Optimal Control

Course Project #2
Optimal Control of a Supersonic Aircraft

November 30, 2023

This project deals with the design and implementation of an optimal control law for a supersonic aircraft with nonlinear drag and lift.

The model is schematically represented in Figure 1.

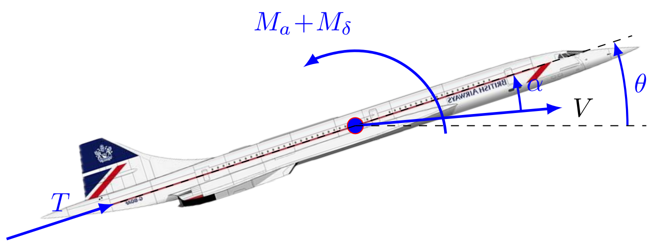

Figure 1: Supersonic Aircraft

The state space consist in $\mathrm{x} = (V, \alpha, \theta, q)$, where V is the longitudinal speed of the aircraft, α is the angle of attack, θ is the pitch and q is the pitch rate. The input is $\mathrm{u} = (\delta_T, \delta_c, \delta_e)$, i.e., the throttle, a canard and an elevator. The model is:

$$
\begin{aligned}
m\dot{V} &= T(V, \delta_T)\cos(\alpha) - D(V) - mg\sin(\theta - \alpha) \\
\dot{\alpha} &= q - \frac{1}{mV}\big(T(V, \delta_T)\sin(\alpha) + L(V) + L_\delta(V, \delta_c, \delta_e) - mg\cos(\theta - \alpha)\big) \\
\dot{\theta} &= q \\
J\dot{q} &= M_a(V) + M_\delta(V, \delta_c, \delta_e).
\end{aligned}
$$

The expressions for the lift $L(V)$, the drag $D(V)$ and the pitching moment $M_a(V)$ are

$$L(V) = \frac{1}{2} C_L \rho V^2$$
$$D(V) = \frac{1}{2} C_D \rho V^2$$
$$M_a(V) = \frac{1}{2} C_M \rho V^2$$

At last, the expressions for the control thrust $T(V, \delta_T)$, the control lift $L_\delta(V, \delta_c, \delta_e)$ and the control moment $M_\delta(V, \delta_c, \delta_e)$ are:

$$T = \frac{1}{2} \rho V^2 C_T \delta_T$$
$$\begin{bmatrix} L_\delta \\ M_\delta \end{bmatrix} = \frac{1}{2} \rho V^2 B \begin{bmatrix} \delta_c \\ \delta_e \end{bmatrix}.$$

All the parameters of the aircraft are available in table 1.

Parameters:	
m	26.82
J	595.9
g	32.17
ρ	0.0011
C_L	0.5
C_D	1.59
C_M	0.5
C_T	3.75
B	$\begin{bmatrix} -3.575 & 0.065 \\ -1.3 & 6.5 \end{bmatrix}$

Table 1: Model parameters.

Hint: For the definition of a meaningful reference trajectory/curve try to keep the velocity V such that its Mach number $M_\infty \in [0.5, 2]$, where $M_\infty = V/a_0$, with $a_0 = 1016$ [ft/s]. Moreover, try to keep $\alpha \in [-10, 10]$.

Task 0 – Problem setup

Discretize the dynamics, write the discrete-time state-space equations and code the `dynamics` function.

Task 1 – Trajectory generation (I)

Compute two equilibria for your system and define a reference curve between the two. Compute the optimal transition to move from one equilibrium to another exploiting the Newton's-like algorithm for optimal control.
Hint: you can exploit any numerical root-finding routine to compute the equilibria.
Hint: define two long constant parts between the two equilibria with a transition in between. Try to keep everything as symmetric as possible, see, e.g., Figure 2.
Hint: for the definition of the reference equilibria, try to parametrize the equilibrium with respect to some state variables, e.g., fix $\alpha \approx 0$, θ and V and find the corresponding inputs and other states.

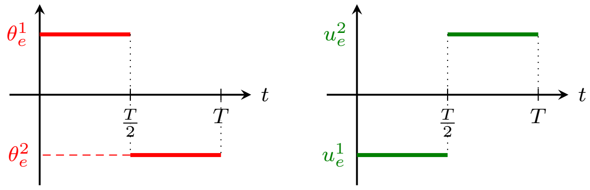

Figure 2: Example of possible desired trajectory for a pendulum system.

Hint: to visualize the results, you can integrate the aircraft position x–y plane, following the dynamics

$$\begin{aligned} \dot{x} &= V\cos(\theta - \alpha) && \text{horizontal position} \\ \dot{y} &= V\sin(\theta - \alpha) && \text{vertical position.} \end{aligned}$$

Task 2 – Trajectory generation (II)

Generate a desired (smooth) state-input curve and perform the trajectory generation task (Task 1) on this new desired trajectory. *Hint:* for the definition of the reference curve, try to parametrize the equilibrium with respect to same state variables, e.g., fix $\alpha \approx 0$, θ and V and find the corresponding inputs and other states. As an example, you may fix $\alpha(t) \equiv 0$, for all t, and vary θ and V.
Hint: as initial guess you may need to compute a quasi-static trajectory, i.e., a collection of equilibria, and generate the first trajectory by tracking this quasi-static trajectory via the feedback matrix solution of an LQR problem computed on the linearization of the system about the quasi-static trajectory with a user-defined cost.

Task 3 – Trajectory tracking via LQR

Linearizing the vehicle dynamics about the (optimal) trajectory $(\mathbf{x}^{\text{opt}}, \mathbf{u}^{\text{opt}})$ computed in Task 2, exploit the LQR algorithm to define the optimal feedback controller to track this reference trajectory. In particular, you need to solve the LQ Problem

$$\begin{aligned} \min_{\substack{\Delta x_1,\dots,\Delta x_T \\ \Delta u_0,\dots,\Delta u_{T-1}}} \quad & \sum_{t=0}^{T-1} \Delta x_t^\top Q^{\text{reg}} \Delta x_t + \Delta u_t^\top R^{\text{reg}} \Delta u_t + \Delta x_T^\top Q_T^{\text{reg}} \Delta x_T \\ \text{subj.to} \quad & \Delta x_{t+1} = A_t^{\text{opt}} \Delta x_t + B_t^{\text{opt}} \Delta u_t \qquad t = 0,\dots,T-1 \\ & x_0 = 0 \end{aligned}$$

where A_t^{opt}, B_t^{opt} represent the linearization of the (nonlinear) system about the optimal trajectory. The cost matrices of the regulator are a degree-of-freedom you have.
Hint: to showcase the tracking performances, consider a perturbed initial condition, i.e., different than x_0^{opt}.

Task 4 – Trajectory tracking via MPC

Linearizing the vehicle dynamics about the (optimal) trajectory $(\mathbf{x}^{\text{opt}}, \mathbf{u}^{\text{opt}})$ computed in Task 2, exploit an MPC algorithm to track this reference trajectory.
Hint: to showcase the tracking performances, consider a perturbed initial condition, i.e., different than x_0^{opt}.

Task 5 – Animation

Produce a simple animation of the vehicle executing Task 3. You can use PYTHON or any other visualization tool.
Hint: to visualize the results, you can integrate the aircraft position x–y plane, following the dynamics

$$\begin{aligned} \dot{x} &= V\cos(\theta) \qquad \text{horizontal position} \\ \dot{y} &= V\sin(\theta) \qquad \text{vertical position.} \end{aligned}$$

Required plots

For Tasks 1-2, you are required to attach to the report the following plots

- Optimal trajectory and desired curve.
- Optimal trajectory, desired curve and few intermediate trajectories.
- Armijo descent direction plot (at least of few initial and final iterations).
- Norm of the descent direction along iterations (semi-logarithmic scale).
- Cost along iterations (semi-logarithmic scale).

For the other tasks, you are required to attach to the report the following plots

- System trajectory and desired (optimal) trajectory.
- Tracking error for different initial conditions.

Guidelines and Hints

- As optimization algorithm, you can use the (regularized) Newton's method for optimal control introduced during the lectures based on the Hessians of the cost only.
- In the definition of the desired curve, you may try to calculate the desired trajectories using a simplified model, e.g., a simplified kinematic model.

Notes

1. Each group must be composed of 3 students (except for exceptional cases to be discussed with the instructor).
2. Any other information and material necessary for the project development will be given during project meetings.
3. The project report must be written in LaTeX and follow the main structure of the attached template.
4. Any email for project support must have the subject:
   "[OPTCON2023]-Group X: rest of the subject".
5. **All** the emails exchanged **must be cc-ed** to professor Notarstefano, dr. Sforni and the other group members.

IMPORTANT: Instructions for the Final Submission

1. The final submission **deadline** is **one** week before the exam date.

2. One member of each group must send an email with subject "[OPTCON2023]-Group X: Submission", with attached a link to a OneDrive folder shared with professor Notarstefano, dr. Sforni and the other group members.

3. The final submission folder must contain:

   - `report_group_XX.pdf`
   - `report` – a folder containing the LaTeX code and `figs` folder (if any)
   - `code` – a folder containing the code, including `README.txt`