The Design, Deployment, and Analysis of SignetLab: A Sensor Network Testbed and Interactive Management Tool

Riccardo Crepaldi, Simone Friso, Albert F. Harris III, Andrea Zanella, and Michele Zorzi
Department of Information Engineering – University of Padova
[riccardo.crepaldi,simone.friso,harris,andrea.zanella,zorzi]@dei.unipd.it

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.

1. INTRODUCTION

The ability to manufacture small, inexpensive computing devices with wireless networking capabilities has led to a large amount of research in the area of sensor network protocol design. The vision of large, self-forming networks of small devices, each equipped with sensing hardware to monitor an environment (e.g., a battlefield or disaster zone), requires the design of communication protocols that are highly scalable (to thousands of nodes), loss-tolerant (the devices can be unreliable and prone to failure), and energy-efficient (sensor devices live on minimal battery power that is not replaceable). A key difficulty in designing protocols for this type of system is the lack of appropriate methods of testing. There are a number of simulation options available (e.g., the ns2 network simulator), but each of these necessarily hides real-world effects (fading properties, anisotropic propagation, etc.).

To address this limitation, a number of testbed solutions have been proposed very recently. The primary focus of these early testbeds has been either on sensor node design [1, 3, 5] or on tools to allow users to timeshare the network [6, 4]. One feature that some of these works cite as a future goal is a tool that allows users to have fine-grained control of experiments during their timeslot as well as real-time feedback from the network. The first contribution of our work is the development of such a tool, which provides a simple interface through which to program, interact with, and receive data from the sensor nodes.

Our second contribution is the analysis of SignetLab, our sensor network testbed. SignetLab is composed of 48 EyesIFXv2 nodes [2], a USB data backplane, and is supported by a software tool that allows node selection, visualization, and network programming. It is deployed in the SigNet research lab (a 10 m by 10 m space) in the University of Padova, DEI building. SignetLab is actively used by researchers at the University of Padova as well as the University of Rome. The analysis of the testbed provides insight into the impact of physical deployment on its functionality.

2. SIGNETLAB SOFTWARE TOOL

The SignetLab software tool was designed to support a number of goals. It should provide a single programming interface to all users that is intuitive to use (i.e., small learning curve). It should be supported on multiple operating systems to allow users to easily integrate it into their own work environment. The tool should also support multiple physical sensor network testbeds (i.e., different node technologies, different node layouts, etc.). Programming nodes (either all or some subset), including compiling and uploading code, should be simple and automated, giving the user as much control as possible during their use of the testbed. Finally, it should be easy for users to add functionality to the tool.

The tool is a Java application and a set of configuration files that set up the environment. When the application starts up, the main window is split into two sections, the GUI node selection pane on the left and the plugin pane on the right.

The GUI node selection pane reproduces the topology of the network as specified in the topology configuration file. The user is able to select the entire set of nodes or any subset of nodes either by clicking on the nodes, dragging a bounding box around them, or by using the selection menu. Once nodes are selected, various plugins can be used to program the nodes and begin code execution. The plugin pane contains various plugins and their interfaces. Users can easily expand the capabilities of the application by using a simple API to write their own plugins.

3. ANALYSIS OF SIGNETLAB

Analysis of the testbed in terms of the environment it provides for protocol experimentation yields insight into the best practices for testbed design and deployment. The fundamental tunable parameter that alters the sensor network environment is the potentiometer setting that adjusts the transmit power of the nodes. This setting determines the distance each node can reach, subject to additional propagation and environmental phenomena (e.g., multipath fading). Analyzing the effects of different potentiometer settings shows the range of environments that the testbed can provide.

We define two metrics to analyze the testbed. Consider the signal propagation from a single sensor node for a given transmit power level. Theoretically, in the absence of any interference or reflections, the area where the signal is received at greater than some average signal strength, , would define a circle. However, in realistic physical environments, there are a number of factors that alter this behavior. In an outdoor environment with no trees, buildings, or other close obstructions, this area may still be well approximated by a circle. For an indoor environment, however, the contour for a given average signal strength would have a very different shape. Our two metrics are defined by inscribing and circumscribing circles to the contour corresponding to each value of the average sig-
nal strength in the graph. We define the greatest continuous distance reached as the radius of the inscribed circle, which is the distance inside which the average received signal strength is guaranteed to be greater than $x$. We define the farthest distance reached as the radius of the circumscribed circle, which is the distance outside which the average received signal strength is guaranteed to be less than $x$. Instead of using received signal strength as our metric to slice the graph, we use percentage of packets received, which is essentially the same, as a received signal strength can always be translated to a probability of packet error.

Using these metrics, we analyzed the characteristics of Signet-Lab in terms of number of hops required to traverse the network (demonstrating its ability to support a variety of sensor network scenarios, including significant multihop behavior) and in terms of the propagation characteristics of different nodes in the network. This analysis gives insight into the effects of node placement on the properties of the network.

Figure 1 presents the greatest continuous distance reached with the potentiometer set at the highest level (255). This graph represents the case where a node is considered reachable if 80% of the transmissions arrive reliably. We also did mappings with other choices of the percentage of packets received; however, their general shape is the same. Nodes in the center reach shorter distances only because they reach the edges of the network, which are about five meters away. The corners of the network can reach about seven meters, meaning that even with the highest potentiometer setting, nodes at the corners cannot reach the entire network with a single hop.

Figure 2 maps these distances for each node in the network. From here it can be seen that at the highest transmit power setting, the edges of the network can reach each other with high probability (though the corners cannot). However, at the lower potentiometer setting (230), the distance is reduced by about 50%.

Again, it should be recognized that the potentiometer can be decreased to 180 while still maintaining some transmission radius. We have maps of the network at all of these settings; however, due to space limitations, they have been omitted here but will be made available on our website (http://www.dei.unipd.it/ricerca/signet); however, the results presented here demonstrate that the testbed achieves the flexibility desired in terms of coverage variability with different potentiometer settings.

4. REFERENCES


