J Netw Syst Manage (2014) 22:23–49
DOI 10.1007/s10922-012-9258-9

Location Prediction Based on a Sector Snapshot for Location-Based Services

Mohammad Sharif Daoud · Aladdin Ayesh · Mustafa Al-Fayoumi · Adrian A. Hopgood

Received: 14 January 2012 / Revised: 1 December 2012 / Accepted: 3 December 2012 / Published online: 4 January 2013

Abstract In location-based services (LBSs), the service is provided based on the users' locations through location determination and mobility realization. Most of the current location prediction research is focused on generalized location models, where the geographic extent is divided into regular-shaped cells. These models are not suitable for certain LBSs where the objectives are to compute and present on-road services. Such techniques are the new Markov-based mobility prediction (NMMP) and prediction location model (PLM) that deal with inner cell structure and different levels of prediction, respectively. The NMMP and PLM techniques suffer from complex computation, accuracy rate regression, and insufficient accuracy. In this paper, a novel cell splitting algorithm is proposed. Also, a new prediction technique is introduced. The cell splitting is universal so it can be applied to all types of cells. Meanwhile, this algorithm is implemented to the Micro cell in parallel with the new prediction technique. The prediction technique, compared with two classic prediction techniques and the experimental results, show the effectiveness and robustness of the new splitting algorithm and prediction technique.

M. Sh. Daoud (✉) · A. Ayesh
Faculty of Technology, De Montfort University, The Gateway, Leicester LE1 9BH, UK
e-mail: m2sharief@yahoo.co.uk

A. Ayesh
e-mail: Aayesh@dmu.ac.uk

M. Al-Fayoumi
College of Computer Engineering and Sciences, Salman Bin Abdulaziz University, Al-Kharj, Saudi Arabia
e-mail: Fayoumi66@yahoo.com

A. A. Hopgood
Sheffield Business School, Sheffield Hallam University, Howard Street, Sheffield S1 1WB, UK
e-mail: A.hopgood@shu.ac.uk

Keywords Mobility · Mobile displacement · Cell-based · Map-based · Markov chain model · GPS · UMTS

1 Introduction

With the advancement of wireless communication and computer technologies, mobile communication has been providing more versatile, portable and affordable networks [1–3] than ever. The number of mobile users on mobile networks has increased rapidly, as the number of mobile users around the world is near 3 billion [4]. The third and subsequent generations of communication do not only bring new technical problems but they also raise a new class of interesting applications. This is due to the change in communication from single-medium oriented into multimedia oriented communication such as image, computing data, Internet services, e-commerce, and video conferences [5, 6].

The rapid technological advances in wireless networks and cellular communication have led to the emergence of the mobile computing paradigm, where information is accessible anywhere and at any time. This new paradigm enables almost unrestricted mobility to the users, which poses a new set of constraints and new kinds of challenges that need to be considered in the design of network protocols and information services.

The two key issues that affect network protocols are mobility and wireless link characteristics. Since mobility became the norm rather than the exception, a user's location information is an additional parameter that needs to be taken into consideration in protocol design. A cost-effective technique should be deployed to locate a certain user as well as efficient data structures and algorithms to manage this fast-changing data.

This proliferation in mobile devices and users' demands gives rise to location-based services (LBSs). They deliver dependent and suitable information relevant to a client's location, hence, narrowing redundancy in the information provided [7]. A key feature of LBSs is that any service requested may need to be answered with different results, depending on the location of the mobile user. Location prediction provides time to prepare services that may be needed by the user in anticipation of requesting them. Especially, services involved with complex computation that may need to extract data and to save time, to ensure that only desired services are available when requested. The mobile communication environment is considered as a restricted dynamic environment [8, 9]. The restriction in such an environment is due to the limitations of the mobile user in terms of processing power, memory, storage, capacity, screen resolution, and battery performance.

This paper discusses a new prediction technique to improve the prediction of locations in LBSs for a micro cell: based-on the introduced splitting model. The proposed prediction technique divides the micro cell into eight equivalent regions (sectors). In this scheme, an update message is sent to the network with the current user location whenever a change in the moving direction of the user is detected. The main contribution of this paper targets the LBS's cost by deploying a Markov chain model that allows intelligent LBSs to minimize the computation cost, consumption of resources and the overall cost of the location management process. The proposed scheme utilizes geometrical and topological techniques allowing users to receive desired services in a timely fashion. In a sense, a developed prediction technique can be utilized to determine a specific request service such as the nearest restaurant, Asynchronous Transfer Mode (ATM), hospital, or a park, in which the proposed works to deliver these services In a timely manner while avoiding the manual filter that may have occurred on the mobile device. Meanwhile, the splitting model is discussed, where the mechanism is used to split different cell types and how the location of the mobile user will be calculated by the model.

The rest of the paper is organized as follows. Section 2 discusses the related work on location prediction for LBSs and their limitations. In Sect. 3, an analytical review of the Markov chain model is presented. A novel Cell Splitting Algorithm is discussed in Sect. 4. The Framework and proposed scheme is introduced in Sect. 5. The simulation model and result analysis is presented in Sect. 6. Section 7 discusses the significant analytical analysis. Finally, the conclusion and future work is presented in Sect. 8.

2 Related Work

Locating users as they move from one place to another in a mobile computing environment is the key to providing continuous services with unrestricted mobility. Therefore, the data management in this environment is especially challenging for the need to process information on the move, to cope with resource limitations, and to deal with heterogeneity. One of the applications of mobile data management is LBSs, which have been identified as one of the most promising areas of research and development [10].

Strategies of location management in mobile environments can be classified into static and dynamic. In the static strategy, the update operation is reduced according to the network topology. This technique suffers some inefficiency especially for users that are located around the Routing Area (RA) boundaries and who cross these boundaries repeatedly. Moreover, RA sizes are fixed for all users as specified by the cellular infrastructure, without considering their individual mobility and service request pattern.

Dynamic location updates have been developed to address and enhance the efficiency of the static strategy [11]. The update operation is initiated according to the user's movement pattern and the frequency of its requesting service. Location is among the most important contextual information for mobile applications. Much of the previous work on LBSs treated location as an additional attribute of the data tables [12, 13]. In this way, location based service queries can be processed like ordinary queries except with additional constraints on the location attribute. Predictive location dynamically was introduced to predict a mobile user's future location based on the current location information, the user's historical mobility pattern, and the auxiliary information. Therefore, the mobility realization and location determination are two factors in location prediction to determine the location of a mobile user at a time *t*.

Francois and Leduc [14] have introduced the accuracy of prediction to evaluate models. Numerous prediction models have been introduced to increase the accuracy of the prediction techniques for users with varying speeds that have been reported in the literature, but none of them can fulfill the optimal accuracy prediction rate and effective cost requirements. The literature is divided into three sections, namely, the cell-based technique, the map-based technique and Prediction Techniques that are based on the Markov chain model.

2.1 Cell-Based Technique

In the cell technique [15–20] a service area is partitioned into several cells; the cell covering the mobile user will page his or her device to establish a radio link in order to track changes in the location of the mobile users.

The cells broadcast their identities and the mobile user periodically listens to the broadcast cell identity and compares it with the cell identity stored in its buffer. If the comparison indicates that the location has been changed then the mobile user sends a location update message to the network [21].

Prediction techniques, based on a cell technique, can be enhanced by heuristic methods and neural networks [22, 23]. Liou and Lu [22] divided the cell into two areas, edge and non-edge. The edge areas have neighboring cells, while the remaining areas are considered as non-edge areas. When the mobile user is in a cell's edge area, the information is passed to a neural network that predicts from the neighbor's cells, the next cell to be visited. Another technique captures some of the mobile user activity and paths. These paths are progressively recorded, giving a history record that is used as input to a neural network to predict the next cell to be visited [23].

The techniques proposed in [22–24] suffered from a long training phase on mobile movements data that where used to build a knowledge base before making predictions. Therefore, the mobile user may change his or her activity, such as movement pattern or visiting a location he/she never visited before, thereby bringing new cases that the techniques have not encountered in training. Hence, the prediction percentages dramatically decrease.

A new splitting-based displacement prediction approach for location-based services (SDPA) [25] has been developed to improve the prediction rate. On the other hand, SDPA reduced the service area in a static manner that is not applicable for a cellular communication network.

2.2 Map-Based Technique

In map matching, using the fussy logic and the data gathered from the Global Positioning System (GPS) is considered as one of the techniques that are proposed to navigate the wheelchair user in a sidewalk area [26]. GPS data and the map of the target area are stored in the server side and the analysis of the data is performed by the fussy logic. Therefore, the incorrect direction will be eliminated, after that, and it will advise the wheelchair to reach a destination [26].

The map matching techniques area proposed in [26–28] suffered from many major drawbacks. All of them are tested and evaluated only for wheelchairs on university campus sidewalk and they work solely outdoors. These techniques are based on GPS navigators. Therefore, anyone who needs to use them must have GPS sensors. However, the GPS sensors lead to extra physical costs, bearing in mind that they may not be applicable for all mobile devices. Moreover, GPS suffers from inaccurate data in narrow roads and high buildings, and is believed to use higher-end GPS relievers to improve the signal, instead of low-end.

In urban areas, the navigation of wheelchairs is difficult because the satellite signal could be very poor in that area. This causes a conflict and an odd drawback for those techniques from the viewpoint of how wheelchair users can be navigated in urban areas while the GPS is not accurate in that area and the wheelchair can't move on highways.

Predictive location model (PLM) is a technique obtained from the Map-Based model without some of its limitations, such as the need to know the end of travel before starting. A service area is modeled as a graph; the edge indicates a road segment, and the intersection of edges is represented as a vertex [29]. During a user's trip on a road, the network generates a trajectory. The trajectory defines a sequence of connected road segments or a sequence of connected vertices between two locations, namely start point and end point. The user trajectory is stored in a database to assist in the prediction of its future trajectory when beginning a new journey. The historical trajectory information stored can be used to infer the number of times the user has traveled on each road segment and the trajectory choice at each intersection. The data are then used to predict the travel of the user.

PLM depends on creating a Dynamic Computational Window (DCW). A DCW is defined as a circular clipping window that centers around the user's current location to retrieve information from a database for location prediction [29]. The size of the DCW dynamically changes relative to the speed of the user. PLM does not allow a given user to visit each of the trajectories more than once for the whole trip. That means the user cannot turn around at an intersection. Extra calculation is needed because the end of travel has not been previously determined.

2.3 Prediction Techniques Based on Markov Chain

There are many different techniques used to enhance the mobility prediction. Markov chain is one of the most commonly used in predictions [30–34].

In [34], the Markov model is used to anticipate the next displacement that is based on the mobility history. The area that is predicted is too large because it contains many cells (Location Area LR). The new mobile entrance to the network may decrease the prediction percentage that is already made by the model.

The models that are introduced in [31, 32] enhance the mobility prediction by using the second order Markov chain. The enhancements have been applied on the computation process or prediction percentage.

The model that is proposed in [32] introduces an efficient mobility prediction by using both incoming and outgoing handoff predictions. Other parameters are used such as road topology, handoff area points inside the cell, cell shape structure and the average time lasting in each service area segment. The model offers acceptable results, however, the implementation cost and response time are significant and the service area that would be predicted is too large, therefore, the manual filter may appear.

In Sun and Blough [31], the user's knowledge is the important key for the prediction process; the user's future knowledge is collected from a mobile side such as a user's diary, e-mail, or instant messaging. The model provides a good prediction percentage when it collects the knowledge, but when no knowledge is available the prediction dramatically decreases. The obstacle is how the knowledge can be collected; thus, this model poses a conflict for mobile user privacy.

The mobility history is considered as the main parameter in the hidden Markov chain for the models that are proposed in [30, 33]. These models will be applicable when the base stations are not managed by the network entity whereas all previous movements of the mobile user are saved and manipulated. Nonetheless, the main drawback is the computation cost that the models need.

Bellahsene and Kloul [35] and Bellahsene et al. [36] have introduced a new mobility prediction architecture that is based on two nodes of the network, global prediction and local prediction. The global prediction works on the enhance gateway while the local prediction works on the base station level. The drawbacks that are addressed in [36] are the delay of the response and the model that does not have a good prediction percentage due to the movements of the mobile user that are described by a random way or the model that does not have enough mobility information about the mobile user.

In [35], the NMMP is introduced where the model appeares as an enhancement to the randomness movement more than those in [36]. The NMMP model is based on two prediction levels, namely, the Global Prediction Algorithm (GPA) and the Local Prediction Algorithm (LPA). The GPA is run by an Enhanced Gateway (EGW) that considers the root of the cellular network. The GPA is responsible for handling the regular user's movements. The LPA is run by an Enhanced Base Stations (EBSs) in order to predict the mobile user's random movements within a cell.

The NMMP handles different types of mobile user mobility movements and it has many drawbacks. The first drawback is the communication cost that is higher in order to solve the PING PONG handover problem although the problem has been already solved in [37, 38]. The communication cost consists of the EGW cost, EBSs' cost, Cell cost, and the cost of routers that are used to connect all of them. Moreover, the communication cost affects the delay of response and updating history. A group of cells that are elected to be in the next displacement belongs to a different router of the EGW that leads to the communication cost being doubled for all processes that are needed.

Secondly, the time management of the NMMP is weak. GPA and LPA are working on different time slots, which means that there is no overlap between the times needed to achieve both of them. In a sense, the total time for the NMMP is the GPA and LPA times. In other words, the total time is the time performed by both the GPA and LPA.

Many techniques have been developed to speed up these schemes, but these improvements consume a significant portion of the overall system resources. Therefore, this paper will endeavor to design a new model that provides a balance between an accuracy rate and power efficiency, suitable for the mobile environment.

3 Markov Chain Model for Prediction

Markov chain models are used for analyzing complex systems and predicting behavior under uncertain dynamic conditions. Furthermore, they can yield present and future states independently of the past states [39, 40].

In real systems, the state is changed from the current state to the next state, or remains in the same state. Therefore, the prediction of Markov chain models is based on a certain probability distribution [41, 42]. The changes from the current state to the next state are called transitions. Each change has a probability that is called the transition probability. Moreover, there are other examples for Markov chain models that are a simple random walk and weather prediction [43].

The probabilities are essential in real systems that are given the probabilities of the preceding states that can be expressed by a transition matrix [44, 43].

$$P = \begin{bmatrix} x & \bar{x} \\ y & \bar{y} \end{bmatrix}$$

$(P)_{i,j}$ is the probability that, if a given state is of type i, it will be followed by a state of type j. When a state of the system is known to be S at time 0, the prediction path can be represented by a vector where the probability of S is 100 % and the complement is 0 %.

$$S(0) = [1 \quad 0]$$

The next state or path of S can be predicted by

$$S(1) = S(0) * P = [1 \quad 0] \begin{bmatrix} x & \bar{x} \\ y & \bar{y} \end{bmatrix} = [z \quad \bar{z}] \tag{1}$$

Here z indicates the probability of the next state, which the user may cross. The general rule to predict N paths that will be crossed is

$$S(N) = S(N-1) * P \tag{2}$$

$$S(N) = S(0) * P^N \tag{3}$$

4 A Novel Cell Splitting Algorithm

This section introduces and describes a novel efficient algorithm for splitting cellular cells and locating the mobile user. Also, the algorithm that is built to handle a certain cell type will be used different cell types (i.e., Pico, micro, macro, and rural) that are based on the symmetry features. In a cellular communication system, the cell is considered as a circle graph.

In this research, the symmetric characteristics of the circle are exploited; one portion of the graph is known, and the remaining portion of the graph can be predicted. The proposed algorithm is utilized in the symmetry by splitting the cell in four quadrants $q = Q_i \,|_{i\in[1,4]}$. The quadrant division depends on the angle path. The angle path (P) is calculated by $P = \sum_{\theta=0}^{360} Q_i$. where Q_i is determined as shown in Eq. (4):

$$Q_i = \begin{cases} i = 1, \ \text{when } \theta \, in \, [0, 90]; \\ i = 2, \ \text{when } \theta \, in \, (180, 90); \\ i = 3, \ \text{when } \theta \, in \, [180, 270]; \\ i = 4, \ \text{when } \theta \, in \, (360, 270); \end{cases} \tag{4}$$

To split a cell area into n equivalent sectors, apply the following rule. Let (x, y) be a point in the xy-plane that is selected randomly from a circular region with radius r and centered at the origin, if the circular region is divided to 2^n sector, where n = 3, 4, 5, …, named $S_1, S_2, S_3, \ldots, S_{2^n}$. Also assume that the centered angle of each sector is

$$\alpha = \frac{360^\circ}{2^n} \tag{5}$$

let λ be the number of sectors in each quadrant. Each quadrant is divided into uniform sectors as follows:

$$\lambda = \frac{2^n}{4} = 2^{n-2} \tag{6}$$

Then, this rule is arranged so the sectors in the plane are at a way that ensures the easy relation in determining the location of the mobile user that has a point (x, y).

Figure 1 shows the cell splitting algorithm where the number of sectors in each quadrant is equivalent to each other's quadrant in the same cell. The numbering sectors is drawn in the way to assist yielding a general algorithm for an unspecific number of sectors regardless of any cell types. The numbering direction is depicted where the arrows direction indicates how the numbering will be in each quadrant. Precise numbering in the first and third quadrants is going clockwise while the numbering in second and fourth quadrants is going counter-clockwise.

To determine the sector within which a mobile user is located, apply the following steps:

Step 1: Identify the quadrant in which a mobile user is located:

To determine the quadrant within which a mobile user is located, the point (x, y) of that mobile user is compared with every interval as in the following Eq. (7)

$$j = \begin{cases} 1 \text{ when } x > 0, \quad y \geq 0 \text{ or } x \geq 0 \quad y > 0; \\ 2 \text{ when } x < 0, \quad y > 0; \\ 3 \text{ when } x < 0, \quad y \leq 0 \text{ or } x \leq 0 \quad y < 0; \\ 4 \text{ when } x > 0, \quad y < 0; \end{cases} \tag{7}$$

where j $\in [1, 4]$ to denote the location of the mobile user that has a point (x, y) with respect to the intended quadrant.

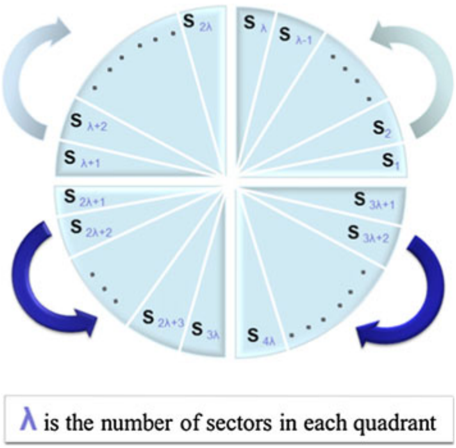

Fig. 1 Cell splitting algorithm scheme

Step 2: Identify the sector in which a quadrant is located. First, compute the R as follows: $R = \left|\frac{y}{x}\right|, \quad 0 \leq R \leq \infty$ then the values of R will be compared within the intervals according to Eq. (8) to determine where each sector belongs.

$$S = \begin{cases} S_{\lambda j-(\lambda-1)} & \text{if} \quad 0 \leq R < \tan\alpha; \\ S_{\lambda j-(\lambda-2)} & \text{if} \quad \tan\alpha \leq R < \tan 2\alpha; \\ S_{\lambda j-(\lambda-3)} & \text{if} \quad \tan 2\alpha \leq R < \tan 3\alpha; \\ . & . \\ . & . \\ . & . \\ S_{\lambda j-1} & \text{if} \quad \tan(\lambda-2)\alpha \leq R < \tan(\lambda-1)\alpha; \\ S_{\lambda j-0} & \text{if} \quad \tan(\lambda-1)\alpha \leq R \leq \infty; \end{cases} \tag{8}$$

where S represents the sector id.

Example: Let $n = 3$ and $(x, y) = (3, 4)$.

Computing the centered angle of each sector can be expressed as: $\alpha = \frac{360^\circ}{8} = 45^\circ$

Using Eq. (6) to compute the number of sectors in each quadrant $\lambda = \frac{8}{4} = 2$, this implies that there are two sectors in each quadrant and the cell is divided into eight sectors.

Using step 1 to identify the quadrant in which a mobile is located, get j = 1 (Q_1). This implies that the mobile user is located in the first quadrant.

Using step 2 to identify the sector in which a quadrant is located $R = \frac{4}{3} = 1.33$, since $\tan\alpha = \tan 45^\circ = 1$, and $\tan 2\alpha = \tan 90^\circ = \infty$, then $1 < R < \infty$ (i.e. $\tan\alpha < R < \tan 2\alpha$), and this implies that $(3, 4) \in S_{\lambda j-(\lambda-2)} = S_{2*1-(2-2)} = S_2$.

5 Location Prediction Based on Sector Snapshot

This section presents a prediction framework and a novel technique called Location Prediction based on Sector Snapshot (LPSS). This technique is based on a third generation mobile network, such as the Universal Mobile Telecommunications System (UMTS).

5.1 Prediction Framework for LPSS

The UMTS is one of the new ‘third generation’ (3G) mobile cellular communication systems being developed within the framework defined by the International Telecommunication Union (ITU) and known as International Mobile Telecommunications-2000 (IMT-2000). The UMTS aims to provide a broadband, packet-based service for transmitting video, text, digitized voice, and multimedia at data rates of up to 2\,Mb/s while remaining cost effective.

In order to demonstrate the architecture of a UMTS network, the elements of a network are introduced as the architecture of UMTS. Figure 2 illustrates the UMTS’s architecture. The UMTS is divided into three major parts: the air interface, the UMTS Terrestrial Radio Access Network (UTRAN), and the UMTS core network. The base stations and the Radio Network Controllers (RNCs) are collectively known as the UTRAN. From the UTRAN to the core network, the RNC will decide where the traffic will be transmitted. Packet traffic is sent to a Serving General Packet Radio Service (GPRS) Support Node (SGSN) and then to the Gateway GPRS Support Node (GGSN). The functions of the GGSN are very similar to the normal Internet Protocol (IP) gateway, which transfers the received packets to the appropriate Internet address. On the other hand, if there is a voice call from a subscriber, the RNC will transmit the traffic to the Mobile Switching Center (MSC). If the subscriber is already authenticated, the MSC switches the phone call to another MSC (if the call is to another mobile subscriber), otherwise the call will be switched to the Gateway MSC (GMSC) (if the call is to the public fixed phone network) [45–47].

After introducing High-Speed Downlink Packet Access (HSDPA) technology to the UMTS network, the transmission rates expected from such wireless

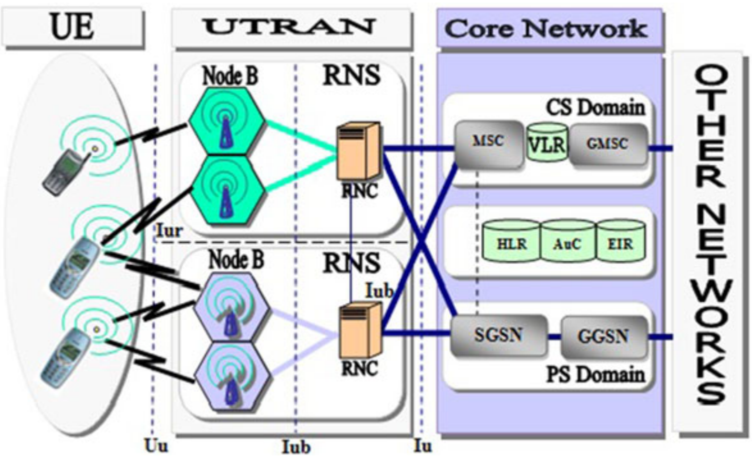

Fig. 2 UMTS architecture

communications are up to 10 Mbps [48]. For the implementation of our proposed work (LPSS), no GPS receivers are required since the control is done by the base station.

In third generation mobile networks, regions are divided into cells. The radius of a cell in a populated area is 250 m [49], whereas the non-populated areas are covered by larger size cells. This fact leads to delivering a massive amount of information. As a result, this information may degrade the accuracy of services provided to the user. In such a case, there is a need for manual filtering. Manual filtering often lets the mobile users use their devices while moving through and interacting with dynamic environments. This is intended to increase the relevance of the information retrieved by users of mobile information systems and remove results that are deemed irrelevant to a user's location. This process conflicts with the restrictions of a mobile user, such as the power consumption, storage space, screen resolution and battery performance, and low computing power and resources.

Furthermore, the manual filtering would take extra time to improve the precision of the retrieved information. This time is usually long and sometimes leads to delivery of incorrect information due to the movement of the user to a new location that has different information from the previous location. This appears especially when a large number of results are returned to the mobile user.

These filters have been implemented in LBS. These problems cannot fulfill the requirements of LBS in terms of accuracy of the prediction rate and cost effectiveness. However, these problems can be avoided and the requirements of LBS can be enhanced by three processes. First, the time by which the service is requested by the user and the time within which the user gets the service are relatively short to fit with the period of staying in that location for a specific period, whereas the proportionality between the two periods reasonably allows the user to benefit from the information associated with its current location before moving on to a new location, especially if the user is in constant motion. Secondly, the volume of results returned to users of mobile information systems is small.

Finally, enhancing the accuracy prediction leads to retrieval of information that is relevant to a user's potential future location.

A new idea for LBSs' prediction is proposed, which is called LPSS. The LPSS employs operations in a circle. The symmetry between the cell and the circle is exploited. By considering the symmetry characteristics in each quadrant, the circle section in the second quadrant of the x y plane can be generated by noting that the two circle sections are symmetric with respect to the y-axis. Furthermore, the circle section in the third and fourth quadrants can be obtained from the sections in the first and the second quadrants by considering the symmetry of the x-axis.

Based on the proposed cell splitting algorithm, which is discussed earlier in Sect. 4, each quadrant is divided into uniform sectors (λ). We know that the number of sectors in a cell (2^n), where n can be 3, 4, 5, …. In this paper the number of sectors has been chosen to be 8; whereas n = 3, which has been empirically tested for splitting the cell into the small region to be covered. Table 1 illustrates this technique. It is notable from the table that the area from Micro type using 8 sectors (98,214.28571) is smaller than the extreme area in Pico type (125,714.2857).

Table 1 Optimal area of sector for LPSS

Cell types	Radius (m)	Area (m^2)				
		No splitting	2 Sectors	4 Sectors	8 Sectors	16 Sectors
Pico	100	31,428.57143	–	–	–	–
	200	**125,714.2857**	–	–	–	–
Micro	200	125,714.2857	62,857.14286	31,428.57143	15,714.28571	7,857.142857
	500	785,714.2857	392,857.1429	196,428.5714	**98,214.28571**	49,107.14286

The LPSS addreses these problems by dividing each cell into eight equivalent sectors (small region). This technique reduces the number of relevant services within the small coverage area of each cell.

It is depicted in Table 1, that the 8 splitting sectors are chosen to be the intermediate between 4 and 16 sectors splitting. Based on 4 sectors, the service area is still large that between 4 and 16. This leads to send a huge amount of information/data to the mobile user, and this fact violates the LBSs' constraints. Meanwhile, even if using 16 sectors will result in a small service area, in this situation a crucial drawback is addressed in that the number of decisions to be made will be increased, and this affects the prediction rate.

The geometry used in the proposed cell splitting algorithm helps to reduce the volume of results returned to the users of mobile information systems. It thereby avoids the need for manual filtering and improving the precision of the information retrieved, increasing the accuracy prediction and meeting the characteristics of the mobile device such as the power consumption, storage space, and low computing power and resources.

5.2 Description LPSS Technique

In order to demonstrate the LPSS technique, a set of parameters is defined. Table 2 summarises the parameters needed to perform the LPSS technique.

To split a cell area into eight equivalent sectors, the proposed splitting algorithm in Sect. 4 is used where n=3, $\alpha = 45°$ and $\lambda = 2$ by using Eqs. (5) and (6), respectively. The eight sectors in cell C_j will be illustrated as in Eq. (9)

Table 2 LPSS parameters

Parameters	Description
j	ID of the cell
i	The Sector ID where the mobile user is located, the current location L_k at current time T_k
k	The sequence time for mobile user movements, the next location L_{k+1} will be predicted at T_{k+1}, which is the later time

$$C_j = \sum_{i=1}^{8} Sec_{c_j,i} \tag{9}$$

where i = 1, 2, …, 8 is the ID of the sector.

To determine the sector within which a mobile user is located, longitude (x) and latitude (y) of the mobile user are processed by step 1 and step 2 which are described in Sect. 4. As a result, the sector where the mobile user is located will be determined.

The dynamic movement of a mobile user through a period of time to T_{k+1} will result in changing the current location in a new neighboring sector. After a set time interval, the mobile user will have moved through a number of sectors. These sectors are stored in a database to assist in predicting a new sector to be entered.

When a new mobile user self-registers, he or she does not yet have a record in the database. The historical movement of the mobile user is derived from all mobile users, or more precisely from neighboring users. Historical data that are stored in the server can be expressed as in Eq. (10):

$$H_{(Sec_{c_j,i},t_k)} = \begin{bmatrix} N_{m,i}/N_m & N_{m+1,i}/N_m & N_{m+2,i}/N_m & \cdots & N_{m+n,i}/N_m \\ N_{m,i+1}/N_{m+1} & N_{m+1,i+1}/N_{m+1} & N_{m+2,i+1}/N_{m+1} & \cdots & N_{m+n,i+1}/N_{m+1} \\ . & . & . & \cdots & \cdots \\ . & . & . & \cdots & \cdots \\ N_{1,n}/N_n & N_{2,n}/N_n & N_{3,n}/N_n & \cdots & N_{n,n}/N_n \end{bmatrix} \tag{10}$$

where N_m is the number of the traversal over sector m, and $N_{m,i}$ is the number of times the user has entered sector i when the user had been on sector m. When the user locates at $Sec_{c_j,i}$ at T_k then the available sectors at T_{k+1} are $Sec_{c_j,i+1}, Sec_{c_j,i-1}$ and the facing sector in the neighboring cell $Sec_{c_p,q}$, where p is the neighboring cell

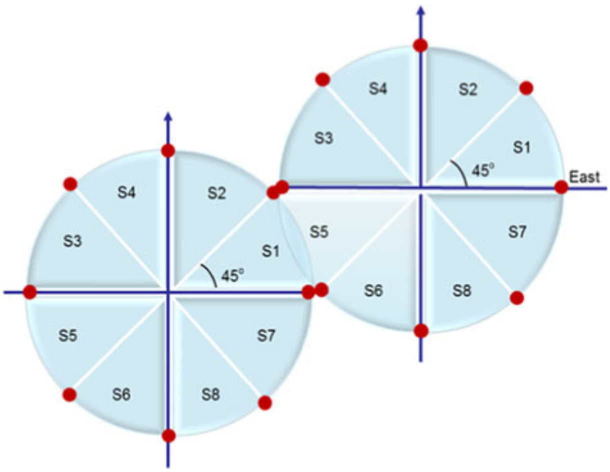

Fig. 3 Movement of mobile user between sectors in two cells

ID and q is the facing sector *ID*. Based on Fig. 3, when the user is located at $Sec_{c_1,1}$, then $N_1 = 1, N_{2,1} = N_{8,1} = N_{Sec_{c_2,5},1} = 1/3$.

The historical matrix is periodically updated to achieve consistency. It is also updated when N_j is incremented by one and $N(j, i)$ is incremented by one. To reach the probabilistic information for the predicted next state, a transition matrix is needed as in Eq. (11):

$$TP(Sec_{c_j,i}, t_k) = \begin{bmatrix} P(Sec_{c_j,1}, Sec_{c_j,1}) & P(Sec_{c_j,2}, Sec_{c_j,1}) & \ldots & P(Sec_{c_j,n}, Sec_{c_j,1}) \\ P(Sec_{c_j,1}, Sec_{c_j,2}) & P(Sec_{c_j,2}, Sec_{c_j,2}) & \ldots & P(Sec_{c_j,n}, Sec_{c_j,2}) \\ P(Sec_{c_j,1}, Sec_{c_j,3}) & P(Sec_{c_j,2}, Sec_{c_j,3}) & \ldots & P(Sec_{c_j,n}, Sec_{c_j,3}) \\ . & . & \ldots & \ldots \\ . & . & \ldots & \ldots \\ P(Sec_{c_j,1}, Sec_{c_j,n}) & P(Sec_{c_j,2}, Sec_{c_j,n}) & \ldots & P(Sec_{c_j,n}, Sec_{c_j,n}) \end{bmatrix} \tag{11}$$

The current state of a mobile user after registration in a network can be represented as in Eq. (12):

$$Currentstate = [1 \quad 0 \quad 0 \quad 0] \tag{12}$$

So the next state will be predicted after multiplying Eq. (12) by Eq. (11). The resultant vector is expressed in Eq. (13):

$$Pr = [Pr_0 \quad Pr_1 \quad Pr_2 \quad Pr_3] \tag{13}$$

Where Pr is the probability that the mobile user will travel to surrounding sectors, and $Pr_0 + Pr_1 + Pr_2 + Pr_3 = 1$.

Logically, the values of Pr will give the indication of the next sector to be visited in the next state since the highest Pr will give the highest probability of the sector. Generally, to generate more predictable sectors for further states, Eq. (13) will be multiplied by transition matrix 11. In other words, the operation that resulted in Eq. (13) will be repeated.

Discussions are now presented on the phases used in simulating the LPSS technique and the steps that follow. Figure 4 shows the pseudo-code of LPSS. *InitilaizeEnvironmentParameters()* initializes the parameters, which is an important phase in the simulation. However, all parameters were initialized before the simulation began.

To ensure that the accuracy of the results is not affected by previous turns, the procedure of splitting cells is a predefined step. In this process, a virtual splitting of each cell in a network is performed in order to produce eight sectors. The splitting process is done once, without the need to recalculate because it is not affected by natural changes such as closed roads, maintenance, and congestion. Consequently, the splitting is excluded from the computation cost as it has been processed before running the technique. The splitting process is done by *SplitSector(cells)*, giving Eqs. (4–6). When the execution is started, the information about the sectors is stored in the database by *CellSector(cellid, sectorid)*.

When a mobile user registers on a network, the current *x, y* are provided to LPSS through the base station where the mobile user is located. The specific sector in which the mobile user is located will be calculated by passing the *x, y* to find the

```
Procedure InitilaizeEnvironmentParameters ()
Procedure SplitCellToRegion ()
   while (cell in a specific cellular network)
         call SplitSector(cells)
         call CellSector(cellid, sectortid)
   End while
End procedure

Procedure MobileRunning()
   while (mobile user is still moving)
        get angle of the mobile user
        call FindSector(x,y)
        call MatrixCalculation( )
        call prediction ( )
        call UpdateEnvironmentParameters ()
   End while
End Procedure
```

Fig. 4 Pseudo-code of sector snapshot for location-based services

sector *FindSector(double x, double y)* using step 1 and step 2, which are described in Sect. 4. The output for this step is the sector identification where the mobile user is located.

The mobile user in LPSS can be in one of two states, FIRST REGISTRATION of HOME USER. In the FIRST REGISTRATION state, the historical movement matrix builds from the neighbors. This information is stored in a database. In the HOME USER state, although there is a historical movement matrix, there remains a need to update it to ensure the matrix is consistent and up-to-date. The two states are implemented by *MatrixCalculation()*, giving Eqs. (10–12).

Transition and probability matrices are built based on the historical movement matrix. Therefore, the historical movement matrix must be up-to-date for the mobile user movements. The probabilities of each sector around the mobile user are derived based on the transition matrix. In the meantime, the current state vector for the mobile user is initialized and then multiplied by the transition matrix. The result is the probability that the user will be moving to each sector around the user, i.e., the potential sectors. This process is done by *Prediction ()*, giving Eq. (13).

All of the above steps are repeated until some cases occur, such as out of coverage, or the simulation time has expired. In each case, the parameter environment will be updated by *UpdateEnvironmentParameter ()*.

6 Discussion of Simulation and Results Analysis

6.1 Parameter Setup and Environment

Any mobile network simulation needs to be done over a very large coverage area. That is why the parameter setting is set using the parameter assumption, as described in pervious research and standardized over 3GPP specifications [50, 51], as shown in Table 3.

The simulations are done over Pentium IV computers with 2 GB RAM and a CPU speed of 3 GHz. The operating system used was Windows XP, where the LAN speed was 100 Mbps.

A simulator was created using Java programming language for the LPSS, in which the algorithm based on Markov chain models is implemented and tested. The number of cells in the simulated experiments varies between one, two, three, five, fifty and one hundred cells with a fixed radius of 250 m each. The movement with different speeds (slow pedestrian speed, fast pedestrian, slow vehicle, and fast vehicle with measures of (5.6, 11.2, 44.8 and 89.6 km/h), respectively) was recorded to train the program to learn how the mobile user moves during different trips. Different samples of data were used to test the performance of the LPSS. In addition, the pause time for each movement was 20 s. The transmission rate was about 8 Mbps.Those parameters were the keys that the simulator applied for 1,800 s (see Fig. 5, a visual representation for the simulation). The movement was recorded to train the program to learn how the mobile user moved during different trips. Different samples of data were used to test the performance of the LPSS.

Each experiment consisted of 10 different iterations to improve accuracy. Each experiment took five hours, as shown in Table 3.

6.2 Simulation Model

The proposed model was simulated to calculate the prediction ratio and the cost per unit of time for mobile users with different mobility for each request ratio. The implementation was constructed as follows:

1. **Network Modeling:** The same cell identification coding system was used as that in [52]. Each cell had a unique identifier, determined by its x and

Table 3 Simulation parameters

Parameter	Value
Number of cells	100
Cell radius	250 m
Transmission rate	8 Mbps
Simulation time	18,000 s
Iterations	10
Pause time	20 s
Velocity of UE	
Slow pedestrian	5.6 k/h
Fast pedestrian	11.2 k/h
Slow vehicle	44.8 k/h
Fast vehicle	89.6 k/h

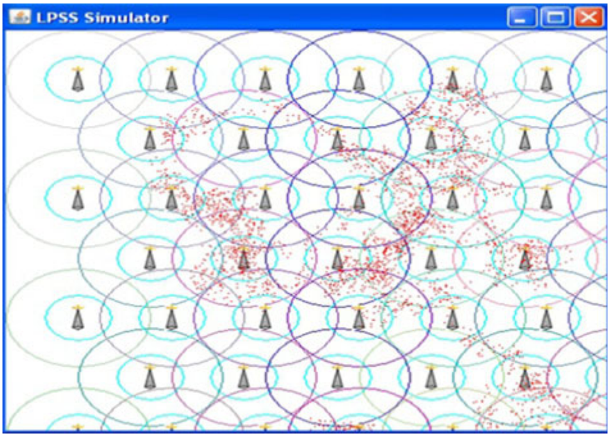

Fig. 5 LPSS environment

y coordinates. Numbering started from the center cell and expanded radially across the entire network.

2. **Request Service Modeling:** Services arrived for mobile users according to a Poisson distribution with mean $m = 1$ per unit of time.
3. **Mobility Modeling:** The Random-Way point mobility model and simulation model implemented the users' cell residence time with a Poisson distribution with mean r.

6.3 Experiments and Result Analysis

Corresponding with the prediction performance analysis, some phases of experiments were designed to evaluate the proposed technique, which included: Phase-1: experiments that evaluated the prediction accuracy, which is the ratio between the number of correct predictions and the total number of predictions [14]. Phase-2: experiments that evaluated the memory usage reduction. Phase-3: experiments that evaluated the execution time. Phase-4: experiments that evaluated the prediction rate over time.

In phase 1, the performance of the LPSS was compared with both NMMP and PLM techniquees in terms of prediction accuracy rate. Figure 6 shows that LPSS improved the prediction rate compared with both NMMP and PLM. This improvement was due to the nature of the mobility pattern in which the mobile user traveled into a series of predicted cells and the changing number of cells visited within a trip. This conclusion is generally valid, though the improvement may differ with a different set of assumptions.

Moreover, Fig. 6 shows that the percentage of correct predictions in the LPSS was more than 95.61 %, as the mobile user moved through one cell, compared with 76.98 and 65.13 % in both the NMMP and PLM techniques, respectively. An increase in the number of cells leads to a decrease in the correct prediction rate in both techniques. The explanation for the achieved results could be stated as follows:

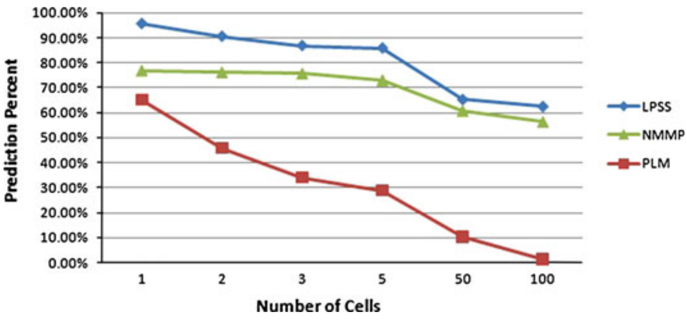

Fig. 6 Prediction performance for sector snapshot for location-based services

when the mobile user moves over more than one cell, the end trip can be reached from different routes and through different cells due to the size of the cells, so such a prediction will be low. On the other hand, in one cell, the movement of the mobile user in the LPSS is kept within the bounds of the sectors within the same cell. Therefore, the mobile user tries to leave the sector to the neighboring sectors and is still within a relatively small area. Thus, the prediction percentage logically increases. In the NMMP technique, the number of selection choices is larger than LPSS, which decreases the accuracy of the next displacement for the mobile user. On the other hand, while using the PLM technique, the movement of the mobile user within one cell will involve many intersections in different routes.

It is acknowledged that the main factor for location based services is the knowledge of the next location of a given user movement. This fact is satisfied more by LPSS than by both NMMP and PLM, respectively. Table 4 summarizes this satisfaction by taking the overall average correct prediction rate for LPSS, NMMP, and PLM.

Each cell has a circular radius shape of approximately 250 m. Therefore, the area of each cell approximately equals 0.2 km^2 and the area of five cells equals 1 km^2. LPSS shows a high correct prediction rate in areas whose size is approximately five

Table 4 Prediction Rate for LPSS, NMMP, and PLM

Number of cells	LPSS (%)	NMMP (%)	PLM (%)
1	95.61	76.98	65.13
2	90.54	76.27	45.90
3	86.85	75.80	34.05
5	85.87	73.13	28.84
50	65.38	10.34	60.89
100	62.58	56.45	1.50
Prediction rate average	81.1365	30.96	69.92

cells. In practice, this area is sufficient for users' activities in urban areas as it is a typical size for a city center, university campus, or small town.

In phase 2, the importance of the memory usage reduction was utilized. To test the effectiveness of the memory usage reduction, different numbers of cells and the coverage area were studied. The average of memory usage in each technique was used to test the memory reduction. The LPSS technique required 14.65 KB for space storage while both NMMP and PLM required 27.77 KB and 121.91, respectively.

Figure 7 illustrates the differences in total memory usage among the LPSS, NMMP, and the PLM techniques. In comparison with the different techniques, the proposed LPSS technique performed better in reducing the memory usage. This conclusion is generally valid, though the improvement may differ with a different set of assumptions. This improvement is due to the division of the cell into a set of sections, where each section acts as a serving region that retrieves the information related to that section only.

Furthermore, the mobility pattern can be used in which the mobile user travels into a series of predicted cells and changes ocurs in the number of cells that were visited within a trip. Please note that whenever the number of cells increased, the memory usage increased. This is due to the fact that whenever more cells were used, more computation and memory was used. This fact will be discussed in phase 3. In conclusion, the mobility pattern factor considers of the other different factors that they may reduce the memory usage.

In phase 3, the execution time was studied in all three different techniques: the LPSS, NMMP, and the PLM. Figure 8 shows the results of the three techniques in terms of execution time using the same method that was used in both phases 1 and 2, with a variation in the number of cells from 1 to 10.

In comparison with the execution for the three techniques, the execution time for the proposed LPSS technique performed better. Figure 8 summarizes the results of the LPSS execution time compared to the results of the NMMP and PLM execution time.

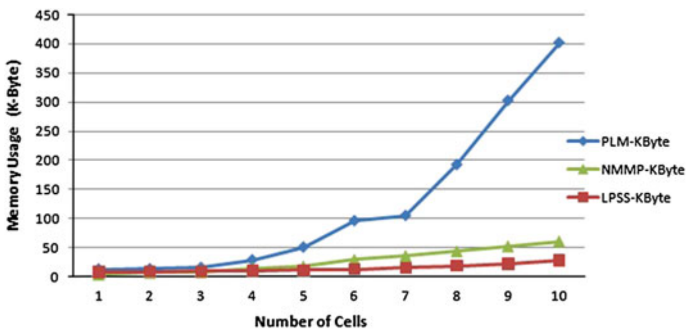

Fig. 7 Memory usage for LPSS, NMMP and PLM

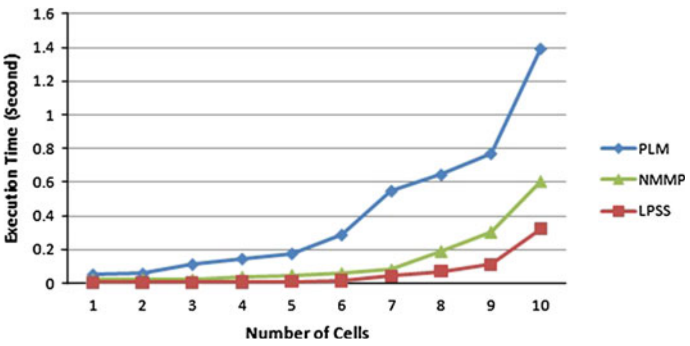

Fig. 8 The excution time for LPSS, NMMP and PLM

The LPSS technique outperforms both NMMP and PLM in every number of cells, but this is especially apparent when the number of cells increases. For example, when the number of cells is 1, the LPSS technique requires about 4.22 ms, while the NMMP and the PLM techniques require 20.89 and 52.97 ms, respectively. When the number of cells increases to 10, LPSS requires only 0.322 s, while both NMMP and PLM require about 0.602 and 1.39 s, respectively.

The mechanism of the LPSS technique depends on the virtual splitting of the cell into sectors. This virtual splitting does not affect the built in database and structures, whenever there is any change in the physical serviced sector. On the other hand, the PLM acts as a "roundabout". The real change in intersections, between roads and physical features, leads to costs for updating the database and structures, which are related to the user predictions. Also, the NMMP technique works on the whole cell area as a unit without splitting it, which leads to usage of manual filtering. This is done because a huge amount of data/information will be delivered to the mobile user.

Furthermore, PLM has a large number of possibilities when a user wants to decide the next road, so the user faces a larger number of possibilities than in the LPSS technique. The nature prediction of the NMMP technique depends on the two levels of prediction described earlier, which lead to more possibilities compared to the LPSS technique.

Finally, the overall number of possibilities in both the NMMP and the PLM techniques is larger than the possibilities in the LPSS, which decreases the correct prediction percentage and increases the execution time.

To complete phase 4, over a period of time the prediction behavior is carried out for improving the prediction mechanism. It also considered as one of the important factors to improve the prediction mechanism. The outcome of this behavior will make a good measure for the robustness of the mechanism from the mobile user's side. This is addition to the ability to deal with challenges such as visiting a new location that has never been visited before and utilizing mobile user neighbors' behavior. Figure 9 shows a description for the prediction rate variations according to 180 days.

At this point, two factors are needed to be considered. These are the robustness and the variation of the prediction rate when the algorithm reaches the steady state. Moreover, there is a very tight relation between those two factors in that, whenever the variation of the prediction rate increases the robustness will decrease, making it as inverse relationship.

Furthermore, a very light regression is noticed in the LPSS mechanism where it is very clear that a lot of regressions take place in both NMMP and PLM after reaching the steady state.

Additionally, both NMMP and PLM are working with different procedures that depend on combination of two levels of prediction and road segments, respectively. This leads to prediction states that never have been handled before in addition to what is noticed in Fig. 9 of a very clear regression after reaching the steady state by both mechanisms. The proposed LPSS mechanism is working on one level, the NMMP is working on two levels, while the PLM depends on making an extensive number of divisions, as shown in Table 5. Table 5 summarizes the nature of LPSS, NMMP, and PLM.

In this analysis, the performance was evaluated by adopting an evaluation methodology to gauge the impact of the enhancement technique on the accuracy prediction, memory usage, complexity cost, and prediction rate over time. The methodology that was used to test and validate the comparisons among the LPSS, NMMP, and the PLM, as well as between analytical and simulation results, is as discusses in Sect. 7.

7 Analytical Analysis

This section investigates and analyses the NMMP and PLM prediction techniques for UMTS mobile networks and the proposed LPSS prediction technique. The investigation is done by two comparisons as follows:

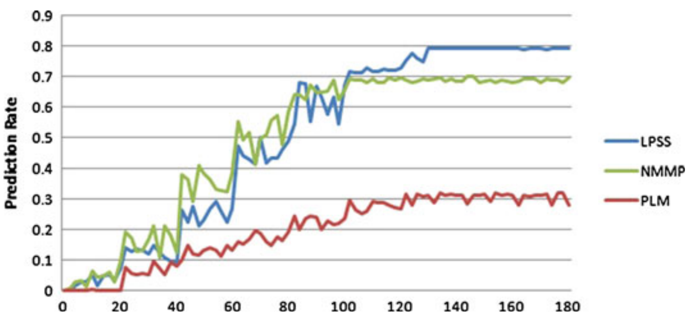

Fig. 9 Prediction rate according to time for LPSS, NMMP and PLM

Table 5 Techniques nature for LPSS, NMMP, and PLM

	LPSS	NMMP	PLM
Number of levels	1	2	1
Number of divisions	Lowest	Medium	Largest

First, compare LPSS with PLM. Suppose the probability of making a decision at each intersection is *P(A)* for the PLM technique and *P(B)* for the LPSS technique. If A_1, A_2, A_3, A_4, …, A_n are the required decision points to reach from point x to point y using PLM, then $P(A_1)$, $P(A_2)$, $P(A_3)$, $P(A_4)$, …, $P(A_n)$ are the probabilities associated with these decisions, respectively.

If B_1, B_2, B_3, B_4, …, B_m are the required decision points to reach from x to y using LPSS, then $P(B_1)$, $P(B_2)$, $P(B_3)$, $P(B_4)$, …, $P(B_m)$ are the probabilities associated with these decisions, respectively, where $m < n$ and $P(A) \cong P(B)$ the worst case of LPSS.

Since $P(A_1)$, $P(A_2)$, $P(A_3)$, $P(A_4)$, …, $P(A_n)$ are independent, i.e., $P(A_1 \cap A_2) = P(A_1)P(A_2)$ and $P(B_1)$, $P(B_2)$, $P(B_3)$, $P(B_4)$, …, $P(B_m)$ are independent, i.e., $P(B_1 \cap B_2) = P(B_1)P(B_2)$ then:

$$P(y|A) = (P(A))^n \tag{14}$$

$$P(y|B) = (P(B))^m \tag{15}$$

where $0 < P(A) \cong P(B) < 1$ and since $m < n$ then $(P(B))^m > (P(A))^n$, i.e. $P(A)$ converges to zero faster than *P(B)*.

Secondly, compare LPSS with NMMP. In NMMP there are two levels of prediction, where level 1 has *6* decisions:*C1, C2, …, C6* and level 2 has two decisions in the best case: *D1* and *D2*.

The probability of decisions in level 1 are *P(C1), P(C2), …, P(C6)*, which are independent with $P(C1) = P(C2) = \cdots = P(C6)$. The probabilities of decisions in Level 2 are *P(D1)* and *P(D2)*, which are independent with $P(D1) = P(D2)$.

Let $P(C1) + P(C2) + \cdots + P(C6) = P(C)$ and $P(D1) + P(D2) = P(D)$. Then $P(C) + P(D) = 1$ and $P(C) = 1 - P(D)$.

Since level 2 is only reached in the case of failure on level 1, *P(C)* is the probability of success and *P(D)* is probability of failure. Hence, *P(success)=P(C)=1-P(D)=1-P(failure)*. This is a Bernoulli distribution with two possible outcomes [53], where *{x:1 = success, 0 = Failure}*, so the probability of *P(C)* to success = $f(x) = P(C)^x\,(1 - P(C))^{(1-x)}$ and *x:{0,1}*.

Here, there are *6* independent for C and 2 independent decisions for D, with a total of *8* decisions. Therefore, $P(C) = 6/8$ and $P(D) = 2/8$. Under the NMMP process, it is a hierarchical prediction process, with level 1 followed by level 2. Therefore, if $x = 1$ in the Bernoulli distribution, $P(C_i) = 1/6$, $1 \le i \le 6$, and if $x = 0$ then $P(D_j) = 1/2$, $j = 1$ *or* 2.

Then, the successful prediction at Level 1 is

$$P(C_i|x = 1) = 6/8 \times 1/6 = 1/8 \tag{16}$$

and the failed prediction at level 1 will lead to level 2 with

$$P(D_i|x = 0) = 2/8 \times 1/2 = 1/8 \qquad (17)$$

While in the proposed LPSS, the correct prediction can take only 1 of 3 cases. Therefore, the correct prediction probability is 1/3. Based on Eqs. (16) and (17), the correct prediction for NMMP is 1/8, so 1/3 > 1/8, i.e., NMMP converges to zero faster than LPSS.

Since the number of possibilities used in the NMMP technique and PLM are larger than the LPSS technique, the execution time and memory usage are larger in NMMP and PLM, i.e., if the time required for processing each decision point is t, then the total time required to process n decision points in NMMP and PLM are $t \times 8$ and $n \times t$, respectively. Similarly the time required for processing m decision points using LPSS is $m \times t$, since $m < n$ and $m < 8$, then $m \times t < n \times t$ and $m \times t < 8 \times t$.

8 Conclusion and Future Work

A novel Splitting cell algorithm and an efficient Location Prediction Sector Snapshot (LPSS) are suggested. The new splitting algorithm is applicable for all cell types, and the efficiency in determining the location of the mobile user's movement is also achieved.

The new prediction technique, LPSS, is more efficient than the NMMP and the PLM. The suggested technique also provides other characteristics. For example, LPSS minimizes the computation cost, consumption of resources, and the overall cost of the location management process. Also, the LPSS reduces the service area and the number of predicted routes during the mobile user trip, by dividing the cell into eight equivalent sectors. Consequently, the LPSS technique improves the location prediction probability over NMMP and PLM. In addition, the average complexity requirements for execution time and usage space are smaller than for the NMMP and PLM techniques. The simulation results have demonstrated that the average prediction accuracy rate, the memory usage, the execution time, and the robustness and regression degree of prediction rate over time are improved when compared with the NMMP and PLM techniques.

Finally, PLM does not allow a given user to visit each of the trajectories more than once for the whole trip. That means PLM is not a practical technique. LPSS does not suffer from this problem. Meanwhile, NMMP is based on a hierarchical prediction process, and this leads to an increase in message passing on the network, delay time and overhead on resource which these were avoided in the LPSS.

Many refinements can be made to improve the prediction rate and accelerate the proposed technique. For instance, the LPSS currently works at a Micro cell level by dividing a cell into eight regions (sectors), but this technique can be altered to work on a higher level than cells, e.g., at the level of Routing Areas (RAs) or works belonging to another RA prediction technique. At the RA level, the RA contains a group of cells, and a location prediction will specify the next RA that a mobile user will visit. Meanwhile, the LPSS evaluates all movement probabilities for the next RA before the mobile user enters it, as shown in the result discussion. LPSS will give a better prediction rate because it will work on a set of cells under the same RA. LPSS illustrated a better prediction rate when the number of cells was less than five cells; thus, LPSS is worthy to work on a less number of cells as much as possible.

Moreover, the Macro and Rural cells still need more investigation to determine how many sectors are suitable for both of them, and the proposed splitting algorithm is applicable for dynamic splitting. Therefore, the short cut for dynamic splitting also needs more investigation.

Several challenges remain. For example, is there any other technique that offers the same accuracy rate as we propose but with better performance? is the design of a prediction technique for LBSs that utilizes information about user location through location-aware mobile devices to provide services, such as the nearest features of interest, perhaps based on other multiple assumptions but with better performance?

References

1. Passerini, K., Patten, K., Bartolacci, M.R., Fjermestad, J.: Reflections and trends in the expansion of cellular wireless services in the U.S. and China. Commun. ACM **50**(10), 25–28 (2007)
2. Sun, Y., Belding-Royer, E.M., Gao, X., Kempf, J.: Real-time traffic support in heterogeneous mobile networks. Wirel. Netw. **13**(4), 431–445 (2007)
3. Wadear, R., Fagoonee, L.: Beyond third generation (B3G) mobile communication: challenges, broadband access and Europe. In: Mobility ’06: Proceedings of the 3rd International Conference on Mobile Technology, Applications & Systems, p. 5. ACM, New York, NY (2006)
4. Induruwa, A.: Mobile phone forensics: an overview of technical and legal aspects. Int. J. Electron. Secur. Digit. Forensic **2**(2), 169–181 (2009). doi:10.1504/IJESDF.2009.024901
5. Fitzek, F., Schulte, G., Reisslein, M.: System architecture for billing of multi-player games in a wireless environment using GSM/UMTS and WLAN services. In: NetGames ’02: Proceedings of the 1st Workshop on Network and System Support for Games, pp. 58–64. ACM, New York, NY (2002)
6. Iftikhar, M., Landfeldt, B., Caglar, M.: Traffic engineering and QoS control between wireless diffserv domains using PQ and LLQ. In: MobiWac ’07: Proceedings of the 5th ACM International Workshop on Mobility Management and Wireless Access, pp. 120–129. ACM, New York, NY (2007)
7. Masri, S.A., Hunaiti, Z.: The impact of zoning concept on data-flow management within LBS system components. IJHCR **1**(1), 43–63 (2010). doi:10.4018/jhcr.2010090903
8. Escalle, P.G., Giner, V.C., Oltra J.M.: Reducing location updates and paging costs in a pcs network. IEEE Transactions on Wireless Communications **1**(1), 200–209 (2002)
9. Zheng, J., Zhang, Y., Wang, L., Chen, J.: Adaptive location update area design for wireless cellular networks under 2D Markov walk model. Comput. Commun. **30**(9), 2060–2069 (2007)
10. Barbar, D.: Mobile computing and databases-a survey. IEEE Trans. Knowl. Data Eng. **11**, 108–117 (1999). doi:10.1109/69.755619
11. Wu, S.Y., Wu, K.T.: Effective location based services with dynamic data management in mobile environments. Wirel. Netw. **12**(3), 369–381 (2006). doi:10.1007/s11276-005-5280-0
12. Dunham, M.H., Kumar, V.: Location dependent data and its management in mobile databases. In: DEXA ’98: Proceedings of the 9th International Workshop on Database and Expert Systems Applications, p. 414. IEEE Computer Society, Washington, DC (1998)
13. Seydim, A.Y., Dunham, M.H., Kumar, V.: Location dependent query processing. In: MobiDe ’01: Proceedings of the 2nd ACM International Workshop on Data Engineering for Wireless and Mobile Access, pp. 47–53. ACM, New York, NY (2001). doi:10.1145/376868.376895
14. Francois, J.-M., Leduc, G.: Mobility prediction’s influence on QoS in wireless networks: a study on a call admission algorithm. In: WIOPT ’05: Proceedings of the Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, pp. 238–247. IEEE Computer Society, Washington, DC (2005). doi:10.1109/WIOPT.2005.29

15. Das, S.K., Sen, S.K.: Adaptive location prediction strategies based on a hierarchical network model in a cellular mobile environment. Comput. J. **42**(6), 473–486 (1999)
16. Biesterfeld, J., Ennigrou, E., Jobmann, K.: Location prediction in mobile networks with neural networks. In: The International Workshop on Applications of Neural Networks to Telecommunications, pp. 207–214 (1997)
17. Kubach, U.: A map-based, context-aware hoarding mechanism. In: Berichtskolloquium des Graduiertenkollegs Parallele und Verteilte Systeme, University of Stuttgart, Germany (2000)
18. Vijay Kumar, B.P., Venkataram, P.: Prediction-based location management using multilayer neural networks. J. Indian Inst. Sci. **82**, 7–21 (2002)
19. Shah, S.H., Nahrstedt, K.: Predictive location-based QoS routing in mobile ad hoc networks. In: IEEE International Conference on Communications (ICC '02), vol. 2, pp. 1022–1027, New York, NY (2002)
20. Kubach, U., Rothermel, K.: An adaptive, location-aware hoarding mechanism. In: Proceedings. ISCC 2000. Fifth IEEE Symposium on Computers and Communications, pp. 615–620, Antibes-Juan les Pins, France (2000)
21. Holma, H., Toskala, A.: WCDMA for UMTS: radio access for third generation mobile communications. Trans. Netw. **9**(6), 790–800 (2001)
22. Liou, S.-C., Lu, H.-C.: Applied neural network for location prediction and resources reservation scheme in wireless networks. In: International Conference on Communication Technology Proceedings, ICCT 2003, IEEE, vol. 2, pp. 958–961 (2003)
23. Capka, J., Boutaba, R.: Mobility Prediction in Wireless Networks Using Neural Networks, vol. 3271, pp. 320–333. Springer, Berlin, Heidelberg (2004)
24. Sadilek, A., Kautz, H., Bigham, J.P.: Finding your friends and following them to where you are. In: Proceedings of the fifth ACM international conference on Web search and data mining (WSDM '12), pp. 723–732. ACM, New York, NY (2012). doi:10.1145/2124295.2124380
25. Daoud, M.Sh., Ayesh, A., Hopgood, A.A., Al-Fayoumi, M.: A new splitting-based displacement prediction approach for location-based services. In: 2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 392–397 (2011). doi:10.1109/ICSMC.2011.6083697
26. Ren, M., Karimi, H.: A fuzzy logic map matching for wheelchair navigation. GPS Solutions **15**, 1–10 (2011). doi:10.1007/s10291-011-0229-5
27. Ren, M., Karimi, H.A.: A hidden Markov model-based map-matching algorithm for wheelchair navigation. J. Navig. **62**(3), 383–395 (2009). doi:10.1017/S0373463309005347
28. Ren, M., Karimi, H.A.: A chain-code-based map matching algorithm for wheelchair navigation. Trans. GIS **13**(2), 197–214 (2009). doi:10.1111/j.1467-9671.2009.01147.x
29. Karimi, H.A., Liu, X.: A predictive location model for location-based services. In: GIS '03: Proceedings of the 11th ACM International Symposium on Advances in Geographic Information Systems, pp. 126–133. ACM, New York, NY (2003)
30. Wu, C.-F., Lee, L.-T., Tao, D.-F.: An HMM prediction and throttling-based call admission control scheme for wireless multimedia networks. Comput. Math. Appl. **54**(3), 364–378 (2007). doi:10.1016/j.camwa.2007.01.035
31. Sun, M.H., Blough, D.M.: Mobility prediction using future knowledge. In: Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM '07), pp. 235–239. ACM, New York, NY (2007). doi:10.1145/1298126.1298167
32. Soh, W.-S., Kim, H.S.: A predictive bandwidth reservation scheme using mobile positioning and road topology information. IEEE/ACM Trans. Netw. **14**(5), 1078–1091 (2006). doi:10.1109/TNET.2006.882899
33. François, J.-M., Leduc, G., Martin, S.: Evaluation d'une méthode de prédiction des déplacements de terminaux dans les réseaux mobiles. In: Actes de Colloque Francophone sur l'Ingénierie des Protocoles, pp. 189–202. Paris, France (2003)
34. Liu, G., Maguire, G.: A class of mobile motion prediction algorithms for wireless mobile computing and communications. Mobile Netw. Appl. **1**(2), 113–121 (1996). doi:10.1007/BF01193332
35. Bellahsene, S., Kloul, L.: A new Markov-based mobility prediction algorithm for mobile networks. In: Proceedings of the 7th European Performance Engineering Conference on Computer Performance Engineering (EPEW '10), pp. 37–50. Springer-Verlag, Berlin, Heidelberg (2010). http://dl.acm.org/citation.cfm?id=1926981.1926986
36. Bellahsene, S., Kloul, L., Barth, D.: A hierarchical prediction model for two nodes-based IP mobile networks. In: Proceedings of the 12th ACM International Conference on Modeling, Analysis and

Simulation of Wireless and Mobile Systems (MSWiM '09), pp. 173–180. ACM, New York, NY (2009). doi:10.1145/1641804.1641835
37. Pang, A.-C., Lin, Y.-B., Tsai, H.-M., Agrawal, P.: Serving radio network controller relocation for UMTS all-IP network. IEEE J. Sel. Areas Commun. **22**(4), 617–629 (2004). doi:10.1109/JSAC.2004.825962
38. Lin, P., Lin, Y.-B., Chlamtac, I.: Overflow control for UMTS high-speed downlink packet access. IEEE Trans. Wirel. Commun. **3**(2), 524–532 (2004). doi:10.1109/TWC.2003.821152
39. Migallón, H., Migallón, V., Penadés, J.: Alternating two-stage methods for consistent linear systems with applications to the parallel solution of Markov chains. Adv. Eng. Softw. **41**(1), 13–21 (2010). doi:10.1016/j.advengsoft.2008.12.021
40. Buchholz, P.: Structured analysis techniques for large Markov chains. In: SMCtools '06: Proceeding from the 2006 Workshop on Tools for Solving Structured Markov Chains, p. 2. ACM, New York, NY (2006). doi:10.1145/1190366.1190367
41. Brooks, S.P., Roberts, G.O.: Convergence assessment techniques for Markov chain Monte Carlo. Stat. Comput. **8**(4), 319–335 (1998). doi:10.1023/A:1008820505350
42. Andreyevich, A.: "Rasprostranenie zakona bol'shih chisel na velichiny, zavisyaschie drug ot druga". Izvestiya Fiziko-matematicheskogo obschestva pri Kazanskom universitete. In: 2-ya seriya, vol. 15, pp. 135–156 (1906)
43. Andreyevich, A.: Extension of the limit theorems of probability theory to a sum of variables connected in a chain, reprinted in Appendix B of R. Howard. In: Dynamic Probabilistic Systems, vol. 1: Markov Chains. John Wiley and Sons (1971)
44. Markovski, J., Sokolova, A., Trčka, N., de Vink, E.P.: Compositionality for Markov reward chains with fast and silent transitions. Perform. Eval. **66**(8), 435–452 (2009). doi:10.1016/j.peva.2009.01.001
45. Bahl, P., Padmanabhan, V.: RADAR: an in-building RF-based user location and tracking system. In: Proceedings of IEEE Infocom, pp. 775–784 (2000)
46. Ashbrook, D., Starner, T.: Learning significant locations and predicting user movement with GPS. In: Proceedings of IEEE Sixth International Symposium on Wearable Computing, pp. 101–108 (2002)
47. Moustafa, M., Habib, I., Naghshineh, M.: GAME based dynamic resource scheduling in QoS aware radio access networks. Soft Comput. **9**(2), 101–115 (2005)
48. Harmatos, J.: Planning of UMTS Core Networks. In: 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 2, pp. 740–744 (2002)
49. Wardlaw, M.I.: Intelligence and mobility for BT's next generation networks. BT Technol. J. **23**(1), 28–47 (2005)
50. Al-Manthari, B., Nasser, N., Hassanein, H.: Fair channel quality-based scheduling scheme for HSDPA system. In: AICCSA '06: Proceedings of the IEEE International Conference on Computer Systems and Applications, pp. 221–227. IEEE Computer Society, Washington, DC (2006). doi:10.1109/AICCSA.2006.205093
51. 3GPP: Third Generation Partnership Project: High Speed Downlink Packet Access (HSDPA); Overall Description 3GPP TS 25.308, release 5 (2003). http://www.3gpp.org
52. Hwang, H., Chang, M., Tseng, C.: A direction based location update scheme with a line paging strategy for PCS Networks. IEEE Commun. Lett. **4**, 149–151 (2000)
53. Johnson, N.L., Kemp, A.W., Kotz, S.: Univariate Discrete Distributions, p. 29, 3rd edn. Wiley-Interscience, New York (2005)

Author Biographies

Mohammad Sharif Daoud has been awarded B.Sc. (Hons) degree in computer information system from Al-Zaytoonah University of Jordan in 2004. Four years later (2008), he has been awarded MSc degree in computer science from The University of Jordan. Furthermore, he is currently studying a Ph.D. course in communication and media in location based-services at De Montfort University in the United Kingdom. His specialist areas include wireless and mobile networks, mobility prediction, ants' colony optimisation, MultiAgent and real time multimedia over UMT All-IP network.

Aladdin Ayesh M.Sc. (Essex, 1995), Ph.D. (LJMU, 2000), has a chartered status in engineering (CEng), science (CSci) and as IT practitioner (CITP). He is a fellow of British Computer Society (FBCS), a senior member of IEEE, and a Reader in AI at De Montfort University. He lectures on range of subjects including multimedia, game programming, multi-agents and robotic systems, and system analysis and design. His research involves theoretical and practical work in the areas of swarm intelligence, cognitive systems and Arabic natural language processing. He has over 100 published works in international journals, conferences, edited books and two scientific books.

Mustafa A. Al-Fayoumi received the B.S. degree in computer science from Yarmouk University, Irbid, Jordan, in 1988. He received the M.S. degree in computer science from the University of Jordan, Amman, Jordan, in 2003. In 2009, he received a Ph.D. degree in computer science from the Faculty of Science and Technology at Anglia University, UK. In 2009, he joined the Al-Zaytoonah University, in Jordan, as an assistant professor. Currently, he is assistant professor and chairman of computer science department at Salman Bin Abdul Aziz University, Saudi Arabia. His research interests include areas like computer security, cryptography, identification and authentication, wireless and mobile networks security, e-application security, simulation and modeling, algorithm analyzes and design, information retrieval and any other topics related to them. He has published more than 10 research papers in international journals and conferences.

Adrian Hopgood joined Sheffield Hallam University in 2011, having previously worked for De Montfort University, Nottingham Trent University, the Open University, Telstra Research Laboratories and Systems Designers PLC. He has published over 100 papers and supervised 15 PhD projects to completion. He holds a doctorate from the University of Oxford and a bachelor's degree from the University of Bristol.