Automated Driving Using External Perception

Individual Lab Report - ILR09
November 2,2023

Team E - Outersense

Author:
Jash Shah

Team Members:

Atharv Pulapaka
Dhanesh Pamnani
Jash Shah
Ronit Hire
Shreyas Jha

Contents

1 Individual Progress

1.1 Planning subsystem

I have been working on developing a multi-layer planning architecture for our self-driving car system. This hierarchical framework integrates high-level mission planning with low-level trajectory generation and control.

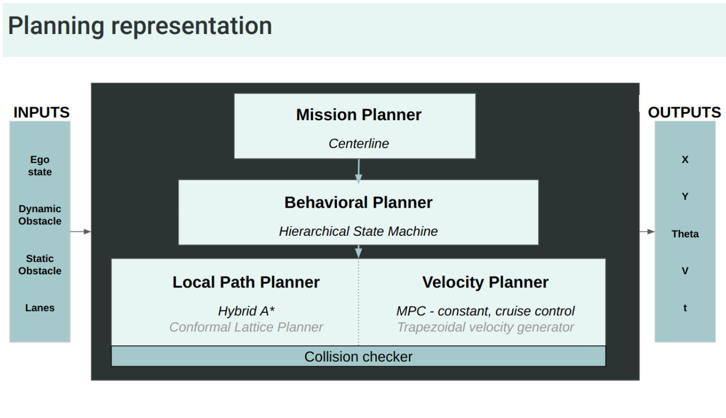

Figure 1: Planning diagram

1.1.1 Mission Planner

The topmost layer is the mission planner which defines the overall route plan for the vehicle based on destination goals. For our current system, this involves following a fixed series of centerline waypoints along the track. The mission planner provides long-term guidance but lacks environment details.

1.1.2 Behavioral Planner

The behavioral planner handles discrete medium-term decisions like overtaking, merging lanes, stopping at intersections. It leverages perception to make tactical choices while conforming to the broader mission plan. Our behavioral planner employs a finite state machine with four key states:

- **Stop**: This state brings the vehicle to a complete stop when required, such as at intersections.
- **Approach to Stop**: This transitions the vehicle speed in a controlled manner to achieve a stop safely.
- **Follow Leader**: This state maintains safe following distance from a lead vehicle using perceived position data.
- **Track Speed**: In the absence of lead vehicles, this state tracks the centerline at a reference velocity.

The planner switches between these states based on perceived environment and events from the mission plan, enabling situational tactical control. For our current structured track, the Follow Leader and Track Speed modes predominantly govern the vehicle behavior, while the stopping states add capabilities for future expansion.

1.1.3 Local Planner

This module generates precise, collision-free trajectories in short time horizons, translating behavioral choices into continuous paths. Our local planner is based on a hybrid A* algorithm that combines grid and sampling techniques. It plans over a local window, invoking re-planning as needed to accommodate dynamic obstacles.

1.1.4 Velocity Planner

The velocity planner regulates speed along the trajectory from the local planner. It smooths the profile and modulates velocity based on comfort, safety and dynamic constraints. Our system employs Model Predictive Control for this layer.

1.1.5 Collision Checking

An additional collision checker validates the local trajectory against obstacle occupancy data from perception, flagging any segments in collision. This allows incorporating reactive safety into the layered planning pipeline.

1.2 Hybrid A* Algorithm Details

The Hybrid A* planner is the meat of our planner:

1.2.1 Hybrid A* Algorithm

The hybrid A* planner discretizes the continuous search space into a grid representation. This allows systematic exploration using graph search algorithms like A*. However, a key challenge with standard grid-based methods is the massive search space they entail, making it computationally intractable for real-time performance. This is where the hybrid approach comes in. It introduces random sampling of the free configuration space to guide and focus the A* graph search, avoiding exploration of irrelevant areas.

Specifically, the algorithm first initializes the grid map, defining key parameters like resolution and the heuristic cost function. It then randomly samples points in the free space around obstacles. These sampled points act as intermediate milestones that the A* search uses to guide its exploration towards the goal. The grid search expands nodes, moving towards the samples based on cost heuristics. Once the goal is reached, the optimal path is extracted by tracing back the lowest cost grid nodes. A smoothing function is finally applied to remove jagged motions resulting from the discrete grid representation.

This hybrid strategy reaps multiple advantages. The guided search based on sampling is far more efficient than blind A* expansion. The use of a grid retains the optimality guarantees of A* unlike other sampling planners. The balance between guided exploration and optimal graph search makes it widely applicable for self-driving cars navigating complex environments.

1.2.2 Implementation Details

The implementation of the hybrid A* planner involved the following key steps:

First, a grid-based world representation was defined with the flexibility to configure resolution and other parameters. Functions to map continuous spatial positions of obstacles and vehicles onto discrete grid nodes were written. A sampling module was created to randomly sample points around obstacles according to a specified density. The core A* search logic was then adapted to integrate the sampling-based guidance during node expansions. Cost heuristics combined path length estimates and proximity to samples to drive the search. Finally, a post-processing spline-smoothing step was added to refine the raw grid path into a continuous trajectory.

The algorithm was integrated with the Gazebo simulation environment and ROS nodes to enable closed-loop testing. The performance was found satisfactory in efficiently computing smooth, collision-free paths for the simulated vehicle in real-time, even in cluttered environments with narrow passages. Further testing in the real-world is planned as the next step.

2 Challenges

Implementing a functional planning system posed several key challenges:

Firstly, integrating the planner with our custom vehicle model and map representation required significant effort. Adapting the algorithms to work with the specific constraints and coordinate frames was non-trivial.

Secondly, tuning the various planner parameters like grid resolution, sampling density and heuristic weights for optimal performance was an iterative process requiring extensive experimentation and testing.

Further, meeting real-time constraints was difficult, especially for computationally expensive searches on large grids. This needs optimizations like multi-threading, efficient data structures and algorithmic improvements.

Another challenge was smoothing the jagged grid-based paths without compromising optimality. This required developing custom smoothing techniques tailored for our system.

Finally, handling complex dynamic environments and external sensor data within the planner for robust performance remains an ongoing research challenge.

3 Teamwork

In terms of teamwork, we collaborated closely as a team to tackle critical tasks efficiently. Each member contributed their expertise to ensure the project's success.

Ronit: Ronit focused primarily on the perception subsystem. He worked on removing dependencies on ArUco markers to make detection more robust. Additionally, Ronit played a key role in integration and testing of the overall system.

Dhanesh: Dhanesh's expertise is in planning. He implemented the custom vehicle model and map representation needed by the hybrid A* planner. We collaborated extensively on adapting the planner and tuning it for optimal performance.

Shreyas: Shreyas concentrated his efforts on refining the state estimation module. He tuned the parameters to achieve precise odometry from the VESC IMU. He was also instrumental in vehicle integration and field testing.

Atharv: Atharv worked on augmenting the longitudinal control system, adapting the PID cruise controller for maintaining safe distances. He provided important controls perspective while integrating the planner.

Through close teamwork, we were able to combine our complementary strengths to tackle the various facets of this complex project efficiently.

4 Future Work

4.1 Personal

In the upcoming phases, I plan to focus on:

1. Fixing the speed issue in the planner to ensure planning at the required rate.
2. Enhancing the planner to account for static obstacles and navigate around them.
3. Rigorously testing the planner and tuning parameters to improve performance.
4. Smoothing integration of the planner with other subsystems like perception and control.

4.2 Team

As a team, our future goals are:

1. Integrating all subsystems including perception, planning and control for closed loop performance.
2. Tuning the state estimation module for accurate and reliable odometry data.
3. Refining the VESC parameters and control on the actual vehicle.
4. Rigorously testing the integrated system, identifying and resolving issues.
5. Incrementally enhancing individual subsystems for better overall performance.

Through systematic teamwork, we aim to synergize our individual strengths to take the integrated autonomous car platform to the next level.