A Destination Prediction Model based on Historical Data, Contextual Knowledge and Spatial Conceptual Maps

Apollinaire Nadembega[1]
[1] Univ. of Montreal, Montreal, Canada
nadembea@iro.umontreal.ca

Tarik Taleb[2]
[2] NEC Europe, Heidelberg, Germany
talebtarik@ieee.org

Abdelhakim Hafid[3]
[3] Univ. of Montreal, Montreal, Canada
ahafid@iro.umontreal.ca

Abstract –
Mobile Wireless Network technology has enabled the development of increasingly diverse applications and devices resulting in an exponential growth in usage and services. One challenge in mobility management is the movement prediction. Prediction of the user's longer-term movement (e.g., 10 min in advance) with reasonable accuracy is very important to a broad range of services. To cope with this challenge, this paper proposes a new method to estimate a user's future destination, called Destination Prediction Model (DPM). This method combines two types of approaches: one based on the use of filtered historical movement pattern and another based on contextual knowledge; both approaches use spatial conceptual maps. The filter is based on the day and the time of the day to increase accuracy. The current movement direction, that takes into account the recent data, is used by the proposed method to reduce historical and contextual knowledge mistakes. Simulations are conducted using real-life data to evaluate the performance of the proposed model. For subjects with low predictability degree, DPM reaches an average prediction accuracy of 79%; it reaches 91% for subjects with high predictability and 86% for other subjects. Simulation results also indicate that DPM significantly reduces the impact of learning period and the remaining distance to reach the destination on prediction performance. In the future, we plan to extend our research work by proposing a full Path Prediction Model (PPM) based on the Destination Prediction Model (DPM).

I. INTRODUCTION

Along with ongoing advances in multimedia processing, mobile communications and networking technologies, mobile multimedia streaming services have become extremely popular among an ever-growing community of mobile users. To ensure a good Quality of Experience (QoE) of these mobile services, bandwidth fluctuations, due to users' mobility and during the service course, need to be minimized. Indeed, mobile users frequently change their points of attachment to the network. A user may then experience different data streaming rates due to disparity in the bandwidth availability at the different visited cells along the movement path of the user. Frequent changes in streaming rates, mainly those with high magnitude, may severely impact the perceived QoE.

To ensure acceptable QoE for mobile users, the network needs to ensure a uniform data exchange rate during the entire (or partial) course of a streaming service while a user is on the move. Towards this end, we proposed [1] a framework that integrates user mobility prediction models with resource availability prediction models to keep a constant or less fluctuating streaming rate and to ultimately ensure steady QoE. In the devised framework, an important requirement consists of the fact that the network must have prior exact knowledge on the mobile user's path, along with departure and arrival times at every cell along the movement path of the user. For this purpose, this paper proposes a new method to estimate a user's future destination, called Destination Prediction Model (DPM). This method combines two types of approaches: one based on the use of historical movement pattern and another based on contextual knowledge. Both approaches use spatial conceptual maps. The statistical data extracted from historical movement are filtered; The filter is based on the type of the day (e.g., weekend, week day) and the time of the day to increase accuracy. The current movement direction, that takes into account the recent data, is used by the proposed method to reduce historical and contextual knowledge mistakes. The proposed DPM method builds on the work in [1], wherein mobility prediction is based on contextual knowledge.

The remainder of this paper is organized as follows. Section II introduces related work. Section III presents the proposed method based on historical movement pattern and contextual knowledge, whilst Section IV evaluates its performance. The paper concludes in Section V, with a summary recapping the main advantages and achievements of the proposed method.

II. RELATED WORK

In the following, we present a brief overview on existing mobility modelling methods. According to [2], mobility models should emulate real life mobility in a reasonable way; therefore, they should be associated with a specific place. Mobility models can be classified into two groups: (1) Random-based mobility models; and (2) non-random based mobility models. The former are not realistic and are thus not able to emulate the real-life mobility of users. The latter are more suitable to model the mobility of users. In general, they take into account three facts: temporal dependency (constrains of physical laws; e.g., speed), spatial dependency (constraints of neighbouring nodes), and geographic restriction (constraints of the environment; e.g., highways).

The work in [3] presents three types of movement models. In the Random Walk model the users follow no rule as their movement is completely independent of position, other users or movement history. The Gravity Mobility model assigns values indicating a given level of attractiveness to certain areas. The higher the attractiveness of an area is, the higher the probability that a user will try to reach that area. This model provides a balanced mixture between deterministic and random parts. The Path Following Mobility model gives a sequence of areas to reach or cross during a mobile user's movement. In each area, the gravity model is used. Furthermore, when an area is reached, the gravity model is used to select a new target area. The gravity and path following mobility models are seen more appropriate to model the movement of mobile network users.

In [4-6], different predictors are compared. In [4], different sets of traces using several different predictors such as the Markov O(1), O(2), O(3) and also the LZ Predictor are compared. The results indicate that a larger data structure and higher complexity do not necessarily help in making better predictions. In [5], different generic pattern detection algorithms are compared. These algorithms are Decision Trees (DT), Instance Based (IB) and Support Vector Machines (SVM). DTs are generated using a training set. The trees define a set of rules, whereby based on the values at the different positions inside the movement sequence, a path down to the last node (leaf) in the tree is followed. IBs usually keep the input sequences unprocessed and retrieve the best matching sequence and resulting target base station by calculating the distance to up to *k* stored sequences. SVMs classify the data by finding hyper planes separating the trace data into subsets for target prediction. The results demonstrate that the three algorithms exhibit similar behaviour depending on the maximum length of available path sequences. Additionally, it was demonstrated that a combination of more than one algorithm reduces the overall rate of erroneous predictions.

Most existing mobility models are based on historical data of motion or mobility trace files of the mobile users [7-9]; more specifically, they make use of the historical data to predict, using different schemes, the movements of users. The authors use Hidden Markov Model (HMM), first order of Markov model or second order of Markov model while the authors in [10] use Bayesian network theory to compute the probability of next destination (e.g., next wireless cell). These approaches do not take into consideration the source and the future destination of the current trip. In [11], Kalman filter is used to extract parameters, such as speed and pause time, from real user traces; it is reported that these parameters follow a log-normal distribution and depend on roads and walkways. In [12], authors introduced a hierarchical Markov model that can learn and infer user's daily movements through an urban community. They described a system that creates a probabilistic model of user's daily movement patterns using unsupervised learning from raw GPS data. To achieve efficient inference, they applied Rao–Blackwellized particle filters at multiple levels of the model hierarchy. Significant locations are extracted from user traces by detecting places where GPS signals are lost. These techniques are based on an assumption that the user's movements follow a specific pattern and exhibit some regularity. In this case, a training phase is first required during which regular movement patterns are detected and stored. User's movement behaviour may be highly uncertain and assumptions about user's movement patterns should be made with utmost care. Therefore, whenever the user is located in new locations or when there is a slight change in the user's mobility patterns, the accuracy of the prediction considerably suffers.

Another set of mobility models is based on users' knowledge or their habits [1, 13]. In [1], the Dempster-shafer' theory is applied to the knowledge of user's preferences and goals to predict his/her mobility; the work does not make any assumption about the availability of users' movement history. The authors in [13, 14] applied the social theory to the structure of the relationships among individual users to predict their movements while the authors in [15] defined mobility models based on daily planned activities; they assumed that users move from home to work, from work to restaurant, from restaurant to work, from work to leisure, and return home in the evening.

All these approaches do not take into account the difference between the days of the week and the time of the day to evaluate their estimations. Considering the source and the current direction of the trip is highly important to estimate the future destination. It is possible that a group of potential future destinations can be reached using the same path during a given duration. These destinations will be grouped to form one cluster; this increases the accuracy of destination prediction. Finally, combining the contextual knowledge with historical data reduces the impact of historical movement on prediction accuracy.

III. PROPOSED DESTINATION PREDICTION MODEL

In [16], the authors introduced a framework, shown in Fig. 1, that assists in avoiding frequent changes in the streaming rates of mobile multimedia services and ultimately ensuring acceptable QoE. The proposed DPM approach is implemented at the User's Equipment (UE) Mobility Predictor (MP) as shown in Fig. 1.

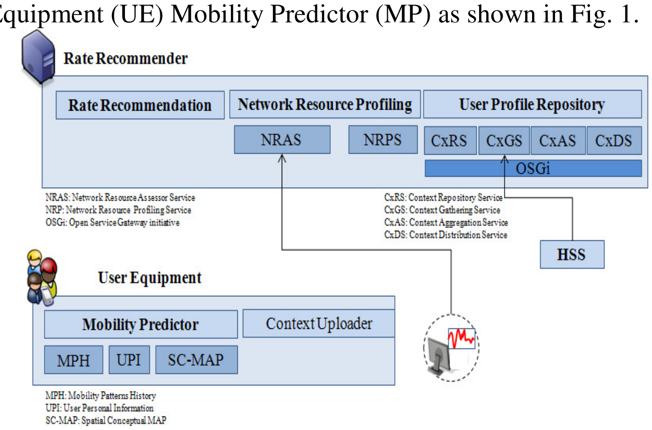

Fig. 1: Key components of RR and mobile terminals (details on the functionality of each unit in the framework are available at [16]).

Note that a UE maintains a database (DB) which records data about user movements, user context and his/her living area. We assume the availability of static data about geographic areas (topology/map of roads), called Navigation Map (NM); the map contains coordinates of road intersections and coordinates of user's Frequently Visited Locations – FVLs – (home, office, shopping mall, etc). We assume that UEs embed a technology, such as GPS, that allows recording the used road segments taking into account day and time. All user possible locations (home, office, road intersection, restaurant, etc) are considered as nodes. Each node is identified by a node ID, latitude and longitude. A road segment is the road portion between two intersections; a road segment consists of edge ID, direction of navigation, node ID 1 and node ID 2. The User Movement Trace (UMT) contains user ID, date, time and node ID that represents user location at date and time. Various algorithms could be used for gathering these data. Using UMT, an algorithm extracts User FVL Trace (UFVLT) that contains User ID, date, arrival time, departure time and node ID.

Algorithm 1 presents a pseudo-code for recording data. This pseudo-code is run by UEs. At each time slot *t*, acceleration *a* and geographic coordinate (*latitude and longitude*) are measured.

Algorithm 1: **Pseudo-code for movement data gathering.**

Input : User_id, NM , a_s , d_s

Output : UMT and UFVLT

Variables :

- ✓ Arrival: is a boolean which marks user arrival in the specific location
- ✓ Departure: is a boolean which marks user departure from the specific location

```
   ✓ Var: is a data set recorder
   ✓ node: is a node
Functions
   ✓ getNode(lat,lon) : it is used to get the node where latitude=lat and
     longitude=lon from DB.
   ✓ existNode(lat,lon): it is used to check whether the node where lati-
     tude=lat and longitude=lon exists in DB.
   ✓ Node(lat,lon) : it is used to create a new node with a generated node_id
     and latitude=lat and longitude=lon
   ✓ getDate(): it is used to get current date
   ✓ getTime(): it is used to get current time

each t sec
    sample a                    // acceleration
    sample (lat, lon)           // location
    if NM.existNode(lat,lon)    // (lat, lon) is into NM
        insert into UMT values (User_id, getDate(),getTime(), NM.getNode(lat,lon))
    if |a| < a_s and Arrival==true
        if UC.FVLC.existNode(lat,lon)// (lat, lon) is into UC.FVLC
            node= new Node(UC.FVLC.getNode(lat,lon))
        else if NM.existNode(lat,lon)   // (lat, lon) is into NM
            node= new Node(NM.getNode(lat,lon))
        else
            node=new Node(lat,lon)
            insert into NM values (User_id,node)
        end if
        insert into UFVLT values (User_id, getDate(),getTime(),NULL,node)
        Departure=True
        Arrival=False
        var(user_id)=user_id
        var(node_id)=node_id
        var(time)=time          // keeping some values
    else if |a| ≥ a_s and Departure==true
        if getTime() – var(time) >= d_s
            update UFVLT set departure_time=getTime()
                         where user_id=var(user_id) and
                         node_id=var(node_id)
        else
            delete from UFVLT where user_id=var(user_id) and node_id=var(node_id)
        end if
        Departure=False
        Arrival=True
    end if
end each
```

When the current position coordinate is a road intersection or a user's FVL, this current position and current timestamp (date and time) are recorded into the user's UMT. If the measured acceleration is smaller than a given threshold a_s, the user is deemed not moving. Thus, if this position is recorded in user's context as FVL, it is recorded with the current date into UFVLT. Time of arrival, recorded at FVL, corresponds to the time when the acceleration falls below a_s.

Departure time from a FVL is recorded when acceleration exceeds a_s. Note that to designate a location visited by a user as FVL, the user needs to reside at the location for a time period longer than a determined duration threshold d_s.

User Contextual (UC) information is gathered and organized in six categories as shown in Table I. The user context database may be built (1) by having users fill in a questionnaire and explicitly express their interests with regard to different places within their living area, and (2) by having users continuously registering both their tasks and scheduled appointments.

Hereunder, we describe how our proposed DPM scheme predicts a destination using the above described User Context database. As stated earlier, in DPM, a destination is a location/node where user resides for a time period beyond a given threshold d_s. To avoid opportunistic positions, this location is extracted from UFVLT according to a frequency of visits f.

Table I: User Contextual (UC) information structure.

Personal Context (PC)	Frequently Visited Location Context (FVLC)	Task Context (TC)	Interest Context (IC)	Calendar Context (CC)	Day Type (DT)
• User ID • Name • Age	• Location (NM_node) • Location name • Preferable day • Earliest preferable time • Latest preferable time • Duration • Characteristics • Importance • Frequency	• Task name • Earliest time • Deadline • Duration • Characteristics • Importance • Frequency	• Interest name • Preferable day • Earliest preferable time • Latest preferable time • Duration • Characteristics • Importance • Frequency	• Location (NM_node) • Date • Time • Characteristics	• From (date) • To (date) • Workday • no workday

A cluster of destinations is a set of destinations that can be reached by a user using the same path during a given travel duration (d). For the purpose of destination clustering, we apply the direction function to select potential targets.

$$o : [0,180] \rightarrow [0,1]$$

$$o(\theta) = 1 - \frac{\theta}{180}$$

Thus, we get the weight of the deviation of current motion (*c*) to each location (*j*) as:

$$o(\theta_j^c) = 1 - \frac{\theta_j^c}{180}$$

We fix a threshold θ_{seuil} and compute $o(\theta_{seuil}) = \varepsilon$. The selected locations (*j*) are those that belong to the following set:

$$\Psi = \bigcup_j \left\{ j \middle| o(\theta_j^c) \geq \varepsilon \right\}$$

Clustering is applied on the set Ψ to take into account the direction of current motion. for better understanding, we use the example shown in Fig. 2. Using the direction of the user's current motion in the figure, we form one cluster C1. But, the addition of the green motion and the red motion produces two distinct clusters, C11 and C12, with smaller sizes compared to C1. Intuitively, the prediction becomes more accurate when the size of clusters decreases.

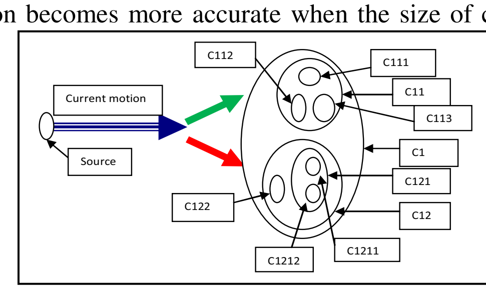

Fig. 2: A destination clustering example.

As stated before, our DPM approach uses historical data and contextual knowledge. The belief function adopted in [1] is used. This function uses contextual information from UC to infer the next location for each formed cluster C_i: $Bel(UC, c_i)$. Generally speaking, using only contextual information is not enough to get a good prediction, hence the application of the belief function on the cluster of destinations to improve the prediction accuracy. Additionally, we evaluate the probability of a user going into each formed cluster C_i taking into account user historical motion. This probability is evaluated based on the current time (t) and the type of the current day (j) (e.g., working day, vacation day, etc) as follows:

$$proba(c_i)_j^t = \frac{frequency(c_i)_j^t}{\sum_{s=1}^{n} frequency(c_s)_j^t}$$

where n denotes the number of formed clusters and $frequency(c_s)_j^t$ denotes the frequency of visits to cluster C_s during the same day type j and the same time t of the day j. To tradeoff between the two techniques, we use a weight α computed as follows:

$$b : [0, N] \rightarrow [0, 1]$$

$$\alpha = b(nbj) = 1 - \frac{nbj}{N}$$

where N and nbj denote the number of days and the number of days used to learn user habit, respectively. The weight of each cluster is evaluated as follows:

$$w(c_i)_j^t = \alpha Bel(UC, c_i) + (1 - \alpha) proba(c_i)_j^t$$

The prediction accuracy is computed as follows:

$$A_d = \frac{n_{bp}}{n_{tp}}$$

where n_{bp} and n_{tp} denote the number of good estimates and the total number of estimates, respectively.

IV. Performance Evaluation

To evaluate the performance (i.e., prediction accuracy) of our proposed DPM scheme, we developed a program in java that uses the historical data and contextual information recordings in the database ; we use the MIT media laboratory's database available from the Reality Mining Project [17]. The subjects from this project consisted of students and staff (94 persons) at a major university during the months between September 2004 and June 2005. The database contains information on the cell tower a user is connecting to and the corresponding time of residence in the cell coverage area.

We run our simulations, using only these subjects who do not reside on campus; the objective is to consider subjects with high level of movements in large geographic areas; this will allow for better evaluation of our proposed approach. To take into account different types of subjects according to the motion predictability, we defined 3 groups (not at all predictable, somewhat predictable and very predictable) and we identified ten subjects (from the database) per group for simulation. Table II shows the values of the parameters used in the simulations.

Table II: simulation parameters.

Parameters	Fixed values	Parameters	Fixed values
d_s	15 minutes	f	1/30 of total days of data collection
θ_{seuil}	45^0	N	30 days
d	10 minutes		

It is worth noting that in some situations DPM may not return a solution (i.e., destination); in this case, in our simulations, we consider that DPM returned "wrong" destination; this has an impact on the accuracy computation of DPM. To evaluate the performance of DPM without taking into account these situations, we include one set of simulations shown in Fig. 4 (for lack of space, we did not include all sets of simulations we run).

In all the scenarios, we used ¾ of the total duration of data collection to learn users' habits (as learning phase) and the destination is predicted using the source as the departure position except where the learning duration and the distance between the source and destination are used as performance metrics.

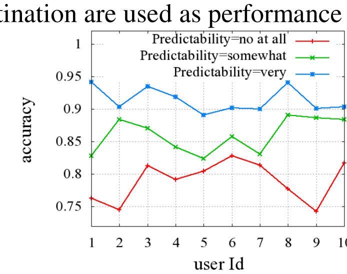

Fig. 3: Accuracy of DPM

The simulation results are shown in Figs. 3-7. Fig. 3 plots the accuracy of our prediction model for different degrees of users' predictability. The graph shows that regardless of the degree of predictability, the accuracy of DPM exceeds 75%. This good performance is attributable to the fact that DPM considers direction to sort out potential clusters for next destination prediction. For subjects with the lowest predictability degree, DPM reaches an average of 79%. It reaches 91% for subjects with high predictability degree and 86% for other subjects.

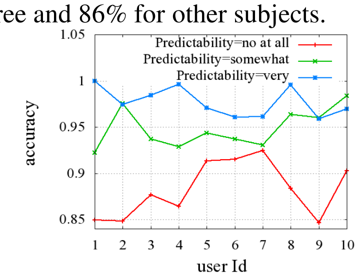

Fig. 4: Accuracy of DPM: "no solution returned" is not considered in accuracy computation

Fig. 4 exhibits the performance of DPM when the case of "no solution returned" is not taken into account. From the figure, it becomes apparent that regardless of the degree of predictability, the accuracy of DPM increases from 75% to 85%. For subjects with highest predictability, performance increases from 91% to 98%. This shows that when our DPM predicts a destination, the accuracy increases. For subjects with highest predictability 1, 4 and 8, whenever the DPM has a prediction was good.

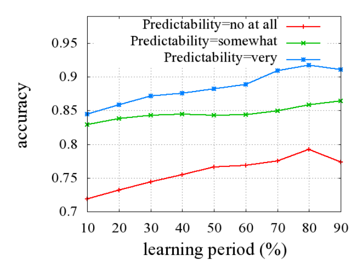

Fig. 5: Accuracy of DPM versus learning phase duration

Fig. 5 shows the impact of the duration of the learning phase duration on the prediction accuracy. In this experiment, the duration of the learning phase is expressed in percentage of the total period of data collection. We observe a small increase (< 7%) in prediction accuracy for the simulated learning phase durations; this shows that the impact of the learning period duration is low. For 10%, the performance of DPM exceeds 72%. This good performance is indeed attributable to the adoption of the contextual knowledge based belief function used in [1] by our proposed DPM.

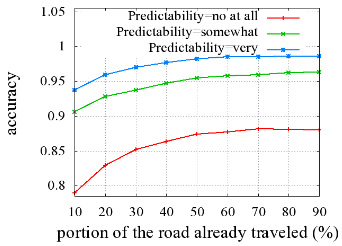

Fig. 6: Accuracy of DPM versus the remaining distance to reach the destination

Fig. 6 highlights the impact of the portion of the road already travelled on the performance of DPM. In this scenario, the destination is inferred using distinct locations as the departure position. These locations represent different percentages of the path from source to destination. Fig. 6 shows that DPM performance is not significantly impacted by the portion of the road already travelled. For subjects with highest predictability, we note a small increase in prediction accuracy (< 4%), due most probably to the usage of the direction function and clustering in DPM. When user is far from source, several destinations become possible candidates. These destination candidates are grouped and the number of potential destinations is reduced along the distance travelled from the source. Thus, direction and clustering minimize the impact of distance between current location and target destination on prediction accuracy.

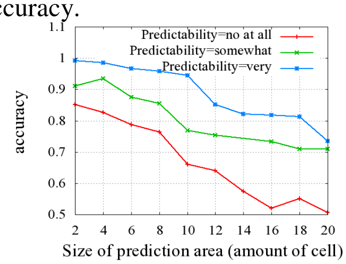

Fig. 7: Accuracy of DPM versus the size of prediction area

Fig. 7 shows the impact of the prediction area size on the performance of DPM; the amount of cells between a source and a destination is used as the evaluation criteria (X-axis).

The performance of DPM decreases rapidly for the subjects with lowest predictability degree. However, it remains bigger than 50% when the distance between source and destination exceeds 20 cells.

V. Concluding Remarks

In this paper, we proposed a new method, called Destination Prediction Model (DPM), to estimate a user's future destination. Unlike existing approaches, the proposed method bases its prediction on both historical movement pattern and knowledge on user's contextual information, using spatial conceptual maps. In addition, DPM filters historical data based on the type of day and the time of the day to evaluate the probability of a location to be the future destination. The current movement direction of a user is used to reduce prediction errors.

The performance of DPM was evaluated using real-life data. The simulation results exhibit the good performance of DPM in predicting the destination. For subjects with low predictability degree, DPM reaches an average prediction accuracy of 79%. It reaches 91% for subjects with high predictability and 86% for other subjects. Simulation results also indicate that DPM significantly reduces the impact of learning period and the remaining distance to reach the destination on prediction performance. In the future, we plan to extend our research work by proposing a full Path Prediction Model (PPM) based on the Destination Prediction Model (DPM).

REFERENCES

[1] N. Samaan and A. Karmouch, "A mobility prediction architecture based on contextual knowledge and spatial conceptual maps," *IEEE Transactions on Mobile Computing,* vol. 4, pp. 537-551, Nov.-Dec. 2005.

[2] F. Bai and A. Helmy, "A survey of mobility models," in *Wireless Ad Hoc and Sensor Networks, Kluwer Academic Publishers*, 2004.

[3] J. Markoulidakis, *et al.*, "Mobility modeling in third-generation mobile telecommunications systems," *IEEE Personal Communications,* vol. 4, pp. 41-56, August 1997.

[4] J. Kim and A. Helmy, "Poster abstract: the challenges of accurate mobility prediction for ultra mobile users," *SIGMOBILE Mob. Comput. Commun. Rev.,* vol. 13, pp. 58-61, July 2009.

[5] S. Michaelis and C. Wietfeld, "Comparison of User Mobility Pattern Prediction Algorithms to increase Handover Trigger Accuracy," in *Proc. of the 63rd IEEE Vehicular Technology Conference (VTC 2006-Spring)*, 2006, pp. 952-956.

[6] I. Butun, *et al.*, "Impact of mobility prediction on the performance of Cognitive Radio networks," in *2010 Wireless Telecommunications Symposium (WTS)*, 2010, pp. 1-5.

[7] H. Wei-Jen, *et al.*, "Modeling Spatial and Temporal Dependencies of User Mobility in Wireless Mobile Networks," *Transactions on Networking, IEEE/ACM,* vol. 17, pp. 1564-1577, Oct. 2009.

[8] W. Wanalertlak, *et al.*, "Behavior-based mobility prediction for seamless handoffs in mobile wireless networks," *Wirel. Netw.,* vol. 17, pp. 645-658, April 2011.

[9] H. Abu-Ghazaleh and A. S. Alfa, "Application of Mobility Prediction in Wireless Networks Using Markov Renewal Theory," *IEEE Transactions on Vehicular Technology,* vol. 59, pp. 788-802, Feb. 2010.

[10] S. Akoush and A. Sameh, "Mobile user movement prediction using bayesian learning for neural networks," in *Proc. of the 2007 international conference on Wireless communications and mobile computing*, 2007, pp. 191-196.

[11] M. Kim, *et al.*, "Extracting a mobility model from real user traces," in *Proc. of the 26th Annual IEEE Conference on Computer Communications (INFOCOM'06)*, 2006, pp. 1–13.

[12] L. Liao, *et al.*, "Learning and inferring transportation routines," *Artificial Intelligence,* vol. 171, pp. 311-331, April 2007.

[13] M. Musolesi and C. Mascolo, "Designing mobility models based on social network theory," *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 11, pp. 59 - 70, July 2007.

[14] D. Yuan, *et al.*, "Experimental analysis of user mobility pattern in mobile social networks," in *Proc. of the 2011 IEEE Wireless Communications and Networking Conference (WCNC)*, 2011, pp. 1086-1090.

[15] F. Ekman, *et al.*, "Working day movement model," in *Proc. of the 1st ACM SIGMOBILE workshop on Mobility models for Networking Research (MobilityModels'08)*, 2008, pp. 33–40.

[16] Tarik Taleb, *et al.*, "Mobility-Aware Streaming Rate Recommendation System," presented at the IEEE Global Communications Conference (GLOBECOM 2011), Houston, Texas, USA, Dec. 2011.

[17] N. Eagle, *et al.*, "Inferring Social Network Structure using Mobile Phone Data," *Proceedings of the National Academy of Sciences (PNAS),* vol. 106, pp. 15274-15278, September 2009.