Lessons Learned Constructing a Wireless Ad Hoc Network Test Bed

Sushant Jadhav, Timothy X Brown, Sheetalkumar Doshi, Daniel Henkel, Roshan George Thekkekunnel, *University of Colorado, Boulder, CO 80303*

***Abstract*—This paper describes lessons learned constructing a wireless mobile ad hoc network test bed at the University of Colorado at Boulder. Research work in the area of ad hoc wireless networking has mainly been conducted using simulation software with little substantial work using actual test beds. The test bed described in this paper is designed to facilitate experiments based on wireless ad hoc networks including radios mounted at fixed sites, on ground vehicles, and in small Unmanned Aerial Vehicles (UAVs). Laptop computers, PDAs, and special-purpose ad hoc radios running the Dynamic Source Routing (DSR) protocol over IEEE 802.11b serve as nodes in this test bed.**

The paper discusses various components of the test bed and also makes a note of issues faced during experimentation and lessons learned in this process.

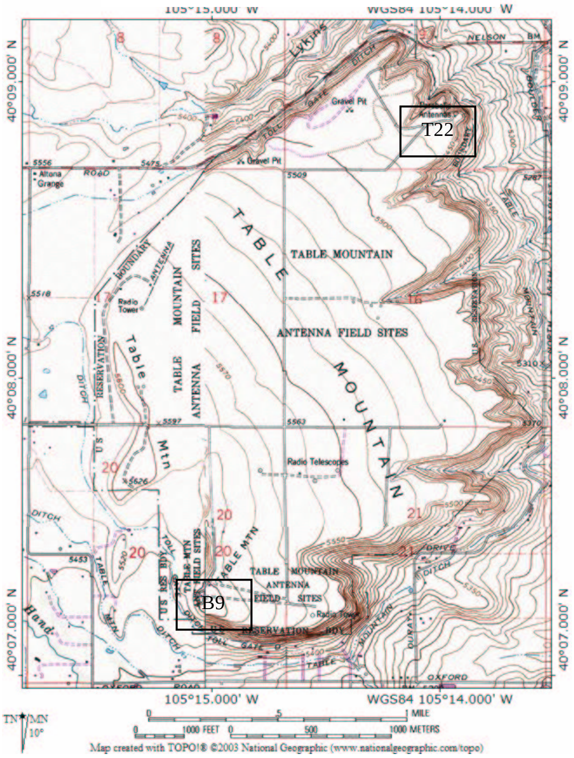
Index Terms—**Dynamic Source Routing Protocol, Unmanned Aerial Vehicle, Mesh Network Radio.**

I. INTRODUCTION

WIRELESS mobile ad hoc networks have been researched extensively in the past using simulation tools primarily to evaluate various ad hoc routing protocols [6][23]. Real test bed implementations have been limited, and the efforts so far can broadly be categorized as bench top, indoor, fixed outdoor, and mobile outdoor setups.

Bench top test beds enable significant experimentation within the confines of a room with the aid of MAC filtering, RF attenuators, or other emulation techniques to shrink the wireless range [8][10][11][20][24]. These test beds are ideally suited for protocol development and testing but are limited in capturing all of the significant behaviors.

Indoor test beds within a building provide a realistic environment for experimentation [3][13][18]. Indoor applications can be tested effectively in this environment but mobility and propagation of the outdoor environment cannot be replicated here. Full scale outdoor environments introduce realistic hardware, propagation, and interference issues not captured by simulated networks. Most of these test beds though are restricted to fixed sites [1][13][21]. Outdoor mobile efforts include test beds to study mobility of nodes in wireless networks [14][15][16][17] and an effort to evaluate the deployment of sensor networks in UAVs [2]. The testbed described in this paper differs from earlier efforts as it is not limited to a single ad hoc routing protocol and is designed specifically to collect performance statistics using benchmark tests. The results are reproducible and the performance statistics collected can be analyzed at a minute level.



Map created with TOPO!® ©2003 National Geographic (www.nationalgeographic.com/topo)

Fig. 1. Table Mountain Test Site

The test bed is designed to accommodate communication among arbitrary combinations of fixed, ground vehicle, and UAV nodes. It consists of ad hoc radio nodes placed at various positions on a test site. This site is located on the

---

Manuscript received February 18, 2005.

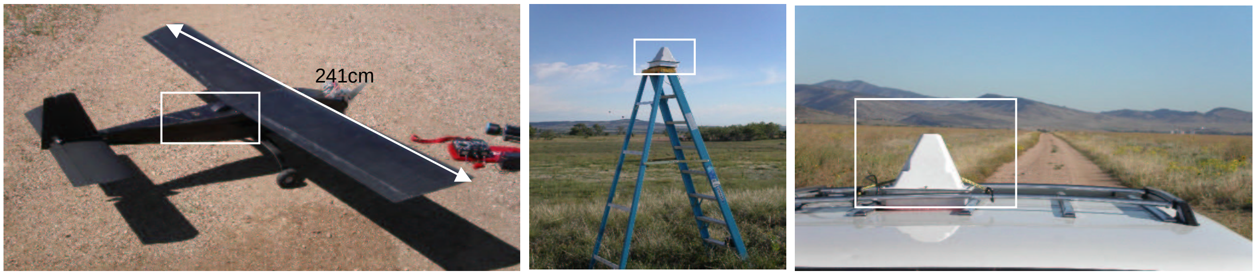
S. S. Jadhav is a student majoring in Interdisciplinary Telecommunications (phone: 201-600-0198; e-mail: Sushant.Jadhav@colorado.edu).

T. X. Brown is a professor for the Electrical and Computer Engineering and Interdisciplinary Telecommunications departments (phone: 303-492-1630; e-mail: Timxb@colorado.edu).

S. Doshi is a doctoral student majoring in Electrical and Computer Engineering (e-mail: Sheetalkumar.Doshi@colorado.edu).

D. Henkel is a researcher with the Interdisciplinary Telecommunications Program (e-mail: henk@gmx.com).

R. G. Thekkekunnel is a student majoring in Electrical and Computer Engineering (e-mail: Roshan-george.Thekkekunnel@colorado.edu).



(a) UAV (b) Fixed (c) Ground Mobile

Figure 2. Heterogeneous Ad Hoc Radio Nodes

Table Mountain Field Site that is 15 km north of Boulder, Colorado with a Network Operating Center (NOC) located on the University of Colorado at Boulder (UCB) campus.

The infrastructure at the site includes range instrumentation, monitoring and telemetry support plus a high-speed fiber optic data network that has broadband access to the Internet for moving collected real time test data to the UCB campus NOC and the NOC servers. Campus and range engineers can in turn access these servers during and after experimental testing. The test bed is designed to meet design objectives that allow convenient and routine wireless communication testing while storing data in an easily accessible form for real-time and post-experiment analysis.

II. TEST BED

The test bed consists of three basic components: the actual test site, hardware components, and the software. Each of these is discussed in further detail in the following sections.

A. *Test Site*

The test site is a 7 km^2 large flat area in which the use of radio transmitters is controlled. Figure 1 shows a United States Geological Survey topographical map of the Table Mountain Field Site where experiments and demonstrations are conducted. The range is a flattop mesa. At the north end a graded area marked “gravel pit” serves as a ready-made UAV airstrip. Two buildings marked B9 and T22 serve as range operator areas.

B. *Hardware Components*

The network experiments use a combination of up to 5 ground nodes, 3 UAV nodes and 2 laptop-based nodes that are deployed around the Table Mountain facility. The 5 ground nodes can all be placed at specific Table Mountain range locations, or they can be placed on vehicles and driven around the range, depending on experimental requirements. Figure 2 shows these devices at a fixed field site, mounted on a ground and on an aerial vehicle.

Our approach for testing for this test bed consists of four efforts: (1) ad hoc network software, (2) communications hardware, (3) UAV platform, and (4) test bed monitoring architecture. The ad hoc network software combined with the communication hardware we denote the mesh network radio (MNR).

These MNRs that serve as wireless end-user nodes pose special challenges. They must support a variety of configurations that include being placed inside small UAVs, driven around in vehicles, incorporated into handheld communication devices or placed at various fixed sites. For the mobile environment, the MNR devices must be tough, small, and power efficient. Yet another design goal is that the nodes be built from low-cost commercial off the shelf (COTS) IEEE 802.11 radio components, computers and electronics.

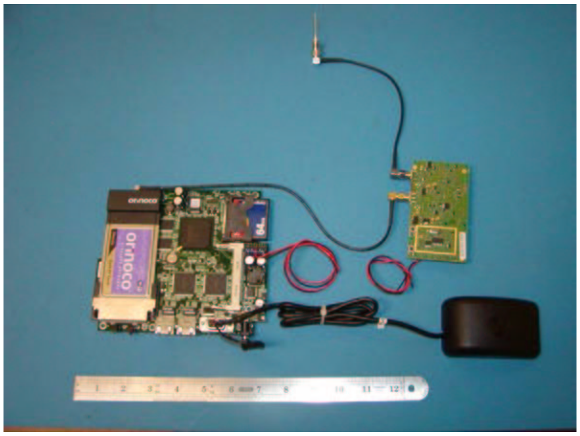


Figure 3. MNR Radio Components

The MNR hardware includes a Soekris Model 4511 single board computer, an Orinoco 802.11b card, a Fidelity-Comtech bidirectional amplifier with up to 1W output and a GPS receiver. A MNR is shown in Figure 3 without the enclosure. The integrated antenna enclosure is 21 cm (8.3 inches) across.

A design challenge was to select hardware conducive to outdoor environments that could be used in ground nodes, mobile nodes, as well as in UAV nodes. The enclosed and environmentally protected MNRs are used in fixed site deployments and mounted on vehicles that drive through and around the test range. The non-enclosed MNRs are placed inside the UAVs. This universality in addition to its relative endurance characteristics was the reason for selecting the Soekris single board computer (SBC) for this testbed. The Soekris SBC features a 486-class processor running at 100 MHz. It has 64 MB of RAM and a compact flash socket for flash memory storage up to 256 MB. The Soekris computer

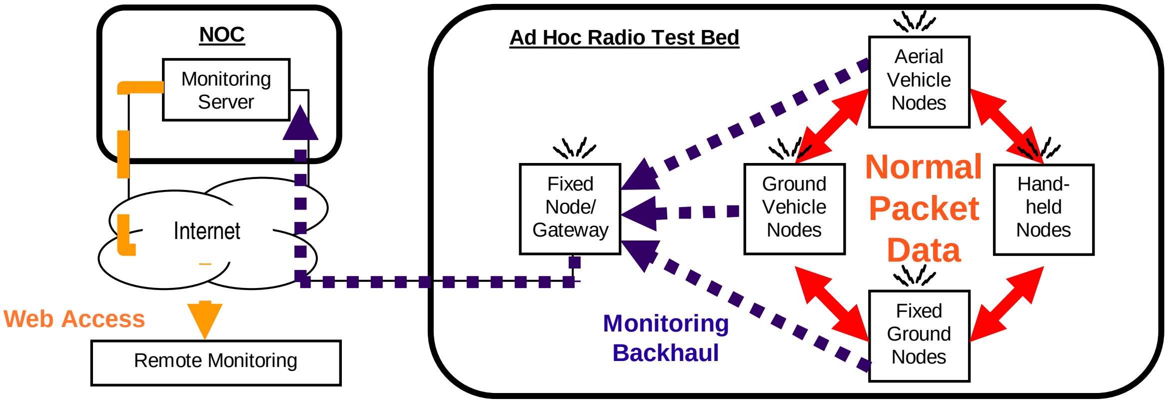


Figure 4. Normal data traffic (red solid) is monitored by each node. Periodically each node sends a report on the data (blue dotted) to the monitor server. This data can be viewed remotely over the Internet (yellow dashed) via a web-based GUI.

can run a variety of operating systems, including OpenBSD, NetBSD, Linux, and a number of real-time operating systems. Sockets for PCMCIA and miniPCI cards make it suitable for the test bed purposes. Two Ethernet ports as well as one RS232 serial interface allow for easy configuration and upgrade of the system. The RF amplifier gain is adjustable from 100mW to 1 W. The GPS receiver-sampling rate is 1 sample per second. The MNR requires an external power supply, which is provided by a small 7Ah sealed, lead calcium driven battery that powers the node for seven to eight hours.

The ad hoc radio is based on the 802.11b MAC protocol which is low-cost, has known behavior, and is readily available. The channel rate is fixed at 2 Mbps since this communicates over longer ranges than the higher rates while meeting a design target of one-hop throughputs in excess of 1 Mbps.

The UAV is a Telemaster-based design. The plane has a 15 kg maximum takeoff weight and 5 kg payload. Top speed of the plane is 170 km per hour. The plane has been constructed in-house at the University of Colorado using carbon fiber composites that make it lightweight but robust.

C. Software

A software design goal is that it be open source and available without license. The software runs under the Linux operating system (the WISP-Dist distribution, a stripped down version of Linux whose size is 8 MB) and is suitable for the radio nodes.

We used an implementation of DSR with the Click modular router [12]. Click has proven effective in several protocol implementations, e.g., the AODV implementation by Neufeld et al. [18], the DSR implementation by Doshi et al.[5] and the GRID project at MIT [21]. The router software can be configured to run at the user level using a driver program or in the Linux kernel as a kernel module. When Click runs in the user level mode, it requires a kernel tap that captures packets that are destined to or from the kernel. This allows the packets to be manipulated in the user-space and also allows for the re-insertion of the packets into the kernel stack. When Click runs as a kernel module, it can steal packets from the network devices before Linux gets an opportunity to handle them. It sends packets directly to the devices as well as to Linux for normal processing. The kernel module version of Click is used for the MNRs. This gives higher performance because the router runs as a part of the Click kernel module.

The radios run the dynamic source routing protocol (DSR) communicating with the other nodes via 802.11b. The reason for selecting DSR as the routing protocol is because it belongs to the on-demand family of ad-hoc routing protocols. DSR unlike table driven routing protocols has less overhead and only seeks a route to a destination when it has data to send. When a node needs to send a packet, it initiates a route request process among other nodes in the network to establish a route. DSR also uses source routing whereby a packet source precisely specifies which route the packet will follow. We have implemented DSR ourselves using the Click modular router. With our own implementation we are free to modify the protocol as per our experimental needs. The DSR software has been ported to a number of other devices including laptops and handheld computers.

The nodes are stateless as they use a RAM file system and no permanent state is stored on the node. This makes the system more robust to power interruptions and other disturbances during operation.

III. MONITORING ARCHITECTURE

There are several challenges that must be solved for a test bed to be effective. The data from the test bed should be available in real time and minute details of the routing behavior should be monitored for further data analysis. The ad hoc networking is complex with control distributed across the ad hoc nodes. The test bed should scale to 10's of monitored nodes. Hence, the monitoring scheme should have minimal impact on the normal operation of the network.

These constraints limit some approaches. The real time collection requirement precludes simply storing monitoring data on each node to be collected after the experiment. The limited UAV payload forces the use of the ad hoc network itself for collecting monitoring data rather than a separate radio for monitoring feed back. The distributed behavior suggests that data has to be centrally collected and correlated between nodes. The scaling and interference constraints imply that the monitoring should use minimal computing, storage, and bandwidth resources. The monitoring approach is shown in Figure 4.

A. *Architecture*

Monitoring is done at a routing level. The monitoring collects the packet statistics of time of arrival, type of packet, packet sequence number, and packet size. The type of packet field indicates if the packet has been received or sent to transmit and the transport layer type (UDP/TCP/ICMP). It also logs information about the DSR control packets like Route Requests, Route Replies and Route Errors.

The monitoring also collects information about UTC time, latitude, longitude and altitude of the node's current position from the GPS attached to the serial port of the node. A unique feature of the monitoring is the inclusion of optional user defined text messages to annotate events during operation of the network. Scripts running on the node can send messages to the monitoring module to be included in the monitoring information.

B. *Packetization and Routing*

The information collected by the monitoring module is packetized and a monitor sequence number is added to the packet that is unique per node. Packetization is triggered every 10 seconds or whenever the estimated packet size of the monitoring information equals 1000 bytes. The module now buffers the monitoring packet and a packet copy is passed on as an application layer packet to a module that adds to it a UDP/IP header with its destination as the gateway node. This packet now is passed to the DSR router that routes this packet to the gateway node over the mesh network.

C. *Reliable Delivery to the Gateway*

The gateway node DSR router receives the DSR source routed monitoring packet, strips off the DSR header and recognizes it as a monitoring packet. It then sends back a Monitoring ACK packet to the node that sourced the monitoring packet. The node on receiving the ACK removes the corresponding monitoring packet from its buffer and is clear to transmit the next monitoring packet it has lined up in the buffer. If the node does not receive the ACK packet, it keeps retransmitting the same monitoring packet till it eventually gets an ACK from the gateway for that packet. Each retry occurs every 10 seconds. If the packet is buffered for over 1 hour it is dropped and the next packet in the buffer is transmitted.

D. *Gateway to UCB monitoring server*

The gateway on receiving the monitoring packets redirects them to the UCB monitoring server by modifying the UDP/IP header and forwarding these modified packets to the test-site router. This router then routes the packet to the UCB monitoring server via the Internet by reading the IP address off the new IP header.

IV. IRIDIUM LINK

The test site has a T1 backhaul to the Internet. Alternate backhauls were considered if the test bed were to move to a test site without Internet connectivity. For this purpose, an Iridium satellite link was tested. A single Iridium phone has a nominal 2.4 kbps data throughput. The data throughput can be increased by aggregating multiple phones using a Multi-Link Point-to-Point Protocol (MLPPP) connection. MLPPP is natively supported in Microsoft Windows XP. Using the Internet Connection Sharing feature in Windows any computer can be configured to act as a gateway, providing Internet access to any connected node on the Test Bed. The Internet service is through Iridium Satellite LLC, which provides access to their Internet gateway.

A phone connects to the computer through a serial port. Multiple Iridium phones are connected via serial cables to a 4 port serial to USB adapter. This allows any laptop with one available USB port to control up to 4 Iridium phones. The Iridium phone is Motorola model number 9505. The Motorola 9505 functions exactly like a standard 2400 bps modem requiring only an additional initialization string.

V. EXPERIMENTATION PLAN

The goal of the experimentation was two-fold. One was to demonstrate that a group of UAVs could successfully form a network among themselves. The second was to show the effect a UAV might have on a group of nodes on the ground. To this end, 28 experiments were designed. Each experiment was a combination of a deployment scenario and a test procedure. Five of these procedures are described below.

1. Throughput: TCP connection throughput when no other traffic is present.
2. Latency: 1 sec ping connectivity and delay when the network is lightly loaded.
3. Congestion: TCP connection throughput when there are competing data flows.
4. Subjective: typical network application performance as perceived by a user.
5. Range: throughput as a function of separation between two nodes.

These experiments were carried out over a period of 4 months with over 40 test days and 60 UAV flights.

VI. LESSONS LEARNED

This section discusses the implications of the testing conducted on the test bed.

A. *Fundamental Test bed Capabilities*

The test bed demonstrated that a combination of airborne and ground nodes can participate together to produce a useful communication network. The ad hoc networking enables networks to form quickly and automatically as nodes come in range of each other.

We observed good performance with broadband connectivity (about 250 kbps) and low latency (30 ms) suitable for real-time applications on a network with up to 3 hops.

B. *Mesh Network Radios*

The MNR design worked well for our experiments. The radios were generally robust to weather and manhandling. They operated for long periods (7-8 hours) with a small lead acid battery. Occasionally the software on the boards would hang-up or reboot, but, these events reduced over time as the software was modified and improved. The stateless software design prevented problems with abrupt power outages and power-cycle reboots. However we observed a large variation in performance between laptop nodes and MNRs. In addition the MNRs had varying performances based on whether the routing software was run in the userlevel or in the Linux kernel. The results are listed in table 1.

TABLE 1

1 Hop Throughput (Mbps)			1 Hop Delay (ms)		
Laptop	MNR		Laptop	MNR	
	User level	Kernel		User Level	Kernel
1.5	0.60	1.3	6	20	10

Processing capabilities of the MNR were found inadequate to simultaneously support the networking, monitoring, and application software. Additionally a limit to network performance was found with the 100 MHz MNR processor. Throughput was always 15% higher when using faster laptop computers rather than the MNR single board computer (SBC). A processor with at least twice the computing speed of the 100 MHz 486-based processor is needed.

C. *Software*

Our software runs on the WISP-Dist distribution of Linux. We see potential for improvement in switching to another distribution of Linux called Pebble which offers better measurement capabilities and a larger tool-set of Linux tools. The only tradeoff in using Pebble is that it is 64MB in size as opposed to 8 MB occupied by the current WISP-Dist distribution. In addition the use of Atheros cards, which are miniPCI to provide antenna diversity and a semi open-source MAC layer, would be an improvement over the current system. The use of miniPCI cards is again a good solution as it eliminates vibration problems to a large extent while saving space and reducing the weight of the radio.

In terms of network performance measurements, we identified some issues and possible solutions for the same. The first issue was the lack of a proper throughput measurement tool for wireless scenarios. Though Netperf was adequate for most measurements with fixed nodes, it is inadequate for highly variable scenarios such as with mobile nodes. The Netperf tool relies on control packet exchanges at the start and at the end of a test without which it cannot obtain results. This led to the tests going into long stalls when nodes went out of range. To counter this we had scripts written which would kill the Netperf process after a specified duration of time and report the throughput as 0 kbps.

Antenna pattern and UAV banking effects: The UAV had only one antenna on the bottom center of the main fuselage and thus the antenna pattern was not isotropic. This when combined with the banking of the UAVs during flight led to temporary outages in the radio link. A possible solution would be spatial antenna diversity. Full-size planes often mount antennas above and below the fuselage. The Orinoco 802.11b cards have only a single antenna connector, but other cards with diversity antennas could be deployed to overcome this problem.

DSR timing optimization: Temporary outages in radio links adversely affected the throughputs. Careful measurements showed that the UAVs banked typically for durations of 3-4 seconds while completing a turn and our protocol was not fine-tuned to react to such temporary outages. We were able to go back to the design of our DSR protocol and tune various timeout parameters; namely routing level acknowledgment (Ack) timeouts and routing level packet Ack detection in order to improve the network performance, especially when faced with temporary outages in the presence of UAVs. For all tests the DSR protocol had timeouts set to 5 seconds. This could be reduced to a much lower value of 500 ms which is still higher than the typical round trip time (RTT) found in this network in the presence of heavy traffic. Also the protocol was set to check for Acks for every 10th packet in order to reduce processing at each node. The best value in order to make the protocol reactive to temporary outages would be to check the Acks for every packet, but this would increase the processing cost tremendously and we observed hits in throughput values. Hence an optimum value of 3 was chosen for this parameter. With these set of values we observed a hit of 6% in throughput values, however we could determine that with these new settings the protocol was sensitive to temporary outages spanning up to 3 seconds. The tradeoff here is that one can make the DSR protocol very reactive; however this would be at the cost of increased processing which would conversely result in performance degradation. Hence a fine balance needs to be maintained between the two choices.

D. *Monitoring*

The monitoring software was able to collect detailed per-packet information. In a medium load network scenario 360 monitoring packets were generated per hour with an aggregate size of 144 Kbytes. An earlier design that only collected aggregate packet statistics was found to be inadequate to discern performance events. With the new design, network activity can be displayed by route, by packet type, and by time. The graphical user interface (GUI) is important for understanding the complex network data and often helped explain anomalous behavior observed in testing. The automatic capture of data and entry into the database provides a permanent, accessible archive of the testing. It also enabled a real-time situational awareness map showing the node positions and network activity. This was important for evaluating the progress of testing that was not possible from a single point on the test bed.

E. Iridium Backhaul Link

The Iridium backhaul combined four separate phones to create a single aggregated link. This combined link achieves a throughput of 6.7 kbps with a delay of 1 sec. This works well for monitoring data traffic. For web browsing, the connection is slow but usable. For real time traffic to the Internet, this was not usable. The connections lasted for hours at a time when the phones were placed with a clear view of the horizon. When placed among buildings, connections would often drop.

VII. CONCLUSION

The test bed is an invaluable resource for evaluating networking technologies. This capability has been demonstrated in a basic assessment of a wireless network of terrestrial and aerial 802.11b nodes. The test bed can continue to operate as an assessment tool for new technologies. Based on our initial work and work in the foreseeable future we envision the following technologies that can be assessed with the technology:

1. Different physical layers such as lower frequency transmitters.
2. Alternative environments including urban and mountainous.
3. Alternative IEEE 802 wireless MAC protocols including 802.11a, 802.11g, and 802.16 (when available).
4. Alternative ad hoc routing protocols.
5. New hardware packages and antennas.
6. Network applications and scenarios such as cooperative UAV operations
7. Security and/or jamming protocols and hardware.

Current projects for the test bed are investigating ad hoc networking between multiple small UAVs for so-called flocking applications.

REFERENCES

[1] Chambers, B.A., "The Grid Roofnet: a Rooftop Ad Hoc Wireless Network," *Master's Thesis, Massachusetts Institute of Technology, MA,* June 2002

[2] Corke, P., Hrabar, S., Peterson, R., Rus, D., Saripalli, S., and Sukhatme, G., "Autonomous deployment and repair of a sensor network using an unmanned aerial vehicle," *ICRA 2004*, 2004

[3] Chin, K., Judge, J., Williams, A., Kermode, R., "Implementation Experience with MANET Routing Protocols", *Comp. Comm. Rev.*, 32(5):49-59, Nov. 2002.

[4] Desilva, S., Das, S., "Experimental Evaluation of a Wireless Ad Hoc Network," *Proc. of the 9th Intl. Conf. on Comp. Comm. & Networks*, Oct. 2000.

[5] Doshi, S. Bhandare, S., Brown, T. X , "An On-demand minimum energy routing protocol for a wireless ad hoc network," *Mobile Computing and Communications Review, vol. 6, no. 2*, July 2002

[6] Gu, D.L., Pei, G., Ly, H., Gerla, M., Zhang, B., Hong, X., "UAV aided intelligent routing for ad-hoc wireless network in single-area theater," *IEEE Wireless Comm. & Networking Conf.*, no. 1, Sep. 2000, pp. 1220–5

[7] Heusse, M.; Rousseau, F.; Berger-Sabbatel, G.; Duda, A.; "Performance anomaly of 802.11b," INFOCOM 2003. *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*. IEEE, Volume: 2, 30 March - 3 April 2003 Page(s): 836-843

[8] Jin, Z., Liang, B., Shu, Y., Yang, O.W.W., "Designing and Implementing A Mobile Ad hoc Network Testbed," *Proc. of the IEEE Canadian Conf. on Elec. & Comp. Eng.*, pp.1559–64, Winnipeg, May 12-15, 2002

[9] Johnson, D., Maltz, D., "Dynamic Source Routing in Ad HocWireless Networks," *Mobile Computing*, Chapter 5, pp. 153-181, Kluwer, 1996

[10] Kaba, J.T., Raichle, D.R., "Testbed on a Desktop: Strategies and Techniques to Support Multi-hop MANET Routing Protocol Development," *Proc. of the 2nd ACM Int. Symposium on Mobile Ad Hoc Networking & Computing* 2001, Long Beach, CA

[11] Ke, O., Maltz, D.A., Johnson. D.B., "Emulation of Multi-Hop Wireless Ad Hoc Networks," *Proc. of the Seventh Int. Work. on Mob. Multimedia Comm.* (MOMUC), IEEE Comm. Soc., Tokyo, Oct. 2000

[12] Kohler, E., Morris, R., Chen, B., Jannotti, J., Kaashoek, M.F., "The Click Modular Router," *ACM Tran. on Comp. Systems*, vol. 18, no. 3, pp. 263–297, August 2000. http://www.pdos.lcs.mit.edu/click

[13] Lundgren, H., Lundberg, D., Nielsen, J., Nordstrom, E., Tschudin, C., "A Large-scale Testbed for Reproducible Ad hoc Protocol Evaluations," *3rd annual IEEE Wireless Comm. & Network Conf.* (WCNC) 2002

[14] Maltz, D.A., Broch, J., Johnson, D.B., "Experiences Designing and Building a Multi-Hop Wireless Ad Hoc Network Testbed," CMU School of Computer Science Technical Report CMU-CS-99-116, March 1999

[15] Maltz, D.A., Broch, J., Johnson, D.B., "Quantitative Lessons From a Full-Scale Multi-Hop Wireless Ad Hoc Network Testbed," *Proc. of the IEEE Wireless Comm. & Network Conf.*, IEEE, Chicago, Sep. 2000

[16] Morris, R., Jannotti, J., Kaashoek, F., Li, J., De Couto, D., "CarNet: A Scalable Ad Hoc Wireless Network System," *9th ACM SIGOPS European Work.*, Kolding, Denmark, September 2000

[17] Mustafa E., Lee D., Datta R., Ko J., Puri A., Sengupta R., Varaiya P, "WTRP – Wireless Token Ring Protocol," submitted to the *IEEE Transactions on Vehicular Technology,* October 2002

[18] Neufeld, M. , Jain, A., Grunwald, D., "Nsclick: Bridging Network Simulation and Deployment," *MSWiM* 2002

[19] Raychaudhuri, D., Seskar, I., Ott, M., Ganu, S., Ramachandran, K., Kremo, H., Siracusa, R., Liu H., Singh, M., "Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols," *Proc. of the IEEE Wireless & Network Conference* (WCNC), 2005

[20] Sanghani, S., Brown, T.X, Bhandare, S., Doshi, S., "EWANT: The Emulated Wireless Ad Hoc Network Testbed," *Proc. of the IEEE Wireless Comm. & Network Conference* (WCNC), 16-20 March, 2003

[21] The Grid Ad Hoc Networking Project at MIT, http://www.pdos.lcs.mit.edu/grid/index.html

[22] Weber, S., Cahill, V., Clarke S., Haahr, M., "Wireless Ad Hoc Network for Dublin: A Large-Scale Ad Hoc Network Test-Bed," *ERCIM News*, vol. 54, 2003.

[23] Xu, K., Hong, X., Gerla, M., Ly, H., Gu, D.L., "LANDMARK Routing In Large Wireless Battlefield Networks Using UAVs," Proc. of *IEEE Military Comm. Conf.* (MILCOM), McLean, VA, Oct. 2001.

[24] Zhang, Y., Li, W., "An Integrated Environment for Testing Mobile Ad-Hoc Networks," *Proc. of 3rd ACM Int. Symposium on Mobile Ad-Hoc Networks & Comp. (MobiHoc)*, pp. 104-11, Lausanne, Switzerland, Jun. 2002