NLSC Quadcopter Project

Task 2: Feedback Linearization

Due Date: May 19th, 2023 @ 11:59PM

Instructions:

- Check the Introduction file that specifies the system model.
- Use the Q&A forum in Moodle to ask questions.
- The submission portal will be available in Moodle. It will automatically be closed at the specified deadline.
- Submissions will only be considered if submitted before the deadline.
- Use the filenames for deliverables as defined below.
- If you choose to deliver digitized handwritten notes, make sure they are clearly visible.

---

2 Feedback Linearization

2.1 Goal

The goal of this task is to apply feedback linearization to effectively control the yaw dynamics of a quadcopter.

2.2 Problem setup

The equations of motion of a quadrotor have nonlinearities arising due to to the kinematics and dynamics. Feedback linearization is a technique which aims to find a coordinate transformation in which the input-output map can be described by linear equations. Although such a design is feasible for the full quadcopter model, it requires the usage of artificial inputs [1] [2]. Instead, in this task of the project, a simpler setup is considered where the position of the quadcopter along $x - y - z$ axes are controlled by an LQR controller (which is given to you), and the yaw dynamics are controlled using feedback linearization (which you are supposed to design).

Recall that the dynamics of the quadcopter are given by the equations

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} \frac{1}{m}\left[(\cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi))u_1 - k_x\dot{x}\right] \\ \frac{1}{m}\left[(\cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi))u_1 - k_y\dot{y}\right] \\ \frac{1}{m}\left[\cos\phi\cos\theta u_1 - mg - k_z\dot{z}\right] \end{bmatrix}, \tag{1}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{1}{I_x}\left[(I_y - I_z)qr + u_2 - k_p p\right] \\ \frac{1}{I_y}\left[(I_z - I_x)pr + u_3 - k_q q\right] \\ \frac{1}{I_z}\left[(I_x - I_y)pq + u_4 - k_r r\right] \end{bmatrix}, \tag{2}$$

and the Euler rate vector $\dot{\Theta} = \begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^\top$ is related to the angular velocity according to

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{3}$$

The full dynamics of the quadcopter are provided to you in the `Introduction.pdf` file. As seen in the linear control task, at any equilibrium point, the linearized dynamics of x, y, z positions and the yaw angle ψ are completely decoupled at any fixed ψ. Because a given LQR controller is used to control the x, y and z positions, the inputs u_1, u_2 and u_3 are already available. For this exercise, the decoupled controller for the x, y and z positions is provided to you in the `Simulink` file (in the block `xyzController`). You are required to design a feedback linearization based controller for the yaw dynamics using the state measurements and the outputs of the block `xyzController`.

2.3 Analysis and controller design

Task 2.a: Using ψ as the output and u_4 as the input, compute the relative degree of the system. Is the system input output linearizable from the input u_4 to the output $y = \psi$?

Task 2.b: Given a reference signal $\psi_{\text{ref}}(t)$, design an asymptotically stable tracking controller for yaw control using input-output linearization from the input u_4 to output $y = \psi$. For the design, you can assume that the derivatives $\dot{\psi}_{\text{ref}}(t)$, $\ddot{\psi}_{\text{ref}}(t)$ are available to you online, and are bounded. Explain the design procedure.

2.4 Control implementation and tuning

Task 2.c: Implement the designed controller in `Simulink`. You should only modify the `yawController` sub-block from the template, within the `Feedback Linearization` block. Note that a constant reference is chosen here, so $\dot{\psi}_{\text{ref}}(t)$, $\ddot{\psi}_{\text{ref}}(t)$ are known to be 0.

Task 2.d: Tune your controller such that the maximum overshoot is less than 5% and settling time (to within 4% of a step change) is less than 5 seconds. Simulate the time evolution of the system for the given initial condition and reference setpoint, and plot the orientation ψ, θ, ϕ as a function of time. In this plot, indicate the overshoot and settling time achieved by your controller, highlighting the percentage values using annotations.

2.5 Deliverables

- `task2abd.pdf` file with the answers to Tasks 2.a, 2.b and 2.d. Digitized handwritten notes are accepted.
- `task2c.m` `Matlab` file with the controller implemented in Task 2.c. This file should only contain a `Matlab` function which uses the same inputs and outputs as in the `yawController` block of template provided to you.

References

[1] A. Benallegue, A. Mokhtari, and L. Fridman, "Feedback linearization and high order sliding mode observer for a quadrotor uav," in *International Workshop on Variable Structure Systems, 2006. VSS'06.* IEEE, 2006, pp. 365–372.

[2] V. Mistler, A. Benallegue, and N. M'sirdi, "Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback," in *Proceedings 10th IEEE international workshop on robot and human interactive communication. Roman 2001 (Cat. no. 01th8591).* IEEE, 2001, pp. 586–593.