Summer Reading 11: Real-Time Trajectory Planning for Autonomous Driving with Gaussian Process and Incremental Refinement

Social Robot Navigation Project @ Bot Intelligence Group

Paper:
Real-Time Trajectory Planning for Autonomous Driving with Gaussian Process and Incremental Refinement
Jie Cheng1, Yingbing Chen2, Qingwen Zhang2, Lu Gan2 and Ming Liu1,2

Summary:

Abstract

- The authors propose an efficient trajectory planning system for autonomous driving in complex dynamic scenarios through iterative and incremental path-speed optimization.
  - **Gaussian Process (GP) path planner**
    - First, a path planner based on Gaussian process generates a continuous arc-length parameterized path in the Frenet frame, considering static obstacle avoidance and curvature constraints.
  - **Efficient S-t Graph Search Method** (w/ local truncation mechanism)
    - An efficient s-t graph search method is introduced to find a speed profile along the generated path to deal with dynamic environments.
  - **Incremental Refinement (Incremental Path Speed Adjustment)**
    - Finally, the path and speed are optimized incrementally and iteratively to ensure kinodynamic feasibility.

Introduction

- The trajectory planning system needs to handle static obstacles, dynamic traffic participants and various constraints (e.g., speed limits, kinodynamic feasibility) in real-time.
- Existing works can be categorized into 2 main classes:
  1. Spatio-temporal planning (s-t planning; s-t approaches)
     - The main advantage of these approaches is that they can consider spatial and temporal constraints simultaneously.
       - However, finding an initial guess in the spatio-temporal space is non-trivial, sometimes even harder than solving the optimization problem.
       - As a result, these approaches are prone to local optimum.
  2. Path-speed decoupled planning (path-speed decoupled approaches)
     - Path-speed decoupled approaches break the high dimensional problem into two easier subproblems:
       - Planing a path
       - Generating a speed profile along that path.
     - The benefits are two folds:
       - Firstly, they do not require a preliminary result in the s-t domain.
       - Secondly, they usually enjoy higher efficiency and flexibility.
     - However, they are generally much harder to guarantee kinodynamic feasibility due to the decomposition structure.
- Apart from the issues mentioned above, most existing works have difficulty handling the curvature constraint.

        - Some ignore the constraint by assuming the resulting trajectory is smooth, which is not reasonable in some scenarios (e.g., sharp turns).
        - Others have to approximate or relax the constraint using a heuristic limit, leading to an overly conservative or even infeasible solution.
    - The authors propose an efficient trajectory planning framework to overcome the problems of existing methods. The authors' approach can be categorized as path-speed decoupled planning in general.
        - Firstly, this path planning problem is converted to a probabilistic inference problem with Gaussian process (GP).
            - A path planner incorporating GP generates a collision-free path under strict curvature constraints.
        - Analytic formulation of curvature constraint is used and integrated into the path planning process.
        - To avoid local optimum, a novel and efficient s-t graph search method is introduced.
        - Finally, an incremental path speed adjustment method is adopted to ensure kinodynamic feasibility.
    - The proposed approach is able to directly consider the curvature constraint without approximations, and efficiently generate high-quality kinodynamically feasible trajectories.

Related Work

- Iterative Path-speed Refinement
    - The online iterative path-speed refinement technique is widely adopted in the unmanned aerial vehicle (UAV) community.
        - However, this technique is less commonly used for AVs (Autonomous Vehicles).
            - The main reason is that various non-linearities (e.g. collision constraints on a rectangular vehicle body, nonlinear vehicle dynamics) make the optimization problem for AVs expensive to solve iteratively.
- Motion Planning via Probabilistic Inference
    - Recent progress on motion planning via probabilistic inference has opened a new window onto the motion planning problem.
    - Mukadam et al. propose Gaussian process motion planning (GPMP), where they show GP driven by linear, time-varying stochastic differential equations (LTV
    - SDEs) is the appropriate tool connecting motion planning and probabilistic inference.
    - The follow-up GPMP2 uses factor graphs to model the planning problem, and supports fast replanning by making use of an incremental Bayes tree solver.
    - The authors use the Gaussian Process (GP) for path planning, and theoretically investigate the connection between this planning method and the well-known jerk optimal solution.

System Overview

- The trajectory planning process contains 3 major steps as depicted in Fig. 2 (blue part).
    - First, a GP-based path planner generates a path in the Frenet frame.
    - Second, the speed planning module finds an optimal speed-profile along that path with the efficient s-t graph search.
    - Finally, when the result trajectory fails the feasibility check, the authors adjust the current path with the incremental refinement module

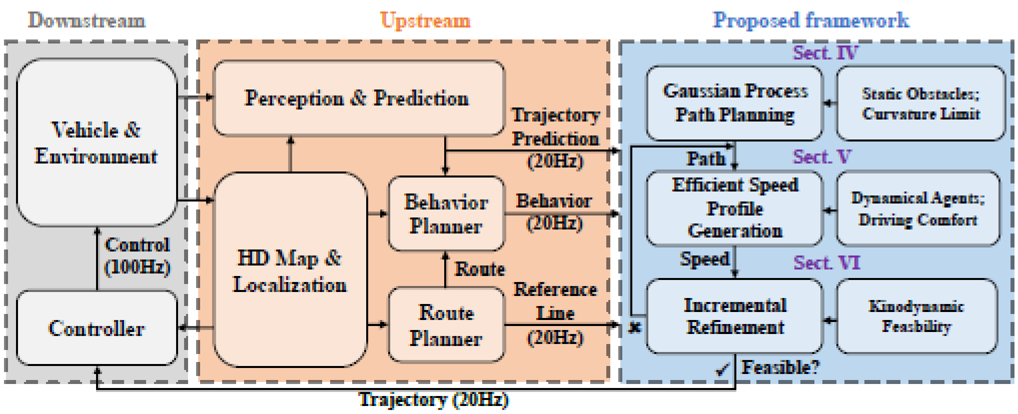


Fig. 2. Overview of the proposed trajectory planning framework and its relationship with other system components.

-

Gaussian Process (GP) Path Planning

- The authors consider a path in the Frenet frame.
- The authors convert the path planning into a probabilistic inference problem with Gaussian Process (GP).
- Two major constraints need to be considered in path planning:
    - collision avoidance
    - curvature constraints
- Path constraints are modeled as the likelihood function.

Efficient Speed-Profile Generation

- Efficient S-t Graph Search
    - In speed planning, local minimums inevitably exist.
        - For example, in a merging scenario, the AV can choose to either decelerate to yield or accelerate to overtake the human driver.
        - To escape from the local minimum and ensure safety, a searching, or sampling step is necessary
    - The authors' proposed s-t graph search algorithm consists of 3 main steps:
        - trajectory projection

        - A dynamical agent is critical if its future trajectory has possible intersections with the AV's desired path. For each critical agent, a sequence of trajectory points are sampled within time intervals. Then check at every trajectory point whether the bounding box of the agent at that position has an overlap with the desired path. If so, a blocked region is marked in the s-t graph, by finding its projection w.r.t. the desired path.
    - forward expansion
        - During the expansion, any child state, which falls into the blocked regions, will be abandoned. After expansion, each survived child state will repeat that process until reaching the maximum planning duration or path length.
    - local truncation
        - The number of states will grow exponentially during the expansion, which would be problematic for real-time implementation. To reduce the computation costs, the authors propose a local truncation mechanism.
- S-t Curve Smoothing
    - Although the authors efficiently find a speed-profile, the authors implicitly assume that the acceleration changes instantaneously during the forward expansion.
    - To improve the driving comfort, a smoothing step is desired. The authors fit the coarse speed-profile with a piecewise Bezier curve and construct the overall problem as a quadratic programming (QP).

Incremental Refinement

- The final trajectory may violate the lateral acceleration constraint if we overestimate.
- Existing solutions mostly adopt an iterative optimization approach, which is computationally expensive.
- The authors propose to iteratively refine the trajectory in an incremental manner.
    - Factor Graph and Bayes Tree
        - A factor graph is a bipartite graph that represents the factorization of a function.
        - Through variable elimination, a factor graph is converted to a Bayes tree, which is an efficient tool for incremental inference.
    - Incremental Trajectory Refinement
        - If the lateral acceleration of the trajectory point *i* exceeds the allowable value a_max_lat, a new factor is added to the factor graph, which constrains the maximum lateral acceleration of the path.
        - When the Bayes tree updates, only the parts of the path associated with the newly added factors are updated. Thereafter, the authors regenerate the speed profile for the new path. This process is repeated until all constraints are satisfied or the maximum number of iterations is reached.

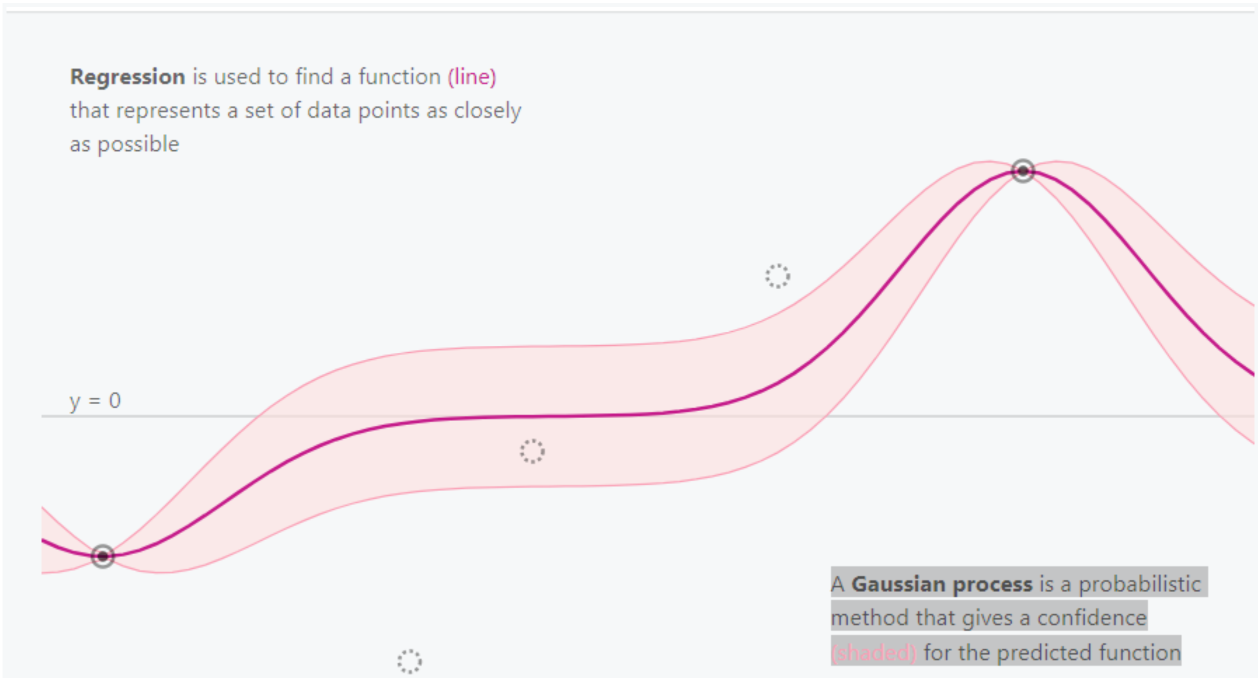
Conclusions

- The authors presented a real-time, kinodynamic trajectory planning framework in dynamic environments.
- The authors introduced a Gaussian Process (GP) path planner that generates a collision-free path under curvature constraints.
- An efficient s-t graph search method with a local truncation mechanism was also proposed to enable fast speed-profile generation.
- Kinodynamic feasibility is guaranteed by the novel incremental refinement scheme.

Glossary:

- S-t graph: The speed time graph (time on the x-axis; speed on the y-axis)

- Gaussian Process (GP): A Gaussian process is a probabilistic method that gives a confidence(shaded) for the predicted function

  https://distill.pub/2019/visual-exploration-gaussian-processes/

-