

Writeup for HW6 Programming Part

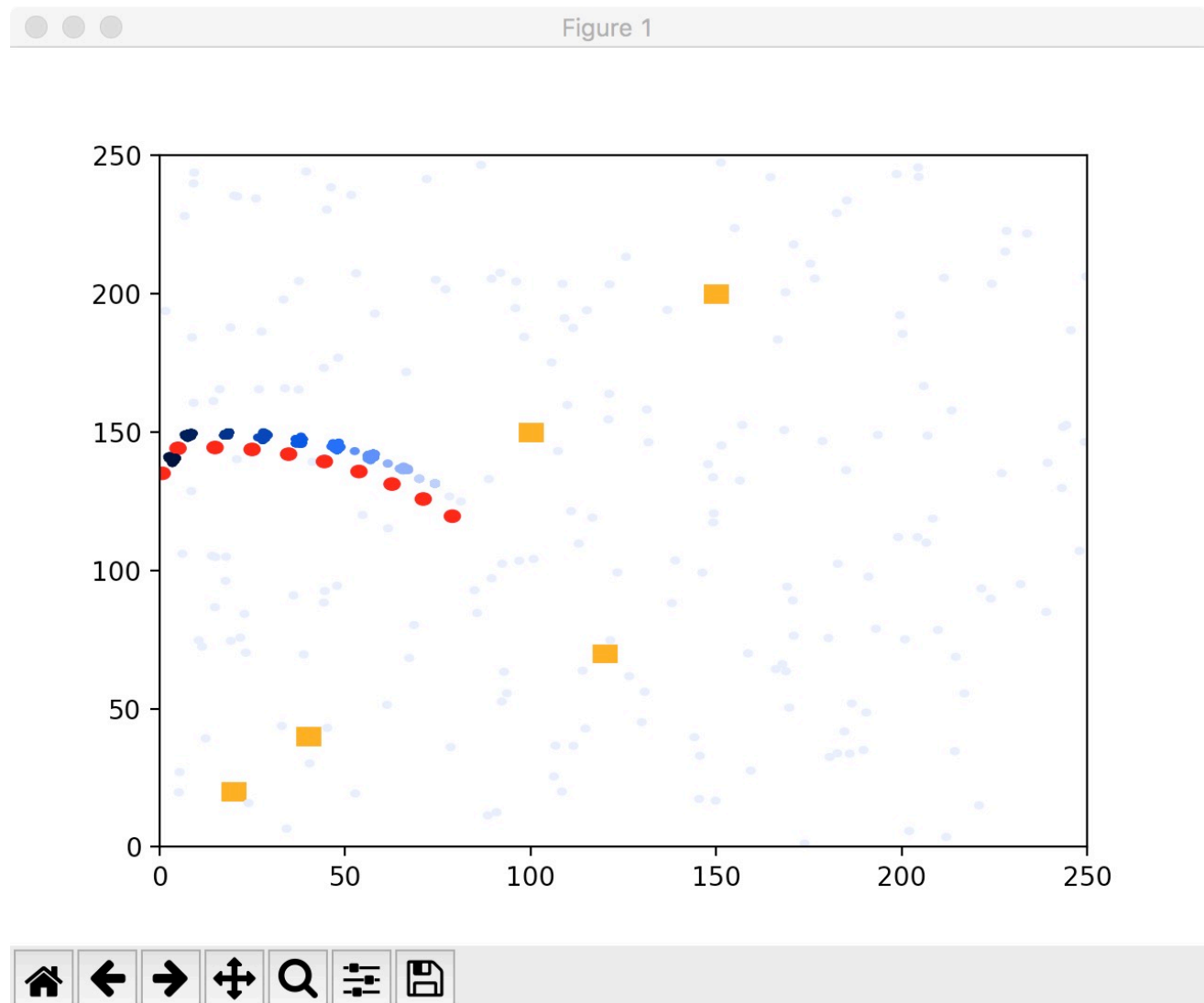
Names: Jing Qian & Qing Teng

UNIs: jq2282 & qt2126

Implementation choices:

1) Motion model

The robot turns a small angle α (left as a hyperparameter, default is 0.1 radians), then moves forward by 10 units each time. However, if it detects a landmark or a wall along its path, it will turn by another α , and continue doing so until the path is clear, before moving forward. As we can see in this screenshot, when the robot is close to the wall, it turns by a much sharper angle than before.



2) Normalizing weights

We calculated the weights for the particles using the method given, and normalized them by dividing by their sum. We did not use log probabilities because normalizing fixed the underflow issue.

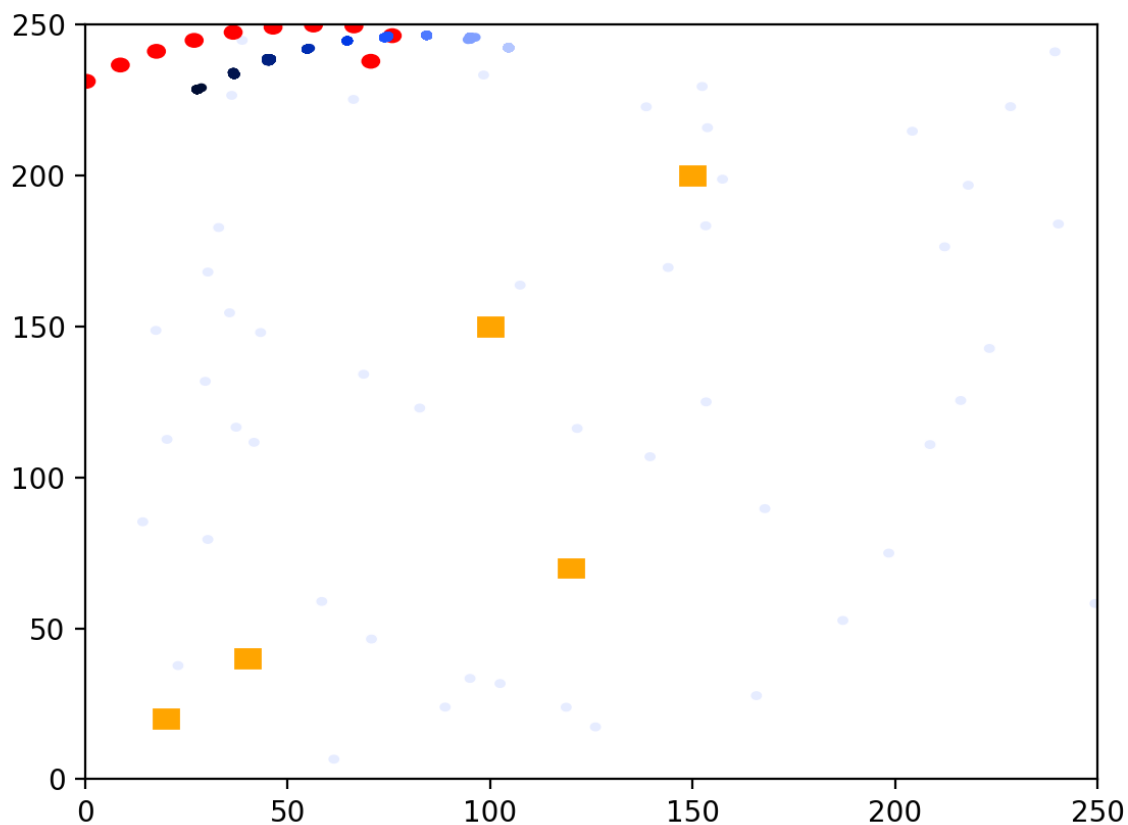
Effectiveness of weighting and resampling:

It is very effective, because initially the particles are scattered everywhere, but they quickly converge to be near the robot's actual location. You can see the comparison between having different numbers of particles in the next section.

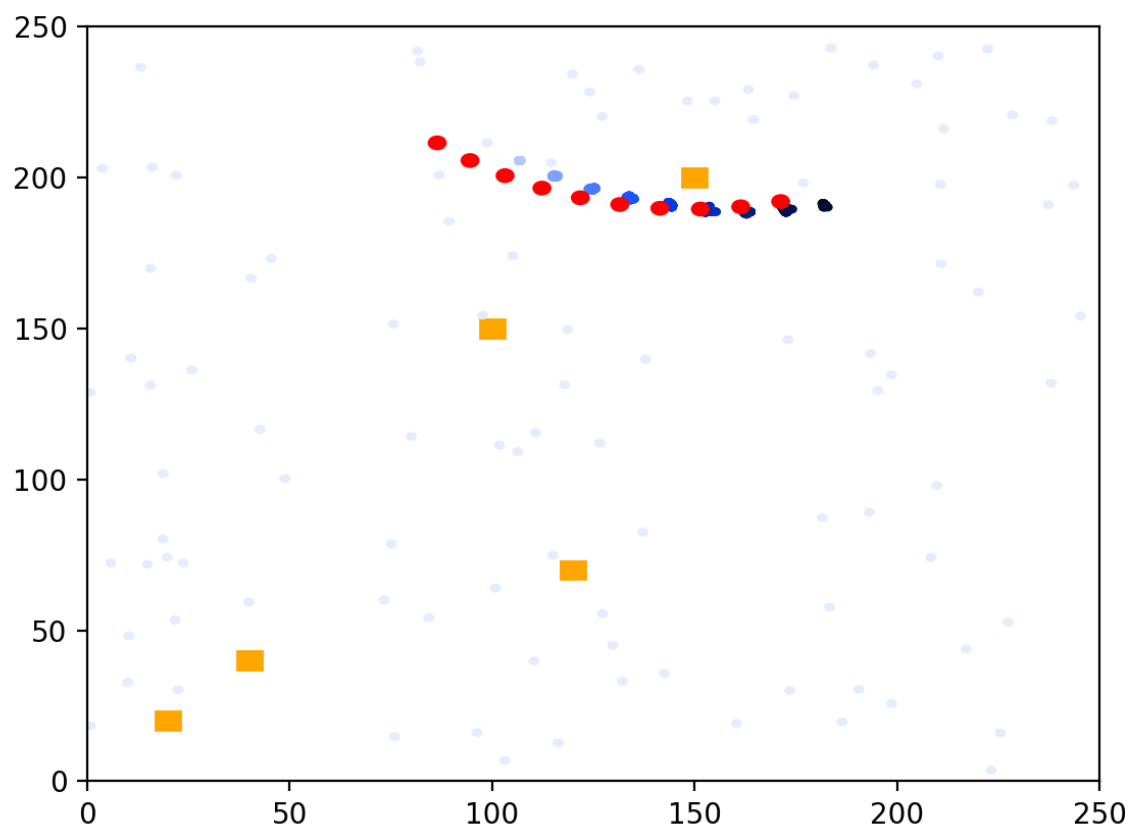
In our first video, we are using the provided `world.txt`, and the convergence is very fast. We also tested on a custom world file `world1.txt` (included in the submission), with more obstacles, and convergence is slower, as can be seen in our second video.

Results from different numbers of particles:

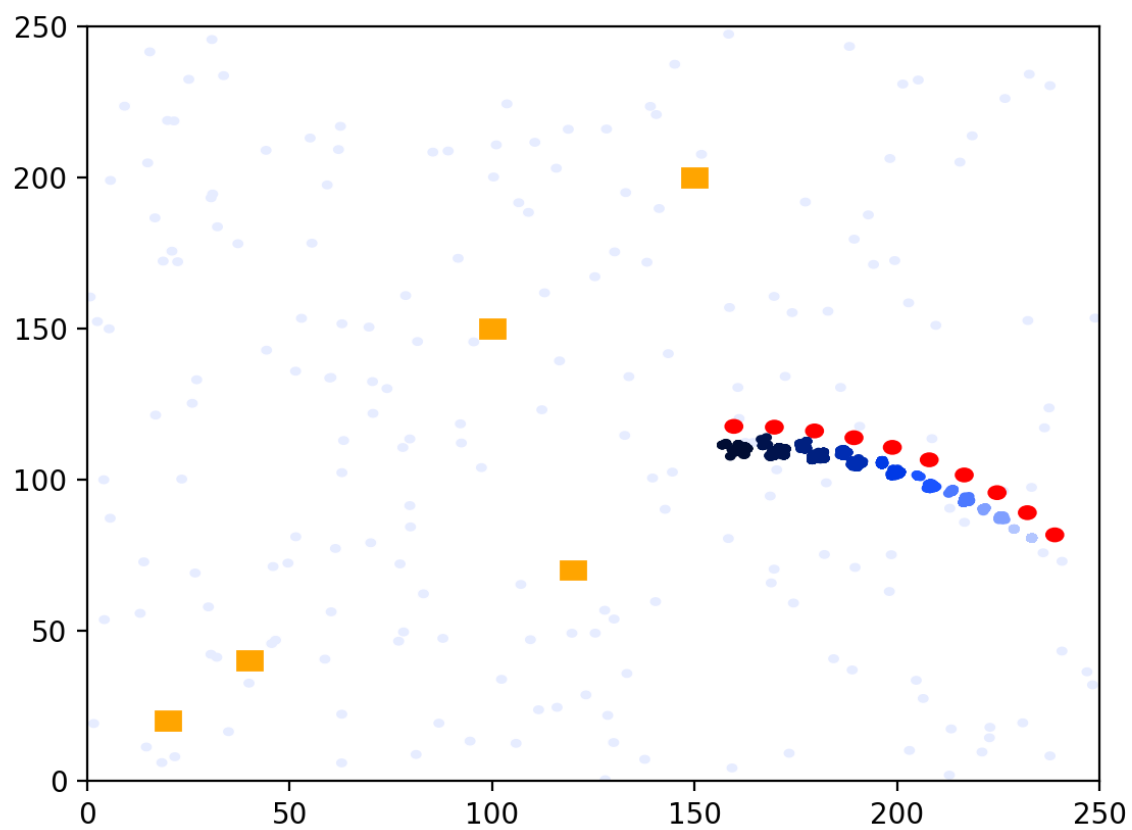
50 particles:



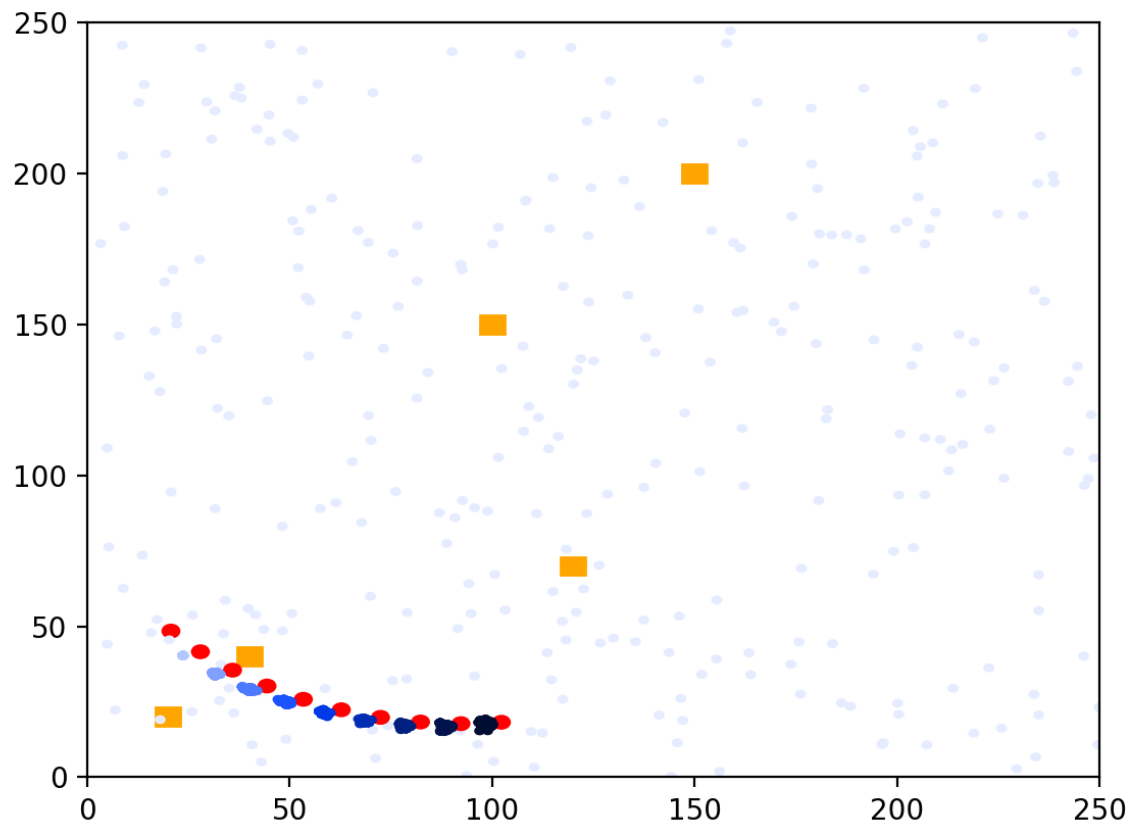
100 particles:



200 particles:



300 particles:



As we can see, the error is smaller when we have more particles. We chose to use 200 particles in the end.