

CSE 397/498-013
Introduction to Mobile Robotics

Homework: Dead Reckoning via an Inertial Navigation System (INS)

Report Due Date: Tuesday, 4 Oct 05 submitted via Blackboard PRIOR TO THE START OF CLASS

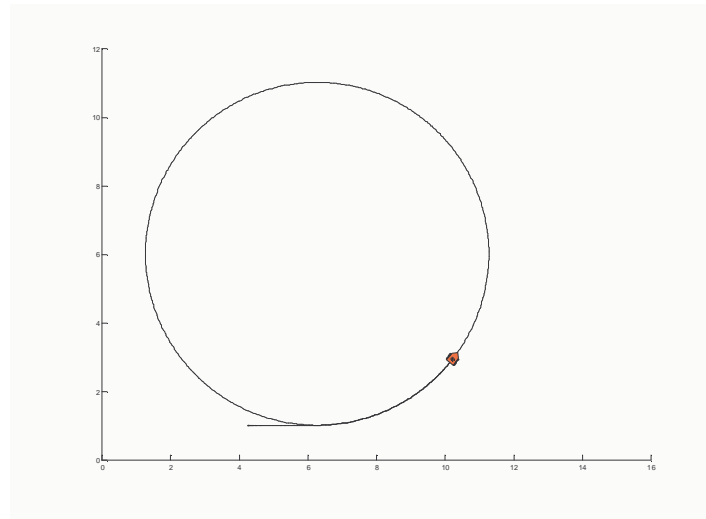


Figure 1: Trajectory estimated from sample input file.

A. Objectives:

1. Investigate the use of inertial sensors (accelerometers and a rate gyroscope) for dead reckoning in a differential drive mobile robot.
2. Investigate the effects of small biases on inertial navigation performance.

B. Assumptions: In all of the INS experiments, we will assume that

1. Our inertial measurement unit (IMU) consists of x and y-axis accelerometers, and a z-axis rate gyroscope.
2. The IMU frame is coincident with the robot frame.
3. Our kinematic model for a differential drive robot still applies.

C. Requirements:

1. You **MUST** use Matlab to complete this assignment.
2. This is an individual assignment. Each student is required to submit his/her own work in order to receive credit.
3. As stated on the web page, if you turn this assignment in late without coordinating with me first you will receive a 0.

D. An aside into structured lighting.

1. Do the derivations of equations 4.12-4.14 in your text.
2. How will a small error in detector u translate into an error in depth? Show this mathematically.

E. Inertial Navigation:

1. Write a function with the following prototype:

```
function hwk2(inputFile)
```

where `inputfile` is the name of a file consisting of robot accelerations in the X_R and Y_R directions (in m/s^2), as well as angular velocity $\dot{\theta}$.

2. Your function will:
 - a. Take a mouse click input from the user and create a robot at that position with $\theta=0$.
 - b. Read in the inertial sensor measurements (sampled at 50Hz).
 - c. Use these to update the pose of the robot using the `moveRobot` function.
 - d. We will assume that the robot is starting from a stopped position.
3. Some helpful Matlab functions:
 - a. `ginput`
 - b. `help` If you don't know what `ginput` does, type `help ginput` at the command prompt (it can be faster than the html help)
4. You can use the sample input file to test the functionality of your program. If it is working correctly, the robot should move at constant speed in a circle going counterclockwise (see Figure 1).
5. You should also test your program with other input files of your own creation to ensure proper functionality of your program.

F. Dead reckoning with bias:

1. In this exercise, we will test the effects of bias in inertial measurements on dead reckoning performance.
2. Generate an input file which reflects the robot moving from standstill to 2 m/s in 5 seconds under constant x-acceleration, and then travels a total distance of 50 meters. Run this trial using your `hwk2` function and include the figure in your report.
3. Repeat the above trial by introducing bias to the measurements obtained in F.2 above to reflect:
 - a. Bias to the x accelerometer of 0.005 g
 - b. Bias to the y accelerometer of 0.005 g
 - c. Bias to the y accelerometer of 0.005 g and bias to the angular velocity of 0.01 rad/s

Include a figure from each trial in your report. Discuss the effects of each component, to include error magnitude, rate of error growth, etc.

G. Additional Questions:

1. Some odometry systems integrate INS components to improve performance. For the wheel bias example from the first homework, discuss how you might improve odometry performance using:
 - a. A single accelerometer
 - b. A single rate gyroscope.

H. Turn in:

1. A write-up, to include images from all simulation trials as well as your Matlab source code.
2. The input files you generated for part F above.