CDS 112/Ae 103a

Problem 1. In this problem, you will use the maximum principle to show that the shortest path between two points is a straight line. We model the problem by constructing a control system

$$\dot{x} = u,$$

where $x \in \mathbb{R}^2$ is the position in the plane and $u \in \mathbb{R}^2$ is the velocity vector along the curve. Suppose we wish to find a curve of minimal length connecting $x(0) = x_0$ and $x(1) = x_f$. To minimize the length, we minimize the integral of the velocity along the curve,

$$J = \int_0^1 \|\dot{x}\| \, dt = \int_0^1 \sqrt{\dot{x}^\mathsf{T} \dot{x}} \, dt,$$

subject to to the initial and final state constraints. Use the maximum principle to show that the minimal length path is a straight line.

Problem 2. Consider the optimal control problem for the system

$$\dot{x} = -ax + bu,$$

where $x = \mathbb{R}$ is a scalar state, $u \in \mathbb{R}$ is the input, the initial state $x(t_0)$ is given, and $a, b \in \mathbb{R}$ are positive constants. (Note that this system is not quite the same as the one in Example 3.2.) The cost function is given by

$$J = \tfrac{1}{2} \int_{t_0}^{t_f} u^2(t) \, dt + \tfrac{1}{2} c x^2(t_f),$$

where the terminal time t_f is given and c is a constant.

(a) Solve explicitly for the optimal control $u^*(t)$ and the corresponding state $x^*(t)$ in terms of t_0, t_f, $x(t_0)$ and t and describe what happens to the terminal state $x^*(t_f)$ as $c \to \infty$.

(b) Let $a = 1$, $b = 1$, $c = 1$, and $t_\mathrm{f} - t_0 = 1$. Solve the optimal control problem numerically using python-control (or MATLAB) and compare it to your analytical solution by plotting the state and input trajectories for each solution. Take the initial conditions as $x(t_0) = 1$, 5, and 10.

(c) Suppose that we wish to have the final state to be exactly zero. Change the optimization problem to impose a final constraint instead of a final cost. Plot the state and input trajectories, and compare the computation times for the results in 2b to the computation time using a terminal constraint.

Problem 3. Consider the dynamics of the vectored thrust aircraft described in Examples 2.4 and 3.5. The equations of motion are given by

$$\begin{aligned}
m\ddot{x} &= F_1 \cos\theta - F_2 \sin\theta - c\dot{x}, \\
m\ddot{y} &= F_1 \sin\theta + F_2 \cos\theta - c\dot{y} - mg, \\
J\ddot{\theta} &= rF_1.
\end{aligned} \tag{1}$$

with parameter values

$$m = 4 \text{ kg}, \qquad J = 0.0475 \text{ kg m}^2, \qquad r = 0.25 \text{ m}, \qquad g = 9.8 \text{ m/s}^2, \qquad c = 0.05 \text{ Ns/m},$$

which corresponds roughly to the values for the Caltech ducted fan flight control testbed.

We wish to generate an optimal trajectory for the system that corresponds to moving the system for an initial hovering position to a hovering position one meter to the right $(x_f = x_0 + 1)$.

For each of the parts below, you should solve for the optimal input, simulate the (open loop) system, and plot the xy trajectory of the system, along with the angle θ and inputs F_1 and F_2 over the time interval. In addition, create a time and record the following information for each approach:

- the computation time required;

- the final position (for the open loop system);

- the weighted integrated cost of the input along the trajectory:

$$\int_0^T \left(10F_1^2(\tau) + (F_2 - mg)^2(\tau) \right) d\tau. \qquad (2)$$

(a) Solve for an optimal trajectory using a quadratic cost from the final point with weights

$$Q_x = \text{diag}([1, 1, 10, 0, 0, 0]), \qquad Q_u = \text{diag}([10, 1]).$$

This cost function attempts to minimize the angular deviation θ and the sideways force F_1.

(b) Re-solve the problem using Bezier curves as the basis functions for the inputs. This should give you smoother inputs and a nicer response.

(c) Re-solve the problem using a terminal cost $V(x(T)) = x(T)^\mathsf{T} P_1 x(T)$ to try to get the system closer to the final value. You should try adjusting the cost along the trajectory Q_x versus the terminal cost P_1 to minimize the weighted integrated cost (2).

(d) Re-solve the problem using a terminal *constraint* to try to get the system closer to the final value. Adjust the cost along the trajectory to try to minimize the cost in equation (2). (Hint: you may have to use an initial guess to get the optimization to converge.)

(e) If $c = 0$, it can be shown that this system is differentially flat (see Example 2.4). Setting $c = 0$, re-solve the optimization problem using differential flatness. (The flatness mappings can be found in the file pvtol.py, available on the companion website.)