Analysis for a Simplified Two-Degree-of-Freedom Model of a Wing-Aileron System involving Coupled Motion

Walker Davidson * Red ID: 821692385
San Diego State University, San Diego, California, 92115, United States

Dylan Shaffer † Red ID: 820416253
San Diego State University, San Diego, California, 92115, United States

Roelle Nogra ‡ Red ID: 818335005
San Diego State University, San Diego, California, 92115, United States

Karl Parks § Red ID: 819618742
San Diego State University, San Diego, California, 92115, United States

Jaden Bowyer ¶ Red ID: 821253245
San Diego State University, San Diego, California, 92115, United States

A modal analysis of the coupled motion present in a wing-aileron system consisting of both springs and dampers was performed. Assuming small perturbations of the system, the equations of motion were derived and analyzed. Through this analysis, the equations and characteristics of the system are to be obtained and considered by linearizing the system and normalizing the modes. Once the equations of the system are obtained they are plotted for a visual confirmation of results.

Nomenclature

θ_1	=	Generalized coordinate 1
θ_2	=	Generalized coordinate 2
$\theta_1 0$	=	Generalized coordinate 1 at time 0
$\theta_2 0$	=	Generalized coordinate 2 at time 0
$\dot{\theta_1}$	=	Velocity 1
$\dot{\theta_2}$	=	Velocity 2
$\dot{\theta_1 0}$	=	Velocity 1 at time 0
$\dot{\theta_2 0}$	=	Velocity 2 at time 0
Θ_1	=	Amplitude 1 mode 1
Θ_2	=	Amplitude 2 mode 2
A	=	Point A, Fixed Translation
B	=	Point B, Internal Hinge
G_1	=	Location of Center of Mass of the Wing
G_2	=	Location of Center of Mass of the Aileron
c	=	Distance from point A to G_1
a	=	Distance from point A to point B
d	=	Distance from point B to point G_2
k_A	=	Spring stiffness at point A
k_B	=	Spring stiffness at point B

*Student, San Diego State University Aerospace Engineering Department
†Student, San Diego State University Aerospace Engineering Department
‡Student, San Diego State University Aerospace Engineering Department
§Student, San Diego State University Aerospace Engineering Department
¶Student, San Diego State University Aerospace Engineering Department

C_A	=	Dashpot coefficient at point A
$I_{G_1}^{w}$	=	Mass moment of inertia with respect to the center of mass of the wing
$I_{G_2}^{ail}$	=	Mass moment of inertia with respect to the center of mass of the aileron
m_1	=	Mass of wing
m_2	=	Mass of aileron
α	=	Given parameter
β	=	Given parameter
γ	=	Given parameter
ϕ	=	Given parameter
FBD	=	Free Body Diagram

Introduction

By obtaining a simplified model of the given wing-aileron system, the reaction forces and inertial forces can be taken into consideration to obtain the equations of motion for the system. Once the equations of motion are simplified and linearized for small perturbations, modal analysis and normalization allows the natural frequencies and modes of the system to be obtained and analyzed. We used the Mechanical Vibrations text to assist in the completion of this report. [1] Once the equations of motion, modes, and natural frequencies have been obtained, the results of free vibration are plotted in MATLAB.

Discussion Part 2

Now the wing-aileron system below is considered with a dashpot at point A and springs at both points A and B:

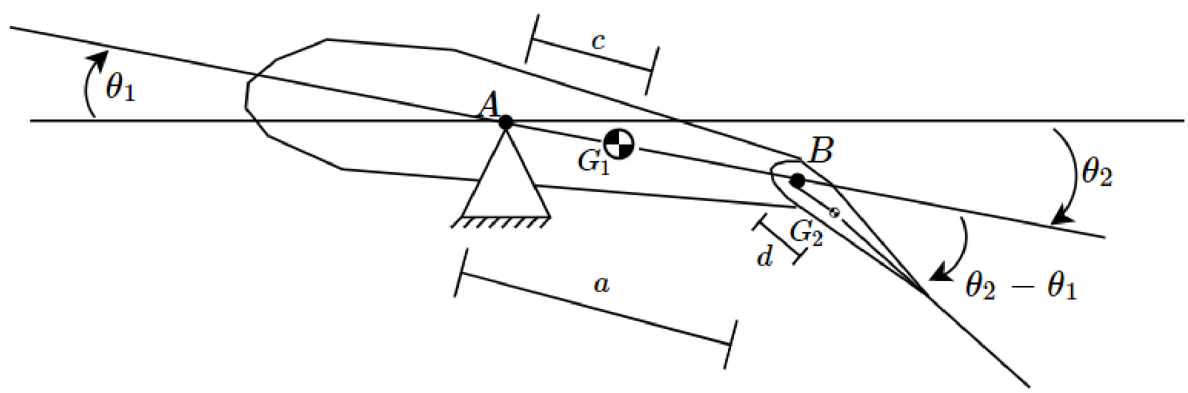


Fig. 1 Wing-Aileron System

Assignment #1 - Creating the Free Body Diagrams and Equations of Motion

We begin by producing a free body diagram of the Aileron only, including the inertial and moment of inertial forces.

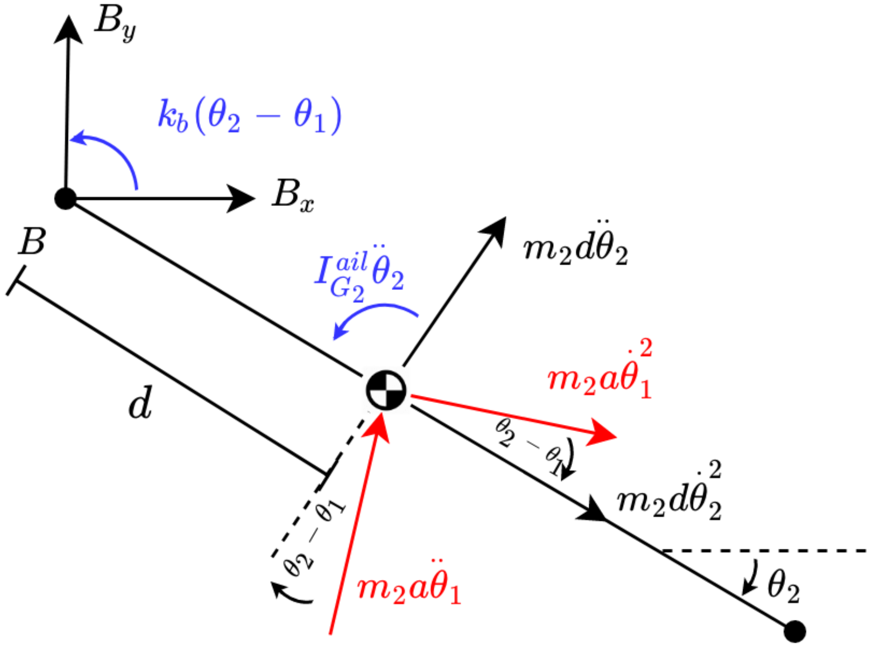


Fig. 2 Aileron Free Body Diagram

From here the first equation of motion can be derived by calculating the moment about B

$$\Sigma M_B = k_b(\theta_2 - \theta_1) + I^{ail}_{G_2}\ddot{\theta}_2 + m_2 d^2\ddot{\theta}_2 + m_2 ad\ddot{\theta}_1 cos(\theta_2 - \theta_1) + m_2 a\dot{\theta}_2^{\,2} sin(\theta_2 - \theta_1) = 0 \tag{1}$$

Next, the free body diagram of the entire system with the inertial and moment of inertial forces included can be produced.

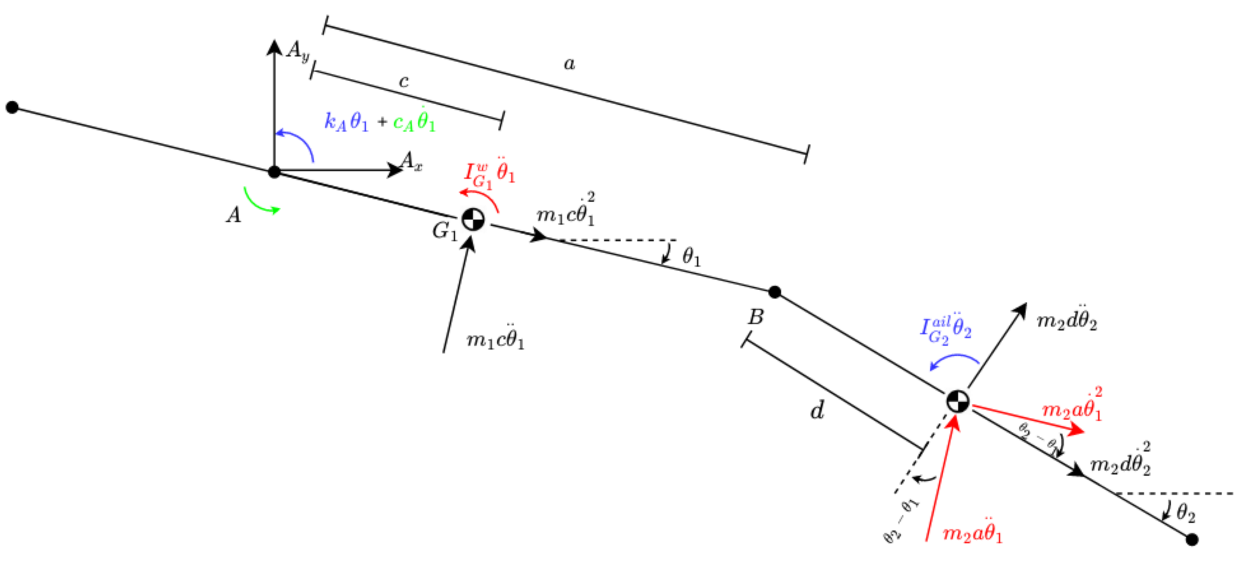


Fig. 3 System Free Body Diagram

Now, the second equation of motion can be obtain using the moment about A.

$$\begin{gathered}\Sigma M_A = k_A\theta_1 + c_A\dot{\theta}_1 + I^w_{G_1}\ddot{\theta}_1 + m_1c^2\ddot{\theta}_1 + I^{ail}_{G_2}\ddot{\theta}_2 \\ + m_2a[a + dcos(\theta_2 - \theta_1)]\ddot{\theta}_1 - m_2adsin(\theta_2 - \theta_1)\dot{\theta}_2^2 \\ + m_2dcos(\theta_2 - \theta_1)[a + dcos(\theta_2 - \theta_1)]\ddot{\theta}_2 = 0\end{gathered} \tag{2}$$

From equation 1

$$I^{ail}_{G2}\ddot{\theta}_2 + m_2adcos(\theta_2 - \theta_1)\dot{\theta}_1 = -m_2d^2\ddot{\theta}_2 + k_B\theta_1 - k_B\theta_2 + m_2a\dot{\theta_2}^2 sin(\theta_2 - \theta_1) \tag{3}$$

Substituting equation 3 into 2

$$\begin{gathered}I^w_{G1}\ddot{\theta}_1 + m_1c^2\ddot{\theta}_1 + m_2a^2\ddot{\theta}_1 - m_2d^2\ddot{\theta}_2 \\ + m_2dcos(\theta_2 - \theta_1)[a + dcons(\theta_2 - \theta_1)]\ddot{\theta}_2 + c_A\dot{\theta}_1 \\ - m_2adsin(\theta_2 - \theta_1)\dot{\theta}_2^2 + k_A\theta_1 + k_B\theta_1 - k_B\theta_2 + +m_2a\dot{\theta_2}^2 sin(\theta_2 - \theta_1) = 0\end{gathered} \tag{4}$$

A geometric approach was used to validate the inertial forces acting on G_2. Observe the acceleration in the geometric approach is negative the acceleration of the inertial force.

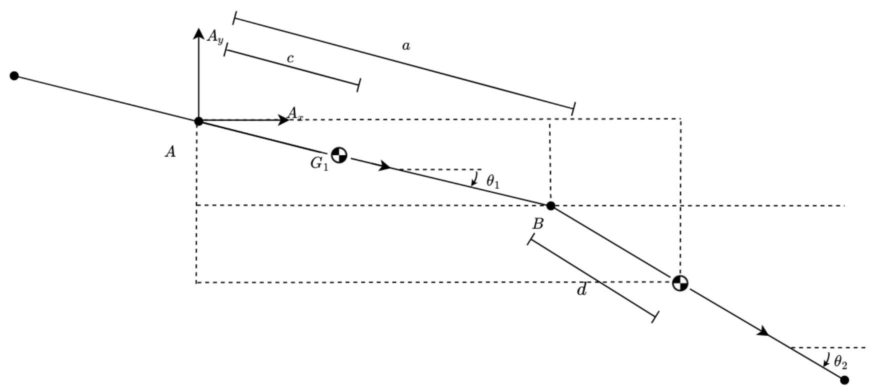


Fig. 4 Geometric Approach

Only the motion along x is required.

$$p_x(t) = acos(\theta_1(t)) + dcos(\theta_2(t)) \tag{5}$$

$$V_x(t) = -asin(\theta_1(t)\dot{\theta}_1(t) - dsin(\theta_2(t))\dot{\theta}_2(t) \tag{6}$$

$$a_x(t) = -acos(\theta_1(t))\dot{\theta}_1(t))^2 - \ddot{\theta}_1(t)(asin(\theta_1(t)) - dcos(\theta_2(t))\dot{\theta}_2(t)^2 - \ddot{\theta}_2(t)(dsin(\theta_2(t)) \tag{7}$$

Assignment #2 - Matrices

From here, we assume a small perturbation of the general coordinates. Taylor series expansion of sin and cos for small angles yield:

$$cos(\theta) \approx 1 \tag{8}$$

$$sin(\theta) \approx \theta \tag{9}$$

Any small value squared leads to much a smaller value so the small time derivatives squared can assumed to be approximately zero.

$$\dot{\theta}^2 \approx 0 \tag{10}$$

Using the above considerations, Equations 4 and 1 now become

$$I_{G1}^{w}\ddot{\theta}_1 + m_1c^2\ddot{\theta}_1 + m_2a^2\ddot{\theta}_1 + m_2ad\ddot{\theta}_2 + c_A\dot{\theta}_1 + k_B\theta_1 - k_B\theta_2 = 0 \tag{11}$$

$$m_2ad\ddot{\theta}_1 + I_{G2}^{ail}\ddot{\theta}_2 + m_2d^2\ddot{\theta}_2 - k_B\theta_1 + k_B\theta_2 = 0 \tag{12}$$

Now, the equations of motion can be placed into a matrix. Observe the symmetric matrices, which are a result of Maxwell-Betti reciprocal work theorem [1].

$$\begin{bmatrix} I_{G_1}^{w} + m_1c^2 + m_2a^2 & m_2ad \\ m_2ad & I_{G_2}^{ail} + m_2a^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} c_A & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} k_A + k_B & -k_B \\ -k_B & k_B \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{13}$$

Where:

$$Mass\ matrix = \begin{bmatrix} I_{G_1}^{w} + m_1c^2 + m_2a^2 & m_2ad \\ m_2ad & I_{G_2}^{ail} + m_2a^2 \end{bmatrix} \tag{14}$$

$$Damping\ matrix = \begin{bmatrix} c_A & 0 \\ 0 & 0 \end{bmatrix} \tag{15}$$

$$Stiffness\ matrix = \begin{bmatrix} k_A + k_B & -k_B \\ -k_B & k_B \end{bmatrix} \tag{16}$$

Assignment #3 - Analysis using Parameter Set 1

If we consider the following parameters:

$$k_A = \alpha k_B \equiv \alpha k \quad a = \beta d \quad I_{G_2} = \gamma m_2 d^2 \quad I_{G_1}^{w} = \Phi I_{G_2}^{ail} \quad \frac{k}{m_2 d^2} = 25 \quad \alpha = 10 \quad \beta = 4 \quad \gamma = 2 \quad \Phi = 5 \tag{17}$$

Equation 13 is now

$$\begin{bmatrix} 30m_2d^2 & 4m_2d^2 \\ 4m_2d^2 & 3m_2d^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta_1} \\ \ddot{\theta_2} \end{bmatrix} + \begin{bmatrix} 11k & -k \\ -k & k \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{18}$$

Where:

$$\theta_1 = \Theta_1 cos(\omega t + \phi) \tag{19}$$

$$\theta_2 = \Theta_2 cos(\omega t + \phi) \tag{20}$$

And:

$$\ddot{\theta_1} = -\Theta_1 \omega^2 cos(\omega t + \phi) \tag{21}$$

$$\ddot{\theta_2} = -\Theta_2 \omega^2 cos(\omega t + \phi) \tag{22}$$

Substituting equations 19, 20, 21, and 22 into Equation 18:

$$-\omega^2 \begin{bmatrix} 30m_2d^2 & 4m_2d^2 \\ 4m_2d^2 & 3m_2d^2 \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} cos(\omega t + \phi) + \begin{bmatrix} 11k & -k \\ -k & k \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} cos(\omega t + \phi) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{23}$$

Simplifying:

$$\begin{bmatrix} -30\omega^2 m_2d^2 + 11k & -4\omega^2 m_2d^2 - k \\ -4\omega^2 m_2d^2 - k & -3\omega^2 m_2d^2 + k \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} cos(\omega t + \phi) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{24}$$

By Taking the Determinate of the first matrix of equation 24 equal to zero, and solving for ω, the frequencies at which the equations of motion become dependent can be determined:

$$det \begin{vmatrix} -30\omega^2 m_2d^2 + 11k & -4\omega^2 m_2d^2 - k \\ -4\omega^2 m_2d^2 - k & -3\omega^2 m_2d^2 + k \end{vmatrix} = (-30\omega^2 m_2d^2 + 11k)(-3\omega^2 m_2d^2 + k) - (-4\omega^2 m_2d^2 - k)^2 = 0 \tag{25}$$

Expanding:

$$90\omega w^2 m_2 d^4 - 30\omega^2 d^2 k - 33\omega^2 m_2 d^2 k + 4k^2 - 16\omega^4 m_2 d^4 - 8\omega^2 m_2 d^2 k - k^2 = 0 \tag{26}$$

Simplifying:

$$74 m_2^2 d^4 \omega^4 - 71 k m_2 d^2 \omega^2 + 10k^2 = 0 \tag{27}$$

We can solve for ω using the quadratic equation:

$$\omega^2 = \frac{71km_2d^2 \pm \sqrt{(71km_2d^2)^2 - 4(74m_2^2d^4)(10k)}}{2(74m_2^2d^4)} \tag{28}$$

With an additional substitution from the given parameters,

$$\frac{k}{m_2d^2} = 25 \tag{29}$$

$$\omega^2 = 4.2875,\ 19.699 \tag{30}$$

$$\omega = 2.0707,\ 4.4385 \tag{31}$$

From Equation 24:

$$(-4\omega^2 m_2 d^2 - k)\Theta_1 + (-3\omega^2 m_2 d^2 + k)\Theta_2 = 0 \tag{32}$$

Solving for the amplitude ratios:

$$(4\omega^2 m_2 d^2 - k)\Theta_1 = (-3\omega^2 m_2 d^2 + k)\Theta_2 \tag{33}$$

$$r = \frac{\Theta_2}{\Theta_1} = \frac{4\omega^2 m_2 d^2 - k}{-3\omega^2 m_2 d^2 + k} = \frac{4\omega^2 + \frac{k}{m_2 d^2}}{-3\omega^2 + \frac{k}{m_2 d^2}} \tag{34}$$

Substitute equation 29 and 30 into 34 provides the amplitude ratios:

$$r^{(1)} = 3.47 \quad r^{(2)} = -3.04 \tag{35}$$

Now the modes are known

$$mode1 : \vec{\theta}^{(1)}(t) = \begin{bmatrix} \Theta_1^{(1)} \\ 3.47\Theta_1^{(1)} \end{bmatrix} cos(2.07t + \phi_1) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{36}$$

$$mode2 : \vec{\theta}^{(2)}(t) = \begin{bmatrix} \Theta_1^{(2)} \\ -3.04\Theta_1^{(2)} \end{bmatrix} cos(4.44t + \phi_2) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{37}$$

The second mode frequency is nearly twice as large which will impact the motion of each angle. Also observe that the second amplitude ratio will result in one negative amplitude for mode 2.

A general solution to our equations of motion is a linear combination of the modes.

$$\vec{\theta}(t) = c_1\vec{\theta}^{(1)}(t) + c_2\vec{\theta}^{(2)}(t) \tag{38}$$

Normalizing $mode1$ and $mode2$ so their modulus are $\sqrt{3}$ and $\sqrt{5}$ respectively.

$$\vec{\theta_1} = \theta_1^{(1)}\sqrt{3} + \theta_1^{(2)}\sqrt{5} \tag{39}$$

$$\vec{\theta_2} = \theta_2^{(1)}\sqrt{3} + \theta_2^{(2)}\sqrt{5} \tag{40}$$

Consider the following initial conditions:

$$\begin{gathered} \theta_1(t=0) = \theta_{10} = \frac{2}{180}\pi \ \ rad \qquad \dot{\theta}_1(t=0) = \dot{\theta}_{10} = -\frac{3}{2\cdot 180}\pi \ \ rad/s \\ \theta_2(t=0) = \theta_{20} = \frac{1}{180}\pi \ \ rad \qquad \dot{\theta}_2(t=0) = \dot{\theta}_{20} = -\frac{5}{180}\pi \ \ rad/s \end{gathered} \tag{41}$$

Further expanding Equation 38

$$\theta_1 = c_1\theta_1^{(1)} + c_2\theta_1^{(2)} = \sqrt{3}\Theta_1^{(1)}cos(\omega_1 t + \phi^{(1)}) + \sqrt{5}\Theta_1^{(2)}cos(\omega_2 t + \phi^{(2)}) \tag{42}$$

$$\theta_2 = c_1\theta_2^{(1)} + c_2\theta_2^{(2)} = \sqrt{3}\Theta_2^{(1)}cos(\omega_1 t + \phi^{(1)}) + \sqrt{5}\Theta_2^{(2)}cos(\omega_2 t + \phi^{(2)}) \tag{43}$$

Where:

$$r_1\Theta_1^{(1)} = \Theta_2^{(1)} \; and \; r_2\Theta_1^{(2)} = \Theta_2^{(2)} \tag{44}$$

Substituting Equation 44 into Equation 43:

$$\theta_2 = c_1\theta_2^{(1)} + c_2\theta_2^{(2)} = \sqrt{3}r_1\Theta_1^{(1)}cos(\omega_1 t + \phi^{(1)}) + \sqrt{5}r_2\Theta_1^{(2)}cos(\omega_2 t + \phi^{(2)}) \tag{45}$$

And solving for each amplitude and phase angle yields the following equations:

$$\Theta_1^{(1)} = \sqrt{\left[\frac{\theta_{10}}{\sqrt{3}} - \frac{\theta_{20} - r_1\theta_{10}}{\sqrt{3}(r_2 - r_1)}\right]^2 + \left[\frac{-\dot{\theta_{10}}}{\omega_1\sqrt{3}} - \frac{\dot{\theta_{20}} - r_1\dot{\theta_{10}}}{\omega_1\sqrt{3}(r_1 - r_2)}\right]^2} \tag{46}$$

$$\Theta_1^{(2)} = \sqrt{\left[\frac{\theta_{20} - r_1\theta_{10}}{\sqrt{5}(r_2 - r_1)}\right]^2 + \left[\frac{\dot{\theta_{20}} - r_1\dot{\theta_{10}}}{\sqrt{5}(r_1 - r_2)}\right]^2} \tag{47}$$

$$\phi^{(1)} = cos^{-1}\left(\frac{\theta_{10}}{\Theta_1^{(1)}\sqrt{3}} - \frac{\theta_{20} - r_1\theta_{10}}{\Theta_1^{(1)}\sqrt{3}(r_2 - r_1)}\right) \tag{48}$$

$$\phi^{(2)} = cos^{-1}\left(\frac{\theta_{20} - r_1\theta_{10}}{\Theta_1^{(2)}\sqrt{5}(r_2 - r_1)}\right) \tag{49}$$

Solving for the values:

$$\Theta_1^{(1)} = 0.0131 \tag{50}$$

$$\Theta_2^{(1)} = 0.0454 \tag{51}$$

$$\Theta_1^{(2)} = 0.0071 \tag{52}$$

$$\Theta_2^{(2)} = -0.0217 \tag{53}$$

$$\phi^{(1)} = 0.5776 \; rad \tag{54}$$

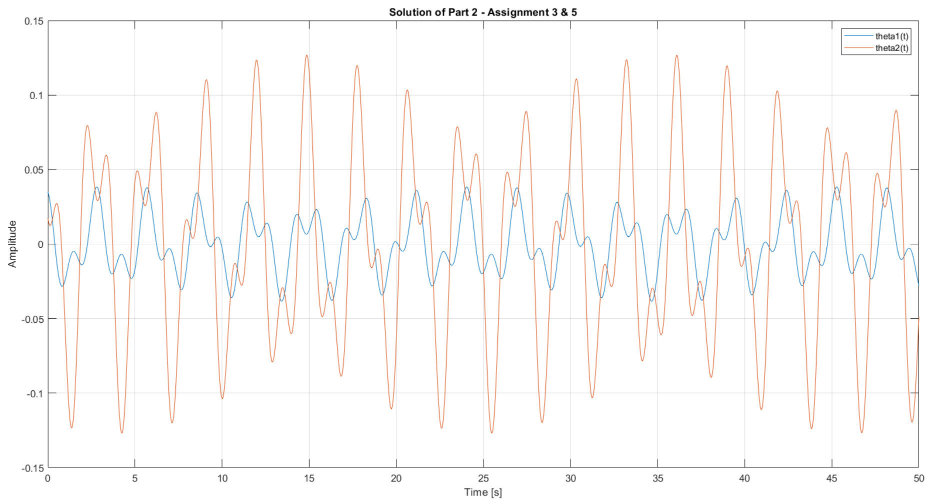
$$\phi^{(2)} = 0.0352 \; rad \tag{55}$$

Observe the general solution obtained is the linear superposition of each mode:

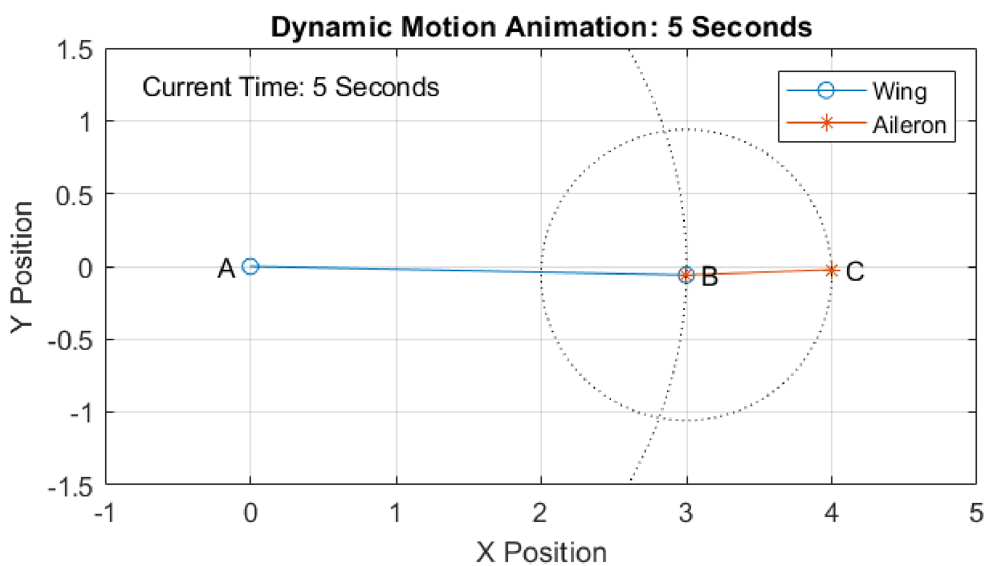
$$\theta_1(t) = 0.0131\sqrt{3}cos(2.07t + 0.5776) + 0.0071\sqrt{5}cos(4.44t + 0.0352) \tag{56}$$

$$\theta_2(t) = 0.0454\sqrt{3}cos(2.07t + 0.5776) - 0.0217\sqrt{5}cos(4.44t + 0.0352) \tag{57}$$

The actual free vibration solution is plotted below for 50 seconds. The MATLAB file included will automatically plot the below content as Figure 1 using the Assignment 3 case. Observe, the superposition of each mode part can be noticed, such as a larger sin wave which is noticeable over a 20 second period from peek to peek.



For additional visualization and confirmation of results, we created an animation using arbitrary lengths for the wing and aileron from their point of rotation, set to 3 and 1, respectively. An image is included below, revealing the location of point B and C at 5 seconds. If the MATLAB file is run, the animation will automatically start on Figure 2 using the Assignment 3 case.



Assignment #4 - Analysis using Parameter Set 2

$$k_A = k \quad k_B = 0 \quad a = \beta d \quad I_{G2}^{ail} = \gamma m2d^2 \quad I_{G1}^{w} = \Phi I_{G2}^{ail} \quad \frac{k}{m_2 d^2} = 25 \quad \alpha = 10 \quad \beta = 4 \quad \gamma = 2 \quad \Phi = 5 \tag{58}$$

Assuming linearity, the above relations and considerations can be made, effectively reducing Equation 18 into the following.

$$\begin{bmatrix} 30m_2d^2 & 4m_2d^2 \\ 4m_2d^2 & 3m_2d^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta_1} \\ \ddot{\theta_2} \end{bmatrix} + \begin{bmatrix} k & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{59}$$

Once again:

$$\theta_1 = \Theta_1 cos(\omega t + \phi) \tag{60}$$

$$\theta_2 = \Theta_2 cos(\omega t + \phi) \tag{61}$$

And:

$$\ddot{\theta_1} = -\Theta_1 \omega^2 cos(\omega t + \phi) \tag{62}$$

$$\ddot{\theta_2} = -\Theta_2 \omega^2 cos(\omega t + \phi) \tag{63}$$

Therefore:

$$-\omega^2 \begin{bmatrix} 30m_2d^2 & 4m_2d^2 \\ 4m_2d^2 & 3m_2d^2 \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} cos(\omega t + \phi) + \begin{bmatrix} k & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} cos(\omega t + \phi) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{64}$$

$$[A] = \begin{bmatrix} -30m_2d^2 & -4m_2d^2 \\ -4m_2d^2 & -3m_2d^2 \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} cos(\omega t + \phi) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{65}$$

Taking the determinant of the frequency matrix [A], the natural frequency of the system according to the given conditions are found as presented below.

$$\begin{gathered} det[A] = (-30\omega^2 m_2 d^2 + k)(-3\omega^2 m_2 d) - (-4\omega^2 m_2 d^2)^2 = 0 \\ \omega^2(74m_2^2 d^4 \omega^2 - 3km_2 d) = 0 \\ \omega^2 = 0 \quad \omega = 0,\ 1.007 \end{gathered} \tag{66}$$

Solving for the amplitude ratio

$$\begin{gathered} -4\omega^2 m_2 d^2 \Theta_1 - 3\omega^2 m_2 d^2 \Theta_2 = 0 \\ -4\Theta_1 = 3\Theta_2 \end{gathered} \tag{67}$$

$$r = \frac{\Theta_2}{\Theta_1} = \frac{-4}{3} \tag{68}$$

$$mode1 : \begin{bmatrix} \Theta_1^{(1)} \\ \frac{-4}{3}\Theta_1^{(1)} \end{bmatrix} cos(\phi_1) \tag{69}$$

$$mode2 : \begin{bmatrix} \Theta_1^{(2)} \\ \frac{-4}{3}\Theta_1^{(2)} \end{bmatrix} cos(1.01 + \phi_2) \tag{70}$$

Observe, the first mode represents rigid body motion. Meaning in this mode, translation may be present but deformation will not occur. For mode 2, ω has a tangible value, signifying that this is not a rigid body mode and deformation may occur.

A general solution to our equations of motion is a linear combination of the modes.

$$\vec{\theta}(t) = c_1\vec{\theta}^{(1)}(t) + c_2\vec{\theta}^{(2)}(t) \tag{71}$$

Normalizing $mode1$ and $mode2$ to so their modulus are $\sqrt{3}$ and $\sqrt{5}$ respectively.

$$\vec{\theta_1} = \theta_1^{(1)}\sqrt{3} + \theta_1^{(2)}\sqrt{5} \tag{72}$$

$$\vec{\theta_2} = \theta_2^{(1)}\sqrt{3} + \theta_2^{(2)}\sqrt{5} \tag{73}$$

Through normalizing the modes, we can scale the values obtained to an arbitrary or prespecified desired value to simplify the problem as shown in equations 72 through 73.

Assignment #5 - MATLAB Code

The MATLAB Code developed for this portion of the assignment is included in the Appendix. The MATLAB Code calculates the Frequencies and Modes of a 2 DoF Dynamic System and outputs to the command window. The code currently is setup for analyzing the initial conditions and parameters specified in Assignment 3 and plots the solution over a given time interval. The code also includes an animation of the motion for a simplified non-linear wing-aileron model. The two figures the .m script generates are included in Assignment 3.

Conclusion

The wing-aileron model provided valuable insight into the possible vibrations experience on the system. Once the system was deconstructed into its individual components, several key observations were discovered: All matrices are symmetric. Amplitude ratios can be negative. Superposition of modes allows a presentation of an actual free vibration.

There are several other interesting aspects that can be added into the system. If a dampening coefficient value is supplied, the equation of motion and the actual free vibration would show a decaying motion. Similar to Homework 2, adding the Lift and Aerodynamic Moment to the FBD and taking into consideration how those values change throughout the vibration would add another interesting component to the model. Also similar to Homework 2, adding a linear spring at the point of rotation would allow a 3 DoF freedom analysis to take place. From this simplified model, a considerably amount of information could be extracted for an actual aircraft, such as the stability constraints of the wing. If an unmanned aerial vehicle is desired, adding the vibration component of course would affect a PID controller for example.

Appendix

MATLAB Code

```matlab
% Part 2 Assignment 5
% Description: Calculates the Frequencies and Modes of a 2 DoF Dynamic
% System. Uses initial conditions and plots the solution over a given time
% interval. Also includes an animation of the motion for a simplified
% non-linear wing-aileron model.

% Class: AE 410 Vibrations
% Assignment: Final Project - Part 2 Assignment 5
% Authors: Karl Parks, Jaden Boywer, Dylan Shaffer, Walker Davidson, Roelle Nogra
% Date: 12/12/2019

% USER: Input Matrices are in the Following Section after the Symbolics

% Disclaimer: This code was developed originally for a 2 DoF non-linear
% model and therefore most of the notation is expressed using the greek
% letter theta to represent changes in angle. Nearly all of the the .m
% script will still work for linear motion.

% MATLAB R2018a or newer and the Symbolic Math Toolbox are required.

clear; clc; close all;

%% Symbolic Definitions required for Assignment 3
% m1 = mass_1
% m2 = mass_2
% c = distance c
% d = distance d
% a = distance a
% IGw = I_G^wing
% IGail = I_G^aileron
% kA = k_A
% kB = k_B

% sphi = small phi
% w = omega

syms m1 m2 c d a IGw IGail kA kB
```

```
syms k alpha beta gamma sphi w

alpha = 10;
beta = 4;
gamma = 2;
sphi = 5;
kB = k; % 0 for Assignment 4
kA = alpha*k; % k for Assignment 4
a = beta*d;
c = a/4;
m1 = 4*m2;
IGail = gamma*m2*d^2;
IGw = sphi*IGail;

%% Input Matrices
% input mass matrix
M = [IGw + m1*c^2 + m2*a^2, m2*a*d;
     m2*a*d                , IGail + m2*d^2];

% input stiffness matrix
K = [kA + kB, -kB;
     -kB     ,  kB];

% additional example matrices below, uncomment to use
% input mass matrix
% M = [10 0
%      0  4];
%
% % input stiffness matrix
% K = [3 2
%      2 4];

%% Beginning Calculations
combinedM = -w^2*M + K; %combined matrix -> frequency matrix

% show matrices
M
K
combinedM

detW2 = det(combinedM); %frequency equations
syms temp
detTemp = subs(detW2, [w^4 w^2], [temp^2, temp]);
solveDet = solve(detTemp); % solve quadratic equation
wTemp = double(simplify(subs(solveDet, m2*d^2, k/25)));
if abs(sqrt(wTemp(1))) < abs(sqrt(wTemp(2))) %smaller omega for mode 1
    w1 = sqrt(wTemp(1));
    w2 = sqrt(wTemp(2));
else
    w1 = sqrt(wTemp(2));
    w2 = sqrt(wTemp(1));
end

% show w values
```

```
fprintf('Omega1: %2.4f\n',w1);
fprintf('Omega2: %2.4f\n',w2);

%% Amplitude Ratios
r = -combinedM(2,1)/combinedM(2,2);
r = simplify(subs(r, m2*d^2, k/25));
r1 = double(simplify(subs(r,w,w1)));
r2 = double(simplify(subs(r,w,w2)));

% show r values
fprintf('r1: %2.4f\n',r1);
fprintf('r2: %2.4f\n',r2);

%% Display Modes (without initial conditions applied)

digits(4);
syms bth1_1 bth1_2 t phi_1 phi_2

fprintf('\nMode 1:\n');
disp([bth1_1; vpa(r1)*bth1_1]*cos(vpa(w1)*t + vpa(phi_1)));

fprintf('Mode 2:\n');
disp([bth1_2; vpa(r2)*bth1_2]*cos(vpa(w2)*t + vpa(phi_2)));

%% Initial Conditions: Amplitudes and Phase Angles
% th1_t = equation of motion for theta_1
% th1_0 = theta_1 position inital condition
% th1d_0 = theta_1_dot velocity inital condition
% bth1_1 = big_theta_1 (1)
% bth2_1 = big_theta_2 (2)
% phi1 = phi_1
% w1 = omega_1
% r1 = ratio_1

th1_0 = 2*pi/180;
th2_0 = 1*pi/180;

th1d_0 = (-3*pi)/(2*180);
th2d_0 = (-5*pi)/180;

C1 = sqrt(3);
C2 = sqrt(5);
% C1 = 1;
% C2 = 1;

% Original Amplitudes (No Normalization)
% bth1_1 = (1/(r2-r1))*sqrt((r2*th1_0 - th2_0)^2+(-r2*th1d_0 + th2d_0)^2/w1^2)
    ;
% bth2_1 = r1*bth1_1;
% bth1_2 = (1/(r2-r1))*sqrt((r1*th1_0 - th2_0)^2+(-r1*th1d_0 + th2d_0)^2/w2^2)
    ;
% bth2_2 = r2*bth1_2;

% Normalized Amplitudes
```

```
temp1 = ((th1_0/C1)-((th2_0 - r1*th1_0)/(C1*(r2-r1))))^2;
temp2 = ((-th1d_0/(w1*C1))-((th2d_0 - r1*th1d_0)/(w1*C1*(r1-r2))))^2;
bth1_1 = sqrt(temp1 + temp2);
bth1_2 = sqrt(((th2_0 - r1*th1_0)/(C2*(r2-r1)))^2 + ((th2d_0 - r1*th1d_0)/(C2
    *(r1-r2)))^2);

if isinf(bth1_1) == 1
    disp('Can Not Continue');
    return;
end

bth2_1 = r1*bth1_1;
bth2_2 = r2*bth1_2;

fprintf('Big Theta1(1): %2.4f\n',bth1_1);
fprintf('Big Theta2(1): %2.4f\n',bth2_1);
fprintf('Big Theta1(2): %2.4f\n',bth1_2);
fprintf('Big Theta2(2): %2.4f\n',bth2_2);

% Original Phase Angles (No Normalization)
% phi1 = atan((-r2*th1d_0 + th2d_0)/(w1*(r2*th1_0 - th2_0)));
% phi2 = atan((r1*th1d_0 - th2d_0)/(w2*(-r1*th1_0 + th2_0)));

% Normalized Phase Angles
phi1 = acos((th1_0/(bth1_1*C1)) - ((th2_0 - r1*th1_0)/(bth1_1*C1*(r2-r1))));
phi2 = acos((th2_0 - r1*th1_0)/(bth1_2*C2*(r2-r1)));

fprintf('Phi 1: %2.4f rad, %2.4f deg\n',phi1,phi1*180/pi);
fprintf('Phi 2: %2.4f rad, %2.4f deg\n',phi2,phi2*180/pi);

%% Display Modes (normalization applied with initial conditions)

fprintf('\nMode 1:\n');
disp([vpa(bth1_1); vpa(r1)*vpa(bth1_1)]*cos(vpa(w1)*t + vpa(phi1)));

fprintf('Mode 2:\n');
disp([vpa(bth1_2); vpa(r2)*vpa(bth1_2)]*cos(vpa(w2)*t + vpa(phi2)));

%% Begin Plot Work
totalTime = 50; %seconds, adjust here
numOfSteps = totalTime*20; %20 steps per second
t = linspace(0,totalTime,numOfSteps);

% Expression for theta_1(t) and theta_2(t) using superposition of modes.
th1_t = C1*bth1_1*cos(w1*t + phi1) + C2*bth1_2*cos(w2*t + phi2);
th2_t = C1*bth2_1*cos(w1*t + phi1) + C2*bth2_2*cos(w2*t + phi2);

fprintf('Initial Val\t== Evaluated @ t(0)\n');
fprintf('%0.4f \t\t== %0.4f\n',th1_0, th1_t(1));
fprintf('%0.4f \t\t== %0.4f\n',th2_0, th2_t(1));

assignmentFig = figure();
plot(t,th1_t);
hold on;
```

```
grid on;
plot(t,th2_t);
xlabel('Time [s]');
ylabel('Amplitude');
title('Solution of Part 2 - Assignment 3 & 5');
legend('theta1(t)','theta2(t)');

%% Begin Animation Work (only applies to non-linear motion)
% Will this get us some extra points for going above and beyond???
% Thanks Demasi!!!

percentageOfTime = 0.1; % animation time (avoiding 50 seconds)
speedFactor = 1; % scale playback speed
angleScale = 1; % show extreme motion
titleName = 'Dynamic Motion Animation: ' + string(floor(totalTime*
    percentageOfTime)) + ' Seconds';
fprintf('\nStarting and running animation... please wait ~ %0.0f seconds.\n',
    floor(totalTime*percentageOfTime));

centerWing = [0 0]; %[x y] coordinate
l1 = 3; % arbitrary values for wing length
l2 = 1; % arbitrary values for aileron length
Bx = [centerWing(1) 0]; % x vec for point B (end of wing)
By = [centerWing(2) 0];
Cx = [Bx(2) 0]; % x vec for point C (end of alieron)
Cy = [By(2) 0];

% circular motion
thCircle = linspace(0,2*pi,500);
rCircle = l1;
xCircle = centerWing(1)+rCircle*cos(thCircle);
yCircle = centerWing(1)+rCircle*sin(thCircle);
rCircle2 = l2;

animationFig = figure();
wing = plot(Bx,By,'-o');
grid on;
hold on;
aileron = plot(Cx,Cy,'-*');
plot(xCircle, yCircle, ':k');
plotCircle2 = plot(xCircle, yCircle, ':k');

xlabel('X Position');
ylabel('Y Position');
title(titleName);
legend('Wing', 'Aileron');
Amark = text(centerWing(1),centerWing(2),'A  ','HorizontalAlignment','right');
Bmark = text(Bx(2),By(2),'  B');
Cmark = text(Cx(2),Cy(2),'  C');
timeSpotx = -0.75;
timeMark = text(timeSpotx,1.25,'Current Time: ');

axis equal;
axis([-1 5 -1.5 1.5]);
```

```
a = tic; % start timer
for i = 1:floor(numOfSteps*percentageOfTime)
    % calculate position of point B
    Bx(2) = centerWing(1) + cos(th1_t(i)*angleScale)*l1;
    By(2) = centerWing(2) + sin(th1_t(i)*angleScale)*l1;
    % start position of point C (at B)
    Cx(1) = Bx(2);
    Cy(1) = By(2);
    % calculate position of point C
    Cx(2) = Bx(2) + cos(th2_t(i)*angleScale)*l2;
    Cy(2) = By(2) + sin(th2_t(i)*angleScale)*l2;

    delete(Bmark);
    delete(Cmark);
    delete(timeMark);
    set(wing,'XData',Bx,'YData',By);
    Bmark = text(Bx(2),By(2),'  B');
    set(aileron,'XData',Cx,'YData',Cy);
    Cmark = text(Cx(2),Cy(2),'  C');
    xCircle2 = Bx(2)+rCircle2*cos(thCircle);
    yCircle2 = By(2)+rCircle2*sin(thCircle);
    set(plotCircle2,'XData',xCircle2,'YData',yCircle2);

    insertTime = 'Current Time: ' + string(floor((i/numOfSteps)*totalTime)) + ' Seconds';
    timeMark = text(timeSpotx,1.25,insertTime);

    b = toc(a); % check timer
    while b*speedFactor < ((totalTime)/(numOfSteps))
        drawnow % update screen
        b = toc(a); % check timer
    end
    a = tic; % reset timer after updating
end
drawnow;

% End of Code
fprintf('\nEnd of Code - Scroll up for more information.\n');
```

References

[1] Rao, S. S., *Mechanical Vibrations*, 5th ed., Pearson Education, Inc., 2011.