Location Updates and Probabilistic Tracking Algorithms for Mobile Cellular Networks[*]

John C.S. Lui[†], Cedric C.F. Fong
Department of Computer Science & Engineering
The Chinese University of Hong Kong, Hong Kong
email: {cslui,cffong}@cse.cuhk.edu.hk

H.W. Chan
Department of Computer Science
Hong Kong University, Hong Kong
email: hwchan@csis.hku.hk

Abstract

In this work, we develop a novel mathematical model to analyze different location update protocols for mobile cellular network. Our model can capture many important features of user movements, such as geographically dependent movement, directional movement and non-exponential distribution of cell dwell times. We propose to use the transient method to predict the location of a mobile user. This method is efficient and, more importantly, able to address the temporal change in the current location of a mobile user. Therefore, the accuracy of the prediction of the location of a user can be enhanced. We also propose a new location update protocol which can attain a lower tracking cost and also adapt to different mobility patterns. Transient performance analysis is also carried out and the results are compared with other protocols.

1 Introduction

In this work, we consider the *mobile terminal tracking* problem, which is a mechanism used in the cellular network so as to enable the personal communication network (PCN) to locate a disconnected mobile computer and re-establish a connection with it. The mobile terminal tracking problem has been studied extensively [4, 3, 5, 9, 10, 1] with the target at minimizing the consumption of system resources, especially communication overhead. This is important because the limited system resources are shared by a large number of mobile computers. An effective utilization of these resources can increase the system capacity and also enhance performance.

If a mobile computer moves from one registration area to another, it will send a location update message to the base station in the cell in the new registration area. The static part of the PCN (wired network) will then update the records about the location of that mobile computer. Therefore, a location update consumes wireless communication bandwidth, database update overhead and the system signaling cost inside the wired network.

The configuration and protocol used in the GSM or IS-41 standards, however, provides potential performance problems. Whenever a mobile computer moves from one registration area to another, an update message will be sent. As a result, if a computer moves around the boundary between two adjacent registration areas, unnecessary overhead is incurred. In order to alleviate this problem, other dynamic approaches to assign the registration area are proposed [1, 4, 9, 13]. Rather than fixing the boundaries of the registration areas, one alternative is to draw a new registration area for each update such that the reporting cell will be the center of the new area. This registration area can be specified in terms of a number of *rings* of cells around the last reporting cell. When a mobile computer leaves its registration area, it means that it has left its last reporting location for a certain distance. Then, an update message has to be sent. It is called the distance-based protocol [1, 9]. Apart from the distance-based protocol, another protocol, called the time-based protocol, is also proposed in [4, 10]. In this protocol, a specific time limit is set. If a mobile computer has not sent any location update for this time limit, it will send an update message to the wired system.

In the above various update protocols, the mobile computer does not need to send update messages for every change in its cell location. As a result, the wired system does not know the exact location of the computer. Whenever the wired system tries to establish a communication channel with the computer, it has to find out where the computer currently is. This involves a *paging* process [2, 5, 14, 11] in which the base stations send polling messages to the cells in the network and wait for the reply of the target computer. This process all induces system overhead in both system signal and wireless bandwidth consumption. If the wired network knows the exact location of a mobile computer, the paging cost can be reduced to a minimum by polling only the cell in which the computer is situated. On the other extreme case, if the wired network does not have any information about the location of the mobile computer, cells all over the wireless network have to be polled. This costs the maximum system overhead.

Therefore, another goal of the location update protocol is to maintain the location information (though imprecise) of the mobile computers, so that the paging cost can be lowered. Furthermore, the more accurate the location information is, the lower the paging cost can be attained. On the other hand, extra costs are required for the mobile computers to send location update messages more frequently so that those information can be more up-to-date.

In [1, 4, 9, 10], the authors have done mathematical analyses to show the performance of the tracking protocols in balancing these tradeoffs. In [1, 4], however,

---

[†]Corresponding author

they assume that the user has a uniform movement pattern, *i.e.* the user moves from one cell to one of the neighboring cells with even probability and at the same movement rate. Moreover, they use steady state analysis to quantify the performance of their protocol, which is only valid for their simplified movement model. These assumptions may not be general enough to incorporate realistic situations. A user usually has a unique, non-symmetric movement pattern over the cellular network. In addition, the cost of paging depends on the actual movement of the mobile user. As a result, steady state analyses may not be appropriate.

One contribution of this paper is to develop a mathematical framework that enables us to model general user movement patterns inside the cellular network. This model is more general than the one proposed in [1, 4] because it allows non-uniform movement of a mobile terminal inside the cellular network. This is much more realistic as under practical cases where users have general habitual movement patterns. Moreover, we can also incorporate other movement information, such as directional movement, so that the prediction accuracy can be enhanced. Also, we do not restrict that the distribution of cell dwell time of a mobile computer be exponential. Another important advantage of this model is to provide a unified framework to compare the performance of different tracking protocols. Under this framework, we can also analyze the impacts of different movement patterns to the protocols. Last but not the least, we have proposed a new protocol, the reverse-guessing protocol. This protocol uses an heuristic based on the prediction result obtained in our proposed method. We find that our method outperforms the time-based and the distance-based methods. The main advantage of this protocol is that our reverse-guessing protocol is more adaptive in choosing the right time for the mobile computer to send an update message under general movement patterns.

This paper is organized as follows. In Section 2, we present our mathematical model for the prediction of the current location of a mobile computer. The evaluation method will also be presented. Next, in Section 3, we present the time-based update and the distance-based protocols. Also, we will illustrate using the transient analysis to evaluate the protocol. In Section 3.3, we propose our new update protocol and the rationale behind it. Lastly, in Section 4, we carry out experiments and show the effectiveness of our protocol method as compared to the existing methods and proposals.

2 Movement & Location Prediction

In this section, we present a novel movement model for a mobile computer inside a cellular network. To begin with, we develop a movement model that can model geographically dependent movement patterns that are more realistic in real situations. A method to predict the probable locations of a mobile computer is then presented. Afterwards, we extend our model to capture more important movement parameters such as directional movement and non-exponential distribution of cell dwell times. Lastly, we introduce the approximation method to enhance the efficiency of the prediction method.

2.1 Movement Model

In general, the movement pattern of a mobile computer is independent of others. So, henceforth, we consider only the movement of a single mobile computer inside the network. In our model, there is a cellular network that is composed of a collection $\mathcal{C}$ of N cells, which are labeled $c_1, c_2, \ldots, c_N$ respectively. The cells are closely packed in an orderly hexagonal fashion.

We model the mobility of the computer by an $N \times N$ *mobility matrix* $\mathbf{Q} = [q_{ij}]$ where q_{ij} is the *rate* at which the mobile computer moves from c_i to c_j and $q_{ii} = -\sum_{j:j \neq i} q_{ij}$ for all i.

It is worth noting that we specify the values of q_{ii} in such a way that the sum of each row in the matrix becomes zero. Using this formulation, the mobility matrix $\mathbf{Q}$ can be employed as a transition rate matrix which governs a continuous time Markov process representing the movement of a mobile computer. Let this process be $\mathcal{X}_m = \{X_m(t) : t \geq 0\}$ whose state space consists of all possible cell locations $c_1, c_2, \ldots, c_N$ of the cellular network. The subscript m means that the last location update took place at cell c_m, *i.e.* $X_m(0) = c_m$; and t is the time elapsed from the last location update.

2.2 Location Prediction

At the time of every location update, the wired network will know the exact location of the mobile computer. As a result, the Markov process should be restarted to reflect the addition of location information. However, in the period between two successive location updates, the mobile computer can move across cells inside the network when they are disconnected from the network. Therefore, the wired network will not know the exact location of the mobile computer during the period. In order to minimize the paging cost, the wired system has to predict the probable cell locations that the mobile computer can be situated in.

There are many research works on minimizing the cost of paging by employing the location probabilities [11, 2, 14]—a vector of probabilities in which element i represents the probability that the mobile computer is situated in cell c_i. In their analyses, however, they do not address any method to obtain the location probability. Here, we devise a method to evaluate this probability vector under the model we have described.

Instead of deriving the steady state location probabilities, we derive the *transient* location probabilities, *i.e.* the probabilities vary with the time since the last location update. It is more accurate if the time at which an incoming call to the mobile computer is short. For an extreme case, if a call arrives just after the mobile computer sends the location update, it is likely that the mobile computer is still inside the last reporting cell. By using the transient method to obtain the location probabilities, we can account for these cases. Lastly, note that if the time for an incoming call is long, the probabilities will approach the steady state ones.

Let the transient location probability vector of the Markov process $\mathcal{X}_m$ be $\boldsymbol{\pi}_m(t) = \langle \pi_{1|m}(t), \pi_{2|m}(t), \ldots,$

$\pi_{N|m}(t) >$, where $\pi_{i|m}(t)$ is the probability that the computer is located at cell c_i at time t since the last report of location at cell c_m. Note that the initial probability vector $\boldsymbol{\pi}_m(0)$ is a unit vector where

$$\pi_{i|m}(0) = \begin{cases} 0 & \forall i \neq m \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

Here, we use the uniformization method [6, 7] to find the location probabilities of the mobile computer at a particular time. This method is well-known for its efficiency and numerical stability. In our case, we now have a continuous time Markov process $\mathcal{X}_m$ which is governed by the initial location probability vector $\boldsymbol{\pi}_m(0)$ and transition rate matrix $\mathbf{Q}$. By uniformization, $\mathcal{X}_m$ can be transformed into a discrete time Markov chain with the same state space $\mathcal{C}$. The transition events of this discretized chain are governed by a Poisson process with rate $\Lambda = \max\{-q_{ii} : 1 \leq i \leq N\}$. The step transition matrix for this Markov chain can be computed as $\mathbf{P} = \mathbf{I} + \mathbf{Q}/\Lambda$. By conditioning on the number of transitions over the time interval $(0, t)$, we can establish the following equation for the transient location probability vector:

$$\boldsymbol{\pi}_m(t) = \sum_{n=0}^{\infty} e^{-\Lambda t} \frac{(\Lambda t)^n}{n!} \mathbf{p}_m(n) \qquad \text{where} \tag{2}$$

$$\mathbf{p}_m(n) = \begin{cases} \mathbf{p}_m(n-1)\mathbf{P} & \text{for } n = 1, 2, \ldots \\ \boldsymbol{\pi}_m(0) & \text{for } n = 0 \end{cases} \tag{3}$$

is the probability vector for the discrete Markov chain after n transitions. The i-th element $p_{i|m}(n)$ in the vector $\mathbf{p}_m(n)$ is the probability that the mobile computer is located at cell c_i after n transitions since its last report of location at cell c_m.

Lastly, we define $\boldsymbol{\pi} = <\pi_1, \pi_2, \ldots, \pi_N>$ to be the steady location probability vector of the mobile computer in the network, where π_m represents the steady probability that the mobile computer is at cell c_m. This probability vector, which is the solution to the equation $\boldsymbol{\pi} = \boldsymbol{\pi}P$, can be found by standard iterative methods [12].

2.3 Directional Movement and Non-Exponential Cell Dwell Time

In this section, we try to include in our model the directional movement and non-exponential cell dwell time property. This is done by splitting up the "cell states" of the continuous-time Markov chain of the movement model.

In order to include the information of directional movement into the Markov model, we split a cell state c_i into several component states: $<c_i, d_1>$, $<c_i, d_2>$, ..., where the second component d_j in the component state $<c_i, d_j>$ represents the moving direction of the mobile user. The probabilities of the states derived will no longer correspond to the location probabilities. Instead, each state probability $\pi_{i,d_j|m}(t)$ corresponds to the (location, direction) pair. To obtain the location probability for a particular cell c_i, we need to take the summation of the location for all possible directions of movement, i.e. $\pi_{i|m}(t) = \sum_{j=1}^{N_d} \pi_{i,d_j|m}(t)$. The extension to accommodate a more general distribution of cell dwell times can be achieved using a similar technique.

3 Performance Analysis

As we have argued in the previous section and in [8], it is more accurate to derive the transient location probability vector of mobile computer. We first present the analysis for the time-based protocol, then the distance-based protocol and lastly, the reverse-guess protocol.

3.1 Time-Based Protocol

In the time-based protocol, a time limit T is specified *a priori* for limiting the period length. If the period lasts for this time limit without any paging event, the mobile computer will send an update message to the wired network. On the other hand, if there is an incoming call arriving at the wired network before the time limit, the network subsystem will page for the mobile computer. When the mobile computer receives this paging signal, it will reply the subsystem, and a communication session will be started. This period ends with a paging cost.

Note that if the mobile computer moves at a higher rate, the expected distance traveled by the mobile computer between two subsequent location updates will be larger. Consequently, we may need to readjust the value of the time limit to a smaller value. In general, a mobile computer may have a non-uniform movement pattern. Consequently, we need to have a separate optimal value for the mobile computer at each of the cell in the network. More importantly, as the movement pattern of the mobile computer changes over time, the system has to re-compute all the optimal values. This increases the maintenance cost vastly.

In our derivation of the cost rate, we consider a single time period that begins with a location update and ends with either another location update or paging. We will first derive the expected tracking cost of the period and then the expected length duration. The expected cost rate is then the ratio of the expected cost and the period length.

If the time period terminates with a location update, i.e. the mobile computer sends an update message to the base station, then an update cost of U is incurred. This period will also end at time T, which is the time limit for the time-based protocol. In another case, if there is a call arrives at time t ($0 < t \leq T$) in the period, the static network will page for the mobile computer. If the mobile computer remains inside the location area, the cost of polling is KC, where C is the overhead for polling a single cell, $K = 3(d + 1)d + 1$ where cells are less than or equal to d apart from the reporting cell. However, if the computer is outside the location area, the *overall* cost of polling for the computer is NC. Let $\mathbf{r}_m = <r_{i|m} : i = 1, 2, \ldots, N>$ be a column vector such that $r_{i|m} = 1$ if cell c_i is at a distance less than or equal to d from c_m, and $r_{i|m} = 0$ otherwise. Using the location probabilities derived previously, the probability of the mobile computer staying inside the area is $\boldsymbol{\pi}_m(t)\mathbf{r}_m$ and that of outside the area

is $1 - \boldsymbol{\pi}_m(t)\mathbf{r}_m$. Therefore the polling cost when an incoming call arrives at time t is

$$KC[\boldsymbol{\pi}_m(t)\mathbf{r}_m] + NC[1 - \boldsymbol{\pi}_m(t)\mathbf{r}_m] = \\ NC - (N{-}K)C\boldsymbol{\pi}_m(t)\mathbf{r}_m \quad (4)$$

Assume that the arrivals of the incoming calls form a Poisson process with parameter μ. The probability distribution function of the time for the call arrival is $P(t) = 1 - e^{-\mu t}$. Subsequently, the probability that the time period terminates with an incoming call is given by $1 - e^{-\mu T}$, while the probability that the period terminates with the delivery of an update message by the mobile computer is $e^{-\mu T}$, where T is the time limit for the time-based protocol. Unconditioning update and paging cases, we have the expected cost of tracking the mobile computer for one single period:

$$C_m^T(T) \quad = \quad Ue^{-\mu T} + \\ NC - (N{-}K)C\boldsymbol{\pi}_m(t)\mathbf{r}_m\} \, \mu e^{-\mu t} \, dt \, (5)$$

Substitute Equation (3) into the above formula and then rearrange the expression (refer to [8] for exact derivation), we now have

$$C_m^T(T) \quad = \quad Ue^{-\mu T} + NC(1 - e^{-\mu T}) - (N{-}K) \\ C\mu T \sum_{n=0}^{\infty} e^{-(\Lambda+\mu)T} \frac{[(\Lambda + \mu)T]^n}{n!} f(n) \, (6)$$

where $f(n)$ can be evaluated recursively as

$$f(n+1) = \frac{n+1}{n+2}f(n) + \left(\frac{\Lambda}{\Lambda+\mu}\right)^{n+1} \frac{\mathbf{p}_m(n+1)\mathbf{r}_m}{n+2} \quad (7)$$
$$f(0) = \mathbf{p}_m(0)\mathbf{r}_m \quad (8)$$

For the duration of the time period, if the period ends with a location update, the period should be T. Otherwise, if the period ends with an arriving call, the length t of the period can be defined by the probability density function $p(t) = \mu e^{-\mu t}$. Therefore, the distribution of the duration of the time period is exponential truncated at T with rate μ; and the expected duration of the period is given by

$$D_m^T(T) = \int_0^T t\mu e^{-\mu t}dt + Te^{-\mu T} = \frac{1 - e^{-\mu T}}{\mu} \quad (9)$$

Note that both the expected cost and the expected duration of each time period are conditioned on the last reporting cell location, c_m, of the mobile computer. By removing this condition, we have the *expected cost rate* for this time based protocol with parameter T as the following:

$$E_T(T) = \frac{\sum_{m=1}^{N} C_m^T(T)\pi_m}{\sum_{m=1}^{N} D_m^T(T)\pi_m} \quad (10)$$

In this protocol, we want to find an optimal parameter T^* such that the average expected cost $E_T(T^*)$ is minimized. However, in our mobility model, we allow a non-uniform computer movement pattern over the cellular network. Instead of finding a single optimal value of the time limit for this protocol, it would be theoretically better to find an optimal vector of time limits $\mathbf{T}^* = <T_i^* : 1 \le i \le N>$ such that each element T_i^* represents the optimal value when the mobile computer is in cell c_i, i.e. $C_i^T(T_i^*)/D_i^T(T_i^*)$ is minimum for all $1 \le i \le N$. These values can be found for each cell separately because they are independent of each other.

3.2 Distance-Based Protocol

In this protocol, the mobile computer will always listen to the broadcast channel from the base station to locate itself inside the cellular network, even though when there may not be a communication channel between them. A distance limit is defined in this protocol. When the mobile computer detects that it has gone farther away from the last reporting location than the distance limit, it will send a location update message. When an incoming call arrives, the wired system will page over the cellular network. On the other hand, we can see that in this protocol, the mobile computer will definitely be inside the registration area since if it were not, an update notification will be sent. Therefore, it is not necessary for the wired system to page outside the location area. In this regard, the paging cost can be lowered. Note that if the distance limit is set to zero, this protocol will become the always-update protocol.

We model the time for the location update as the *first passage time* for the Markov process to change some states (cells within the distance bound) to some other states (cell out of the distance bound). The distribution of the time is a *phase distribution* [8], of which we denote the probability density function as $f_m(t)$. By deriving the expression for $f_m(t)$, we can get the expression for the expected cost for a location update period, $C_m^D(d)$, given that the last location update is at cell c_m. The expression is as shown in the following:

$$C_m^D(d) = \int_0^{\infty} Uf_m(t)e^{-\mu t}dt + \int_0^{\infty} KC\mu e^{-\mu t}[1{-}F_m(t)]dt \quad (11)$$

where $F_m(t)$ is the probability distribution function for $f_m(t)$. Using similar argument, we can also derive the expected length, $D_m^D(d)$, of the period given that that the last reporting cell is c_m. The expression is

$$D_m^D(d) = \int_0^{\infty} tf_m(t)e^{-\mu t}dt + \int_0^{\infty} t\mu e^{-\mu t}[1{-}F_m(t)]dt \quad (12)$$

The remaining question is how to obtain the phase distribution function $f_m(t)$. Due to the lack of space, we refer the reader to [8] for detail derivation.

In this protocol, we are able to find the optimal value d^* such that the average expected cost $E_D(d^*)$ is minimized. Again, because our movement model allows a non-uniform movement pattern of a mobile computer inside the network, it is advantageous to evaluate an optimal vector $\mathbf{d}^* = <d_i^* : 1 \le i \le N>$ where each element d_i^* represents the optimal distance value for the cell c_i. Since the values for each element in the vector is independent, we can derive the optimal value for each cell separately.

3.3 The Reverse-Guessing Protocol

Here, we propose a new algorithm, the *reverse-guessing* algorithm, that accounts for the paging cost and the update cost in a more direct manner. In this protocol, we assume that the mobile computer will also have the statistics about its movement patterns. It will also use the *same* prediction method as the wired network system so that it can know what the wired system predict on its current location. The protocol operates as the following. At certain time intervals, the computer evaluates its location probability in the same way as the wired network (as presented in Section 2). Then we rearrange cells such that the location probabilities are in decreasing order of magnitudes. The mobile computer then looks up the position of its current cell in the probability sequence. We define the position as the *rank*. If the rank is greater than a predefined limit for the protocol, then an update message will be sent. For detail description of the protocol, please refer to [8].

The main reason to choose the rank as the heuristic is very intuitive. In the analysis of paging algorithms, like [11], we can see that the cost of paging is directly related to the rank of the mobile computer's current cell in the location probability sequence. In other words, if the value of the rank is larger, the actual paging cost will be higher. Therefore, we can use the rank as a more direct estimate of the paging cost. Another important point is that there are many advantages for this protocol. First of all, it is an online protocol. The mobile computer will keep track of its current location and use this information to obtain a more precise estimate of the paging cost. Moreover, the value of the heuristic (rank) for our reverse-guessing protocol is insensitive to different mobility patterns. This results in a more adaptive optimization as compared to the time-based and the distance-based protocols. Lastly, the protocol involves only the calculation of the location probability, which is efficient enough to be implemented inside a mobile computer. Moreover, we can lower the complexity of the prediction at the mobile computer by the reducing the problem size.

Let us carry out the performance analysis of this protocol. Mathematically, we have the cost evaluation functions for the expected cost and duration of the reverse-guessing protocol, as the ones for the distance-based protocol (Equations (11) and (12)):

$$C_m^R(R) = \int_0^\infty U g_m(t) e^{-\mu t} dt + \int_0^\infty \{NC - (N-K)C\boldsymbol{\pi}_m(t)\mathbf{r}_m\}$$
$$\mu e^{-\mu t}[1 - G_m(t)]dt \qquad (13)$$
$$D_m^R(R) = \int_0^\infty t g_m(t) e^{-\mu t} dt + \int_0^\infty t\mu e^{-\mu t}[1 - G_m(t)]dt. \qquad (14)$$

where $g_m(t)$ is the probability density function for the time until a location update is sent by the mobile computer and $G_m(t)$ is the probability distribution function. In each equation, the first term corresponds to the case that a location update message is sent by the mobile computer; while the second term represents the case that an incoming call arrives before the delivery of the location update message. The term $\{NC - (N-K)C\boldsymbol{\pi}_m(t)\mathbf{r}_m\}$ is the expected paging cost at time t and is the same as that in the time-based protocol (Equation (4)).

Let us also define that the threshold value of the reverse-guessing protocol be R. The distribution function is then defined as

$$G_m(t) = \sum_{i=R+1}^{N} \pi_{i|m}^P(t) \qquad (15)$$

In [8], we derive the computation procedure of computing tight upper and lower bounds of the expected values. The upper bound of the cost rate for the reverse-guessing protocol can be obtained by

$$\max[E_R(R)] = \frac{\sum_{m=1}^{N} \max[C_m^R(R)]\pi_m}{\sum_{m=1}^{N} \min[D_m^R(R)]\pi_m} \qquad (16)$$

and the lower bound is

$$\min[E_R(R)] = \frac{\sum_{m=1}^{N} \min[C_m^R(R)]\pi_m}{\sum_{m=1}^{N} \max[D_m^R(R)]\pi_m} \qquad (17)$$

One advantage of this protocol over the time-based and the distance-based protocols is that the optimized value of G does not change for each starting cell c_m. This can be seen in the experiment in the next section. It is because, in this protocol, we try to relate the location update event at the mobile computer to the accuracy of the prediction of the wired system. This accuracy has little dependency on the movement patterns of the mobile computer. As a result, the protocol is more adaptive to changes in the movement pattern of the mobile computer.

4 Experiment

In this section, we carry out the experiment and compare the performance results of the time-based protocol, the distance-based protocol and the proposed reverse guessing protocol. This experiment is divided into four cases, each of them differs from each other by the mobility pattern. In the first case—the uniform mobility—the mobility of the mobile computer is uniform and symmetric across the whole cellular network. This case occurs when the mobile user is inside a prosperous area of a city, where probably random walk is the most realistic model. In the second case—the random mobility—the move rate of the computer from one cell to another cell is generated randomly from uniform distribution. In the third, or the zone case, the computer has a tendency to stay inside a geographical area. In the last case, the circle case, the computer moves along a circular path. Many common user movement patterns can be generated by superimposing the different movement patterns mentioned above.

In all these cases, we carried out five protocols. The first one corresponds to the case that the mobile computer will not send any update message. The second one corresponds to the case that the computer sends an update whenever it enters a different cell. The

Protocol	Uniform	Random	Zone	Circle
No Update	24.6926	15.2988	7.8195	3.09715
Always Update	100.679	100.683	100.677	100.681
Time-Based	18.9124(9)	12.1886(13)	7.6240(18)	3.09715(∞)
Distance-Based	23.1667(8)	15.9603(8)	7.6983(11)	3.03874(11)
Reverse-Guessing	13.4576(30)	9.13518(30)	5.79265(30)	3.02828(30)

Table 1: Expected Cost for various protocols under different mobility patterns

last three protocols are time-based, distance-based and reverse-guessing protocols respectively.

The configuration is a network composed of 721 cells. The only difference is the assignment of the move rates from one cell to another cell, which are now geographically dependent according to the different cases mentioned above. For a fair comparison, however, we have normalized the move rates in such a way that the maximum move rate is set to be once per unit time. The performance results for the experiment is tabulated above. In this table the minimum costs for different protocols under different movement patterns are recorded. Also, the optimal parameter values (*i.e.* time period limit for the time-based protocol, the distance limit for the distance-based protocol, and the threshold for the rank in the reverse-guessing protocol) are recorded inside the brackets. ¿From this result, we can see that the performance results of different protocol vary under different movement patterns. Moreover, the optimal parameter settings for the time-based and the distance-based protocol also change with different movement patterns. The most important of all, however, is that the performance of the reverse-guessing protocol is always the best among all five protocols.

5 Conclusion

Mobile terminal tracking is an essential mechanism for the personal communication network to locate a disconnected mobile computer so as to re-establish a connection to the computer. There are many protocols proposed with the target at balancing the update cost with the paging cost, such as the time-based protocol and the distance-based protocol. To test the performance of these protocols, researchers have devised some mathematical models to evaluate the system costs. However, their models may not be capable of capturing real life characteristics of user movement that are important in the performance evaluation. To cope with these deficiencies, we have proposed a novel, more general mathematical framework to include those important properties. In addition to these analyses, we have also taken advantage of the prediction result to devise the reverse-guessing protocol. This protocol can obtain an optimal paging process. In general, the reverse-guessing protocol uses a better heuristic to help the mobile computer determine when to send an update message. This protocol can result in a lower system cost for mobile terminal tracking.

References

[1] Ian F. Akyildiz and Joseph S. M. Ho. Dynamic Mobile User Location Update for Wireless PCS Networks. In *ACM Journal of Wireless Networks*, volume 1. Baltzer Publishers, 1995.

[2] Daniel O. Awduche, Aura Ganz, and Arthur Gaylord. An Optimal Search Strategy for Mobile Stations. In *Proceedings of IEEE International Conference on Universal Personal Communications*, 1996.

[3] Amotz Bar-Noy and Ilan Kessler. Tracking Mobile Users in Wireless Communications Networks. In *Proceedings of IEEE INFOCOM*, 1993.

[4] Amotz Bar-Noy, Ilan Kessler, and Moshe Sidi. Mobile users: To Update or not to Update? In *Proceedings of IEEE INFOCOM*, 1994.

[5] S. K. Das and S. K. Sen. A New Location Update Strategy for Cellular Networks and Its Implementation Using a Genetic Algorithm. In *Proceedings of the Third ACM/IEEE Conference on Mobile Computing and Networking*, September 1997.

[6] Edmundo de Souza e Silva and H. Richard Gail. Analyzing Scheduled Maintainence Policies for Repairable Computer Systems. In *IEEE Transactions on Computers*, November 1990.

[7] A. Jenson. Markov Chains as an Aid in the Study of Markov Processes. In *Skand. Aktuarietidskr*, 1953.

[8] H.W. Chan John C.S. Lui, C.C.F. Fong. Location Updates and Probabilistic Tracking Algorithms for Mobile Cellular Networks. Technical report, Department of Computer Science & Engineering, the University of Hong Kong, 1999.

[9] Upamanyu Madhow, Michael L. Honig, and Ken Steiglitz. Optimization of Wireless Resources for Personal Communications Mobility Tracking. In *Proceedings of IEEE Infocom*, 1994.

[10] C. Rose. Minimizing the Average Cost of Paging and Registration: A Timer-Based Method. In *ACM Journal of Wireless Networks*, volume 2. Baltzer Publishers, 1996.

[11] C. Rose and R. Yates. Minimizing the Average Cost of Paging under Delay Constraints. In *ACM Journal of Wireless Networks*, volume 1. Baltzer Publishers, 1995.

[12] William J. Stewart. *Introduction to Numerical Solution of Markov Chains*. Princeton University Press, 1994.

[13] Leandros Tassiulas and Farooq M. Anjum. A Hierarchical Multiresolution Registration Structure for Mobility Track. In *Proceedings of IEEE International Conference on Universal Personal Communications*, 1996.

[14] Markku Verkama. Optimal Paging — A Search Theory Approach. In *Proceedings of IEEE International Conference on Universal Personal Communications*, 1996.