A Taxonomy and Survey of Microscopic Mobility Models from the Mobile Networking Domain

JOANNE TREURNIET, Defence Research and Development Canada, Ottawa Research Centre

A mobility model is used to generate the trajectories of mobile nodes in simulations when developing new algorithms for mobile networks. A model must realistically reflect the scenario in which the technology will be used to reliably validate the algorithm. Considerable progress has been made toward realistic mobility models in the academic literature, and models have become quite complex. A consistent taxonomy has not yet been established for this field. A new multifaceted taxonomy is presented in this work that provides a framework for authors to clearly and consistently describe their models, making them easier to understand and reproduce. By surveying the application field of mobile communication networks, a common nomenclature and a high-level view of existing literature are provided, which are required to reduce duplication of effort and to enable a better sense of the way forward. A tactical scenario demonstrates the application of the taxonomy to model construction.

14

Categories and Subject Descriptors: A.1 [**Introductory and Survey**]; I.6.5 [**Simulation and Modeling**]: Model Development—*Modeling methodologies*

General Terms: Algorithms, Experimentation, Performance, Reliability, Security, Verification

Additional Key Words and Phrases: Survey, taxonomy, mobile network, mobility model

ACM Reference Format:
Joanne Treurniet. 2014. A taxonomy and survey of microscopic mobility models from the mobile networking domain. ACM Comput. Surv. 47, 1, Article 14 (June 2014), 32 pages.
DOI: http://dx.doi.org/10.1145/2616973

1. INTRODUCTION

Mobile networks are formed dynamically, with the location of a node dependent on the movement of a device with its owner. Collecting these node movements, or traces, provides the ability to recreate the network dynamics for the purpose of validating new protocols, routing algorithms, and identity/trust management schemes in the laboratory. However, such trace data is resource intensive to collect, has a fixed number of nodes, and represents a single dataset. The number of nodes in the network can greatly affect connectivity and hence the performance of new technologies, and robust statistical sampling requires the analysis of multiple datasets. For these reasons, simulations that enable flexible trace generation are often employed as a first step. A mobility

Author's address: 3701 Carling Avenue, Ottawa, ON K1A 0Z4; email: Joanne.Treurniet@drdc-rddc.gc.ca.

DOI: http://dx.doi.org/10.1145/2616973

model provides the basis for generating a mobile node's trajectory in a simulation. The choice of model is important because any assumptions made in the model will alter the node trajectories, which can have a dramatic effect on the apparent performance of the technology being studied [Hong et al. 1999; Camp et al. 2002; Bai et al. 2003; Basgeet et al. 2003; Yoon et al. 2003; Blakely and Lowekamp 2004; Lagar-Cavilla et al. 2007; Chen et al. 2010]. The choice of scenario has also been shown to be important [Helgason et al. 2010]. Different application areas require scenarios with different types and degrees of randomness. For example, the tactical military domain has a low degree of randomness because missions are carried out with a purpose and in a particular environment, whereas simulating everyday pedestrian movement has a higher degree of randomness. Together, these results indicate that choosing the right model for a given scenario is paramount to validating the research. This recognition has led to increasingly complex models.

Research and development in mobile networks is expanding, and effectively communicating the models is important for the advancement of the field. However, communication issues in the mobile networking literature have been found to lead to difficulties with the reproducibility of simulation results, as demonstrated in Kurkowski et al. [2005] for relatively simple models. This applies equally to the models on which these simulations are based. The growing complexity of mobility models makes clear, concise, and complete model descriptions ever more difficult, invariably leading to omissions.

Standardizing the way we talk about mobility models will facilitate communication and access to the literature by nonexperts. A mobility model taxonomy composed of six elements is proposed in this work, including Spatial Constraints, Target Selection, Pathfinding, Motion, Pause Time, and Group Dynamics. By considering each element separately, the taxonomy provides a framework through which researchers can completely and systematically describe their work, improving communication and reproducibility. It also provides a common nomenclature and a high-level view of what already exists, which is required to reduce duplication of effort and to enable a better sense of the way forward.

Because mobility models are cross-disciplinary, it would be impractical to carry out an exhaustive survey of the state of the art. To manage the scope, this survey is limited to microscopic models from the mobile communications literature, excluding models developed explicitly for vehicular networks.

2. CLASSIFICATION APPROACH

A variety of approaches to classifying models in the literature have been used. Bettstetter [2001] classifies the models from the point of view of their degree of randomness. Several classify the literature in terms of *entity* versus *group* mobility (e.g., Camp et al. [2002], Babaei et al. [2007], and Nabi et al. [2011]). Other classifications distinguish whether or not a model accounts for obstacles, such as *geometric* versus *nongeometric* [Williams and Huang 2009], *unguided* versus *guided* [Aravind and Tahir 2010], and *free space* versus *geographic* [Ahmed et al. 2010]. Musolesi and Mascolo [2009] use a classification scheme consisting of *synthetic*, *trace-based,* and *social network* models, which is not comprehensive. Stepanov et al. [2005] divide the models into *random*, *area constrained*, *profile based*, *trace based*, and *integrated*. Since the field is reaching a point where most models fall into the *integrated* category, this classification is no longer viable. Aschenbruck et al. [2011] classify models in terms of dependencies and restrictions on node movement: *temporal* (influenced by past movement), *spatial* (influenced by surrounding nodes), *geographic* (confined to an area), *no dependencies*, or *hybrid*. Like the previous taxonomy, many models fall under *hybrid* in this taxonomy.

The preceding classification schemes fail to provide a crisp separation because they attempt to classify the models in their entirety, when most recent models are actually

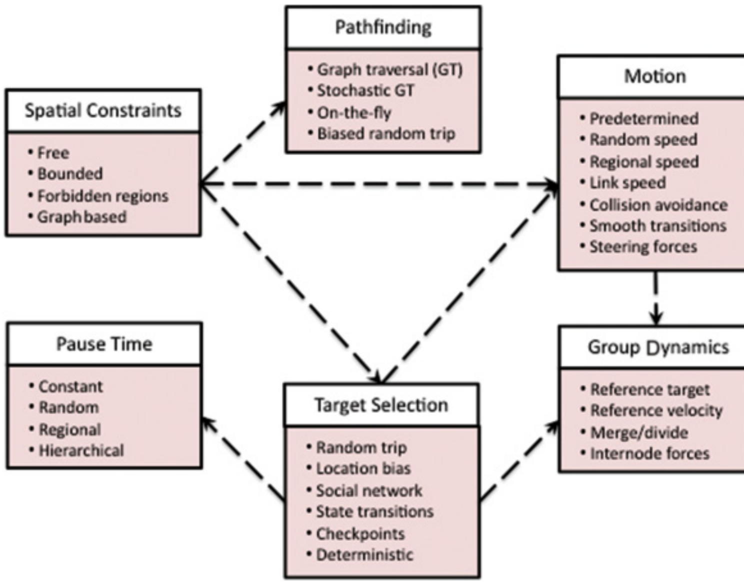

Fig. 1. The taxonomy of mobility model elements, shown with examples of methods that can be applied. Methods may be combined, and more methods can be added as work in the field progresses. Model elements may influence one another, as indicated by the dashed lines.

composed of multiple model elements. Figure 1 shows the proposed taxonomy for the elements of a mobility model. The elements were selected by considering what has been recognized in several framework descriptions as separate stages in a simulation (e.g., Choi et al. [2007], Basu et al. [2008], Holliday [2008], Papageorgiou et al. [2009], and Medina et al. [2010]), then assessing the commonalities of existing models along with what makes each model unique. Sample categories of methods observed in the literature are given in the shaded boxes of Figure 1, and the dashed links indicate the potential for a method within a model element to affect the choices available in other elements.

Describing a model using model elements requires a high-level description of the model that includes any relationships between the model elements, followed by a detailed description of each of the individual elements. All models must define the space along with any constraints (Spatial Constraints, Section 2.1), the process by which nodes can select their next desired location (Target Selection, Section 2.2), and the dynamics of how they get there, which can be more than simple linear motion (Motion, Section 2.4). The path they will follow to get to the next target (Pathfinding, Section 2.3) is only required when a node must pass through a set of intermediate points en route to its target. Optionally, nodes can pause between targets (Pause Time, Section 2.5), and they can behave as groups (Group Dynamics, Section 2.6). Some mobility models are hierarchical, with a secondary model embedded within a primary model; the elements must be described for each model in the hierarchy.

In the sections that follow, the model elements are discussed separately, giving examples of methods that have been applied in published work for each; these were selected to give a flavour of the state of the art. Methods may be combined, and more methods can be added as work in the field progresses. Conceptual frameworks for multiple-model simulations and software frameworks are discussed in Section 3. In Section 4, examples are given for how to both describe and construct a model using the taxonomy, and the taxonomy is applied to a selection of models to show similarities and trends in terms of these elements.

2.1. Spatial Constraints

Every model must address how the simulation area or volume is defined, which in turn sets limitations on node positions. Node positions can be unlimited, real but constrained to a bounded area, constrained to be outside of obstacles, or constrained to graph vertices and edges. Restricting nodes to a grid is a special case of a graph where the vertices are evenly spaced.

It has been shown that the scenario and its structure have a far greater effect on contact time, contact rate, and intercontact time measurements than accurate simulation parameters [Helgason et al. 2010], indicating that incorporating spatial restrictions in a mobility model will have a substantial effect on results. As shown in Figure 1, Spatial Constraints can directly affect the choice of methods available for Target Selection and Pathfinding elements.

2.1.1. Unlimited Free Space. In the simplest approach, nodes are allowed to move without restriction [Rubin and Choi 1997; Sánchez and Manzoni 2001].

2.1.2. Bounded Free Space. By adding boundaries to the unlimited free space approach, the nodes are constrained to a finite space or subspace. Periodic boundary conditions, where the simulation is executed on a torus, make a node reappear on the opposite side of the simulation plane (i.e., “wrap around”) with the same direction and velocity [Haas 1997]. Nodes can enter and exit the boundaries of the simulation via a random arrival process such as a Poisson, Erlang process [Rubin and Choi 1997; Borrel et al. 2005; Helgason et al. 2010]. Boundaries can be implicitly enforced via the Target Selection method by restricting the target to be within or on the boundaries [Johnson and Maltz 1996; Royer et al. 2001] or by defining the behaviour of a node when it encounters a boundary, such as choosing a new random direction pointed into the simulation space [Tolety 1999].

2.1.3. Forbidden Regions. Regions of the simulation space that nodes are forbidden to occupy are useful for modelling buildings and other obstacles (e.g., Aschenbruck et al. [2007], Papageorgiou et al. [2009], and Wu et al. [2011]).

2.1.4. Graph Constrained. Nodes can be constrained to graph edges and vertices to restrict their motion and targets.[1] The simplest graph is a polar grid [Bei-Zhan et al. 2007] or Cartesian grid [Chiang 1998; Basagni et al. 1999; Davies 2000; Bai et al. 2003; Washington and Iziduh 2009; Bhandari et al. 2010]. The graph can be augmented with a higher density of vertices in preferred areas [Gloor et al. 2004; Medina et al. 2010]. A synthetic graph can be generated by randomly selecting the length and end-point positions of horizontal and vertical grid edges [Zheng et al. 2010], or selecting vertex positions randomly [Kang et al. 2011] or using an algorithm that generates scale-free vertex distributions [Lee et al. 2012].

Graphs can be constructed from geographical maps [Scourias and Kunz 1999; Tian et al. 2002; Stepanov et al. 2005; Schwamborn et al. 2010; Sousa et al. 2011], where the vertices of the graph may be points on physical infrastructure or locations that a user might visit, and the edges connect the vertices to create allowed paths. Additional vertices can be inserted to represent finer-grained detail [Ekman et al. 2008; Kim et al. 2009]. Random street maps can be constructed using algorithms designed to reflect the characteristics of urban and rural roads [Barthélemy and Flammini 2008; Bitner et al. 2009; Strano et al. 2012].

[1]For clarity, graphs will be composed of *vertices* and *edges* to differentiate them from the mobile network’s nodes and communication links.

2.1.5. Forbidden Regions and Graph Constrained. Obstacles can form the basis of a graph, to ensure that nodes avoid them. A Voronoi diagram [Okabe et al. 2000] created from the obstacle vertices results in a graph where the edges lie between obstacles [Jardosh et al. 2005; Babaei et al. 2007]. Conceptually, any edge on the Voronoi diagram that intersects with an obstacle boundary becomes a "doorway." Alternatively, a graph can be constructed from a predefined set of vertices by placing an edge between two vertices if there is no obstacle between them [Aschenbruck et al. 2010a].

2.1.6. Bounded Free Space and Graph Constrained. To increase randomness in graph-based methods, they can be combined with bounded free space methods to constrain motion to a region within a certain distance of a graph element. For example, the vertices and edges can form the skeleton of the area within which nodes are constrained [Zhou et al. 2004], or each vertex can mark the centre of a square region from which a node can select a target [Ahmed et al. 2010].

2.2. Target Selection

A *target* is a point in the simulation space that defines a node's desired coordinates. The Target Selection model element defines the method by which a node's next target is selected. Note that spatial constraints or desired node behaviour may require a route to be defined between the node's current position and the target; this is discussed in the Pathfinding model element (Section 2.3).

2.2.1. Random Trips. An early definition of a *random walk* is a random process consisting of a sequence of discrete steps of fixed length in a random direction [Pearson 1905]. A generic term, *random trip*, was introduced for variants of the random walk that consist of random, independent node movements [Le Boudec and Vojnović 2006]. A node on a random trip chooses a new target stochastically via probability distribution functions (PDFs) for position, direction, distance, and/or travel time. The form of the PDF and its parameters can change the target selection behaviour:

—*Uniform:* A uniform distribution can be used to select a value with equal probability throughout a range [Rubin and Choi 1997; Royer et al. 2001; Rhee et al. 2011].
—*Normal:* A normal or Gaussian distribution reflects a tendency to select values near a mean value [Tolety 1999; Kang et al. 2011].
—*Exponential:* An exponential distribution is used to favour smaller values, with the probability decaying rapidly as values increase [Jardosh et al. 2005].
—*Power Law:* A power-law distribution also favours smaller values. A Lévy distribution, associated with patterns seen in nature, can be approximated to a truncated power-law distribution for distance under some conditions [Song et al. 2010; Rhee et al. 2011; Lee et al. 2012], although it has been shown that a Gamma distribution is a better fit to data collected on various animals [Edwards et al. 2007].

Discrete time and space. Discrete space random walks, also known as cellular automata, are constrained to a unit grid. The Target Selection method is invoked at each timestep, and direction is the random variable. Each potential direction is assigned a probability, and at each timestep, a node chooses a new direction based on these probabilities [Basagni et al. 1999; Bai et al. 2003]. Staying in the current cell can be an option [Perera et al. 2002; Kraaier and Killat 2004]. Motion in the x and y dimensions can be controlled separately by using two Markov chains, with transition probabilities assigned to states that can increment, decrement, or leave the coordinate unchanged [Chiang 1998].

Discrete time. In discrete time random trips, a node generates a new target at each timestep. The target coordinates can be sampled over the entire simulation space, or to enforce local movement, they can be sampled equivalently as a displacement in the x and y dimensions [Sánchez and Manzoni 2001; Piórkowski et al. 2009], as a distance and direction from the current position, or as a speed and direction [Tolety 1999; Bettstetter 2001; Kang et al. 2011]. Since the timestep size is fixed, the distance is directly proportional to the speed of a node; hence, this type of Target Selection method also determines node speed in the Motion element. Speed and direction chosen relative to current values yields smoother motion [Tolety 1999; Aravind and Cui 2008; Kang et al. 2011].

Variable time and space. In variable time and space random trips, movements may be of variable length in a random direction and variable in time. The target selection can implement this by stochastically selecting position, direction, distance, speed, and/or time (due to the relationship between speed, distance, and time). A node can sample its target coordinates directly from the entire simulation space [Johnson and Maltz 1996]; however, when there is a spatial constraint of forbidden regions, the Target Selection model must disallow targets selected in the forbidden regions [Papageorgiou et al. 2009; Wu et al. 2011]. When a node must be constrained to a particular region, the next target can be randomly selected from within the region [Aravind and Cui 2008; Batabyal and Bhaumik 2012]. When speed is defined in the Motion element, sampling time and direction from separate distributions yields a random target [Chen et al. 2007].

Random vertex. For graph-constrained models, space is discretized, although not necessarily into regular intervals. Selecting a vertex randomly over all vertices via a uniform PDF [Tian et al. 2002], or one dependent on distance [Jardosh et al. 2005; Lee et al. 2012], is the simplest target selection approach. The set of potential targets can be limited by selecting a subset of the graph's vertices from which the node may choose such as only allowing destinations at the outer edge of the graph [Sousa et al. 2011], or selecting from vertices associated with a type of activity [Zheng et al. 2010].

2.2.2. Location Bias. Location bias methods introduce an increased probability of selecting a target at a point, or within a region, that is more attractive based on some factor. The reason for the attraction may be an event that would attract nodes (e.g., a first responders scenario), or areas of the simulation space that are preferable (e.g., popular areas of a college campus). The attractiveness of a location can be dynamic and can be affected by distance.

Attraction points, also known as hotspots, can be static throughout the simulation, can be set to appear at specified times in the simulation [Jardosh et al. 2005; Huang et al. 2008], or can arrive and disappear following a random process such as a Poisson process [Borrel et al. 2005]. In the absolute case, hotspots become potential targets by virtue of their existence. In scenarios where hotspots appear dynamically, a hotspot may become the target for the nearest node [Schwamborn et al. 2010] or group of nodes [Ng and Zhang 2005; Rollo and Komenda 2009], or a random subset of nodes may be forced to select it as their next target [Jardosh et al. 2005]. Other approaches include constraining the target selection to a specified region around the hotspot [Huang et al. 2008] and having a node choose a target probabilistically from the set of hotspots. Hotspot regions can be defined, and a node can stochastically choose whether to be inside or outside a hotspot region, then choose a random point inside the selected region [Khadivi et al. 2006; Batabyal and Bhaumik 2012].

By dividing the simulation space into cells, location bias over all cells can be implemented by assigning a probability to each cell. This probability can be derived from empirical data [Nunes and Obraczka 2011], or it can be a function that is inversely proportional to the distance to the cell [Bhandari et al. 2010]—or for an embedded graph, the number of shortest path hops to the cell [Babaei et al. 2007]. The next target may be a specific point inside the selected region, such as a graph vertex [Babaei et al. 2007; Bhandari et al. 2010], or a random point within the selected region [Nunes and Obraczka 2011].

Methods of extracting hotspot regions from empirical data are given in Nunes and Obraczka [2011] and Kim et al. [2006]. The former is cell based, whereas the latter represents hotspots as two-dimensional normal distributions. The latter approach addresses problems associated with determining the appropriate cell size: if too small, a hotspot can become diluted; if too large, a hotspot region can be overestimated.

2.2.3. Random Trips with Location Bias. Location bias can be combined with discrete time and space random trip methods by letting the probability of selecting each direction be dynamic to simulate the appearance of an attraction point in the simulation [Guo et al. 2010].

2.2.4. Social Interaction. Scenarios involving people may require a means of reflecting a node's desire to be near other nodes. This can be accomplished through the application of network theory. In a social network, the strength of the relationships between every pair of nodes is defined in its social interaction matrix, which in network theory terminology is equivalent to a weighted adjacency matrix. Each element (i, j) of the matrix defines the strength of the interaction between nodes i and j. There are several models that can be used to create a social interaction matrix by generating networks with specific qualities, including:

—*Small World:* A sparsely connected, decentralized, highly clustered network with a large number of nodes [Watts 1999], also known as the Caveman model, implemented in Musolesi and Mascolo [2007], Fischer et al. [2010], and Boldrini and Passarella [2010].
—*Scale Free:* A network that is scale invariant—that is, the number of social connections for any given node is independent of the number of nodes in the network [Barabási and Albert 1999], used by Herrmann [2003].
—*Holme-Kim:* A scale-free network with additional clustering [Holme and Kim 2002], used by Fischer et al. [2010].
—*Toivonen:* A highly clustered network where highly connected nodes are connected to other highly connected nodes and the distribution of node degree is broad [Toivonen et al. 2006], used by Fischer et al. [2010].
—*Assortatively Mixed Network:* A network in which nodes connect preferentially to other nodes that are like them in some way [Newman 2003]. For example, in Wang et al. [2011], the attributes of gender, age, academic major, and type of research institute are applied.

To select a node's next target, one can use the social interaction matrix to choose a community—and from that select, a location. The social interaction matrix can be analyzed to identify cliques (groups of fully connected nodes) [Herrmann 2003] or communities (densely connected groups) [Musolesi and Mascolo 2007; Wang et al. 2011], for instance using the Girvan-Newman [Newman and Girvan 2004] or modularity maximization [Kang et al. 2011] algorithms. A node's next target community can be chosen as that with the maximum total social attraction [Musolesi and Mascolo 2007; Wang et al. 2011] or can be selected probabilistically over all communities based on the total social attraction [Boldrini and Passarella 2010]. Once the community has been selected, the target coordinates could be randomly selected within the selected community's assigned region [Musolesi and Mascolo 2007; Boldrini and Passarella 2010] or taken as the location of the node in the selected community that has the highest individual social interaction value [Wang et al. 2011].

Alternatively, the attraction to a location or region might increase with its "popularity." This could be implemented by assigning to it a PDF that is a periodic function of time, increasing at certain times of day. The probability of selecting a location can be computed as proportional to the number of nodes at or going to the location [Borrel et al. 2005]. By dividing the simulation space into rectangular cells, a probability can be assigned to each cell that is a function of its popularity. One approach is to initially place nodes using the growth and rewiring properties of a scale-free network, where an initial cell is selected with a probability proportional to the number of nodes currently in each cell [Lim et al. 2010]. In another approach, the popularity of a cell might be interpreted from each node's individual perspective, as equivalent to the population of the region as last perceived by the node [Mei and Stefa 2009].

There are other social factors that may influence a node's selection, such as the social attraction of a node to the "owner" of a location [Fischer et al. 2010]. A node repulsion factor can be used to reduce the frequency of meetings between nodes with a weak social attraction caused by them having a common social acquaintance [Fischer et al. 2010].

2.2.5. Social Interaction with Location Bias. In a combined method, the social interactions can be augmented to include attraction to a point or region to select the next target. If the desired behaviour of a node is to prefer shorter distances, the attraction to a location can be implemented as a decaying function of distance [Borrel et al. 2005; Mei and Stefa 2009; Boldrini and Passarella 2010]. A node may have an introverted nature, which can be represented using the above distance and popularity terms modified by a factor that increases the effect of the distance term and decreases the effect of the popularity term, and vice versa for extroverted nodes [Mei and Stefa 2009].

2.2.6. State Transitions. When a characteristic of a node can be expressed in terms of a finite number of "states" and the probabilities associated with a node's next state depend only on its current state, the target selection process can be expressed as a finite state machine. A state may reflect the node's modality, where it is, or what it is doing. A transition probability matrix (or transition table) can be used to select the next target; when a node makes a transition from one state to another, it selects its next target based on its new state. State transition methods are commonly used as the first stage in a multiple-method target selection process.

If the states of a model represent the geographical region that a node occupies, such as in a model where the simulation space is divided into cells, the new state determines the region and the target could be selected as a random point in that region. The transition matrix can be populated such that a node must choose an adjacent cell [Liang and Sheng 2005], or defined to reflect a desired node density for each cell [Ueno et al. 2011], which may be useful for including obstacles by disallowing regions of the simulation space. The states could equivalently be defined as communities, each of which is assigned a geographic space, that a node may choose to move between [Hsu et al. 2009]. This allows for overlapping regions.

If the state is instead the *type* of cell [Murray and Pesch 2004], the transition matrix can be defined to reflect a time-dependent probability that a node in one type of cell will move to another type of cell. For example, in the morning, a node in a "town" might have a higher probability of moving to a "city" cell, and the reverse in the evening. If a different cell type is selected, a cell of that type is chosen randomly.

When the node's states are activities, the transition matrix gives the probability that a node chooses its next activity based on its current activity. The location of the next target is determined by the new activity. Each activity may have a fixed location [Stepanov et al. 2005], a target may be randomly selected from a set of locations associated with the selected activity [Hsu et al. 2005; Zheng et al. 2010], or it may be selected within a region associated with an activity using a uniform [Scourias and

Kunz 1999] or normal [Cho et al. 2011] distribution. Fixed activity locations and regions of activity can be combined [Schwamborn et al. 2010]. Transition probabilities can be derived from time usage surveys [Scourias and Kunz 1999; Kim et al. 2009; Zheng et al. 2010] or extracted from traces [Hsu et al. 2005; Stepanov et al. 2005; Kim et al. 2006]. Multiple tables can be used to reflect activities being more likely to occur at a certain time of day [Scourias and Kunz 1999; Hsu et al. 2005, 2009]. For example, a node may be more likely to go out to eat at mealtimes. Time dependence can be introduced by defining the transition probabilities as a continuous function of time [Cho et al. 2011], which smooths the transitions between time periods, and increases the flexibility of the model by allowing the transition probability to be a continuous function of the time since the activity was last done.

Finally, when the states reflect the node's current modality—for example, its mood, state of being, or other intrinsic characteristic—the transition matrix can give the probability of switching to another modality. The selected modality then determines the next target selection method. For example, consider a node with two states, *exploring* and *revisiting*. For either state, a node explores (visits a new random target) with a probability p that is a power-law function of the number of locations it has already visited, and with $1 - p$ it revisits a previously visited location [Song et al. 2010; Munjal et al. 2011]. This has been shown to agree with empirical data.

2.2.7. Location Bias with State Transitions and Social Attraction. In an approach that combines modality state transitions with location bias, one cell is defined as a node's home cell, and a node's state is either inside its home cell or in an outside cell [Boldrini and Passarella 2010]. When in an outside cell, it selects a random target in its current cell with a probability p that reflects its desire to be away from home, and with $1 - p$ selects a random target in its home cell. When at home, the probability of choosing an outside cell is a function of social attraction.

2.2.8. Deterministic. Deterministic targets do not have a stochastic element to them; the targets are either manually predefined or selected via rules. For example, in a scenario where all of the vertices on a graph must be visited, a node's target can be selected as the nearest nonvisited vertex [Aschenbruck et al. 2010a]. A deterministic method based on the social interaction matrix uses a predefined set of meeting locations as targets [Herrmann 2003].

2.2.9. Checkpoint List. A checkpoint list is an ordered set of targets that a node must visit and used in conjunction with another Target Selection method. In the simplest checkpoint list, a node visits a manual set of targets in the order assigned [Hong et al. 1999]. A node's checkpoint list can be selected as a random subset of a larger set of targets, ordered stochastically according to a power-law function of the distance to the target [Lee et al. 2012]. Another approach is to generate targets for each node as a random walk consisting of a random number of steps and ordering the checkpoint list using a PDF that may be uniform [Tuduce and Gross 2005].

Time can provide the stochastic aspect of a checkpoint list. For example, sequential activities selected via an activity-based transition matrix can be assigned times such that a randomly selected pause time and maximum travel time allows for the next activity to be reached [Zheng et al. 2010], or cliques identified using the social interaction matrix can be assigned checkpoint times such that all cliques are able to meet [Herrmann 2003].

2.3. Pathfinding

When the simulation space has obstacles, a node may not be able to move directly to its next target; in these cases, a path can be defined that consists of a set of intermediate points through which the node must pass. In other cases, a model might require a node to behave nonlinearly between targets. The Pathfinding model element determines the coordinates of intermediate points between targets.

2.3.1. Graph Traversal Algorithms. When the Spatial Constraint is limited to graph edges and vertices, the node must find a path through the intermediate vertices to its target vertex. Graph traversal algorithms are applied to determine the path. When the Spatial Constraint calls for forbidden regions, a path from source to target must be found that avoids obstacles or undesirable areas. In such cases, a graph can be created using the obstacle vertices, a vertex for the source, and a vertex for the target [Aschenbruck et al. 2007; Medina et al. 2010; Wu et al. 2011]. A graph traversal algorithm can then be applied to find the path.

The Dijkstra shortest path algorithm [Rosen 1988] is the most commonly referenced algorithm for determining the path with lowest cost between a given source and target vertex in the graph. The A^* algorithm is an alternative to Dijkstra's algorithm [Schwamborn et al. 2010]. The cost can include variables such as edge length [Tian et al. 2002; Jardosh et al. 2005; Babaei et al. 2007; Ekman et al. 2008], edge type [Ahmed et al. 2010], speed [Zheng et al. 2010], time [Mogre et al. 2007], and edge or vertex desirability [Medina et al. 2010].

For very large graphs, speed can be increased by implementing the graph traversal algorithm in a hierarchical manner, determining the optimal route at a high level and increasing granularity just along that route [Buckland 2005; Kim et al. 2009].

2.3.2. Stochastic Graph Traversal. To add diversity to a graph traversal algorithm, edges can be assigned a probability of being selected that reflects their attractiveness [Stepanov et al. 2005]. When Spatial Constraints combine graph-based and bounded free space, the coordinates of the intermediate path points can be randomly sampled within a square region around the vertex selected using a standard graph traversal algorithm to give a more realistic variation in movement, where the size of the region reflects the freedom of movement [Ahmed et al. 2010].

2.3.3. On-the-Fly. In on-the-fly methods, a node makes a decision at each intermediate path point. For on-the-fly graph traversal, a node makes a decision at every vertex. A node may stochastically favour edges in the direction of the target that have a higher desirability [Sousa et al. 2011], or a probability can be assigned to each outgoing edge based on empirical data, perhaps taking into account the node's previous vertex, current vertex, origin, and target [Yoon et al. 2006].

To navigate around obstacles in free space on-the-fly, an iterative process can be used [Papageorgiou et al. 2012]: when a straight line between the current position and the target intersects an obstacle, the next intermediate point is the vertex of the obstacle that is closest to the final destination. The process repeats until the target is reached.

2.3.4. Biased Random Trips. To create a guided random walk from origin to target [Murray and Pesch 2004], a node can move a specified distance in each timestep at a random angle heavily biased toward the target, until the node is within a specified distance of the target. Alternatively, at each timestep, a node could choose a random point inside a square region directly in front of the node, centred along the vector connecting its current position to its target [Rollo and Komenda 2009].

The path to a target may need to follow a pattern where at first the displacements are large, but the displacement becomes incrementally smaller as it approaches the target. This has been suggested for search and rescue operations [Ng and Zhang 2005].

To generate paths similar to empirical data, a set of points can be generated randomly within a rectangle defined by the origin and target at opposite corners, with their traversal order sorted by distance from the target [Kim et al. 2006]. The number of points can be drawn randomly from a normal distribution centred on the average number of points observed in the data for walks between the current region and the target region.

2.4. Motion

The Motion model element dictates the speed and direction of a node's movement from one target to the next, or to its next intermediate point if there is pathfinding. It includes methods that influence a node's velocity or acceleration vector. Note that velocity is composed of both speed and direction; for linear motion between two targets, the direction is predetermined and speed defines the Motion model.

It is important to specify when the contributing variables are resampled in the Motion method—for example, for each node at initialization, after a target has been reached, or at each timestep. A random process (e.g., Poisson or exponential) can be applied at each timestep to decide whether to change the speed or direction [Bettstetter 2001]. A semi-Markov process[2] with phases of motion *speed up*, *middle smooth*, *slow down*, and *pause* [Zhao and Wang 2009] will change its Motion model at each transition, remaining in each phase for a random duration.

2.4.1. Predetermined Speed. Trivially, the speed of all nodes can be the same throughout the simulation [Batabyal and Bhaumik 2012]. For a scenario where nodes must reach predefined locations on a predefined schedule, the speed between each checkpoint can be set so that the next checkpoint will be reached on time [Hong et al. 1999].

2.4.2. Random Speed Distributions. The PDF for node speed reflects the node's desired speed behaviour:

—*Uniform*: A uniform distribution between a maximum and minimum value is common (e.g., Johnson and Maltz [1996], Hsu et al. [2009], Boldrini and Passarella [2010], Lim et al. [2010], and Wang et al. [2011]). It has been shown that this distribution should not include zero [Yoon et al. 2003].
—*Normal*: A normal distribution centred on the average speed [Kim et al. 2009] is consistent with pedestrian behaviour [Helbing et al. 2000].
—*Lognormal*: The speed observed in empirical data has been shown to fit a lognormal distribution [Kim et al. 2006].
—*Discontinuous*: A discontinuous PDF can be constructed to reflect a tendency toward a set of preferred speeds [Bettstetter 2001], where the PDF is uniform except at each preferred speed where a delta function creates a spike in the probability.

Alternatively, if the travel distance to the target is set, the time for the motion can be sampled from a random distribution from which the speed can be calculated. A simple uniform PDF [Tuduce and Gross 2005] can be applied; however, both gamma [Zheng et al. 2010] and power-law [Rhee et al. 2011] distributions have been shown to fit empirical data.

2.4.3. Regional Speeds. To specify areas of high or low speed, regions of the simulation space can be assigned unique distributions from which to select speeds [Aschenbruck et al. 2007; Piórkowski et al. 2009; Du et al. 2012]. Speed distributions can be defined to reflect caution in regions that a node has not yet visited [Aschenbruck et al. 2010a]. If a target has been visited already, a node can choose a speed from a distribution yielding higher speeds; otherwise, a node chooses from a distribution yielding lower speeds.

[2]A semi-Markov process is a continuous-time stochastic process where the state transitions form a Markov chain, and the time between jumps are random variables whose distribution functions may depend on the two states between which the move is made.

2.4.4. Graph Edge Speed. Graph-based models can have a speed associated with each edge [Sousa et al. 2011], which can be selected a priori from a distribution—for example, uniform [Schwamborn et al. 2010]. Time-dependent speed limits assigned to each edge can be used to truncate speeds drawn from another distribution [Zheng et al. 2010].

2.4.5. Speed Adjustment for Collision Avoidance. When nodes move along the same graph edge at different speeds, speeds must be adjusted if collisions are to be avoided. This can be accomplished by checking each pair of nodes i and j at each timestep and modifying their speeds accordingly: if node i is approaching node j from behind, the distance between them is smaller than the safe distance, and the speed of node i exceeds that of node j, node i's speed is reduced to that of node j [Bai et al. 2003]. Collision avoidance can also be accomplished by adjusting the speed via an empirically derived relationship between distance and speed [Kim et al. 2009].

2.4.6. Smooth Transitions. To reduce sudden changes in velocity, the next speed and direction can be sampled from normal distributions centred on the current speed and direction [Tolety 1999; Aravind and Cui 2008; Zhao and Wang 2009; Kang et al. 2011]. Further smoothing can be accomplished by applying a constant acceleration until the desired speed is reached [Zhao and Wang 2009], or by applying a random acceleration term in the transition to a new speed [Bai et al. 2003; Guo et al. 2010; Bettstetter 2001]. For a transition to a new direction, the change can be implemented incrementally over a predefined period of time [Bettstetter 2001].

2.4.7. Steering Forces. Motion can be governed by steering forces acting on nodes, where targets and the environment exert attractive and repulsive forces on nodes. Individual parameters could be defined for every node to tune its behaviour. Note that all forces between nodes are covered in the Group Dynamics element as Internode forces (Section 2.6.3).

The OpenSteer library [Reynolds 1999] provides the following individual node steering forces:

—*Seek*: Radially align the velocity toward the target.
—*Flee*: Radially align the velocity away from the target.
—*Pursuit*: Predict the future position of a moving target and apply *seek* to the predicted target.
—*Evasion*: Predict the future position of a moving target and apply *flee* to the predicted target.
—*Arrival*: Similar to seek, but when the node is inside a stopping radius of the target, the velocity is ramped down linearly.
—*Obstacle Avoidance*: Avoid obstacles by keeping an imaginary cylinder of free space in front of the node. If an obstacle intersects with the cylinder, a steering force to the side is applied.
—*Wander*: Apply random steering but smooth the motion by constraining the next random force to the surface of a sphere located slightly ahead of the node.
—*Path Following*: Apply a steering force that constrains the node to a tube of a specified radius that follows a defined path curve. The position of the node in the next step is computed, and if this is outside of the tube, *seek* is used to steer the node to the closest point on the curve.
—*Containment*: Similar to *path following*, if the future position is outside of the container, a *seek* steering force brings the node back toward an inside point.
—*Flow Field Following*: If the next position of a node will be inside a flow field, a steering force aligning with the flow is applied.

Subsets of these, or minor variations thereof, are implemented in Legendre et al. [2006], Holliday [2008], and Medina et al. [2010]. Obstacle avoidance can be accomplished using other forms of the force, such as an exponential decay [Williams and Huang 2009]. Forces of a form inspired by Newton's universal law of gravitation can attract a node to a point in the simulation space, inversely proportional to a power of the distance of the node from the point [Nelson et al. 2007; Du et al. 2012].

Solving the equations of motion. The change in velocity and position can be derived from forces using Newton's first law, where $\vec{F} = m\vec{a}$ gives the acceleration vector corresponding to the summed force. Using unit mass, Euler integration can be used to solve for the new velocity and position [Reynolds 1999; Legendre et al. 2006; Holliday 2008; Medina et al. 2010][3]:

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \vec{a}(t+\Delta t)\Delta t, \tag{1}$$

$$\vec{x}(t+\Delta t) = \vec{x}(t) + \vec{v}(t+\Delta t)\Delta t. \tag{2}$$

The velocity can be truncated to upper and lower bounds to reflect realistic motion limits [Reynolds 1999; Liu et al. 2010].

As humans do not follow the same acceleration rules as other objects, it has been suggested [Nelson et al. 2007] that rather than smoothly accelerating over a timestep, humans change to a new constant velocity very quickly. In this case, the velocity can evolve independently of the current velocity, $\vec{v}(t+\Delta t) = \vec{a}(t+\Delta t)\Delta t$.

2.4.8. Steering Forces with Other Methods. When steering forces are used to influence the motion of a node that is otherwise obeying another Motion method (e.g., constant velocity toward a target, or a random walk), the two can be combined using methods that act on the velocity. Assuming unit mass and a small enough timestep, the contribution of the force can be approximated in terms of a change in velocity, $\Delta\vec{v}_F = \vec{F}\Delta t$; a node's velocity can be updated by vector addition of the velocity change due to the forces with the velocity change due to the other Motion method [Rossi et al. 2005]. The position can then be updated via Equation 2. In another method, the total steering force is applied only to the direction vector by adding the direction of the summed forces to the direction vector generated by the target selection method [Williams and Huang 2009; Du et al. 2012].

On a three-dimensional surface, the z-plane can be considered separately from the (x, y) plane by letting the gravitational force alter the speed of a node and letting a path defined on the (x, y) plane determine the direction [Liu et al. 2010]. To solve for the new position, the velocity can be determined via Equation 1, but the new position is solved by advancing in the (x, y) plane in very small timesteps, with each move placing the node on the surface at the corresponding z coordinate, stopping when the required displacement $v\Delta t$ has been reached.

2.5. Pause Time

The Pause Time model element determines how long a node waits between reaching its target and proceeding to the next. The Pause Time model element is optional.

2.5.1. Constant. The pause time can be set to be constant throughout the simulation, using the same value for all nodes [Johnson and Maltz 1996; Papageorgiou et al. 2009; Lim et al. 2010; Ueno et al. 2011] or using a different constant value for each node.

[3] Legendre et al. [2006] use Euler integration (Eq. 1) for velocity; however, the position is given in the paper as $\vec{x}(t+\Delta t) = \vec{x}(t) + \vec{v}(t+\Delta t)\Delta t + \vec{a}(t)\Delta t^2/2$. This is possibly a transcription error, as it is close to the second-order Taylor expansion of position about t, consistent with Leapfrog integration [Frenkel and Smit 2002].

2.5.2. Random Distribution. A new pause time can be selected from a random distribution for each node after a target has been reached. Parameters for the PDFs can be extracted from empirical data. Examples from the literature include:

—*Uniform*: Tian et al. [2002], Borrel et al. [2005], Hsu et al. [2009], and Wang et al. [2011].
—*Exponential*: Murray and Pesch [2004]. Fit to empirical data in Kim et al. [2009].
—*Truncated Power Law*: Mei and Stefa [2009] and Rhee et al. [2011]. Fit to empirical data in Song et al. [2010].
—*Pareto*: Ekman et al. [2008]. Fit to empirical data in Tuduce and Gross [2005].
—*Log-normal*: Fit to empirical data in Kim et al. [2006].
—*Weibull*: Fit to empirical data in Zheng et al. [2010] and Schwamborn et al. [2010].

Empirical data can also be used directly by binning the observed durations to create a cumulative distribution function (CDF) [Scourias and Kunz 1999; Hsu et al. 2005]. The parameters can vary depending on the time period [Scourias and Kunz 1999]. For example, a node may visit a coffee shop for a longer period over the lunch hour.

2.5.3. Regional. The pause times can be defined to be geography dependent by using different parameters for a pause time PDF in different regions. This may be useful in scenarios where users tend to stop for longer periods of time in particular areas, such as an airport lounge or a hotspot region. This can be implemented by dividing the entire simulation space into rectangular cells [Liang and Sheng 2005] or treating hotspot regions individually and all space outside the hotspots as one region [Hsu et al. 2005; Kim et al. 2006].

2.5.4. Task Based. The pause time distribution can be parameterized to reflect a node's task. A separate set of parameters can be used for each activity type [Zheng et al. 2010], or a separate empirically derived pause time CDF can be generated for each activity type [Scourias and Kunz 1999].

2.6. Group Dynamics

The optional Group Dynamics model element describes node behaviours that result in groups of nodes staying spatially close to one another. Groups behaviours may be defined relative to a reference, which can be a designated leader node or an imaginary node. The behaviour of a reference node is described by the other models elements; the dynamics described in this section are for the follower nodes. Group behaviours can also be brought about by defining forces between the nodes. Rules that form or split groups of nodes are another aspect of group dynamics.

The Group Dynamics element is used for models where the group is the central concept—that is, a node must ask "what is my group doing?" before making a decision. It excludes methods where nodes do not directly influence one another and can make decisions individually, outside of the group construct. These methods are covered in the Target Selection model element, such as social interaction or location bias, where nodes are assigned a community or activity that uses a predefined static spatial region. Although clustering emerges from these methods, it is brought about indirectly.

2.6.1. Relative to a Reference. The target and motion of a follower node can be defined relative to a reference point, which can be a designated leader or an imaginary reference node. A group's leader can be predetermined, or if a social interaction matrix is present [Kang et al. 2011], choosing the most socially attractive node in the group as the leader. There are two approaches to selecting targets for followers in the literature; in the first, followers select a target position relative to the reference node's target and a speed relative to the reference node's speed, whereas in the second, the followers choose their velocity vector (direction and speed) relative to the reference node's velocity.

When the reference chooses its target and speed, each follower node in the group can be assigned the same target [Rollo and Komenda 2009] or a target some distance from the reference, which can be as simple as a predetermined fixed offset from the target for each node [Reidt and Wolthusen 2007]. Random selection methods include selecting a displacement $(\Delta x, \Delta y)$ [Sánchez and Manzoni 2001], a random point within a square or circular region about the reference point [Ng and Zhang 2005], or a random vector (r, θ) with its origin at the reference point. The distributions used to generate the random vector can be selected to reflect the desired shape of the group, such as circular [Hong et al. 1999], elliptical [Chen et al. 2010], or a diamond aligned with the direction of the reference point [Ning et al. 2008]. The corresponding speeds may be the same for all nodes in the group [Hong et al. 1999], the nodes in a group may select their speed from a uniform distribution bounded to an interval around a common group speed [Ng and Zhang 2005], or each node's speed may be modified such that all nodes in the group arrive at their targets simultaneously [Gu et al. 2011].

For models where the reference node selects a direction and speed (i.e., a random trip with variable time and space), a follower node's velocity may be computed as the vector sum of the reference node's velocity with a randomly drawn velocity deviation vector [Wang and Li 2002]. Or, they could be selected from speed and direction distributions centred at those of the reference node [Kang et al. 2011].

2.6.2. Merge and Divide. Scenarios may require groups to merge or divide. Merging allows a node or group of nodes to join another group, which is useful for scenarios where a mobile device is transferred to another user, or a user boards a vehicle with other mobile devices. For this case, groups may be merged deterministically, such as at set time intervals following this rule: if the distance between the groups is less than some critical distance, they are merged and the centre of the new group is randomly chosen within the overlapping area of the old groups [Gu et al. 2011]. When a scenario involves nodes being assigned to tasks, the group merging rule can be that nearby nodes not currently assigned to a task form a group by proceeding to a rendezvous point before moving to the task location as a group [Rollo and Komenda 2009]. Group merging can be used to ensure that there are not too many small groups with no target. For instance, a rule might be that if a group is small and has not had a target assigned to it for a threshold time, its nodes are merged into another group [Ng and Zhang 2005].

Group division provides the ability for some nodes in the group to change their target, which could be used for a scenario in which a user drops their mobile device. This could be implemented as a random process, such as by randomly selecting the number of groups to be divided at periodic intervals, and if selected, splitting a group by selecting two new group targets and randomly assigning each node to one group or the other [Gu et al. 2011]. Group division is also useful for ensuring that there are enough groups to reach every target; this can be accomplished deterministically as follows: if there is no available group for a new target, the group that is nearest to the new target and of sufficient size will divide [Ng and Zhang 2005]. In a scenario where all paths in a graph must be covered, such as in a building search scenario [Aschenbruck et al. 2010a], at each intersection with more than one path choice the group can split to fully cover the building.

2.6.3. Internode Forces. Internode forces can introduce complex group behaviours. Steering forces that result in group behaviour include:

—*Separation*: Move away from nearby nodes by adding a steering force opposite to them. The form of the force usually decays with distance, such as a power law [Reynolds 1999; Legendre et al. 2006; Borrel et al. 2009; Liu et al. 2010] or an exponential decay [Williams and Huang 2009]. The magnitude of the force may depend on the angle of the interaction, being greater for nodes in front and less for nodes to the side [Legendre et al. 2006].

—*Cohesion*: Keep nodes together by finding the mean position of all others in the group and applying a force toward it [Reynolds 1999; Legendre et al. 2006], or by computing the force with a function that has an attractive peak around a desired distance [Williams and Huang 2009].

—*Leader Following*: Keep nodes in the vicinity of a leader (an existing node or a reference point) by applying a force toward the leader that increases with distance [Liu et al. 2010] or is constant within a fixed radius of the leader and decays with distance [Rossi et al. 2005]. The former case is equivalent to Cohesion using a reference point at the group's mean position. This can also be accomplished by setting an *arrive* point (see Section 2.4.7) just behind the leader [Holliday 2008].

—*Alignment*: Align the velocities with a steering vector that is the difference between the group average velocity and the node's velocity [Reynolds 1999; Legendre et al. 2006; Holliday 2008; Liu et al. 2010].

—*Unaligned Collision Avoidance*: Avoid a potential future collision with another node by steering to avoid the site of that potential collision [Reynolds 1999].

When the forces have been computed, the velocity and position can be updated using the methods discussed in Section 2.4.7.

3. FRAMEWORKS

Conceptual frameworks are concepts or ideas around which a model can be designed. Conceptual frameworks that allow the implementation of multiple mobility models are important for scenarios where node behaviour varies depending on location or task, the involvement of hierarchical authorities, or intrinsic behavioural characteristics (e.g., modelling people and vehicles). Software frameworks describe software design and/or development approaches.

3.1. Conceptual Frameworks for Multiple Mobility Models

In a given scenario, a node's mobility model may need to vary depending on its circumstances, such as its location, activity, or role. To reflect a change in behaviour caused by a node's position in the simulation space, a mobility model can be assigned to a static region, where all nodes obey that model when inside that region [Lu et al. 2006; Güneş et al. 2007]. A model can capture behaviour determined by the role of a node (e.g., soldier vs. pedestrian, car vs. ambulance) by allowing each node [Boschi et al. 2008; Fongen et al. 2009] or group of nodes [Blakely and Lowekamp 2004; Reidt and Wolthusen 2007] to follow a different model. The behaviour of a node may also vary depending on both its role and its current location. This could be implemented by using a different set of model parameters when a node is in a given region along with a role-based Target Selection method [Aschenbruck et al. 2007], or by defining the model and parameters dynamically based on a node's role and location relative to an event [Nelson et al. 2007; Huang et al. 2008].

Mobility models can be applied hierarchically. For instance, a node may select and move to a target within a region using a primary model, and move around inside that region under a secondary model for a period of time until the higher-level model selects a new region [Ng and Zhang 2005; Ekman et al. 2008; Kim et al. 2009]. The secondary model can be selected to reflect a behaviour associated with the current activity [Holliday 2008]. This could apply to scenarios where a node must go to a region to complete a task, such as shopping or search and rescue operations. In such models, an additional pause time can be used to define how long a node spends obeying the secondary model. For instance, a power law distributed pause time determines how long a node moves about in a cell via a secondary model in Nguyen et al. [2011].

Steering forces can be used exclusively to replicate arbitrary mobility models, when each node has its own independent dynamic steering behaviour rules [Medina et al. 2010]. Obstacles and paths can be modelled in a cell-based environment as attractive or repulsive forces acting on the nodes. Targets can be specified as an attractive force for each node or group based on their activities, and internode forces can be applied to dictate group behaviour. The form of the force functions and the parameters can vary for each node depending on desired behaviour.

In handling multiple hierarchical groups such as those seen in military structures, it may be helpful to have each group of nodes move relative to their own coordinate system. This can be implemented by placing the origin of the coordinate system of a subordinate group at the coordinates of its parent object [Fongen et al. 2009] or at a reference point [Blakely and Lowekamp 2004].

3.2. Software Frameworks

BonnMotion [Aschenbruck et al. 2010b], the Texas Unified Framework [Choi et al. 2007], MobiSim [Mousavi et al. 2007], and the Mobility Model Simulation Environment (MOMOSE) [Boschi et al. 2008] are open source, extensible object-oriented frameworks in which arbitrary models can be implemented. In MobiSim and MOMOSE, a mobility model is associated with a subset of the nodes, and multiple models can be used in a simulation. In BonnMotion, a single model class is defined for all nodes, and different behaviours can be assigned to different node types within the class. In the Texas framework, the model is defined as a method of the node class.

Open source network simulation software such as NS-2/NS-3 [NS-3 Consortium 2012] and The ONE [Keränen et al. 2009] contain basic mobility models, can be extended to include new mobility models, and can read trace files generated by the previous. Commercial products such as QualNet [Scalable Network Technologies 2013] and Legion Studio [Legion Software 2013] contain interfaces allowing for the specification of scenarios but are less readily extensible.

4. TAXONOMY APPLICATION

The primary purpose of the taxonomy is to assist authors in completely describing their models as a unified whole so that readers can more easily digest the various aspects of a model. To illustrate this aspect, the taxonomy is used to describe a relatively complex model in Section 4.1. An additional benefit of the taxonomy is that it can provide researchers with a framework within which to assess their requirements of a mobility model. Two example scenarios are assessed in Section 4.2. Further, the taxonomy allows a means of analysing the literature for trends, which is discussed in Section 4.3.

4.1. Example Model Description

A model must be described at a high level before breaking it down into its elements, and the elements each require a thorough description. If there are multiple models used, such as in models with multiple node types, the model for each must be described. To give a flavour of how the taxonomy might be applied, a brief description of the Mission Critical Mobility (MCM) model [Papageorgiou et al. 2012] is given in terms of the taxonomy elements. MCM was selected because it is a relatively complex model that has multiple node types, obstacles, on-the-fly pathfinding, group dynamics, and target selection that is affected by external factors.

High-level concepts and behaviours. There are two node classes in MCM—emergency workers and medical staff—each following a slightly different cycle that defines their behaviour. Emergency workers get a target, move to the target, call for reinforcements if necessary, and pause. Medical staff go to a fixed home base location before beginning the same cycle. Nodes are divided into groups of the same class, and each group is assigned a leader node. MCM is based on the concept of events, which have coordinates associated with them, and a severity: normal, serious, or complex. After a group arrives at an event, reinforcements are called based on the severity of the event. Each class of node has a FIFO queue associated with it that contains the coordinates of serious or complex events waiting for reinforcements. If the event is serious, its coordinates are added to the queue associated with the group's class. If an event is complex, its coordinates are added to both queues. In terms of the taxonomy elements, we have the following:

—*Spatial Constraints*: Obstacles are represented by rectangular forbidden regions in a bounded free space (Sections 2.1.2 and 2.1.3).
—*Target Selection*: The group's designated leader checks the event queue corresponding to its class for a target, and if one exists, it chooses that as its next target. Otherwise, the leader's next target becomes the coordinates of a new event: the coordinates are sampled from a uniform random distribution, discarding coordinates within forbidden regions (Section 2.2.1, *variable time and space random trip*). The event severity is selected from a uniform distribution.
—*Pathfinding*: An on-the-fly method of creating a path from obstacles recursively draws a line between current position and target; if an obstacle is crossed, the vertex on the nearest edge of the obstacle that is closest to the target is added as a path point (Section 2.3.3).
—*Pause Time*: Constant over the entire simulation, the same value for all nodes (Section 2.5.1).
—*Motion*: Speed is selected from a uniform distribution, for each node each time a new target is selected; the motion is linear from path point to path point (Section 2.4.2).
—*Group Dynamics*: Followers are assigned a target relative to the group leader's target at a constant offset (Section 2.6.1).

4.2. Scenario-Based Model Requirements

A scenario gives a description of node behaviours in a given setting. These behaviours must be reflected in the corresponding mobility model. The taxonomy provides a framework to aid in constructing such a model. In this application, the scenario definition leads to a set of behavioural requirements. For each element in the taxonomy, model requirements are derived from the behavioural requirements, which are then used to guide the choice of method. Two examples are given in this section. First, a "toy" problem of visitor movements at a state fair is presented, which gives an illustration of a derivation of model requirements based on a simplified, albeit unrealistic, scenario. Second, a more complex example of a military tactical mobile ad hoc network (MANET) is given to show the process for a real-world scenario.

4.2.1. State Fair. Consider a state fair that consists of six games, all situated evenly within a 50m square. There is a gate at the centre of the western boundary from which visitors may enter and exit, but once inside, visitors can pass through one another and through the game booths. Each game can be played only once by each visitor, and to play a game, a visitor stands on a spot located at the centre of the south side of its booth. Each game takes around 6 minutes to play and can be played simultaneously by multiple players. On the day of the fair, one visitor arrives every 5 minutes, prepared with an ordered list of the games they will play. A visitor must visit at least three but no more than six booths. Visitors move directly from one game to the next, and everyone moves at the same speed of 2km/h inside the fairground.

The goal is to create a model that will give us the trajectories of the nodes over time for this scenario. For each model element, the required behaviour of the model is assessed to yield an appropriate method, as follows:

—*Spatial Constraints*: The fair occupies a defined area, and nodes enter and exit the space via one preset entry point on the boundary. Since visitors can pass through the booths, obstacle avoidance is not required. Hence, the spatial constraints are bounded free space (Section 2.1.2) with constant periodic arrivals at the entry point.
—*Target Selection*: Prior to entering, each visitor generates a list of games to visit with no repetition, uniform random length between three and six, and randomly ordered. From the perspective of a visitor, a single (x, y) coordinate represents the booth's location. A viable method is a random trip (Section 2.2.1) in conjunction with a checkpoint list using these coordinates (Section 2.2.9). Since a visitor leaves the fair after playing the games on its list, the gate coordinates are appended to each checkpoint list.
—*Pathfinding*: Since there are no obstacles and no interactions between nodes, there is no requirement for nonlinear motion between targets; hence, this element is not required.
—*Motion*: The motion between path points is linear at a constant speed that is the same for all nodes (Section 2.4.1).
—*Pause Time*: The time spent at any booth is assumed to follow a normal PDF centred on 6 minutes, which is a random pause time method (Section 2.5.2).
—*Group Dynamics*: Nodes do not interact with each other, so this element is not required.

This toy model can be made more complex by adding requirements, which may affect one or more model elements (as shown in Figure 1). For example, introducing obstacle avoidance affects *Spatial Constraints* and *Target Selection*, and may also affect the *Pathfinding* and *Motion* elements.

4.2.2. Military MANET. A more complex example is the military tactical MANET scenario, for which several requirements analyses have been published [Burbank et al. 2006; Perisa et al. 2007; Holliday 2008; Aschenbruck et al. 2008; Papageorgiou et al. 2009; Fongen et al. 2009]. The scenario used in this example is the second phase of a multinational peacekeeping mission to secure a chemical factory on hostile ground [Salmanian 2003]. A ship provides a stationary local command centre on a nearby body of water; unmanned aerial vehicles (UAVs) provide network connectivity; and troops move in armoured personnel vehicles (APVs), in tanks, and on foot. The region is mountainous with harsh conditions. The troops move toward the factory in APVs and tanks through a mountain pass. One tank remains in the pass to secure it and to provide connectivity to the command centre. On arrival, the troops exit the vehicles and secure the factory. We assume that this occurs as described in Aschenbruck et al. [2010a]: units are divided into small groups that enter the building consecutively, and each group secures a small part of the building in such a way that the fallback path is always secured.

In the following analysis, the required behaviour of each element of the model is derived from the scenario description; these are then assessed to identify some potential approaches:

—*Spatial Constraints*: The scenario is carried out in a defined region that has obstacles such as buildings and bodies of water, and regions that are more or less preferable based on terrain suitability and safety. Inside the factory, it has indoor obstacles such as walls. Graph-based methods with forbidden regions (Section 2.1.5) and bounded free space combined with graph constraints (Section 2.1.6) satisfy the requirements.

The graph can be designed with a higher density of vertices in preferred areas (Section 2.1.4). Free space methods with forbidden regions (Section 2.1.3) will prevent the selection of targets inside an obstacle but do not account for less preferable regions; this method would have to be paired with another method such as steering forces (Section 2.4.7).

—*Target Selection*: The strong planning aspects of military operations means that the model should allow a tactical level of target selection for nodes or groups of nodes, based on their role in the mission, its region of operations, and the terrain. For example, a UAV's role is to fly in a circular pattern above the region of operations, and the last tank to reach the mountain pass stays there for the remainder of the scenario. Deterministic target selection with checkpoints (Sections 2.2.8 and 2.2.9) allows for preplanned targets for each node. To introduce variations in movement patterns, location bias methods (Section 2.2.2) could be used to bias target selection to within the intended region of operations.

—*Pathfinding*: The Pathfinding method must avoid obstacles and choose a route that maximizes safety and ease of passage, and accommodate different behaviours for each node type (e.g., a vehicle will view a narrow path differently than ground troops). Graph traversal methods (Section 2.3.1) and biased random trips (Section 2.3.4) may provide the required behaviour. Stochastic graph traversal (Section 2.3.2) introduces variations in path selection.

—*Motion*: The Motion element must accommodate differences in movement patterns for different types of nodes and be able to accommodate differences in their intrinsic properties (e.g., average speed). The average speed should change in hostile environments. This requirement is satisfied by using regional speeds (Section 2.4.3) for free space environments and a graph edge speed (Section 2.4.4) for graph-based environments.

—*Pause*: The pause time after reaching a target should reflect the individual task of the node based on mission goals—hence, task-based pause times (Section 2.5.4) is an appropriate method. A PDF centred on the task-based pause time could be used to introduce randomness (Section 2.5.2).

—*Group Dynamics*: The hierarchical command structure dictates that group behaviour should reflect the concept of leaders and followers in a variety of group formations. This can be accomplished via target or velocity selection relative to a reference point (Section 2.6.1), or by applying internode forces (Section 2.6.3). Group dynamics must allow soldier nodes to enter and exit vehicle nodes, and behave appropriately in both cases. This can be achieved by moving the occupants of a vehicle relative to the vehicle's reference point (Section 2.6.1). The group dynamics inside the factory are dictated by the need to merge and divide (Section 2.6.2) in the process of securing the building.

Other requirements not explicit in the scenario description can be included. If the activities of adversarial forces are included, the model could incorporate dynamic changes to the environment, such as explosions that may make a route inaccessible, or adversary nodes that should be avoided. These may affect the *Target Selection*, *Pathfinding*, or *Motion* elements. The inclusion of battle casualties would result in the need to be capable of node removal, or of a change in node state to reflect the damage. If realistic motion is required, the model must provide smooth motion, prevent unintentional collisions, and disallow two nodes of the same type occupying the same physical space. *Motion* methods that satisfy these requirements are given in Sections 2.4.5, 2.4.6, and 2.4.7.

4.3. High-Level Survey

The taxonomy can also be used to provide a high-level view of the literature. Tables I through IV show the taxonomy applied to a selection of models, building on those

Table I. A Selection of Simple Models with Random or Location-Biased Target Selection and No Pathfinding or Group Dynamics Element

Each "X" denotes the model's category of methods; a "1" or "2" in the Target Selection element indicates a hierarchy of target selection methods.

	Spatial			Target						Path				Motion							Pause				Group		
Key	Bounded space	Forbidden areas	Graph constrained	Random	Location bias	Social network	State transitions	Checkpoints	Deterministic	Graph traversal	On-the-fly	Stochastic GT	Biased random trip	Predefined	Random speed	Regional speed	Edge speed	Collision avoidance	Smooth	Steering forces	Constant	Random	Regional	Task based	Reference	Merge/divide	Steering forces
RWP96	X			X											X						X						
Smooth01	X			X											X				X								
RBP02	X			X											X						X						
Manhat03			X	X											X			X	X								
Pixel04			X	X											X							X					
Pragma05	X				X										X							X					
HotWP06	X				X										X	X							X				
ModWP06	X			X											X	X						X					
Ripple07	X			X											X												
Subway07	X			X					X					X							X						
HeteroRW09	X			X											X	X											
Cluster10	X				X										X						X						
TLevyW11	X			X											X							X					
STEPS11	X			2	1										X												
SLAW12			X		X			X							X							X					
Affinity12	X				X									X													

previously classified in Aschenbruck et al. [2011]. The list of models is not exhaustive; specifically, it excludes simple variations of non–random waypoint (RWP) random trip models and models that did not specify all of the aspects of the taxonomy. Software and multimodel frameworks were excluded because the tabular presentation would be ineffective. Models are referred to in the tables by an identifying key with a two-digit suffix representing the year of publication; the key-reference mapping for the tables is given in Table V. The tables are sorted by year of publication, from which the seminal works become more evident. It is important to note, however, that the publication date is for the referenced cited, which is not necessarily the first appearance of the model, particularly when the citation is a journal article. In the tables, a model is said to be bounded if it states its dimensions or discusses boundary conditions or regions; otherwise, it is classified as free space.

Table I shows the RWP model and some of its variants. RWP variants with graph-based spatial constraints choose a random adjacent vertex, but otherwise they are bounded free space models that choose a random target, some with location bias. This table shows that variations of RWP continue to appear in the literature.

In Table II, models with random or location-biased Target Selection methods that use a graph traversal technique for Pathfinding are shown. Jardosh et al. [2005] (Obstacle05 in Table V) were the first to address obstacles directly by creating a graph from the obstacles. Tian et al. [2002] (Graph02) were the first to apply graph traversal with a random Target Selection method, and the method became quite popular in 2007. However, Table IV shows that Scourias and Kunz [1999] (Activity99) were the first to use graph traversal with state transition methods, making theirs the first

Table II. A Selection of Models with Random or Location-Biased Target Selection That Use Graph Traversal Techniques for Pathfinding and Do Not Model Group Dynamics Explicitly
Each "X" denotes the model's category of methods.

	Spatial			Target						Path				Motion							Pause				Group		
Key	Bounded space	Forbidden areas	Graph constrained	Random	Location bias	Social network	State transitions	Checkpoints	Deterministic	Graph traversal	On-the-fly	Stochastic GT	Biased random trip	Predefined	Random speed	Regional speed	Edge speed	Collision avoidance	Smooth	Steering forces	Constant	Random	Regional	Task based	Reference	Merge/divide	Steering forces
Graph02			X	X						X					X							X					
Obstacle05		X	X	X	X					X					X							X					
Statistical06			X	X							X				X						X		X				
GraphRW07			X	X							X				X							X					
GraphWP07			X	X						X					X							X					
OMBAA07		X	X		X					X					X							X					
Disaster07	X	X		X					X	X					X												
CORPS08		X			X					X					X												
Human09		X		X							X				X						X						
DiAm10			X	X						X					X		X					X					
RandStreet10			X	X						X					X							X					
Anchor10	X		X	X	X							X			X	X											
Sousa11			X	X	X						X			X			X										

graph traversal Pathfinding overall. There is an apparent lack of overlap between graph-based Spatial Constraints and Group Dynamics within the sample of models.

Table III shows a selection of models that use an explicit Group Dynamics method. Hong et al. [1999] (RPGM99) were the first to introduce group motion by using a reference point; similar models were developed later with slight variations. The first combining group methods with graph-based spatial constraints was Zhou et al. [2004] (VTrack04). Steering forces were first implemented to bring about group behaviours by Legendre et al. [2006] (Behavioural06).

Table IV shows a selection of models that use state transition methods or social interaction methods for the Target Selection element. In this sample of models, Scourias and Kunz [1999] (Activity99) were the earliest implementers of state transition methods. Although the notion was introduced by Herrmann [2003], the table shows that Musolesi and Mascolo [2007] (CommunityM07) published the first complete model in the mobile communications field to use social networks to choose a target region. Mei and Stefa [2009] (SmallWorld09) followed with a two-stage variation, similar to that of the Community model in Boldrini and Passarella [2010] (CommunityB10). This was augmented in the Home Cell model (HomeCell10) with state transitions and a location bias. Because social interaction methods implicitly lead to group-like behaviours, it is not surprising that social models do not employ explicit Group Dynamics methods; however, it is interesting to note the lack of overlap between models using state transitions and explicit Group Dynamics methods.

By considering the Motion element in all of the tables, it can be seen that Bai et al. [2003] were the first to introduce collision avoidance in this sample of models, and smoothing of motion was introduced in Bettstetter [2001]. The use of steering forces to control motion was first introduced in Legendre et al. [2006] and was combined with other methods to control motion in Williams and Huang [2009] and Liu et al. [2010].

Table III. A Selection of Models That Explicitly Incorporate Group Behaviours
Each "X" denotes a category of methods; a "1" or "2" in the Target Selection element indicates a hierarchy of target selection methods.

	Spatial			Target						Path				Motion							Pause				Group		
Key	Bounded space	Forbidden areas	Graph constrained	Random	Location bias	Social network	State transitions	Checkpoints	Deterministic	Graph traversal	On-the-fly	Stochastic GT	Biased random trip	Predefined	Random speed	Regional speed	Edge speed	Collision avoidance	Smooth	Steering forces	Constant	Random	Regional	Task based	Reference	Merge/divide	Steering forces
RPGM99	X							X	X					X											X		
Column01				X											X										X		
Structured04	X							X	X					X											X		
VTrack04	X		X	X									X		X										X	X	
RefRegion05	X			1					2				X	X							X[b]	X[b]		X	X	X	
Coalition05			X						X						X[a]										X		
Behavioural06	X								X											X							X
Diamond08	X			X											X										X		
GroupForce09				X											X					X					X		X
Skiing10	X			X										X						X							X
Tactical10		X	X						X						X	X	X					X			X	X	
SocComm11			X	X											X										X		
Mission12	X	X		X							X				X								X		X		

[a]Assumed, specifies an average speed but no distribution.
[b]Assumed, nodes "could stop for a pause time τ."

5. DISCUSSION

Mobility models are needed to test research ideas for mobile networks, but the choice of model can dramatically affect the test results. Simple mobility models are sufficient for initial testing of new ideas for protocols, routing algorithms, and security schemes but are insufficient for proving their viability in real-world conditions. Numerous models have been developed in recent years to address the need for realism, but the RWP model continues to dominate [Kurkowski et al. 2005]. Factors contributing to the small uptake of newer models may include a lack of understanding of the models themselves, or of which model is appropriate for their research. The taxonomy presented in this work can assist with these issues.

The adoption of a taxonomy that enables a systematic definition of mobility models will enhance communication by allowing authors and readers to think of their models in terms of their various components while keeping all of the pertinent information together. The literature contains some incomplete or disjointed model descriptions that may be impeding their uptake. In future literature, authors can use the taxonomy as a framework in which to completely and systematically describe their models. This, in addition to addressing the issues identified in Kurkowski et al. [2005], will increase reproducibility, which facilitates scientific progress. Moreover, to discuss the common characteristics of the models for the taxonomy, a consistent nomenclature had to be established. The use of common language will support clearer communications between researchers and allow similarities and differences between models to be more easily recognized. As the taxonomy matures, it will enable a more economical communication of ideas and concepts, relieving authors of the burden of having to explain established concepts and making novel contributions more evident.

Table IV. A Selection of Models That Utilize State Transitions and Social Interactions
Each "X" denotes the model's category of methods; a "1" or "2" in the Target Selection element indicates a hierarchy of target selection methods.

	Spatial			Target						Path				Motion							Pause				Group		
Key	Bounded space	Forbidden areas	Graph constrained	Random	Location bias	Social network	State transitions	Checkpoints	Deterministic	Graph traversal	On-the-fly	Stochastic GT	Biased random trip	Predefined	Random speed	Regional speed	Edge speed	Collision avoidance	Smooth	Steering forces	Constant	Random	Regional	Task based	Reference	Merge/divide	Steering forces
Activity99			X	2			1		2	X				X[b]							X	X					
Murray04	X			2			1						X		X[c]												
AreaGraph05	X		X				1		2						X							X		X			
WeightedWP05	X						X								X[c]												
Mold05	X						X							X								X	X				
WLAN05	X			2			1								X							X					
Stepanov05			X				X			X		X			X			X				X					
Kim06	X			2[a]			1						X		X								X				
CommunityM07	X					X									X												
Workday08			X	2			1			X					X							X					
RealMobGen08	X						X						X		X							X					
TVComm09	X			2			1								X							X					
SmallWorld09	X			2		1									X[d]												
Udel09			X	2			1								X			X				X					
Agenda10			X	2			1	X		X					X		X					X		X			
CommunityB10	X			2		1									X												
HomeCell10	X			2	1	1	1								X												
SMOOTH11				2		1	1		2						X							X					

[a]Assumed, but unspecified.
[b]Assumed, specifies a "system wide average speed."
[c]Assumed due to reference to RWP.
[d]Unit step size and random distance means that speed will necessarily be random.

A mobility model must reflect the variety of scenarios under which the technology will be applied. Many models exist in the literature, with significant repetition in each of the model elements, as shown in Tables I through IV. Rather than attempting to select a model in its totality, a researcher requiring a mobility model for testing purposes can select an appropriate method for each model element based on requirements derived from the use case scenario (cf. Section 4.2). If an appropriate method does not exist, the new method will be certain to contribute novel features to the research community. A guideline that maps requirements to existing methods for a variety of scenarios would be useful future work.

It has been proposed that the RWP model continues to be used due to its simplicity [Munjal et al. 2011]. For researchers who are not mobility model experts, validating their work depends on understanding the context of the simulation results, which makes complex models unappealing. It can also be difficult for the nonexpert to choose parameters for complex models. For example, in force-based models, the steering forces can be difficult to parameterize [Tripp et al. 2010]. Excess complexity can impede scientific progress in this respect. In their simulations, Helgason et al. [2010] showed that network measurements were relatively insensitive to changes in the distribution of node speeds, node arrival process parameters, node size, and minimum node

Table V. Mapping of Model Names to Reference

Key	Reference	Key	Reference
Activity99	[Scourias and Kunz 1999]	OMBAA07	[Babaei et al. 2007]
Affinity12	[Batabyal and Bhaumik 2012]	Pixel04	[Kraaier and Killat 2004]
Agenda10	[Zheng et al. 2010]	Pragma05	[Borrel et al. 2005]
Anchor10	[Ahmed et al. 2010]	RandStreet10	[Aschenbruck and Schwamborn 2010]
AreaGraph05	[Bittner et al. 2005]	RBP02	[Bettstetter and Wagner 2002]
Behavioural06	[Legendre et al. 2006]	RealMobGen08	[Walsh et al. 2008]
Clustered10	[Lim et al. 2010]	RefRegion05	[Ng and Zhang 2005]
Coalition07	[Reidt and Wolthusen 2007]	Ripple07	[Chen et al. 2007]
Column01	[Sánchez and Manzoni 2001]	RPGM99	[Hong et al. 1999]
CommunityB10	[Boldrini and Passarella 2010]	RWP96	[Johnson and Maltz 1996]
CommunityM07	[Musolesi and Mascolo 2007]	Skiing10	[Liu et al. 2010]
CORPS08	[Huang et al. 2008]	SLAW12	[Lee et al. 2012]
DiAm10	[Schwamborn et al. 2010]	SmallWorld09	[Mei and Stefa 2009]
Diamond08	[Ning et al. 2008]	Smooth01	[Bettstetter 2001]
Disaster07	[Aschenbruck et al. 2007]	SMOOTH11	[Munjal et al. 2011]
Graph02	[Tian et al. 2002]	SocComm11	[Kang et al. 2011]
GraphRW07	[Mogre et al. 2007]	Sousa11	[Sousa et al. 2011]
GraphWP07	[Mogre et al. 2007]	Statistical06	[Yoon et al. 2006]
GroupForce09	[Williams and Huang 2009]	Stepanov05	[Stepanov et al. 2005]
HeteroRW09	[Piórkowski et al. 2009]	STEPS11	[Nguyen et al. 2011]
HomeCell10	[Boldrini and Passarella 2010]	Structured04	[Blakely and Lowekamp 2004]
HotWP06	[Khadivi et al. 2006]	Subway07	[Toubiana et al. 2007]
Human09	[Papageorgiou et al. 2009]	Tactical10	[Aschenbruck et al. 2010a]
Kim06	[Kim et al. 2006]	TLevyW11	[Rhee et al. 2011]
Manhat03	[Bai et al. 2003]	TVComm09	[Hsu et al. 2009]
Mission12	[Papageorgiou et al. 2012]	Udel09	[Kim et al. 2009]
ModWP06	[Hyytia et al. 2006]	VTrack04	[Zhou et al. 2004]
Mold05	[Liang and Sheng 2005]	WeightedWP05	[Hsu et al. 2005]
Murray04	[Murray and Pesch 2004]	WLAN05	[Tuduce and Gross 2005]
Obstacle05	[Jardosh et al. 2005]	Workday08	[Ekman et al. 2008]

separation, whereas the scenario itself had greater impact. Further research into the level of realism required to simulate mobile networks in a variety of scenarios may help to identify unneeded complexity.

The RWP model may also continue to be used due to the lack of other suitable benchmarks for the field. A novel approach to the development of benchmarks involves populating a database with parameters used in the force-based Universal Mobility Modeling Framework (UMMF) [Medina et al. 2010], with the resultant network measurements (e.g., contact time, intercontact time) from the simulation [Tripp et al. 2010]. The user would then specify desired network measurements, and the database would return the UMMF parameters that would result in the desired network properties. Since UMMF has been shown to be capable of recreating other models, the approach is promising.

Techniques and models from other domains have been applied to the mobile networking domain, but there are more that can be considered. The social sciences have generated models of pedestrian dynamics, particularly in crowds [Pelechano et al. 2005; Bellomo and Dogbe 2011], and the tourism domain has generated models of both pedestrian and vehicular traffic (e.g., [Gloor et al. 2004; Steiner et al. 2007]. Pathfinding and collision avoidance are used in the robotics domain [Gelenbe et al. 1997]. Some of the methods described here are taken from simulation methods in the physical sciences;

although it is well established in microscopic simulation that Verlet integration is superior to Euler integration [Frenkel and Smit 2002], it has not been established at this scale whether the extra complexity is necessary.

Although only a portion of the literature on mobility models is presented in Tables I through IV, the analysis with respect to the taxonomy supports a better understanding of past work and the current state of the art, which may help guide the way ahead. As more literature is examined, it may be logical to extend the taxonomy to include new optional elements, such as *Initialization* and *Event Dynamics*. The application of standards such as a taxonomy and a common nomenclature will enhance understanding and encourage the increased adoption of more realistic models in the mobile communications field.

ACKNOWLEDGMENTS

The author would like to thank Dr. Daniel R. Matusek and the anonymous reviewers, whose advice and comments substantially improved this article.

REFERENCES

Sabbir Ahmed, Gour C. Karmakar, and Joarder Kamruzzaman. 2010. An environment-aware mobility model for wireless ad hoc network. *Computer Networks* 54, 9 (June 2010), 1470–1489.

Alex Aravind and Hassan Tahir. 2010. Towards modeling realistic mobility for performance evaluations in MANET. In *Proceedings of the 9th International Conference on Ad-Hoc, Mobile and Wireless Networks (ADHOC-NOW'10)*. Springer-Verlag, 109–122.

Alex A. Aravind and Xiang Cui. 2008. RMobiGen: A trace generation, visualization, and performance analysis tool for random mobility models. In *Proceedings of the 2008 Spring Simulation Multiconference (SpringSim'08)*. 403–412.

Nils Aschenbruck, Raphael Ernst, Elmar Gerhards-Padilla, and Matthias Schwamborn. 2010b. BonnMotion: A mobility scenario generation and analysis tool. In *Proceedings of the 7th International ICST Conference on Simulation Tools and Techniques (SIMUTools'10)*. 1–10.

Nils Aschenbruck, Raphael Ernst, and Peter Martini. 2010a. Indoor mobility modelling. In *Proceedings of the 2010 GLOBECOM Workshop on Mobile Computing and Emerging Communication Networks (GC Wkshps'10)*. IEEE, 1264–1269.

Nils Aschenbruck, Elmar Gerhards-Padilla, Michael Gerharz, Matthias Frank, and Peter Martini. 2007. Modelling mobility in disaster area scenarios. In *Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems*. ACM, 4–12.

Nils Aschenbruck, Elmar Gerhards-Padilla, and Peter Martini. 2008. A survey on mobility models for performance analysis in tactical mobile networks. *Journal of Telecommunications and Information Technology* 2, 54–61.

Nils Aschenbruck, Aarti Munjal, and Tracy Camp. 2011. Trace-based mobility modeling for multi-hop wireless networks. *Computer Communications* 34, 704–714.

Nils Aschenbruck and Matthias Schwamborn. 2010. Synthetic map-based mobility traces for the performance evaluation in opportunistic networks. In *Proceedings of the 2nd International Workshop on Mobile Opportunistic Networking*. ACM, 143–146.

Hamideh Babaei, Mahmood Fathy, and Morteza Romoozi. 2007. Obstacle mobility model based on activity area in ad hoc networks. In *Proceedings of the 2007 International Conference on Computational Science and Its Applications (ICCSA'07 Part II)*. Springer-Verlag Berlin, 804–817.

Fan Bai, Narayanan Sadagopan, and Ahmed Helmy. 2003. The IMPORTANT framework for analyzing the Impact of Mobility on Performance of RouTing protocols for Adhoc NeTworks. *AdHoc Networks Journal* 1, 383–403.

Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *Science* 286, 5439 (October 1999), 509–512.

Marc Barthélemy and Alessandro Flammini. 2008. Modeling urban street patterns. *Physical Review Letters* 100, 13 (April 2008), 138702.

Stefan Basagni, Imrich Chlamtac, and Violet R. Syrotiuk. 1999. Dynamic source routing for ad hoc networks using the global positioning system. In *Proceedings of the 1999 IEEE Wireless Communications and Networking Conference (WCNC'99)*. IEEE, 301–305.

Dharmayashdev Rai Basgeet, Pascal Dugenie, Alistair Munro, Dritan Kaleshi, and James Irvine. 2003. SMM: Mathematical framework of a scalable mobility model. In *Proceedings of the 6th ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWIM'03)*. ACM, 74–81.

Prithwish Basu, Alberto Medina, Oleksiy Pikalo, and Cesar Santivanez. 2008. Universal mobility modeling for MANETs. In *Proceedings of the 2008 Annual Conference of the International Technology Alliance (ACITA'08)*. Available at https://www.usukitacs.com/node/1487.

Suvadip Batabyal and Paramik Bhaumik. 2012. Improving network performance with affinity based mobility model in opportunistic network. In *Proceedings of the 2012 Wireless Telecommunications Symposium (WTS'12)*. IEEE, 1–7.

Wang Bei-Zhan, Wang Ya-Ping, Wang Wei, and Lou Run Yu. 2007. MANET loss tomography based on circle-movement mobile model. In *Proceedings of the 1st International Symposium on Data, Privacy, and E-Commerce (ISDPE'07)*. IEEE, 122–127.

Nicola Bellomo and Christian Dogbe. 2011. On the modeling of traffic and crowds: A survey of models, speculations, and perspectives. *SIAM Review* 53, 3 (August 2011), 409–463.

Christian Bettstetter. 2001. Mobility modeling in wireless networks: Categorization, smooth movement, and border effects. *Mobile Computing and Communications Review* 5, 3 (July 2001), 55–66.

Christian Bettstetter and Christian Wagner. 2002. The spatial node distribution of the random waypoint mobility model. In *Proceedings of Mobile Ad-Hoc Netzwerke, 1. deutscher Workshop über Mobile Ad-Hoc Netzwerke (WMAN'02)*. 41–58.

Shiddhartha Raj Bhandari, Gyu Moung Lee, and Noël Crespi. 2010. Mobility model for user's realistic behavior in mobile ad hoc network. In *Proceedings of the 8th Annual Conference on Communication Networks and Services Research (CSNR'10)*. IEEE, 102–107.

Agnieszka Bitner, Robert Hołyst, and Marcin Fiałkowski. 2009. From complex structures to complex processes: Percolation theory applied to the formation of a city. *Physical Review E* 80, 3 (September 2009), 037102.

Sven Bittner, Wolf-Ulrich Raffel, and Manuel Scholz. 2005. The area graph-based mobility model and its impact on data dissemination. In *Proceedings of the 3rd International Conference on Pervasive Computing and Communications Workshops (PerCom'05)*. IEEE, 268–272.

Ken Blakely and Bruce Lowekamp. 2004. A structured group mobility model for the simulation of mobile ad hoc networks. In *Proceedings of the 2nd International Workshop on Mobility Management and Wireless Access Protocols (MobiWac'04)*. ACM, 111–118.

Chiara Boldrini and Andrea Passarella. 2010. HCMM: Modelling spatial and temporal properties of human mobility driven by users' social relationships. *Computer Communications* 33, 9, 1056–1074.

Vincent Borrel, Marcelo Dias de Amorim, and Serge Fdida. 2005. A preferential attachment gathering mobility model. *IEEE Communications Letters* 9, 10 (October 2005), 900–902.

Vincent Borrel, Franck Legendre, Marcelo de Amorim, and Serge Fdida. 2009. SIMPS: Using sociology for personal mobility. *IEEE/ACM Transactions on Networking* 17, 3 (June 2009), 831–842.

Stefano Boschi, Miriam Di lanni, Pilu Crescenzi, Gianluca Rossi, and Paola Vocca. 2008. MOMOSE: A mobility model simulation environment for mobile wireless ad-hoc networks. In *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (Simutools'08). 1–10.*

Mat Buckland. 2005. *Programming Game AI by Example*. Jones & Bartlett Learning.

Jack L. Burbank, Philip F. Chimento, Brian K. Haberman, and William T. Kasch. 2006. Key challenges of military tactical networking and the elusive promise of MANET technology. *IEEE Communications Magazine* 44, 11 (November 2006), 39–45.

Tracy Camp, Jeff Boleng, and Vanessa Davies. 2002. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing* 2, 5 (August 2002), 483–502.

Chun-Hung Chen, Ho-Ting Wu, and Kai-Wei Ke. 2007. General ripple mobility model of speed-time pairing. In *Proceedings of the 2007 Wireless Communications and Networking Conference (WCNC'07)*. IEEE, 4057–4062.

Wenjie Chen, Lihua Dou, Yujin Li, and Yunchuan Wei. 2010. Ellipse group mobility model for tactical MANET. In *Proceedings of the 2010 Chinese Control and Decision Conference (CCDC'10)*. IEEE, 2288–2293.

Ching-Chuan Chiang. 1998. *Wireless Network Multicasting*. Ph.D. Dissertation. University of California, Los Angeles.

Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: User movement in location-based social networks. In *Proceedings of the 17th International Conference on Knowledge Discovery and Data Mining (KDD'11)*. ACM, 1082–1090.

Noun Choi, Alieza Mahdian, Ravi Prakash, S. Venkatesan, Neeraj Mittal, Albert J. Anderson, Eric Redding, and Robert Butler. 2007. A unified framework of node mobility models. In *Proceedings of the 2007 IEEE Military Communications Conference (MILCOM'07)*. IEEE, 1–7.

Vanessa Davies. 2000. *Evaluating Mobility Models within an Ad Hoc Network*. Master's Thesis. Colorado School of Mines, Golden, CO.

Zhenguo Du, Peilin Hong, and Kaiping Xue. 2012. Hotspot gravitation mobility model. *IEEE Communications Letters* 16, 2 (February 2012), 193–195.

Andrew M. Edwards, Richard A. Phillips, Nicholas W. Watkins, Mervyn P. Freeman, Eugene J. Murphy, Vsevolod Afanasyev, Sergey V. Buldyrev, Marcos G. E. da Luz, Ernesto P. Raposo, H. Eugene Stanley, and Gandhimohan M. Viswanathan. 2007. Revisiting Lévy flight search patterns of wandering albatrosses, bumblebees and deer. *Nature* 449, 7165 (October 2007), 1044–1048.

Frans Ekman, Ari Keränen, Jouni Karvo, and Jörg Ott. 2008. Working day movement model. In *Proceedings of the 1st ACM SIGMOBILE Workshop on Mobility Models (MobilityModels'08)*. ACM, 33–40.

Daniel Fischer, Klaus Herrmann, and Kurt Rothermel. 2010. GeSoMo: A general social mobility model for delay tolerant networks. In *Proceedings of the 7th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS'10)*. IEEE, 99–108.

Anders Fongen, Morten Gjellerud, and Eli Winjum. 2009. A military mobility model for MANET research. In *Proceedings of the 2009 IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN'09)*.

Daan Frenkel and Berend Smit. 2002. *Understanding Molecular Simulation: From Algorithms to Applications*. Academic Press.

Eroi Gelenbe, Nestor Schmajuk, John Staddon, and John Reif. 1997. Autonomous search by robots and animals: A survey. *Robotics and Autonomous Systems* 22, 1 (November 1997), 23–34.

Christian Gloor, Pascal Stucki, and Kai Nagel. 2004. Hybrid techniques for pedestrian simulations. In *Proceedings of the 4th Swiss Transport Research Conference (STRC'04)*. Available at http://www.strc.ch/conferences/2004.

Yanying Gu, R. Venkatesha Prasad, and Ignas Niemegeers. 2011. Mobility modeling for personal networks. *Wireless Personal Communications* 58, 2 (May 2011), 169–196.

Mesut Güneş, Martin Wenig, and Alexander Zimmermann. 2007. Improving MANET simulation results: Deploying realistic mobility and radio wave propagation models. In *Proceedings of the 12th IEEE Symposium on Computers and Communications (ISCC'07)*. IEEE, 39–44.

Hao Guo, Zhipeng Gao, Heng Zhang, Zhili Guan, Xingyu Chen, and Xuesong Qiu. 2010. A hotspot attraction driven user mobility model and direction deciding algorithm. In *Proceedings of the 2010 IEEE Symposium on Computers and Communications (ISCC'10)*. IEEE, 960–962.

Zygmunt J. Haas. 1997. A new routing protocol for the reconfigurable wireless networks. In *Proceedings of the 6th IEEE International Conference on Universal Personal Communications (ICUPC'97)*. IEEE, 562–566.

Dirk Helbing, Illes Farkas, and Tamas Vicsek. 2000. Simulating dynamical features of escape panic. *Nature* 407, 6803 (September 2000), 487–490.

Ólafur Helgason, Sylvia T. Kouyoumdjieva, and Gunnar Karlsson. 2010. Does mobility matter? In *Proceedings of the 7th International Conference on Wireless On-Demand Network Systems and Services (WONS'10)*. IEEE, 9–16.

Klaus Herrmann. 2003. Modeling the sociological aspects of mobility in ad hoc networks. In *Proceedings of the 6th ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWIM'03)*. ACM, 128–129.

Peter Holliday. 2008. SWARMM: A mobility modelling tool for tactical military networks. In *Proceedings of the 2008 IEEE Military Communications Conference (MILCOM'08)*. IEEE, 1–7.

Petter Holme and Beom Jun Kim. 2002. Growing scale-free networks with tunable clustering. *Physical Review E* 65, 2 (February 2002), 026107.

Xiaoyan Hong, Mario Gerla, Guangyu Pei, and Ching-Chuan Chiang. 1999. A group mobility model for ad hoc wireless networks. In *Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'99)*. ACM, 53–60.

Wei-jen Hsu, Kashyap Merchant, Haw-wei Shu, Chih-hsin Hsu, and Ahmed Helmy. 2005. Weighted waypoint mobility model and its impact on ad hoc networks. *Mobile Computing and Communications Review* 9, 1 (January 2005), 59–63.

Wei-Jen Hsu, Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Ahmed Helmy. 2009. Modeling spatial and temporal dependencies of user mobility in wireless mobile networks. *IEEE/ACM Transactions on Networking* 17, 5 (October 2009), 1564–1577.

Ying Huang, Wenbo He, Klara Nahrstedt, and Whay C. Lee. 2008. CORPS: Event-driven mobility model for first responders in incident scene. In *Proceedings of the 2008 IEEE Military Communications Conference (MILCOM'08)*. IEEE, 1–7.

Esa Hyytia, Pasi Lassila, and Jorma Virtamo. 2006. A Markovian waypoint mobility model with application to hotspot modeling. In *Proceedings of the 2006 IEEE International Conference on Communications (ICC'06)*. IEEE, 979–986.

Amit P. Jardosh, Elizabeth M. Belding-Royer, Kevin C. Almeroth, and Subhash Suri. 2005. Real-world environment models for mobile network evaluation. *IEEE Journal on Selected Areas in Communications* 23, 3 (March 2005), 622–632.

David B. Johnson and David A. Maltz. 1996. Dynamic source routing in ad hoc wireless networks. In Mobile Computing. The Kluwer International Series in Engineering and Computer Science. Vol. 353. Kluwer Academic Publishers, 153–181.

GuoDong Kang, Michel Diaz, Tanguy Perennou, Patrick Senac, and Jiadong Xu. 2011. Mobility model based on social community detection scheme. In *Proceedings of the 2011 Cross Strait Quad-Regional Radio Science and Wireless Technology Conference (CSQRWC'11)*. IEEE, 769–773.

Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. 2009. The ONE simulator for DTN protocol evaluation. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques (SIMUTools'09)*. 55:1–55:10.

Pejman Khadivi, Shadrokh Samavi, Hossein Saidi, and Terence D. Todd. 2006. Mobility model for hybrid WLAN/cellular systems. *Iranian Journal of Electrical and Computer Engineering* 5, 1 (Winter-Spring 2006), 11.

Jonghyun Kim, Vinay Sridhara, and Stephan Bohacek. 2009. Realistic mobility simulation of urban mesh networks. *Ad Hoc Networks* 7, 2 (March 2009), 411–430.

Minkyong Kim, David Kotz, and Songkuk Kim. 2006. Extracting a mobility model from real user traces. In *Proceedings of the 2006 IEEE Conference on Computer Communications (INFOCOM'06)*. IEEE, 1–13.

Jan Kraaier and Ulrich Killat. 2004. Calculating mobility parameters for a predefined stationary user distribution. In *Proceedings of the 12th IEEE International Conference on Networks (ICON'04)*. IEEE, 41–45.

Stuart Kurkowski, Tracy Camp, and Michael Colagrosso. 2005. MANET simulation studies: The incredibles. *Mobile Computing and Communications Review* 9, 4 (October 2005), 50–61.

H. Andrés Lagar-Cavilla, Gerard Baron, Thomas E. Hart, Lionel Litty, and Eyal de Lara. 2007. On the robustness of simple indoor MANET simulation models. *Ad Hoc and Sensor Wireless Networks* 4, 4 (2007), 321–354.

Jean-Yves Le Boudec and Milan Vojnović. 2006. The random trip model: Stability, stationary regime, and perfect simulation. *IEEE/ACM Transactions on Networking* 14, 6 (December 2006), 1153–1166.

Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. 2012. SLAW: Self-similar least-action human walk. *IEEE/ACM Transactions on Networking* 20, 2 (April 2012), 515–529.

Franck Legendre, Vincent Borrel, M. Dias de Amorim, and Serge Fdida. 2006. Reconsidering microscopic mobility modeling for self-organizing networks. *IEEE Network* 20, 6 (November 2006), 4–12.

Legion Software. 2013. Legion Studio. Retrieved June 2, 2014, from http://www.legion.com/legion-studio.

Shih-Tsung Liang and Wei-Hsueh Sheng. 2005. The mold mobility model for mobile wireless networks. In *Proceedings of the 1st International Conference on Wireless Internet*. IEEE, 79–84.

Sunho Lim, Chansu Yu, and Chita R. Das. 2010. A realistic mobility model for wireless networks of scale-free node connectivity. *International Journal of Mobile Communications* 8, 3, 351–369.

Xin Liu, Carey Williamson, and Jon Rokne. 2010. Physics-based modeling of skier mobility and avalanche rescue in mountainous terrain. In *Proceedings of the 35th IEEE Conference on Local Computer Networks (LCN'10)*. IEEE, 645–652.

Gang Lu, Gordon Manson, and Demetrios Belis. 2006. Mobility modeling in mobile ad hoc networks with environment-aware. *Journal of Networks* 1, 1, 54–63.

Alberto Medina, Gonca Gursun, Prithwish Basu, and Ibrahim Matta. 2010. On the universal generation of mobility models. In *Proceedings of the 18th Annual IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'10)*. 444–446.

Alessandro Mei and Julinda Stefa. 2009. SWIM: A simple model to generate small mobile worlds. In *Proceedings of the 2009 IEEE Conference on Computer Communications (INFOCOM'09)*. IEEE, 2106–2113.

Parag S. Mogre, Matthias Hollick, Nico d' Heureuse, Hans Werner Heckel, Tronje Krop, and Ralf Steinmetz. 2007. A graph-based simple mobility model. In *Proceedings of the 2007 ITG-GI Conference on Communication in Distributed Systems (KiVS'07)*. 1–12.

Seyed Morteza Mousavi, Hamid R. Rabiee, M. Moshref, and A. Dabirmoghaddam. 2007. Mobisim: A framework for simulation of mobility models in mobile ad-hoc networks. In *Proceedings of the 2007 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'07)*. IEEE, 82–82.

Aarti Munjal, Tracy Camp, and William C. Navidi. 2011. SMOOTH: A simple way to model human walks. *Mobile Computing and Communications Review* 14, 4 (October 2011), 34–36.

Ken Murray and Dirk Pesch. 2004. Mobility modelling using attracting points for cellular network environments. In *Proceedings of the 2004 Information Technology and Telecommunications Conference*. 87–93.

Mirco Musolesi and Cecilia Mascolo. 2007. Designing mobility models based on social network theory. *Mobile Computing and Communications Review* 11, 3 (July 2007), 59.

Mirco Musolesi and Cecilia Mascolo. 2009. Mobility models for systems evaluation. In *Middleware for Network Eccentric and Mobile Applications*. Springer, 43–62.

Majid Nabi, Marc Geilen, and Twan Basten. 2011. MoBAN: A configurable mobility model for wireless body area networks. In *Proceedings of the 4th International Conference on Simulation Tools and Techniques (SIMUTools'11)*. 168–177.

Samuel C. Nelson, Albert F. Harris III, and Robin Kravets. 2007. Event-driven, role-based mobility in disaster recovery networks. In *Proceedings of the 2nd ACM Workshop on Challenged Networks (CHANTS'07)*. ACM Press, New York, NY, 27–34.

Mark E. J. Newman. 2003. Mixing patterns in networks. *Physical Review E* 67, 2 (February 2003), 026126.

Mark E. J. Newman and Michelle Girvan. 2004. Finding and evaluating community structure in networks. *Physical Review E* 69, 2 (February 2004), 026113.

Jim M. Ng and Yan Zhang. 2005. Reference region group mobility model for ad hoc networks. In *Proceedings of the 2nd IFIP International Conference on Wireless and Optical Communications Networks (WOCN'05)*. IEEE, 290–294.

Anh Nguyen, Patrick Sénac, Victor Ramiro, and Michel Diaz. 2011. STEPS: An approach for human mobility modeling. In *Proceedings of the 10th International IFIP TC 6 Conference on Networking (NETWORKING'11)*. Springer-Verlag, Berlin, 254–265.

Li Ning, Guo Yan, Dong Chao, and Wang Jinlong. 2008. Diamond group mobility model for ad hoc network in military. In *Proceedings of the 9th International Conference on Signal Processing (ICSP'08)*. IEEE, 2754–2756.

NS-3 Consortium. 2012. NS-3. Retrieved June 2, 2014, from http://www.nsnam.org.

Bruno A. A. Nunes and Katia Obraczka. 2011. On the invariance of spatial node density for realistic mobility modeling. In *Proceedings of the 8th International Conference on Mobile Ad Hoc and Sensor Systems (MASS'11)*. IEEE, 322–331.

Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, and Sung Nok Chiu. 2000. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams* (2nd ed.). Wiley, New York, NY.

Christos Papageorgiou, Konstantinos Birkos, Tasos Dagiuklas, and Stavros Kotsopoulos. 2009. An obstacle-aware human mobility model for ad hoc networks. In *Proceedings of the 2009 IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'09)*. IEEE, 1–9.

Christos Papageorgiou, Konstantinos Birkos, Tasos Dagiuklas, and Stavros Kotsopoulos. 2012. Modeling human mobility in obstacle-constrained ad hoc networks. *Ad Hoc Networks* 10, 3 (May 2012), 421–434.

Karl Pearson. 1905. The problem of the random walk. *Nature* 72, 1865 (July 1905), 294.

Nuria Pelechano, Kevin O'Brien, Barry Silverman, and Norman Badler. 2005. Crowd simulation incorporating agent psychological models, roles and communication. In *Proceedings of the 1st International Workshop on Crowd Simulation (V-CROWDS'05)*. 21–30.

Ranjit Perera, Andreas Eisenblätter, E. R. Fledderus, Carmelita Görg, Michael Scheutzow, and S. C. J. Verwijmeren. 2002. *Pixel oriented mobility modelling for UMTS network simulations*. Technical Report IST-2000-28088. IST Mobile Summit. 828–831 pages.

Dennis Perisa, Alan Allwright, and Peyam Pourbeik. 2007. Structural dynamics of war game MANETs. In *Proceedings of the 2007 International Symposium on Communications and Information Technologies (ISCIT'07)*. 830–835.

Michal Piórkowski, Natasa Sarafijanovic-Djukic, and Matthias Grossglauser. 2009. A parsimonious model of mobile partitioned networks with clustering. In *Proceedings of the 1st International Communication Systems and Networks and Workshops (COMSNETS'09)*. IEEE, 1–10.

Steffen Reidt and Stephen D. Wolthusen. 2007. Efficient trust authority distribution in tactical MANET environments. In *Proceedings of the 2007 IEEE Military Communications Conference (MILCOM'07)*. IEEE, 1–7.

Craig W. Reynolds. 1999. Steering behaviors for autonomous characters. In *Game Developers Conference*. Miller Freeman Game Group, San Jose, CA, 763–782. Library available at http://opensteer.sourceforge.net.

Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seong Joon Kim, and Song Chong. 2011. On the Levy-walk nature of human mobility. *IEEE/ACM Transactions on Networking* 19, 3 (June 2011), 630–643.

Milan Rollo and Antonń Komenda. 2009. Mobility model for tactical networks. In *Proceedings of the 4th International Conference on Industrial Applications of Holonic and Multi-Agent Systems for Manufacturing (Lecture Notes in Computer Science)*, Vol. 5696. Springer, 254–265.

Kenneth H. Rosen. 1988. *Discrete Mathematics and Its Applications*. Random House.

Michele Rossi, Leonardo Badia, Nicola Bui, and Michele Zorzi. 2005. On group mobility patterns and their exploitation to logically aggregate terminals in wireless networks. In *Proceedings of the 62nd IEEE Vehicular Technology Conference (VTC'05-Fall)*, Vol. 3. IEEE, 1863–1867.

Elizabeth M. Royer, P. Michael Melliar-Smith, and Louise E. Moser. 2001. An analysis of the optimum node density for ad hoc mobile networks. In *Proceedings of the 2001 IEEE International Conference on Communications (ICC'01)*. IEEE, 857–861.

Izhak Rubin and Cheon Won Choi. 1997. Impact of the location area structure on the performance of signaling channels in wireless cellular networks. *IEEE Communications Magazine* 35, 2 (February 1997), 108–115.

Mazda Salmanian. 2003. *Military wireless network information operation scenarios*. Technical Report DRDC Ottawa TM-2003-241. Defence R&D Canada, Ottawa.

Miguel Sánchez and Pietro Manzoni. 2001. ANEJOS: A Java based simulator for ad hoc networks. *Future Generation Computer Systems* 17, 5 (March 2001), 573–583.

Scalable Network Technologies. 2013. QualNet. Retrieved June 2, 2014, from http://web.scalable-networks.com/content/qualnet.

Matthias Schwamborn, Nils Aschenbruck, and Peter Martini. 2010. A realistic trace-based mobility model for first responder scenarios. In *Proceedings of the 13th ACM International Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWIM'10)*. ACM, 266–274.

John Scourias and Thomas Kunz. 1999. An activity-based mobility model and location management simulation framework. In *Proceedings of the 2nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'99)*. ACM, 61–68.

Chaoming Song, Tal Koren, Pu Wang, and Albert-László Barabási. 2010. Modelling the scaling properties of human mobility. *Nature Physics* 6, 10 (September 2010), 818–823.

Ivo Sousa, Paula Queluz, António Rodrigues, and Pedro Vieira. 2011. Realistic mobility modeling of pedestrian traffic in wireless networks. In *Proceedings of the 2011 IEEE International Conference on Computer as a Tool (EUROCON'11)*. IEEE, 1–4.

Albert Steiner, Michel Philipp, and Alex Schmid. 2007. Parameter estimation for a pedestrian simulation model. In *Proceedings of the 2007 Swiss Transport Research Conference (STRC'07)*. Available at http://www.strc.ch/conferences/2007.

Illya Stepanov, Pedro Jose Marron, and Kurt Rothermel. 2005. Mobility modeling of outdoor scenarios for MANETs. In *Proceedings of the 38th Annual Symposium on Simulation (ANSS'05)*. IEEE, 312–322.

Emanuele Strano, Vincenzo Nicosia, Vito Latora, Sergio Porta, and Marc Barthélemy. 2012. Elementary processes governing the evolution of road networks. *Scientific Reports* 2 (March 2012), Article 296.

Jing Tian, Jorg Hahner, Christian Becker, Illya Stepanov, and Kurt Rothermel. 2002. Graph-based mobility model for mobile ad hoc network simulation. In *Proceedings of the 35th Annual Simulation Symposium (ANSS'02)*. IEEE, 337–344.

Riitta Toivonen, Jukka-Pekka Onnela, Jari Saramäki, Jörkki Hyvönen, and Kimmo Kaski. 2006. A model for social networks. *Physica A* 371, 2 (November 2006), 851–860.

Viswanath Tolety. 1999. *Load Reduction in Ad Hoc Networks Using Mobile Servers*. Master's Thesis. Colorado School of Mines, Golden, CO.

Vincent Toubiana, Houda Labiod, and Bennet Fischer. 2007. Event based mobility model for subway scenarios. In *Proceedings of the 3rd International Conference on Wireless and Mobile Computing, Networking and Communications (WiMOB'07)*. IEEE Computer Society, Washington, DC, 79.

Howard Tripp, Robert Hancock, Alberto Medina, and Prithwish Basu. 2010. Obtaining generic mobility models from network properties and trace data. In *Proceedings of the 2010 Annual Conference of the International Technology Alliance (ACITA'10)*. Retrieved June 2, 2014, from https://www.usukita.org/node/836.

Cristian Tuduce and Thomas Gross. 2005. A mobility model based on WLAN traces and its validation. In *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'05)*. IEEE, 664–674.

Eijiro Ueno, Akihito Hiromori, Hirozumi Yamaguchi, and Teruo Higashino. 2011. A simple mobility model realizing designated node distributions and natural node movement. In *Proceedings of the 8th International Conference on Mobile Ad Hoc and Sensor Systems (MASS'11)*. IEEE, 302–311.

Chris Walsh, Arta Doci, and Tracy Camp. 2008. A call to arms: It's time for REAL mobility models. *Mobile Computing and Communications Review* 12, 1 (January 2008), 34–36.

Jingxin Wang, Jian Yuan, Xiuming Shan, Zhenming Feng, Junwei Geng, and Ilsun You. 2011. SaMob: A social attributes based mobility model for ad hoc networks. In *Proceedings of the 5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS'11)*. IEEE, 444–449.

Karen H. Wang and Baochun Li. 2002. Group mobility and partition prediction in wireless ad-hoc networks. In *Proceedings of the 2002 IEEE International Conference on Communications (ICC'02)*. IEEE, 1017–1021.

Alicia Nicki Washington and Rotimi Iziduh. 2009. Modeling of military networks using group mobility models. In *Proceedings of the 2009 International Conference on Information Technology: New Generations (ITNG'09)*. IEEE, 1670–1671.

Duncan J. Watts. 1999. *Small Worlds: The Dynamics of Networks Between Order and Randomness*. Princeton University Press.

Sean A. Williams and Dijiang Huang. 2009. Group force mobility model and its obstacle avoidance capability. *Acta Astronautica* 65, 7–8 (October-November 2009), 949–957.

Di Wu, Juanjuan Li, and Jiangchuan Liu. 2011. A random obstacle-based mobility model for delay-tolerant networking. *International Journal of Network Management* 21, 4 (July 2011), 326–339.

Jungkeun Yoon, Mingyan Liu, and Brian Noble. 2003. Random waypoint considered harmful. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'03)*. IEEE, 1312–1321.

Jungkeun Yoon, Brian D. Noble, Mingyan Liu, and Minkyong Kim. 2006. Building realistic mobility models from coarse-grained traces. In *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*. ACM, 177–190.

Ming Zhao and Wenye Wang. 2009. A unified mobility model for analysis and simulation of mobile wireless networks. *Wireless Networks* 15, 3 (April 2009), 365–389.

Qunwei Zheng, Xiaoyan Hong, Jun Liu, and David Cordes. 2010. Agenda driven mobility modelling. *International Journal of Ad Hoc and Ubiquitous Computing* 5, 1 (December 2010), 22–36.

Biao Zhou, Kaixin Xu, and Mario Gerla. 2004. Group and swarm mobility models for ad hoc network scenarios using virtual tracks. In *Proceedings of the 2004 IEEE Military Communications Conference (MILCOM'04)*. IEEE, 289–294.

Received March 2013; revised October 2013 and February 2014; accepted April 2014