To illustrate this mechanism, consider the following tracking strategy, in which the timer T_i is given by $T_i = T_0/i$, where i is the distance, in terms of cells, traveled by the user. Given that the user moves in a velocity of v cells per time unit, the upper bound on the distance, in terms of cells, the user can travel without transmitting a registration message, is the maximal value of i that satisfies $i = vt \leqslant vT_0/i$, where t is the time duration, measured in time slots, since the last registration message transmitted by the user. Hence, this distance is bounded from above by $\sqrt{T_0 v}$.

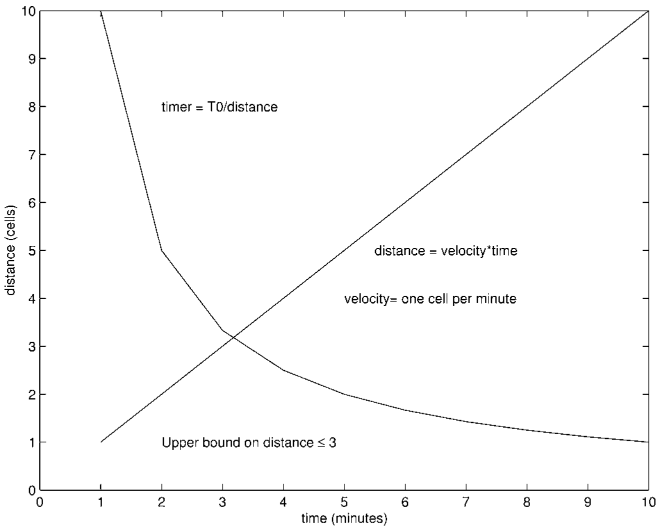

Figure 1. Upper bound on the uncertainty in user location for a user with maximal velocity of one cell crossing per minute, and a timer $T_i = T_0/i$, $i = 1, 2, \ldots$, and $T_0 = 10$ minutes.

Figure 1 depicts the upper bound on the uncertainty in user location for a user with maximal velocity of one cell crossing per minute, and a timer $T_i = T_0/i$, $i = 1, 2, \ldots$, and $T_0 = 10$ minutes. The maximal distance that the user can travel before transmitting a registration message is less than or equal to 3 cells. Hence, the paging cost is bounded from above to the number of cells at a distance no longer than 3 cells. Since the uncertainty in user location depends only on the time duration since its last registration message until the paging event, and on the upper bound on the user velocity, this uncertainty can be easily computed by the system. Consequently, an efficient paging strategy should search for the user only in a restricted area, that is determined by the (computed) uncertainty in the user location.

5. The (topology-independent) distance tracking mechanism

Below we describe the method used for measuring the distance traveled by the user.

Definition 5.1. The *adjacency list* $A(x, R)$ is defined as the list of all the neighbors of a cell x at a distance less than or equal to R.

The distance tracking mechanism consists of two elements:

- Each cell has a unique code (ID) that identifies the cell.
- Each cell periodically transmits its ID and the adjacency list $A(x, R)$ of all its neighbors at a distance less than or equal to a *radius* R, where R is a pre-defined parameter that depends on the system topology. The *adjacency list* $A(x, R)$ includes the ID codes and the distances of all the neighbors of the cell x, up to a distance R. For each $y \in A(x, R)$, x transmits the ID of y, and $d(x, y)$. The *adjacency list* is announced by the base station as a broadcast message, through the down-link control channel (e.g., DCCH in GSM networks).
- The user *learns* about the system topology and its location, and estimates its distance from its last location known to the system, by listening to the *adjacency list* $A(x, R)$ *while roaming between the cells in the network.* The user computes the distances between the visited cells by comparing the *adjacency lists* $A(x1, R)$, $A(x2, R), \ldots$, where $x1, x2, \ldots$ are the cells visited by the user.

The method is independent of the system topology. However, the radius R of the *adjacency list* required for an accurate estimation of the distance traveled by the user, depends on the system topology. This dependency is studied in this section.

Remark 5.1. Currently in CDMA networks each cell transmits a unique ID that identifies the cell, and a list of all its nearest neighbors, in order to provide soft hand-off between cells. This information can be easily used for location management, at actually no cost. As we show later, for most practical cases it is sufficient to use $R = 1$ for efficient distance-based tracking.

The user computes its distance from its last known location, using the *adjacency lists* that were received during its roaming. The computed distance, denoted as d^*, is evaluated as follows:

- Let $x(t)$ denotes the user location at time t.
- The user *path*, denoted by P_u, is therefore the locations $x(0), x(1), x(2), \ldots, x(t-1), x(t)$.
- The *Personal Mobility Graph* of a user u, denoted by G_u, is defined as:

  $$G_u = \{y \in A(x(t'), R),\ 0 \leqslant t' \leqslant t\},$$

  where $A(x(t'), R)$ is the *adjacency list* of the cell $x(t')$. The set G_u is therefore the collection off all the locations visited by the user u, and all their neighbors at a distance less than or equal to R.

The distance of $x(t)$ from the user u location, say at time $t = 0$, $x(0)$, is estimated by:

$$d^*(x(t), x(0)) = \min\{d^*(x(t), x) + d^*(x, x(0))\}, \quad x \in G_u. \tag{1}$$

Hence, $d^*(x(t), x(0))$ is the shortest path in G_u between $x(t)$ and $x(0)$. Note that it follows directly from the definition of G_u that $x(t) \in G_u$, $x(0) \in G_u$, and that $P_u \subseteq G_u$. Consequently, $d^*(x(t), x(0))$ always exists. Using the triangular equality and the definition of d^*, we get that for all locations $x1, x2$, the following equation holds:

$$d^*(x1, x2) \geqslant d(x1, x2), \quad \forall x1, x2. \tag{2}$$

Given that $d(x1, x2) \leqslant R$, the following lemma holds:

Lemma 5.1. If either $x1 \in P_u$, or $x2 \in P_u$, then $d(x1, x2) \leqslant R \Rightarrow d^*(x1, x2) = d(x1, x2);\ \forall x1, x2$.

Proof. The condition $d(x1, x2) \leqslant R$ implies that $x2 \in A(x1, R)$ and $x1 \in A(x2, R)$. Hence: $d^*(x1, x2) = d(x1, x2)$. □

Our concern is in the distance traveled from the user last location known to the system. Hence, our interest is in the accuracy of $d^*(x1, x2)$, where both $x1$ and $x2$ are locations visited by the user. Under this condition, the following lemma holds:

Lemma 5.2. Given that $x1$ and $x2$ are locations visited by the user, the condition $d(x1, x2) \leqslant 2R$ implies that $d^*(x1, x2) = d(x1, x2)$.

Proof. The condition $d(x1, x2) \leqslant 2R$ implies that there exists a cell x', such that x' belongs to the shortest path from $x1$ to $x2$, and $x' \in A(x1, R) \cap A(x2, R)$. For example, if $d(x1, x2) = 2R$ then $\exists x',\ d(x1, x') = R = d(x2, x')$. It follows from the definition of d^* and lemma 5.1 that: $d^*(x1, x2) = d^*(x1, x') + d^*(x', x2) = d(x1, x') + d(x', x2) = d(x1, x2)$. □

Lemma 5.3. The condition $A(x1, R) \cap A(x2, R) = \emptyset$ implies that $d(x1, x2) > 2R$.

The proof follows directly from the definition of $A(x, R)$ and the proof of lemma 5.2. Lemmas 5.2 and 5.3 imply that the distance-based strategy [4,10] can be implemented up to a distance of $2R + 1$, *independently of the network topology*. The proof of this claim follows directly from the next lemma.

Lemma 5.4. Consider a user that its last known location at time $t = 0$ is $x(0)$, and its location at time t is $x(t)$. Then, the conditions that $d^*(x(t-1), x(0)) = 2R$, and $A(x(t), R) \cap A(x(0), R) = \emptyset$ imply that $d^*(x(t), x(0)) = d(x(t), x(0)) = 2R + 1$.

Proof. Using lemma 5.2, we get that:

$$\begin{aligned} d^*\big(x(t), x(0)\big) &\leqslant d^*\big(x(t-1), x(0)\big) + d^*\big(x(t-1), x(t)\big) \\ &= d^*\big(x(t-1), x(0)\big) + 1 = 2R + 1. \end{aligned}$$

Using that with equation (2) and lemma 5.3, we get that $2R + 1 \geqslant d^*(x(t), x(0)) \geqslant d(x(t), x(0)) > 2R$. Hence: $d^*(x(t), x(0)) = d(x(t), x(0)) = 2R + 1$. □

Using lemmas 5.1–5.4, we get that under the condition that $x1, x2 \in G_u$, and in particular, given that $x1, x2$ are locations visited by the user, the condition $d(x1, x2) > 2R$ implies that:

$$d^*(x1, x2) \geqslant d(x1, x2) > 2R. \tag{3}$$

Remark 5.2. Lemma 5.2 and equation (3) imply that even if the estimated distance d^* is inaccurate, the result of using d^* for distance-based tracking would be, in the worst case, a slight increase in the registration rate.

Definition. The *diameter* D of a graph G is defined as the length of the longest distance between two nodes in the graph.

Lemma 5.5. A sufficient condition under which $d^*(x1, x2) = d(x1, x2)$, where $x1$ and $x2$ are any cells visited by the user is that $R = \lfloor D/2 \rfloor$, where D is the diameter of the mobility graph G.

Proof. Using that for any cells $x1, x2 \in G$, $d(x1, x2) \leqslant D \leqslant 2R + 1$, the proof follows directly from lemmas 5.2 and 5.3. □

Lemma 5.6. Given a mobility graph G, a sufficient condition under which $d^*(x1, x2) = d(x1, x2)$, where $x1$ and $x2$ are any cells visited by the user, is that the maximal length of a cycle in G is at most $4R + 2$.

The proof is given in appendix A.

Remark 5.3. Lemma 5.6 implies that whenever the mobility graph has no cycles (i.e. the mobility graph is a forest graph), a sufficient conditions under which $d^*(x1, x2) = d(x1, x2)$, where $x1$ and $x2$ are any cells visited by the user, is that $R = 1$. For example, in tree graphs, or line graphs, it is sufficient to use $R = 1$ to implement a distance-based tracking strategy.

Lemma 5.6 implies that under the condition that the maximal cycle length in the *mobility graph* G is n, a sufficient condition under which $d^*(x1, x2) = d(x1, x2)$, where $x1$ and $x2$ are cells visited by the user, is that $R = \lceil (n-2)/4 \rceil$. In particular, if G is a ring graph of length n, a sufficient condition is that $R = \lceil (n-2)/4 \rceil$.

Lemma 5.7. Given a mobility graph G, a sufficient condition under which $d^*(x1, x2) = d(x1, x2)$, where $x1$ and $x2$ are any cells visited by the user, is that $R = \min\{\lfloor D/2 \rfloor, \lceil (n-2)/4 \rceil\}$, where D is the diameter of G, and n is the maximal length of a cycle in G.

Proof. The proof follows directly from lemmas 5.5 and 5.6. □

6. Performance analysis

In this section we show that the interactive tracking strategy is superior to the existing tracking methods used in current systems. The interactive tracking strategy also performs better than the distance-based strategy, that is considered as the most efficient tracking strategy, in the sense that the performance sensitivity of the interactive scheme to the user parameters is much better than the performance sensitivity of the distance based strategy. An uncertainty in the user mobility pattern has only a limited effect on the tracking cost under the interactive tracking strategy, while under the distance-based strategy it may significantly change the cost of tracking.

6.1. The timer-based method

Many systems incorporate the IS-41 standard with a fixed (static) timer. Below we show that the interactive tracking strategy outperforms the timer-based method in the sense that both the registration rate and the paging cost under the interactive tracking strategy are lower than that of the timer-based method.

The registration rate under the timer-based method, using a timer T, is given by:

$$R_{\text{timer}} = \frac{1}{T}. \tag{4}$$

Consider a special case of the interactive tracking strategy, under which $T_i = T$, $\forall i$, where T is the timer used under the timer-based method. The *residential interval* denoted by θ is defined as the time interval in which the user resides in its cell. The user mobility pattern consists of *residential intervals* and cell movements, such that each *residential interval* is terminated by a cell boundary crossing. Let $\alpha(T)$ be the probability that the *residential interval* is less than or equal to T:

$$\alpha(T) = \Pr[\theta \leqslant T]. \tag{5}$$

The registration rate under the interactive strategy is therefore given by:

$$\begin{aligned} R_{\text{I}} &= \frac{\alpha(T)}{T} + \sum_{t=T+1}^{\infty} \frac{\Pr[\theta = t]}{t} \\ &< \frac{\alpha(T)}{T} + \frac{1}{T}\sum_{t=T+1}^{\infty} \Pr[\theta = t] \\ &= \frac{\alpha(T)}{T} + \frac{1-\alpha(T)}{T} = \frac{1}{T} = R_{\text{timer}}. \end{aligned} \tag{6}$$

The paging cost under the interactive strategy is certainly lower than that of the timer-based method. To show that, consider the user *residential interval* θ. If $\theta \leqslant T$, the registration rate, and hence the paging cost under the interactive strategy are equal to that of the timer-based method. Otherwise, if $\theta > T$, the user performs the "always update" strategy, hence its paging cost is minimal. A simple paging strategy for the interactive strategy is the following: The user is paged first at its last known location, and if not found, at increasing distance from its last known location. For example – at the second paging cycle we search at all the cells at a distance one from the user last known location. At the last paging cycles the search is conducted at all the cells in the location area in which we did not search before. The number of paging cycles depends on the constraint on the paging delay. Assuming the same paging strategy for the timer-based method, let S_{timer} denotes the search (paging) cost under the timer-based method, in terms of number of cells need to be searched. Clearly, $S_{\text{timer}} > 1$. The search cost under the interactive strategy is given by:

$$S_I = \alpha(T)S_{\text{timer}} + [1-\alpha(T)]1 < S_{\text{timer}}. \tag{7}$$

Hence, both the registration rate and the paging cost under the interactive tracking strategy are lower than that of the timer-based method.

6.2. The geographic-based (GB) method

The interactive strategy is to be implemented *in addition* to the GB method, hence it can only enhance its performance. The reason for that is that under the GB method, users register only at the boundaries of the LA's. Hence, there are non-utilized system resources *within* the LA. These resources can, and should be, used for registration. Taking into consideration that under the interactive strategy the registration activity is governed by the system, *only* non-utilized system resources are used for registration, *at practically no cost.* A detailed description of the control mechanism which governs the registration rate as a function of the load on the control channel can be found in [13]. Thus, the paging cost under the interactive strategy would be certainly lower than that of the GB strategy, without increasing the actual cost of registration.

6.3. The distance-based method

The distance-based strategy [4,10] is considered as the most efficient tracking strategy. In this section we show that the distance-based strategy is in fact a special case of the interactive strategy. Consider the case in which $T_i = \infty$ for $\forall i < D$, and $T_i = 1$, for $\forall i \geqslant D$, where D is a pre-defined parameter. This variant of the interactive strategy is in fact the distance-based strategy with a parameter D. The main advantage of the interactive strategy over the distance-based strategy is its *robustness*. The optimal distance-threshold is very sensitive to the user velocity. If the distance threshold is too high, the user never registers, and if it is too low, many unnecessary registration messages are expected. For example, the optimal distance-threshold for pedestrians is very small, while for a vehicular user driving in a highway the optimal distance-threshold is much higher. Hence, the performance of the distance-based strategy is very sensitive to an accurate estimation of the user mobility pattern. Unfortunately, such estimation is not a simple task. To overcome this drawback, the interactive strategy uses a combination of the distance-based tracking with a distance-sensitive timer, to accommodate users with uncertain mobility parameters. This combination provides a distance-driven tracking strategy, in which the registration rate can be adjusted to the user velocity, such that highly mobile users update their location more often than lowly mobile users. Consequently, *the worst case performance of the interactive strategy is bounded from above by a pre-defined parameter*. For example, if $T_i = T$, $\forall i$, the registration rate is bounded from above by $1/T$, and the paging cost is bounded from above by the maximal distance a user can travel in T time units. Hence, the combination of a distance-based tracking with a distance-sensitive timer forms a mechanism that bounds from above the *total* tracking cost.

6.4. Comparison and numerical results

To demonstrate the efficiency of the interactive tracking, its performance is compared to the performance of the existing methods currently used in actual systems.

Consider the special case of the interactive tracking, under which $T_i = T, \forall i$. Let t' be the time interval length between two consecutive registration messages. The expected value of t' is given by:

$$\begin{aligned} E[t'] &= T\sum_{t=1}^{T}\Pr[\theta = t] + \sum_{t=T+1}^{\infty}\Pr[\theta = t]t \\ &\geqslant \sum_{t=1}^{\infty}\Pr[\theta = t]t = E[\theta], \end{aligned} \tag{8}$$

where $E[\theta]$ is the expected *residential interval* θ, defined in section 6.1. It therefore follows from equation (8) that the expected time interval between two consecutive registration messages under the interactive tracking, is higher than the expected *residential interval* of the user. The registration rate R_{I} under this strategy is given by:

$$\begin{aligned} R_{\mathrm{I}} &= \sum_{t=1}^{T}\frac{\Pr[\theta = t]}{T} + \sum_{t=T+1}^{\infty}\frac{\Pr[\theta = t]}{t} \\ &\leqslant \sum_{t=1}^{\infty}\frac{\Pr[\theta = t]}{t} = \sum_{t=1}^{\infty}\frac{(\Pr[\theta = t])^2}{\Pr[\theta = t]t}. \end{aligned} \tag{9}$$

Equations (8)–(9) suggest that the registration rate under the interactive tracking decreases with $E[\theta]$ and, for a good approximation, the registration rate R_{I} can be well estimated by $1/E[\theta]$.

Figure 2 depicts the expected registration rate under the timer-based method, and the expected registration rate R_{I}, as a function of $E[\theta]$. Using equations (8)–(9), R_{I} is estimated by $1/E[\theta]$. The registration rate under the interactive tracking is significantly smaller than the expected registration rate under the timer-based method. This behavior is supported by equation (6). Real cellular networks data indicates that the most loaded time period in a typical working day is between 10 a.m. to 14 a.m. At that time period many users tends to move very rarely. Hence, $E[\theta]$ is expected to be very large, compared to T. As a consequence, the registration rate under the interactive scheme has the potential to be up to 10 times smaller than that of the timer-based method, at the highest loaded time period of a typical working day. Note that, as we show in section 6.1, the paging cost under the interactive tracking is *always* lower than that of the timer-based method (equation (6)).

To compare the performance of the GB method to that of the interactive tracking, we evaluate the paging cost under the following search strategy: Let D be the maximal distance the user can travel during T time units. The search for the user is conducted at its last known location. If it is not found, than the search is conducted simultaneously at all the cells at a distance less than or equal to D. If the user is not found – the search is conducted at all remaining cells in the LA. Using lemma 4.1, the paging cost S_{I} under the variant of the interactive tracking defined above, for a two dimensional system, is bounded from above by:

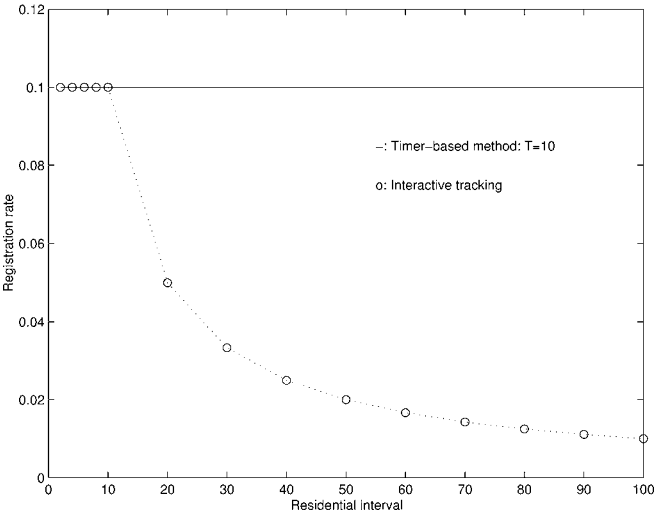

Figure 2. The expected registration rate under the interactive tracking R_{I}, in comparison to the expected registration rate under the timer-based method, as a function of the expected residential interval $E[\theta]$.

$$S_{\mathrm{I}} < 1\cdot\Pr[\theta > T] + O\big(D^2\big)\big(1 - \Pr[\theta > T]\big) = S_{\mathrm{I}}'. \tag{10}$$

The *user mobility* $M(D, \lambda)$ is defined as the probability to travel a distance greater than D between two successive paging event: $M(D,\lambda) = \Pr[\mathit{distance} > D \mid \tau = 1/\lambda]$, where τ is the user *roaming interval*, and λ is the rate of incoming calls directed to the user. Let $\alpha_0 = \Pr[\mathit{distance} = 0 \mid \tau = 1/\lambda]$. The paging cost under the GB strategy, using the same paging strategy described above, is given by:

$$S_{\mathrm{GB}} = 1\cdot\alpha_0 + NM(D,\lambda) + O\big(D^2\big)\big(1 - M(D,\lambda) - \alpha_0\big), \tag{11}$$

where N is the number of cells within the LA. Figure 3 depicts the upper bound on the paging cost under the interactive tracking S'_{I} and the paging cost under the GB strategy S_{GB}, as a function of the *user mobility* $M(D,\lambda)$. We consider a two dimensional system consists of 100 cells, in which $D = 2$, and the number of cells at a distance less than or equal to 2 is 25 cells ($[2D + 1]^2$). It is assumed that the *user mobility* is given by: $M(D,\lambda) = 0.5(1 - \Pr[\theta > T])$, and that $\alpha_0 = 0.3\Pr[\theta > T]$. It can be seen that the worst case paging cost under the interactive tracking S_{I}', is significantly lower than the expected paging cost under the GB strategy. Note that due to the use of a load-sensitive timer, only non-utilized system resources are used for registration, under the interactive tracking.

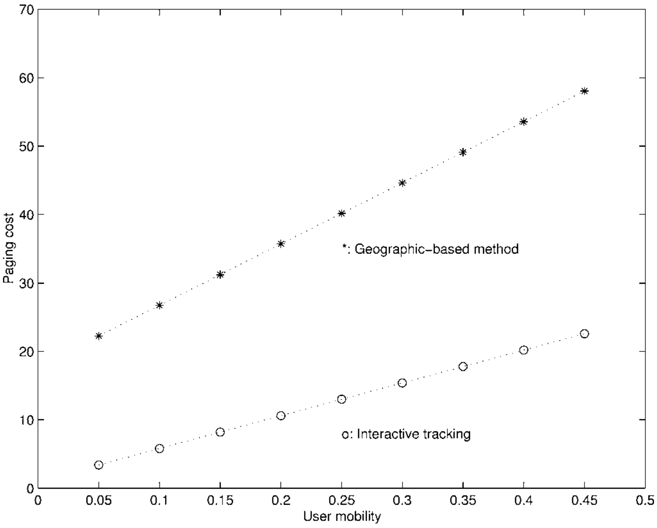

Figure 3. The upper bound on the paging cost under the interactive tracking S_I', in comparison to the expected paging cost under the GB strategy, as a function of the *user mobility* $M(D, \lambda)$.

7. Summary and concluding remarks

In this work we presented an interactive tracking strategy. The user–network interaction is used for an interactive distance tracking, in which the user learns about the system topology, using location information announced by the system as a broadcast message. We showed that the proposed interactive tracking can only perform better than the existing methods used in current systems. Numerical results demonstrate a significant reduction in the expected tracking cost. Under realistic conditions, the proposed method has the potential to reduce the paging cost in about 80%, in comparison to the GB strategy, without increasing the actual cost of registration. The registration rate can be reduced in about 90%, *in addition* to lower paging cost, in comparison to the timer-based method, as used in current networks.

The distance tracking algorithm suggested in this study is independent of the system topology. However, the cost of its implementation does depend on the system topology. Using a distance-based strategy, it is sufficient for most practical cases to use a threshold less than or equal to 3 cells [7]. Moreover, lemma 5.2 and equation (3) imply that even if the proposed distance tracking method is inaccurate, the result, in the worst case, would be a slight increase in the registration rate. For these reasons, and since the diameter of a typical location area used in actual systems is, for a good approximation, about 10 cells, then for most practical cases it is sufficient to inform the user only the list of the nearest neighbors to its location. *This information is already available to the users in CDMA networks, in order to enable soft hand-off.* Taking into consideration that the third generation wireless systems will be based on CDMA, this information can be used *at actually no cost* for efficient distance tracking.

Appendix A. Proof of lemma 5.6

The condition $d^*(x1, x2) > d(x1, x2)$ implies that there exists a path from $x2$ to $x1$, that goes through cells unvisited by the user, that is shorter than the path traveled by the user from $x1$ to $x2$. Lemma 5.2 implies that $d(x1, x2) > 2R$, otherwise $d^*(x1, x2) = d(x1, x2)$. The condition $d^*(x1, x2) > d(x1, x2) > 2R$ therefore implies that $d(x1, x2) \geqslant 2R + 1$, implying that: $d^*(x1, x2) \geqslant d(x1, x2) + 1 \geqslant 2R + 2$.

Consider the minimal case in which $d(x1, x2) = 2R + 1$ and $d^*(x1, x2) = 2R + 2$. Under this condition, none of the cells along the shortest path from $x1$ to $x2$ can be visited by the user. Otherwise, if there exists a cell, say x', such that x' was visited by the user u, and x' is on the shortest path from $x1$ to $x2$, then $d(x', x1) \leqslant 2R$, $d(x', x2) \leqslant 2R$, and $x' \in G_u$. Using lemma 5.2 and the definition of d^*, it follows that $d^*(x1, x2) = d^*(x1, x') + d^*(x', x2) = d(x1, x2) = 2R + 1$. Since $d^*(x1, x2) > d(x1, x2)$, such a cell x' does not exist. Hence, there exists a cycle in G, that goes through $x1$ and $x2$, whose length is at least $(2R + 2) + (2R + 1) = 4R + 3$. Thus, the condition $d^*(x1, x2) > d(x1, x2)$ implies that there exists a cycle in G, whose length is at least $4R + 3$. Therefore, a sufficient condition under which $d^*(x1, x2) = d(x1, x2)$ for any cells $x1, x2$ visited by the user, is that the maximal length of a cycle in G is at most $4R + 2$. $\square$

References

[1] I.F. Akyildiz, J.S.M. Ho and Y.B. Lin, Movement based location update and selective paging schemes, IEEE/ACM Transactions on Networking 4(4) (1996) 629–638.

[2] A. Bar-Noy and I. Kessler, Tracking mobile users in wireless networks, IEEE Transactions on Information Theory 39 (1993) 1877–1886.

[3] A. Bar-Noy, I. Kessler and M. Naghshineh, Topology-based tracking strategies for personal communication networks, ACM Journal of Mobile Networks and Applications (1996).

[4] A. Bar-Noy, I. Kessler and M. Sidi, Mobile users: to update or not to update? Wireless Networks 1(2) (1995) 175–185.

[5] A. Bhattacharya and S.K. Das, LeZi-Update: An information-theoretic approach to track mobile users in PCS networks, in: *Proc. of ACM MobiCom* (1999).

[6] EIA/TIA, Cellular radio-telecommunications inter-system operations, Technical Report IS-41, Revision C (1995).

[7] J.S.M. Ho and I.F. Akyildiz, Mobile user location update and paging under delay constraints, Wireless Networks 1(4) (1995) 413–425.

[8] L. Kleinrock, *Queueing Systems*, Vols 1, 2 (Wiley, New York, 1976).

[9] H. Levy and Z. Naor, Active tracking: Locating mobile users in personal communication service networks, Wireless Networks 5(6) (1999) 467–477.

[10] U. Madhow, L. Honig and K. Steiglitz, Optimization of wireless resources for personal communications mobility tracking, IEEE Transactions on Networking 3(6) (1995) 698–707.

[11] K.S. Meier-Hellstern, E. Alonso and D. O'Neil, The use of SS7 and GSM to support high density personal communications, in: *Wireless Communication: Future Directions*, eds. J.M. Holtzman and D.J. Goodman (Kluwer, 1993) pp. 55–68.

[12] Z. Naor and H. Levy, Minimizing the wireless cost of tracking mobile users: An adaptive threshold scheme, in: *Proc. of IEEE INFOCOM'98* (1998) pp. 720–727.

[13] Z. Naor and H. Levy, LATS: A load-adaptive threshold scheme for tracking mobile users, IEEE/ACM Transactions on Networking 7(6) (1999) 808–817.

[14] Z. Naor and H. Levy, Cell Identification Codes for tracking mobile users, in: *Proc. of IEEE INFOCOM'99* (1999) pp. 28–35.

[15] E. Pitoura and G. Samaras, Locating objects in mobile computing. IEEE Transactions on Knowledge and Data Engineering 13(4) (2001) 571–592.

[16] C. Rose, Minimizing the average cost of paging and registration: a timer-based method, ACM Journal of Wireless Networks 2(2) (1996) 109–116.

[17] C. Rose and R. Yates, Minimizing the average cost of paging under delay constraints, ACM Journal of Wireless Networks 1(2) (1995) 211–219.

[18] O. Wolfson, A. Sistla, S. Chamberlain and Y. Yesha, Updating and querying databases that track mobile units, Distributed and Parallel Databases Journal 7(3) (1999) 257–287.

Zohar Naor received the B.Sc., M.Sc. and Ph.D. degrees in computer science from Tel-Aviv university, Israel, in 1987, 1992, and 2000, respectively. From 1988 to 1995 he has worked in the areas of digital communication and wireless networks. His current research interests include wireless networks, mobile networks, networking and distributed systems.
E-mail: naorz@post.tau.ac.il