EECS 270 – Robot Algorithms
Lab N. 4 – Fall 2017

Important Information

1. This lab is due by the deadline specified in CatCourses.
2. You solution must be submitted electronically through CatCourses.
3. Labs MUST be solved individually.
4. Your solution must be coded in Matlab and strictly follow the given requirements. Failure to comply with the desired structure for the input/output files leads to an immediate 0.

Particle filter

Using the same environment presented in Lab 3, solve the *global localization problem* using a particle filter approach. Consider the following suggestions and constraints while solving this exercise.

1. At initialization time generate M particles uniformly distributed over $\mathcal{C} = [-4, 6] \times [-3, 10] \times [0, 2\pi)$.
2. Note that this time the robot does not start from $(0, 0, 0)$, but rather from an unknown pose.
3. To sample a new pose x_t starting from x_{t-1} and u_t, use the algorithm (`sample_motion_model_velocity`, see next page) and use $\alpha_{1...4} = 0.01$; $\alpha_{5,6} = 0.0005$. For the function `sample`, use a Gaussian distribution with 0 mean (you can use Matlab's built in `randn` function).
4. For the sensor model $p(z|x)$ consider the algorithm `beam_range_finder_model` given in the next page. Note that for the environment at hand we have $K = 3$. Assume p_{hit} is Normally distributed with variance/standard deviation $\sigma^2_{hit} = 0.001$.

Questions:

1. Try different values of M and comment about the tradeoff between speed and accuracy. Justify your observations with numbers.
2. Based on the best value of M that you determined, provide an estimate for the final pose of the robot. Explain how you compute your answer.

3. Produce a figure showing the distribution of the particles at the end (use the same M you used in the previous question and plot just the x, y components).

Submit your code and answers via UCM crops. Add also a short description about how your code can be run.

Algorithm 1 Sample_Motion_Model_Velocity(u_t, x_{t-1})

$\hat{v} = v + \text{sample}(\alpha_1|v| + \alpha_2|\omega|)$
$\hat{\omega} = \omega + \text{sample}(\alpha_3|v| + \alpha_4|\omega|)$
$\hat{\gamma} = \text{sample}(\alpha_5|v| + \alpha_6|\omega|)$
$x' = x - \frac{\hat{v}}{\hat{\omega}} \sin\theta + \frac{\hat{v}}{\hat{\omega}} \sin(\theta + \hat{\omega}\Delta t)$
$y' = y + \frac{\hat{v}}{\hat{\omega}} \cos\theta - \frac{\hat{v}}{\hat{\omega}} \cos(\theta + \hat{\omega}\Delta t)$
$\theta' = \theta + \hat{\omega}\Delta t + \hat{\gamma}\Delta t$
return $x_t = (x', y', \theta')^T$

Algorithm 2 Beam_Range_Finder_Model(z_t, x_t, m)

$q = 1$
for $k = 1$ to K **do**
  Compute $z_t^{k^*}$, i.e. the correct value based on x_t. Let $p_{hit} = N(z_t^{k^*}, \sigma_{hit}^2)$
  $q = q \cdot p_{hit}(z_k^t)$
end for
return q