Making Sensor Networks Accessible to Undergraduates Through Activity-Based Laboratory Materials

Jens Mache
Lewis & Clark College
jmache@lclark.edu

Damon Tyman, Nirupama Bulusu
Portland State University
damont@pdx.edu, nbulusu@cs.pdx.edu

Abstract

In 2003, the MIT Technology Review magazine listed wireless sensor networks as one of "Ten Emerging Technologies That Will Change the World" [2]. Five years later, are wireless sensor networks ready for the undergraduate classroom?

Our goal is to develop lab exercises that are suitable for activity-driven teaching of undergraduate students. We here describe our lab exercises and our experience class-testing them in the Winter quarter of 2008 at Portland State University.

1. Introduction

Whereas sensor networks are poised to become a ubiquitous part of the computing landscape, sensor network education has been primarily offered at the advanced graduate level. This is especially so because sensor networking draws from many disparate fields of study. The goal of our work is to develop lab exercises that are suitable for activity-driven teaching of sensor networks to undergraduate students.

With input from industry, students, and an educational consultant, we are developing exemplary lab exercises, identifying topics that are appropriate, clarifying prerequisite knowledge and preparatory material, and presenting the material in a format that is suitable for undergraduates. The significance of this work lies in clarifying the prerequisite knowledge to deploying and programming sensor networks, and building a foundation for teaching these topics to undergraduates.

A first iteration of a lab-based sensor network course was class-tested at Portland State University in the Winter quarter of 2008. We used the SunSPOT devices [1] which have been made publicly available by Sun Microsystems since April 2007. SunSPOTs are programmed in Java, relieving the necessity of students having to learn TinyOS and NesC upfront, as would be the case with the popular Berkeley mote devices. Assuming a basic familiarity with Java, the learning curve for working with the SPOTs is very reasonable, and thus allows for a concentration on sensor network concepts such as localization, wireless communication, and power management (instead of language and operating system details).

2. Our Lab Exercises

2.1 Introductory

The purpose of Lab1 is to introduce students to the SunSPOTs and familiarize them with the basics of working with them. We use the demo that comes by default with the SPOTs, "the ectoplasmic bouncing ball". This is effective in that the demo is fun to experiment with, encourage students to peel the cover and use the switches on the SPOTs, and demonstrates radio communication between the SPOTs as well as the use of a SunSPOT host application that runs on the desktop and uses the basestation.

Lab2 is a follow-up to and continuation of Lab1. The primary purpose is to get students up to speed with deploying new applications onto the SunSPOTs. They are taught how to do this, and familiarized with SPOT functionality introduced into NetBeans by the SPOT-specific modules. The Telemetry demo is used as the application which students first deploy and then test. This ensures that they have properly set up and used a shared basestation and run a host application that interacts with remote SPOTs (if they did not try the bouncing ball host application at the end of Lab1). As an exercise, students are asked to modify code for the first time by changing their host application to only communicate with the SPOT that they specify.

2.2 Communication

The intent of Lab3 is to teach radio communication between remote SPOTs. Addressing lossy wireless communication is one of the significant challenges of wireless sensor network deployments. Of the two provided protocols, radiostream and radiogram, radiogram is chosen. Once one is familiar with one, however, learning to use the other is trivial (especially once one knows which resources to turn to). An application is presented and gone through tutorial-style. The application starts from the code snippet given in the SunSPOT Developer's Guide and builds out of it a program where each SPOT increments a number at a set interval and sends the result out over the radio while simultaneously displaying numbers received over the radio. This way, the number each SPOT displays is not its own, but the number sent by another node. Students deploy and test the application after understanding the code. As a homework following this lab, students are tasked to use what they learned to test packet loss between remote SPOTs and better understand packet loss, link asymmetry, irregular coverage and other vagaries of radio communication.

After learning how to handle radio communication in Lab3, students are in Lab4 introduced to interacting with the Internet through HTTP on a SunSPOT. This lab, which also follows a tutorial-style approach, again starts from a code snippet from the Developer's Guide and builds out of it an application where a SPOT gets and parses a web page off the Internet. For parsing, whenever certain tags are encountered, the SPOT flashes its LED's a designated color (depending on the tag). For this to work, a base-station running the socket-proxy application must be connected to an Internet-connected computer. As a follow-up to this lab, we have recently developed a new lab that goes further and demonstrates using HTTP POST requests (instead of just GET requests as is seen in Lab4) to publish sensor data on Internet portals, such as SensorMap.

2.3 Localization

Lab5 is meant to teach localization. Spatial context is important to sensing applications, and thus node localization is a fundamental building block for sensor networks. At this point, students have become quite comfortable programming SPOT applications on their own and we deemed tutorial-style labs not necessary any more. Instead, starting with Lab5, the focus is more on completing a challenge on one's own rather than following a tutorial to make something work. The challenge in Lab5 is to program a remote SPOT to localize itself. Beacons that broadcast fixed coordinates are set up around the room, and students are asked to implement a well-known algorithm that calculates a position from the beacon coordinates received. The algorithm given was a simple weighted centroid approach where a SPOT's coordinate in any dimension is the quotient of the weighted sum of beacon coordinates (in that dimension) divided by the sum of the weights. For weights, students used the RSSI value for signal strength available through the SPOT API.

2.4 Power Management

The theme of Lab6 is power management. In unattended wireless sensor networks, reducing power consumption and conserving energy are the dominant system design goals. For this lab, students are asked to investigate power consumption characteristics of SunSPOTs through the tools provided by the API. Students are instructed to take measurements of the current being drawn at a certain point in time and the maximum current drawn between two events for a number of different scenarios. Both measurements are available through API calls; the latter approach was primarily used as it is usually more consistent and meaningful. Students investigate three questions. For the first question, students investigate which of the SunSPOT tasks, radio transmit, receive, or computation consumes the most power. Another goal of the lab was for students to understand the distinction between sleep and deep sleep modes of the SunSPOTs. The last question is aimed at making sure students understand the conditions necessary for deep sleep and are able to make sure they are met as well as test that they truly understand what happens when the SPOT goes into deep sleep.

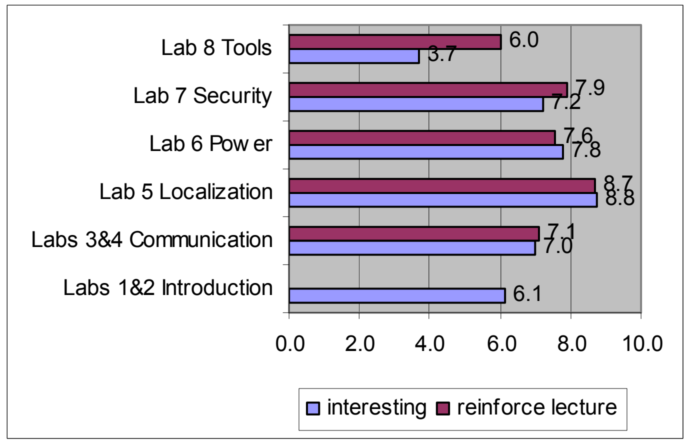

Figure 1: Feedback from nine students whether the first 8 labs were interesting and reinforced lecture topics, on a scale from 0 (very low) to 10 (very high).

2.5 Security

Lab7 addresses the topic of security. Security in sensor networks is challenging because of the resource-constraints of the devices and the inherent vulnerabilities of the wireless medium. Many sensor network security schemes use symmetric key cryptography to accommodate the resource-constraints, and therefore many key pre-distribution protocols for sensor networks have been proposed. This lab followed a security lecture in class where a specific key distribution scheme, the Eschenauer-Gligor Scheme, was introduced. This lab challenges students to explore the characteristics of that scheme. A key manager was provided by which remote SPOTs could request a set of keys and have the specified number of keys sent back by the key manager. The task of the student in this lab is to program two SPOTs to each request Y keys, where Y is a variable number (repeated 5 times) for many different values of Y and record the percentage of times that at least one key appears on both nodes. They are to do this until they arrive at a reasonable conclusion about a good number of keys to use for an approximately 50% chance of a key match between any two nodes.

2.6 Other Platforms and Tools

The final lab, Lab8, is meant to introduce the TinyOS environment and Micaz motes from Crossbow to the students. Students are first directed to the official TinyOS tutorials for a brief introduction to NesC. This is so that they can see the basic structure of the language, but not with the intent of them gaining much ability programming in it. They are subsequently directed to deploy a few example applications onto the Micaz's to learn the process and see what capabilities these motes have. Finally, they are tasked with running an application on the TOSSIM simulator and exploring the various features it provides.

2.7 Contour Tracking

Many sensor network applications often perform a few common canonical tasks – target detection and tracking, contour tracking, target classification and so forth. As a final assignment, students built a contour tracking application for a grid of SunSPOT nodes. The goal for the sensor network is to determine where the contour of a single light source occurs. For example, when a flashlight that produces a circular area of light is placed in the middle of the nodes, the sensor network should be able to determine between which nodes the contour of that circle falls. It was thus necessary for students to establish a ground truth for the contour to ensure that their network was responding properly. Many students used the LEDs on the SPOTs, as in many of the other labs, which allowed for quick visual feedback from the network, but was not a requirement.

3. Class-testing and Student Feedback

A first iteration of the lab-based sensor network course was taught at Portland State University in the winter quarter of 2008. Nine students provided feedback (before the final contour tracking assignment was due), see Fig. 1 and Table 1.

Table 1: How the first 8 lab exercise were ranked by nine students (rank 1 is best).

	s1	s2	s3	s4	s5	s6	s7	s8	s9	avg
Labs 1&2 Introduction	5	5	4	6	5	5	5	4	4	4.8
Labs 3&4 Communication	4	3	5	4	4	4	4	2	1	3.4
Lab 5 Localization	1	2	1	1	1	3	1	2	2	1.6
Lab 6 Power	3	4	2	2	2	2	2	1	3	2.3
Lab 7 Security	2	1	3	3	3	1	3	4	5	2.8
Lab 8 Tools	6	6	6	5	6	6	6	6	6	5.9

In addition to the quantitative feedback, the students also gave us verbal feedback. Unanimously, the students agreed that the labs were very useful and probably the best part of the course. In the words of one student, "The course is very lab oriented and it helps in understanding concepts. The lab instructions are very clearly defined. We both learned about the concepts pertaining to the subject matter, and also got to practice working with the technology hands-on."

On the flip side, students requested more orientation about techniques for debugging the Sun SPOTs and possibly further simplifying the Demo code already provided by Sun Microsystems.

4. Discussion

What went well

- Students responded well to the SunSPOTs; they were engaged and motivated to experiment on their own.
- The Java-programmable SunSPOTs allow for concentration on concepts over language-details.
- Our labs covered several core sensor network topics such as communication, localization, power management and security.

What can improve

- Create new labs on sensor calibration, and wireless multi-hop routing.
- Create primer on debugging tools and techniques for the SPOTs.
- Develop a macro-programming lab, possibly a TinyDB clone for the SPOTs.
- Expand listing of background material resources for each lab.
- Simplify instructions for Labs 1-3.
- Make lab 4 more challenging; transition to requiring more coding of the student as in later labs.
- Find a way to introduce TinyOS that is engaging and neither too overwhelming nor too basic.

.

Acknowledgements

This work is supported by NSF grants CNS-0720914 & CNS-0720545 and by an equipment donation by Sun Microsystems.

We would like to thank Jeanette Palmiter, as well as Suman Nath, Mark Yarvis and Chieh-Yih Wan for helpful discussions.

References

[1] SunSPOT device, http://sunspotworld.com

[2] MIT Technology Review, "10 Emerging Technologies That Will Change the World", http://www.technologyreview.com/Infotech/13060/page2/, Feb. 2003.