CS 3468
Lab 5

Jared Wallace

Objectives

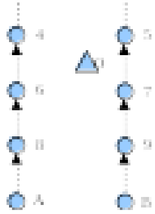
1. Build a sensor application that detects and locates mobile object
2. Understand the pro's and con's of teamwork

Overview

In this project, y'all will build a sensor network to track a mobile object that carries a light. In the network, sensors will detect the object by sensing the light. Because each sensor has a different distance to the light, the light intensity recorded by each sensor will vary according to the distance from the source. By collecting and analyzing all sensing data, the network will be able to detect and locate the mobile object.

The network will look like the diagram below, where the triangle symbol represents the source light, the circle symbols represent sensors and the square symbol represents the base station. The base station will be what collects the data from the sensors and computes the location of the source light.

Because the network area is large, sensors farther away from the base station will be unable to communicate directly with the base station. Instead, their data must traverse other sensors in order to reach the base station. The dashed lines represent how that data is forwarded through other sensors to reach the base station.



Team

This will be a whole class effort. Each student will have the responsibility of programming their own sensor. You will be provided one local base station app for debugging purposes. When we build the final network, one bad sensor node will break the network (there's the con of teamwork).

I will give you the following information:

- The ID of the sensor
- The ID of the local base station (your personal debugging base station)
- The ID of the next hop
- The ID of the previous hop

Tasks

Implementation of the network takes two steps. First, each student will make a local application to ensure the sensor can correctly detect light and transmit data. Then, all sensors will be connected to form a tracking network with a static route.

Part 1

Create an application that can:

1. Sense light every two seconds
2. Calculate and transmit the average of every three data points and transmit that data to the base station (that's once every six seconds for the mathematically challenged)
3. Utilize the LED's to signal the status of the sensor (green for sampling, red for transmitting)

Packet Structure

Also defined in sense.h, the structure is as follows:

1. sender_id: the ID of the sensor that originates the DATA packet, 2 bytes.
2. cnt: the sequence number, 2 bytes. The sequence number is generated by the originating sensor and is incremented for each new DATA packet.
3. val: the light intensity, 2 bytes.
4. fwd_id: the ID of the sensor who is forwarding the DATA packet packet, 2 bytes.
5. rec_id: the ID of the base station, 2 bytes.

Part 2

Implement the tracking network as follows: (network will use a static route, as discussed)

1. Sense light every 2 seconds,
2. Send the average of three data points to the next hop every 6 seconds in a DATA packet.
3. Use LEDs to indicate the status of the sensors.
4. Forward the received DATA packets from the previous hop to the next hop.

Your design and implementation should have the following structure:

1. The application should have two separate modules: sensing and forwarding. The two modules should be in two different .nc files.
2. The sensing module should indicate if a new data item is generated and sent.
3. The forwarding module should indicate when a new packet is received and forwarded.
4. If the transmission fails, the forwarding module should either buffer or discard outgoing packets. It's your choice which behavior to implement.

Lab Report

1. Please demonstrate your program to the lab instructor and let him check your code at the end of the current lab project.
2. Your project report is due at the beginning of the next lab.
3. Grading criteria
   - Demonstration, 15 percent
   - Code, 15 percent
   - Report, 70 percent

Report instructions

Format:

1. Include your name and ID in the first page
2. Font size of at least 10pt
3. Single spaced
4. Maximum of 8 pages (I will take points off for exceeding this without any good reason)
5. Please submit as PDF online, and turn in a hard copy

Content:

1. Introduction (10 percent of your report grade) Please summarize the task of this lab and what you have learned in the lab. Also include how you coordinated with the other lab members.

2. The code (90 percent of your report grade) Describe the design and implementation of your code, both the sensing and the forwarding modules. At a minimum, include the following:

   (a) How are the two modules you wrote organized in respect to the other modules?

   (b) What interfaces do your modules provide or use? What do those modules' interfaces do?

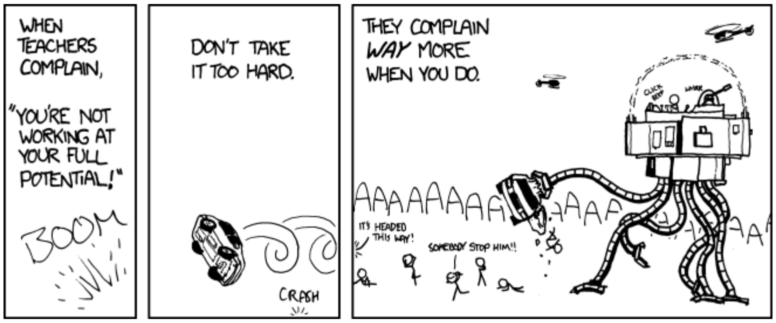
   (c) What data structure does the module use to buffer outgoing packets?

   (d) What will happen if too many packets are received before they can be forwarded out? How could this situation happen? (specify some hard numbers)

   (e) Does the module have the ability to detect transmission failure (sending and receiving)? If a transmission fails, what will your module do?

   (f) What will happen if a received packet does not have the right format? How does the module check it?

   (g) List any other design concerns not already discussed.



The bunch of disadvantaged kids I was tutoring became too good at writing, and their essays were forcing me to confront painful existential questions, so I started trying to turn them on to drugs and crime instead.