Homework 3:
(1) Kinematic Feedback Control, and (2) Using Laser for Feature Extraction

Assigned: Thursday, Sept. 18, 2008
Due: Friday, Oct. 3 at 23:59:59 (i.e., before midnight)
(any homework turned in after 00:05:00 on Oct. 4 will be considered late, and will receive a grade of '0')

Part 1: Short answers about Kinematics

1. List three simplifying assumptions we make throughout the kinematic analyses that we have been studying that are not really true in the real world.

2. Let's say I super-glue the steering of my bicycle (at a small angle). How many ICR's does the modified bicycle have? Explain your answer.

3. Is the synchro-drive holonomic? Explain your answer.

4. Consider a bicycle where both wheels can be steered (the two-steer). Which is more maneuverable, the two-steer or the synchro-drive robot? Explain your answer.

Part 2: Implementing Kinematic Feedback Control

In this part of the homework, you will implement the feedback control approach for a differential drive vehicle, as discussed in Section 3.6.2 of our text. You will illustrate this control by driving your robot through a series of goal poses (specified below). Your vehicle should stop at each pose, then move on to the following pose, until the last specified pose is reached.

Your robot should be the normal differential drive Pioneer 2DX robot.

Use the same environment as in Part 2 of your Homework #2. However, you should only use one robot, and its starting pose should be (0,0,0). You won't need any beacons. In this homework, you must create your own .cfg and .world files to achieve this starting scenario. This can be done by modifying some of the earlier .cfg and .world files you have used in other assignments. (Note that the parsing of the .cfg and .world files that Player/Stage performs is not as "smart" as you might like. For example, the parsing is very sensitive to spaces placed in the wrong place. To avoid headaches, model your .cfg and .world files as closely as possible to examples you are already given, and pay attention to spaces!)

Here is how your starting simulation should appear:
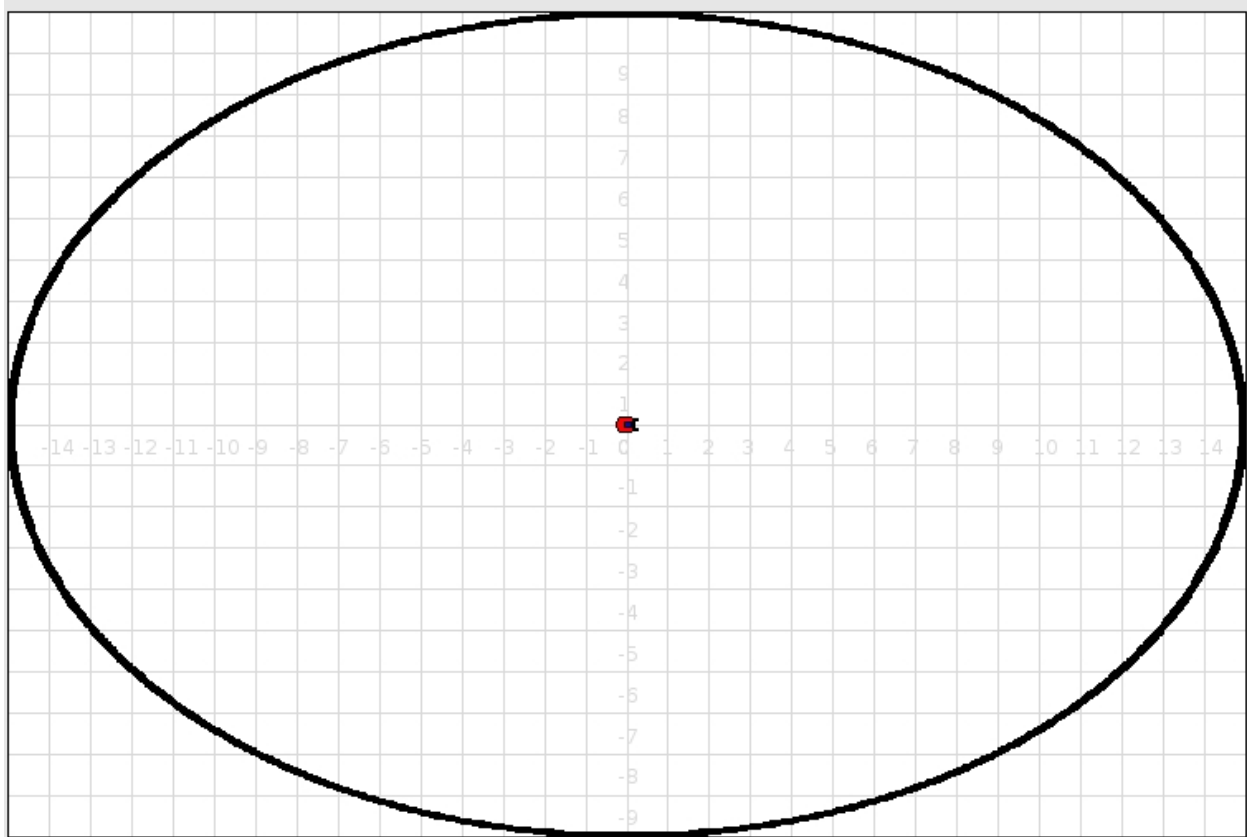
To test out your kinematic control, move your robot through the following series of goal poses; remember, the robot should stop at each pose:

(0,0,0) → (5,3,0) → (2,7,270) → (-7,4,0) → (-7, -3, 0) → (0, -3, 90) → (7, -5, 180) → (0, 0, 0)

What to submit for Part 2:
- A screenshot of the robot trace moving through each goal pose, from the starting position to the final goal pose.

Part 3: Feature extraction using laser data

Graduate Student Version:
In this part of the homework, you will make use of a laser range scanner on a mobile robot to determine the size of 3 boxes in the robot's environment.   The assumption is that the robot starts at the left side of a room at pose (-10, 3, 0), and 3 boxes of different sizes and orientations are lined up along the *x* axis, as shown here:
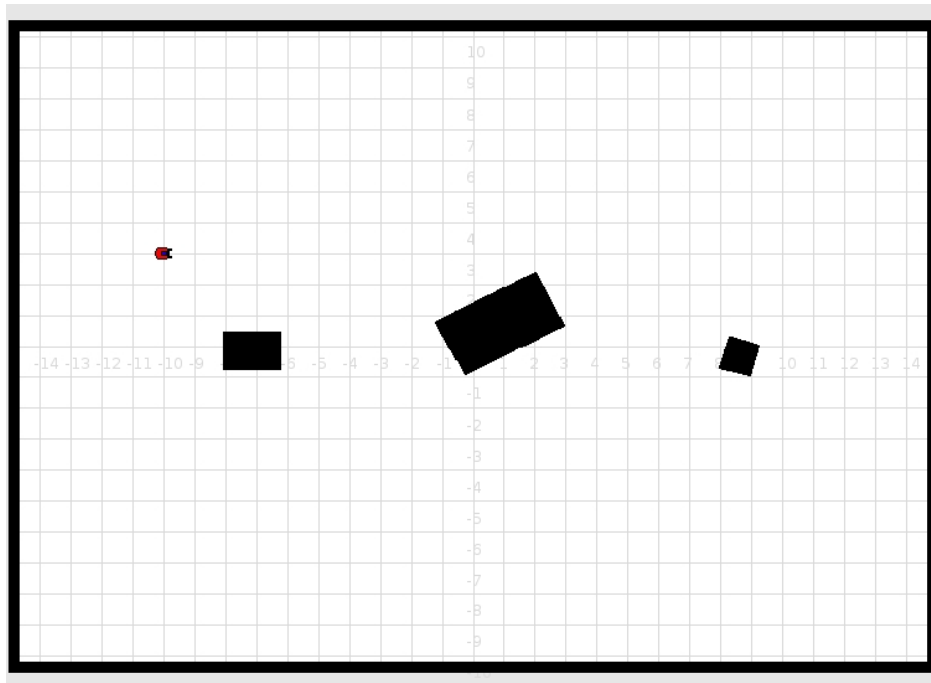
You don't know in advance the exact positions of the boxes, but you do know that some part of each box touches the *x* axis (i.e., where *y* = 0). And, you also know that the *x* value of the starting position of your robot is to the left of the minimum *x* value of the first box (from the left). (You can make use of the GPS localization on the robot to determine where your robot is.)

You are being provided with .cfg and .world files for this problem, called HW3-Part3.cfg and HW3-Part3.world, as well as the above bitmap image. These are on the class website, on the "Homework Assignments" page. *Be sure to use the correct environment – i.e., the one with 3 boxes as above; do this by uncommenting/commenting the proper bitmap in the .world file.*

Your job is to write robot code that moves the robot around each box and outputs the length of each side of the found boxes. Note that even though these boxes will be rectangular, you should go around all 4 sides of each box to determine its size, based on the robot's calculations using the laser. It doesn't matter which side you start with. Thus, the overall output should be something like:

```
Found Box 1:
        Side 1 = 2.76m
        Side 2 = 1.35m
        Side 3 = 2.68m
        Side 4 = 1.28m

Found Box 2:
        Side 1 = 2.13m
        Side 2 = 3.82m
        Side 3 = 2.27m
        Side 4 = 3.90m

(etc. for Box 3)
```

The focus of this exercise is in using the laser scanner data to fit a line to each side of the box to estimate its size. This is a feature extraction problem, as described in Sections 4.3.1.1 and 4.3.1.2 in our text.

What to submit for Part 3:
- A screenshot of the robot trace as it moves through the environment.
- A listing of your robot code output.

<u>***Undergraduate Student Version:***</u>

Your assignment is similar to the graduate students' version, except you only have to deal with finding the size of 1 box. Here, the assumption is that the robot starts at the left side of a room at pose (-10, 3, 0), and 1 box of some size and orientation is on the *x* axis, as shown here:
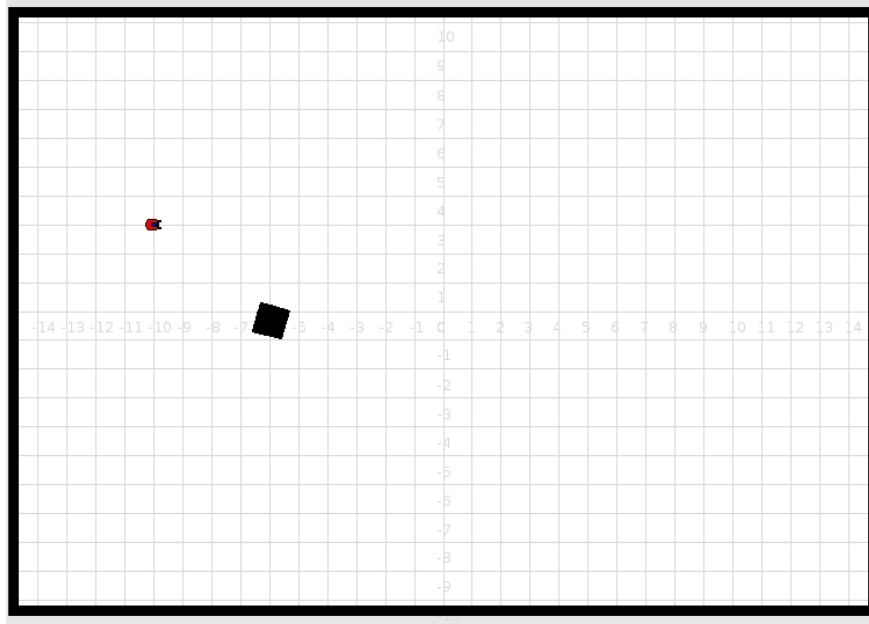
You don't know in advance the exact position of box, but you do know that some part of the box touches the *x* axis (i.e., where $y = 0$). And, you also know that the *x* value of the starting position of your robot is to the left of the minimum *x* value of the box. (You can make use of the GPS localization on the robot to determine where your robot is.)

You are being provided with .cfg and .world files for this problem, called HW3-Part3.cfg and HW3-Part3.world, as well as the above bitmap file. These are on the class website, on the "Homework Assignments" page.

Your job is to write robot code that finds the box, and outputs the length of each side of the found box. Note that even though this box will be rectangular, you should go around all 4 sides of the box to determine the length of each side, based on the robot's calculations using the laser. It doesn't matter which side you start with. Thus, the overall output should be something like:

```
Found Box:
      Side 1 = 2.76m
      Side 2 = 1.35m
      Side 3 = 2.68m
      Side 4 = 1.28m
```

The focus of this exercise is in using the laser scanner data to fit a line to each side of the box to estimate its shape. This is a feature extraction problem, as described in Sections 4.3.1.1 and 4.3.1.2 in our text.

What to submit for Part 3:
- A screenshot of the robot trace as it moves through the environment.
- A listing of your robot code output.

SUBMITTING YOUR HOMEWORK:

Place all your files in a single directory. These files should include:

- Your written answers, screenshots, and robot output as requested above, in a single pdf file called "yourlastname-HW3.pdf")

- For Part 2:
  - Your configuration file, called "HW3-Part2.cfg"
  - Your world file, called "HW3-Part2.world"
  - Your makefile, called "makefile" or "Makefile"
  - Your robot control code, called "yourlastname-HW3-Part2.cc"
  - A README file giving the command line arguments to run your code (it may simply be "./yourlastname-HW3-Part2", but if you require command line arguments, please specify them here).

- For Part 3:
  - Your configuration file, called "HW3-Part3.cfg"
  - Your world file, called "HW3-Part3.world"
  - Your makefile, called "makefile" or "Makefile"
  - Your robot control code, called "yourlastname-HW3-Part3.cc" .
  - A README file giving the command line arguments to run your code (it may simply be "./yourlastname-HW3-Part3", but if you require command line arguments, please specify them here).

Remove all other unnecessary files. Use the submit script **494mr_submit** or **594mr_submit** to submit your files. (These will be emailed to Dr. Parker.)