A Quantitative Evaluation of the MeshTest Wireless Testbed [*]

Brenton Walker Charles Clancy
Laboratory for Telecommunications Sciences
US Department of Defense
brenton@LTSnet.net clancy@LTSnet.net

ABSTRACT

The MeshTest testbed is designed to subject wireless devices and protocols to realistic and repeatable mobile scenarios, including multi-hop and disconnected topologies. The testbed makes it possible to efficiently test real implementations of ad-hoc and delay-tolerant routing protocols. With the completion of its mobility control software, the testbed has recently become operational for mobile scenarios. In this paper we report on some of the challenges encountered in building the testbed and implementing the control software, switch calibration, and some basic testbed results. We compare the basic results achieved on the testbed to results from live tests with comparable real systems.

Since we first introduced plans for the MeshTest testbed, multiple researchers have expressed an interest in building similar setups. Therefore we believe it is valuable to share our experiences and lessons learned so far in bringing this testbed into operation.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication; C.4 [**Performance of Systems**]: Measurement techniques

General Terms

Design, Experimentation, Measurement

1. INTRODUCTION

Mobile Ad-Hoc wireless networks have become one of the most heavily studied new wireless technologies. They can potentially be a very cost-effective mechanism for supporting large, distributed communications, with applications ranging from mobile wireless broadband to sensor networks. However, realistic and repeatable simulation environments are difficult to come by.

Network simulators, such as ns-2 and OPNET, offer repeatability, but it is difficult to quantify their accuracy. Their RF propagation and interference models are based on vast oversimplifications of electromagnetic wave theory. They also rely on simplified implementations of protocols that do not always accurately reflect the issues that may arise with a real operating system and real hardware.

At the other extreme, one can build real implementations and take them out into the field and run live tests, for example [1]. If one wants to experiment with mobility, however, such tests can be expensive and time-consuming, and are difficult to choreograph and control. Test management and data gathering will also be an issue, as there may not be an out-of-band control channel dedicated to these purposes.

Because of these difficulties many researchers have built laboratory-based testbeds which offer varying degrees of realism. These typically involve many wireless devices in a relatively small area all communicating over the air [6, 3]. But these too have shortcomings. In particular, it can be very difficult to create reproducible, multi-hop networks with such short distances between nodes. Also, testing mobility either involves physically moving devices around the room or using simulation techniques with questionable accuracy.

More accurate wireless test environments require a hybrid of the two. One must be able to reproducibly create arbitrary topologies that more accurately reflect real physics. This can be done by either making more sophisticated simulators [5] or more simulation-like testbeds. We have chosen the latter approach. A good overview of prominent wireless testbeds and how well they meet researchers' needs is given in [4].

1.1 The Theory Behind the Testbed

MeshTest consists of a rack of 12 computers in shielded enclosures, an RF matrix switch, and a server that provides experiment control, as depicted in Figure 1. The RF from each computer's WiFi card is cabled through the enclosures and into the matrix switch. The enclosures prevent inadvertent cross-talk, and the matrix switch allows us to arbitrarily control the attenuation between the devices.

Figure 2 shows the logical construction of an $n \times b$ switch. It has n inputs that connect through nb digital attenuators to b buses. Each bus has a direct, unattenuated, external connection. Note that the RF switch only simulates inter-

[*]This work was funded by the Laboratory for Telecommunications Sciences, US Department of Defense, and supported by the University of Maryland and Rutgers University. The opinions expressed in this paper reflect those of the authors, and do not necessarily represent those of the Department of Defense or US Federal Government.

TRIDENTCOM 2008, 17th − 20th Mar 2008, Innsbruck, Austria.
ISBN 978-963-9799-24-0
DOI 10.4108/icst.tridentcom.2008.3132

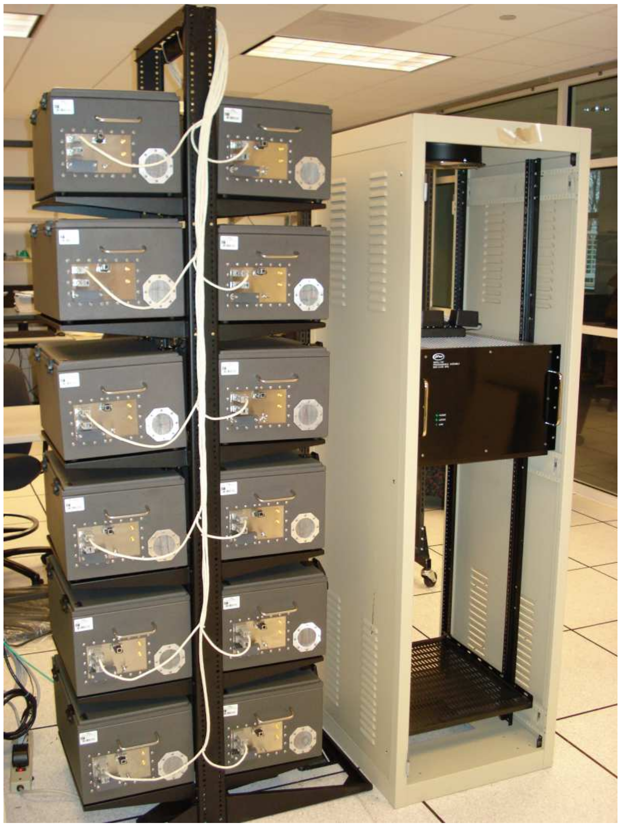


Figure 1: Assembly and connection of the shielded enclosures and RF switch.

node channel loss, and not propagation times. Propagation delay can potentially be simulated by delaying packet processing in software at the receiver.

While any device, from cellular telephones to software-defined radios, may be placed into the enclosures, the default configuration involves 802.11-based computers. Through a partnership with Rutgers, nodes and simulation management software from their ORBIT testbed [6] have been acquired.

Programming the digital attenuator settings for the matrix switch is not as obvious as it may seem. In particular, taking an arbitrary physical arrangement of devices one can easily compute a matrix, L, of inter-node attenuations, but adapting these values for the switch's matrix of attenuators can be challenging. In Figure 2, we can see that n nodes connect through nb attenuators to b buses. Let S be an $n \times b$ matrix representing the settings of the nb attenuators. In [2] we show that finding appropriate attenuator settings is equivalent to finding S such that

$$\Lambda . * \, S^T S = L \qquad (1)$$

where $.*$ is MATLAB notation for entry-wise multiplication of matrices, rather than standard matrix multiplication, and Λ is the insertion loss of the switch.

In [2] we also show that one can use simulated annealing [7] to compute an approximate decomposition, S, for a variety of scenarios, including static topologies, mobile topologies, and situations that involve multi-path fading.

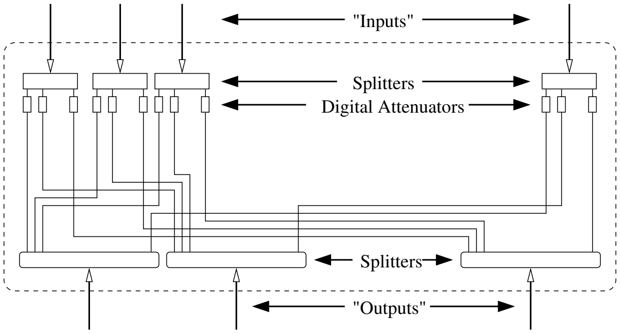


Figure 2: RF matrix switch diagram, showing n upper I/O ports, b lower I/O ports, and nb Ethernet-controlled digital attenuators with ranges 0-127 dB

2. THE RF MATRIX SWITCH

The heart of the testbed is the Matrix Switch of programmable attenuators. In designing the testbed we made a number of assumptions about how the switch would behave; some of which were valid and some of which needed adjustment. The key properties which we discuss here are:

- The accuracy of the programmable attenuators.
- The RF isolation between pairs of inputs.
- The insertion loss between pairs of inputs.
- The validity of our matrix multiplication model for combining signals.
- The effect of phase discontinuities caused by our digital attenuators

With both the accuracy of the attenuators and the RF isolation between inputs the switch performed as we expected. We anticipated that the programmable attenuators in the switch would be fairly accurate, and they were. We spot-checked the attenuations over a number of paths using a signal generator and spectrum analyzer. Also, so far we have been unable to detect any RF leakage above the noise floor. We have found that nodes in the shielded enclosures which are not connected through unattenuated paths are unable to detect one another.

2.1 Terminology

Figure 2 is a rough diagram of the matrix switch's construction. The switch was intended to combine up to n =16 signals, supplied on the **"Inputs"** out to b =4 **"Outputs"**, as labeled on the figure. For our applications we attach nodes exclusively to the "Inputs" and do not use the "Outputs", except occasionally for measurements and calibration. Our long term plan involves using the "Outputs" to attach multiple matrix switches together to form a much larger testbed.

For a matrix switch with b buses there are b different **paths** a signal can take between any pair of inputs. There are $(nb)^2$ total paths through the switch. The attenuation over any path is controlled by the two attenuators it passes through.

2.2 Insertion Loss

By "insertion loss" we mean the ratio of the power supplied on input i to the power propagated out through input j when both attenuators along a single path from i to j are set to 0dB. Insertion loss is the cumulative result of a number of factors: non-ideal connectors and cabling both inside and outside the switch, non-ideal attenuators and splitters, and energy division in the splitters.

We measured insertion loss over individual paths by feeding an unmodulated 2.4GHz sine wave of known power into input i, attaching a spectrum analyzer to another input, j, and setting the two attenuators on a single path between them to zero. The difference between the input power and output power is the insertion loss for that pair of inputs over that path.

We found that the insertion loss between all pairs of inputs was approximately 45dB, indicating that signals transmitted between all pairs of inputs traverse essentially the same type of path. We also observed that putting a terminator on an output jack effectively eliminated any signal on any path through the splitter connected to that output. We concluded that almost all of the energy propagated back to the inputs is due to reflection at the outputs. This leads to insertion losses that are higher than anticipated, but fairly uniform over different pairs of inputs.

An insertion loss of 45dB may seem severe, but at $2.4GHz$ it is equivalent to the free-space path loss over 1.8 meters, and does not significantly limit the variety of scenarios we can simulate on the testbed.

2.3 The Matrix Multiplication Model

Our algorithm for deriving attenuator settings that approximate a given physical scenario is based on a mathematical abstraction that models the matrix of attenuations achieved between pairs of switch inputs as a product of the matrix of attenuator settings with its transpose. A key assumption of this model is that signals following different paths through the switch will constructively interfere, resulting in a more powerful signal at the other inputs. For this to happen the combined signals have to be in phase (or close to it). We found that this is not always the case. On the other hand, given the nature of the typical attenuator settings our algorithm computes, we have concluded that even in the worst cases any destructive interference is well within the margin of error one would expect from fading.

It is not surprising that signals following different paths through the matrix switch would come out with differing phase. At 2.4GHz one wavelength is only $\frac{1}{8}$ meters, so a cable length difference of 6 cm would put the two signals in opposite phase. Also, digital attenuators will have an effect on the phase which varies with the attenuation applied.

Combining signals that are out of phase leads to destructive interference which manifests itself when we open up multiple paths between a pair of inputs. We identified several pairs of inputs which exhibited particularly good signal combining properties, and two pairs of inputs and paths which were particularly bad. By attaching a signal generator to an input and a high-frequency oscilloscope to the outputs corresponding to suspect paths, we were able to observe that the signals on the interfering paths were out of phase by as much as $15-20^\circ$ at the output, whereas the signals on the constructive paths were very close to in phase.

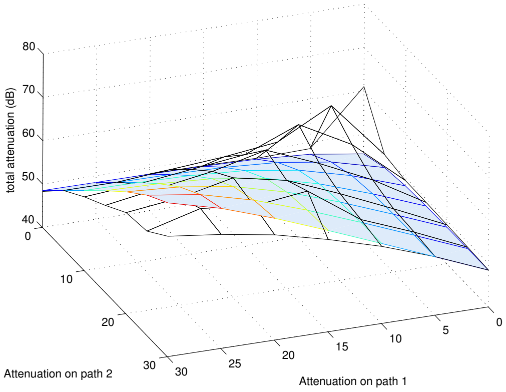


Figure 3: The results of balancing signal power between two paths which are out of phase. The light blue surface is the theoretically correct total attenuation and the black wire frame is experimentally measured attenuations. The interference is worst when the power on the two paths is the same. In practice the loudest path will be 10+dB higher than the others, and any interference will be negligible.

Fortunately, the attenuator settings produced by our factorization algorithm tend to concentrate most of the energy between any particular pair of inputs onto one path through the switch. Since the differences in attenuation between the different paths are typically 10dB or more, even in the worst cases the stronger signal dominates any potential interfering signals. Figure 3 shows the received signal power for varying attenuation imbalances along two different paths for one particularly troublesome pair of inputs.

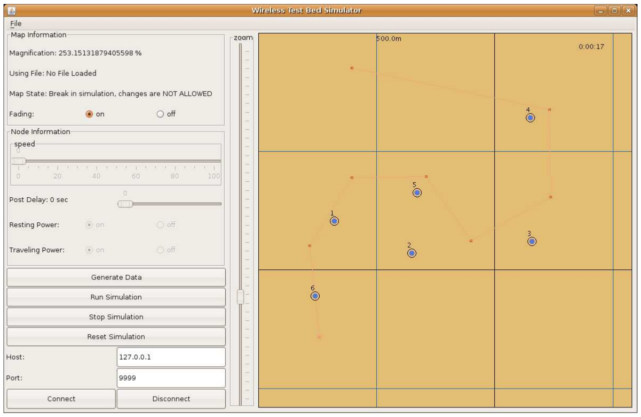
2.4 Phase Discontinuities

Our digital attenuators set their attenuation by switching fixed attenuators in and out of the path of the signal. If such a switch occurs during a transmission, it will almost certainly cause a phase discontinuity, even for small changes in attenuation. This could potentially disrupt modulation schemes such as OFDM. The alternative would be to use analog devices whose attenuation is controlled by a continuously variable voltage. Unfortunately such devices do not have the dynamic range we require, and would each require an individually calibrated voltage control.

We used iperf [9] to transmit packets between two nodes and report the number of packet drops over each 10s interval. We found 31000kbps to be the highest rate at which we could consistently maintain a median packet drop rate of 0 over a single path with no attenuator switching. To test the practical effects of phase discontinuities, we repeatedly switched a single attenuator along that path between 15dB and 16dB at regular intervals. We found that for switching intervals of less than 0.01s, the number of packet drops was related to the switching rate. On the other hand, even in the worst cases, simulated fading requires attenuator updates every $0.1s$. In this range the median packet drop rate under full load was one packet per $10s$. We conclude that for our purposes the phase discontinuities caused by the digital attenuators will not interfere with testbed experiments.

3. SWITCH CONTROL SOFTWARE

The MeshTest testbed uses ORBIT's testbed management software to control the nodes. The additional piece neces-



Magnification: 253.15131879405598 %

Using File: No File Loaded

Map State: Break in simulation, changes are NOT ALLOWED

Fading: ● on ○ off

Post Delay: 0 sec

Resting Power: ● on ○ off

Traveling Power: ● on ○ off

Host: 127.0.0.1

Port: 9999

Figure 4: A screen-shot of the mobility management program. The program allows the user to construct a mobility scenario, computes the attenuator settings and sends them to the switch during an experiment.

sary to make MeshTest work is software to map physical arrangements of nodes to the appropriate attenuator settings, and upload those settings to the switch in real-time during an experiment. Over the summer, students built a GUI-based program to perform these functions.

3.1 Performance of the Software

The switch control software must take a matrix of desired path losses and produce switch settings which approximate the desired attenuations. We have shown that this task is equivalent to finding an approximate solution to (1). In a mobile scenario the attenuator settings must be updated to reflect the changing node positions at least every 2-3 seconds. Each new set of attenuator settings requires us to do a simulated annealing climb to factor a new path-loss matrix. Our prototype algorithm was built in MATLAB and sometimes took several minutes to converge to a solution. Such computational demands would be impractical for an operational testbed. By modifying the search to better reflect the physical metaphor underlying simulated annealing, and experimenting with a variety of different search parameters, we arrived at a factorization algorithm that gives acceptable matrix factorizations in a few seconds.

We evaluate the quality of the switch settings by computing the difference between the desired path loss matrix and the attenuations achieved by the switch settings. Specifically we look at the sum of the non-diagonal entries of

$$\Delta = \left| -10\log(L) + 10\log(\Lambda .* S^T S) \right| \qquad (2)$$

divided by $2n(n-1)$. This represents the absolute average error between corresponding entries in L and $(\Lambda .* S^T S)$.

Figure 5 shows the mean error as a function of the number of nodes in a 1km by 1km square. The approximation becomes slightly worse as more nodes are added, but the results are still excellent and well within the margin of error expected from fading. The mean error for two nodes is higher than one might expect because there is only one attenuation to approximate, and rounding errors (the switch attenuators only take integer values) push this up to about 0.5dB.

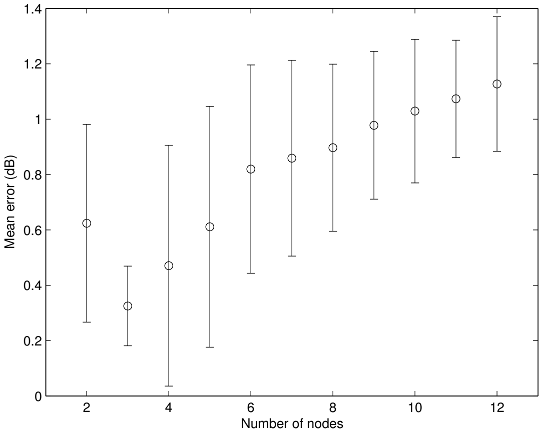


Figure 5: The mean error in attenuations between any pair of nodes as a function of the number of nodes in the experiment. Nodes were placed uniformly randomly in a 1km by 1km square. Results are the mean of 100 independent trials. Error bars are one standard deviation.

3.2 Simulating the Effects of Fading

The switch control software also gives the user the option to run the experiment with "fading" enabled. In the real world, fading refers to random fluctuations in signal quality due to multipath, phase issues, transient obstructions, atmospheric variations, or movement. We simulate fading by randomly perturbing the attenuations experienced between nodes according to a log-normal random process as described in [2]. Specifically, if $l_{i,j}$ is the expected path loss between nodes i and j, our method effectively produces an actual attenuation of $X \cdot l_{i,j}$, where X is a log-normal random variable with mean zero, and variance appropriate to the environment being simulated. For example in an indoor environment we model fading with variance of ±10dB.

Once the attenuator settings are computed for a given physical node arrangement we can apply our random perturbations to it without refactoring the path loss matrix. When fading is enabled we upload newly perturbed attenuator settings to the switch every $0.1s$, which is a reasonable coherence time for signals undergoing a low Doppler spread.

4. PRELIMINARY EXPERIMENTS

We and other researchers [8] have begun using the testbed for a variety of experiments. Here we describe some very basic tests which keep the focus on the performance of the testbed. In the first two tests we simply program all the nodes to act as 802.11 base stations and broadcast beacons while listening for the other nodes' beacons. The PRISM header attached to the received beacons contains signal strength and RSSI information, which is the metric we focus on. In these experiments the fields extracted were:

- time since the beginning of the trace
- source of the beacon
- signal strength and RSSI (PRISM header)
- noise level (PRISM header)

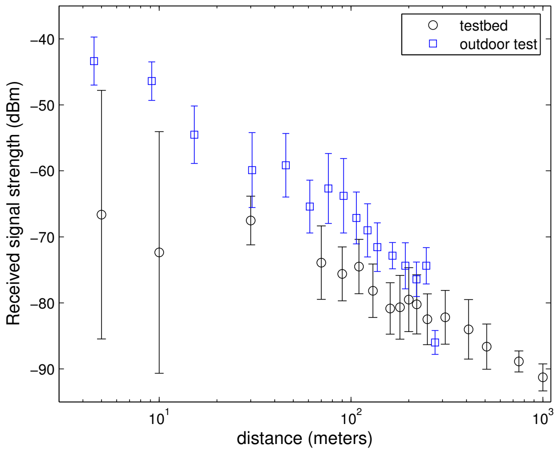


Figure 6: Signal strength results from both the testbed and the field test. The results show that the conditions of the field test were more favorable than our simulated path-loss model assumes. The underlying variation in signal strength with distance appears the same for distances over about 10m.

The quantities recorded in the PRISM header are not particularly precise measures of channel quality. One problem is that the nodes' Atheros-based wireless cards have an automatic gain control which cannot be disabled. Also the transmitters' output power is not necessarily constant and may depend on the noise environment. In practice, however, we do observe variations in signal strength with distance. In order to check the realism of the testbed we need to see if the variations experienced by nodes in the testbed are comparable to variations experienced by nodes in real life.

We describe three types of tests here. In the first type of test we position a pair of nodes a certain simulated distance apart and track the signal quality. We compare the results of this test to results from comparable hardware in an outdoor environment. The second type of test is what we call the "drive-by" test. A series of fixed access points are placed in the simulated space and a mobile node passes by them at a fixed speed, monitoring the signal strength of each one. In the third type of test we position several nodes in close proximity, program them to send data at a constant rate, and observe the drop in per-node throughput as more nodes begin transmitting.

4.1 Stationary Measurements

In this set of experiments we placed a pair of stationary nodes a certain (simulated) distance apart and let them transmit and record beacons for 10 seconds. We repeated the experiment nine times for varying distances. In each test we placed ten other nodes (for a total of twelve) randomly around a 500m×500m square in the simulated space. The other ten nodes were not activated. The purpose was just to make the attenuator settings as difficult as possible to compute. Between individual tests we would jumble the inactive nodes and re-compute the attenuator settings.

We replicated this experiment outdoors using two Linux laptops with Atheros-based 802.11 cards. One was set up to broadcast probes and the other was set up to listen for that station's probes. We measured off distances from 15ft to 900ft in an open field with few obstructions and at several fixed distances took two 10 second samples. The transmitting laptop was held at a fixed height of 3ft off the ground and was on the top of a slight hill. The laptops were always held so that the users were facing each other.

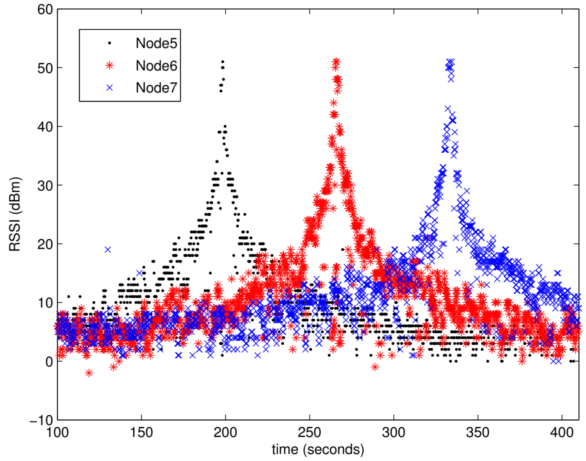


Figure 7: RSSI results vs time for the beacons received by the mobile node as it passes by a line of nodes acting as access points. The results highlight the consistency of the testbed.

Figure 6 shows signal strength vs distance results for this set of experiments. The results show that the conditions of the field test were more favorable than our path loss model predicted. We believe this is mostly because the laptops have built-in antennas which each give a couple dB gain. Differences like this just represent a linear adjustment to the target path-loss matrix. The experiment is a success because the underlying variation in signal strength with distance matches fairly well. The noise measurements for both experiments were consistently about −95dBm, so plotting the RSSI values is redundant.

The two data points at short distances show, however, that the testbed has difficulty simulating path loss over short distances (less than 10m). Because of insertion loss, the matrix switch cannot produce attenuations of less than about 45dB. This is not a serious obstacle to the testbed's intended purpose of testing ad-hoc and delay-tolerant wireless protocols and devices.

4.2 The "Drive-by" Experiments

This experiment was carried out on the testbed. Nodes acting as base stations, sending out beacons every 0.1 seconds were placed along a line at 500m intervals. Another node moved past them at 30km/h, recording the beacons. There were 5 active stationary nodes and one mobile node. The entire experiment ran for about six minutes. Figure 7 shows the RSSI values of the beacons received by the mobile node from three of the base stations vs time.

The signals from the base stations clearly get stronger as the mobile node approaches and then fade out as it moves away. The results demonstrate the consistency of the nodes, the attenuator settings generated, and the behavior of the

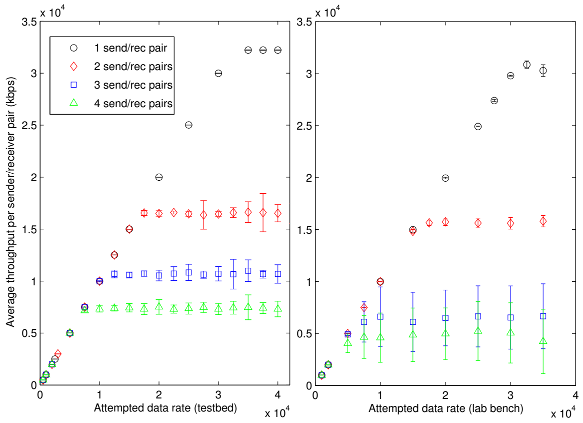


Figure 8: Average throughput per sender/receiver pair as a function of the attempted transmission rate for varying numbers of sender/receiver pairs for the experiment described in section 4.3. The effective throughput for each pair goes down as the number of pairs sharing the medium increases.

matrix switch regardless of which inputs the nodes are connected to. They also show again how the testbed can produce the appropriate attenuations between may nodes simultaneously.

4.3 The RF Switch as a Shared Medium

One of the advantages of the MeshTest testbed over network simulators or wireless emulators is that the devices experience actual physical interference from each other. We performed a simple experiment to show how the RF matrix switch acts as a realistic shared medium when several pairs of nodes are communicating at once.

In this experiment eight 802.11g nodes were randomly arranged in a 30m square. The nodes were configured to operate in ad-hoc mode and were grouped into sender/receiver pairs. We used iperf [9] to send UDP traffic between each pair at a fixed rate. By varying the number of active pairs, we were able to observe the drop in throughput achieved by each pair as the number of sender-receiver pairs increased. The results illustrate how the 802.11 MAC protocol efficiently divides the available bandwidth, and also how the RF matrix switch behaves as a shared medium for a group of nodes placed in a small simulated area.

As a comparison we performed the same test with antennas attached to each node instead of running through the matrix switch. This setup is more analogous to that in the ORBIT testbed. The results are shown on the right in figure 8. We see similar behavior for one and two pairs of nodes, but for three and four pairs the experiment becomes unpredictable and difficult to control. Setting several nodes near each other in a lab provides a poor RF environment for testing, and we observed that nodes would begin dropping their modulation rates in response to the noisy environment. Of course RF noise can be introduced into the MeshTest testbed if if a noisy environment is desired. This comparison highlights experimental control and reproducibility we enjoy with MeshTest.

5. CONCLUSIONS AND FUTURE WORK

MeshTest is now a functional wireless testbed which we expect will be especially valuable for evaluating ad-hoc and delay-tolerant protocols. MeshTest combines the strengths of the ORBIT testbed with realistic simulated node mobility, allowing users to repeatably test mobile scenarios with multi-hop topologies. The mobility management software is independent of the type of nodes used, allowing us to experiment with any RF device that fits in the shielded enclosures.

We have shown that the testbed can realistically and consistently simulate the path loss experienced by real 802.11 nodes in a real outdoor environment. We have also shown that it reasonably and consistently approximates the attenuations in mobile experiments involving multiple nodes. The nodes can also be subjected to randomly fluctuating attenuations, simulating the effects of multipath fading.

In the future we plan to make the mobility management software faster and implement random mobility models. Also we are designing and building an expanded testbed that will accommodate up to 32 nodes.

Acknowledgment

The authors would like to thank Ian Vo for his work developing, implementing, and testing the switch/mobility management software during Summer 2007. We also thank Aaron Schulman for sharing his ORBIT scripts with us and explaining some of the strange behavior in the wireless traces.

6. REFERENCES

[1] B. A. Chambers. The grid roofnet: A rooftop ad hoc wireless network. MIT Masters Thesis, June 2002.

[2] T. Clancy and B. Walker. Meshtest: Laboratory testbed for large wireless topologies. IEEE TridentCOM 2007.

[3] P. De, A. Raniwala, S. Sharma, and T. Chiueh. MiNT: A miniaturized network testbed for mobile wireless research. IEEE INFOCOM 2005.

[4] P. De, A. Raniwala, S. Sharma, and T. Chiueh. Design considerations for a multihop wireless network testbed. *IEEE Communications Magazine*, 2005.

[5] T. Lin, S. F. Midkiff, and J. S. Park. A dynamic topology switch for the emulation of wireless mobile ad hoc networks. In *LCN '02: Proceedings of the 27th Annual IEEE Conference on Local Computer Networks*, pages 791–798, Washington, DC, USA, 2002. IEEE Computer Society.

[6] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh. Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols. IEEE WCNC 2005.

[7] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2002.

[8] A. Schulman, D. Levin, and N. Spring. On the fidelity of 802.11 packet traces. In *PAM*, 2008. To appear.

[9] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and K. Gibbs. Iperf - the tcp/udp bandwidth measurement tool. http://dast.nlanr.net/Projects/Iperf/.