NA 565 - Fall 2023

Trajectory Optimization

October 4, 2023

UNIVERSITY OF MICHIGAN

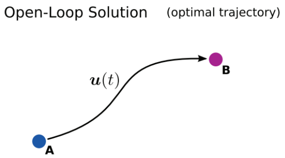
What is Trajectory Optimization?

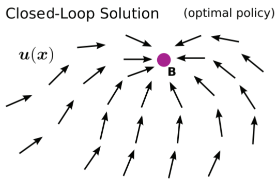
- Trajectory optimization is a collection of techniques that are used to find open-loop solutions to an optimal control problem.

- The solution to a trajectory optimization problem is a sequence of controls $u^*(t)$, given as a function of time, that moves a system from a single initial state to some final state.

Open-Loop vs. Closed-Loop Solutions

- An optimal trajectory starting from any point in the state space can be recovered from a closed-loop solution by a simple simulation.
- The sequence of controls, combined with the initial state, define a single trajectory that the system takes through state space.





Recall: State Feedback Control

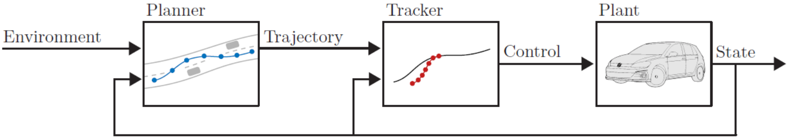
- In LQR, we have a linear state feedback

$$u_k = Kx_k.$$

- In MPC, we have a complicated state feedback obtained by solving an online optimization problem

$$u_k = \mu_{\mathsf{MPC}}(x_k).$$

Trajectory Optimization vs. Tracking Controller (MPC)



Trajectory Optimization Problem:

$$
\begin{aligned}
&\min_{x_t, u_t} N(x_{t_N}) + \int_0^{t_N} L(x_t, u_t) dt \\
\text{s.t.} \quad & \dot{x}_t = f(x_t, u_t) \\
& x(0) = x_{\mathsf{init}}, x(t_N) = x_{\mathsf{des}} \\
& + \text{additional constraints}
\end{aligned}
$$

Remark

Additional constraints include path constraints (additional equality or inequality constraints involving x_t and u_t) and bounds on the state and input.

Why not just solve this optimization problem?

- The decision variables, x_t and u_t are vector-valued functions. This makes the problem infinite-dimensional.

$$
\begin{aligned}
&\min_{x_t, u_t} N(x_{t_N}) + \int_0^{t_N} L(x_t, u_t) dt \\
\text{s.t.} \quad & \dot{x}_t = f(x_t, u_t) \\
& x(0) = x_{\text{init}}, x(t_N) = x_{\text{des}} \\
& + \text{additional constraints}
\end{aligned}
$$

Strategies for Solving the Problem

There are two strategies for approaching the problem:

- Indirect methods:
  Indirect methods analytically construct the necessary and sufficient conditions for optimality, e.g., Euler–Lagrange equation, Hamilton–Jacobi–Bellman equation.

- Direct methods

Both direct and indirect methods can be solved via *computation*:

- Shooting: Only solve for inputs (and boundary states) and simulate the state via forward simulation of dynamics. Good for simple control and no path constraints.
- Collocation: Parametrize the problem via many trajectory segments described by polynomials. Allows for complicated control and path constraints.

- Indirect methods: Optimize then discretize.
- Direct methods: Discretize then optimize.

We focus on direct methods. Specifically, direct collocation has been successful in robotics.

- Direct shooting
- Direct collocation

Example (click!)

Convex Formulations for Linear Systems, Recall: LQR

- $$\min_{x_k \in \mathbb{R}^n, u_k \in \mathbb{R}^m} \ \frac{1}{2} x_N^\mathsf{T} L x_N + \sum_{k=0}^{N-1} \frac{1}{2} x_k^\mathsf{T} Q x_k + \frac{1}{2} u_k^\mathsf{T} R u_k$$
  $$\text{s.t.} \quad x_{k+1} = A_k x_k + B_k u_k, k = 0, \dots, N-1$$
  $$x_0 = x_{\mathsf{init}}, x_N = x_{\mathsf{des}}$$
- Variables are state sequence $x_1, \dots, x_N$ and input sequence $u_1, \dots, u_{N-1}$.
- Constraints are linear dynamics equations and the initial and final state.

- In LQR, constraints are linear dynamics equations and the initial and final state.
- If additional constraints are affine, then the problem is a convex QP.

Nonconvex Trajectory Optimization: Direct Transcription

$$\min_{x_k,u_k} N(x_N) + \sum_{k=0}^{N-1} L_k(x_k,u_k)$$

$$\text{s.t.} \quad x_{k+1} = f_k(x_k,u_k), k = 0,\ldots,N-1$$

$$x_0 = x_{\mathsf{init}}, x_N = x_{\mathsf{des}}$$

$+$ additional constraints

Remark

$x_{k+1} = x_k + \int_{t_k}^{t_{k+1}} f(x_t,u_t)dt := f_k(x_k,u_k).$

Continue ...

Click below!

Trajectory Optimization

Next Time

- Trajectory Optimization Examples in Python

References and Further Reading

- R. Tedrake, Underactuated Robotics
  https://underactuated.csail.mit.edu/
- M. Kelly, 2017. An introduction to trajectory optimization: How to do your own direct collocation. SIAM Review, 59(4), pp.849-904.
- P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. Del Prete, 2022. Optimization-based control for dynamic legged robots. https://arxiv.org/abs/2211.11644