SignetLab: Deployable Sensor Network Testbed and Management Tool

Riccardo Crepaldi, Albert Harris, Alberto Scarpa, Andrea Zanella, and Michele Zorzi
Department of Information Engineering – University of Padova

[riccardo.crepaldi,harris,alscarpa,andrea.zanella,zorzi]@dei.unipd.it

1. INTRODUCTION

The drive to move sensor network research from theory to deployment has dramatically increased interest in sensor network testbed design. Such testbeds have the goal of supporting rapid deployment of protocols on real hardware without requiring the use of specialized sensor networks each time. The key challenges to meeting this goal include providing realistic environments for testing the protocols (i.e., indoor testbeds may not approximate outdoor deployments accurately) and having a simple programming and experimentation interface. If the testbeds do not provide realistic environments for the testing of protocols, they may fail to bridge the gap between simulation and deployment and if they are not easy to use and maintain, they could end up stifling research rather than augmenting it.

The SignetLab demonstration shows two fundamental contributions. First, we present our deployable testbed. A critical feature of the SignetLab testbed is the ability to rapidly deploy it in a number of scenarios (e.g., inside a lab or outdoors). This allows protocols to be tested in environments that are very similar to actual deployment scenarios. To facilitate this deployability, we have implemented a novel in-network programming module to accurately and efficiently program nodes while they are deployed. A key difference between our in-network programming module and previous work [7, 9] is the built-in reliability mechanism that accurately programs all nodes in the network, via multihop wireless links. Second, we present our testbed management and visualization tool. This tool provides the programming interface to the network as well as the data visualization module. Through the tool, nodes can be managed at a fine-grained level, giving researchers the necessary control to perform a large variety of experiments.

2. RELATED WORK

There have been a number of previous testbed designs; however, most of this work has focused either on mobile ad hoc networks [1, 8] or on specific protocol testing (e.g., power control [2, 10] or mobility [5]). Two testbeds, however, have particular relevance to our work: MoteLab [11] and Mobile Emulab [3, 6]. Both of these solutions aim to maximize testbed utilization among different users. To this end, each provides a web interface through which users can schedule time to use the sensor network. Fine-grained control over the network during a user’s timeslot is not provided by these solutions, although it is a desired capability [11]. Our work is orthogonal to both of these solutions in the sense that either scheduling solution could be combined with our testbed tool.

In addition to work on testbeds themselves, there have been attempts to provide an in-network programming solution for sensor networks (e.g., Deluge [7]). None of these solutions reliably programs all nodes in a given network and they reliably support only micaMotes [12].

3. SIGNETLAB HARDWARE

SignetLab was built using the EyesIFXv2 nodes, which were developed during a European research project on self-organizing energy-efficient sensor networks [4]. Each node is equipped with an ultra-low power MSP 430 processor with 10KB on chip RAM and 48 KB flash/ROM, with an additional 512 KB serial EPROM. The radio chip is a low power FSK/ASK transceiver. It provides half-duplex, low data rate communication in the 868 MHz band, with a sensitivity of $<-109$ dBm, enabling up to 64 Kbps wireless connectivity.

The EyesIFXv2 is equipped with an on-board antenna and an SMA-connector for an external antenna. The transceiver accepts a supply voltage of 5.5 V, with a current of 9 mA in receive mode and 12 mA in transmit mode. The transmit power can be modulated by means of a digital potentiometer with 255 settings (although only 180 to 255 produce useful transmit power variations). The nodes come with onboard temperature and light sensors as well as an SPI expansion port for including additional sensing capabilities. The nodes can be powered either by batteries with a capacity of 1,000 mAh or through a power supply connected via an external polarized connector or a USB connection.

4. SIGNETLAB SOFTWARE TOOL

The SignetLab software tool was designed to provide a single programming interface to all users facilitating rapid testing and deployment of sensor network protocols. Additionally, it is supported on multiple platforms, is highly customizable, and is easily extensible. Programming nodes (either all or some subset), including compiling and uploading code, is simple and automated, while giving the user as much control of the testbed as possible.

To accomplish all of this, we designed a management and visualization tool as a Java application. The main application window gives the user access to a number of tools, including the compilation and execution pane, a serial output viewer, a direct input pane to send commands directly to nodes, and a visualization pane. All

Categories and Subject Descriptors: C.2.1 [Network Architecture and Design]: Wireless communication.

General Terms: Performance, Design.

Keywords: Wireless sensor network, testbed, software management tool.

Copyright is held by the author/owner(s).

SenSys 06, November 1–3, 2006, Boulder, Colorado, USA.
ACM 1-59593-343-3/06/0011.
these panes are fully customizable and we have provided an API through which new panes can be built by users. The visualization pane allows the display of arbitrary node topologies through a simple configuration file. Also, an API is provided for users to display data obtained from the sensor network in a variety of ways (e.g., simple bars denoting state as in Figure 1).

The tool also provides an in-network programming module. This module is critical for a deployable testbed to eliminate the need for long cables to connect the nodes. For example, attempting to test a number of protocols for comparison in a particular deployment without an in-network programming module would require either cables connecting all nodes in the deployment, which might be prohibited by the terrain, or the nodes to be gathered up and redeployed for each protocol. Redeploying the network in exactly the same way is, however, not only very time consuming, but also very difficult to accomplish. Our in-network programming module uses a combination of FEC and ARQ to ensure accurate programming of the network, with any un-recoverable errors (e.g., due to hardware failure), being reported back to the researcher.

5. DESCRIPTION OF DEMONSTRATION

The goal of this demonstration is to show the effectiveness of the SignetLab sensor network testbed as a rapid protocol testing tool. Therefore, we will deploy the testbed at the conference and use the management and visualization tool to compile, install, and run a sensor network application. While the application is running, the visualization component of the tool will be used to monitor the functionality of the application in real time. Furthermore, data collection will be performed by the management tool, allowing post-run analysis. We intend to reprogram the nodes after each run to demonstrate the speed at which the network can be reconfigured using the tool.

The demonstration application consists of an environmental monitoring application used to detect fires. The application uses a simple routing algorithm to propagate periodic temperature readings at a coarse granularity from each node. In the event of a large increase in temperature, the nodes sensing the increase begin sending updates more frequently. The monitoring station (i.e., the sink node) runs the visualization tool and displays, using a plugin written for the application (see Figure 1), each node and a number of bars showing the perceived state of the environment. In the event a fire is detected, a fire icon is displayed.

6. REFERENCES