

UC Santa Cruz

UC Santa Cruz Previously Published Works

Title

MANET Protocol Simulations Considered Harmful: The Case for Benchmarking

Permalink

<https://escholarship.org/uc/item/5ww6x4r2>

Author

Garcia-Luna-Aceves, J.J.

Publication Date

2013-08-01

Peer reviewed

MANET PROTOCOL SIMULATIONS CONSIDERED HARMFUL: THE CASE FOR BENCHMARKING

DANIEL HIRANANDANI, KATIA OBRACZKA, AND J.J. GARCIA-LUNA-ACEVES,
UNIVERSITY OF CALIFORNIA, SANTA

ABSTRACT

In this article, we investigate the current best practices in simulation-based multi-hop wireless ad-hoc network (MANET) protocol evaluation. We extend a prior characterization of the settings and parameters used in MANET simulations by studying the papers published in one of the premier mobile networking conferences between 2006 and 2010. We find that there are still several configuration pitfalls which many papers fall victim to, which in turn damages the integrity of the results as well as any research aimed at reproducing and extending these results. We then describe the simulation “design space” of MANET routing in terms of its basic dimensions and corresponding parameters. We also propose four “auxiliary” metrics to increase simulation integrity. We conclude with several example scenarios that promote modeling simulations after real-world situations.

INTRODUCTION

Network simulations are extensively used in the design and evaluation of computer networks and their protocols. There are many reasons why network practitioners and researchers turn to simulations either as an alternative or to complement actual “live” experiments. Some of the main reasons for the popularity of network simulators are ease of experiment reproducibility and scalability. On the other hand, it is critical that simulation scenarios accurately and adequately reflect the real environments and conditions under which the network systems being studied will operate.

Unfortunately, as pointed out by previous surveys on the topic [1], this is often not the case. Not only that but, our survey of the papers published in the ACM MobiHoc conferences between 2006 and 2010 (more details on the survey are provided later in the article) indicates that most of the papers that used simulations as their experimental platform do not fully disclose the settings and parameters used. This lack of full disclosure calls into question the quality and reproducibility of the experiments: not only it is not possible for a third-party to reliably achieve the same results but also it questions the validity

of the conclusions that are based on the simulation results. As research communities thrive on extending the work of others, the lack of full knowledge of the experimental methodology used by previous efforts is a serious inhibiting factor. Even worse, we fear that the researchers themselves do not know what parameters they are using. Relying on default values in a simulator will likely produce different results between simulator versions, and will certainly produce different results when comparing different simulators.

Additionally, quite often the designers of the protocols are the ones designing the tests by which their protocols are evaluated. Consequently, there tends to be a bias where the developer designs the experiments that will highlight the positive features of their protocols. So it is not always the case that the tests thoroughly expose the protocol to the full spectrum of operating conditions.

At this point, it is interesting to look at how some other disciplines perform experimental evaluations. Benchmarking, i.e., running a set of standard tests for relative performance assessment is widely adopted in computer architecture, VLSI, compilers, and databases, to name a few.

The overarching goal of this effort is to promote benchmarking as the standard best practice when designing and studying the performance of computer networks and their protocols. This article is our first step towards this goal: here, we highlight the issues that currently exist with designing and documenting simulation experiments and introduce the basis for developing a set of guidelines for benchmarking network protocols using simulations. As a starting point, we focus on routing protocols for wireless multi-hop ad hoc networks (MANETs). MANETs refer to infrastructure-less networks where there is no functional distinction between hosts and routers: all nodes can originate, sink, as well as forward traffic. MANET routing protocols reflect the wide variety of MANETs which can take on many forms ranging from static, dense, and homogeneous networks to highly mobile, sparse, and even connectivity-challenged networks.

The next step will be to take the simulation guidelines we produce and develop a suite of benchmarking scripts to run any MANET rout-

ing protocol against a set of scenarios and produce results that can be compared to other MANET routing protocols. We will then create a publicly-accessible repository to store the benchmarking scripts, as well as results of these benchmarks. These tools will ensure that the community:

- Has reliable, reproducible, and rigorous tests
- Is able to view and use as baseline the results of other protocols without having to re-run simulations to reproduce them
- Is able to easily produce relative performance results for their own protocols

The article is organized as follows. The next section puts our work in the context of previous and related work. We then provide a snapshot of the current best practices in conducting and reporting simulation experiments. We go on to describe the basis to create a benchmarking suite for MANET routing by outlining the design space of MANET routing simulation parameters. Our conclusions and directions for future work are discussed in the last two sections.

RELATED WORK

There has been a few efforts that have focused on studying the validity of simulation-based protocol evaluation. However, as will become clear later, the community as a whole has not been following the recommendations provided by previous work.

For example, the work presented in [2] reported important statistics for the simulation-based papers accepted to the ACM MobiHoc conference up to 2006. This article was a very important milestone as it brought to light the current best practices in simulation-based evaluation of MANET protocols. Our work leverages on this effort and goes a step further: it shows that current practices in simulation-based MANET protocol evaluation are practically unchanged; it then describes the design space of MANET routing protocols in terms of its fundamental parameters as the basis for the evaluation guidelines for these protocols.

In [3] two key auxiliary metrics were introduced: average shortest path hop count and average network partitioning. The proposed metrics are periodically measured over the duration of the simulation and provide feedback on the effectiveness of the scenario being used. Between these two auxiliary metrics and the third auxiliary metric recently introduced by [4], the average neighbor count, a researcher can identify if their simulation scenario has too few or too large of average hop count distances, too little connectivity which leads to network partitioning, and too dense or too sparse of a network. These auxiliary metrics do provide excellent information about a scenario; in this article, we propose four additional metrics that capture important information about traffic workload. These metrics are described later.

The use of simple models is proposed in [1], which surveyed the papers published in the ACM MobiHoc 2008. They found that 59 percent of the papers did not run meaningful comparative studies: they either did not compare against “truly competing solution(s)” or did not

compare their solution(s) to any other protocol. We argue that we can eliminate this problem with our future work aimed at creating a publicly available standardized set of routing protocol benchmarks.

The work in [5] reports a survey similar to [8] in which they studied 280 papers on simulations of peer-to-peer systems and found that 71 papers did not even state which simulator was used.

Five principles are shown in [6] which reinforces the importance of having repeatable, rigorous, complete, statistically and empirically valid simulations. It stresses the need for researchers to include all parameters and configurations used in their experiments.

In [7] the implementation differences of IEEE 802.11 was studied by comparing two different simulators as well as the differences in 802.11 within multiple ns-2 versions. This study finds that the results between simulators are quite different (although they also find that the difference is minimal when the ns-2 802.11 MAC is ported into OMNeT++).

The work described in [8] also finds in 2006 that very little has changed in the MANET community in terms of simulation-based evaluation methodology. Almost 90 percent of papers do not even specify the simulator version, and over half do not specify the number of simulation runs. These omissions make accurately reproducing and re-using previous results near impossible.

In the area of CPU benchmarking, the Standard Performance Evaluation Corporation (SPEC) [9] has a standardized suite of tests that evaluate the performance of a processor. This suite of tests comprises many smaller tests using various real-world applications including compression, compiling, discrete event simulation, and speech recognition. When new processors are vetted against previous processors, these SPEC tests are run on the processor that produces a single numeric value that can be compared to the other processors to determine a hierarchy of rank. While a single value is not telling enough to describe a routing protocol's performance, the idea of running a battery of standard tests on different routing protocols and having measurable as well as comparable metrics is very applicable and desirable to the MANET community.

CURRENT MANET SIMULATION BEST PRACTICES

In order to characterize the current state-of-the-art in evaluating MANET protocols, we conducted a survey of the full papers accepted into the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc) for the years 2006 through 2010. We chose to study MobiHoc because of the prior surveys which identified the common problems in papers accepted into previous editions of the same conference. We also focused on MobiHoc as it is regarded as a highly selective and prestigious conference in the MANET community. We consider the results of this survey to be a “best practices” survey in that they represent the configurations used by some

The overarching goal of this effort is to promote benchmarking as the standard best practice when designing and studying the performance of computer networks and their protocols. This article is our first step towards this goal.

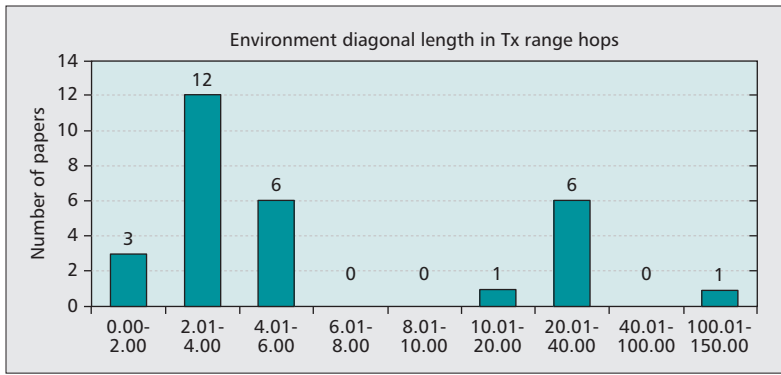


Figure 1. Measuring MobiHoc 2006–2010 simulation scenario environment area in terms of the number of transmission range hops across the area’s diagonal.

of the most influential papers published in the MANET community. Although we were able to find 12 of the 25 papers submitted to the most recent MobiHoc 2011 (which occurred near the time of this paper’s submission deadline), we chose to omit all of the MobiHoc 2011 papers from our survey to preserve completeness of each year’s worth of papers.

In addition to the MobiHoc survey, we extended the survey to include 82 papers which simulated routing protocols from a variety of conferences to get a broader idea of what the “average” MANET routing protocol simulation scenario looks like. Besides broadening the universe of papers surveyed, we extended the survey outside of MobiHoc for a number of other reasons, including: fewer routing protocol papers have been published in MobiHoc in the recent years compared to the previous study; few papers actually specify their configurations and parameters out of the routing papers published in recent MobiHoc years (which further demonstrates the issue at hand).

MOBIHOC SURVEY

Out of the 159 MobiHoc papers we reviewed, while we noticed that simulation is still an often-used tool to evaluate protocol performance (105 papers or 66.0 percent used simulations), only 47 out of the 105 simulation-based MobiHoc papers specified which simulator was used (44.8 percent). Comparing the popularity of individual simulators used in these published papers to a survey of the 2000–2005 MobiHoc papers that used simulation-based evaluations [8], we found that Network Simulator 2 (ns-2) tied with custom simulators (13 of the 47 simulated papers for each, or 27.7 percent each). This is quite different from the previous study, which ranked ns-2 at 43.8 percent and self-developed simulators at 27.3 percent. Matlab, however, had the largest increase in declared popularity moving from only 3.8 percent proclaimed-usage in the previous study to 21.3 percent in our survey.

Some of the more startling results show that out of the 25 papers which identify the mobility model used, 8 papers (32 percent) use the Random Waypoint Mobility Model which has been shown to exhibit undesirable behavior and thus produce unreliable results [10]. Additionally, of the 15 papers which identify how source and

destination pairs were chosen to be the source and sink of a flow, random selection was the most popular at 86.6 percent which has two major issues. The first is that there is no guarantee on the minimum number of hops for the path between the source and the destination; thus, the source and destination pairs can potentially be next to each other and traffic between them does not require any routing. Secondly, the nodes that are chosen might produce traffic flows that do not overlap. Overlapping flows stress the network’s ability to handle multiple flows in terms of queuing and processing power, so scenarios with zero concurrent flows could produce artificially optimistic results.

We also noticed that the papers surveyed tend to run a single scenario multiple times. While we certainly encourage such practice to reduce the effect of outlying results, we need to address the problem of using the same Pseudo Random Number Generator’s (PNRG) seed. It was noticed that in ns-2 which uses a fixed PRNG seed of 12345, rerunning simulations without ever changing the seed will produce identical results [2]. Nevertheless, this problem is still quite prevalent today. To quantify the gravity of this issue, of the 39 papers that declared running more than one simulation run, only 5 papers (12.8 percent) addressed changing the PNRG seed for each run.

As a measure of how arbitrary topologies are chosen, we measured the longest distance in an environment (the diagonal of the rectangular environments) in terms of a node’s transmission range. This distance spans the longest point-to-point distance in the environment, and roughly shows the maximum possible diameter of the network. Figure 1 shows that the values range from as little as 0.35 hops all the way up to 141 hops.

In order to better characterize the quality of scenarios used, we also calculated one of our proposed auxiliary metrics, the Average Node Density of the network which is described in more detail later for the papers that listed the environment dimensions, number of nodes, and node transmission range. The results are shown in Table 2 sorted by density, and they show a very wide range of values ranging from 400 nodes per cell down to 0.025 nodes per cell.

A fairly common occurrence was for papers to declare that they were unable to provide details on the simulation configurations due to the “constrained space” of the article. We realize that this is an understandable concern, however it is not acceptable. It is, however, perfectly acceptable to include a URL that links to the research group’s or individual’s website that contains information about the simulations. Taking this one step further, we also encourage including contact information on the website to give others a way to obtain the code used in the simulations.

As a fitting example, while we lack the space in this article to provide the full results of the MobiHoc survey, we show a snapshot of the results in Table 1 and Table 2. We also invite the readers to visit <http://inrg.cse.ucsc.edu/> to view the more detailed version of these statistics.

EXTENDED SURVEY

In this extended survey, we noticed that a few patterns emerged in the values of parameters chosen, and we list them in Table 3. We found that the following is an appropriate “average simulation” in the sense that these were the most-often occurring values of parameters in our study. While we found that ns-2 was the most commonly used simulator, the parameter values shown in Table 3 apply to all simulators.

We can see from Table 3 that the average scenario randomly places nodes in the environment, randomly moves them, and then randomly selects which nodes will communicate with each other. We do believe in adding randomness to simulations to introduce variance, but randomizing all of these aspects will make the scenario overly synthetic, and thus far from reality.

SIMULATION DESIGN SPACE AND GUIDELINES

In this section, we lay out the basis for specifying a set of guidelines to standardize the evaluation of MANET routing protocols. Such guidelines will enable not only accurate and unbiased performance assessment, but also sharing of tools and results. We start by identifying the main performance “dimensions” of MANET routing, namely: topology, traffic, and mobility. We have found that many of the values used in simulation studies are largely synthetic, i.e., they are not consistent with real-world scenarios. Since the values of these parameters largely depend on the driving application(s), we provide some example real-world situations along with logical ranges of values that could be used when simulating them.

TOPOLOGY

The topology used by simulation experiments describes physically the environment where the simulation takes place. It typically includes: width, height, terrain, channel characteristics (e.g., path loss, fading, etc.), number of nodes, the nodes’ transmission ranges, and the method for determining where to place the nodes (node distribution/placement).

These parameters are often modified individually to produce “new” simulation scenarios. However, some parameters may have a “collective” effect. This is the case, for example, of number of nodes, size of the area, and transmission range, all of which affect the density of the network being simulated, which, in turn, can have significant impact on the performance of MANET routing.

Examples have shown how easy it is to create two scenarios that are effectively measuring the same things when changing the environment size, average node speed, and transmission range, and how important it is to identify the environment size and node speed in terms of the transmission range of the node [2]. We propose that this should be taken one step further to include the number of nodes in these measures as well. Since the environment size, node speed, and transmission range effectively describe the node density of a scenario, it is logical to see

Description	Totals	Percentage
Used simulation for protocol evaluations	105 of 159	66.0%
Specified which simulator was used	47 of 105	44.8%
Used ns-2 as their simulator	13 of 47	27.7%
Used Qualnet as their simulator	3 of 47	6.4%
Used a custom simulator	13 of 47	27.7%
Used JIST/SWANS as their simulator	3 of 47	6.4%
Used Matlab as their simulator	10 of 47	21.3%
Used Opnet as their simulator	1 of 47	2.1%
Used QNS as their simulator	2 of 47	4.3%
Used TOSSIM as their simulator	1 of 47	4.3%
Used Silhouette as their simulator	1 of 47	2.1%
Stated the simulator version	5 of 105	5.8%
Used mobility	25 of 105	23.8%
Used the random waypoint mobility model	8 of 25	32.0%
Used the random walk mobility model	3 of 25	12.0%
Used the Brownian motion model	2 of 25	8.0%
Used the random direction mobility model	1 of 25	4.0%
Used other mobility models	4 of 25	16.0%
Used mobility traces	7 of 25	28.0%
Used the UMass DieselNet trace	1 of 7	14.3%
Used the MIT Reality trace	2 of 7	28.6%
Used the Intel Labs trace	1 of 7	14.3%
Used the Infocom DTN trace	1 of 7	14.3%
Used the Huggle traces	1 of 7	14.3%
Used other (unnamed) traces	2 of 7	28.6%
Declared node distribution methods	34 of 105	32.4%
Used random distribution	14 of 34	41.2%
Used uniform distribution	10 of 34	29.4%
Used perturbed grid distribution	2 of 34	5.9%
Used Poisson distribution	5 of 34	14.7%
Used clusters	2 of 34	5.9%
Used power-law distribution	1 of 34	2.9%
Stated Packet Rate Distribution	9 of 105	8.6%
Used CBR Flows	7 of 9	77.8%
Used VBR Flows	2 of 9	22.2%
Stated source-destination pair selection	15 of 105	14.3%
Used random selection	13 of 15	86.7%
Used fixed selection	2 of 15	13.3%
Stated the environment dimensions	51 of 105	48.6%
Stated the radio transmission fange	37 of 105	35.2%
Stated the number of Nodes	66 of 105	62.9%
Stated the number of simulation runs	38 of 105	36.2%
Used multiple PNRG seeds	6 of 105	5.7%
Accounted for steady-state (traffic or mobility or both)	7 of 105	6.7%
Stated metrics collected	66 of 105	62.9%
Measured delivery ratio	16 of 66	24.2%
Measured delay	19 of 66	28.9%
Measured overhead	4 of 66	6.1%
Measured throughput	12 of 66	18.2%
Measured average hop count	5 of 66	7.6%
Measured average node degree	5 of 66	7.6%
Measured energy consumption	4 of 66	6.1%
Measured network diameter	1 of 66	1.5%

Table 1. *MobiHoc 2006–2010 simulation survey.*

that the number of nodes plays a significant role in the density as well. When designing a scenario, careful thought should be put into these

four parameters because they are not independent of each other - they all affect the network density.

TRAFFIC

Modeling traffic includes parameters such as when nodes start and stop sending data, the number of traffic flows, the selection process for determining source and destination nodes, and the packet rate distribution. While values assigned to these parameters should reflect the driving application(s), there are a few guidelines that should be followed when selecting values for these parameters.

As pointed out earlier, the most common method for choosing traffic source-destination pairs is random selection. Therefore, it is impor-

tant to note some of the adverse effects this methodology can introduce. For example, selecting source-destination pairs at random may cause, as side-effect, the number of hops in the path to be abnormally small or large. It may also mean that no node might ever have to route for more than one flow at a time. However, it is necessary to subject network protocols in general, and routing in particular to heavier as well as non-uniform traffic loads. Therefore, other traffic source-destination selection policies in addition to uniform selection need to be employed. We also propose an additional auxiliary metric that will measure the average number of concurrent flows in the nodes along the routed paths. We recognize that this auxiliary metric will provide a coarse-grain analysis of the problem, however it will address the issue of not even knowing whether scenarios contain any overlapping flows at all.

MOBILITY

Mobility models, like topology and traffic, are very application dependent, and while some mobility models are more popular than others, they are not necessarily the best models to use. Take for example Random Waypoint Mobility: it is still the most used mobility model, but it does not produce realistic movement for applications such as human walks [4]. Therefore, using mobility models such as Self-similar Least Action Walk (SLAW) would be preferred instead. Additionally, there are traces such as MIT Reality [11] as well as others available through CRAW-DAD [12] which can provide realistic node mobility.

While we strongly recommend the use of real-world mobility traces (as shown in the mobile sample scenarios described later and summarized in Table 4), we realize it is not always possible to collect relevant traces. For these kinds of situations, we urge readers to reference research such as [13] which analyzes and compares the use of WLAN, GPS, and synthetic traces in the context of producing realistic human mobility models.

METRICS

We make a distinction between metrics and auxiliary metrics. We view metrics as being used to evaluate the performance of a protocol whereas auxiliary metrics are used to evaluate the effectiveness of a simulation scenario. There is a danger in some of the auxiliary metrics because there are *similar* metrics that are more common to use (such as average hop count), but they don't provide as valuable of information (compared to average shortest path hop count).

Performance Metrics — In our extended survey of MANET routing protocol papers, we found that there were several de facto metrics for evaluating the performance of a protocol. These metrics are shown in Fig. 2, and include Delivery Ratio (number of packets received divided by the number of packets sent), Average Delay (Average over all received packets' arrival time minus departure time), Throughput (Total number of bytes sent divided by the total amount of time data was being sent), Overhead (Total number

# Nodes	Dimensions	Tx Range	Avg Density
10000	1500 m × 1500 m	150 m	400.000
20	300 m × 300 m	600 m	320.000
1000	500 m × 500 m	120 m	230.400
100	1000 m × 1000 m	600 m	144.000
300	500 m × 500 m	100 m	48.000
512	4000 m × 4000 m	600 m	46.080
160	100 m × 100 m	20 m	25.600
50	100 m × 100 m	32 m	20.480
20	500 m × 500 m	250 m	20.000
250	300 m × 300 m	40 m	17.778
100	1250 m × 1250 m	250 m	16.000
1000	4000 m × 4000 m	250 m	15.625
150	600 m × 600 m	88 m	12.907
3000	610 m × 610 m	20 m	12.900
50	1000 m × 1000 m	250 m	12.500
80	600 m × 600 m	100 m	8.889
50	1 m × 1 m	0.20 m	8.000
54	100 m × 100 m	18 m	6.998
64000	1000 m × 1000 m	5 m	6.400
500	200 m × 200 m	6 m	1.800
100	100 m × 100 m	3 m	0.360
100	5000 m × 5000 m	150 m	0.360
10	4000 m × 4000 m	100 m	0.025

Table 2. *MobiHoc 2006–2010 average densities.*

of bytes used for transmitting routing information), and Energy Consumption, while the most common auxiliary metric was Average Node Degree (Average Number of Node Neighbors). Some metrics were combined for comparisons because there are several variations of the metrics that can alternatively be used (such as throughput can be alternatively measured as goodput - the total number of bytes received divided by the amount of time data was being sent).

Auxiliary Metrics — We propose that there are four more important auxiliary metrics that should be accounted for in determining the effectiveness of a simulation scenario: Average Node Density, the Average Number of Concurrent Flows, the Number of Unroutable Packets, and the Average Source-Destination Distance in Transmission Range Hops.

Average Node Density — Average Node Density is a coarse measure of how dense or sparse a network is. In order to calculate the Average Node Density of a network, we assume a grid distribution of the nodes across the environment, and we also approximate a node's transmission area as a square with sides of length $2 * T_r$ where T_r is the Transmission Range of the node. We first divide the environment area (*width * height*) into cells of the same size as the node's transmission area, and then we divide the number of nodes by the number of cells which gives us the number of nodes per cell. This auxiliary metric is very easy to calculate, and can give valuable insight into the scenario long before the scenario is simulated. It is important to note that since the density of the network will change over time in a mobile network, so in order to more accurately calculate the density of a dynamic network, measurements are taken periodically throughout the simulation to capture a sequence of instantaneous node densities.

Average Number of Concurrent Flows — The Average Number of Concurrent Flows is a measure of traffic flows that overlap in time. This metric will allow researchers to design simulations with higher numbers of concurrent flows to not only ensure that the network experiences higher congestion which will in turn demonstrate the worst-case performance of a protocol, it will also expose synthetic scenarios. Nodes routing multiple flows simultaneously will experience a higher level of stress at all layers: contention and collisions at the PHY/MAC layers, queueing in the routing layer, and QoS in the application. Real-world situations tend to have at least a few nodes in which flows converge upon such as fixed data sinks in smart energy meter networks [6] or emergency response situations where local clusters communicate internally before forwarding data to upstream clusters, so the lack of simulating flows converging upon certain points in the network will lead to unrealistic results.

We also considered the related, and more powerful auxiliary metric, Average Number of Overlapping Flows. This metric gives insight into the number of concurrent flows, as well as detailing how the flows interact with each other. It is

Parameter	Value
Simulator used	ns-2
Simulation duration	900 s
Env. dimensions	1000 m × 1000 m, 1500 m × 300 m
Number of nodes	50 or 100
Node transmission range	250 m
MAC protocol used	802.11
Initial node distribution	Random
Node mobility model	Random waypoint
Packet size	512 bytes
Packet send rate S-D pair selection	4 packets/s
Number of traffic flows	Random 20

Table 3. Extended simulation survey “average scenario.”

entirely possible to have a high number of concurrent flows in a network where none of the flows overlap, so the network would not be as stressed as it would first appear. Counting the number of overlapping flows, however, would provide information regarding how many flows were being routed by each intermediate node in the path. Nodes having to simultaneously route for many flows would experience a higher load, and would thus produce more meaningful results when trying to stress the network. The drawbacks in implementing this auxiliary metric lie in needing to expose the path a packet will take to determine which nodes are routing for that flow.

Since paths are determined by the routing protocol, and those paths can change due to mobility, network load, available energy on the node, etc., this metric would need to keep a mapping between a flow and the path(s) it takes at each instance of time that the network is probed. Since some paths are pre-determined, it might be necessary to look inside the routing tables as packets are being generated which reduces the ability to calculate this metric without requiring specific code for each routing protocol. A better approach would be to have nodes record packet sequence numbers when they are sent or received so that the path can be determined and mapped back to the flow that the packets belong to in post-processing. The drawback of this approach is that it requires a significant amount of memory and post-processing power, so we leave this metric to the future work as an improvement upon the Average Number of Concurrent Flows.

Number of Unroutable Packets — The Number of Unroutable Packets is a measure of the packets dropped due to the routing protocol which includes no route existing between the source and destination nodes, and the TTL being exceeded on a packet. While Delivery Ratio includes one aspect of this auxiliary metric since it counts the number of packets that *were* able to be delivered, the packets that were not able to be delivered can be due to effects of the PHY, MAC, or routing layers - they are all grouped together to count towards dropped packets. In

Smart energy meters	
Parameter	Value
Simulation duration	3600 s
Environment dimensions	250 m × 250 m
Number of nodes	240
Number of sources	240
Number of destinations	4
Source-dest pair selection	All sources, fixed sinks
Initial node distribution	Fixed
Node mobility model	N/A
Node transmission range	25 m
Packet size	512–1024 bytes
Per node Packet send rate	5 packets per hour
MASE Seismic Monitoring	
Value	Parameter
Simulation duration	900 s
Environment dimensions	50 km × 550 km
Number of nodes	100
Number of sources	100
Number of destinations	1
Source-dest pair selection	All sources, single fixed sink
Initial node distribution	Fixed
Node mobility model	N/A
Node transmission range	6500 m
Packet size	512–1024 bytes
Per node packet send rate	8 packets per second
Campus DTN	
Value	Parameter
Simulation duration	3600 s
Environment dimensions	1000 m × 1000 m
Number of nodes	19
Number of sources	19
Number of destinations	18
Source-dest pair selection	All sources, all sinks
Initial node distribution	Fixed
Node mobility model	Mobility trace
Node transmission range	250 m
Packet size	160 bytes
Per node packet send rate	2 packets per minute
San Francisco Taxi VANET	
Value	Parameter
Simulation duration	900 s
Environment dimensions	9500 m × 6230 m
Number of nodes	283
Number of sources	283
Number of destinations	10
Source-dest pair selection	Random sources, random fixed sinks
Initial node distribution	Fixed
Node mobility model	Mobility trace
Node transmission range	750 m
Packet size	512 bytes
Per node packet send rate	33, 66, 100, 333, 666, 1000 kb/s

Table 4. Sample parameters for real-world scenarios.

evaluating a routing protocol, it is important to know how many packets were dropped due to the physical and MAC layers, however those dropped packets should not be attributed to the routing protocol.

While our guidelines can be applied to any simulation platform, as an example, looking at the ns-3 v3.10 AODV routing protocols' source code, there are 12 separate places where packets can be dropped due to: non-existent routes, the output interface being mismatched, duplicate packets, invalid header type, creating loops, or TTL exceeded. When we evaluate a routing protocol, we only want to the number of packets dropped due to non-existent routes and TTL exceeded which accounts for 7 of the 12 drop cases in the code. Although over-counting in 5 places may seem insignificant, we are not including the number of events which would cause a packet to be dropped in the MAC or PHY layers, and the number of dropped packets due to the PHY/MAC layers could be quite substantial in a dense network.

Average Number of Source-Destination Distances in Tx Range Hops — The Average Number of Source-Destination Distance in Transmission Range Hops is a measure of the physical distance between source and destination nodes, which in turn determines the minimum number of hops a routing protocol could utilize to send data between the nodes. This metric takes the difference between the two nodes and divides it by the node's transmission range, T_r , to determine the line-of-sight minimum number of hops.

EXAMPLE SCENARIOS

We provide two static and two mobile example scenarios in this section with the intent of showing that it is possible to create scenarios that are applicable to real-world situations. Instead of arbitrarily choosing parameter values as we saw in our extended survey, we urge the importance of evaluating protocols with meaningful simulations. The parameters and values are listed in Table 4.

The first scenario is a section consisting of four city blocks (2×2) equipped with smart energy meters that are able to wirelessly communicate energy usage data back to a central data collector unit. The density of the environment will depend on how many houses or apartment units are built on a city block, but based on U.S. Census data we can estimate 15 housing units per acre, and we can estimate a medium-sized city block as $125 \text{ m} \times 125 \text{ m}$ which gives us approximately 60 housing units per city block. The energy data sent back to the hub is quite small at 512–1024 bytes for commands and meter registers, and it is also fairly infrequent seeing as the meters would not need to update more than a few times per hour. Since multiple meters are capable of sending data at the same time, the possibility of concurrent flows increases for the nodes closer to the central data collector unit. Since there is a central unit acting as a data sink, the destination node is always fixed, but the source nodes can be chosen at an estimated one every hour.

The second scenario is modeled after the

Middle America Subduction Experiment (MASE). MASE is a sensor network that monitors seismic activity in Mexico, and reports the data back to the collaborating research labs for processing. This network consists of 100 nodes spanning 550 km from Acapulco, Mexico to Tampico, Mexico, each of which is equipped with an 802.11 radio. Looking at a map of the locations of the nodes, we can estimate that the width of this environment is approximately 50 km since the nodes are laid across a fairly straight line. We were unable to find specific data regarding the traffic these nodes send, but we are able to estimate the values based on reports of 20–40 Mbytes of bandwidth per node per day with an estimated 5 minutes of transmit time per hour when in low power mode.

The third scenario uses the CRAWDAD KAIST mobility trace that were collected by recording the movement of students on the KAIST college campus [15]. The mobility trace is a collection of position recordings for 61 nodes over the course of over 22 hours and in a 3900 m × 8700 m area. Due to this being a mobility trace monitoring real people, not all of the 61 nodes were active over the course of the entire simulation, so we used a snapshot of the trace that represented the time with the highest amount of node movement. To select this section of time, we created a script to first discretize time into buckets, then count the number of movements made by unique nodes, and then select the sequence of buckets with the highest number of movements for unique nodes. This limited the scope of the trace to a section of one hour in length which is sufficient for this study because we intend to evaluate how routing protocols operate in Disruption Tolerant Networks (DTNs), not specifically to gather results about the entirety of this mobility trace.

Lastly, the fourth scenario uses the CRAWDAD mobility traces which captured the positions of San Francisco taxis while they were operating [16]. The full trace contains coordinates of 536 nodes spanning a 3600 km × 4200 km area over 575 h. This amount of mobility data not only is significantly more data than we need for this test, but we also suspect at least a few of the coordinates were stray positions that are intended to be filtered out since the size of the entire is many times larger than the San Francisco peninsula. To restrict the trace data, we targeted nodes in just the San Francisco peninsula which resulted in a 9500 m × 6230 m area. We also used a similar process like was done with the KAIST trace to reduce the amount of the trace to 900 s from the full 575 h, and this limited the number of active nodes in the area to 283. The period of 900 s that was selected was a sequence of time that had a large amount of unique node movement so as to keep the number of active nodes high.

FUTURE WORK

The goal of this article is to remind the community that there is a problem in the current way simulations of network protocols, specifically MANET protocols, are performed. In an effort to aid create better practices guiding network

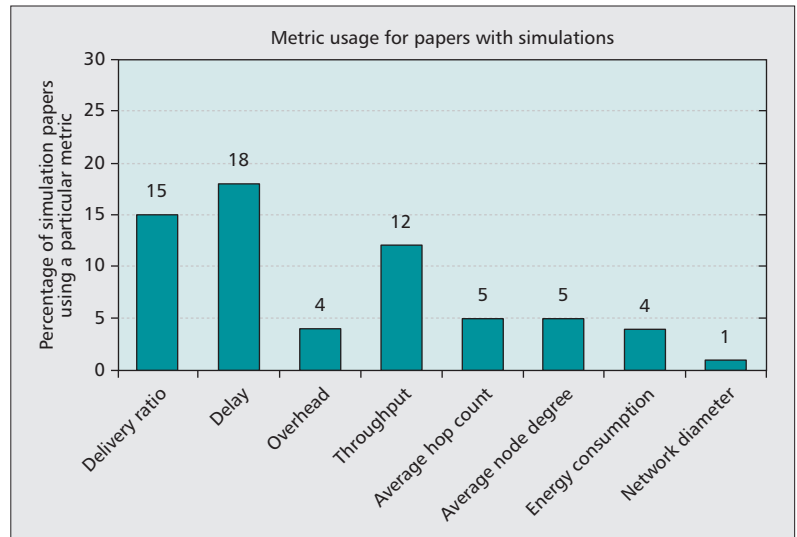


Figure 2. Percentage of papers found in MobiHoc 2006–2010 that used the indicated metrics.

protocol simulations, we will propose a set of concrete guidelines for constructing MANET routing protocol simulations. We have also been working on a suite of scripts designed to help benchmark MANET routing protocols. These benchmarks will thoroughly test a protocol’s performance under many different scenarios, and the results will be stored for later use. That way researchers wanting to compare their protocol’s performance to others will be able to simply look up another protocol’s scores — they will not need to rerun the benchmarks for an already benchmarked protocol. This approach will provide an unbiased approach to network protocol evaluation, and will also save researchers’ time by not having to develop their own test scenarios.

CONCLUSIONS

As intuitive as it seems, it is extremely important to document the settings of an environment for any scientific test, and this is no different for MANET routing protocol simulations. There have been several studies showing the differences between protocol implementations between simulators, so simply relying on default values in a simulator can produce wildly different results when comparing one simulator to another. We have shown that there is still a problem with omitting parameters and values used in simulations, and we have also shown that some of the values selected correspond to synthetic scenarios. We urge the MANET community to use realistic scenarios, and to improve upon the current state of recording and sharing simulation scenarios. We hope to see a drastic improvement in the next survey of MANET simulation papers.

REFERENCES

- [1] I. Stojmenovic, “Simulations in Wireless Sensor and Ad Hoc Networks: Matching and Advancing Models, Metrics, and Solutions,” *IEEE Commun. Mag.*, vol. 46, no. 12, Dec. 2008, pp. 102–7.
- [2] S. Kurkowski, T. Camp, and M. Colagrosso, “MANET

- Simulation Studies: The Incredibles," *SIGMOBILE Mob. Comp. Commun. Rev.*, col. 9, no. 4, Oct. 2005, pp. 50–61.
- [3] S. Kurkowski, T. Camp, and W. Navidi, "Two Standards for Rigorous MANET Routing Protocol Evaluation," *IEEE Int'l Conf. Mobile Ad Hoc and Sensor Systems*, Oct. 2006, pp. 256–66.
- [4] A. Munjal, T. Camp, and W.C. Navidi, "Constructing Rigorous MANET Simulation Scenarios with Realistic Mobility," *2010 Euro. Wireless Conf.*, Apr. 2010, pp. 817–24.
- [5] S. Naicken *et al.*, "The State of Peer-to-Peer Simulators and Simulations," *SIGCOMM Comp. Commun. Rev.*, vol. 37, no. 2, Mar. 2007, pp. 95–98.
- [6] D. Kidston and T. Kunz, "Towards Network Simulations Credibility: Lessons from Applying Five Key Principles," *IEEE MILCOM*, Nov. 2008, pp. 1–6.
- [7] P. Di *et al.*, "Towards Comparable Network Simulations," 2008.
- [8] T. R. Andel and A. Yasinsac, "On the Credibility of Manet Simulations," *Computer*, vol. 39, no. 7, July 2006, pp. 48–54.
- [9] Standard Performance Evaluation Corporation, <http://www.spec.org/>.
- [10] J. Yoon, M. Liu, and B. Noble, "Random Waypoint Considered Harmful," *IEEE INFOCOM'03*, vol. 2, Mar. 2003, pp. 1312–21.
- [11] N. Eagle, "MIT Media Lab: Reality Mining," 2009, <http://reality.media.mit.edu/>.
- [12] Dartmouth College, "A Community Resource for Archiving Wireless Data At Dartmouth," 2001, <http://crawdad.cs.dartmouth.edu/>.
- [13] B. A. Nunes and K. Obraczka, "On the Invariance of Spatial Node Density for Realistic Mobility Modeling," *Proc. 8th IEEE Int'l Conf. Mobile Ad Hoc and Sensor Systems*, 2011, pp. 322–31.
- [14] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "CRAWDAD Data Set EPFL/Mobility (v. 2009-02-24)," Feb. 2009, downloaded from <http://crawdad.cs.dartmouth.edu/epfl/mobility>.
- [15] I. Rhee *et al.*, "CRAWDAD Data Set NCSU/Mobilitymodels (v. 2009-07-23)," July 2009, downloaded from <http://crawdad.cs.dartmouth.edu/ncsu/mobilitymodels>.

ADDITIONAL READING

- [1] Pacific Gas and Electric Company, "SmartMeter System — How It Works," 2010, <http://www.pge.com/myhome/customerservice/smart-meter/howitworks/>.

BIOGRAPHIES

DANIEL HIRANANDANI (dhiranan@soe.ucsc.edu) is a Software Engineer in the Platforms Networking group at Google. Prior to joining Google, he earned an MS in Computer Engineering from the University of California, Santa Cruz in 2012, and a BS in Computer Engineering from California Polytechnic State University, San Luis Obispo in 2009. While his research mainly focuses on Mobile Ad Hoc Networks and localization/mapping using submersible robots, his research interests also include large-scale networks, embedded systems, and autonomous navigation.

KATIA OBRACZKA is Professor of Computer Engineering at UC Santa Cruz. Before joining UCSC, she held a research scientist position at USC's Information Sciences Institute and a joint appointment at USC's Computer Science Department. Her research interests span the areas of computer networks, distributed systems, and Internet information systems. She is the director of the Internetwork Research Group (i-NRG) at UCSC and has been a PI and a co-PI in a number of projects sponsored by government agencies (NSF, DARPA, NASA, ARO, DoE, AFOSR) as well as industry. Prof. Obraczka has edited one book, wrote a number of book chapters, and published over 200 technical papers in journals and conferences. She is a senior member of the IEEE.

J. J. GARCIA-LUNA-ACEVES [F] holds the Jack Baskin Endowed Chair of Computer Engineering at the University of California, Santa Cruz (UCSC), is Chair of the Computer Engineering Department, and is a Principal Scientist at the Xerox Palo Alto Research Center (PARC). He is a fellow of the ACM, and AAAS. He received the IEEE Computer Society Technical Achievement Award in 2011, and the IEEE Communications Society AHSN TC Technical Recognition Award in 2012.