Rohan Chandrasekar

+1 412 909 9897 | rohan.csekar@gmail.com | LinkedIn | GitHub | Website

Education

Carnegie Mellon University — Pittsburgh,PA
MS Mechanical Engineering-Specialization:Robotics and Control Systems GPA:3.92/4 — Aug 2022-May 2024

Vellore Institute of Technology — Chennai,India
B.Tech Electrical and Electronics Engineering GPA:8.79/10 — July 2017 -June 2021

Experience

Biorobotics Lab, Carnegie Mellon University — Pittsburgh,PA
Graduate Research Assistant-Multi-Model Perception Uber Good — October 2022-May 2024

***Mentor**:Dr.Matthew J.Travers,Carnegie Mellon University*

Autonomy
- Played a pivotal role in developing a multi-robot system from the ground up using **ROS**.
- Developed a trajectory library based local planner for an RC car by incorporating the kinematic bicycle model and integrating it with voxel grids.
- Developed a resilient decentralized multi-robot planning and coordination move-out-of-the-way algorithm that involves collaboration of diverse multi-agent robots like the Spot by Boston Dynamics.

Systems Engineering and UI
- Designed a user interface utilizing the ROS integrated Qt-based inteface for the system.
- Developed and seamlessly integrated **decentralized** and **centralized** behavior trees within a high-speed, diverse multi-robot system designed for tasks such as search and rescue, exploration, and reaching destinations in complex, unstructured environments.
- Significantly enhanced the system's performance by achieving a CPU usage of less than **3%**.This involved successfully incorporating the **BehaviorTree.CPP** library into the system stack.

DevOps
- Developed the CI/CD pipeline by setting up unit tests using rostest for the stack and integrating it with self-hosted runners with Github Actions.

Hardware
- Conducted testing for a payload comprising a Jetson Xavier, lidar, cameras, and an IMU.

IIT Madras — Chennai,India
Project Associate — December 2021-July 2022

***Mentor**:Dr.Anil Prabhakar,Professor,IIT Madras*
- Developed Python-based algorithms to both reduce the thickness of samples and test/qualify them using **OpenCV**.
- Involved in developing the user interface of the product using **Tkinter**.

Projects

Controlling a wheeled double pendulum robot — Oct 2023-Dec 2023
- Control strategies such as PID, LQR, and Input Shaping were implemented to stablize a wheeled double pendulum and simultaneously preventing oscillations in the upper pendulum.
- The system was modeled using Lagrangian mechanics, and controllers along with a Luenberger observer were designed using MATLAB.
- Furthermore, the effectiveness of these controllers was tested on hardware by modifying an Elegoo Tumbller self-balancing robot to include a second pendulum.

Real-time Lattice Based A* Planning for RC Cars — Oct 2023-Dec 2023
- Developed a local planner focusing on kinodynamically feasible paths for an RC car using a kinodynamic bicycle model.
- Utilized A* graph search with euclidean distance heuristic, culling paths hitting obstacles identified by simulated Velodyne lidar.
- Implemented path execution with iLQR controller and demonstrated successful testing in a simulated Gazebo world.

Building a constrained NLP solver using ProxQP — Feb 2023-May 2023
- Utilized Sequential Quadratic Programming (SQP) to transform the Nonlinear Programming (NLP) problem into a Quadratic Programming (QP) problem by linearization.
- Solved the QP using ProxQP and compared it with OSQP(Operator Splitting Quadratic Program)
- Enhanced performance by a factor of eight using ProxQP as compared to the existing OSQP

Control optimization of a race car using Webots — Oct 2022-Dec 2022
- Utilized state-space analysis and control theory principles to fine-tune parameters and achieve peak performance
- Effectively incorporated a combination of controllers, including PID, LQR (Linear Quadratic Regulator), State Feedback, and an MPC(Model Predictive Controller), into the race car simulation.
- Reduced the track traversal time by a factor of two over the basic PID controller.

Technical Skills

- **Application Software**: MATLAB,Simulink,Robot Operating System(ROS),Thingspeak,Gazebo
- **Programming Languages**: C,C++,Python,Julia
- **Libraries/Tools**: Linux programming,Docker,Git,CMake,PyTorch,OpenCV,Matplotlib,Sklearn,Pandas,Tkinter