WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2002; **2**:483–502 (DOI: 10.1002/wcm.72)

A survey of mobility models for ad hoc network research‡

Tracy Camp,*,† Jeff Boleng and Vanessa Davies

Department of Mathematics and
Computer Sciences
Colorado School of Mines
Golden, CO
U.S.A.

Summary

In the performance evaluation of a protocol for an ad hoc network, the protocol should be tested under realistic conditions including, but not limited to, a sensible transmission range, limited buffer space for the storage of messages, representative data traffic models and realistic movements of the mobile users (i.e. a mobility model). This paper is a survey of mobility models that are used in the simulations of ad hoc networks. We describe several mobility models that represent mobile nodes whose movements are independent of each other (i.e. entity mobility models) and several mobility models that represent mobile nodes whose movements are dependent on each other (i.e. group mobility models). The goal of this paper is to present a number of mobility models in order to offer researchers more informed choices when they are deciding on a mobility model to use in their performance evaluations. Lastly, we present simulation results that illustrate the importance of choosing a mobility model in the simulation of an ad hoc network protocol. Specifically, we illustrate how the performance results of an ad hoc network protocol drastically change as a result of changing the mobility model simulated. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS
ad hoc networks
mobility models
entity mobility models
group mobility models

*Correspondence to: Tracy Camp, Department of Mathematics and Computer Sciences, Colorado School of Mines, Golden, CO 80401, U.S.A.
†E-mail: tcamp@mines.edu
‡Research group's URL is http://toilers.mines.edu.
Contract/grant sponsor: NSF; contract/grant numbers: ANI-9996156 and ANI-0073699.

1. Introduction

In order to thoroughly simulate a new protocol for an ad hoc network, it is imperative to use a mobility model that accurately represents the mobile nodes (MNs) that will eventually utilize the given protocol. Only in this type of scenario is it possible to determine whether the proposed protocol will be useful when implemented. Currently, there are two types of mobility models used in the simulation of networks: traces and synthetic models [1]. Traces are those mobility patterns that are observed in real-life systems. Traces provide accurate information, especially when they involve a large number of participants and an appropriately long observation period. However, new network environments (e.g. ad hoc networks) are not easily modeled if traces have not yet been created. In this type of situation, it is necessary to use synthetic models. Synthetic models attempt to realistically represent the behaviors of MNs without the use of traces. In this paper, we present several synthetic mobility models that have been proposed for (or used in) the performance evaluation of ad hoc network protocols.

A mobility model should attempt to mimic the movements of real MNs. Changes in speed and direction must occur and they must occur in reasonable time slots. For example, we would not want MNs to travel in straight lines at constant speeds throughout the course of the entire simulation because real MNs would not travel in such a restricted manner. In Section 2, we discuss seven different synthetic entity mobility models for ad hoc networks:

1. *Random Walk Mobility Model (including its many derivatives)*: A simple mobility model based on random directions and speeds.
2. *Random Waypoint Mobility Model*: A model that includes pause times between changes in destination and speed.
3. *Random Direction Mobility Model*: A model that forces MNs to travel to the edge of the simulation area before changing direction and speed.
4. *A Boundless Simulation Area Mobility Model*: A model that converts a 2-D rectangular simulation area into a torus-shaped simulation area.
5. *Gauss–Markov Mobility Model*: A model that uses one tuning parameter to vary the degree of randomness in the mobility pattern.
6. *A Probabilistic Version of the Random Walk Mobility Model*: A model that utilizes a set of probabilities to determine the next position of an MN.
7. *City Section Mobility Model*: A simulation area that represents streets within a city.

There are other synthetic entity mobility models available for the performance evaluation of a protocol in a cellular network or personal communication system (PCS). Although some of these mobility models could be adapted to an ad hoc network, this paper focuses on those models that have been proposed for (or used in) the performance evaluation of an ad hoc network.

In Section 3, we present five group mobility models that allow researchers to simulate situations in which the MNs' decisions on movement depend upon the other MNs in the group.

1. *Exponential Correlated Random Mobility Model*: A group mobility model that uses a motion function to create movements.
2. *Column Mobility Model*: A group mobility model in which the set of MNs form a line and are uniformly moving forward in a particular direction.
3. *Nomadic Community Mobility Model*: A group mobility model in which a set of MNs move together from one location to another.
4. *Pursue Mobility Model*: A group mobility model in which a set of MNs follow a given target.
5. *Reference Point Group Mobility Model*: A group mobility model in which group movements are based upon the path traveled by a logical center.

In all five group mobility models, random motion of each individual MN within a given group occurs.

In Section 4, we illustrate that a mobility model has a large effect on the performance evaluation of an ad hoc network protocol. In other words, we show how the performance results of an ad hoc network protocol significantly change when the mobility model in the simulation is changed. The results presented prove the importance of choosing an appropriate mobility model (or models) for a given performance evaluation.

We survey a number of synthetic mobility models used in ad hoc network simulations in this paper. The details of the models provide a good resource to researchers when they are deciding upon a mobility model to use in their performance evaluations. In addition, implementations of all the mobility models described in this paper (except Exponential Correlated Random Mobility Model) are available at *http://toilers.mines.edu*.

2. Entity Mobility Models

In this section, we present seven mobility models that have been proposed for (or used in) the performance evaluation of an ad hoc network protocol. The first two models presented, the Random Walk Mobility Model and the Random Waypoint Mobility Model, are the two most common mobility models used by researchers. Thus, we discuss these two models in more depth than the other five models presented.

2.1. Random Walk

2.1.1. Overview

The Random Walk Mobility Model was first described mathematically by Einstein in 1926 [2]. Since many entities in nature move in extremely unpredictable ways, the Random Walk Mobility Model was developed to mimic this erratic movement [3]. In this mobility model, an MN moves from its current location to a new location by randomly choosing a direction and speed in which to travel. The new speed and direction are both chosen from predefined ranges, [*speedmin, speedmax*] and $[0, 2\pi]$, respectively. Each movement in the Random Walk Mobility Model occurs in either a constant time interval t or a constant distance traveled d, at the end of which a new direction and speed are calculated. If an MN that moves according to this model reaches a simulation boundary, it 'bounces' off the simulation border with an angle determined by the incoming direction. The MN then continues along this new path.

Many derivatives of the Random Walk Mobility Model have been developed including the 1-D, 2-D, 3-D and d-D walks. In 1921, Polya proved that a random walk on a 1-D or 2-D surface returns to the origin with complete certainty, that is, a probability of 1.0 [4]. This characteristic ensures that the random walk represents a mobility model that tests the movements of entities around their starting points, without worry of the entities wandering away never to return.

The 2-D Random Walk Mobility Model is of special interest, since the Earth's surface is modeled using a 2-D representation. Figure 1 shows an example of the movement observed from this 2-D model. The MN begins its movement in the center of the $300\ \text{m} \times 600\ \text{m}$ simulation area or position (150, 300). At each point, the MN randomly chooses a direction between 0 and 2π and a speed between 0 and 10 m s^{-1}. The MN is allowed to travel for 60 s before changing direction and speed. In the Random Walk Mobility Model, an MN may change direction after traveling a specified distance instead of a specified time. We illustrate this variation of the model in Figure 2. In this example, the MN travels for a total of 10 steps (instead of 60 s) before changing its direction and speed. Unlike Figure 1, each movement of the MN in Figure 2 is exactly the same distance.

The Random Walk Mobility Model is a widely used mobility model (e.g. [5–8]), which is sometimes referred to as Brownian Motion. In its use, the model is sometimes simplified. For example, Basagni *et al.* [9] simplified the Random Walk Mobility Model by assigning the same speed to every MN in the simulation.

2.1.2. Discussion

The Random Walk Mobility Model is a memoryless mobility pattern because it retains no knowledge

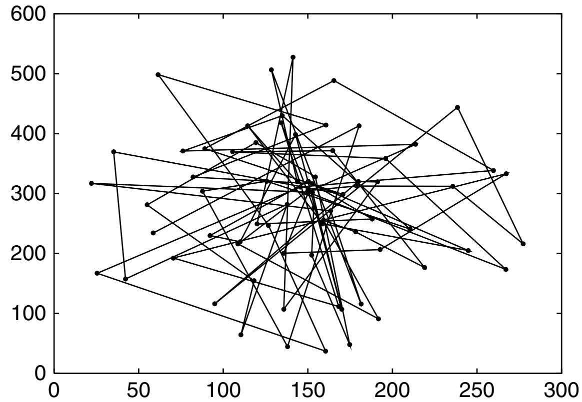

Fig. 1. Traveling pattern of an MN using the 2-D Random Walk Mobility Model (time).

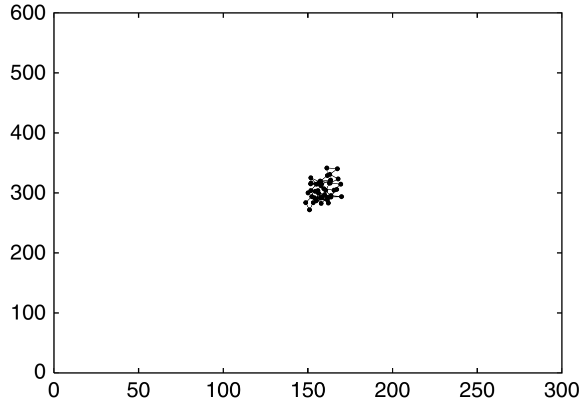

Fig. 2. Traveling pattern of an MN using the 2-D Random Walk Mobility Model (distance).

concerning its past locations and speed values [10]. The current speed and direction of an MN is independent of its past speed and direction [11]. This characteristic can generate unrealistic movements such as sudden stops and sharp turns (see Figure 1). (Other models, such as the Gauss–Markov Mobility Model, which we discuss in Section 2.5, can fix this discrepancy.)

If the specified time (or specified distance) an MN moves in the Random Walk Mobility Model is short, then the movement pattern is a random roaming pattern restricted to a small portion of the simulation area. Some simulation studies using this mobility model (e.g. References [6,9]) set the specified time to one clock tick or the specified distance to one step. Figure 2 illustrates the static nature obtained in the Random Walk Mobility Model when the MN is allowed to move 10 steps (not one) before changing direction; as shown, the MN does not roam far from its initial position. In summary, if the goal of the performance investigation is to evaluate a semi-static network, then the parameter to change an MNs' direction should be given a small value. Otherwise, a larger value should be used.

2.2. Random Waypoint

2.2.1. Overview

The Random Waypoint Mobility Model includes pause times between changes in direction and/or speed [12]. An MN begins by staying in one location for a certain period of time (i.e. a pause time). Once this time expires, the MN chooses a random destination in the simulation area and a speed that is uniformly distributed between [*minspeed, maxspeed*]. The MN then travels toward the newly chosen destination at the selected speed. Upon arrival, the MN pauses for a specified time period before starting the process again.

Figure 3 shows an example traveling pattern of an MN using the Random Waypoint Mobility Model starting at a randomly chosen point or position (133, 180); the speed of the MN in the figure is uniformly chosen between 0 and 10 m s^{-1}. We note that the movement pattern of an MN using the Random Waypoint Mobility Model is similar to the Random Walk Mobility Model if pause time is zero and [*minspeed, maxspeed*] = [*speedmin, speedmax*].

The Random Waypoint Mobility Model is also a widely used mobility model (e.g. References [13–16]). In addition, the model is sometimes simplified. For example, Ko and Vaidya [17] uses the Random Waypoint Mobility Model without pause times.

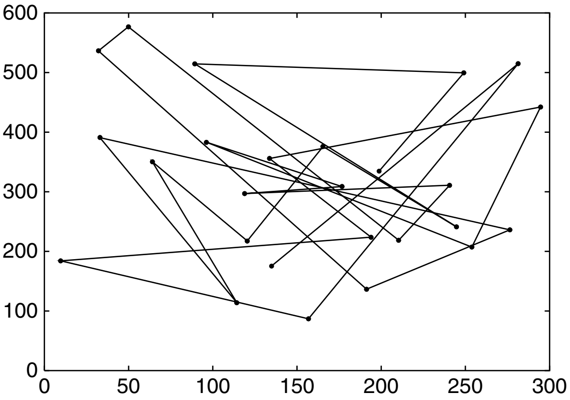

Fig. 3. Traveling pattern of an MN using the Random Waypoint Mobility Model.

2.2.2. Discussion

In most of the performance investigations that use the Random Waypoint Mobility Model, the MNs are initially distributed randomly around the simulation area. This initial random distribution of MNs is *not representative of the manner in which nodes distribute themselves when moving*. Figure 4 illustrates the cumulative average MN neighbor percentage for MNs using the Random Waypoint Mobility Model as time progresses (speed is 1 m s^{-1} and pause time is zero). The average MN neighbor percentage is the cumulative percentage of total MNs that are a given MNs' neighbor. For example, if there are 50 MNs in the network and a node has 10 neighbors, then the node's current neighbor percentage is 20%. A neighbor of an MN is a node within the MNs' transmission range. As shown, there is high variability

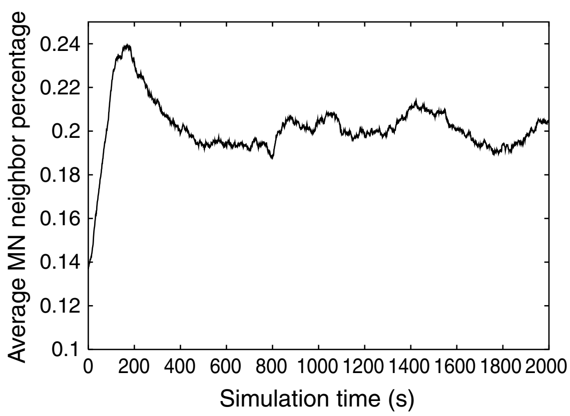

Fig. 4. Average neighbor percentage versus time.

during the first 600 s of simulation time. This high variability in average MN neighbor percentage will produce high variability in performance results unless the simulation results are calculated from long simulation runs [18].

In the following, we present three possible solutions to avoid this initialization problem. Firstly, save the locations of the MNs after a simulation has executed long enough to be past this initial high variability, and use this position file as the initial starting point of the MNs in all future simulations. Secondly, initially distribute the MNs in a manner that maps to a distribution more common to the model. For example, initially placing the MNs in a triangle distribution may distribute nodes in the Random Waypoint Mobility Model more accurately than initially placing the MNs randomly in the simulation area [19]. Lastly, discard the initial 1000 s of simulation time produced by the Random Waypoint Mobility Model in each simulation trial. (Discarding 1000 s of simulation time ensures that the initialization problem is removed even if the MNs move slowly. In other words, we can discard fewer seconds of simulation time for faster moving MNs.) Discarding the initial 1000 s of simulation time has an added benefit over the first solution proposed. Specifically, this simple solution ensures that each simulation has a random initial configuration.

There is also a complex relationship between node speed and pause time in the Random Waypoint Mobility Model. For example, a scenario with fast MNs and long pause times actually produces a more stable network than a scenario with slower MNs and shorter pause times. Figure 5 gives the link breakage rate of MNs using the Random Waypoint Mobility Model as a function of pause times and speeds. The figure illustrates that long pause times (i.e. over 20 s) produce a stable network (i.e. few link changes per MN) even at high speeds [18]. In other words, the figure indicates that the mobile network is quite stable for all pause times over 20 s. (See Reference [20] for an in-depth discussion.)

If the Random Waypoint Mobility Model is used in a performance evaluation, appropriate parameters need to be evaluated. For example, the Random Waypoint Mobility Model is used to evaluate a multicast protocol for ad hoc networks in Reference [21]. In this performance investigation, the speed of the mobile nodes was varied between 0 and 1 m s^{-1}, the pause time of the mobile nodes was varied between 60 and 300 s and each simulation executed for 300 s. With such slow speeds and large

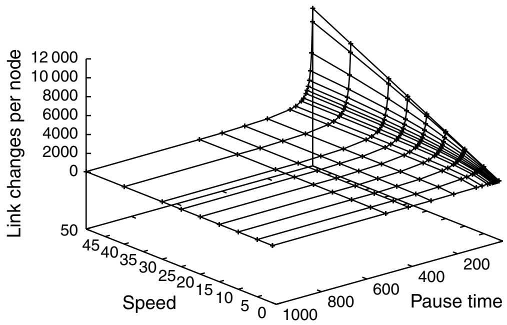

Fig. 5. Link breakage versus speed versus pause time.

pause times, the network topology hardly changes. In other words, the results presented in Reference [21] are only valid for an ad hoc network scenario with MNs that barely move.

2.3. Random Direction

The Random Direction Mobility Model [22] was created to overcome *density waves* in the average number of neighbors produced by the Random Waypoint Mobility Model. A density wave is the clustering of nodes in one part of the simulation area. In the case of the Random Waypoint Mobility Model, this clustering occurs near the center of the simulation area. In the Random Waypoint Mobility Model, the probability of an MN choosing a new destination that is located in the center of the simulation area or a destination that requires travel through the middle of the simulation area, is high. (This trend is illustrated in Figure 3.) Thus, the MNs appear to converge, disperse and converge again¶.

In order to alleviate this type of behavior and promote a semi-constant number of neighbors throughout the simulation, the Random Direction Mobility Model was developed [22]. In this model, MNs choose a random direction in which to travel similar to the Random Walk Mobility Model. An MN then travels to the border of the simulation area in that direction. Once the simulation boundary is reached, the MN pauses for a specified time, chooses another angular direction (between 0 and 180 degrees) and continues the process. Figure 6 shows an example path of an MN, which begins in the center of the simulation area

¶ An autocorrelation test on the number of neighbors obtained from MNs moving with the Random Waypoint Mobility Model reveals that there is no deterministic pattern to the mobility model; thus, we question the conclusion that density waves in the average number of neighbors actually exist [23].

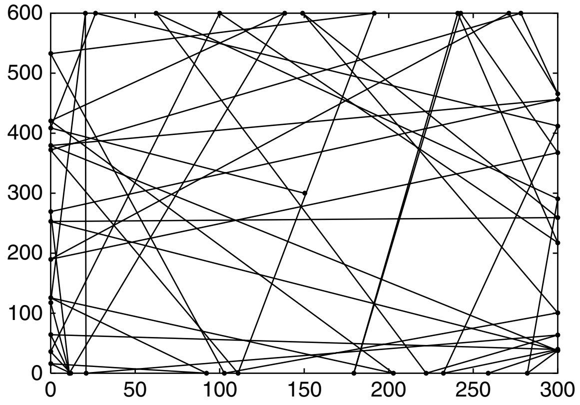

Fig. 6. Traveling pattern of an MN using the Random Direction Mobility Model.

or position (150, 300), using the Random Direction Mobility Model. The dots in the figure illustrate when the MN has reached a border, paused and then chosen a new direction. Since the MNs travel to and usually pause at the border of the simulation area, the average hop count for data packets using the Random Direction Mobility Model will be much higher than the average hop count of most other mobility models (e.g. Random Waypoint Mobility Model). In addition, network partitions will be more likely with the Random Direction Mobility Model compared to other mobility models.

A slight modification to the Random Direction Mobility Model is the Modified Random Direction Mobility Model [22]. In this modified version, MNs continue to choose random directions but they are no longer forced to travel to the simulation boundary before stopping to change direction. Instead, an MN chooses a random direction and selects a destination anywhere along that direction of travel. The MN then pauses at this destination before choosing a new random direction. This modification to the Random Direction Mobility Model produces movement patterns that could be simulated by the Random Walk Mobility Model with pause times.

2.4. A Boundless Simulation Area

In the Boundless Simulation Area Mobility Model, a relationship between the previous direction of travel and velocity of an MN with its current direction of travel and velocity exists [24]. A velocity vector $\overline{v} = (v, \theta)$ is used to describe an MNs' velocity v as well as its direction θ; the MNs' position is represented as (x, y). Both the velocity vector and the position are updated at every Δt time steps according to the following formulas:

$$v(t + \Delta t) = \min\{\max[v(t) + \Delta v, 0], V_{\max}\};$$

$$\theta(t + \Delta t) = \theta(t) + \Delta\theta;$$

$$x(t + \Delta t) = x(t) + v(t) * \cos\theta(t);$$

$$y(t + \Delta t) = y(t) + v(t) * \sin\theta(t);$$

where $V_{\max}$ is the maximum velocity defined in the simulation, Δv is the change in velocity that is uniformly distributed between $[-A_{\max} * \Delta t, A_{\max} * \Delta t]$, $A_{\max}$ is the maximum acceleration of a given MN, $\Delta\theta$ is the change in direction that is uniformly distributed between $[-\alpha * \Delta t, \alpha * \Delta t]$ and α is the maximum angular change in the direction an MN is traveling.

The Boundless Simulation Area Mobility Model is also different in how the boundary of a simulation area is handled. In all the mobility models previously mentioned, MNs reflect off or stop moving once they reach a simulation boundary. In the Boundless Simulation Area Mobility Model, MNs that reach one side of the simulation area continue traveling and reappear on the opposite side of the simulation area. This technique creates a torus-shaped simulation area allowing MNs to travel unobstructed. Figure 7 illustrates this concept. The rectangular area on the left side of Figure 7 is transformed into the torus shape on the right side of Figure 7 in two steps; first we fold the simulation area so that the top border ($y = Y\max$) lies against the bottom border ($y = 0$), forming a cylinder and then we fold the resulting cylinder so that both open circular ends connect. Figure 8 illustrates an example path of an MN using the Boundless Simulation Area Mobility Model, in which $V_{\max}$ is 10 m s^{-1}, $A_{\max}$ is 10 m (s^2)$^{-1}$, α is $\pi/2$ or 90 degrees and Δt is 0.1 s; the MN begins in the center of the simulation area or position (150, 300) and moves for 500 s. The triangles in the figure illustrate when the MN reaches a boundary and the dots illustrate where the MN reappears.

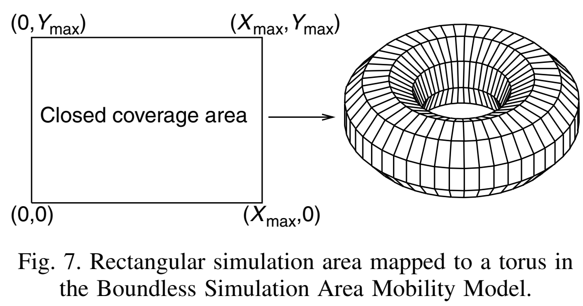

Fig. 7. Rectangular simulation area mapped to a torus in the Boundless Simulation Area Mobility Model.

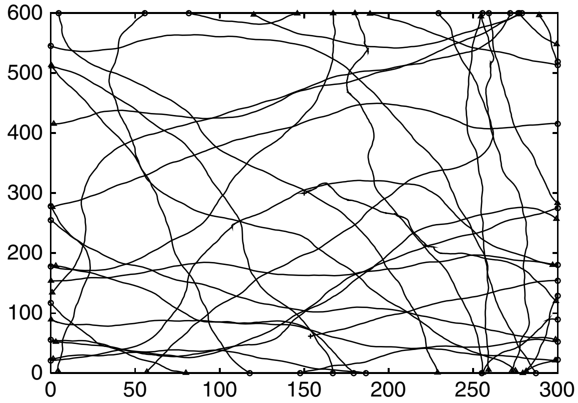

Fig. 8. Traveling pattern of an MN using the Boundless Simulation Area Mobility Model.

2.5. Gauss–Markov

The Gauss–Markov Mobility Model was originally proposed for the simulation of a PCS [10]; however, this model has been used for the simulation of an ad hoc network protocol [25]. In this section, we describe how the model was implemented in Reference [25].

The Gauss–Markov Mobility Model was designed to adapt to different levels of randomness via one tuning parameter. Initially each MN is assigned a current speed and direction. At fixed intervals of time, n, movement occurs by updating the speed and direction of each MN. Specifically, the value of speed and direction at the n^{th} instance is calculated on the basis of the value of speed and direction at the $(n-1)^{\text{st}}$ instance and a random variable using the following equations:

$$s_n = \alpha s_{n-1} + (1-\alpha)\overline{s} + \sqrt{(1-\alpha^2)}s_{x_{n-1}}$$

$$d_n = \alpha d_{n-1} + (1-\alpha)\overline{d} + \sqrt{(1-\alpha^2)}d_{x_{n-1}}$$

where s_n and d_n are the new speed and direction of the MN at time interval n; α, where $0 \le \alpha \le 1$, is the tuning parameter used to vary the randomness; $\overline{s}$ and $\overline{d}$ are constants representing the mean value of speed and direction as $n \to \infty$ and $s_{x_{n-1}}$ and $d_{x_{n-1}}$ are random variables from a Gaussian distribution. Totally random values (or Brownian motion) are obtained by setting $\alpha = 0$ and linear motion is obtained by setting $\alpha = 1$ [10]. Intermediate levels of randomness are obtained by varying the value of α between 0 and 1.

At each time interval, the next location is calculated on the basis of the current location, speed and direction of movement. Specifically, at time interval n, an MNs' position is given by the equations

$$x_n = x_{n-1} + s_{n-1}\cos d_{n-1}$$

$$y_n = y_{n-1} + s_{n-1}\sin d_{n-1}$$

where (x_n, y_n) and (x_{n-1}, y_{n-1}) are the x and y coordinates of the MNs' position at the n^{th} and $(n-1)^{\text{st}}$ time intervals, respectively, and s_{n-1} and d_{n-1} are the speed and direction of the MN, respectively, at the $(n-1)^{\text{st}}$ time interval.

To ensure that an MN does not remain near an edge of the grid for a long period of time, the MNs are forced away from an edge when they move within a certain distance of the edge. This is done by modifying the mean direction variable $\overline{d}$ in the above direction equation. For example, when an MN is near the right edge of the simulation grid, the value $\overline{d}$ is changed to 180 degrees. Thus, the MNs' new direction is away from the right edge of the simulation grid. The values of mean direction for different locations in the simulation grid are shown in Figure 9.

Figure 10 illustrates an example traveling pattern of an MN using the Gauss–Markov Mobility Model; the MN begins its movement in the center of the simulation area or position (150, 300) and moves for 1000 s. In Figure 10, n is 1 s, α is 0.75, $s_{x_{n-1}}$ and $d_{x_{n-1}}$ are chosen from a random Gaussian distribution with mean equal to zero and standard deviation equal to one. The value of $\overline{s}$ is fixed at 10 m s^{-1}; the value of $\overline{d}$ is initially 90 degrees but changes over time according to the edge proximity of the node.

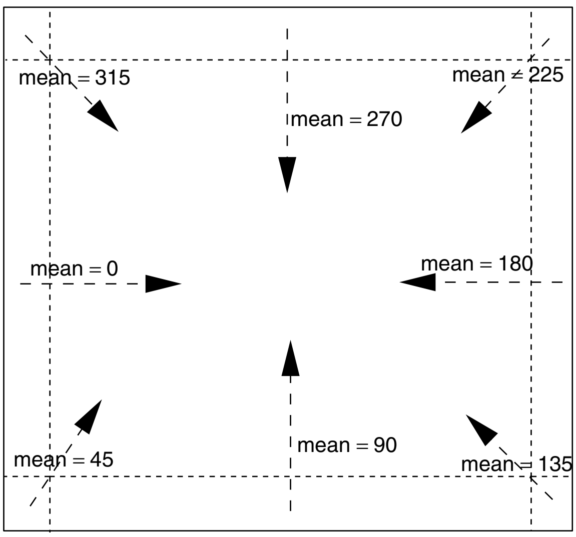

Fig. 9. Change of mean angle near the edges (in degrees).

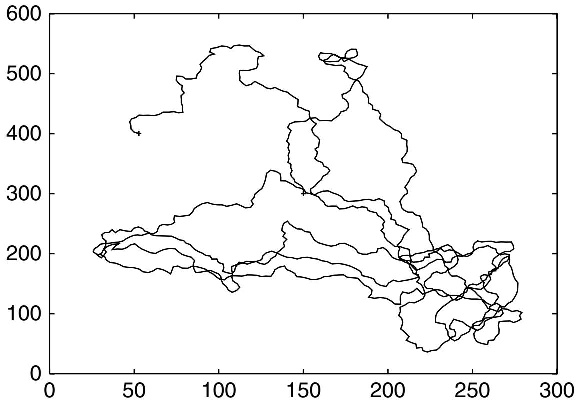

Fig. 10. Traveling pattern of an MN using the Gauss–Markov Mobility Model.

As shown in Figure 10, the Gauss–Markov Mobility Model can eliminate the sudden stops and sharp turns encountered in the Random Walk Mobility Model (see Section 2.1) by allowing past velocities (and directions) to influence future velocities (and directions).

The above description is how the Gauss–Markov Mobility Model was implemented in Reference [25]. Other implementations of the model exist. For example, the Markov process can be applied to the x and y equations directly instead of through speed and direction variables; in addition, a velocity vector can be used instead of a direction equation.

2.6. A Probabilistic Version of Random Walk

Chiang's mobility model utilizes a probability matrix to determine the position of a particular MN in the next time step, which is represented by three different states for position x and three different states for position y [26]. State 0 represents the current (x or y) position of a given MN, State 1 represents the MNs' previous (x or y) position and State 2 represents the MNs' next position if the MN continues to move in the same direction. The probability matrix used is

$$P = \begin{bmatrix} P(0,0) & P(0,1) & P(0,2) \\ P(1,0) & P(1,1) & P(1,2) \\ P(2,0) & P(2,1) & P(2,2) \end{bmatrix}$$

where each entry $P(a, b)$ represents the probability that an MN will go from State a to State b. The values within this matrix are used for updates to both the MNs' x and y positions. In Chiang's simulator, each node moves randomly with a preset average speed. The following matrix contains the values Chiang used to calculate x and y movements:

$$P1 = \begin{bmatrix} 0 & 0.5 & 0.5 \\ 0.3 & 0.7 & 0 \\ 0.3 & 0 & 0.7 \end{bmatrix}$$

These values are illustrated via a flow chart in Figure 11. With the values defined, an MN may take a step in any of the four possible directions (i.e. north, south, east or west) as long as it continues to move (i.e. no pause time). In addition, the probability of the MN continuing to follow the same direction is higher than the probability of the MN changing directions. Lastly, the values defined prohibit movements between the previous and next positions without passing through the current location. This implementation produces probabilistic rather than purely random movements, which may yield more realistic behaviors. For example, as people complete their daily tasks, they tend to continue moving in a semi-constant forward direction. Rarely do we suddenly turn around to retrace our steps, and we almost never take random steps hoping that we may eventually wind up somewhere relevant to our tasks. However, choosing appropriate values of $P(a, b)$ may prove difficult, if not impossible, for individual simulations unless traces are available for a given movement scenario.

Figure 12 illustrates an example traveling pattern of an MN using the Probabilistic Version of the Random Walk Mobility Model; the MN begins its movement in the center of the simulation area or position (150, 300) and moves according to the above probability matrix and state description. The step size is set to 10 for this example. As shown, the MN moves in straight lines for periods of time and does not show the highly variable direction seen in the Random Walk Mobility Model of Section 2.1.

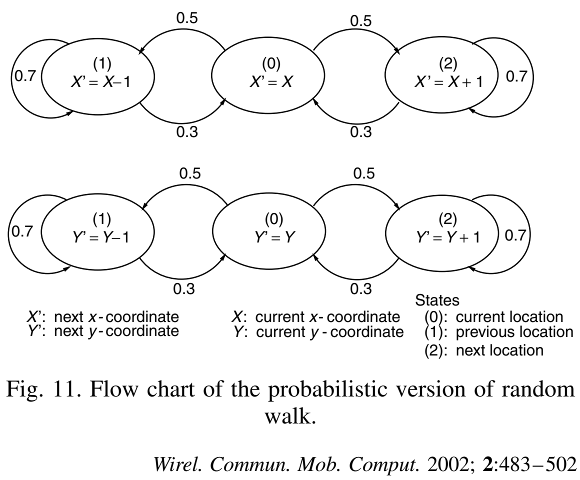

Fig. 11. Flow chart of the probabilistic version of random walk.

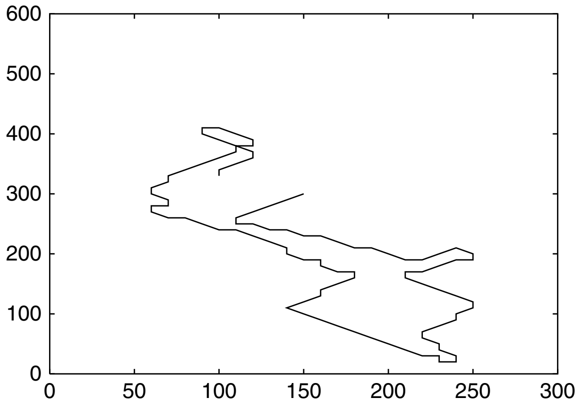

Fig. 12. Traveling pattern of an MN using the Probabilistic Version of the Random Walk Mobility Model.

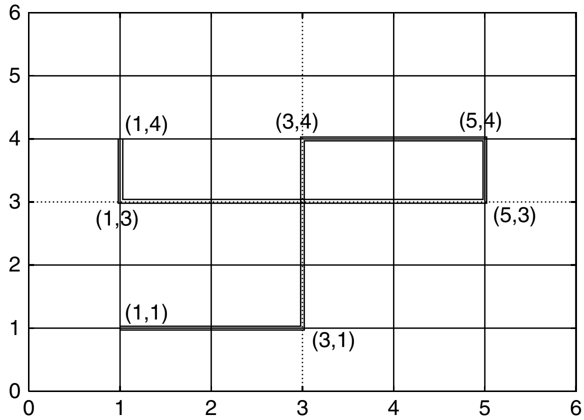

Fig. 13. Traveling pattern of an MN using the City Section Mobility Model.

2.7. City Section Mobility Model

In the City Section Mobility Model, the simulation area is a street network that represents a section of a city where the ad hoc network exists [3]. The streets and speed limits on the streets are based on the type of city being simulated. For example, the streets may form a grid in the downtown area of the city with a high-speed highway near the border of the simulation area to represent a loop around the city. Each MN begins the simulation at a defined point on some street. An MN then randomly chooses a destination, also represented by a point on some street. The movement algorithm from the current destination to the new destination locates a path corresponding to the shortest travel time between the two points; in addition, safe driving characteristics such as a speed limit and a minimum distance allowed between any two MNs exists. Upon reaching the destination, the MN pauses for a specified time and then randomly chooses another destination (i.e. a point on some street) and repeats the process.

Figure 13 shows the movements of an MN using an example city section in the City Section Mobility Model. Within this example, the center-most vertical and horizontal streets are designated as midspeed roads (i.e. $x = 3$ and $y = 3$), similar to main thoroughfares within a city; all other roads are considered to be slow residential roads. An MN starts the simulation at (1,1), moves to (5,4) and then moves to (1,4). The dashed lines in Figure 13 indicate the midspeed roads and the double lines represent streets traveled by the MN in our example. As shown, both moves from (1,1) to (5,4) and from (5,4) to (1,4) use midspeed roads.

The City Section Mobility Model provides realistic movements for a section of a city since it severely restricts the traveling behavior of MNs. In other words, all MNs must follow predefined paths and behavior guidelines (e.g. traffic laws). In the real world, MNs do not have the ability to roam freely without regard to obstacles and traffic regulations. In addition, people typically tend to travel in similar patterns when driving across town or walking across campus. Enforcing that all MNs follow predefined paths will increase the average hop count in the simulations compared to other mobility models [3].

Improvements to the City Section Mobility Model are the following: include pause times at certain intersections and destinations, incorporate acceleration and deceleration and account for higher/lower concentrations of MNs depending on the time of day. In addition, the model should be expanded to include a larger simulation area, an increased number of streets, a high-speed road along the border of the simulation area and other novel path-finding algorithms.

3. Group Mobility Models

In Section 2, we discuss mobility models that represent multiple MNs whose actions are completely independent of each other. In an ad hoc network, however, there are many situations in which it is necessary to model the behavior of MNs as they move together. For example, a group of soldiers in a military scenario may be assigned the task of searching a particular plot of land in order to destroy land mines, capture enemy attackers or simply work together in a cooperative manner to accomplish a common goal.

In order to model such situations, a group mobility model is needed to simulate this cooperative characteristic. In this section, we present five group mobility models. We note that four of the five group mobility models are closely related. The most general of these four models is the Reference Point Group Mobility (RPGM) model. Specifically, three of our group mobility models (Column, Nomadic and Pursue) can be implemented as special cases of the RPGM model.

3.1. Exponential Correlated Random Mobility Model

According to Reference [11], one of the first group mobility models to be proposed is the Exponential Correlated Random Mobility Model. In this model, a motion function is used to create MN movements. Given a position (MN or group) at time t, $\vec{b}(t)$ is used to define the next position (MN or group) at time $t+1$, $\vec{b}(t+1)$:

$$b(t+1) = b(t)e^{-\frac{1}{\tau}} + \left(\sigma\sqrt{1-\left(e^{-\frac{1}{\tau}}\right)^2}\right) r$$

where τ adjusts the rate of change from the MNs' previous location to its new location (i.e. small τ equates to large change) and τ is a random Gaussian variable with variance σ. Unfortunately, it is not easy to create a given motion pattern by selecting appropriate values for (τ, σ) in the Exponential Correlated Random Mobility Model [11]. The next four group mobility models improve upon this drawback.

3.2. Column Mobility Model

The Column Mobility Model proves useful for scanning or searching purposes [2]. This model represents a set of MNs that move around a given line (or column), which is moving in a forward direction (e.g. a row of soldiers marching together toward their enemy). A slight modification of the Column Mobility Model allows the individual MNs to follow one another (e.g. a group of young children walking in a single-file line to their classroom).

For the implementation of this model, an initial reference grid (forming a column of MNs) is defined [27]. Each MN is then placed in relation to its reference point in the reference grid; the MN is then allowed to move randomly around its reference point via an entity mobility model. (The authors propose using the Random Walk Mobility Model, which is described in Section 2.1, as the entity mobility model.) The new reference point for a given MN is defined as

$$\begin{aligned} new\_reference\_point = {} & old\_reference\_point \\ & + advance\_vector \end{aligned}$$

where *old_reference_point* is the MNs' previous reference point and *advance_vector* is a predefined offset that moves the reference grid. The predefined offset that moves the reference grid is calculated via a random distance and a random angle (between 0 and π since movement is in a forward direction only). Since the same predefined offset is used for all MNs, the reference grid is a 1-D line.

Figure 14 gives an illustration of four MNs moving in the Column Mobility Model. As shown, the MNs roam closely around their respective reference points. When the reference grid moves (on the basis of a random distance and a random angle), the MNs follow the grid and then continue to roam around their respective reference points. Figure 15 illustrates the simulated movement of two groups (three MNs are in each group) using the Column Mobility Model. One group in the figure is using the original Column Mobility Model, in which the MNs move perpendicular to the direction of movement. The second group is using the modified Column Mobility Model, in which the MNs move parallel to the direction of movement. We obtained these movement patterns for the Column Mobility Model using a variation of our RPGM model implementation (see Section 3.5).

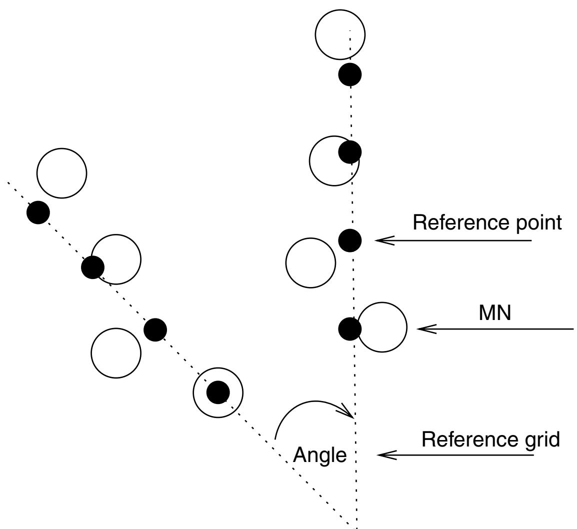

Fig. 14. Movements of four MNs using the Column Mobility Model.

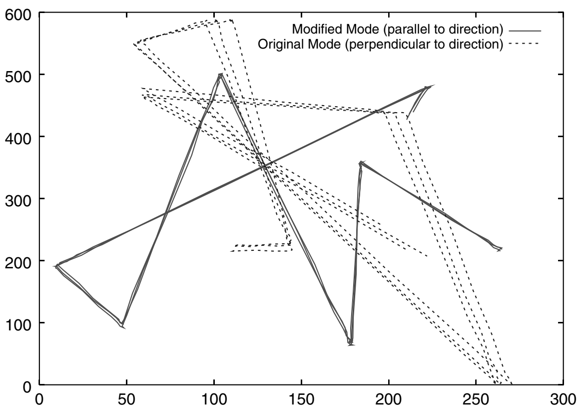

Fig. 15. Traveling pattern of MNs using the Column Mobility Model.

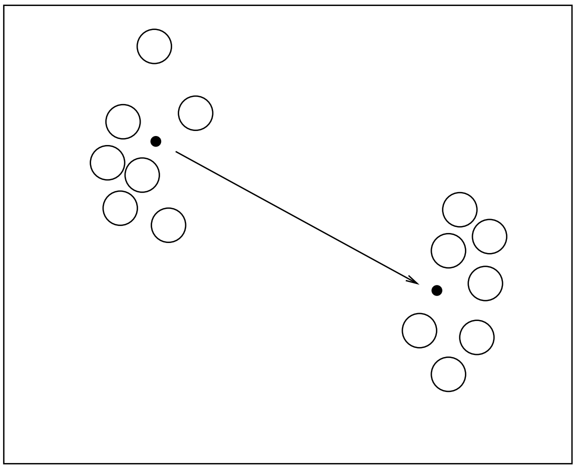

Fig. 16. Movements of seven MNs using the Nomadic Community Mobility Model.

3.3. Nomadic Community Mobility Model

Just as ancient nomadic societies moved from location to location, the Nomadic Community Mobility Model represents groups of MNs that collectively move from one point to another [2,27]. Within each community or group of MNs, individuals maintain their own personal 'spaces' where they move in random ways. Numerous applications exist for this type of scenario. For example, consider a class of students touring an art museum. The class would move from one location to another together; however, the students within the class would roam around a particular location individually.

In the Nomadic Community Mobility Model, each MN uses an entity mobility model (e.g. the Random Walk Mobility Model) to roam around a given reference point. When the reference point changes, all MNs in the group travel to the new area defined by the reference point and then begin roaming around the new reference point. The parameters for the entity mobility model define how far an MN may roam from the reference point.

Compared to the Column Mobility Model, the MNs in the Nomadic Community Mobility Model share a common reference point versus an individual reference point in a column. Thus, we would expect the MNs to be less constrained in their movement around the defined reference point. For example, in the Column Mobility Model, the MNs may only travel for two seconds before changing direction and speed; in the Nomadic Community Mobility Model, the MNs may be allowed to travel for 60 s before changing direction and speed. Figure 16 gives an illustration of seven MNs moving with the Nomadic Community Mobility Model. The reference point (represented by a small black dot) moves from one location to another; as shown, the MNs follow the movement of the reference point. While we do not illustrate a simulated movement pattern for the Nomadic Community Mobility Model, one could easily be created by using the implementation of the RPGM model (see Section 3.5).

3.4. Pursue Mobility Model

The Pursue Mobility Model is also defined in Reference [2,27]. As the name implies, the Pursue Mobility Model attempts to represent MNs tracking a particular target. For example, this model could represent police officers attempting to catch an escaped criminal. The Pursue Mobility Model consists of a single update equation for the new position of each MN:

$$new\_position = old\_position + acceleration$$
$$(target - old\_position) + random\_vector$$

where *acceleration(target-old_position)* is information on the movement of the MN being pursued and *random_vector* is a random offset for each MN. The *random_vector* value is obtained via an entity mobility model (e.g. the Random Walk Mobility Model); the amount of randomness for each MN is limited in order to maintain effective tracking of the MN being pursued. The current position of an MN, a random vector and an acceleration function are combined to calculate the next position of the MN. Figure 17 gives an illustration of six MNs moving with the Pursue Mobility Model. The white node represents the node being pursued and the solid black nodes represent the

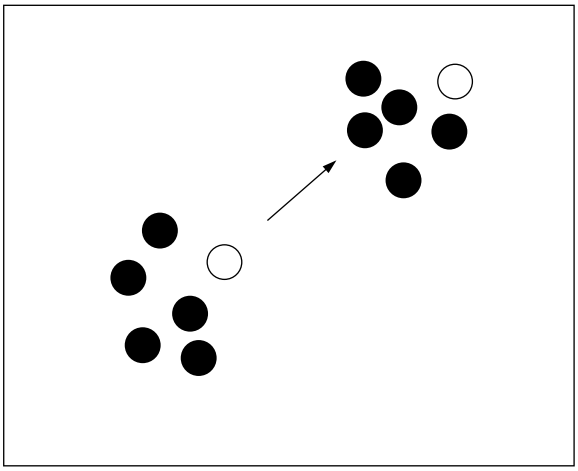

Fig. 17. Movements of six MNs using the Pursue Mobility Model.

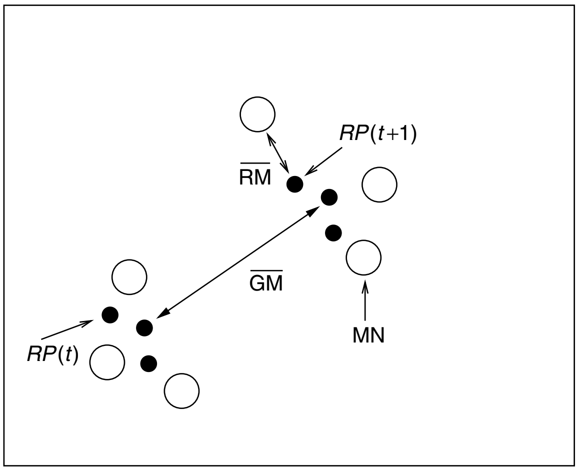

Fig. 18. Movements of three MNs' using the RPGM model.

pursuing nodes. Again, a simulated movement pattern for the Pursue Mobility Model could easily be generated using the implementation of the RPGM model (see Section 3.5).

3.5. Reference Point Group Mobility Model

The RPGM model represents the random motion of a group of MNs as well as the random motion of each individual MN within the group [11]. Group movements are based on the path traveled by a logical center for the group. The logical center for the group is used to calculate group motion via a group motion vector, $\vec{GM}$. The motion of the group center completely characterizes the movement of its corresponding group of MNs, including their direction and speed. Individual MNs randomly move about their own predefined reference points, whose movements depend on the group movement. As the individual reference points move from time t to $t+1$, their locations are updated according to the group's logical center. Once the updated reference points, $RP(t+1)$, are calculated, they are combined with a random motion vector, $\vec{RM}$, to represent the random motion of each MN about its individual reference point.

Figure 18 gives an illustration of three MNs moving with the RPGM model. The figure illustrates that, at time t, three black dots exist to represent the reference points, $RP(t)$, for the three MNs. As shown, the RPGM model uses a group motion vector $\vec{GM}$ to calculate each MNs' new reference point, $RP(t+1)$, at time $t+1$; as stated, $\vec{GM}$ may be randomly chosen or predefined. The new position for each MN is then calculated by summing a random motion vector, $\vec{RM}$, with the new reference point. The length of $\vec{RM}$ is uniformly distributed within a specified radius centered at $RP(t+1)$ and its direction is uniformly distributed between 0 and 2π.

Movement patterns using the RPGM model are shown in Figures 19 and 20. Figure 19 is an illustration of three MNs moving together as one group. Figure 20 is an illustration of five groups moving, such that each group has a different number of MNs. Both the movement of the logical center for each group, and the random motion of each individual MN within the group, are implemented via the Random Waypoint Mobility Model. One difference, however, is that individual MNs do not use pause times while the group is moving. Pause times are only used when the group reference point reaches a destination and all group nodes pause for the same period of time.

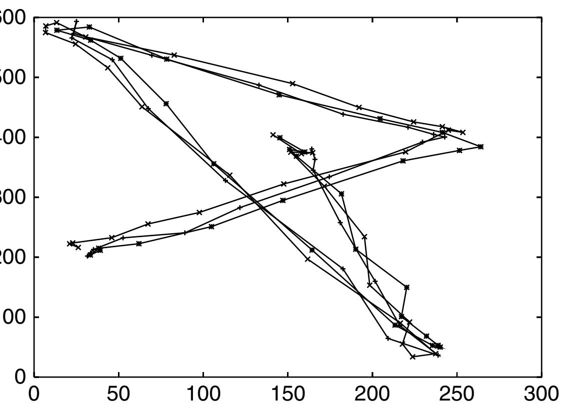

Fig. 19. Traveling pattern of one group (three MNs) using the RPGM model.

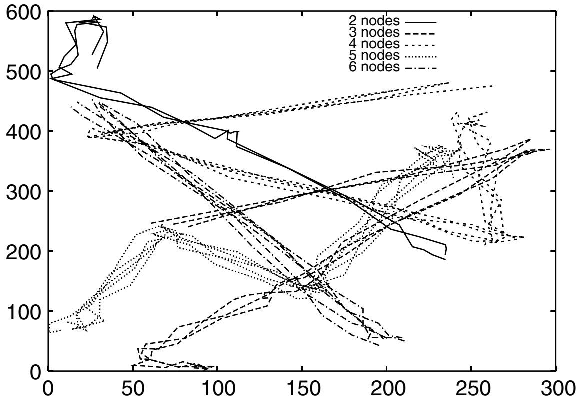

Fig. 20. Traveling pattern of five groups using the RPGM model.

The RPGM model was designed to depict scenarios such as an avalanche rescue. During an available rescue, the responding team consisting of human and canine members work cooperatively. The human guides tend to set a general path for the dogs to follow, since they usually know the approximate location of victims. The dogs each create their own 'random' paths around the general area chosen by their human counterparts.

The RPGM model was originally defined in Reference [11] and then used in Reference [28]. If appropriate group paths are chosen, along with proper initial locations for various groups, many different mobility applications may be represented with the RPGM model. In Reference [11], three applications for the RPGM model are defined. Firstly, the In-place Mobility Model partitions a given geographical area such that each subset of the original area is assigned to a specific group; the specified group then operates only within that geographic subset. Secondly, the Overlap Mobility Model simulates several different groups, each of which has a different purpose, working in the same geographic region; each group within this model may have different characteristics than other groups within the same geographical boundary. For example, in disaster recovery of a geographical area, one might encounter a rescue personnel team, a medical team and a psychologist team, each of which have unique traveling patterns, speeds and behaviors. Lastly, the Convention Mobility Model divides a given area into smaller subsets and allows the groups to move in a similar pattern throughout each subset. Similar to the Overlap Mobility Model, some groups in the Convention Mobility Model may travel faster than others.

As mentioned, Figure 15, which is an illustration of the Column Mobility Model, was created via our RPGM model implementation. To create this movement pattern, we added the following restriction to the RPGM model: all the nodes' reference points in a group must be in a column that is either perpendicular or parallel to the direction of travel. While simulated movement patterns are not illustrated in Sections 3.3 and 3.4, an implementation of the Nomadic Community Mobility Model and the Pursue Mobility Model are obtained from our RPGM model implementation. Specifically, we use a value of zero for the input parameter reference point separation in our RPGM model implementation to ensure that all the individual node reference points are the same as the group reference point.

4. Importance of Choosing a Mobility Model

In this section, we illustrate that the choice of a mobility model can have a significant effect on the performance investigation of an ad hoc network protocol. The results presented illustrate the importance of choosing an appropriate mobility model (or models) for the performance evaluation of a given ad hoc network protocol.

We use ns-2 [29] to compare the performance of the Random Walk Mobility Model, the Random Waypoint Mobility Model, the Random Direction Mobility Model and the RPGM model via a simulation with 50 MNs. (Table I details how the 50 MNs are separated into groups for the RPGM model.) Two sets of results are presented for the RPGM model; one set of results consists of intergroup communication only, and the other set of results consists of 50% intergroup communication and 50% intragroup communication (see details below). Each MN in the simulations has a 100 m transmission range, and the routing of packets is accomplished with the Dynamic Source Routing

Table I. Groups specified in the RPGM model.

Number of group members	Number of these groups	Total nodes	Per cent of total
2	7	14	28%
3	4	12	24%
4	2	8	16%
5	2	10	20%
6	1	6	12%
TOTAL	16	50	100%

Protocol (DSR) [12]. The parameters for these four mobility models were chosen in a way to simulate path movements that were as similar as possible. For example, in the Random Walk Mobility Model, the MN changes directions after moving a distance of 100 m, which produces movement patterns similar to the Random Waypoint Mobility Model when pause time is zero.

DSR is a source routing protocol that determines routes on demand. In a source routing protocol, each packet carries the full route (a sequenced list of nodes) that the packet should be able to traverse in its header. In an on-demand (or reactive) routing protocol such as DSR, a route to a destination is requested only when there is data to send to that destination, and a route to that destination is unknown or expired. We chose DSR since it performs well in many of the performance evaluations of unicast routing protocols (e.g. References [13,16]).

The ns-2 code used in our simulations of DSR was obtained from Reference [30]. The simulations are executed for 2010 s; however, our results are gathered from 1010 s of simulated time and data is only sent from 1000 to 2000 s of simulation time, which accounts for the conclusions drawn from Figure 4. Our communication model is similar to the communication model used in References [13] and [16]. For the entity mobility models and RPGM with all intergroup communication, we have 20 CBR (constant bit rate) sources sending packets at a rate of one packet per second to 20 different receivers. In other words, 20 000 packets are transmitted between 20 peers. For the RPGM results with both intergroup and intragroup communication, 10 000 packets are transmitted between 20 peers in different groups (1 packet every 2 s) and 10 000 packets are transmitted within groups (1 packet every 5 s). All packets are of size 64 bytes. We avoid unnecessary contention in the transmission of packets by offsetting the transmission of a data packet by 0.0001 s.

All the performance results presented are an average of 10 different simulation trials. The initial locations of the MNs in each trial are random (i.e. via the uniform distribution). We calculate a 95% confidence interval for the unknown mean and we plot these confidence intervals on the figures. Since most of the confidence intervals are quite small (in fact, some of the intervals are smaller than the symbol used to represent the mean on our plots), we are convinced that our simulation results precisely represent the unknown mean.

In our comparison of the four mobility models, we consider the following performance metrics obtained from the DSR protocol: data packet delivery ratio, end-to-end delay, average hop count and protocol overhead. The data packet delivery ratio is the ratio of the number of data packets delivered to the destination nodes divided by the number of data packets transmitted by the source nodes.

Figures 21 and 22 illustrate the performance (i.e. data packet delivery ratio and end-to-end delay) of DSR with the four mobility models chosen. Figure 23 illustrates the average hop count versus speed, which helps us understand these two

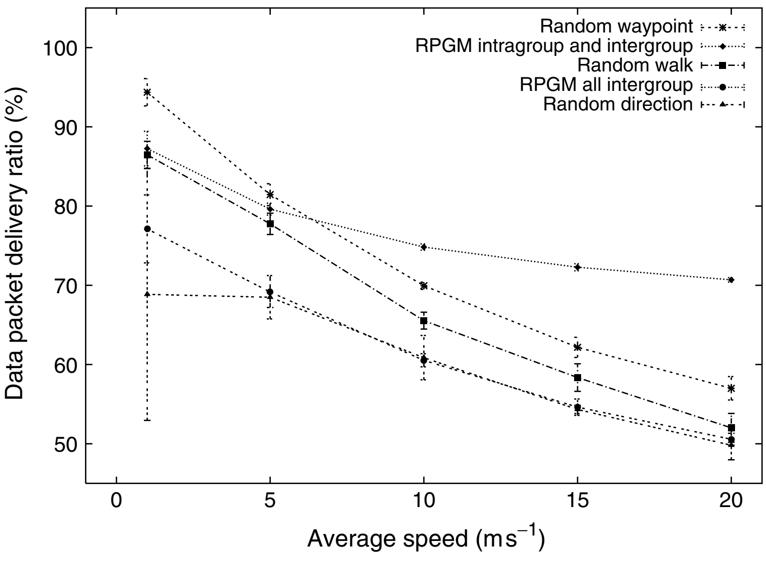

Fig. 21. Data packet delivery ratio versus speed.

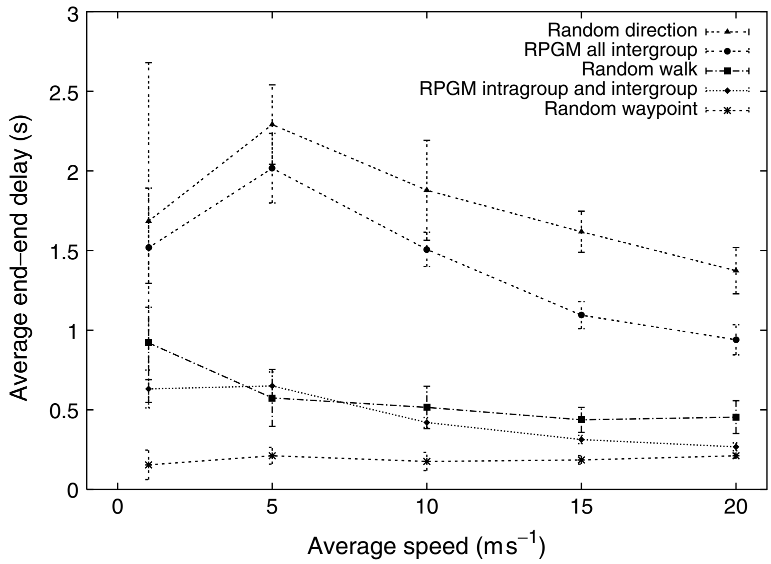

Fig. 22. Average end-to-end delay (s).

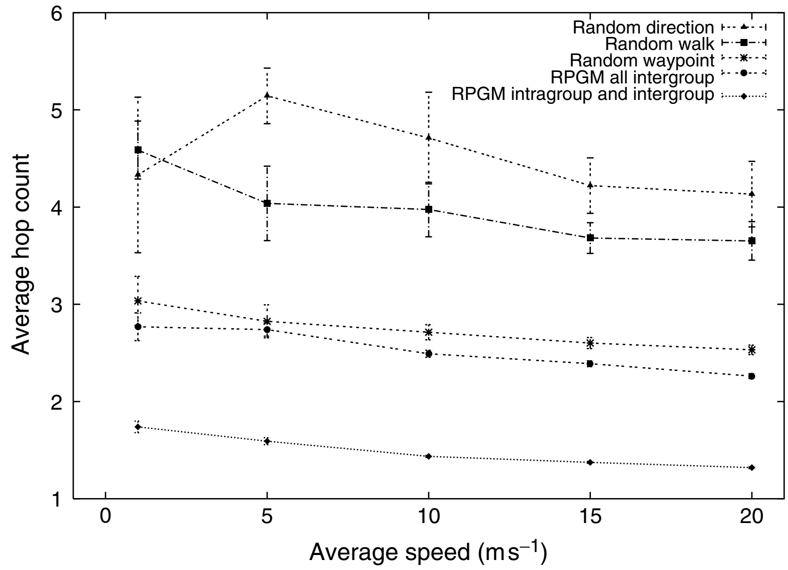

Fig. 23. Hop count versus speed.

performance figures. The three figures combine to illustrate that the Random Waypoint Mobility Model stresses DSR less than the other two entity mobility models. Specifically, the Random Waypoint Mobility Model has the highest data packet delivery ratio, the lowest end-to-end delay and the lowest average hop count compared to the Random Walk Mobility Model and Random Direction Mobility Model. These results exist since MNs using the Random Waypoint Mobility Model are often traveling through (or to) the center of the simulation area.

The Random Direction Mobility Model has the highest average hop count, the highest end-to-end delay and the lowest data packet delivery ratio since the Random Direction Mobility Model has each MN move to the border of the simulation area before changing direction. Thus, hop counts between a sender and receiver are higher and transient network partitions are more likely in the Random Direction Mobility Model compared to the other two entity mobility models. The performance of DSR when using the Random Walk Mobility Model falls

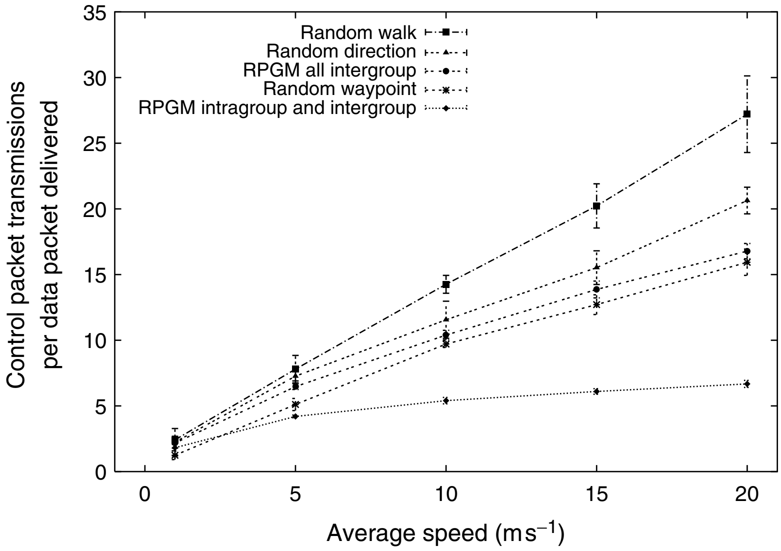

Fig. 24. Control packet overhead versus speed.

between these two extremes. Lastly, we note that the confidence intervals of the Random Walk Mobility Model and Random Direction Mobility Model are the largest; more variation in movement patterns exist in these two mobility models.

The RPGM model with only intergroup communication has approximately the same hop count as the Random Waypoint Mobility Model (see Figure 23). As mentioned, both a group's movement and an MNs' movement within a group in the RPGM model is done via the Random Waypoint Mobility Model. Thus, we would expect the hop counts for received packets to be similar between these two simulations. The RPGM model with only intergroup communication has a much lower data packet delivery ratio and higher end-to-end delay than the results for the Random Waypoint Mobility Model. Since only 16 groups exist (see Table I) in the RPGM model simulation, the network will be much sparser than the 50 MNs that roam in the Random Waypoint Mobility Model. And, since all communication is between groups, the performance of the mobility model in terms of data packet delivery ratio and end-to-end delay will suffer from transient partitions that exist in the sparse network.

The RPGM model with both intergroup and intragroup communication has the lowest average hop count (see Figure 23), since 50% of the packets transmitted are sent within the groups. Low average hop count corresponds to a high data packet delivery ratio, which is illustrated in Figure 21. The data packet delivery ratio is not, however, as high as one would expect; since 50% of the packets are transmitted between groups, these packets are sometimes dropped because of the transient partitions that occur. Figure 22 illustrates that the partitions also affect the end-to-end delay of the results for the RPGM model with both intergroup and intragroup communication.

Figures 24 and 25 illustrate the overhead DSR requires with each of the four chosen mobility models. Figure 24 shows the number of control packet transmissions for each data packet delivered as speed increases. Figure 25 illustrates the number of control byte transmissions (in both control packets and data packets) for each data packet delivered as speed increases. Since the RPGM model with both intergroup and intragroup communication has the lowest average hop count, this model requires the least amount of overhead. MNs moving with the Random Walk Mobility Model and the Random Direction Mobility Model have the highest average hop count and as a result these two models require the highest amount of overhead.

5. Conclusions

Conclusion 1: The performance of an ad hoc network protocol can vary significantly with different mobility models. Figures 21 to 25 illustrate the performance of one ad hoc network routing protocol with different mobility models. As shown, the performance of the protocol is greatly affected by the mobility model.

Conclusion 2: The performance of an ad hoc network protocol can vary significantly when the same

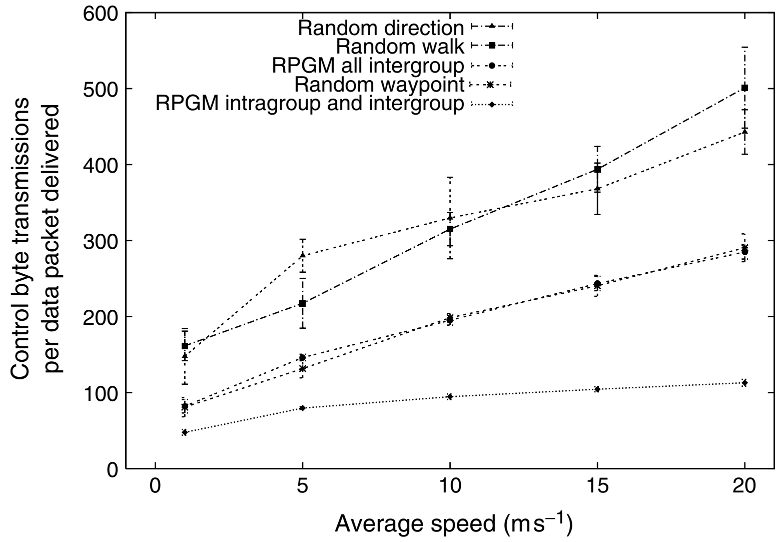

Fig. 25. Control byte overhead versus speed.

mobility model is used with different parameters. Figures 1 and 2 illustrate the widely different movement patterns that can occur with the Random Walk Mobility Model when different input parameters are used. When evaluated, these different movement patterns lead to widely different performance results.

Conclusion 3: The selection of a mobility model may require a data traffic pattern that significantly influences protocol performance. For instance, if a group mobility model is simulated, then protocol evaluation should be done with a portion of the traffic local to the group. Intragroup communication changes a protocol's performance dramatically, compared with the same mobility scenarios and all intergroup communication (see Figures 21–25).

Conclusion 4: The performance of an ad hoc network protocol should be evaluated with the mobility model that most closely matches the expected real-world scenario. In fact, the anticipated real-world scenario can aid the development of the ad hoc network protocol significantly. However, since the development of ad hoc networks is relatively new, we do not yet know what a realistic model is for a given scenario. In fact, we are just beginning to see realistic trace files for PCS or cellular networks. In Reference [31], results are presented on how often MNs move and how far they move for the Metricom radio network. Traffic patterns and on-line behavior for wireless users of a high-speed wireless access network are presented in Reference [32].

Conclusion 5: If the expected real-world scenario is unknown, then researchers should make an informed choice about the mobility model to use[‖]. The following list summarizes our conclusions for the seven synthetic entity mobility models for ad hoc networks.

1. The Random Walk Mobility Model with a small input parameter (distance or time) produces Brownian motion and, therefore, basically evaluates a static network (see Figure 2) when used in a performance investigation. A large input parameter (distance or time) is similar to the Random Waypoint Mobility Model without pause times (see Figure 1 and Figure 3). The main difference between these two mobility models is that MNs are more likely to cluster in the center of the simulation area with the Random Waypoint Mobility Model.
2. The Random Waypoint Mobility Model is used in many prominent simulation studies of ad hoc network protocols. It is flexible, and it appears to create realistic mobility patterns for the way people might move in, for example, a conference setting or museum (see Figure 3). One concern with this model is the straight movement pattern created by the MN to the next chosen destination.
3. The Random Direction Mobility Model (see Figure 6) is an unrealistic model because it is

[‖] Since a single mobility model is unlikely to depict the behavior of the MNs in all scenarios, it may be best to evaluate an ad hoc network protocol with multiple mobility models.

unlikely that people would spread themselves evenly throughout an area (a building or a city). In addition, it is unlikely that people will only pause at the edge of a given area. The Modified Random Direction Mobility Model allows MNs to pause and change directions before reaching the simulation boundary; this version, however, is similar to the Random Walk Mobility Model with pause times.

4. The Boundless Simulation Area Mobility Model provides movement patterns that one might expect in the real world (see Figure 8). In addition, this model is the only one that allows MNs to travel unobstructed in the simulation area, thus removing any simulation edge effects from the performance evaluation. One concern, however, is the undesired side effects that would occur from allowing the MNs to move around a torus. For example, one static MN and one MN that continues to move in the same direction become neighbors again and again. In addition, a simulation area without edges would force modification of the radio propagation model to wrap transmissions from one edge of the area to the other.
5. The Gauss–Markov Mobility Model also provides movement patterns that one might expect in the real world (see Figure 10), if appropriate parameters are chosen. In addition, the method used to force MNs away from the edges of the simulation area (thus avoiding undesired edge effects) is of note.
6. While the Probabilistic Random Walk Mobility Model also provides movement patterns that one might expect in the real world (see Figure 12), choosing appropriate parameters for the probability matrix may be difficult. This model could become useful, however, when we have scenario trace data that we want to model.
7. The City Section Mobility Model (see Figure 13) appears to create realistic movements for a section of a city, since it severely restricts the traveling behavior of MNs; MNs do not have the ability to roam freely without regard to obstacles and other traffic regulations. Further development of this model (e.g. to use realistic city maps) is desired.

Regarding the five synthetic group mobility models for ad hoc networks, the following list summarizes our conclusions.

1. The Exponential Correlated Random Mobility Model appears to theoretically describe all other mobility models. However, selecting appropriate parameter values is (almost) impossible.
2. The Column, Nomadic Community and Pursue Mobility Models are useful group mobility models for specific realistic scenarios. The movement patterns provided by these three mobility models can be obtained by changing the parameters associated with the Reference Point Group Mobility Model.
3. The RPGM Model is a generic method for handling group mobility. An entity mobility model (or models) needs to be specified to handle both the movement of a group of MNs and the movement of the individual MNs within the group. The input parameters of the RPGM model allow the flexibility to implement the Column, Nomadic Community and Pursue Mobility Models.

In summary, if a group mobility model is desired, we recommend using the RPGM Model with appropriate parameters. If an entity mobility model is desired, we recommend using either the Random Waypoint Mobility Model, the Random Walk Mobility Model (if clustering in the middle of the simulation area is undesired) or the Gauss–Markov Mobility Model. However, a preferred entity mobility model combines the strengths of the current entity mobility models (see Conclusion 7). As mentioned, implementations of all the mobility models described in this paper (except Exponential Correlated Random Mobility Model) are available at *http://toilers.mines.edu*. Our implementations allow a user to create either a gnuplot figure or an ns-2 mobility file of a given mobility model.

Conclusion 6: The results of DSR presented in Figures 21 to 25 differ greatly from the results presented in References [13] and [16]. As an example, all the data packet delivery ratios presented in Reference [13] for DSR (using the Random Waypoint Mobility Model) are over 95%. Their results are not comparable to ours because of the differences in our simulation environments. (For example, the maximum average speed considered in Reference [13] is only 10 m s^{-1}; our maximum average speed is 20 m s^{-1}) Furthermore, the metric used for the *x*-axes in Reference [13] is pause time, rather than speed. As discussed in Section 2.2.2 (see Figure 5), speed has a much greater impact than pause time on link breakage rates [18]. The results presented in Reference [16] are taken from only 250 s of simulation time. As shown in Figure 4, there is high variability in the average number of neighbors during the initial seconds of simulation time for MNs using the Random Waypoint Mobility Model. Since the authors of Reference [16] do not present confidence intervals for the unknown mean in the random scenarios, the precision of their estimates cannot be determined. (See Reference [33] for more details on the differences between the simulation environments in References [13,16] and herein.)

Conclusion 7: Further research on mobility models for ad hoc network protocol evaluation is needed. One avenue of future work is to devote further effort in examining the movements of entities in the real world to produce accurate mobility models. A second avenue is to develop a new model that combines the best attributes of some of the models. For example, a new model could handle edges via the method in the Gauss–Markov Mobility Model and then combine the movement patterns of the Boundless Simulation Area Mobility Model and the Random Waypoint (or Random Walk) Mobility Model. A third avenue is to develop a minimum mobility model standard for performance evaluation. This minimum standard would allow us to evaluate different mobility models more thoroughly. Lastly, we should examine the method used to choose a future MN location. In other words, the similarities and differences between mobility models that randomly select directions and mobility models that randomly select specific locations should be analyzed.

Acknowledgments

We thank William Navidi, Felipe Perrone, and the anonymous referees for providing helpful suggestions that improved the quality of this paper.

References

1. Sanchez M, Manzoni P. A java-based ad hoc networks simulator. In *Proceedings of the SCS Western Multiconference Web-based Simulation Track*, January 1999.
2. Sanchez M, Manzoni P. Anejos: A java based simulator for ad-hoc networks. *Future Generation Computer Systems* 2001; **17**(5): 573–583.
3. Davies V. *Evaluating Mobility Models Within an Ad Hoc Network*. Master's thesis, Colorado School of Mines, Colorado, 2000.
4. Weisstein EW. *The CRC Concise Encyclopedia of Mathematics*. CRC Press: Boca Raton, FL, 1998.
5. Bar-Noy A, Kessler I, Sidi M. Mobile users: to update or not to update?. In *Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 1994; pp. 570–576.
6. Garcia-Luna-Aceves JJ, Madrga EL. A multicast routing protocol for ad-hoc networks. In *Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 1999; pp. 784–792.
7. Rubin I, Choi C. Impact of the location area structure on the performance of signaling channels in wireless cellular networks. *IEEE Communications Magazine* 1997; **35**(2): 108–115.
8. Zonoozi M, Dassanayake P. User mobility modeling and characterization of mobility pattern. *IEEE Journal on Selected Areas in Communications* 1997; **15**(7): 1239–1252.
9. Basagni S, Chlamtac I, Syrotiuk VR, Woodward BA. A distance routing effect algorithm for mobility (DREAM). In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, 1998; pp. 76–84.
10. Liang B, Haas Z. Predictive distance-based mobility management for PCS networks. In *Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, March 1999.
11. Hong X, Gerla M, Pei G, Chiang C. A group mobility model for ad hoc wireless networks. In *Proceedings of the ACM International Workshop on Modeling and Simulation of Wireless and Mobile Systems (MSWiM)*, August 1999.
12. Johnson D, Maltz D. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*. Imelinsky T, Korth H (eds). Kluwer Academic Publishers: Norwell, MA, 1996; pp. 153–181.
13. Broch J, Maltz D, Johnson D, Hu Y, Jetcheva J. Multi-hop wireless ad hoc network routing protocols. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, 1998; pp. 85–97.
14. Chiang C, Gerla M. On-demand multicast in mobile wireless networks. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP)*, 1998.
15. Garcia-Luna-Aceves JJ, Spohn M. Source-tree routing in wireless networks. In *Proceedings of the 7th International Conference on Network Protocols (ICNP)*, 1999.
16. Johansson P, Larsson T, Hedman N, Mielczarek B, Degermark M. Routing protocols for mobile ad-hoc networks—a comparative performance analysis. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, 1999; pp. 195–206.
17. Ko Y, Vaidya NH. Location-aided routing (LAR) in mobile ad hoc networks. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, 1998; pp. 66–75.
18. Boleng J. Normalizing mobility characteristics and enabling adaptive protocols for ad hoc networks. In *Proceedings of the Local and Metropolitan Area Networks Workshop (LANMAN)*, March 2001; pp. 9–12.
19. Brown TX. Personal correspondence. April 2001.
20. Karp B. *Geographic Routing for Wireless Networks*. Ph.D. thesis, Harvard University, Cambridge, MA, 2000.
21. Royer EM, Perkins C. Multicast operation of the ad-hoc on-demand distance vector routing protocol. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, 1999; pp. 207–218.
22. Royer E, Melliar-Smith PM, Moser L. An analysis of the optimum node density for ad hoc mobile networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, 2001.
23. Navidi W. Personal correspondence. October 2000.
24. Haas Z. A new routing protocol for reconfigurable wireless networks. In *Proceedings of the IEEE International Conference on Universal Personal Communications (ICUPC)*, October 1997; pp. 562–565.
25. Tolety V. *Load Reduction in Ad Hoc Networks Using Mobile Servers*. Master's thesis, Colorado School of Mines, Colorado, 1999.

26. Chiang C. *Wireless Network Multicasting*. Ph.D. thesis, University of California, Los Angeles, 1998.
27. Sanchez M. Mobility models. http://www.disca.upv.es/misan/mobmodel.htm; Page accessed on May 30th, 2002.
28. Pei G, Gerla M, Hong X, Chiang C. A wireless hierarchical routing protocol with group mobility. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, September 1999.
29. The VINT Project. The network simulator -ns-2. http://www.isi.edu/nsnam/ns/; Page accessed 30 May 2002.
30. The Rice University Monarch Project. Monarch extensions to the ns simulator. URL: http://www.monarch.cs.rice.edu/; Page accessed 30 May 2002.
31. Tang D, Baker M. Analysis of a metropolitan-area wireless network. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 13–23, 1999.
32. Hutchins R, Zegura EW. Measurements from a campus wireless network. In *Proceedings of the IEEE International Conference on Communications (ICC)*, 2002.
33. Camp T, Boleng J, Williams B, Wilcox L, Navidi W. Performance comparison of two location based routing protocols for ad hoc networks. In *Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2002; pp. 1678–1687.

Authors' Biographies

Tracy Camp received her Ph.D. degree in Computer Science from The College of William and Mary in 1993. Currently she is an associate professor of computer science at the Colorado School of Mines. Her main field of research is in the area of mobile computing and networking; she is presently the principal investigator of three grants and a co-investigator of one grant totaling over one million dollars in federal funding. Dr Camp and her family (husband, toddler, dog and cat) are vegetarians that tremendously enjoy living in the foothills of the Rocky Mountains.

Jeff Boleng is an Air Force Communications and Computer Systems Officer currently serving in Seoul, South Korea. He received his Ph.D. and M.S. degrees in Mathematical and Computer Sciences from Colorado School of Mines in 2002 and 1997, respectively, and his B.S. in computer science from the U.S. Air Force Academy in 1991. He will return to the Air Force Academy faculty in 2003. Dr Boleng has a wide variety of research interests and is currently focusing on mobile communications and networking.

Vanessa Davies received a B.S. in 1999 and an M.S. in 2001 at the Colorado School of Mines in computer science. After working at Lockheed Martin Mission Systems for a year in Santa Maria, California, she is currently participating in licensing procedures to provide financial analysis for First Command Financial Planning in San Antonio, Texas. Her hobbies include reading, dog training, piano and staying in shape with pilates.