RnD experiments analysis

Mohammed Waseem

1 Particle Filter Benchmarking: Linear and Rotational Motion Analysis

In this experiment, I conducted analyses comparing the performance of three distinct particle filter algorithms. Specifically, I evaluated AMCL alongside two variants of the optimal particle filter. The first variant involves interpolating 2D lidar data into 3D, while the second directly utilizes 2D lidar data. These experiments were carried out using a TurtleBot3 within a custom environment on ROS Noetic.

The initial experiment was to check the performance of AMCL under the default parameters provided by the Turtlebot3 package.

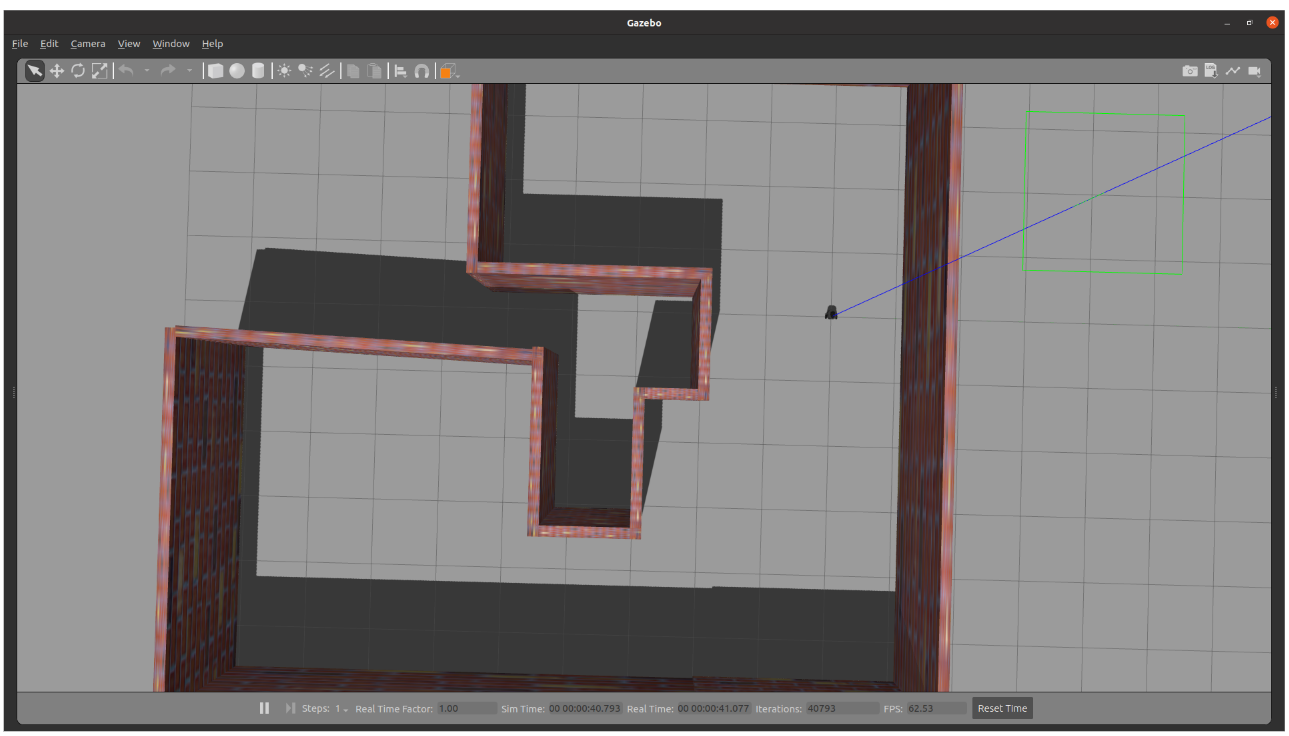

Figure 1: gazebo setup

1. **Linear Motion Accuracy Test:**
    - Turtlebot3 moved linearly along the x-axis with a velocity of 0.22 m/s, covering a distance of approximately 5 meters.
    - Both the Odom position and the filter position started at (0,0) before the motion.
    - The robot's trajectory during this experiment was a straight line along the x-axis.
    - The experiment was repeated for particle quantities of 1, 3, 50, 100, 250, and 500.
    - Data were collected for the RMSE between Odom and filter positions for each run, and an average of the entire run is taken at the end.
    - The same experiment was conducted for two variants of the optimal particle filter under similar conditions and the results were plotted.

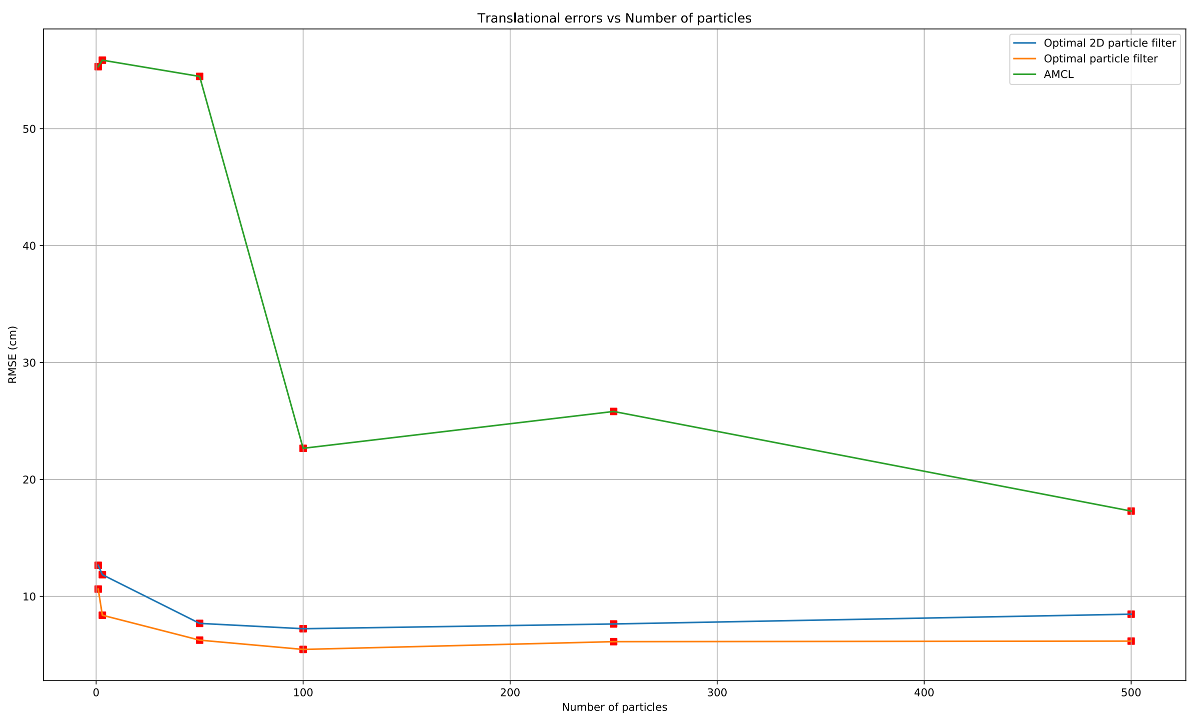

Figure 2: translational error vs number of particles

2. **Rotational Motion Accuracy Test:**

   - Turtlebot3 executed a full circular motion with a constant linear velocity of 0.22 m/s and an angular velocity of -0.5 rad/s.
   - Both Odom position and filter position started at (0,0) before the circular motion.
   - The robot's trajectory during this experiment was a circle.
   - The experiment was repeated for the same particle quantities (1, 3, 50, 100, 250, and 500).
   - Data was collected for the RMSE between Odom and filter positions for each run and an average of the entire run is taken at the end.
   - The same experiment was conducted for two variants of optimal particle filter under similar conditions and the results were plotted.

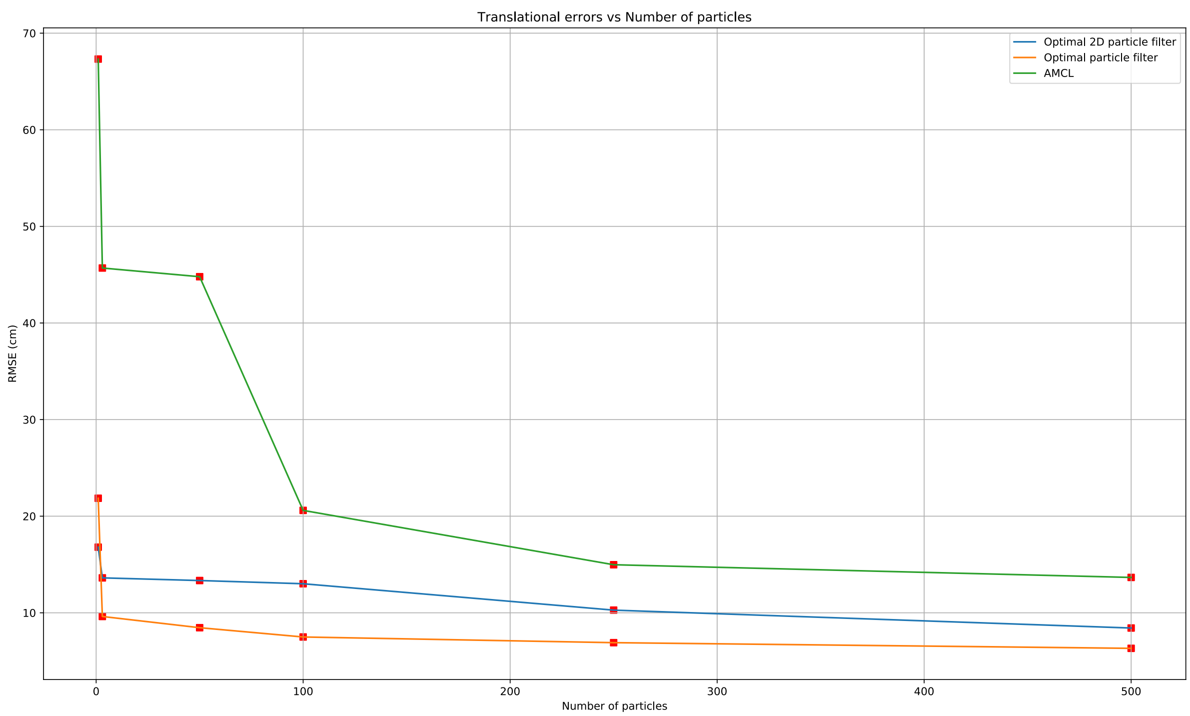

Figure 3: rotational error vs number of particles

3. **Random Motion Accuracy Test:**

- The Turtlebot3 is subjected to a random motion test with a mix of translational and rotational velocities. Velocities are dynamically determined by a Python code which has predefined velocities.
- Both Odom position and filter position started at (0,0) before the this motion.
- The robot's trajectory during this experiment was a mix of translation and rotation.
- The experiment was repeated for the same particle quantities (1, 3, 50, 100, 250, and 500).
- Data was collected for the RMSE between Odom and filter positions for each run and an average of the entire run is taken at the end.
- The same experiment was conducted for two variants of optimal particle filter under similar conditions and the results were plotted.

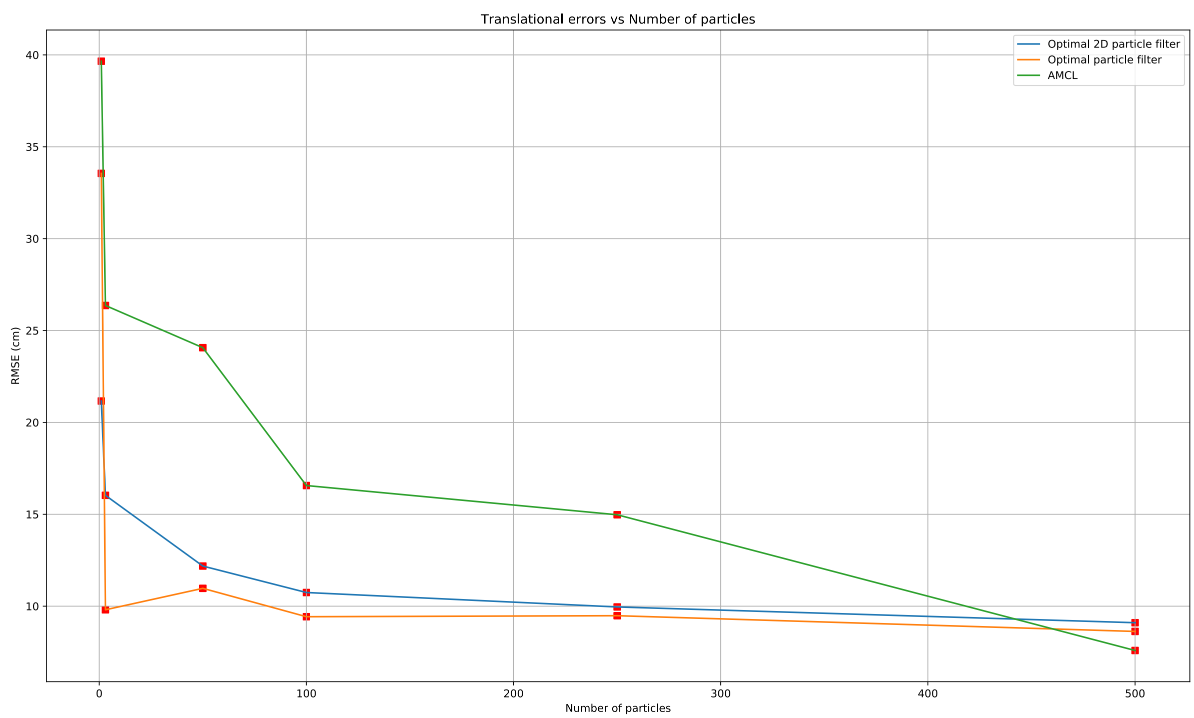

Figure 4: random motion error vs number of particles

In conclusion, the experiment showcased a consistent trend of significantly better accuracy in the optimal particle filter compared to AMCL. Even with fewer particles, such as 1 or 3, the optimal particle filter outperformed AMCL, giving better results compared to the 500 particles in AMCL. This shows the optimal particle filter is more efficient, resulting in more accurate localization.

2 Dynamic Adaptability of Particle Filter: Adaptive Response to Manual Displacements

In this experiments, With the same setup. I moved the Turtlebot3 manually, lifting the robot and moving along linear x and y axes, and rotational movement. This dynamic scenario aimed to simulate real-world scenarios where the robot encounters robot kidnap problem where the robot is picked and placed at a random position.

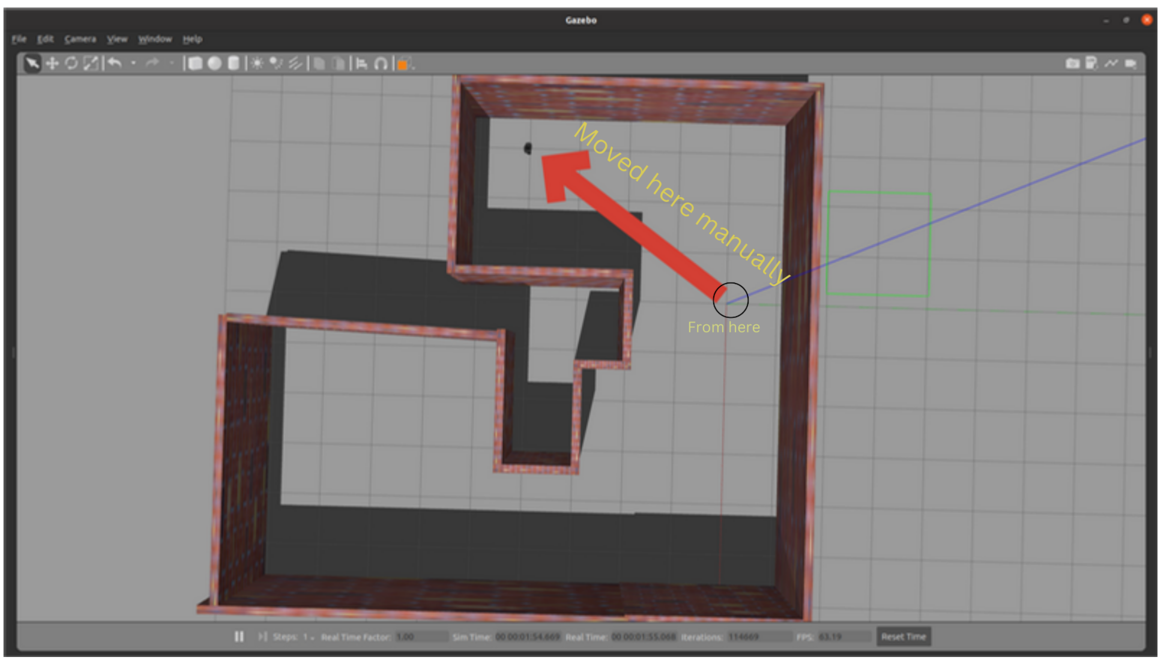

Figure 5: gazebo setup

1. **Linear Motion Accuracy Test:**
   - After manual displacements, an accuracy test for linear motion was performed
   - The same experiment as above was done in this experiment and plotted the results for both algorithms

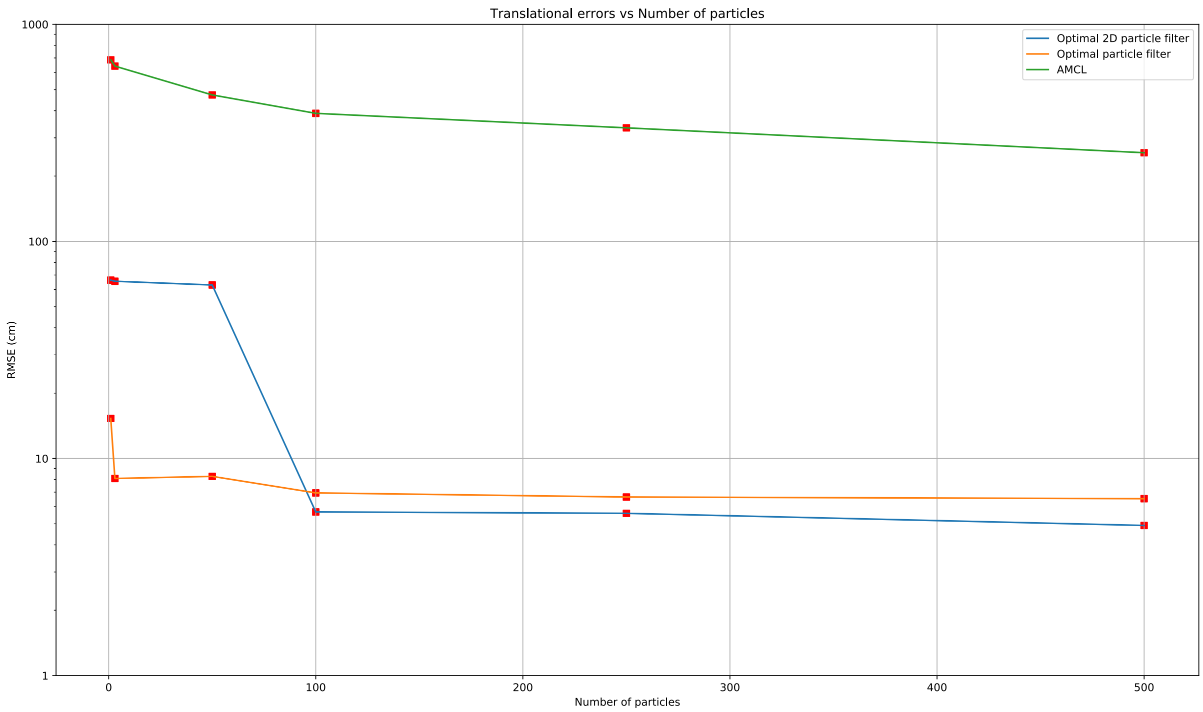

Figure 6: LOG graph of translational error vs number of particles

2. **Rotational Motion Accuracy Test:**
   - After manual displacements, an accuracy test for rotational motion was performed
   - The same experiment as above was done in this experiment and plotted the results for both algorithms

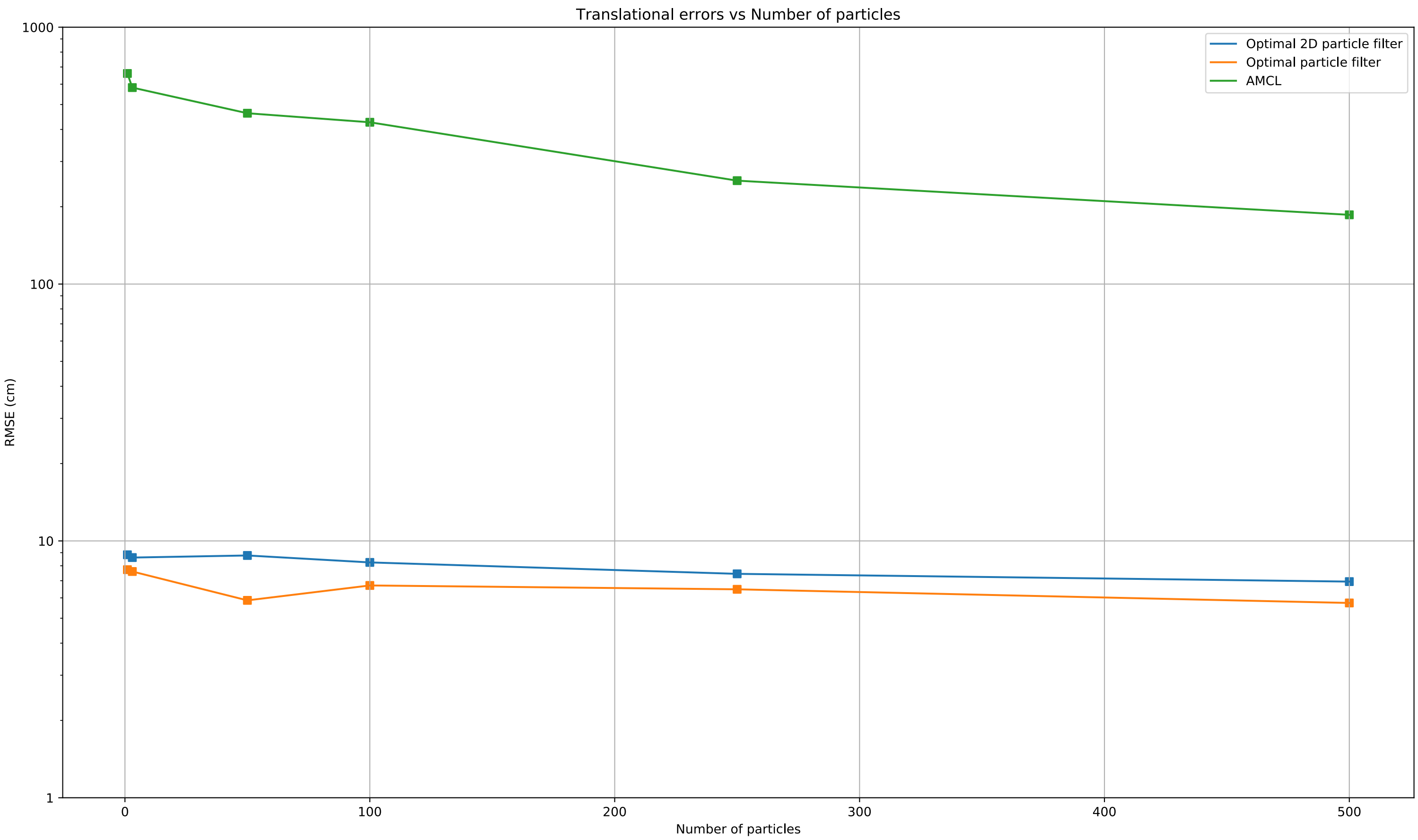

Figure 7: LOG graph of rotational error vs number of particles

3. **Random Motion Accuracy Test:**
   - After manual displacements, an accuracy test for Random motion was performed
   - The same experiment as above was done in this experiment and plotted the results for both algorithms

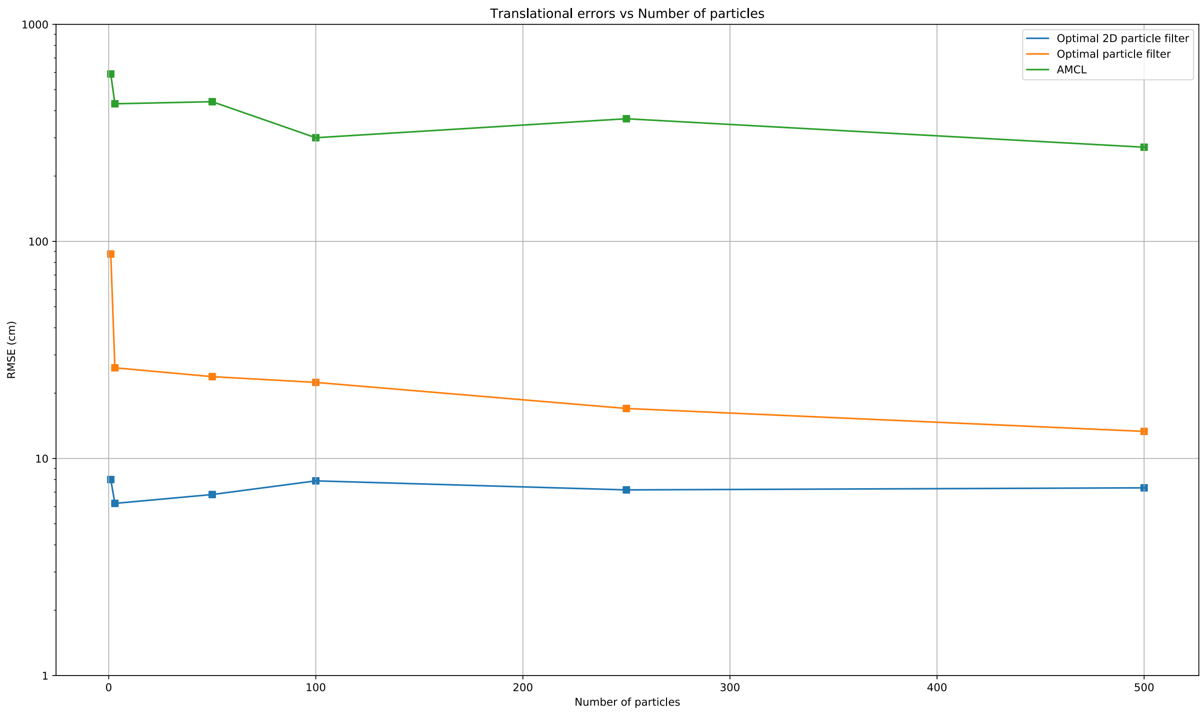

Figure 8: LOG graph of random motion error vs number of particles

4. **Comprehensive Analysis of Particle Filter Performance post Robot Kidnap Scenario:**

    - Conducted extensive analysis after the robot kidnap scenario, plotting individual error values for particles 1, 3, 50, 100, 250, and 500 in each run for both AMCL and Optimal Filter without computing averages.
    - This comprehensive examination provides a detailed insight into the performance of both algorithms across varying particle quantities and how they recover after the robot is placed somewhere else.
    - The plotted results highlight the Optimal Particle Filter's effective recovery post the robot kidnap scenario, in contrast to AMCL, which struggles to regain accurate localization.

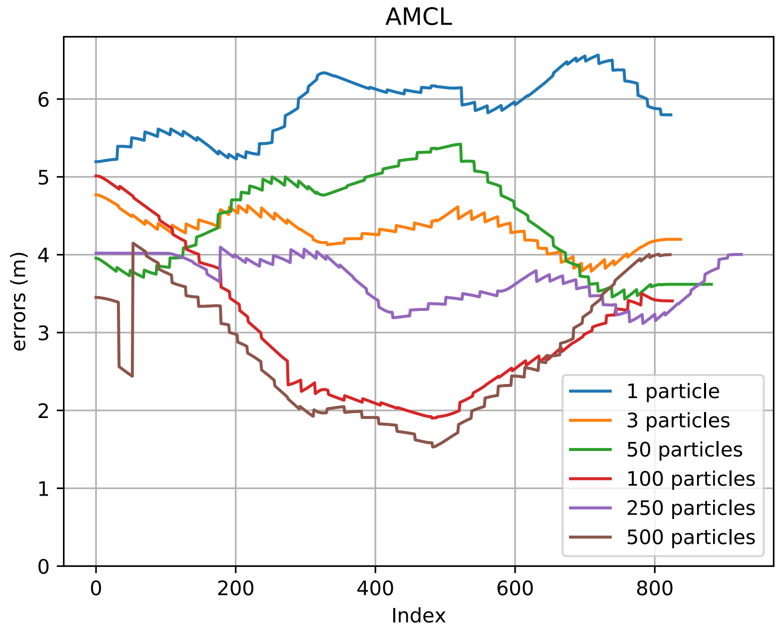

Figure 9: AMCL Errors for Different Particle Quantities

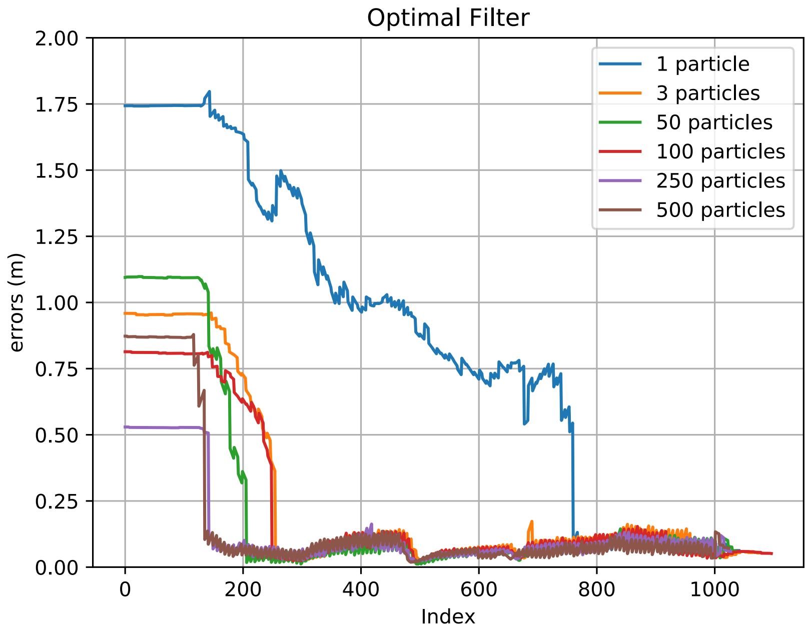

Figure 10: Optimal Filter Errors for Different Particle Quantities

In conclusion, this experiment too showcased a consistent trend of significantly better accuracy in the optimal particle filter compared to AMCL. When faced with a challenging scenario of robot kidnap, where the robot has lifted and placed in different positions, the AMCL algorithm struggled to recover effectively, even with higher particle count of 500. In contrast, the optimal particle filter demonstrated better results, adaptability and efficient localization during the robot kidnap scenario, even with a reduced number of particles. This again shows optimal particle filter is better at handling complex and dynamic situations, highlighting more efficient, resulting in more accurate localization

3 Accuracy and Robustness

The experiment setup for **Linear Motion**, **Rotational Motion**, **Random Motion**, and **Kidnapped Pose** tests is the same. The only addition is that each experiment was conducted for **5 runs** for each particle quantity (1, 3, 50, 100, 250, and 500) and for both AMCL and Optimal 2D algorithms. The accuracy and robustness were evaluated and visualized using **line graphs** (average RMSE). For each particle quantity and for each run, the **average of the entire run** is plotted in **light color**, and the **average of 5 runs** is plotted in **dark color** to provide a clearer distinction between individual and overall performance. **Box plots** were also used for robustness (spread of RMSE across runs).

In the **Kidnapped Pose** test, only **random motion** was performed, not all three motion types (linear, rotational, random).

1. **Random Wrong Initial Pose Test**

    - In this experiment, the robot started at (0,0), but the initial pose was altered randomly within a 1-meter radius of the origin using RViz 2D Pose Estimation.
    - This was performed under **random motion** conditions to evaluate how well the algorithms could handle inaccurate starting positions.
    - The experiment was conducted for particle quantities of 1, 3, 50, 100, 250, and 500, with **5 runs** for each.
    - Results were evaluated for both **accuracy** and **robustness**, with **line graphs** and **box plots** used for visualization of RMSE and spread.

A table has been added to show the time taken to localize (in seconds) in the Kidnapped Pose test and the Random Wrong Initial Pose test for both AMCL and Optimal algorithms. The table includes data for each particle quantity (1, 3, 50, 100, 250, and 500), with 5 runs conducted for each configuration. The time values are given in seconds, and NaN represents a case where the robot never localized successfully within the given complete run.

No. of particles	**AMCL**					**Optimal**				
	1	2	3	4	5	1	2	3	4	5
1	NaN	NaN	NaN	NaN	NaN	0	0	0	0	0
3	NaN	NaN	NaN	NaN	NaN	0	0	0	0	0
50	NaN	NaN	NaN	NaN	1.867	0	0	0	0	0
100	NaN	NaN	NaN	NaN	NaN	0	0	0	0	15.338
250	NaN	NaN	NaN	2.066	NaN	0	0	0	6.8	0
500	NaN	NaN	NaN	43.034	2.733	0	0	0	0	0.02

Table 1: AMCL and Optimal Time take to Localize (seconds) - Robot Kidnap Test

No. of particles	**AMCL**					**Optimal**				
	1	2	3	4	5	1	2	3	4	5
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN	1.173	18.816	0	24.174	8.386
50	NaN	NaN	NaN	NaN	NaN	6.169	10.732	0	2.336	10.987
100	NaN	3.035	5.101	3.201	NaN	4.71	1.597	22.367	23.267	5.474
250	9,7	13.667	2.17	1.303	1.434	27.654	9.845	18.463	29.124	0
500	NaN	7.4	11.166	4.9	4	0.5	5.933	32	30.834	2.07

Table 2: AMCL and Optimal Time take to Localize (seconds) - Random Wrong Initial Pose Test

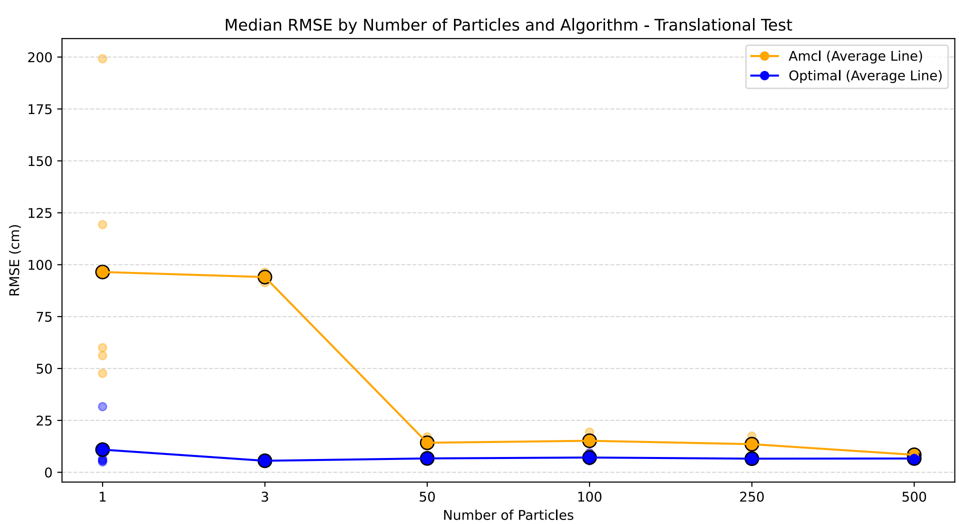

Figure 11: Median RMSE - Translational Test

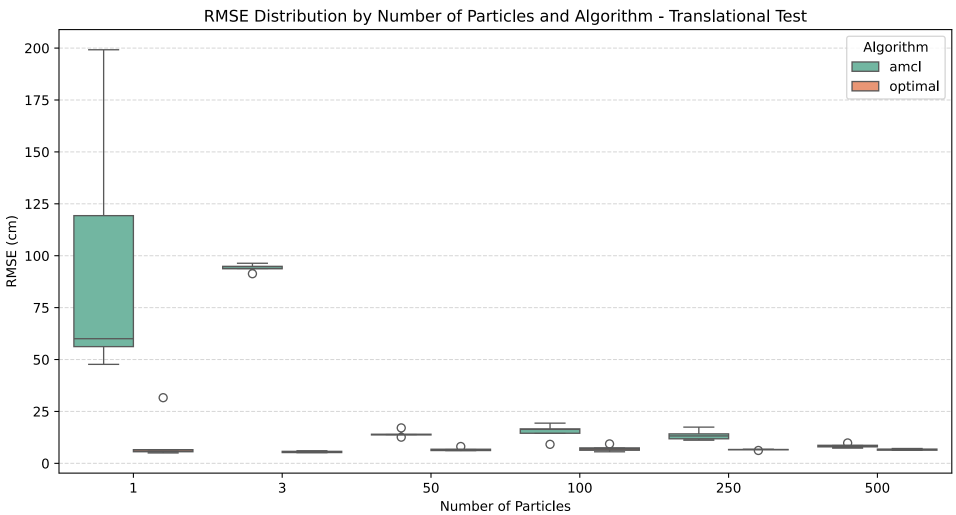

Figure 12: Box Plot - Translational Test

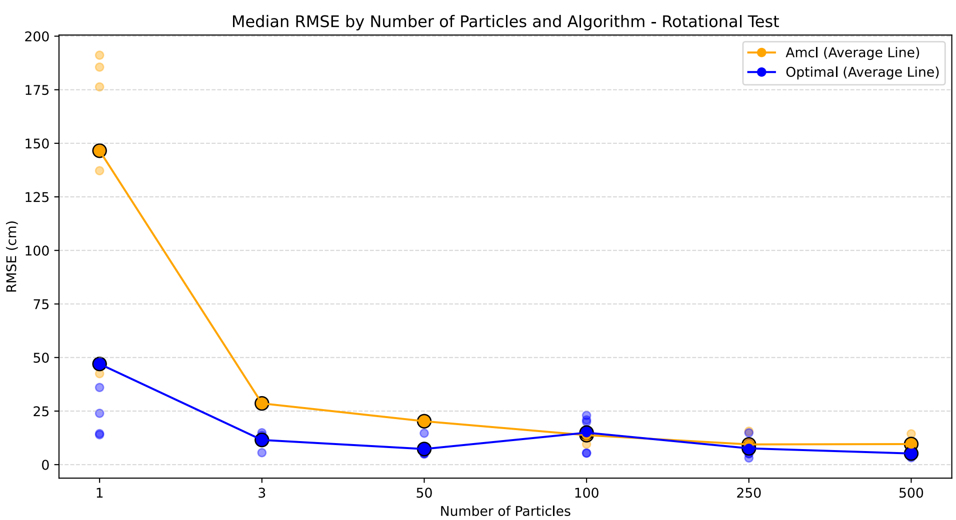

Figure 13: Median RMSE - Rotational Test

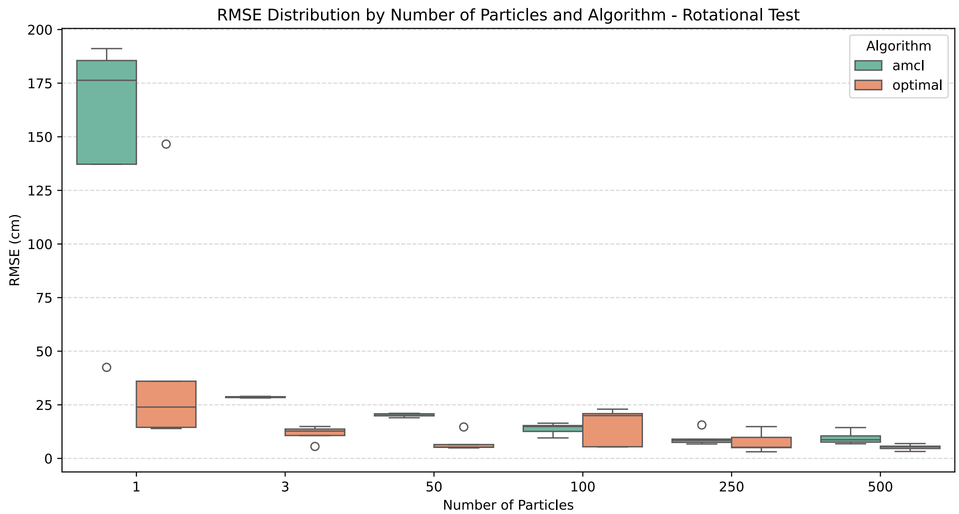

Figure 14: Box Plot - Rotational Test

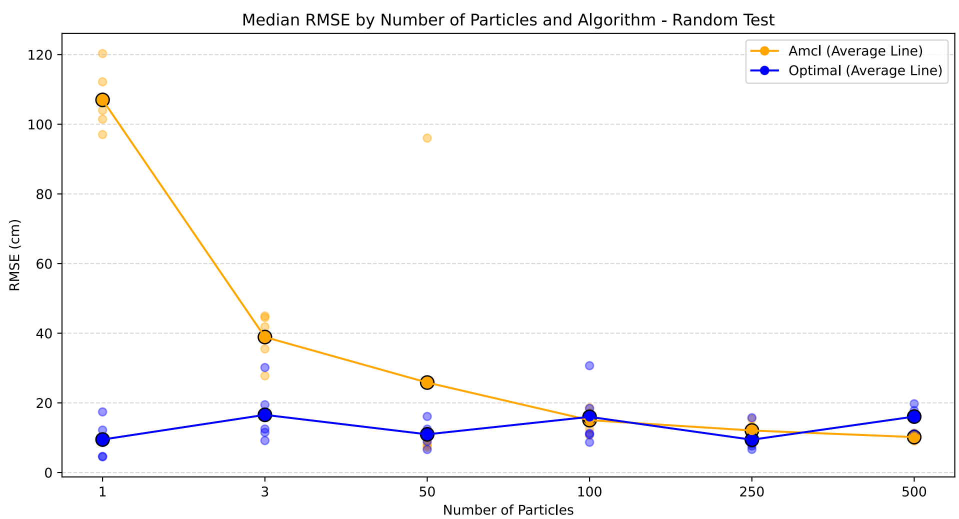

Figure 15: Median RMSE - Random Test

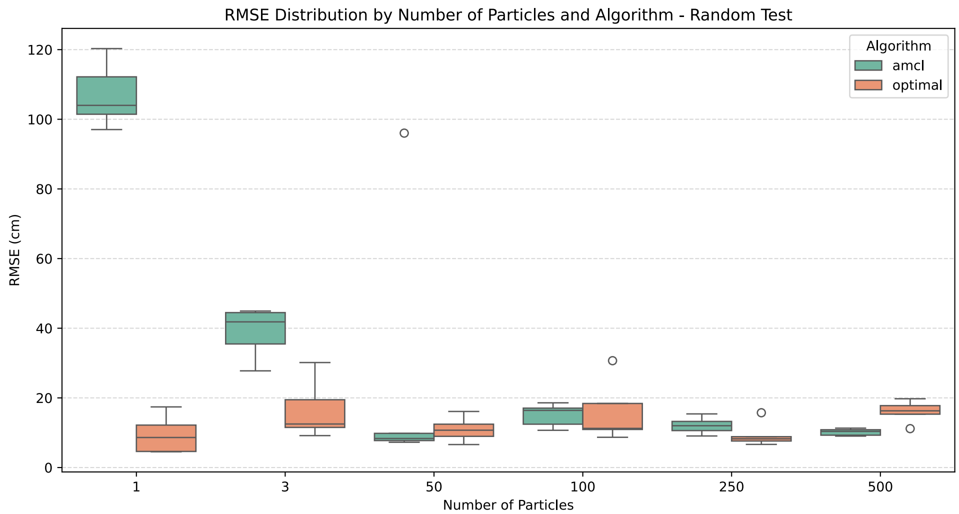

Figure 16: Box Plot - Random Test

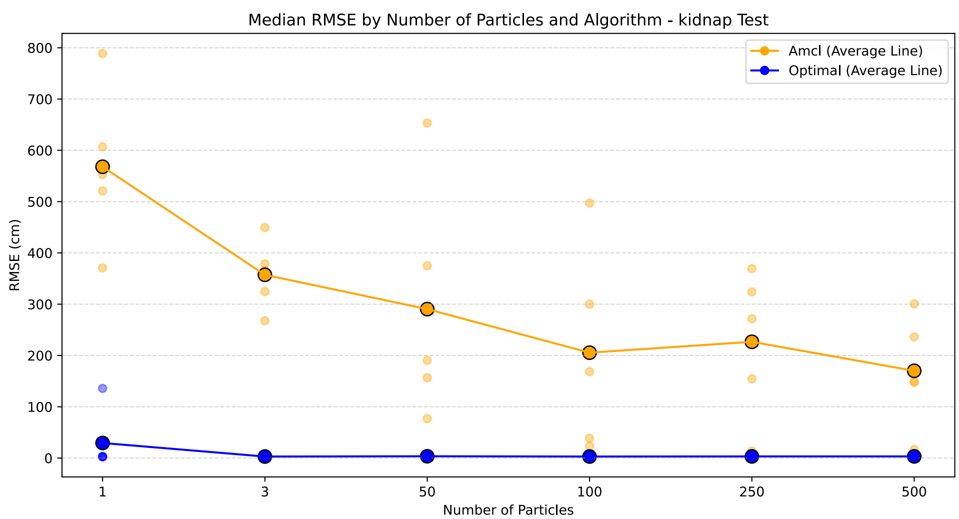

Figure 17: Median RMSE - Kidnap Test

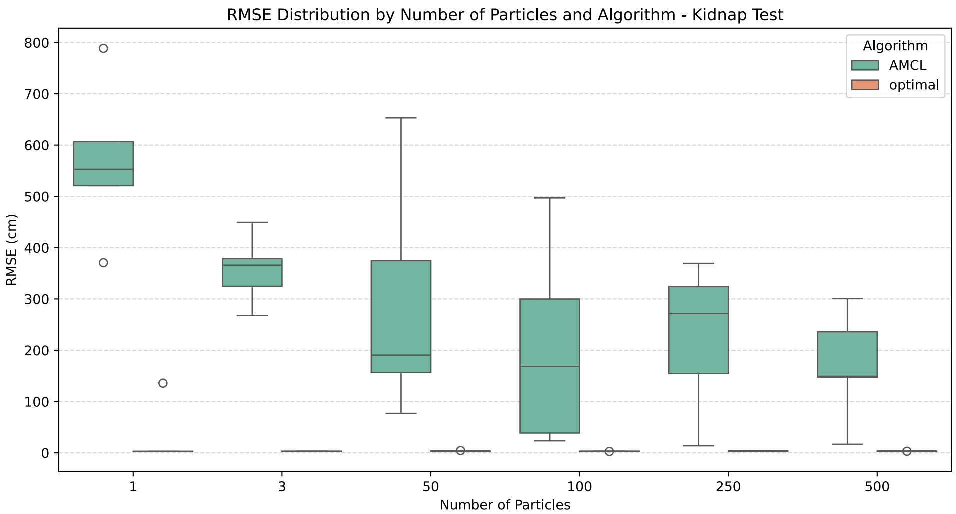

Figure 18: Box Plot - Kidnap Test

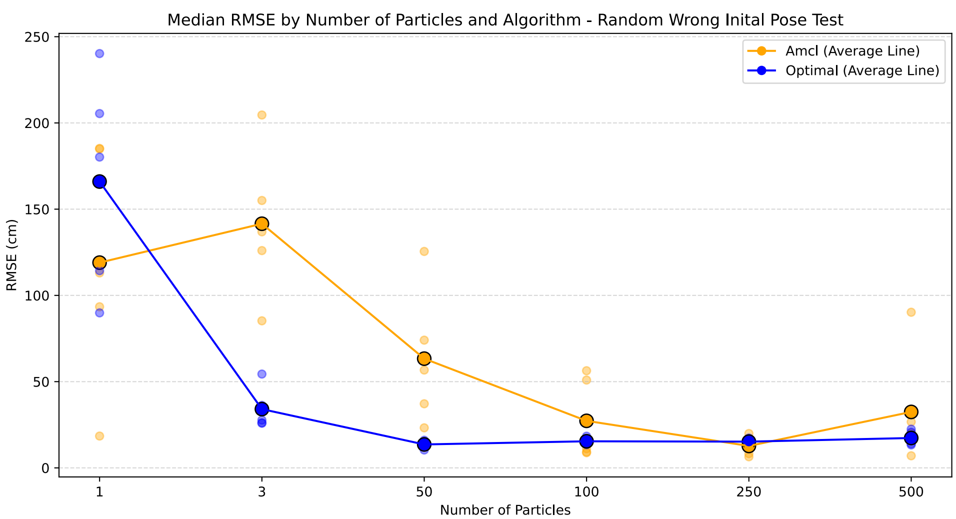

Figure 19: Median RMSE - Random Wrong Initial Pose Test

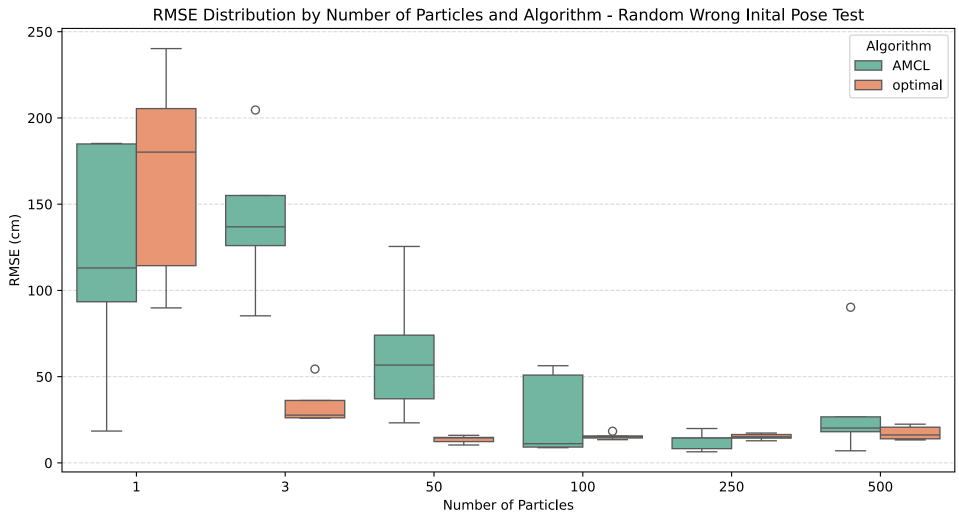

Figure 20: Box Plot -Random Wrong Initial Pose Test

4 Robustness Testing in Various Environments

In this section, I conducted robustness testing of the particle filter algorithms in three distinct environments to evaluate their localization performance. Robustness to ambiguity was evaluated with simulated data from two challenging environments: a set of three identical square rooms of 10m length, and a large circular room of 20m diameter. The results demonstrated that the optimal particle filter could localize accurately in two out of the three environments but struggled in the circular room.

A successful run is defined as one in which, after the complete run, the pose of the robot is within a 0.5 meter radius of the true pose.

Robustness to map discrepancies was also evaluated. A simulation test was carried out in a simple environment consisting of a planar corridor and a door (identical rooms). The door is left open during mapping and closed during localization.

In circular room a large unmapped partition occludes the exit way during localization, such that there is a ambiguity in the generated LiDAR scans. Ideally, the localization system must capture this uncertainty through appropriate dispersion of particles such that the robot is able to recover its exact pose as it reaches the exit way.

An accuracy test was also conducted, with 5 runs for each particle quantity (1, 3, 50, 100, 250, and 500) for both AMCL and Optimal 2D algorithms. The accuracy was evaluated based on the RMSE between the estimated and true positions. Additionally, the time complexity of the Optimal particle filter algorithm was calculated, using the system specifications of an AMD Ryzen 9 processor, to assess the computational efficiency of the algorithm under different particle quantities.

1. **Three Identical Rooms:**

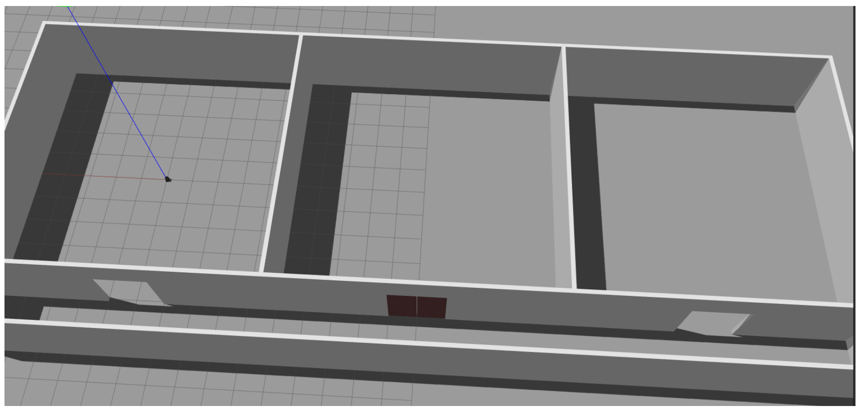

Figure 21: Three identical square rooms of 10m length environment

- This environment consisted of three identical rooms with closed doors, posing a challenge for the particle filter due to the lack of unique features within each room.
- The following plot shows the line graphs for RMSE and a box plot in the identical rooms environment.
- A table has been added to show the successful runs for each particle quantity (1, 3, 50, 100, 250, and 500) for both AMCL and Optimal algorithms, where each particle quantity was tested over 5 runs. In this table, a 0 indicates that the robot was not localized by the end of the run, and a 1 indicates that the robot was successfully localized, with its pose within a 0.5-meter radius of the true pose at the end of the run.

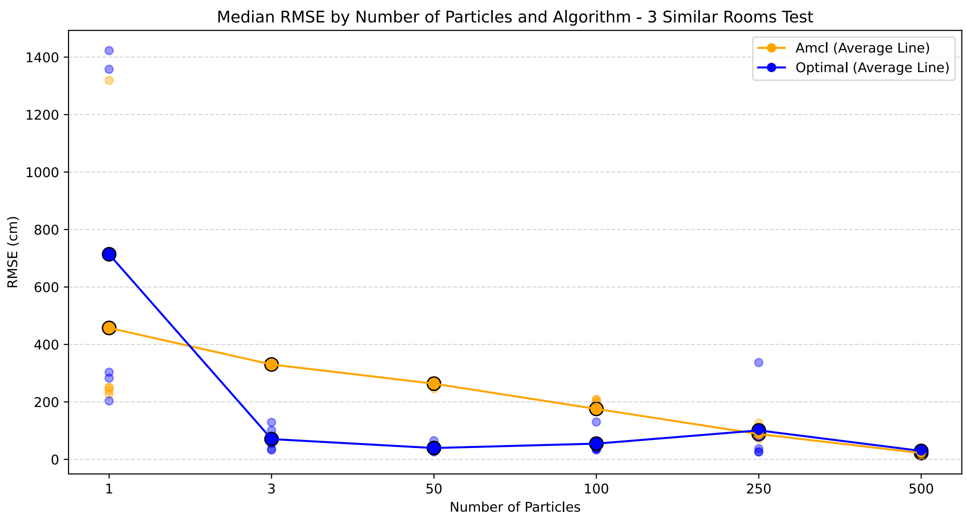

Figure 22: Median RMSE - Similar Rooms Test

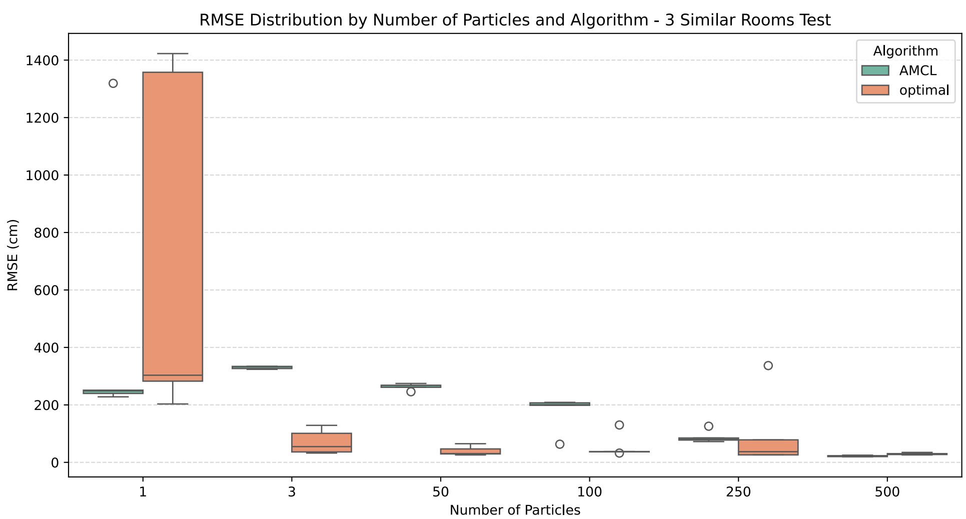

Figure 23: Box Plot - Similar Rooms Test

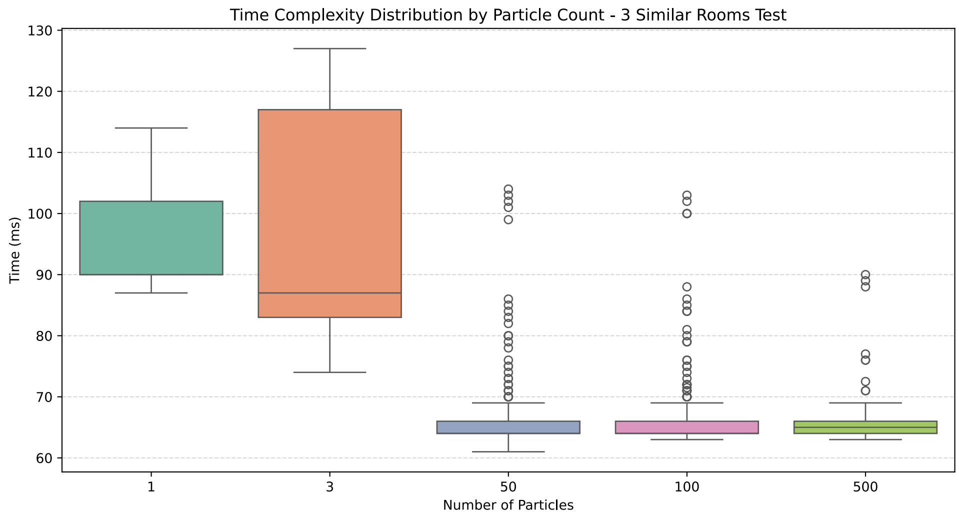

Figure 24: Time Complexity Distribution by Particle Count - Similar Rooms Test

Number of particles	**AMCL**					**Optimal**				
	1	2	3	4	5	1	2	3	4	5
1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	1	1	1	1
50	0	0	0	0	0	1	1	1	1	1
100	0	0	0	0	0	1	1	1	1	1
250	1	1	1	1	1	1	1	1	1	1
500	1	1	1	1	1	1	1	1	1	1

Table 3: AMCL and Optimal Successful Runs

2. **Circular Room:**

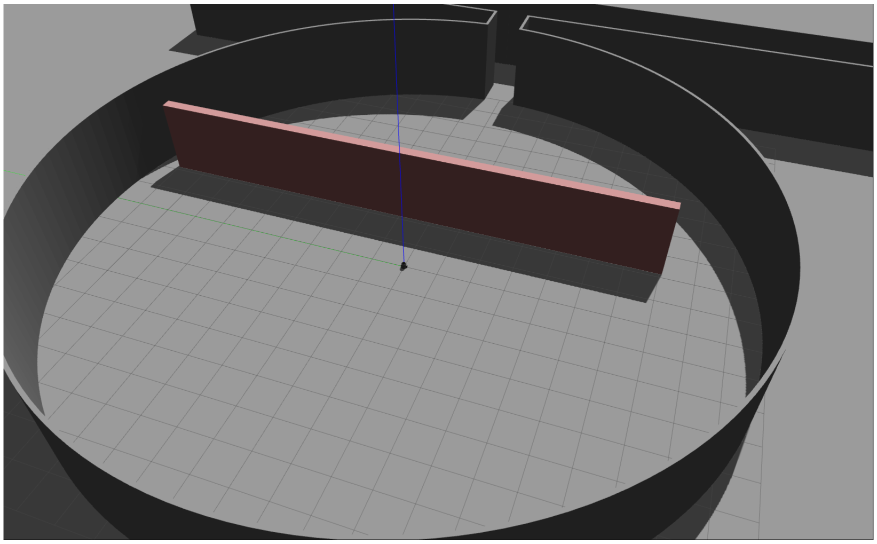

Figure 25: Circular room of 20m diameter environment

- This environment consisted of 20m diamenter circular room, posing a challenge for the particle filter due to the lack of unique features.
- The following plot shows the line graphs for RMSE and a box plot in the environment of the circular room.

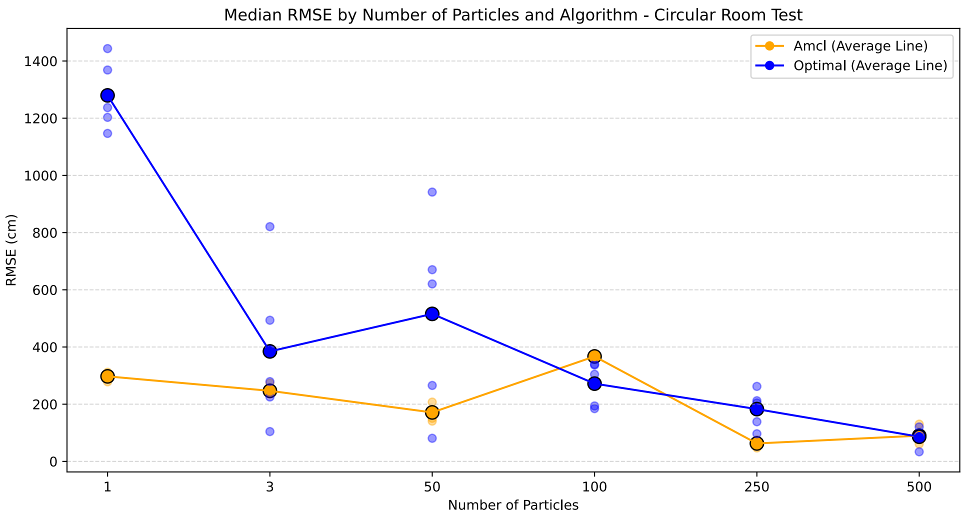

Figure 26: Median RMSE - Circular Rooms Test

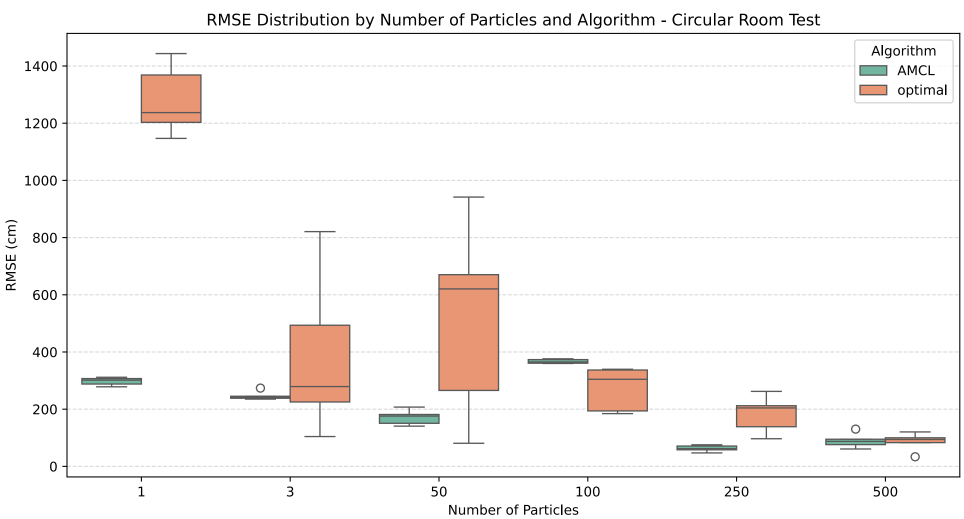

Figure 27: Box Plot - Circular Rooms Test

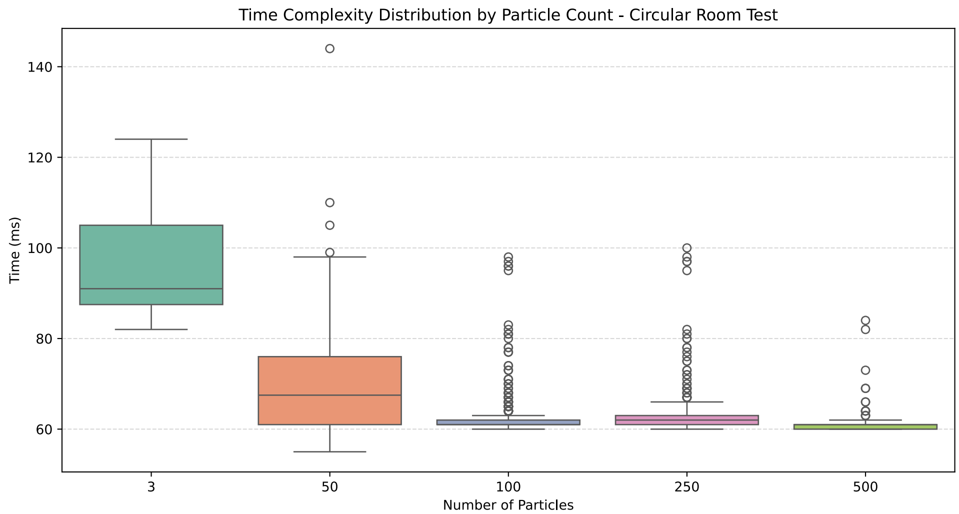

Figure 28: Time Complexity Distribution by Particle Count - Circular Room Test

Number of particles	**AMCL**					**Optimal**				
	1	2	3	4	5	1	2	3	4	5
1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	1	0	0	0	0
100	0	0	0	0	0	0	1	0	0	0
250	0	0	0	0	0	0	0	1	1	1
500	0	0	0	0	0	1	1	1	0	0

Table 4: AMCL and Optimal Successful Runs