CALIFORNIA INSTITUTE OF TECHNOLOGY
Computing and Mathematical Sciences

CDS 110

Eric Mazumdar — **Problem Set #5** — Issued: 29 Oct 2024
Fall 2024 — Due: 05 Nov 2024

Problem 1. Trajectory Tracking (60 pts)
In this problem, you will apply the concepts learned in class to develop a trajectory tracking controller for a car. The problem is divided into two parts: Part 1 will be tackled in Problem Set 5, with the rest given in Set 6.

Often in practice, we find that the systems that we are trying to control or analyze are extremely complicated with lots of moving parts and consequently a very high order (number of states). In this problem, we will be looking at controlling a car which–if we considered all the possible moving parts–would have way too many states to reason about. Therefore, instead of using the full car dynamics, we will use what's called a *reduced order model* of the car which only considers the most pertinent parts of the dynamics. In this case, we'll be using the classic *bicycle model* where we disregard the car's width, but we have the same steering and driving dynamics, as given below. Let $p_x^{\mathcal{I}}$ and $p_y^{\mathcal{I}}$ be the 2 dimensional positions of the car in the inertial frame ($\mathcal{I}$) and θ the orientation of the robot. Let $v_x^{\mathcal{B}}$ and $v_y^{\mathcal{B}}$ be the linear velocities in the body frame ($\mathcal{B}$) and ω the angular velocity around the vertical z-axis of the $\mathcal{B}$ frame. The dynamic model can be expressed as:

$$
\begin{aligned}
\dot{p}_x^{\mathcal{I}} &= v_x^{\mathcal{B}} \cos\theta - v_y^{\mathcal{B}} \sin\theta \\
\dot{p}_y^{\mathcal{I}} &= v_x^{\mathcal{B}} \sin\theta + v_y^{\mathcal{B}} \cos\theta \\
m\left(\dot{v}_x^{\mathcal{B}} - \omega v_y^{\mathcal{B}}\right) &= F_{xr} + F_{xf}\cos\left(u_\delta\right) - F_{yf}\sin\left(u_\delta\right), \\
m\left(\dot{v}_y^{\mathcal{B}} + \omega v_x^{\mathcal{B}}\right) &= F_{yr} + F_{xf}\sin\left(u_\delta\right) + F_{yf}\cos\left(u_\delta\right), \\
I_z\dot{\omega} &= \frac{L}{2}F_{xf}\sin\left(u_\delta\right) + \frac{L}{2}F_{yf}\cos\left(u_\delta\right) - \frac{L}{2}F_{yr},
\end{aligned}
\tag{1}
$$

where L is the wheelbase length, F_{xf} and F_{xr} are the front and rear tire forward forces, m is the vehicle mass, I_z is the vehicle inertia about the vertical axis intersecting the center of mass, and the lateral forces are $F_{yf} \approx C_y\alpha_f, F_{yr} \approx C_y\alpha_r$, where C_y is the tire cornering stiffness, α_r and α_f are two tire slip angles, and u_δ is the steering angle.

Note: designing a controller for such a system can be complicated because of nonlinearities. Oftentimes, a linearized model for the velocity is employed, by separating the longitudinal controller of the car from the lateral controller, and modelling the forward velocity as a first order time delay system, with τ the time delay coefficient.

$$
\begin{aligned}
\dot{p}_x^{\mathcal{I}} &= v_x^{\mathcal{B}} \cos\theta - v_y^{\mathcal{B}} \sin\theta \\
\dot{p}_y^{\mathcal{I}} &= v_x^{\mathcal{B}} \sin\theta + v_y^{\mathcal{B}} \cos\theta \\
\dot{v}_x^{\mathcal{B}} &= -\frac{1}{\tau}v_x^{\mathcal{B}} + \frac{1}{\tau}u_v \\
\dot{v}_y^{\mathcal{B}} &= -\frac{C_y}{m v_x^{\mathcal{B}}}v_y^{\mathcal{B}} - v_x^{\mathcal{B}}\omega + \frac{C_y}{m}u_\delta \\
\dot{\omega} &= -\frac{L^2 C_y}{2 I_z v_x^{\mathcal{B}}}\omega + \frac{L C_y}{2 I_z}u_\delta
\end{aligned}
\tag{2}
$$

Solution. See: `https://github.com/lupusorina/cds_110_hw5` for the code.

Task 0: Make a drawing of the system and mark the important variables from the problem statement (i.e., velocities, angles, etc)

Solution. See Fig. 1.

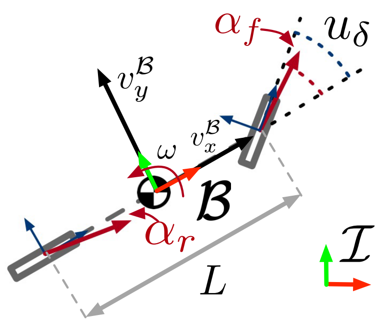

Figure 1: Model

Task 1: Simulate the equations of motion in (2).
Start with an initial condition: $[p_x^{\mathcal{I}} = 0, p_y^{\mathcal{I}} = 0, \theta = 0, v_x^{\mathcal{B}} = 1.0, v_y^{\mathcal{B}} = 0, \omega = 0]$. Input different values for u_δ (steering input) and u_v (velocity input) and plot the system's behaviour. Use the following parameters $m = 11.5$ kg, $L = 0.4$ m, $I_z = 0.5$ kg m^2, $C_y = 100$ N/rad, $\tau = 0.1$ s. With a nonzero control input of choice, plot the states (do not forget to specify what control input was used). Analyze the resulting plots to assess whether the simulated behavior matches with a real car and your intuition.
You can use the systems.py file in `https://github.com/lupusorina/cds_110_hw5` as a starting point.

Solution. See Fig. 2.

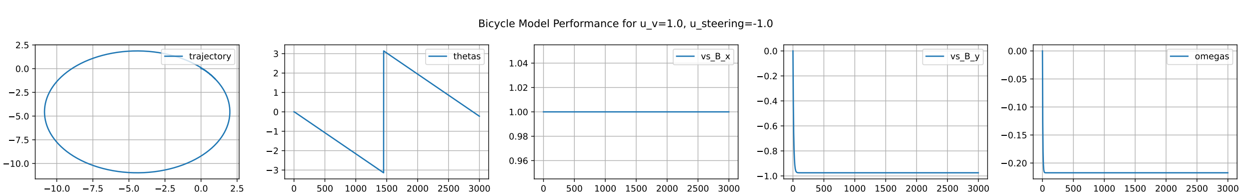

Figure 2: Performance

Task 2: (Longitudinal control) The linear velocity in the x-direction is modelled as

$$\dot{v}_x^{\mathcal{B}} = -\frac{1}{\tau} v_x^{\mathcal{B}} + \frac{1}{\tau} u_v,$$

with τ being a time-constant (i.e., 0.1 s). Design a linear controller for this system and show convergence using Lyapunov theory.

Solution. We write the 1D differential equation as

$$\dot{x}(t) = ax(t) + bu,$$

and try the controller

$$u_v = -b^{-1}(k_p(x - x_d) + ax_d).$$

with $k_p > 0$. The closed-loop system becomes

$$\dot{x} = ax - (k_p(x - x_d) + ax_d).$$

Note the error $e = x - x_d$. Using this notation, the closed-loop system is

$$\dot{e} = ae - k_p e.$$

We pick a Lyapunov function as

$$V = 1/2e^2$$

and compute its derivative.

$$\begin{aligned} \dot{V} &= e\dot{e} \\ &= e(ae - k_p e) \\ &= ae^2 - k_p e^2 \\ &= (a - k_p)e^2 < 0. \end{aligned}$$

Therefore, the velocity error converges exponentially to 0. $(e \to 0)$

Task 3: Test this controller on your car simulator going in a straight line. Plot the tracking performance. Include the desired regulation point for the velocity.

Solution. See Fig. 3.

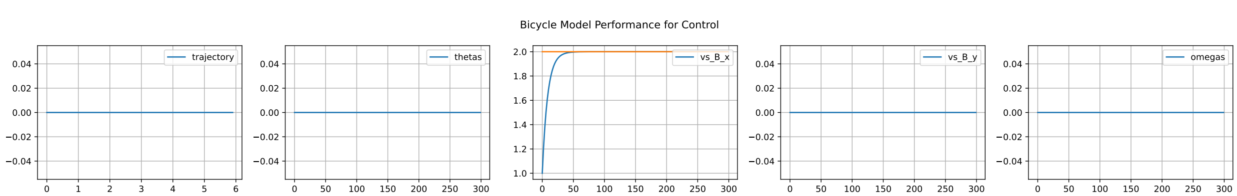

Figure 3: Performance

Task 4: Experiment with different trajectories (circular, figure 8) from the Github link. Plot the variables such as position and velocity in the inertial frame, as well as the desired velocities transformed in the body frame. Note: these trajectories will be used for the second part of the problem set.
`https://github.com/lupusorina/cds_110_hw5`.

Solution. See Fig. 4 and Fig. 5.

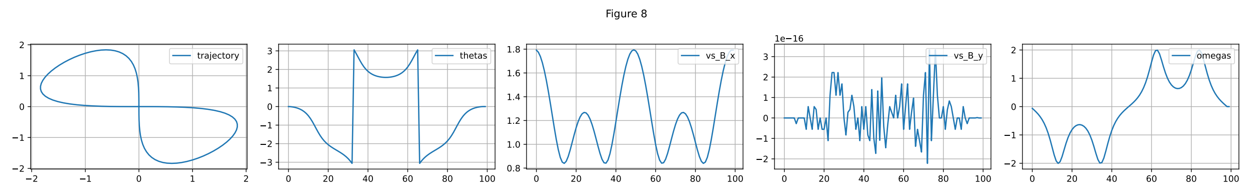

Figure 4: Figure 8

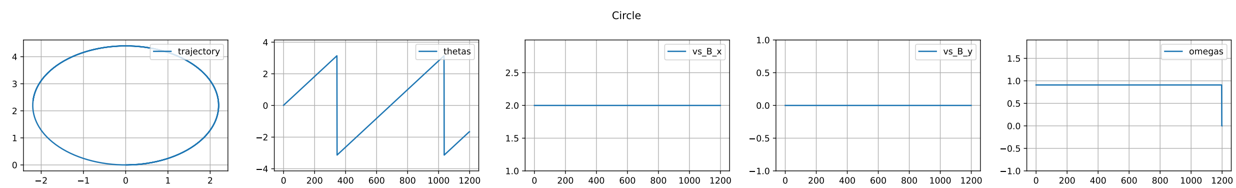

Figure 5: Circle

Problem 2. **(40 pts)** Consider the double integrator

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u = Ax + Bu$$

Find a state feedback that minimizes the quadratic cost function

$$J = \int_0^\infty \left(q_1 x_1^2 + q_2 x_2^2 + q_u u^2\right) dt$$

where $q_1 \geq 0$ is the penalty on position, $q_2 \geq 0$ is the penalty on velocity, and $q_u > 0$ is the penalty on control actions.
Task 1: Analyze the coefficients of the closed loop characteristic polynomial and explore how they depend on the penalties.
Hint: Compare the closed loop characteristic polynomial with the standard second-order polynomial $s^2 + 2\zeta_0\omega_0 s + \omega_0^2$.
Task 2: Implement the LQR in Python and verify that **K** obtained analytically is the same as **K** obtained in Task 1. You can use numerical values of choice.

Solution. Solution. The solution is given by the algebraic Riccati equation (7.31 FBS) which in this case can be solved analytically. We have

$$Q_x = \begin{pmatrix} q_1 & 0 \\ 0 & q_2 \end{pmatrix}, \quad Q_u = q_u$$

Furthermore

$$PA = \begin{pmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & p_{11} \\ 0 & p_{12} \end{pmatrix}$$
$$PB = \begin{pmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} p_{12} \\ p_{22} \end{pmatrix}$$

and the elements of the algebraic Riccati equation becomes

$$\begin{aligned} 11: &\quad 0 - \tfrac{p_{12}^2}{q_u} + q_1 = 0 \\ 12: &\quad p_{11} - \tfrac{p_{12}p_{22}}{q_u} = 0 \\ 22: &\quad 2p_{12} - \tfrac{p_{22}^2}{q_u} + q_2 = 0 \end{aligned}$$

The positive definite solution is

$$p_{11} = \sqrt{q_1 q_2 + 2q_1\sqrt{q_1 q_u}}, \quad p_{12} = \sqrt{q_1 q_u}, \quad p_{22} = \sqrt{q_2 q_u + 2q_u\sqrt{q_1 q_u}},$$

and the controller gains are

$$K = Q_u^{-1} B^T P = \frac{1}{q_u} \begin{pmatrix} p_{12} & p_{22} \end{pmatrix} = \begin{pmatrix} \sqrt{q_1/q_u} & \sqrt{q_2/q_u + 2\sqrt{q_1/q_u}} \end{pmatrix}$$

The closed loop characteristic polynomial is

$$\begin{aligned} \det(sI - A + BK) &= \det \begin{pmatrix} s & -1 \\ k_1 & s + k_2 \end{pmatrix} = s^2 + k_2 s + k_1 \\ &= s^2 + s\sqrt{q_2/q_u + 2\sqrt{q_1/q_u}} + \sqrt{q_1/q_u} \end{aligned}$$

Comparing this with the standard second-order polynomial $s^2 + 2\zeta_0\omega_0 s + \omega_0^2$ we find that

$$q_1 = q_u \omega_0^4, \quad q_2 = 2q_u \left(2\zeta_0^2 - 1\right) \omega_0^2$$

Notice that:

- The controller gains depend on the ratios q_1/q_u and q_2/q_u.

- The weight q_1 is proportional to the fourth power of the frequency ω_0 which implies that large changes in q_1 are required to obtain significant changes in response speed.
- The weight q_2 is proportional to the second power of the frequency ω_0.
- Critical damping $\left(\zeta_0 = \sqrt{2}/2\right)$ is obtained for $q_2 = 0$.

It is generally true that the gains only depend on the relative magnitudes of Q_x and Q_u and that large changes (orders of magnitude) of the weights are required to obtain significant changes in system performance. It is also true that changing weights on states associated with position influences response speed while those corresponding to velocity influences damping.