AA222 Final Project Proposal

Philippe Weingertner and Minnie Ho

May 21, 2020

1 Problem Formulation

This project proposal investigates trajectory optimization in the presence of obstacles [10, 2, 5, 12]. One such application for this class of problems is that of autonomous driving, where we have an ego vehicle and dynamic obstacles (vehicles, pedestrians) which may intersect our desired trajectory and which we wish to avoid using motion planning and control. Trajectory optimization problems minimize a cost function which takes into account start and terminal states, as well as cost along the trajectory path. The design space is subject to constraints on the states and control input at sampled time points.

We define the problem as follows:

- The state is defined by $x = [s, \dot{s}] \in \mathbb{R}^2$ a longitudinal position and speed, along a predefined path.
- The control is defined by $u = [\ddot{s}]$ a longitudinal acceleration (or deceleration) command.

Given a starting state $\mathbf{x_0}$ and a target state $\mathbf{x_T}$, we want to find the optimal control sequence which minimizes a quadratic cost function subject to the system dynamics and obstacle avoidance constraints. We have to deal with 10 objects crossing our path at different time instants. The problem is discretized and acceleration or deceleration commands may be executed every 250 ms.

$$\min_{u_0,\ldots,u_{T-1}} (x_T - x_{\text{ref}})^\intercal Q_T (x_T - x_{\text{ref}}) + \sum_{k=0}^{T-1} (x_k - x_{\text{ref}})^\intercal Q (x_k - x_{\text{ref}}) + u_k^\intercal R u_k$$

$$\text{subject to} \begin{cases} x_{k,\min} \le x_k \le x_{k,\max} \\ u_{k,\min} \le u_k \le u_{k,\max} \\ x_{k+1} = A_d x_k + B_d u_k \\ x_0 = x_{\text{init}} \\ \forall\, (t_{\text{col}}, s_{\text{col}})_{i \in [1,10]} \quad x_{t_{\text{col}}^{(i)}}[1] < s_{\text{col}}^{(i)} - \Delta_{\text{safety}} \text{ or } x_{t_{\text{col}}^{(i)}}[1] > s_{\text{col}}^{(i)} + \Delta_{\text{safety}} \end{cases}$$

Where we use a Linear Dynamics approximation with a Constant Acceleration model in between 2 time steps:

$$\begin{bmatrix} s \\ \dot{s} \end{bmatrix}_{k+1} = A_d \begin{bmatrix} s \\ \dot{s} \end{bmatrix}_k + B_d \begin{bmatrix} \ddot{s} \end{bmatrix}_k \text{ with } A_d = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, B_d = \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{bmatrix}$$

2 Approach

There are two main classes of trajectory optimization problems: direct and indirect. In the direct method, the trajectory problem is first discretized and then optimized - differential and integral equations are sampled, and the problem is transformed into a optimization program [6] and [4]. An example of a direct method is direct collocation, where the state and control trajectories are approximated using polynomial splines. Once this is done, the trajectory problem can be converted into a non-linear program and solved using NLP solvers.

In indirect methods, we must first analytically construct the necessary and sufficient conditions for optimality, discretize these conditions, and then solve the optimization problem. An example of an indirect method is one which does not explicitly represent states and rather only parametrizes the controls [1]. The Iterative-Linear Quadratic Regulator (iLQR) is an example of an indirect method that uses the first-order derivatives of the dynamics, while Differential Dynamic Programming is an indirect method that typically requires the second-order derivatives of the system dynamics.

We propose to investigate and compare some of these trajectory optimization approaches when applied to the context of autonomous driving. We note that the extra challenge here compared to typical settings is that we are dealing with obstacle avoidance constraints on top of pure dynamics constraint. Moreover these constraints are expressed with a set of OR constraints: we may proceed or yield to avoid collisions.

3 Measurement of Success

We use four metrics to evaluate the performance of our different approaches.

(1) The main success metric is the percentage of cases where we reach a target state without collision.

(2) The second metric is the agent runtime.

(3) The third metric is a comfort metric: the number of hard-braking decisions.

(4) The fourth metric relates to efficiency: how fast we reach a target while complying to some speed limitation.

References

[1] John T. Betts. "Survey of Numerical Methods for Trajectory Optimization". In: *Journal of Guidance, Control, and Dynamics* 21.2 (1998), pp. 193–207. DOI: `10.2514/2.4231`.

[2] Haoyang Fan et al. *Baidu Apollo EM Motion Planner*. 2018. eprint: `arXiv:1807.08048`.

[3] C. R. Hargraves and S. W. Paris. "Direct trajectory optimization using nonlinear programming and collocation". In: *Astrodynamics 1985*. Aug. 1986, pp. 3–12.

[4] T. A. Howell, B. E. Jackson, and Z. Manchester. "ALTRO: A Fast Solver for Constrained Trajectory Optimization". In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2019, pp. 7674–7679.

[5] Christos Katrakazas et al. "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions". In: 2015.

[6] Matthew Kelly. "An Introduction to Trajectory Optimization: How to Do Your Own Direct Collocation". In: *SIAM Review* 59.4 (2017), pp. 849–904. DOI: `10.1137/16m1062569`.

[7] Mykel J. Kochenderfer and Tim A. Wheeler. *Algorithms for Optimization*. The MIT Press, 2019.

[8] C. Liu, W. Zhan, and M. Tomizuka. "Speed profile planning in dynamic environments via temporal optimization". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. 2017, pp. 154–159.

[9] Changliu Liu, Chung-Yen Lin, and Masayoshi Tomizuka. "The Convex Feasible Set Algorithm for Real Time Optimization in Motion Planning". In: *SIAM J. Control and Optimization* 56 (2017), pp. 2712–2733.

[10] Ugo Rosolia, Stijn Bruyne, and A.G. Alleyne. "Autonomous Vehicle Control: A Nonconvex Approach for Obstacle Avoidance". In: *IEEE Transactions on Control Systems Technology* 25 (June 2016), pp. 1–16. DOI: `10.1109/TCST.2016.2569468`.

[11] Andreas Wachter and Lorenz Biegler. "On the Implementation of an Interior Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming". In: *Mathematical programming* 106 (Mar. 2006), pp. 25–57. DOI: `10.1007/s10107-004-0559-y`.

[12] Xiaojing Zhang, Alexander Liniger, and Francesco Borrelli. "Optimization-Based Collision Avoidance". In: *ArXiv* abs/1711.03449 (2020).