Measurement-Based Approaches for Accurate Simulation of 802.11-based Wireless Networks

Anand Kashyap
Computer Science Dept.
Stony Brook University
Stony Brook, NY, USA
anand@cs.sunysb.edu

Samrat Ganguly
NEC Corporation of America
New York, NY, USA
samrat.ganguly@necam.com

Samir R. Das
Computer Science Dept.
Stony Brook University
Stony Brook, NY, USA
samir@cs.sunysb.edu

ABSTRACT

In this work, we address the issue of unrealistic simulations of wireless networks using a measurement-based approach. The idea is to use empirical modeling using measurement data as a mechanism to model physical layer behavior. We demonstrate the power of this approach for 802.11-based networks using ns2, a packet-level network simulator. Specifically, we develop two versions of the ns2 simulator that model the wireless physical layer with different levels of fidelity. In both versions, the deferral and reception model are built using measurements. For propagation modeling, one version uses direct measurements and the other uses an empirically derived model. In validation experiments with a 12-node mesh testbed, both these versions were found to be reasonably accurate (85 percentile errors within about 10% of the capacity) relative to regular simulations (85 percentile errors within 50% of capacity).

Categories and Subject Descriptors

I.6.5 [**Simulation and Modeling**]: Model Development—*Modeling methodologies*.; C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Experimentation, Measurement.

Keywords

Network simulator, wireless network capacity, wireless interference, 802.11.

1. INTRODUCTION

Simulation-based modeling is a useful tool for evaluating performance of network protocols. Simulations served the networking community well for wired networking regime. However, simulations for wireless networks have often been questioned [13, 6], primarily due to the lack of realistic lower layer models. However, the research community has not yet practiced serious validation exercises for wireless network simulators barring minor exceptions [14]. Our goal in this work is to revisit the issue of unrealistic simulation models of wireless networks for the lower layers, and address the problem using a new approach that uses measurement-based modeling.

Network simulators widely used in wireless networking literature such as ns-2 [4], qualnet [3], opnet [2] etc. implement the network protocol layers in the same fashion as in a real system. The upper layer implementations (such as transport and network) are fairly accurate. This is because they are implemented in software in a real system. This makes it easier to model them in the simulation software. This is also true for MAC-layer models as detailed specs and firmware implementations are available to a serious simulation modeler. However, the wireless physical layer has been hard to model. While theoretical models do exist, they make assumptions on the propagation environment and the interface characteristics and use various model parameters (e.g., path loss exponent) that are hard to instantiate. Also, often such models work at a much finer timescale (at the bit or symbol level, e.g.) while popular network simulators operate at a packet-level time scale. Making the timescale finer may cause a serious slowdown of the simulator eliminating the scalability benefit – one possible reason why such attempts have not been seriously pursued yet. On the other hand, research has shown that physical layer modeling can make serious impact on the upper layer protocol performance [19] thereby making realistic modeling all the more important.

Our goal in this paper is to propose measurement-based approaches to model the physical layer of protocol stack so that not only popular packet level simulators can still be used, but also the simulation accuracy is vastly improved. The approach is not simulator specific, but we have used ns-2 because of its popularity. Similarly, our work is not MAC/radio specific, but we focus in this paper on 802.11 because of its ubiquity. We identify three components that comprehensively capture the physical layer behavior in an 802.11-based network. They are (i) signal propagation model, (ii) carrier sensing model on the sender side, and (iii) packet reception model on the receiver side. We propose measurement-based approaches to model the above three components. The idea is to use measurements to preserve realism where analytical models are inadequate.

We validate the accuracy of the measurement-based approaches vis-a-vis direct experimentation on a 12-node 802.11-based indoor mesh network testbed. Our general conclusion is that the technique is very accurate when measurement

MSWiM'08, October 27–31, 2008, Vancouver, BC, Canada.

data from an actual testbed is available. When complete testbed is not available for measurements, measured data from a limited set of nodes can also be used for modeling using the proposed approach while providing high level accuracy compared to existing simulations approaches. Our hope is that this study will encourage the wireless networking research community to use measurement-based techniques for simulation studies. Wide adoption will also lead to reuse of measurement-based models making the approach very cost-effective in terms of effort.

The rest of the paper is organized as follows. In Section 2, we present a background of our approach. In Section 3, we present the actual measurement based models we use. Section 4 presents the validation of different physical layer modeling approaches with respect to measurements from a real network. We then present the related works in Section 5, and then conclude in Section 6.

2. APPROACH

The physical layer components in an 802.11 network simulator can be classified into three broad categories – (i) *radio propagation model*, (ii) *deferral or carrier sense model* on the sender side, and (iii) *packet reception model* on the receiver side. We describe our approach to handle them below.

2.1 Propagation

Typically, wireless network simulators assume a generic propagation model, such as free space model or two-ray ground reflection model coupled with a shadowing model [16] as in ns-2. Naturally, such a generic model may not be appropriate for the propagation environment to be evaluated. Further, parameters of such models (e.g., path loss exponents) still need to be instantiated. Our approach here is as follows.

a. *If a testbed is available*, we perform direct measurement on the testbed to determine propagation behavior. Here, the receiver simply measures the received signal strength (RSS) and no real modeling is performed. This requires only $O(N)$ measurements for an N node network. Each node can transmit a beacon and every other node simply measures the RSS. Commodity 802.11 interfaces allow such measurements.[1]

b. *If a testbed is not available (but a pair of network nodes are available)*, we model the propagation behavior using an empirical, measurement-based approach in the environment being considered. This is not unlike early work in cellular communications that gave rise to popular empirically derived models such as Okumura-Hata models [16]. A similar modeling approach has also been considered in outdoor 802.11-based networks with reasonable accuracy [7].

2.2 Deferral and Reception

Carrier sensing in 802.11 cards is implemented using a channel acquisition module, which determines whether the channel is idle for transmission. This is modeled in simulators by using a carrier sense threshold, and a received signal with higher power than this threshold makes the channel busy. It has been observed [15] that carrier sensing between a pair of nodes is not deterministic, and in practice, if a pair of nodes attempt to transmit simultaneously, the probability that one node defers due to other may be a value somewhere between 0 and 1.

Modeling the packet reception is harder. This depends on signal to interference plus noise ratio or SINR, where signal is the received signal power and interference is the aggregate of the interference powers received at the receiver. Interference is simply signal transmitted by any node other than the designated transmitter. Fundamentally, SINR affects the bit-error rate (BER) in a received packet [16]. The SINR vs. BER relationship typically depends on receiver design and modulation used. BER ultimately affects PER (packet-error rate) depending on the coding used. Note again the probabilistic nature of packet reception. Usually, there is a sharp fall in BER (and hence PER) with increasing SINR. Thus, often simulators simplify this by assuming a simple two-step function to model SINR vs. PER relationship. This essentially translates to the so-called *capture threshold*, signifying an SINR threshold needed for successful packet reception. Even when modulation/coding specific SINR vs. PER relationship can be used (the best case), it is unclear whether a universal theoretically based model would suffice for any interface.

2.3 Modeling Strategy

Direct measurements are possible for modeling the propagation behavior in Section 2.1 (using $O(N)$ measurements). However, similar direct measurements are not possible for modeling deferral or packet reception behavior, *even when a testbed is available*. The reason is that all possible subsets of transmitting nodes must be considered, requiring an exponential number of measurement steps. This requires us to take an empirical modeling approach that still only uses $O(N)$ measurement steps and the rest is done via modeling. The modeling part assumes that only aggregate interference power is important to determine deferral or reception, and not individual interference powers or number of interferers. Note that this assumption should be true in theory. We have indeed performed limited amount validations to test this out (reported in the next section).

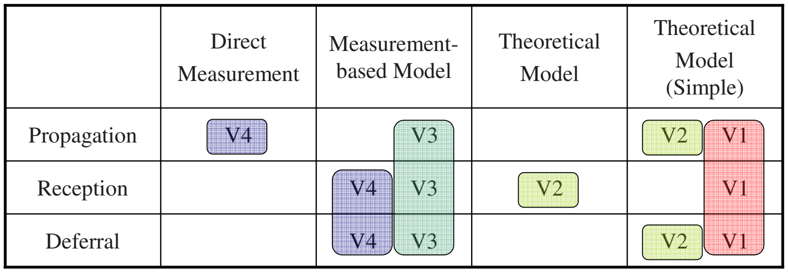
We develop several versions of the ns2 simulator, *only differing in the physical layer implementation*. To describe the simulators better, let us categorize the propagation, deferral and packet reception modeling in the simulators in 4 categories. See Figure 1. We name the simulator versions V1 to V4, with increasing complexity. V3 and V4 replace the entire physical layer by our measurement-based model. The difference in V3 and V4 is that in V4, direct RSS measurements are used to model propagation (note (a) in Section 2.1); while in V3, a model is used for propagation that is derived from measurements (note (b) in Section 2.1).

V1 and V2 use simpler models. V1 is very similar to the default ns-2 simulator. Here, the propagation model is a free space propagation model, reception is based on a SINR threshold,[2] and deferral is based on a carrier sense threshold. These thresholds are tuned using measurement data

---

[1] Note that there are subtleties here that commodity cards allow RSS measurements only when the packet is received correctly. Prior measurement studies indicated that impact of this is relatively minor [17].

[2] The default ns2 has an even simpler reception model, where it simply compares signal with one interferer only at a time. V1 makes it somewhat more realistic by using a true SINR computation.



	Direct Measurement	Measurement-based Model	Theoretical Model	Theoretical Model (Simple)
Propagation	V4	V3		V2 V1
Reception		V4 V3	V2	V1
Deferral		V4 V3		V2 V1

Figure 1: Versions of the simulators considered and the models used by them.

as a guide. V2 differs from V1 in that it uses a somewhat more sophisticated model for packet reception based on the theoretically derived PER vs. SINR curves [5].

3. MEASUREMENT-BASED MODELS

In this section, we present the measurement-based models we use for the simulators. All measurements were done on our experimental testbed consisting of 12 Dell Latitude D520 laptops running Linux 2.6.15 kernel. The testbed is located in one floor of an office-cum-lab environment. See Figure 2 for a network diagram. Each laptop uses a DLink AirPremier DWL-AG660 802.11/a/b/g PC card with Atheros AR5212 chipset. The Madwifi driver, Version 0.9.6 [1] is used. The cards are configured in ad hoc mode when used as transmitter, and in monitor mode, when used as receiver. RSS measurements use the appropriate field in the prism monitoring header which is obtained whenever a packet is captured when the card is in monitor mode. The value reported by Atheros cards is $10\log_{10}(S + I/N)$, where S is the signal strength and I is the interference, N is fixed at -95dBm (noise floor). All experiments reported here are done for 802.11b. We also did similar set of validations for 802.11a and have very similar experience. But we choose to present 802.11b results here as it gives longer range links and has a rich set of interferences in our testbed. The experiments are done in nights when interference from external 802.11 networks is expected to be minimal. All experiments are done at the lowest PHY-layer rate (1 Mbps) and with large (1400 bytes) packet sizes.

3.1 Modeling Propagation

Radio propagation in indoor environment is a complex phenomena. There are three main factors that play a role in determining the received signal power – path loss, shadowing and multipath fading [16]. At a high level, path loss describes the exponential decay of signal power with distance, with the exponent depending on the propagation environment. Shadowing describes random variation of path loss in similar propagation environment, commonly modeled by a Normal distribution in dB (log-normal shadowing). Following a similar modeling work [7] we ignore multipath fading due to its modeling complexity and impact only in small time and spatial scales.

Combining path loss and log-normal shadowing, we have

$$P_{dB}(d) = P_{dB}(d_0) - 10\alpha log_{10}\left(\frac{d}{d_0}\right) + X_\epsilon, \qquad (1)$$

where $P_{dB}(d)$ is the received signal power at distance d, d_0 is a reference distance where power measurement is already available, α is the path loss exponent, and X_ϵ is a Normal random variable in dB having a standard deviation of ϵ dB and zero mean. The path loss exponent α is 2 in free space, but is higher in a cluttered environment.

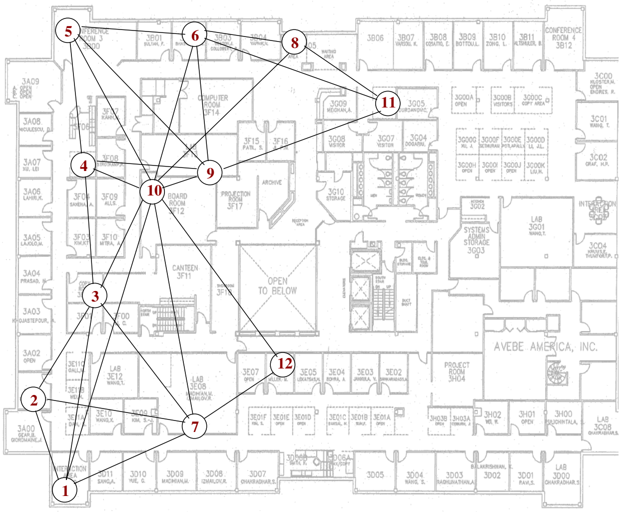


Figure 2: Locations of the nodes on the floor map and links with more than 90% delivery ratio. Width of the map is 60m.

We use an empirical method to estimate α and ϵ from measurement data following similar work in [7]. We collect average RSS values for each pair of nodes in the testbed from 132 separate measurements (12 transmitters × 11 receivers) and use least square linear regression to find the path loss exponent for our testbed environment. Figure 3(a) shows the scatterplot, and the fitted line, which gives the path loss exponent α as 4.66. Similarly, ϵ is estimated by fitting a

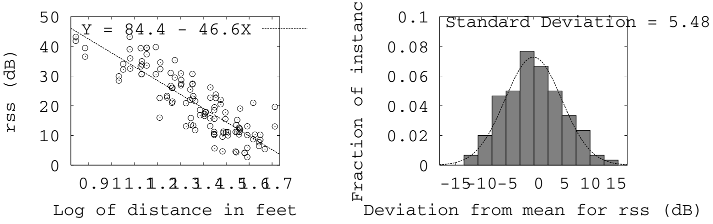
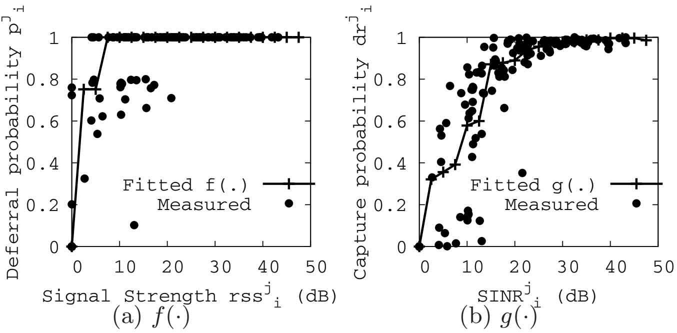


Figure 3: (a) Measurement data showing received signal strength vs. distance and also the least-square fit. (b) Empirical estimation of the shadowing model.



(a) $f(\cdot)$ (b) $g(\cdot)$

Figure 4: Determining (a) deferral and (b) packet reception probabilities.

Normal distribution for the error values in the above regression. See Figure 3(b). We get $\epsilon = 5.48$. We use this model in the simulators V3. Note that if a complete testbed is not available, but only a couple of nodes are available, we can still create this model by performing a large number of RSS measurements by placing just two nodes in different random locations in the test environment.

3.2 Modeling Deferral

The first step is to create an empirical relationship for the probability of deferral between two nodes based on received signal strengths. We express this relationship as a function $f(\cdot)$, such that $p^j_i = f(rss^j_i)$, where p^j_i is the (deferral) probability that node i defers to the transmission of node j and rss^j_i denotes the measured values of average signal strength of packets transmitted from node j and received at node i. We determine function $f(\cdot)$ simply by taking two nodes and positioning them in many random locations in the test environment, and then directly measuring the RSS values between them as well as the deferral probability.

The deferral probability is measured as follows. Both nodes attempt to broadcast UDP packets as fast as possible. Thus, they always have backlogged traffic. We measure the transmit rate (rate at which a node is transmitting packets on the air) of each node. We also measure transmit rate when the node is transmitting alone. The ratio of these two rates gives the deferral probability p. A large number of such measurements $< p, rss >$ are taken and are shown in the scatterplot of Figure 4(a). $f(\cdot)$ is estimated as the linear interpolation of average values of p for small buckets of rss values. Further, it is assumed that deferral probability p depends only on the sum of rss values if multiple transmitters are present. Thus,

$$p^Y_i = f\left(\sum_{j\in Y} rss^j_i\right), \tag{2}$$

where Y denotes a set of active transmitters.

3.3 Modeling Packet Reception

A similar approach is taken for modeling the packet reception behavior. Define *delivery ratio* dr^j_i from node j to node i as the fraction of packets received by i that are transmitted by j in the absence of any other interfering transmitter. Let us define $dr^j_i(Y)$ as the delivery ratio from j to i in presence of the set of interferers Y. Our first task is to model dr^j_i as $dr^j_i = g(rss^j_i/\text{noise})$. This simply relates packet reception (capture) probability to SNR, the ratio of the received signal strength and noise. Here rss^j_i denotes the average signal strength of packets received from j to i in absence of interference.

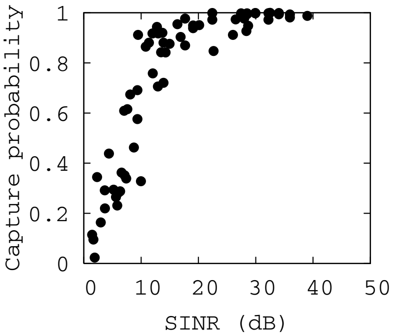


Figure 5: Capture probability versus SINR in presence of one interferer.

Once the function $g(\cdot)$ has been modeled, $dr^j_i(Y)$ can be expressed as follows:

$$dr^j_i(Y) = g\left(SINR^j_i(Y)\right), \tag{3}$$

where,

$$SINR^j_i(Y) = \frac{rss^j_i}{\sum_{k\in Y} rss^k_i + \text{noise}}. \tag{4}$$

As in the case of equation 2, the above equation also requires only pairwise measured rss values in the deployed network.

A set of experiments as before is devised to empirically model $g(\cdot)$. Two nodes are placed in many random locations. One of them transmits broadcast UDP packets and the other receives. The average dr and rss values are recorded at the receiver. The scatterplot in Figure 4(b) shows the experimentally obtained values. The function $g(\cdot)$ is obtained via interpolation as before. As stated before, these results are for the lowest PHY-layer rate (1 Mbps), and thus the $g(\cdot)$ function is specific to this data rate. Similar experiments must be done at all data rates to get the rate specific $g(\cdot)$ functions.

Note that the empirical technique above measures SINR without any interferer (thus, actually SNR) with an assumed noise floor (-95dBm). We have also validated that indeed when one or more interferer is added, the function $g(\cdot)$ estimated above still holds. Figure 5 shows the experimentally obtained delivery ratio vs. SINR scatterplot in the presence of 1 interferer. Note the similarity of this plot with Figure 4(b). This provides credence to our approach that function $g(\cdot)$ can be modeled using measurements without any interferer, and thus requires only $O(N)$ measurement steps.

4. EVALUATION

We evaluate the accuracy of the simulators on the target network – the 12-node mesh testbed described before. Average RSS (rss) and delivery ratio (dr) values for all link pairs in the network are collected. Here, each node takes turn to transmit UDP broadcast packets and every other node measures the average rss and dr values. This process is similar to measurements reported in [15, 17]. This takes $O(N)$ steps for an N node network. The rss measurements are used to seed simulator V4, while the rss and dr measure-

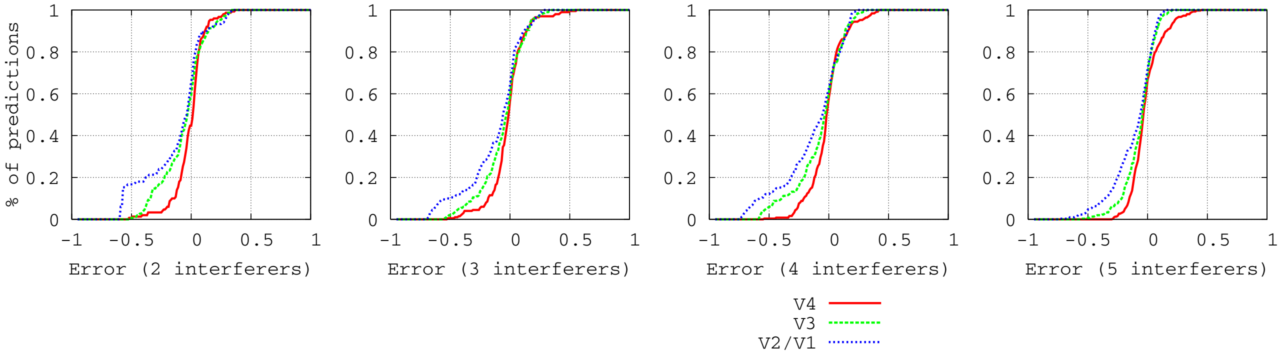


Figure 6: CDF of error between the estimated and measured transmission capacity of senders.

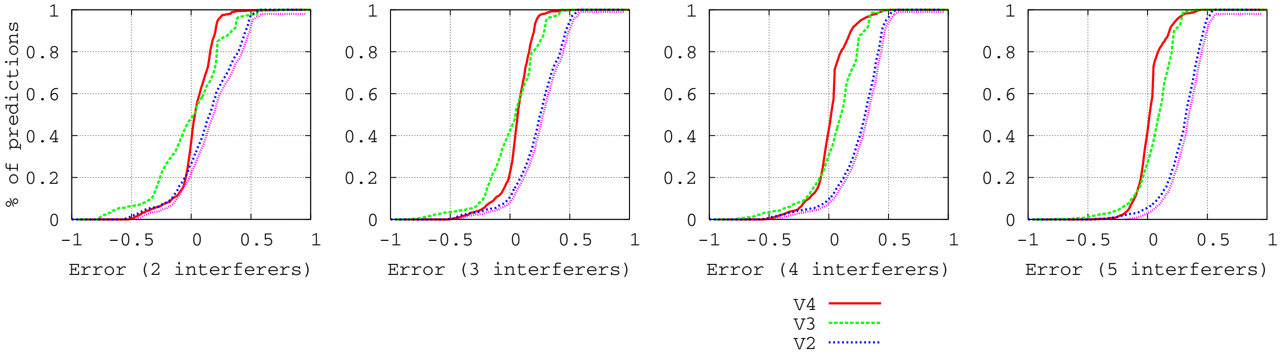


Figure 7: CDF of the error between the estimated and measured throughput capacity on links.

ments are used to create the deferral and reception model for simulators V4 and V3.

For validation, we perform direct measurements on the testbed to evaluate link capacities and then compare them with those estimated by the various versions of the simulators. In each validation experiment, n nodes are chosen from the testbed as transmitters while the remaining $12-n$ nodes act as receivers. Each transmitter then broadcasts packets as fast as possible (to model saturated traffic) for 60 seconds. At the end of this time period, the throughput on each one of the $n(12-n)$ links is measured by counting the number of packets received from each sender. For each such link, there are $n-1$ interferers. We also measure the transmission capacity (number of packets *actually transmitted* in the air per second) for each transmitter. This quantity is reported by the card to the Madwifi driver.

We have performed validation experiments with up to 5 interferers. When $n = 2$, it is a single interferer scenario. Here, we have measured all possible combinations of such scenarios, which require 66 experiments, and provide data for 132 transmitters, and 1320 links. When $3 \leq n \leq 6$, we randomly pick 50 random sets of n transmitters each, which results in data for $50n$ transmitters, and $50n(12-n)$ links. Overall, we have performed 266 sets of experiments resulting in 7820 data points in the plots to be presented next.

Figure 6 shows the CDF of the absolute error (i.e., estimated − measured) in the sender side transmission capacity for the various simulators. We present capacity normalized to the channel capacity. Since V1 and V2 use the same deferral and propagation model, the transmission capacity of these two simulators are identical. Note that V4 is quite accurate – the error is within 10% of the channel capacity 85% of the times. V3 is less accurate than V4 (the error is within 15% of capacity 85% of the times), because V3 uses a model for propagation rather than using direct measurement. V1 and V2 *underestimate* the transmission capacity significantly, likely because they model a weaker path loss. This results in more deferral and lower transmission capacity.

Exactly similarly, we present the absolute error between estimated and measured link throughput capacities at the receiver side in Figure 7. Once again, note the excellent accuracy for V4 followed by V3. The 85 percentile error for V4 and V3 is 10% and 15% of capacity, respectively. Note again V2 and V1 provide very poor estimation, *overestimating* the capacity this time. In the case for V2 and V1, the throughput capacity is almost the same as the transmission capacity as collisions rarely happen because of almost perfect deferral. In reality, however, many more packets are actually transmitted, but many of them actually lead to collisions leading to much poorer received throughput.

The take-home message from these results is that careful measurement-based modeling can be successfully used to develop accurate simulators (V4 and V3). When measurements are not used, even when best possible strategies are used in the simulation models (e.g., V2), the errors are very high. For example, for estimating throughput capacity, for 85% of the scenarios, the error in V2 increases to 50% of the channel capacity.

5. RELATED WORK

In [13], the authors describe unrealistic assumptions often made in wireless network simulators. They also develop a simulator that they validate against real experiments; however they report experiments related to propagation modeling only. Several emulation approaches are described to validate wireless ad hoc network simulations in [10]. However, here comparisons against real networks are not reported. In a recent comprehensive article [6] the authors survey many questionable practices for simulating mobile ad hoc networks. They note inadequate modeling of protocols and lack of validations as two major issues. They also note other issues such as improper documentation, or lack of statistical validity that are not explored in our paper. In [14], a validation approach has been developed using direct execution simulators for ad hoc networks.

In [9], the effect of detail in wireless network simulations and how they influence the conclusions are studied. In [19], a careful study is done using different simulators that shows how the details in physical layer modeling can impact upper layer protocol performance in a simulator. Physical layer emulations [11] and various hybrid approaches [20] have recently been promoted to impart realism to modeling studies. However, they are quite complex, require significant amount of hardware and are yet to be widely adopted.

The measurement approaches discussed in the paper have similarities with several recent works, such as [17, 7, 15, 8, 12] for 802.11 networks and [18] for Berkeley mote-based networks. These papers emphasize the significance of using measurements over analytical modeling. Some of these papers also promote using just pairwise signal strength measurements between nodes to model interference and its impact. We utilize these ideas in our paper in the context of creating an accurate and realistic wireless network simulator.

6. CONCLUSIONS

In this work, we have demonstrated that empirical modeling of the physical layer is necessary in building more accurate wireless network simulators. We have specifically focused on 802.11 and developed two versions of the popular ns-2 simulator that model the wireless physical layer with different levels of fidelity. In both versions, the deferral and reception model are built using measurements. For the propagation modeling, one version (V4) uses direct measurements and the other (V3) uses a modeling approach. In validation experiments over a 12-node mesh testbed, both these versions were found to be reasonably accurate (85 percentile errors about 10% of capacity). Simulation errors in more traditional simulation models were found to be unacceptably high (85 percentile error within about 50% of capacity). Our future work will focus on improving the error margins and validating our simulators with unicast traffic, with relayed traffic, etc. The eventual goal is to make such simulators using various measurement data sets available to the research community for evaluating protocol performance.

Acknowledgments

This work was partially supported by the following NSF grants: CNS-0751121, CNS-0721665, CNS-0721455, CNS-0713186, CNS-0519734 and CNS-0423460.

7. REFERENCES

[1] Multiband Atheros Driver for WiFi (MADWIFI). http://sourceforge.net/projects/madwifi/.

[2] OpNet. http://opnet.com.

[3] QualNet. http://scalable-networks.com.

[4] The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns.

[5] HFA3863 Data Sheet: Direct Sequence Spread Spectrum Baseband Processor with Rake Receiver and Equalizer. Intersil Corporation, 2000.

[6] TR Andel and A. Yasinsac. On the credibility of manet simulations. *Computer*, 39(7):48–54, 2006.

[7] Joseph Camp, Joshua Robinson, Christopher Steger, and Edward Knightly. Measurement driven deployment of a two-tier urban mesh access network. In *Proc. ACM MobiSys Conference*, 2006.

[8] S. Das, D. Koutsonikolas, Y. Hu, and D. Peroulis. Characterizing multi-way interference in wireless mesh networks. In *Proc. ACM WINTECH Workshop*, 2005.

[9] J. Heidemann, N. Bulusu, J. Elson, C. Intanagonwiwat, K. Lan, Y. Xu, W. Ye, D. Estrin, and R. Govindan. Effects of detail in wireless network simulation. In *Proceedings of the SCS Multiconference on Distributed Simulation*, 2001.

[10] D. Johnson. Validation of wireless and mobile network models and simulation. In *Proceedings of the DARPA/NIST Network Simulation Validation Workshop, Fairfax, Virginia, USA*, 1999.

[11] Glenn Judd and Peter Steenkiste. Repeatable and realistic wireless experimentation through physical emulation. *SIGCOMM Comput. Commun. Rev.*, 34(1):63–68, 2004.

[12] Anand Kashyap, Samrat Ganguly, and Samir R. Das. A measurement-based approach to modeling link capacity in 802.11-based wireless networks. In *Proceedings of ACM MobiCom*, pages 242–253, 2007.

[13] D. Kotz, C. Newport, R.S. Gray, J. Liu, Y. Yuan, and C. Elliott. Experimental evaluation of wireless simulation assumptions. *Proc. ACM MSWiM Symposium*, pages 78–82, 2004.

[14] J. Liu, Y. Yuan, D.M. Nicol, R.S. Gray, C.C. Newport, D. Kotz, and L.F. Perrone. Simulation validation using direct execution of wireless Ad-Hoc routing protocols. In *Proc. PADS Workshop*, pages 7–16, 2004.

[15] J. Padhye, S. Agarwal, V. Padmanabhan, L. Qiu, A. Rao, and B. Zill. Estimation of link interference in static multi-hop wireless networks. In *IMC*, 2005.

[16] Theodore S. Rappaport. *Wireless Communications: Principles and Practice*. IEEE Press, Piscataway, NJ, USA, 1996.

[17] Charles Reis, Ratul Mahajan, Maya Rodrig, David Wetherall, and John Zahorjan. Measurement-based models of delivery and interference in static wireless networks. In *SIGCOMM*, 2006.

[18] D. Son, B. Krishnamachari, and J. Heidemann. Experimental study of concurrent transmission in wireless sensor networks. In *Proc. ACM SenSys*, 2006.

[19] Mineo Takai, Jay Martin, and Rajive Bagrodia. Effects of wireless physical layer modeling in mobile ad hoc networks. In *Proc. ACM MobiHoc*, 2001.

[20] J. Zhou, Z. Ji, M. Varshney, Z. Xu, Y. Yang, M. Marina, and R. Bagrodia. WHYNET: a hybrid testbed for large-scale, heterogeneous and adaptive wireless networks. In *Proc. WINTECH*, 2006.