An Application of Markov Jump Process Model for Activity-Based Indoor Mobility Prediction in Wireless Networks

Joanna Kołodziej
Department of Mathematics and Computer Science
University of Bielsko-Biała
Bielsko-Biała, Poland
jkolodziej@ath.bielsko.pl

Samee Ullah Khan
Department of Electrical and Computer Engineering
North Dakota State University
Fargo, ND, 58108–5060, USA
samee.khan@ndsu.edu

Lizhe Wang
School of Informatics and Computing
Indiana University
Bloomington, IN 47405
wanglizh@indiana.edu

Nasro Min-Allah and Sajjad A. Madani
Department of Computer Science
COMSATS Institute of Information Technology
Pakistan
nasar@comsats.edu.pk, madani@ciit.net.pk

Nasir Ghani
Department of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131–0001, USA
nghani@ece.unm.edu

Hongxiang Li
Department of Electrical and Computer Engineering
University of Louisville
Louisville, KY 40292, USA
hongxiangli@gmail.com

***Abstract*—One of the most important objectives of a wireless network is to facilitate a prediction of users' mobility regardless of their point of attachment to the network. In indoor environments the effective users' motion prediction system and wireless localization technology play an important role in all aspects of people's daily lives.**

In this paper we propose an activity-based continuous-time Markov model to define and predict the human movement patterns. This model is a simple extension of an *Activity based Mobility Prediction algorithm using Markov modeling (AMPuMM)* technique. Both models are experimentally evaluated in realistic small university campus scenario. The obtained results show us the high efficiency of the jump methodology in the prediction of the students' activities in the indoor campus environment.

***Keywords*-Indoor Environment, Wireless Network, Markov Chain, Markov Jump Process**

I. Introduction

The foremost objectives of a wireless network is to facilitate the communication of mobile users and the widespread tracking and prediction of their mobility regardless of their point of attachment to the network. In indoor environments the effective users' motion prediction system and wireless localization technology play an important role in all aspects of people's daily lives, including e.g. living assistant, navigation, emergency detection, surveillance/tracking of target-of-interest, evacuation purposes, and many other location-based services. Prediction techniques that are currently used do not consider the motivation behind the movement of mobile nodes and incur huge overheads to manage and manipulate the information required to make predictions.

User's mobility prediction is an important maneuver that determines the location of the user in the network by the manipulation of the available information about the user's activity. The prediction accuracy depends on the user mobility model and the prediction methodology. Many models assume a basically random movement of the user. While this is sufficient to simulate the performance of network level protocols, this assumption is not suitable for application level evaluation.

To overcome such limitations we propose in this paper an extension of the *Activity based Mobility Prediction algorithm using Markov modeling (AMPuMM)*, presented in [8], by the implementation of the Markov jump continuous-time process [5] framework to predict the future location of the users. The presented user's mobility model is a component of the general activity-based model and the user's movement patterns are defined as the paths in a multi-graph representing the physical environment.

The remainder of the paper is organized as follows. In Sec. II we define the activity-based model for the indoor user's behavior. In Sec. III the main idea of the AMPuMM model and its extension by using the continuous-time Markov process are presented. We experimentally evaluate our approach against the AMPuMM model in Sec. IV. The paper is ended in Section VI with some conclusions and final remarks.

II. Activity-Based Model

In activity-based modeling a typical daily user's behavior is characterized as sequences of user's activities derived from a set of parameters. The model's type can be distinguished by the way it illustrates the users decisions of when and how an activity is carried out.

An activity-based user-centered approach to various types of wireless networks is presented in [4] and [7]. This model derives an integrated view on mobility and network usage from a users real-world activity perspective. It is based on the results of the users' decisions and activities and enables their executions by connecting the locations of the consecutive activities. The main idea of the model is presented in Fig. 1.

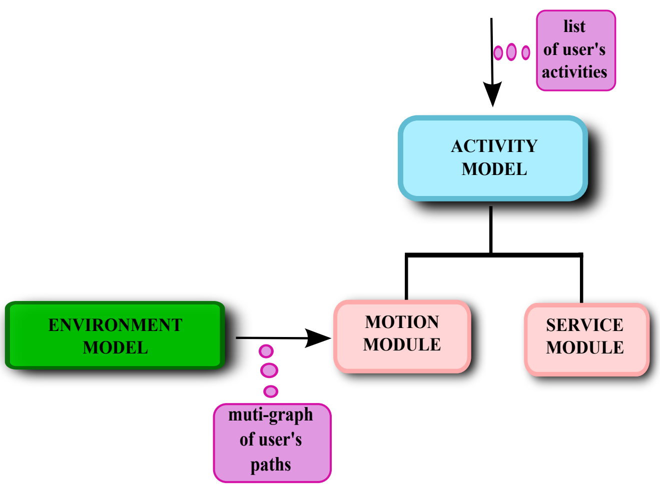

Figure 1. Activity-based model modules

A general model consists of the four following modules (compare with [4]):

- Activity Model,
- Motion Module,
- Service Module,
- Environment Model.

In *Motion* and *Service* modules the necessary movements from a given user's activity and needful services for the activity support are defined. In this paper however, we mainly focus on the remaining two components of the model, namely ***Activity*** and ***Environment*** models.

A. Activity Model

The main aim of the activity model is a transformation of an abstract list of possible non-networking activities into a concrete activity schedule. Each activity is characterized by the parameters presented in Table I.

To define the activity model, first we have to specify a list of the non-networking user's actions, which are classified into several parametrized groups. The concept of an action class leads to the collective term of an activity.

Table I
Users' activity criteria

Parameter	Description
Starting time:	a fixed point of time, in which the activities either start or are scheduled
Duration:	the time, in which the activities finish or they are predicted to be completed; the activities' duration is usually defined by using some specified random distribution
Priority:	the criterion, which can be specified by the user in order to make some preferences in the activities. The user's activities are scheduled in the order of their priority when time conflicts arise. Each user can specify its own set of priorities

Let us consider an university computer science department as a hypothetic scenario for such a model. Typical actions of the staff members include providing the lectures, seminars and lab work, providing the exams, preparation to the classes, research work, going to the library, attending departmental meetings, etc. There is another class of possible social activities, which must be also analyzed, i.e. going to the cafeteria, eating lunch, relaxing. In the case of lack of the detailed specification of the teaching topics (low level of details), the seminars, labs and lectures can be generalized to an activity of a course. These actions typically share common locations and are both regular and recurring events.

All user's activities in the departmental model can be also classified under the duration time criterion to those with a fixed starting time like providing the lecture, and free-floating activities like borrowing a book from the library. Usually, free-floating activities have durations that adhere to random distributions. To avoid the overlapping of the activities in time, the users can additional define their own priorities. The activities of higher priority take precedence over activities of lower priority when there are conflicting starting or ending times.

The activity model is used mainly for the calculation of the concrete activity schedules for a user.

B. Environment Model

The environment model is usually represented as an *undirected multigraph* $G = (V, E(w))$, where:

- V - is a set of the vertices in the graph. We define it as sum $V = T \cup L$, where T denotes the transition nodes and L - a set of the location nodes;
- $E(w)$ - is a set of graph multiple edges. The parameter w denotes the *width* of the path, which is expressed as a number of edges for the paths between two location or transition nodes.

An example of the graph structure related to departmental model from the previous section is presented in Figure 2.

The edges in the graph define the walkable paths in and around the department. By using the edge 'width' parameter

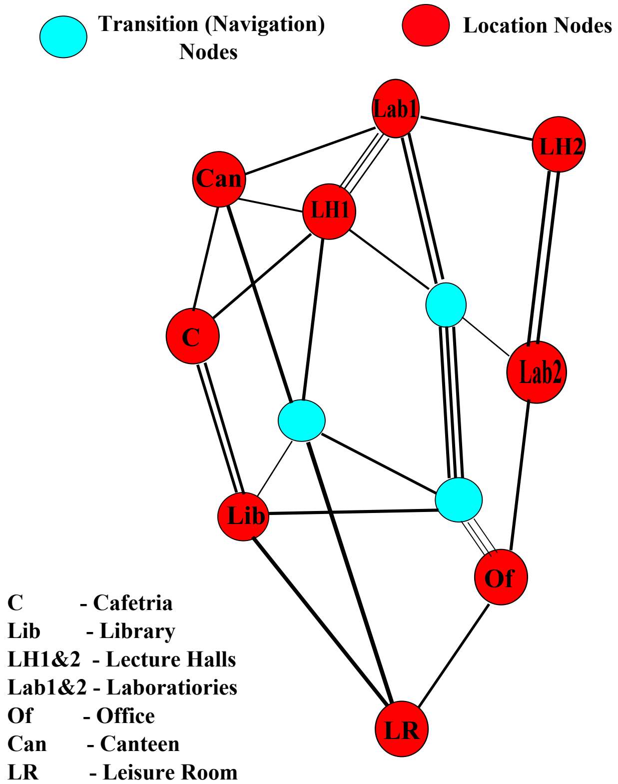

Figure 2. Activity-based model modules

we can model a realistic group movement, which means that individuals are able to move side by side on one of these parallel paths. We call as *preferred paths* the paths, which are most commonly 'generated' by particular users.

The vertices of the graph are divided into two following groups:

- ***transition nodes***, which represent the intersections of the users' mobility trials and users' activity locations like cafeteria, library, lecture room, lab room, etc;
- ***location nodes***, which are interpreted as 'likely destinations' during any movement sequence.

The environmental multi-graph and the activity model are the basis for prediction the mobility of the users in the system (see [3]; [10]). The use of the analytical mobility models is facilitated by the particular location node shape and type. Reaching a vertex connected to an activity location is related to a start of the execution of such activity. In the following section we present the user's mobility model based on the Markov jump process.

III. Activity-Based Mobility Models

Activity Markov-based models accumulate and store information about the mobility behavior of the users in terms of the time sequence in which activities are performed. I this section we firstly highlight the main idea of the *Activity-based Mobility Prediction model using Markov modeling (AMPuMM)* [8] proposed by Mathivaruni et al. for the prediction of future location of the mobile nodes in wireless networks. Then we define a continuous-time Markov jump process [1] for modeling an actual dynamics of the system and the user's movement prediction with the environmental multi-graph as a 'road map'. We prove that AMPuMM model is a simple approximation of our Markov-based solution.

A. Activity-based User's Mobility Prediction using Markov Modeling (AMPuMM)

The main aim of the Activity-based user's Mobility Prediction using Markov Modeling algorithm (AMPuMM), presented in [8], is to define and analyze activity patterns for each user. These patterns are generated based on the activity monitoring provided in a specified time interval. The whole process is modeled as a Markov chain with seven typical activity states, where transition probabilities are calculated to predict an $(n+1)$-th day user's location using an information from the past n days. The user's activities are classified into two groups: navigation activities and location activities. The authors propose in [8] a minimum threshold value t_{min}, which is set for an activity to graduate as a location activity.

Formally, the AMPuMM can be defined in the following way:

Definition 3.1

Let us denote by $A_7 = [a_1, a_2, \ldots, a_7]$ the vector of seven activity states, and by $N(a_i)$ – a number of time slots in which the activity a_i is performed. The process of activity transition is modeled as a homogeneous Markov chain with the states from the set A_7 and a transition probability matrix P defined as follows:

$$P = \begin{bmatrix} p_{(a_1,a_1)} & \cdots & p_{(a_1,a_7)} \\ p_{(a_2,a_1)} & \cdots & p_{(a_2,a_7)} \\ \vdots & \vdots & \vdots \\ p_{(a_7,a_1)} & \cdots & p_{(a_7,a_7)}, \end{bmatrix} \tag{1}$$

where

$$p_{(a_i,a_j)} = \frac{N(a_i - a_j)}{N(a_i' - a_j)}, \tag{2}$$

$N(a_i - a_j)$ denotes the number of time slots in which the activity a_j follows activity a_i, and $N(a_i' - a_j)$ is the number of time slots in which the activity a_j doesn't follow the activity a_i.

An initial state probability vector is defined by $\Lambda(a_i) = \frac{N(a_i)}{N(a_i')}$, and the next state probability vector is defined by $A_7(t) = A_7(t-1) \times P$, where $A_7(t)$ is the 'state' of the activity vector A_7 in the t-th time slot. It means that each day (time period) of the observation can be additionally divided into the several time segments, and, instead of one transition matrix for the whole day, few transition matrices for EACH individual activity can be defined and considered. Thus, the generated Markov chain has a multidimensional transition matrix, which in fact increase the model complexity.

B. An extension of AMPuMM model - Markov jump process approach

The main drawback of the AMPuMM model presented in Sec. III-A is the difficulties in the implementation of the multidimensional transition matrix. It is very hard to extract the information about the joint actions of the users. It can be concluded from the experimental evaluation of the model performed in [8], that this approach can be effective just for a small area model (not so many activities and users) and in the case of ignorance of the 'transition (navigation)' nodes. The activity in AMPuMM model is restricted just to an action of the user, which terminates in a specified location. Another drawback is the restriction of such model just to the discrete time case, which means that to achieve a good prediction it is necessary to consider many time slots, which raises the model complexity.

To avoid such disadvantages we propose to model the user's mobility as the *Markov continuous-time jump process* [1], which can be defined as follows:

Definition 3.2 *Let us denote by $A = [a_1, \ldots, a_n]$ a vector of all states of the system interpreted as an user's locations. A Markov jump process over A is a random variable $X(t)$, parameterized by time $t \in [0; \infty)$, with an initial state a_0 at time interval from $t = 0$ until t_1, a set of of jump times – $t_1, t_2, \ldots$ and transition probabilities – $r(a_i, a_j)$, such that:*

- $r(a_i, a_i) = 0, \forall i = 1, \ldots, n;$
- $\sum_{a_j} r(a_j, a_i) = 1, \forall a_i \in A.$

Transition probability distribution for the process is defined by the following formula:

$$P(\tau \leq t, X(\tau) = a_j, X(0) = a_i) = r(a_j, a_i) \cdot F_{a_i}(t). \quad (3)$$

where the time, in which the system remains in the state a_i is modeled by the exponential distribution:

$$F_{a_i}(t) = 1 - e^{-\gamma_{a_i} t} \quad (4)$$

with the following probability density function: $f_{a_i}(t) = \gamma_{a_i} e^{-\gamma_{a_i} t}$.

It follows from the above definition that the process stays in the state a_j during the time interval $[t_j, t_i)$ and then jump to the next state a_i. In the general case, the distribution function F_{a_i} may be different for different states.

Let us denote by $p((b, a) \mid t)$ the conditional probability that the jump process is in state b at time t given that it was in state a at time 0. Given times $0 < t_1 < t_2 < \ldots < t_n < s$ and $t > 0$, the Markov property for a jump process is defined as follows:

$$\begin{aligned} &P((X(t+s) = b) \mid (X(s) = a; X(t_n) = a_n; \ldots \\ &\ldots; X(t_1) = a_1)) = p((b, a) \mid t) \end{aligned} \quad (5)$$

With the Markov property, joint probabilities for Markov jump process can be expressed as

$$\begin{aligned} &P(X(t+s) = a; X(s) = b; X(0) = c) = p((a, b) \mid t)\cdot \\ &\cdot p((b, c) \mid s) \cdot P(X(0) = c) \end{aligned} \quad (6)$$

for $t, s > 0$. Hence the Chapman-Kolmogorov equation [1] for the Markov jump process is defined in the following way:

$$\begin{aligned} &p((a, c) \mid (t+s)) = \sum p((a, b) \mid t) \cdot p((b, c) \mid s), \\ &\forall a, b, c \in A \end{aligned} \quad (7)$$

Usually a geometrically distributed discrete random variable can be approximated by a continuous time exponentially distributed random variable (see [1]). Therefore, we can demonstrate now that the AMPuMM model can be approximated by our Markov jump process.

Proposition 3.1: Let us denote by $\hat{A} = [a_1, \ldots, a_k]$ the vector of all possible activities of the users, which is in fact a simple generalization of the vector A_7 specified in Sec. III-A (we consider k activities and in the very special case k can be 7). Let $\hat{P}$ be the user's activity transition probability matrix defined as follows:

$$\hat{P} = \begin{bmatrix} p_{(a_1,a_1)} & \cdots & p_{(a_1,a_k)} \\ p_{(a_2,a_1)} & \cdots & p_{(a_2,a_k)} \\ \vdots & \vdots & \vdots \\ p_{(a_k,a_1)} & \cdots & p_{(a_k,a_k)} \end{bmatrix}. \quad (8)$$

The transition of the user's activities is modeled as Markov jump process, in which transition probabilities are defined in the following way:

$$r(a_j, a_i) = \frac{p_{(a_j,a_i)}}{p_g(i)} \quad (9)$$

for $j \neq i$ and $p_{(a_j,a_i)}$ is the probability of transformation from state a_j to a_i specified for the matrix in Eq. 8. The probability of leaving the state a_i is calculated as follows:

$$p_g(i) \equiv \sum_{j \neq i} p_{(a_j,a_i)}. \quad (10)$$

Proof: The probability that a Markov chain remains in state a_i for n steps can be calculated as follows: if $p_{(a_i,a_i)} = 0$, then the only possibility is $n = 0$; it must always make a transition and never stays in the same state. Otherwise, the probability is calculated using the following formula:

$$p^n_{(a_i,a_i)} = e^{n \cdot ln(p_{(a_i,a_i)})} = e^{n \cdot ln(1 - p_g(i))} \quad (11)$$

For $p_g(i) \ll 1$, we achieve $ln(1 - p_g(i)) \approx -p_g(i)$, so then this probability is exponentially distributed with rate $p_g(i)$. This suggests that transition probabilities specified in Eq. 9 satisfy the requirements for a Markov jump process, i.e.:

$$\begin{aligned} &a) \quad r(a_i, a_i) = 0; \\ &b) \quad \textstyle\sum_{a_j} r(a_j, a_i) = \frac{\sum_j p_{(a_j,a_i)}}{p_g(i) = \sum_j p_{(a_j,a_i)}} = 1. \end{aligned} \quad (12)$$

For $i \neq j$ we have $\gamma_{a_i} \equiv \gamma p_g(i)$ where γ^{-1} has units of time. This defines the time scale of the jump process, and the choice $\gamma = 1$ makes the n step of the Markov chain equivalent to $t = n$ in the Markov jump process.

■

IV. Experimental Evaluation

We performed a simple experimental analysis in order to evaluate the proposed solution in a realistic indoor environment and to compare its efficiency with AMPuMM model prediction mechanism. The environmental model we used is presented in Fig. 2 and is based on the university campus in Bielsko-Biała. We identified cafeteria, canteen, library and leisure room as the most important nodes at common area for students of all departments and defined them as four location states in two considered Markov-based models. The analyzed campus part was approximated by a $500m \times 400m$ area.

We invited for our experiment 35 undergraduate students of the 6-th semester in computer science and asked them to fill out a daily schedule of their activities. The data was collected in the period of four weeks, since March 7, 2001 till April 3, 2011. The students had to specify where they had been or will be at given points in time (8, 10, 12, 13, 14, 16, 18 oclock). The first three weeks worth of activity data was separated from the last week worth. This allowed for activity habits to be learned from the first three weeks (learning set) and the last week (test set) to be used for testing purposes. For analysis of the activity data, we use actual lecture schedules and opening hours of campus facilities.

For each user, based on the results collected in the first three weeks of the experiment, we estimated the probabilities of moving between four location states distinguished for both Markov jump and AMPuMM models. The executions of the models have been repeated 30 times. To assess the models quantitatively, we used a distance measure, which determines the optimal average deviation between the generated users' motion patterns and those calculated by using the data collected in the fourth week of experiment. We also specified rates of predictability for both methods by estimating the relative errors of the prediction of the users' activities in the fourth week. Those two metrics were averaged over the users number. The obtained experimental results averaged over 30 runs are presented in Fig. 3 and Fig. 4.

It can be observed, that on average the generated users are about 115–250 meters from their real positions. This average distance error is rather significant. This may come from the fact that both Markov-based model tries to fill the whole simulation area uniformly, but there are few activity locations at the edges of the campus. The students are more likely to be in the middle of the area because they have to cross the center of the simulation area very often in order to reach their next location. Another very important reason is that the most important nodes of the environmental graph are located in different buildings. In this case even small error in prediction of the user's future trajectory has a strong impact on the distance error estimation. Markov jump model has been outperformed by AMPuMM on Mondays and Wednesdays, but the difference in distances is small. It is quite different for prediction errors results, where AMPuMM has been beaten in all instances. The errors in the users' activity prediction was much greater at the beginning of the week and before the weekends.

The results we achieved for both methods can be obviously improved by using the electronic devices for moni-

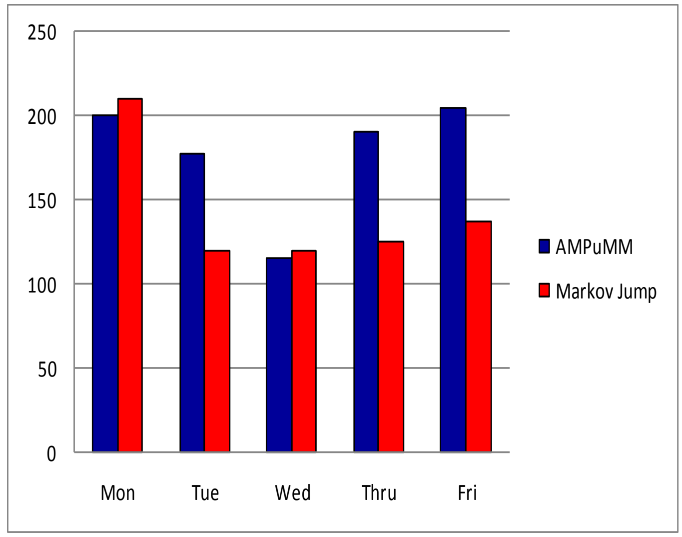

Figure 3. Optimal average distance deviation per weekday [m] (averaged over 30 runs)

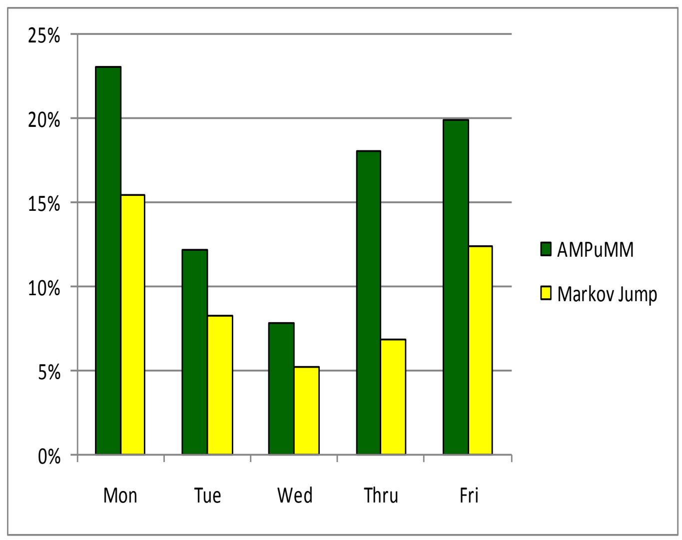

Figure 4. Users' activity prediction error per weekday [%] (averaged over 30 runs)

toring the students. It will allow much better estimation of the duration times for all activities. The students' sample selected to the experiment was also not very large and thus will not provide results with a high significance. However, we believe that our analysis helps to detect some major trends for the further research.

V. Related Work

Although the tracking of the users' location can be effectively solved in most cases by GPS-based navigating systems, a prediction of the future users' locations in complex environments remains a challenging task [12]. The researchers usually focus on the study of stochastic or learning mechanisms of the generation of users' movement patterns or simple extraction of the high-level information from collected raw location data. This extraction is realized by using some specialized clustering algorithms. D. Ashbrook et al. [2] combined a simple modification of k-means clustering algorithm and the Markov chain model. The main drawback of this approach is the complexity of the Markov module. Another Markov-based approach is demonstrated in [6]. The authors used a combination of recurrent self-organizing maps (RSOM) and the Markov chain model for the generation of the users' trajectories at the university campus. Future movement is predicted based on past movement trajectories. D.J. Patterson et al. [9] proposed a method to be used in the current transportation mode which used a dynamic Bayesian network model. Domain knowledge was incorporated into the Bayesian network model and the parameters of their network were learned using an expectation-maximization (EM) algorithm. Another interesting example of the application of the Bayesian network for 'learning' the 'type' of museum visitor (greedy, selective, or busy) from tracking their locations of can be found in [11].

VI. Conclusions

In this paper we developed an activity-based continuous-time Markov model to define and predict the human movement patterns in indoor environment. We used the Markov jump method for the specification of the user's mobility as the stochastic continuous-time process with a negative exponential time distribution. We formally proved a coherence of our model with Activity-based user's Mobility Prediction using Markov Modeling proposed in [8]. A simple experimental analysis in realistic scenario of a small university campus confirmed the improvement of the users activity prediction results by using the Markov jump methodology. Our approach, by electronic real-time navigation tools, may be used as an additional input into intelligent building automation systems, which can make it a crucial issue especially in the disaster management.

References

[1] Ait-Sahalia Y., Hansen L. P. and Scheinkman J. A.: "Operator Methods for Continuous-Time Markov Processes", In the *Handbook of Financial Econometrics*, Vol 1, Chapter 1, Elsevier Press, 2009.

[2] Ashbrook, D., and Starner, T.: "Using GPS to Learn Significant Locations and Predict Movement Across Multiple Users", *Personal and Ubiquitous Computing*, vol. 7, no. 5, pp. 275-286, 2003.

[3] Bettstetter C.: "Mobility Modeling in Wireless Networks: Categorization, Smooth Movement, and Border Effects", *Mobile Computing and Communications Review*, Vol. 5, No. 3, pp. 55–67.

[4] Breyer T., Klein M., Obreiter P., and König-Ries, B.: "Activity-Based User Modeling in Service-Oriented Ad-hoc-Networks", in*First Working Conference on Wireless On-Demand Network Systems (WONS 2004)*, Trento, Italy .

[5] Clifford P.: "Markov random fields in statistics". In G. R. Grimmett and D. J. A. Welsh, editors, *Disorder in Physical Systems*, pp. 19-32, Oxford University Press, Oxford, 1990.

[6] Han, S-J., and Cho, S-B.: "Predicting User's Movement with a Combination of Self-Organizing Map and Markov Model", in *Proc. of ICANN 2006, Part II, S. Kollias et al. (Eds.)*, Springer LNCS vol. 4132, pp. 884–893, 2006.

[7] König-Ries, Klein M., and Breyer T.: "Activity-Based User Modeling in Wireless Networks", *Mobile Networks and Applications*, Vol. 11, No. 2, pp. 267–277.

[8] Mathivaruni, R.V. , and Vaidehi, V.: "An Activity Based Mobility Prediction Strategy Using Markov Modeling for Wireless Networks", in *Proc. of the World Congress on Engineering and Computer Science 2008 WCECS 2008*, San Francisco, USA, 2008, pp. 379–384.

[9] Patterson, D., Liao, L., Fox, D., and Kautz, H.: "Inferring High-Level Behavior from Low- Level Sensors", in *Proc. of the 5-th International Conference on Ubiquitous Computing*, Seattle, WA, pp. 73–89, 2003.

[10] Scourias J., and Kunz T.: "An Activity-based Mobility Model and Location Management Simulation Framework ", in *Proc. of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '99)* , ACM, pp. 61–68.

[11] Sparacino, F.: "Sto(ry)chastics: A Bayesian Network Architecture for User Modeling and Computational Storytelling for Interactive Spaces", in *Proc. of the 5-th International Conference on Ubiquitous Computing*, Seattle, WA, pp. 54–72, 2003.

[12] Zhang, D., Xia, F., Yang, Z. and Yao, L.: "Localization Technologies for Indoor Human Tracking", in *Proc. of the 5th International Conference on Future Information Technology (FutureTech)*, May 2010, Busan, Korea.