ME 598

Introduction to Robotics

Fall 2023

Final Project

Group 4

December 18, 2023

"I pledge my Honor that I have abided by the Stevens Honor System."

Hunter Schmidt

"I pledge that I have abided by the Graduate Student Code of Academic Integrity."

Muthathal Chinniah, Sophie Dewil, Sathya Prasad Reddy Patlolla, Vignesh Pagilla

This report has been prepared by:

1. Lab Leader: Hunter Schmidt
2. Muthathal Chinniah
3. Sophie Dewil
4. Sathya Prasad Reddy Patlolla
5. Vignesh Pagilla

Abstract:

In this lab we simulated a Mobile robot in MATLAB using Robotics Playground App in MATLAB. The robot was simulated in an environment with two objects and walls on all four sides as obstacles. These two objects can be a unique and distinct combination of any two colors in red, blue and green. The robot's environment also had three colored zones red, blue and green. The robot had sensors in it to measure each colored object's distance and angles and also front and rear range sensors to detect obstacles and encoders for odometry readings. The task was that the robot should move appropriately from a known position to push both the objects to their respective colored zone. All five team members worked together in the lab. After trying different methods for implementing the task like MATLAB function, SIMULINK model and stateflow model, the best method was chosen for the final report and video. Abstract and introduction of the report was done by Muthathal Chinniah. Procedure of the report was done by Sathya Prasad Reddy Patlolla. Results were documented by Hunter Schmidt. Discussion was reported by Sophie Dewil. Feedback part of the report was done by Vignesh Pagilla. Conclusion was done by Hunter Schmidt and Vignesh Pagilla.

Introduction:

A mobile robot is a machine that can move from one place to another without or with very minimal human assistance. That implies that the robot has to know what is happening around it. In other words, the robot should have the ability to perceive its environment, localize itself in that environment and the ability to move to the required position without any collision. Considering a real time mobility task, if a person wants to travel from Stevens Institute Babbio Center to the Hoboken Path station, the person should have a valid means to travel like a vehicle or the person should walk and be able to perceive the environment to reach the station without hitting anyone or anything. Equating this to the mobile robot, the path station is the robot's goal position, the vehicle is robot's motor driver and the eyes of the person to perceive the environment is the robot's sensors. The working of mobile robots was better understood while working on this project.

Procedure:

1. We began by utilizing xPose and yPose readings to guide the robot to the center of the arena.
2. Then Pivoted the robot to the right using distance and angle readings from the object detector to align with the object.
3. Made the robot move towards the object until the respective color's distance reduces to approximately 0.4, securing it in the claw or plough attachment.
4. Now, Resumed the pivoting to align with the color zone for the present object using tPose readings.

5. Further, Separated tPose readings for left and right initialization guide movement trajectories. Pushed the object into its zone until the front distance sensor reading reaches a threshold value of 0.7.
6. Executed a reversing maneuver and pivot to approach the right object. Once the right object is in its designated zone, repeat the reversing maneuver and navigate back home.
7. Work distribution involved assigning specific color object-pushing tasks to each team member.
8. This approach ensured equal participation and streamlined execution of tasks, enhancing overall efficiency.
9. Conducted rigorous testing of the stateflow chart with different color scenarios.
10. This testing phase validated the adaptability of the algorithm to diverse color configurations.
11. Following the algorithm implementation, comprehensive runtime calculations were performed for each process with the stateflow model shown in Fig 5.1.
12. This analysis provided valuable insights into the efficiency and computational costs associated with the developed algorithm, contributing to process optimization.

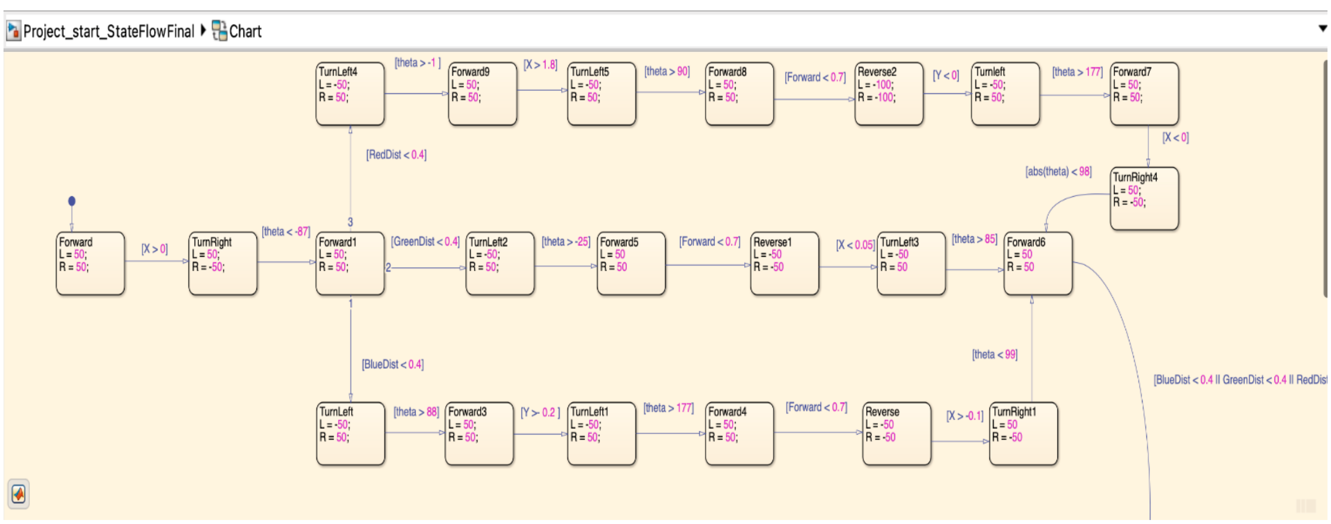


Fig 5.1(a) Stateflow chart to move the objects with three colours on the left side.

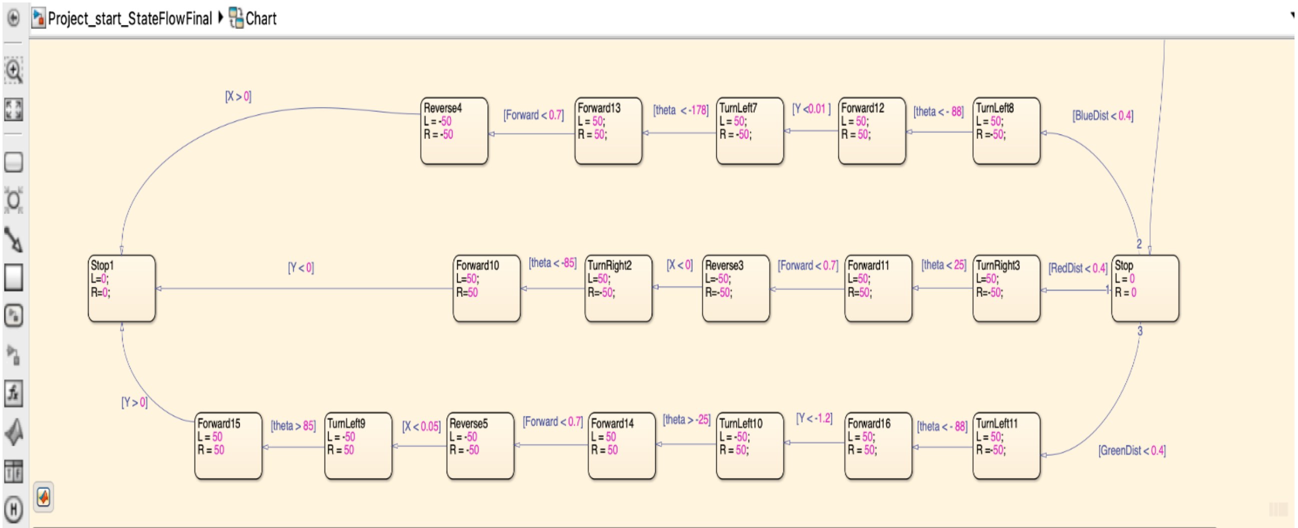


Fig 5.1(b) Stateflow chart to move the objects with three colours on the Right side.

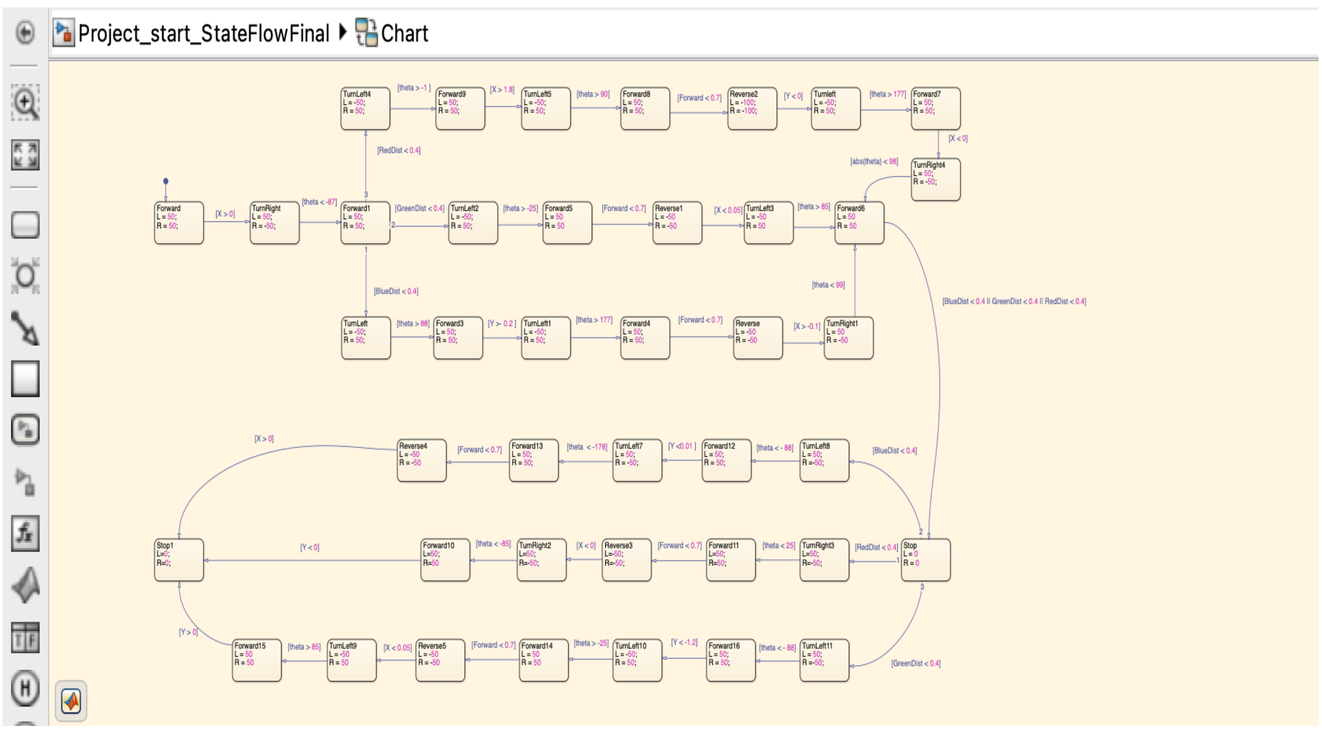


Fig 5.1(c) Combined stateflow with tasks of moving objects on both sides left and right into the matching arena

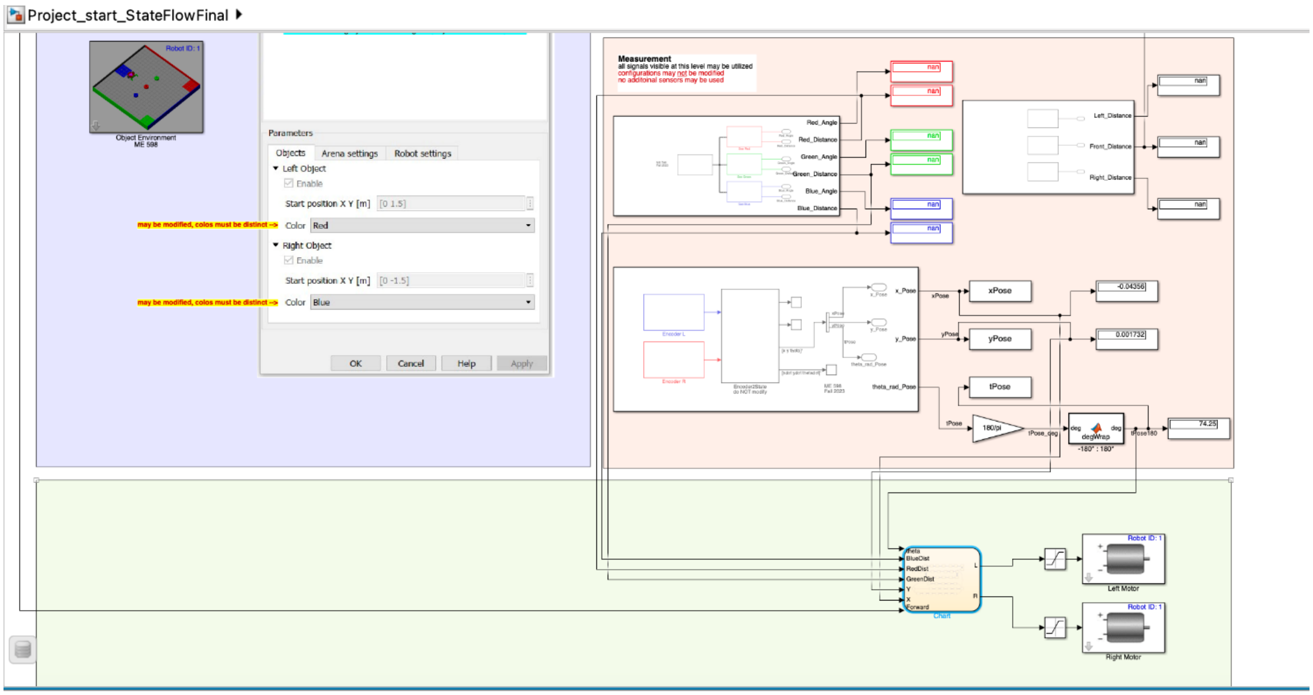


Fig 5.1(d) Simulink chart created and connected with the stateflow chart

Results:

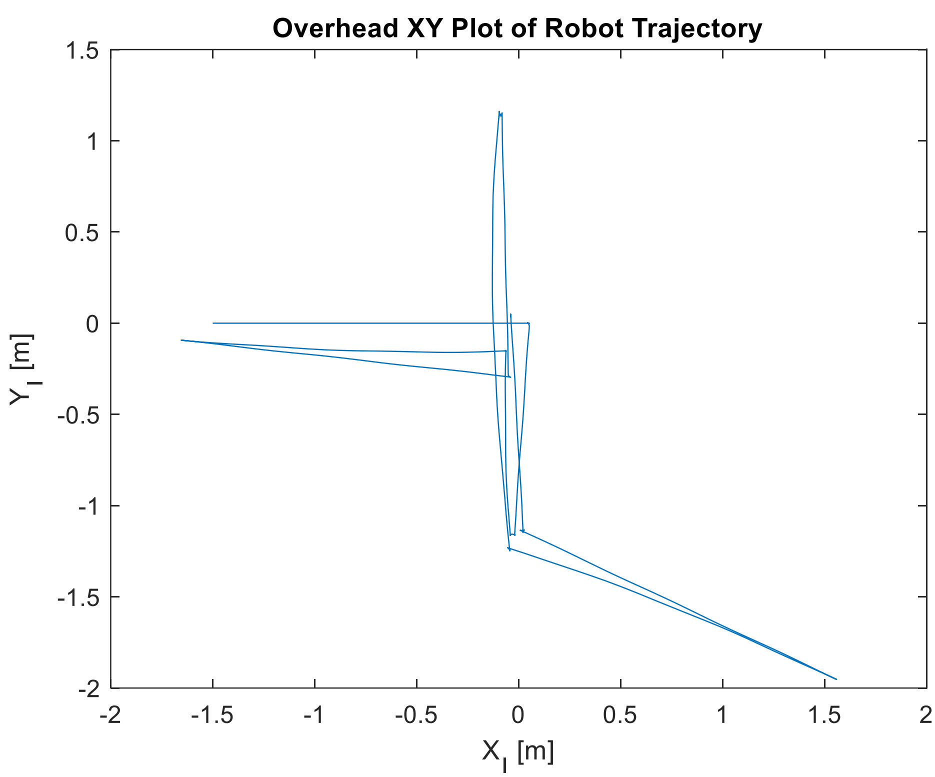


Figure 5.2: XY Plot for Trial 1 with Right Object Blue and Left Object Green

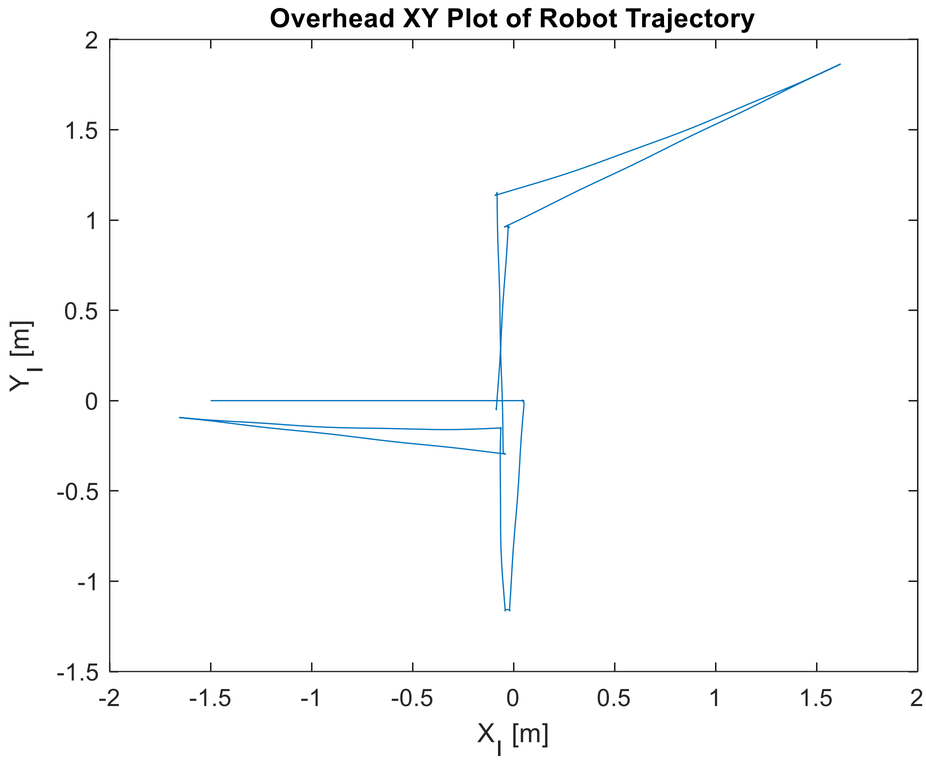


Figure 5.3: XY Plot for Trial 2 with Right Object Blue and Left Object Red

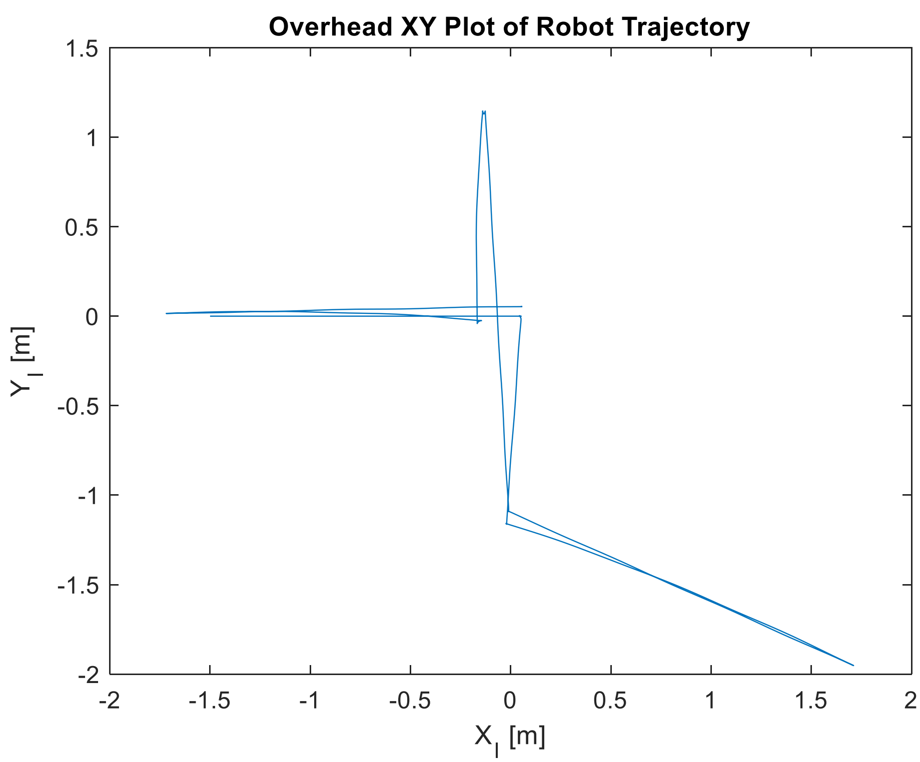


Figure 5.4: XY Plot for Trial 3 with Right Object Green and Left Object Blue

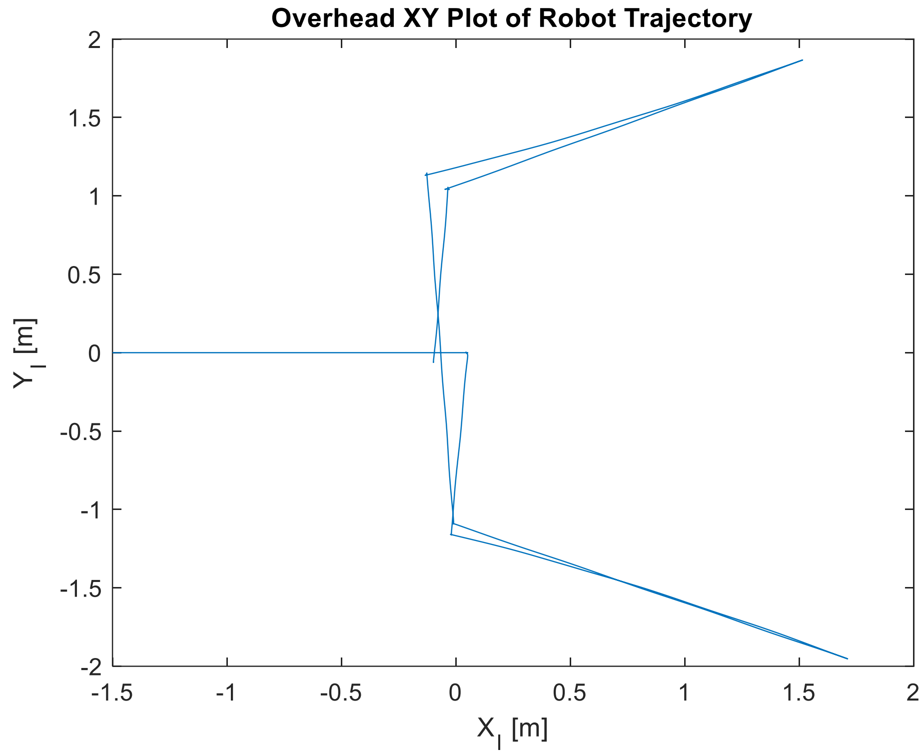


Figure 5.5: XY Plot for Trial 4 with Right Object Green and Left Object Red

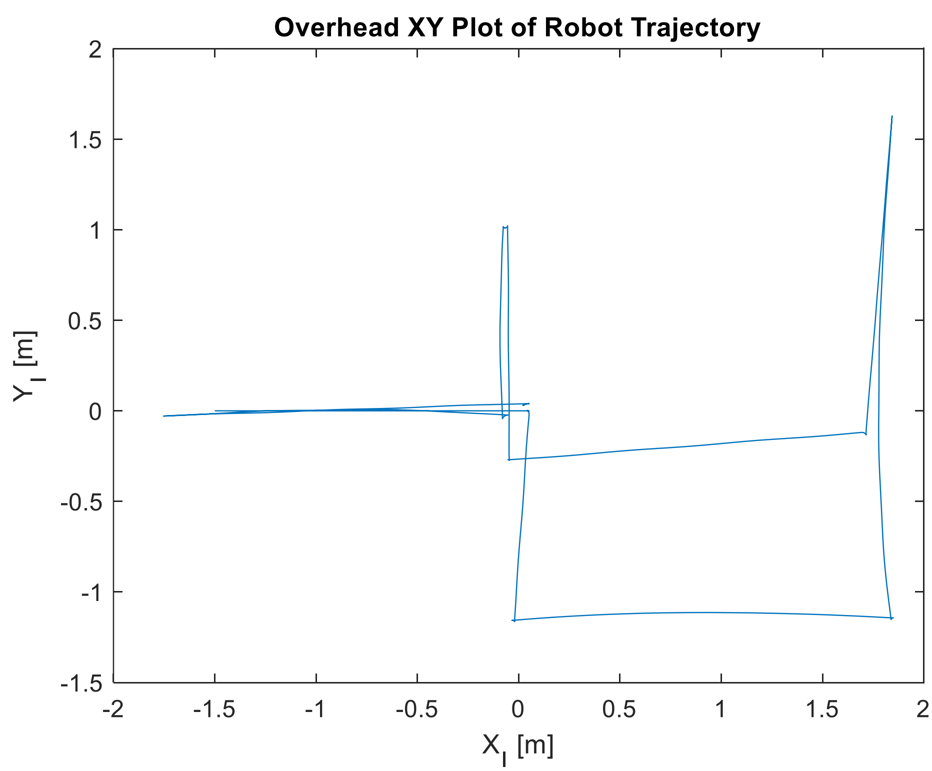


Figure 5.6: XY Plot for Trial 5 with Right Object Red and Left Object Blue

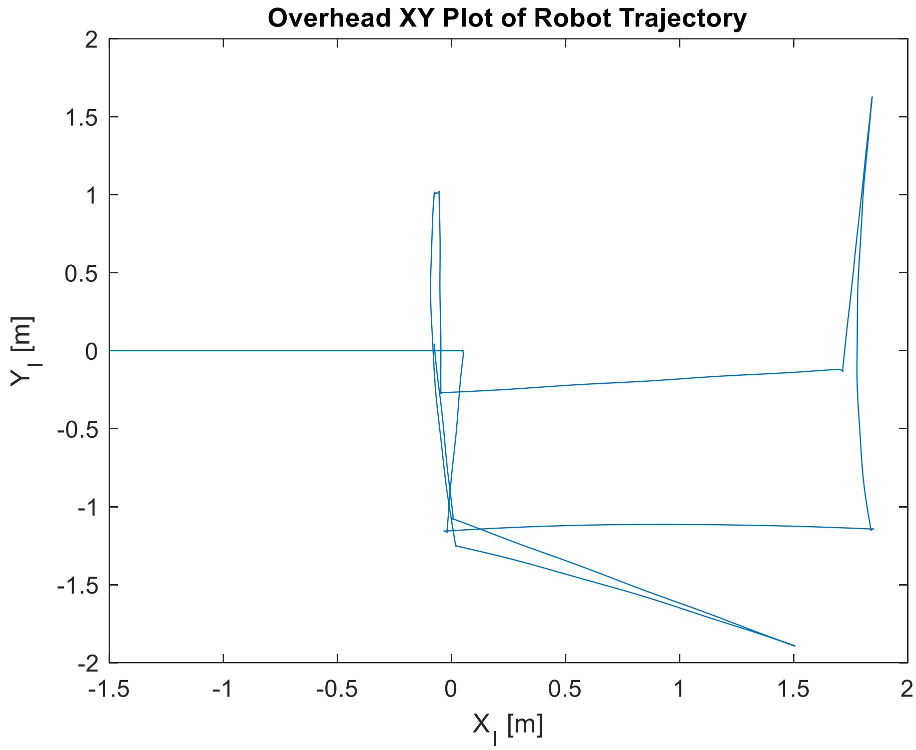


Figure 5.7: XY Plot for Trial 6 with Right Object Red and Left Object Green

Table 5.1: Table of Objects Sorted Successfully and Times for Each Trial

Trial	Objects Sorted Successfully	Time (seconds)
1	Right Object Blue and Left Object Green	150
2	Right Object Blue and Left Object Red	125
3	Right Object Green and Left Object Blue	125
4	Right Object Green and Left Object Red	115
5	Right Object Red and Left Object Blue	155
6	Right Object Red and Left Object Green	170

Discussion:

Our system consistently performed with 100% accuracy over numerous tests. We were able to effectively sort objects of all three colors into their corresponding zones. The general Stateflow pipeline was conserved between object placement and color, with some specific adaptations depending on the situation. In general, our robot started in a home location, moved towards the fixed location of an object, sensed which object it was holding, and pivoted towards the corresponding target location. This is a stable method, but some color orders introduced environmental elements requiring novel planning. For example, the green object needed more careful tuning when it was the second object to be moved. What originally seemed like an error in the branch for the second green object turned out to be a misalignment caused by the first object. After several testing, it was noticed that the robot approached the second object differently if the first object was blue vs red. For when the second object was red or blue, this did not create error as the zones for those objects are closer to the second object. However, since the second green object is so far from its zone, the difference in initial conditions compounded to produce significant error where the object would not be fully within the zone. To solve this, the initial red and blue branches were tuned more accurately as well as the second green object being altered. After some additional testing the sweet spot was found for accurate repeatability for all color combinations.

In addition, the red target location was in the upper left corner of the environment; if the red object is in the first location (i.e., center right) the robot runs the risk of colliding with the second target (i.e., center left) while bringing the red object to the red target location. Consequently, for trials such as these an alternative route was mapped out to bring the robot around the obstacle (the second object). This is an effective method of maintaining accuracy in this task but has shown a downside. That is, the time to completion is significantly greater in trials where the first object is red, than in trials with a different first color. In fact, a two-way t test displays a significant difference between those two types of trials with a p value < 0.05. With more time it could be possible to speed up those trials and reduce the difference in timing. Perhaps an approach could be to use machine learning to allow the object to traverse its environment in the most intuitive way possible. However, even with the decrease in speed our robot can move all objects into their correct location.

Student Feedback

Overall, our team, Group R4, demonstrated strong collaboration and effective communication throughout the project. Regular team meetings allowed us to discuss progress, challenges, and share insights. Clear communication facilitated a smooth workflow and ensured that everyone was on the same page regarding project objectives and tasks.

1. Difficulties Encountered:

Throughout the project, we faced a few challenges, notably in fine-tuning the robot's movements and ensuring precise object sorting. Integrating Stateflow with MATLAB posed a learning curve for some team members, requiring additional time to grasp its functionalities. Debugging complex algorithms for autonomous sorting was another area that demanded collaborative problem-solving.

2. Thoughts on the Project and Time Investment:

Overall, the project was both challenging and rewarding. Developing a MATLAB simulation involving Stateflow provided an excellent opportunity to apply theoretical knowledge to a real-world scenario. As a whole, team members dedicated approximately 40-50 hours to complete the project, including research, coding, debugging, and documentation.

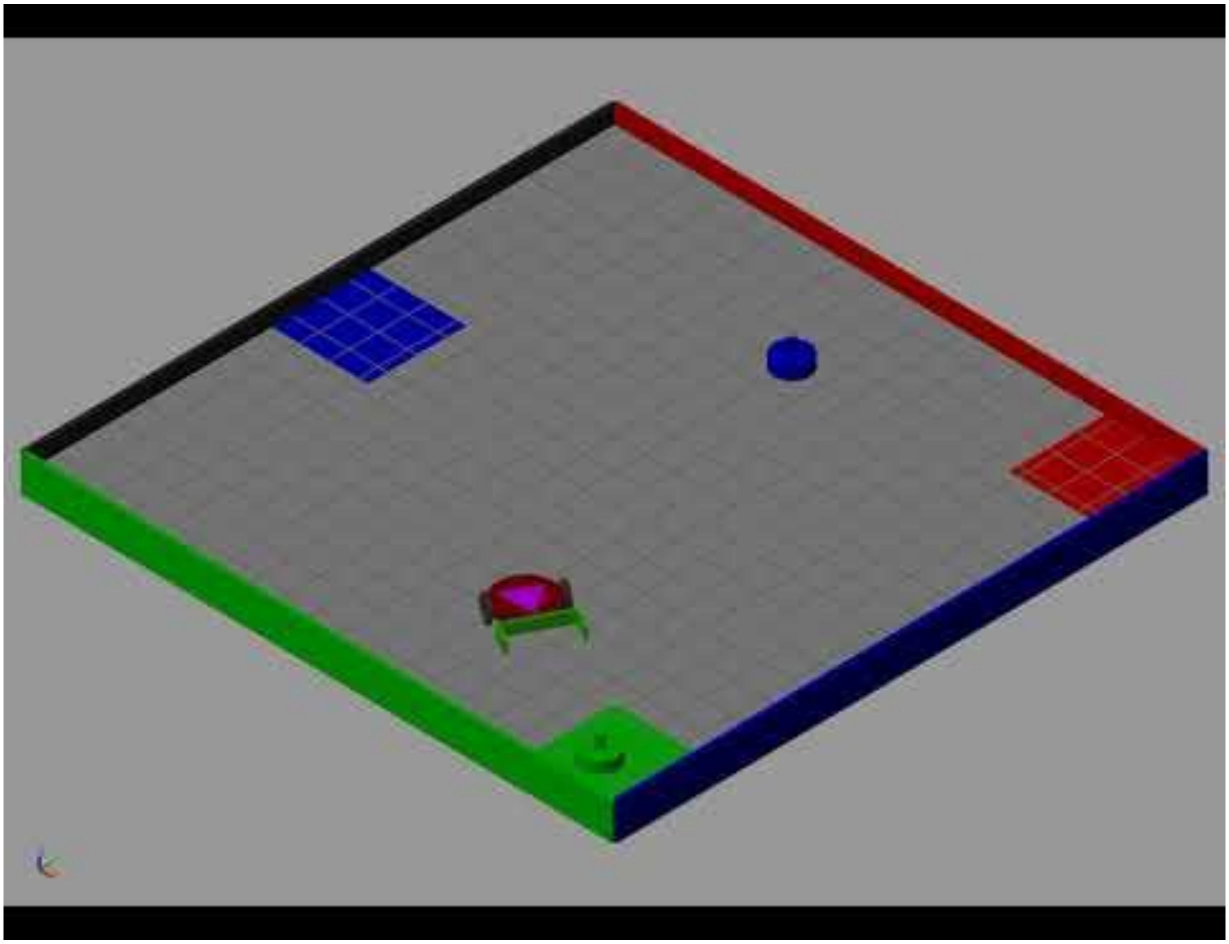
3. Recommendation for the Future:

We believe that this project offers a valuable learning experience for future students interested in robotics and control systems. While challenging, it provides a hands-on opportunity to apply MATLAB and Stateflow concepts to practical problem-solving. We recommend this project for future cohorts, with the suggestion to allocate additional time for familiarizing oneself with Stateflow's intricacies.

Conclusion:

In conclusion, Group R4 has successfully navigated the complexities of developing a MATLAB simulation for autonomous object sorting. The MATLAB Simulink model provided an opportunity to apply everything we have learned throughout the semester in a real-world environment. The project required integration of sensor readings and motor outputs with control algorithms to successfully operate the robot autonomously. The project also introduced the Stateflow environment in MATLAB, which was an extremely useful tool for creating step-based code. The realistic physics within the simulation created circumstances where the robot would be in a different location than the motor sensors perceived, and this provided new challenges for the group to overcome as this strayed from our assumptions of ideal cases from earlier in the course. The solution required novel and innovative usage of the available sensors to tune the algorithm so that the robot could operate completely autonomously and be successful in guiding both objects to their desired zones for any combinations of colors. Our collective efforts, perseverance in overcoming challenges, and the ultimate success of the project are a testament to our commitment to learning and collaboration.

Appendix A: Video of Trial 1



https://youtu.be/GjHFHPD27Xs?si=iuhwhSON-HvhUYGb