Impact of Mobility Prediction on the Temporal Stability of MANET Clustering Algorithms*

Aravindhan Venkateswaran†, Venkatesh Sarangan‡, Natarajan Gautam§1, Raj Acharya†

†Department of Comp. Sci. & Engr.
Pennsylvania State University
University Park, PA, 16802
+1 (814) 865-9505
{venkates, acharya}@cse.psu.edu

‡Comp. Sci. Department
Oklahoma State University
Stillwater, OK, 74078
+1 (405) 744-5672
saranga@cs.okstate.edu

§Department of Ind. Engr.
Texas A&M University
College Station, TX, 77843
+1 (979) 845-5458
gautam@iemail.tamu.edu

ABSTRACT
Scalability issues for routing in mobile ad hoc networks (MANETs) have been typically addressed using hybrid routing schemes operating in a hierarchical network architecture. Several clustering schemes have been proposed to dynamically identify and maintain hierarchy in MANETs. To achieve significant performance gains, it is important that the underlying clustering scheme is able to identify stable clusters such that the cost associated with maintaining the clustered architecture is minimized. In this paper, we study the impact of mobility prediction schemes on the temporal stability of the clusters obtained using a mobility-aware clustering framework. We investigate the performance of the prediction schemes with respect to Gauss-Markov, Random Waypoint, and Reference Point Group mobility models under varying network and mobility conditions. Our results indicate that while mobility prediction significantly improves temporal stability of the clusters, an accurate mobility tracking algorithm need not always lead to an accurate mobility prediction scheme.

Categories and Subject Descriptors
C.2.1 [**Computer Communication Networks**]: Network Architecture and Design – wireless communication; I.6.m [**Simulation and Modeling**]: Miscellaneous.

General Terms
Algorithms, Performance.

Keywords
Ad hoc networks, clustering, mobility prediction.

1. INTRODUCTION
Advances in wireless communication and the widespread use of mobile and handheld devices has resulted in an increasing popularity of mobile ad hoc networks (MANETs) – networks that consist of a collection of geographically distributed nodes that communicate with each other over a wireless medium. MANETs do not have a fixed infrastructure in place and communication takes place through wireless links among mobile hosts. Moreover, limited transmission range of nodes often results in a multi-hop communication scenario, where several hosts may need to relay a packet before it reaches its final destination [1].

The mobility of nodes coupled with the transient nature of wireless media often results in a highly dynamic network topology. This makes the task of routing in an ad hoc network more difficult when compared to a wired network. Routing protocols in ad hoc networks can be broadly classified into two types: reactive and proactive. However, a flat structure exclusively based on proactive or reactive routing does not perform well in large dynamic MANETs [2]. Consequently, a hierarchical architecture is essential for enhancing the routing performance in large-scale MANETs [6]. Unlike wired networks, it is essential to have a dynamic scheme to identify and maintain a hierarchy in an ad hoc network. A clustering scheme in MANET organizes the mobile nodes in the network into virtual groups known as clusters, based on certain criteria. A cluster typically consists of a cluster head and its member nodes. A clustered architecture provides an effective means for topology management, since topology changes local to a cluster need not be propagated across the whole network. Also, typically only the cluster heads are involved in route discovery which significantly reduces the control overhead associated with the routing process. There are many papers in the literature which focus on presenting an effective and efficient clustering scheme for MANETs. A survey of such clustering schemes is presented in [6].

An important argument against introducing a hierarchy in an ad hoc network is that, the overhead associated with maintaining the hierarchy may outweigh its potential benefits. For instance, the membership of a cluster can frequently change as the nodes move in and out of the range of the cluster heads. Hence, the clustering process may have to be run frequently creating additional computational overhead. Thus, it is important for a clustering scheme to identify stable clusters by minimizing the frequency of membership changes. Since it is not possible to partition the network into clusters which do not change at all, we need to

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PE-WASUN'05, October 10–13, 2005, Montreal, Quebec, Canada.
Copyright 2005 ACM 1-59593-182-1/05/0010...$5.00.

*This work is supported in part by the NSF ITR grant 0219747.

[1] This research was performed when the author was at Pennsylvania State University.

design clustering schemes that exhibit temporal stability (i.e., identify clusters with a long life-time) in order to effectively apply hierarchical routing techniques. One way to achieve this is to use mobility prediction to identify clusters consisting of nodes that show some temporal similarity in their mobility patterns. Such an approach can also help in introducing a notion of quasi-stability in an otherwise unstable network topology. Several schemes that utilize mobility prediction for clustering in MANETs have been proposed [10], [13]. However, to the best of our knowledge, there is no work in the literature that analyzes the impact of different mobility prediction schemes on the stability of the clusters under varying network and mobility conditions. Such a study can provide an impartial view on the efficacy of mobility prediction schemes and help researchers in making an informed selection of prediction models appropriate to their network environment.

In this paper, we compare the performance of two generic mobility prediction algorithms: (1) Mobility Prediction using the Link Expiration Time [7] and (2) Mobility Prediction using Linear Autoregressive Models [8]. We restrict our analysis to these two prediction schemes as, unlike other schemes in the literature, only these two schemes are independent of the underlying model that defines the node mobility. Further, these two schemes do not require the network to have any well-known virtual cluster centers [10] or waypoints [12] and hence are independent of the network architecture as well. In order to analyze the performance of these prediction schemes, we propose a simple framework for a mobility prediction-based clustering scheme that aims to provide temporal guarantees on link availability between nodes. Simulations are performed to evaluate the temporal stability of the clusters defined in terms of the metrics – Cluster Survival Time, Cluster Residence Time and Number of Reaffiliations. We compare the results with a clustering framework that is mobility sensitive, but does not utilize mobility prediction such as WCA (Weighted Clustering Algorithm [5]), in order to better understand the efficacy of mobility prediction.

The rest of the paper is organized as follows. In section 2, we present a summary of significant contributions in the areas of clustering and mobility prediction algorithms for MANETs. The proposed prediction based clustering framework is described in section 3. In section 4, we present a detailed experimental analysis on the performance of the two prediction algorithms. Our conclusions are presented in section 5.

2. RELATED WORK

Several existing approaches utilize mobility prediction schemes to design efficient routing protocols for MANETs. In [7], William et al. compute the Link Expiration Time (LET) to predict the duration of a wireless link between two nodes in the network. Their approach assumes that the direction and speed of motion of the mobile nodes does not change during the prediction interval. This simple mechanism is then applied to enhance the reliability of existing unicast and multicast ad hoc routing protocols. In [12], an offline algorithm is proposed to predict link durations in the worst-case scenario for an urban MANET. The predicted link durations are then utilized to design a routing algorithm which finds minimum cost paths with required duration guarantees.

Dynamic clustering in ad hoc networks has also been extensively studied in the literature. Several distributed clustering algorithms for MANETs have been proposed. While some schemes try to balance the energy consumption for mobile nodes, others aim to minimize the clustering-related maintenance costs. Combined metrics based clustering schemes take a number of metrics into account for cluster configuration. The Weighted Clustering Algorithm (WCA) [7], is one such scheme, where four parameters are considered in the clusterhead election procedure which are representative of the degree, transmission power, mobility, and battery power of the mobile nodes. Such a scheme can flexibly tune the parameters to suit to different scenarios. Reference [6] presents a comprehensive survey of various MANET clustering schemes that exist in the literature.

In this paper, we consider a clustering framework that utilizes mobility prediction for identifying temporally stable clusters. One of the earliest approaches to utilize mobility prediction in clustering was the Distributed Dynamic Clustering Algorithm proposed by McDonald et al. in [13]. DDCA employs the (α, t)-clustering scheme, wherein generated clusters have the property that the path between any two nodes in the cluster will be available for time t seconds with a probability of at least α. Though this prediction scheme gives such a strong characterization, it is applicable only for those scenarios where the nodes follow a random walk mobility model. A (p,t,d)-clustering model is proposed in [10] which is based on mobility prediction derived from data compression techniques. The clustering is achieved by dividing the network into circular regions referred to as virtual clusters. A virtual cluster becomes an actual cluster whenever mobile nodes exist in it.

In [9], Zaidi et al. propose a two tier composite model of node mobility that captures the group behavior in a mobile ad hoc network. They use a first order autoregressive (AR-1) mobility model, originally proposed in [8] to track the mobility state evolution of an individual node. Their results indicate that with appropriate model parameters, AR-1 model is capable of representing a wide range of mobility patterns. A dynamic scheme to automatically recognize group mobility behavior in MANETs is also proposed in [9]. Though this could be considered as a clustering scheme, there is no explicit mobility prediction involved in the approach. Group mobility is identified by means of a correlation index test between the estimated mobility states of the individual nodes.

In this paper, we build a framework for a mobility-prediction based clustering algorithm to analyze the performance of two generic mobility prediction schemes: (1) Mobility Prediction using Link Expiration Time and (2) Mobility Prediction using Autoregressive Models. These mobility prediction schemes are evaluated through the clustering framework under three different mobility models: (1) Gauss-Markov mobility model, (2) Random Waypoint mobility model and (3) Reference Point Group Mobility (RPGM) model. We also compare the results against the WCA which is a mobility-aware clustering framework that does not utilize mobility prediction.

3. A PREDICTIVE CLUSTERING FRAMEWORK

In this section, we describe a simple mobility prediction-based clustering scheme that aims to provide temporal guarantees on the availability of links between mobile nodes. We assume that every node in the network has a unique id, which could be the node's IP address or a combination of one or more ids. Every node is also aware of its geographical location and mobility information either via GPS or using mechanisms such as [15] that use signal strength measurements.

3.1 Terminologies

We model a mobile ad hoc network as an undirected graph $G = (V, E)$, where V is the set of all mobile nodes, and E is the set of the undirected links between them. A link (u, v) is said to exist between nodes u and v, if and only if both are in the transmission ranges of each other. Let N_j denote the set of all nodes in the one hop neighborhood of node j. Cluster C_j is a set of nodes such that $C_j = \{u \mid u \in N_j\}$ for some $j \in V$. In addition, the members in the set C_j satisfy certain constraints, which will be discussed later on. Node j is called as the *seed* or the *clusterhead* of the cluster C_j. Other nodes in the cluster are referred to as the *member nodes*. We define the *residence time,* τ_j^k of node k, as the amount of time k spends being a part of cluster j, before getting affiliated to another cluster. A node can get affiliated to another cluster if it moves outside the range of a clusterhead.

3.2 Algorithm Specification

The proposed clustering framework aims to partition the network into clusters consisting of nodes that exhibit temporal similarity in their mobility pattern. The design of this framework is motivated by the *(α,t)*-clustering scheme originally proposed in [13]. Specifically, in order to join a cluster C_j, a node i must satisfy the following conditions:

1. $i \in N_j$
2. $\tau_j^k \geq T_j$, where T_j is the admission criteria associated with the cluster C_j.

A clusterhead uses the mobility prediction scheme to check if a given node can satisfy the admission criteria, before admitting the node in its cluster. The algorithm is designed to run continuously and asynchronously on each active node in the network, avoiding the need for a centralized control or periodic reclustering.

Every cluster head periodically broadcasts HELLO messages to the nodes in its neighborhood. The HELLO message contains the clusterhead's admission criteria, location, and mobility profile. Upon activation, a node rapidly seeks to join a feasible cluster based on the advertisements from the neighboring clusterheads. If there are multiple feasible clusters, the node joins the cluster with maximum number of member nodes. If no clusters are detected, the node itself becomes a clusterhead and starts broadcasting periodic HELLO messages. Adjacent un-clustered nodes are prevented from each forming a new cluster by forcing nodes with higher identifiers to back off and try again as described in [13].

Cluster maintenance is performed based on a soft-state approach. Each member node maintains timers that are reset on receiving the periodic HELLO messages from their cluster heads. If a member node does not receive the HELLO message from its clusterhead within a stipulated time, the associated timer goes off to indicate one of two possibilities: (1) the member node has moved out of the clusterhead's transmission range, or (2) the clusterhead has died. In both these cases, the member node tries to find out if there are any other feasible clusters in its neighborhood that it can join. If none is available, it becomes a clusterhead on its own and starts broadcasting periodic HELLO messages.

Similar to the HELLO message, the member nodes in a cluster send periodic MEMBER_UPDATE messages to the clusterhead. Every clusterhead proactively maintains the location and mobility information of all the nodes in its cluster. If a MEMBER_UPDATE message is not received within the stipulated time, it is assumed that the node has moved out of the transmission range of the clusterhead and is no longer considered a part of the cluster.

3.3 Mobility Prediction Schemes

In this section, we present an overview of the two mobility prediction schemes considered in this paper. The choice of these prediction schemes is due to the fact that unlike other schemes in the literature, both these schemes are independent of the underlying model that defines the node mobility and of the network architecture.

Link Expiration Time: The Link Expiration Time (LET) is a simple prediction scheme that determines the duration of a wireless link between two mobile nodes by assuming that their speed and direction of movement remains constant. Let the location of node i and node j at time t be given by (x_i, y_i) and (x_j, y_j). Also, let $\vec{v}_i$ and $\vec{v}_j$ be the speeds, and θ_i and θ_j be the directions of the nodes i and j respectively. If the transmission range of the nodes is r, then the Link Expiration Time, D_t, of the link between the two nodes, as defined in [7], is given by

$$D_t = \frac{-(ab+cd) + \sqrt{(a^2+c^2)r^2 - (ad-bc)^2}}{a^2+c^2} \quad \ldots(1)$$

where

$$a = v_i \cos\theta_i - v_j \cos\theta_j$$

$$b = x_i - x_j$$

$$c = v_i \sin\theta_i - v_j \sin\theta_j$$

$$d = y_i - y_j$$

The LET gives an *upper bound* on the estimate of the residence time of a node in a cluster. In the proposed clustering framework, when LET-based prediction is used, a node is allowed to join a cluster only if the predicted LET of the link between the node and the clusterhead is greater than the cluster's admission criteria.

Linear First Order Autoregressive Model: The linear first order autoregressive (AR-1) model, as defined in [8], has been shown to effectively track the movement of a mobile node irrespective of the underlying mobility model. In an AR-1 model, the mobility state of a node at time *n* is defined by the column vector

$$s_n = \left[x_n, \dot{x}_n, y_n, \dot{y}_n, \ddot{x}_n, \ddot{y}_n\right]$$

where x_n and y_n specify the position, $\dot{x}_n$ and $\dot{y}_n$ specify the velocity, and $\ddot{x}_n$ and $\ddot{y}_n$ specify the acceleration of the mobile node in the *x* and *y* directions in a two-dimensional grid. The AR-1 model for the mobility state $\boldsymbol{s_n}$ of a node is given by:

$$s_{n+1} = As_n + w_n \qquad \ldots (2)$$

where A is a 6×6 transformation matrix, the vector w_n is a 6×1 discrete-time zero mean white Gaussian process, with a covariance matrix Q. The matrices A and Q are called the parameters of the model and are estimated based on a training data which allows the model to accurately characterize a wide class of mobility patterns. The parameters of the AR-1 model are updated periodically using the actual observed values. If the state information $\boldsymbol{s_n}$ at any time *n* is available, it is possible to predict the mobility state $\boldsymbol{s_{n+m}}$ at any time $n+m$ in the future using the following equation

$$s_{n+m} = A^m s_n \qquad \ldots (3)$$

In our experimental analysis, we use the AR-1 model to track the node movement and to predict the residence time of a node in a cluster. A node *i* is allowed to join a cluster C_j only if the estimated residence time is *at least* T_j (the admission criteria of the cluster).

4. Experimentation Results

In this section, we present the results from detailed simulation experiments carried out using the OPNET simulation software [16]. Before we discuss the results, we first describe the mobility models and the performance metrics used to evaluate the prediction schemes.

4.1 Mobility Models

We model the movement of nodes in the network using three mobility models: (1) Gauss-Markov, (2) Random Waypoint and (3) RPGM mobility models. Although random node mobility has been widely used, there are a number of applications of ad hoc networks in tactical communications such as emergency response teams, battlefields, etc., where nodes do not exhibit complete random motion. Therefore, in order to effectively study the performance of any clustering algorithm for an ad hoc network, we need to have mobility models that simulate realistic movement of mobile nodes. Hence, we selected the Gauss-Markov mobility model which allows us to control the randomness in the movement pattern.

We consider the random waypoint mobility model as a worst case scenario for any mobility prediction scheme. While good mobility prediction schemes should be successful in identifying explicit group mobility in the network, accurate mobility prediction in the presence of absolute random mobility is tough, if not impossible. The RPGM model introduces explicit group mobility in the network. An effective mobility prediction scheme should be able to identify the groups accurately. Therefore, in order to evaluate the strengths of the prediction schemes, we also conduct simulations consisting of groups of nodes, each moving independent of each other in an overlapping fashion. Reference [3] presents a comprehensive description of the above mentioned mobility models.

4.2 Performance Metrics

The primary goal of using a mobility prediction scheme is to enable the underlying clustering framework to provide temporal guarantees on the availability of routes to all the nodes within a cluster. In order to analyze the performance of the prediction schemes, we consider the following factors:

(1) The clusters identified through mobility prediction should exhibit temporal stability, i.e., there should be minimal changes in the membership of a cluster over a specified duration of time.

(2) The overhead associated with cluster maintenance should be minimized.

(3) The number of clusters in the network should be minimized to achieve scalability.

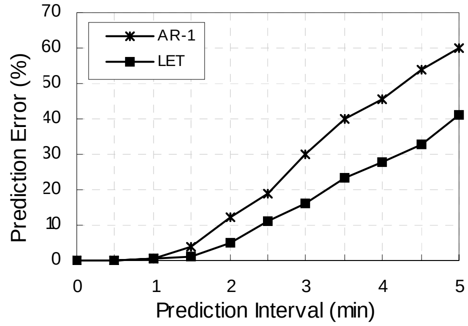
Metric for temporal stability: One way to evaluate cluster stability would be to observe the duration of time for which the membership a cluster remains unchanged. However, in the absence of explicit group mobility, it is very unlikely that nodes will remain with the same cluster for a long duration of time. Nevertheless, the stability of both inter- and intra-cluster routes critically depends on the frequency of the nodes leaving a cluster. Hence, we define the cluster survival time as the amount of time between two consecutive events of nodes leaving the cluster. We also record the cluster residence time which is the average amount of time spent by a node in a cluster. The cluster residence time is also a good measure of the stability of a cluster [13]. It is similar to cell residence time in cellular systems, which is a determinant of the distribution and rate of handoffs.

Metric for maintenance overhead: The maintenance overhead of the clustering algorithm can be evaluated using the reaffiliation count which represents the number of times mobile nodes change their cluster affiliations. A higher reaffiliation count means higher control traffic overhead since all active routes to the node need to be updated.

Metric for scalability: Finally, it is important to minimize the number of clusters in the network in order to improve scalability. Nevertheless, a clustering algorithm need not result in an minimal number of clusters, as long as the resulting clusters are relatively stable.

Measure for prediction accuracy: The performance of the clustering scheme is heavily dependent on the accuracy of the prediction algorithm. The decision to allow nodes to join a cluster is based on the future position of the nodes in the network, as estimated by the mobility prediction algorithm. We define the prediction error as the fraction of times a prediction turns out to be incorrect, i.e., the fraction of times a node leaves a cluster without satisfying its admission criteria.

It is important to note that the above mentioned metrics are not independent of each other. However, they all indicate the



(a) Percentage Error in Prediction

performance of the prediction scheme from different aspects. Clearly, there is a tradeoff between the size of the clusters and the stability of the clusters. A small cluster implies higher cluster survival times since membership changes will be less frequent. However, it is desirable to have clusters with multitude of nodes to localize the effects of topological changes. An optimal clustering scheme would be one which maximizes the stability of the clusters while still resulting in highly populated clusters.

4.3 Experimental Setup

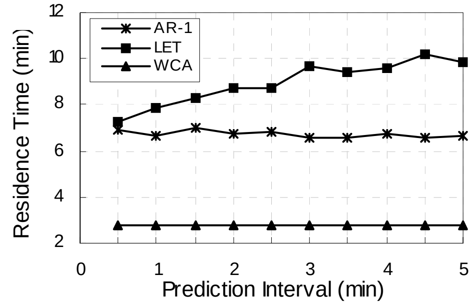
We simulate an ad hoc network consisting of 70 mobile nodes on a 1000m x 1000m grid. We used the OPNET simulator and each of the simulation runs were carried out for a 5 hour time period. We compared three clustering schemes, namely, LET-based predictive scheme, AR-1 model-based predictive scheme and WCA. For each of the simulation runs, the AR-1 model was initially trained on a data set consisting of 600 data points. During the course of the simulation period, the parameters of the model were updated with the observed values at intervals of 30 seconds.

4.4 Results with Gauss-Markov mobility model

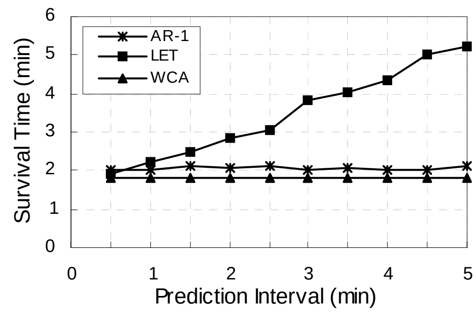
Gauss-Markov mobility model gives us the ability to control the randomness in the movement patterns of the mobile nodes through a tuning parameter α [3]. If α=1, the movement of the nodes is completely linear whereas a value of α = 0, results in random node movements. For our simulations, α was set at 0.8.

Sensitivity to prediction interval: In our clustering framework, a node is allowed to join an existing cluster in the network only if it satisfies the admission criteria associated with the cluster. A clusterhead uses the mobility prediction scheme to check if a node satisfies the admission criteria. A stricter admission criterion would require the mobility prediction scheme to predict the movement of the nodes over a larger interval of time. If the predictions were to be accurate, as the admission criteria is increased, the resulting cluster will exhibit greater temporal stability with high cluster survival and residence times.

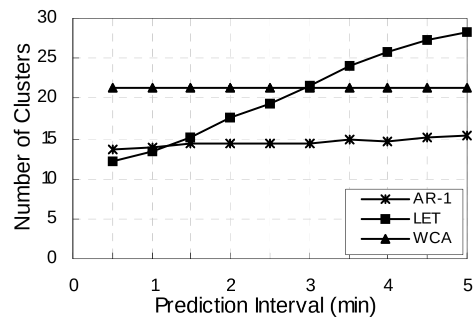
The graphs in Figure 1(a) indicate that both the schemes have a high prediction error over intervals greater than 2 minutes. We also observe that while the first order linear autoregressive model accurately tracks node mobility, it has significantly higher prediction error, making it unsuitable for *multi-step predictions* (predictions over large intervals). In our simulations, we used the



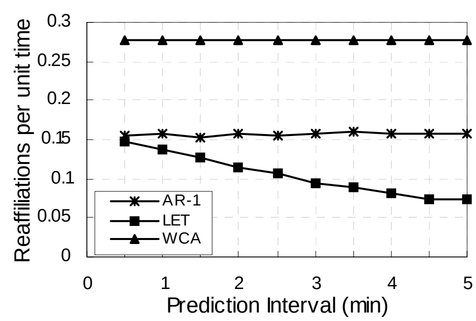
(b) Average cluster residence time



(c) Average cluster survival time

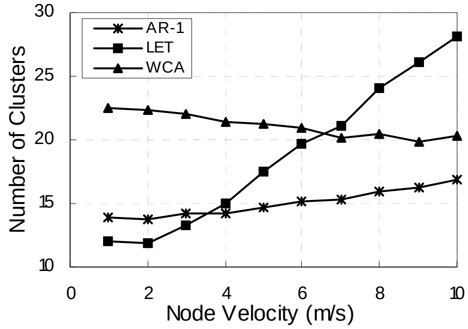


(d) Average number of clusters

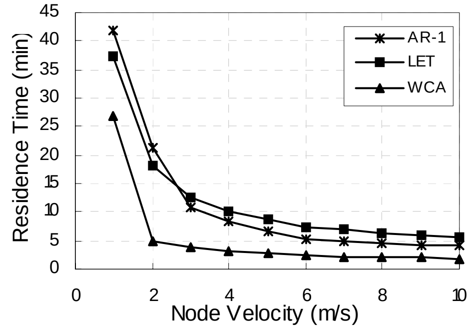


(e) Reaffiliations per unit time

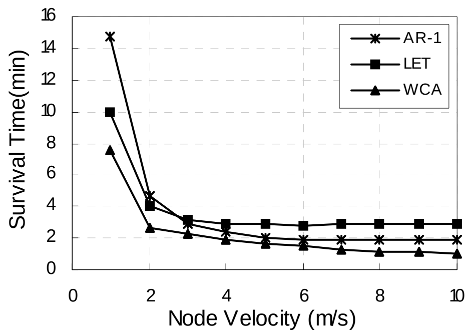
Figure 1. Effect of prediction interval under Gauss-Markov mobility model



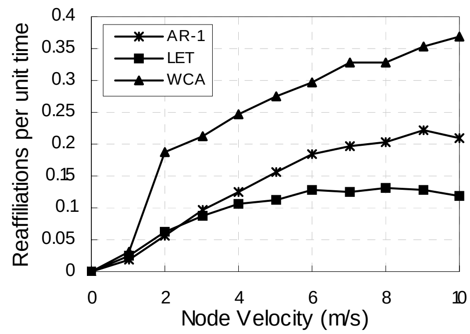
(a) Average number of clusters



(b) Average cluster residence time



(c) Average cluster survival time



(d) Reaffiliations per unit time

Figure 2. Effect of node speed under Gauss-Markov Mobility Model

prediction scheme referred to as the *plug-in* predictor which is obtained by repeatedly using the fitted model with unknown future values replaced by their own forecasts. This often results in high prediction errors over large intervals especially if the model order has not been fitted well [14]. Erroneous predictions often lead to the inclusion of some nodes in a cluster that in reality, do not satisfy the admission criteria. As a result, high prediction errors in the AR-1 based scheme severely degrade its performance as indicated by cluster survival and residence times in figure 1(b) and 1(c).

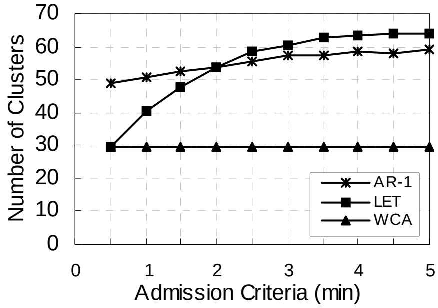
The LET-based scheme, on the other hand, does result in clusters with increasing survival times as the admission criteria of the clusters is increased. This is due to the fact that, in the absence of total random movement, a linear approximation of the movement of the nodes over a short interval of time holds good. *Nevertheless, both the mobility prediction-based schemes result in better temporal stability when compared to WCA which is insensitive to the admission criteria*. A prediction-based scheme also significantly increases the cluster residence time and hence the stability of routes in the network. Stable and long-lived clusters also result in significantly less maintenance overhead which can be verified in Figure 1(e).

Sensitivity to node speed: In the next set of experiments, we vary the average speed of the mobile nodes from 1 m/s to 10 m/s. The admission criteria of the clusters is fixed at 2 minutes with a uniform transmission range of 250 meters for all nodes. As the average speed of the mobile node increases, they tend to move in and out of the clusters more frequently resulting in a highly dynamic network topology. Consequently, maintaining the temporal stability of the clusters becomes increasingly difficult. Nevertheless, a good mobility prediction scheme should be able to accurately identify nodes that meet the admission criteria of the clusters even at higher speeds. As a result, a clustering scheme that uses mobility prediction should adapt the cluster size to node mobility while maintaining the temporal stability of the clusters. Specifically, at low speeds, it results in less number of clusters with larger cluster size, while the average number of clusters in the network gradually increase in response to higher speeds.

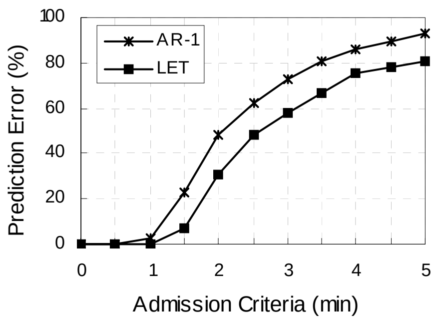
From our results, we observe that both the prediction schemes exhibit this trend as can be seen in figure 2. For typical walking speeds (less than 2 m/s), the AR-1 model-based scheme results in highly stable clusters in comparison to the LET-based scheme as seen in figures 2(b) and 2(c). However, the LET-based scheme adapts well to increasing node speeds making it more suitable at higher speeds. While the performance degradation of the AR-1 model-based scheme could be offset partially by updating the model parameters more frequently, it will significantly increase the computational overhead. Nevertheless, we observe that both the prediction schemes do result in clusters with better temporal stability when compared to WCA which also has significantly high number of reaffiliations as shown in figure 2(d).

4.5 Results with Random Waypoint mobility model

The accuracy of a mobility prediction algorithm is directly related to the movement patterns of the nodes in the network. In the presence of total random movement patterns, it is almost



(a) Average number of clusters



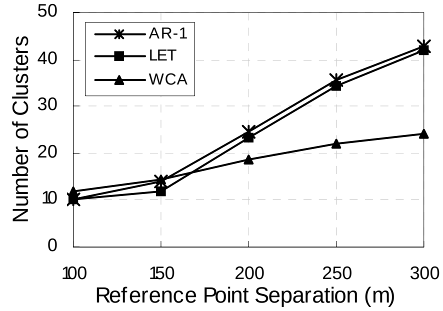
(b) Percentage error in prediction

Figure 3. Performance under random waypoint mobility model

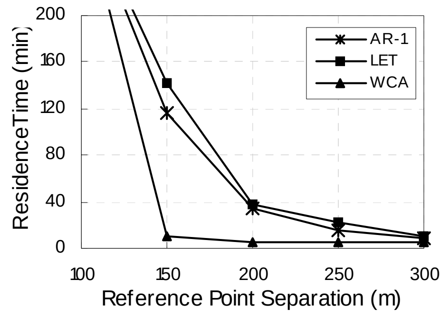
impossible for any prediction algorithm to perform well. We consider this case as a worst case scenario for clustering schemes that rely on mobility prediction. In figure 4(a), we observe that the both the prediction schemes result in an almost un-clustered network for high cluster admission criterion. This is so because, every node forms its own cluster since the prediction scheme is unable to identify any feasible cluster. WCA always yields a well-clustered architecture since it does not try to meet any admission constraint. Though the AR-1 model has been shown to be successful in accurately tracking random node movement[9], multi-step prediction is worse than LET since it tries to model the mobility of the nodes using a linear model. *Clearly, in the presence of random node mobility, it is advisable to use an algorithm that does not rely on mobility prediction***.**

4.6 Results with RPGM model

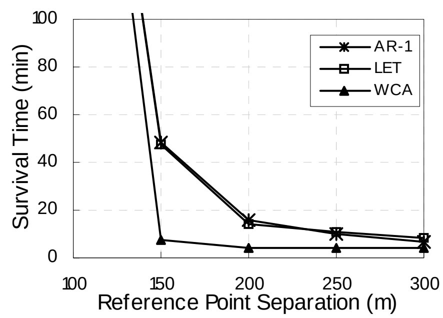
The Reference Point Group Mobility (RPGM) model represents the random motion of a group of nodes as well as the individual nodes with the group. Each group in the network is represented by its logical center. Individual mobile nodes randomly move about their own pre-defined reference points, whose movements depend on the group movement. Therefore, an accurate clustering scheme should be able to identify such explicit group mobility in the network. However, in order to evaluate the strengths of the prediction schemes, we conducted simulations consisting of 10 groups of 7 nodes each moving independent of each other in an overlapping fashion in 2000m x 2000m grid. The reference point



(a) Average number of clusters



(b) Average cluster residence time



(c) Average cluster survival time

Figure 4. Performance under RPGM model

separation is increased across subsequent runs of the simulation to simulate groups which are loosely coupled. The transmission range of the nodes is fixed at 250 meters and the admission criterion for the prediction schemes is set at 3 minutes. In figure 5(a), we plot the number of clusters identified by the clustering schemes with respect to increasing reference point separation. We observe that when the nodes within a cluster are tightly coupled together, all the three schemes are able to accurately identify the groups in the network. But as the separation between the reference points is increased, even the nodes within a group are far apart from each other making it impossible to identify the groups using a single hop clustering scheme. As a result, the number of clusters in the network increases steadily. However, both the prediction schemes have similar performance in terms of all the performance metrics. For node separations less than 150 meters, there is no change in the clustered topology once the actual groups are identified. Thus, the cluster residence times and the cluster survival times equal the duration of the simulation. The performance of WCA rapidly degrades in comparison to the predictive schemes as illustrated in figures 5(b) and 5(c). However, as the reference point separation is increased, individual node mobility (which is similar to random waypoint mobility model) starts to significantly influence the results.

5. CONCLUSIONS

In this paper, we studied the effect of mobility prediction on the temporal stability of clusters in MANETs. We used a simple mobility-aware clustering framework to compare the performance of two generic mobility prediction algorithms: (1) Mobility Prediction using the Link Expiration Time and (2) Mobility Prediction using Linear Autoregressive Models. Based on our simulation results, we make the following conclusions:

1. When the nodes do not exhibit total random motion, a predictive clustering scheme significantly improves the temporal stability of the clusters when compared to a mobility-aware non-predictive scheme. However, there is a tradeoff between the stability and the size of the clusters.
2. In the presence of total random node mobility, it is advisable to use an algorithm that does not rely on mobility prediction.
3. In the presence of explicit group mobility, both predictive and non-predictive clustering schemes are successful in accurately identifying the groups. However, when the separation with the nodes in a group increases, mobility prediction helps in improving the temporal stability of the identified clusters. However, the performance gains are restricted due the one-hop clustering scheme used.
4. A predictive clustering scheme is able to adapt to varying network conditions by dynamically adjusting the cluster size in order to guarantee temporal stability.
5. While the AR-1 model has been shown to accurately track node mobility [8], it does not always result in accurate mobility predictions.

We make the following conclusions about the comparative study of the two prediction schemes.

1. The AR-1 model with the recursive *plug-in* predictor results in higher multi-step prediction errors in comparison to the LET-based scheme. However, for predictions over a small interval of time, the AR-1 based scheme results in an optimal number of clusters with comparable survival times.
2. The AR-1 model-based scheme performs well under low mobility whereas the LET-based scheme is a more suitable choice at higher speeds since the performance of the AR-1 model degrades faster than the LET-based scheme with increasing node speeds.
3. In the presence of explicit group mobility, both the prediction schemes perform equally well.

In our simulations, all the clusters in the network have a fixed admission criterion. We are currently investigating various approaches to arrive at better values for the admission criteria of individual clusters based on parameters such as node velocity and transmission range.

6. REFERENCES

[1] Y.-C. Tseng, S.-L. Wu, W.-H. Liao, and C.-M. Chao, “Location awareness in ad hoc wireless mobile networks,” IEEE Computer, 34(6), pp. 46-52, June 2001.

[2] P. Guptar and P. R. Kumar, “The capacity of wireless networks,” IEEE Transactions on Info. Theory, vol. IT-46.2, March 2000, pp. 384-404

[3] T. Camp, J. Boleng, and V. Davies, “A survey of mobility models for adhoc network research,” Wireless Comm. & Mobile Comp. (WCMC), vol. 2, no. 5, pp. 483-502, 2002.

[4] A.B. McDonald and T.F. Znati, “A mobility-based framework for adaptive clustering in wireless ad hoc networks,” IEEE Journal on Selected Areas in Communications 17(8), pp.1466–1487, 1999.

[5] M. Chatterjee, S. K. Das, and D. Turgut, “WCA: A weighted clustering algorithm for mobile ad hoc networks”, Journal of Cluster Computing, Special issue on Mobile Ad hoc Networking, No.5, pp. 193-204, 2002.

[6] J. Y. Yu and P. H. J. Chong, “A survey of clustering schemes for mobile ad hoc networks,” IEEE Communications Surveys, Vol 7, No. 1, First Quarter 2005.

[7] W Su, S-J Lee, and M. Gerla, “Mobility prediction and routing in ad hoc wireless networks,” International Journal of Network Management, Vol. 11, Jan-Feb 2001, pp. 3-30

[8] Z. R. Zaidi and B.L. Mark, “Mobility estimation for wireless networks based on an autoregressive model,” in Proceedings of IEEE Globecom, Dec. 2004.

[9] Z. R. Zaidi, B.L. Mark, and R. K. Thomas, “A two-tier representation of node mobility in ad hoc networks,” in Proceedings of IEEE SECON, October 2004.

[10] S. Sivavakeesar, G. Pavlou, and A. Liotta, “Stable clustering through mobility prediction for large-scale multihop intelligent ad hoc networks,” IEEE Wireless Comm. and Networking Conference, March 2004

[11] S. Jian, D. He, and J. Rao, “A prediction-based link availability estimation for mobile ad hoc networks,” in Proceedings of IEEE INFOCOM, 2001.

[12] J. Tang, G. Xue, and W. Zhang, “Reliable routing in mobile ad hoc networks based on mobility prediction”, in IEEE International Conference on Mobile Ad Hoc and Sensor Systems, October 2004.

[13] A.B. McDonald and T.F. Znati, “Design and simulation of a distributed dynamic clustering algorithm for multimode routing in wireless ad hoc networks,” Simulation, Vol. 78, No. 7, pp. 408-422, July 2002.

[14] C-K Ing, “Selecting optimal multistep predictors for autoregressive processes of unknown order,” The Annals of Statistics, Vol. 32, No. 2, pp. 693-722, 2004

[15] S. Capkun, M. Hamdi, J. P. Hubaux, "GPS-free positioning in mobile da-hoc networks," in Proceedings of HICSS, Hawaii, January 2001, pp. 10

[16] OPNET simulator, http://www.opnet.com