CS545 LAB 2: Monte Carlo Localization

Seokjoo Yoo seokjooy@usc.edu
Stephen Iota iota@usc.edu
Pranavi Jalapati jalapati@usc.edu
Satya Narapata narapata@usc.edu

Viterbi School of Engineering, University of Southern California

Note for graders: We, as a lab group, were unable to get `numba` working on any of our machines. So, we had to use a small number of particles in order for the experiments to finish in time. However, a higher variance in the motion model will decrease its performance.

Lab 2

1. **Q**: *How does each of the model variance parameters affect the performance of the localizer? Give possible reasons for the behavior you see.*

   A: A very small sensor variance can result in very confident yet incorrect estimates; really good observation models result in bad particle filters. Conversely, a higher motion model variance(Linear_model_val_x, Linear_model_val_y, Angular_model_var) will lead to higher uncertainty in localization.

2. **Q**: *How does the number of particles affect the behavior of the localizer?*

   A: Using a higher number of particles better represents next state distribution $p(\mathbf{x}_t|\mathbf{u}_t, \mathbf{x}_{t-1})$ before integrating the current measurement at timestep t. Assuming this distribution is Normal, the localizer is more likely to use the expected value (the mean), and less likely to be affected by outliers drawn during resampling (particles with low associated weights).

   With a low number of particles, there may be no particles to sample from in the vicinity of the correct state - not enough particles to cover all relevant regions with high likelihood. In this case, a few bad draws during the resampling step can cause the estimated and target position distributions to diverge significantly.

3. **Q**: *Can a particle filter with a single particle perform well? Why or why not? What if it starts in the correct position?*

   A: When using only a single particle, the resampling procedure becomes redundant - the single particle is propagated at each iteration and the measurement probability $p(\mathbf{z}_t|\mathbf{x}_t)$ does not provide any additional information to the filter. If the motion model is highly accurate, i.e. we trust $p(\mathbf{x}_t|\mathbf{u}_t)$ to have low variance and noise, and the initial belief $p(\mathbf{x}_0)$ is highly accurate, the model can still produce a bad particle filter. The updates of the belief of the particle filter are likely to result in confident (closely clustered), yet incorrect estimates.

Extra credit 3

Better particle filter through clustering. For this exercise, we modified the `GetMeanPose` method of the `ParticleFilter` class. We used a KMean clustering model to cluster the x, y, θ translations into 8 clusters, then used the average of the particles in the cluster with the highest frequency respectively to estimate the posterior distribution.

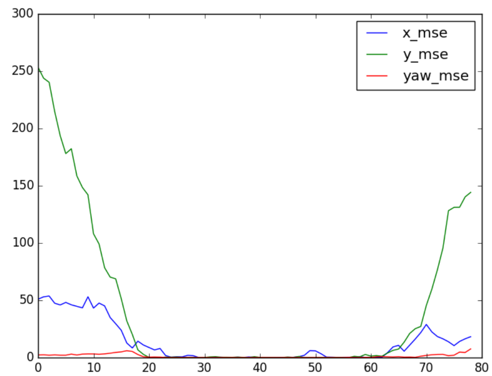

Figure 1: Error graph generated by the localizer, low number of particles(200) due to hardware constraint.