Problem 4 (30 points)**.** Using 4 measurement directions, keep the sensor limit constant at 15, run your algorithm with number of particles 500, 1000, 1500. How does changing the number of particles influence the estimation accuracy, converging speed and computational cost of the algorithm? Record a video of one of the three runs. The video should include the turtle map window. Provide a link to the video and include it in the report.

https://drive.google.com/drive/folders/1s_mjOR1UJbJXIeA6fg2ZDpUKpbv9dQXd?usp=share_link

Increasing number of particles increases computational cost and converging speed. Although the estimation accuracy increases with a high number of particles, because when the number of particles is small, they may govern the distribution, ultimately leading to an inaccurate representation of state distribution.

Problem 5 (30 points)**.** Using 4 measurement directions, keep the number of particles constant at 1000, run your algorithm with sensor limit 15, 20, 25. How does changing sensor limit influence the estimation accuracy, converging speed and computation cost of the algorithm? Record a video of one of the three runs. The video should include the turtle map window. Provide a link to the video and include it in the report.

https://drive.google.com/drive/folders/1s_mjOR1UJbJXIeA6fg2ZDpUKpbv9dQXd?usp=share_link

With increasing the sensor limit, the estimation of system state is more accurate as the particle weight updates are better ultimately leading to a better localization. Converging speed decreases and computational cost can vary based on case-by-case basis.

Problem 6 (10 points)**.** Does the particle filter you implemented performs evenly well through the whole environment after converging? More specifically, does the your particle filter have larger prediction error in some regions of the environment then other regions? If yes, can you explain why this is happening?

No, it does not perform evenly well because in some areas it confuses similar environments in other parts of the map. In some parts of the map the prediction error is larger, as seen in figures 1 and 2. This is prominent in corridors without doors.

Problem 7 (20 points)**.** Modify the LidarProcessing module and the sensor model so that they can make measurements in 8 directions. Run your algorithm with number of particles 1000 and sensor limit 20. How does having more sensor data influence the estimation accuracy and converging speed of the algorithm? Record a video of the run. Besides the turtle map window, the video should include the RViz window of the sensor measurements. Provide a link to the video and include it in the report.

More sensor data provides a better estimation accuracy and converging speed.

https://drive.google.com/drive/folders/1s_mjOR1UJbJXIeA6fg2ZDpUKpbv9dQXd?usp=share_link

Problem 8 (5 bonus points)**.** Activate the measurement noise as outlined in Section 3.4. This will result in occasional absence of the Lidar measurements. Run your algorithm with particles 1500 and sensor limit 25. How does the missing lidar measurement influence the estimation accuracy and converging speed? Provide a link to the video and include it in the report.

https://drive.google.com/drive/folders/1s_mjOR1UJbJXIeA6fg2ZDpUKpbv9dQXd?usp=share_link

Problem 9 (5 bonus points)**.** Modify the distribution of your initial particles to be within the top-right quadrant. Run your algorithm with particles 500 and sensor limit 15. How does the initial distribution of the particles influence the estimation accuracy and converging speed? Provide a link to the video and include it in the report.