Modeling Mobility for Networked Mobile Cyber-Physical Systems

Mohannad A. Alharthi[*]
School of Computing
Queen's University
Kingston, ON, Canada
harthi@cs.queensu.ca

Abd-Elhamid M. Taha
Electrical Engineering
Alfaisal University
Riyadh, Saudi Arabia
ataha@alfaisal.edu

ABSTRACT

Simulation plays a considerable role in validating Cyber-Physical Systems (CPSs) as it substantially reduces the costs and risks in the design-testing cycles. Reliable simulations, however, mandate realistic modeling for both the cyber and the physical aspects. This is especially the case in various networked mobile CPSs (e.g., excavation robots and vehicular networks), where costs and risks may become substantial. Our interest in this work is to briefly survey how mobility is modeled in state-of-the-art network simulators. The survey considers representative models commonly used in the literature, their enhancements, and their persisting limitations. It also reports on some recent attempts to integrate network simulators with physical modeling environments.

1. INTRODUCTION

Credibility in network simulators (e.g., ns-3 [17] and OPNET [20]) owes much to their maturity in processing networks of various scales, structures, protocols and interfaces. Such simulators are commonly based on an event-driven structure, which lends itself to a reasonable set of optimizations for time and accuracy. Within the context of wireless networks, considerations need to be made for the specific aspects of the wireless communication medium and mobility — both of which are continuous-time phenomena, and which need careful modeling to ensure simulation reliability.

Earlier efforts on modeling mobility can be found in works analyzing the performance of cellular networks [10]. Such models, commonly characterized as being "macroscopic", were not concerned with the specific location of a mobile node, but rather the vicinity (specific cell or sector) within which the node lies. Models were hence reduced to a queuing-based view of the cellular network landscape where nodes either reside at a specific cell or move between cells at certain assumed rates, essentially creating a birth-death process. This level of detail sufficed the (capacity limited) nature of cellular networks at the time, when resource allocations were made either in time or frequency, and the specific effects of node location can be safely ignored.

Interest in "microscopic" mobility where the coordinate location of a node is computed began to rise toward the end of the 1990s, mainly due to a resurging interest in mobile ad hoc networks (MANETs). Communications in MANETs do not go through a central, controlling node such as an access point or base station, but are rather made directly between nodes. Evaluation of a MANET protocol thus heavily relied on knowing where a node is at, in turn resulting in the introduction of a plethora of mobility models. Among these models are the Random Waypoint and the Reference Point Group Mobility (RPGM) models [8] (explained in Section 2), which are popular in mainstream network simulators.

The role of microscopic mobility models became more important as the evolution of cellular networks led to a paradigm of "interference limited" allocations where network capacity became directly related to where the mobile node was at, and the data rate at which it is transmitting. This was more the case as cellular networks shifted from connection-oriented operation to a datagram-centric one, and as cell size (i.e., a cell's coverage area) was substantially reduced, especially in urban areas. Meanwhile, the introduction of location-based services meant that mobility modeling also mattered in simulated service evaluations.

Today, efforts continue to generate realistic mobility in modeling wirelessly networked cyber- physical systems (CPSs), where physical behavior is related to both the data exchanged and the nodes' surrounding physical context. Examples of such systems can be found in Vehicular Ad Hoc Networks (VANETs), where serious stakes depend on inter-vehicle communications and communications between vehicles and road-side infrastructures, significantly depending on vehicle location and speed. Similar emphasis can also be found in analyzing networked robotics and swarms. Increasingly, the importance of realistic mobility modeling has led to integrated simulation environments.

In this work we offer a survey on microscopic mobility models that have been used in network simulators. We aim at highlighting the advantages and disadvantages of the different models, especially the impact of the mobility model choice on the correctness of network simulation. We also review representative integrations targetting application domains such as VANETs and networked robotics.

---

[*] Mohannad A. Alharthi is also affiliated with King Abdulaziz University (KAU), Jeddah, Saudi Arabia.

CyPhy '14, April 14 - 17 2014, Berlin, Germany

http://dx.doi.org/10.1145/2593458.2593468.

2. OVERVIEW OF MOBILITY MODELS

A mobility model should be capable of generating various realistic mobility behaviors, including changes in speed and direction. Mobility models can be generally categorized into two categories: trace-based and synthetic. Trace-based models are essentially data collected from real networks used to dictate the movement of nodes in the simulation. Such models do offer realistic mobility, but do not allow for reactive simulations, and may be hard to attain. Synthetic mobility models, on the other hand, compute the mobility based on predefined settings, and respond to changes in the simulated environment.

Generally, synthetic models can be classified based on randomness in speeds and directions or destinations, temporal dependency of velocity, and whether geographic obstacles or restrictions exist. As well, some models also capture group mobility [2].

2.1 Random Entity Mobility Models

One class of mobility models is based on random movements of mobile nodes. In this class, each node is assigned a random speed and either a random direction or destination. Attributes assigned to one node are independent from those assigned to all other nodes in the simulation. Two popular random entity mobility models are the Random Waypoint and the Random Walk mobility models [2].

In the Random Waypoint model, random destinations in the simulation area are selected for each node. Each node then travels to the selected destination at a random speed selected uniformly from a predefined range of speeds. When a node reaches its destination, it pauses for a predefined period of time, then selects a new random destination. This process continues until the end of the simulation. The behavior of nodes using the Waypoint model can be tuned by adjusting parameters such as minimum and maximum speed, and pause time.

The Random Walk mobility model is a variant of the Random Waypoint model. In this model, each node selects a random direction from a predefined range of directions, instead of choosing a location in the simulation area. The node moves in the selected direction for a specified period of time or a specified distance, then it chooses a new direction without resting before traveling in the new direction. If a node reaches the boundary of the simulation area, the node is reflected back to the simulation area.

A particular phenomenon that was observed in the Random Waypoint is the high probability by which nodes converge to the center of the simulation area as the simulation progresses. This phenomenon is called “density waves”. A mobility model called the Random Direction [24] reduces these density waves by making nodes travel in the selected direction to a selected destination on the boundary. Instead of bouncing, nodes pause for a short period of time, then have a new direction chosen for them towards the center.

2.1.1 Handling Geographical Constraints

Some models take into account geographic restrictions and obstacles in order to simulate situations that take place in urban areas, or areas that involve physical obstacles in the environment. In such models, however, geographical constraints are handled by limiting node mobility to fixed paths – an assumption that can be justified given that much street or in-building mobility is path-constrained. The Pathway model [28], for example, is a graph-based model that mimics the map of a city. Vertices of the graph represent buildings while streets are represented by edges. Initially nodes are placed on random vertices. Nodes then move continuously to random destinations (buildings or vertices) using the shortest path available through the graph. The movement involves pauses for short intervals upon reaching a destination and before moving to a new random destination. Another example can be found in the City Section model [3, 25] (also called the Manhattan Grid Model), which limits node mobility to a rectangular grid that represents streets within an urban area. In this model, nodes move on the street according to predefined traffic laws such as speed limits. Similar to the Pathway model, nodes also move to random destinations using the shortest path available, and stop temporarily at destinations.

A model that captures changes in both mobility trajectory and the wireless communication channel is the Obstacle mobility model [9]. It does so, however, by employing several realistic movement patterns in the simulation area. The simulation field consists of a set of obstacles (buildings) and a pathway graph connecting them. Obstacles has an effect on trajectories of nodes as well as on the radio channel. Pathways between obstacles are computed prior to the simulation to allow movement between and through buildings. Additionally, radio transmission is blocked completely by obstacles. Once a destination (building) is selected for a node, a travel path is chosen along the graph edges according the shortest path possible. Upon reaching the destination, nodes pause for a period of time and select a new destination, performing a continuous process.

2.1.2 Models with Temporal Dependency

The aforementioned mobility models, especially random entity models, can be commonly found in mainstream simulators such as GloMoSim [6], OPNET [20], ns-2 [16] and ns-3 [17], or can be available through their extensions. The models, nevertheless, suffer from several drawbacks that affect the accuracy of the simulation. For instance, nodes in Random Waypoint and Random Direction undergo unrealistic movements such as taking sharp turns and sudden stops (i.e., without any deceleration) [1]. Memorylessness is also a common characteristic in such models, whereby no correlation or dependence is enforced between consecutively assigned speeds. Several models have been proposed to overcome these drawback, including the Gauss-Markov and Smooth Mobility models described below.

The Gauss-Markov model [14] introduces temporal dependency of nodes’ velocity by having the velocity of a node at a certain time period depending on the velocity at the previous time period [14]. The model further incorporates a tunable memory level parameter for adjusting the randomness of the model in order to be able to produce various simulation scenarios. The value of the parameter determines the degree of which the new velocity is dependent on the previous velocity or a randomly generated value.

The Smooth Mobility model [1] is another model that utilizes temporal dependency. The model targets the problem of sharp turns and sudden accelerations by introducing smooth increments of speeds and direction change. Both velocity of nodes and direction are affected by previously used velocities and directions. This smoothing more closely matches more realistic movement with respect to speed, ac-

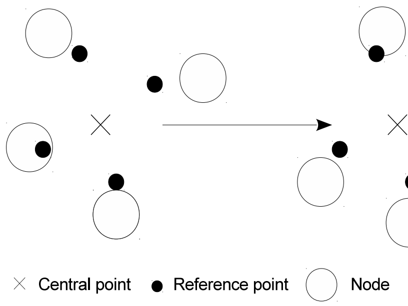

Figure 1: The movement of a group in RPGM

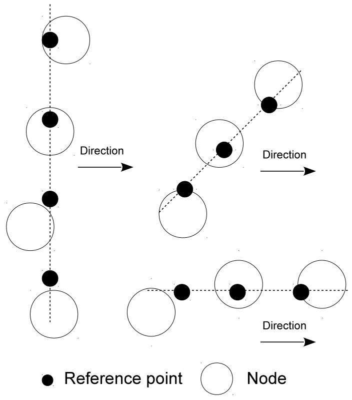

Figure 2: Possible configurations of the Column model

celeration and direction change. This behavior can be adjusted by modifying two parameters: the acceleration rate and the maximum allowed direction change in a period of time [1].

An important aspect of utilizing a mobility model for validation is how long it maintains specific evaluation criteria. Controlling average speed distribution or average longest distance covered becomes of particular importance in certain scenarios, e.g., where handover protocols are evaluated in cellular networks. An evaluation of steady-state characteristics of the Random Waypoint model was made in [1]. It was found that nodes in Random Waypoint have a tendency to cluster at the center of the simulation area as the simulation progresses. Suggestions were also made in [1] to alleviate the effects of this behavior.

2.2 Group Mobility Models

In the models described above, nodes move independently from each other and do not capture instances where a group of nodes can change their mobility, for example, following some leader node or in response to the movement of an external entity. To this end, group mobility models capture the behavior of nodes that work in some form of cooperation. Generally, a group model allows for two movements, a movement of the group collectively, and movements of individual nodes within the group. Models such as Random Walk are utilized for individual node movements. The Reference Point Group Mobility model (RPGM) is a model that works in this manner, which allows for various group mobility patterns including column, nomadic, and pursue mobility [8, 26]. Such patterns have direct applications in simulating scenarios involving rescue, sensing and tracking, and military operations.

In RPGM, the movement of the group is directed by a central point to the group, a "logical center" that determines the path the group takes, as well as the speed, acceleration, and direction. Moreover, each node is paired with a reference point that follows the movement of the center. Initially, nodes are uniformly distributed within a certain range around the central point. Nodes are also placed around their reference points. As the group advances towards the goal, the reference points advance, and each node follows its reference point and then moves around it according to a random entity model. Accordingly, the central point defines the trajectory of the group as a whole (either random or predefined path), while reference points define the movements of individual nodes around their reference points within the group. Figure 1 shows the placement of nodes around their reference points as they move from one location to another. This model is designed to produce various scenarios that involve multiple groups if provided proper combination of parameters [8].

In column mobility, nodes form a straight line, which can have any angle with respect to the direction in order to form a line of nodes either standing side by side facing the goal, or behind each other (like a waiting queue). Figure 2 shows possible configurations of this model. Each node in the group has a reference point where the node can move around it according to a random entity mobility model. Reference points advance towards the destination direction at each time step, which leads each node to update its location around its reference point. Meanwhile, in nomadic mobility nodes move together from one place to another as a single group, sharing the same reference point, with each node moving around the reference point as dictated by a random entity model. Finally, nodes in pursue mobility track a leader node but has their mobility dictated by a random entity model to the pursued node to maintain a degree of randomness.

3. IMPACT OF MOBILITY ON PERFORMANCE EVALUATIONS

Increasing interest in reliably simulating MANETs motivated the distinct efforts made in [23, 2]. The element goal of any network simulation is to test various aspects in an environment that, as much as possible, matches practical deployment and operation. In these two works, the choice of mobility model was shown to have a non-negligible effect on the results. Affected performance metrics included energy consumption, routing overhead, path length (hop count) traveled by the packets, end-to-end delay, and packet delivery rate. Intuitively, faster movements increase energy consumption — not only for energy expended in mobility, but also in packet re-transmissions. Indeed, a frequent change of network topology affects packet delivery due to broken links as a pair of nodes grow distantly apart. A node's surroundings (movement or communication obstacles), directly dependent on the node's location, also affect simulation results [23].

The work in [2] also examines the effect of mobility model choice. In [2], the Random Waypoint model is shown to result in a high packet delivery rate and low end-to- end delay and hop count due to the model's density wave phenomenon. Meanwhile, and for the same set of protocols evaluated, choosing the Random Direction model results in higher hop counts and end-to-end delays, but a low packet delivery rate. The difference in the results from those of the Random Waypoint was largely attributed to the border effect utilized in the Random Direction model, as this function eliminates the density wave phenomenon.

Other studies have been presented within the MANET contexts, but the two noted here suffice to highlight the serious consequences of selecting one mobility model over the other. These consequences are not alleviated by a passive conformity in selecting a specific model for most performance evaluations. At the same time, the models discussed thus far, including the smoothed models, still lack in representing a natural mobility environments — the need for which has increased substantially when considerations for aspects such as actuator qualities, terrain changes, impacts, and driving patterns have to be made.

The need for careful selection is not limited to mobility models. In many simulations for wireless mobile networks, accurate modeling needs to involve both mobility and the wireless channel. The effect of using a more accurate channel model on the formation of a group of networked autonomous robots was studied in [15]. The work evaluated the impact on packet drops and network topology in a field with obstacles. The scenario involved robots following a leader that has a predefined path, and where follower robots are expected to avoid obstacles in the field to maintain good communication channels. In such a setting, dropped packets directly influence the robots' movements made to better end-to-end communication. To demonstrate the effect of accurate channel modeling, the authors utilized Matlab for computing the channel conditions instead of utilizing the network simulator's own computation. Considerable variations in the results were shown to result from the choice of the channel model.

4. MODELING MOBILITY FOR VANETs AND NETWORKED ROBOTICS

Simulating networks of wireless networked CPSs requires careful attention, if only for the possibly sizable human and monetary assets at stake. In what follows, we focus on how mobility is modeled for two specific CPS applications, namely VANETs and networked robotics. In both these applications, the specific coordinate location of the CPS may have non-negligible consequences. In the example noted above, for example, the specific location of a robot directly affected communication quality. In the context of VANETs, correct location information becomes a crucial safety requirement [4].

4.1 Mobility for VANETs

In simulating VANETs, researchers have used real mobility traces since they are clearly more accurate than random models or other models such as the Manhattan grid model. Real traces can be hard to collect in order to be used in simulation; furthermore, it is not possible to simulate specific scenarios other than what is available in the traces, which makes this approach very limited. Recent applications of VANETs and Inter-Vehicle Communication (IVC) protocols require more detailed modeling at the microscopic level, describing various aspects of vehicles, traffic constraints, in addition to driver behavior models. Microscopic models emulate the behavior of individual vehicles independently. Additionally, driver-behavior is a challenging aspect since the impact involves individual vehicles (microscopic effect) as well as the high level flow of traffic, which affects mobility in general [4]. There are a number of simulators that have been developed for road traffic simulation such as VISSIM [29], and SUMO [11].

4.2 Mobility for Networked Robotics

Network simulators offer advanced and detailed models of communications as well as network protocols and routing. However, there is a need for models of physical robotics and their dynamics within the surrounding environment. This need is realized by existing robotic simulators, which model various components of robotics, including mechanical parts such as wheels and legs, engines, sensors, and actuators. Moreover, a detailed modeling of the surrounding environment is possible with such simulators, allowing for a realistic simulation of robotics, including robot mobility [27]. Additionally, robot simulators utilize controllers used in actual robots. A controller program can be used unchanged either in the simulation or in the real robot. This allows researchers to use the same controllers similarly. However, usually these simulators do not have accurate models of the network, if they have any network simulation capability, they have only simplistic models. ARGoS [21] and Player/Stage [22] are two examples of open source robotic simulators.

In order to utilize the mobility produced by those simulators, traces can be extracted from them, and then can be passed on to a dedicated network simulator. However, this process is not practical and is considered inadequate for evaluating dynamic control algorithms and mobility scenarios. In an attempt to fulfill this requirement, integrated simulation, also called co-simulation, were explored where a robotics simulator and a network simulator work jointly as a unified simulation environment.

5. CO-SIMULATION

Co-simulation was introduced to overcome the lack of complete realistic models of mobility in network simulators as well as the lack of detailed network models in modeling environments used to model physical systems, including robotic simulators. In such an integrated setting, the whole CPS simulator can provide realistic models for the physics aspects of the system, surrounding environment, control components, etc., while the network simulator provides realistic networking through network models for various protocol operations across the network stack.

Co-simulation environments work through interaction between heterogeneous models in both simulators. That is, typically the integration combines solver-dependent continuous-time models (e.g., Simulink, Modelica, or a robotic simulator) with discrete-event models represented by network models provided by ns-3. To give an example, consider performing a simulation of a set of networked robots. A robotic simulator can be used to utilize its robotic and controller models, and a network simulator can be used to simulate the network. To perform the simulation, this would require constructing models that represent the robots in the robotic simulator. In the network simulator, we would need to construct a set of network nodes for each robot in order to represent the network capability of robots. In this simulation environment, robots should be linked to their respective network nodes during the simulation. Robot models will perform a control algorithm that may include sending messages over the network between robots. This communication functionality are forwarded to the network simulator.

The network simulator can utilize mobility information (i.e., positions of nodes) from the robotic simulator and updates the mobility of nodes in the network simulator in order to perform the network simulation. The network simulator is informed about the messages exchanged between robots (e.g., as a result of the control algorithm being simulated) in order to simulate the networking activity. As nodes receive packets, respective robots in the robotic simulator are notified to consider messages received.

This form of dynamic interaction brings several challenges in developing and utilizing co-simulation environments. For instance, simulation time synchronization, and the execution mechanism required to enable the co-simulation are key aspects. It is important to note that the evaluated network performance might be affected by this synchronization process depending on the execution mechanism, since the two simulators alternate execution, and exchange information at fixed time steps.

5.1 Co-simulation Tools

A number of integrated simulation tools have been recently developed for VANETs and networked robotics. The following are representative examples.

- ARGoS-NS3 [12]: A simulator for multi-robot systems developed by IDalle Molle Institute of Artificial Intelligence (IDSIA). The goal is to make the simulation of multi-robotic systems more realistic by adding the communication aspect of the simulation, which is accomplished by incorporating a network simulator. This framework provides an interface between ARGoS, a robotic simulator, and ns-2 or ns-3.
- BARAKA [13]: A simulator that targets Sensor and Actor Networks (SANETs). It integrates physical aspects of the simulation with the networking aspects. OMNeT++ [19] is used along with the Mobility Framework for OMNeT++ for network simulation. The Open Dynamics Engine (ODE) [18], an open source library for simulating rigid body dynamics, is used for simulating physical robots. Robot controllers are written as application layer programs. BARAKA combines ODE with OMNeT++ for making an integrated simulator for understanding the effects that physics and communication have on each other.
- iTETRIS [7]: An integrated simulation platform for large-scale cooperative road traffic management. iTETRIS integrates SUMO (Simulation or Urban Mobility) and ns-3. It provides a realistic environment based on an open source architecture that links the traffic simulator with the network simulator in real time, allowing dynamic control of traffic based on the urban mobility provided by SUMO and the wireless network simulated in ns-3.

5.2 Discussion

Performance of co-simulator environments becomes a crucial concern because of the nature of integrated simulation. The performance of a co-simulation environment depends on several factors, such as the size/scale of the simulation, including objects being simulated (network nodes, robots or vehicles), the level of detail in physical and networking models and the individual simulators used. However, it is expected that co-simulation will be demanding for processing power. If we consider network simulation alone, a simulator such as ns-3 shows competent performance compared to other simulators as shown in [12] which offers a performance evaluation of ARGoS-NS3. Therefore, choosing efficient simulators is likely to make the integrated simulator work better.

An advantage of this integration is more realistic modeling of mobility, as the produced mobility behavior is the result of an actual physical simulation, not a result of synthesized or randomly generated movements, leading to more reliable network simulation dependent on mobility behavior. In addition, general physical modeling can be utilized in some networked physical processes where mobility is not necessarily part of the simulated experiment.

Modeling CPSs introduces several challenges due to the heterogeneity of CPSs. As several engineering and science domains are involved, the modeling and design process requires the cooperation of several experts from each domain involved. This might also require using several modeling and development tools. Another challenge is the actual modeling of the system. CPSs naturally are composed of physical phenomena, computations and communications which are modeled as a blend of continuous-time and discrete-event models. Physical continuous-time models may be modeled using differential equations which rely on specialized solvers usually available in physical modeling tools. In the context of networked CPSs, combined network/physics-based simulation tools can greatly benefit researchers and designers of networked CPSs, by integrating the modeling capabilities of physical modeling tools with network simulators.

In addition to the efforts noted above, other efforts are currently underway to formalize how the operation of such integrated environments, can be enhanced by design, e.g., [5].

6. CONCLUSION

Our aim in this work was to instigate a discussion on how mobility is modeled in network simulations, especially when mobile cyber-physical systems are involved. While network simulators have reached considerable maturity on the “cyber” side (computations, networking, etc.), efforts persist in generating more reliable physical models. For mobility, the efforts ranged from smoothing generated node mobility to emphasize natural dependences in time and space, to deriving models from urban and highway vehicular traces, to integrating network simulators with physical modeling environments. It is our understanding that the topic merits further exploration, especially in terms of validation of mobility models, and increasing the accessibility to scalable realistic simulations.

7. REFERENCES

[1] C. Bettstetter. Smooth is better than sharp: a random mobility model for simulation of wireless networks. In *Proceedings of the 4th ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, MSWIM '01, pages 19–27, New York, NY, USA, 2001. ACM.

[2] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5):483–502, 2002.

[3] V. A. Davies. Evaluating mobility models within an ad hoc network. Master's thesis, Colorado School of Mines, 2000.

[4] F. Dressler, C. Sommer, D. Eckhoff, and O. Tonguz. Toward realistic simulation of intervehicle communication. *IEEE Vehicular Technology Magazine*, 6(3):43 –51, sept. 2011.

[5] E. Eyisi, J. Bai, D. Riley, J. Weng, W. Yan, Y. Xue, X. Koutsoukos, and J. Sztipanovits. NCSWT: An integrated modeling and simulation tool for networked control systems. *Simulation Modelling Practice and Theory*, 27(0):90 – 111, 2012.

[6] GloMoSim. http://pcl.cs.ucla.edu/projects/glomosim/.

[7] J. Gozálvez, S. Turksma, L. Lan, O. Lazaro, F. Cartolano, E. Robert, D. Krajzewicz, R. Bauza, F. Filali, M. Röckl, et al. iTETRIS: the framework for large-scale research on the impact of cooperative wireless vehicular communications systems in traffic efficiency. In *Information and Communications Technologies (ICT-MobileSubmit 2009)*, 2009.

[8] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang. 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, pages 53–60, New York, NY, USA, 1999. ACM.

[9] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri. Towards realistic mobility models for mobile ad hoc networks. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, MobiCom '03, pages 217–229, New York, NY, USA, 2003. ACM.

[10] J. Jobin, M. Faloutsos, S. Tripathi, and S. Krishnamurthy. Understanding the effects of hotspots in wireless cellular networks. In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 4 vol. (xxxv+2866), march 2004.

[11] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner. Sumo (simulation of urban mobility). In *Proceedings of the 4th Middle East Symposium on Simulation and Modelling*, pages 183–187, 2002.

[12] M. Kudelski, M. Cinus, L. Gambardella, and G. A. Di Caro. A framework for realistic simulation of networked multi-robot systems. Technical report, IDSIA / USI-SUPSI. Dalle Molle Institute for Artificial Intelligence, Galleria 2, 6928 Manno, Switzerland, Apr. 2012.

[13] T. Labella, I. Dietrich, and F. Dressler. Hybrid simulation of sensor and actor networks with baraka. *Wireless Networks*, 16:1525–1539, 2010.

[14] B. Liang and Z. Haas. Predictive distance-based mobility management for pcs networks. In *Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1377 –1384 vol.3, mar 1999.

[15] A. Muppalla and B. S. Manoj. The impact of communication and terrain characteristics on the accuracy of robot formation. In *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*, ACWR '11, pages 497–502, New York, NY, USA, 2011. ACM.

[16] The network simulator - ns-2. http://www.isi.edu/nsnam/ns/.

[17] ns-3. http://www.nsnam.org/.

[18] Open dynamics engine. http://www.ode.org/.

[19] OMNeT++. http://www.omnetpp.org/.

[20] OPNET modeler. http://www.opnet.com/solutions/network_rd/modeler.html.

[21] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle, T. Stirling, A. Gutierrez, L. M. Gambardella, and M. Dorigo. Argos: A modular, multi-engine simulator for heterogeneous swarm robotics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5027 –5034, sept. 2011.

[22] The player project. http://playerstage.sourceforge.net.

[23] P. Prabhakaran and R. Sankar. Impact of realistic mobility models on wireless networks performance. In *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pages 329 –334, june 2006.

[24] E. Royer, P. Melliar-Smith, and L. Moser. An analysis of the optimum node density for ad hoc mobile networks. In *IEEE International Conference on Communications*, volume 3, pages 857 –861 vol.3, 2001.

[25] A. K. Saha and D. B. Johnson. Modeling mobility for vehicular ad-hoc networks. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, VANET '04, pages 91–92, New York, NY, USA, 2004. ACM.

[26] M. Sánchez and P. Manzoni. Anejos: a java based simulator for ad hoc networks. *Future Generation Computer Systems*, 17(5):573 – 583, 2001. I: Best of Websim99. II: Traffic Simulation.

[27] G. S. Sukhatme, S. Brizius, and G. A. Bekey. Evaluating the mobility of a wheeled robot using dynamic modeling *. *Advanced Robotics*, 12(5):579–591, 1997.

[28] J. Tian, J. Hahner, C. Becker, I. Stepanov, and K. Rothermel. Graph-based mobility model for mobile ad hoc network simulation. In *35th Annual Simulation Symposium*, pages 337 – 344, april 2002.

[29] PTV Vissim. http://vision-traffic.ptvgroup.com/en-us/products/ptv-vissim/.