# Explorebots: A Mobile Network Experimentation Testbed

Teresa A. Dahlberg
University of North Carolina-Charlotte
9201 University City Blvd.
Charlotte, NC 28223-0001
704-687-8553
tdahlber@uncc.edu

Asis Nasipuri
University of North Carolina-Charlotte
9201 University City Blvd.
Charlotte, NC 28223-0001
704-687-3876
anasipur@uncc.edu

Craig Taylor
Southwest Research Institute
6220 Culebra Road
San Antonio, Texas 78228-0510
craig.taylor@swri.org

## ABSTRACT

In this paper, we detail our development of *Explorebots*—expandable, vision- and sensor-equipped wireless robots built around MICA motes. We developed Explorebots as a dynamic outreach for an NSF-funded Girl Scouts project. We've extended the capabilities of Explorebots to comprise a mobile network experimentation testbed. The testbed will support experimental analysis of protocols for mobile multi-hop networks. The low-cost Explorebots enable repeatable experiments without complete reliance on human subjects for mobility.

## Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design – *wireless communication*.

## General Terms

Design, Experimentation.

## Keywords

Mobile robotics, multi-hop networks, sensor networks, testbed.

## 1. INTRODUCTION

A growing body of research addresses network architectures with mobile nodes, mobile routers, and multiple wireless hops to access points. Examples are found for mesh, hybrid, and sensor networks. Although a majority of this research relies on computer simulations for performance evaluations, experimental validation is often crucial for evaluating the effect of real life parameters that are difficult to model in simulation or analytical platforms. Examples include understanding the impact of wireless path loss and interference, battery dissipation, and mobility on protocol performance.

A challenge for experimental validation of mobile networking protocols is to conduct repeatable experiments without complete reliance on human subjects to carry devices. Consequently, there is a need for development of autonomous mobile platforms equipped with hardware and software that may be easily programmed to perform a wide range of tasks to emulate desired mobility models as well as perform wireless communication, sensing, and networking functions. Although several experimental platforms have been proposed to address these needs [1 – 4], differences in cost, capabilities, and applicability motivate the development of new experimental platforms to suit targeted experiments.

In this paper we introduce the development of low cost, flexible autonomous mobile robots that comprise a testbed that can be used for a wide range of indoor experimentation on multi-hop ad hoc and sensor networking. Some of the key components of our ongoing research that motivates this work include localization and tracking algorithms for sensor networks and routing algorithms for hybrid (cellular/ad hoc) networks. Our prior wireless experimentation testbed requires manual movement of nodes to change network topology or to study the impact of mobility on protocol performance. To automate mobility, we incorporated use of *Explorebots*—expandable, vision and sensor equipped wireless robots. The major features of Explorebots are:

- Powerful on-board microprocessor for accurate robotic movement control.
- Interface with MICA wireless sensor nodes [5] to provide a wide range of sensing and communication functions.
- Built-in electronic compass and ranging devices that can be used for navigation.
- On board camera with wireless transmitter.

We initially developed the Explorebots for a dynamic outreach for an NSF-funded project for the Girls Scouts, Hornets Nest Council. We have since, extended the capabilities of the Explorebots to enhance our mobile network experimentation testbed.

In section 2, we detail the architecture, hardware, communication interface, and software tools used for the Explorebots. A demonstration of the application of Explorebots for the Girl Scouts outreach is described in section 3. Proposed research experiments that we will be conducting for wireless network experimentation are explained in section 4. Section 5 describes related work, and section 6 provides conclusions.

*SIGCOMM'05 Workshops*, August 22–26, 2005, Philadelphia, PA, USA.

## 2. EXPLOREBOTS DESIGN

### 2.1 Architecture

Figure 1 illustrates the Explorebots modular architecture and interfaces. The Mobile Platform manages robot movement. The Mobility Control and Robot Control & Communications modules enable remote control of the robot. Sensor Input/Output and Video Communications modules enable monitoring of feedback from the robots. Except for the Sensor I/O module, which interfaces with both, the modular architecture separates modules that manage robot monitoring and control (red blocks) from the protocols that are the subject of the experimental analysis (blue blocks).

Our objective was to identify a small-footprint robot that is easily maintainable, extensible, flexible and low cost, to support, initially an outreach demonstration, and ultimately an array of research experiments for mobile multi-hop architectures. During our initial search for off-the-shelf robots, we found nothing that met all of our criteria. Most had cost ranges from $3-6K, were not easily expandable, or, were from suppliers who went out of business. As a result, we developed the Explorebots, shown in Figure 2.

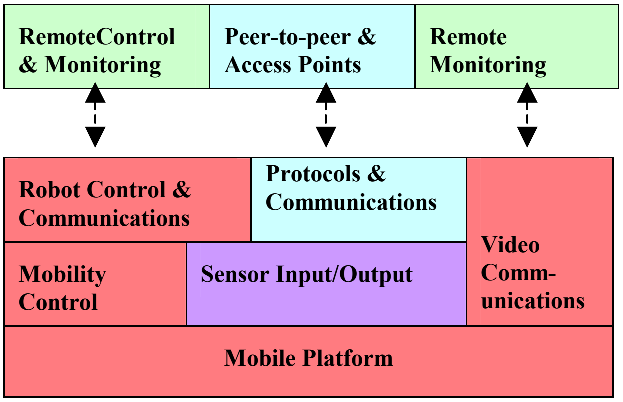

**Figure 1. Explorebots architecture**

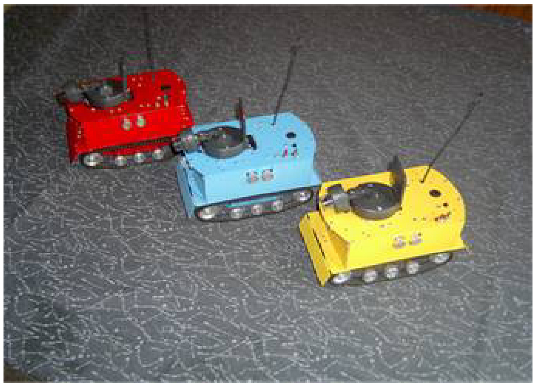

**Figure 2. Phase 1 Explorebots.**

The modular architecture supports multiple mobile platforms, flexible communications, programmable microprocessor, and multiple sensor support. Components are mostly off-the-shelf, from multiple suppliers. Each Explorebot can be assembled by an undergraduate electrical and computer engineering or computer science student in a couple of months for about $1,100.

### 2.2 Hardware Components

The primary robot components are mobile platform, microprocessor, communications, sensing elements, camera, and battery, as illustrated in Figure 3 and described next.

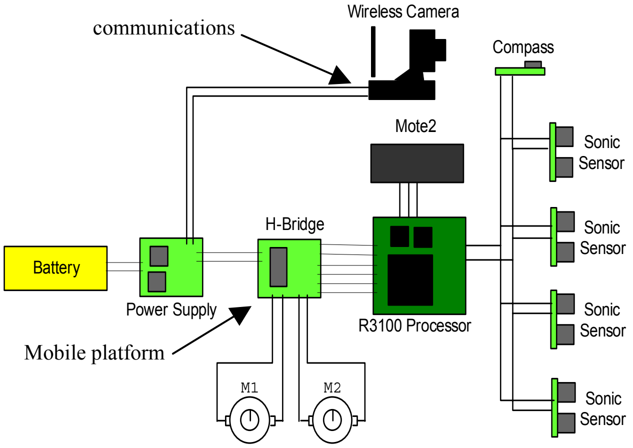

**Figure 3. Explorebots hardware architecture.**

Figure 4 shows the two level Rogue ATV base kit [6] selected for the mobile platform. The Rogue uses a track-based system that includes base, motors, gears, and H-Bridge driver. The dual H-Bridge driver allows for easy interfacing between the processor and the motors for individual control of each motor with speed and forward/reverse directions. The interface to the motor gear box is based on pulse width modulation (PWM), a common motor control method. This drive scheme allows for future changes to different bases with minimum system redesign.

**Figure 4. Rogue ATV robotic base kit.**

The Rabbit Semiconductor R3000 [7] was selected as the core microprocessor. It has an 8-bit architecture available as a complete module with all of the necessary supporting circuitry, as shown in Figure 5.

In module form, only power and I/O pin connections are required, memory and flash is included, and motor control is enabled via PWM interfaces. A Dynamic C/Assembler development environment is supported with TCP/IP library.

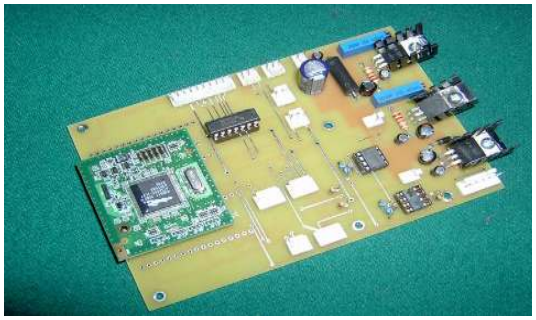

Figure 5. Rabbit 3000 microprocessor.

The power supply must provide three different voltages for the operation of the robot (Processor Power, 3.3 volts, Logic Power, 5.0 volts, and Motor Power, 4.0 volts) in a small, easily chargeable package. Because of the heavy inductance load of the motor, one adjustable power source provides the processor and motor power and one fixed power source provides the logic and device power. After much experimentation with NI-MH batteries, newer Lithium-Polymer batteries were selected. A battery of 11.1 volts, 1500 mA-hr was found that easily fit in the limited storage area of the robot. For safety, a battery (manufactured by Great Planes) was selected that has separate charging connection with safety circuits to prevent overcharging of individual cells. An issue with Lithium-Polymer batteries is to prevent discharging of the cells below 2.5 volts. For the battery applied in this design, this means that the voltage cannot drop below 7.5 volts or damage to the batteries will occur. To protect the batteries a simple circuit was designed based on a voltage-monitoring device, the MAX8212[1] [8].

Sensor interfacing using the industrial I2C communications bus provides the main connection with external sensor devices. This method is a two-wire serial communication protocol with one line representing data, and the second line used as clock control. Because of its wide use in embedded systems, there is a wide variety of sensing devices available. With this approach, new sensors configurations do not require reworking of the robot hardware.

Sonic based ranging sensors using the SRF08 modules [10] (see Figure 6), are located on all sides of the Explorebot, providing range sensing from 3 cm to 6 m.

A magnetic 2-axis compass, the CMPS03 [11], shown in Figure 7, provides heading or directional sensing.

In addition to the I2C sensors, groups of sensors require direct wiring to the processor I/O pins. Two examples are tactile and velocity-distance sensing, described below.

Two tactile sensors (see Figure 8) are on the forward face and one tactile sensor is on the aft face of the robots. The sensors reside under hinged bumper flaps to detect and report contact with obstacles.

Figure 6. Sonic-based ranging sensors

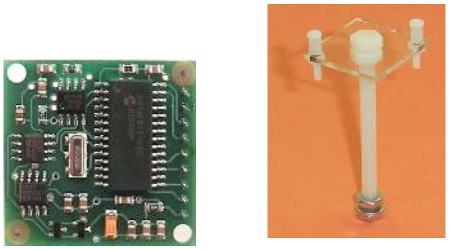

Fig. 7: Heading or Direction Sensing

We custom designed the velocity and distance sensor shown in Figure 8. It is a rotation counter with an optical sensor that is attached to the mobile platform gear train. One of the gears for each motor has four holes that cause the optical sensor to pulse each time a hole passes it. The robot software maintains a counter that sums the pulses and relates them to time. With this information, it is possible to calculate the speed and distance traveled by each track.

Figure 8. Tactile sensor and custom velocity-distance sensor

A separate wireless camera system (the XCam2 320x240 color [9], as shown in Figure 9) was mounted on each robot to support 15 fps streaming video to a remote computer. The XCam2 battery pack was powered from the robot by removing supporting circuitry and installing it in the robot.

Figure 9. X10 Cam2 wireless camera.

---

[1] Finding a battery and charging solution was one of the most challenging issues. Email any of the authors for additional details.

The installations of the tactile, range, and compass sensors are shown in Figure 10.

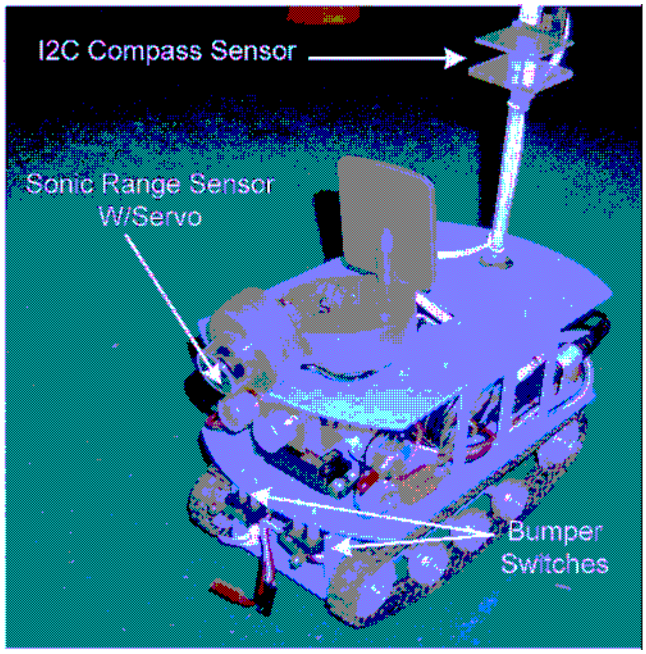

**Figure 10. Explorebot with installed range, tactile (in bumpers) and compass sensors.**

## 2.3 Communications

Wireless communications to the Explorebot is via processor serial port to a separate module that includes Crossbow’s Mica2 Motes, shown in Figure 11, and Berkeley’s RTS operating system, Tiny OS. Although the range of the Mica2 mote is limited, it supports a number of experimental scenarios, and the communication module can be easily replaced (for example, in our follow-on work, motes are replaced by 802.11).

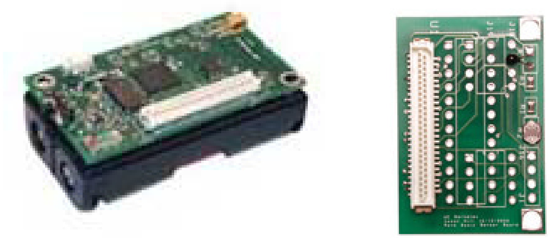

**Figure 11. Mica2 mote and MTS101 basic sensor board**

## 2.4 Supporting Software

The rabbit R3000 processor supports real time operation through a round-robin scheduling or through inclusion of MicroII OS. We used the scheduling scheme with internal and external hardware interrupts. Figure 12 diagrams the architecture.

Serial communications between the processor and the Mica2 Mote is via use of a standard RS232 library provided with the rabbit processor. Configurations required to implement the serial interface are contained within the Mote-R3100 library.

The core of the Mote-3100 library is the communications protocol used by the robot to define commands and information packets. This implementation is a packet protocol that is embedded within the payload of Tiny OS message packets.

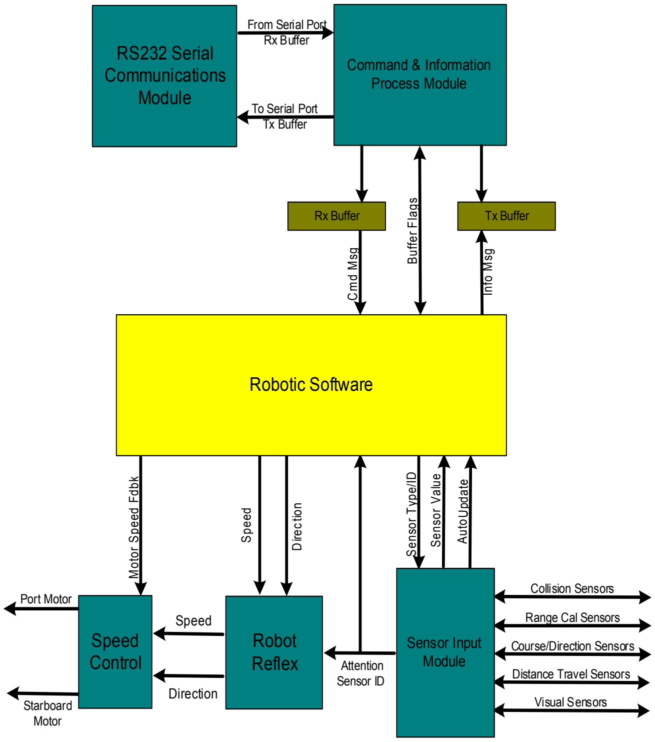

**Figure 12. Explorebot software architecture**

The sensor input module uses the I2C serial bus to interface to a number of sensors, each having a unique address. Communications with a sensor is a simple matter of sending an attention message to the sensor’s address, and then requesting a register value or sending a register value. For sensors having their own send/receive sequence, additions to the sensor library are required.

The current standard sensor library supports two sensor types, the SRF08 sonic range sensor and the CMPS03 compass module. The miscellaneous sensor library contains the code required either for sensors that connect to the processor directly or through means other than the I2C bus. In the initial release of the library, three sensors are included: Bumper Switch Input, Low Battery Input, and Gear Rotation Counter. As with other libraries, this library includes the initialization algorithm and any required support functions.

The motor control functions were also incorporated into a library. The original library contained only the simple speed control as outlined above, however efforts are underway to expand the library to include other control algorithms such as PID control. The purpose of all of the libraries that we developed is to include a wide range of functionality to support a number of different test scenarios.

## 3. GIRLS ARE IT!

We initially developed the three Explorebots shown in Figure 1 for the Girls are I.T.! outreach project [12]. The Explorebots were interconnected as shown in Figure 13. Each robot has a bumper in the front and back, four sonic range sensors (one on each side of the robot), and a wireless camera. Each robot communicates with a server which routes information to three individual workstation PCs.

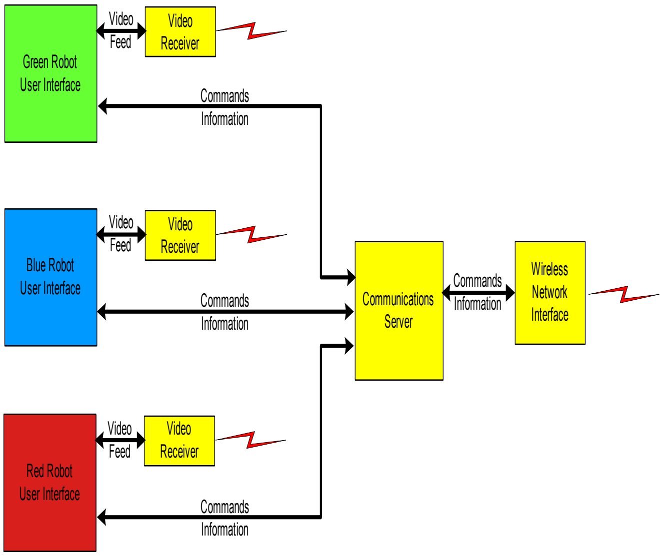

**Figure 13. A demonstration testbed using Explorebots**

At each workstation, the operator (a girl scout) explores a remote terrain via remote control of a robot with the video feed from the robot's camera as the only visual aid. Commands from the operator workstation to the robot instruct the robot to move in a forward or reverse direction or to turn to the right or left.

Figure 14 shows the operator's GUI, which interfaces with a java application that maintains communication with a mica mote residing on the robot. A picture of a partially built robot is in the video view of the figure. The direction control buttons are in the center, and a speed control slider is center bottom. Feedback from robot sensors is shown to the left (temperature and light intensity) and to the right. On the right, the distance from the robots bumpers to an obstacle is shown. A robot's compass reading (north, east, south, west) is shown with an arrow. And, green, yellow, and red collision indicators are meant to warn the operator of impending or occurring collisions.

Each robot resides within one level of a three-level terrain inside of a mobile computer lab. The terrain is hidden behind a curtain at the back of the bus, while the operator workstations are placed near the front of the bus. Each terrain level is decorated according to a different theme (underwater exploration, a disaster scene among a network of highways, and a wildlife reserve). The terrain is flat, and the obstacles are either plastic toys or pictures secured within plastic frames. All obstacles are secured to the floor of the terrain. The Explorebots travel very well on the flat surface and will roll over objects that are less than about two inches high. For this demonstration activity, we wanted the robots to move around obstacles, not over them. Therefore, obstacles chosen had to be substantial enough in size, and secured well to the floor, to prevent the robots from rolling over the obstacles. (In our follow-on work, we've since experimented with different base kits. The choice of base kits is very dependent on the intended environment: what types of obstacles exist, if the intent is to move over or around or back off from obstacles.)

**Figure 14. Robot workstation user interface**

The choice of obstacle size was also influenced by limitations on the camera view. We did not use remote control or automatic vertical tilting of the camera. This limited what could be captured by the cameras.

Because our emphasis was remote control of the robots, the challenges with this activity were proper positioning of the cameras, providing feedback (via bumpers and other sensors) to the operator in a timely manner, and delivering operator control commands to the robots in a timely manner. In extending the Explorebots to form a networking research testbed, our focus changed to autonomous navigation of the robots, as we mention next.

## 4. RESEARCH TESTBED

Several expanded versions of Explorebots have been (and are still) being built to incorporate mobile nodes into our existing experimental research testbed. Additional sensors are added to support autonomous navigation experiments. Additionally, the side range sensors have been replaced with a forward sensor that is positioned with a servo under control of the robot. The forward bumper has also been split to identify partial collisions that may require a turn away from an object. Hardware modifications have been minimized because of the flexible nature of the Rabbit RCM module and the use of I2C serial sensors.

Currently, Explorebots can be instructed to move to specific location coordinates within a 4'×6' rectangular test area that is bounded by wooden railing. The desired location coordinates are entered using the GUI.

Our current research using the Explorebots is focused on the following issues:

***Incorporation of localization:*** We implemented a low cost directionality based localization scheme that uses optical beacon signals to allow mica2 wireless sensors to determine their locations [13]. This feature will be programmed into the Explorebots to utilize location information for moving to specified coordinates.

***Target localization:*** We are exploring algorithms for implementing autonomous robotic movements that are driven by processing sensor signals such as sound and light. In this application, the Explorebot will communicate with an array of mica2 motes located in the test area to move towards a specific acoustic (such as a tone generator) or photo source.

***Hybrid routing protocols:*** One of our objectives is to evaluate routing protocols for hybrid access networks. The idea is to use a set of wireless routers on mobile platforms that provide optimum multi-hop routes from access points to a pool of wireless users. Optimum coverage and QoS of the wireless network will depend on appropriate positioning of the wireless routers. Our goal is to design a prototype multi-hop network using mica2 motes on Explorebots to emulate wireless routers. Although such a wireless network is very different from a typical wireless LAN such as one using 802.11, this platform would adequately capture the indoor wireless propagation effects for evaluating our route selection, positioning and power control algorithms.

## 5. RELATED WORK

Although robotics is not a new field, building experimental testbeds with mobile wireless-controlled robots is emerging as a key area for many researchers in wireless networks. Intel based mobile robots are used by researchers at the Center for Embedded Networked Sensing (CENS) to perform environmental monitoring tasks [3,14]. They address design issues such as coverage, navigation, network deployment and maintenance by mobile robots. A mobile wireless robotic testbed has been developed by the University of Utah that allows users to create their own experiments over the web [1]. Columbia University's AVENUE robot was designed to automatically construct wireless signal maps in outdoor environments [2]. Intel Research leads robotic research groups in about 20 universities, who developed pilot projects that include robotic dogs, insects and helicopters [4]. The common features of all these mobile platforms include the presence of onboard processor to manage internal (within the robot) systems such as motion control, sensors, actuators, etc., and external control from a computer using a wireless interface.

The tremendous scope of applications and the extent of interest for experimentation with such platforms led to the development of the Robotics Engineering Task Force (RETF) [15], whose charter is to develop standards for protocols that could be used as interoperable building blocks for mobile robots. While the focus of the RETF is on software development only, one of their goals is to develop specifications for key hardware features to be used in such reusable and interoperable mobile robot platforms, such as sensors, actuators, and communication interfaces.

## 6. CONCLUSION

We present the architecture and hardware details of an autonomous robot used for mobile testbed experimentation. These robots (Explorebots) have a range of useful hardware and software tools that provide a low cost solution to obtaining experimental validation of ad hoc and sensor networking protocols in an indoor laboratory scale testbed. Relevant details required for implementing Explorebots and our current research involving them are described. Additional details are found at http://www.cs.uncc.edu/~tdahlber.

## 7. ACKNOWLEDGMENTS

This work was partially supported by NSF grants EIA 0130799 "Experimental Testbed for Mobile Network Protocols" and EIA 0204398 "Girls are IT!". The authors wish to thank our students Jeremy Yager and Chezerae Ramos for the many hours they spent on this project. Finally, the authors thank the reviewers for their thoughtful comments.

## 8. REFERENCES

[1] B. White, J. Lepreau, S. Guruprasad, "Lowering the Barrier to Wireless and Mobile Experimentation", HotNets-I, Oct. 2002.

[2] P. Blaer and P. K. Allen, "TopBot:Automated Network Topology Detection with a Mobile Robot", ICRA 2003.

[3] M. A. Batalin and G. S. Sukhatme, "Coverage, Exploration and Deployment by a Mobile Robot and Communication Network", Telecomm Syst. Vol. 26, No. 2, pp. 181 – 196, 2004.

[4] J. Butler, "Robotics and Microelectronics: Mobile Robots as Gateways into Wireless Sensor Networks", Technology@Intel Magazine, May 2003.

[5] http://www.xbow.com

[6] http://www.hobbyengineering.com/SectionRK.html#CatRKROGUE

[7] http://www.rabbitsemiconductor.com/documentation/docs/manuals/RCM3100/UsersManual/aspecs.htm

[8] http://korea.maxim-ic.com/parts.cfm/p/MAX8212

[9] http://www.x10.com/products/x10_vk45a.htm

[10] http://info.hobbyengineering.com/specs/devantech-srf04-tech.pdf

[11] http://info.hobbyengineering.com/specs/devantech-cmps03-tech.pdf

[12] http://www.girlsareit.org

[13] A. Nasipuri and K. Li, "A Directionality based Location Discovery Scheme for Wireless Sensor Networks", in Proceedings of the *First ACM Intl. Workshop on Wireless Sensor Networks and Applications (WSNA'02),* 2002.

[14] http://robotics.usc.edu/

[15] http://www.robo-etf.org/