

# Experimental Evaluation of RF Modems for Use in Fleets of Multiple Cooperating Autonomous Undersea Vehicles

Radim Bartoš, Venkata S. Gorla,  
Leon N. Cyril, and Rohit Sharma  
Department of Computer Science  
University of New Hampshire, Durham, NH 03824  
Email: {rbartos, vsx2, lno2, rohits}@cs.unh.edu

Rick J. Komerska and Steven G. Chappell  
Autonomous Undersea Systems Institute  
Lee, NH 03824  
{komerska, chappell}@ausi.org

**Abstract**— This paper presents the results of the experimental evaluation of three radio frequency (RF) modems for use as communication infrastructure among multiple surfaced cooperating autonomous undersea vehicles (AUVs), gateway buoys, and land or ship based operators. RF modems are inherently more complex than their wired counterparts which makes it difficult to estimate the performance they deliver to an application. Throughput, communication latency, and latency variation (jitter) are used as the measures of performance. The experiments were designed to subject the modems to the traffic patterns common in the AUV fleets. The results of the presented experiments should help to set realistic expectations of RF modem performance and aid in the design of comprehensive communication solutions for AUVs.

## I. INTRODUCTION

The complexity of scenarios in which underwater and above-water communication networks are deployed is steadily increasing. Simple point-to-point methods designed for communication between an operator and a single underwater vehicle are no longer sufficient to satisfy the requirements of the missions that are envisioned for fleets of multiple cooperating autonomous vehicles. Communication infrastructure to support such missions must address, among many other issues, an efficient use of the available communication channel capacity, shared channel access control, multi-hop message delivery, and must support a wide range of communication demands from the applications. The research reported in this paper is motivated by the long-endurance Solar-powered Autonomous Undersea Vehicles (SAUVs) developed by AUSI and its partner organizations with funding from the Office of Naval Research.

Transmission rate and latency are the basic measures used to evaluate a communication link. This study presents these measures for all modems under consideration. Additionally, tests of latency variation (jitter) were carried out. Even though many modems present their services to the users as a byte-oriented serial port, they often internally use some form of multi-byte frame oriented communication. This feature impacts the latency that the user experiences depending on the fit between the framing mechanism in the modem and the actions of the user. This is significant since many uses

of modems involve some form of user packetization. Since mismatch between modem and user packetization may lead to degraded performance, it is important to understand whether and how internal modem packetization is implemented in a device. Some RF modems support multi-hop communication, often in a way that is completely hidden from the user. While this simplifies the setup and use, it is important to understand the impact such techniques have on the modem performance. An intermediate node consumes energy for transmissions of forwarded messages and this action may not be reported to the energy manager of the vehicle. Furthermore, the transmission capacity of this node can be reduced.

## II. RF MODEMS

Three RF modems have been used in the experiments described in this paper. FreeWave [1] shown in Fig. 1, MaxStream [2] shown in Fig. 2, and NovaRoam [3] shown in Fig. 3. Table I outlines and compares some of the basic characteristics of the modems. Although the FreeWave modems have been widely used by the AUV community, the trend toward networking together AUVs and other ocean instrument platforms motivates us to explore other RF solutions which better support these networking aspects.

Unlike their wired counterparts, RF modems are relatively complex devices. All modems under consideration used some form of internal packetization, supported addressing of transmissions, and allowed point-to-multipoint communication. The MaxStream and NovaRoam modems support strong encryption of transmitted data. The rest of the section focuses on three aspects of the RF modem design: the connectivity with the host computer, ease of control and management, and the support for multi-hop transmissions.

### A. Modem Interfaces

Each of the three modems under consideration used a different technology for interfacing with the host computer. The FreeWave modem is equipped with a standard serial port, the MaxStream modem connects using a USB port

TABLE I  
RF MODEM OVERVIEW

	<i>FreeWave</i>	<i>MaxStream</i>	<i>NovaRoam</i>
Manufacturer	FreeWave Technologies, Inc.	MaxStream, Inc.	Nova Engineering, Inc.
Model	FGR-115RC	XTend-PKG-U	EH900
Dimensions (mm)	165 × 74 × 57	140 × 70 × 29	160 × 132 × 33
Weight (g)	441	200	400
Band	900 MHz ISM	900 MHz ISM	900 MHz ISM
Transmit power	5 mW to 1 W	1 mW to 1 W	10 mW to 1 W
RF data rate	9600/19200 bps	9600/115200 bps	100/400 kbps
Host interface	serial	USB	Ethernet/serial
Multi-hop routing	static	no	dynamic



Fig. 1. FreeWave RF modems



Fig. 2. MaxStream RF modems



Fig. 3. NovaRoam RF modems

that emulates a serial port, and the NovaRoam modem provides Ethernet connectivity together with a serial port. OEM versions of the modems are available, having different size and weights. Performance characteristics, however, should be identical. For example, the MaxStream modems also support serial and Ethernet versions of the model we tested. As will be shown later, the modem performance is mostly limited by the capabilities of the RF interface. The different choices of the host interface have no significant impact on the overall performance. Availability of the particular interface technology on the host system is the most significant factor. Older, embedded computers may not support USB or Ethernet interfaces.

The only issue that we encountered with the host interfaces was with the USB interface of the MaxStream modems and the Linux operating system. The USB product identifier (PID) of the modem is not recognized by the current release of Linux kernel drivers even though the chipset used in the modem has been supported by the kernel for several years. A minor edit of the *ftdi\_sio* module source code and recompilation addressed the problem. A brief note describing the fix, directly applicable only to an old version of the kernel, is provided by the

manufacturer. The note, however, contains enough information to apply the fix to the current version of the kernel module.

### B. Modem Configuration and Management

Ideally, a modem should provide a dual interface for configuration and management. One for use by a human operator, where ease and intuitiveness of use are important factors, and a second to be used for automated control and management by a program.

1) *FreeWave configuration*: The FreeWave modems use a simple, menu-based console interface. The main issue that significantly impacts the usability of the modem is that the modem can be brought to a control mode only by either physically pressing the Reset button or by power cycling the modem. There is no command, equivalent to “+++” in the standard Hayes command set, that would bring the modem to the command mode. To overcome this limitation, embedded real-world installations of the modem, such as those in the SAUV, resort to the use of vehicle power management to cycle the modem’s power in order to put it in a re-configuration mode.

2) *MaxStream configuration*: The MaxStream modems use a rich set of AT commands to get the status of the modem and to control all aspects of its operation. A special binary mode is available to further simplify the control of the modem from a program by eliminating the string to value conversion. A simple Windows-based application comes with the modem to read/write configuration parameters and to run simple tests. It acts as a front-end for issuing modem commands and does not provide any additional controls. While useful, it is not essential for the operation of the modem.

3) *NovaRoam configuration*: Configuring the NovaRoam modems is possible only through a web-based interface similar to those found on home-use routers. The web interface uses non-standard features that prevent the pages from being rendered in a usable form in browsers other than Microsoft Internet Explorer. This is a significant problem when the host computers are not Windows based. There is no obvious way to manage the device from a program. Given the anticipated types of applications, it would be highly desirable to have, for example, SNMP-based management capabilities such as those available in commercial-grade 802.11 access points.

The modem lacks an obvious way to bring itself to a factory default state without having access to the web interface. It would be highly desirable to have a reset button on the device.

### C. Multi-hop Communication

The transmission range of a RF device is always limited and, therefore, in scenarios where more than a pair of modems is deployed, it is desirable to use the modems not only as a source and a sink of traffic but also to use them as relay nodes capable of forwarding traffic to nodes that are not in direct communication range. The FreeWave and NovaRoam modems support multi-hop communication. In the case of the FreeWave, the routes are static and set up by the operator using modem management. There is no provision for automatic

rerouting due to a network topology change or a node failure. If a route is set to use an intermediate node, the traffic is routed through it even if the source and target nodes are in communication range. Failure of the intermediate node will cause a loss of connectivity.

The NovaRoam modems in the routing mode use the standard AODV ad hoc routing protocol [4], [5] that supports automatic route discovery and rerouting in cases of a topology change or a node failure. From the perspective of the end users the wireless network appears as a single IP subnet. The actions of the ad hoc routing protocol are completely transparent to the end-user.

## III. EXPERIMENT METHODOLOGY

In all the experiments, the modems were connected to laptops running Fedora Core 5 distribution of the Linux operating system with the default build of the kernel version 2.6.16-1.2122\_FC5. The kernel module *ftdi\_sio* was modified and recompiled to make it recognize the MaxStream modems as valid USB serial devices.

### A. Throughput Measurements

Experiment design is based on expected types of network traffic which may be encountered in fleets of multiple cooperating autonomous undersea vehicles. The throughput experiments emulate file transfer commonly found when mission plans are downloaded into the vehicles and collected data are uploaded to a shore station. Such transfers typically use some form of acknowledgment-based error control. The presented experiments utilize standard kermit file transfer [6] timing to determine the application data throughput. Default kermit parameters were used in the experiments unless noted otherwise.

In the case of the NovaRoam modems which support Ethernet/IP connectivity, *netperf* [7], a widely used tool to benchmark network performance, was used. To account for the cost of providing reliable application data transfer, the TCP stream throughput test of *netperf* (TCP\_STREAM) was chosen.

### B. Latency and Jitter Measurements

Command and control messages together with emergency signaling form another important traffic category in the fleets of AUVs. While their network throughput requirements are often small, the latency they experience and its variation (jitter) should be kept at the minimum. In the presented experiments, two latency scenarios were considered. In the first scenario, the round-trip time of a single character was measured using two custom programs, one that measures the time taken to send and receive back a single character and a second that acts as a simple character echo client, sending back every byte it receives.

The second scenario emulates a request-response message exchange. In this case, blocks of 64 bytes were sent, completely received by the echo client and then sent back. The time to complete the entire transaction was measured. The

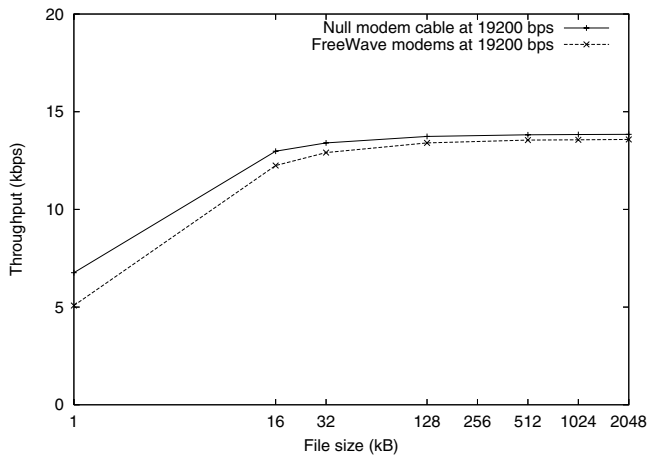


Fig. 4. Throughput of FreeWave modems tested using kermit file transfer

motivation for this experiment was to determine the increase in round-trip latency as the message length increases.

The latency measurements obtained in this set of experiments were also used to calculate the jitter. From the several methods used to calculate jitter [8], we picked *average jitter*, defined as the average of the absolute values of differences between message latency and the average latency, and *min/max jitter*, defined as the difference between the maximum and the minimum message latencies. Average jitter approximates how much the latency deviates from the average, min/max latency gives the worst case scenario which would be useful to know though it may not occur frequently. The results presented below indicate significant differences between the values of the two jitter calculation methods.

The *netperf* program was used in the testing of the NovaRoam modems. The experiments were conducted using the TCP request-response test (TCP\_RR) for message lengths of 1 and 64 bytes. It has to be noted that, unlike the case of request-response experiments for serial ports, 40+ bytes of IP, TCP, and TCP option headers were added by the protocol stack to the load sent by the modems. All *netperf* experiments were performed at  $\pm 20\%$  accuracy with 95% confidence.

#### IV. EXPERIMENT RESULTS

##### A. Throughput Experiments

As outlined in the previous section, two methods were used to measure the application throughput of the modems: kermit file transfer and TCP throughput tested using *netperf*. Figs. 4 and 5 show the throughput of the FreeWave and MaxStream modems respectively. In both cases, the throughput achieved over a null modem cable at the same serial port rate is included to show the impact of RF modem use. Table II gives the results of *netperf*-based throughput experiments for NovaRoam modems.

##### B. Distance Experiments

The measurements of the maximum throughput vs. distance were carried out by setting up one modem on the third floor

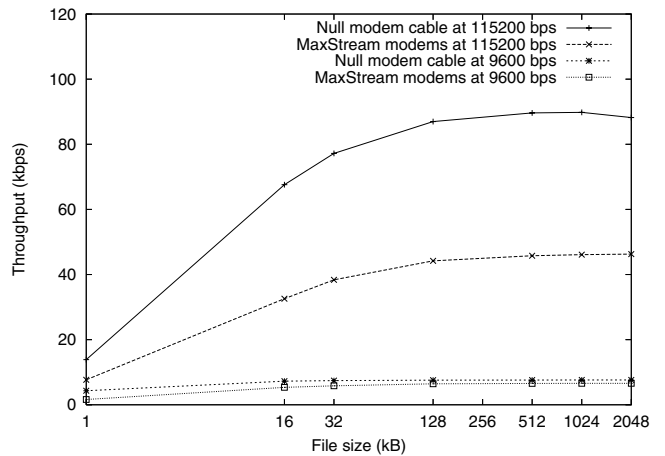


Fig. 5. Throughput of MaxStream modems tested using kermit file transfer

TABLE II  
THROUGHPUT AND ROUND-TRIP LATENCY OF NOVAROAM MODEMS

Experiment	RF Data Rate (kbps)	Router mode	Bridge mode
TCP throughput (kbps)	100	48.3	37.2
	400	203.5	186.4
TCP 1-byte request-response (ms)	100	54.4	52.9
	400	14.5	12.3
TCP 64-byte request response (ms)	100	70.0	63.2
	400	19.0	17.7

of a UNH campus building. The other modem was located in a parked car at various points on the university campus. For most of the test-car locations, there were trees and overhead wires but no buildings in the line of sight. The campus is relatively flat. All experiments were carried out with modems set to their maximum transmit power (1 W for all). Lower transmit powers led to significantly reduced range. Significant spatial fluctuations in performance were observed during the experimentation. It is very likely that significantly different results will be obtained in a different setting. The value of the presented measurements is in the relative comparison of modem performance. The results of throughput vs. distance tests in Figs. 6, 7, and 8 show that the farthest distance was achieved by the FreeWave modem, followed by NovaRoam and MaxStream.

##### C. Latency Experiments

Just like the throughput tests, the latency experiments were carried out with the modems in relatively close proximity in an environment with no significant interference. The test program reports latency as experienced by the application. A part of the measured latency is due to the processing of the transmitted data by the operating systems and the serial port driver. To show the magnitude of the unavoidable operating system

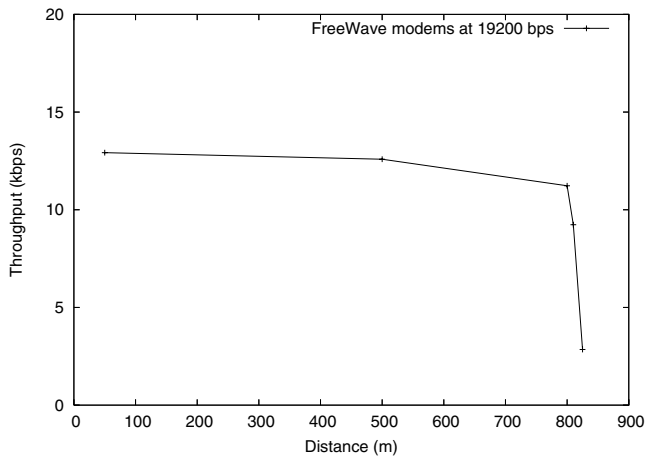


Fig. 6. Throughput vs. distance of FreeWave modems tested using kermit 32 KB file transfer

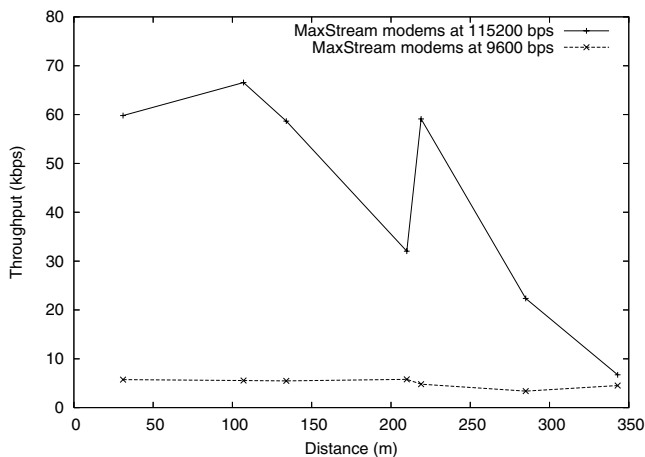


Fig. 7. Throughput vs. distance of MaxStream modems tested using kermit file transfer (file sizes 32 KB for 9600 bps experiments and 128 KB for 115200 bps experiments)

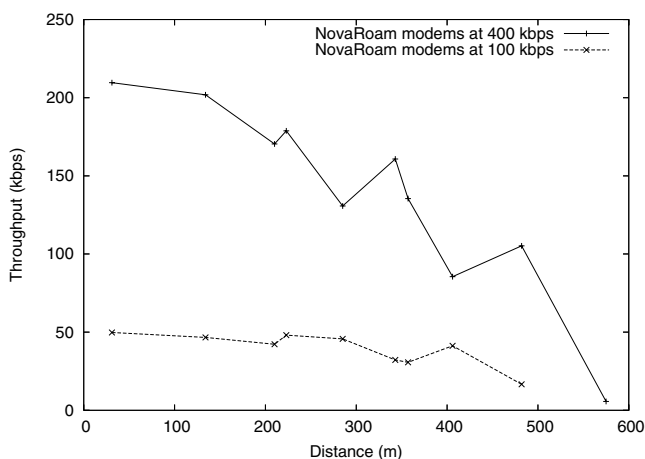


Fig. 8. Throughput vs. distance of NovaRoam modems tested using *netperf*

TABLE III  
LATENCY AND JITTER FOR 1-BYTE REQUEST-RESPONSE EXPERIMENTS

Modem Type	Rate (bps)	Latency (ms)	Jitter (ms)	
			Average	Min/Max
Loopback connector	9600	5.85	0.08	6.51
	19200	2.95	0.12	1.21
	115200	0.90	0.04	1.29
Null modem cable and a character-loopback client	9600	15.70	0.48	12.01
	19200	7.97	0.31	14.00
	115200	4.01	0.59	44.73
FreeWave w/o repeater	19200	20.95	2.33	21.06
	with a repeater (2 hops)	19200	88.40	14.41
MaxStream	9600	46.17	3.16	33.01
	115200	45.09	4.64	17.89

latency, two additional experiments were carried out. One with a simple loopback connector that feeds the transmitted data back to the receiver giving the time spent at the source of the request. The second experiment, where the RF modem link is replaced with a null modem cable, captures the processing on the responding side. Results for both tests are shown for the same serial port rates as those used for communication with the RF modems.

The results for 1-byte request-response and 64-byte request-response experiments are summarized in Tables III and IV respectively. The 64-byte request-response test was not carried out for the FreeWave modems due to their unavailability for this experiment. Overall, the inclusion of RF modems caused an increase in latency in the order of several hundreds of percent regardless of the type of the RF modem used. While the average jitter remained relatively low, there was significant min/max jitter. Since this was observed even in the case of the null modem cable experiments, we attribute it to the interrupt handling within the operating system. The latency of a 2-hop static FreeWave route was found to be more than four times that of a single hop communication. Furthermore, significant average and min/max jitter were observed.

The latency of the NovaRoam modems in the router and bridge modes, measured using *netperf*'s TCP-based request-response tests, is shown in Table II. The measured latency for the bridge mode was slightly less than the latency for the router mode. This was expected since the bridge mode requires less processing at the end nodes than the router mode.

#### D. Additional Experiments

The MaxStream modems offer three RF modes of operation: *streaming mode*, *acknowledged mode* (default), and *multi-transmit mode*. In the streaming mode the transmissions are not acknowledged which, according to the manufacturer, leads to improved latency and jitter performance at the expense of reduced immunity to interference. The acknowledged mode introduces acknowledgments and attempts to retransmit up to a user controllable number of times. The multi-transmit mode

TABLE IV

LATENCY AND JITTER FOR 64-BYTE REQUEST-RESPONSE EXPERIMENTS

Modem Type	Rate (bps)	Latency (ms)	Jitter (ms)	
			Average	Min/Max
Null modem cable and a message-loopback client	9600	135.76	0.13	2.01
	115200	14.24	1.90	6.08
MaxStream	9600	319.98	1.33	32.01
	115200	80.46	1.33	14.26

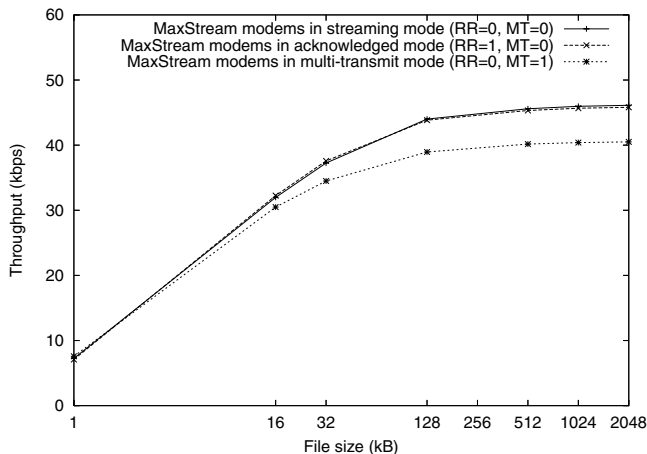


Fig. 9. Throughput of MaxStream modems under different modes (RF rate 115200 bps)

uses multiple (user specified) forced transmissions of the same packet to achieve higher reliability at the expense of reduced throughput and increased latency. The presented experiments measured the impact of the modes on the throughput under relatively favorable conditions (modems in close proximity and no significant interference). Fig. 9 shows that there was no significant degradation in throughput between the streaming and the acknowledged modes. This can be attributed to relatively large kermit data packets compared to the size of the acknowledgments. Two copies of the packet were sent in the case of multi-transmit mode. The throughput degraded slightly compared to the other modes, suggesting that the data transmission time, which is doubled in this mode, is only one of the factors that affect throughput performance.

Even though the MaxStream modems provide byte-oriented serial connectivity, internally they use a framing mechanism and send sequences of characters as packets. Packetization adds overhead and can lead to potentially undesirable cross-layer interferences. Many applications that utilize underlying serial communication have their own framing or packetization method. Performance degradation may occur in the case of a mismatch between packet lengths of the application and that of the modem. Kermit file transfer is an example of an application that divides transmitted data into frames and also performs its own error control. Fig. 10 shows throughput vs. the maximum RF packet size for the MaxStream modems tested using 512

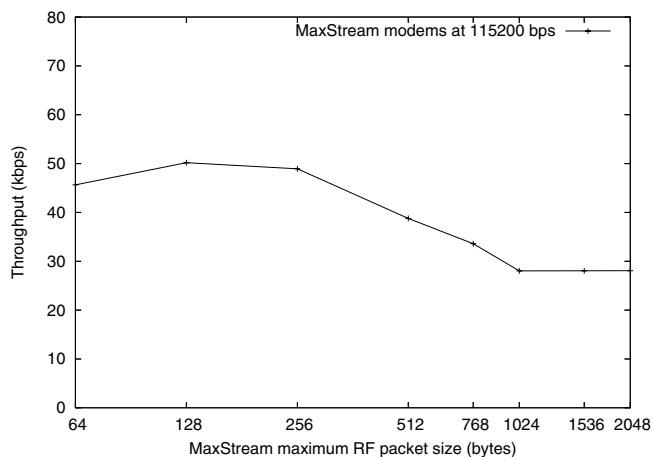


Fig. 10. Throughput vs. maximum RF packet size for MaxStream modems tested using 512 KB kermit file transfer at 115200 bps RF rate, kermit packet size fixed at 1024 bytes

KB kermit file transfer at the 115200 bps RF rate, with the kermit packet size fixed at 1024 bytes. The results suggest that a shorter RF packet size leads to higher throughput.

## V. CONCLUSION

This paper presents results of experimental evaluation of performance of three types of RF modems having application as surface communication infrastructure for multiple cooperating AUVs. A testing methodology, based on AUV network usage scenarios, has been devised and the results of experiments testing throughput, range, latency, and jitter performance are presented. Understanding of RF modem performance characteristics will, among other uses, aid in the design of comprehensive higher-level communication methodologies for AUVs.

There were more experiment results that could not be included here due to the space limitation. Please, check <http://www.cs.unh.edu/cnrg/rf-modems> for an extended version of this paper.

## ACKNOWLEDGMENT

This research was supported in part by grant N00014-05-1-0666 from the Office of Naval Research.

## REFERENCES

- [1] FreeWave Technologies, Inc., “FreeWave Spread Spectrum Wireless Data Transceiver – User Manual,” Version 2.6, 2004.
- [2] MaxStream, Inc., “9XTend-PKG-U USB RF Modem – Product Manual,” Version 1.2.4, 2005.
- [3] Nova Engineering, Inc., “NovaRoam EH900 Mobile Router – User Manual,” Rev. 1.4, 2006.
- [4] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” IETF RFC 3561, July 2003.
- [5] Nova Engineering, Inc. “Mobile Ad Hoc Networking (MANET) with AODV,” White Paper, 2003.
- [6] F. da Cruz and C. M. Gianone, “Using C-Kermit Communication Software,” Second Edition, Digital Press, 1997.
- [7] Netperf, [Available online] <http://www.netperf.org/>.
- [8] A. Chadda and R. Bartoš, “Quality of Service Testing Methodology,” Department of Computer Science Technical Report CS-TR 05-01, January 2005.