

# A Testbed Emulator for Cross-Layer Studies in Mobile Ad Hoc Wireless Networks<sup>†</sup>

Jae-seung Yeom, Nawaporn Wisitpongphan, Sooksan Panichpapiboon, and Ozan K. Tonguz

Electrical and Computer Engineering  
Carnegie Mellon University  
Pittsburgh, PA 15213-3890, USA

Email: {jyeom, nawaporn, sooksan, tonguz}@ece.cmu.edu

**Abstract**—Mobile ad hoc wireless networks have been extensively studied as they make possible various interesting future applications. However, most of the existing studies focus on showing the overall impact of mobility from topology changes and signal variations, while they do not clearly show how inter-operability among protocol layers affects the network performance. To address this issue, we developed a testbed which enables us to investigate the cross-layer interaction among protocol layers at Carnegie Mellon University (CMU). In this paper, we report on the testbed as well as on our experiences and findings from it.

## I. INTRODUCTION

Mobile ad hoc wireless networks (MANET) enable many interesting future applications such as inter-vehicle networks [1], [2]. The most unique and useful feature of MANET is the dynamic configuration of networks which provides moving users with connectivity on the fly in the absence of a fixed infrastructure. Various ad hoc routing schemes are proposed to handle dynamic topology change and on-demand route discovery. However, the performance of such networks when there is mobility is not nearly as good as the performance when there is no mobility [3], [4]. Numerous proposals have been made to improve the performance of MANET and many studies have been carried out based on analysis, simulation, emulation, and experiment. While most of the existing frameworks to study MANET focus on showing the overall impact of topology changes and signal variations induced by mobility, one of the key factors affecting the performance of MANET is how well the layers in a protocol stack cooperate to handle the situation induced by mobility. Therefore, we have built a testbed which enables us to study the interaction among protocol layers.

CMU-Emulator (CMU-EMU), shown in Figure 1, is designed as a cost-efficient MANET testbed based on 802.11b for research and education purposes. In addition, it offers an easier way to repeat and manage experiments than real-world experiments do. We use a set of inexpensive off-the-shelf hardware – signal attenuators, external antennas, laptops, and so on – to miniaturize the testbed environment into the size of a table. Hence, while it can be easily reconstructed for an affordable cost, it provides meaningful data and valuable lessons. In this paper, we share our experience in building

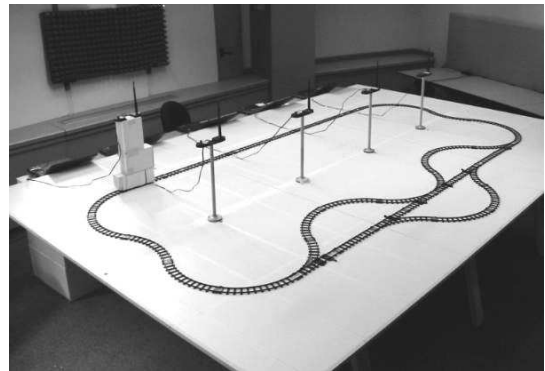


Fig. 1. The CMU-Emulator (CMU-EMU) testbed setup.

the testbed, methods of cross-layer study and experimental results. To the best of our knowledge, no experimental results with cross-layer traces from the entire protocol stack has been reported so far.

Cross-layer considerations in designing protocols for MANET have recently received increasing attention [5]–[7]. Cross-layer designing of protocols aims to enable protocols to avoid erroneous interactions and cooperate efficiently, and to achieve performance improvements consequently. Our testbed could be useful especially in such studies. It has been shown that applying independently designed cross-layer protocols together may result in unintended adverse effects [8], [9]. In addition, it is noted that cross-layer design without caution may hinder further advancement [8], [9]. To demonstrate such problems, a testbed should not be limited to show the only interaction among a partial set of protocol layers. Therefore, in our testbed, cross-layer interaction is traced throughout the entire protocol stack.

The rest of the paper is organized as follows. In Section II, we discuss previous studies in experimental and emulation approaches and also provide the background information about the carrier sense mechanism of 802.11b. In Section III, we discuss the design and implementation choices as well as the limitations of our testbed. In Section IV, we present some of the experimental results obtained on our testbed, compare the results with those from existing simulative or analytical studies, and investigate the cross-layer interaction observed from the experiments. In Section V, we conclude the paper.

<sup>†</sup>This work was supported in part by an Intel Grant to Carnegie Mellon University for establishing a Wireless Educational Laboratory.

## II. BACKGROUND

### A. Related Work

Simulation models simplify the behavior of real-world networks and sometimes fail to capture the important effects of their behavior [10]–[14]. In addition, many of the previous simulation studies mainly focus on the improvement in performance seen by using the abstract models of network behavior. Moreover, they often concentrate on the limited aspects of the network behavior. Therefore, it is rather vague as to how and when they would be helpful in real world networks and how their proposals would co-operate with other proposals. Unfortunately, this is also true with experimental studies because various factors affect results simultaneously in uncontrolled ways while making it difficult to pinpoint the exact cause of the end result [2], [15]–[17]. To obtain manageable and repeatable experimentation, various studies rely on emulation. In general, they can be categorized into four approaches: the MAC-filter approach, the signal attenuation approach, the hybrid approach, and the approach that relies on specialized hardware.

The basic idea of the MAC-filter approach is to filter out some packets between the MAC layer and the upper layer in the protocol stack to emulate the MAC layer connectivity [18]. Then, wireless nodes can be physically placed anywhere within the radio contact range regardless of the locations in experimental scenarios. Therefore, certain experiments can be easily managed and repeated. However, the physical environment seen by the physical and medium access layer is very different from that of the real-world, thereby limiting the use of cross-layer studies.

The signal attenuation approach tries to scale-down the real-world test area into a manageable area as small as a room or a table by reducing the signal power and the propagation range [19]–[22]. In [19], radio frequency (RF) signals propagate through cables, attenuators, and circulators. Therefore, it is extremely difficult to create an arbitrary topology. Moreover, the mobility of a node is emulated by variable attenuators, which are very expensive. In [20], mobility is emulated by using 1-to-4 RF multiplexers. Although the testbed provides a highly manageable and repeatable environment, the abrupt switching in the location of a node only accommodates coarse mobility, which is not adequate for many cross-layer studies.

In the hybrid approach proposed in [22], the layers above the link layer are simulated by *ns-2* and the signal strength is attenuated. This approach is useful for validating the simulation models by comparing the result of the testbed with those from pure simulations. The hybrid approach discussed in [23] replaces the physical layer in the real network stack with that of simulation, such that experiments can be easily repeated while using all the real protocol implementations above the physical layer. This approach is especially useful for designing and debugging protocols at the early stage of development. In addition, this eliminates the need for developing two versions of a protocol: one for the simulator and the other for the real networks. However, the amount of traffic load handled by the

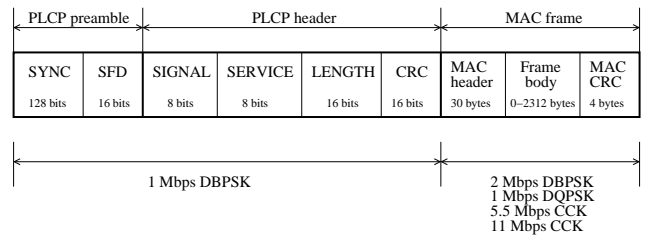


Fig. 2. The physical layer convergence protocol data unit (PPDU) format

network is limited not by the channel characteristics but by the computational capability of the physical layer simulation.

A Field-Programmable Gate Array (FPGA) component is used to emulate the effects of signal propagation in [24]. RF signals transmitted by wireless cards are mixed with the local oscillator signals and digitized such that they can be fed into an FPGA-based digital signal processing (DSP) module. A central controller manages the experiment and emulates the movement of nodes by controlling the DSP module. This approach achieves a high degree of repeatability and provides a framework for achieving a high degree of realism while overcoming the computational limit of simulation approaches.

The reprogrammable Embedded Sensor Boards (ESB) are used offering a flexible platform to study the network behavior in the real environment [25]. Although new MAC layer protocols can be reprogrammed into the control component of the board, the RF module equipped on the board is not compatible with the 802.11b standard.

### B. Carrier Sense and Signal Reception

In this section, we discuss the basics of the carrier sensing mechanism of the 802.11b protocol, which are necessary to understand our experimental results with inter-node interferences. To determine whether the channel is idle or busy, the collision avoidance strategy of the 802.11 protocol relies on the *Clear Channel Assessment* (CCA) mechanism. CCA has two sensing methods: the physical sensing and the virtual sensing (the RTS/CST protocol). In our experiments, the RTS/CST option of the device driver is turned off, and only the physical sensing is used. In the physical sensing, the Physical (PHY) layer reports a medium as ‘busy’ to the Medium Access Control (MAC) layer when the incoming Physical Layer Convergence Protocol (PLCP) preamble signal, shown in Figure 2, is detected by a receiver or when the IR variation in the band between 1 MHz and 10 MHz exceeds a certain threshold. The former sets the Carrier Sense (CS) value and the latter sets the Energy Detect (ED) value in the receiver circuit. The medium is declared ‘idle’ otherwise. When receiving a signal, a receiver detects the preamble of the PLCP protocol data unit (PPDU) of the signal and locks onto it. Then, it verifies the Cyclic Redundancy Code (CRC), the format and the length of the PLCP packet in turn. Once an error is detected, the rest of the signal is ignored. Otherwise, the MAC frame encapsulated in the PPDU is passed up to the MAC layer. Then, the MAC data CRC is checked. The successfully received MAC data is passed up to the higher protocol layer. The number of frames

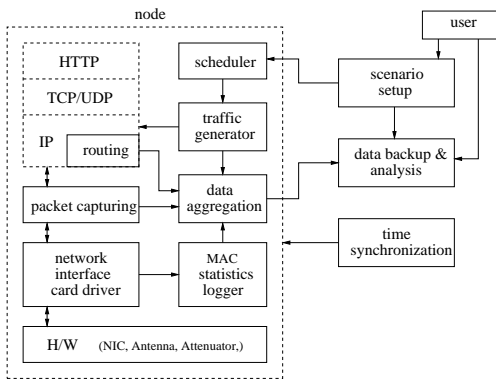


Fig. 3. System architecture

received with error is  $N_{Err} = N_{PLCP\_Fmt} + N_{PLCP\_Len} + N_{PLCP\_CRC} + N_{MAC\_CRC}$ , where  $N_{PLCP\_Fmt}$ ,  $N_{PLCP\_Len}$ ,  $N_{PLCP\_CRC}$  and  $N_{MAC\_CRC}$  are the number of frames received with the PLCP format error, the PLCP length error, the PLCP CRC error and the MAC CRC error respectively. The frame error rate at the link layer is defined as  $FER = N_{Err} / (N_{Err} + N_{OK})$ , where  $N_{OK}$  is the number of packets received without error at the MAC layer.

### III. TESTBED DESIGN AND IMPLEMENTATION

Because of the typical range of 802.11b signals (roughly 100-250 m), multi-hop wireless ad hoc network experiments would require a huge space. Clearly, it is not easy to manage experiments under such conditions. Moreover, due to the random nature of real life, it is difficult to obtain a repeatable test environment. To address the manageability and repeatability issues, we built a scaled-down testbed small enough to fit in a room. To make this possible, we use a specific set of hardware.

#### A. System Architecture and Workflow

The testbed system architecture is illustrated in Figure 3. The main functional components of the testbed are software tools to set up and run experiments and to collect and analyze data as well as hardware. The hardware help to create a physical environment required for experiments on the testbed. The basic workflow of an experiment includes time synchronization, job scheduling (traffic generation and data collection), and the preprocessing, backup, processing and analysis of data. Traffic generation and data collection tasks are scheduled to run during a test. Time synchronization is necessary to coordinate the jobs on multiple laptops, and it is done over a wired Ethernet. Traffic is generated using appropriate applications, such as `netperf`<sup>1</sup>, `ping`, etc. Then, the traffic sent and received by each node is captured and dumped into files for later analysis, using `tethereal`<sup>2</sup>. The

<sup>1</sup>A well-known network performance measuring tool.

<sup>2</sup>A packet capturing tool. Packets are captured between the device driver and the network protocol stack of operating systems and dumped into a file. Each packet in a dump file includes the headers of the link layer, the IP layer and the TCP/UDP layer. When a dump file is read, the header of each packet can be printed out on `stdout`. Then, formatted text processing tools such as `awk` can be used to extract the necessary information.

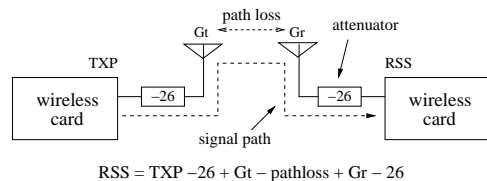


Fig. 4. Attenuators are used to reduce the signal propagation range such that multi-hop experiments can be performed in a small area

device driver of a Cisco 802.11b ethernet adapter reports various diagnostic values from the PHY/MAC layer. These values are obtained by a custom-made daemon via `/proc` interfaces. Routing tables are logged periodically – many routing daemons already produce routing table log files. We currently use the `AODV-UU 0.7.2` routing protocol implementation [17]. Test configuration parameters are also saved as well as data. The preprocessing, backup and main processing of data are usually carried out in a batch processing fashion. Many tools are developed to support this in shell scripts and `awk` scripts utilizing various pre-existing software tools. The script tools are developed on the `Linux` (kernel 2.4.20) system. The TCP New Reno is the TCP implementation used in the testbed.

#### B. Hardware Components

Hardware components are chosen to obtain the three desired properties of testbed: manageability, repeatability, and cost efficiency. The first property is obtained by reducing the testbed to a manageable size and by automating experiment procedures. The second property is obtained by avoiding or minimizing random effects in signal propagation, such as multi-path effects. The use of major hardware components is described in the following.

The signal strength is reduced by using attenuators to fit the testbed in a small room. By default, 26 dB attenuators (20 dB + 6 dB) are attached to each laptop between the wireless adapter and the external antenna, such that the total 52 dB attenuation is applied to a signal path for any transmitter and receiver pair, as shown in Figure 4. The wireless adapter card model used, Cisco aironet LMC352, does not have a built-in internal antenna but only has connection ports for two external antennas. Although many wireless adapter models with built-in antennas also provide connection ports for external antennas, we chose to use the one with no internal antenna because the built-in antennas are not completely bypassed and emanate leakage. In addition, this adapter model has another useful feature, the multiple choices of the transmit power: 1 mW, 5 mW, 20 mW, 30 mW, 50 mW and 100 mW. Figure 5 shows that the communication range in the testbed is reduced to around 70 cm by the default setting: the 5 mW transmit power, the 11 Mbps data rate, the 52 dB attenuation and the channel 8. The testbed size is approximately  $375 \times 250 \text{ cm}^2$ , and can hold up to 4-hop communication routes using 5 laptops with the default setting. The caveat in choosing the amount of attenuation is the leakage from the card, although it is weaker than that from those with built-in antennas. The strength of leakage at the auxiliary antenna port of a card is measured around 0.9%

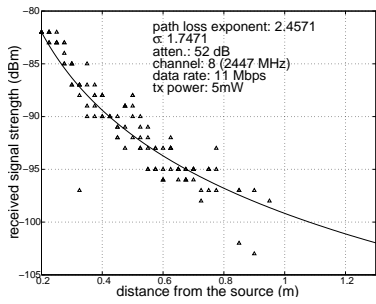


Fig. 5. Signal propagation in the testbed.

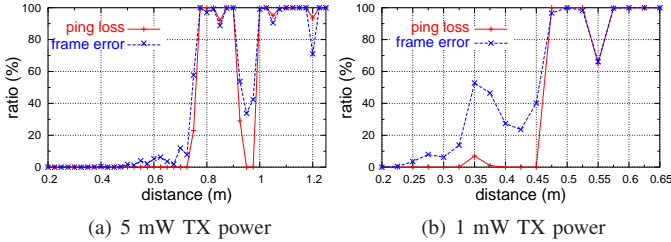


Fig. 6. Frame error rates and ping loss rates are measured over different distances between a source and a destination configured with the 11 Mbps data rate, the 52 dB attenuation, and the 0.2 sec of ping interval

of the total transmit power, which is lower than the power of legitimate transmit signal by 20 dB. The leakage is also subject to path loss but lacks the advantage of 4.28 dBi antenna gain. However, if the attenuation on the legitimate signal path is too high, leakage dominates the radio propagation. The maximum amount of attenuation applicable needs to be found considering the distance between external antennas and laptops and the distance between external antennas. All the attenuation could have been applied only to the receiver side. However, in such a case, multi-path effects in the indoor environment become highly significant.

Without the assumption of isotropic radiation pattern of antennas in a horizontal plane, it would not have been feasible to set up a test due to the complexity of propagation patterns. Therefore, we use external antennas, Cisco Air-Ant3351, with omni-directional radiation patterns, which have a 2.14 dBi gain. Although the signal range is shrunken to miniaturize the testbed, other physical properties remain the same. For example multi-path fading effects are still related to the order of the wavelength of signal, which has not been shrunken. Therefore, to mitigate offering the misleading environment, we try minimizing such effects. Moreover, minimizing uncontrollable effects helps us to see the causal relations in cross-layer interaction better. On these accounts, the near field effect is avoided by placing antennas sufficiently apart from each other, i.e.,  $20\text{ cm} \geq \frac{2L^2}{\lambda}$ , where  $\lambda$  is the wavelength of the signal and  $L$  is the length of the antennas. The multi-path effects from the ground is avoided by adjusting the height of antennas,  $h$ , to 48 cm, where  $\frac{4h^2}{\lambda} \geq \text{the testbed size}$ .

### C. Reduced Signal Range

As shown in Figure 2, the modulation format of the PLCP preamble and payloads may be different, resulting in a dis-

TABLE I  
RANGE SUMMARY

TX Power (mW)	Data Rate (Mbps)	Attenuation (dB)	Effective Communication Range (cm)	Maximum Communication Range (cm)
5	11	50	75	175
5	11	52	70	122
5	11	56	40	70
1	11	50	37	72
1	11	52	30	57
1	1	52	77	160

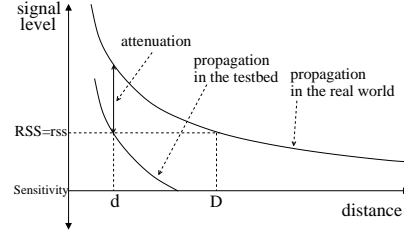


Fig. 7. Mapping a distance in the testbed to the counterpart in the real world by identifying the distance where the received signal strengths are equal.

crepancy between the carrier sensing range and the data communication range. According to the card manufacturer's specification, the receiver sensitivities to the modulation of PLCP preamble and that of payloads may differ by 9 dB. Figure 6 shows the ping packet loss rates and frame error rates which are measured over various distances between a source and a destination. Unlike the case with a simulated environment, a communication range does not have a clear-cut boundary in the real world. Therefore, it is difficult to describe it by a single value. Table I summarizes the effective communication ranges and the maximum communication ranges observed in the testbed. There is a dramatic increase in the frame error rate around the boundary of the effective communication area. The maximum communication range refers to the distance beyond which no ping packet is received. An interference range is much larger than an effective communication range. With the 5 mW transmit power, the 11 Mbps data rate, and the 52 dB attenuation, interference occurred even at the distance of 3.4 times of the effective communication range.

### D. Limitations

The given environment of the testbed is not ideal, since the campus network infrastructure uses the channels 1, 6, and 10. We chose to use the channel 8, because the cells of the channels 6 and 10 are farther than that of the channel 1 from the testbed. The testbed is placed in a half-underground room, where the external interference is weaker than that above the ground. Fortunately, the strength of the residual interference is diminished by the attenuators at receivers and the walls around the room. Consequently, we find that the given environment is sufficient for our purpose.

To mimic mobility, we use toy trains, which cost only \$10 per set, by placing an antenna on top of each train. Since the antenna is connected to a laptop, the movement of a train requires a human to carry the laptop around the testbed.

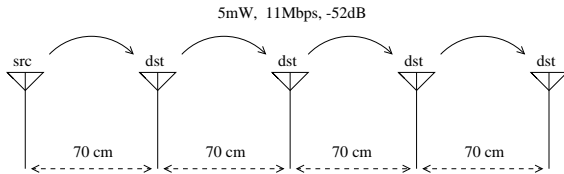


Fig. 8. The experimental setup for the Multi-hop TCP throughput study. Five nodes are aligned in a line and separated as far as possible while remaining in the data communication ranges of immediate neighbors only.

However, the movement of a train is far more steady than that of a human. Therefore, this leaves out the problem of human choreography [17]. Although the track configurations limit the mobility models in experiments, they still support many useful scenarios where we gain meaningful insight into the impact of mobility from the realistic environment. The movement of a train is interpreted by projecting the distances on the testbed to those in the real world, as described in Figure 7. However, such a distance mapping method requires that the propagation model of the testbed and that of the real world have an identical path loss exponent. Otherwise, when a pair of two different distances in the testbed are mapped into a pair in the real world, their relative differences are inconsistent [21]. Therefore, the only environment that the testbed emulates is the one that can be described by the propagation model of the identical path loss exponent. In addition, the doppler effect at the speed of a train differs from that at the projected speed.

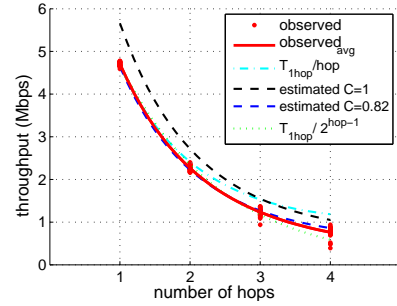
#### IV. EXPERIMENT

##### A. Multi-hop TCP Throughput Performance

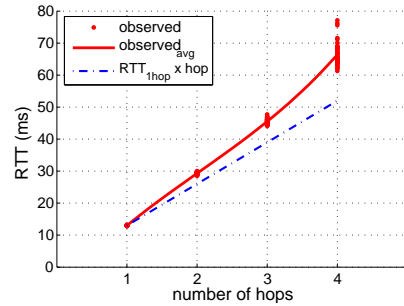
The goal of this experiment is to measure the reference multi-hop TCP throughput performance in a simple chain topology without mobility, and to study how the MAC protocol behavior is related to the performance.

1) *Experimental Setup*: Five nodes are aligned and separated as far as possible while remaining in good communication ranges such that only neighbors can hear each other, as shown in Figure 8. This separation is likely to minimize inter-node interference and provide the maximum capacity available. TCP throughput is measured over different numbers of hops for 1 minute each. Since TCP is a reliable transport protocol, *throughput* is measured at the source by counting only the number of acknowledged segments among all the segments transmitted. The *round-trip-time* (RTT) is the time between the transmission of a segment and the reception of the acknowledgement of the transmission. Retransmitted segments or cumulatively acknowledged segments are ignored. Traffic is generated by `netperf`, such that a traffic source always has data to send and tries to transmit data as quickly as possible. We use `tcptrace` to measure the average RTT, and `tethereal` and `awk` to count the number of lost packets and calculate the TCP throughput from packet dump files.

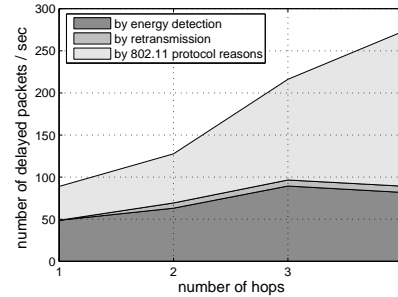
2) *Experimental Results and Observations*: There are many factors affecting the TCP throughput performance in ad hoc wireless networks. Capacity is one of the factors, and how it limits the throughput is discussed in [26]. When there



(a) The TCP throughput vs. the number of hops



(b) The round-trip-time vs. the number of hops



(c) The number of packets deferred or retransmitted at the MAC layer

Fig. 9. Inter-node interference causes an increase in delay mostly via collision avoidance and retransmission at the MAC layer. The increased delay decreases the aggregate throughput.

are only two nodes, transmitter and receiver, in an ad hoc wireless network, the contention is minimum and the capacity is highest. In such a case, the throughput is bounded by  $\frac{1460}{1500} \times 6.06 \approx 5.90$  Mbps, where the IP packet size is 1500 bytes with the 40 byte TCP/IP header, and the theoretical maximum throughput of 802.11b MAC layer is 6.06 Mbps when the 11 Mbps data rate is used and the RTS/CTS is not used [27]. If the radios of non-neighbor nodes do not interfere with each other, channel utilization would be  $\frac{1}{3}$  of the maximum because three consecutive nodes in the chain forms a group where only one of them can transmit at a time; for example, when the first node transmits to the second node, the third node cannot transmit without interfering with the second node. As the hop length of connection increases in a chain topology, the throughput is bounded by the utilization calculated in this way [26]. They also discovered that the throughput is often far less than the expected bound due to

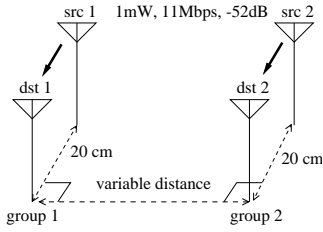


Fig. 10. The experimental setup for the impact of inter-node interference study. Two transmitter/receiver pairs contend for medium when they are close.

the inefficiency of MAC layer in the simulation. Although they suggest various explanations for the causes related to the MAC layer behavior, they present no detailed evidence seen from the MAC layer as in Figure 9(c). In our case, as discussed in Section III-C, radios interfere far beyond the effective communication range. Even the fifth node may not transmit without interfering with the second node when the first node communicates with the second node. It is observed that the interference in this experimental setting covers the area beyond three times of the effective communication range in which case the utilization is bounded by  $\frac{1}{5}$  as the chain gets longer. In our case, it already reaches below  $\frac{1}{5}$  for the chain of five nodes, as shown in Figure 9(a). We plan to investigate this further with more nodes to study the details of the MAC layer impacts on the utilization bound. In the mean time, we conduct another experiment to study spatial reuse with a different setting, as discussed in Section IV-B.

The relationship between RTT and TCP throughput is described in much of the literature. For example, according to the study in [28], TCP throughput is bounded by

$$BW = \frac{C \cdot MSS}{RTT \sqrt{p}}, \quad (1)$$

where  $MSS$  is the message segment size,  $p$  is the packet loss rate, and  $C$  is a constant dependent on the TCP implementation used, the ACK strategy and the loss mechanism.  $C$  is normally less than 1. The RTT values are measured over different hops, as shown in Figure 9(b), and the resultant TCP throughput is shown in Figure 9(a). Figure 9(a) shows that the measured throughput is bounded by  $BW$  with  $C = 1$  and close to  $BW$  with  $C = 0.82$  as calculated by (1). It is also shown that the throughput decreases exponentially rather than proportionally. The  $p$  values observed in our experiment are 0.024, 0.023, 0.025, and 0.028 for one-hop, two-hop, three-hop and four-hop routes respectively. Figure 9(b) shows that the gap between the measured RTT values and the ideally expected values – the RTT over 1-hop routes multiplied by the number of hops – increases as the hop length increases. Figure 9(c) shows what contributes to the RTT increase at the MAC layer. It is shown that the collision avoidance is a more dominant contributor to RTT than collision is. Therefore, designing an efficient carrier sensing method and a collision avoidance mechanism can be the key to performance improvement.

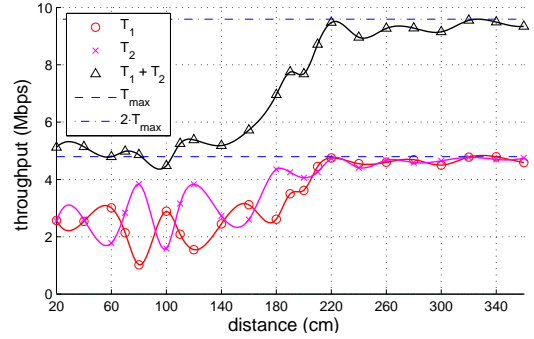


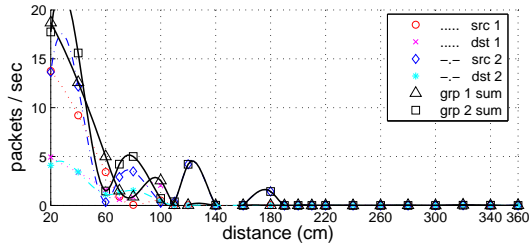
Fig. 11. The impact on the TCP throughput by inter-node interference. Throughputs of each TX/RX pair ( $T_1$  and  $T_2$ ) and aggregate throughput ( $T_1 + T_2$ ) are shown.

### B. Collision Avoidance and Medium Utilization

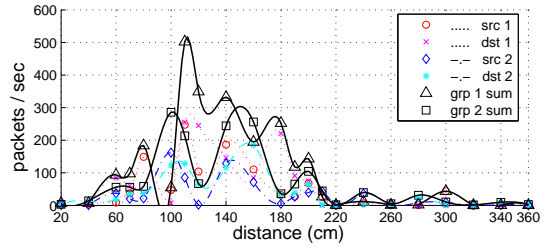
The goal of this experiment is to demonstrate the capability of the testbed to show how inter-node interference is spatially distributed and how the medium is spatially reused.

1) *Experimental Setup*: Four nodes are positioned as shown in Figure 10. There are two communication groups in each of which a source node transmits TCP data stream to its destination node in the group. These two groups try to communicate simultaneously, possibly contending to acquire the medium. Then, we study how they interfere with each other over various distances. A source and a destination are separated by the fixed distance of 20 cm, while the distance between the two groups varies. The TCP throughput of each group is measured at every 20 cm to see how interference is spatially distributed and how the medium is spatially reused. The transmission power used in this experiment is 1 mW, and the effective communication range is around 25-30 cm. The other parameters are the same as those described in Section IV-A.

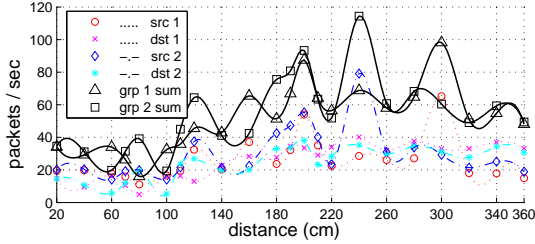
2) *Experimental Results and Observations*: The TCP throughput result is presented in Figure 11. From this, we can roughly estimate that the spatial reuse factor is between  $\frac{1}{4} = \frac{25}{100}$  and  $\frac{1}{9} = \frac{25}{225}$ . In the simulation study discussed in [26], the throughput is bounded by the reuse factor of  $\frac{1}{7}$  in a chain topology. As discussed in Section IV-A, RTT is an important factor on TCP throughput. RTT can be increased either by the increased hop-length of routes, the collision avoidance at the MAC layer or the retransmission due to collision. In Section IV-A, we observed that the collision avoidance is the dominant source of RTT with the size of network in which most of the nodes are within the carrier sense range of each other. When the two communication groups are close, collision is avoided by PLCP preamble detection, as observed in Figure 12(a). Then, as the distance between them increases, it is avoided by RF energy detection, as observed in Figure 12(b). During the transition between the avoidance schemes, there is a peak of retransmissions at the MAC layer, as observed in Figure 12(c). However, the number of packet drops perceived by the upper layer of the MAC layer is small and evenly distributed over the distances, as observed in Figure 12(d); a packet drop is reported to the above layer when the MAC



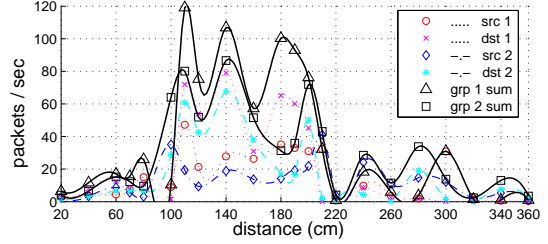
(a) The number of packets deferred due to 802.11 protocol reasons



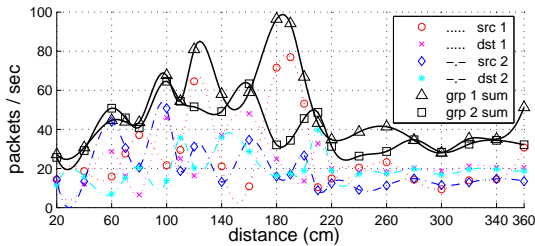
(a) The number of PLCP format errors



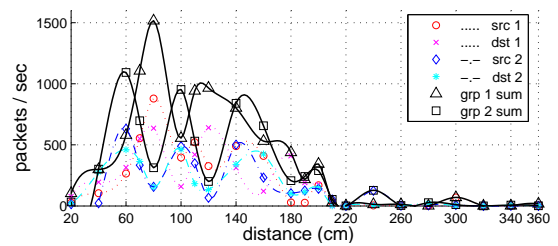
(b) The number of packets deferred due to RF energy detection



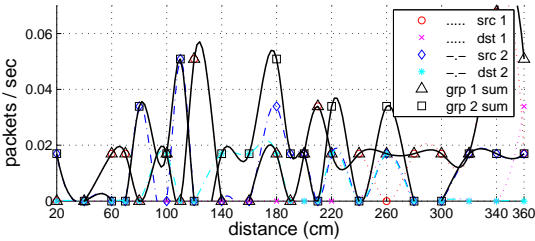
(b) The number of PLCP CRC errors



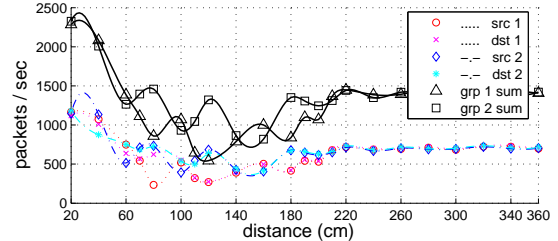
(c) The number of packets transmitted at the MAC layer with retrying



(c) The number of MAC CRC errors



(d) The number of packets retried up to the maximum retransmission limit but failed to transmit



(d) The number of MAC frames received successfully

Fig. 12. The impact of inter-node interference vs. the distance between two communication groups. Delay is increased by collision avoidance and retransmission due to collision at the MAC layer, which affect spatial medium utilization and round-trip-time at the transport layer

layer retransmission fails after a certain number of retries. Figure 13 shows why the packets had to be retransmitted at the MAC layer. When the groups are close, each of them overhear the communication of the other, as observed in Figure 13(d). As the distance between the groups increases beyond the effective communication range, 30 cm, the receivers start receiving corrupted packets, as observed in Figure 13(c). Note that the peak in Figure 13(c) comes before the peak in Figure 12(c). It is because signal is sometimes not received by the PHY layer as the distance increases, as observed in Figure 13(a) and 13(b). Thus, no packet is handed over to the MAC layer. Beyond 220 cm, interference seems to be insignificant except that the number of deferred packets due to energy

Fig. 13. Collision is caused by inter-node interference. The frame errors show the collision statistics.

detection peaks around it. The alternating pattern observed in Figure 11 requires further study especially regarding the fairness issue in the MAC layer [29].

### C. Mobility Experiment and Cross-Layer Study

In this experiment, we demonstrate the capability of our testbed to trace the cross-operations in an entire protocol stack in studying the impact of mobility on the TCP throughput performance. For this purpose, we compare the TCP throughput measured with different movement patterns, and discuss how the cross-operations between protocol layers made the difference in the TCP throughput performance.

1) *Experimental Setup*: Four of the total five nodes are placed as in the experiment discussed in Section IV-A with a slight adjustment in the distance between them, as shown

in Figure 14. The rest of the parameters are configured as the same as described in Section IV-A, except that node one is now moving on the track around the other four stationary nodes. The track is designed in such a way that any position on the track is covered by the radio of at least one of the stationary nodes. While running on the track, node one receives traffic from node two which may have been relayed via other stationary nodes. The throughput of the TCP connection is measured with three different mobility patterns. In pattern one, node one moves on the straight path (the outer path) shown in Figure 14 at the average speed of  $13 \text{ cm/sec}$ , which corresponds to the real world speed of  $62 \text{ km/hrs}$  ( $\approx 38 \text{ miles/hrs}$ ) according to the mapping suggested in Section III-D. The inner curved-path shown in Figure 14 is used for patterns two and three. While the speed used in pattern two is the same as that of pattern one, the average speed used in pattern three is  $6.3 \text{ cm/sec}$ , which corresponds to approximately  $30 \text{ km/hrs}$  ( $\approx 18.6 \text{ mile/hrs}$ ). In pattern two, node one passes the area in which signals are heard from multiple nodes more often than it does in pattern one. In pattern three, node one moves on the same path as it does in pattern two but at a slower speed. Therefore, comparing the performances from three different mobility patterns would give us a rough idea of the impact of the speed, interference and handoff.

2) *Experimental Results and Observations*: To quantify the impact of mobility on the TCP throughput, we use two metrics. One is the *expected throughput*, which is proposed in [3] as follows:

$$Throughput_e = \frac{\sum_{h=1}^{\infty} (Throughput(h) \cdot T(h))}{\sum_{h=1}^{\infty} T(h)}, \quad (2)$$

where  $h$  is the number of hops,  $T(h)$  is the total duration of the existence of  $h$ -hop routes, and  $Throughput(h)$  is the throughput measured over  $h$ -hop routes without mobility involved. Then the impact of mobility is quantified as the difference between the expected throughput and the measured throughput with mobility. To understand the detail of the impact, we propose another metric which takes connectivity into consideration as follows:

$$Throughput_{e-conn} = \frac{\sum_{h=1}^{\infty} (Throughput(h) \cdot connectivity(h) \cdot T(h))}{\sum_{h=1}^{\infty} T(h)}, \quad (3)$$

where  $connectivity(h)$  is the ratio of the connected period over the total time that a connection takes on  $h$ -hop routes, which is the sum of the connected period, half of the disconnected period during the reroutes to other hop-length paths, and the disconnected period during the reroutes to the same hop-length paths. Using this metric, we can concentrate on the impact of mobility excluding that from the connectivity performance of routing protocols. However,  $connectivity(h)$  is currently measured only at the source. In other words, the current calculation does not consider unidirectional links or stale routing table entries. Therefore, a tighter bound exists.

The mobility pattern two results in a lower throughput than pattern one and three, encountering multiple-coverage areas more often and more quickly respectively, as shown in Figure

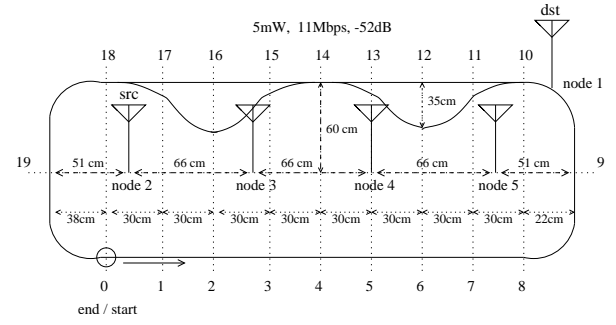


Fig. 14. The experimental setup for the study of mobility impact. One node is moving around the other four nodes while communicating with one of them.

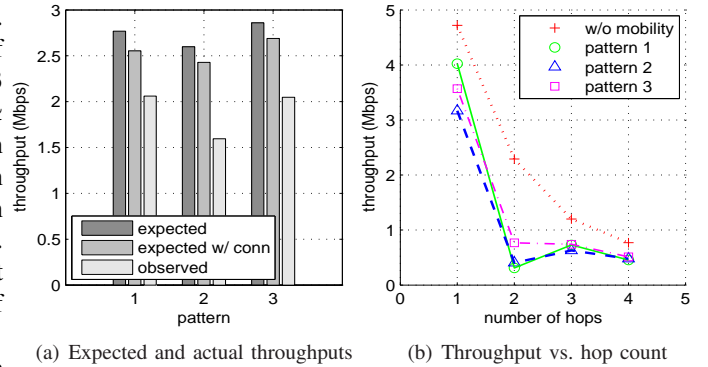


Fig. 15. The expected throughput and the measured throughput for each mobility pattern.

15. In such areas, signals are received from multiple stations, links become unstable due to inter-node interferences, capture effects, frequent route changes, and TCP congestion control back-offs. [3], [18], [29]–[31]. We observed that rerouting occurs 7, 10 and 19 times in mobility patterns one, two and three respectively. Therefore, the throughput with pattern two could have been affected more significantly by the rerouting overhead than that with pattern one, while the throughput with pattern three could have been such than that with pattern two. However, Figure 15(a) shows that the actual throughput with pattern three is almost the same as that with pattern one, and the gaps between the expected throughput with and without connectivity consideration are relatively even and small with the three patterns. In addition, Figure 15(a) also shows that the gap between the expected throughput with connectivity consideration and the actual throughput is the largest with pattern two, which suggests that the throughput degradation due to the other causes than the rerouting overhead is the largest with pattern two. Therefore, the rerouting overhead, which affects the connectivity, is not likely to be the major reason of the performance degradation in this experiment. However, it does not necessarily imply that the connectivity performance is sufficient because it is possible that the significant damage is already done at the other layers and little is left to be affected by connectivity. Thus, improvement in other factors may reveal the problems of connectivity. Figure 15(b) shows that the throughput is degraded especially over the two-hop routes. Therefore, we further examine the cross-

layer interactions while paying attention to what happens over the two-hop routes.

Figure 15(b) also shows that when the route is comprised of more than two hops, the difference between the throughput with and without mobility is only marginal. This does not imply that the impact of mobility is insignificant as the hop length of a route increases. Instead, the impact of mobility is obscured by other adverse effects and resurfaces when those are reduced. We believe that the performance degradation is mainly due to the misbehavior of TCP in the presence of inter-node interference.

To explain the reason behind our hypothesis, we display one of the data set obtained from a test using mobility pattern one in Figure 16. When node one is crossing the effective communication boundary of node two and node three, it experiences inter-node interferences from the both nodes, which explains the fact that the number of retransmitted packets per second at the MAC layer by node two and the number of packets dropped due to MAC CRC error at node three sharply increase at around 3 seconds, as shown in Figure 16(e) and 16(f). This is the moment when the route should change from the one-hop route to the two-hop route. However, Figure 16(d) shows that the link between node three and node one, which would have been the second link of the two-hop route, is broken. Moreover, since node one can hear both node two and node three, the shortest route, which has the direct connection to node two, is chosen regardless of the link quality. Although, node one is directly connected to node two even at around 6 seconds, the quality of the link is very poor, resulting in a large number of retransmissions at the MAC layer, as shown in Figure 16(e). Finally, the TCP starts the binary back-off at around 6 seconds, as shown in Figure 16(b). Then, the throughput drops to zero although it is shown as non-zero in Figure 16(a). The reason is that the throughput is averaged over the period between the transmission of a segment and either the last reception of the acknowledgement after the transmission and before the end of the sampling granularity period or the first reception of the acknowledgement after the sampling period. Since packets are lost around 6 seconds, and packets start getting through and the throughput peaks instantly at around 14 second, the averaged throughput between these events is non-zero. This is the moment when node one is passing by position three in Figure 14, as shown in Figure 16(g). Finally, at around 9 seconds (at position four), the route from node two to node one is updated to a two-hop route, as shown in Figure 16(c). However, at this point, node one is already crossing the effective communication boundary of node three and node four. Therefore, not only the link between node one and node two is invalid then, but also the two-hop route is almost outdated. They both find the three-hop route and the TCP restarts transmitting at around 14 seconds.

The summary of observations is as follows. Even with the ideal link conditions, the fast node movement can outdated the routing table fairly quickly. When the speed is even faster, a newly discovered route becomes invalid even before being used. In general, the faster the speed, the more time

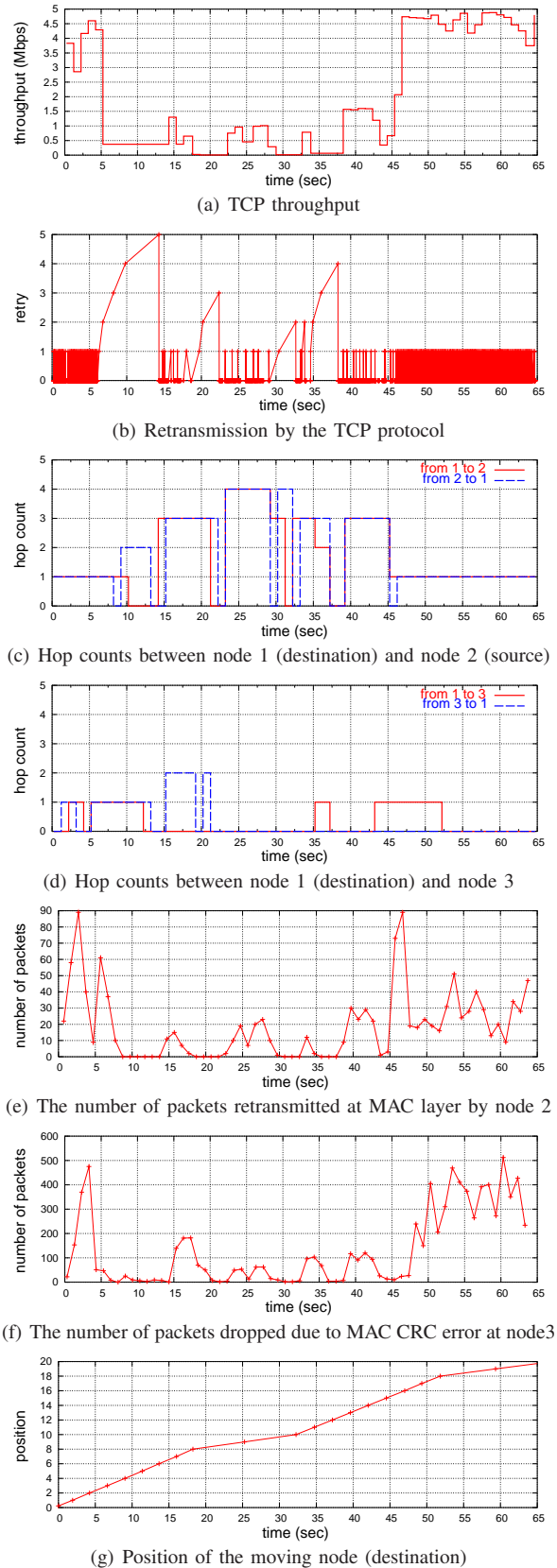


Fig. 16. Cross-layer viewpoint of the network. This sample data is measured with mobility pattern 1.

spent on route discoveries, the longer the overall disconnected period. When there is disconnection or route change, the TCP congestion control initiates binary exponential back-off, increasing the idle period. In addition, when a destination node is passing through the area where it can hear from multiple nodes, the routing protocol becomes unstable. This is due to the fact that routing tables get updated too frequently due to the unstable or asymmetric radio propagation. Then, packets get dropped very easily, causing the TCP, which is unaware of what is happening in the other layers, to timeout and back off. Another observation is that TCP packets are often much longer than routing control packets. If such packets are sent over a multiple signal coverage area, they are more vulnerable to bit errors than the small routing control packets. Therefore, even when a route is stable, TCP may experience significant performance degradation.

In conclusion, routing protocols must be designed to cope with unstable links, which stems from the inter-node interference problem and mobility. TCP congestion control needs to be aware of the node mobility as well as the error-prone wireless link. New PHY/MAC protocol design may be required to reduce the inter-node interference in MANET. Improving the performance of MANET cannot easily be achieved by improving a single layer in the protocol stack. Rather, it requires improving the inter-operability between different layers. Therefore, it is important for a testbed to demonstrate cross-layer interaction.

## V. CONCLUSIONS

We have built an ad hoc wireless emulation testbed to study the cross-layer interactions in the protocol stack of a MANET. We have demonstrated the capability of our testbed to trace inter-operability between different layers via a number of experiments with simple MANET scenarios. We have presented the results from our experiments and compared them with some simulation results reported in the literature. Furthermore, we have presented the trace of network behavior from the cross-layer perspective over space and time using simple scenarios. We have analyzed the cross-layer impact of mobility on the throughput performance over a multi-hop communication route. We believe that the experimental methodology and results presented in this paper may help many other studies, such as cross-layer protocol design, performance evaluation, network planning, and so on.

## ACKNOWLEDGMENT

The authors would like to thank Junggil Choi, Evsen Yanmaz, and Sun-Chien (Cecil) Wei for their help and advice.

## REFERENCES

- [1] O. K. Tonguz and G. Ferrari, *Ad Hoc Wireless Networks: A Communication-Theoretic Perspective*, John Wiley and Sons, 2006.
- [2] J. P. Singh et al., "Cross-layer Multi-hop Wireless Routing for Inter-Vehicle Communication," *IEEE Tridentcom*, pp. 92-101, Mar. 2006.
- [3] G. Holland and N. Vaidya, "Analysis of TCP Performance over Mobile Ad Hoc Networks," *ACM MobiCom*, pp. 219-230, Aug. 1999.
- [4] J. P. Singh et al., "Wireless LAN Performance Under Varied Stress Conditions in Vehicular Traffic Scenarios," in *IEEE VTC*, pp. 743-747, Sept. 2002.
- [5] S. Shakkottai et al., "Cross-Layer Design for Wireless Networks," *IEEE Communications Magazine*, vol. 41, no. 10, pp. 74-80, Oct. 2003.
- [6] E. Setton et al., "Cross-layer Design of Ad Hoc Networks for Real-Time Video Streaming," *IEEE Wireless Communications*, vol. 12, no. 4, pp. 59-65, Aug. 2005.
- [7] X. Yu, "Improving TCP Performance over Mobile Ad Hoc Networks by Exploiting Cross-Layer Information Awareness," *ACM MobiCom*, pp. 231-244, Sept. 2004.
- [8] V. Kawadia and P. R. Kumar, "A Cautionary Perspective on Cross-Layer Design," *IEEE Wireless Communications*, vol. 12, no. 1, pp. 3-11, Feb. 2005.
- [9] V. Srivastava and M. Motani, "Cross-Layer Design: A Survey and the Road Ahead," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 112-119, Dec. 2005.
- [10] J. Heidemann et al., "Effects of Detail in Wireless Network Simulation," *SCS Multiconference on Distributed Simulation*, pp. 3-11, Sept. 2000.
- [11] S. Desilva and S. R. Das, "Experimental Evaluation of a Wireless Ad Hoc Network," *IEEE ICCCN*, pp. 528-534, Oct. 2000.
- [12] D. Cavin et al., "On the Accuracy of MANET Simulators," *ACM POMC*, pp. 38-43, Oct. 2002.
- [13] J. Liu et al., "Simulation Validation Using Direct Execution of Wireless Ad-Hoc Routing Protocols," *ACM/IEEE PADS*, pp. 7-16, May 2004.
- [14] D. Kotz et al., "Experimental Evaluation of Wireless Simulation Assumptions," *ACM MSWiM*, Oct. 2004.
- [15] V. Kawadia and P. R. Kumar, "Experimental Investigations into TCP Performance over Wireless Multihop Networks," *ACM SIGCOMM workshop*, pp. 29-34, Aug. 2005.
- [16] D. Johnson et al., "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks," *Ad Hoc Networking*, Chapter 5, pp. 139-172, Addison-Wesley, 2001.
- [17] E. Nordstrom et al., "A Testbed and Methodology for Experimental Evaluation of Wireless Mobile Ad Hoc Networks," *IEEE Tridentcom*, pp. 100-109, Feb. 2005.
- [18] Y. Lee et al., "Experimental Evaluation of LANMAR, a Scalable Ad-Hoc Routing Protocol," *IEEE WCNC*, vol. 4, pp. 2032-2037, Mar. 2005.
- [19] J. T. Kaba and D. R. Raichle, "Testbed on a desktop: strategies and techniques to support multi-hop MANET routing protocol development," *ACM MobiHoc*, pp. 164 - 172, 2001.
- [20] S. Sanghani et al., "EWANT: Emulated wireless ad hoc network testbed," *IEEE WCNC*, pp. 1884-1849, Mar. 2003.
- [21] J. Yeom and J. Choi, "A multi-hop mobile ad hoc network emulator," Master's thesis, Carnegie Mellon University, Dec. 2003.
- [22] P. De et al., "MiNT: A Miniaturized Network Testbed for Mobile Wireless Research," *IEEE Infocom*, vol. 4, pp. 2731-2742, Mar. 2005.
- [23] J. Flynn et al., "JEmu: A Real Time Emulation System for Mobile Ad Hoc Networks," in *Proc. of the 1st Joint IEI/IEEE Symposium on Telecommunications Systems Research*, Nov. 2001.
- [24] G. Judd and P. Steenkiste, "Repeatable and Realistic Wireless Experimentation through Physical Emulation," *ACM HotNets-II*, Nov. 2003.
- [25] Y. I. Jerschow et al., "A Real-World Framework to Evaluate Cross-Layer Protocols for Wireless Multihop Networks," *ACM REALMAN*, pp. 1-6, May 2006.
- [26] J. Li et al., "Capacity of Ad Hoc Wireless Networks," *ACM MobiCom*, pp. 61-69, July 2001.
- [27] J. Jun, P. Peddabachagari and M. Sichitiu, "Theoretical Maximum Throughput of IEEE 802.11 and its Applications," in *Proc. of IEEE Network Computing and Applications*, pp. 249-256, April 2003.
- [28] M. Mathis et al., "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm," *SIGCOMM, Computer Communications Review*, vol. 27, no. 3, pp. 67-82, July 1997.
- [29] S. Ganu et al., "Methods for Restoring MAC Layer Fairness in IEEE 802.11 Networks with Physical Layer Capture," *ACM REALMAN 06*, pp. 7-14, May 2006.
- [30] M. K. Marina and S. R. Das, "Routing Performance in the Presence of Unidirectional Links in Multihop Wireless Networks," *ACM MobiHoc*, pp. 12-23, 2002.
- [31] K. Chin et al., "Implementation Experience with MANET Routing Protocols," *ACM SIGCOMM Computer Communications Review*, vol. 32, no. 5, pp. 49-59, Nov. 2002.