Lab 5

Homing – Hybrid Control

Reading: *Introduction to AI Robotics (Ch. 7), Lec. 5-2*

(Demonstration due in class on **Thursday**)

(Code and Memo due in Moodle drop box by midnight on **Sunday at midnight**)

Read this entire lab procedure before coming to lab.

****

Purpose: The purpose of this lab is to use a type of locomotion called ***homing*** or ***docking*** with hybrid control to move the CEENBoT toward a light beacon. There will be a static light source placed in the environment which the robot can easily sense. The goal will be for the robot to move toward the beacon and stop just before hitting it. There will be no fixed path to the beacon, the robot should follow walls until the beacon is sensed, it should then leave the wall, keep track of its state and use the move to goal behavior to dock on the source while avoiding any obstacles along the way. Lastly, the robot should then turn 180 degrees and return to the wall to continue following as near as possible to the spot where it left. This will only be possible if the robot has kept track of its state.

Objectives: At the conclusion of this lab, the student should be able to:

- Use the photoresistor to sense a light beacon in the robot's environment
- Implement a hybrid controller on a mobile robot using a light sensor
- Integrate homing and docking into the behavior-based controller designed in prior labs

Equipment: Base Robot

2 photoresistors

range and contact sensors

Software: AVR Studio 4 (32-bit) available at http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725

WinAVR GCC toolchain (http://winavr.sourceforge.net/)

CEENBoT API static library available at

http://www.digital-brain.info/downloads/capi324v221-v1.09.002R.zip

****

LAB PROCEDURE

****

Part 1 – Move To Goal (Homing)

1. The hybrid control architecture that you will implement to home the robot includes a reactive layer (obstacle avoidance, wall following, move to goal, path update), middle layer (arbitrator), and deliberative layer (current state, path plan back to wall). This architecture is shown in Figure 1. Your code should be written in a modular fashion with functions such that it is evident where the planning, sensing and acting take place.

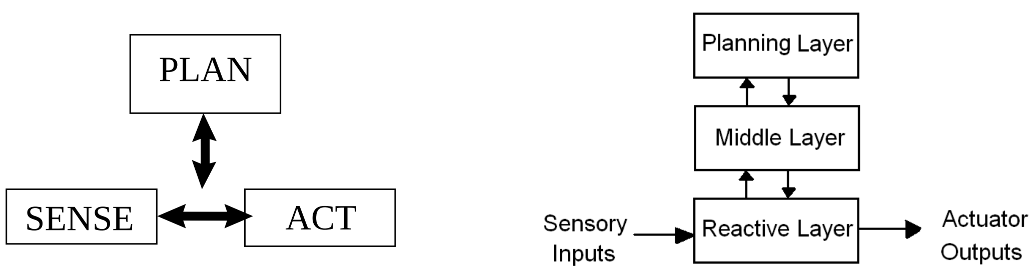

Figure 1: Homing Hybrid Control Architecture

2. The partial world map (representation) includes direction to the beacon and back to the wall with respect to the robot's current pose. This representation will be input into the deliberative layer for path planning. Updates to the path will be based upon feedback from the distance, heading and photoresistors. The middle layer will be used to make decisions about whether path updates are handled in the deliberative or reactive layer. The reactive layer will handle obstacle avoidance, wall following and move to goal behaviors. The robot should turn around and follow the path to drive back to the wall.

3. Based upon the above model, write code to home the CEENBoT robot to the light source (see Figure 2). The robot should come within one foot of the beacon without touching it.

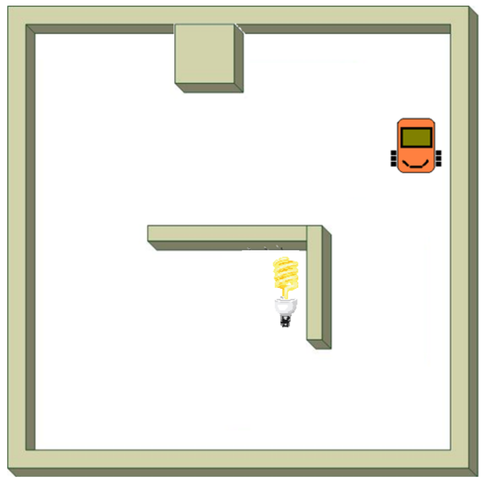

Figure 2: Robot homing

4. Test your final control algorithm for several different robot start points or beacon locations and summarize the results in your lab memo.

Part 2 – Dock the Robot and Return to the Wall

1. Improve the homing routine implemented in part 1 by docking the robot (back to the light). The robot should then follow the original path to drive back to the wall and continue to follow at the point where it left off.

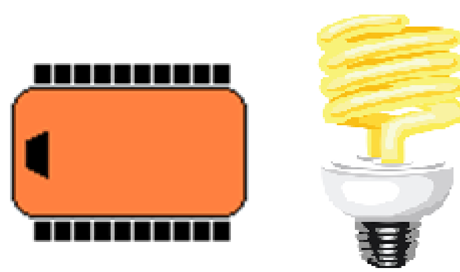

Figure 3: Robot Docking

Demonstration:

The demonstration of the program for lab 5 will include three parts. In the first part, the robot will be placed in the environment, wander until it finds a wall and then follow the wall until the beacon is detected. The robot should then move to goal and stop within one foot of the beacon. In the second part, the robot should turn and dock on the beacon. Lastly, the robot should return to the wall as close as possible to where it left off and continue to follow the wall.

Program:

The program should be properly commented and modular with each new behavior representing a new function call. The design of the subsumption architecture should be evident from the program layout. You should use the GUI, keypad, LCD and speech module as needed to illustrate robot state, input and output data.

Memo:

The following list provides the basic guidelines for writing a technical memorandum.

- ✓ Format
  - Begins with Date, To , From, Subject
  - Font no larger than 12 point font
  - Spacing no larger than double space
  - Written as a paragraph not bulleted list
  - No longer than three pages of text
- ✓ Writing
  - Memo is organized in a logical order
  - Writing is direct, concise and to the point
  - Written in first person from lab partners
  - Correct grammar, no spelling errors
- ✓ Content
  - Starts with a statement of purpose
  - Discusses the strategy or pseudocode for implementing the robot paths (may include a flow chart)
  - Discusses the tests and methods performed
  - States the results including error analysis
  - Shows data tables with error analysis and required plots or graphs
  - Answers all questions posed in the lab procedure
  - Clear statement of conclusions

Questions to Answer in the Memo:

1. What does the hybrid control architecture for your design look like? What was on the planning layer? Middle layer? Reactive layer?
2. What was your general strategy for planning the path back to the wall from the beacon?
3. How reliable was the photoresistor at detecting different objects at various shapes, sizes and distances. Compare and contrast sensor data.
4. How significant was the difference in photoresistor voltages for the left and right sides. How did you use this difference to extract directional information to move the robot toward the beacon?
5. How significant was the difference in sensor data based upon distance from the source? How did you use this difference to extract distance information to move the robot toward the beacon?
6. How did the architecture respond to differences in robot start position or beacon location?
7. How did the robot's hybrid controller respond to dynamic changes in the environment (i.e. other robots and people) and compare this to purely deliberative control.
8. Were there any challenges in implementing the homing routine?
9. What could you do to improve the robot homing?
10. How did docking the robot modify the control architecture or algorithm?

Grading Rubric:

The lab is worth a total of 30 points and is graded by the following rubric.

Points	**Demonstration**	**Code**	**Memo**
10	Excellent work, the robot performs exactly as required	Properly commented, easy to follow with modular components	Follows all guidelines and answers all questions posed
7.5	Performs most of the functionality with minor failures	Partial comments and/or not modular with objects	Does not answer some questions and/or has spelling, grammatical, content errors
5	Performs some of the functionality but with major failures or parts missing	No comments, not modular, not easy to follow	Multiple grammatical, format, content, spelling errors, questions not

			answered
0	Meets none of the design specifications or not submitted	Not submitted	Not submitted

Submission Requirements:

You must submit you properly commented code as a zipped folder of the C file and the lab memo in a zipped folder by **11:59 pm on Sunday** to the Moodle Course Drop box. Your code should be modular with functions and classes in order to make it more readable. You should use the push buttons, buzzers and LCD to command the robot and indicate the robot state during program execution.