On the Impact of Realism of Mobility Models for Wireless Networks

Hector Flores†, Rudolf Riedi†‡¶
†Dept of Statistics, ‡Dept of ECE Rice University, Houston, TX
¶Dept of Telecommunications, EIF-FR, Fribourg, Switzerland
and Fernfachhochschule Schweiz, Brig, Switzerland
Email: hflores@rice.edu, riedi@rice.edu

Stephan Eidenbenz§, Nick Hengartner§
§CCS-3
Los Alamos National Laboratory, Los Alamos, NM
Email: eidenben@lanl.gov, nickh@lanl.gov

***Abstract*—We present PedSims, a suite of mobility models spanning the entire range from the classic Random Waypoint (RWP), to correlated movement, to trace-level-quality social-activity-based mobility. Instrumenting PedSims at various levels of structural complexity we assess and quantify the impact of real world mobility patterns, such as common interests or synchronous behavior, on ad hoc network performance. Thus, PedSims provides not only a flexible platform for mobility simulation but also a means to strike a trade-off between a tolerable error and computational complexity. Notably, as our study of DSR, AODV and SAFARI [19] finds, neglecting to capture mobility patterns may result in inversion of the ranking of routing protocols as well as in over- and under-estimation of network performance. Further contributions include a bias correction for the sampling of node-node connection time.**

I. Introduction

In theoretical applications of Mobile Adhoc Networks (MANETs), it is becoming increasingly clear that realistic mobility models need to be deployed to increase the validity of claims made regarding all other network properties, such as routing efficiency or scalability [7]–[9], [15]. However, the building of high fidelity mobility models can be a task unto itself. From a modeling perspective, it is be desirable to balance the competing aims of realistic mobility behavior and simplicity of implementation and computation. The ideal compromise should be as simple as possible while preserving the quantitative results from wireless communication simulations and applications. On one end of the spectrum we have Random Walk, Random Waypoint, Random Direction, Gauss-Markov, Boundless Random, etc. [10], [14]. Their beauty is in their simplicity of implementation, but they do not compare well to real movement in a populated city. On the more complex end there are models like TRANSIMS [8], [9] that simulate automobile movement metropolitan areas. Their approach to mobility modeling rests on modeling the behavior of virtual individuals executing realistic day-plans consisting of performing activities at various locations and moving between these locations on a virtual road network in simulated time. The amount of time to process a single TRANSIMS mobility simulation for a medium sized city such as Portland is on the order of days or longer, depending on how much preprocessing has already been done. Clearly this is too unwieldy for most mobile networking experiment frameworks.

A compromise must be made between realism and simplicity of implementation in a way that impacts as little as possible the simulation results for a desired wireless application. To study this trade-off, we propose *PedSims*, a flexible mobility simulation platform. We implement a hierarchy of pedestrian mobility models with increasing complexity within PedSims and study topological properties of the resulting ad hoc networks via several metrics. We continue to compare three ad hoc routing protocols, i.e., DSR [11], AODV [20] and Safari [19], in order to assess the relevance of certain features of faithfulness in mobility simulation on network performance estimation as well as on protocol ranking. With this study we hope to provide insight into how to strike an ideal tradeoff between margin of errors and computational complexity.

II. Related Work

In recent years, a great interest in mobility models has been demonstrated by a large number of proposals, such as [1], [2]. For a general overview we refer the reader to excellent surveys [3], [4]. The idea to modify the well-known Random WayPoint (RWP) mobility model is not new. A modification, proposed by Feeley et. al. [5], proposes adding a set of fixed, finite attractor points as the set of targets for RWP which is similar to a variation we evaluated. Further, there exist a variety of non-Markovian mobility models that attempt to mimic social structures (group and cellular models as described in [4].

As a particular implementation of PedSims we have chosen, by convenience, the campus and student population of Rice University. Campus-specific models have been proposed before. The design of PedSims lends much of its structure from TRANSIMS, a traffic simulator developed at LANL. However, no software was reused in generating PedSims, except for the UPMOST Visualizer from TRANSIMS' parent project (see [7]–[9]).

III. PedSims: Pedestrian Simulator

The structure of PedSims consists, similar to mobility simulators traditionally used in the ad hoc network community, of three elements: a population of objects, a space within which the objects move, and a set of rules determining how objects move. The most crucial novelty of PedSims resides in a set of mobility rules which are sensitive to time and reflect a *heterogeneity* among the mobile objects. This structure allows for *emerging* correlation of mobility such as clustering and co-motion since several nodes may choose to visit the same location around the same time. However, the structure underlying PedSims does not include correlated decisions, i.e., coordination among nodes.

A. PedSims Population, Space and Interaction

We elaborate on the three elements of PedSims.

First, the **population** of PedSims consists of mobile objects or nodes, called *mobiles*, which are endowed with a set of *attributes*. These attributes allow PedSims to faithfully represent an heterogeneous population where members exhibit different mobility patterns. We may think of attributes as indicating interests and preferences, or purposes and duties.

Second, the **space** is assumed to be a connected subset of an Euclidean space of two or three dimensions. However, more general settings can be accommodated by changing the format of the output PedSims produces. Given in this space is a set of locations called *destinations*, as well as a set of *paths* on which the mobiles move from destination to destination. Finally, a set of *walking rules* provides a unique, consistent way of how to get from any given location on a path to any destination. By default, walking rules implement shortest paths.

Third, the so-called **interaction model** allows to randomly choose a sequence of destinations that a mobile will visit together with the times at which it will depart from the destinations and move to the next. At expiration of the current elapse time, the interaction model takes as input the attributes of a mobile, its location and the current time; it returns a *weighted list* where each entry consists of three items: (i) a possible next destination for the mobile, (ii) a total elapse time and (iii) a weight. The weight is used to randomly choose the next destination of the mobile from that list, where higher weights correspond to higher chances of being selected.

The full mobility trace of a mobile is obtained by moving the mobile to the randomly chosen destination, having it wait there until the corresponding elapse time passed, to draw a new destination and elapse time, and to repeat until the simulation ends or the mobile leaves the simulated world.

B. Instrumenting Pedsims: Levels of Complexity

The interaction model is clearly the core structure of PedSims and its most distinguishing feature. It also harbors the true complexity of the model. Most notable is the possibility of incorporating an explicit time dependence of decisions made. On a first level of time-complexity, the interaction model simply provides a weighted list of destinations with elapse time depending only on the attributes, but not on the decision time. This corresponds to a simple model of visitors to a **theme park**. That is, a person visiting a theme park full of attractions (destinations) will only ever choose an attraction based on personal preferences (attributes). On a second level of time-complexity, the weighted list depends on time, modeling a theme park where attractions or interests —such as hunger— change according to the time of day, but where visitors don't plan ahead.

On a third level of time-complexity we may model a theme park with visitors that are planning ahead for certain **appointments**, i.e. destinations and arrival times not to be missed. To this end, the interaction model enforces restrictions which depend not only on current location but also current time and which constrain the mobile from wandering off too far. This is achieved by assigning zero weight to destinations and/or elapse times which are in conflict with the appointment. In a most simple case, where mobiles may move along any straight line in one given velocity, zero weight is assigned to all destinations outside ellipses formed by current location and future appointment destinations, with semi-axis shrinking appropriately as the appointment times draw closer.

C. Implementing Random WayPoint and Variants in Pedsims

Successively instrumenting PedSims' structure we suggest the following variants of RWP which we study in the sequel.
RWP [11] Here, the population consists of a number of mobiles with identical attributes, or none at all. The space of destinations forms a rectangle in the plane, or a cube in space, with straight paths connecting them. The interaction model says, that independently of attributes, current time and location, a new destination is chosen randomly with uniform distribution, a random velocity at which to move to the destination, as well as a random waiting time at the destination which can be computed as an elapse time minus the travel time.
RWP(STT): Space restrictions Restricting in the above RWP the set of destinations to a finite set in the plane we arrive at a variant of Pedsims termed RWP(STT) (straight-to-target). Notably, the constraint implies that the mobiles are now restricted to a finite set of paths which has the potential to effect length and robustness of multi-hop communication paths.
RWP(CS): Time constraint Enforcing certain decision times in RWP(SST) through the interaction model nodes will now depart from their chosen path or destination at synchronized times, adding temporal correlation to the already existing albeit small spatial correlation in the mobility.

D. RiceSims: Simulating real-world mobility in PedSims

Aiming at a faithful simulation of a real world population we chose to implement student life at Rice University.

To this end, the population was set to be a student body of 300 with their defining attributes being gender, age, major, whether in residence or not, walking slow or fast, and taking many or few classes corresponding to averages over the last few years as available. The spatial details, i.e., the destinations and paths were obtained from a GIS accurate map of actual buildings and walk ways on Rice campus.

Two variants were implemented: **RiceSims** uses the full spatial information as discussed above, with paths computed using Dijkstra's shortest path algorithm. **Ricesims(STT)** replaces actual paths by straight lines between destinations. This reduces correlation in mobility somewhat, as mobiles have to share paths less often, yet it distributes them better over space; thus, wireless connectivity could potentially improve but multi-hop paths could be of shorter duration.

The *interaction model of RiceSims* is structured as a theme park with planning ahead:

- Appointments of mobiles correspond to realistic and consistent class times, chosen randomly according to the attribute "major". Classes start at full hours and last 50 minutes, followed by 10 minutes break.

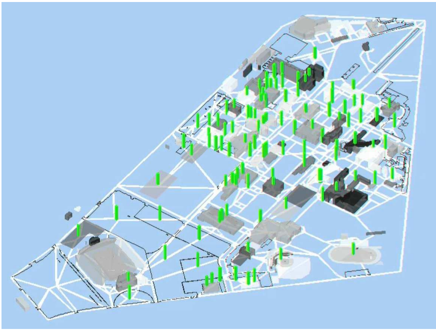

Fig. 1. Ricesims Visualization 3D

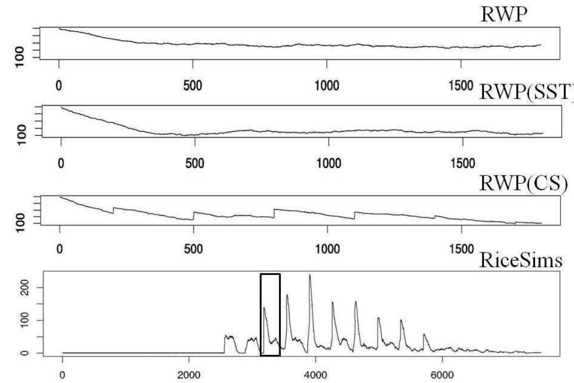

Fig. 2. Number of People Moving. The RiceSims trace is in 10's of secs and corresponds to an entire day. For the analysis, only 1800 secs (from 3150 to 3330) were used. The RWP versions quickly settle into a stationary regime (CS with periodic component). We show only 1800 secs simulation time; otherwise no details would be visible.

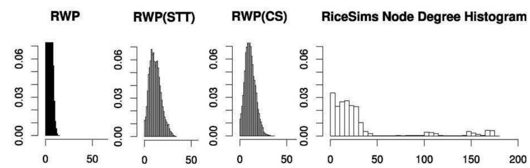

Fig. 3. Node Degree Distributions

- Entry point and time constitute the mobile's first destination and elapse time, chosen randomly among the student parking lots and housings according to attribute, and consistent with the first appointment.
- Additional destinations (besides appointments) consist of "Campus Exit Points", "Office", "Gym", "Soccer field", "Dining Halls", and "Dormitory" with weights according to attributes. Only destination and elapse times consistent with the next appointment are allowed. Elapse times are chosen uniformly from the interval of feasible times.
- End of Schedule: Once a mobile leaves campus by selecting an "Exit Point", it will move from there only if it has another appointment to keep.

E. Visualization/Observations

We utilized the Upmost Visualizer (designed for the Upmost project of which Transims is a subset) [9] to generate an animation of the simulation, both as a verification of our process and for first observations (see Figure 1).

IV. Mobility Experiments

This section discusses several metrics of multi-hop wireless networks implied by our mobility models, across time and across the various mobility models. In doing so, we adapt Principle Component Analysis for the temporal study of metrics and implement a bias-correction necessary for estimating connectivity duration.

A. Hierarchy of Mobility Models

The various mobility models in our hierarchy (see IV-A):

- RWP: Classical Random Waypoint
- RWP(STT): RWP (Straight To finite many Targets)
- RWP(CS): RWP(STT) with synchronous decision times
- Ricesims(STT): Students with realistic classes and recreational activities moving on straight lines
- Ricesims: Students moving along shortest campus paths.

Each model was inspected using via node spatial distribution and real-time animation privided by iNSpect [8]. For consistency, we generated each model with 300 nodes, moving at speeds between 2-4 miles per hour. For the RWP variants, a fixed wait time of 300 seconds was specified. The synchronous times in RWP(CS) were set at every 1/6 interval of simulation time. The total simulation time was 1800 seconds.

B. Number of People Moving

The simplest measure of network instability counts the number of individuals that move at any given time, an important parameter for ad hoc routing protocols. Figure 2 shows bursts of activities for Ricesims and RWP(CS) by design while mobility is stationary for RWP and RWP(STT).

It has been argued that reaching a steady state (in terms of node speed, spatial distribution, etc.) is a desireable quality in a mobility model [2], but for an application with correlated movement, the number of people active is bursty and non-stationary, raising several interesting new questions about the appropriate use of steady state calculations.

C. Node Degree Distribution

At least two ways to study node degree present themselves. First, histograms over network and time summarize the overall node degree (Fig. 3). Second, insight on the temporal aspect is gained via an average node degree as a function of time (Fig. 4). The conclusions from these metrics are partly to be expected but not trivial. For RWP and RWP(SST), the plots confirms a stationary behavior. They also impressively demonstrate the disruptive impact of bursts of mobility on topology and, indirectly, on connectivity. Notably and somewhat surprisingly, the synchronized timing of mobility alone achieves hardly any impact, as the RWP(CS) model shows. This suggests that correlated mobility and clustering of nodes has a larger impact on topology in terms of node degree. Indeed, in RWP(STT) changes of direction and speed

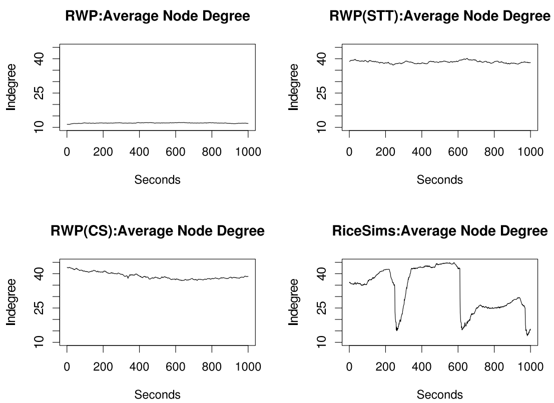

Fig. 4. Average Node Degree Time Series

are made simultaneously across a percentage of nodes but there is no guarantee that spatial clustering occurs.

With Ricesims, the relatively small number of appointment locations (class rooms) insure that mobiles come and remain together (high node degree) with sharp changes during breaks. Notably, the average degree is lower around 30 than the fully connected models RWP(STT) and RWP(CS) around 40.

A third alternative way to summarize the node degree consists in treating each time series as a single multivariate observation, i.e., a vector, and to apply principle component analysis (PCA). In a nutshell, PCA indicates to what degree certain "most typical temporal behaviors" (called principle components), are present in each mobile's time series (see [6] for details). for the first three models RWP, RWP(STT) and RWP(CS) we observe hardly any structure in the PCA decomposition, meaning that the principal components are present in a random and unpredictable manner in the mobiles. For the Ricesims models, on the other hand, a handful of clusters of mobiles show up in the PCA, meaning that similar proportion of the principle components are present in each mobile of a given cluster. In other words, mobiles can be divided into groups of similar behavior (clusters in the PCA). How this grouping relate to attributes is ongoing work.

D. Estimating the Duration of Node-pair Connections

The time series of average node degree is a good indicator of global connectivity. More telling would be the actual connection times between any two nodes. Since this computationally untractable for large node populations in long time scenarios we estimate the average duration of a node-node connection via the following *Sampling Strategy*:

1) pick a random time t
2) pick a random member (mobile) M_k
3) pick at random a different member M_n within the radio range of M_k (range does not have to be a disc)
4) going both forward and backward in time calculate the length Y_k of the longest time interval which contains t and during which M_k and M_n stayed within range.

Bias Adopting this sampling procedure is quite natural, yet it introduces a *length bias* one needs to be aware of. For clarity, we denote the sampled (observed) random variable by Y to contrast with the desired random variable L, i.e., the actual length of a node-node connection. Note that Y and L assume the same range of values, but with different chances.

The mental picture of drawing a sample of Y is as follows: Disjoint intervals $\{I_j\}_{j=1}^N$ with lengths $|I_j|$ independently drawn from the distribution of durations, i.e., connection lengths L are set next to each other in random order, forming a large interval $[0, T]$. To sample an interval I_j, we choose a random time t in $[0, T]$ and record the length of the interval t falls into, say l_i. In the notation of our sampling, this leads to a sample $Y = l_i$. Repeating M times with the same intervals I_j, we count the number m_i of thus sampled lengths $Y = l_i$. Dividing by the total number M of samples we find that $m_i/M \simeq P[Y = l_i]$. However, this is not the desired probability $P[L = l_i]$. Indeed, a moment's thought convinces the reader that long intervals are picked with higher chance. This introduces a bias towards large durations, i.e., we would *overestimate the duration time* if we left this bias uncorrected.

Bias Correction To relate $P[Y = l_i]$ to $p_i := P[L = l_i]$ let us assume that n_i of the intervals $\{I_j\}_{j=1}^N$ possess the length l_i. Clearly $\sum_i n_i = N$ and $\sum_i l_i n_i = T$; also, $p_i \simeq \frac{n_i}{N}$ and $\frac{T}{N} \simeq \mathbb{E}[L]$ for large N. Now, since the probability of t falling into an interval of length exactly equal l_i is computed as the length of the favorable intervals divided by the total length of intervals, we find

$$P[Y = l_i] = \frac{l_i n_i}{T} = \frac{l_i \cdot n_i/N}{T/N} \simeq p_i \frac{l_i}{\mathbb{E}[L]} \quad \text{for } N \text{ large.}$$

This is intuitively appealing since it indicates that the bias is proportional to the interval length and our *un-biased* estimate of p_i becomes

$$\hat{p}_i = \frac{m_i}{M} \frac{1}{l_i} \cdot \mathbb{E}[L]. \tag{1}$$

Thus, we unbias the simple-minded estimator m_i/M by dividing it by l_i; the constant factor $\mathbb{E}[L]$ can be viewed as a normalizing constant which ensures that $\sum_i \hat{p}_i = 1$ for N large. Thus, we may approximate it by simply normalizing the estimated probabilities $\hat{p}_i$.

Un-biased Estimation of Duration So far, we have assumed that L and Y are discrete for simplicity of the argument. When sampling the actual real world duration times, which are continuous in nature, we proceed via histograms. Denoting the mid-points of the histogram bins by l_i, the count of values of Y that fall in bin with mid-point l_i by m_i then (1) remains formally the same: we only need to *divide* m_i *by* l_i and *re-normalize* the histogram to arrive at $\hat{p}_i$.

Estimated Duration of node-node connections Figure 5 summarizes the results of the analysis.

The analysis suggests that mobility models that constrain mobility on a fixed path appear to have more occurrences of longer paired movement than RWP. Also, the underlying group structure that exists in Ricesims appears to be captured in this metric, as many mobiles have similar schedules, and hence, modestly greater paired movement. Increased measured correlation between individuals will likely cause this metric to differentiate Ricesims from the other models even further.

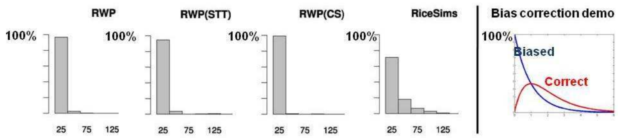

Fig. 5. Histogram of Connection Times in secs (unbiased estimation). Far right: Estimation of a gamma distribution. Without bias correction, the estimated density appears exponential (blue decaying curve), with bias correction the mode of the gamma distribution becomes apparent (red single hump curve). Since bias was corrected we can trust that no mode is present in the estimated Connection Times.

V. Network Protocol Experiments

We have conducted extensive simulation studies under these mobility models comparing the re-active protocols AODV [20] and DSR [11], as well as the pro-active SAFARI [19] routing protocol.

A. Simulation Details

Mobility For RWP, RWP(STT), RWP(CS) the space consisted of a rectangular area of size 1850m by 1550m, which is comparable in area to Rice campus, the space for RiceSims and RiceSime(STT). We created 78 destinations for all except RWP. The total simulation time was 30 minutes for 300 mobiles. For the RiceSims and RiceSims(STT) case, these 30 minutes were taken from the period of 08:45 - 09:15 AM, which includes a period of intense mobility as the students depart class at 8:50. We created one mobility trace for each of the five mobility model.
Communication We used CBR (Constant-Bit-Rate) sessions with a sending rate of 256 Kbps, whose duration we choose from an exponential distribution with mean at 4 seconds. Source and destination devices were selected uniformly at random for each session. We selected session starting times either randomly spread out over the duration of the simulation (**random sessions**) or in a correlated fashion (**correlated sessions**), where a device is more likely to start a session if devices in its intermediate neighborhood recently started a session. An average of 1200 sessions were generated for each simulation run. We created 30 random session sets and 30 correlated session sets.
Radio range We chose the transmission range of the mobile devices to be either 40m or 100m.
Run time Simulating with RiceSims the *mobility traces* for 500 nodes corresponding to a full days worth (86400 secs) took roughly 3 mins. Network simulation appears to be the demanding portion of the overall simulation.

All other settings were at ns-2 default.

B. Observations

We simulated each routing protocol (AODV, DSR, and SAFARI) using the 30 random and the 30 correlated session sets for the mobility trace of each mobility model and for each of the two transmission ranges. For each simulation run, we computed throughput and number of send operations which we report in full to privide a sense of the spread caused by random initializations, instead of reporting only averages over the runs (see Figure 6). We point out only the most remarkable conclusions.
Mobility models. A finite set of destination points results in better connectivity that all routing protocols are able to turn into an increase of throughput at the cost of a modest increase in network load. This is particularly noticeable in the 40m range scenarios (compare RWP to the rest) where throughput grows by almost a full magnitude but persists in the 100m range scenarios. Thus, simple RWP underestimates performance in the presence of preferred target locations of mobiles.

Correlated movement such as starting to move at common time and location along joint paths obviously further increases connectivity at least among certain nodes. But it also has the potential to increase the life time of multi-hop routes and makes route repair more efficient. This is well demonstrated when comparing RWP(SST) to the higher structured mobility models in the 40m range scenario with poor connectivity. While lack of correlated movement is a liability for RWT(STT) in the 40m scenario, it turns into an asset in the 100m case as it is able to spread out traffic load more evenly while connectivity is ensured in most cases. Even higher mobility-correlation is present for RWP(CS) which allows protocols to further increase performance in most scenarios.

RiceSims and RiceSims(STT) perform very similar to each other in the 40m scenarios. In the 100m scenario, RiceSims achieves a slightly better throughput than RiceSims(STT) for AODV and SAFARI as following walk-ways leads to better connectivity. Note that the RiceSims mobility models never correspond to the largest throughput in any of the scenarios.

In conclusion, more simplistic mobility models would over- or under-estimate performance on an urban scenario such as Rice campus and run the risk of inverting the ranking between different routing protocols.
Protocols. The reactive AODV and DSR protocols overall perform similar to each other, which is in accordance with earlier results. Minor differences could be due to implementation details of the respective versions in ns-2 that favors AODV, in particular with the way route recovery is handled in version ns-2.2.27. Interestingly, DSR performs as expected for the RiceSims and RiceSims(STT) models.

SAFARI tends to outperform the other protocols, most convincingly in the realistic 100m transmission range scenario, at the cost of higher overhead. Pro-activity is an asset in the 100m scenario, it is a liability in the 40m scenario, where the incurred network load is almost a magnitude higher than its competitors.
Session types. Having correlated sessions or just random sessions does not change relative protocol rankings. Random sessions overall lead to better throughput by about a factor of two. In the case of the re-active DSR and AODV protocols, this improvement comes at the cost of about two-fold increased network load. The pro-active SAFARI protocol keeps its

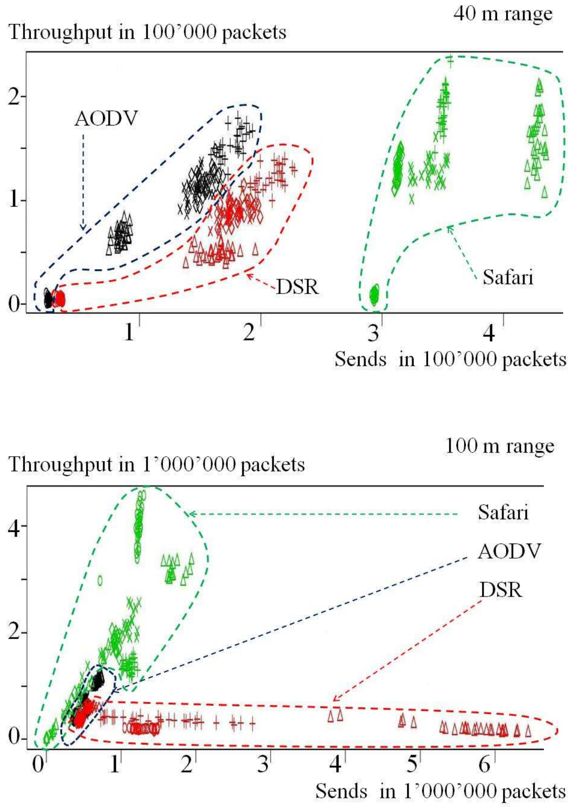

Fig. 6. Throughput by Sends for correlated sessions (random sessions see text). Range: 40m on top, 100m on bottom. Protocols: AODV black, DSR red, Safari green. Mobility models: RWP o, RWP(SST) Δ, RWP(CS) +, RiceSims(SST) $\diamond$, RiceSims x. Each data point represents one simulation run with the indicated protocols and settings.

network load constant. We chose our simulation parameters deliberately such that congestion would not be the main issue in getting data across the network. Testing high-load scenarios is ongoing work.

VI. Conclusions

Pedsims provides a flexible platform on which to study the relevance of mobility patterns on network performance and receive insight on how to trade off accuracy against computational complexity and run time. PedSims comes together with an implementation of the social and geographic context of a small university campus called RiceSims, which is still at a computationally manageable level.

Our study shows that simplifications in the modeling of mobility, though often necessary, may lead to over- or underestimation of network performance as well as to inversion of protocol ranking. Findings indicate that implementing common destinations and departure times of nodes —if present in the real world scenario to be modelled— are an essential and simple step to take.

Further contributions include a bias correction for the sampling of node-node connection time as well as a ranking of certain ad hoc routing protocols in certain mobility scenarios. Future work will address the use of real world data sets.

VII. Acknowledgements

Financial support came from NSF under grant number ANI-0338856. We would like to thank the Rice University GIS data center for their valuable assistance.

References

[1] C. Bettstetter, C. Wagner. The spatial node distribution of the random waypoint mobility model. In Proceedings of German Workshop on Mobile Ad Hoc networks (WMAN), Ulm, Germany, March 2002.

[2] J. Boudec, M. Vojnovic. Perfect Simulation and Stationarity of a Class of Mobility Models. Infocom 2005.

[3] T. Camp, J. Boleng, V. Davies. A Survey of Mobility Models for Ad Hoc Network Research. Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, 2(5):483 502, 2002.

[4] V. Davies. Evaluating Mobility Models Within an Ad Hoc Network. Masters thesis, Colorado School of Mines, Mathematical and Computer Sciences, 2000.

[5] M. J. Feeley, N. Hutchinson, S. Ray. Realistic Mobility for Mobile Ad Hoc Network Simulation, in 3rd International Conference on AD-HOC Networks and Wireless (ADHOCNOW), Vancouver, BC, July 2004.

[6] H. Flores, S. Eidenbenz, R. Riedi, N. Hengartner. Describing MANETS: Principal Component Analysis of Sparse Mobility Traces. Third ACM International Workshop on Performance Evaluation of Wireless Ad Hoc Sensor, and Ubiquitous Networks (PE-WASUN 2006).

[7] W. Hsu, K. Merchant, H. Shu, C. Hsu, H. Helmy. Weighted Waypoint Mobility Model and its Impact on Ad Hoc Networks - Mobicom 2004 poster.

[8] S. Kurkowski, T. Camp, N. Mushell, M. Colagrosso. A Visualization and Analysis Tool for NS-2 Wireless Simulations: iNSpect. Colorado School of Mines, 2004.

[9] LANL TRANSIMS-3.0 documentation set. Technical report, Los Alamos National Laboratory, 2002. URL http://transims.tsasa.lanl.gov/TRANSIMS_3-0_docs.html.

[10] A. Jardosh, E. Belding-Royer, K. Almeroth, S. Suri. Real-world Environment Models for Mobile Network Evaluation. Journal on Selected Areas in Communications special issue on Wireless Ad hoc Networks, March 2005.

[11] D. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks, in Mobile Computing, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers, 1996, ch. 5, pp. 153-181.

[12] MobiREAL: A Realistic Network Simulator. URL http://www.mobireal.net/

[13] K. Maeda, K. Sato, K. Konishi, A. Yamasaki, A. Uchiyama, H. Yamaguchi, K. Yasumoto, and T. Higashino. Getting Urban Pedestrian Flow from Simple Observation: Realistic Mobility Generation in Wireless Network Simulation. in Proceedings of the 8th ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM2005), 2005.

[14] A. Saha, D. Johnson. Modeling mobility for vehicular ad-hoc networks. VANET 04: Proceedings of ACM workshop on Vehicular Ad Hoc Networks, pages 91 92, New York, NY, USA, 2004. ACM Press.

[15] P. Nain, D. Towsley, B. Liu, Z. Liu. Properties of Random Direction Models. INRIA technical report RR-5284, July 2004.

[16] H. Pucha, S. Das, Y.Hu. The Performance Impact of Traffic Patterns on Routing Protocols in Mobile Ad Hoc Networks. MSWiM 04 October 46, 2004,

[17] M. Sarela. Measuring the Effects of Mobility on Reactive Ad hoc Routing Protocols. Helsinki University of Technology Laboratory for Theoretical Computer Science Research Reports 91, 2004.

[18] Y. Tseng, W. Liao, S. Wu. Mobile Ad Hoc Networks and Routing Protocols in: Handbook of Wireless Networks and Mobile Computing, I. Stojmenovic (ed.), Wiley and Sons, New York, pp. 195 - 218, 2002.

[19] S. Du, M. Khan, S. PalChaudhuri, A. Post, A. Saha, P. Druschel, D. Johnson, R. Riedi. Safari: Self-Organizing Hierarchical Architecture for Scalable Ad Hoc Networking, Ad Hoc Networks, 6(4), pp. 485–507, (June 2008).

[20] C. Perkins, E. Royer. Ad hoc On-Demand Distance Vector Routing, Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, 1999.