A mobility and traffic generation framework for modeling and simulating ad hoc communication networks[1]

Chris Barrett[a], Martin Drozda[b,c], Madhav V. Marathe[a], S.S. Ravi[c,*] and James P. Smith[a]
[a]*Basic and Applied Simulation Science* (*CCS-5*) *Los Alamos National Laboratory, MS M997, P.O. Box 1663, Los Alamos, NM 87545, USA*
E-mail: {*barrett,marathe,jpsmith*}*@lanl.gov*
[b]*University of Hamover, FG Simulation and Modellierung, Welfengarton 1, 30167 Hannover, Germang*
E-mail: drozda@sim.uni-hannover.de
[c]*Part of the work was done while the authors were visiting the Basic and Applied Simulation Sciences Group at the Los Alamos National Laboratory*

Abstract. We present a generic mobility and traffic generation framework that can be incorporated into a tool for modeling and simulating large scale ad hoc networks. Three components of this framework, namely a mobility data generator (MDG), a graph structure generator (GSG) and an occlusion modification tool (OMT) allow a variety of mobility models to be incorporated into the tool. The MDG module generates positions of transceivers at specified time instants. The GSG module constructs the graph corresponding to the ad hoc network from the mobility data provided by MDG. The OMT module modifies the connectivity of the graph produced by GSG to allow for occlusion effects. With two other modules, namely an activity data generator (ADG) which generates packet transmission activities for transceivers and a packet activity simulator (PAS) which simulates the movement and interaction of packets among the transceivers, the framework allows the modeling and simulation of ad hoc communication networks. The design of the framework allows a user to incorporate various realistic parameters crucial in the simulation.

We illustrate the utility of our framework through a comparative study of three mobility models. Two of these are synthetic models (random waypoint and exponentially correlated mobility) proposed in the literature. The third model is based on an urban population mobility modeling tool (TRANSIMS) developed at the Los Alamos National Laboratory. This tool is capable of providing comprehensive information about the demographics, mobility and interactions of members of a large urban population. A comparison of these models is carried out by computing a variety of parameters associated with the graph structures generated by the models. There has recently been interest in the structural properties of graphs that arise in real world systems. We examine two aspects of this for the graphs created by the mobility models: change associated with power control (range of transceivers) and variation in time as transceivers move in space.

Keywords: Mobile and ad hoc networks, mobility models, simulation and modeling, graph theory

1. Introduction

1.1. Simulation of ad hoc networks

An *ad hoc wireless mobile network* is a collection of mobile transceivers that communicate via radio transmission. There is no wireline network to support the

[1] The work is supported by the Department of Energy under Contract W-7405-ENG-36.

[*] Corresponding author. Department of Computer Science, University at Albany - SUNY, Albany, NY 12222, USA. Supported by a grant from Los Alamos National Laboratory and by NSF Grant CCR-97-34936. E-mail: ravi@cs.albany.edu.

ISSN 1058-9244/04/$17.00 © 2004 – IOS Press and the authors. All rights reserved

movement of packets. The communication network is formed spontaneously when transceivers activate their radios. Moreover, movement of the transceivers constantly changes the connectivity of the network. These networks are gaining popularity due to their applicability in a number of important situations such as battlefield communications, emergency management and response, etc. The topic of designing and building large scale ad hoc networks is an area of active research.

The dynamic connectivity of the network that is induced by the constant movement of transceivers implies that the performance of the communication network depends crucially on how the underlying transceivers move. Indeed, as has been observed in the literature (see for example [9,10,21,26,27,38]), parameters such as the velocity of transceivers, their mobility patterns and their spatial density significantly affect the network performance.

The goal of the *AdHopNet* project at the Los Alamos National Laboratory (LANL) is to develop a scalable simulation-based tool that can be used for the design and analysis of ad hoc communication networks being built as a part of SUO-SAS project at DARPA [8] and the Urban Infrastructure Suite of analytical tools being developed as a part of the National Infrastructure Simulation and Analysis Capability (NISAC). The focus of this paper is on some modules of this simulation-based analysis tool. These modules produce mobility patterns, and the data generated by these modules in conjunction with a module that generates communication activities is used to drive the packet movement simulator of AdHopNet. The modules for generating mobility data were developed originally for AdHopNet. However, they can be used in conjunction with other simulation tools such as *GloMoSim* and *ns-2* [3,4]. The flexible framework within which the modules operate allows us to create both synthetic and realistic mobility patterns along with appropriate communication activities (or calling patterns). Although our framework is designed for ad hoc networks, it can be used to obtain an understanding of the effects of mobility on a wired backbone network that contains mobile units and *base stations*. In such networks, each base station is a stationary unit that communicates with one or more mobile units. At any time, each mobile unit is assigned to a base station. A mobile unit can communicate directly with only its assigned base station. The base stations communicate among themselves through a wired backbone network. (The backbone may also involve satellite links.)

1.2. Basic terminology

We now present some terminology that is used throughout this paper. The *range* of a transceiver is the geographic distance over which the packets sent by the transceiver can be received. Different transceivers may have different ranges. The distance metric used is the Euclidean (L_2) metric. Thus, if the range of a transceiver T is r, then a packet sent by T can be received only by the transceivers (or base stations) that are within or on the circle of radius r centered at the point occupied by T.

In an ad hoc network, a *hop* refers to the movement of a packet directly from one transceiver T to another transceiver T' which is within the range of T. In a wired backbone mobile network, a hop may refer to either the movement of a packet between a mobile unit and its base station or between a pair of base stations.

The *out neighbors* of a transceiver T are those transceivers that can receive packets from T in one hop. Similarly, the *in neighbors* of a transceiver T are the transceivers from which T can receive packets in one hop. The terms “in neighbor” and “out neighbor” arise from a natural directed graph representation of a network: the nodes of the graph correspond to transceivers (and base stations) and a directed edge (u, v) implies that the transceiver (or base station) represented by v is in the range of the transceiver (or base station) represented by u.

1.3. Summary of contributions

This paper presents three novel contributions: a generic framework for simulation, realistic mobility modeling and a graph analysis of ad hoc networks. The first part is a generic framework for mobility and traffic generation that can be incorporated into tools for modeling and simulating ad hoc communication networks. This framework, discussed in Section 3, is flexible enough to support a variety of modules for each component. We illustrate the flexibility of the framework by showing how it can be used to compare several mobility models. Further, it is possible to compose the components of this framework in different ways to obtain different modeling scenarios (see Section 3.2). A similar framework used in the TRANSIMS system [6] has allowed studies ranging from equity analyses of transportation system improvements through detailed studies of response strategies for chemical and biological attack. Despite this paper demonstrating only the applicability of the framework in the analysis of mobility models, such a wide range of potential applications is also likely for this system.

Using the framework with all other modules fixed, we carry out an experimental study of three mobility models. Two of these are synthetic models (random waypoint and exponentially correlated mobility) proposed in the literature [12,22,32]. The third model is based on a detailed urban population mobility modeling tool (TRANSIMS) developed at the Los Alamos National Laboratory [6]. This is the first time a realistic urban population mobility model has been used in modeling ad hoc networks. In fact, it is the only existing urban population mobility model with its level of detail. Its comparison with the simpler popular mobility models is interesting–there are significant differences.

We note that the approach used in this paper to compare mobility models is different from those used in other papers in the literature. For example, reference [13] compares spatial distributions of nodes under several mobility models. In reference [21], the comparison is based on parameters such as average displacement from the starting point, rates of links coming up or going down and the rate of packet delivery. Our work compares mobility models by computing a variety of measures for the graphs generated by each of the models (Section 5). The measures computed include degree distribution, clustering coefficient and diameter. Computing such measures enables us to investigate a fundamental question: Are mobile networks scale-free? This property has proven to be pervasive in real-world networks [2] such as scientific collaboration, movie actors, protein folding, linguistics, etc. For instance, Govindan and Tangmunarunkit [19] and Faloutsos et al. [18] studied this phenomenon and showed that internet networks have the scale-free property. To our knowledge this has not been investigated for ad hoc networks. Our preliminary results indicate that the structural properties of ad hoc networks are different from internet router networks, in particular that mobile networks are not scale-free. Also, unlike previous analyses of systems that can be modeled as graphs, there is an intrinsic time dependence in mobile networks. We find that measured quantities (such as the degree distribution) are invariant on short time scales (seconds), but can change significantly on longer time scales. This is important in light of recent work by the research community [2,34,36]. All of the measurements also show variation with the range (power) of transceivers. The experimental analysis reported in this paper is a first step in exploring structural properties of graphs arising from mobility models.

The remainder of this paper is organized as follows. Section 2 presents a brief review of mobility models proposed in the literature. Section 3 provides a discussion of the components of our framework. Section 4 presents a more detailed discussion of the modules that were used in generating the experimental results reported in this paper. Section 5 provides an analysis of the experimental results. Finally, a discussion of our ongoing and future research is presented in Section 6.

2. A brief review of known mobility models

A number of mobility models that can be used in the simulation of ad hoc networks have been proposed in the literature. We limit the presentation to models that are closely related to the ones used in our experimental study. For a discussion of other models and a classification of models, we refer the reader to [13,14], and the references therein. All of these models are very simple compared to TRANSIMS, discussed briefly in Section 4.1.3.

In the *random mobility model* discussed in [38], the position of each transceiver at time $(t + 1)$ is a random displacement from its position at time t. This implies in essence that the speed and direction are both random variables that have no correlation with their current values. As discussed in the literature, this model tends to produce unrealistic choppy motion with sharp turns, sudden stops, etc.

Basagni et al. [11] study an extension of this model, where the speed is held constant but direction is a uniform random variable over a specified range. In Ko and Vaidya's model [25], the transceivers can move along a prespecified set of paths made up of segments. The segment lengths are exponentially distributed and the direction of each segment is chosen uniformly at random. Speed is also assumed to be uniformly distributed within a window of size s around the current speed v. The model of Das et al. [16] selects a sequence of sub-destinations in an on-line fashion. When at a given sub-destination, the transceiver selects its next sub-destination and speed. It then travels to this sub-destination along a straight line connecting the current and the next sub-destination. This model can be seen to be an extension of the model of Basagni et al. [11].

The *random waypoint mobility model* discussed in [22] specifies a sequence of pause and motion periods for each transceiver. During the motion period, a randomly chosen sub-destination is reached using a constant speed chosen uniformly at random between zero and a maximum allowable limit. A recent paper [13] incorporates additional features into the random waypoint model with a view to making the model more realistic. In that work, speeds are limited to a few that are characteristic of automobiles. Further, the speed and direction are coupled with a minimum turn radius assuming a particular coefficient of static friction and a flat road. For other variants of the random waypoint model, see [20,27].

All the above models have one common theme: they specify mobility rules for individual transceivers. Researchers have also identified another type of mobility model, called the *group mobility model*. In such a model, the set of transceivers is partitioned into groups. The movements of individual transceivers within a group are strongly correlated. Such models aim to provide realistic mobility data when ad hoc networks are used in an emergency response or a military setting. Two basic models of this type have been proposed in the literature.

The first is the *Exponentially Correlated Random Mobility* (ECRM) model studied in [12,21,32]. Using this model, one can control the movement of a group independently of the movement of the other groups and the nodes within the group. At each step, a group undergoes a randomly chosen displacement along a randomly chosen direction.

A more sophisticated model referred to as the *Reference Point Group Mobility* (RPGM) model is proposed in [21]. By an appropriate setting of the parameters, RPGM can mimic many other known mobility models. An informal description of this model is as follows. The transceivers are partitioned into a specified number of groups. Each group has a logical center which can be assigned a specific movement model. Transceivers within a group move together as a group within an annulus around the group center, with a small amount of random movement. Such movements can be readily implemented (for instance) by maintaining prespecified lower and upper bounds on the distance of each point from its group center.

3. Conceptual design of the framework

We have developed a prototype software tool that allows a user to substitute arbitrary models and programs into the various components of the simulation, called modules. The modules are run independently, and exchange of information between the modules is accomplished using files, which can also be considered as state-saves for readers more accustomed to single, large, complex, amalgamated programs (that are rarely easy to debug). This approach also allows the user to combine the modules in interesting ways to create very complex simulations without having to recode anything (see Section 3.2). Commercial packages can also be used as drop-in replacements with a bit of data filtering.

This paper demonstrates the versatility of the framework by considering three different mobility modules. As in [21], our tool is flexible enough to support group and individual mobility models. In addition, it supports a more realistic mobility model, obtained from a large scale microscopic mobility simulator called TRANSIMS, also developed at Los Alamos [6]. Using TRANSIMS, one can produce extremely high fidelity mobility models that have been validated against realistic road traffic and many other population mobility characteristics in present-day urban areas.

Our framework combines the mobility models with a communication activity generator that produces synthetic calling patterns in which parameters such as the size and number of packets, level of burstiness, etc. can be controlled. The communication activity patterns can be generated so that the synthetic patterns match realistic patterns in a statistical sense. The mobility data and communication activity generators have been designed with the goal of providing input data to ad hoc network simulators.

3.1. A functional view of the prototype simulator

A schematic diagram showing the various modules of the prototype simulator is presented in Fig. 1. This schematic diagram represents a functional view of the system, connected in its most simple configuration. Care has been taken to ensure that the design is responsive to future enhancements. The simulator that is being developed as part of the AdHopNet project at the Los Alamos National Laboratory uses these modules along with other more sophisticated modules to permit simulations of large-scale ad hoc networks (consisting of millions of transceivers). A brief discussion of the functions carried out by the various modules is presented below.

Mobility Data Generator (*MDG*)*:* The function of this component is to generate the positions of transceivers with respect to time. This module allows transceivers to become idle for some period of time and rejoin the network at a later time. Further, the module permits new transceivers to join the network and existing transceivers to leave the network permanently. The

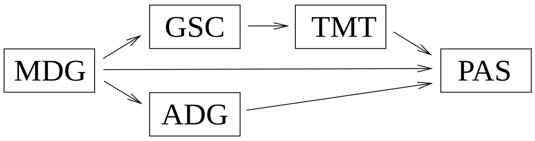


Fig. 1. A schematic diagram of the simulator, showing only "feed-forward" paths and a simple composition of the modules.

design is flexible enough to incorporate various mobility models discussed earlier and also to incorporate mobility models derived from realistic mobility data.

Graph Structure Generator (*GSG*)*:* The function of this module is to construct the directed (or undirected) graph corresponding to the ad hoc network from the mobility data provided by MDG. The type of output that is produced by this module is as follows. At each time instant when mobility data for transceivers is available, the GSG constructs the out neighbors of each active transceiver T knowing the range of T and the positions of all other active transceivers. The module has been decoupled from MDG so that it can be made compatible with many different mobility simulation tools. Moreover, the module is flexible enough so that the existence of an edge between a pair of transceivers A and B can be a function of the given radio propagation model. For example, we can use the line of sight model or a two way path loss model [12] to decide whether or not two transceivers can talk to each other.

Occlusion Modification Tool (*OMT*)*:* The inputs to this module are a graph (as produced by GSG) and information about occlusions in the terrain. At an abstract level, the effect of occlusions can be seen as deleting a subset of the edges in the ad hoc network formed by the transceivers. This could be used to account for many forms of obstruction, including buildings or other physical barriers, stationary electronic interference, etc. The initial use of the module was to handle constraints due to different types of terrain. For the purposes of this paper, this module was implemented to support edge-cutting barriers in two dimensions, where information regarding barriers is given as a list of point pairs. Again, the reason for making this a separate module is that the effect of occlusions on the existence of an edge can now be decided post hoc, independent of the mobility and graph generation models.

Activity Data Generator (*ADG*)*:* The purpose of ADG is to generate *sessions*, where a session is a sequence of packets to be sent from a chosen source to a chosen destination. The ADG module generates sessions for a specified fraction of the transceivers that are active at a given time. For each session, the data generated by the ADG specifies the source, the destination, the number of packets in the session, the size of each packet and the rate at which the source generates the packets. Simulating a session with varying packet sizes, rates, etc., can be done using multiple "logical" sessions to emulate a single "actual" session.

Packet Activity Simulator (*PAS*)*:* This module simulates the packet flow in the system using the activity data and the graph structure generated by ADG and GSG respectively. Thus, this module can begin to execute only after all the other modules have finished their execution. It is this module that simulates the interaction of packets with the network and each other, and resolves the contention for resources.

The Activity Data Generator and Packet Activity Simulator that we have implemented will be discussed more fully in subsequent publications, in part because each is interesting enough to stand on its own. For this work we have not used our own million-node PAS, but the freely available GloMoSim application [3]. Although not discussed, we have linked the *ns-2* simulator into our framework as the PAS, and also the commercial QualNet simulator. The modular nature of the framework makes such substitutions possible, and relatively simple.

3.2. Composition of modules

The modular nature of the components and our modeling methodology provides the ability to simulate a variety of scenarios by appropriate compositions of modules. Figure 1 illustrates only the most simple feed-forward-once configuration of the simulator. The real power of the methodology is realized only through the use of feedback and sophisticated combinations of the modules. We present two simple examples below to illustrate the idea.

A troop movement exercise could be simulated by running a clustered random waypoint model once for each unit of soldiers, then running each group independently through the graph connection module, and finally merging all the data. So long as there are no duplicate transceiver IDs, this emulates troop deployment where each unit has its own non-interfering channel of communication and pattern of movement. For a simple modification, one member of each unit (say, the commander) could be selected to have all other selected transceivers (one from each unit) added to its connected node list. This allows commanders of various units to have a second channel amongst themselves.

A second example is in the modeling of self-adapting networks, where each transceiver adjusts its transmission range so that its number of neighbors is within specified bounds. In this case, GSG would need to be iterated with each node searching for a power value that yields a number of neighbors satisfying the specified bounds. Such a simple feedback loop could be drawn into Fig. 1 as a self loop on the GSG. Degree optimization and topology control is considered in part due to recent work by Royer et al. [33], Ramanathan and Rosales-Hain [31], Liu et al. [28] and by Kleinrock and Silvester's classic work on the topic [24]. Self-adapting networks are important in the context of power-aware protocols (see for example [23]). The mobility modeling tool presented in this paper can be used in simulating such protocols.

The above examples serve to illustrate the flexibility of the framework with respect to mobility models. The framework allows a similar flexibility with respect to the other components as well. For example, if one wants to study the effects of different models for activity generation in ad hoc networks, it is a simple matter to substitute different versions of the ADG module without changing the other modules. Likewise, one can also study the effects of different types of occlusions by substituting different versions of the OMT module.

4. Implementation details

In this section, we provide some implementation details regarding three modules (namely, the mobility data generator, the graph structure generator and the occlusion modification tool), which are used in the comparison of the mobility models. Implementation details regarding the activity data generator and the packet activity simulator have not been included as they are not germane to the experimental study reported in this paper.

4.1. Mobility data generator

For this study, we implemented three mobility models in the MDG. The first two models (namely, random waypoint and exponentially correlated random mobility) were discussed in Section 2, and we introduce no new features to these models from previously published work. The third model (the TRANSIMS mobility model) is from a continuing research effort in urban population mobility at the Los Alamos National Laboratory. Each of the mobility models was implemented to have the same average node density and average node speed, where those numbers were determined from the TRANSIMS tool to be 160 nodes per square kilometer and 16.4 meters per second, respectively. The simulation region was set to be a square of area 6.25 square kilometers.

4.1.1. Random waypoint

The random waypoint (RW) mobility model [22] is the simplest of the three, but still has more structure than a simple random walk model. It is an event driven model, where each transceiver is given a location and a speed at which to move to that location from the starting position. When a transceiver reaches its new position, it randomly chooses a new position and speed and starts moving after an optional delay. As implemented here, the optional delay was always set to zero seconds, and the new positions (as well as the initial positions) are each drawn from a uniform random distribution over the entire simulation area. A snapshot of the node positions at time $= 100$ seconds after the start of the simulation is shown in Fig. 2. Because the new positions are distributed uniformly randomly over the entire simulation area, any snapshot will yield a similar distribution of node positions. Snapshots at other timesteps are not shown because they are qualitatively (and approximately quantitatively) indistinguishable.

It is important to note that the distribution of new speed choices is not the same as the distribution of instantaneous speeds. Nodes do not choose a time at which to travel with a given velocity (which would make the distributions the same); rather, they choose a new position. It takes longer for a slow moving node to reach its new destination than it would for a fast node. Thus, the slow moving nodes stay in the slow-moving state longer. This gives a higher weight to the slow speeds in a temporal (or spatial) average of speed. In order to achieve an average node speed of 16.4 meters per second, matching the TRANSIMS data, it was necessary to have the nodes randomly choose a new speed from the uniform distribution from zero to 67 meters per second. The resulting speed distributions discussed in Section 5 show this effect for the RW model.

Under the RW model, at each discrete timestep, each node updates its position according to its velocity. If it reaches its next velocity-change position, it chooses a new velocity and moves in that direction. Thus, the position and velocity of a node at any instant can be computed in $O(1)$ time. Therefore, if there are N nodes and the number of time steps simulated is τ, the total time needed to compute the data set for all the time steps is $O(N\tau)$.

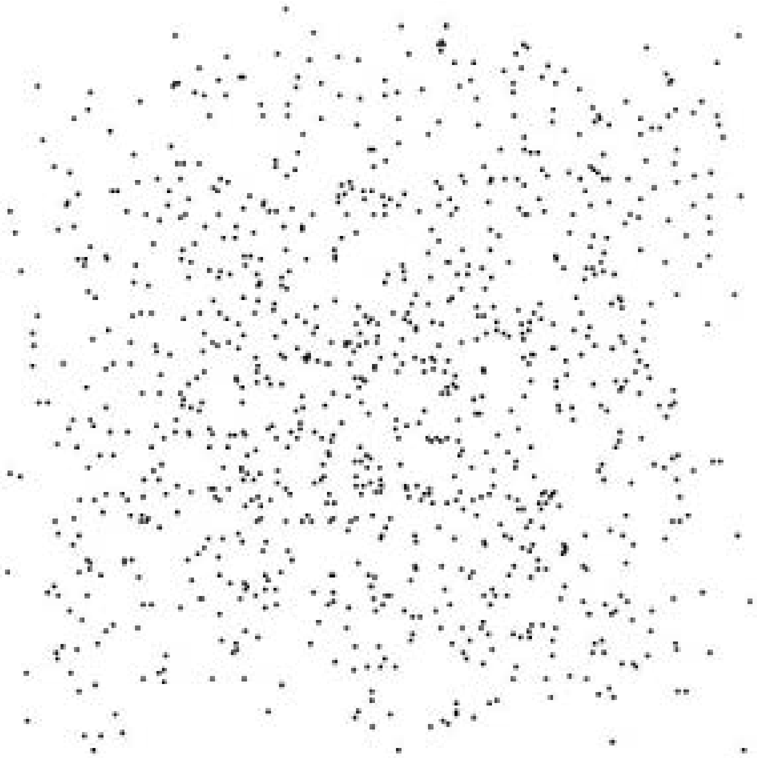


Fig. 2. A snapshot of an instance of transceiver positions in the Random Waypoint model.

4.1.2. *Exponentially correlated random mobility (ECRM)*

Our interpretation of the ECRM model is as follows. Each node is a member of a group. At each timestep, a node chooses a new position, $\overline{x}_n$, relative to its group center, $\overline{x}_g$, and each group center chooses a new position relative to the center of the simulation coordinate system. Thus at any timestep t, the position of a node in the simulation coordinate system is given by $\overline{x}_n + \overline{x}_g$. Both node and group motions follow the same equation (relative to their respective coordinate systems) given below.

$$\overline{x}_i(t+1) = \overline{x}_i(t) \cdot e^{-1/\tau} + s \cdot \sigma \cdot \overline{r} \cdot \sqrt{1 - e^{-2/\tau}} \quad (1)$$

Here, i is either n or g representing the node position or the group position. The decay of positions back to the center of the group is controlled, in part, by the τ parameter. The extent of motion about the group center is characterized in part by the quantity $s \cdot \sigma$. The random component of the motion is controlled by the vector random variable $\overline{r}$. Any distribution can be used for this random variable, however, researchers [12,21, 32] seem to prefer one with radial dependence. In order to best compare the results from ECRM to the other two simulations that where characterized by a square simulation area, we use a random vector variable that has both x and y components distributed uniformly randomly in a range of ± 1.

The motion of each group and each node is characterized by its τ and $s \cdot \sigma$ parameters, which could be chosen according to any scheme the user desires. We have chosen to use a single pair of values for τ and $s \cdot \sigma$ for all of the groups and another for all of the nodes. For the group centers, we used $\tau = 3000$ and $s \cdot \sigma = 740$ meters. For nodes, we used $\tau = 1500$ and $s \cdot \sigma = 300$ meters. These were chosen to yield the expected average speed of 16.4 meters per second with an extent of motion covering approximately the same simulation area as the RW model in a reasonable amount of time. There was no boundary to the ECRM simulation area; the model itself constrains the extent of nodes. Two snapshots of node positions generated using the ECRM model are shown in Fig. 3.

Like the RW model, the algorithm used for ECRM is also very simple. For each time step, each group moves to its new coordinates, then every member of the group moves relative to the group center. The time used to generate the position of any node per time instant is

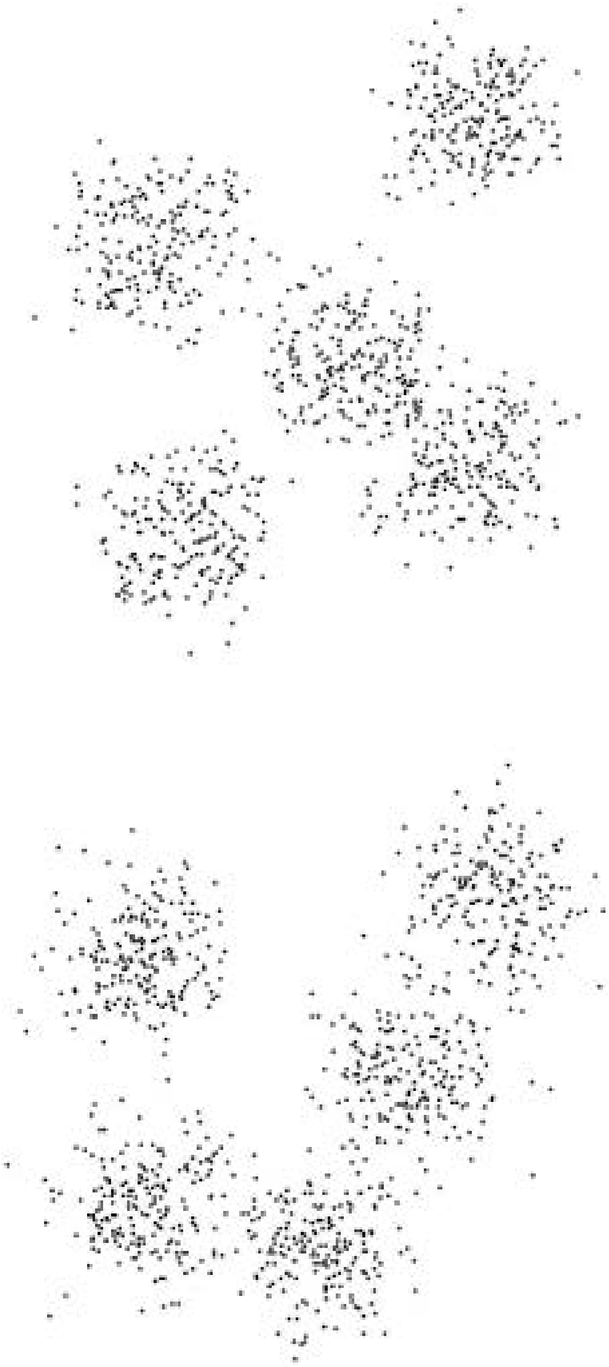


Fig. 3. Two snapshots of transceiver positions in the ECRM mobility model. The upper panel is at time zero (after a 200 second run to remove the effects of initial positions), and the lower panel is 900 seconds later. Because of the grouping nature of the model, there is a noticeable difference in the distribution of positions with respect to time.

$O(1)$. Since each node is in only one group, the total time needed to generate ECRM mobility data for N nodes and τ time instants is $O(N\tau)$.

4.1.3. TRANSIMS – realistic traffic

The third model used in this study incorporates realistic mobility patterns. By "realistic", we mean move-

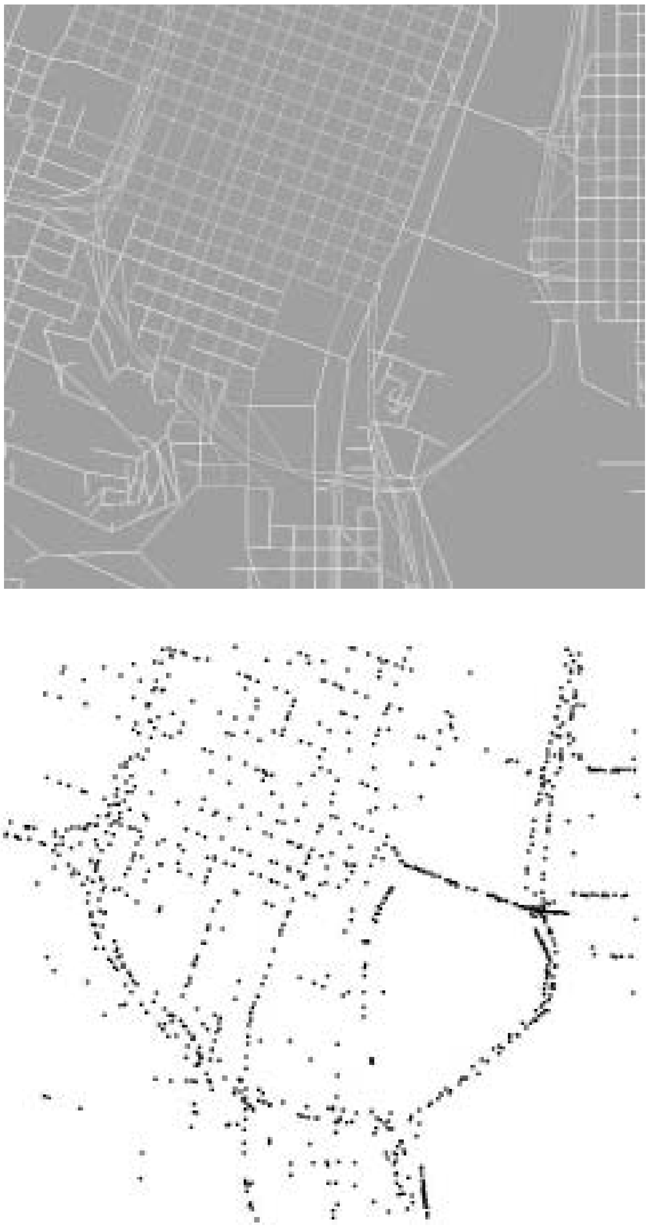


Fig. 4. The upper panel shows the road network on which vehicles are moving, and the lower panel is a snapshot of those vehicles at 7 am in a TRANSIMS simulation of Portland, Oregon. The gap on the right of each figure is the Willamette river.

ments that are likely to arise in operational ad hoc networks in urban areas. In this context, the discussion is pertinent to commercial uses of ad hoc networks, although the ideas are generic enough to be applicable in military and emergency response settings. The tool we used to generate these mobility patterns is TRANSIMS, a large-scale microscopic transportation simulation environment [6]. TRANSIMS produces traffic patterns (positions of all travelers on a second by second basis) for a large metropolis. It is beyond the scope of this paper to completely describe the TRANSIMS urban population mobility models and simulation. The interested reader should consult the many technical publications and the several thousands of pages of documentation for the system, both available on-line (www-transims.tsasa.lanl.gov).

TRANSIMS has the ability to carry out highly detailed simulations of very large transportation net-

Input:

(a) The range for each transceiver.

(b) Time instants τ_j, $1 \leq j \leq m$; for each time instant τ_j, the coordinates of each transceiver that is active at τ_j. (Each X-coordinate is an integer in the range [0 .. XMAX] and each Y-coordinate is an integer in the range [0 .. YMAX].)

Output: For each time instant τ_j and each transceiver T that is active at time τ_j, the number and list of transceivers to which T has an outgoing edge.

```
1. Obtain the range for each transceiver from the input.
2. for $j = 1$ to $m$ do
    (i) Construct a list $L$ of all transceivers that are active at time $\tau_j$. (The information for each
        transceiver $T$ in $L$ includes the coordinates of $T$.)
    (ii) Initialize the Location Table LT such that for each $i$, $0 \leq i \leq$ XMAX, LT($i$) points to a list
        of transceivers from $L$ whose X-coordinate $= i$; the Y-coordinates of the transceivers in
        LT($i$) are in increasing order.
    (iii) for each transceiver $T$ in $L$ do
        (a) Use the Location Table to construct a list $L'$ of the transceivers in $L$ to which $T$
            has an outgoing edge. (The details of this step are given in Section 4.2.)
        (b) Write the number of transceivers in $L'$ and the IDs of the transceivers in $L'$ to the
            output.
```

Fig. 5. Outline of the Graph Generator.

works. Thus, it can be used to produce mobility patterns on an extremely large scale (1.5 million people with 1 million links). Both for clarity of demonstration (the figures of nodes and networks included here are already somewhat crowded) and in order to be compatible with ad hoc network simulators (PAS) such as *ns-2* and GloMoSim, which can handle no more than about 1500 transceivers, we selected a small part of downtown Portland, OR, for generating experimental results. The top panel of Fig. 4 shows the chosen area. The area is a square 2.5 km on a side, with approximately 1000 vehicles present at any time. This represents a little less than 0.0009% of the Portland simulation area used by TRANSIMS. (The comparative analyses shown below were done only on timesteps having nearly 1000 vehicles, for fair comparison with the other two models.)

It is important to note that we used only vehicle positions in the simulation area, which introduces a feature not in the other two models; that is, the TRANSIMS model allows transceivers to enter or leave the simulation area (and therefore the network), or to appear intermittently. This is also a common scenario in ad hoc networks where the structure of the network changes spontaneously. Since the simulation was designed to handle such changes, our approach doesn't cause difficulties for the other models. This illustrates how realistic mobility data differs significantly from that produced by simple synthetic models.

The road network underlying the area of simulation from which data was extracted is shown in the first panel of Fig. 4. In the second panel is a snapshot of vehicle positions at 7 am. Like the RW model, the plot of vehicle positions qualitatively changes little in short time spans. However, it can change a great deal at different times of the day: morning rush hour has very different traffic compared to 3 am.

The reader should not underestimate the complexity or difficulty of modeling realistic urban populations to the extent that is done in TRANSIMS. The system is far too complicated to provide a simple complexity measure to any component. The overall source code (written in C++) consists of over a half million lines, and the size of a dataset for a typical run is over 100 GB. All of the output has never been recorded for a single run, but it would very likely exceed tens of Terabytes. TRANSIMS was done over a period 8 years and is a work of approximately two hundred person-years [6].

4.2. Graph structure generator

For the experimental study reported in this paper, the graph generator module generates the directed graph

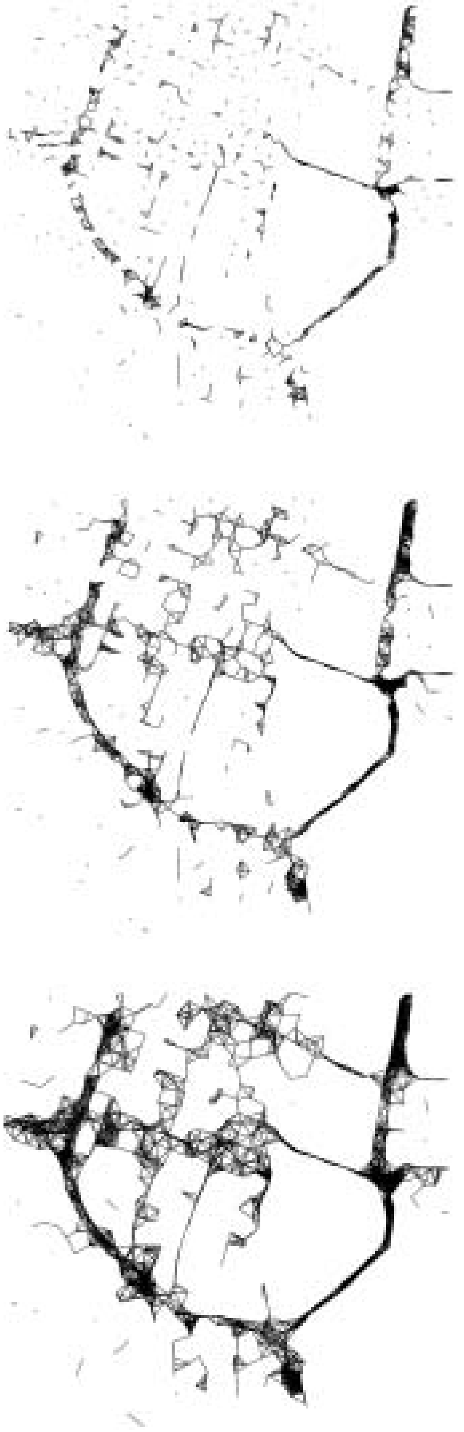


Fig. 6. Three graphs resulting from different connection radii on the same set of node positions. From top to bottom, the radii are 50 m, 75 m and 100 m. Notice that even with an unreasonably large range, there are still many components in the interior of the network.

corresponding to the ad hoc network, given the coordinates and the range of each transceiver. More specifically, for each transceiver T, the module generates the number and list of transceivers to which T has an *outgoing* edge. An outgoing edge from a transceiver T to a transceiver T_1 indicates that T_1 can receive a packet sent by T. When all transceivers have the same range, the ad hoc network can be represented by an undirected graph, where each undirected edge indicates a bidirectional link between a pair of transceivers.

The graph generation algorithm uses a simple data structure to reduce the computation time. We refer to this data structure as the *Location Table* in the following discussion. The data structure relies on the following two assumptions:

1. The X and Y coordinates of any transceiver are integers in the range 0 to XMAX and 0 to YMAX respectively.
2. Each grid point is occupied by at most one transceiver.

For TRANSIMS, these assumptions are automatically satisfied through a simple translation of the coordinate system. (TRANSIMS uses single-occupancy integer coordinates based on a geographic information system.) For the other models, the above assumptions were enforced by a preprocessing step that suitably rounds the coordinates of mobile units and shifts overlapping positions. The data used for this study had a sparse enough spatial distribution of nodes that no shifting of positions was required.

The Location Table (LT) is an array of size XMAX+1. Each entry of LT is a pointer to a list of transceiver IDs. More specifically, for each i, $0 \leqslant i \leqslant \texttt{XMAX}$, LT($i$) is a pointer to a list of transceiver IDs such that the X-coordinate of each transceiver in the list is equal to i, and the list is in *increasing* order of the Y-coordinates of the transceivers. As positions of transceivers are read from the input file, it is a simple matter to insert them into LT satisfying these conditions.

Suppose T is a transceiver with coordinates (x, y) and range r. Let T' be another transceiver with coordinates (x', y'). Then, under the L_2 distance metric, it can be verified that T has an outgoing edge to T' if and only if both of the following conditions are satisfied.

1. $x' \in [x_l, x_h]$, where $x_l = \max\{0, x - r\}$ and $x_h = \min\{x + r, \texttt{XMAX}\}$.
2. $y' \in [y_l, y_h]$, where $y_l = \max\{0, y - D\}$ and $y_h = \min\{\texttt{YMAX}, y + D\}$ with $D = \lfloor\sqrt{r^2 - (x - x')^2}\rfloor$.

So, to find the out neighbors of a transceiver T, we need to consider only those transceivers whose X-coordinate values are in the range $[x_l, x_h]$. For each such X-coordinate, say i, we can obtain a list of all the transceivers whose Y-coordinates are in the range $[y_l, y_h]$ by a simple scan of the list LT(i). This scan is facilitated by having the list in sorted order of Y coordinates. An outline of the algorithm used to generate the edges in the graph at various time steps is given in Fig. 5.

For a system with N nodes, the above algorithm has a worst-case running time of $O(N^2)$ for each time instant, since each node may have outgoing edges to $O(N)$ other nodes. Thus, if the algorithm is run for τ time instants, the worst-case time for graph generation is $O(N^2\,\tau)$. However, if the transceivers are distributed uniformly over the grid, the expected number of transceivers to which a given transceiver has outgoing edges will be significantly smaller, and the expected running time will be much better than that given by the worst-case bound. The algorithm was implemented in C++ and it works well for small to medium values of transceiver ranges. Our tests indicated that this approach is fast enough for the prototype version of the ad hoc network simulator. With larger ranges and higher node densities, more sophisticated search methods may be needed.

As mentioned above, when all the transceivers have the same range, the communication network formed by the transceivers at any time can be represented by an undirected graph. Figure 6 illustrates the graphs produced for the TRANSIMS mobility model at 7 am for three different ranges (also called connection radii). At the shortest range, namely 50 meters, the graph is composed of many small components. At 100 meters, there is one large connected component, and most of the nodes are part of this component. However, even with a range of 150 meters, the graph has a number of small components. This is a common feature of graphs based on nodes that are not uniformly distributed, and can be seen in both the TRANSIMS model and ECRM, but not Random Waypoint (Figs 6 and 8, and Table 1).

As observed in Section 3.2, it is possible to compose the modules of Fig. 1 to obtain a self-adapting ad hoc network. To illustrate this, we present results from an experiment conducted using the TRANSIMS mobility model on the section of Portland, OR, shown in the top panel of Fig. 4. Our goal was to adjust the range of each transceiver so that the degree of each transceiver in the resulting network is approximately 7. This search can be implemented by iterating the graph generator module, since the outdegree of each node is a monotonically non-decreasing function of the range. The resulting range distribution for the TRANSIMS node positions at 7 am is shown in Fig. 7. The degree distribution has 46 nodes of degree 6 and 49 nodes of degree 8. Because all of these nodes have two nodes at exactly the same range, they are unable to achieve the degree value 7. A similar explanation can be given for nodes of degree 5 and 9.

4.3. *Occlusion modification tool*

Given the set of all edges that can possibly exist within the maximum ranges of the devices (exactly what is produced by the GSG), any physical effects such as terrain, buildings, reflection, or signal attenuation can act only to reduce the members of this set. Occlusions include anything that can interrupt a physical link between nodes. For the implementation used in this study, occlusions are generated externally as a set of barriers in two dimensions; that is, the barriers are lines. The Occlusion Modification Tool (OMT) takes the set of barriers and the graph produced by the Graph Generator, and removes all the edges that cross any of the barriers. The resulting graph is the one that represents the usable physical links between nodes; the edges that were removed would never have been perceived. If E denotes the number of edges in the input graph and B denotes the number of barriers, the running time of the OMT module is $O(EB)$.

An example to illustrate the functionality of this module is shown in Fig. 8. The upper panel of this figure shows the undirected graph without any barriers. The positions of the nodes in the graph were generated using the random waypoint model. The lower panel shows the modified graph when two barriers are added. The reader will notice that many edges in the top right quadrant of the upper panel are missing in the lower panel; these edges were removed by a barrier in the upper right quadrant. Likewise, many edges to the left of the center of the upper panel are missing in the lower panel because of a barrier in the bottom left quadrant.

The graph structure generator (GSG) and the occlusion modification tool (OMT) are two of the components in our framework. In the study reported here, conceptually simple versions of these two modules have been implemented to demonstrate the usefulness of the framework. More sophisticated versions of GSG and OMT incorporating features of radio propagation models (such as reflections and signal attenuation) can indeed be used in an application based on our framework.

Table 1

Summary statistics for a sample of the test graphs. R is the range of each node (in meters), time refers to the number of seconds after the decay of initial transients for RW and ECRM and after 7 am for TRANSIMS. Size is the number of edges, D is the node degree, C is the node clustering coefficient, and N is the number of connected components in the graph. The parameter d denotes the graph diameter.

model	R	time	size	ave-D	ave-C	N	d
ECRM	50	0	1991	3.98	0.611	144	29
ECRM	50	10	1958	3.91	0.588	138	36
ECRM	50	900	2040	4.08	0.600	162	22
RW	50	0	828	1.65	0.573	421	19
RW	50	10	869	1.73	0.614	410	13
RW	50	900	860	1.72	0.587	400	11
TRANSIMS	50	0	2376	4.74	0.672	205	55
TRANSIMS	50	10	2433	4.86	0.656	191	35
TRANSIMS	50	103	2562	5.11	0.651	171	44
ECRM	75	0	4312	8.62	0.602	36	32
ECRM	75	10	4396	8.79	0.613	43	24
ECRM	75	900	4529	9.05	0.603	60	28
RW	75	0	1795	3.59	0.579	128	58
RW	75	10	1884	3.76	0.612	130	65
RW	75	900	1895	3.79	0.606	126	53
TRANSIMS	75	0	4394	8.77	0.678	58	57
TRANSIMS	75	10	4561	9.12	0.662	61	72
TRANSIMS	75	103	4844	9.66	0.686	69	88
ECRM	100	0	7575	15.1	0.617	17	31
ECRM	100	10	7752	15.5	0.620	18	22
ECRM	100	900	8009	16.0	0.617	25	39
RW	100	0	3248	6.49	0.593	31	47
RW	100	10	3291	6.58	0.596	38	42
RW	100	900	3339	6.67	0.603	41	41
TRANSIMS	100	0	6758	13.5	0.681	17	40
TRANSIMS	100	10	6952	13.9	0.683	22	44
TRANSIMS	100	103	7332	14.6	0.685	24	53
ECRM	150	0	16386	32.7	0.655	3	20
ECRM	150	10	16651	33.3	0.658	5	19
ECRM	150	900	17374	34.7	0.654	8	24
RW	150	0	7301	14.6	0.605	5	28
RW	150	10	7293	14.5	0.607	8	27
RW	150	900	7349	14.6	0.604	5	26
TRANSIMS	150	0	11997	23.9	0.701	7	23
TRANSIMS	150	10	12417	24.8	0.696	6	23
TRANSIMS	150	103	13298	26.5	0.703	8	22

5. Analysis of results

5.1. *Definitions of graph parameters*

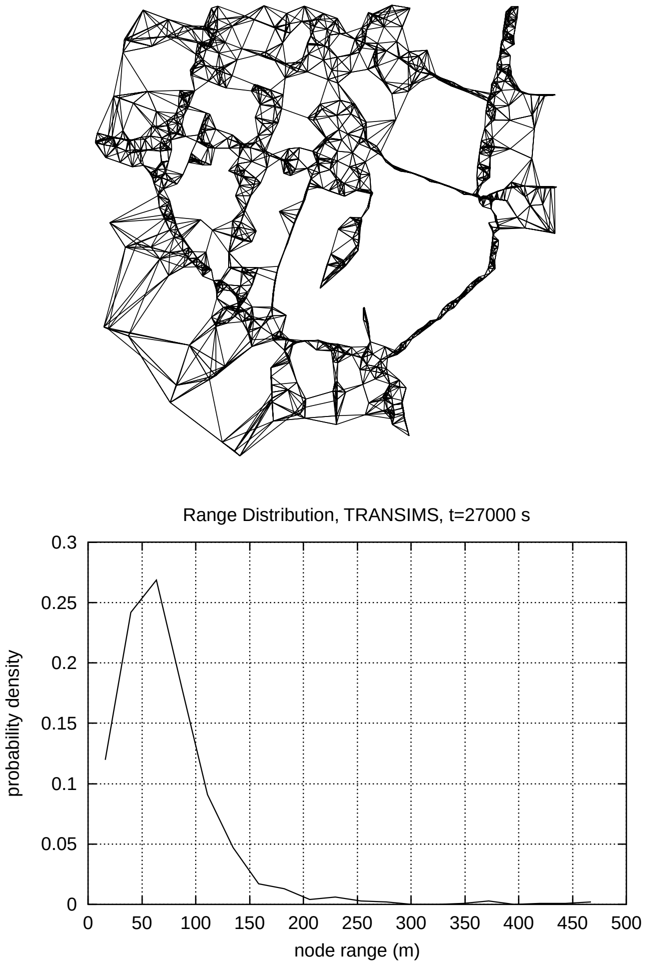
Our experimental methodology for comparing the various mobility models is based on the values of certain parameters of the graphs generated by those models. Below, we provide formal definitions of those graph theoretic parameters. Readers interested in more complete descriptions of these metrics or others should consult one of the references on introductory or advanced graph analysis [37,15,35,2].

We start with the definition of the *clustering coefficient* of a node [35]. Consider a graph $G(V, E)$ and let v be a node of G. Let N_v denote the set of nodes adjacent to v in G, not including v itself. Let $G_v(N_v, E_v)$ denote the subgraph of G induced on the node set N_v. The clustering coefficient of v, denoted by C_v, is given by

$$C_v \;=\; \frac{2\,|E_v|}{|N_v|\,(|N_v|-1)}.$$

Thus, C_v denotes the ratio of the actual number of edges in G_v to the maximum possible number of edges in G_v (which occurs when G_v is a complete graph). Obviously, C_v is a real number in the range 0.0 to 1.0. A low value of C_v indicates that the neighbors of v are not well connected to each other.

Next, we define two other graph parameters, namely the *radius* and *diameter* [15]. We first present the definitions of these parameters for connected graphs and



count	degree
1	5
46	6
902	7
49	8
3	9

Fig. 7. Self-adapting network with degree approximately 7. The upper panel is the resulting graph, the middle is the range distribution that produced it, and the lower panel is the resulting degree distribution.

extend them to graphs that are not necessarily connected.

Let $G(V, E)$ be a graph. For any pair of nodes u and v, let $d(u, v)$ denote the length of (i.e., the number of edges in) a shortest path between u and v in G. For a node v, the radius r_v is defined by

$$r_v = \max\{d(u, v) : u \in V\}.$$

The radius of the graph r_G is defined by

$$r_G = \min\{r_v : v \in V\}.$$

The diameter of the graph d_G is defined by

$$d_G = \max\{r_v : v \in V\}.$$

Equivalently, the diameter of the graph d_G can also be defined by

$$d_G = \max\{d(u, v) : u, v \in V\}.$$

As is well known, the diameter of a connected ad hoc network is an upper bound on the number of hops needed for a message to travel from one node to another.

We can extend the definitions of radius and diameter to graphs that are not necessarily connected as follows. The radius of a graph G can be defined as the minimum value among the radii of all the connected components

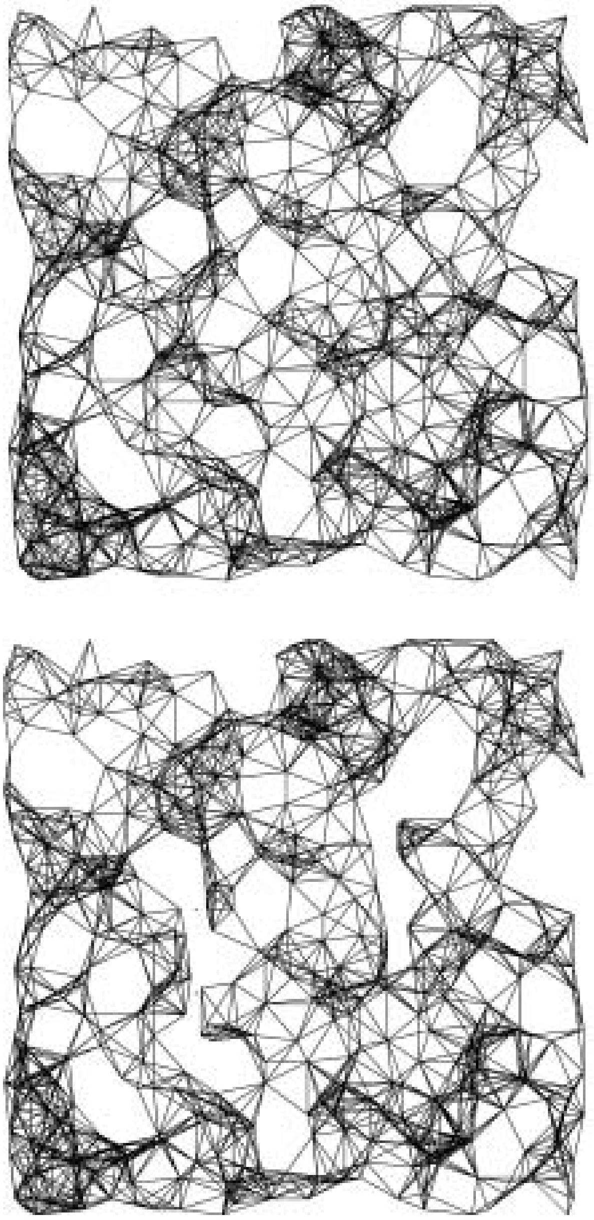


Fig. 8. The upper panel shows a timestep of the RW model, and the lower panel shows the same with two vertical barriers erected in the lower-left and the upper right.

in G. Similarly, the diameter of a graph G can be defined as the maximum value among the diameters of all the connected components in G. We use these extended definitions in the following subsection.

5.2. Analysis

It was observed that the different mobility models produce patterns of motion or even instantaneous snapshots of node positions that are very different. Further, it was pointed out that variations in the ranges of nodes can also produce significantly different results. In this section we make an attempt at quantifying these differences.

Time in TRANSIMS is measured in seconds from 7 am (itself 25200 seconds from midnight). As synthetic mobility models, both RW and ECRM required some time from the beginning of the simulation for sta-

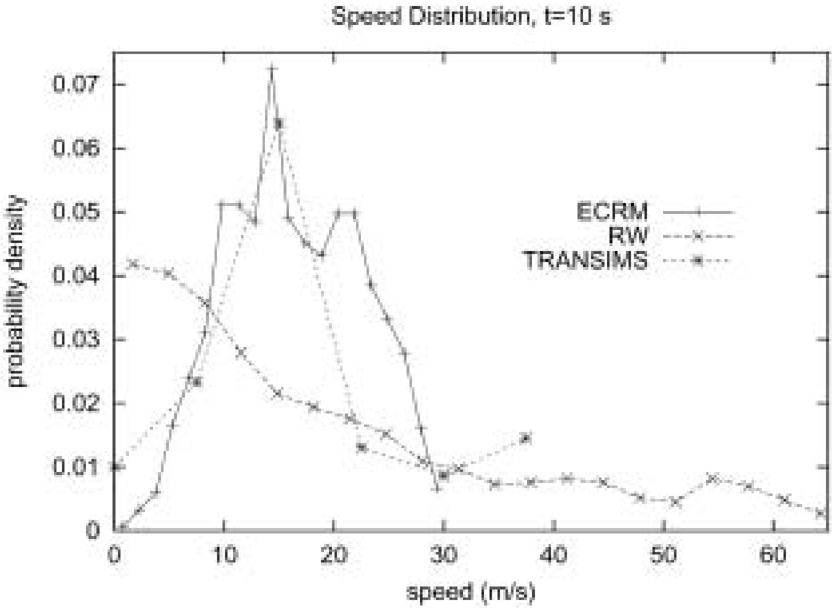


Fig. 9. Distributions of node speed for each of the three mobility models at a single instant of time.

bilization of artifacts caused by a selection of initial positions not characteristic of the model. The RW model was given 100 seconds, and ECRM 200 seconds. Time values quoted in the figures and in the discussion below are relative to those times.

All of the mobility models were calibrated to produce the same average speed at each timestep such that they would match the average speed of the extracted TRANSIMS data, namely 16.4 meters per second. Figure 9 shows the full distributions of speed producing those averages for a single timestep: $t = 10$ seconds (210 seconds after simulation start for ECRM, 110 seconds after start for RW, and 27010 seconds for TRANSIMS). ECRM and TRANSIMS have similar profiles, but RW is more flat with a much longer tail. As previously mentioned, the fact that in RW slow moving nodes stay in the slow-moving state much longer than fast moving nodes creates a distribution that is weighted towards slow speeds. Thus to reach the same average, the long tail visible in Fig. 9 is required.

The remainder of this section provides a comparison of the experimental results obtained for the three mobility models. As mentioned earlier, the comparison relies on various graph theoretic measures for the networks generated by the mobility models.

A widely used measure of a graph is the degree distribution. Here, the degree of a node is defined as the number of out-neighbors of that node. Figure 10 illustrates the degree distributions for each of the three models at a single timestep (namely, $t = 10$ seconds, which was also used in the speed distributions shown in Fig. 9) for three different range values. As with the speed distributions, RW is less like TRANSIMS than ECRM. However, while both TRANSIMS and ECRM have skewed distributions centered about the same value, the TRANSIMS distribution is skewed towards low degree, while the ECRM distribution is skewed towards high degree. The RW distribution is cleanly peaked at about half of the degree of the other two. The TRANSIMS distributions also have noticeably longer tails than the other two.

The degree distributions for RW change only by a small amount throughout the simulation (after the initial settle-in period) because the node positions maintain a nearly constant distribution with respect to the simulation area. The TRANSIMS data also changes little within a few hundred seconds, but as the overall traffic changes in the metropolitan area, changes become more pronounced. For example, while there are only 1000 vehicles in the simulation area at 7 am, at 7:30 am, there are almost double that number; as a consequence, the graph formed by the nodes would change significantly. Because ECRM has travelers in groups that have macroscopic movements, there can be significant changes with respect to time as whole groups become disconnected from the rest of the graph. Within a group, the degree distribution maintains a more constant form (like the RW model). Figure 11 shows ECRM degree

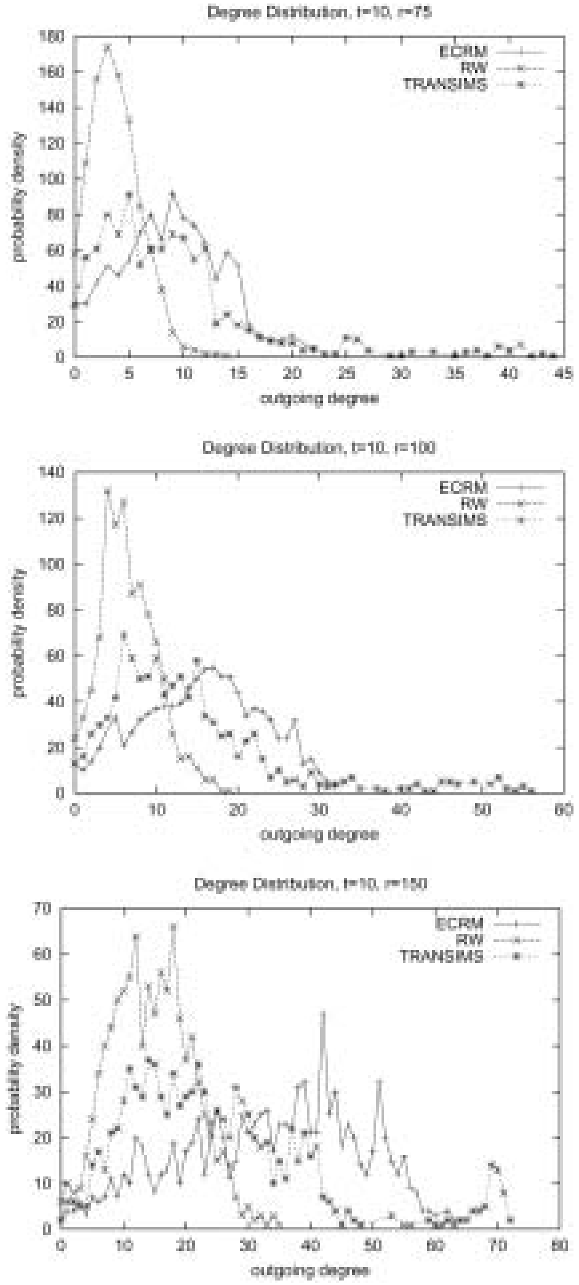


Fig. 10. Degree distributions for the three mobility models at a single instant of time for three different connection radii. The top panel is for $R = 75$ m, the middle panel for $R = 100$ m, and the bottom panel for $R = 150$ m.

distributions at three timesteps. There is little change in 10 seconds, but after 900 seconds (i.e., 15 minutes), there is a noticeable flattening of the distribution. Two of the distributions correspond to the snapshots shown in Fig. 3.

A currently active line of research is the study of scale-free networks [2]. Many diverse systems have been shown to have properties associated with scale-free networks. A scale-free network is defined in [5] to be one for which the degree distribution has a power-law tail:

$$P(k) \sim k^{-\gamma} \tag{2}$$

where $P(k)$ is the distribution function that gives the probability that a randomly selected node has degree exactly k and γ is a constant. Thus, if a degree distribution can be reasonably fit to a power law, then it is one that satisfies Eq. (2), and the network is therefore scale-free. Such a degree distribution does not have a characteristic scale. It can be seen from the previous degree plots that the networks generated by the different mobility models are not scale-free: they have a characteristic degree. It is therefore meaningful to consider the typical degree of a mobile network (although protocols that imply the use of nonuniform ranges or the presence of occlusions may change this situation). The top panel of Fig. 12 shows the data presented in the middle panel of Fig. 10 on a log scale. The middle panel of Fig. 12 shows the effect of transceiver range on the degree distribution. All three models behave the same way, with increasing range stretching the degree distribution to higher values, but RW is the smoothest and TRANSIMS the noisiest (not shown). There is also no time variation in the qualitative differences between ranges (also not shown). Time varying aspects of networks have not been considered much in recent research, but they are of paramount importance for mobile networks. For a given range, the temporal variation of the degree distribution is shown in the bottom panel of Fig. 12. It can be seen from that figure that the RW distributions do not change significantly with time. The ECRM model was already shown to have noticeable time dependence in Fig. 11. For TRANSIMS, note that timesteps were selected so as to have close to 1000 transceivers in each timestep, and nodes are tightly constrained to the roads. Thus, when the density of nodes changes dramatically, as it does in any urban area with respect to time, there will be a corresponding change in all of the graph parameters. However, when the density remains constant, these results suggest that the degree distribution and the clustering coefficient distribution (discussed below) will be static. Of course, degree distributions by no means completely characterize a mobile network; this observation will be discussed in Section 6.

Figure 13 shows the clustering coefficient distributions for all nodes in each graph at the timestep labeled

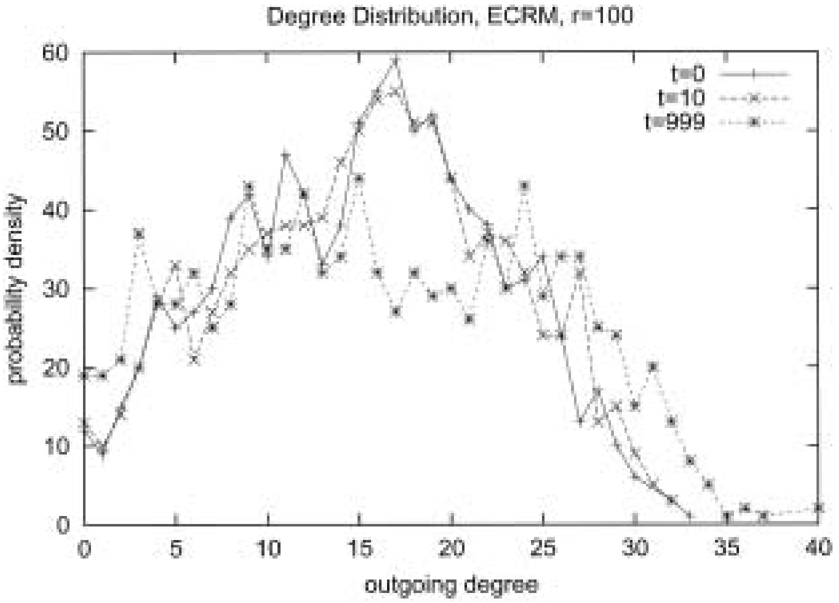


Fig. 11. Degree distributions for the ECRM model at three instances of time. Because there is a possibly significant difference in the positions of nodes with respect to time, the corresponding degree distribution may also change significantly, as shown.

10 seconds (see above) for each of the mobility models when each node has a range of 150 meters. The clustering coefficients for lower ranges were too noisy to draw any clear conclusions. The ECRM distribution is more tightly peaked than the other two, and there is a clear difference in the average clustering coefficients of the models.

A graph-level summary of statistics is shown in Table 1. Included there are several time snapshots for each scenario (mobility model and connection range). The columns labeled 'size', 'ave-D', 'ave-C', 'N' and 'd' represent respectively the number of edges in the graph, the average outdegree, the average clustering coefficient, the number of connected components and the diameter. Each graph in the RW and ECRM models has exactly 1000 nodes; The TRANSIMS graphs have 1002, 1002 and 1003 nodes respectively. The graph size has a clear trend: for a given connection range, the RW model usually has the lowest, while the ECRM usually has the highest number of edges. The average outdegree has a similar trend, although ECRM and TRANSIMS do overlap slightly more. Not unexpectedly, both graph size and average out-degree increase monotonically with connection range. The average clustering coefficients range from 0.573 to 0.703. In general, RW graphs have lower and TRANSIMS graphs have higher values, and lower connection radii have lower values. However, neither of these statements is true in all cases. The number of connected components in the graphs decreases with increase in the range, as would be expected; however, in these experiments, the number of components did not drop down to one. (This may be partly due to boundary effects.)

For the graphs created from the mobility models, the radius measurement is not very meaningful because most of the graphs produced have small components consisting of only one or two nodes. The radius for almost all graphs in Table 1 is 1, and it is not reported in the table. The diameter values range from 11 to 88. (For comparison, the diameter of a square grid with 1024 nodes is 62.) Graphs generated using the RW model cover the range uniformly. Graphs generated using the ECRM model tend toward the lower two thirds and those generated using TRANSIMS to the higher two thirds. Graphs created with a range of 50 meters occupy the lower half of the values, but no significant conclusions can be drawn for other ranges.

The diameter values shown in Table 1 exhibit a behavior that merits explanation. For each of the mobility models, the table shows that increasing the range may increase the diameter of the graph. (For example, at time 900, the diameters under the RW models for the two range values of 50 and 75 meters are 11 and 53 respectively.) To understand this, the reader should bear in mind that the diameter values presented in the table represent the largest of the diameters of the connected components. When the range is 50 meters, the number of nodes in each component is small. So, the diameter

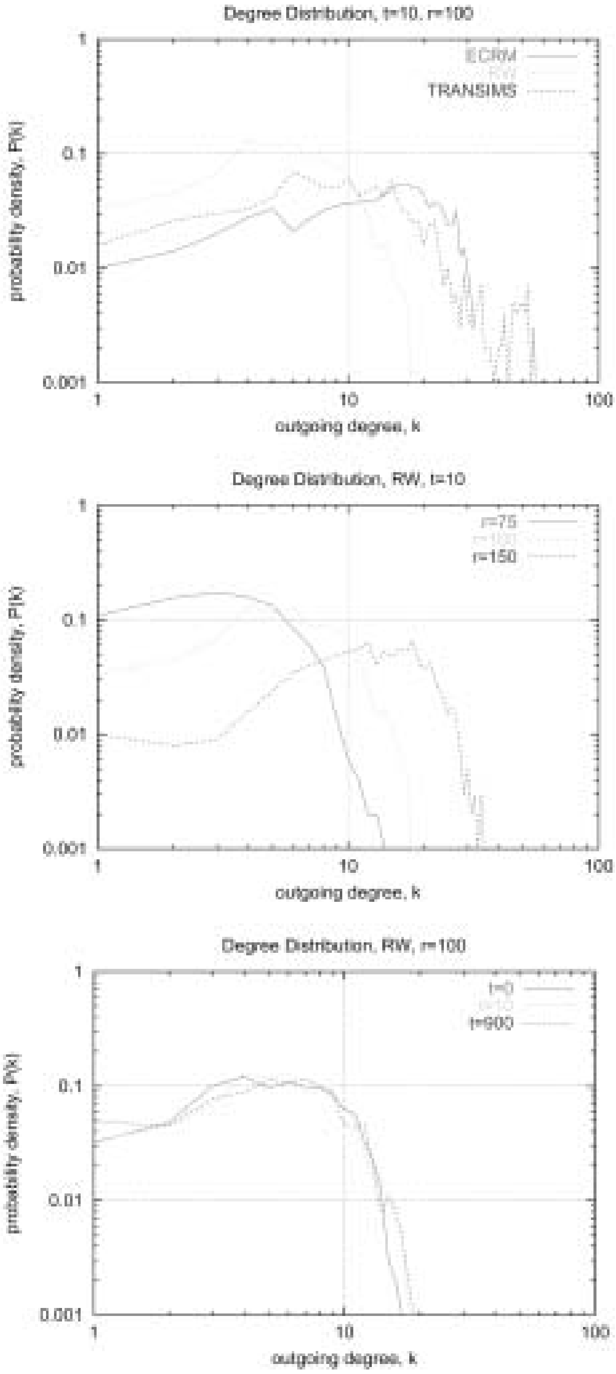


Fig. 12. Degree distributions shown on logarithmic axes to more clearly illustrate their non-scale-free nature. The top panel is the same as the middle panel of Fig. 10, but on log axes. The middle panel shows the effect of range in RW at a fixed time, and the bottom panel shows the time variation at a fixed range for RW.

is also small. When the range is increased to 75 meters, several small components get merged together with a relatively small number of additional edges. This creates sparsely connected larger components with large diameters. As the range is increased to 150 meters, many new edges are introduced, and the connections between components becomes denser. This causes the diameter values (as well as the number of components)

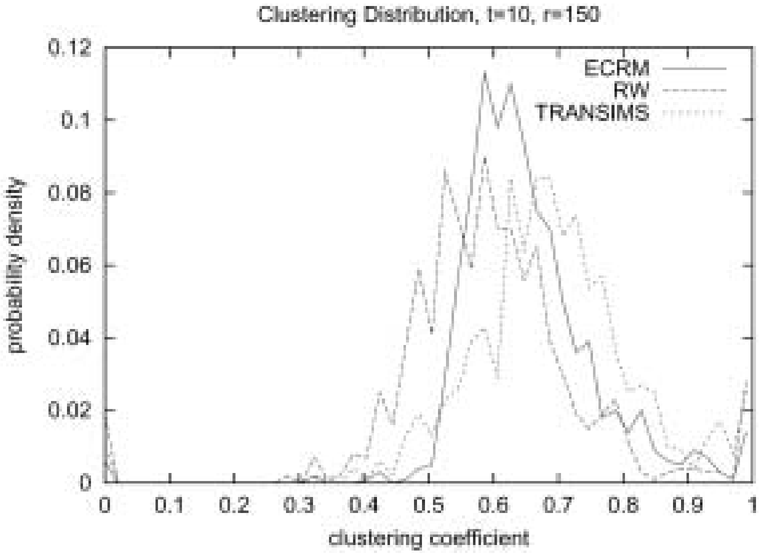


Fig. 13. Clustering coefficient distributions for each of the three mobility models with a connection range of 150 m.

to decrease.

6. Ongoing and future work

We have described the components of a framework for generating mobility patterns and traffic activities that can be used to drive simulations of ad hoc communication networks. The experimental analysis presented in the previous section showed that there are measurable differences between synthetic and realistic mobility patterns. The work described above is ongoing, and a number of enhancements are currently being considered and implemented. We are also carrying out a detailed analysis of the structural properties of mobile networks. We mention some of the more significant ones below.

1. The most significant extension is to fully incorporate TRANSIMS-based mobility generating mechanisms in our framework. While this is not conceptually difficult, a number of implementation difficulties must be addressed. For example, ad hoc network simulators discussed in the open literature cannot handle more than about 1500 transceivers while the data generated from Portland, Oregon, has 1.5 million travelers. In other words, transportation simulators handle numbers of travelers that are three orders of magnitude larger than what can be handled by current ad hoc network simulators. Thus, issues such as selecting a subset of travelers such that they do not form small disjoint components, efficiently updating positions of a large number of users, etc. must be carefully considered. We have recently demonstrated a distributed PAS that can handle millions of nodes in approximately real time, but it is not yet full featured. Thus, the modeling framework by itself does not impose a limit on the size of networks that can be modeled and simulated.
2. We are investigating methods for generating realistic communication activity patterns that can be fed into an ad hoc network simulator. We are using a statistical methodology developed as a part of TRANSIMS to accomplish this goal, and early results are encouraging.
3. Having more edges in an ad hoc network is desirable from the point of view of connectivity. For a given placement of nodes, if it is necessary to increase the number of edges, then the ranges of one or more nodes (i.e., the transmission powers used by those nodes) must be increased. Such a step may increase the interference among the nodes and consequently have a negative impact on the performance of the network. Therefore, controlling transmission powers of nodes is important from the point of view of network performance. Also, the dynamics of the packets in the network also changes substantially with varying mobility models. This topic will be discussed in detail in a different paper [7]. Figure 14 shows a sample of such effects, plotting the average frac-

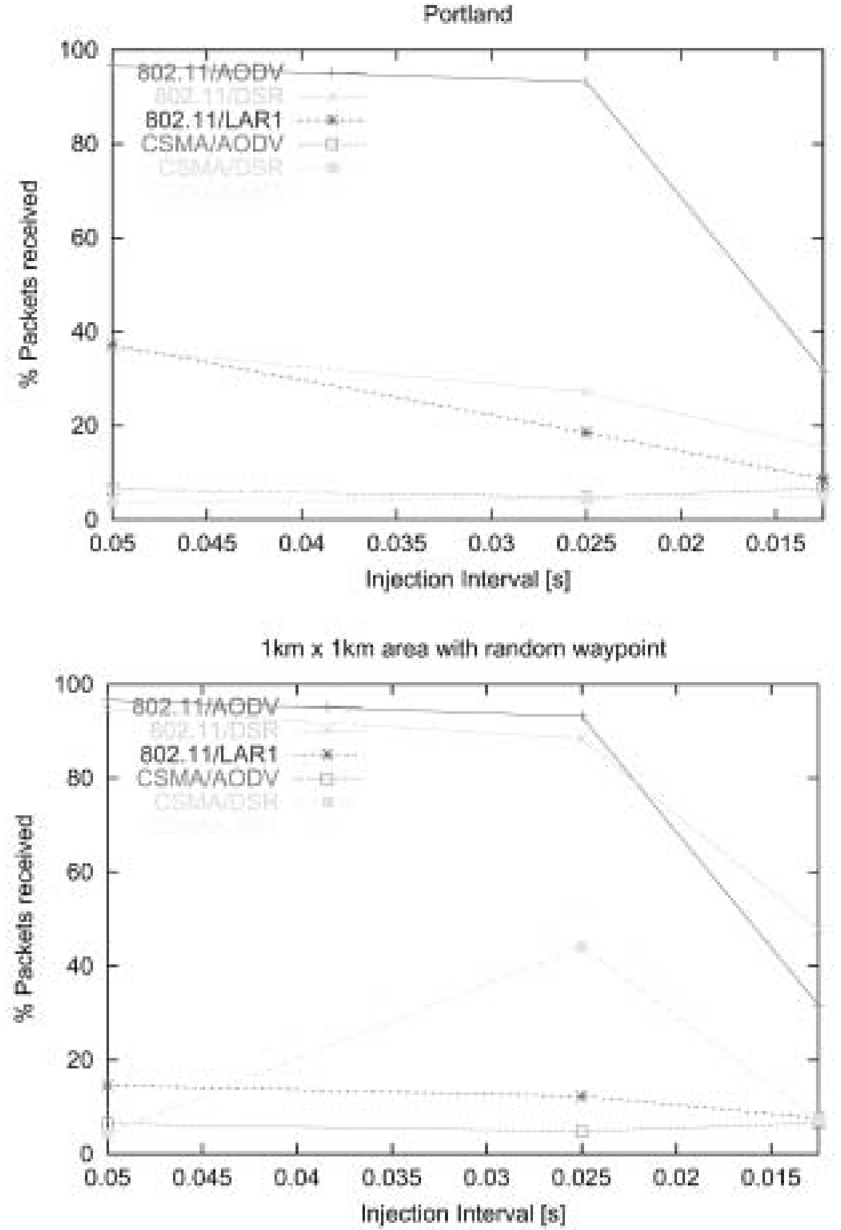


Fig. 14. The importance of mobility models on routing protocols.

tion of packets received by a destination for ten randomly chosen node pairs at each injection rate. This study was completed using GloMoSim [3] as PAS. The MAC layer protocols were either 802.11 or CSMA, and the routing protocols were AODV, DSR, or LAR1 [1,22,25,29,30]. It is obvious from Fig. 14 that there are substantial differences in performance caused by the mobility model. A complete analysis of the influence of mobility models on protocols will be presented in [7].

4. The main reason for computing the graph parameters is to understand how the underlying structural properties of time varying graphs affect the performance of the network. We note that in the case of graphs arising from mobility models, time varying properties are extremely important, as opposed to the systems described in [2]. Much more work needs to be done in this respect, and a paper that explores the time variation of the struc-

tural properties of mobile networks is in preparation [17].

Acknowledgments

We thank the referees for a careful reading of the paper. Their suggestions helped us to clarify several technical issues and also to significantly improve the presentation.

References

[1] Wireless LAN Medium Access Control (MAC) and Physical (PHY) Layer Specification, *IEEE Standard IEEE* **802**(11) (June, 1999).

[2] R. Albert and A.-L. Barabási, Statistical Mechanics of Complex Networks, Reviews of Modern Physics, 2001, citeseer.nj.nec.com/442178.html.

[3] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia and M. Gerla, GloMoSim: A Scalable Network Simulation Environment, UCLA Computer Science Department Technical Report 990027, May, 1999.

[4] S. Bajaj, L. Breslau, D. Estrin, K. Fall, S. Floyd, P. Haldar, M. Handley, A. Helmy, J. Heidemann, P. Huang, S. Kumar, S. McCanne, R. Rejaie, P. Sharma, K. Varadhan, Ya Xu, H. Yu and D. Zappala, Improving Simulation for Network Research. USC Technical Report 99-702, March, 1999.

[5] A.-L. Barabási and R. Albert. Emergence of Scaling in Random Networks, *Science* **286** (1999), 509–512.

[6] C. Barrett, R. Beckman, K. Berkbigler, K. Bisset, B. Bush, K. Campbell, S. Eubank, K. Henson, J. Hurford, D. Kubicek, M. Marathe, P. Romero, J. Smith, L. Smith, P. Speckman, P. Stretz, G. Thayer, E. Van Eeckhout and M. Williams, TRANSIMS (TRansportation ANalysis SIMulation System) 3.0 LA-UR-00-1724, 1725, 1755, 1766, and 1767. Los Alamos National Laboratory, 2000, URL: http://transims.tsasa.lanl.gov.

[7] C. Barrett, M. Drozda, M.V. Marathe and J.P. Smith, *Mobility and Performance of Wireless Ad-hoc Network Protocols*, In preparation, 2003.

[8] C. Barrett, M.V. Marathe, S.S. Ravi, C.M. Reidys and J.P. Smith, Advances in Simulation-based Design and Analysis of Ad-Hoc Networks, Los Alamos Technical report, LAUR-01-1644, 2001.

[9] C. Barrett, M. Drozda, A. Marathe and M. Marathe, *Characterizing the Interaction Between Routing and MAC Protocols in Ad-Hoc Networks*, Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC 2002), Lausanne, Switzerland, June, 2002.

[10] C. Barrett, M. Drozda, A. Marathe and M. Marathe, *Analyzing Interaction Between Network Protocols, Topology and Traffic in Wireless Radio Networks*, Proc. IEEE Wireless Communications and Networking Conference (WCNC'03), New Orleans, Louisiana, IEEE Press, 2003.

[11] S. Basagni, I. Chlamtac, V.R. Syrotiuk and B.A. Woodward, *A Distance Routing Effect Algorithm for Mobility* (*DREAM*), ACM/IEEE International Conference on Mobile Computing and Communication, MobiCom'98, 1998, pp. 76–84.

[12] M. Bergamo, R.R. Hain, K. Kasera, D. Li, R. Ramanathan and M. Steenstrup, MMWN preliminary design (draft), 1998, URL: ftp.bbn.com/pub/ramanath/mmwn-design.ps.

[13] C. Bettstetter, *Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks*, ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, (MSWiM'01), Rome, Italy, July, 2001, pp. 19–27.

[14] T. Camp, J. Boleng and V. Davies, *Mobility Models for Ad Hoc Network Simulations*, To appear in Wireless Communication & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, 2002.

[15] G. Chartrand and L. Lesniak, Graphs and Digraphs. Wadsworth and Brooks/Cole Publishers, Monterey, CA, 1986.

[16] S. Das, R. Castaneda and J. Yan, Simulation Based Performance Evaluation of Mobile, Ad Hoc Network Routing Protocols, *Mobile Networks and Applications* **5** (July, 2000), 179–189.

[17] D.C. Englehart, M.V. Marathe, M. Morin, S.S. Ravi and J.P. Smith, *Graph structure of Ad-hoc Mobility Models*, In preparation, 2003.

[18] M. Faloutsos, P. Faloutsos and C. Faloutsos, *On Power-law Relationships of the Internet Topology*, Proc. SIGCOMM, 1999, pp. 251–262.

[19] R. Govindan and H. Tangmunarunkit, *Heuristics for Internet Map Discovery*, Proc. IEEE INFOCOM, Tel Aviv, Israel, 2000, pp. 1371–1380.

[20] Z.J. Haas, *A New Routing Protocol for the Reconfigurable Wireless Networks*, Proc. 6th IEEE International Conf. on Universal Personal Communication, (ICPUC), 1997, pp. 562–566.

[21] X. Hong, M. Gerla, G. Pei and C. Chiang, *A Group Mobility Model for Ad hoc Wireless Networks*, Proc. 2nd ACM International Workshop on Modeling and Simulation of Wireless and Mobile Systems, (MSWiM'99), Seattle, WA, Aug. 1999, pp. 53–60.

[22] D. Johnson and D. Maltz, Dynamic Source Routing in Ad Hoc Wireless Networks, in: *Mobile Computing*, Tomasz Imielinski and Hank Korth, eds, Chapter 5, Kluwer Academic Publishers, 1996, pp. 153–181.

[23] C.E. Jones, K.M. Sivalingam, P. Agrawal and J.C. Chen, A Survey of Energy Efficient Network Protocols for Wireless Networks, *Wireless Networks* **7** (2001), 343–358.

[24] L. Kleinrock and J. Sylvester, *Optimum Transmission Radii for Packet Radio Networks or Why Six is a Magic Number*, Proc. IEEE National Telecommunications Conference, Birmingham, AL, Dec. 1978, pp. 4.3.1–4.3.5.

[25] Y.-B. Ko and N. Vaidya, *Location Aided Routing* (*LAR*) *in Mobile Ad Hoc Networks*, Proc. ACM/IEEE International Conference on Mobile Computing and Communication, (MobiCom'98), 1998, pp. 66–75.

[26] K. Leung, W. Massey and W. Whitt, Traffic Models for Wireless Communication Networks, *Proc. IEEE INFOCOM* **3** (1994), Toronto, CA, 1029–1037.

[27] B. Liang and Z.J. Haas, Predictive Distance-Based Mobility Management for PCS Networks, *Proc. IEEE INFOCOM* **3** (1999), New York, NY, 1377–1384.

[28] E.L. Lloyd, R. Liu, M.V. Marathe, R. Ramanathan and S.S. Ravi, Algorithmic Aspects of Topology Control Problems for Ad Hoc Networks, *Proc. Third ACM International Symposium on Mobile Ad Hoc Networking and Computing* (MobiHoc 2002), Lusanne, Switzerland, June 2002, pp. 123–134.

[29] A. Nasipuri, J. Zhuang and S.R. Das, A Multichannel CSMA MAC Protocol for Multihop Wireless Networks, Proc. IEEE

Wireless Communications and Networking Conference, (WCNC'99), Sept., 1999.

[30] C.E. Perkins and E.M. Royer, Ad-hoc On-Demand Distance Vector Routing, *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, Feb. 1999, pp. 90–100.

[31] R. Ramanathan and R. Rosales-Hain, *Topology Control of Multihop Wireless Networks using Transmit Power Adjustment*, Proc. IEEE INFOCOM, Tel Aviv, Israel, MAr.2000, pp. 404–413.

[32] S. Ramanathan and M. Steenstrup, Hierarchically-Organized, Multihop Mobile Wireless Networks for Quality-of-Service Support, *Mobile Networks and Applications* **3** (1998), 101–119.

[33] E.M. Royer, P.M. Melliar-Smith and L.E. Moser, An Analysis of the Optimum Node Density for Ad hoc Mobile Networks. Submitted for publication, citeseer.nj.nec.com/475080.html.

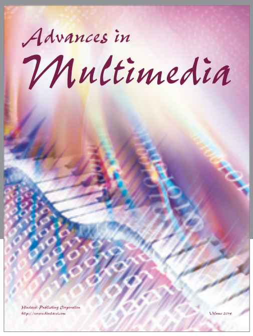
[34] S.H. Strogatz, Exploring Complex Networks, *Nature* **410** (2001), 268–276.

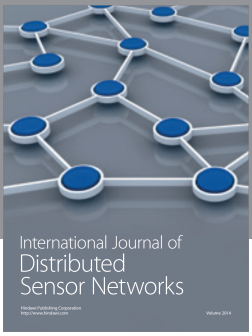
[35] D.J. Watts, *Small Worlds: The Dynamics of Networks between Order and Randomness*, Princeton University Press, Princeton, NJ, 1999.

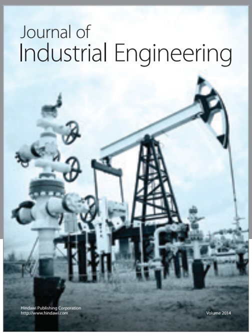
[36] D.J. Watts and S.H. Strogatz, Collective Dynamics of Small-world Networks, *Nature* **393** (1998), 440–442.

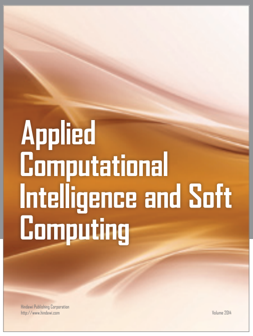
[37] D.B. West, *Introduction to Graph Theory*, Prentice Hall, Upper Saddle River, NJ, 2001.

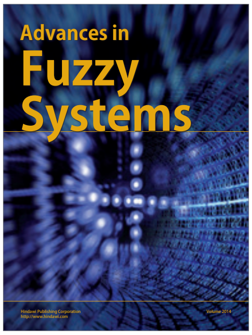
[38] M. Zonoozi and P. Dassanayake, User Mobility Modeling and Characterization of Mobility Patterns, *IEEE Trans. Selected Areas in Communications* **15** (Sept. 1997), 1239–1252.











Hindawi

Submit your manuscripts at
http://www.hindawi.com