MAE 3260

Pitch Perfect

Aircraft Pitch Control Linearization

Michelle Mobius, Audrey Garon, Parker McLaughlin, Kyle Dattner

December 2025

Outline

Abstract

We will be looking into an aircraft pitch control model. We will specify our own design requirements based on existing aircrafts. We will linearize the system by simplifying the nonlinear equations of motion into linear equations of motion. This can be done based on a steady flight assumption (straight and level cruise) that assumes there is not much disturbance in the system: i.e. linear and angular velocities are constant. We plan to focus specifically on pitch control in response to disturbances to the system. This is our topic of choice because it builds on the knowledge we have gained so far in the course, while challenging ourselves to apply it to a system we have not yet focused on. We are curious about aircraft in particular because the models for both the system and disturbances are more complex than other systems that we have worked with in the past.

Students/Roles

Student	Task/Role
Michelle Mobius	Michelle chose the simplifying assumptions for the model and derived the model and design requirements with the rest of the group. She created the equations of motion and also graphed the system response. She wrote code for the MATLAB script to generate the Bode plots, poles of the system, impulse response, and step response. She wrote analysis on the space model, requirements, model simplifications, and equations of motion. She helped format the final report.
Audrey Garon	Audrey chose the simplifying assumptions for the model and derived the model and design requirements with the rest of the group. She created the equations of motion and also graphed the system response. She wrote code for the MATLAB analysis scripts to generate plots (Bode, impulse, step, poles of the system). She also formatted the report and added figures and references. In the report, she also wrote analysis paragraphs about the Bode plots and impulse response graph.
Parker McLaughlin	Parker researched various models for aircraft longitudinal stability. She chose the simplifying assumptions for the model and helped verify the correctness of the model with the rest of the group. She derived the transfer function, drew the block diagram, and typed up and formatted equations for the report.
Kyle Dattner	Kyle chose the simplifying assumptions for the model and derived the model and design requirements with the rest of the group. Kyle did research on modeling the disturbance and made the Bode plots for the closed-loop system.

List of MAE 3260 concepts or skills used in this group work:

- ODEs
- TFs
- State space
- Block diagrams
- Feedback control law
- Disturbance rejection

Design Requirements

We designed our aircraft pitch control model based on the flight specifications and requirements of a standard Airbus A320. We specifically chose the Airbus A320 because it is the best-selling commercial aircraft, making it a highly reliable platform to analyze with millions of flight miles and relatively easily available specifications.

Our feedback controller should overshoot less than 25%, have a rise time less than 8 seconds, settling time less than 30 seconds, steady state error less than 5% and maximum acceleration of 3.5 m/s^2 [2].

Model Simplifications

We are assuming that the aircraft is in longitudinal equilibrium, operating at steady cruise and velocity. We also assume small perturbations about an equilibrium; this enables us to use a Taylor expansion to linearize the equations of motion of the system. We assume that the angle of attack is nearly constant and the lift generated by the wings is approximately equal to the weight of the aircraft.

Airbus A320 Specifications:

- α = cruise angle of attack = 3.49°
- C_L = coefficient of lift = 0.15
- C_D = coefficient of drag = 0.0324
- C_M = coefficient of pitch moment = -0.3
- C_W = coefficient of weight = 0.15 (equal to C_L due to steady flight)
- i_{yy} = normalized moment of inertia = 2,500,630
- ρ = air density = 1.225 kg/m^3
- S = wing area = 122.4 m^2
- m = maximum takeoff weight = 68000 kg
- $\bar{c}$ = average chord length = 0.75 m
- $\mu = \frac{\rho S \bar{c}}{4m}$ = 0.0004134375 Pa*s
- $\Omega = \frac{2U}{\bar{c}}$

The constants above were taken from [1].

Equations of Motion

We linearized the equations of motion of the system [3] to be decoupled into longitudinal and lateral equations. We based this model on a similar study done at the University of Michigan to linearize the equations of motion of a Boeing aircraft [2]. The generalized equations of motion are:

$$\dot{\alpha} = \mu\Omega\sigma\left[-\ (C_L + C_D)\alpha + \frac{1}{(\mu - C_L)}q\ -\ (C_W sin\gamma)\delta\ +\ C_L\right]$$

$$\dot{q} = \frac{\mu\Omega}{2i_{yy}}\left[\left[C_M -\ \eta(C_L + C_D)\alpha\right] + \left[C_M +\ \sigma C_M(1\ -\ \mu C_L)\right]q\ +\ (\eta C_W sin\gamma)\delta\right]$$

$$\dot{\theta} =\ \Omega q$$

Based on the Airbus A320 specifications, we derive the following quantities:

$$\mu\ = \frac{\rho S\bar{c}}{4m} =\ 0.0004134375\ Pa\ \cdot\ s$$

$$\Omega\ = \frac{2U}{\bar{c}} =\ 644.4453\ Hz$$

$$\sigma\ = \frac{1}{1+\mu C_L}\ =\ 0.9994$$

$$\eta\ =\ \mu\sigma C_M =-\ 0.0001239$$

Plugging in these variables and aircraft specifications to our equations of motion, we find the equations of motion for the Airbus A320:

$$\dot{\alpha} =-\ 0.045\alpha\ -\ 2.95q\ -\ 0.055u$$

$$\dot{q} =\ 3.1\alpha\ -\ 2.85q\ +\ 1.3u$$

$$\dot{\theta} =\ 644.445q$$

State Space Model

We then developed the state space model of the aircraft pitch control system based on the system's linearized differential equations [3]. We identified the three essential state variables: angle of attack (α), pitch angle (θ), pitch rate (q) to create our state space matrices to solve the governing equations. The angle of attack is critical to our model because it describes the angle between the wing's chord length and direction of the airflow. The pitch rate tells us the rate of change of the pitch angle and the pitch angle is the positional telemetry of the aircraft we want to control.

Our state space model is of the canonical form: $\dot{x}$= Ax+Bu and y=Cx+Du [2]. The state represents the internal condition of the system, the inputs are the physical variables of the system, and the eigenvalues of the matrix A of the system are the n poles of the system.

Here is our state space model:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -0.045 & -2.95 & 0 \\ 3.1 & -2.85 & 0 \\ 0 & 644.445 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} -0.055 \\ 1.3 \\ 0 \end{bmatrix} [\delta]$$

$$y = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \theta \end{bmatrix}$$

Figure 1: State space model of an Airbus A320 pitch control model in steady cruise and at constant velocity.

Transfer Functions

In order to simulate our system, we also needed to derive the transfer function. First, we write our equations of motion in a general form, where a, b, c, e, f, g, h are constant coefficients. We use these variables to simplify our derivation. Next, we convert to the Laplace domain:

$$\text{Laplace}[\dot{\alpha} =- A\alpha + Bq + C\delta, \dot{q} =- E\alpha - Fq + G\delta, \dot{\theta} = Hq]$$
$$sA(s) =- aA(s) + bQ(s) + c\Delta(s) \text{ (1)}$$
$$sQ(s) =- eA(s) - fQ(s) + g\Delta(s) \text{ (2)}$$
$$s\Theta(s) = hQ(s) \text{ (3)}$$

We now plug the three transfer functions into each other to get a transfer function in terms of θ, the pitch angle, and δ, the trim angle.

$$\text{From (1): } sA(s) + aA(s) = bQ(s) + c\Delta(s)$$
$$A(s) = \frac{bQ(s)+c\Delta(s)}{s+a}$$
$$\text{From (2): } sQ(s) + fQ(s) =- eA(s) + g\Delta(s)$$
$$Q(s) = \frac{-eA(s)+g\Delta(s)}{s+f}$$
$$\text{Plug (1) into (2): } Q(s)(s + f) =- e(\frac{bQ(s)+c\Delta(s)}{s+a}) + g\Delta(s)$$
$$Q(s) = \frac{g(s+a)-ec}{(s+f)(s+a)+eb}\Delta(s)$$
$$\text{Plug into (3): } s\Theta(s) = hQ(s)$$
$$\Theta(s) = \frac{h}{s} \cdot \frac{g(s+a)-ec}{(s+f)(s+a)+eb}\Delta(s)$$
$$\Theta(s) = \frac{hg(s+a)-hec}{s(s+f)(s+a)+seb}\Delta(s)$$
$$\Theta(s) = \frac{(hg)s+(hga-hec)}{s^3+(a+f)s^2+(af+eb)s}\Delta(s)$$

Finally, we plug in the values for our constant coefficients:

$$\dot{\alpha} = -A\alpha + Bq + C\delta = -0.045\alpha - 2.95q - 0.055u$$
$$\dot{q} = -E\alpha - Fq + G\delta = 3.1\alpha - 2.85q + 1.3u$$
$$\dot{\theta} = Hq = 644.445q$$
$$\Theta(s) = \frac{837.78s+134.05}{s^3+2.895s^2-7.891s}\Delta(s)$$

Bode Plots: Open Loop System

Using MATLAB, we create a magnitude and phase plot for our system.

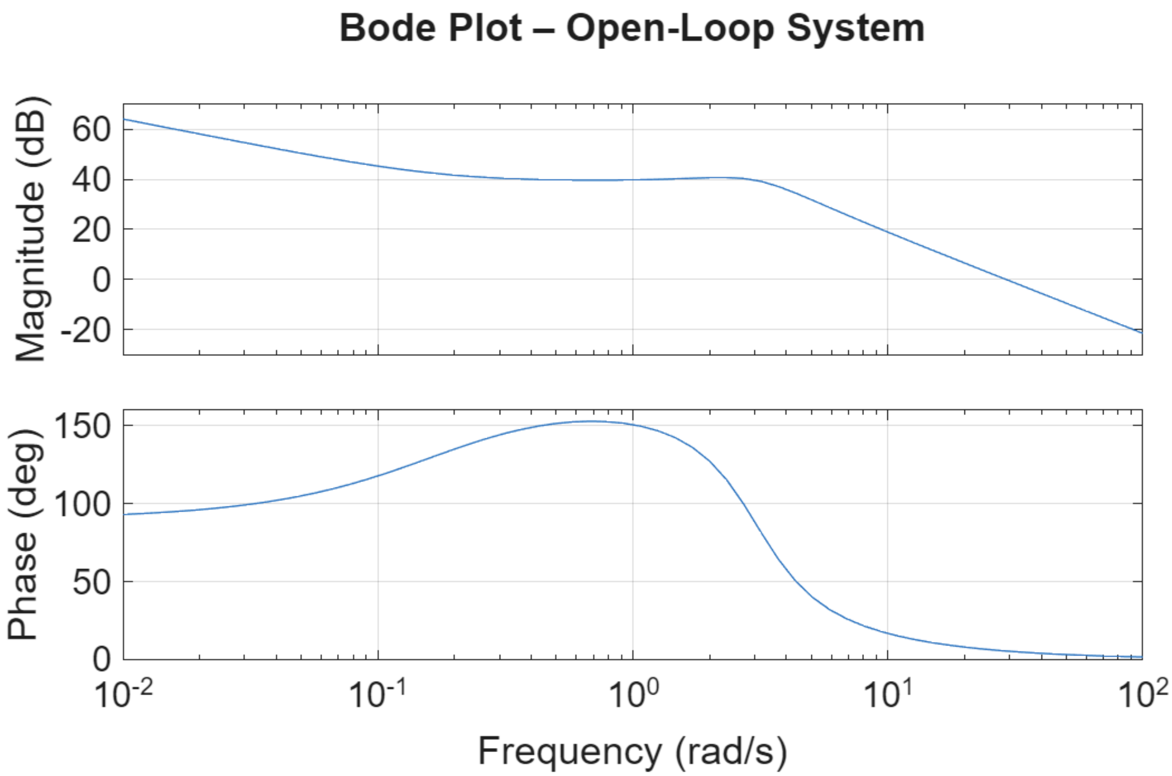

Figure 2: Bode plot of our open-loop system.

The magnitude plot (Figure 2) of the A320 pitch-angle system shows that the aircraft has very high low-frequency gain, starting around 60 dB, which means that even a small elevator input produces a large change in pitch angle. The high magnitude at low frequencies comes from the fact that the system is very sensitive to slow variations in pitch. When the input changes slowly, the aircraft's angle continues increasing, so the Bode plot shows a large gain at those frequencies. As frequency increases, the magnitude gradually decreases, and after about 4 rad/s it begins to fall at roughly –40 dB per decade, which is typical behavior for a system dominated by two poles, such as the aircraft's short-period.

The phase plot (Figure 2) starts around +90°, which simply shows that the aircraft reacts noticeably to slow pitch motions. As the frequency increases to about 1 rad/s, the phase rises toward +150°, meaning the aircraft begins to react a little earlier to changes in pitch because of its short-period motion. After this point, the phase steadily drops toward 0°, showing that at high frequencies the aircraft can't keep up and behaves more like a system that lags behind the input. The magnitude plot never crosses 0 dB, which means the open-loop system does not have a clear point where it would naturally balance gain and stability. Because of this and the limited phase margin, the system is only marginally stable. In practical terms, this explains why the aircraft responds strongly but slowly, doesn't oscillate very much, and takes a long time to settle after a disturbance, which is one of the main reasons why real aircraft use feedback control to improve pitch damping and make the airplane easier to handle.

System Poles

Using our transfer function, we calculate the following values for our system's poles using MATLAB:

$$0 + 0i, \quad -1.4475 + 2.6792i, \quad -1.4475 - 2.6792i$$

Our system has three poles, one at 0 + 0i and one complex conjugate pair (as you can see above). The pole near the origin is reflective of a true integrator and the complex conjugate pole describes a natural oscillatory motion that dampens over time.

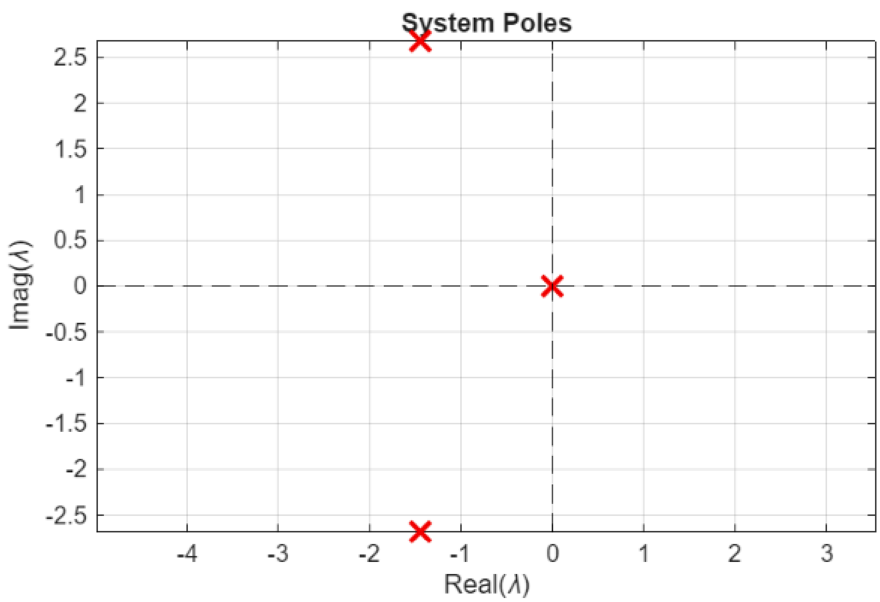

Figure 3: System poles of our Airbus A320 open-loop system.

These pole locations are as we expect for the longitudinal stability of an aircraft. Aircraft tend to show two forms of dynamic responses to a disturbance: a short-period response, which is highly damped and has a short duration, and a long-period response [4]. The long-period response is known as the "phugoid response" and has much less damping, so it persists over a longer duration.

Due to our model simplifications, we do not see the short-period response poles on our plot. However, we can see the 2 poles associated with the phugoid response in the left half of the imaginary plane (Figure 3). These poles have a low damping ratio and low frequency, as evidenced by their relatively large angle to the x-axis and their small distance from the origin. It is good that these poles are located in the left half of the imaginary plane–otherwise, the aircraft would be unstable and would have a tendency to enter a steep climb or dive upon slight disturbances in pitch.

Step Response: Open Loop System

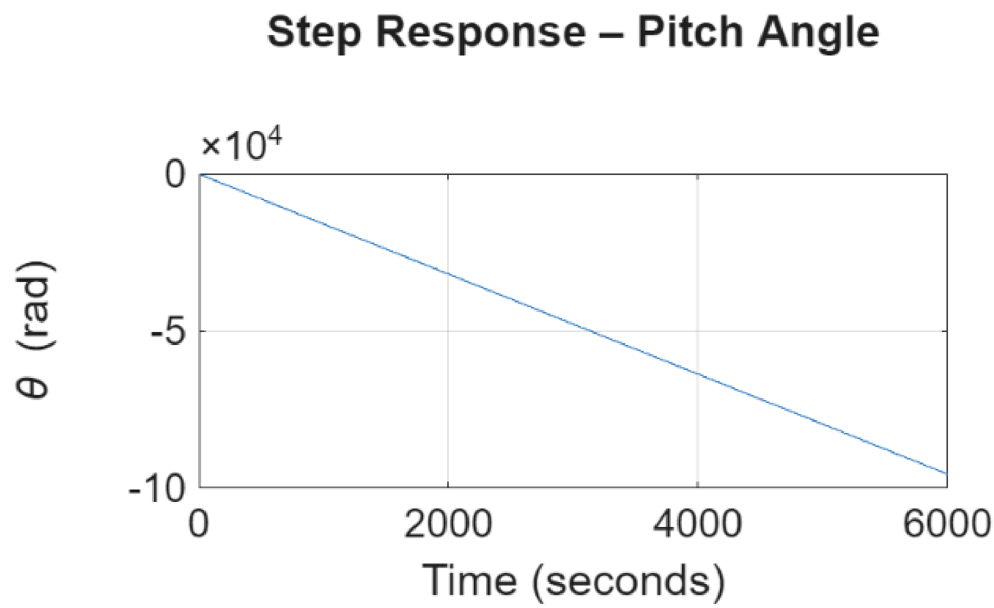

Figure 7: The Step Response of our open-loop system.

Impulse Response: Open Loop

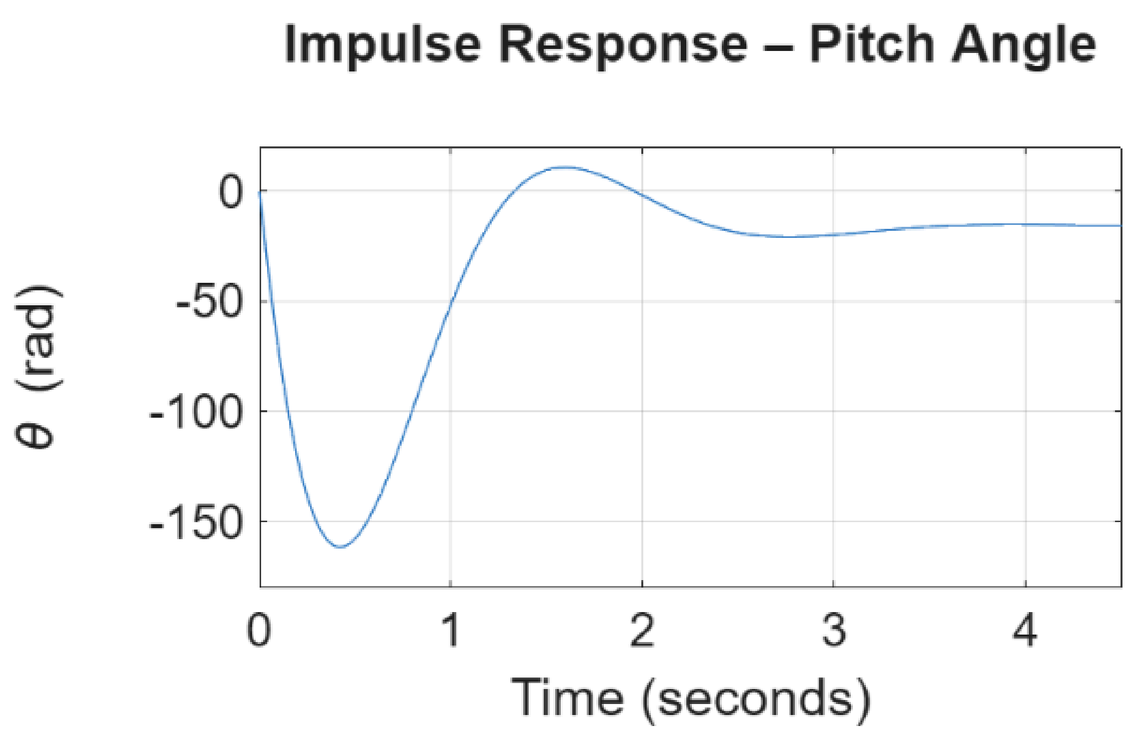

Figure 8: The Impulse Response of our open-loop system.

The impulse response shows that the aircraft's pitch angle behaves like a lightly damped second-order system. The pitch angle drops sharply right after the impulse, showing that the aircraft reacts very strongly at the start before it begins to settle out. It then swings upward past zero into a smaller overshoot before gradually settling back toward equilibrium. This oscillatory behavior means the system has complex poles with relatively low damping, so the response oscillates but still decays over time. The fact that the oscillations get smaller and eventually settle shows the system is stable, but the large first peak and noticeable overshoot suggest it is not very well-damped. Overall, the response is stable, underdamped, and dominated by its natural mode of oscillation.

Block Diagram

In order to add feedback control in our system, we now implement a PID controller. The block diagram for the closed loop system is shown below:

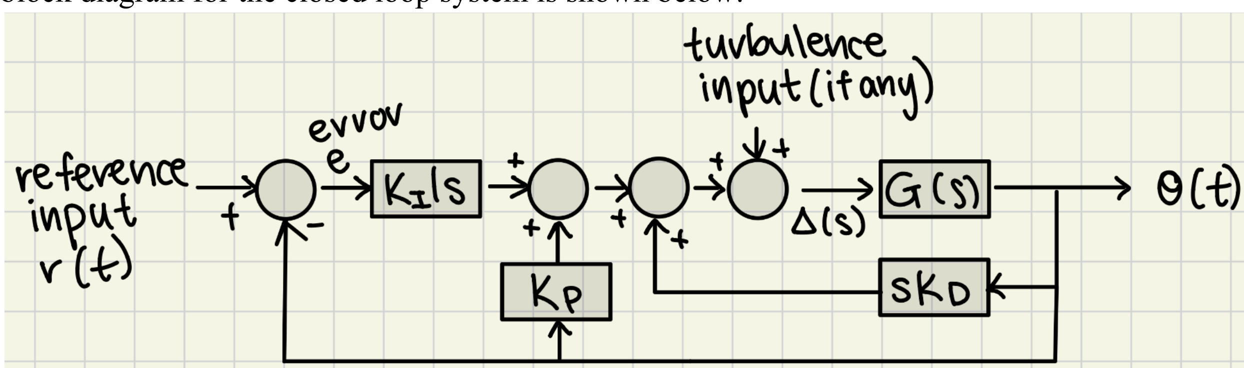

Figure 4: Block diagram of our Airbus A320 pitch control model in steady cruise and at constant velocity.

We must now re-derive the system transfer function. The transfer function for the full controller is now:

$$C(s) = K_P + \frac{K_I}{s} + sK_D$$

The input to the controller is the error between the output and the reference input:

$$E(s) = R(s) - \Theta(s)$$

We are now able to find the full loop transfer function:

$$\Theta(s) = C(s)G(s) \cdot (R(s) - \Theta(s))$$

$$\frac{\Theta(s)}{R(s)} = \frac{C(s)G(s)}{1+C(s)G(s)}$$

$$\frac{\Theta(s)}{R(s)} = \frac{\left(K_P+\frac{K_I}{s}+sK_D\right)\left(\frac{837.78s+134.05}{s^3+2.895s^2-7.891s}\right)}{1+\left(K_P+\frac{K_I}{s}+sK_D\right)\left(\frac{837.78s+134.05}{s^3+2.895s^2-7.891s}\right)}$$

$$\frac{\Theta(s)}{R(s)} = \frac{\left(K_I+s(K_D+K_P)\right)(837.78s+134.05)}{s^2\left(s^2+2.895s-7.891\right)+\left(K_I+s(K_D+K_P)\right)(837.78s+134.05)}$$

We can now use this transfer function to analyze the behavior of the system with feedback.

Bode Plots: Closed Loop System

Using MATLAB, we create a magnitude and phase plot for our closed loop system. For the sake of simplicity we inputted sample Kp, Ki, and Kd values of 5, however in an actual system we would fine tune these values to adjust for our system constraints.

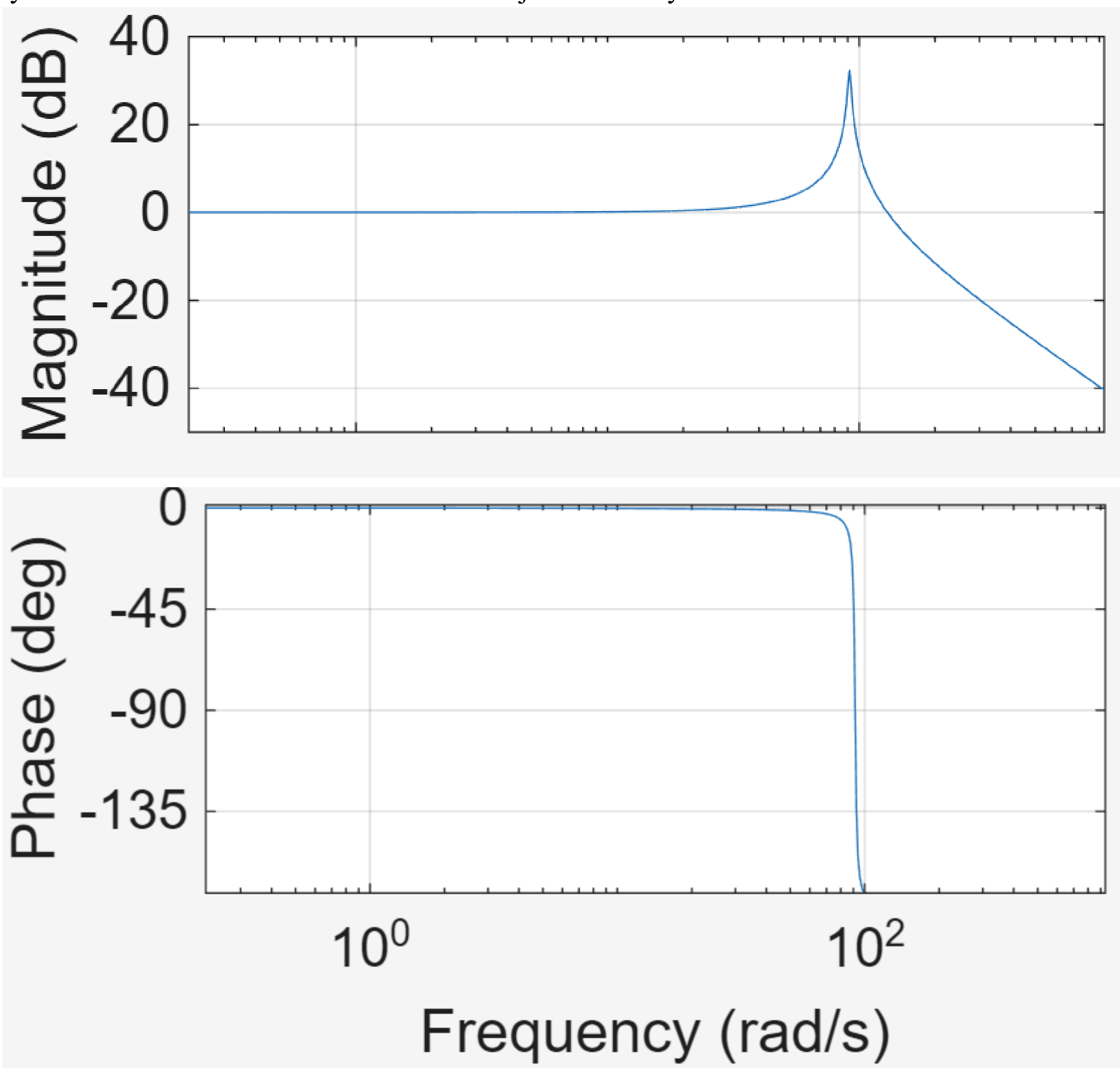

Figure 9: Closed-loop Bode plot.

Disturbance Rejection

Most advanced technical models of turbulence use the Dryden Turbulence Model to represent variable wind gusts. The Dryden model is represented in both spatial (**Ω**) and temporal form (ω), in three dimensions, with a function (**Φ**) representing the power spectral density of the turbulence [7]. However, this is a very complicated function that is difficult to model and would not incorporate well into our transfer function, making it difficult to use for our purposes.

To model turbulence in a useful way we can use the tools that we've used throughout this class.

1. An impulse function ($C\ *\ \delta(t)$) can be used to represent a sudden gust of wind of intensity C that instantly goes away.
2. A unit step function ($C\ *\ 1(t)$) can be used to represent a continuous gust of wind of intensity C.
3. A sin function ($C\ *\ sin(\omega t\ -\ \phi)$) can be used to represent wind gusts with temporally varying intensity.

These three functions cover the primary turbulence scenarios that we will want to model, and can be incorporated into our function to obtain a relevant output.

Appendix A: Matlab Code (for analysis)

```matlab
%our linearized Airbus A320 state space model
A = [-0.045 -2.95 0;3.10 -2.85 0;0 644.445 0];
B = [-0.055; 1.30; 0];
C = [0 0 1];
D = 0;

sys = ss(A,B,C,D);

%poles of the system
poles = eig(A);
disp('Poles of the System:');
disp(poles);

%plotting the poles of our sys
figure;
plot(real(poles), imag(poles), 'rx', 'MarkerSize', 12, 'LineWidth', 2);
hold on;
xline(0,'k--'); yline(0,'k--');
grid on;
xlabel('Real(\lambda)'); ylabel('Imag(\lambda)');
title('System Poles');
axis equal;

%plotting step response of our sys
figure;
step(sys);
grid on;
title('Step Response - Pitch Angle');
ylabel('\theta (rad)');

%plotting impulse response of our sys
figure;
impulse(sys);
grid on;
title('Impulse Response - Pitch Angle');
ylabel('\theta (rad)');

%bode plot of our sys
figure;
bode(sys);
grid on;
title('Bode Plot - Open-Loop System');
```
Figure 10: Matlab code for the making of the different plots.

Appendix B: References

1. "A320-100." *aircraftinvestigation.info*, https://aircraftinvestigation.info/airplanes/A320-100.html.
2. "Aircraft Pitch: System Modeling." *Control Tutorials for MATLAB and Simulink*, University of Michigan, https://ctms.engin.umich.edu/CTMS/index.php?example=AircraftPitch&section=SystemModeling.
3. *Chapter 4 Dynamical Equations for Flight Vehicles. courses.cit.cornell.edu*, https://courses.cit.cornell.edu/mae5070/DynamicEquations.pdf.
4. Leishman, J. Gordon. *Aircraft Stability & Control. Introduction to Aerospace Flight Vehicles*, Embry-Riddle Aeronautical University, 2023, https://eaglepubs.erau.edu/introductiontoaerospaceflightvehicles/chapter/aircraft-stability-control/.
5. MIT OpenCourseWare. "Lecture 13: Aircraft Stability and Control." *16.333 Aircraft Stability and Control*, https://ocw.mit.edu/courses/16-333-aircraft-stability-and-control-fall-2004/76c576f96472bf3052a038afba82feea_lecture_13.pdf.
6. Stanford University. "Aircraft." *Stanford Engineering Everywhere*, https://see.stanford.edu/materials/lsoeldsee263/14-aircraft.pdf.
7. United States, National Aeronautics and Space Administration. *Title Unknown (Report ID 20190000875)*. NASA Technical Reports Server (NTRS), https://ntrs.nasa.gov/api/citations/20190000875/downloads/20190000875.pdf.