ECSE 4490/6490 January 10, 2022

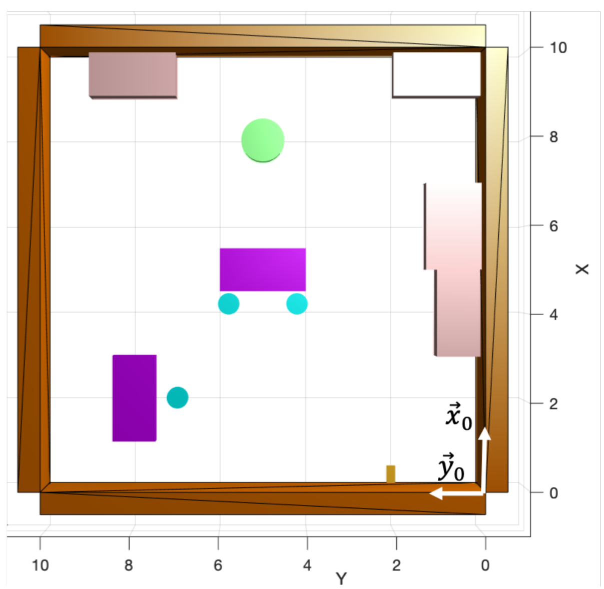
Robotics II

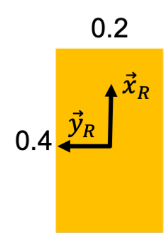
Mini-Project # 1

Assigned: January 10, 2022
Due: February 3, 2022
Full Score: 100 points

Please submit your project report to Gradescope and program code to your GitHub. Discussion with your peers is encouraged, but you must hand in your own work. Verbatim copying (whether you copy from someone or let someone copy your work) is considered cheating and will result in zero for the project grade and notification to the Class Dean and Dean of Students. Multiple instances of cheating will result in failing of the course.

1. Consider a room with some furniture as shown in the figure below. A mobile robot with length, width, height: .4×.2×.2 m, shown in gold. The robot zero configuration is as indicated





The MATLAB file `room.m` that was used to generate this scene is provided in the Google Doc.

The idealized mobile robot kinematics is given by (true kinematics will be a perturbed version)

$$\dot{x} = \cos(\theta)u_1$$
$$\dot{y} = \sin(\theta)u_1$$
$$\dot{\theta} = u_2$$

where (u_1, u_2) are the forward and turning velocities respectively. Write this as

$$\dot{q} = f(q)u$$

where

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, \quad f(q) = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix}, \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}.$$

Suppose the following sensors are available:

(a) Proximity sensor: detecting the nearest distance between the robot and the environment. This has a limited range.

(b) IMU: robot acceleration and turning velocity, in the body frame.

(c) Pin-hole camera: projection onto the camera image plane

(d) Local GPS sensors: the distance from the center of the robot to the corners of the room at the top of the ceiling. The error increases with distance.

(e) Lidar: scanner at the geometric center of the robot. It can generate N equally spaced scans in 360° to measure the distance to the environment.

You can also propose landmarks, e.g., markers for cameras. You may propose the inclusion or exclusion of sensors for your project subject to instructor's consent. Please feel free to exchange ideas with other students in class (via WebEx Projet 1 subspace).

2. Your project should include the following components

   (a) **Robot Model Uncertainty**: Assume some discrepancy between the true robot kinematics and the best guess model. One possibility is to include an additive noise to the robot kinematic model, i.e., the truth model is $\dot{q} = f(q)u + w$ where w is a zero mean Gaussian random (vector) variable.

   (b) **Robot Motion**: Update the robot state q based on the commanded velocity input, u and current robot state q using the assume truth model.

   (c) **Sensor Noise Model**: Each sensor measurement should include some noise. The noise model should be at least an additive zero mean Gaussian random variable with constant noise covariance (which may assumed known and be included in your algorithm).

   (d) **Sensor Measurements**: Given the robot state, generate your chosen sensor measurement. In the case of additive noise, $y = h(q) + v$, where v is a zero mean Gaussian random variable with known noise covariance.

   (e) **Localization**: Estimate the robot state at time t, $\hat{q}(t)$, based on current and past sensor output measurements, $\{y(\tau) : 0 \le \tau \le t\}$, past inputs $\{u(\tau) : 0 \le \tau \le t\}$, assumed kinematic model, sensor noise statistics, and known landmark locations.

   (f) **Mapping:** Given the robot state history, $\{q(t), t \in [0, T]\}$, sensor output measurements, and sensor noise statistics, estimate the landmark locations and generate a map of the room.

   (g) **Simultaneous Localization and Mapping (SLAM)**: Estimate the robot state and the landmark locations simultaneous based on the past sensor output measurements, noise statistics, and inputs.

(h) **Room Coverage**: Devise an algorithm to generate the input command $u(t)$ for the robot to cover the room. For example, it could be a random walk algorithm for the robot to move in a straight line in a random direction until it runs close to an obstacle. Then change to an arbitrary random direction away from the obstacle.

(i) **Evaluation and Comparison**: Compare sensor-based estimation vs. dead reckoning vs. ground truth. Evaluate the impact of noise covariance and possibly different distributions (e.g., non-Gaussian).