Probabilisti Robotics
Monte Carlo Localization - Laboratory exercise

This document will guide you through the 2^{nd} practical work related with the Localization subject of the Autonomous Robots course.

1. **The goal**

The goal of this practical exercise is to implement a Montecarlo Localization (MCL) algorithm to localize an one-dimensional Mobile robot being moved, with constant velocity, in a hallway. You will program all the code in Matlab.

2. **The MCL Problem**

The MCL localization is an implementation of the Markovian localization problem where the involved pdfs are represented through samples (particles) and the Bayes filter is implemented through the Particle Filter. Markov localization addresses the problem of state estimation from sensor data. Markov localization is a probabilistic algorithm: Instead of maintaining a single hypothesis as to where in the world a robot might be, Markov localization maintains a probability distribution over the space of all such hypotheses. The probabilistic representation allows it to weigh these different hypotheses in a mathematically sound way.

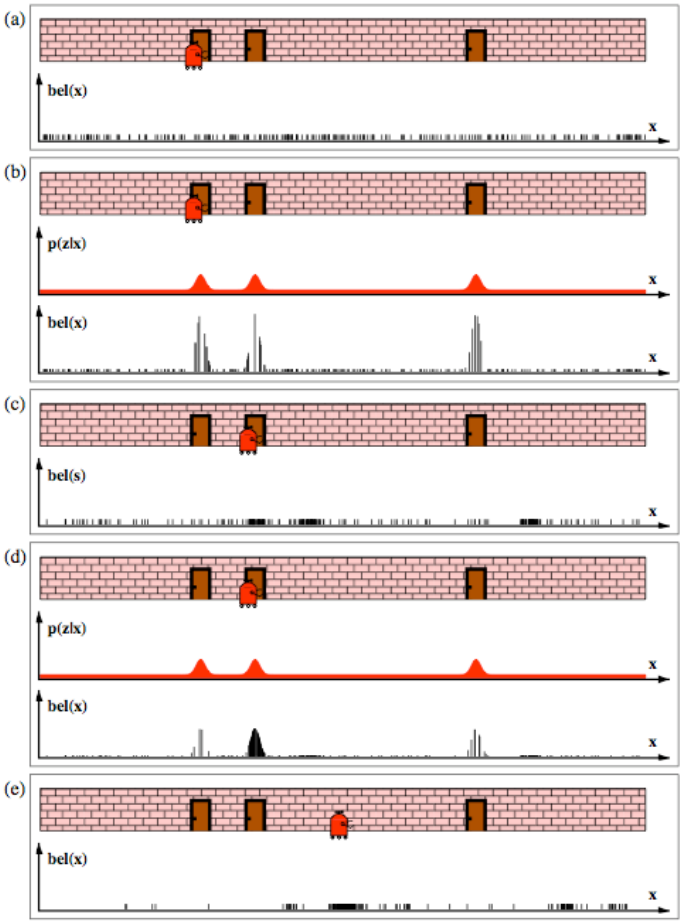

Figure 1 MCL of a Mobile Robot Moving in a Hallway.

3. The Algorithm

The Figure 3 shows the *pseudocode* of the grid localization algorithm to be implemented.

Algorithm Particle_filter($\mathcal{X}_{t-1}, u_t, z_t$)**:**
    $\bar{\mathcal{X}}_t = \mathcal{X}_t = \emptyset$
    for $m = 1$ to M do
        sample $x_t^{[m]} \sim p(x_t \mid u_t, x_{t-1}^{[m]})$
        $w_t^{[m]} = p(z_t \mid x_t^{[m]})$
        $\bar{\mathcal{X}}_t = \bar{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
    endfor
    for $m = 1$ to M do
        draw i with probability $\propto w_t^{[i]}$
        add $x_t^{[i]}$ to $\mathcal{X}_t$
    endfor
    return $\mathcal{X}_t$

Figure 2 Particle filter algorithms

Algorithm MCL($\mathcal{X}_{t-1}, u_t, z_t, m$):
$\bar{\mathcal{X}}_t = \mathcal{X}_t = \emptyset$
for $m = 1$ to M do
$x_t^{[m]} =$ **sample_motion_model**$(u_t, x_{t-1}^{[m]})$
$w_t^{[m]} =$ **measurement_model**$(z_t, x_t^{[m]}, m)$
$\bar{\mathcal{X}}_t = \bar{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
endfor
for $m = 1$ to M do
draw i with probability $\propto w_t^{[i]}$
add $x_t^{[i]}$ to $\mathcal{X}_t$
endfor
return $\mathcal{X}_t$

Figure 3 MCL Algorithm3

4. Work to do

During these lab session you must complete the m-file containing the MCL algorithm. These are the steps you must follow:

a) First: Read the m-file and try to understand how it works. Even though it is incomplete, it is possible to run it. For the animation to work, it is necessary to have the animation.mat (see previous lab) file in the same directory of the m-file.

b) Second: Read the help of the **random** and **pdf** MATLAB functions. They will be useful for this lab.

c) Third: Program the *sample_motion_model* and the *MCL* functions (focusing on the prediction section of the last one) and complete the main iteration.

d) Fourth: Program the *measurement_model* and the *MCL* function (focusing on the update section of the last one).

e) Fifth: Program the resampling step of the MCL function.

All the code sections to be programmed are tagged with the tag %COMPLETE ….. Feel free to change/update whatever other section you desire.

f) After the lab session: Write a report explaining your solution. Include the final m-file.

Note: One session is allocated for this lab.