Apollo Type Flexible Launch Vehicle

The flight vehicle in this example is similar to an Apollo/Saturn type cylindrical booster such as the one shown in the picture. It consists of two stages and contains a spacecraft to be inserted into earth orbit. The analysis that follows is during first stage and at maximum dynamic pressure. During first stage the vehicle is powered by a single 3 million pounds thrust solid rocket motor (SRM) that gimbals in pitch and yaw and provides acceleration in the x axis. Because of its cylindrical symmetry the pitch and yaw axes are identical and we shall, therefore, analyze stability only in the pitch and roll axes. The roll axis is controlled by 8 RCS thrusters (4 pairs) located around the peripheral of the cylinder at 90 degrees apart. Two left/ right pairs are firing in the ±z direction and two top/ bottom pairs are firing in the ±y direction. Since the vehicle is long and skinny it is considerably flexible and the main purpose of this study is to investigate the effects of structural flexibility on vehicle stability. The objective of our analysis is to define a satisfactory stiffness of the conical inter-stage structure that joins the first and second stages together. We will, therefore, analyze and compare the stability of the control system using three separate finite element model structures defined as: nominal stiffness, soft, and stiff structural models. We will also analyze at the vehicle responses to attitude step commands and to wind-gust disturbances, measured from several gyros and accelerometer sensors located in different parts of the vehicle. For pitch axis stabilization the flight control system uses attitude and rate feedback from a gyro. It also uses angle of attack feedback from a vane sensor that is located in the front part of the vehicle. The alpha-feedback is also needed for load-relief during high Q-bar conditions. We conclude that the load-relief system can reduce the aerodynamic loading on the structure by as much as 18%. Our study in roll is to define the dead-band in the RCS jet control logic. We will create, therefore, a roll axis flexible vehicle model, design the RCS logic, and define a dead-band that is small enough for tight roll attitude control while avoiding too much thruster chattering.

Analysis

The analysis is separated in two parts: (a) the pitch axis analysis that uses the TVC, gyros and alpha feedback for flight control, and load-relief, and (b) the roll axis analysis that uses the RCS jets for roll control. The Flixan data files for the pitch analysis are located in the directory "*Flixan\Examples \Apollo \MaxQ Pitch*", and the data files for the roll analysis are in the directory "*Flixan\Examples \Apollo\RCS Jets*". The Matlab/ Simulink analysis is performed in the "Mat" subdirectories.

Pitch Axis Analysis

For pitch analysis we will create three flexible vehicle state-space models using different inter-stage stiffnesses. We will design a flight control system (gains and filters) that stabilizes the pitch axis. We will combine the vehicle, control system and TVC actuator systems together to create closed-loop simulation models for Matlab/ Simulink. We will also create state-space models for open-loop frequency response analysis that will be used to measure the system's stability margins. The closed-loop simulation models will also be used to determine the system's response to noisy gust disturbances.

Input Data Files

The launch vehicle input data is in file "*Apollo-MaxQ.Inp*". There are two vehicle data sets in this file: a data set for creating a rigid-body model, "*Apollo-20, Stage-1, Max-Q, Rigid Body*", and another set for creating flexible vehicle state-space models "*Apollo-20, Stage-1, Max-Q, 20 Flex modes*". The purpose of this analysis is to compare vehicle stability using three different inter-stage structure designs: a nominal inter-stage stiffness design, a soft design, and a hard inter-stage design. We have created, therefore, three sets of modal data files from Nastran finite elements models. The three sets of modal data are identical in every way except for the inter-stage stiffness, and each file contains 71 flexible modes (rigid-body modes are not included). There is a nominal stiffness modal data file "*Apollo-Nom.Mod*", a soft modal data file "*Apollo-Soft.Mod*", and a very stiff inter-stage structure file "*Apollo-Hard.Mod*". The pivoting TVC nozzle is not included in the finite elements model. The engine resonance at the pivot caused by the actuator piston, load, and backup structures is included in the actuator model. There is also a locations or "nodes" identification file "*Apollo_Stg1.Nod*" that is used in the mode selection process.

We will use the three modal data files to create separate vehicle systems identical in every way except for the type of flex modes and save them in three separate system files. The flight vehicle modeling program generates fully coupled (roll, pitch, and yaw) state-space vehicle systems. We will run it three times to generate the three vehicle systems of variable inter-stage stiffness. The vehicle input data file is always the same "*Apollo-MaxQ.Inp*". The output system files, however, are different in each case. For the nominal stiffness we save the vehicle systems in file "*Apollo-MQ-Nom.Qdr*", for the soft vehicle we use systems file "*Apollo-MQ-Soft.Qdr*", and for the stiff vehicle we save the systems in file "*Apollo-MQ-Hard.Qdr*". A set of 20 preselected modes is used for each inter-stage stiffness. The three sets of selected modes are also included in input file "*Apollo-MaxQ.Inp*". The input file in addition to the vehicle and selected modal data-sets it also contains model truncation data to extract the pitch subsystem.

Although it is feasible in the analysis to use the fully coupled vehicle system that includes all 3 axes, it is however inconvenient to use systems that are larger than necessary. We will use, therefore, the system truncation (system extraction) Flixan utility program to extract only the system states, inputs and outputs that relate only to the vehicle motion in the pitch direction, and to truncate or eliminate the remaining variables. The system truncation instructions are identified by Flixan from the label "CREATE A NEW SYSTEM FROM AN OLD", followed by the title of the new (reduced size) system in the next line "*Apollo-20, Stage-1, Max-Q, 20 Flex modes, Pitch Axis*". The next line contains the title of the original fully coupled system "*Apollo-20, Stage-1, Max-Q, 20 Flex modes*". The remaining lines in the data set are system truncation instructions telling the program which inputs, states, and outputs are to be retained from the original system. The truncated system is saved in the systems file below the original system. There are two sets of system truncation instructions in file "*Apollo-MaxQ.Inp*". One set is for extracting

a pitch axis rigid-body system from the coupled rigid-body system and another set is for extracting a pitch axis flexible system from the coupled flexible body system.
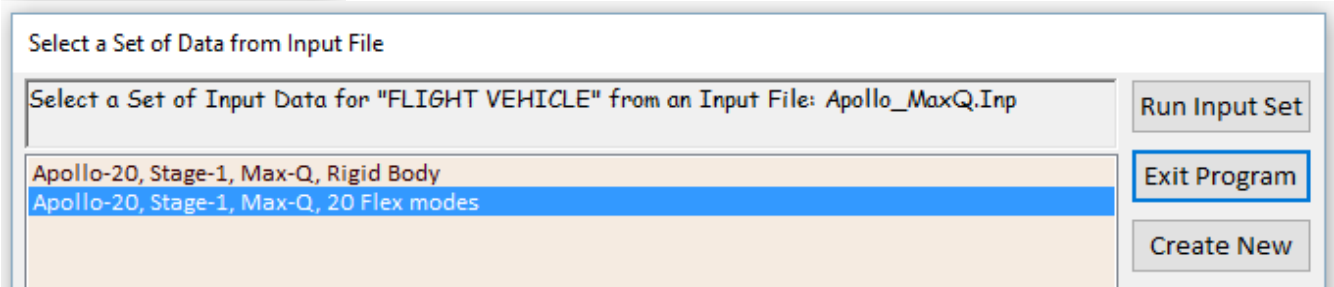
Pitch Mode Selection

Our first task before beginning the analysis is to select the flex modes from the Nastran files. Let us go back to the input data file "*Apollo-MaxQ.Inp*". This file contains three sets of 20 preselected pitch axis modes. So far we have assumed that the modal sets are already selected from three Nastran files, scaled and ready to be processed by the flight vehicle modeling program. We will now describe the process of selecting and scaling a set of pitch dominant modes from one of the Nastran ".Mod" files. This process is performed by the mode selection program as follows. Start the Flixan program, select the project folder "*Flixan\Examples\Apollo\MaxQ Pitch*", and from the Flixan menu select "*Program Functions*", "*Flight Vehicle/ Spacecraft Modeling Tools*", and then "*Flex Mode Selection*", as shown.
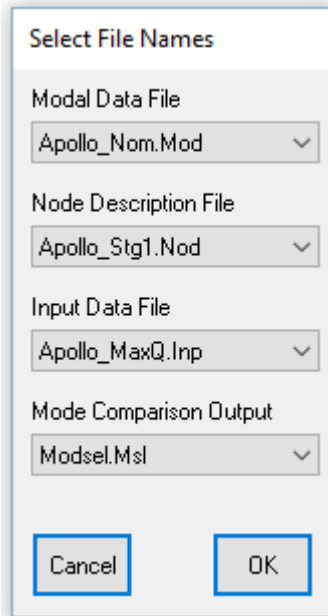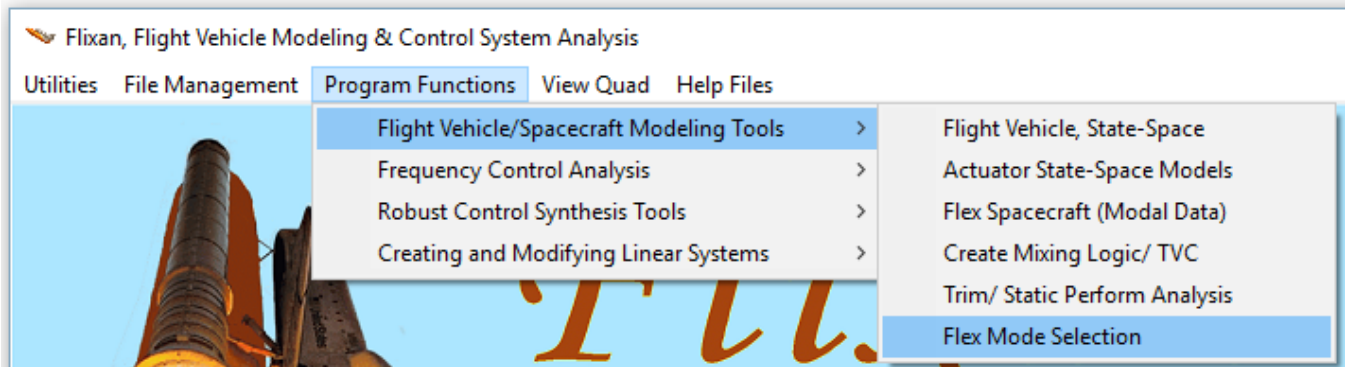
In this menu you must select the following files:

(a) The Nastran generated modal data file "*Apollo_Nom.Mod*" that contains the mode frequencies, the mode shapes and slopes for 71 modes at specific locations (nodes) which are important for our flight control analysis.
(b) The nodes file "*Apollo_Stg1.Nod*" that identifies the vehicle locations which are included in the modal data file.
(c) The input data file "*Apollo_MaxQ.Inp*" that contains the vehicle parameters.
(d) Select also an output file to save the mode comparison results.

In the following menu you must select the flight vehicle model and click on "*Run Input Set*". The program uses this data-set to identify the vehicle excitation and sensor points.

The user must also enter a short label to be included in the title of the selected modes set, to identify, for example, the type of modes.

At this point a bar plot appears and used for selecting the dominant modes. It is showing the modal strengths by means of vertical red bars. The height of each bar is logarithmically proportional to the relative mode strength. The strong modes are tall and the weak modes are short. The modal strength is measured relative to the weakest mode. The user manually selects some of the strongest modes by pointing the mouse cursor at a bar and clicking the mouse. The modes change color from red to green when they are selected. There are no rigid-body modes in this case. We select 20 flex modes and press the "enter" button to complete the mode selection. Some additional information is also entered which is included as comments.

Mode Strength Comparison (use mouse to select the strongest modes)
Select Dominant Modes of: APOLLO 20 MODAL DATA, NOMINAL INTERSTAGE ST
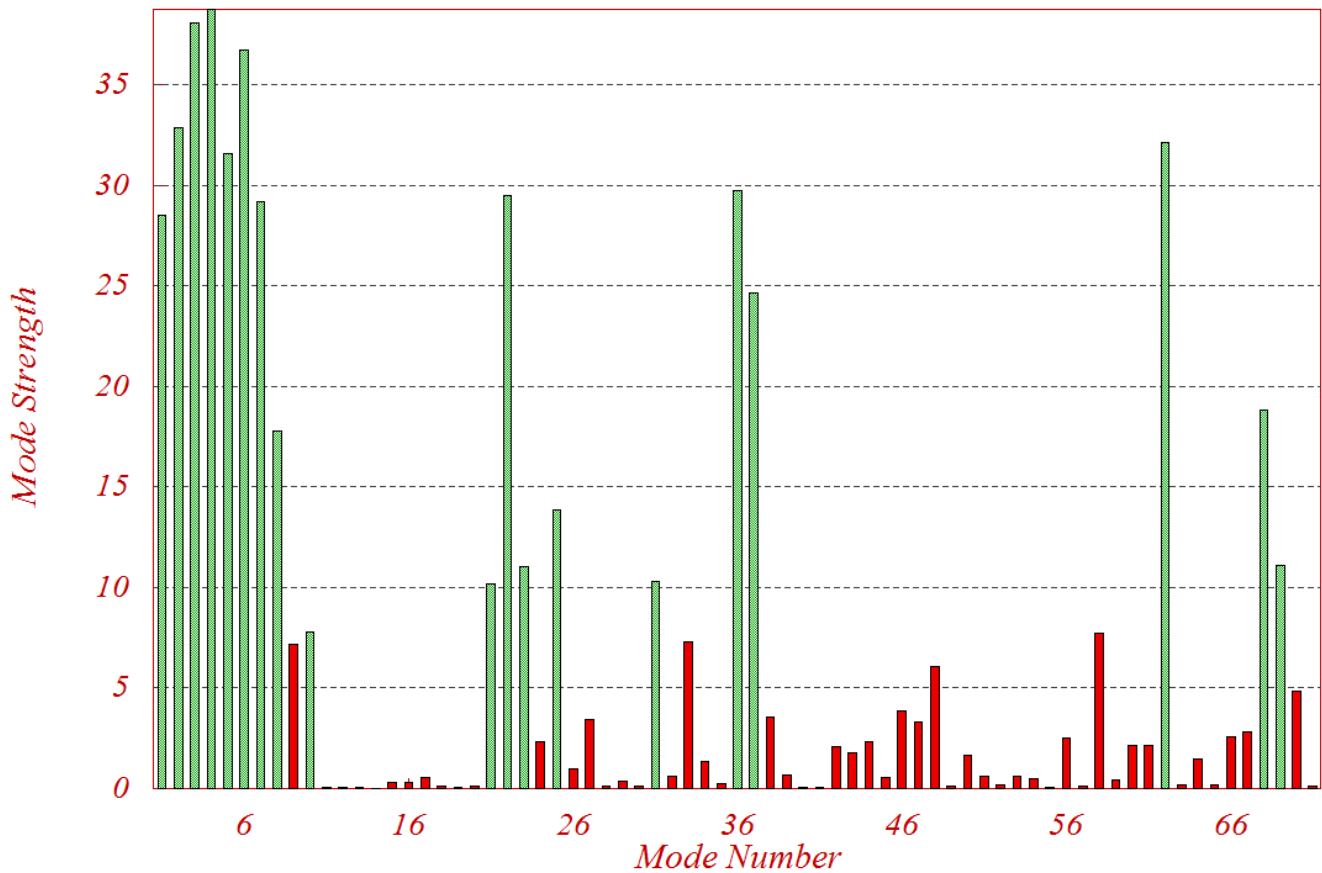
In Figure 2, the gimbal acceleration output from the actuator block (yellow) connects to the gimbal acceleration input in the vehicle model (green). The acceleration is used by the vehicle model to calculate the TWD forces which are combined with the TVC forces to generate the total forces at the gimbal.

Flight Control System with Load-Relief

The flight control system consists mainly of feedback from three vehicle outputs: the attitude θ, vehicle rate q, and the measured angle of attack α_s at the alpha sensor. It includes some additional filters for stability: a low pass filter, and a notch filter for attenuating the second flexible mode. The three control gains were calculated using classical control methods.
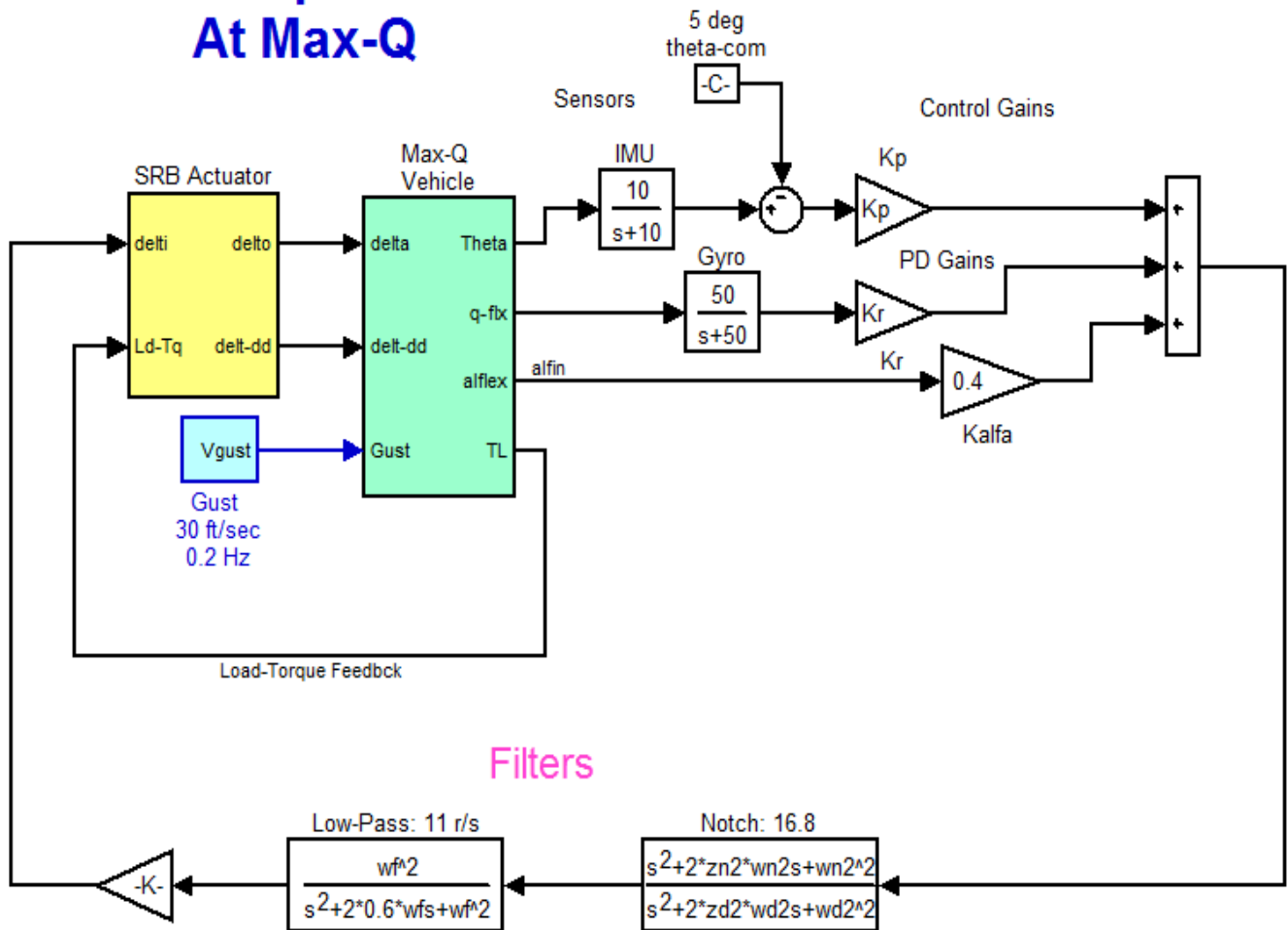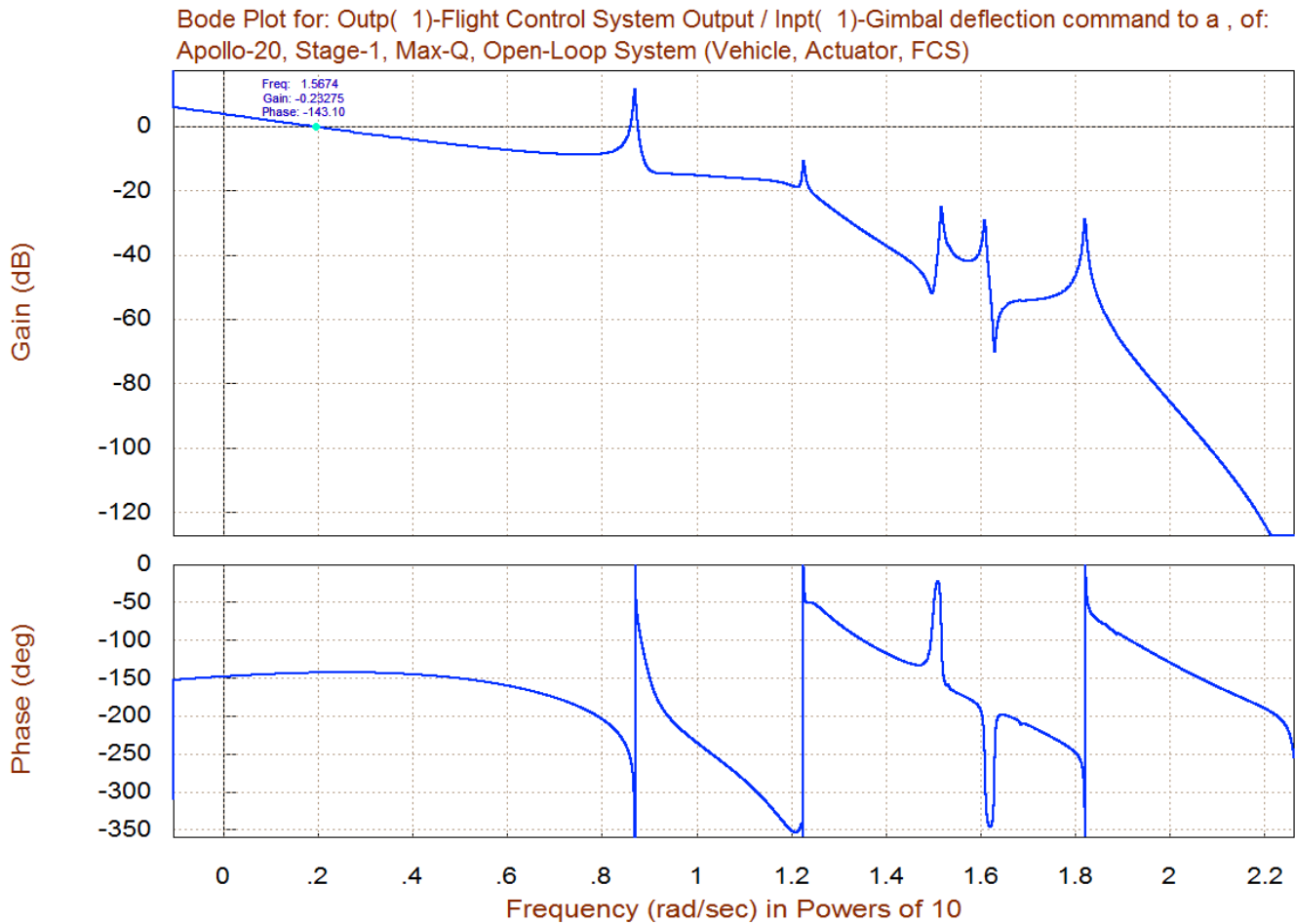

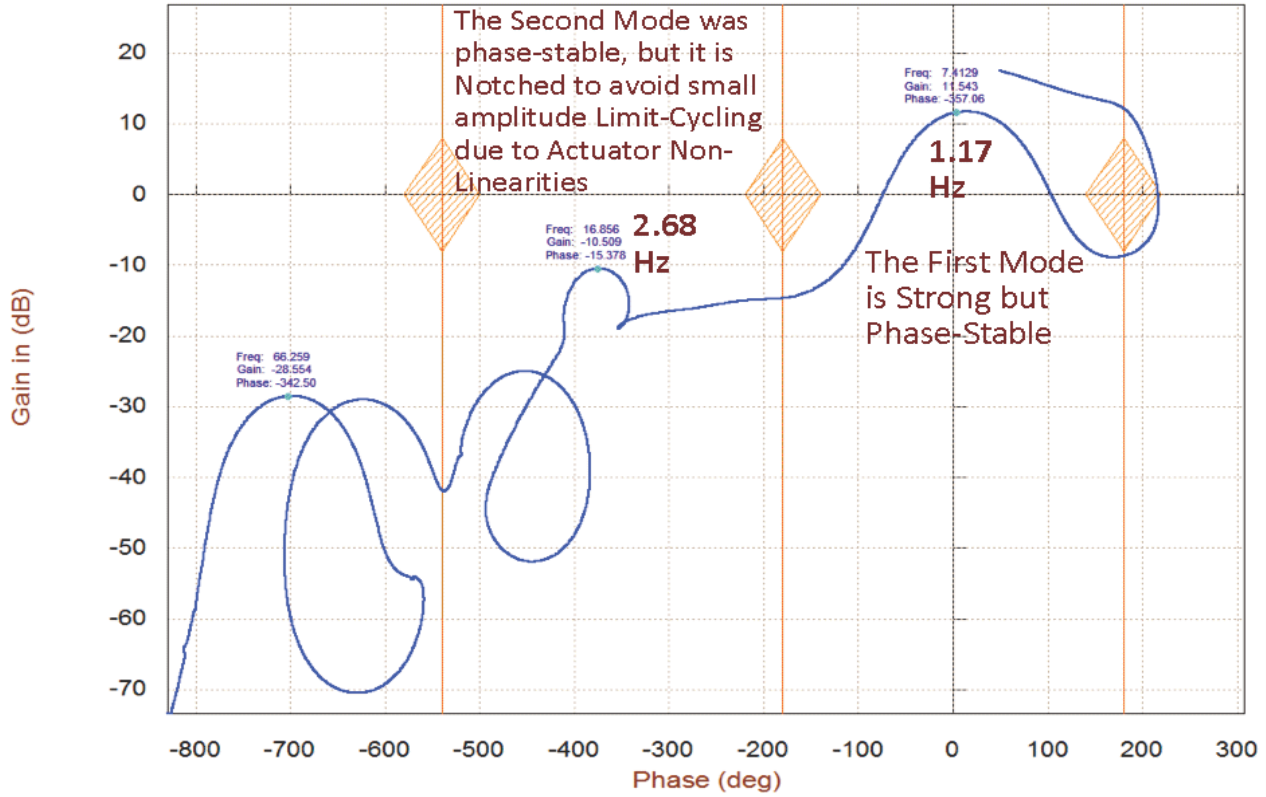**Figure 2 Closed-Loop Simulation Model**

Stability Analysis

The system stability is measured by cutting the control loop at the actuator input and calculating the frequency response of the open-loop system between the δ_{com} input to the actuator and the filter outputs. Then we plot the frequency response on a Bode or Nichols chart to measure the phase and gain margins. This process is implemented in a Matlab script file "run.m" which is located in directory "\Apollo \MaxQ Pitch \Mat". The script loads the nominal stiffness pitch flight vehicle "Vehi_20flx.m" and the actuator "actuator.m" systems into the Matlab workspace, calculated the flight control gains, generates the linear open-loop Simulink model "OpenFlx.Mdl", and generates the Bode and Nichols plots as shown in Figure 3 (a & b) below. The Simulink model file "*OpenFlx.Mdl*" is similar to Figure 2 but it has the control loop opened at the actuator input. It includes the vehicle and actuator state-space quadruple matrices inside the green and yellow blocks. The open-loop configuration is also implemented in the Flixan input file "*Apollo_MaxQ.Inp*" and its title is "*Apollo-20, Stage-1, Max-Q, Open-Loop System (Vehicle, Actuator, FCS)*" .

The Bode plot shows a cross-over frequency point (bandwidth) of 0.24 Hz. The Nichol's plot shows that the first flex mode at 1.17 Hz is phase stabilized. The phase margin is 39 (degrees) and the gain margin is 9 (dB). The 2.68 Hz second mode is attenuated by the notch filter, although it was well behaved in phase and it would otherwise be phase-stable without filtering. The reason for gain stabilizing it is because at low amplitudes the actuator behavior becomes sluggish and as the frequency increases the phase delay increases unpredictably to the point where the resonance would drift towards the left, cross the (+) point, and cause limit cycling. So it is better to gain stabilize it with a notch.

Bode Plot for: Outp(1)-Flight Control System Output / Inpt(1)-Gimbal deflection command to a , of:
Apollo-20, Stage-1, Max-Q, Open-Loop System (Vehicle, Actuator, FCS)

Freq: 1.5674
Gain: -0.23275
Phase: -143.10

Nichols Plot for: Outp(1)-Flight Control System Output / Inpt(1)-Gimbal deflection command to a , of:
Apollo-20, Stage-1, Max-Q, Open-Loop System (Vehicle, Actuator, FCS)

The Second Mode was phase-stable, but it is Notched to avoid small amplitude Limit-Cycling due to Actuator Non-Linearities

Freq: 7.4129
Gain: 11.543
Phase: -357.06

1.17 Hz

Freq: 16.856
Gain: -10.509
Phase: -15.378

2.68 Hz

The First Mode is Strong but Phase-Stable

Freq: 66.259
Gain: -28.554
Phase: -342.50

Gain in (dB)

Phase (deg)

Nyquist Plot for: Outp(1)-Flight Control System Output / Inpt(1)-Gimbal deflection command to a , of:
Apollo-20, Stage-1, Max-Q, Open-Loop System (Vehicle, Actuator, FCS)

Real: 2.5011
Imag: 2.3456
Freq: 7.3890
Gain: 3.4289
Phase: -316.84

1.17 Hz

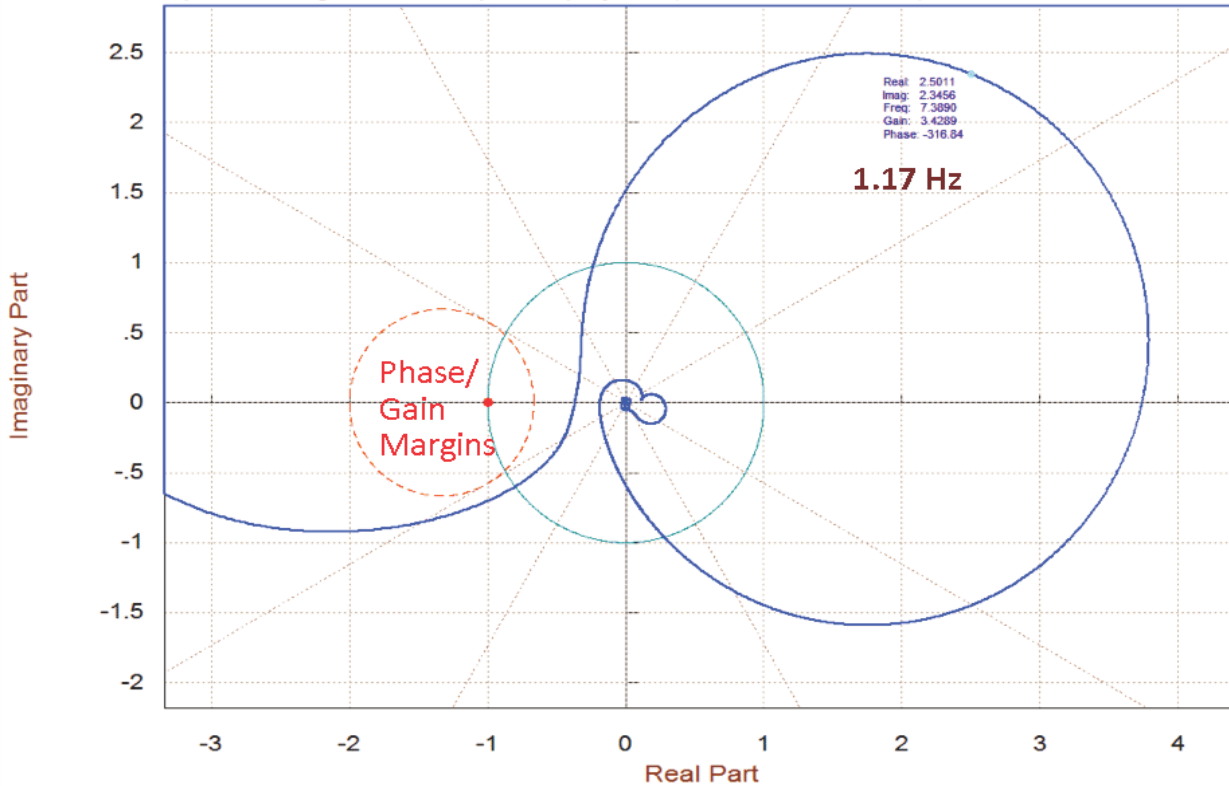Phase/ Gain Margins

Imaginary Part

Real Part

Figure 4 Nichols and Nyquist Plots of the Nominal Stiffness Open-Loop System obtained using the Flixan frequency response program

The frequency domain stability analysis is performed using both Flixan and Matlab programs. Figure 5 is similar to Figure 3 and compares stability between the three structures of different inter-stage stiffnesses. In Matlab the plots are generated using the script file "run3.m" in directory "\Apollo\MaxQ Pitch \Mat". The script loads the three pitch vehicle and the actuator state-space systems into the Matlab workspace, calculates the Nichol's and Bode plots, and plots all three together in different colors, as shown above. The three pitch vehicle Matlab function files: "Vehi_20flx", "Vehi_20rif", and "Vehi_20sif", correspond to the nominal, hard, and soft inter-stage stiffnesses. They were created from the system title "*Apollo-20, Stage-1, Max-Q, 20 Flex modes, Pitch Axis*", from the three system files "Apollo-MQ-Nom.Qdr", "Apollo-MQ-Hard.Qdr", and "Apollo-MQ-Soft.Qdr", and they were exported into Matlab format using the Flixan "Export to Matlab" utility.

The green curve represents the nominal structure frequency response (same as in Figure 3). The blue corresponds to the stiff interface structure, and the red corresponds to the soft interface structure. The mode frequencies are slightly different between structures due to the variation in stiffness. In all 3 cases the first mode is phase stable and the second mode is attenuated with the notch filter. The rigid body stability margins are acceptable in all 3 cases. In the soft case, however, the phase margin of the first mode is reduced to 32 (degrees), which implies, that further softening will drive it towards instability. The second mode is gain stable in all three cases because of notch filtering. The attenuation of the second mode is greater in the nominal stiffness case (green) because the notch is tuned at mode-2 resonance (2.68 Hz). Figures 6 and 7 are similar, including the Nyquist Diagram. They were created using the Flixan frequency response program and they compare stability of stiff versus soft structures.
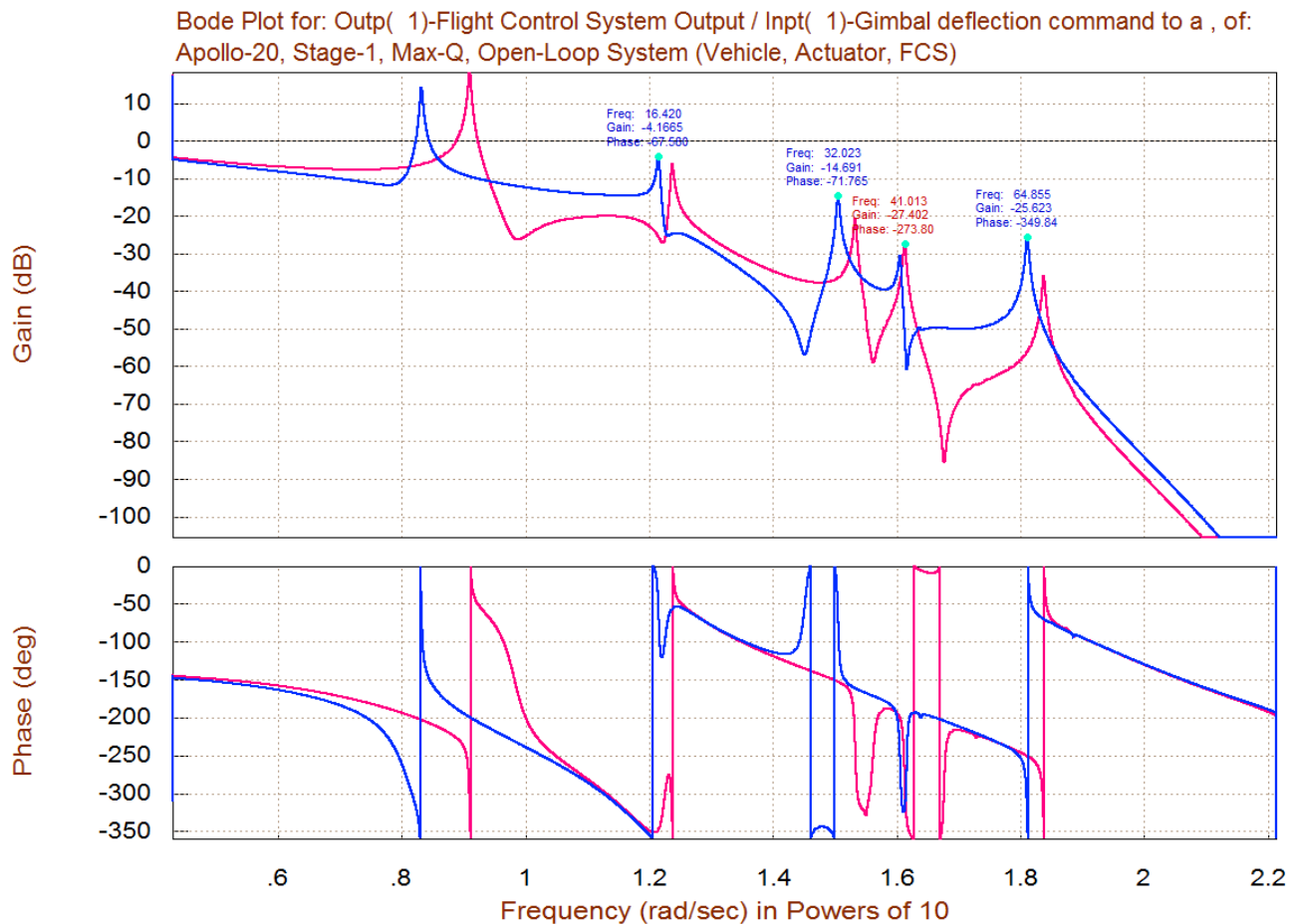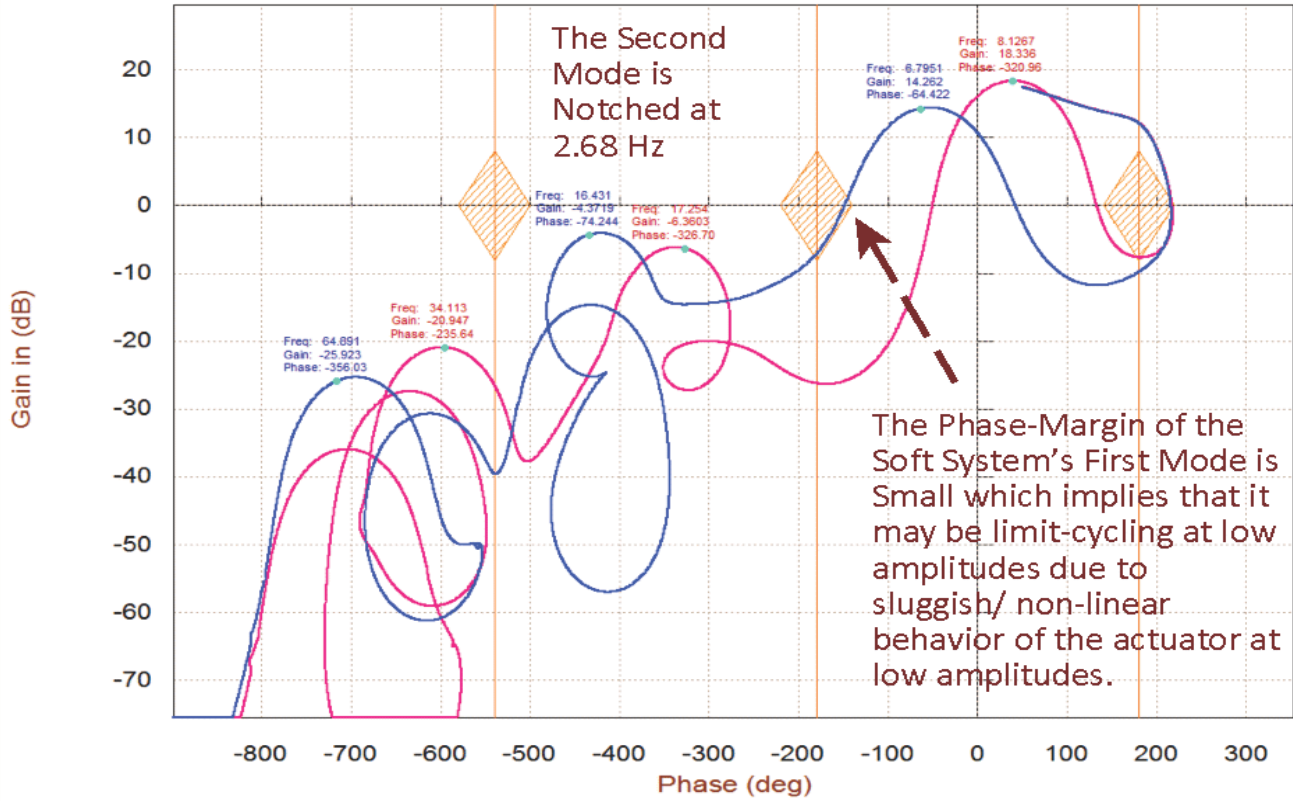

Figure 6 Frequency Response Comparison between the Stiff (red) and the Soft (blue) systems calculated using the Flixan Frequency Response Utility Program
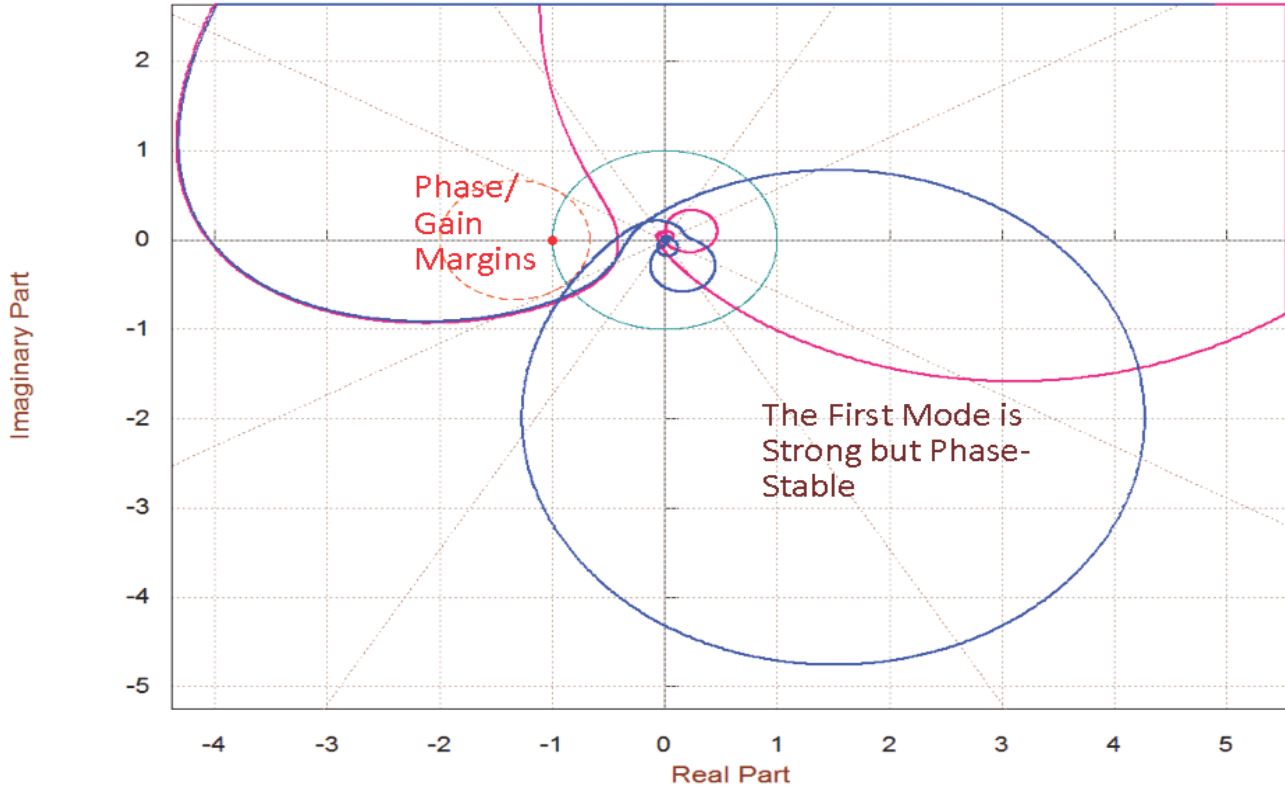
Figure 7 Nichols and Nyquist Plots Comparison between the Stiff (red) and the Soft (blue) systems calculated using the Flixan Frequency Response Utility Program

Simulation Results

The Simulink model "Sim.Mdl", shown in Figure 2 located in directory "\Apollo\MaxQ Pitch\Mat" is used to generate the simulations. The model has two inputs: an attitude command (θ-com), and a wind gust disturbance input. The attitude command comes from guidance, which is either closed-loop guidance or a table look-up attitude versus time command. The gust disturbance is a noisy waveform representing the wind velocity relative to the vehicle, in (ft/sec). The direction of the wind relative to the vehicle is defined in the vehicle input data. The magnitude of the wind is limited to 30 (ft/sec) peak-to-peak, and its bandwidth is limited to 0.2 Hz. The input selection, either attitude command or disturbance input can be turned "on" or "off" from the Simulink model. When the simulation is complete, a Matlab script file "pl.m" is used to plot the simulation results which are as shown in the figures below.
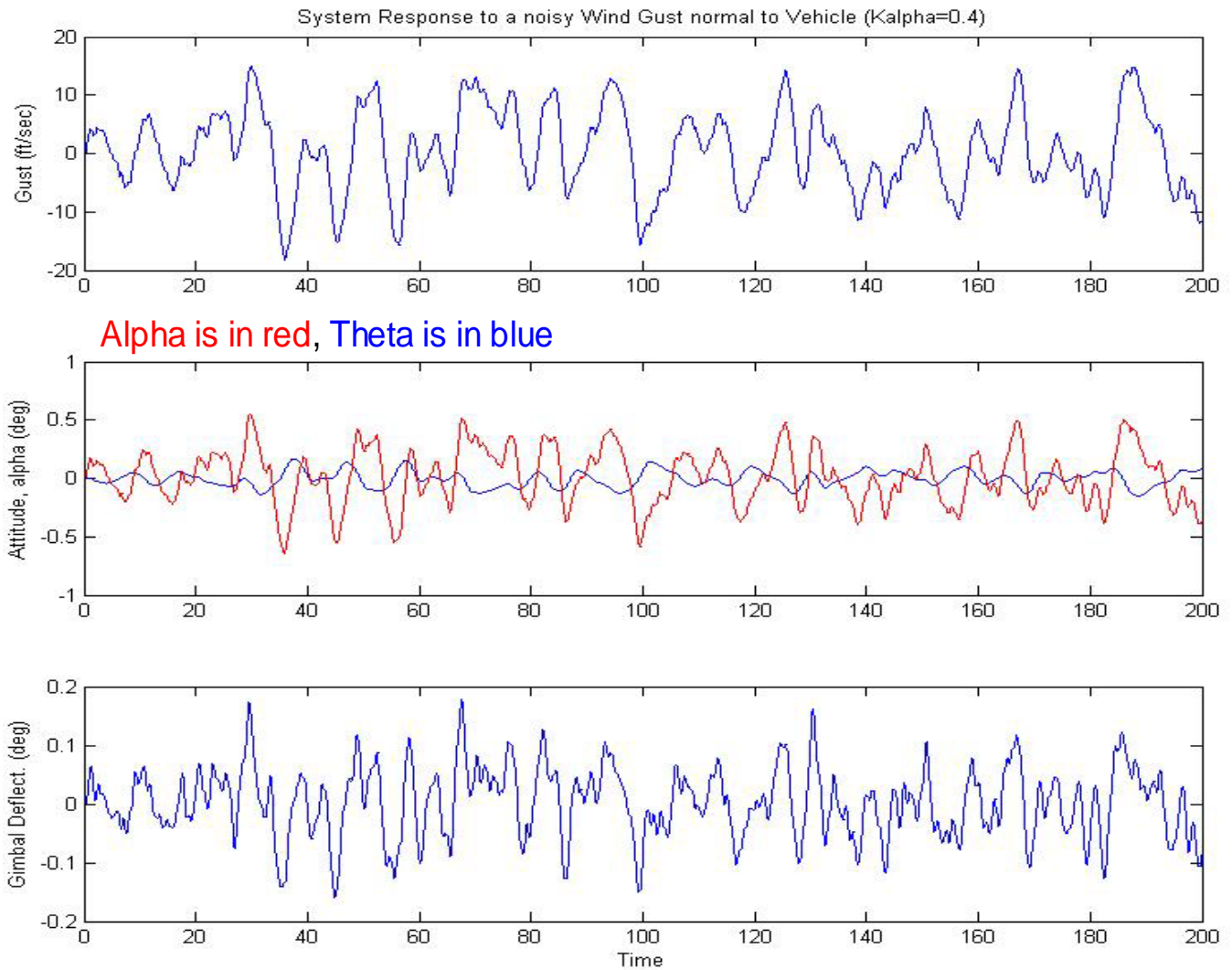

Figure 8 Vehicle Attitude, Alpha, and Gimbal Responses to a Noisy Wind Gust (shown above)

Figure 10 above compares the angle of attack and the gimbal responses to the wind gust disturbance between two cases. Case-a, (shown in red) is calculated using the nominal flight control gains. Case-b, (blue) is calculated with the load-relief (alpha) gain Ka reduced to ¼ of its nominal value, which represents a 75% reduction in load-relief. The alpha comparison shows that the reduction in load-relief gain increases the structural loading by approximately 18%, due to an increase in alpha, which demonstrates that the load-relief alleviates some of the normal loading, as it is supposed to do. The load alleviation is restricted, however, due to limitation in the flight control system bandwidth.

Tracking the Guidance Command

Figures 11 and 12 show the vehicle responses to 5 degrees pitch attitude command. The pitch attitude (blue) reaches 4 degrees relatively fast and then it converges slowly towards the commanded attitude. Remember that this flight condition is at maximum dynamic pressure and the load-relief is preventing it from reaching the 5 degrees very fast. During Max-Q the flight control system is also attempting to keep α small and it is trading some of the command following performance for reducing α and the normal loads. The vehicle is, therefore, not expected to have a great command following performance at Max-Q. The angle of attack is shown in red. It peaks up to 4 degrees before it starts decreasing towards zero as a result of the alpha-feedback. The plot also shows the gimbal deflection which reaches its peak value at 3 degrees. This is acceptable, because we normally are not commanding big attitude steps.
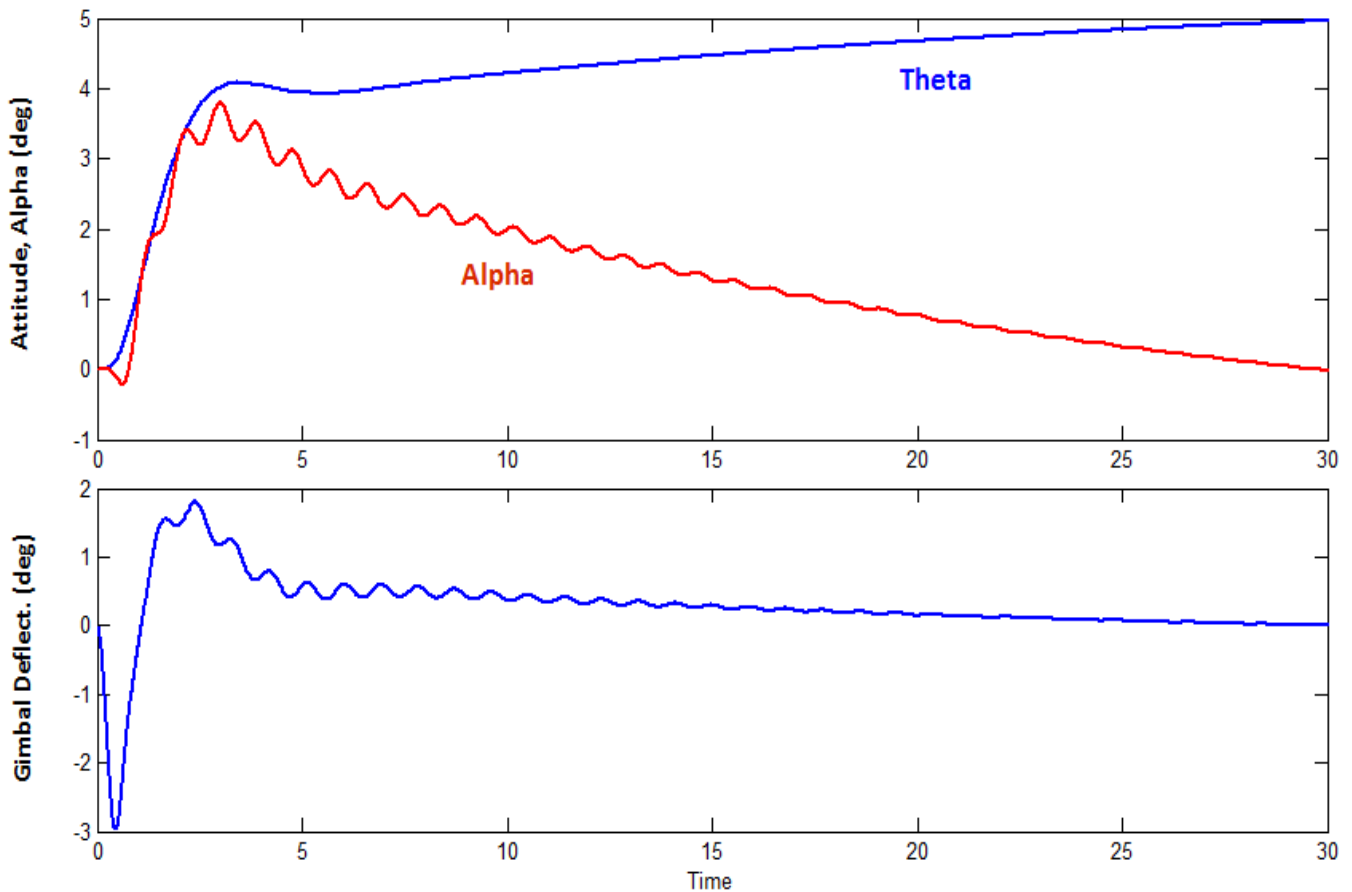

Figure 11 Vehicle Response to 5 degrees attitude change command in presence of wind gusts

Figure 12 shows the three rate gyro and three accelerometer responses to the attitude change command. The green responses are measured from the top of the vehicle where the payload module is located. The red responses are measured at the inter-stage location. The blue responses are obtained from the flight control sensors which are located in the 1st stage booster. Notice how flexibility is minimal at the FCS rate-gyro (blue).
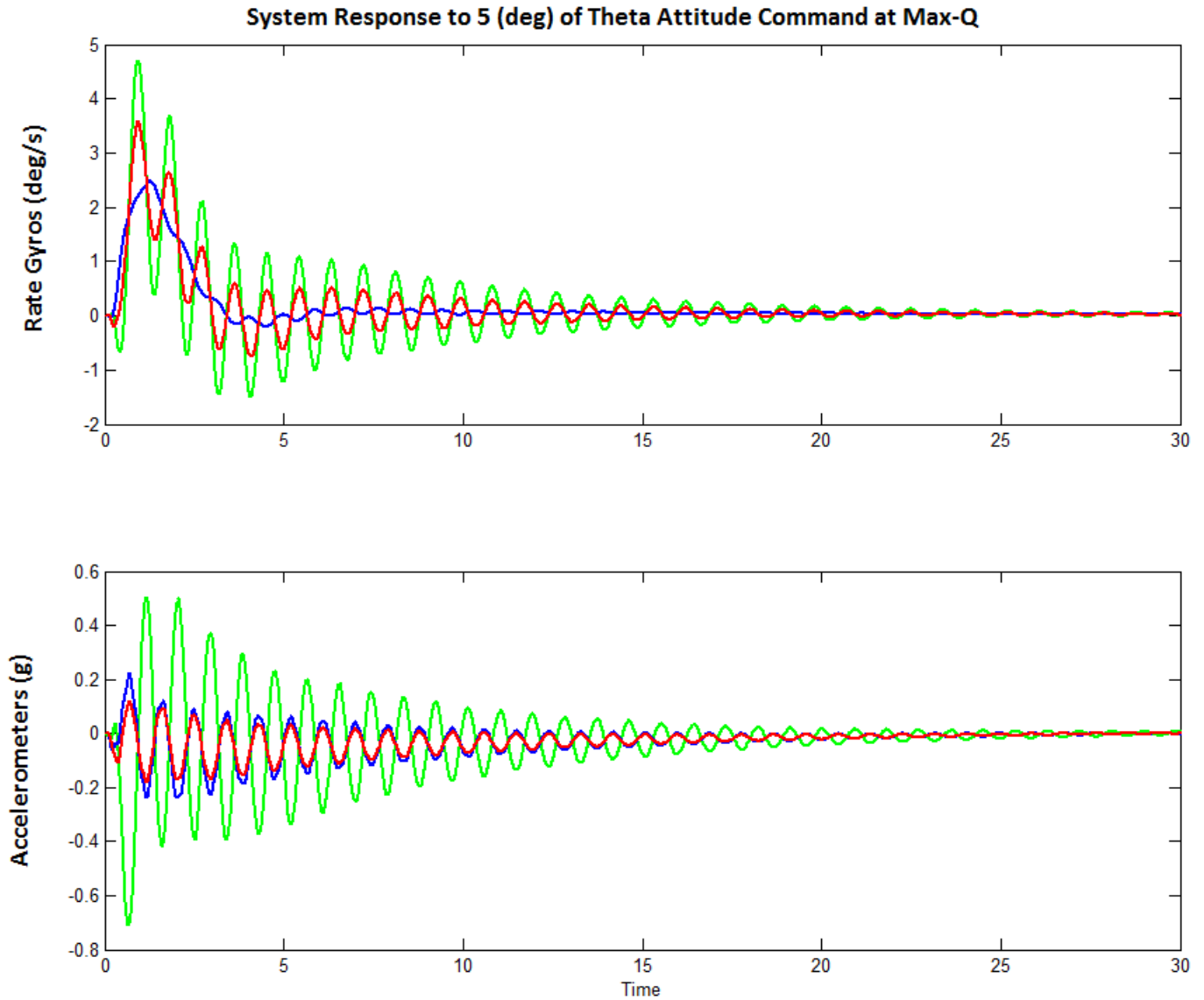

Figure 12 Rate Gyro and Accelerometer Responses to the 5 degrees attitude command (with gusts)

It is, therefore, necessary to use some additional jets for roll control in order to prevent the vehicle from drifting in roll during flight. In this example we are using a cluster of 8 RCS jets. The jets are on/ off devices of 600 (lb) thrust. They are placed in pairs in four locations around the peripheral of the cylindrical booster, near the bottom of the second stage, see figure on the right. Each pair is firing in opposite (±y and ±z) directions to provide positive and negative thrust at four locations.

We will generate a flexible vehicle model to be used for roll axis analysis. We will also develop a simple RCS bang-bang control logic, couple the vehicle model with the control logic and simulate it in the Matlab/ Simulink environment. This will help us define the control system dead-zone.
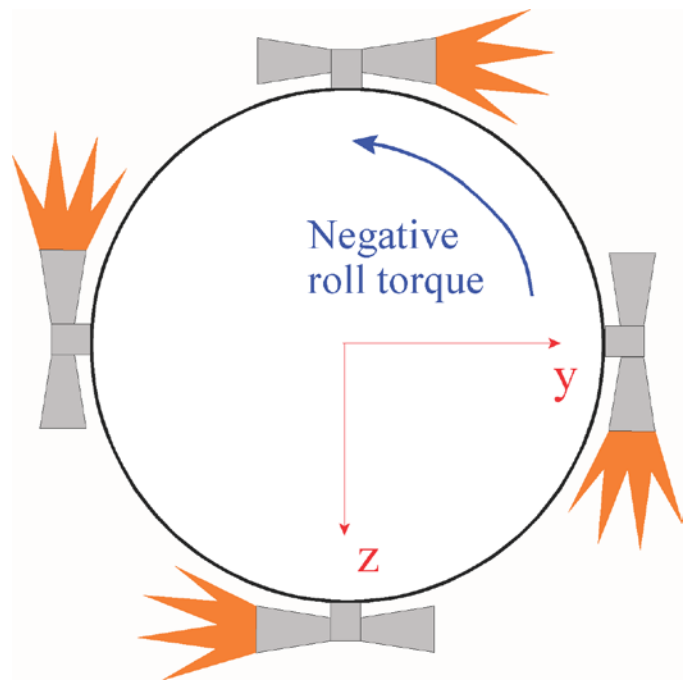

Data Files

The data files for generating the roll axis vehicle models are located in directory "*Flixan\ Examples\ Apollo\RCS Jets*". The input data file is "*Apollo_MQ_RCS.Inp*" and it contains two flight vehicle modeling data sets: a rigid-body and a flexible vehicle set. The flex vehicle title is "*Apollo-20, Stage-1, Max-Q, Flex Body, RCS Control*". This vehicle has a gimbaling engine of fixed thrust that does not couple into roll and four RCS thrusters. The four reaction control thrusters are ±600 (lb) each used for roll control. Actually, each thruster represents a pair of jets mounted back to back, as shown above, that can fire in opposite directions. The thrust directions (D_y and D_z) of the four RCS thrusters in the input data are oriented to provide thrusts in the ±z and the ±y directions for roll control. They are defined as "Throttlable", that is, capable to provide thrust variation between -600 to +600 (lb) in the state-space system by just varying the throttle command input of the corresponding thruster between -1 and +1. This means that the thrust is included in the model and you must not drive the vehicle system with the actual thrust at its inputs, but the throttle command input must be varied between -1 to +1. Exceeding this range would exceed the thrust capability of the hardware. In the closed-loop simulations the throttle command comes from the reaction control system (RCS) software, and since we are dealing with "on/ off" devices in this example, the commands to the vehicle inputs that correspond to the RCS pairs should be either, {-1, 0, or +1} and nothing in between. Values outside this range are also unacceptable and they will produce meaningless results. In the vehicle input data we have included two roll rate gyro sensors. One sensor is located near the top of the stack and the other sensor is located near the top of the first stage booster.

RCS Control/ Simulation Results

The simulation is performed using the Simulink model "*RCSim_flx.Mdl*" shown in Figure 14 below. The Matlab files are in folder "Examples\Apollo\RCS Jets\Mat". The jet control logic is implemented in the Simulink model as a Matlab function file "*Jet_Con.m*". It employs a very simple phase-plane control law that combines the roll rate and attitude errors together when the combined signal exceeds 0.1 it fires all four jets in the proper direction. If the combined error is between 0.05 and 0.1 it fires only two jets. Otherwise, if the magnitude of the error signal is less than 0.05, it does not fire any jets. The script file "pl.m" is used for plotting the simulation data in Matlab.
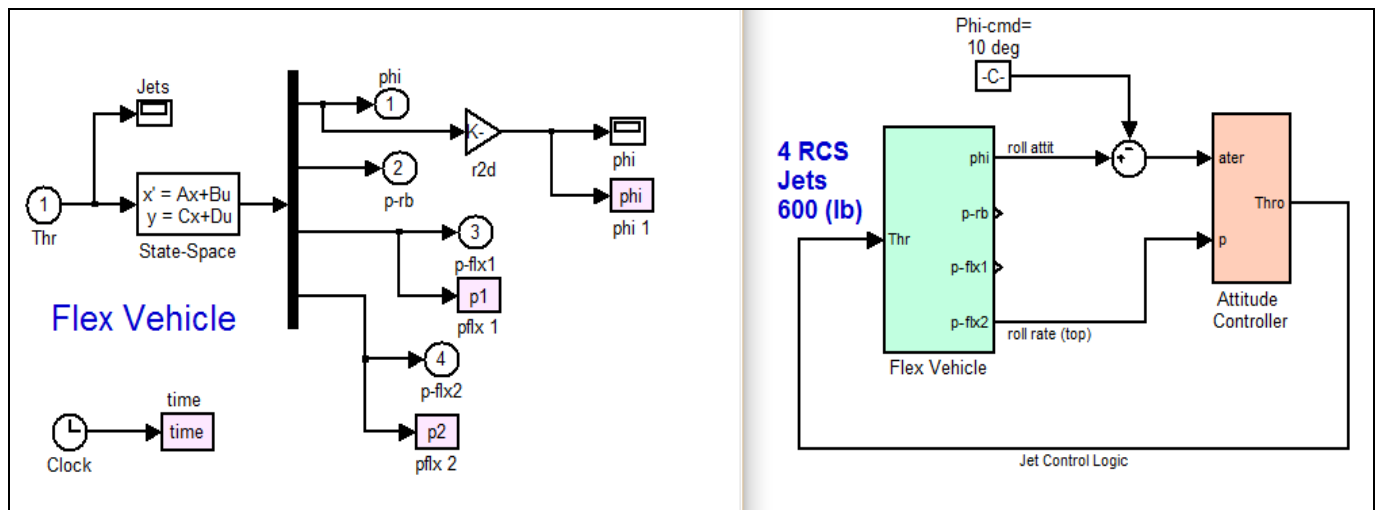

```
function Thr= Jet_Con(ater,p)
% Thr= Jet_Con(ater,p)
db=0.2/57.295;            % dead-band (rad)
er= (2.0*ater+3.0*p);    % error signal

if abs(er)>db & abs(er)<2*db;
                         Thr=-sign(er)*[1,0,1,0];
elseif abs(er)>=2*db;  Thr=-sign(er)*[1,1,1,1];
else;                    Thr=[0,0,0,0];
end
```

Figure 14 Roll Axis Simulink Model file "RCSim_flx.mdl"

The results in Figure 15 consist of three plots that show the system response to 10 degrees of roll command. The first plot shows the roll attitude which approaches the 10 (deg) commanded value. The second plot shows the rate response at the two rate gyros. The rate at the space module (red) is slightly higher than the other gyro which is located near the top of the stage-1 booster (blue). The third plot shows the RCS firing history. Initially, all four jets fire in the positive direction (green) to get the vehicle rolling. Then the 4 thruster directions reverse, firing in the negative torque direction to slow it down. Finally, only two jets chatter (red) until the error becomes sufficiently small.

Closed-Loop Roll Response to a 10 (deg) Roll command, using a 0.05 (deg) dead-band

Red is the roll rate gyro response at the space module (front)
Blue is the roll rate gyro response at the Stage-1 booster (top)

all 4 jets +ve

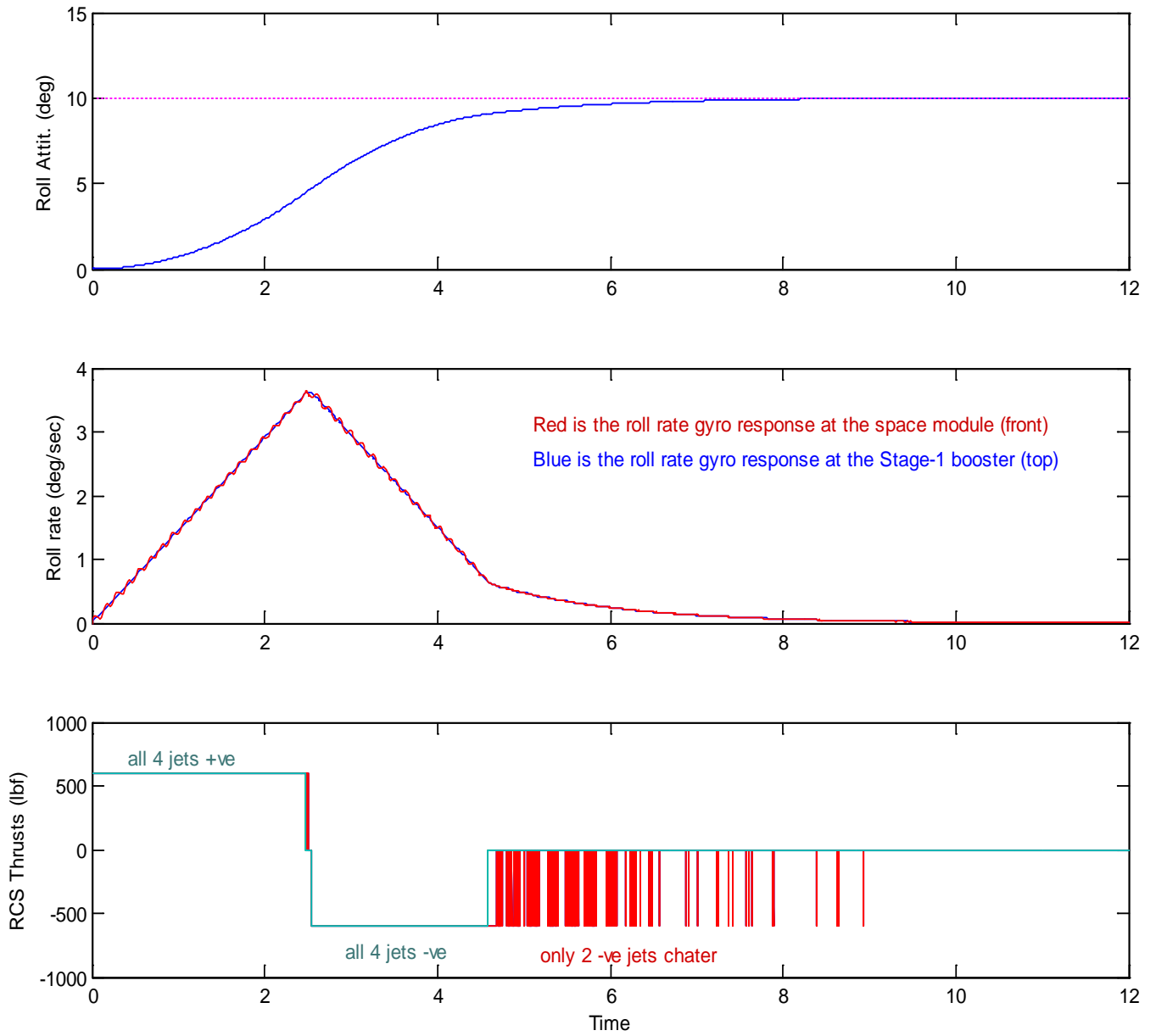all 4 jets -ve

only 2 -ve jets chater

Figure 15 Roll Reaction Control System response to 10 degrees attitude change command