Mobile Networks and Applications 1 (1996) 49–56

Topology-based tracking strategies for personal communication networks

Amotz Bar-Noy [a,1], Ilan Kessler [b] and Mahmoud Naghshineh [c]

[a] *Dept. Of EE-Systems, Tel Aviv University, Tel Aviv 69978, Israel*
[b] *Qualcomm Israeli, P.O.B. 719, Haifa 31000, Israel*
[c] *IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA*

Abstract. This paper explores tracking strategies for mobile users in personal communication networks which are based on the *topology* of the cells. We introduce the notion of *topology-based strategies* in a very general form. In particular, the known paging areas, overlapping paging areas, reporting centers, and distance-based strategies are covered by this notion. We then compare two topology-based strategies with the time-based strategy on the line and mesh cell topology.

1. Introduction

1.1. The tracking problem

Location independent communication is one of the key requirements of mobile communication systems such as Personal Communication Networks (PCN). In contrast to fixed communication systems in which every user is attached to a physical port of the network and calls can be directed to specific ports, in a mobile communication environment calls have to be directed to an access point with which the user can establish a connection. This leads to a cellular architecture. The cellular architecture comprises two levels – a stationary level and a mobile level. The stationary level consists of fixed *base stations* that are interconnected through a fixed network, referred to as the *backbone* network. The mobile level consists of mobile units that communicate with the base stations via wireless links. The geographic areas within which mobile units can communicate with a particular base station is referred to as a *cell*. Neighboring cells overlap with each other, thus ensuring continuity of communications. The mobile units communicate among themselves as well as with the fixed information networks through the base stations and the backbone network. (See Fig. 1.)

Roaming users introduce a new problem in connection setup or call initiation process. In order to be able to call a mobile user, the network must be able to locate this user. In other words, the network must find a base station to which the called user can establish a connection. This is referred to as the mobile users *tracking* problem. There are two simple approaches to tracking mobile users, that illustrate the fundamental trade-off in this problem [2]. In the first approach, the network does not try to keep track of users as they move between cells, and starts a search for a mobile user only after this user is called. This approach is referred to as the *Never Update* strategy. This strategy may result in a long search time and a large number of search messages that are broadcasted in a large domain of the network. Both of these drawbacks are highly undesirable since the wireless channel bandwidth is scarce and a paged mobile needs to be located in a rather short period of time. In the second approach, which is referred to as the *Always Update* strategy, the network gets updated every time mobile users move between cells. As a result there are no search messages and paged users are located in a short time. However, updating the new location each time mobile users cross boundaries of cells is again expensive because it is an extensive usage of the wireless bandwidth and the batteries of the mobile terminals. Thus, the problem is to find tracking strategies that minimize the totals cost of update and search.

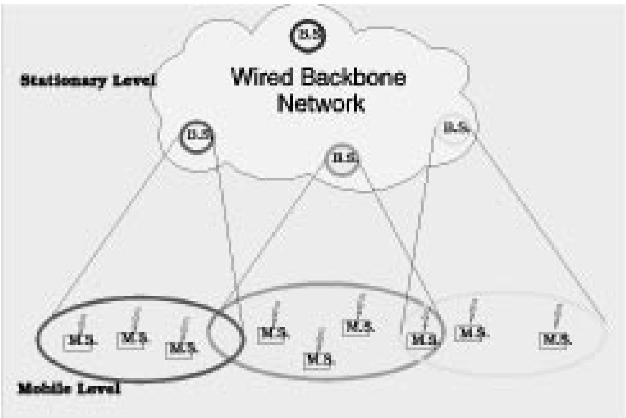


Fig. 1. The cellular architecture. A B.S. is a base station and an M.S. is a mobile station.

Remark. The problem of how to maintain the location

---

[1] Also with IBM T.J. Watson Research Center Yorktown Heights, NY 10598, USA.

(an exact location or an area that contains the exact location) of users in the backbone network is a directory management problem. In the context of mobile users this issue was investigated in [1] and [8]. In this paper we ignore this issue and assume that all the nodes in the backbone network poses the same knowledge regarding the locations of all mobile users.

1.2. Known tracking strategies

In the following we briefly describe tracking strategies that were studied in the literature. For each strategy we describe the *Update* rule and the *Find* rule. The *Update* rule defines when or where a user should report to the system its new location. The *Find* rule defines where to search for a particular user.

We distinguish between two types of tracking strategies. In the first type, the system decides when and where mobile users should update their location. Consequently, if two users travel along the same route they report in the same locations. We call such strategies *static* strategies. In the second type, called *dynamic* strategies, the mobile users themselves decide when and where to report. As a consequence, two mobile users may report in different locations even though they travel along the same route.

The simplest dynamic strategy is the *time-based* strategy [6]. In this strategy, each mobile user updates the system every T units of time. The search for a mobile user is conducted in all cells to which the mobile user can reach in T units of time from the last cell in which it has reported. The *movement-based* strategy and the *distance-based* strategy are also dynamic strategies (see [3] and [7]). In the movement-based strategy each mobile user updates the system whenever it has completed M movements between cells. The search for a mobile user is conducted in all cells to which the mobile user can reach by M movements from the last cell in which it has reported. In the distance-based strategy, each mobile user updates the system whenever it is in distance D from the last cell in which it has reported. The search for a mobile user is conducted in all cells whose distance from the last cell in which the user has reported is less than D.

The simplest static strategy is the *paging areas* strategy. In this strategy, the cells are clustered into disjoint groups which are referred to as paging areas. Mobile users report their location only when they arrive at a new paging area. The search for a mobile user is conducted only in one particular paging area. The paging areas strategy has the following drawback. A mobile user traveling along the border of a paging area could send many update messages if it has an oscillatory pattern of movement. The *overlapping paging areas* strategy is similar to the paging areas strategy, except that here the cells are clustered into non-disjoint groups. This allows the elimination of the above drawback. In contrast to the paging areas strategy where the mobile user has one choice of how to select its new paging area, in the overlapping paging areas strategy there could be many choices. In [5], the new paging area is determined by the base station associated with the new cell, independently of where the user came from and of its last paging area.

Another static strategy with a different approach is the *reporting centers* strategy [2]. Here, a subset of all cells is selected and designated as reporting cells. The idea is that mobile users report only upon entering a reporting cell, and a search for any user is restricted to the vicinity of a reporting cell. The vicinity of a cell is the set of all cells to which a mobile user can move without crossing another reporting cell. Assuming that each base station is transmitting a beacon every fixed period of time, the reporting centers strategy may be implemented by including in the beacon one bit which indicates whether the base station is a reporting center or not. Note that if a mobile user stays for a long time around a particular reporting cell, moving into and out of this cell many times, this may result in many unnecessary updates. To avoid this, one can add the rule that a mobile user never reports in two consecutive visits to the same reporting center. In fact, there could be many other such rules. We refer to the reporting centers strategy with a restricting rule as the *generalized reporting centers* strategy. It is not difficult to see that the overlapping paging areas and the generalized reporting centers strategies are essentially the same. This is true since one can designate all the border cells of all paging areas as reporting centers and add the appropriate rules.

1.3. Contribution of the paper

In this paper we introduce the notion of *topology-based* strategies. We concentrate on strategies that do not depend on the full history of the users and / or system. This class of strategies contains many of the known strategies. We then explore tracking strategies in personal communication networks which are based on the topology of the cells. We compare the paging area strategy, the generalized reporting centers strategy (the overlapping paging areas strategy), and the time-based strategy. The comparisons are done on the line and mesh topologies. We show that on the line topology, for any i.i.d. patterns of movements the generalized reporting centers strategy outperforms the time-based strategy and the paging area strategy. For the mesh topology, we define a special generalized reporting centers strategy. Then we show that for reasonable i.i.d. patterns of movements, the generalized reporting centers strategy outperforms the time-based strategy and the paging area strategy. However, there exist patterns of movements in which the paging areas strategy and the time based strategy are superior to the generalized reporting centers strategy.

In section 2 we define the notion of topology-based strategies and show how this definition covers existing strategies. In section 3 we analyze an i.i.d. pattern of movement for both the line and the mesh topologies for various tracking strategies. In section 4 we compare the performance of the strategies analyzed in section 3, by presenting some analytic results and some numerical results.

2. Topology-based tracking strategies

In this section we first define the class of topology-based tracking strategies, and then show how known strategies fall into this class.

2.1. Definition

We first introduce a number of notions which are required for the definition of topology-based strategies. Let the *mobility graph* of the network be the directed graph in which each vertex corresponds to a cell, and there is a directed edge from vertex v to vertex w if and only if the corresponding cells overlap. Note that when overlapping is a symmetric property, it follows that if there is a directed edge from vertex v to vertex w then there is also a directed edge in the opposite direction.

In every tracking strategy, at any time there is associated with each user a set of cells that defines the possible locations of the user at that time. Thus, if there is a need to locate the user, the system searches the user only within these cells. This is formalized as follows. Let $G = (V, E)$ be a mobility directed graph where $V = \{1, \dots, n\}$. At time t there is associated with user x a set of vertices, $\lambda_x(t)$, that contains the vertex in which user x is located at time t. The set $\lambda_x(t)$ is referred to as the *location area* of user x at time t. Let $S_1, \dots, S_k$ be all possible location areas in G under the particular tracking strategy that is being used for user x, and denote $\mathcal{S} = \{S_1, \dots, S_k\}$. Clearly $\cup_{i=1}^{k} S_i = V$, i.e., $\mathcal{S}$ is a cover for the mobility graph G.

As an example, consider the two extreme strategies, the Always Update and the Never Update strategies, described in the introduction. Under the Always Update strategy, each set in the partition is a vertex of G, i.e., $\mathcal{S} = \{\{1\}, \dots, \{n\}\}$. Under the Never Update strategy there is only one set in the partition which contains all the vertices of G, i.e., $\mathcal{S} = \{V\}$.

At any time t, the identity of the location area $\lambda_x(t)$ is explicitly or implicitly known to user x. The rules by which $\lambda_x(t)$ is determined depend on the particular tracking strategy that is being used. Whenever the location area of user x changes (i.e., $\lambda_x(t) \neq \lambda_x(t^-)$), the user sends an update message to the fixed network, and in this way lets the system know about the new location area.

We define a *topology-based* strategy to be a strategy in which the location area $\lambda_x(t)$ depends only on the following: (i) The current cell in which the user is located, (ii) the previous cell in which the user was located, and (iii) the location area that the user belonged to while being in the previous cell. More precisely, assume that user x is in vertex v at time t' and first moves out from v at time t. Let w be the new vertex at time t. Then $\lambda_x(t)$ is a function of (v, w) and $\lambda_x(t')$ only. We denote this function by f. Thus, by definition $f : E \times \mathcal{S} \to \mathcal{S}$. Note that in order that the strategy be feasible, $f((v, w), S_i)$ is defined only if $v \in S_i$, and the value of $f((v, w), S_i)$ is a set S_j such that $w \in S_j$.

Remark. Our definition for topology-based strategies includes only strategies with "Markovian" memory, i.e., the current location area does not depend on cells or location areas in the past except the last ones. However, this definition can be generalized to include strategies in which the current location area depends on the entire history of the user (or any part of it).

2.2. Examples: known strategies

The simplest topology-based strategy is the paging areas strategy. Here, the partition $\mathcal{S}$ consists of all the paging areas. Since $\mathcal{S}$ is a disjoint partition, it follows that the function f is uniquely defined as follows: $f((v, w), S_i) = S_i$ if v and w belong to the same paging area S_i and $f((v, w), S_i) = S_j$ if v belongs to S_i and w belongs to S_j.

The overlapping paging areas strategy is also a topology-based strategy. Again, the partition $\mathcal{S}$ consists of all the paging areas. However, the function f can be chosen in many ways. In [5], for instance, $f((v, w), S_i) = f((u, w), S_j$ for any $u \neq v$ as long as $w \notin S_i$ and $w \notin S_j$. That is, f is chosen such that the new location area depends on the current cell and it is a function of w only.

In the reporting centers strategy the number of sets in $\mathcal{S}$ is the same as the number of reporting centers. Each set is a vicinity of some reporting center. Since each reporting center belongs to only one vicinity (its own), it follows that the function f is uniquely defined.

The distance-based strategy with parameter D can be described as follows. The partition $\mathcal{S}$ consists of n sets. For each vertex $v \in V$ the set S_v contains all the vertices $w \in V$ such that the distance in G between v and w is less than D. The function f is defined as follows: $f((v, w), S_u) = S_u$ if $w \in S_u$ and $f((v, w), S_u) = S_w$ if $w \notin S_u$.

The time-based and the movement-based strategies are not topology-based strategies.

3. Analysis

In this section we analyze the performance of the paging areas strategy and the generalized reporting centers strategy, both are topology-based strategies, as well as the performance of the time-based strategy (which is not a topology-based strategy). The results obtained in this section will be used in the next section to compare the various strategies.

3.1. The model

We first introduce the mathematical model that is used for the analysis. We consider two topologies – the infinite line and the infinite mesh. In the infinite line mobility graph, for every integer i cells i and $i+1$ are neighboring cells. Thus, a mobile user that is in cell i, can only move to cells $i+1$ or $i-1$ or remain in cell i. In the infinite mesh mobility graph, for any two integers i and j cell (i,j) is a neighbor of the following eight cells: $(i \pm 1, j \pm 1), (i \pm 1, j), (i, j \pm 1)$. Therefore, a mobile user that is in cell (i,j) can either move to one of these eight cells or remain in cell (i,j).

To model the movement of the mobile users in the network we make the following assumptions. These assumptions facilitate the analysis.

- We assume that time is slotted.
- We assume that the movements of every user are independent of the movements of other users.
- For each user we assume an independent and identically distributed (i.i.d.) movement pattern, as follows. In the line mobility graph, if a user is in cell i at the beginning of a slot, then during the slot it moves to cell $i+1$ with probability p, moves to cell $i-1$ with probability q, or remains in cell i with probability $1-p-q$ $(0 < p+q \leq 1)$, independently of its movements in other slots. In the mesh mobility graph, if a user is in cell (i,j) at the beginning of the slot, then during the slot it moves to one of the eight neighbors of cell (i,j) or remains in cell (i,j) according to the following rule: Along each axis the user moves as in the line mobility graph with probabilities p_x and q_x (for the X axis) and p_y and q_y (for the Y axis), respectively, where the movements along the X and Y axes are independent.
- We assume that updates and searches are done at the beginning of slots, such that if a user is both updating and being searched in the same slot then the update operation precedes the search. Movements between cells in any slot are assumed to start after all updates and searches associated with that slot are done, and all movements are completed by the end of the slot.
- We consider only distance-based strategies that incur symmetric location areas. Consequently, we assume that searches take place only after the system has reached a steady-state, and the time of search is chosen independently of the evolution of the system. This implies that the probability that the system is in a particular state at the time of search is the stationary probability of that state. We assume that the search operation is fast, and terminates within one slot.

The performance measures that we consider in this paper are the expected number of update messages per slot transmitted by a user, denoted by $\mathcal{U}_s$, and the number of all possible cells in which the mobile user can be since its last update, denoted by $\mathcal{A}_s$, where s is the strategy under which the measure is taken.

Since the movements of different users are independent, it suffices to calculate the above performance measures for one user. Thus, throughout the analysis presented in the sequel we refer to a particular user.

3.2. Paging areas strategy

Consider the line mobility graph, and let the line be partitioned into segments of length L, each of which is a paging area. Let the cells of each paging area be numbered from left to right by $1, 2, \dots, L$. Let $\{X_i, i \geq 0\}$ be the cell in which the user is located at time i. Then $\{X_i, i \geq 0\}$ is a Markov chain with stationary probabilities $\pi_d = 1/L$ for all $d = 1, \dots, L$. The user reports if and only if either it is in cell 1 and moves to the left or it is in cell L and moves to the right. Therefore,

$$\mathcal{U}_{\mathrm{PA}}^{\mathrm{line}}(p,q,L) = (p+q)/L\,. \tag{1}$$

Since there are L cells in a paging area, the search cost is given by

$$\mathcal{A}_{\mathrm{PA}}^{\mathrm{line}}(L) = L\,. \tag{2}$$

Consider now the mesh mobility graph, and let the mesh be partitioned into rectangles of size $L_x \times L_y$, each of which is a paging area. As in the one dimensional case, the location of the user at time i can be described by a Markov chain. In fact, since the movements along the X and Y axes are independent, the above Markov chain can be decomposed into two identical and independent Markov chains, one for each dimension. Let R_x (R_y) be the event on the X (Y) Markov chain due to which the user reports. Let $\pi(R_x)$ and $\pi(R_y)$ be the stationary probabilities of these events. We have

$$\begin{aligned}\mathcal{U}_{\mathrm{PA}}^{\mathrm{mesh}} &= \pi(R_x \cup R_y) = \pi(R_x) + \pi(R_y) - \pi(R_x)\pi(R_y)\\ &= \mathcal{U}(x) + \mathcal{U}(y) - \mathcal{U}(x)\mathcal{U}(y)\,,\end{aligned} \tag{3}$$

where $\mathcal{U}(x)$ and $\mathcal{U}(y)$ are the update rates due to movements on the X and Y axes respectively. Thus, we can obtain the update rates in the two-dimensional case using the results obtained above. This yields

$$\begin{aligned}&\mathcal{U}_{\mathrm{PA}}^{\mathrm{mesh}}(p_x, q_x, p_y, q_y, L_x, L_y)\\ &\quad= \frac{p_x + q_x}{L_x} + \frac{p_y + q_y}{L_y} - \frac{p_x p_y + p_x q_y + q_x p_y + q_x q_y}{L_x \cdot L_y}\,.\end{aligned} \tag{4}$$

The search cost would be simply the number of base stations or cells in the paging area which is

$$\mathcal{A}_{\mathrm{PA}}^{\mathrm{mesh}}(L) = L_x \cdot L_y\,. \tag{5}$$

3.3. Generalized reporting centers strategy

On the line mobility graph, let cells iK be reporting centers for every integer i and some parameter K. Each vicinity is of size $2K-1$ and two neighboring vicinities intersect in $K-1$ cells. We adopt the rule that a mobile user never reports in two consecutive visits to the same reporting center.

Let T_i be the number of slots that elapse between the ith and the $(i+1)$st reports of the mobile user. To obtain $E(T_i)$ we note that this is a problem of random walk on a line with absorbing barriers at 0 and $2K$, where the user starts from the initial position K, and where the user moves with probability $p+q \leq 1$. Thus the T_i's are independent and identically distributed random variables, and therefore $\mathcal{U}_{\mathrm{RC}}(p,q,K) = 1/E(T_1)$. An expression for $E(T_1)$ can be obtained from [4], which yields

$$\mathcal{U}_{\mathrm{RC}}^{\mathrm{line}}(p,q,K) = \begin{cases} \frac{(q-p)}{K}\frac{\left(\left(\frac{q}{p}\right)^K+1\right)}{\left(\left(\frac{q}{p}\right)^K-1\right)} & \text{if } q \neq p\,, \\ \frac{2p}{K^2} & \text{if } q = p\,. \end{cases} \tag{6}$$

The search cost is the number of vertices in the vicinity of the reporting center which is

$$\mathcal{A}_{\mathrm{RC}}^{\mathrm{line}}(K) = 2K-1\,. \tag{7}$$

On the mesh mobility graph we design the overlapping paging areas by partitioning the mesh into rectangles of size $(2K_x-1)\times(2K_y-1)$, each of which is a paging area such that two adjacent paging areas intersect in a rectangle of size $K_x \times K_y$.

The implementation of these overlapping paging areas by generalized reporting centers is as follows. Each K_xth column and K_yth row are designated in their entire as reporting centers. Each mobile user maintains two numbers: the last column at which it has reported and the last row at which it has reported. The mobile user updates its location only if at least one of these two numbers is changed. This can be done by numbering the rows and the columns alternately by 0 and 1. The mobile user need only to remember if the last row or column is 0 or 1.

Similarly to the case of paging areas, it can be shown that (3) holds also in this case. This yields

$$\begin{aligned}\mathcal{U}_{\mathrm{RC}}^{\mathrm{mesh}}(p_x,q_x,p_y,q_y,K_x,K_y) &= \mathcal{U}_{\mathrm{RC}}^{x}(p_x,q_x,K_x,K_y) \\ &\quad + \mathcal{U}_{\mathrm{RC}}^{y}(p_y,q_y,K_x,K_y) - \mathcal{U}_{\mathrm{RC}}^{x}(p_x,q_x,K_x,K_y) \\ &\quad \times \mathcal{U}_{\mathrm{RC}}^{y}(p_y,q_y,K_x,K_y)\,. \end{aligned} \tag{8}$$

The search cost in the generalized reporting centers strategy is the size of the vicinity which is given by

$$\mathcal{A}_{\mathrm{RC}}^{\mathrm{mesh}}(K) = (2K_x-1)(2K_y-1)\,. \tag{9}$$

3.4. Time-based strategy

In the time-based strategy, the mobile user transmits an updating message once in T slots. Therefore, the update cost is independent of the structure of the mobility graph and is

$$\mathcal{U}_{\mathrm{Time}}^{\mathrm{line}}(T) = \mathcal{U}_{\mathrm{Time}}^{\mathrm{mesh}}(T) = \frac{1}{T}\,. \tag{10}$$

On the line mobility graph, when a mobile user is paged, it might be located in one of the $2T-1$ cells centered around the cell to which the mobile user last reported. Therefore, the search cost is

$$\mathcal{A}_{\mathrm{Time}}^{\mathrm{line}}(T) = 2T-1\,. \tag{11}$$

On the mesh mobility graph when a mobile user is paged, it might be located in one of the $(2T-1)^2$ cells arranged as a square and centered around the cell to which the mobile last reported. Therefore, the search cost is

$$\mathcal{A}_{\mathrm{Time}}^{\mathrm{mesh}}(T) = (2T-1)^2\,. \tag{12}$$

4. Comparisons

In this section we compare the performance of the three strategies analyzed in the previous section. In subsections 4.1 and 4.2 we present some analytic comparisons and in subsection 4.3 we present some numerical results. For simplicity we assume that in the mesh topology, $L_x = L_y$ for the paging areas strategy and $K_x = K_y$ for the generalized reporting centers strategy. Similar results hold for the more general case.

4.1. Analytic comparisons – the line

The comparisons are done by letting

$$\mathcal{A}_{\mathrm{PA}}^{\mathrm{line}} = \mathcal{A}_{\mathrm{RC}}^{\mathrm{line}} = \mathcal{A}_{\mathrm{T}}^{\mathrm{line}} = 2T-1$$

and then comparing $\mathcal{U}_{\mathrm{PA}}^{\mathrm{line}}$, $\mathcal{U}_{\mathrm{RC}}^{\mathrm{line}}$, and $\mathcal{U}_{\mathrm{T}}^{\mathrm{line}}$ pairwise.

The paging areas strategy vs. the time-based strategy: We get that $\mathcal{U}_{\mathrm{PA}}^{\mathrm{line}} = (p+q)/(2T-1)$ which is smaller than $\mathcal{U}_{\mathrm{T}}^{\mathrm{line}} = 1/T$ when $T \geq 2$. For large T, $\mathcal{U}_{\mathrm{PA}}^{\mathrm{line}}$ is less than half of $\mathcal{U}_{\mathrm{T}}^{\mathrm{line}}$. The advantage of the paging areas strategy over the time-based strategy increases when the mobile user is more stationary, i.e., when $p+q$ decreases.

The generalized reporting centers strategy vs. the time-based strategy: For large T and fixed $p \neq q$, we get that $\mathcal{U}_{\mathrm{RC}}^{\mathrm{line}} \approx |q-p|/T$ which is smaller than $\mathcal{U}_{\mathrm{T}}^{\mathrm{line}} = 1/T$ as long as $q < 1$ and $p < 1$. The advantage of the generalized reporting centers strategy over the time-based strat-

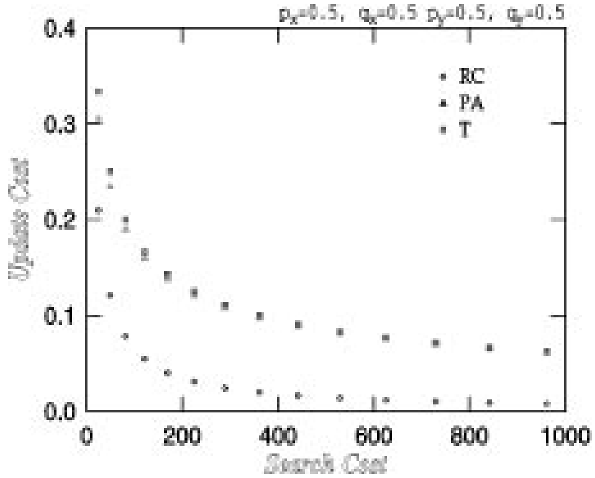


Fig. 2. Mobiles are always moving randomly in all directions.

egy increases when the walk of the mobile user is more random, i.e., when $|p-q|$ is small.

The paging areas strategy vs. the generalized reporting centers strategy:

- $p=q$: It follows that the generalized reporting centers strategy is superior, since $\mathcal{U}_{\mathrm{RC}}^{\mathrm{line}}=2p/T^2\leq 2p/(2T-1)=\mathcal{U}_{\mathrm{PA}}^{\mathrm{line}}$ for $T\geq 1$. For large T the advantage is substantial.
- $q\neq p$: For large T, it follows that $\mathcal{U}_{\mathrm{RC}}^{\mathrm{line}}\approx|q-p|/T$ and $\mathcal{U}_{\mathrm{PA}}^{\mathrm{line}}\approx(q+p)/2T$. Therefore, the paging areas strategy is better when $p\leq q/3$ and worse otherwise. This means that when the mobile "prefers" one direction the paging areas strategy is better and when its walk is more random the generalized reporting centers strategy is better.

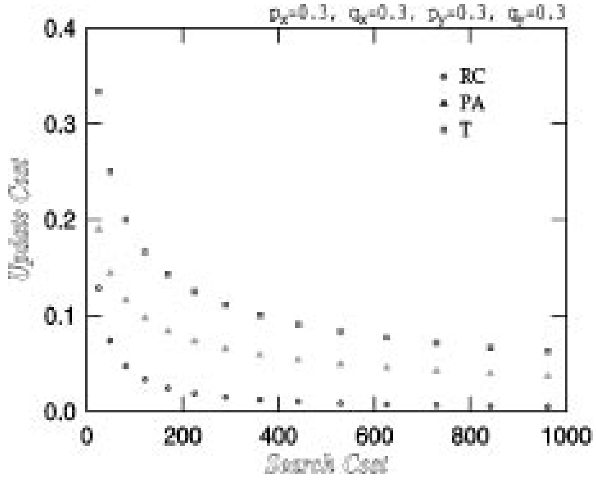


Fig. 3. Mobiles are moving randomly in all directions but stay in their place with some probability.

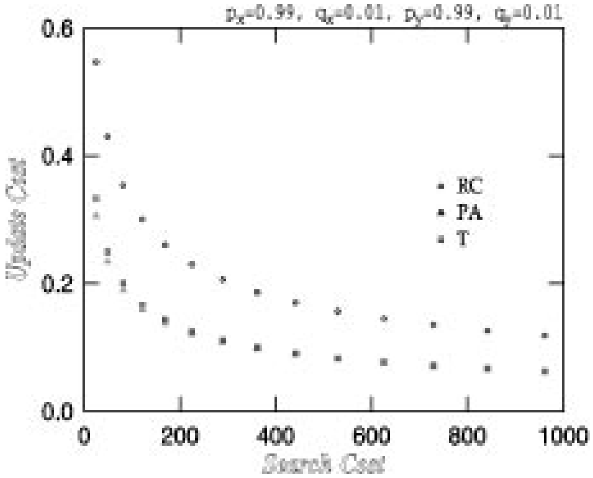


Fig. 4. Mobiles are always moving in one direction.

4.2. Analytic comparisons – the mesh

The pairwise comparisons of $\mathcal{U}_{\mathrm{PA}}^{\mathrm{mesh}}$, $\mathcal{U}_{\mathrm{RC}}^{\mathrm{mesh}}$, and $\mathcal{U}_{\mathrm{T}}^{\mathrm{mesh}}$ are done as follows. We assume that

$$\mathcal{A}_{\mathrm{PA}}^{\mathrm{mesh}}=\mathcal{A}_{\mathrm{RC}}^{\mathrm{mesh}}=\mathcal{A}_{\mathrm{T}}^{\mathrm{mesh}}=(2T-1)^2\,.$$

We then fix $q_x\geq p_x$ and $q_y\geq p_y$ and let T be very large.

The paging areas strategy vs. the time-based strategy: It follows that $\mathcal{U}_{\mathrm{PA}}^{\mathrm{mesh}}\approx(p_x+q_x+p_y+q_y)/2T\leq 1/T$. Again, if the mobile is more stationary the paging areas strategy is better.

The generalized reporting centers strategy vs. the time-based strategy: It follows that $\mathcal{U}_{\mathrm{RC}}^{\mathrm{mesh}}\approx(q_x-p_x+q_y-p_y)/T$. Consequently, the time-based strategy is better if $(q_x+q_y)-(p_x+p_y)>1$, i.e., when the mobile user "prefers" one direction and its walk is not random.

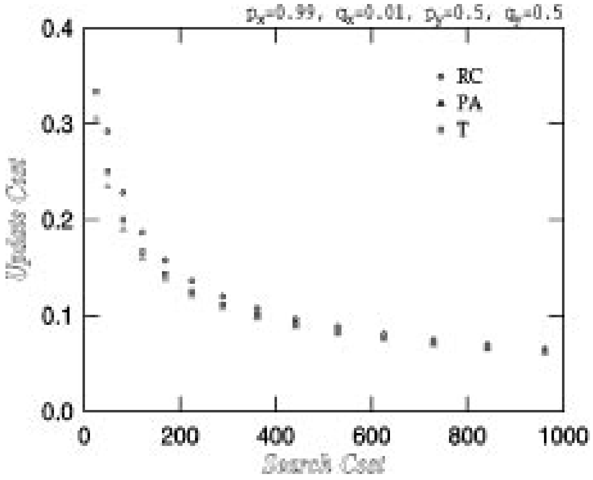


Fig. 5. Mobiles are always moving, to the right in the X direction and randomly in the Y direction.

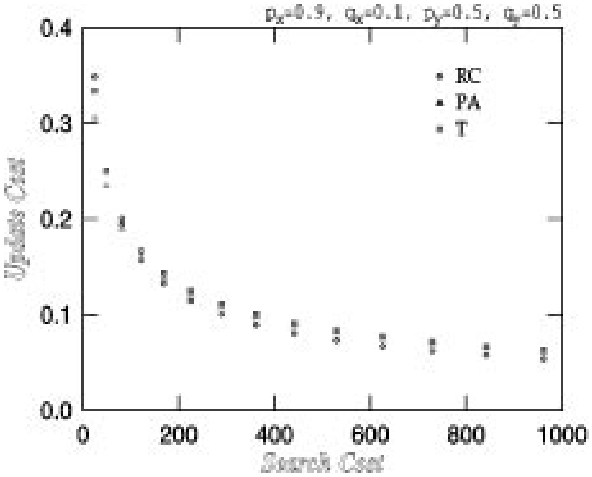


Fig. 6. Mobiles are always moving, mainly to the right in the *X* direction and randomly in the *Y* direction.

The paging areas strategy vs. the generalized reporting centers strategy:

- $p_x = q_x$ and $p_y = q_y$: As in the line graph, since $T^2 \leq 2T$, it follows that the generalized reporting centers strategy is always better and its advantage increases when T is very large.

- $q_x > p_x$ and $q_y > p_y$: Here, the paging areas strategy is better when $3(p_x + p_y) \leq q_x + q_y$. In such cases, the mobile user "prefers" a direction. However, it follows that $\mathcal{U}_{\text{RC}}^{\text{mesh}} \leq 2\mathcal{U}_{\text{PA}}^{\text{mesh}}$ for all values of p_x, q_x, p_y, q_y, while $\mathcal{U}_{\text{PA}}^{\text{mesh}} \ll \mathcal{U}_{\text{RC}}^{\text{mesh}}$ when $q_x - p_x$ and $q_y - p_y$ are very small (as in the previous case).

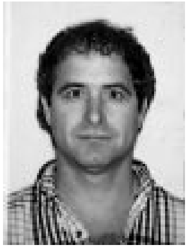
4.3. Numerical comparisons

We compare the performance of the three strategies on the mesh mobility graph in Figs. 2–6. In each figure, we plot the update cost which is the number of update messages a mobile sends per unit of time as a function of the search cost for the paging area strategy (denoted by PA), for the generalized reporting centers strategy (denoted by RC) and for the time-based strategy (denoted by T). Each figure has a different set of values for p_x, q_x, p_y, and q_y.

Figs. 2 and 3 show that in the case of a symmetric movement in all directions RC is more efficient than both PA and T. Also, when the user is not moving all the time, (the case of Fig. 3), PA is better than T. On the extreme, Fig. 4 shows that in a directed movement with very little randomness, PA and T perform better than RC. This is true because in this case there is no need for the complicated paging areas of RC. Fig. 5 depicts the case where the movement is directed in one direction (X component) and random in the other direction. In this case, PA is slightly better than T and T is slightly better than RC. However, if we increase the randomness of movement in the X direction, as in Fig. 6, then RC outperforms PA and T.

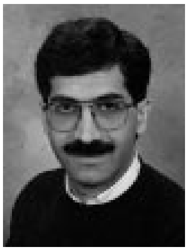
References

[1] B. Awerbuch and D. Peleg, Sparse partitions, *Proc. 31st Symp. on Foundation of Computer Science* (1990) pp. 503–513.
[2] A. Bar-Noy and I. Kessler, Tracking mobile users in wireless networks, IEEE Trans. Inform. Theory 39 (1993) 1877–1886.
[3] A. Bar-Noy, I. Kessler and M. Sidi, Mobile users: To update or not to update?, *Proc. INFOCOMM '94*.
[4] W. Feller, *An Introduction to Probability Theory and its Applications* (Wiley).
[5] Fukumine at al., Method for registering mobile station in mobile communication systemxa, US Patent 5212822 (1993).
[6] W. Lee, *Mobile Cellular Telecommunications Systems* (McGraw-Hill, New York, 1989).
[7] U. Madhow, M.L. Honig and K.S. Steiglitz, Optimization of wireless resources for personal communications mobility tracking, *Proc. INFOCOMM '94*.
[8] S.J. Mullender and P.B.M. Vitányi, Distributed match-making, Algorithmica 3 (1988) 367–391.



Amotz Bar-Noy received the B.Sc. degree in 1981 in mathematics and computer science and the Ph.D. degree in 1987 in computer science, both from the Hebrew University, Israel. From 1987 to 1989 he was a post-doc fellow in Stanford University, California. In 1989 he joined IBM T. J. Watson Research Center, New York, as a Research Staff Member. Since 1995 he is also with the Electrical Engineering Department of Tel Aviv University, Israel. His main areas of interest are: communication networks, combinatorial optimization and algorithms, parallel and distributed computing.
E-mail: amotz@eng.tau.ac.il

Ilan Kessler received the B.Sc., M.Sc., and D.Sc. degrees in electrical engineering from the Technion, Israel Institute of Technology, Haifa, Israel, in 1979, 1986, and 1990, respectively. From 1979 to 1983 he was a technical officer in the Israeli Defense Force, where he was responsible for the development and deployment of mobile data networks. From 1990 to 1994 he was a Research Staff Member at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, where he worked on design and analysis of algorithms for mobile wireless networks and high-speed communication networks. In February 1994 he joined Qualcomm Inc., San Diego, CA, and since August 1994 he is with Qualcomm Israel, Haifa, Israel, where he has been leading a group working on a CDMA-based satellite network.
E-mail: ilan@qualcomm.com



Mahmoud Naghshineh is a Research Staff Member at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, where he currently works in the wireless and mobile networks group. He joined IBM in 1988. From 1988 to 1991, he worked on a variety of research and development projects dealing with design and analysis of local area networks, communication protocols, and fast packet-switched/broadband networks. Since 1991, he has been working in the area of wireless and mobile ATM, wireless access broadband networks, and mobile and wireless local area networks. He received his doctoral degree form Columbia University, New York, in 1994, M.S. in electrical engineering and B.S. in computer engineering from Polytechnic University, New York, in 1991 and 1988, respectively, and the Vordiplom degree in electrical engineering from RWTH Aachen, Germany, in 1985. He is a member of the IEEE communication society, and a member of the IEEE technical committee on computer communications as well as technical committee on personal communications. He has served as a member of technical program committee, session organizer and chairperson for many IEEE conferences and workshops. He is also an editor of the IEEE personal communications magazine. Currently, he is an adjunct faculty member of the department of electrical engineering at Columbia university where he teaches a course on wireless/mobile communications and networking. He has published numerous technical papers and holds a number of IBM awards and patents in the area of high-speed and wireless/mobile networks.
E-mail: Mahmoud@watson.ibm.com