COSC402 Lab 6: Sensor Network Programming

Description

In this lesson, you will practice how to implement a routing scheme in Contiki. Specifically, you will

- Learn how to use the multihop Rime module,
- Learn how to use the announcement mechnasim,
- Learn how to manage a list using the list module,
- Learn how to allocate memory with the memb module.

Reference

Please refer to the following pages for the details of the modules to be used in this lab

- announcement.h (http://contiki.sourceforge.net/docs/2.6/a00318.html)
- list.h (http://contiki.sourceforge.net/docs/2.6/a00206.html)
- memb (http://contiki.sourceforge.net/docs/2.6/a00212.html),
- ctimer (http://contiki.sourceforge.net/docs/2.6/a01666.html).

Programming

The multihop module provides hooks for forwarding packets in a multi-hop fashion, but does not implement any routing protocol. A routing mechanism must be provided by the application or protocol running on top of the multihop module. In this lab, you will implement a very simple routing mechanism: each node maintains a list of neighbors, which it populated through the use of the announcement mechanism. Each node forwards every incoming packet to a random neighbour.
The neighbor list is populated by incoming announcements from neighbors. The program maintains a list of neighbors, where each entry is allocated from a MEMB() (memory block pool). Each neighbor has a timeout so that they do not occupy their list entry for too long.
When a packet arrives to the node, the function forward() is called by the multihop layer. This function picks a random neighbor to send the packet to. The packet may be forwarded by other node in the network until it reaches its final destination (or is discarded in transit due to a transmission error or a collision).

I have emailed you the skeleton code. Create a folder name “COSC402” under contiki-2.6, and further create a subfolder named “lab6”. Save the two files in lab6. Open the multihoprouting.c, and implement the following two functions.

Implement received_announcement()

This function is called when an incoming announcement arrives. The function checks the neighbor table to see if the neighbor is already in the list. If the neighbor is not in the list, a new neighbor table entry is allocated and added to the neighbor table.

Implement forward()

This function is called to forward a packet. The function picks a random neighbor from the neighbor list and returns its address. The multihop layer sends the packet to this address. If no neighbor is found, the function returns NULL to signal to the multihop layer that the packet should be dropped.

Test

Test in COOJA simulator

Create a simulation following these steps:

- create a new simulation and give a name ”multihoprouting”,
- create a mote type by choosing Sky Mote type
- choose the multihoprouting.c in the /TELE402/Multihop_routing/multihoprouting.c, compile it and create it.
- add 5 nodes. In the simulation visualizer, choose UDGM to show the communication range of each node. Adjust the position of each node to make it into a line network.
- start the simulation. Right click on a node and choose the “click button on Sky X” in the pop up menu. check the output in the Mote output window.

Test on sensor hardware

Each of you will be given a sensor mote. Please first burn your mote with an ID that is larger than 1 (refer to the instructions in previous lab). Discuss with your classmate to avoid using the same ID. Open ../Multihop_routing/multihoprouting.c, and go to the line “packetbuf_copyfrom(“hello”, 6)”. Change ”hello” to other string, e.g. your name and your student ID or anything else you like, then modify the length of your string (Remember that the length of the string includes the NULL terminator). Compile it and upload to the sensor mote using the following command:

```
cd contiki2.6/TELE402/Multihop_routing/
make TARGET=sky multihoprouting.upload
```

I will set up a sensor mote serving as the destination of your messages and show the output of my sensor mote to the project screen. After finishing uploading the problem to your mote, look at the projector screen and check if your message has appeared in it. Take your sensor mote to different position in the lab, and see what happens to the output.

Question

What problems does this routing protocol have? How to solve these problems?

Assignment 2

- Due by 4pm on September 13. Late submission will receive penalty (5% per day).
- This assignment contains the code you developed in labs 5 and 6. You just need to email me the following two files: flash.c (lab5) and multihoprouting.c (lab6).