Mobility Prediction based on Graphical Model Learning

Huijun Li, Gerd Ascheid
Institute for Communication Technologies and Embedded Systems
RWTH Aachen University, Templergraben 55, 52056 Aachen, Germany
Email: {huijun.li, gerd.ascheid}@iss.rwth-aachen.de

***Abstract*—Existing mobility prediction algorithms focus on predicting the next cell or interesting regions such as a home zone. But for position- and movement-based optimization of transmission in a cell such coarse-level mobility prediction is not sufficient. In this paper a learning-based graphical model is introduced which allows a fine-level prediction of the movements and velocities of mobile users inside a cell. We divide the mobile users into different user groups by velocities and learn the path patterns and user type transitional probabilities. Based on this a-priori information a three-step mobility prediction algorithm considering positioning error and future user type is proposed. The simulation result shows a better level of prediction accuracy compared to previous methods.**

I. Introduction

Location-based services (LBS) are playing an important role in modern wireless communication networks. Examples of LBS include emergency calls, navigation system and traffic management [1]. Another useful application is the management of channel resources. Mobility prediction can not only help optimize channel resources and beamforming patterns in advance, but also predict optimal handover instances based on historical records, thus, improving or guaranteeing QoS.

Mobile users do not move randomly. They follow habits and are subject to constraints like e.g. the road topology. These repeatable trajectory patterns can be learned from historical records. Several mobility prediction algorithms have been presented in the literature. In [2] a standard Markov chain model was used to predict the next location. Markov Renewal Processes were proposed in [3] for computing the likelihood of the next-cell transition. But the framework is limited to making predictions and estimations for single-step transitions, as it is very reasonable when studying cell transitions. The work in [4] proposed a hierarchical approach by combining historical trajectory data and current movement to predict the next cell-crossing and subsequent cell sequence. In [5] the road topology information was integrated into the prediction algorithm under the assumption of perfect positioning and non-updated digital maps. However, all previous work was only intended to predict the next cell or interesting region (e.g. hospitals, shops, companies, etc.) of the mobile terminal. Mobility prediction at a fine level inside one cell is still an open problem but is a key precondition for intra-cell cognitive resource allocation. It is the aim of this paper.

Although users move with certain habits, different types of users may behave quite differently. For instance, trains always go along the railways. Cars move along the roads but pedestrians may move more randomly. So it is more effective to learn the trajetory pattern for each type of user separately. The change of user groups during one journey must also be considered since it is quite common in real life. That a passenger goes to a bus station on foot and then gets on the bus is a typical example of such case.

In this paper we introduce a learning-based graphical mobility model to do mobility prediction. The average velocities of users are modeled as Gaussian mixture distribution and mobile users are divided into user groups based on parameters learned by means of expectation maximization (EM). All the history trajectories are abstracted as a graphical map with paths and nodes. Paths indicate the road topology and nodes represent crossings of the roads or changing points of user groups. This makes the prediction smarter and efficient. We predict the future path on one hand, on the other hand we also predict the user type. User type determines the average velocity of the mobile user for the next path, so we can calculate user position sequences for the near future in order to implement optimization algorithm related to locations in advance. Our algorithm is studied by the performance of predicting the users' movements under a predefined environment and is also applied to enable long-term window scheduling [6].

The rest of this paper is organized as follows. In section II the graphical mobility model learning is described. Section III presents a mobility prediction method, which is made up of three steps. Simulation results are shown in section IV. Finally a conclusion is given in section V.

II. Graphical Mobility Model Learning

The graphical mobility model is made up of a series of nodes $\mathcal{M} = \{m_1, m_2, ...\}$ and paths $\mathcal{P} = \{p_1, p_2, ...\}$. The nodes represent the crossings of the roads or the changes of user types. The paths stand for geographical trajectories. The graphical path structure is extracted from the history trajectory database, which can be obtained by pattern a clustering method [7]. This process can also be initialized

978-1-4673-1881-5/12/$31.00 ©2012 IEEE

with a priori geographical maps and updated with new trajectory records regularly. Fig. 1 shows an example of a geographical map contour with the range of 1km × 1km and Fig. 2 is the corresponding graphical structure. The predictions are made at the nodes. The proposed graphical model learning contains two parts: user type classification and transitional probability of future path and user type at the nodes.

A. User Type Classification

In order to divide mobile users into user groups, we need a model to describe the main characteristics of different types of users and learn the model parameters.

We assume that there are K types of users corresponding to the graphical structure. Generally speaking, users with similar velocities have similar behaviors and path sequences. So the users are classified into K groups and distinguished by their velocities. If we model the velocity of each user type as a Gaussian distribution, the velocity of all user types has a Gaussian mixture distribution. We assume that the Gaussian distribution for the velocity of user group k has the mean μ_k and variance σ_k^2. Suppose that the users do not change their types between two adjacent nodes m_i and m_{i+1}. If v denotes an average velocity along one path, we have the Gaussian distribution for user type k

$$P(v|\mu_k,\sigma_k) = \frac{1}{\sqrt{2\pi}\sigma_k}\exp(-\frac{(v-\mu_k)^2}{2\sigma_k^2}).$$

The probability of the velocity of a user is written as:

$$P(v) = \sum_{k=1}^{K} P(k)P(v|\mu_k,\sigma_k) \tag{1}$$

where $P(k)$ is the probability of a user belonging to user group k from the total K user groups. The Gaussian mixture model (GMM) parameters $P(k)$, μ_k and σ_k in (1) need to learned from an available velocity set $\boldsymbol{v} = \{v_1, v_2, ...v_N\}$.

A powerful method to estimate the GMM parameters is the EM algorithm, which maximizes the log-likelihood function of the observed velocity set. If we assume that the data are drawn independently from the distribution, the objective of EM is expressed as:

$$\begin{aligned}[\mu,\sigma,P] &= \underset{\mu_k,\sigma_k,P(k)}{\arg\max}\ lnP(\boldsymbol{v}|\mu,\sigma) \\ &= \underset{\mu_k,\sigma_k,P(k)}{\arg\max} \sum_{n=1}^{N} ln \sum_{k=1}^{K} P(k)P(v_n|\mu_k,\sigma_k)\end{aligned} \tag{2}$$

where μ,σ,P are whole sets of $\mu_k,\sigma_k,P(k)$ for all k separately.

The EM algorithm achieves a local maximum of the observed data likelihood function, depending on the initial parameters. It iteratively performs an Expectation (E) step which computes the expectation of the likelihood function using the current parameters and a Maximization (M) step which calculates the parameters maximizing the likelihood function from the E step. Using a similar process as in

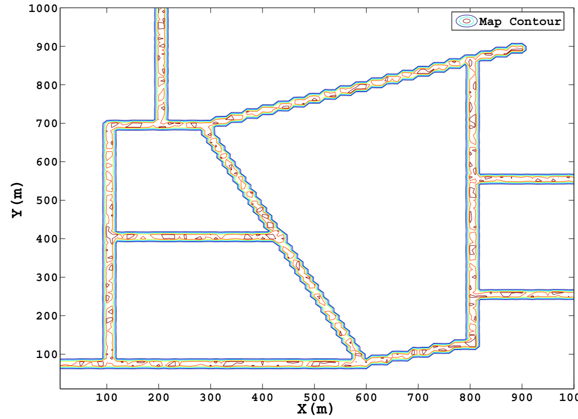

Fig. 1. Example of geographical map contour of 1km × 1km.

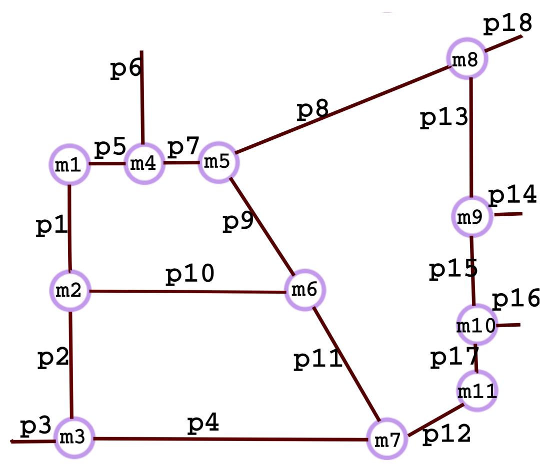

Fig. 2. The graphical structure of trajectories according to Fig. 1.

[8], the EM algorithm based on (2) for our model can be summarized as follows:

1: Initialize the model parameters μ_k, σ_k and $P(k)$, $k = \{1, 2, ..., K\}$.

2: **E step** Compute the posteriori probabilities using current parameters for all k and n where $n = \{1, 2, ..., N\}$

$$P(k|v_n) = \frac{P(k)P(v_n|\mu_k,\sigma_k)}{\sum_{l=1}^{K} P(l)P(v_n|\mu_l,\sigma_l)}. \tag{3}$$

3: **M step** Recompute the parameters using the current probabilities from the E step

$$\mu_k = \frac{1}{N_k}\sum_{n=1}^{N} P(k|v_n)v_n$$

$$\sigma_k^2 = \frac{1}{N_k}\sum_{n=1}^{N} P(k|v_n)(v_n-\mu_k)^2$$

$$P(k) = \frac{N_k}{N}$$

where $N_k = \sum_{n=1}^{N} P(k|v_n)$.

4: Evaluate the log-likelihood function (2). Go to step 2 if convergence is not achieved.

B. Transitional Probability at the nodes

Having a user group classification we can learn the trajectory patterns for each user type separately at the nodes. We assume mobile users can change their user types at nodes. Because users tend to repeat their behavior, the decision of future path and future user type is influenced by the most recent paths and user types. We record the corresponding probabilities in a graphical way.

We describe the most recent L paths and the corresponding user types at every node as $\boldsymbol{P_{past}}$ and $\boldsymbol{K_{past}}$. L is the order of the probabilistic model. The probability of the next path and user type can be expressed as:

$$P(k^i_{next}, p^d_{next} | \boldsymbol{P_{past}}, \boldsymbol{K_{past}}) = \frac{N(k^i_{next}, p^d_{next}, \boldsymbol{P_{past}}, \boldsymbol{K_{past}})}{\sum_{d=1}^{D}\sum_{k=1}^{K} N(k^i_{next}, p^d_{next}, \boldsymbol{P_{past}}, \boldsymbol{K_{past}})} \quad (4)$$

where k^i_{next} and p^d_{next} are the next possible user type i and path d. K and D are the total number of user groups and next possible paths. $N(\cdot)$ indicates the number of paths and user types pair in the history records.

To make it more clear we give a simple example. Fig. 3 illustrates the 1st-order probabilistic graphical model at node m_9 with the direction from path p_{15} to p_{13} and p_{14}. The numbers written in three colors indicate the numbers of the three user types on the paths. It can be seen that 100 mobile users of uses type 1 arrive via path p_{15}, 20 of them continuing via path p_{13} and 70 via path p_{14}. The remaining 10 convert to type 2 continuing via path p_{14} (that is 10 of the $(60+10)$ written in black). Similarly there are 200 users of type 2 travelling on path p_{15}, 120 of which move to path p_{13} and 60 of which move to path p_{14}. The remaining 20 change their user type to type 1 moving towards path p_{14}. From the 60 type 3 users arriving via path p_{15} 40 users continue via path p_{13} and 20 users continue via path p_{14}. We give two example probabilities at node m_9 based on (4) below:

$$P(k_1, p_{13}|p_{15}, k_1) = \frac{20}{100}, \quad P(k_2, p_{14}|p_{15}, k_1) = \frac{10}{100}.$$

III. Mobility Prediction

The mobility prediction is made based on the measurement of past movement, the accuracy of which depends on the used positioning technique. So we propose a three-step mobility prediction algorithm considering positioning errors. First, the position estimation of the passing trajectory is improved by making use of the graphical structure as a priori information. Second, the scattered position measurement points are filtered by a Kalman interpolation filter to form a past trajectory. In the last step the past user type are identified and the future path and user type is predicted with the transitional probabilities.

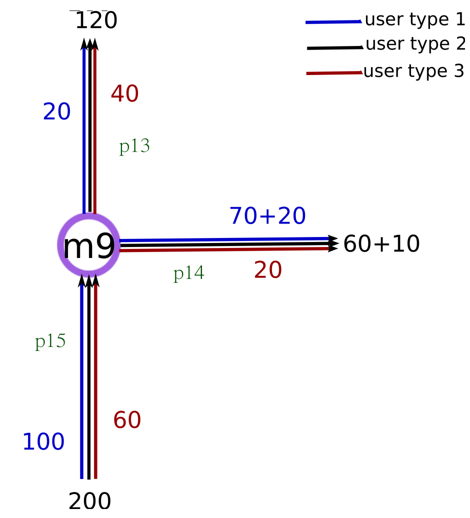

Fig. 3. The probabilistic graphical model at node m_{19} with incoming path p_{15} and outgoing paths p_{13} and p_{14}.

A. Maximum A Posteriori (MAP) Position Estimation

Positioning technology estimates the position of the user but the error can not be ignored. The measurements are probable not along the roads compared with the corresponding geographical map. So we propose a MAP position estimation method to reduce the positioning error with a priori knowledge of the graphical trajectory map.

Let r be the position estimate of the real position variable θ. The a posteriori probability distribution function (p.d.f.) of the real position variable θ is given by the Bayes rule [9]:

$$f(\theta|r) = \frac{f(r|\theta)g(\theta)}{\int f(r|\theta')g(\theta')d\theta'}$$

where $f(r|\theta)$ is the measurement p.d.f. of the position θ. Since the position estimation is normally not accurate, $f(r|\theta)$ may be modeled as Gaussian distribution $\mathcal{N}(\theta, \sigma^2)$. The position error modeling in [10] shows the standard deviations for GPS σ_{GPS} and OTDOA σ_{OTDOA} are 6.7m and 31.4m based on the direct path to the satellite orbits for a specific date and time for certain scenario. $g(\theta)$ is the a priori p.d.f. of the real position θ which can be obtained from the graphical trajectory map as road information. If the width of the road is W, The a priori information can be modeled as a uniform distribution inside the road boundary with a tolerance ϵ:

$$g(\theta) = \begin{cases} \frac{1}{W+2\epsilon} & \text{if } \theta \in [-\frac{W}{2} - \epsilon, \frac{W}{2} + \epsilon] \\ 0 & \text{others} \end{cases}.$$

MAP estimation of position θ is:

$$\hat{\theta}_{MAP}(x) = \arg\max_{\theta} \frac{f(r|\theta)g(\theta)}{\int f(r|\theta')g(\theta')d\theta'} = \arg\max_{\theta} f(r|\theta)g(\theta).$$

B. Kalman Interpolation

After MAP position estimation, a sequence of position measurements with reduced errors are acquired. But they are scattered near the roads and the sequence itself cannot form a meaningful past trajectory, so a Kalman interpolation filter is used subsequently.

To track the mobility of a moving vehicle we need the location and the velocity of it. In an orthogonal coordinate system, the location s is determined by the horizontal and vertical coordinates of the position x and y. The velocity is the differential coefficient of the location parameters $(\dot{x}, \dot{y})$. Suppose that the users' locations and velocities are measured every Δt seconds and the accelerations of both directions $\ddot{x}$ and $\ddot{y}$ are constant for Δt.

The movement model is described as following:

The state vector consists of the position and velocity of a user:

$$\boldsymbol{v_k} = [x_k \ \dot{x_k} \ y_k \ \dot{y_k}]^T.$$

From Newton's laws of motion we can evolve the true state at time k from the state at $k-1$:

$$\boldsymbol{s_k} = \boldsymbol{F}\boldsymbol{s_{k-1}} + \boldsymbol{G}\boldsymbol{a_k} \tag{5}$$

where

$$\boldsymbol{F} = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{G} = \begin{bmatrix} \frac{\Delta t^2}{2} & 0 \\ \Delta t & 0 \\ 0 & \frac{\Delta t^2}{2} \\ 0 & \Delta t \end{bmatrix}$$

and $\boldsymbol{a_k}$ is the acceleration vector

$$\boldsymbol{a_k} = \left[\ddot{x_k} \ \ddot{y_k}\right]^T$$

which can be modeled as Gaussian distribution $\boldsymbol{a_k} \sim \mathcal{N}(0, \Sigma)$ with a priori knowledge:

$$\Sigma = cov(\boldsymbol{a_k}) = \begin{bmatrix} \sigma^2_{a_x} & 0 \\ 0 & \sigma^2_{a_y} \end{bmatrix}.$$

Now the state transition function in (5) can be written as:

$$\boldsymbol{s_k} = \boldsymbol{F}\boldsymbol{s_{k-1}} + \boldsymbol{W_k}$$

where $\boldsymbol{W_k} = \boldsymbol{G}\boldsymbol{a_k} \sim \mathcal{N}(0, \boldsymbol{Q_k})$,

$$\boldsymbol{Q_k} = E[\boldsymbol{G}\boldsymbol{a_k}\boldsymbol{a_k}^T\boldsymbol{G}^T] = \boldsymbol{G}E[\boldsymbol{a_k}\boldsymbol{a_k}^T]\boldsymbol{G} = \boldsymbol{G}\Sigma\boldsymbol{G}^T.$$

At time step k, the measurement of the users' real state vectors can be modeled as:

$$\boldsymbol{z_k} = \boldsymbol{H}\boldsymbol{s_k} + \boldsymbol{v_k}$$

where $\boldsymbol{H}$ is a four-dimensional identity matrix and $\boldsymbol{v_k}$ is measure noise vector

$$\boldsymbol{v_k} = \left[v_{k1} \ v_{k2} \ v_{k3} \ v_{k4}\right]^T$$

where v_{k1}, v_{k2}, v_{k3}, v_{k4} are independent from each other with zero mean and variance σ^2_{k1}, σ^2_{k2}, σ^2_{k3} and σ^2_{k4}. So the covariance will be:

$$\boldsymbol{R_k} = E[\boldsymbol{v_k}\boldsymbol{v_k}^T] = diag(\sigma^2_{k1}, \sigma^2_{k2}, \sigma^2_{k3}, \sigma^2_{k4}).$$

After Kalman interpolation filtering [11] the sequence of position measurements $\boldsymbol{s_{past}}$ are obtained from the updated state estimates. So $\boldsymbol{s_{past}} = [\hat{\boldsymbol{s}}_{\boldsymbol{k_1|k_1}}, \hat{\boldsymbol{s}}_{\boldsymbol{k_2|k_2}}, ...\hat{\boldsymbol{s}}_{\boldsymbol{k_M|k_M}}]$, where M is the number of past velocity samples.

C. Mobility Prediction

In this section the prediction is made at every node. First, we map the filtered sequence $\boldsymbol{s_{past}}$ to the path and user type pattern of the graphical mobility model, in order to match the transitional probabilities stored at the nodes. According to the trajectory map, the filtered recent sequence $\boldsymbol{s_{past}}$ can be written as two vectors $\boldsymbol{p_{past}} = [p_{past1}, p_{past2}, ..., p_{pastL}]$ and $\boldsymbol{v_{past}} = [v_{past1}, v_{past2}, ..., v_{pastL}]$, where L is the order of the model. The former comprises the recent L path sequences and the latter comprises the recent average velocity samples along the L paths.

Second, we determine the user types for the recent L paths from the estimated GMM parameters. The user type k_{pasti} for the velocity sample v_{pasti} on the path p_{pasti} is identified by maximizing the posteriori probability function (3)

$$\begin{aligned} k_{pasti} &= \max_k P(k|v_{pasti}) \\ &= \max_k \frac{P(k)P(v_{pasti}|\mu_k, \sigma_k)}{\sum_{l=1}^{K} P(l)P(v_{pasti}|\mu_l, \sigma_l)}. \end{aligned} \tag{6}$$

So the user type sequence for the recent paths $\boldsymbol{k_{past}} = [k_{past1}, k_{past2}, ..., k_{pastL}]$ is identified from $\boldsymbol{v_{past}}$.

Next we search the recorded history data $\boldsymbol{P_{past}}$ and $\boldsymbol{K_{past}}$ at the node to find the matched pattern for current $\boldsymbol{p_{past}}$ and $\boldsymbol{k_{past}}$. Then the probabilities of next path and user type are computed with (4). The decision is made by choosing the future user type and path pair $[i, d]$ to maximize the probability function

$$[i, d] = \max_{i,d} P(k^i_{next}, p^d_{next}|\boldsymbol{p_{past}}, \boldsymbol{k_{past}}).$$

Then the future position sequence $\boldsymbol{r_{next}}$ is calculated along the chosen path d with the mean velocity of user type i.

IV. Simulation Results

In this section we present the simulation results of the proposed model. The generated geographical map with an area of 1km × 1km in Fig. 1 is available at the base station and used for our implementation. The graphical structure is shown in Fig. 2. We assume there are three user types. The three Gaussian mixture velocity distributions have the means 5, 15, 30 (m/s) and standard deviations 4, 8, 5 (m/s) separately. The probabilities of a user belonging to the three groups are 0.56, 0.28 and 0.16. The probabilities at the nodes are predefined.

Mobility prediction contains three steps. The purpose of the first two steps is to reduce the positioning error which are important for the third step to find the correct trajectory patterns. Table I compares the mean error of raw positioning, after MAP position estimation and after Kalman interpolation for GPS and OTDOA. It can be seen that the

TABLE I
MEAN ERROR OF TRAJECTORY ESTIMATION PROCEDURE FOR DIFFERENT POSITIONING METHODS

mean error	raw positioning	after MAP	after Kalman
GPS	8.4m	7.2m	1.6m
OTDOA	39.2m	37.8m	8.1m

positioning errors are significantly reduced which guarantees finding the correct past paths.

We compare our model with two other algorithms for next-cell prediction which are applied in our graphical model. The ignorant prediction algorithm of [12] randomly picks one of the next paths as the result, regardless of history trajectory records. Transition Matrix (TM) [13] uses a path transition matrix which is calculated from the trajectory history without user grouping to predict the next path. Fig. 4 compares the prediction accuracies. With the user group classification our proposed model has a significant higher precision because the same group users behave similarly. Higher order of the probabilistic model has a better performance since the statistics in Eq. (4) are more reliable. The correct prediction probability depends on the considered scenario. Among the correctly predicted paths some are given the wrong user type, which causes a reduced accuracy of the predicted position sequence r_{next} for these users. Table II below shows the probability of right user type prediction based on the predefined statistics in our simulator.

TABLE II
RATE OF CORRECT USER TYPE PREDICTION

model order	1	2	3
probability	0.69	0.61	0.85

V. CONCLUSION

This paper introduces a graphical mobiltiy model and presents a mobility prediction algorithm based on the model learning. The proposed model utilizes the road topology or the clustered history trajectories to form an intuitive graphical structure. GMM is used to characterize the velocities of mobile users. The users are grouped with the learned GMM parameters, which is more effective than estimating mobility patterns without distinguishing user types. The probabilistic model at the nodes is based on the history trajectory records as a priori information and is updated regularly. The three-step mobility prediction algorithm is tested in a predefined environment and shows an improved accuracy of prediction compared with other algorithms and proves the advantage of our prediction approach. The gain depends on the specific scenario and the training data. The proposed predictor will be applied to future-position related optimization in our future research work.

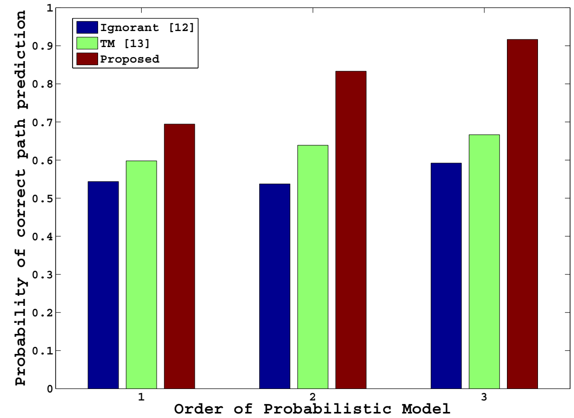

Fig. 4. Probability comparison of correct path prediction for three different methods over varying probabilistic model orders based on the predefined scenario.

REFERENCES

[1] S. Wang, J. Min, and B. Yi, "Location based services for mobiles: Technologies and standards," *IEEE International Conference on Communications (ICC), Beijing*, 2008.

[2] G. Liu and G. Maguire, "A class of mobile motion prediction algorithms for wireless mobile computing and communication," *Mob. Netw. Appl*, pp. 113–121, Oct. 1996.

[3] H. Abu-Ghazaleh and A. S. Alfa, "Mobility prediction and spatial-temporal traffic estimation in wireless networks," *Proc. IEEE 67th VTC-Spring*, pp. 2203–2207, May 2008.

[4] T. Liu, P. Bahl, and I. Chlamtac, "Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 6, pp. 922–936, Agu. 1998.

[5] W.-S. Soh and H. S. Kim, "Dynamic bandwidth reservation in cellular networks using road topology based mobility predictions," *INFOCOM*, March 2004.

[6] H. Li and G. Ascheid, "Long-term window scheduling in multiuser OFDM systems based on large scale fading map," *Submitted*.

[7] D. G. S. Richard O. Duda, Peter E. Hart, *Pattern Classification*, 2nd ed. New York, Wiley, 2001.

[8] C. M. Bishop, *Pattern Recognition and Maschine Learning*. Springer, 2006.

[9] W. M. Bolstad, *Introduction to Bayesian Statistics*, 1st ed. Wiley Interscience, 2007.

[10] M. Neuland, T. Kuener, and M. Amirijoo, "Influence of positioning error on X-map estimation in LTE," *Proc. 73rd IEEE Vehicular Technology Conference (VTC 2011-Spring)*, 2011.

[11] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, pp. 35–45, 1960.

[12] G. Yavas, D. Katsaros, O. Ulusoy, and Y. Manolopoulos, "A data mining approach for location prediction in mobile environments," *Data and Knowledge Engineering 54 (2005)*, pp. 121–146.

[13] S. Rajagopal, R. Srinivasan, R. Narayan, and X. Petit, "GPS-based predictive resource allocation in cellular networks," *IEEE ICON 02*, pp. 229–234, 2002.