Software for Intelligent Robotics

Assignment #4
Out: Tuesday, October 8, 2002
DUE: Thursday, October 17, 2002

Programming Assignment: Logical Sensors

As we studied in class, and as described in the Murphy text (section 6.1.1), logical sensors are a unit of sensing that supplies a particular percept. In this assignment, you will build a logical sensor that returns a polar plot of range data that can either use sonar or laser for constructing the polar plot. This logical sensor will be implemented as a perceptual schema. You will incorporate this into your code for Assignment #3, so that your behaviors can use either laser or sonar for avoiding obstacles (while also wandering, detecting beacon, and going to beacon).

For this assignment, please complete the following within the Nomad200 simulator:

1. Create a perceptual schema called `perceptual_schema_range` that extracts a polar plot of range data (which is the percept) and stores it into a data structure called `polar_range`. (You may define the actual organization of this data structure as you see fit.) This perceptual schema should have two logically equivalent modules for generating the polar plot based on either a) sonar, or b) laser. The perceptual schema should accept a parameter that indicates whether sonar or laser should be used for generating the polar plot of range data. Additionally, two other parameters are accepted by the perceptual schema that give the minimum and maximum angular values for which the polar range plot should be generated. For example, “perceptual_schema_range(laser,0,360)” would fill in the `polar_range` data structure with range values detected using the laser for the complete 360 degrees around the robot, while “perceptual_schema_range(sonar,+45,-45)” would use the sonar to generate a polar range plot within ±45 degrees of the forward direction of the robot. You may define the notation for the parameters that specify the angular range however you like (e.g., using positive and negative angular values, only positive angular values, etc.).

2. Convert your sonar processing code from your earlier assignments into a module within the above perceptual schema, such that it generates the `polar_range` data. The angular range of values generated is based upon the second and third parameters mentioned above.

3. Write a second logically equivalent module for the above perceptual schema that uses the laser range scanner to generate a polar plot of range data, also storing it in the `polar_range` data structure. (NOTE: see comments at end for more discussion regarding the use of the laser range scanner.) The angular range of values generated is based upon the second and third parameters mentioned above (in #1).

4. Convert your code from Assignment #3 (for wandering and going to beacon, etc.) so that the obstacle avoidance behavior no longer accesses the sonar data directly, but instead uses the `polar_range` data structure giving the polar plot of the range around the robot, which is populated by the `perceptual_schema_range` perceptual schema. The perceptual schema should be passed an argument indicating whether sonar or laser should be used to generate the polar plot, as well as the additional arguments indicating the angular values of the range data desired.

5. Run your revised code for wandering and going to beacon (from Assignment #3), using the same map4.txt, with the same beacon detection range of 2000. In this exercise, you will have one beacon position, which is (x,y) = (2700,1900). Run your revised code for two situations:
   a) Using the sonar module in the perceptual schema
   b) Using the laser module in the perceptual schema

You may use any angular range of values for these situations, as long as your code operates properly.
TURN IN THE FOLLOWING:

a) A hard copy of your complete code, fully documented.

b) Two screen dumps of your code from part 5 above, with the first screen dump showing your results using the sonar module, and the second screen dump showing your results using the laser module. In both cases, the beacon location should be (2700,1900).

c) A description of any differences you found in your robot control behavior when using laser versus using sonar for obstacle avoidance.

d) A discussion of the following topic: In an actual application, how would one go about determining which logically equivalent sensing method should be used, when multiple sensing methods are possible? What are the considerations that should be made? Do you think that sensor fusion of both the laser and the sonar would make sense in certain applications? If so, how might that be done? If not, why not?

e) Email an electronic version of your code to parker@cs.utk.edu, naming your code yourlastname-4.c (or .zip, etc.)

NOTES regarding the use of the laser range scanner:

One of the challenges in research in autonomous robotics is that often we have to figure out how to use various robotic components without being given step-by-step instructions or example code. In this exercise, you will need to use the documentation you have been provided to determine how to activate and use the laser scanner in your code. The method for configuring and accessing the laser data is related to how you accessed and used the sonar data, although there are several differences. You should have sufficient information in the documentation provided to determine how to access and use the laser data. However, it is not immediately straightforward, and will require some exploration on your part. The objective of this exercise is not to frustrate you (!), but to give you a small taste for how autonomous robotics research often requires digging through less-than-thorough documentation, and some trial-and-error experimentation, to be able to use a provided sensor or effector. In this exercise, you are welcome to discuss issues related to the configuration and access of the laser data with each other, although you should write your own code for actually using the laser data once you have determined how to obtain it.