ME 354: Automatic Control of Aerospace Vehicles
Final Project Report – The GLUAS

Zachary Fisher
Spring 2019

Table of Contents

Aircraft Overview

This report will examine a prototype level research project at Lehigh University called the GLUAS (gun-launched, unmanned aerial system). The design originates from the military's need for a small, easily deployable, and capable MAV (micro-aerial vehicle) to survey unknown landscapes during warfare for superior intelligence. This current design has been developed to fit inside a 60mm mortar shell with the goal of eventually being condensed into a 40mm grenade launcher barrel for even greater portability and usability.

This aircraft relies only on only two elevons to control flight. Although this aircraft's design calls for 10 individually flexible wing segments, for the purposes of this control system development, the wing will be assumed to be a more like a semicircle.

Figure 1: Top View of GLUAS

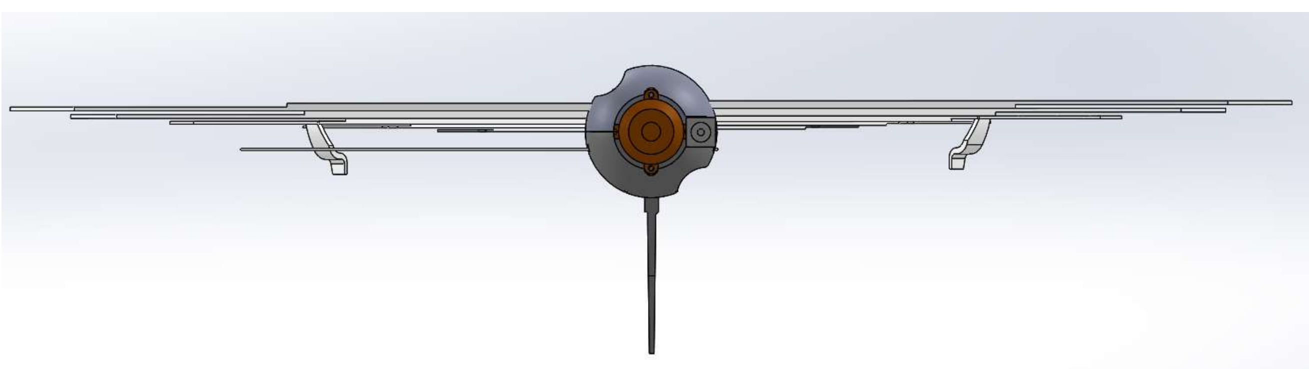

Figure 2: Front View of GLUAS

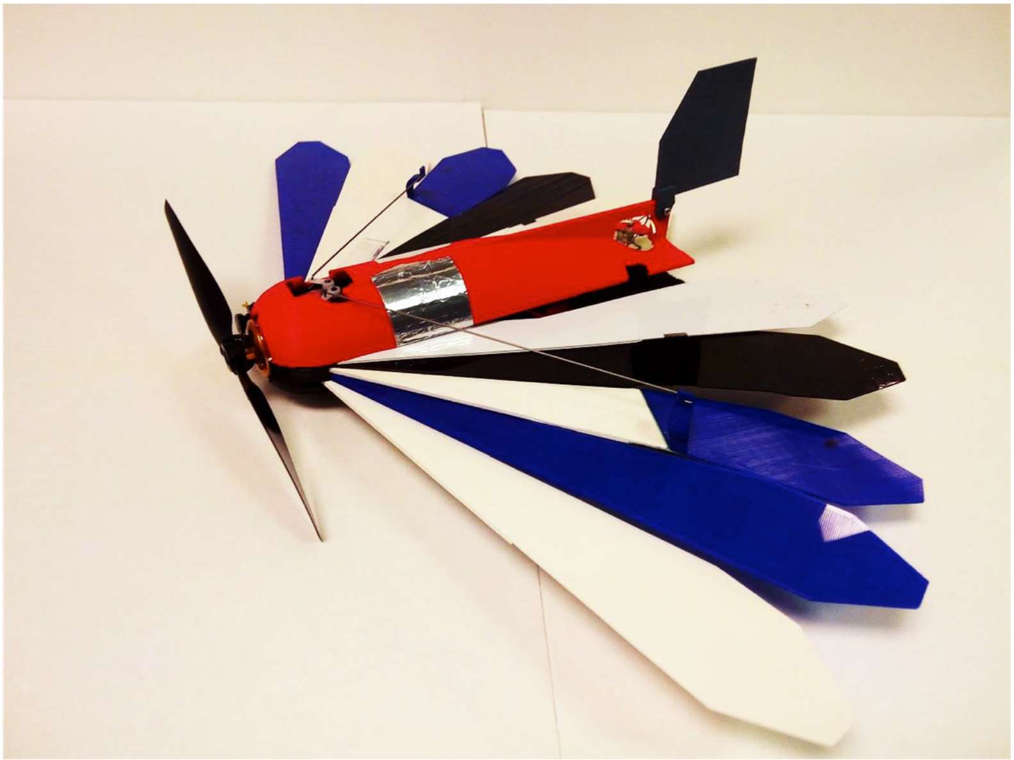
Figure 3: 3D Printed Functional Prototype

Table 1: Aircraft Data

Physical Parameter	Variable	Value (Imperial)	Value (S.I.)
Wing Span	b	1.65 ft	0.504 m
Wing Root Chord	c_r	0.00066 ft	0.201 m
Wing Tip Chord	c_t	0 ft	0 m
Y Distance to MAC	Y_{MAC}	0.367 ft	0.112 m
Wing Surface Area	S	0.9534 ft^2	0.089 m^2
Total Vertical Fin Area	S_V	0.0107 ft^2	0.0099 m^2
Total Aileron Surface Area	S_{ail}	0.1044 ft^2	0.0097 m^2
Aspect Ratio of Vertical Fin	A_V	-	8.177
Aspect Ratio of Wing	A	-	2.856
Dihedral	Γ	-	0 degrees
Geometric Twist	ε	-	0 degrees
Sweep Angle, Leading Edge	Λ_{LE}	-	15 degrees
Sweep Angle, Half Chord	$\Lambda_{c/2}$	-	-3.5 degrees
Fin Sweep Angel, Half Chord	$\Lambda_{V_{c/2}}$	-	7.79 degrees
Total Elevon Area	-	0.1044 ft^2	0.0097 m^2
Total Mass	m	0.558 lbs	253 grams
Cruising speed	u	49.9 ft/s	15.2 m/s
Angle of Attack	α_0	-	3 degrees
Moment of inertia, x	I_{xx}	0.000751 slug*ft^2	1019488 g*mm^2
Moment of inertia, y	I_{yy}	0.00117 slug*ft^2	1592093 g*mm^2
Moment of inertia, z	I_{zz}	0.00188 slug*ft^2	2549070 g*mm^2

The output coordinate system in place in SolidWorks is different than what is traditionally used for an aircraft. The Figure 4 screen image reflects the original moments of inertia within the SolidWorks frame for both the principal axes and the output coordinate system.

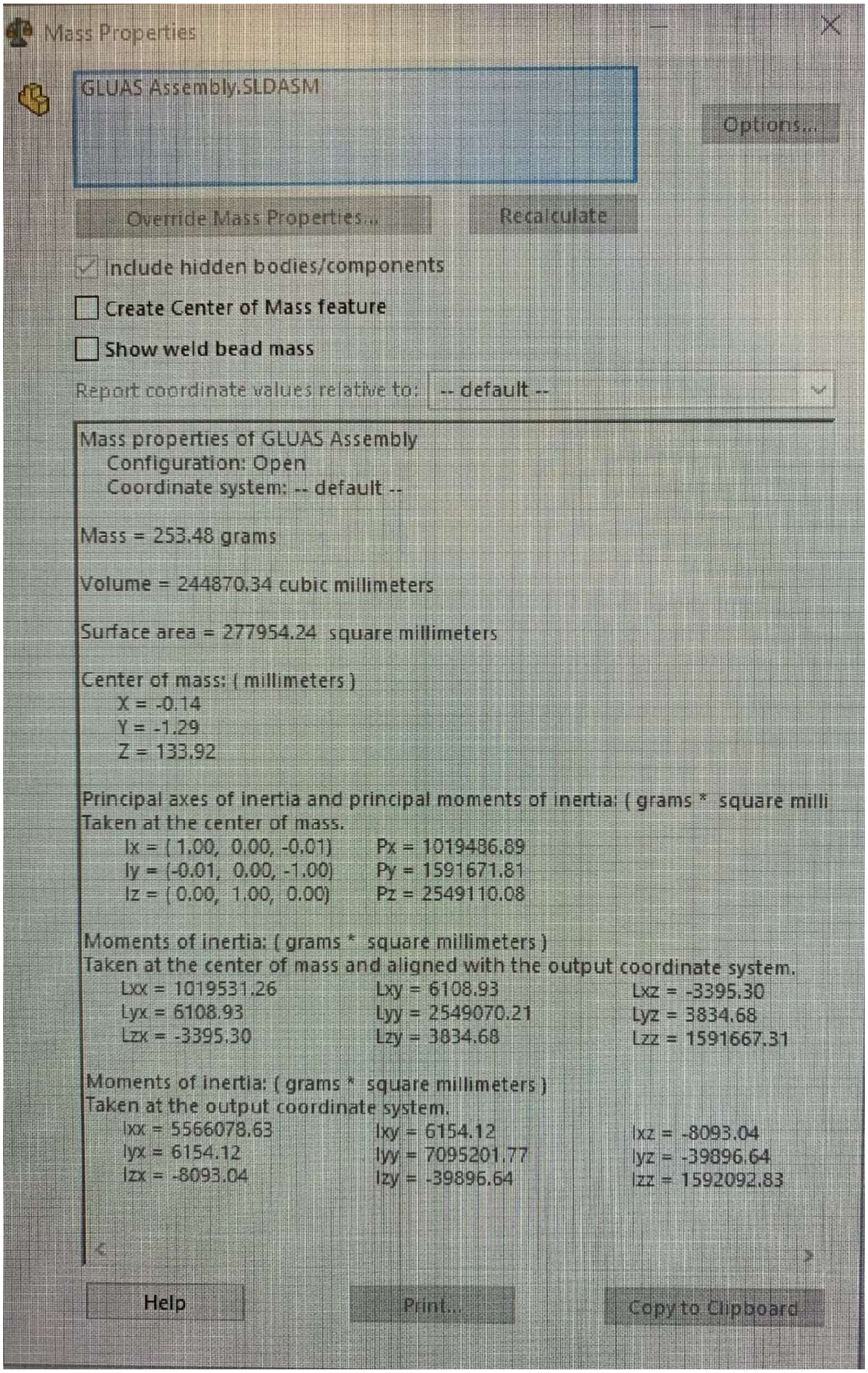

Figure 4: Mass Properties though SolidWorks

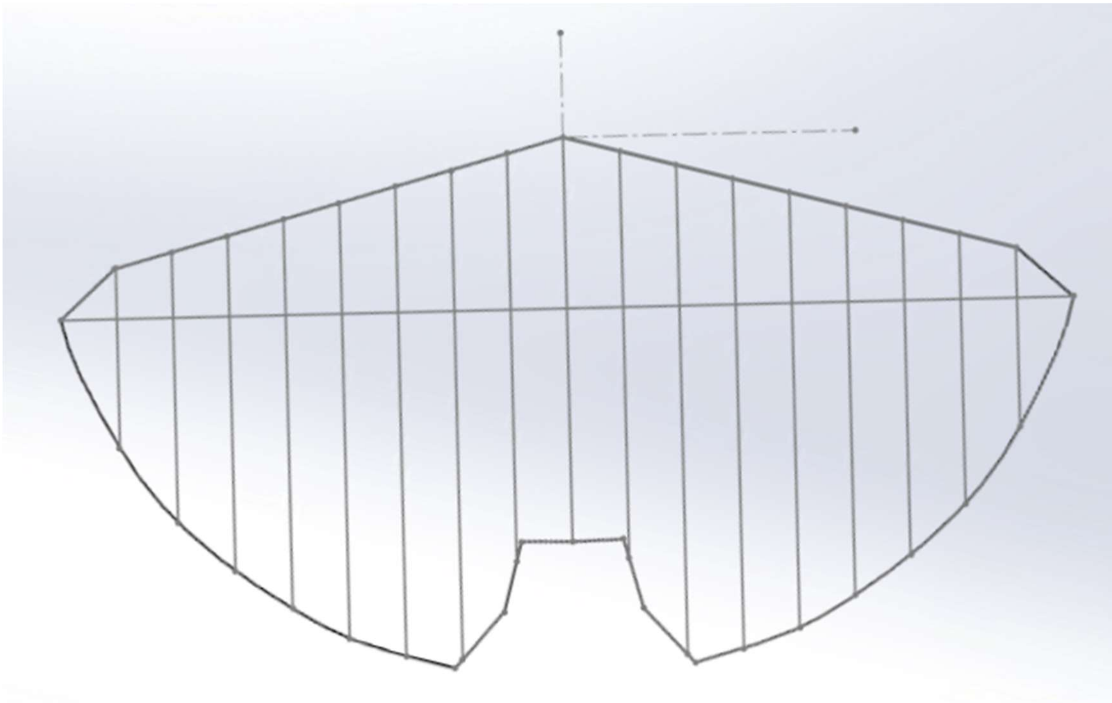

Figure 5: Simplified Wing Planform

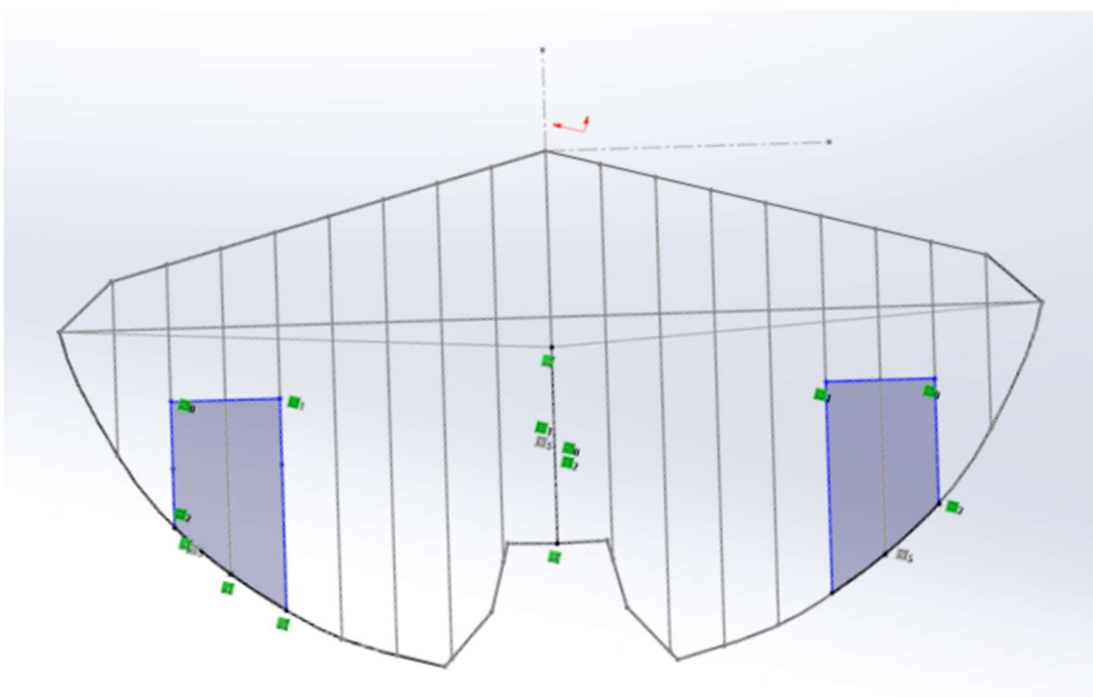

Figure 6: Simplified Wing Planform with Equivalent Elevons Areas

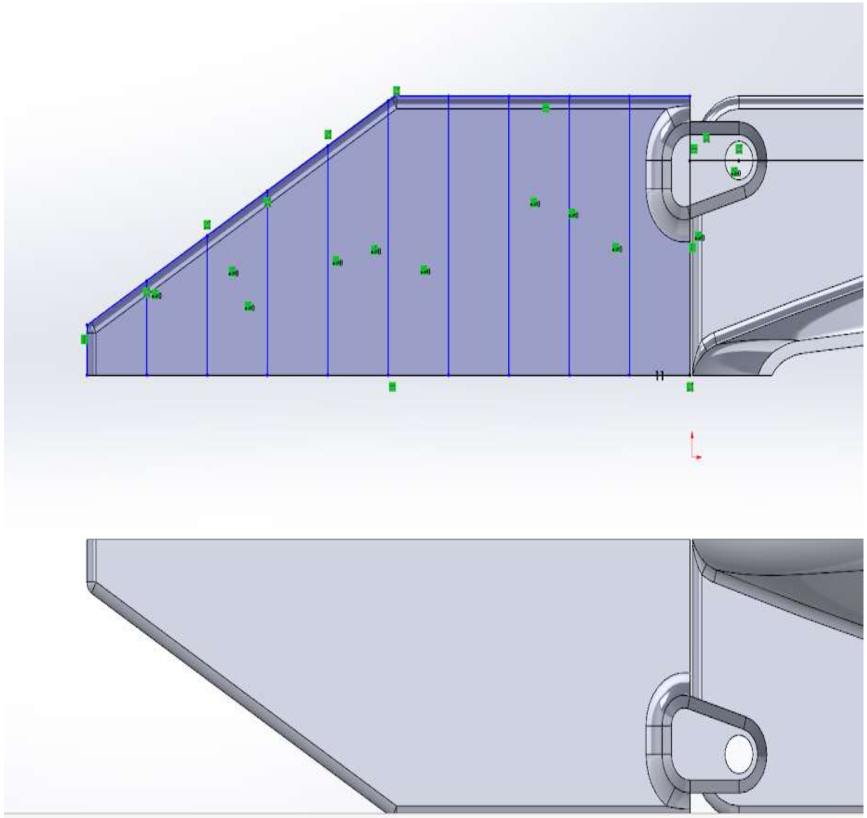

Figure 7: Tail Fin Geometry (Retracted Orientation)

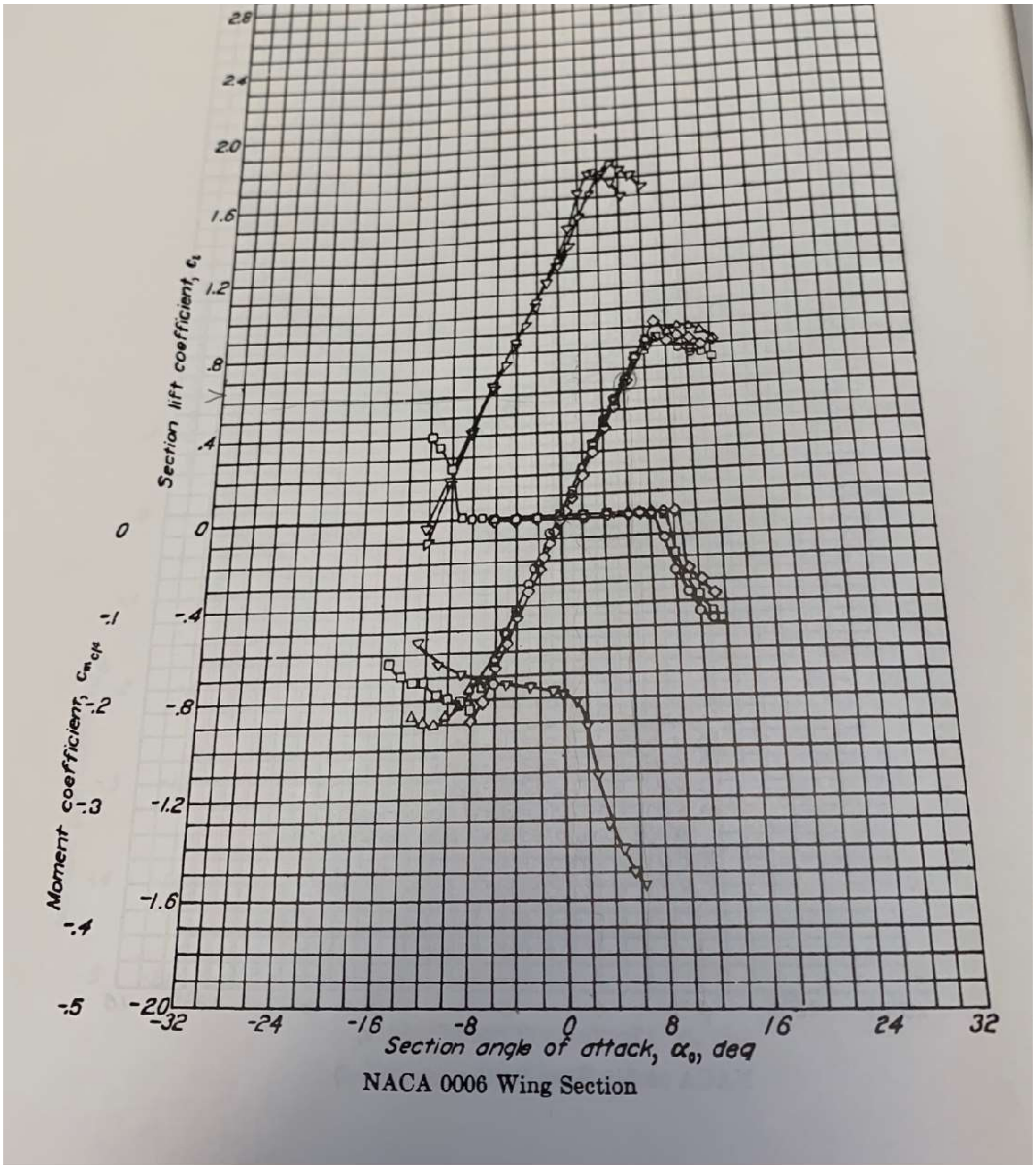

Figure 8: NACA 0006 Wing Section

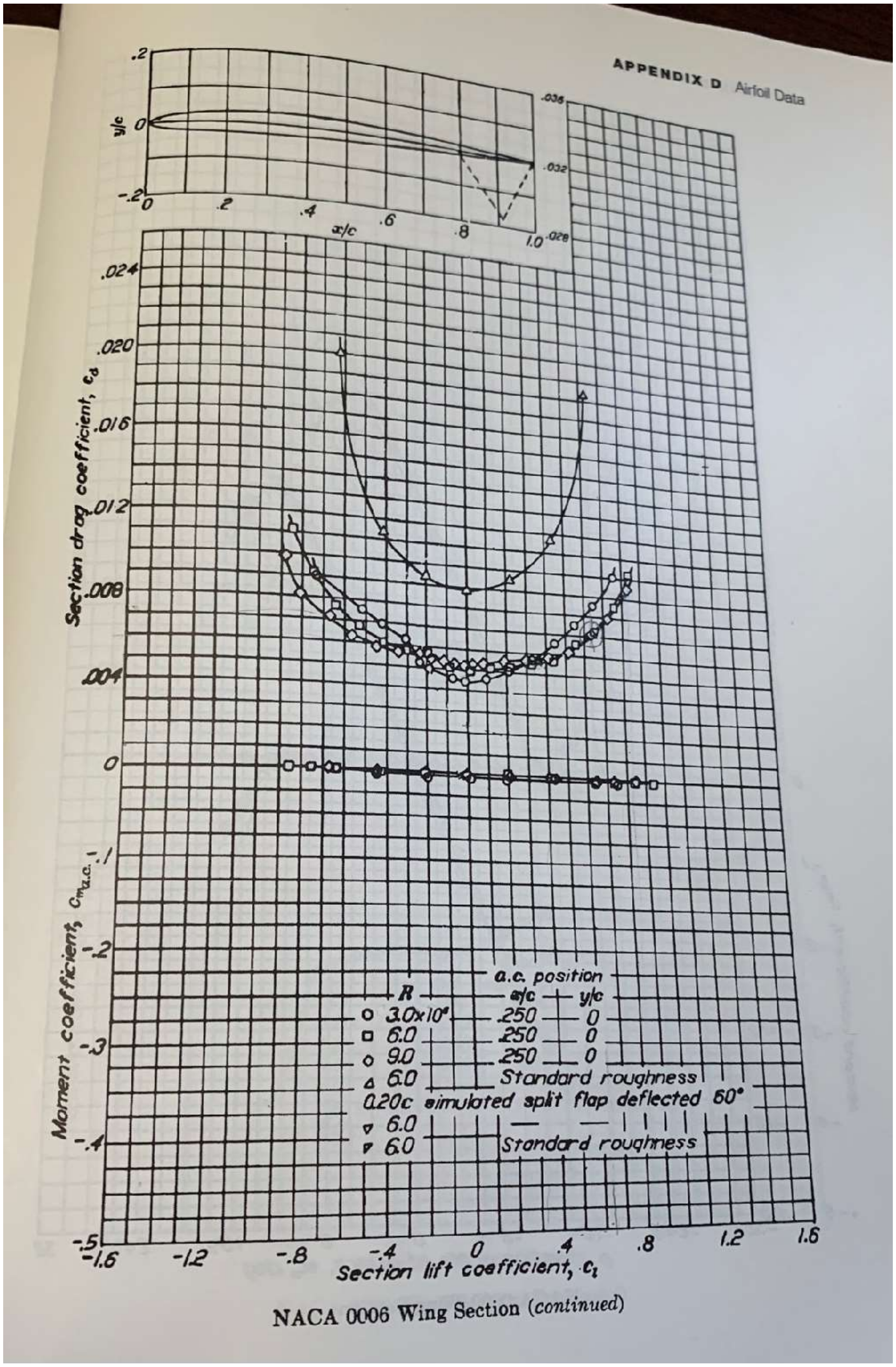

Figure 9: NACA 0006 Wing Section (continued)

MATAERO

Using the Model 5 GLUAS SolidWorks file, a created a the simplified wing geometry seen in Figure 4 above. It assumes a single, nearly flat wing with a slightly increased planform area as a result of merging the wing tip segments. To provide data regarding the airfoil-section angle of attack related to the section lift and section moment coefficients, the wing and tail fins have been assumed to be a NACA 0006 airfoil as described in Figures 7 and 8. Additionally, the wing will be treated as completely rigid with no dihedral for the purposes of these calculations.

Table 2: MATAERO Inputs

Definition	Variable	Value
Number of wings considered	NW	1
Number of airfoil sections	N	[19]
Leading edge x-coordinates	XLE	[-0.2793 -0.1969 -0.1723 -0.1477 -0.1231 -0.0985 -0.0739 -0.0492 -0.0246 0 -0.0246 -0.0492 -0.0739 -0.0985 -0.1231 -0.1477 -0.1723 -0.1969 -0.2793]
Leading edge y-coordinates	YLE	[-0.8268 -0.7366 -0.6431 -0.5512 -0.4594 -0.3675 -0.2756 -0.1837 -0.0919 0 0.0919 0.1837 0.2756 0.3675 0.4594 0.5512 0.6431 0.7366 0.8268]
Chord length of airfoil sections	CHORD	[0 0.2924 0.4408 0.5467 0.6352 0.7104 0.7662 0.7990 0.6645 0.6599 0.6645 0.7990 0.7662 0.7104 0.6352 0.5467 0.4408 0.2924 0]
Root incidence angle	ih	[0]
Vector indicating section characteristics	atwst	[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
Geometric twist	gtwst	[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Index of the wing	refwng	1
Vehicle angle of attack	alpha	3
Airfoil-section angle of attack	AOA	[-6 -4 -2 0 2 4 6 8]
Section lift coefficient	Cl	[-0.65 -0.45 -0.25 0 0.2 0.45 0.65 0.8]
Section moment coefficient	Cm	[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

Table 3: MATAERO Outputs

Definition	Variable	Value
Planform area	S	0.9534 ft^2
Mean aerodynamic chord	MAC	0.6501 ft
Aerodynamic center location	XAC	0.0709 ft
Lift coefficient	CL	0.1921
Pitching moment coefficient (about apex)	CM	-0.0263
Pitching moment coefficient about XAC	CMac	-0.0054
Mean induced downwash angle	MIA	-1.1006 degrees

AVL – MIT

Three input text files were created governing the geometry, mass, and flight conditions for the GLUAS. AVL (Athena Vortex Lattice) was then used to simulate the aircraft and calculate the associated effectiveness coefficients and therefore stability derivatives. Values like the air density (at sea level) and freestream velocity (49.9 ft/s), angle of attack (3 degrees), and other properties relating to the aircraft's mass distribution can easily be modified with subsequent iterations of the GLUAS using the text files in Appendices B, C, and D.

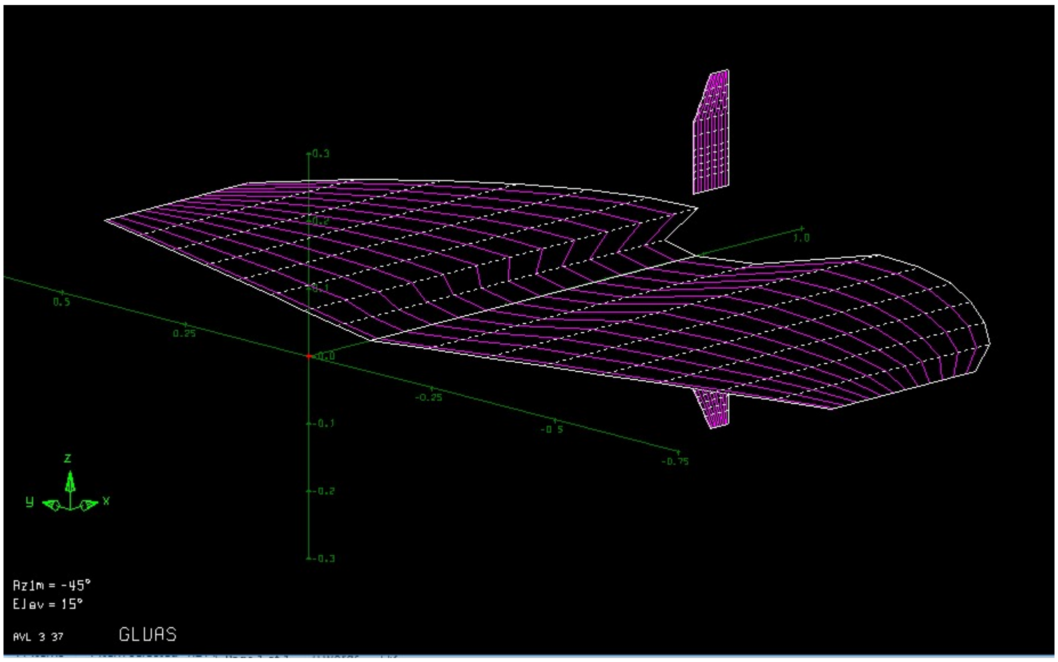

Figure 10: Simulated Geometry in AVL

AVL Outputs:

Vortex Lattice Output -- Total Forces

Configuration: GLUAS
  # Surfaces =   4
  # Strips   = 40
  # Vortices = 432

Sref = 0.95340   Cref = 0.65010   Bref = 1.6535
Xref = 0.35433   Yref = 0.0000   Zref = 0.0000

Standard axis orientation, X fwd, Z down

Run case: 0 deg bank

Alpha = 17.20577   pb/2V = -0.00000   p'b/2V = -0.00000
Beta =   0.00000   qc/2V =   0.00000
Mach =   0.044   rb/2V =   0.00000   r'b/2V =   0.00000

```
 CXtot =   0.00242     Cltot =  -0.00000     Cl'tot =  -0.00000
 CYtot =  -0.00000     Cmtot =   0.00207
 CZtot =  -0.83673     Cntot =   0.00000     Cn'tot =   0.00000

 CLtot =   0.80000
 CDtot =   0.24520
 CDvis =   0.16000     CDind = 0.0851982
 CLff  =   0.82638     CDff  = 0.0900195    | Trefftz
 CYff  =   0.00000         e =    0.8421    | Plane

  aileron        =   0.00002

---------------------------------------------------------------

Stability-axis derivatives...

                     alpha                beta
              ----------------     ----------------
z' force CL |    CLa =   2.413065     CLb =  -0.000000
y  force CY |    CYa =  -0.000001     CYb =  -0.229657
x' mom.  Cl'|    Cla =   0.000000     Clb =  -0.163864
y  mom.  Cm |    Cma =   0.006048     Cmb =   0.000000
z' mom.  Cn'|    Cna =   0.000001     Cnb =   0.070381

                 roll rate  p'              pitch rate  q'             yaw rate  r'
              ----------------          ----------------          ----------------
z' force CL |    CLp =  -0.000000       CLq =   2.873257          CLr =  -0.000000
y  force CY |    CYp =   0.262036       CYq =  -0.000000          CYr =  -0.038697
x' mom.  Cl'|    Clp =  -0.153857       Clq =   0.000000          Clr =   0.164535
y  mom.  Cm |    Cmp =   0.000000       Cmq =  -0.660064          Cmr =   0.000000
z' mom.  Cn'|    Cnp =  -0.049271       Cnq =   0.000000          Cnr =  -0.032935

              aileron      d1
              ----------------
z' force CL |   CLd1 =  -0.000000
y  force CY |   CYd1 =  -0.000775
x' mom.  Cl'|   Cld1 =  -0.001664
y  mom.  Cm |   Cmd1 =   0.000000
z' mom.  Cn'|   Cnd1 =   0.000169
Trefftz drag| CDffd1 =   0.000000
span eff.   |    ed1 =  -0.000000

Neutral point  Xnp =   0.352702

Clb Cnr / Clr Cnb  =   0.466047    (  > 1 if spirally stable )
```

Force Stability Derivative Equations

The following equations are used in the MATLAB code seen in Appendix E.

$$X_u + X_{Pu} = \frac{q_\infty S_W}{m}\left(-\left(C_{D_u} + \frac{2}{U_0}C_{D_0}\right) + \left(C_{P_{X_u}} + \frac{2}{U_0}C_{P_{X_0}}\right)\right)$$

$$X_\alpha = \frac{q_\infty S_W}{m}\left(-C_{D_\alpha} + C_{L_0}\right), \quad X_{\dot\alpha} = -\frac{q_\infty S_W}{m}C_{D_{\dot\alpha}}$$

$$X_q = -\frac{q_\infty S_W}{m}C_{D_q}, \quad X_{\delta_E} = -\frac{q_\infty S_W}{m}C_{D_{\delta_E}}, \quad X_T = -\frac{\cos\left(\phi_T + \alpha_0\right)}{m}$$

$$Y_\beta = \frac{q_\infty S_W}{m}C_{S_\beta}, \quad Y_p = \frac{q_\infty S_W}{m}C_{S_p}, \quad Y_r = \frac{q_\infty S_W}{m}C_{S_r}$$

$$Y_{\delta_A} = \frac{q_\infty S_W}{m}C_{S_{\delta_A}}, \quad Y_{\delta_R} = \frac{q_\infty S_W}{m}C_{S_{\delta_R}}$$

$$Z_u + Z_{Pu} = \frac{q_\infty S_W}{m}\left(-\left(C_{L_u} + \frac{2}{U_0}C_{L_0}\right) + \left(C_{P_{Z_u}} + \frac{2}{U_0}C_{P_{Z_0}}\right)\right)$$

$$Z_\alpha = -\frac{q_\infty S_W}{m}\left(C_{L_\alpha} + C_{D_0}\right), \quad Z_{\dot\alpha} = -\frac{q_\infty S_W}{m}C_{L_{\dot\alpha}}$$

$$Z_q = -\frac{q_\infty S_W}{m}C_{L_q}, \quad Z_{\delta_E} = -\frac{q_\infty S_W}{m}C_{L_{\delta_E}}, \quad Z_T = -\frac{\sin\left(\phi_T + \alpha_0\right)}{m}$$

Moment Stability Derivative Equations

The following equations are used in the MATLAB code seen in Appendix E.

$$L_\beta = \frac{q_\infty S_W b_W}{I_{xx}} C_{L_\beta}, \quad L_p = \frac{q_\infty S_W b_W}{I_{xx}} C_{L_p}, \quad L_r = \frac{q_\infty S_W b_W}{I_{xx}} C_{L_r}$$

$$L_{\delta_A} = \frac{q_\infty S_W b_W}{I_{xx}} C_{L_{\delta_A}}, \quad L_{\delta_R} = \frac{q_\infty S_W b_W}{I_{xx}} C_{L_{\delta_R}}$$

$$M_u + M_{Pu} = \frac{q_\infty S_W \overline{c}_W}{I_{yy}} \left(C_{M_u} + \frac{2}{U_0} C_{M_0} \right) + \left(C_{P_{M_u}} + \frac{2}{U_0} C_{P_{M0}} \right)$$

$$M_\alpha = \frac{q_\infty S_W \overline{c}_W}{I_{yy}} C_{M_\alpha}, \quad M_{P_\alpha} = \frac{q_\infty S_W \overline{c}_W}{I_{yy}} C_{P_{M_\alpha}}, \quad M_{\dot\alpha} = \frac{q_\infty S_W \overline{c}_W}{I_{yy}} C_{M_{\dot\alpha}}$$

$$M_q = \frac{q_\infty S_W \overline{c}_W}{I_{yy}} C_{M_q}, \quad M_{\delta_E} = \frac{q_\infty S_W \overline{c}_W}{I_{yy}} C_{M_{\delta_E}}, \quad M_T = \frac{d_T \cos(\phi_T) - x_T \sin(\phi_T)}{I_{yy}}$$

$$N_\beta = \frac{q_\infty S_W b_W}{I_{zz}} C_{N_\beta}, \quad N_p = \frac{q_\infty S_W b_W}{I_{zz}} C_{N_p}, \quad N_r = \frac{q_\infty S_W b_W}{I_{zz}} C_{N_r}$$

$$N_{\delta_A} = \frac{q_\infty S_W b_W}{I_{zz}} C_{N_{\delta_A}}, \quad N_{\delta_R} = \frac{q_\infty S_W b_W}{I_{zz}} C_{N_{\delta_R}}$$

Longitudinal State Space Model

$$A = \begin{pmatrix} \left(X_u + X_{P_u} + \frac{X_{\dot\alpha}\left(Z_u + Z_{P_u}\right)}{U_0 - Z_{\dot\alpha}}\right) & \left(X_\alpha + \frac{X_{\dot\alpha} Z_\alpha}{U_0 - Z_{\dot\alpha}}\right) & -g & \left(X_q + X_{\dot\alpha}\left(\frac{U_0 + Z_q}{U_0 - Z_{\dot\alpha}}\right)\right) & 0 \\ \left(\frac{Z_u + Z_{P_u}}{U_0 - Z_{\dot\alpha}}\right) & \left(\frac{Z_\alpha}{U_0 - Z_{\dot\alpha}}\right) & 0 & \left(\frac{U_0 + Z_q}{U_0 - Z_{\dot\alpha}}\right) & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \left(M_u + M_{P_u} + \frac{M_{\dot\alpha}\left(Z_u + Z_{P_u}\right)}{U_0 - Z_{\dot\alpha}}\right) & \left(M_\alpha + M_{P_\alpha} + \frac{M_{\dot\alpha} Z_\alpha}{U_0 - Z_{\dot\alpha}}\right) & 0 & \left(M_q + M_{\dot\alpha}\left(\frac{U_0 + Z_q}{U_0 - Z_{\dot\alpha}}\right)\right) & 0 \\ 0 & -U_0 & U_0 & 0 & 0 \end{pmatrix}$$

$A =$

-0.03244	8.75432	-32.2	0	0
-0.007409	-0.260840	0	1	0
0	0	0	1	0
0.820363	38.36785	0	-1145.9617	0
0	-49.9	49.9	0	0

$$B = \begin{pmatrix} \left(X_{\delta_E} + \frac{X_{\dot\alpha} Z_{\delta_E}}{U_0 - Z_{\dot\alpha}}\right) & \left(X_T + \frac{X_{\dot\alpha} Z_T}{U_0 - Z_{\dot\alpha}}\right) \\ \left(\frac{Z_{\delta_E}}{U_0 - Z_{\dot\alpha}}\right) & \left(\frac{Z_T}{U_0 - Z_{\dot\alpha}}\right) \\ 0 & 0 \\ \left(M_{\delta_E} + \frac{M_{\dot\alpha} Z_{\delta_E}}{U_0 - Z_{\dot\alpha}}\right) & \left(M_T + \frac{M_{\dot\alpha} Z_T}{U_0 - Z_{\dot\alpha}}\right) \\ 0 & 0 \end{pmatrix}$$

$B =$

0	-1.79211
-.0.35988	0
0	0
-1360.509	0
0	0

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Lateral State Space Model

$$A = \begin{pmatrix} \frac{Y_\beta}{U_0} & \frac{g}{U_0} & \frac{Y_p}{U_0} & \left(\frac{Y_r}{U_0} - 1\right) & 0 \\ 0 & 0 & 1 & 0 & 0 \\ L'_\beta & 0 & L'_p & L'_r & 0 \\ N'_\beta & 0 & N'_p & N'_r & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$A =$

-0.0004050	0.64529	0	-1	0
0	0	1	0	0
-11	0	0	0	0
172.7720	0	-120.9510	-80.8492	0
0	0	0	1	0

$$B = \begin{pmatrix} \frac{Y_{\delta_A}}{U_0} & \frac{Y_{\delta_R}}{U_0} \\ 0 & 0 \\ L'_{\delta_A} & L'_{\delta_R} \\ N'_{\delta_A} & N'_{\delta_R} \\ 0 & 0 \end{pmatrix}$$

$B =$

0	0
0	0
-1.02134	0
0.41486	0
0	0

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Control Input Response Behavior

With respect to time, each of the five graphs per figure correspond to the sideslip angle, bank angle, roll rate, yaw rate, and heading angle. For the longitudinal, inputs consist of the elevator deflection and thrust. For the lateral, inputs consist of the aileron deflection and rudder deflection.

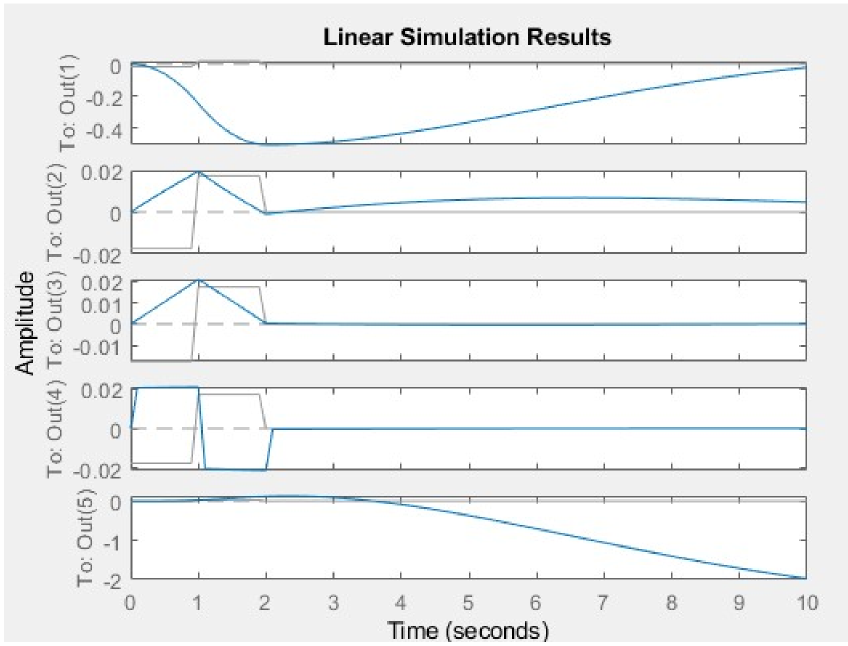

Figure 9: Longitudinal, 2-Second Elevator Doublet Response

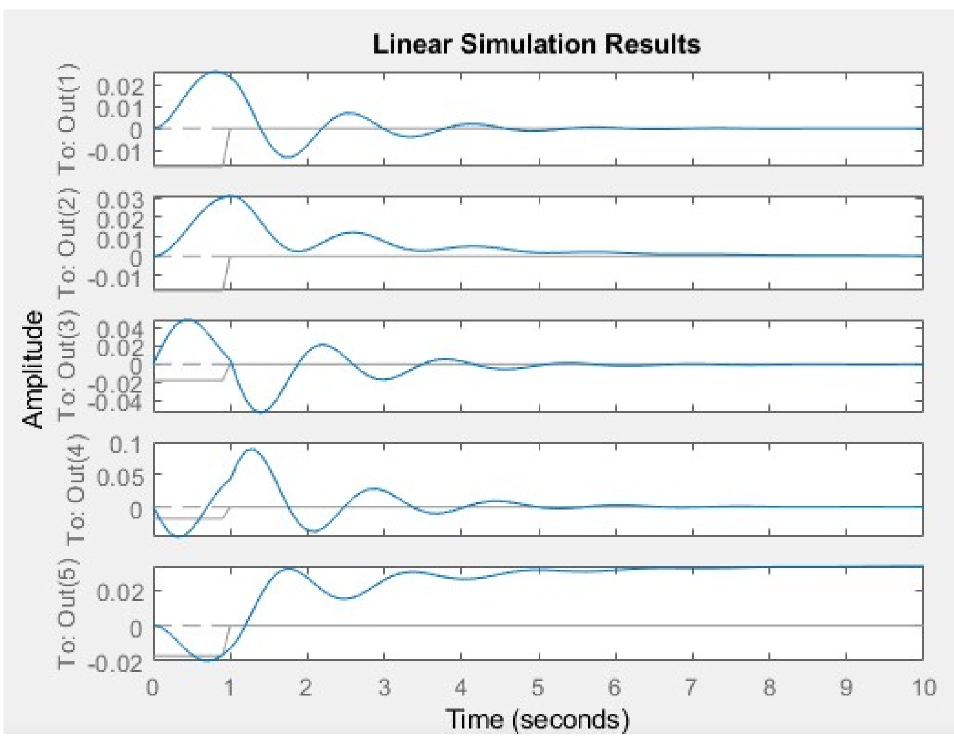

Figure 10: Lateral, 1-Second Aileron Input Response

The eigenvalues for the longitudinal “A” matrix are as follow:

0

-1145.995

-0.14030 + 0.27418i

-0.14030 - 0.27418i

0.020817

The eigenvalues for the lateral “A” matrix are as follow:

0

-0.44971

-0.76369 + 3.94920

-0.76369 – 3.94920

-78.8725

Pitch and Yaw Dampers

The "A" matrices for both the longitudinal and lateral dynamics were augmented with a specific gain. This gain "K" was applied to the existing state space matrices to generate the new "A" matrices as seen below:

$$A_{aug} = A - B * K * C$$

The longitudinal dynamics were improved as seen by the reduced maximum amplitude of the first peak. The lateral dynamics were significantly improved with this method as seen in Figure 12. The oscillation has been damped and each plot reaches its steady state value faster.

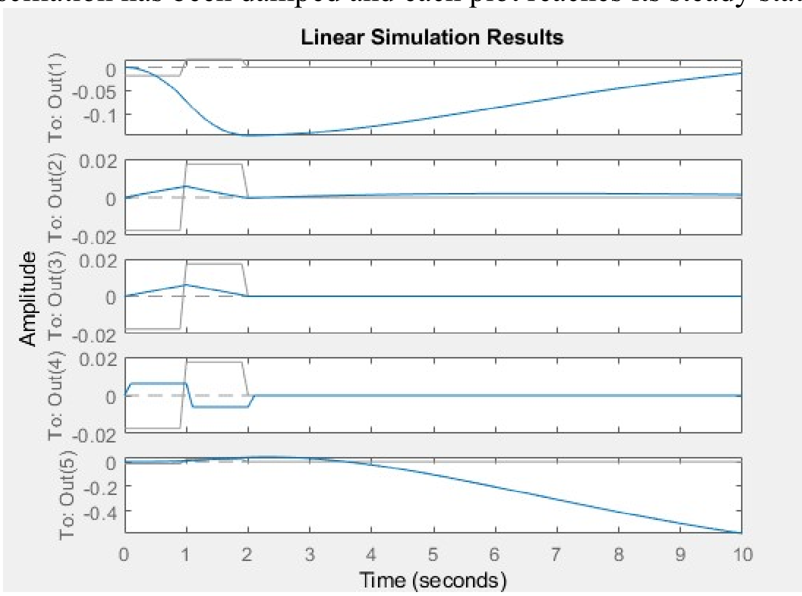

Figure 11: Augmented Longitudinal, 2-Second Elevator Doublet Response

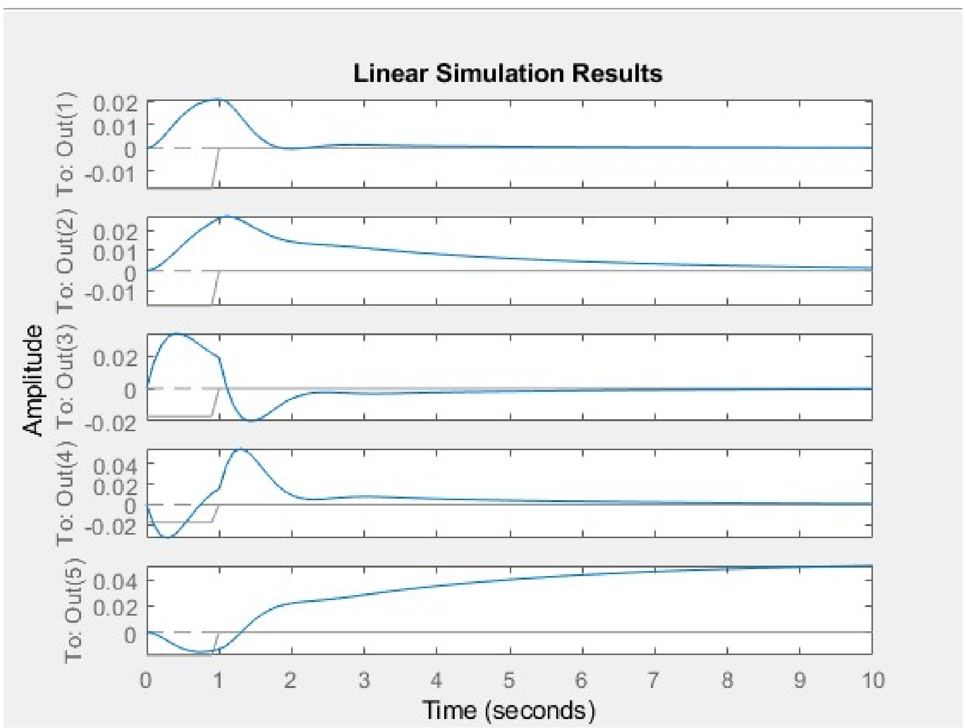

Figure 12: Augmented Lateral, 1-Second Aileron Input Response

The eigenvalues for the augmented longitudinal “A” matrix are as follow:

$$0$$
$$-3866.990$$
$$-0.14568 + 0.24296i$$
$$-0.14568 - 0.24296i$$
$$0.007292$$

The eigenvalues for the augmented lateral “A” matrix are as follow:

$$0$$
$$-75.9526$$
$$-0.301378$$
$$-2.33517 + 3.26911i$$
$$-2.33517 - 3.26911i$$

Autopilot Altitude Response

Using the augmented longitudinal system, a P.I. compensator of (s+3)/s, lead compensator of (s+0.0001)/(s+1), and gain of -0.00019, an open loop altitude system was created. From there, the loop was closed as shown in block diagram form in Figure 13. The Bode diagram in Figure 14 illustrates the behavior of that open loop system with a phase margin of 66 degrees at 0.07 rad/s and a gain margin of 9.4 dB at 0.2 rad/s. When a step change of 100 feet is mandated on the autopilot system, the system takes about 25 seconds to reach that level as shown in Figure 15. There is almost no overshoot nor oscillation.

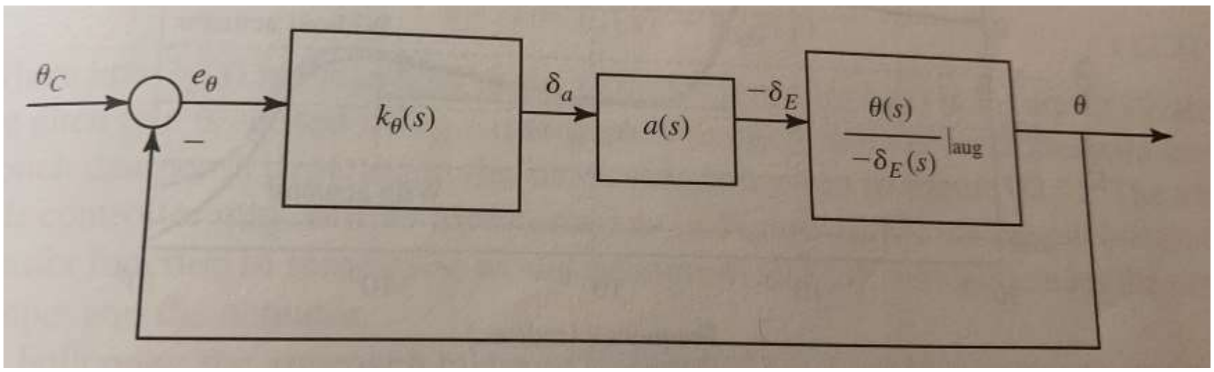

Figure 13: Pitch-Attitude Control Loop Block Diagram

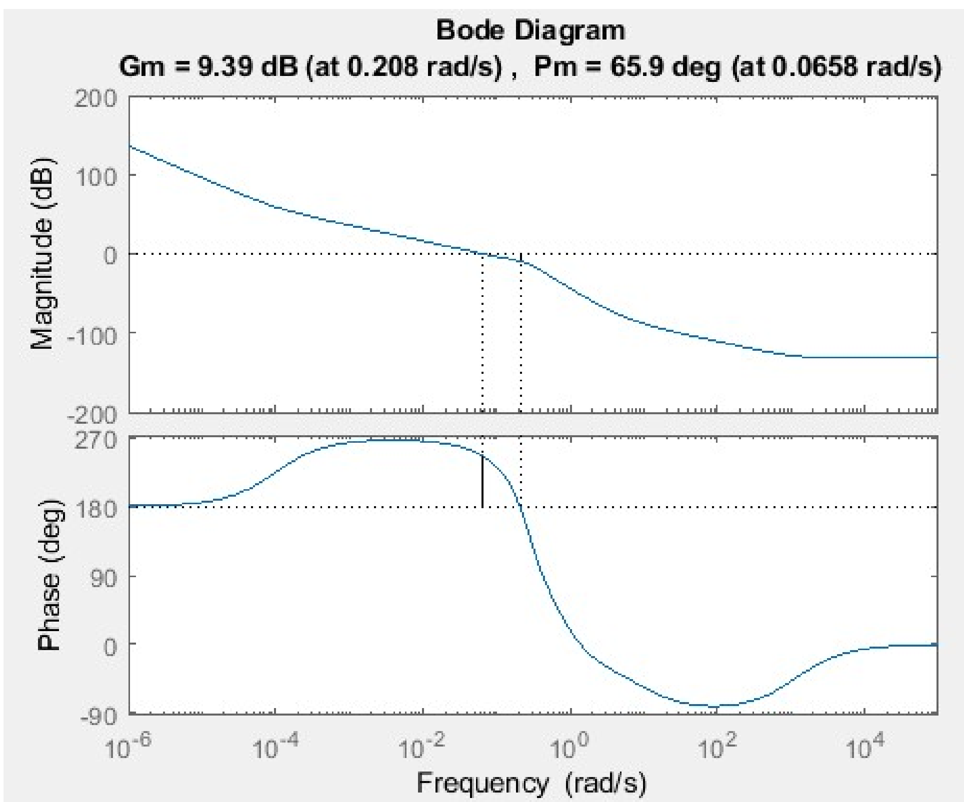

Figure 14: Bode Plot of Compensated Open Loop System

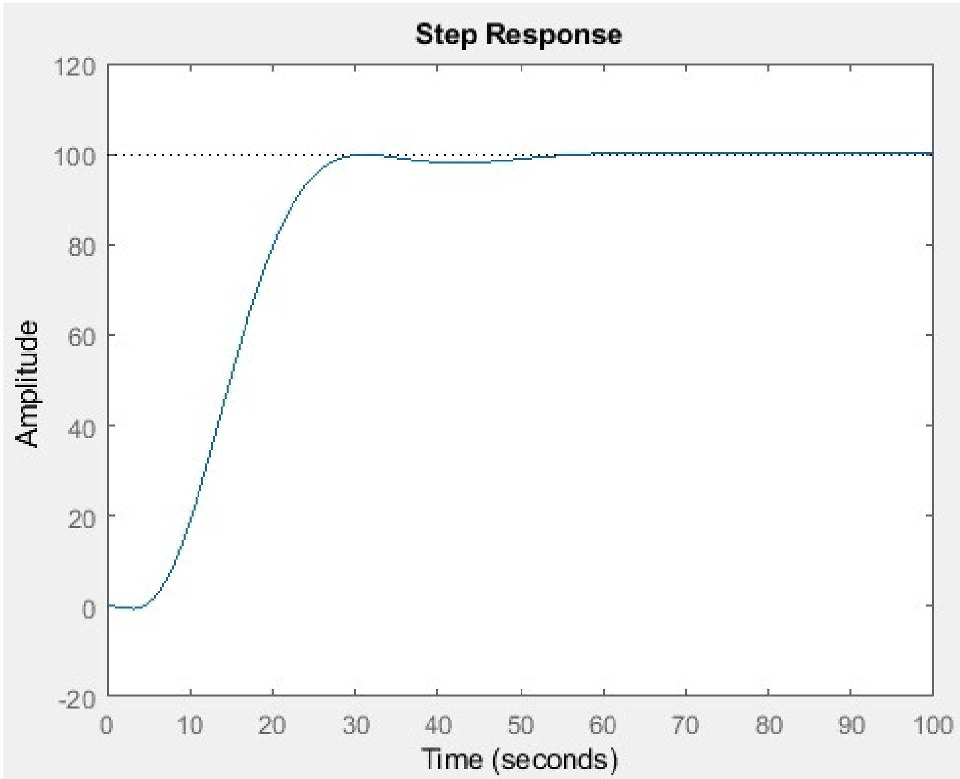

Figure 15: Altitude Response Behavior

Autopilot Heading Response

Using the augmented lateral system, a gain of -0.05, a transfer function based on air speed, an open loop system was created. Subsequently the loop was closed as represented in Figure 16. The collective responses to a 3 degree heading change on β *(rad),* ϕ *(rad), p (rad/sec), r (rad/sec),* and ψ *(rad)* are captured in Figure 17. Figure 18 shows just the heading response (in radians) and takes about 22 seconds to steady out near the target heading change. There is slight overshoot and minimal oscillation.

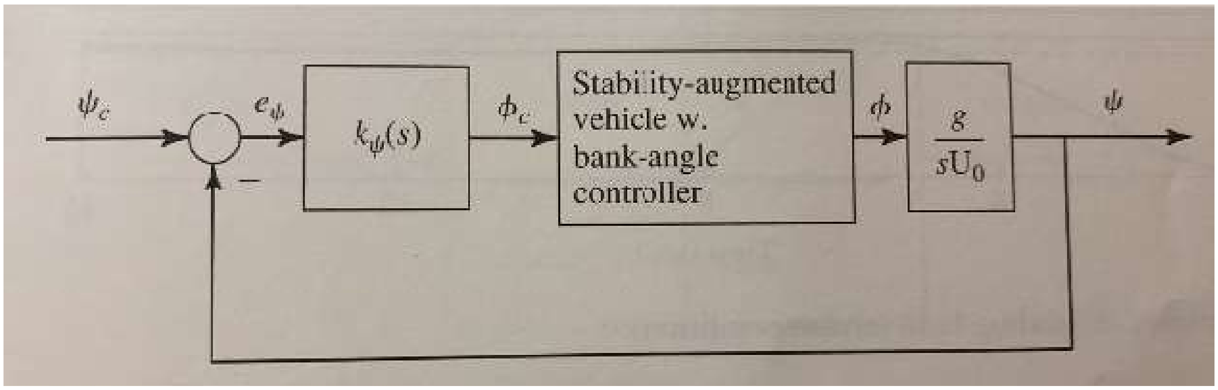

Figure 16: Heading Hold Block Diagram

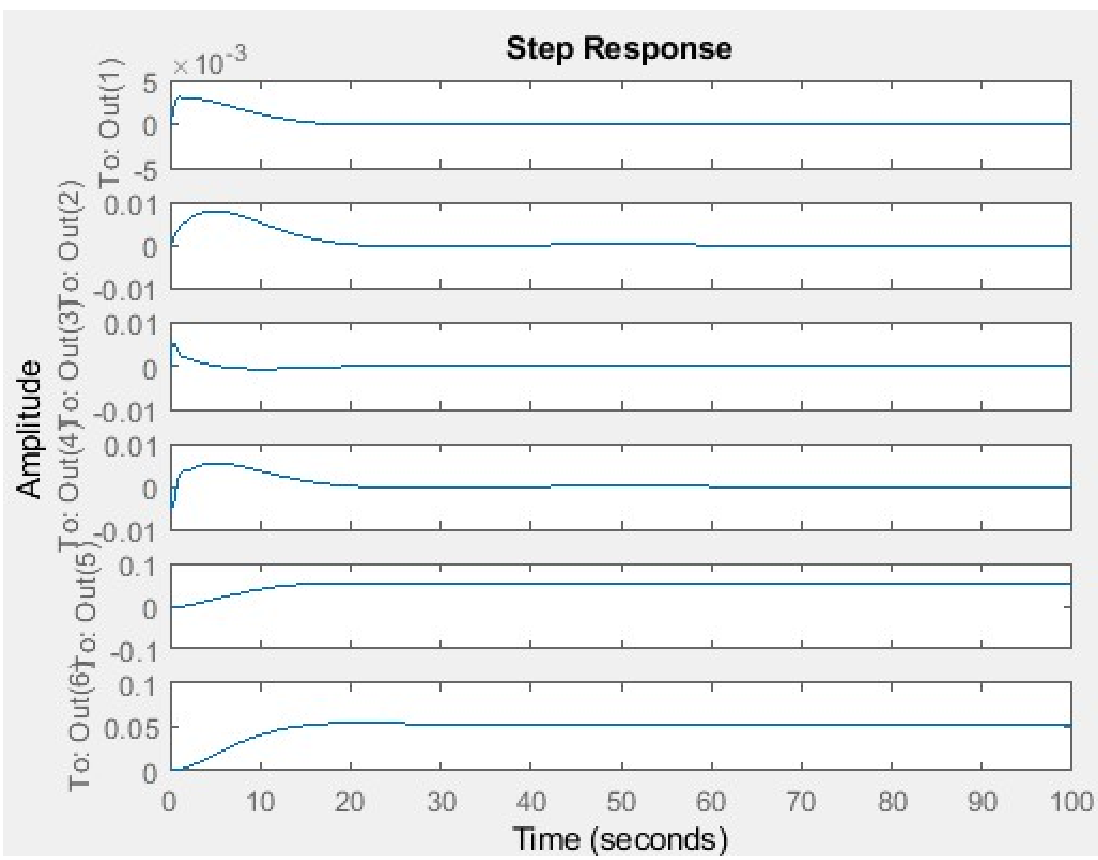

Figure 17: Overall Response to Heading Change

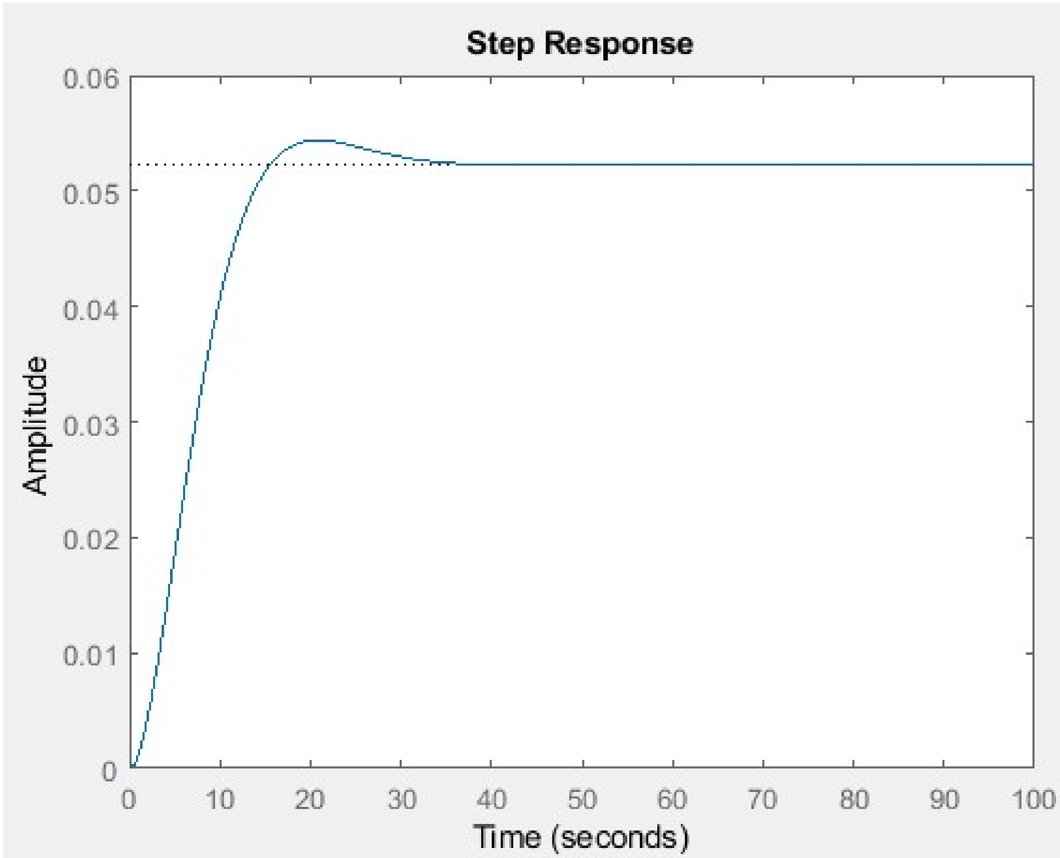

Figure 18: Heading Specific Response

Appendix A: MATAERO MATLAB Code

```
% MATAERO
%
% Copyright © 2011 by David K. Schmidt,
% published and distributed by McGraw-Hill with permission.
%
% This program computes the lift and moment coefficients for
% multiple wing surfaces. The tip boundary conditions of cl=0 is enforced.

NW=1; %Number of wings
N=[19]; %Number of segments subdividing wing
XLE=[-0.2793 -0.1969 -0.1723 -0.1477 -0.1231 -0.0985 -0.0739 -0.0492 -0.0246
0 -0.0246 -0.0492 -0.0739 -0.0985 -0.1231 -0.1477 -0.1723 -0.1969 -0.2793];
%Leading Edge x-coord for each segment (dim is value of N)
YLE=[-0.8268 -0.7366 -0.6431 -0.5512 -0.4594 -0.3675 -0.2756 -0.1837 -0.0919
0 0.0919 0.1837 0.2756 0.3675  0.4594 0.5512 0.6431 0.7366 0.8268]; %LE y-
coord along wing
CHORD=[0 0.2924 0.4408 0.5467 0.6352 0.7104 0.7662 0.7990 0.6645 0.6599
0.6645 0.7990 0.7662 0.7104 0.6352 0.5467 0.4408 0.2924 0];
ih=[0]; %dim equal to val of NW
atwst=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
gtwst=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
refwng=1;
alpha=3;
AOA=[-6 -4 -2 0 2 4 6 8];
Cl=[-0.65 -0.45 -0.25 0 0.2 0.45 0.65 0.8];
Cm=[0 0 0 0 0 0 0 0];

clear G S MAC CL CM MIA
%
% Calculate the 1/4 chord location of each wing section
% and initialize the circulation distribution
%
k=0;
for i=1:NW,
  for j=1:N(i),
    k=k+1;
    xqc(k)=XLE(k)+0.25*CHORD(k);
    aeff=alpha+ih(i)+gtwst(k);
    sctn=atwst(k);
    [cl,cm]=gtclcm(sctn,aeff,AOA,Cl,Cm);
    gamma(k,1)=0.5*CHORD(k)*cl; % Does not include Vinfinity
  end
end
G=gamma;
clear k i j aeff sctn cl cm
%
% Get the Influence Coefficient Matrix
%
AIC=gtAIC(NW,N,xqc,YLE);
%
% Get the zero lift angles of attack for the tip sections
% and initialize the offset distances for the tip vorticies
%
k=0;
```

```
indx=0;
for i=1:NW,
  k=k+1;
  indx=indx+1;
  aol(indx)=fndzlaoa(atwst(k),Cl,AOA);
  offst(indx)=0.0;
  k=k+N(i)-1;
  indx=indx+1;
  aol(indx)=fndzlaoa(atwst(k),Cl,AOA);
  offst(indx)=0.0;
end
clear k indx i
%
% Begin the interation process
%
delta=.7;
cmax=7;
for count=2:cmax,
%
% set offset values such that cl=0 at the tips
  offst=tipbc(NW,N,xqc,YLE,gamma,offst,AIC,alpha,ih,gtwst,aol);
%
% get the induced angles of attack
  aind=gtinducd3(NW,N,xqc,YLE,gamma,offst,AIC);
%
% calculate the new circulation distribution
  k=0;
  for i=1:NW,
    for j=1:N(i),
      k=k+1;
      aeff=alpha+ih(i)+gtwst(k)+aind(k);
      sctn=atwst(k);
      [cl(k),cm(k)]=gtclcm(sctn,aeff,AOA,Cl,Cm);
      newgam=0.5*CHORD(k)*cl(k);
      gamma(k,1)=gamma(k,1)+delta*(newgam-gamma(k,1));
    end
  end
  G(:,count)=gamma;
%
% repeate this procedure
end
%cl
%cm
%aind
clear xqc AIC aol gamma count offst k i j aeff sctn newgam
%
% Now, calculate Lift and Moment coefficients
%
[S,MAC,XAC,CL,CM,MIA]=gtCLaCM(NW,N,XLE,YLE,CHORD,cl,cm,aind);
Sref=S(refwng);
MACref=MAC(refwng);
for i=1:NW,
    CMac(i)=CM(i)+CL(i)*XAC(i)/MAC(i);
end
CLtotal=0.0;
CMtotal=0.0;
CM0tot=0.0;
```

```
for i=1:NW,
  CLtotal=CLtotal+CL(i)*(S(i)/Sref);
  CMtotal=CMtotal+CM(i)*(S(i)/Sref)*(MAC(i)/MACref);
  CM0tot=CM0tot+CMac(i)*(S(i)/Sref)*(MAC(i)/MACref);
end
clear i
S
Sref
MAC
MACref
XAC
%XACref
CL
CLtotal
CM
CMtotal
CMac
CM0tot
MIA
```

Appendix B: AVL – MIT, Geometry File

```
GLUAS
0.0443                  Mach
0    0    0.0        iYsym  iZsym  Zsym
0.9534   0.6501  1.6535    Sref   Cref   Bref    reference area, chord, span
0.354331  0.0   0.0       Xref   Yref   Zref    moment reference location
#
#
#
#=================================================
SURFACE
Wing
12  0   8 0   ! Nchord  Cspace   Nspan  Sspace
INDEX
1
ANGLE
0.0
YDUPLICATE
0.0
SCALE
1  1  1
TRANSLATE
0.0  0.0  0.0

#--------------------------------------------
SECTION
0.1247  0    0.0     0.6599   0    1  0
NACA
0006
#---------------------
SECTION
0.1493  0.0919  0.0    0.6645   0    1 0
NACA
0006
#---------------------
SECTION
0.1739  0.1837  0.0    0.7990   0    1 0
NACA
0006
#-----------------------------------------------
SECTION
0.1986  0.2756  0.0    0.7662   0    1 0
NACA
0006
#-----------------------------------------------
```

```
SECTION
0.2232  0.3675  0.0    0.7104   0    1 0
NACA
0006
#-------------------------------------------------
SECTION
0.2478  0.4594  0.0    0.6352   0    1 0
NACA
0006
CONTROL
#name gain Xhinge XYZhvec SgnDup
aileron  1.0  0.524  0 1 0  -1
#-------------------------------------------------
SECTION
0.2724  0.5512  0.0    0.5467   0    1 0
NACA
0006
CONTROL
aileron  1.0  0.524  0 1 0  -1
#-------------------------------------------------
SECTION
0.2970  0.6431  0.0    0.4408   0    1 0
NACA
0006
CONTROL
aileron  1.0  0.524  0 1 0  -1
#-------------------------------------------------
SECTION
0.3216  0.7366  0.0    0.2924   0   1 0
NACA
0006
#-------------------------------------------------
#Missing the last wing tip section b/c AVL section limit

#==============================================================
SURFACE
Fin Top
10 0  12 0   ! Nchord  Cspace   Nspan  Sspace
ANGLE
0.0
SCALE
1 1 1
TRANSLATE
0.0 0.0  0.0

#-------------------------------------------------
```

```
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7792   0   0.0927  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7792   0   0.1139  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7792   0   0.1351  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7792   0   0.1564  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7792   0   0.1776  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7804   0   0.1988  0.0697    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.7918   0   0.2201  0.0582    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle    Chord   Ainc  Nspanwise  Sspace
0.8032   0   0.2413  0.0468    0
NACA
```

```
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle   Chord   Ainc  Nspanwise  Sspace
0.8146   0   0.2625  0.0354    0
NACA
0006
#-------------------------------------------------
#missing last two sections of tail fin b/c AVL won't accept more sections

#==============================================================
SURFACE
Fin Bottom
10 0  12 0   ! Nchord  Cspace   Nspan  Sspace
ANGLE
0.0
SCALE
1 1 1
TRANSLATE
0.0 0.0  0.0

#-------------------------------------------------
SECTION
!Xle   Yle   Zle   Chord   Ainc  Nspanwise  Sspace
0.7792   0   -0.0927  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle   Chord   Ainc  Nspanwise  Sspace
0.7792   0   -0.1139  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle   Chord   Ainc  Nspanwise  Sspace
0.7792   0   -0.1351  0.0709    0
NACA
0006
#-------------------------------------------------
SECTION
!Xle   Yle   Zle   Chord   Ainc  Nspanwise  Sspace
0.7792   0   -0.1564  0.0709    0
NACA
0006
#-------------------------------------------------
```

```
SECTION
!Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
0.7792    0   -0.1776  0.0709     0
NACA
0006
#------------------------------------------------
SECTION
!Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
0.7804    0   -0.1988  0.0697     0
NACA
0006
#------------------------------------------------
SECTION
!Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
0.7918    0   -0.2201  0.0582     0
NACA
0006
#------------------------------------------------
SECTION
!Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
0.8032    0   -0.2413  0.0468     0
NACA
0006
#------------------------------------------------
SECTION
!Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
0.8146    0   -0.2625  0.0354     0
NACA
0006
#------------------------------------------------
```

Appendix C: AVL – MIT, Mass File

```
#
#  GLUAS
#  Mass & Inertia breakdown
#
#  xyz is location of item's own CG
#  Ixx.. are item's inertias about item's own CG
#
#   x back
#   y right
#   z up
#
#  x,y,z system here must have origin
#  at same location as AVL input file
#
Lunit = 1.0 ft
Munit = 1.0 slug
Tunit = 1.0 s

g   = 32.18
rho = 0.002378

#
#  mass   x     y     z      Ixx     Iyy    Izz   [ Ixy  Ixz  Iyz ]
 0.0173 0.02463 0  0.2239   1.350   0.7509  2.095
```

Appendix D: AVL – MIT, Run File

```
 ---------------------------------------------
 Run case  1:  0 deg bank

 alpha        ->  CL          =   0.80000
 beta         ->  beta        =   0.00000
 pb/2V        ->  pb/2V       =   0.00000
 qc/2V        ->  qc/2V       =   0.00000
 rb/2V        ->  rb/2V       =   0.00000
 aileron      ->  Cl roll mom =   0.00000
 elevator     ->  Cm pitchmom =   0.00000
 rudder       ->  Cn yaw  mom =   0.00000

 alpha     =   3.0         deg
 beta      =   0.00000     deg
 pb/2V     =   0.00000
 qc/2V     =   0.00000
 rb/2V     =   0
 CL        =   0.8
 CDo       =   0.16
 bank      =   0.00000     deg
 elevation =   0.00000     deg
 heading   =   0.00000     deg
 Mach      =   0.04430
 velocity  =   49.8700     ft/s
 density   =  0.0023769    slug/ft^3
 grav.acc. =   32.1800     ft/s^2
 turn_rad. =   0.00000     ft
 load_fac. =   1.00000
 X_cg      =   0.354331
 Y_cg      =   0.00000
 Z_cg      =   0.00000
 mass      =  0.017343     slug
 Ixx       =  0.0007519    slug-ft^2
 Iyy       =  0.001170     slug-ft^2
 Izz       =   0.00188     slug-ft^2
 Ixy       =   0.0         slug-ft^2
 Iyz       =   0.0         slug-ft^2
 Izx       =   0.0         slug-ft^2
 visc CL_a =   0.00000
 visc CL_u =   0.00000
 visc CM_a =   0.00000
 visc CM_u =   0.00000
```

Appendix E: GLUAS MATLAB Code

```
clc; clear all;
format long g

%Aircraft Specs
g=32.2; %accel of grav (ft/s^2)
b_w=1.635; %wing span 0.504 (m) or 1.6535 ft
sweep_LE=15; %Leading edge sweep (deg)
sweep_c2=-3.51; %Half chord sweep (deg)
sweep_vc2=7.79; %tail fin half chord sweep (deg)
alph_wing=3; %AOA of wing (deg)
alph_0wing=0; %AOA of zero lift coefficient
delX_AC=0.0709; % X coordinate of AC (ft)
Y_MAC=0.367*3.28; %Y coordinate of MAC (ft), 0.367 meters
A=2.856; %Aspect Ratio of wing
A_v=8.177; %Aspect ratio of vertical fin
c_lalph=0.1921*(180/pi); %section lift coefficient (1/rad)
a=343*3.28; %speed of sound (ft/s)
u=15.2*3.28; %Cruise speed (ft/s)
M=u/a; %Mach number
B=sqrt(1-M^2);
k=c_lalph/(2*pi);
U_0=49.9; %free stream velocity (ft/s)
q_inf=0.5*0.002378*U_0^2; %dynamic pressure (lb/ft^2)
q_h=0.9*q_inf; %dynamic pressure at tail (horizontal)
S_w=0.9534; %Wing surface area (ft^2)
S_v=0.0107; %Total vertical fin area (ft^2)
S_ail=0.1044; %total aileron surface area (ft^2)
m=0.558; %mass of vehicle (lbs)
I_xx=0.0007519; %slug*ft^2
I_yy=0.00117; %slug*ft^2
I_zz=0.00188; %slug*ft^2

%Effectiveness Calculations
e=0.75; %Oswald's span-efficient factor 'Assumption'
phi_T=0; %pg 257
alph_0=alph_0wing;
cbar_w=0.6501; %MAC (ft)
d_T=0; %pg 257
x_t=1; %DN
T_0=0;% don't know. pg 257

C_Lalph=2.413065;
C_Lroll=-.000001;
C_Lbeta=0;
C_Lalphdot=0;
C_Lq=0; %pg 286
C_LdelE=0.355; %Elevator lift effectiveness /rad
C_L0=0;
C_Lu=0;
C_Lp=0;
C_Lr=0;
C_LdelA=-0.001664; %vehicle's aileron rolling-moment effectiveness
C_LdelR=0; %vehicle's rudder rolling-moment effectiveness
C_L=0.8;
```

```
C_D0=0.16; %Parasite drag coeff of vehicle
C_D=0.2452; %Vehicle drag coefficient
C_Du=0; %used in X_u equation
C_Dalph=2*C_Lalph^2/(pi*A*e); %Vehicle AOA drag effectiveness
C_Dalphdot=0;
C_Dq=0; %pg 286
C_DdelE=0; %vehicle elvator drag effectiveness

C_PXu=0; %pg 269
C_PX0=0;
C_PZu=0; %pg 269
C_PZ0=0;
C_PMu=0; %pg 270
C_PM0=0;
C_PMalph=0;

C_Sbetav=-2*pi*A_v/(2+sqrt((A_v^2)*(B^2)*(1+(tan(sweep_vc2)^2)/B^2)+4));
C_Sbeta=C_Sbetav*q_h*S_v/(q_inf*S_w); %Vehicle sideslip side-force
effectiveness
C_Sp=0; %Steady unaccelerated flight
C_Sr=0;
C_SdelA=0; %Vehicle aileron side-force effectiveness
C_SdelR=0; %Vehicle rudder side-force effectiveness

C_Mu=0;
C_M0=0; %steady unaccelerated flight
C_Malph=0.006048; %vehicle AOA pitching-moment effectiveness
C_Malphdot=-0.0706;
C_Mq=-0.660064; %pg 287
C_MdelE=-0.87; %vehicle elevator pitching-moment effectiveness /rad

C_Nbeta=0.070381; %Vehicle sideslip yawing moment effectiveness
C_N0=0; %steady unaccelerated flight
C_Np=-0.049271;
C_Nr=-0.032935;
C_NdelA=0.000169; %pg 209. vehicle aileron yawing moment effectiveness
C_NdelR=0;

%Force Equations
X_u__X_Pu=(q_inf*S_w/m)*(-1*(C_Du+2*C_D0/U_0)+(C_PXu+2*C_PX0/U_0));
X_alph=-(q_inf*S_w/m)*(-C_Dalph+C_L0);
X_alphdot=-q_inf*S_w*C_Dalphdot/m;
X_q=-q_inf*S_w*C_Dq/m;
X_delE=-q_inf*S_w*C_DdelE/m;
X_T=-cos(phi_T+alph_0)/m;

Y_beta=q_inf*S_w*C_Sbeta/m;
Y_p=q_inf*S_w*C_Sp/m;
Y_r=q_inf*S_w*C_Sr/m;
Y_delA=q_inf*S_w*C_SdelA/m;
Y_delR=q_inf*S_w*C_SdelR/m;

Z_u__Z_Pu=-0.3697;
Z_alph=(-q_inf*S_w/m)*(C_Lalph+C_D0);
```

```
Z_alphdot=-q_inf*S_w*C_Lalphdot/m; %zero
Z_q=-q_inf*S_w*C_Lq/m;
Z_delE=-q_inf*S_w*C_LdelE/m;
Z_T=-sin(phi_T+alph_0)/m;

%Moment Equations
L_beta=-11;%(q_inf*S_w*b_w/I_xx)*C_Lbeta;
L_p=(q_inf*S_w*b_w/I_xx)*C_Lp;
L_r=(q_inf*S_w*b_w/I_xx)*C_Lr;
L_delA=(q_inf*S_w*b_w/I_xx)*C_LdelA;
L_delR=(q_inf*S_w*b_w/I_xx)*C_LdelR;

M_u__M_Pu=(q_inf*S_w*cbar_w/I_yy)*(C_Mu+2*C_M0/U_0)+(C_PMu+2*C_PM0/U_0);
M_alph=(q_inf*S_w*cbar_w/I_yy)*C_Malph;
M_Palph=(q_inf*S_w*cbar_w/I_yy)*C_PMalph;
M_alphdot=(q_inf*S_w*cbar_w/I_yy)*C_Malphdot; %nonzero
M_q=(q_inf*S_w*cbar_w/I_yy)*C_Mq;
M_delE=(q_inf*S_w*cbar_w/I_yy)*C_MdelE;
M_T=(d_T*cos(phi_T)-x_t*sin(phi_T))/I_yy;

N_beta=(q_inf*S_w*b_w/I_zz)*C_Nbeta;
N_p=(q_inf*S_w*b_w/I_zz)*C_Np;
N_r=(q_inf*S_w*b_w/I_zz)*C_Nr;
N_delA=(q_inf*S_w*b_w/I_zz)*C_NdelA;
N_delR=(q_inf*S_w*b_w/I_zz)*C_NdelR;

%Longitudinal Matrices Calculations
A_long=[X_u__X_Pu+(X_alphdot*Z_u__Z_Pu/(U_0-Z_alphdot))
X_alph+(X_alphdot*Z_alph/(U_0-Z_alphdot)) -g X_q+X_alphdot*(U_0+Z_q)/(U_0-
Z_alphdot) 0;
    (Z_u__Z_Pu)/(U_0-Z_alphdot) Z_alph/(U_0-Z_alphdot) 0 (U_0+Z_q)/(U_0-
Z_alphdot) 0;
    0 0 0 1 0;
    M_u__M_Pu+(M_alphdot*(Z_u__Z_Pu)/(U_0-Z_alphdot))
M_alph+M_Palph+M_alphdot*Z_alph/(U_0-Z_alphdot) 0
M_q+M_alphdot*(U_0+Z_q)/(U_0-Z_alphdot) 0;
    0 -U_0 U_0 0 0];
B_long=[X_delE+X_alphdot*Z_delE/(U_0-Z_alphdot) X_T+X_alphdot*Z_T/(U_0-
Z_alphdot);
    Z_delE/(U_0-Z_alphdot) Z_T/(U_0-Z_alphdot);
    0 0;
    M_delE+M_alphdot*Z_delE/(U_0-Z_alphdot) M_T+M_alphdot*Z_T/(U_0-
Z_alphdot);
    0 0];
C=eye(5);
D=zeros(5,2);

%Lateral Matrices Calculations
A_lat=[Y_beta/U_0 g/U_0 Y_p/U_0 (Y_r/U_0)-1 0;
    0 0 1 0 0;
    L_beta 0 L_p L_r 0;
    N_beta 0 N_p N_r 0;
    0 0 0 1 0];
B_lat=[Y_delA/U_0 Y_delR/U_0;
    0 0;
    L_delA L_delR;
```

```
    N_delA N_delR;
    0 0];

%S.S. Systems
sys_lat=ss(A_lat,B_lat,C,D); %LATERAL
sys_long=ss(A_long,B_long,C,D); %LONGITUDINAL

%Dampers
Kq=-2; %longitudinal
A_long_aug=A_long-B_long(:,1)*Kq*C(4,:);
sys_long_aug=ss(A_long_aug,B_long,C,D);

Kr=0.18; %lateral
A_lat_aug=A_lat-B_lat(:,1)*Kr*C(4,:);
sys_lat_aug=ss(A_lat_aug,B_lat,C,D);

%Aileron input
t=[0:.1:10]; %time interval
u_aileron=zeros(2,101);
u_aileron(1,1:10)=-1*(pi/180); %1 sec by 1 degree

%Elevator Doublet
u_elevator=zeros(2,101);
u_elevator(1,1:10)=-1*(pi/180); %1 sec at -1 degree
u_elevator(1,11:20)=1*(pi/180); %1 sec at +1 degree

%lsim plotting
figure(1);
lsim(sys_lat,u_aileron,t) %1 sec aileron input
figure(2);
lsim(sys_long,u_elevator,t) %2 sec elevator doublet
figure(3);
lsim(sys_lat_aug,u_aileron,t) %augmented lat
figure(4);
lsim(sys_long_aug,u_elevator,t) %augmented long

%Eigenvalues
eigs_lat=eig(A_lat)
eigs_long=eig(A_long)
eigs_lat_aug=eig(A_lat_aug)
eigs_long_aug=eig(A_long_aug)

%Autopilot Altitude Hold (longitudinal. pitch-attitude)
zsysauglong=zpk(sys_long_aug);
uelevaug_long=zsysauglong(1,1);
alphaelevaug_long=zsysauglong(2,1);
thetaelevaug_long=zsysauglong(3,1);
qelevaug_long=zsysauglong(4,1);
[z1,p1,k1]=zpkdata(alphaelevaug_long);
[z2,p2,k2]=zpkdata(thetaelevaug_long);
gammatheta=1-zpk(z1,z2,k1/k2);
hgamma=tf(U_0,[1 0]);
sysgam=series(gammatheta,hgamma);
PI=tf([1 3],[1 0]); %PI Compensator
LC=tf([1 0.0001],[1 1]); %Lead Compensator
```

```
Kh=-0.00019; %Gain
sysfinal=Kh*PI*sysgam*LC; %Form Open Loop System for Altitude
figure(5);
margin(sysfinal);
[Afin,Bfin,Cfin,Dfin]=ssdata(sysfinal);
Afincl=Afin-Bfin*1*Cfin;
sysclalt=ss(Afincl,Bfin,Cfin,Dfin);
zsysclalt=zpk(sysclalt);

%Autopilot Heading Hold (lateral. bank-angle)
syscl=sys_lat_aug;
hdg=tf(g/U_0,[1 0]);
syspsi=series(syscl,hdg,[2],[1]);
[Apsi,Bpsi,Cpsi,Dpsi]=ssdata(syspsi);
Kpsi=-0.05;
Bpsi=Bpsi(:,1);
Aclpsi=Apsi-Bpsi*Kpsi*Cpsi;
Bclpsi=Bpsi*Kpsi;
Cclpsi=C;
Cclpsi=[[0;0;0;0;0] C];
Cclpsi(6,:)=Cpsi;
Dclpsi=[0;0;0;0;0;0];
sysclpsi=ss(Aclpsi,Bclpsi,Cclpsi,Dclpsi);
zsysclpsi=zpk(sysclpsi);
hdgcmd=zsysclpsi(6,1);

%Step Change of 100ft in altitude and 3deg in heading
figure(7);
step(100*sysclalt,100) %altitude change response
figure(8);
step(0.05236*sysclpsi,100) %heading response (3deg in radians)
figure(9);
step(0.05236*hdgcmd,100)
```