Intelligent Autonomous System Project3 Report

Gilberto E. Ruiz (ger83)

April 16, 2024

Introduction:

In the initial part of this project, as detailed in the writeup checkup, I was tasked to construct an indoor environment map using wheel encoders and range sensor data. By integrating the information from the mobile robot's wheel odometry with that from a 2D laser range scanner (LIDAR), I successfully created a preliminary 2D occupancy grid map highlighting the walls and obstacles within the given space. The next phase of the project was to refine the map through applying a Simultaneous Localization and Mapping algorithm (SLAM). The entire project was created in an iPython notebook titled "SLAM.ipynb" and was separated into 12 cells. These cells address the completion of the mapping project for the two assigned maps (numbers 20 and 23), as well as the test data provided by the professor (numbers 22 and 24). The main objective of this project was to refine the environment mapping by progressing from a single-particle model to a full-fledged SLAM algorithm involving multiple particles, to improve map accuracy as the robot traveled through the environment.

Problem Formulation:

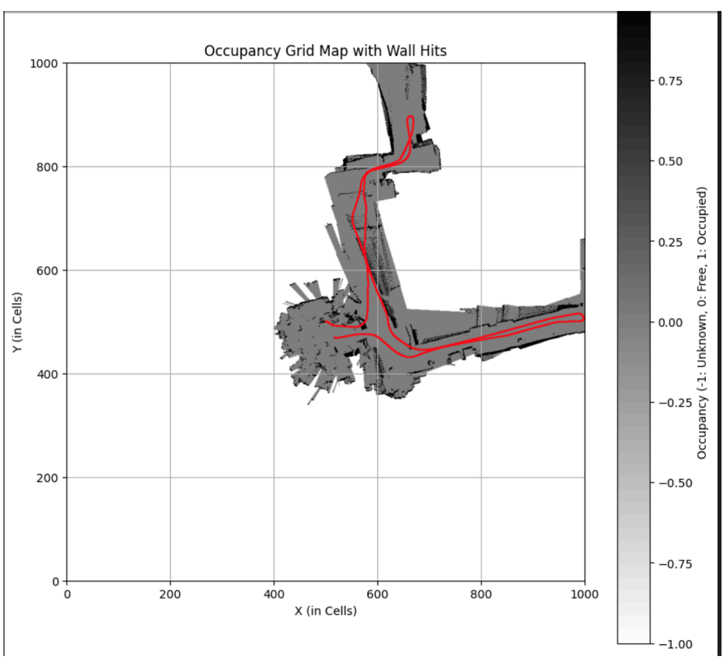
The SLAM (Simultaneous Localization and Mapping) problem requires the robot to construct a map of an unknown environment while concurrently determining its location within that map. This project's formulation involves interpreting data from wheel encoders and a 2D LIDAR sensor to create a detailed occupancy grid representing the environment. The occupancy grid identifies areas as occupied, free, or unknown based on the sensor inputs. The challenge is compounded by the noise inherent in sensor data and the robot's motion, requiring the application of probabilistic methods and particle filters to estimate the robot's pose and refine the map concurrently. The objective is to generate a map with clear delineation of walls and obstacles while maintaining an accurate trajectory of the robot's path, thus enabling it to navigate autonomously and efficiently.

Technical Approach and Results:

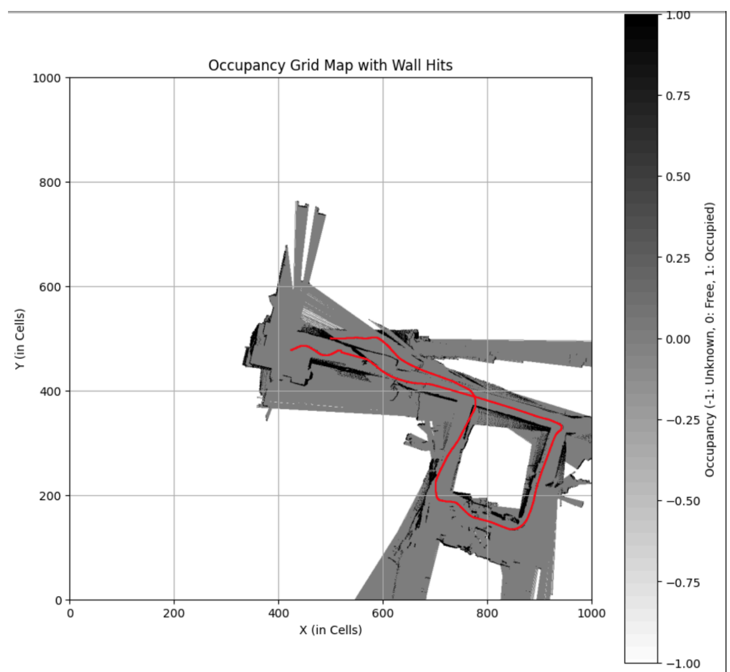
The approach to SLAM began with extracting odometry data to establish the robot's initial trajectory. Using LiDAR data brought the layout of the indoor space into view, enabling a distinction between the open spaces and occupied areas (walls). A particle filter was introduced to my SLAM algorithm due to its robustness in estimating the probable distribution of the robot's path and map concurrently. Each particle represented a hypothetical state of the robot which navigated through the environment based on a motion model, driven by odometry data. The particles were continually assessed and weighted according to the LiDAR feedback, evaluating the compatibility of the data with the predicted states. The particle with the maximal weight typically meant that it had the most accurate estimate of the robot's current state and was thus utilized to update the occupancy grid. The outcome was a comprehensive map delineating the environment with an overlaid robot trajectory, indicative of the enhanced precision achieved though SLAM. Fine-tuning the algorithm parameters was crucial for the balance between the sensitivity of the model and the accuracy of the mapping. The particle weights' update and the resampling process played critical roles in maintaining a computational focus on the most probable states. Below are all the resulting plots I got of the occupancy grid map with one particle on all 4 maps (train data and test data):

Train Data:

Map 20

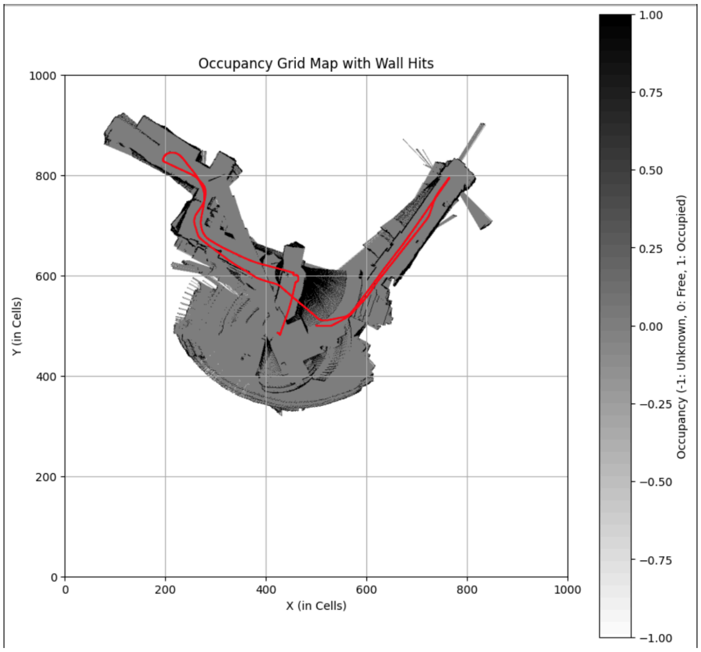


Map 23

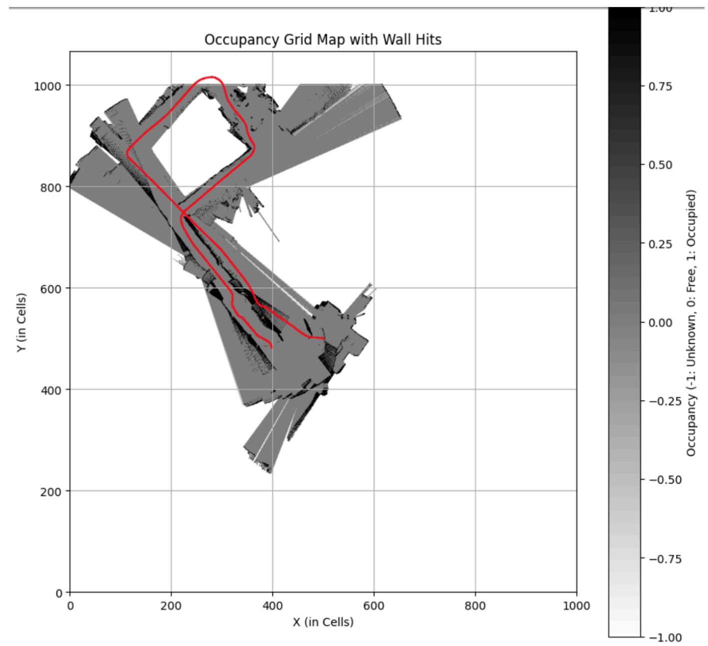


Test Data:

Map 22



Map 24



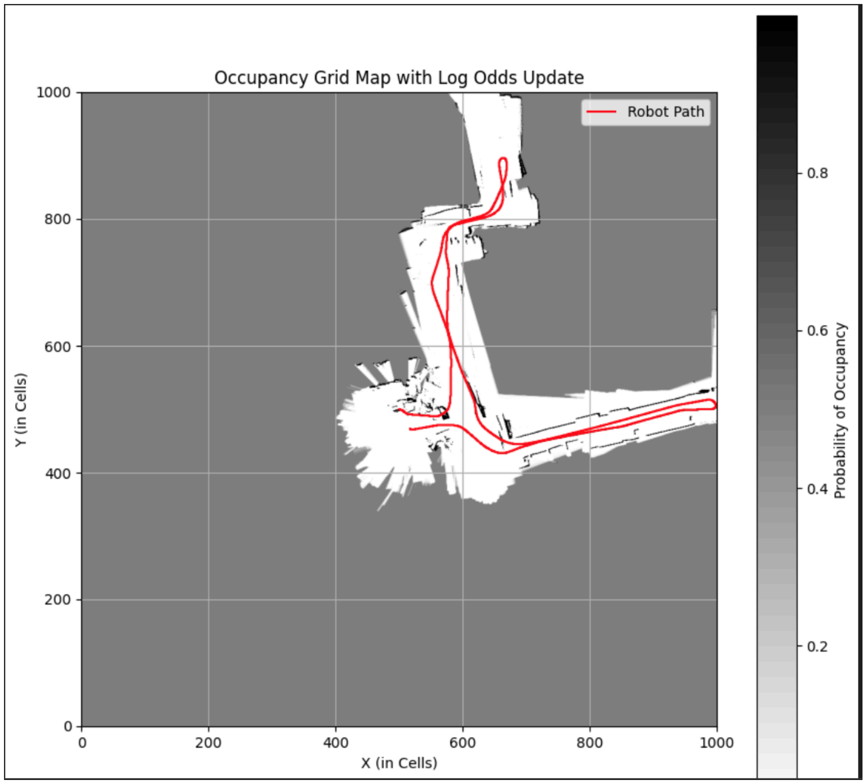
Results and Discussion:

The project's task to implement SLAM resulted in an enhanced mapping system. The integration of wheel odometry and LiDAR data via particle filter algorithm was central to improving accuracy and robot trajectory. This multi-faceted approach produced a more coherent environmental layout, highlighting free spaces and obstacles more distinctly.

Challenges Faces:

Throughout working on this project, I've encountered several challenges:

1. Computational Intensity and Duration: The switch from a straightforward occupancy grid mapping to a log odds update significantly increased computational demands. Specifically, the cell responsible for updating the occupancy grid map utilizing log odds took upwards of 37 minutes to execute, spotlighting the need for algorithm optimization to reduce processing time without compromising on the quality of the map. This was the resulting plot which looked very similar to when I upgraded the occupancy map without log odds:



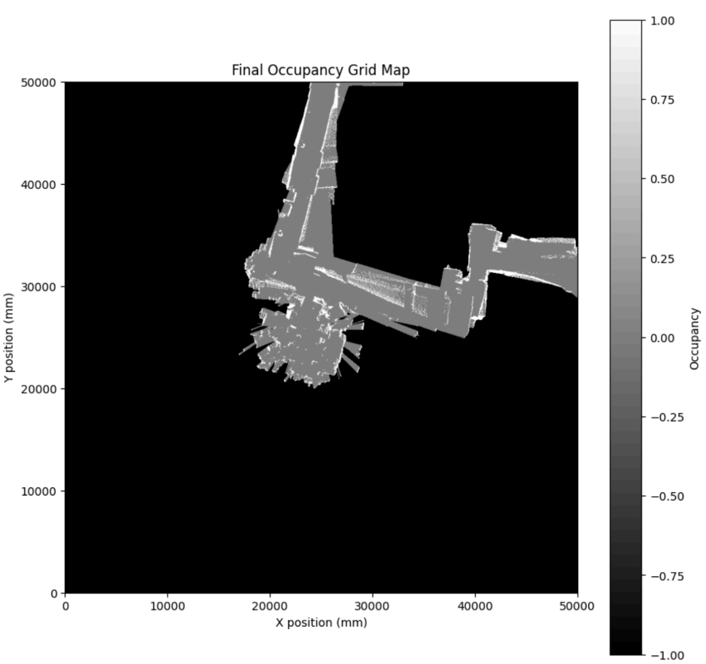
2. Sensor Noise and Data Inconsistencies: Encoded data and LiDAR readings are susceptible to noise and discrepancies. Such inconsistencies posed considerable hurdles in achieving an accurate representation of the environment. To address this, noise filtering techniques and sensor fusion approaches were explored, seeking a balance between real-time processing and reliable mapping.
3. Map Clarity: Despite the advanced methods deployed, the resultant maps occasionally lacked clarity in defining walls and obstacles, primarily due to sensor inaccuracies and the limitations of the particle filter in dealing with sparse and ambiguous data. Fine-tuning of the parameters within the SLAM algorithm was crucial to mitigate this, requiring multiple iterations to determine an optimal balance.
4. Parameter Optimization: The complexity of the SLAM algorithm necessitated meticulous parameter adjustments. Determining the appropriate number of particles, the threshold for resampling, and the log odds' increment and decrement values was a process of trial and error. Too few particles or improper noise settings led to insufficient exploration of the state space, while too many increased the computational burden.

5. Algorithmic Refinement: A significant insight gained was the trade-off between the precision of the grid's resolution and the computational resources required. Initially, larger cells were used to expedite debug and testing phases. Later, finer grids demanded higher particle counts to maintain resolution, highlighting the need for a strategic approach to grid discretization.
6. Data Management: The final stages of the project introduced test datasets to evaluate the SLAM algorithm's robustness in unknown environments. Processing these datasets uncovered additional complexities, emphasizing the need for versatile and adaptive algorithms capable of generalizing across different scenarios.

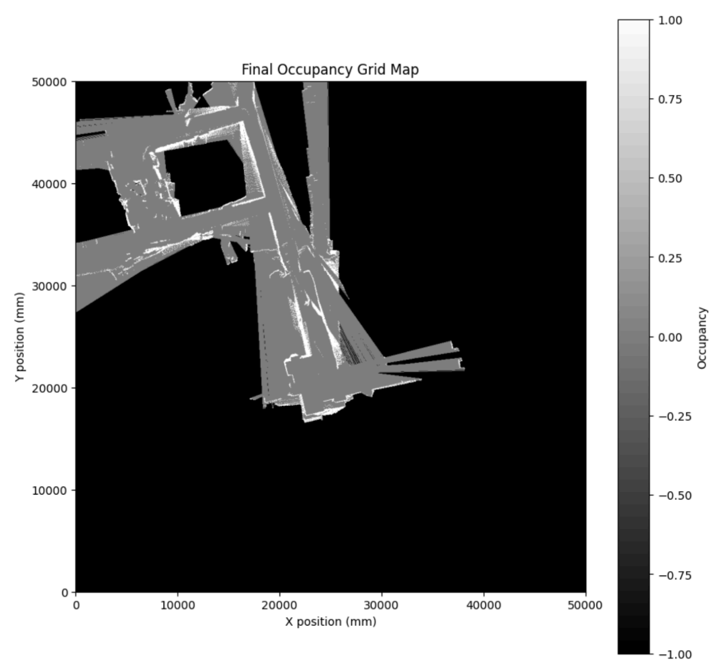
And these were the results of applying my full SLAM algorithm on all 4 maps provided (train and test data)

Train Data:

Map 20

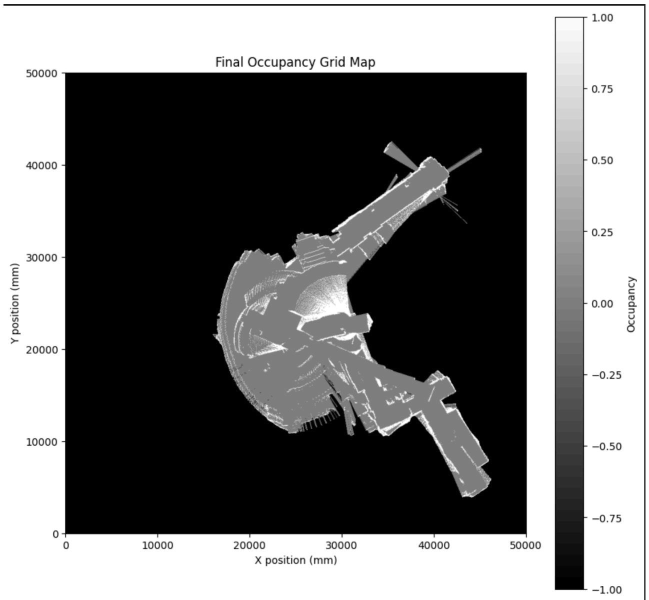


Map 23

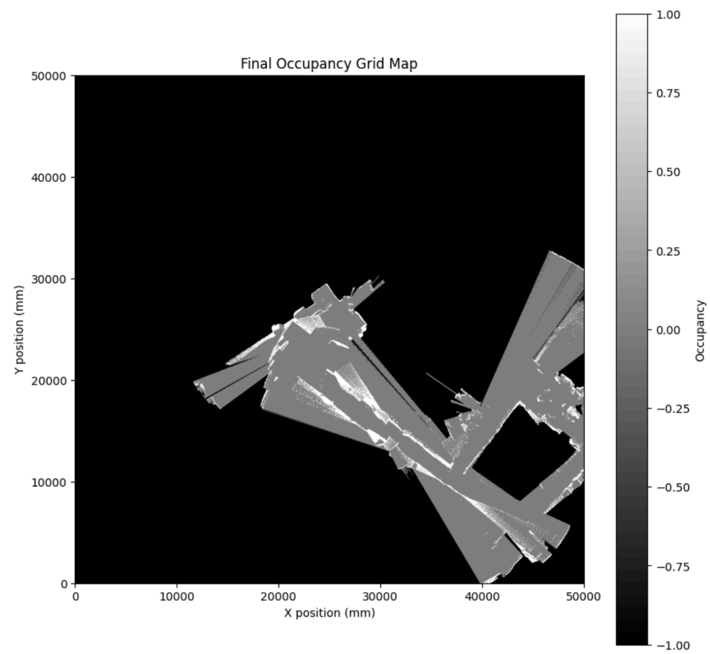


Test Data:

Map 22



Map 24



Conclusion:

The SLAM project demonstrated that creating an accurate map and maintaining reliable robot localization is as challenging as it is critical. Throughout the project, we refined the balance between computational efficiency and the fidelity of the resulting maps. While the implementation showcased promising results, it also brought to light the delicate nature of parameter tuning and the impact of sensor noise on overall performance. The primary takeaway from this project is the necessity for constant iteration and adaptation. The implementation of SLAM is a dynamic process that demands a deep understanding of both the theoretical underpinnings and practical considerations of robotic systems. Moving forward, the insights gained from this subject can serve as a foundation for further exploration into more sophisticated models and algorithms, potentially incorporating machine learning to predict and correct sensor deviations.