Mobility Pattern Recognition in Mobile Ad-Hoc Networks

S. M. Mousavi
Department of Computer Engineering,
Sharif University of Technology
sm_mousavi@ce.sharif.edu

H. R. Rabiee
Department of Computer Engineering,
Sharif University of Technology
& ITRC
rabiee@sharif.edu

M. Moshref
Department of Computer Engineering,
Sharif University of Technology
moshref@ce.sharif.edu

A. Dabirmoghaddam
Department of Computer Engineering,
Sharif University of Technology
Dabir@ce.sharif.edu

ABSTRACT
A Mobile Ad hoc Network (MANET) is a collection of wireless mobile nodes forming a self-configuring network without using any existing infrastructure. Network nodes in a mobile Ad-hoc network move in some motion patterns called mobility models. The mobility models play a very important role in determining the protocol performance in MANET. Thus, it is essential to study and analyze various mobility models and their effect on MANET protocols. If we can recognize the mobility pattern of motion of mobile nodes in our environment we can customize our network protocols to deal with that existing mobility model. In this paper we introduce a new method for classification and pattern recognition of mobility traces into mobility models in mobile Ad-hoc networks. This method uses a simple learning based classification method to recognize the existing mobility model in raw mobility traces which was collected from real motion of mobile Ad-hoc nodes or mobility traces generated by mobility simulators. Our simulation results prove ability of our proposed method to accurately classify given unknown mobility traces into various mobility models.

Keywords
Mobile Ad-Hoc Networks, Mobility Models, Mobility Simulator

1. INTRODUCTION
A mobile ad-hoc network (MANET) is a group of mobile wireless nodes working together to form a network. Such networks can exist without a fixed infrastructure working in an autonomous manner and every mobile device has a maximum transmission power which determines the maximum transmission range of the device. As nodes are mobile, the link connection between two devices can break depending on the spatial orientation of nodes. Mobile ad-hoc networks have numerous applications in sensor networks, disaster relief systems and military operations. Some of the network constraints in mobile ad-hoc networks are limited bandwidth, low battery power of nodes, and frequent link unreliability due to mobility [1].

In order to thoroughly simulate a new protocol for an Ad-Hoc network, it is imperative to use a mobility model that accurately represents the mobile nodes that will eventually utilize the given protocol. Only in this type of scenario it is possible to determine whether or not the proposed protocol will be useful when implemented. Currently there are two types of mobility models used in the simulation of networks: traces and synthetic models. Traces are those mobility patterns that are observed in real life systems. Traces provide accurate information, especially when they involve a large number of participants and an appropriately long observation period. However, new ad-hoc network environments are not easily modeled if traces have not yet been created. In this type of situation it is necessary to use synthetic models. Synthetic models attempt to realistically represent the behaviors of mobile nodes without the use of traces.

The mobility model is designed to describe the movement pattern of mobile users, and how their location, velocity and acceleration change over time. Since mobility patterns may play a significant role in determining the protocol performance, it is desirable for mobility models to emulate the movement pattern of targeted real life applications in a reasonable way. Various researchers proposed different kinds of mobility models, attempting to capture various characteristics of mobility and represent mobility in a somewhat 'realistic' fashion. Much of the current research has focused on the so-called synthetic mobility models that are not trace-driven.

There are several synthetic mobility models proposed to literature to mimic the real motion of mobile nodes in real environments. It is obvious that the performance of mobile Ad-hoc protocols is affected by the motion pattern of mobile nodes. For example the average link duration in RPGM mobility model is much more than link duration of Random Walk model because in RPGM mobility model nodes travel near each other most of the time, therefore the duration time that to neighboring nodes exist in transmission range of each other is much bigger that this duration time in Random Walk mobility model [2,3].

If a network protocol can recognize existing mobility model in a real environment using collected mobility traces, it can customize its behavior to deal with that specific motion pattern of mobile nodes. For example in some protocols which use mobility Prediction methods [4,5] the accuracy of mobility prediction estimator differs in each mobility models. This mobility prediction estimator can be customized in each of the mobility models to make it more accurate and decrease the mobility prediction inaccuracy.

MC'o7 (Mobility'07), September 10-12, 2007, Singapore.

In this paper we introduce a new method for classification of collected unknown (mobility model of the trace is unknown) mobility traces from simulation or real environments into specific mobility models. Our classification method is based on supervised learning based statistical pattern recognition methods and it is similar to k-nearest neighbor classification method [6]. We use 3 features of mobility models called mobility metrics to classify mobility models. These features are: Average Degree of Spatial Dependence, Average Degree of Temporal Dependence and Average Relative Speed [3]. We use these mobility metrics to extract properties of each of mobility traces and use them as classification parameters.

For this purpose we designed and implemented a java based software called Mobility Analyzer to analyze mobility traces and extract various mobility metrics like average link duration, average path duration, average degree of spatial dependence, average degree of temporal dependence, average relative speed and node spatial distribution from them. Mobility Analyzer uses mobility traces generated by MobiSim software [7] in plain text and XML format, analyzes the traces and generates its analysis results in XML format. This software can recognize the mobility pattern of raw mobility traces which are generated by MobiSim and classify them in its supported mobility models using our proposed mobility pattern recognition method.

The rest of the paper is organized as follows: in section 2 we introduce supported mobility models of Mobility Analyzer and a brief explanation of their features. In section 3 we introduce mobility metrics we used in our proposed pattern recognition method and explain our proposed schemes to classify the mobility traces into supported mobility models. Section 4 provides performance evaluation of our proposed method and finally we discuss the conclusion and future works.

2. Supported Mobility Models

In this section we try to briefly introduce supported mobility models of Mobility Analyzer software and explain their parameters, algorithms and properties and also node spatial distribution of each of mobility models.

2.1 Random Waypoint

The Random Waypoint Model was first proposed by Johnson and Maltz [8]. Soon, it became a 'benchmark' mobility model to evaluate the MANET routing protocols, because of its simplicity and wide availability.

The implementation of this mobility model is as follows: as the simulation starts, each mobile node randomly selects one location in the simulation field as the destination. It then travels towards this destination with constant velocity chosen uniformly and randomly from $[0, V_{Max}]$, where the parameter V_{Max} is the maximum allowable velocity for every mobile node [9]. The velocity and direction of a node are chosen independently of other nodes. Upon reaching the destination, the node stops for a duration defined by the 'pause time' parameter. If $T_{Pause} = 0$, this leads to continuous mobility. After this duration, it again chooses another random destination in the simulation field and moves towards it. The whole process is repeated again and again until the simulation ends.

We used MobiSim to generate mobility traces for Random Waypoint Mobility Models. Node Spatial Distribution of Random Waypoint Mobility Model with 20 mobile nodes in 500m*500m simulation area, average speed=20m/s, and maximum pause time=10s, for simulation time=100000sec is shown in figure 1.

As we can see in figure 1 spatial node distribution in Random Waypoint is non-uniform and the node density is maximum at the center region, whereas the node density is almost zero around the boundary of simulation area. This phenomenon is called non-uniform spatial distribution problem in random waypoint.

2.2 Random Direction

The Random Direction model based on similar intuition is proposed by Royer, Melliar-Smith and Moser [10]. This model is able to overcome the non-uniform spatial distribution problem. Instead of selecting a random destination within the simulation field, in the Random Direction model the node randomly and uniformly chooses a direction by which to move along until it reaches the boundary. After the node reaches the boundary of the simulation field it stops with a pause time T, then it randomly and uniformly chooses another direction to travel. This way, the nodes are uniformly distributed within the simulation field.

Node Spatial Distribution of Random Direction Mobility Model with 20 mobile nodes in 500m*500m simulation area, average speed=20m/s, and maximum pause time=10s, for simulation time=100000sec is shown in figure 2.

As we can see spatial node distribution in the simulation field is uniform. In comparison to spatial node distribution shown in figure 1, Random Direction model does not have non-uniform spatial distribution problem.

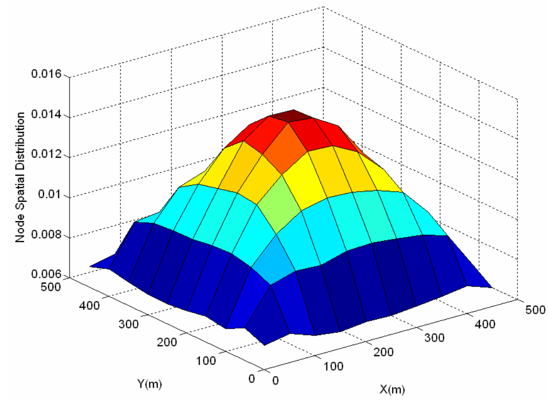

Figure 1: Spatial Node Distribution in Random Waypoint Mobility Model

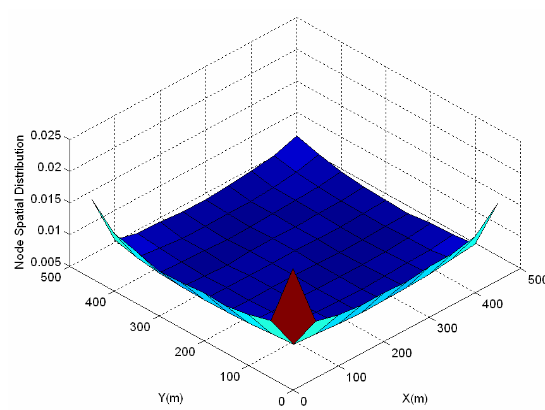

Figure 2: Spatial Node Distribution in Random Direction Mobility Model

2.3 Random Walk

The Random Walk model was originally proposed to emulate the unpredictable movement of particles in physics. It is also referred to as the Brownian Motion. Because some mobile nodes are believed to move in an unexpected way, Random Walk mobility model is proposed to mimic their movement behavior [3]. The Random Walk model has similarities with the Random Waypoint model because the node movement has strong randomness in both models. We can think the Random Walk model as the specific Random Waypoint model with zero pause time.

However, in the Random Walk model, the nodes change their speed and direction at each time interval. For every new interval t, each node randomly and uniformly chooses its new direction $\theta(t)$ from $(0,2\pi]$. In similar way, the new speed follows a uniform distribution from $[0,V_{Max}]$. Therefore, during time interval t, the node moves with the velocity vector $(v(t)\cos(\theta(t)), v(t)\sin(\theta(t)))$. If the node moves according to the above rules and reaches the boundary of simulation field, the leaving node is bounced back to the simulation field with the angle of $\theta(t)$ or $\pi-\theta(t)$, respectively. This effect is called border effect [11] or reflection rule.

Node Spatial Distribution of Random Walk Mobility Model with 20 mobile nodes in 500m*500m simulation area, average speed=20m/s, for simulation time=100000sec is shown in figure 3. As we can see spatial node distribution in the simulation field is uniform.

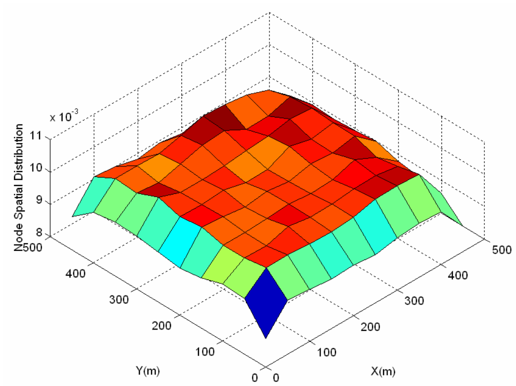

Figure 3: Spatial Node Distribution in Random Walk Mobility Model

2.4 Manhattan

The Manhattan mobility model is usually used to emulate the movement pattern of mobile nodes on streets defined by maps. It can be useful in modeling movement in an urban area where a pervasive computing service between portable devices is provided [3].

Maps are used in this model. The map is composed of a number of horizontal and vertical streets. Each street has two lanes for each direction (north and south direction for vertical streets, east and west for horizontal streets). The mobile node is allowed to move along the grid of horizontal and vertical streets on the map. At an intersection of a horizontal and a vertical street, the mobile node can turn left, right or go straight. This choice is probabilistic: the probability of moving on the same street is 0.5, the probability of turning left is 0.25 and the probability of turning right is 0.25. The velocity of a mobile node at a time slot is dependent on its velocity at the previous time slot. Also, a node's velocity is restricted by the velocity of the node preceding it on the same lane of the street.

If two mobile nodes on the same Manhattan lane are within the safety distance (SD), the velocity of the following node cannot exceed the velocity of preceding node.

The inter-node and intra-node relationships involved are:

If node j is ahead of node i in its lane then:

$$\begin{aligned} &|\vec{V}_i(t+1)| = |\vec{V}_i(t)| + random() * |\vec{a}_i(t)| \\ &\forall i, \forall j, \forall t, D_{i,j}(t) \le SD \Rightarrow |\vec{V}_i(t)| \le |\vec{V}_j(t)| \end{aligned} \qquad (1)$$

The map used in our simulations for Manhattan mobility model is shown in figure 4.

Node Spatial Distribution of Manhattan Model with 20 mobile nodes in 500m*500m simulation area and average speed=20m/s, for simulation time=100000sec is shown in figure 5.

As we can see spatial node distribution in the simulation field is not uniform and we have much more node density in intersections in comparison with lanes. In proposed scheme for simulation of this model in MobiSim, nodes stop for a random pause time between zero and a specified maximum pause time in intersections and choose another destination so the node density in intersections is much more. We added this pause time to Manhattan Model to simulate behavior of cars in intersections due to the traffic lights and pedestrian crossings.

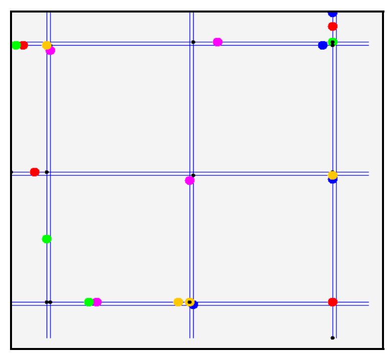

Figure 4: Manhattan Mobility Model Map

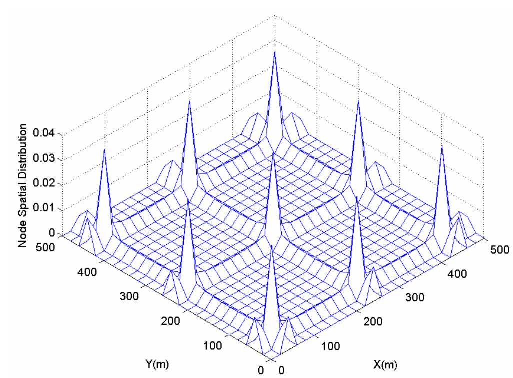

Figure 5: Spatial Node Distribution in Manhattan Mobility Model

2.5 Gauss-Markov

The Gauss-Markov Mobility Model was designed to adapt to different levels of randomness via one tuning parameter [12].

Initially each Mobile Node is assigned a current speed and direction. At fixed intervals of time *n*; node's movement occurs by updating the speed and direction of each mobile node. Specifically, the value of speed and direction at the *nth* instance is calculated based upon the value of speed and direction at the $(n-1)^{st}$ instance and a random variable using the following equation:

$$s_n = as_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha)^2}\, s_{x_{n-1}}$$
$$d_n = ad_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha)^2}\, d_{x_{n-1}} \qquad (2)$$

Where s_n and d_n are the new speed and direction of the mobile node at time interval *n*; α , where $0 \le \alpha \le 1$, is the tuning parameter used to vary the randomness; $\bar{s}$ and $\bar{d}$ are constants representing the mean value of speed and direction as $n \to \infty$; s_{n-1} and d_{n-1} are random variables from a Gaussian distribution. Totally random values or Brownian motion are obtained by setting $\alpha = 0$ and linear motion is obtained by setting $\alpha = 1$ [11]. Intermediate levels of randomness are obtained by varying the value of α between 0 and 1.

At each time interval the next location is calculated based on the current location, speed, and direction of movement. Specifically, at time interval *n*, an Mobile node's position is given by the equations:

$$x_n = x_{n-1} + s_{n-1}\cos d_{n-1}$$
$$y_n = y_{n-1} + s_{n-1}\sin d_{n-1} \qquad (3)$$

Where (x_n, y_n) and (x_{n-1}, y_{n-1}) are the *x* and *y* coordinates of the mobile node's position at the n^{th} and $(n-1)^{st}$ time intervals, respectively, and s_{n-1} and d_{n-1} are the speed and direction of the mobile node, respectively, at the $(n-1)^{st}$ time interval [2,12].

The model parameters are memory factor (α), and random amplitude which is represented with σ. If we set the memory factor to 1 the model behavior becomes like Random Walk and if we set it to zero model behavior becomes like Brownian Motion.

Figure 6 shows traveling pattern of 5 nodes with $a = 0.1$, $\sigma = 5$ and average speed=20m/s.

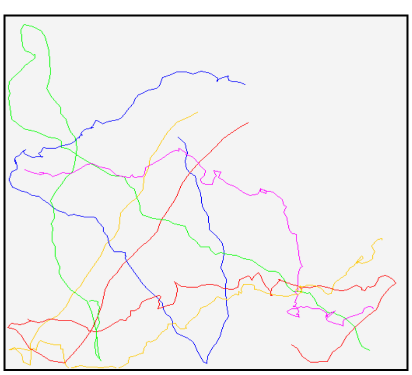

Figure 6: Traveling pattern of Mobile Nodes in Gauss-Markov Mobility Model

Node Spatial Distribution of Gauss-Markov Model with 20 mobile nodes in 500m*500m simulation area and average speed=20m/s, for simulation time=100000sec is shown in figure 7.

As we can see the node spatial distribution in this mobility model is similar to Random Walk node spatial distribution.

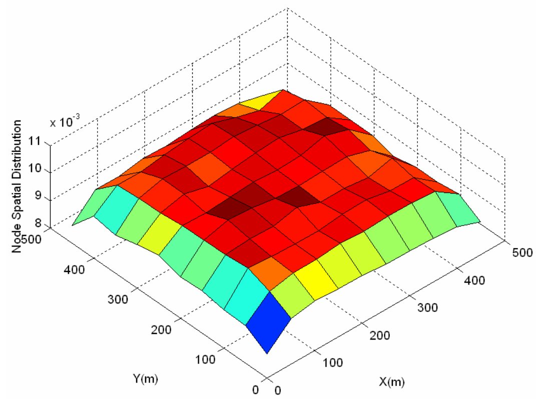

Figure 7: Spatial Node Distribution in Gauss-Markov Mobility

2.6 Reference Point Group Model

In line with the observation that the mobile nodes in MANET tend to coordinate their movement, the Reference Point Group Mobility (RPGM) Model is proposed in [13]. One example of such mobility is that a number of soldiers may move together in a group or platoon. Another example is during disaster relief where various rescue crews (e.g., firemen, policemen and medical assistants) form different groups and work cooperatively.

In the RPGM model, each group has a center, which is either a logical center or a group leader node. For the sake of simplicity, we assume that the center is the group leader. Thus, each group is composed of one leader and a number of members. The movement of the group leader determines the mobility behavior of the entire group. Initially, each member of the group is uniformly distributed in the neighborhood of the group leader. Subsequently, at each instant, each node has a speed and direction that is derived by randomly deviating from that of the group leader. The movement in group mobility can be characterized as follows:

$$\begin{cases} |V_{member}(t)| = |V_{leader}(t)| + random() * SDR * \max speed \\ \theta_{member}(t) = \theta_{leader}(t) + random() * ADR * \max angle \end{cases} \qquad (4)$$

SDR is the Speed Deviation Ratio and ADR is the Angle Deviation Ratio. SDR and ADR are used to control the deviation of the velocity (magnitude and direction) of group members from that of the leader. So model parameters will be ADR, SDR, initial distance (members initial distance from the leader node), and group size which determines number of group nodes.

We used Random Walk Mobility Model for motion behavior of leader node in each group therefore node spatial distribution in RPGM model is similar to Random Walk Mobility Model because the leader node in each node travels with Random Walk Mobility Model.

Figure 8 shows traveling pattern of 2 groups with 5 nodes with average speed=20m/s, SDR=0.05, ADR=0.05, using RPGM Mobility Model.

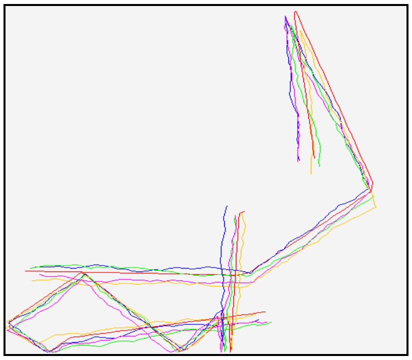

Figure 8: Traveling pattern of Mobile Nodes in RPGM Mobility Model

3. Mobility Pattern Recognition

In this section we introduce used mobility metrics in our learning based classification method and our proposed scheme for pattern recognition of mobility traces in mobile Ah-hoc networks.

Before formally defining the metrics, we introduce some basic terminology that will be used later in the paper:

1. $\vec{V}_i(t)$: velocity vector of node i at time t.
2. $\left|\vec{V}_i(t)\right|$: speed of node i at time t.
3. $\theta_i(t)$: angle made by $\vec{V}_i(t)$ at time t with the Xaxis.
4. $\vec{a}_i(t)$: acceleration vector of node i at time t.
5. $x_i(t)$: X co-ordinate of node i at time t.
6. $y_i(t)$: Y co-ordinate of node i at time t.
7. $D_{i,j}(t)$: Euclidean distance between nodes i and j at t.
8. $RD(\vec{a}(t),\vec{b}(t'))$: relative direction (RD) (or cosine of the angle) between the two vectors $\vec{a}(t)$, $\vec{b}(t')$ is given by: $\frac{\vec{a}(t).\vec{b}(t')}{\left|\vec{a}(t)\right|*\left|\vec{b}(t')\right|}$.
9. $SR(\vec{a}(t),\vec{b}(t'))$: speed ratio (SR) between the two vectors $\vec{a}(t)$, $\vec{b}(t')$ is given by $\frac{\min(\left|\vec{a}(t)\right|,\left|\vec{b}(t)\right|)}{\max(\left|\vec{a}(t)\right|,\left|\vec{b}(t')\right|)}$.

3.1 Mobility metrics

We propose these metrics to differentiate the various mobility patterns used in our study [3]. The basis of differentiation is the extent to which a given mobility pattern captures the characteristics of spatial dependence, temporal dependence and geographic restrictions. In addition to these metrics, we also use the relative speed metric that differentiates mobility patterns based on relative motion. This metric was first proposed in [14].

Degree of Spatial Dependence:

It is a measure of the extent of similarity of the velocities of two nodes that are not too far apart. Formally,

$$D_{spatial}(i,j,t) = RD(\vec{v}_i(t),\vec{v}_j(t)) * SR(\vec{v}_i(t),\vec{v}_j(t)). \quad (5)$$

The value of $D_{spatial}(i,j,t)$ is high when the nodes i and j travel in more or less the same direction and at almost similar speeds. However, $D_{spatial}(i,j,t)$ decreases if the relative direction or the speed ratio decreases.

As it is rare for a node's motion to be spatially dependent on a far off node, we add the condition that

$$D_{i,j}(t) > c_1 * R \Rightarrow D_{spatial(i,j,t)=0}, \quad (6)$$

where $c_1 > 0$ is a constant which will be determined during our experiments.

Average Degree of Spatial Dependence:

It is the value of $D_{spatial}(i,j,t)$ averaged over node pairs and time instants satisfying certain condition. Formally,

$$\overline{D}_{spatial} = \frac{\Sigma_{t=1}^{T}\Sigma_{i=1}^{N}\Sigma_{j=i+1}^{N} D_{spatial}(i,j,t)}{P}, \quad (7)$$

where P is the number of tuples (i,j,t) such that $D_{spatial}(i,j,t) \neq 0$. Thus, if mobile nodes move independently of one another, then the mobility pattern is expected to have a smaller value for $D_{spatial}$. On the other hand, if the node movement is co-ordinated by a central entity, or influenced by nodes in its neighborhood, such that they move in similar directions and at similar speeds, then the mobility pattern is expected to have a higher value for $D_{spatial}$.

Degree of temporal dependence:

It is a measure of the extent of similarity of the velocities of a node at two time slots that are not too far apart. It is a function of the acceleration of the mobile node and the geographic restrictions. Formally,

$$D_{temporal}(i,t,t') = RD(\vec{v}_i(t),\vec{v}_i(t')) * SR(\vec{v}_i(t),\vec{v}_i(t')). \quad (8)$$

The value of $D_{temporal}(i,t,t')$ is high when the node travels in more or less the same direction and almost at the same speed over a certain time interval that can be defined. However, $D_{temporal}(i,t,t')$ decreases if the relative direction or the speed ratio decreases.

Arguing in a way similar to that for $D_{spatial}(i,j,t)$, we have the following condition:

$$\left|t-t'\right| > c_2 \Rightarrow D_{temporal}(i,t,t') = 0, \quad (9)$$

where $c_2 > 0$ is a constant which will be determined during our experiments.

Average degree of temporal dependence:

It is the value of $D_{temporal}(i,t,t')$ averaged over nodes and time instants satisfying certain condition. Formally,

$$\overline{D}_{temporal} = \frac{\Sigma_{i=1}^{N}\Sigma_{t=1}^{T}\Sigma_{t'=1}^{T} D_{temporal}(i,t,t')}{P}. \quad (10)$$

where P is the number of tuples (i,t,t') such that $D_{temporal}(i,t,t') \neq 0$ Thus, if the current velocity of a node is completely independent of its velocity at some previous time step, then the mobility pattern is expected to have a smaller value for $\overline{D}_{temporal}$. However, if the current velocity is strongly dependent on the velocity at some previous time step, then the mobility pattern is expected to have a higher value for $\overline{D}_{temporal}$.

3.1.1 Relative Speed (RS):

We use the standard definition from physics, i.e.

$$RS(i,j,t) = \left|\vec{V}_i(t) - \vec{V}_j(t)\right|. \quad (11)$$

As in the case of $D_{spatial}(i,j,t)$, we add the following condition:

$$D_{i,j}(t) > c_3 * R \Rightarrow RS(i,j,t) = 0, \quad (12)$$

where $c_3 > 0$ is a constant which will be determined during our experiments.

3.1.1.1 Average Relative Speed:

It is the value of $RS(i,j,t)$ averaged over node pairs and time instants satisfying certain condition. Formally,

$$\overline{RS} = \frac{\Sigma_{i=1}^{N}\Sigma_{j=1}^{N}\Sigma_{t=1}^{T} RS(i,j,t)}{P}. \quad (13)$$

where P is the number of tuples (i,j,t) such that $RS(i,j,t) \neq 0$.

3.2 Proposed Classification Method

Our proposed method for classification of mobility models is divided into 2 main phases: Training phase and Classification phase. This method is a supervised learning based classification method.

3.2.1.1 Training Phase:

In training phase we analyze 20 instances of known mobility traces generated by MobiSim for each of mobility models and extract values of 3 mobility metrics for each mobility trace. These mobility metrics are: Average Degree of Spatial Dependence, Average Degree of Temporal Dependence and Average Relative Speed. Then we put them in a three-dimentional matrix and we calculate 3D centroid for each of the mobility model classes. Therefore in this phase the method learns mobility metrics of each of the mobility models by calculating the 3D centroid of each of mobility model classes.

3.2.1.2 Classification Phase

In classification phase we get mobility traces with unknown mobility model and attempt to classify each of them into one of the supported mobility models. We get 120 unknown traces as classification instances from each of the supported mobility models (20 instances for each one). For each of the unknown traces the method calculates values of 3 mobility metrics (Average Degree of Spatial Dependence, Average Degree of Temporal Dependence and Average Relative Speed) and calculates 3D centoid for each of them. Then the method calculates Euclidean distance of the centroid of each of the unknown traces to the centroid of each of the mobility model classes (calculated in training phase). Then the method classifies the unknown trace into the nearest class (lowest Euclidean distance between its centroid and centroids of classes). Using this method each of the unknown traces would be classified into only one of the mobility classes (mobility models).

Our simulation results show significant performance of this method to recognize the mobility model of all unknown traces into one of the supported mobility models classes.

4. Simulation Results

We used MobiSim software to generate 20 known and 20 unknown mobility traces for each mobility model with 20 nodes, for 10000sec, with average speed=40m/s, in a 500*500 simulation region and the same simulation configurations as we mentioned for simulations in section 2, then we used Mobility Analyzer software to calculate mobility metrics of each of the mobility traces. These features (mobility metrics) are organized in 3 feature vector (Average Degree of Spatial Dependence, Average Degree of Temporal Dependence and Average Relative Speed). We considered $c_1, c_3 = 100$ and $c_2 = 1$ (Equations 6,9,12) for calculating these 3 mobility metrics. Figures 9, 10, 11 show 2 of the mobility metrics and their relation in each of the mobility traces in a two dimensional feature vector.

As we can see mobility metrics vary in different mobility models. We discuss each of the mobility models and their features as following.

In RPGM mobility model nodes move near each other as a group with almost similar speed and direction angle therefore this mobility model has very high degree of spatial dependence because there is high similarity in motion of nodes in a group. But in comparison with other mobility models it has lowest relative speed because each of the nodes in a group chooses a random speed and direction according the speed and direction of the group leader so the speed and direction of each of group members are almost same as each other so the value of relative speed for two of group nodes is very low so in average the relative speed of this model would be very low. This model has high temporal dependence because in each instance of time the motion of a mobile node in this mobility model is similar to its motion in previous instance of time.

The Random Waypoint and Random Direction models almost have same mobility metrics because the motion manner in these two mobility models is almost the same. The main difference in their motion manner is: in Random Direction the nodes stop on the regions of simulation area for a random time called pause time, but in Random Waypoint model nodes may stop in any point of simulation region. This can cause a problem called non-uniform node spatial distribution in Random Waypoint. This phenomenon causes a small difference in average relative speed of these 2 mobility models. But other mobility metrics in this mobility models are almost the same therefore the only useful mobility metric which can be used to separate these to mobility models is relative speed. Because of existing of pause time in motion of nodes in these 2 mobility models, movements of mobile nodes have low similarity therefore they have high average relative speed.

Gauss-Markov and Random Walk almost have similar mobility metrics. Both of them have medium average relative speed and low average spatial dependence. Lack of similarity of speed and direction in motion of nodes caused almost high average relative speed and low average spatial dependence in these 2 mobility models. Average temporal dependence is the only mobility metrics that can separate these 2 mobility models. Because of constant speed and direction of a mobile node during a transition in Random Walk mobility model this model has very high average temporal dependence. But in Gauss-Markov mobility models speed and direction of a mobile node changes frequently during a period of time so it has low average temporal dependence.

Manhattan mobility models has low average relative speed and spatial dependence because nodes in this mobility model almost move with similar speed and angle because they must consider the safe distance between each other so neighboring nodes would have almost same speed and angle. This model also has low average temporal dependence because of existing negative and positive acceleration in motion of nodes and pause time in intersections and behind the traffic lights.

In the training phase the mobility analyzer calculates each of the mobility metrics for each of known mobility traces and calculated a 3D centroid for each of the mobility models.

In the classification phase mobility analyzer calculated mobility metrics for each of the unknown given mobility traces and calculates 3D centroid of these 3 mobility metrics then it calculated Euclidian distance between each of the calculated centroids to centroids of each of the mobility classes and assigns each trace in nearest class. With this method we used 120 known mobility traces (20 trace for each model) for training phase and classified 120 unknown mobility traces into 6 mobility models in classification phase.

The classification result is shown in figure 12. As we can see our proposed method can successfully separate features of 6 mobility models and classify unknown mobility traces into right mobility model classes.

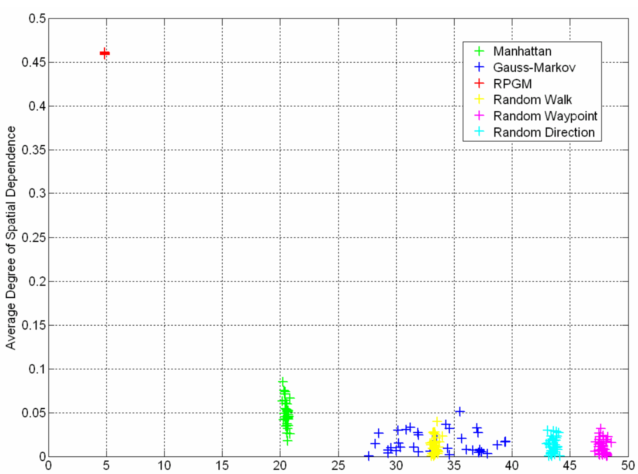

Figure 9: Calculated values of 2 mobility metrics for mobility trace samples

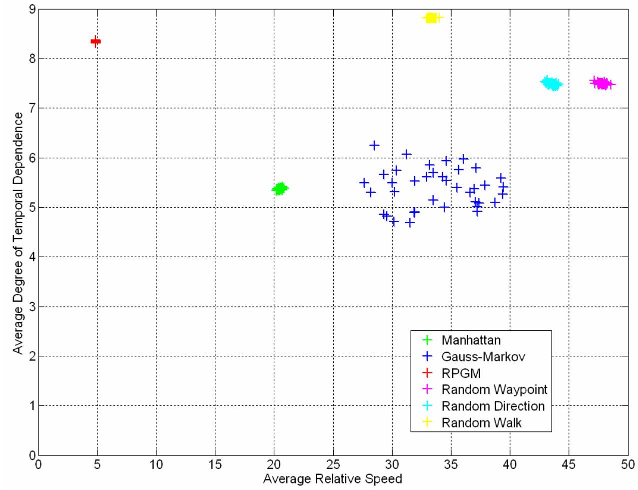

Figure 10: Calculated values of 2 mobility metrics for mobility trace samples

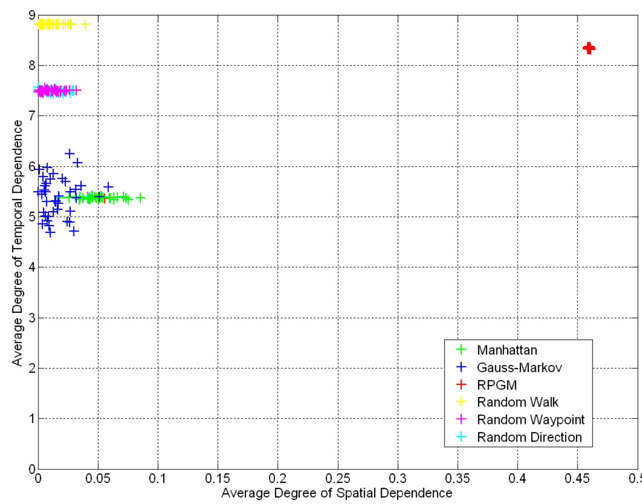

Figure 11: Calculated values of 2 mobility metrics for mobility trace samples

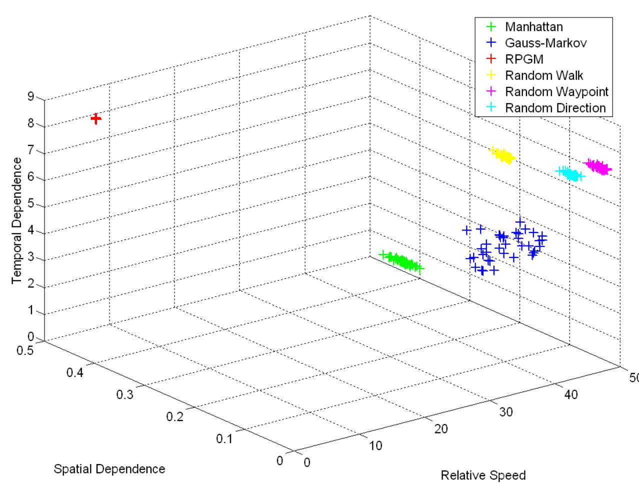

Figure 12: Classified mobility traces into 6 mobility model classes

5. CONCLUSION AND FUTURE WORKS

In this paper we introduced a new method for classification and pattern recognition of mobility models in mobile Ad-hoc networks. This method uses a simple learning based classification method to recognize the existing mobility model in unknown mobility traces collected from real motion of mobile Ad-hoc nodes or mobility traces generated by mobility simulators. Our simulation results show significant performance of this method to recognize the mobility model of all unknown traces into one of the supported mobility model classes.

For the future works we are working on our proposed method to make it more efficient in classification of other mobility models like Freeway, Probabilistic Random Walk, and other mobility models. Also we are working on other mobility metrics like Geographical Restrictions. In addition we are working on other pattern recognition methods like unsupervised learning based clustering methods to make our mobility pattern recognition method more useful and accurate with support of various mobility models.

6. ACKNOWLEDGEMENTS

The authors would like to thank members of Sharif Digital Media Lab (DML) for their invaluable cooperation.

This work was supported by Sharif Advanced Information and Communication Technology Center (AICTC) & Iran Telecommunication Research Center (ITRC).

7. REFERENCES

[1] C. Siva Ram Murthy and B.S. Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Prentice Hall, 2004.

[2] Tracy Camp, Jeff Boleng, Vanessa Davis, A Survey of Mobility Models for Ad Hoc Network Research, Wireless Communication & Mobile Computing (WCMC): Special issue on Mobile AdHoc Networking: Research, Trends and Applications, vol 2, no 5, pp. 483-502, 2002.

[3] Fan Bai, Narayanan Sadagopan, Ahmed Helmy, The Important framework for analyzing the Impact of mobility on performance Of Routing Protocols for Adhoc Networks, Elsevier Journal of Ad Hoc Networks, 2003, pp. 383-403.

[4] W. Su, S-J. Lee, and M. Gerla, Mobility Prediction and Routing in Ad hoc Wireless Networks, International Journal of Network Management, Vol. 11, No. 1, pp.3-30, 2001.

[5] Zeeshan Hameed Mir, Deepesh Man Shrestha, Geun-Hee Cho, Young-Bae Ko, Mobility Aware Distributed Topology Control for Mobile Multi-hop Wireless Networks, ICOINS 2006, LNCS 3961, pp. 257-266 2006.

[6] Richard O. Duda, Peter E. Hart, David G. Stork (2001) *Pattern classification* (2nd edition), Wiley, New York

[7] S. M. Mousavi, M. Moshref, H. R. Rabiee, A.Dabirmoghaddam, "MobiSim: a Framework for Simulation of Mobility Models in Mobile Ad-Hoc Networks, http://ce.sharif.edu/~sm_mousavi/mobisim.html

[8] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols", in Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking(Mobicom98), ACM, October 1998.

[9] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, Advances in network simulation, in IEEE Computer, vol. 33, no. 5, May 2000, pp. 59--67.

[10] E. M. Royer, P. M. Melliar-Smith, and L. E. Moser. An Analysis of the Optimum Node Density for Ad hoc Mobile Networks, in Proceedings of the IEEE International Conference on Communications(ICC), Helsinki, Finland, June 2001.

[11] C. Bettstetter and C. Wagner. The Spatial Node Distribution of the Random Waypoint Mobility Model, in Proc. German Workshop on Mobile Ad-Hoc Networks (WMAN), Ulm, Germany, GI Lecture Notes in Informatics, no. P-11, pp. 41-58, Mar 25-26, 2002.

[12] B. Liang and 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.

[13] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, A group mobility model for ad hoc wireless networks, in Proc. ACM Intern. Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM), August 1999.

[14] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, M. Degermark, Scenario-based performance analysis of routing protocols for mobile ad-hoc networks, in: International Conference on Mobile Computing and Networking (MobiCom_99), 1999, pp. 195–206.