Markov model based location prediction in wireless cellular networks

TAMÁS SZÁLKA, SÁNDOR SZABÓ, PÉTER FÜLÖP

Budapest University of Technology and Economics, Department of Telecommunications
{szalkat, szabos, fepti}@mcl.hu

Keywords: mobility modelling, Markov model, location prediction, cellular network

The efficient dimensioning of cellular wireless access networks depends highly on the accuracy of the underlying mathematical models of user distribution and traffic estimations. Mobility prediction also considered as an effective method contributing to the accuracy of IP multicast based multimedia transmissions, and ad hoc routing algorithms. In this paper we focus on the trade-off between the accuracy and the complexity of the mathematical models used to describe user movements in the network. We propose Markovian mobility models, in order to utilize the additional information present in the mobile user's movement history thus providing more accurate results than other widely used models in the literature. The new models are applicable in real-life scenarios, because these rely on additional information effectively available in cellular networks (e.g. handover history), too. The complexity of the proposed models is analyzed and the accuracy is justified by means of simulation.

1. Introduction

Different mobility models have been proposed in the literature to cope with user mobility in different wireless and mobile networks (e.g. cellular networks, ad hoc networks, etc.). Mobility model based prediction provides useful input for dimensioning and planning of wireless mobile networks, ad hoc routing algorithms, efficient multicast transmission and call admission control [3,8].

One of the well known mobility models is *Random Walk Mobility* model, which is often used in network planning and in analyzing network algorithms, because of its simplicity [1].

In the Random Walk Mobility model [4] the node moves from its current location to a new location by randomly choosing a direction and a speed. The Random Walk model defines user movement from one position to the next one with memoryless, randomly selected speed and direction. Each movement in the Random Walk Mobility Model occurs in either a constant time interval *t* or in a constant distance travelled *d*. When a mobile node reaches a simulation boundary, it simply bounces off the simulation border with an angle determined by the incoming direction, then the node continues moving along this new path. As we mentioned, this is very easy to use, but on the other hand this very simple model presumes unrealistic conditions, like uniform user distribution in the mobile network.

To handle different levels of randomness, one can use the *Gauss-Markov Mobility Model*. In this model initially each node is assigned a current speed and direction, and at fixed intervals of time, the speed and direction of the nodes are updated. The value of speed and direction at any time instance are calculated based upon the value of speed and direction at the previous instance and a random variable [5].

However, in real-life networks, geographical characteristics such as streets and parks influence the cell residence time (dwell time) and movement directions of users in the network, and result in a non-uniform user density. While these models are appropriate for mathematical analysis, easy to use in simulations and for trace-generation, they fail to capture important characteristics of mobility patterns in specific environments, e.g. time variance, location dependence, unique speed and dwell-time distributions [2].

In order to follow user movement patterns more efficient, City Section Mobility, or Mobility Vector model can be used. In the *City Section Mobility Model* a section of a city is represented where an ad hoc network exists [6]. The streets and speed limits on the streets are based on the type of city being simulated. The streets might form a grid in the downtown area of the city with a high-speed highway near the border of the simulation area to represent a loop around the city.

If a flexible mobility framework for hybrid motion patterns is needed, one can rely on the *Mobility Vector* [7] model. A mobility vector expresses the mobility of a node as the sum of two sub vectors: the Base Vector and the Deviation vector. The base vector defines the major direction and speed of the node while the deviation vector stores the mobility deviation from the base vector. The mobility vector is expressed as an acceleration factor in different directions.

Beside the mobility models many works have discussed prediction algorithms, too. The prediction serves as an input for an optimal resource planning.

The *shadow cluster scheme* [9] estimates future resource requirements in a collection of cells in which a mobile is likely to visit in the future (as a "shadow" of the user). The shadow cluster model makes its prediction based on the mobile's previous routes. In this model, the highway traffic with various constant speeds is simulated and users travel in forward and backward directions. The shadow cluster model improves estimation of resources and decision of call admission.

User movements in a cellular network can be described as a time-series of radio cells the user visited. The handover event of active connections (e.g. cell boundary crossing) is recorded in the network management system's logs, thus the information can be extracted from the management system of cellular mobile networks, such as GSM/GPRS/UMTS networks. The user movements are described by the dwell-time and outgoing probabilities (the probability of a user leaving for each neighbouring cell, called the handover vector). These parameters can be calculated for each cell based on the time-series of visited cells of the users. However, in some cases, these parameters – dwell-time and outgoing probabilities – are not enough to capture all the information present in the time-series of user movements. In many situations the movement patterns can be estimated more precisely if the model also considers the conditional probabilities between the incoming and outgoing directions. In this paper we investigate the effect of incorporating this additional information into the mobility models on the accuracy of the models.

The results of this paper are applicable in engineering tasks of network dimensioning, can provide input for more effective Call Admission Control algorithms in order to ensure user's satisfaction and optimal resource usage in cellular wireless mobile networks [3].

2. Cell-centralized Markov mobility model

2.1 Motivation

The memoryless property of the Markov process makes the Markov models in the literature easily applicable. In many models, the Markov processes states are based simply on the physical radio cells, i.e. one state represents one radio cell. In this case, any potentially present additional information in the user movements cannot be included in the model. We propose a model, in which the states of the model are constructed according to a group of cells belonging to typical movement directions.

Our model takes into account the user history during the prediction and merges the neighbor cells into a direction group depending on the user behavior and historical distribution and patterns, yet retaining the memoryless property.

2.2 The model enhancement

In our method the direction of a user is identically distributed between 0 and 2π. The user's speed is between 0 and V_{max}. After moving in a direction with a randomly chosen speed for a given Δt time, the user changes its direction and speed. N_{cell} denotes the number of cells in the network.

A possible classification of cells can be seen in *Figure 1*. Using this case a user can be located in three different states during each time slot in simple Markov-chain based model, the stay state *(S)* and the left-area state *(L)* and the right-area state *(R)*.

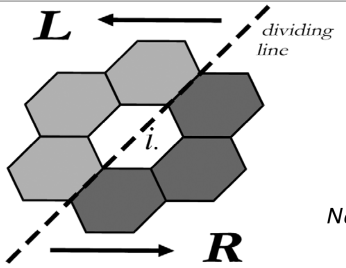


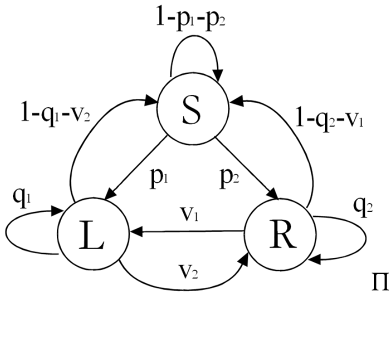
Figure 1. Neighbor cells separated into two groups

The grouping can be derived from the user behavior. If the users in right-hand side cells behave similarly from the current cell's point of view, the neighboring cells will be merged into a common cell group, which represents a state in the Markov model (*R* state). Other grouping methods can be used as well, i.e. a stand-alone cell can constitutes a group also. In our example model each of the two groups (*R* and *L*) contains three cells.

Let us define the random variable *X(t)*, which represents the movement state of a given terminal during time slot *t*. We assume that $\{X(t), t=0,1,2,...\}$ is a Markov-chain with transition probabilities *p, q, v*.

If the user is in state *S* of Markov model for cell *i (current cell)*, it remains in the given cell. If the user is in state *R*, it is in range of the cells on the right-hand side, if in state *L*, it is in the left-hand side of the dividing line.

Since the transition properties are not symmetric, the left-area state and the right-area state have different probabilities. *Figure 2* depicts the Markov-chain and transition (Π) matrix.



$$\Pi = \begin{bmatrix} 1-p_1-p_2 & p_1 & p_2 \\ 1-q_1-v_2 & q_1 & v_2 \\ 1-q_2-v_1 & v_1 & q_2 \end{bmatrix}$$

Figure 2. State diagram and Π matrix of 3-state M-model (M_3)

As we mentioned before, our aim is to use the information present in the previous steps of the users for motion prediction. In this case, the transition probabilities are not handover intensities, but contain more information, specifically:

p – Probability of the event that the user stays for a timeslot in cell *i*, and in the next timeslot it moves one of the cell groups. According to the mentioned model (Fig. 2), p_1 means that the user moves in the next timeslot into one of the right (*R* state) neighbor cells.

q – Probability of the event that the user came into cell *i* from one of the cell groups, and in the next timeslot it moves into the same cell group where it comes from. Thus q_1 means that the user comes from the right-hand side neighbor cells *(R)* and moves into the same, *R* group (but not necessarily into the same cell of the cell group).

v – Probability of the event that the user comes into cell *i* from one of the cell groups, and in the next timeslot it moves into another cell group. According to the above mentioned model (see Fig. 2), v_1 means that the user comes from left neighbor cells *(L)* and moves into the right group *(R)*.

The probabilities introduced above can be calculated from real network traces.

How can we use this model?

How can we predict the user's next steps?

We assume that mobile users are in cell *i* at the beginning of a timeslot *t*. With each *i* cell a previously introduced Markov model is associated, which handles the users movement in this cell. The network operator's log contains the information where the users from cell *i* were in timeslot *t-1* (according to the assumption that they were in one of the neighbor cells). Using this information as *initial distribution P(0)*, we calculate *P(1)* with the transition matrix of the model: $P(1) = P(0) \cdot \Pi$.

P(1) shows the predicted users distribution for the *t+1* timeslot, in other way we predicted where the users will be from cell *i in timeslot t+1*.

The number of users in cell *i* at timeslot *t+1*, $N_i(t+1)$, is given by (1).

$$N_i(t+1) = N_i(t)\cdot P_S{}^i(1) + \frac{1}{3}\sum_{l\in S^i_{adj(L)}} N_l(t)\cdot P_R{}^l(1) + \frac{1}{3}\sum_{r\in S^i_{adj(R)}} N_r(t)\cdot P_L{}^r(1) \qquad (1)$$

where $S^i_{adj(R)}$ means the set of cell indexes from the right-hand side cells of cell *i*, $S^i_{adj(L)}$ means the set of cell indexes from the left-hand side cells of cell *i*.

The steady state probabilities of the Markov model can represent a steady user distribution if the network parameters do not vary.

The balance equations for this Markov-chain are given in Eq. (2).

$$\begin{aligned} P_S\cdot(p_1+p_2) &= P_L\cdot(1-q_1-v_2) + P_R\cdot(1-q_2-v_1) \\ P_L\cdot(1-q_1) &= P_S\cdot p_1 + P_R\cdot v_1 \\ P_R\cdot(1-q_2) &= P_S\cdot p_2 + P_L\cdot v_2 \end{aligned} \qquad (2)$$

We also know that $P_S+P_L+P_R=1$, thus the steady state probabilities can be calculated.

Knowing the result we can predict the number of mobile terminals for time slot *t+1* in a steady state:

$$N_i(t+1) = N_i(t)\cdot P_S{}^i + \frac{1}{3}\sum_{l\in S^i_{adj(L)}} N_l(t)\cdot P_R{}^l + \frac{1}{3}\sum_{r\in S^i_{adj(R)}} N_r(t)\cdot P_L{}^r \qquad (3)$$

As we mentioned earlier, this model is based on possible cell grouping (*R* and *L* states) and performs well when the user's movement has only one typical direction, because in this case the handover intensities of the right-move (or the left-move) cells do not differ significantly.

If we try to predict the user's distribution in a city having irregular, dense road system, or in big parks, then the handover intensities could differ. From this point of view the best way is to represent all of the neighbor cells as a separated Markov state, so we create an *n+1*-state Markov model:

– stationary state *(S)*

– move to neighbor 1...n state $(M_{N1}...M_{Nn})$

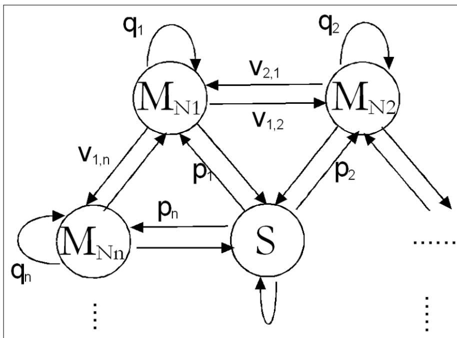
The steady state probabilities can be calculated as in the previous cases (Eq. 4).

$$\begin{aligned} &P_S\cdot\sum_{i=1}^{n} p_i = \sum_{j=1}^{n}(1-q_j-\sum_{l\neq k}^{n} v_{k,l})P_{Ni} \\ &\ldots\ldots \\ &P_{Nk}(1-q_k) = P_S\cdot p_n + \sum_{i\neq k} P_{Ni}\cdot v_{i,k} \qquad 1\le k\le n \\ &\ldots \end{aligned} \qquad (4)$$

Using the steady state result the predicted number of users in the next time slot is given by Eq. (5).

$$N_i(t+1) = N_i(t)\cdot P_S{}^i + \sum_{j\in S^i_{adj}} N_j(t)\cdot P_{M_i}{}^j(j) \qquad (5)$$

where S^i_{adj} means the set of cell indexes from all of the neighbors cells of cell *i*.



$$\Pi = \begin{bmatrix} 1-\sum_{m=1}^{n} p_m & p_1 & p_2 & \cdots & \cdots & \cdots & p_n \\ 1-q_1-v_{1,2}-v_{1,n} & q_1 & v_{1,2} & \cdots & \cdots & \cdots & v_{1,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 1-q_i-v_{i,i+1}-v_{i,i-1} & \cdots & \cdots & v_{i,i-1} & q_i & v_{i,i+1} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 1-q_n-v_{n,1}-v_{n,n-1} & v_{n,1} & \cdots & \cdots & \cdots & v_{n,n-1} & q_n \end{bmatrix}$$

Figure 3. State diagram and Π matrix of n+1-state Markov model (M_{n+1})

2.3 Complexity and accuracy of the model

Based on the Markov model generator method introduced in the previous section, a specific model can be derived depending on the complexity limits and the precision (accuracy) demand.

The accuracy of the model increases as the number of states grows. The number of states grows when the movement history (time dimension) is increased, or when the number of direction (direction dimension) is increased. Increasing the time dimension increases the number of states exponentially, increasing the direction dimension increases it linearly. With the state-space rising, the computational complexity of the Markov steady state calculations also follows a rising curve. The question is the characteristic of these functions and the existence of a theoretical or practical optimum point.

It is assumed that each cell has N neighbors and the 3-state (stay, left and right-move state) model is used to determine the users movement. It is also assumed that $N/2$ cells belong to both left and right Markov-states, and the users are uniformly distributed between cells. A theoretical error can be derived from this assumption since in most cases the user motion pattern does not result in a uniform distribution in the $N/2$ cells. In the worst case the users move with probability 1 into one of the neighbor cells. The error can be measured with the difference between the uniform distribution and the worst case. This difference is given by (6).

$$1-\frac{1}{N/M}+\frac{1}{N/M}((N/M)-1)=2\left(1-\frac{1}{N/M}\right), \qquad (6)$$

where N means the neighbor numbers, and M means the direction numbers in the model.

We measured the computation complexity also as the function of state number. This enables us to compare complexity and prediction error in an easy way. Based on the 1...M-state model the prediction computation is calculated with the costs of Markov steady state mathematical operations and other procedures necessary for transition probabilities. The complexity can be estimated with (M_{states}^3+ M_{states}+ $1/M_{states}$).

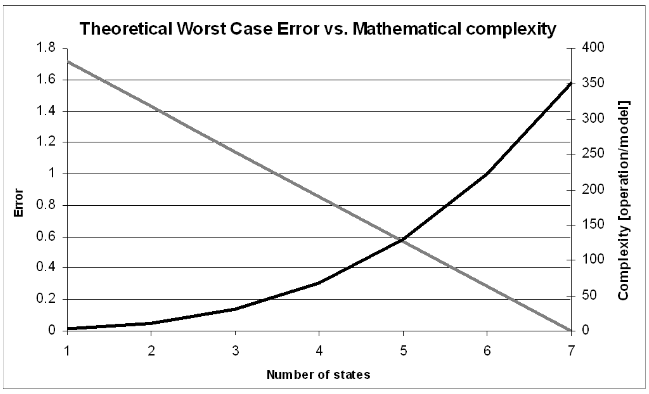


Figure 4. Complexity and accuracy of the calculation

Figure 4 shows the complexity and error characteristic. In the given model calculations the optimal point of operation is around 5 states where error is minimal at this level of error.

In the previous comparison the direction dimension sizing is used only. If we want to rely on the information of the previous steps for the estimation, then movement history has to be introduced into the model. In fact this means that every state in the Markov-chain has to be changed with M states. This causes exponential state number explosion that can be seen in *Figure 5*.

As we have extended our model step by step in time and in direction dimension, its precision increased along with the complexity of the model. In order to minimize

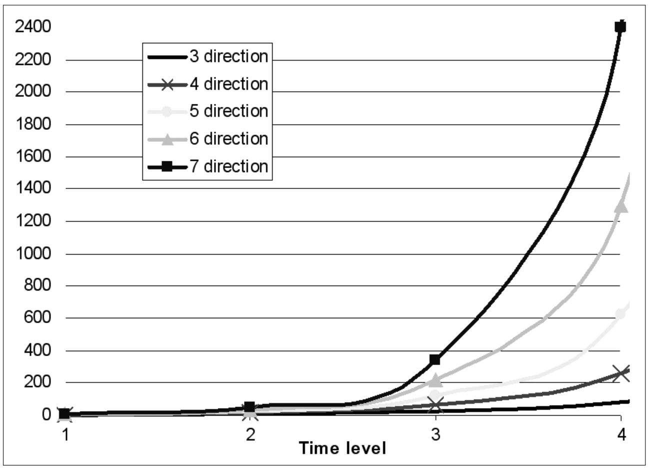


Figure 5.
The increase of number of states in the case of different direction dimensions (3,4,5,6,7 states), and different time dimensions (1,2,3,4)

the complexity, we developed a simple algorithm, which is able to minimize the number of states based on merging adjacent cells. Due to size limitation this algorithm is not discussed in this paper.

2.4 The effect of information of previous visited cells in the model

Neglecting the recent transition series of users in the cluster, i. e. when the model does not take into account in model buildup the previous steps of mobile users, the estimation is less precise.

In this section we show a simple example of the effect of previous user steps on the accuracy of the prediction. Let us consider two routes as shown in *Figure 6/b*. The accuracy of transition probability estimations is better if the model knows where the users came from, compared to the RW-like estimation which cannot differentiate the users on the two routes *(Figure 6/a)*.

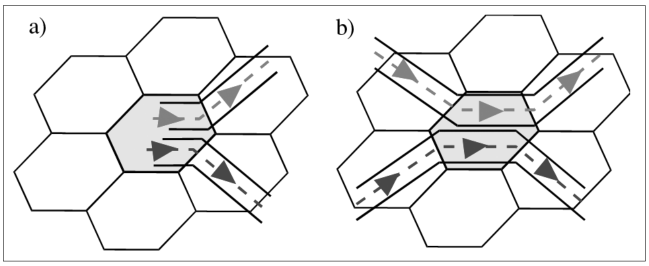


Figure 6. User prediction methods
a. Model without memory, unknown where is the users come from
b. Model with memory, the previous steps of the users taken in account

To find the error rate of a movement-history-less RW model compared to an algorithm that possesses user distribution from history, we use the example cells shown in Fig. 6. Let us define the following parameters:

– the number of incoming users on the upper route at timeslot t is $in1_t$,
– the number of incoming users on the lower route at timeslot t is $in2_t$,

– similarly the number of users leaving on the upper route at timeslot t is $out1_t$,

– the number of users leaving on the lower route at timeslot t is $out2_t$,

and the user movement directions with a simple transition matrix $\mathbf{P}=\begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$.

The RW model in Fig. 6/a calculates the number of leaving users *(out1, out2)* with a history estimation of the **P** matrix, and with the sum of *in1* and *in2* (the total number of users in the observed cell), but without the knowledge of *in1, in2*.

In our proposed algorithm, the use of *in1, in2* means that the model calculates based on the information of the incoming users. The incoming users can be considered uniform or different (marked users). Since the latter is more accurate, in this comparison we use marked users.

Assume that the history estimation of the RW model's **P** matrix is based on the previous timeslot. That is the P($out1_{t+1}$) and P($out2_{t+1}$) probabilities can be substituted with the relative frequency of $out1_t/(out1_t+out2_t)$ and $out2_t/(out1_t+out2_t)$, respectively.

Applying the same assumption on the algorithm with history, the number of leaving users can be calculated with the **P** matrix itself, that is $out1_{t+1}= in1_t{}^*p_{11}+ in2_t{}^*p_{21}$ and $out2_{t+1}= in1_t{}^*p_{12}+ in2_t{}^*p_{22}$. At a given and constant **P** matrix let us assume that the incoming user distribution varies, that is the $in1_t/in2_t$ ratio (Incoming Distribution – ID) changes.

Figure 7 shows the error of RW compared the estimation using history.

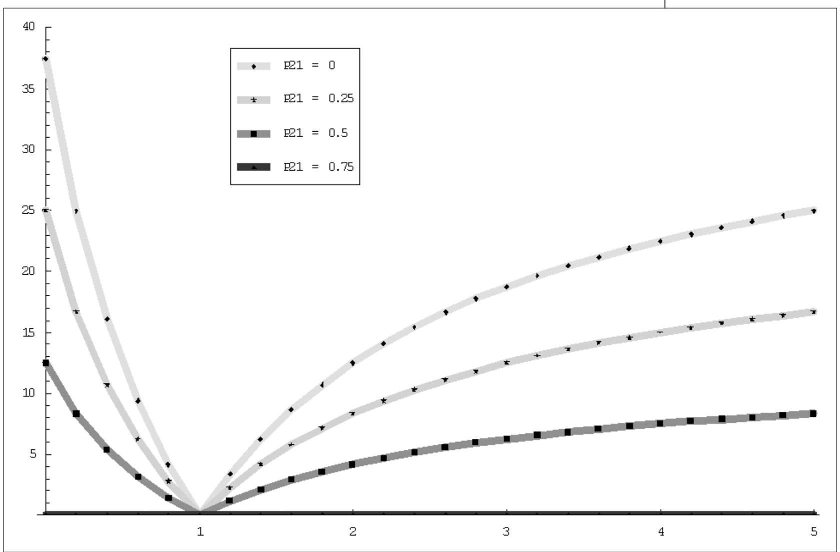


Figure 7. HOV prediction error in percents – Given $in1_{t-1}/in2_{t-1}=1$ and $P=\{\{0.75, 0.25\}.\{p_{21}, 1-p_{21}\}\}$, p_{21} plotted with four different values

The RW model works with error if ID_{t-1} is different from ID_t which is caused by the fact that the RW history **P**-estimation in this special case equals the number of leaving users of the previous timeslot. That is it does not include the actual ID_t value. Contrarily, the history-model calculates with the actual number of incoming users and the **P** matrix itself, which gives the exact probabilities of the leaving users distribution. The error rate caused by the lack of history increases as the variance of ID increases that is the $in1_t/in2_t$ ratio changes.

Using history cannot enhance further the accuracy of the estimation if $p_{21} = p_{11}$ and Fig. 7 shows a constant zero error rate ($p_{11} = p_{21} = 0.75$). In this case the outgoing direction of each user is independent of the incoming direction and the history is useless, since users arriving from each direction are leaving towards a given direction with the same probabilities.

The results show that our proposed movement history significantly increases the accuracy of the model in cases when the ID distribution in an arbitrary cell has high variance, or has periodicities without stationary distribution.

To apply the accuracy with the use of information about the previous user visited cells in the mobility model, we introduce our estimation model discussed in the previous section (M_n, where n denotes the number of Markov states). The states of the model show and store where the users came from. The memory can be interpreted in two meanings. Time dimension memory shows the number of timeslots in the past that the model considers. Thus a model with m time dimensions in time t can calculate the next transition based on the user position in *(t-m, ... t-2, t-1).* Direction dimensions memory shows the number of directions that the model can differentiate. In general a cell cluster consists of hexagonal cells. The direction dimension of a model on this cluster is maximum 6. If transitions from the central cell to two adjacent cells are not differentiated then the direction dimension is decreased by 1.

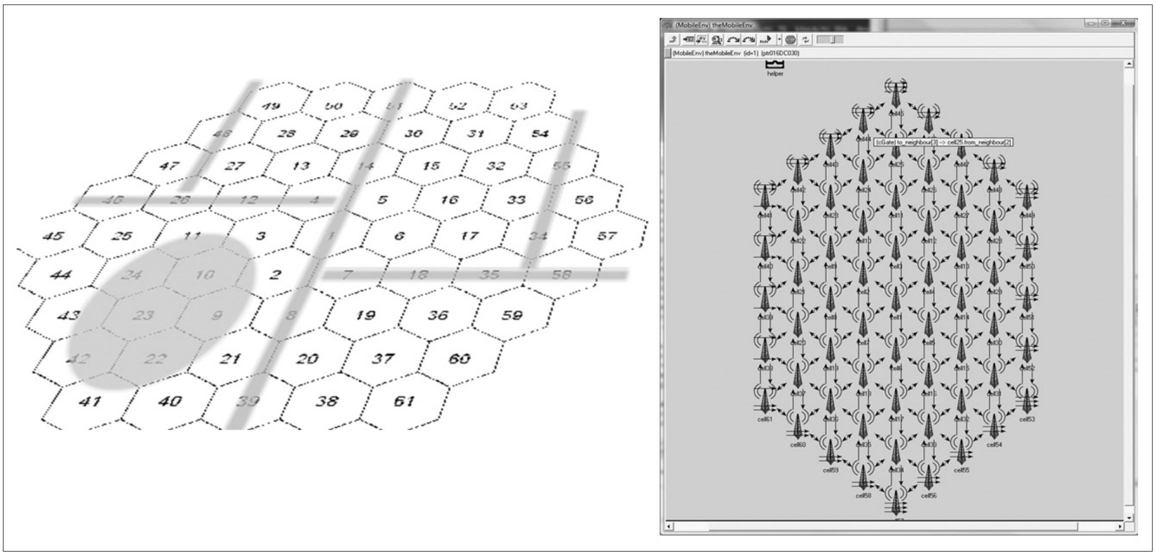
If we use more information from user movement history in prediction than our model does, the complexity increases exponentially. Obviously the direction numbers in the model influence the complexity as well. We come to a simpler model if we use only left and right directions instead of usage of all neighbor cells as a state in Markov model as it was shown in the previous sections.

Our work was motivated to find an optimal size of estimation parameter space with the highest accuracy beside tolerable complexity.

3. Simulation and numerical results

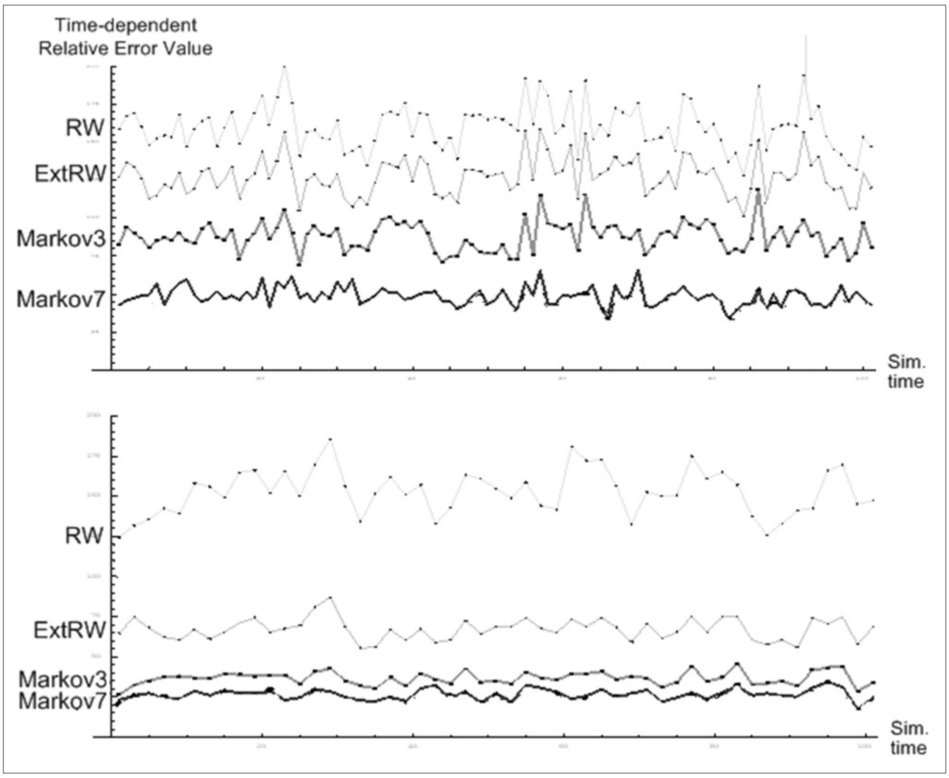
The inaccuracy of the RW-based mobility models depends on the properties of the transition probabilities. The RW model is only capable of accurate prediction of user movements in case of uniform movement distributions (e.g. all elements in the transition probability vector are equal to 1/6). In the simulation we applied

Figure 8.
a. Cell cluster with streets and park
b. OMNET++ simulation



an extended RW model (ExtRW) which was capable of tracking different cell dwell times. ExtRW can model better different user velocities with the correct dwell time parameters than RW which uses a fixed dwell time. With the variable dwell time parameter, the ExtRW simulation model can adapt to different movement velocities, i.e. a user with slow motion spends more time in each cell before he/she initiates handover to one of its neighbors. ExtRW does not cope with different directions of stepping forward. Thus stepping into each neighboring cell has an equal conditional probability. On the condition that at a specified moment a handover is done, the direction is uniformly distributed on the set of neighboring cells.

Figure 9.
TREV values in RW, ExtRW, M_3, M_7 models with λ=(1, 4)



The estimation procedure was validated by a simulation environment of a cell cluster shown in *Figure 8*. The cluster consisted of 61 named cells, the simulation environment included geographical data that are interpreted as streets on the cluster area. The drift of the movement is heading to the streets from neutral areas.

The simulation used 610 mobile terminal (10 for each cell), in the initial state uniformly distributed in the cluster. The average motion velocity of the users is parameterized with a simple PH cell dwell time simulator (reciprocal of exponentially distributed values).

The simulation consists of two parts. The reference simulation is the series of the transitions that the mobiles have initiated between cells. It produces a time-trace that contains the actual location data for each mobile terminal in the network. We have used this reference simulation as if it was a provider's real network trace.

The estimation procedure uses the past and the current reference simulation results to estimate future number of users in each cell. The estimation error is interpreted as the measure of accuracy of each mobility model in this paper.

The prediction starts 100 timeslots after the reference simulation initiation. During the warm-up process the reference simulation produces enough sample data for the correct estimation which uses the previous reference results as an input to estimate the future user distribution. Each user-transition in the 100-timeslot reference period is used to derive transition probabilities, motion speed and

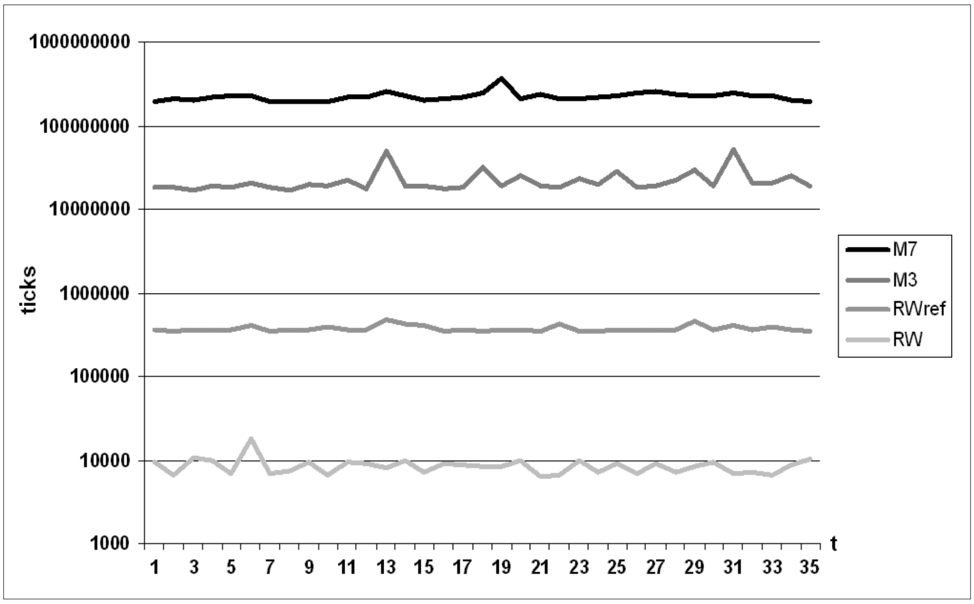


Figure 10.
CPU usage of the models on logarithmical scale

patterns in the simulation cell-space. These patterns serve as an input for the simulation threads of each mobility model. The models have the same input throughout the simulation process so that the results are comparable.

The estimation errors of the models in the simulation were measured with the average error of the cells in each timeslot. It produces a time-dependent relative error value (TREV) in each timeslot for the cell-cluster. TREV shows the average error compared to the actual user number in the cells. It can be seen in *Figure 9* that TREV depends on the dynamics of the motion, basically on the cell dwell time. The generic lambda (λ) parameter affects the motion velocity of the simulated users, the higher value means longer dwell time thus slower motion.

The relative performance of the models can be seen in *Figure 10*. The execution time of the methods of the model is plotted in each timeslot on logarithmic scale.

4. Conclusions

In this paper we proposed an alternative Markov-chain based method. The simulation results proved the analytical properties of the proposed mobility models.

The algorithm with the simple RW model is not capable of precise adaptive location prediction due to its inflexible parameter set. Since the user movement patterns in the simulation are not completely random due to the streets and geographical circumstances, the uniformly distributed Random Walk pattern cannot model it.

The Markov mobility model is the most accurate in the estimation process since it has the ability to calculate with motion direction, speed and the recent handover event (user history) also. The three-state model focuses on cell dwell time since it differentiates only two motion directions which cannot follow general drifts. The n-state model is more sophisticated in terms of both the cell dwell time and motion direction since it is capable of following for example six different drifts in the cell cluster.

The network operator may use a seven-state Markov model to make predictions on the future distribution and location of users among radio cells to justify CAC or other QoS decisions.

Authors

TAMÁS SZÁLKA was born in Budapest in 1982. He received the M.Sc. degree in Computer Science from Budapest University of Technology and Economics (BUTE) in 2006 then he started his Ph.D. studies at BUTE, on the Department of Telecommunications. Beside his Ph.D. studies, he is doing engeneering research as Ph.D. student and fellow researcher in the Mobile Innovation Centre of BUTE. As planned, he will obtain his Ph.D. degree in 2011. His work is relatad to the area of mobililty modeling and location based services of cellular networks.

PÉTER FÜLÖP was born in Mosonmagyaróvár in 1980. He received his M.Sc. degree in Technical Informatics from the Budapest University of Technology and Economics in 2005. Currently he is a PhD student at the Department of Telecommunication and fellow reserarcher in the Mobile Innovation Centre of BUTE. He is working as a system test engineer at Research and Development Department of Ericsson Telecommunications Hungary. His research interest is IP mobility, interworking of heterogeneous mobile networks and movement modeling in cellular, mobile networks. The PhD thesis is going to be written on Complex Mobility Management Systems and Applications.

SÁNDOR SZABÓ was born in 1977. He received the M.Sc. degree in Electronic Engineering from the Budapest University of Technology (BUTE) in 2000 then he started his Ph. D. studies at BUTE. Currently he is Assistant Professor at the Dept. of Telecommunications of BUTE. He leads research projects, and also a project leader at the Mobile Innovation Centre of BUTE. His research interests include mobile and wireless systems, mobility modeling, mobility management and the integration of heterogeneous wired and wireless networks.

References

[1] A. Jardosh, E.M. Belding-Royer, K.C. Almeroth, S. Suri, “Towards Realistic Mobility Models for Mobile Ad Hoc Networks”, In Proceedings of ACM MobiCom, pp.217–229, September 2003.

[2] Jungkeun Yoon, Brian D. Noble, Mingyan Liu, Minkyong Kim, "Building realistic mobility model from coarse-grained wireless trace", In Proceedings of the 4th International Conference on Mobile Systems, Applications and Services (MobiSys), Uppsala, Sweden, June 2006.

[3] S. Michaelis, C. Wietfeld, "Comparison of User Mobility Pattern Prediction Algorithms to increase Handover Trigger Accuracy", IEEE Vehicular Technology Conference, Melbourne, May 2006.

[4] M.M. Zonoozi, P. Dassanayake, "User mobility modelling and characterization of mobility patterns", IEEE Journal on Selected Areas in Communications, Vol. 15, No. 7, September 1997.

[5] B. Liang, Z. Haas, Predictive distance-based mobility management for PCS networks, In Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), March 1999.

[6] V. Davies, Evaluating mobility models within an ad hoc network, Master's thesis, Colorado School of Mines, 2000.

[7] X. Hong, T. Kwon, M. Gerla, D. Gu, G. Pei, "A Mobility Framework for Ad Hoc Wireless Networks", Proceedings of ACM 2nd International Conference on Mobile Data Management (MDM 2001), Hong Kong, January 2001.

[8] Vincent W.-S. Wong, Victor C.M. Leung, "Location Management for Next-Generation Personal Communication Network", IEEE Network, September/October 2000.

[9] Levine et al, "A Resource Estimation and Call Admission Algorithm for Wireless Multimedia Networks Using the Shadow Cluster Concept", IEEE/ACM Trans. on Networking, Februar 1997.

[10] R. Chellappa, A. Jennings, N. Shenoy, "A Comparative Study of Mobility Prediction in Cellular and Ad Hoc Wireless Networks", Proceedings of the IEEE International Conference on Communications 2003 (ICC2003), Alaska, USA, May 2003.