

Space Shuttle Ascent, First Stage



In this example we are going to model the Space Shuttle vehicle during first stage and analyze the flight control system stability and performance at a flight condition where the dynamic pressure is maximum. During first stage the Shuttle propulsion system consists of three Space Shuttle main engines, (SSME) and two solid rocket boosters, (SRB). The vehicle attitude and direction is controlled by gimballing the five engines in pitch and yaw. The vehicle attitude, rate, and acceleration are measured by an IMU, rate gyros, and accelerometer sensors which are located at the top section of the SRBs. Roll attitude and rate are measured by sensors which are located in the orbiter vehicle. Inside the orange external tank (ET) there are two large tanks containing the propulsion fuel, a liquid oxygen (LOX) tank located in the front section, and a liquid hydrogen (LH2) tank in the aft section of the ET. During Max-Q the tanks are partially filled and fuel sloshing is a potential problem. If not properly controlled it may destabilize the vehicle by creating oscillatory disturbance forces. The control surfaces are not used during ascent but they are scheduled open-loop to minimize loading on the aero-surface actuators. Structural flexibility generated by engine gimballing is significant and it is included in the analysis. Aero-elasticity, however, is not as significant during ascent and it was not included in this analysis. It is included in the entry analysis.

The analysis is separated in three sections. In the first section we develop the pitch vehicle model and include 18 symmetric (pitch) structural modes. Fuel sloshing dynamics is also included (along z) for the two tanks which excite the vehicle in the pitch direction. The pitch Shuttle model is then discretized and combined with the digital flight control system to create control analysis models in the

z-domain and the models are used to evaluate the control system stability and performance using Matlab/ Simulink. In the second section we develop dynamic models that include both pitch and lateral dynamics. In addition, 29 flex modes are also included in the dynamic model which are both symmetric (pitch) and anti-symmetric (roll/ yaw). The sloshing of the two propellant tanks is also included in the dynamic model. It is represented by two masses oscillating in two directions perpendicular to the vehicle acceleration vector. The spring-mass analogy is used where the slosh masses are oscillating about the center of the tank, along the y and z directions. The analysis consists of stability analysis in the frequency domain to determine the phase and gain margins, sensitivity to wind gusts, and also time domain simulations of the vehicle in response to attitude commands from guidance and also responses to wind gusts. In the third section we shall redesign the Shuttle Ascent flight control system at Max-Q and attempt to improve the vehicle sensitivity to wind-gust disturbances. The original classical control design demonstrated poor wind-gust sensitivity performance with an undesirable resonance at the expected disturbance frequencies. We will use the H-infinity method which inherently attempts to reduce sensitivity to disturbances.

The data files for this analysis are located in directory "C:\Flixan\Examples\Shuttle Ascent". This directory contains the files used for creating the Shuttle vehicle models and for analyzing its performance in Matlab. There are three subfolders corresponding to the three sections. "Pitch Analysis", and "Coupled Vehicle" where the pitch and lateral stability analysis and simulations are performed using Matlab and Simulink. The folders contain structural flexibility data files (*.Mod, *.Nod), vehicle input data (*.Inp), and system files (*.Qdr). The third subdirectory is "Hinfinity Design" which corresponds to Section 3.

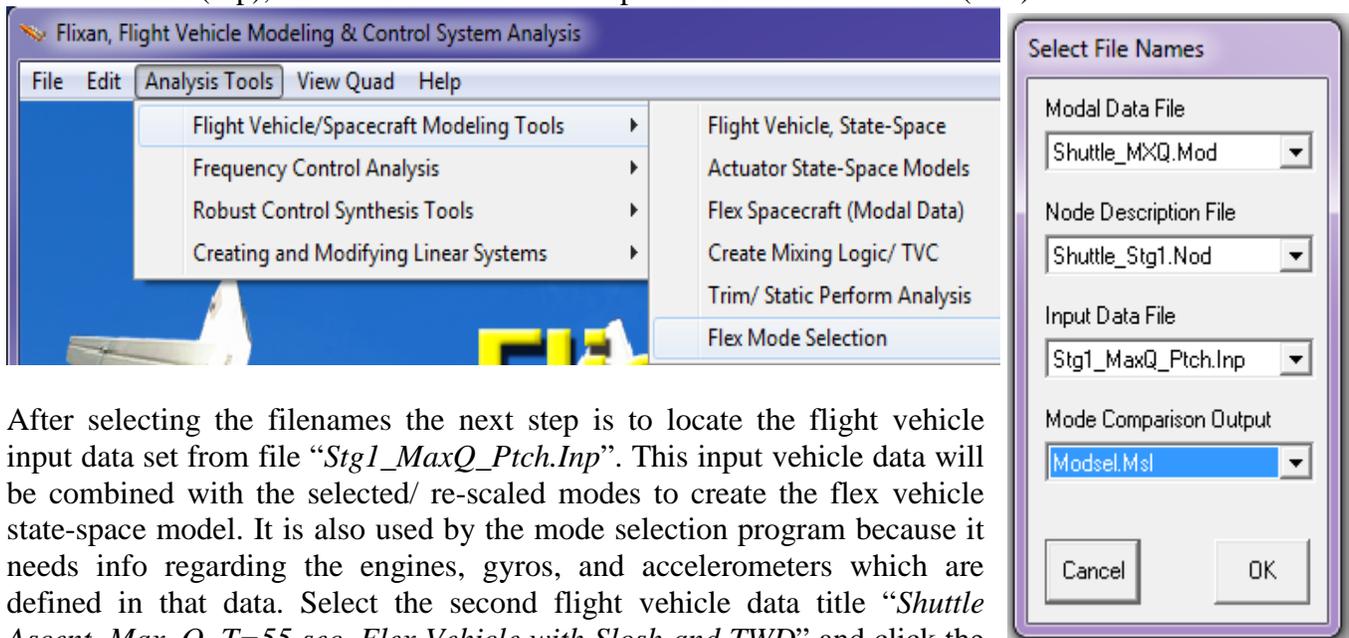
1.0 Pitch Axis Vehicle Modeling and Analysis

In the first part of the analysis we generate the pitch Shuttle model. We will use the flight vehicle modeling program to generate the vehicle state-space models using the vehicle parameters during max-Q which are already saved in the input file "Stg1_MaxQ-Ptch.Inp". The vehicle parameters during max-Q are obtained from six-dof simulations. Two pitch state-space systems are created for this analysis: a simple rigid-body, and a high order system that includes slosh and bending. The titles of two system generated by the modeling program are: "Shuttle Ascent, Max_Q, T=55 sec (Rigid Body)" and "Shuttle Ascent, Max_Q, T=55 sec, Flex Vehicle with Slosh and TWD". The second system contains 18 symmetric structural modes that have already been selected from the modal data file "Shuttle_MXQ.Mod". The set of the 18 selected symmetric modes are also included in the input data file and the title of the set is "Shuttle Ascent, Max_Q, T=55 sec, Rigid Body with Slosh/ TWD, Pitch Modes". The input file is processed by Flixan and the state-space systems used in pitch analysis are saved in the systems file "Stg1_MaxQ_Ptch.Qdr". The Shuttle state-space systems created by the vehicle modeling program include both pitch and lateral dynamics. The pitch flexible system is extracted from the coupled system using the systems extraction and modification utility. The reduced system title is "Shuttle Ascent, Max_Q, T=55 sec, Flex Pitch Vehicle with Slosh and TWD". But before we get to the analysis let us describe how the modes were selected from the modal data file.

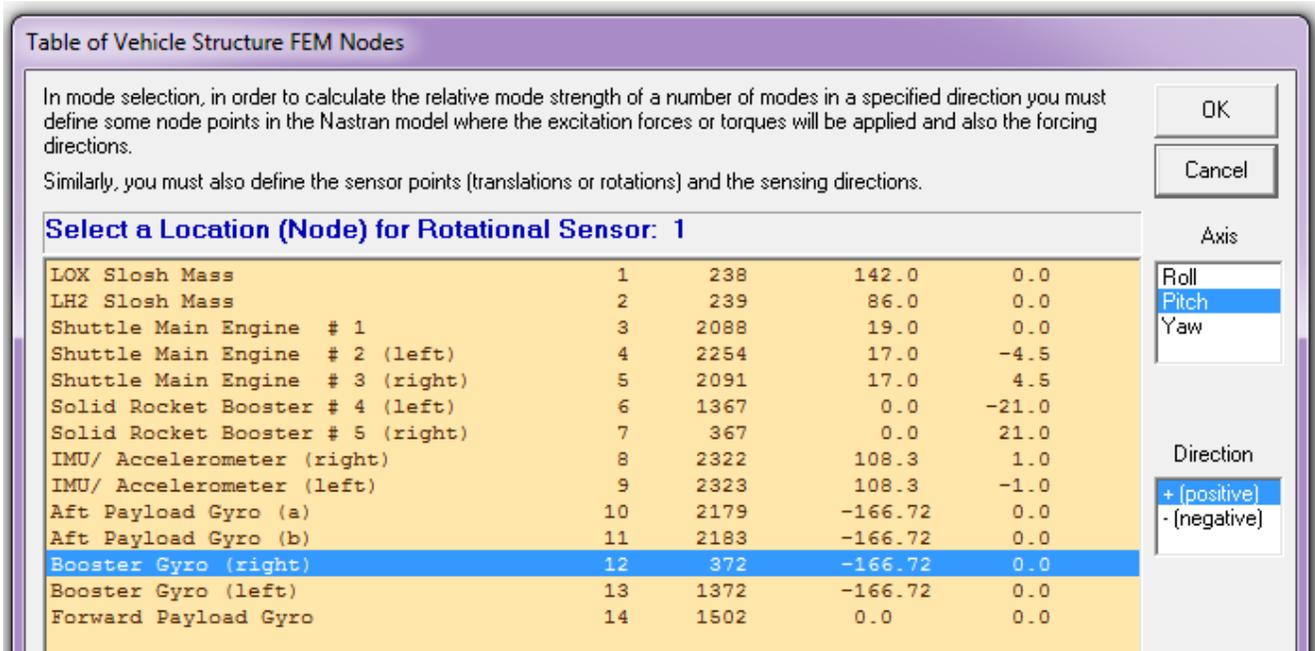
1.1 Pitch Mode Selection

The file that contains the finite element structural modes is “*Shuttle_MXQ.Mod*”. It is a Nastran output that contains the mode shapes and slopes for 95 modes, at 14 vehicle location (also known as nodes). This file is extracted from a big finite element model (FEM) file that includes several hundred modes and many vehicle locations. It is more manageable because it includes fewer modes (only the first 95 modes) and fewer locations (only 14 relevant to control analysis) and it is formatted to be recognizable and readable by the mode selection program. An intermediate program was used (not shown here) to convert the huge Nastran output to this smaller file. This modal data file contains frames of data at each mode frequency. Each frame consists of the mode frequency in (rad/sec), the modal damping coefficient (ζ) which are all set to $\zeta=0.005$, the generalized mass (all set to 10,000 in this case), followed by the mode shapes and slopes at the 14 vehicle locations (3 translations x, y, z, and 3 rotations about x, y, z). The locations which are important for flight control analysis are the engine gimbals, the sensors, and the slosh mass locations. The vehicle locations (nodes) are defined in a separate file “*Shuttle_Stg1.Nod*”, known as the map. This file contains a description for each node, the node numbers (in this case 1 to 14), a node identification number (that is the node number used in Nastran), and the node location in vehicle coordinates (which is only for reference and not used by the program).

The mode selection is a Flixan process by which we select an even smaller number of modes (18 dominant modes in this case) to be combined with the rigid vehicle model. The mode shapes are also scaled during this process in order to be compatible with the rigid-body model. The flex mode selection program is under the “*Flight Vehicle/ Spacecraft Modeling*” group, as shown below. It begins with a small filenames selection menu where the user selects the modal data file, the nodes file, the flight vehicle input data file, and an output filename (Modsel.Msl by default) that shows the relative mode strength at the completion of the mode selection, see below. The modal data input filenames must have an extension (Mod). The node filenames must have an extension (Nod), the vehicle input data filenames (Inp), and the mode selection output filenames extension is (Msl).



After selecting the filenames the next step is to locate the flight vehicle input data set from file “*Stg1_MaxQ_Ptch.Inp*”. This input vehicle data will be combined with the selected/ re-scaled modes to create the flex vehicle state-space model. It is also used by the mode selection program because it needs info regarding the engines, gyros, and accelerometers which are defined in that data. Select the second flight vehicle data title “*Shuttle Ascent, Max_Q, T=55 sec, Flex Vehicle with Slosh and TWD*” and click the “*Run Input Set*” button.



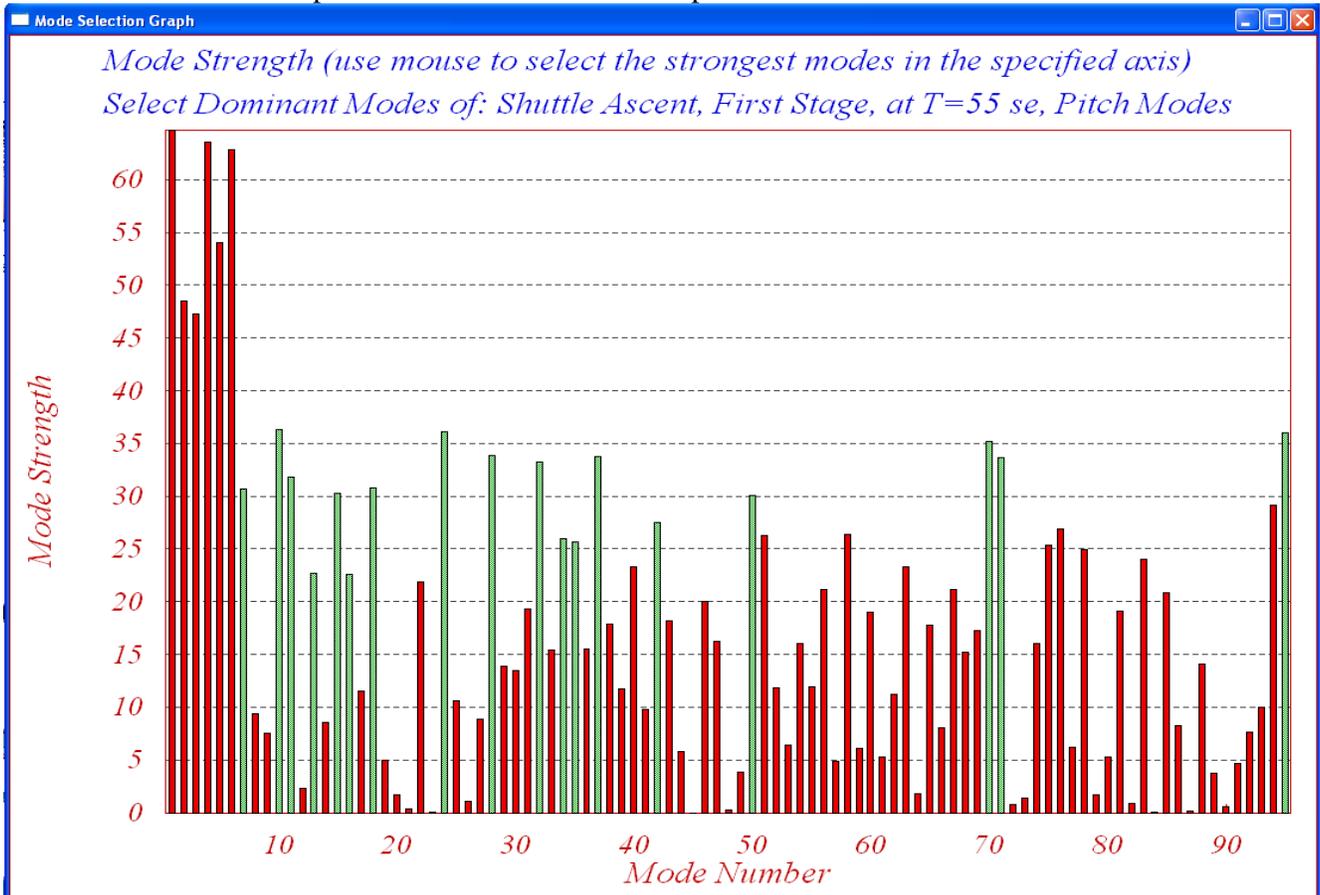
At this point we have identified all the excitation and sensor points and directions for the mode comparison and mode selection process which is mainly in the pitch direction. The modal strength for each mode is determined by the values of the mode shapes at the nodes where the forces are being applied in the force direction, and also by the values of the mode shapes at the sensors in the directions measured. High mode shape values at the excitation and sensor points imply strong contribution from that mode. The mode selection program calculates the mode strength for each mode and saves it in file "Modsel.Msl", from where the user can select the dominant modes graphically.

The selection process, however, is not finished yet because the program needs additional information before allowing the user to select which modes to retain from the modal data file. The mode selection program will create a much smaller subset of the original modal data set and it will save it in the input data file "Stg1-MaxQ-Ptch.Inp" under a separate title. The selected modal data set will contain only the selected few dominant modes and modal shapes at only a few locations that play an important role in the flight vehicle model such as the engine gimbals, the sensors, a disturbance point, and the slosh mass locations. The important vehicle locations (gimbals, sensors, etc) are identified automatically by the mode selection program from the vehicle input data. These locations are matched with the corresponding nodes in the nodes map. The mapping between structural nodes and the vehicle locations in the input data is performed by the user using menus similar to the previous location selection menus but this time it is for a different purpose. The program expects the user to identify nodes for the 5 engines, the 3 gyros (roll, pitch, and yaw), the 2 accelerometers (Ny and Nz), the 2 slosh masses (LOX and LH2), and one disturbance point (not used).

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



At this point a bar chart comes up 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 not a physical number but it is measured relative to the weakest mode. The user manually selects some of the strongest modes by pointing the mouse cursor at the bar and clicking the mouse. The modes change color from red to green when they are selected. Notice, that the first six modes, although strong, they were not selected because they are rigid-body modes. The rigid-body dynamics are already included in the vehicle model. We select 18 flex modes and press the "enter" button to complete the mode selection.



The final step before exiting the mode selection is to type-in some additional reference notes regarding the mode selection process by using a dialog. The notes describe, for example, information regarding what kind of modes were selected and the conditions of mode selection, the excitation and measurement points, directions, etc. This information will be included as comments below the title in the selected modes set, which is saved in the input data file "Stg1_MaxQ_Ptch.Inp". The title of the selected modes set is "Shuttle Ascent, Max_Q, T=55 sec, Rigid Body with Slosh/TWD, Pitch Modes". It contains the frequencies and mode shapes of the selected pitch modes at vehicle locations that correspond to the locations defined in the vehicle data. The title of the selected modes set must also be included at the bottom of the flexible vehicle input data set (below the number of flex modes). This is how the flight vehicle modeling program will associate the flex modes with the vehicle input data.

1.5 Combining the Vehicle with the Actuators

The next step is to combine the vehicle model with the actuators and the flight control system. There are many ways to combine the subsystems together and we will consider two options: (a) combining all the individual subsystems together using Simulink, and (b) use the Flixan systems interconnection utility to combine all the continuous (s-plane) subsystems together in order to create a continuous state-space plant model and then complete the system combination using Simulink by connecting the continuous plant model with the discrete flight control system and to create simulation models in Simulink. We did it both ways, but in this section we shall describe how to combine the pitch vehicle with the two actuators together using Flixan.

The interconnection of the vehicle with the actuator models is not as simple as connecting the actuator output to the deflection input of the vehicle model. There is additional dynamics involved that makes the actuator/ vehicle interconnection a little more complex. There is the tail-wag-dog dynamics, which is a reaction torque on the vehicle caused by the slewing acceleration of the nozzle. The vehicle model in addition to nozzle deflection inputs it also has gimbal acceleration inputs for modeling the tail-wag-dog (TWD) dynamics. There is also the load-torque feedback which is a disturbance torque applied on the actuator gimbal and is attempting to rotate the nozzle against the actuator control torque. It is caused by the vehicle acceleration, both rotational and translational. On the Shuttle and most launch vehicles the load-torque effects are not as significant than the TWD, that usually creates a complex pair of zeros in the transfer function near the $j\omega$ axis. This causes a sharp notch in the open-loop frequency response amplitude of $\{q(\omega)/\delta(\omega)\}$ and a 180° phase reversal at the tail-wags-dog frequency. This happens only if the engine cg is below the gimbal, otherwise, the TWD zeros are on the real axis of the complex plane. In order to model the TWD dynamics two things are needed, an actuator that provides a gimbal acceleration output, that is, in addition to the gimbal output angle, and a flight vehicle state-space model that includes gimbal acceleration inputs. In the vehicle/ actuator interconnection, the gimbal angle outputs from the actuator connect to the gimbal angle inputs of the vehicle model, and the gimbal acceleration outputs from the actuator go to the gimbal acceleration inputs of the vehicle model. This takes care of the tail-wags-dog modeling.

For the load-torque feedback modeling, the actuator in addition to the (δ -com) it must also have a load-torque input, which is a mechanical reaction torque in (ft-lb) on the nozzle gimbal caused by the vehicle motion. The load-torque attempts to rotate the nozzle in a direction that may be against the actuator control torque. In order to model the TVC load-torques correctly a separate actuator model must be included for each engine gimbaling direction. In this case, however the model can be simplified and instead of using 5 separate actuators directly connecting to the 5 vehicle pitch gimbal deflection and acceleration inputs, we lump the 3 SSME actuators in a single SSME model and the two SRB actuators in a single SRB actuator. A single SSME actuator deflection output drives all 3 SSME deflection inputs in the vehicle model via the TVC gains, and the actuator acceleration output drives all 3 SSME acceleration inputs via the same TVC gains. The same applies for the single SRB actuator output deflection driving the two SRB deflection inputs in the vehicle model, and the acceleration output driving the two acceleration inputs. It is mathematically equivalent to the actual physical system and it uses fewer states. The load-torque feedback outputs from the vehicle system are also connected to actuator load-torque inputs. The 3 load-torque outputs from the SSME are averaged to a single signal (Tly_ssme) that goes to the SSME actuator, and also the 2 SRB load-torque outputs are averaged to a single signal (Tly-srb) that goes into the SRB actuator load-torque input. The flight vehicle modeling program creates the additional inputs and outputs required to implement the TWD and the load-torque feedback dynamics when the TWD flag is turned-on in the TVC engines “With TWD”, in the vehicle input data.

Shuttle Stage-1 at Max-Q, Closed-Loop Simulation

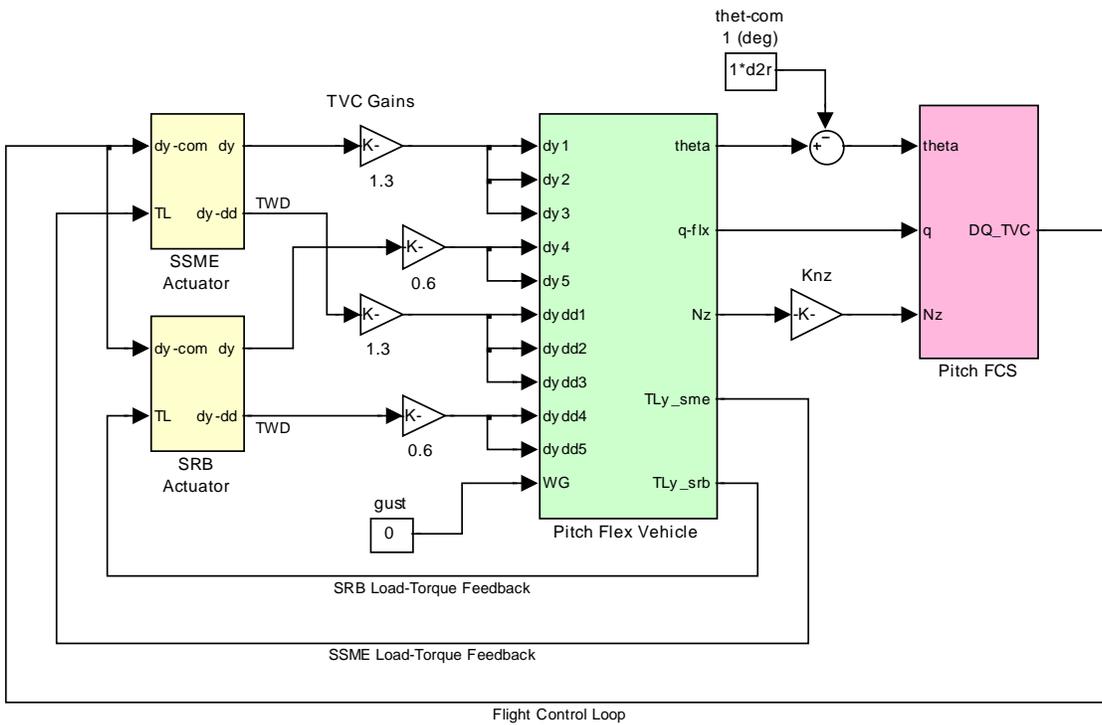
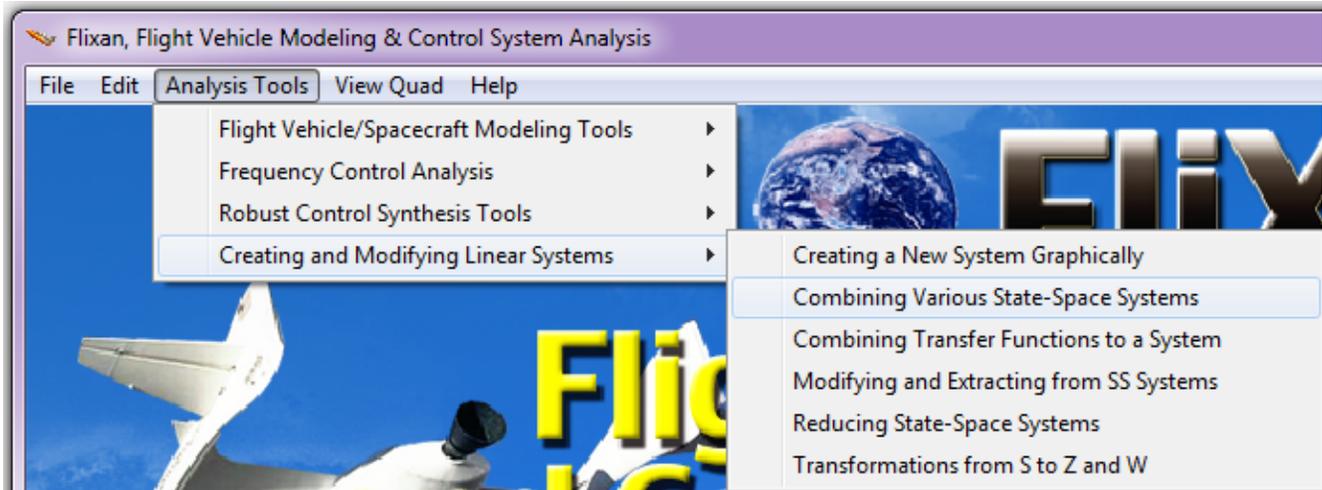


Figure (1.5.1) Pitch Vehicle and Actuators Interconnection

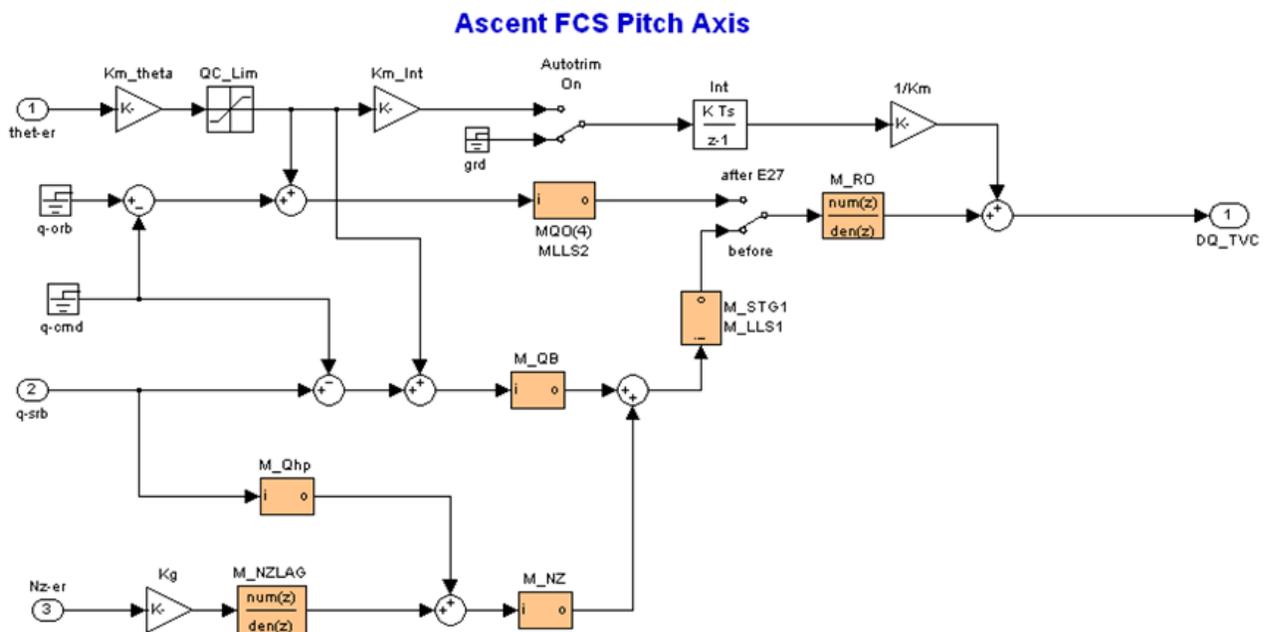
The pitch vehicle and the actuators interconnection is shown in Figure (1.5.1). We use the systems interconnection utility to combine them as shown. The systems interconnection data is in file “*Stg1_MaxQ_Ptch.Inp*”. Its title is “*Pitch Plant (Flex Vehicle, Actuators, TVC, TWD)*”. The state-space systems combination utility is also under the “*Creating and Modifying Linear Systems*” group.



1.6 Pitch Axis Discrete Flight Control System

The pitch flight control system of the Space Shuttle during ascent is defined by a combination of discrete transfer functions in the z-domain. The inputs are: vehicle pitch attitude error (Θ) in (radians), pitch rate (q) in (rad/sec), and normal acceleration (N_z) in (ft/sec²). The output DQ_TVC is the command that drives the pitch actuators via the TVC. The gains and filters are designed strictly for the Max-Q flight condition. In this example we are not emphasizing in the flight control system description and design, but only in the overall vehicle modeling and performance analysis. The pitch flight control system is also included in the input data file “*Stg1_MaxQ_Ptch.Inp*” together with the other data related with this analysis. It is defined in transfer functions form. The transfer functions interconnection title is: “*Pitch Axis Discrete FCS*”, and its sampling rate is 40 (msec). In order to be able to combine this transfer-function defined FCS with the other vehicle subsystems we must convert it to a discrete state-space system. The Flixan transfer functions interconnection utility will be used to create the discrete state-space FCS and will save it in the systems file together with the other systems.

Pitch Flight Control System



The transfer-function data processing utility in Flixan is selected from “*Combining Transfer-Functions to System*” option which is under the “*Creating and Modifying Linear Systems*” group and it is similar to the systems interconnection option. In the filename selection menu you must select the same input data and systems filenames already used in this project. In the transfer functions interconnection selection menu select “*Pitch Axis Discrete FCS*”. Run the selected set and the transfer function interconnections dialog appears showing the interconnection info.

1.8 Pitch Axis Stability Analysis Using Matlab

In the previous section we have created several models that can be used to analyze open-loop stability using frequency response analysis in Matlab. The following Matlab script file "run.m" loads the Flixan generated models to Matlab and performs frequency response analysis. An open-loop model is already created by Flixan and it is saved in file "open-loop.m". However, in this case we will use Simulink to connect the subsystems together open-loop and to perform the linear analysis.

```
% File: run.m
r2d= 180/pi; d2r=1/r2d; Ts=0.04;
% Load Systems from Flixan
[Apf,Bpf,Cpf,Dpf]= pitch_vehi_flex;
[Aac,Bac,Cac,Dac]= ssme_actuator;
[Asb,Bsb,Csb,Dsb]= srb_actuator;
[Afc,Bfc,Cfc,Dfc]= pitch_fcs;
[Apl,Bpl,Cpl,Dpl]= pitch_plant_s;
[Apz,Bpz,Cpz,Dpz]= pitch_plant_z;
[Aol,Bol,Col,Dol]= open_loop;
[Acl,Bcl,Ccl,Dcl]= closed_loop;

% Titles in file "Stg1_MaxQ_Ptch.Inp"
% Shuttle Ascent, Max_Q, T=55 sec, Flex Pitch Vehicle with Slosh
% Shuttle Main Engine Actuator
% Solid Rocket Booster Actuator
% Pitch Axis Discrete FCS
% Pitch Plant (Flex Vehicle, Actuators, TVC, TWD)
% Pitch Plant Discrete
% Open-Loop Pitch (Plant Model + Controller)
% Closed Loop (Plant Model + Controller)

[Ao,Bo,Co,Do]= dlinmod('Pitch_Open_Flex',0.04); % Linearize Open-Loop Simulink model that uses separate blocks
%[Ao,Bo,Co,Do]= dlinmod('Pitch_Open_Flex2',0.04); % Linearize Open-Loop Simulink model that uses combined plant
sys= SS(Ao,Bo,Co,Do,0.04); % Create SS System
%sys= SS(Aol,Bol,Col,Dol,0.04); % Create SS System using the pre-combined z-system "open_loop.m"
w=logspace(-2.4, 2, 24000); % Define Frequ Range
figure(1); Nichols(sys,w) % Plot Nichol's Chart
figure(2); Bode(sys,w) % Plot Bode
figure(3); Nyquist(sys,w) % Plot Nyquist
```

The open-loop Simulink interconnection file "*Pitch_Open_Flex.Mdl*", shown in Figure (1.7.1), has the loop opened at the FCS output. It is used to calculate the frequency response and the stability margins.

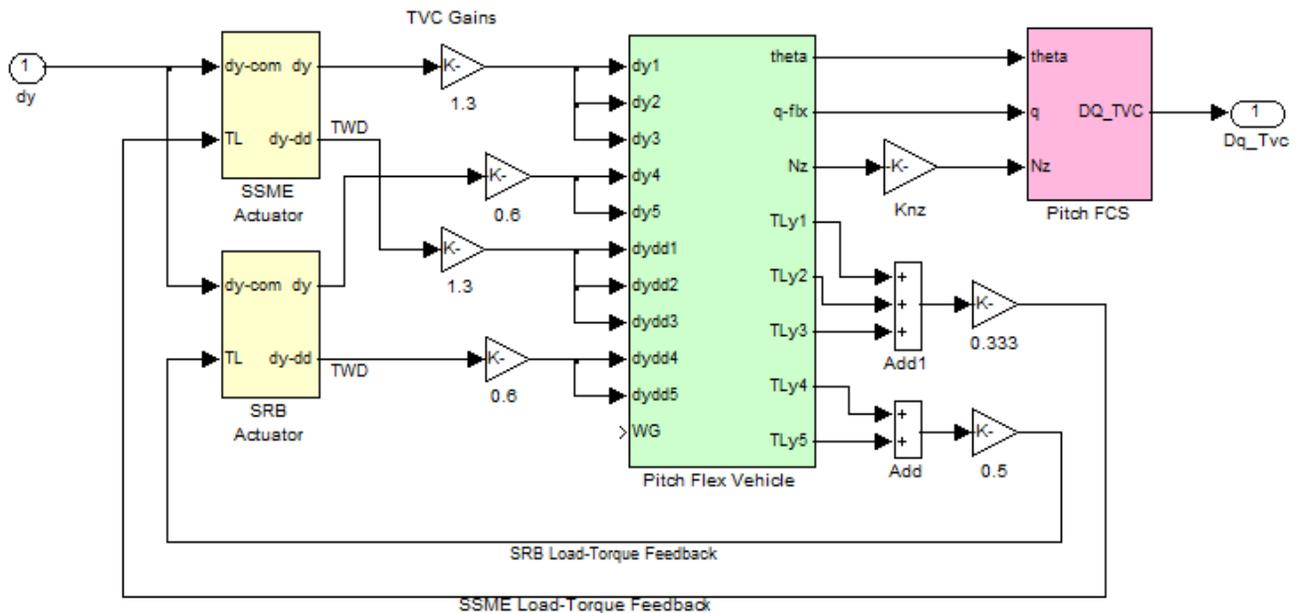


Figure (1.7.1) Open-Loop Analysis Model "Pitch_Open_Flex.Mdl" with the Loop Opened at the FCS Output

This system consists of the continuous pitch flex vehicle from file "*pitch_vehi_flex.m*", the two actuator (SSME and SRB) subsystems, and the TVC gains in series with the discretized flight control system "*pitch_fcs.m*", sampled at 0.04 sec. Matlab calculates the frequency response and plots the Bode, Nichols, and Nyquist diagrams to show the gain and phase margins.

We also have a second open-loop implementation in Simulink file "*Pitch_Open_Flex2.Mdl*", shown in figure (1.7.2) below, and used for tutorial purposes. This implementation connects the FCS with the combined vehicle/ actuator/ TVC plant system: "*Pitch Plant (Flex Vehicle, Actuators, TVC, TWD)*" that was connected together using Flixan, and it was loaded in Matlab from file "*pitch_plant_s.m*". A third implementation uses the open-loop system "*Open-Loop Pitch (Plant Model + Controller)*" generated by Flixan, loaded in Matlab from file "*open_loop.m*". This consists of all subsystems including the FCS. The frequency response results from all three systems are very similar and they are presented in all three: Bode, Nyquist, and Nichols flavors.

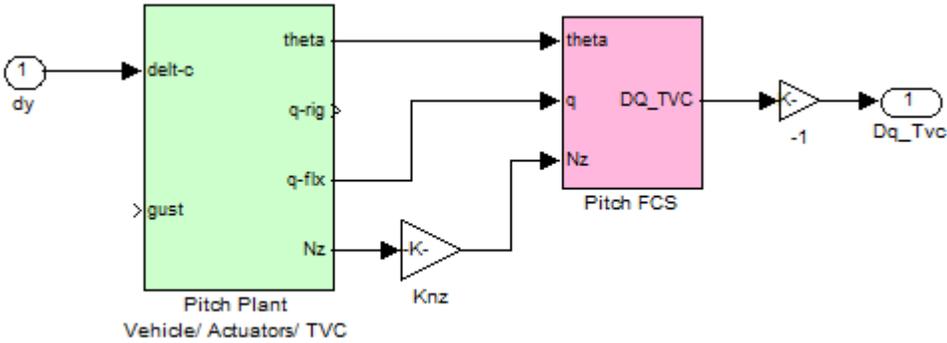
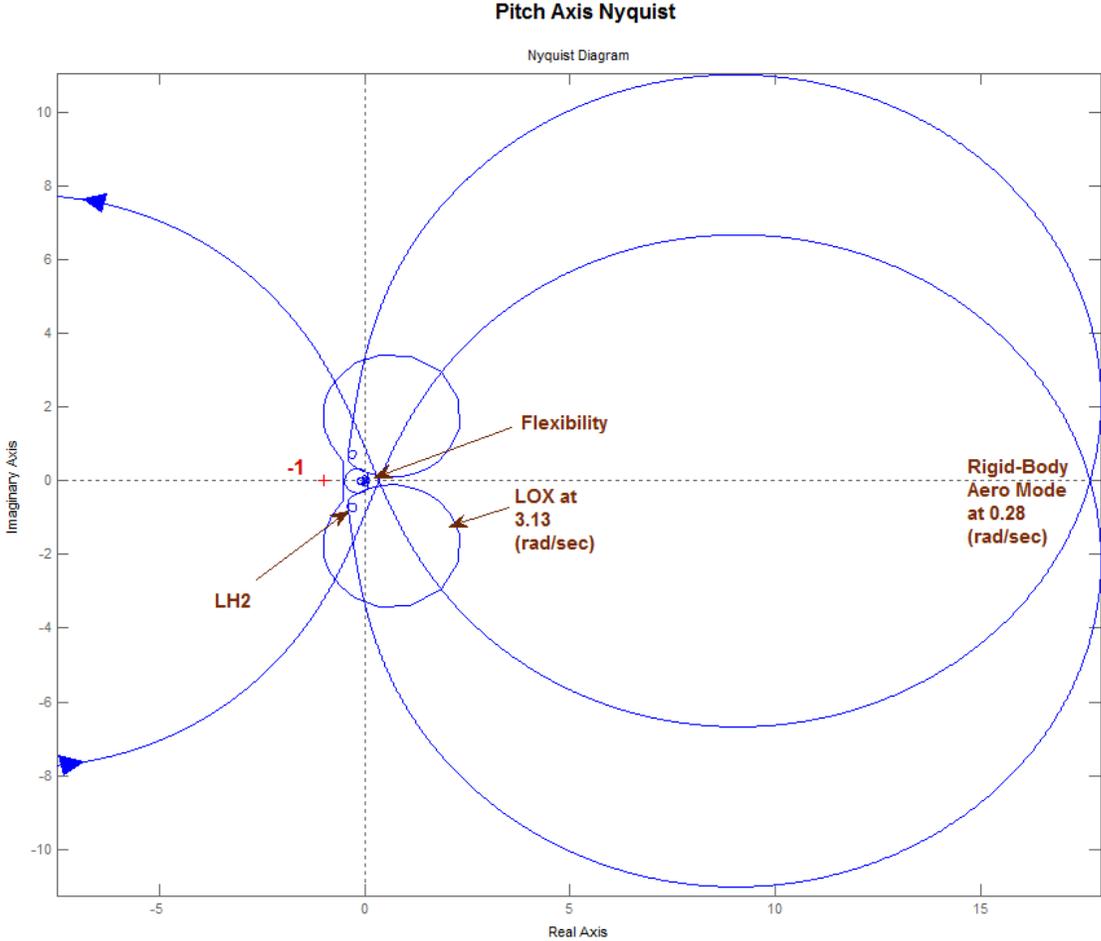
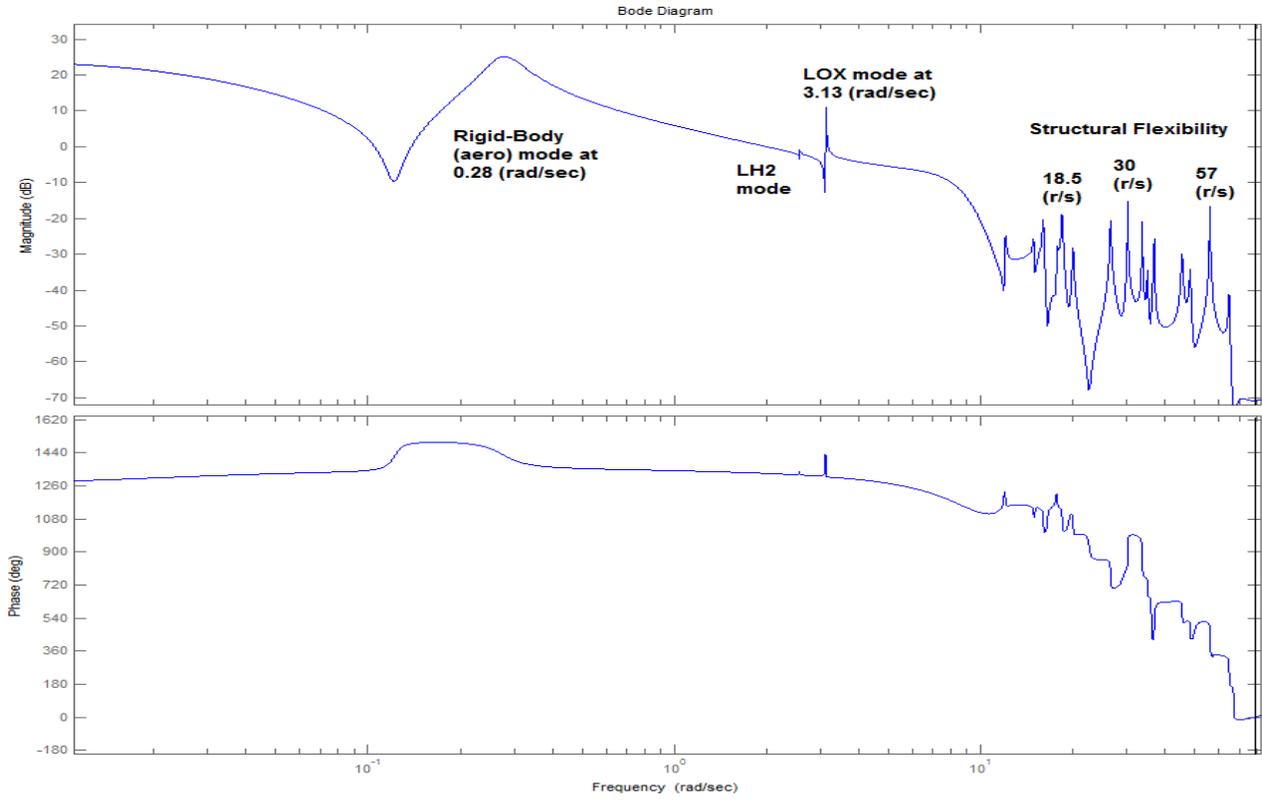


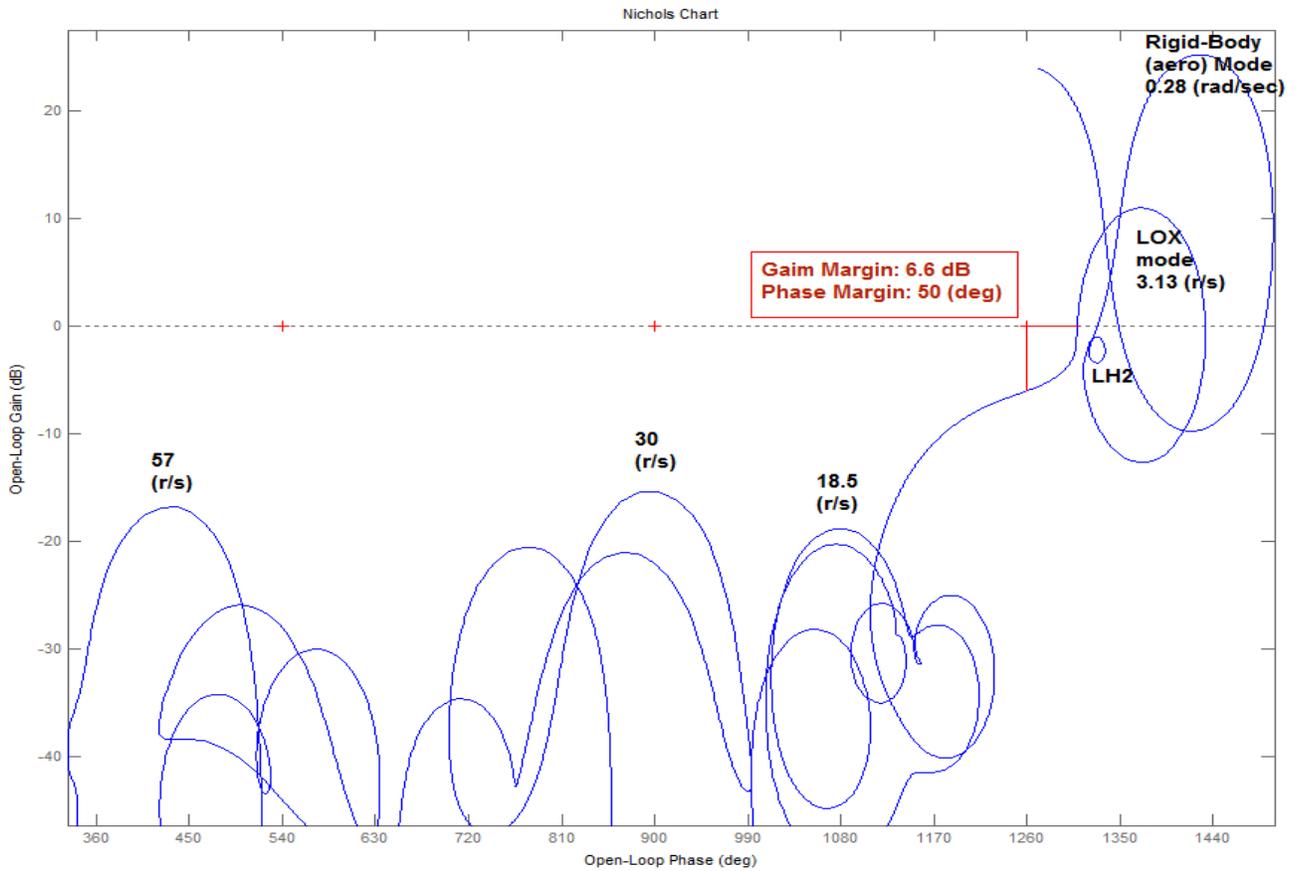
Figure (1.7.2) Open-Loop Analysis Model "*Pitch_Open_Flex2.Mdl*" using the pre-combined plant model



Pitch Axis Bode Plot



Pitch Axis Stability Analysis



1.8 Closed-Loop Simulations

The closed-loop simulations are also performed in Simulink, and there are three simulation models located in folder “C:\Flixan\Examples\Shuttle Ascent\Pitch Analysis”. The first one is implemented in the Simulink model “Pitch_Flex_Sim.Mdl”, shown in figure (1.8.1), where the vehicle, actuators, flight-control, and the TVC are combined together in Simulink.

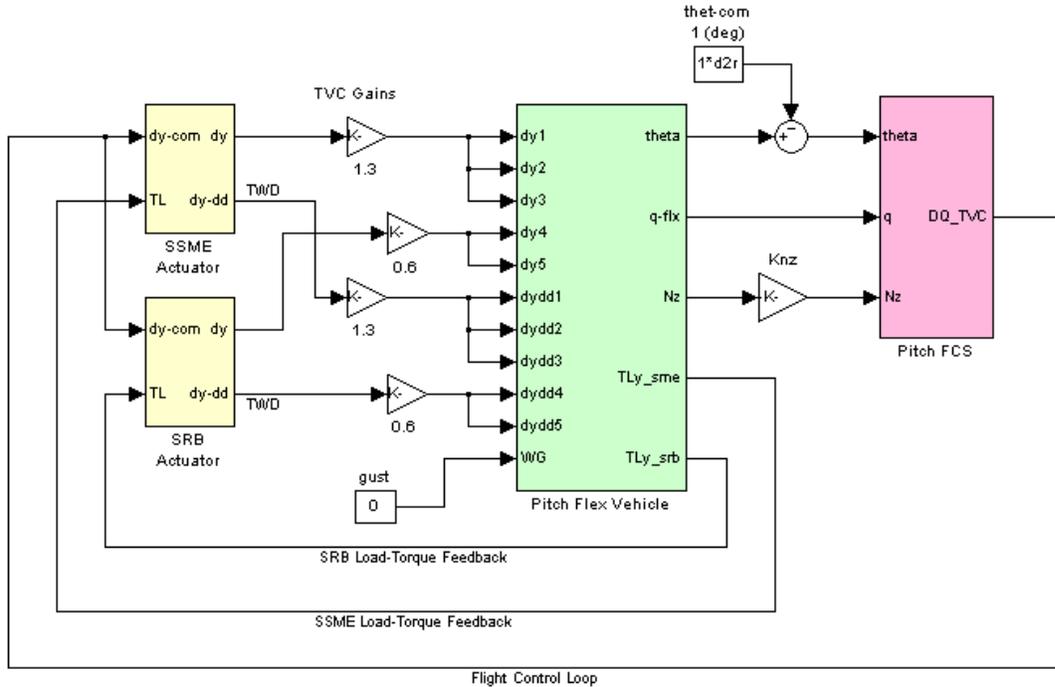


Figure 1.8.1 Shuttle Ascent Closed-Loop Simulation Model “Pitch_Flex_Sim.Mdl”

The second closed-loop simulation model is “Pitch-Flex-Sim2.mdl”, shown in figure (1.8.2). It uses a pre-combined plant model: “Pitch Plant (Flex Vehicle, Actuators, TVC, TWD)”, shown in green below, where the vehicle and actuators were connected together in Flixan.

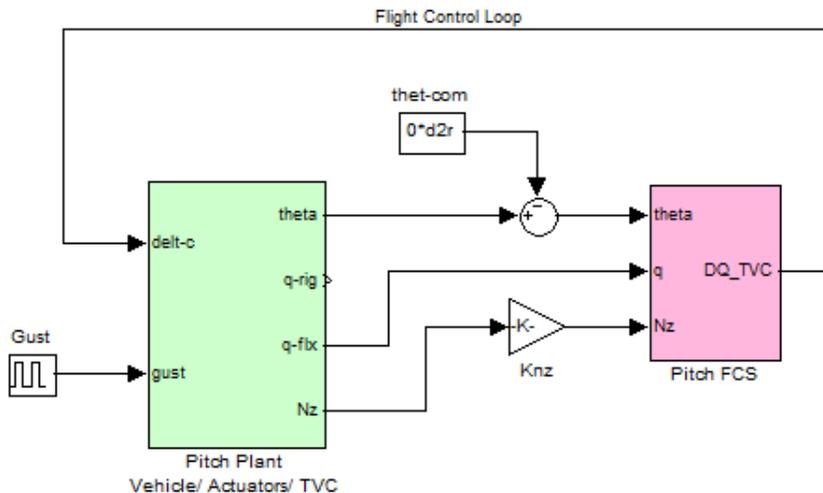


Figure 1.8.2 Shuttle Ascent Closed-Loop Simulation Model “Pitch_Flex_Sim2.Mdl”

The third closed-loop simulation model is “*Pitch-Flex-Sim3.Mdl*”, shown in figure (1.8.3), and it was combined together and discretized in Flixan. The discrete system sample-rate is 40 (msec) and its title is “*Closed Loop (Plant Model + Controller)*”. It is loaded into Matlab from file “*closed_loop.m*”. The simulation results from all three models are very similar.

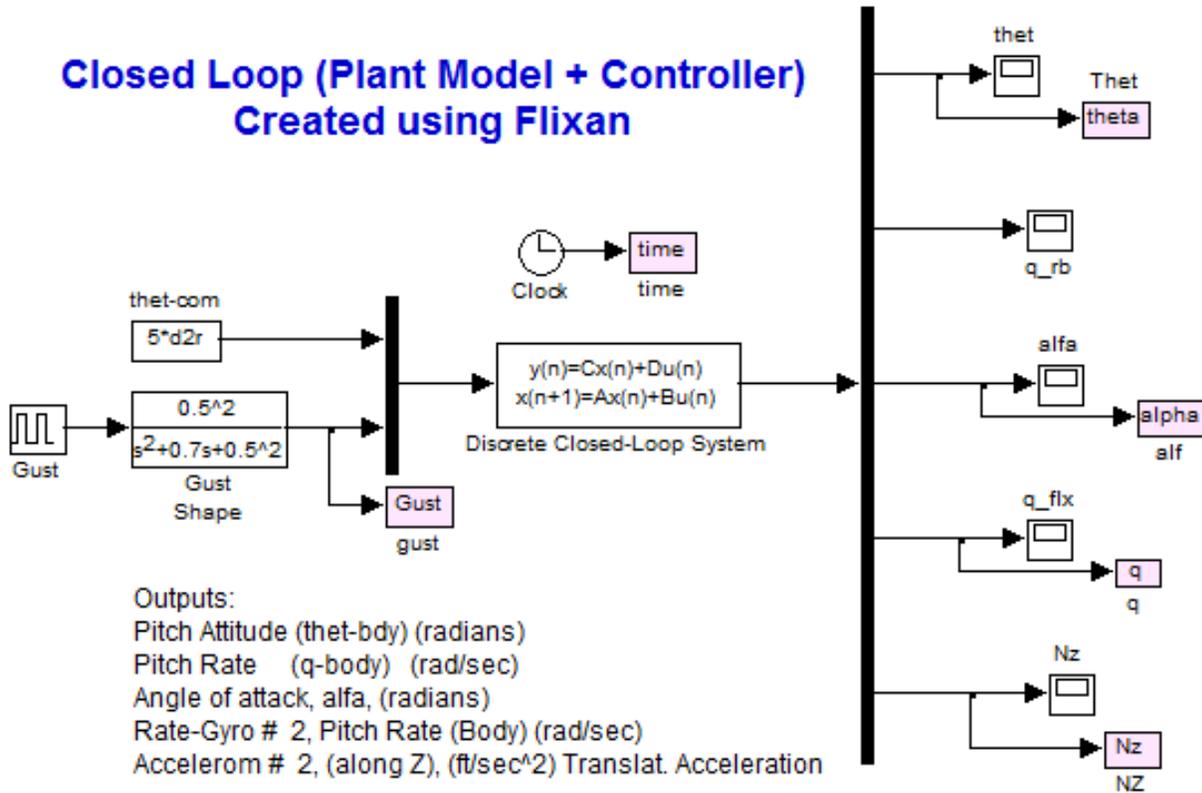
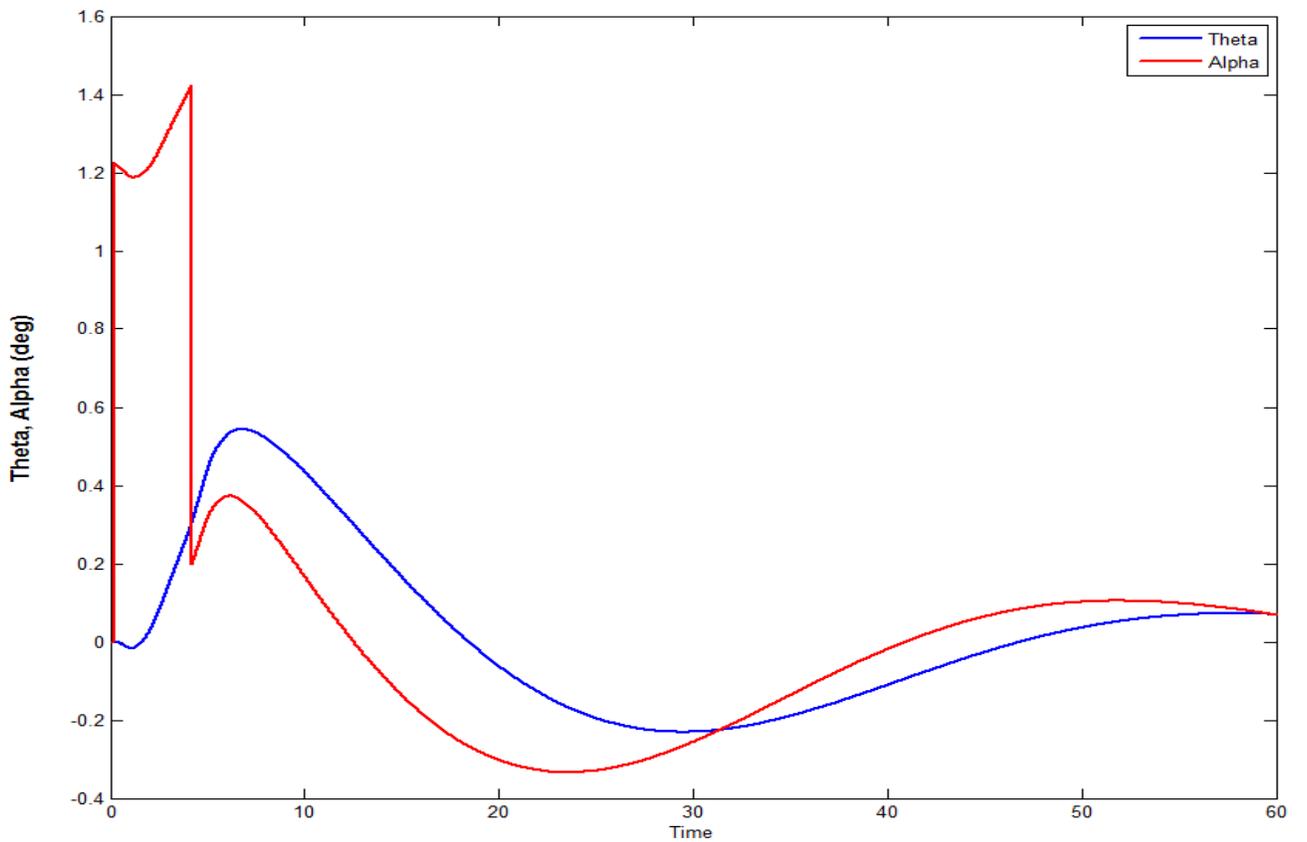
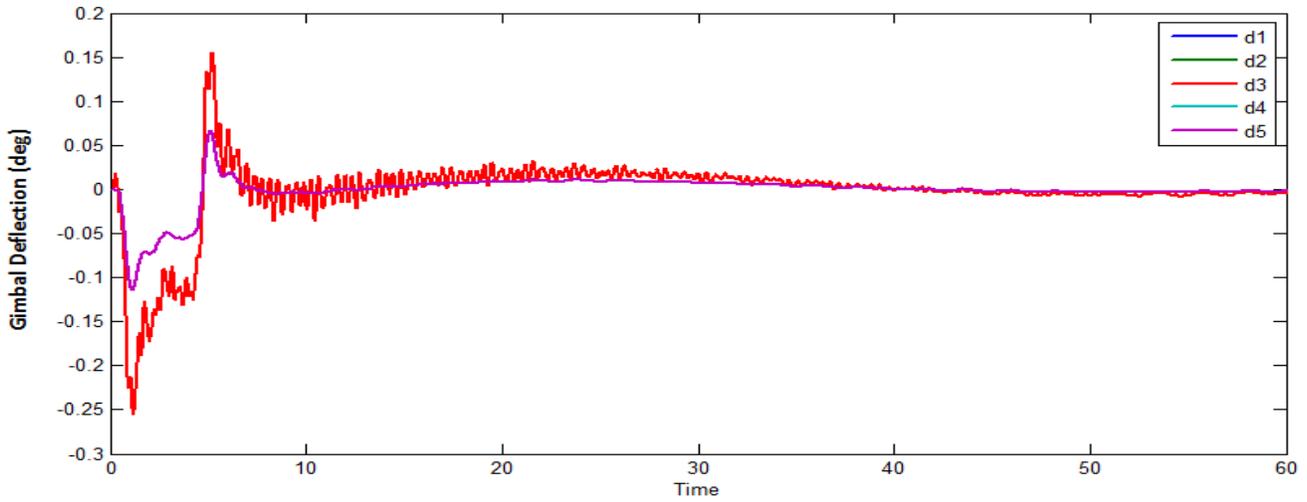
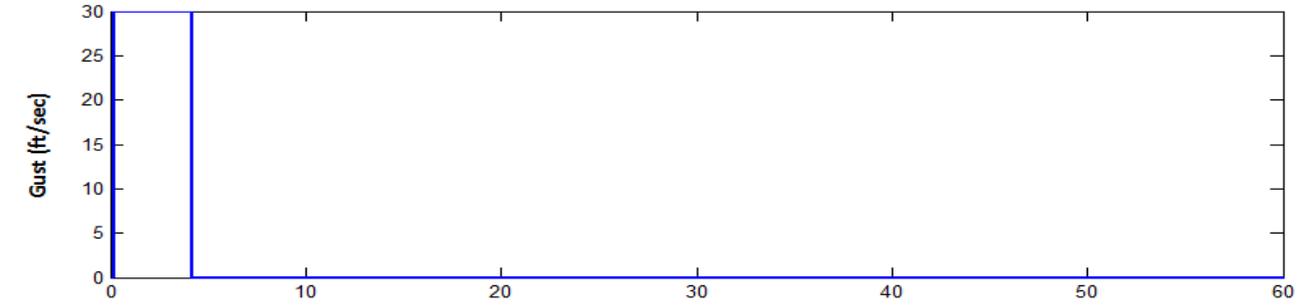


Figure 1.8.3 Shuttle Ascent Closed-Loop Simulation Model “Pitch_Flex_Sim3.Mdl”

The simulation models have two inputs, one representing an attitude (θ_{cmd}) coming from guidance and the second input represents the wind gust velocity. The direction of the wind gust is defined in the vehicle input data. When the simulations are complete a script file “pl.m” is used to plot the results.

Figures (1.8.4), below, show the vehicle response to a sharp wind pulse of 30 (ft/sec). It was obtained using the simulation model “*Pitch_Flex_Sim.Mdl*”. Initially the angle of attack jumps due to the wind incidence and the TVC engines gimbal to counteract the aero disturbance. This excites structure flexibility which is visible at the gyro, the accelerometer, and the gimbals. Slosh oscillations are also visible at the gimbal deflections. The SSME deflections are larger since they have bigger deflection capability than the SRB. The gimbal accelerations contribute to the TWD forces and the load-torques that feed back into the actuator.

Shuttle Ascent Stage-1 Response to 30 (ft/sec) Wind Gust Pulse



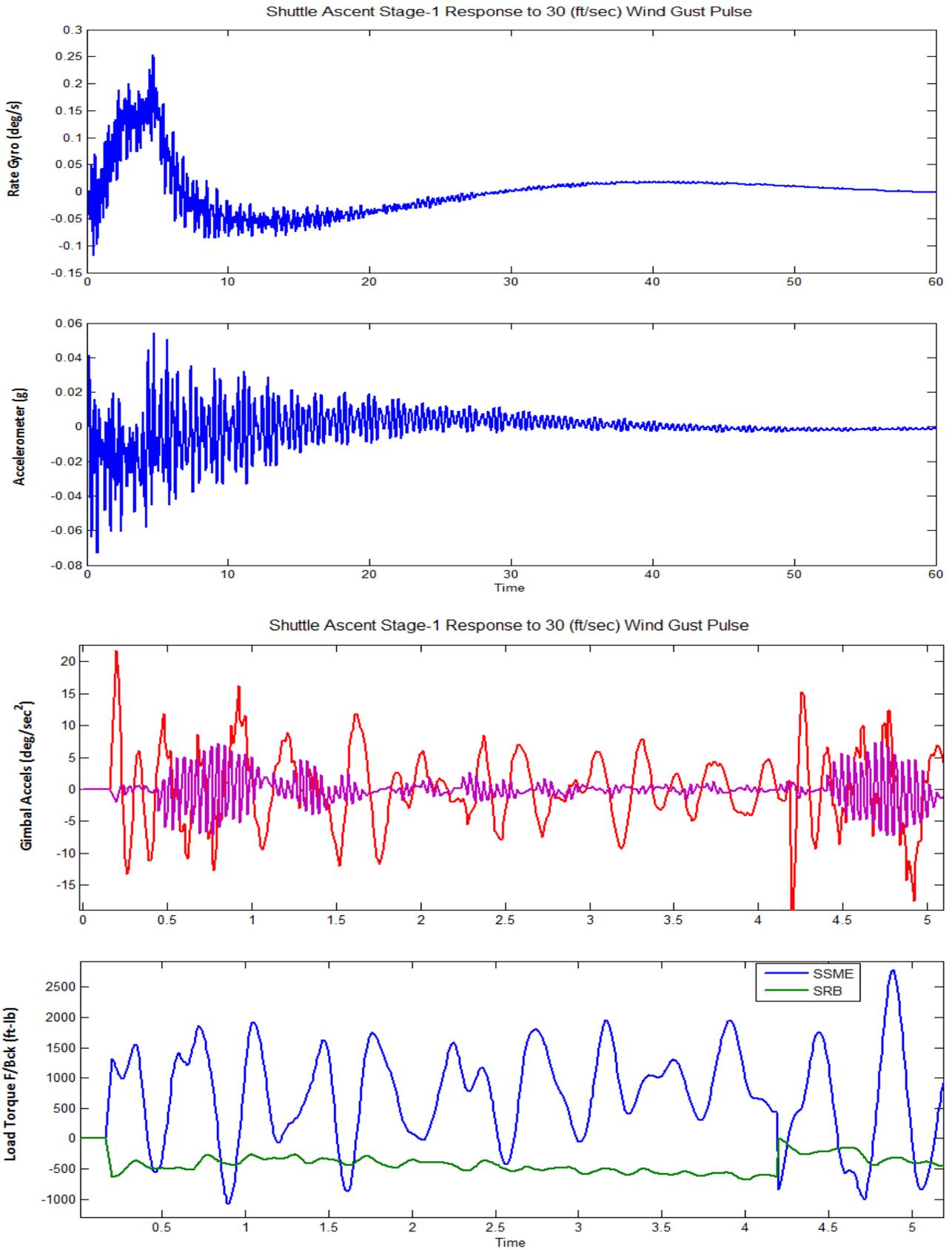
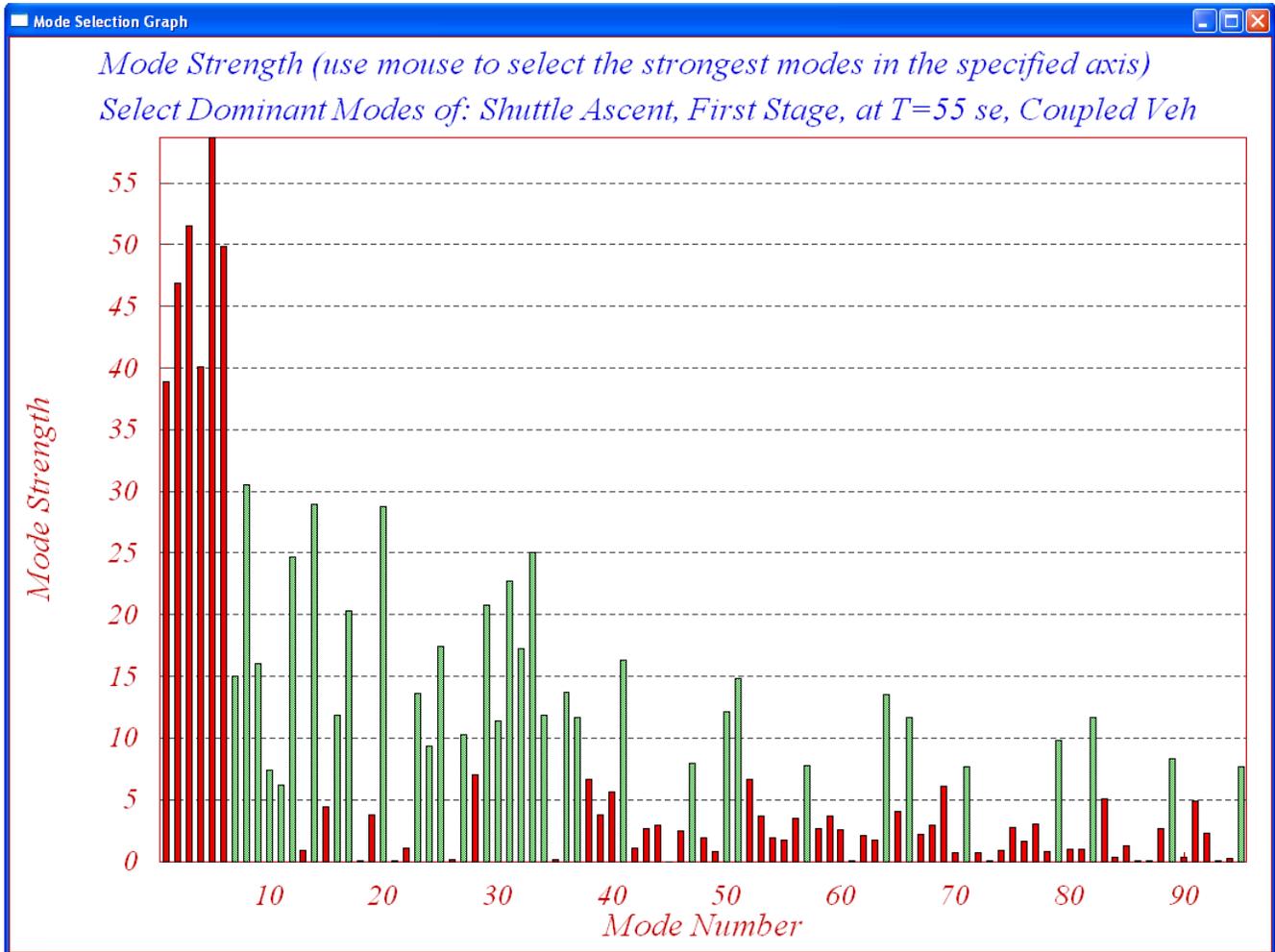


Figure 1.8.4 Response to a Wind Pulse of 30 (ft/sec) amplitude

2.0 Pitch and Lateral Axes Analysis Using a Coupled Model

We will now repeat a similar analysis by using a vehicle model that has the pitch and lateral dynamics fully coupled together and combined with discrete flight control systems, the actuators, the sensors, and the TVC matrix. We will use Flixan to generate state-space models for the Shuttle vehicle, both rigid and flexible, combine the flexible vehicle with the actuators, sensors, TVC, and flight control system, and create open-loop and closed-loop models for frequency response analysis and simulations. The control analysis is performed using the Matlab program. For tutorial purposes we shall demonstrate three types of modeling options using Flixan and Matlab/ Simulink. In the first case we create the individual subsystem models using Flixan and they are combined together and analyzed in Matlab/ Simulink. In the second option the vehicle, actuators, TVC, and sensor subsystems are combined together as a system and it is discretized using Flixan. Then we use Simulink to combine them with the discrete pitch and lateral flight control systems and analyzed them using Matlab. In option three all subsystems are combined together using Flixan, loaded into Matlab and the analysis is performed using Matlab/ Simulink. Notice that the Flixan interconnection utility is more efficient in combining subsystems together because Simulink has difficulty combining systems that involves algebraic loops. In this example we have algebraic loops because of the load-torque mechanical feedback.

The coupled axes analysis is performed in directory: "*C:\Flixan\Examples\Shuttle Ascent\Coupled Vehicle*". The Shuttle vehicle input data and the selected modal data are in file "*Shuttle-Stg1-T55.Inp*". If you look inside this file there are two sets of vehicle data. One set for creating a simple rigid-body model with title: "*Shuttle Ascent, Max_Q, T=55 sec (Rigid Body)*", and a second set for building a high order vehicle model with slosh and structural bending having the title: "*Shuttle Ascent, Max_Q, T=55 sec, Flex Vehicle with Slosh and TWD*". The vehicle modeling program will generate the Shuttle state-space systems from the vehicle data, which are identical with those used in the pitch axis except that the flex model is using a different sets of modal data. It uses a set of 29 modes which include both symmetric and anti-symmetric modes, and have already been pre-selected, scaled, and saved in file "*Shuttle-Stg1-T55.Inp*". The title of the modal data set is "*Flex Modes for Shuttle Ascent, Max_Q, T=55 sec*". Notice that this title must also appear at the bottom of the flex vehicle data-set. It is the last line below the line "Number of Bending Modes: 29". This file also includes two sets of actuator data. One set for modeling the Shuttle main engine dynamics and another set for the Solid Rocket Booster engine. The Flixan generated state-space systems are saved in file "*Shuttle_Stg1-T55.Qdr*". They are also converted to Matlab format by the "Export to Matlab" utility. The Matlab analysis is performed in the same folder. In this example we will not run each Flixan subprogram separately and interactively as in the pitch axis analysis. We have created a batch set that processes all the data-sets in file "*Shuttle-Stg1-T55.Inp*" together in batch mode, faster and without user interaction. The batch consists of a set of processing statements that appear on the top of the input data file. It creates the vehicle models, the actuator and sensor models, and the flight control systems. It combines them together, discretizes them, and transforms them to Matlab for further analysis. The batch set has a title: "*Batch for preparing the Shuttle Ascent, First Stage Max-Q, Coupled Axes Models*" and it is followed by some comment lines that describe the overall analysis performed in this project file "*Shuttle-Stg1-T55.Inp*". In section (2.2) we will discuss the batch execution and describe the function of each individual set of data in the input file, but before we get to that let's talk about the selection of the modal data.



The bar chart above is showing the relative mode strength for each mode and the selected modes are shown in green. In this case we select 29 flex modes (from all 3 directions) and press the "Enter" key to complete the mode selection. The mode selection program will create a smaller subset of the original modal data set and it will save it in the input data file "Shuttle-Stg1-T55.Inp", under a separate title: "Flex Modes for Shuttle Ascent, Max_Q, T=55 sec". The selected modal data-set will contain modal frequencies and mode shapes only for the selected modes and only at the locations defined in the flight vehicle model, such as, the engine gimbals, the sensors, the slosh mass locations, and the disturbance point.

Before closing, the mode selection program the user may also type-in some notes regarding the conditions and criteria applied in the mode selection process for book keeping purposes. The notes describe, for example, information regarding what kind of modes were selected and the conditions of mode selection, the excitation and measurement points, directions, etc. This information will be included as comments below the title in the selected modes set in file "Shuttle-Stg1_T55.Inp". The title of the selected modes set must also be included at the bottom of the flexible vehicle input data set (below the number of flex modes). This is how the flight vehicle modeling program will associate the flex modes with the vehicle input data.

2.3 Lateral Axes Digital Flight Control System

The lateral flight control system is defined as a combination of discrete transfer functions sampled at 40 msec. The transfer function interconnections title is: “*Lateral Axes Discrete FCS (Roll/Yaw)*” and the data are in file “*Shuttle_Stg1_T55.Inp*”. The flight control system inputs are: yaw attitude error (Ψ_e) in (radians), roll attitude error (Φ_e) in (radians), yaw rate (r), roll rate (p), in (rad/sec), and lateral acceleration (N_y) in (ft/sec^2). The outputs are roll and yaw flight control demands (DP_TVC and DR_TVC) to the thrust vector control matrix. The transfer functions interconnection program converts the data to a discrete state-space system. In this case it is executed by the batch. To run the transfer functions program manually, go to “*Analysis Tools*” then to “*Create State-Space Systems*” and then to “*Combine Transfer Functions*”. In the filename selection menu the user selects the input data and systems filenames used in this project. In the transfer functions interconnection title selection select “*Lateral Axes Discrete FCS (Roll/Yaw)*”. Run the selected set and the transfer function interconnections dialog appears showing the interconnection info. Click on “*Run*” to execute it. The program transforms the lateral FCS transfer function interconnection data into a state-space system and saves it in file “*Shuttle_Stg1_T55.Qdr*”, under the same title: “*Lateral Axes Discrete FCS (Roll/Yaw)*”. The discrete FCS is also exported to Matlab as a state-space system m-file “*lateral_fcs.m*” in directory “*\Shuttle Ascent\Coupled Vehicle*”. It will be combined with the vehicle and actuators to perform the lateral stability analysis and simulations.

2.4 Stability Analysis

This example is mainly for tutorial purposes and the stability analysis is performed using three different approaches of combining the various subsystems together. Three open-loop Simulink models were, therefore, created and used for stability analysis. They are dynamically equivalent producing very similar results but they were synthesized differently. Pitch and lateral are not separated but they are dynamically coupled. All models consist of three loops (roll, pitch, and yaw) and for stability analysis only one loop must remain open for frequency response analysis while the other two loops are closed. The Matlab file “*run.m*” loads the state-space subsystems for the vehicle, actuators, pitch and lateral flight control systems, and the TVC matrix to the Matlab workspace. They were created by the batch in section 2.1 and converted to Matlab format as m-functions. The file “*run.m*” uses these open-loop models to calculate the Nichols and Bode plots for stability analysis.

```
r2d= 180/pi; d2r=1/r2d; Ts=0.04; % Radian to degree converts
[Afls,Bfls,Cfls,Dfls]= vehicle_flex_s; % Shuttle Ascent, Max_Q, T=55 sec, Flex Vehicle with Slosh and TWD
[Aflz,Bflz,Cflz,Dflz]= vehicle_flex_z; % Discrete Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)
[Alfc,Blfc,Clfc,Dlfc]= lateral_fcs; % Lateral Flight Control System
[Apfc,Bpfc,Cpfc,Dpfc]= pitch_fcs; % Pitch Flight Control System
[Azpl,Bzpl,Czpl,Dzpl]= vehi_plant_z; % Load Discrete Plant (Vehi,Actuat,Sensors)
[Acts,Bcts,Ccts,Dcts]= actuators_5s; % System of Five Actuators
[Actz,Bctz,Cctz,Dctz]= actuators_5z; % System of Five Actuators, Z-Transform
[Asz, Bsz, Csz, Dsz] = sensors_z; % IMU, Gyro, Accelerometer Sensors, Z-transform
[Aol, Bol, Col, Dol] = open_loop; % Open-Loop System (Plant + FCS)
load TVC.Mat -ascii % Shuttle Stage-1 TVC Matrix at Max-Q

[As,Bs,Cs,Ds]= dlinmod('Sensitivity',0.04); % Wind-Gust Sensitivity Analys Model
[Ao,Bo,Co,Do]= dlinmod('Open_Flex1',0.04); % Continuous Open-Loop Simulink Flex Model-1
%[Ao,Bo,Co,Do]= dlinmod('Open_Flex2',0.04); % Continuous Open-Loop Simulink Flex Model-2

syso= SS(Ao,Bo,Co,Do,Ts); % Create Open-Loop SS System
sys= SS(As,Bs,Cs,Ds,Ts); % Create Closed-Loop SS System
w=logspace(-2.5, 2, 24000); % Define Frequ Range
figure(1); Nichols(syso,w) % Plot Nichol's Chart
figure(2); Bode(syso,w) % Plot Bode
figure(3); Sigma(sys,w) % Plot Nyquist
```

Option-1 All Subsystems Combined in Simulink

Figure (2.4.1) below shows the first open-loop analysis model where the subsystems that were created in Flexan are combined together using the Simulink model “*Open_Flex1.mdl*”. The frequency response analysis is performed using the file (run.m) in Matlab which calls the Simulink model.

Shuttle Stage-1 at Max-Q, Pitch and Lateral Coupled, Open-Loop Model-1

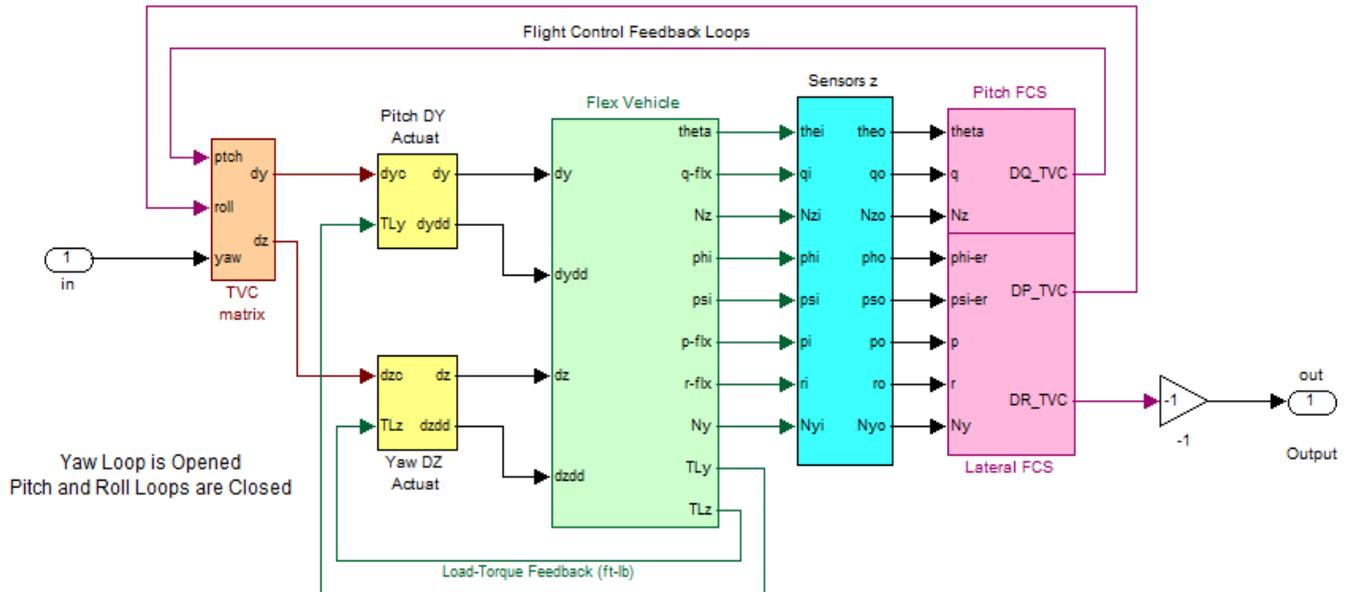
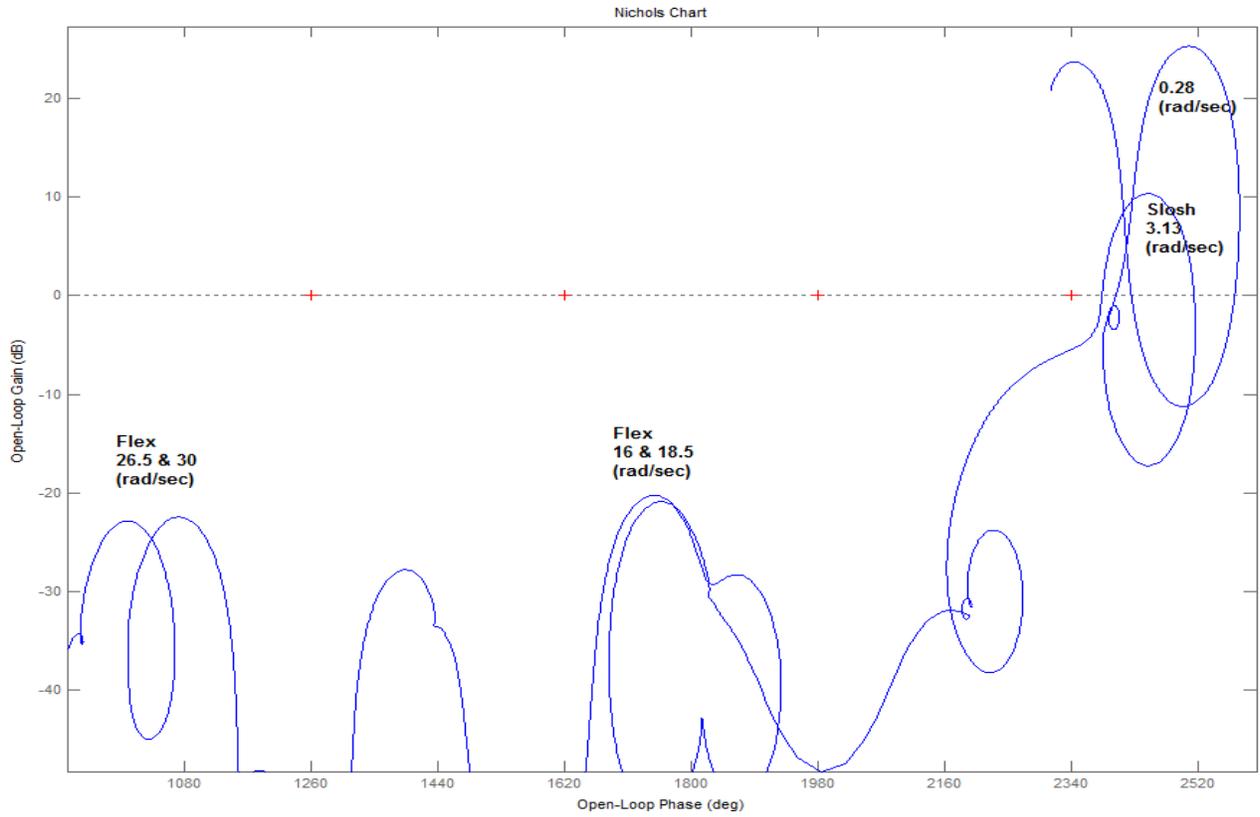


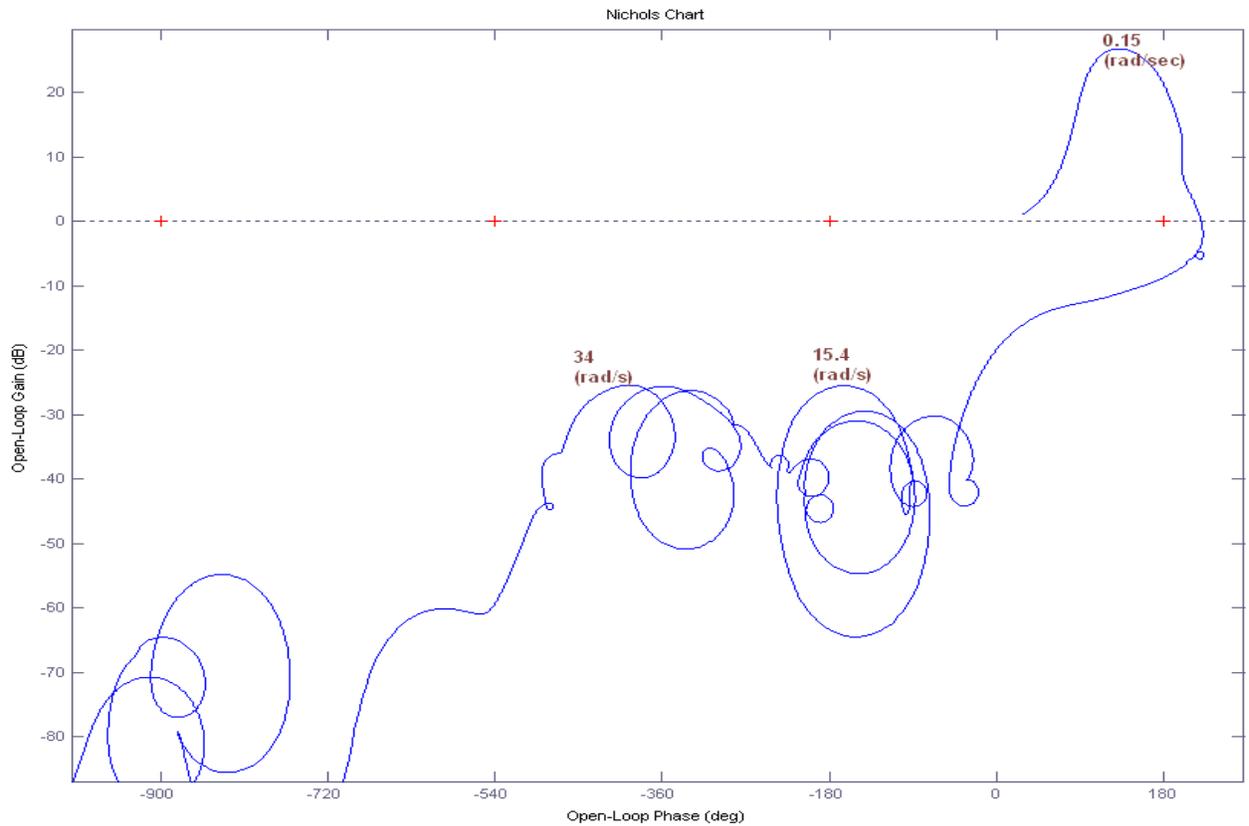
Figure 2.4.1 Simulink Model “*Open_Flex1.mdl*” for Open-Loop Analysis

The Matlab file creates the three Nichols plots, shown in Figure (2.4.2). The Simulink model in Figure (2.4.1) must be modified by opening and closing the appropriate loops before running the file run.m. For stability analysis we must open one loop at a time and close all the other loops. For yaw analysis, for example, we open the yaw loop and close the roll and pitch loops, as shown above. For roll analysis, on the other hand, we close the yaw and pitch loops and calculate the frequency response across the opened roll loop. We must include a (-1) gain in the opened loop in order to place the critical point correctly on the Nichols or the Nyquist plots (assuming positive feedback). The vehicle model is the continuous system “*Shuttle Ascent, Max_Q, T=55 sec, Flex Vehicle with Slosh and TWD*” from file “*vehicle_flex_s.m*”. The pitch and yaw actuator subsystems are “*System of Five Actuators*” loaded from file “*actuators_5s.m*”. The sensors subsystem is “*IMU, Gyro, Accelerometer Sensors, Z-transform*” loaded from file “*sensors_z.m*”. The TVC matrix is (10x3) loaded from file “*TVC.mat*” that converts the roll, pitch, and yaw flight control demands to pitch and yaw actuator deflection commands. The pitch and lateral FCS subsystems are loaded from files “*pitch_fcs.m*” and “*lateral_fcs.m*” respectively.

Pitch Axis Stability Analysis



Roll Axis Stability Analysis



Yaw Axis Stability Analysis

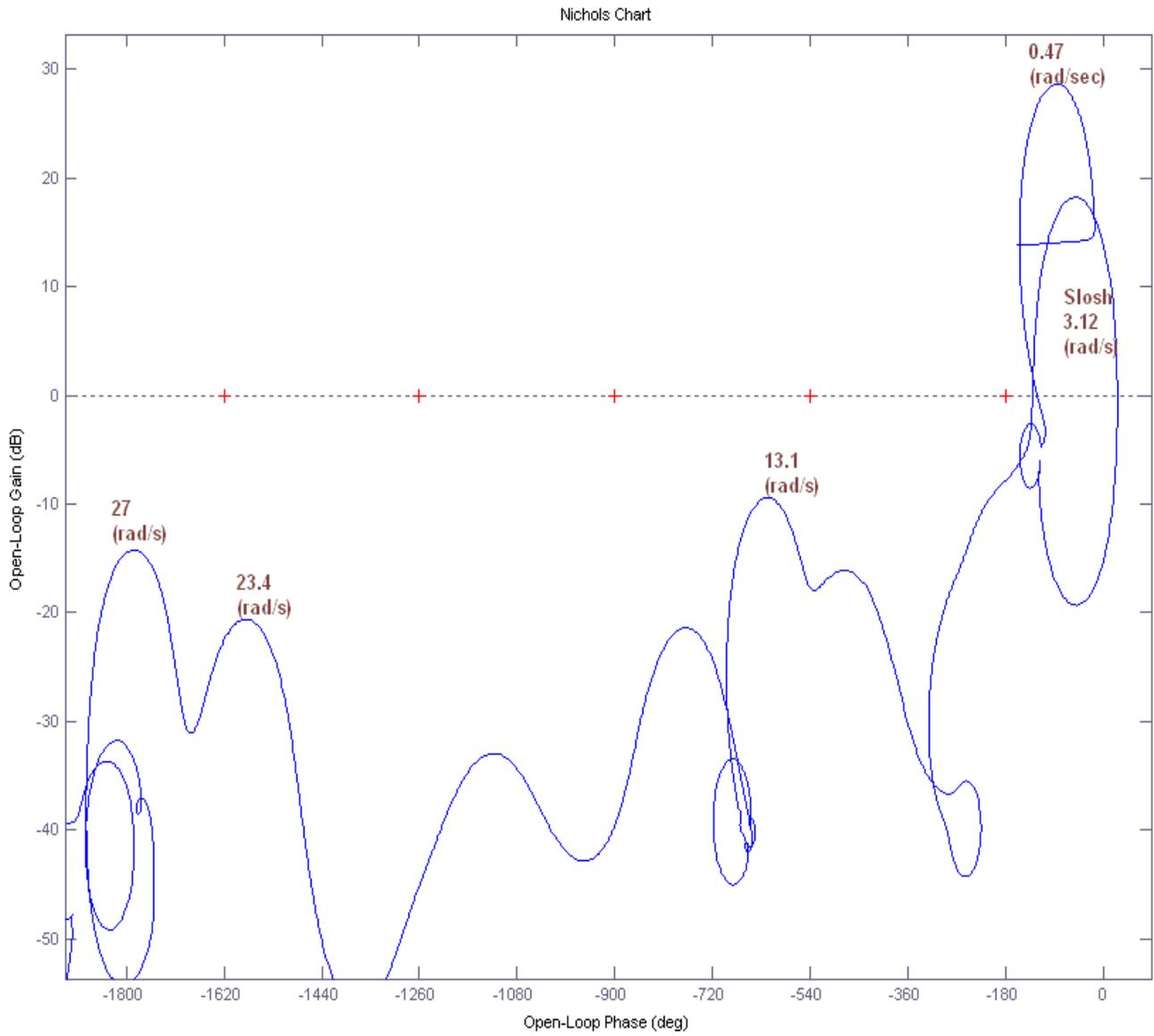


Figure (2.4.2) Nichols plots showing the stability margins in Roll, Pitch, and Yaw directions

Option-2 Plant Model Combined Using Flixan

The second open-loop configuration model uses the subsystems that were partially combined in Flixan and created the discrete plant “*Discrete Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)*”. This model is loaded from file “*vehicle_flex_z.m*” and it is now running inside the green block in the Simulink model “*Open_Flex2.mdl*”, shown in Figure (2.4.3) below with the roll loop opened. The file (run.m) is used for the frequency response analysis.

Shuttle Stage-1 at Max-Q, Pitch and Lateral together, Open-Loop Model-2

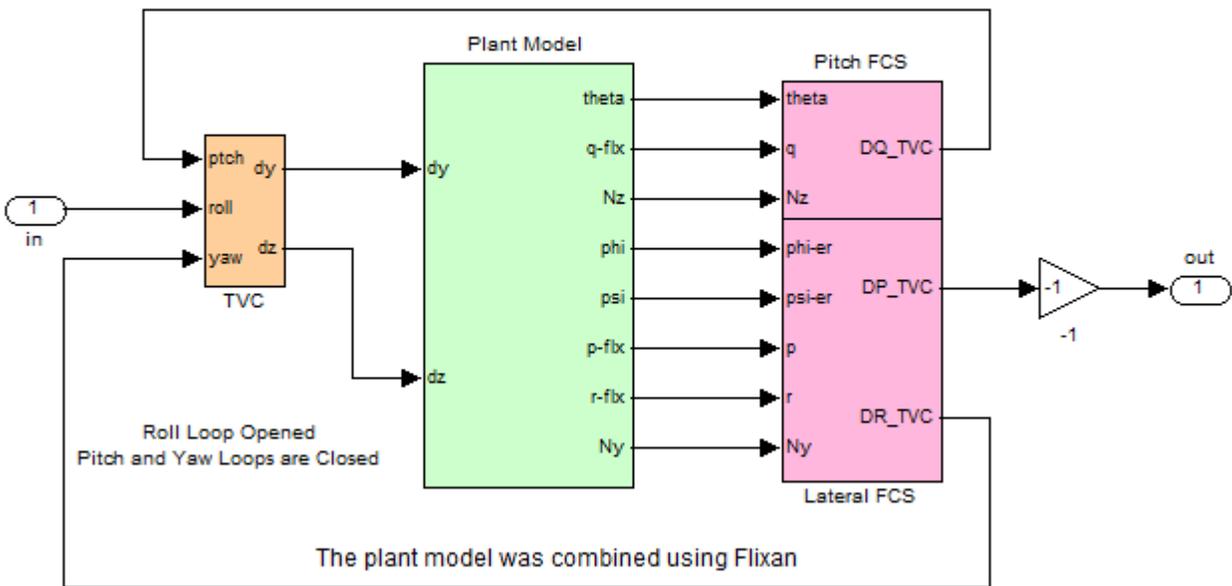


Figure 2.4.3 Simulink Model “*Open_Flex2.mdl*” for Open-Loop Analysis Using Option-2, (the plant model is combined using Flixan)

The Nichols plots in this case are very similar to the option-1 plots and they are not shown. There is a 4° difference, however, in the phase-margin. Option-1 has 4° less phase margin, possibly because Simulink has difficulty handling the 10 algebraic loops in the load-torque mechanical feedback.

The sensitivity model is used by the Matlab file “run.m” to calculate the sigma plot shown in Figure (2.5.2) below. It shows that the closed-loop system is not only sensitive to the structure modes, but that it is also sensitive to aerodynamic disturbances at 0.8 (rad/sec). This sensitivity is mainly in the lateral direction and it shows that the Space Shuttle during ascent and at high dynamic pressures is sensitive to lateral wind disturbances in the frequency range between 0.1 to 0.14 Hz. This is an important finding because the vehicle is excited by wind disturbances in this frequency range as it ascends through the various layers of atmosphere that have different wind speeds. The big vertical stabilizer increases its vulnerability to lateral winds.

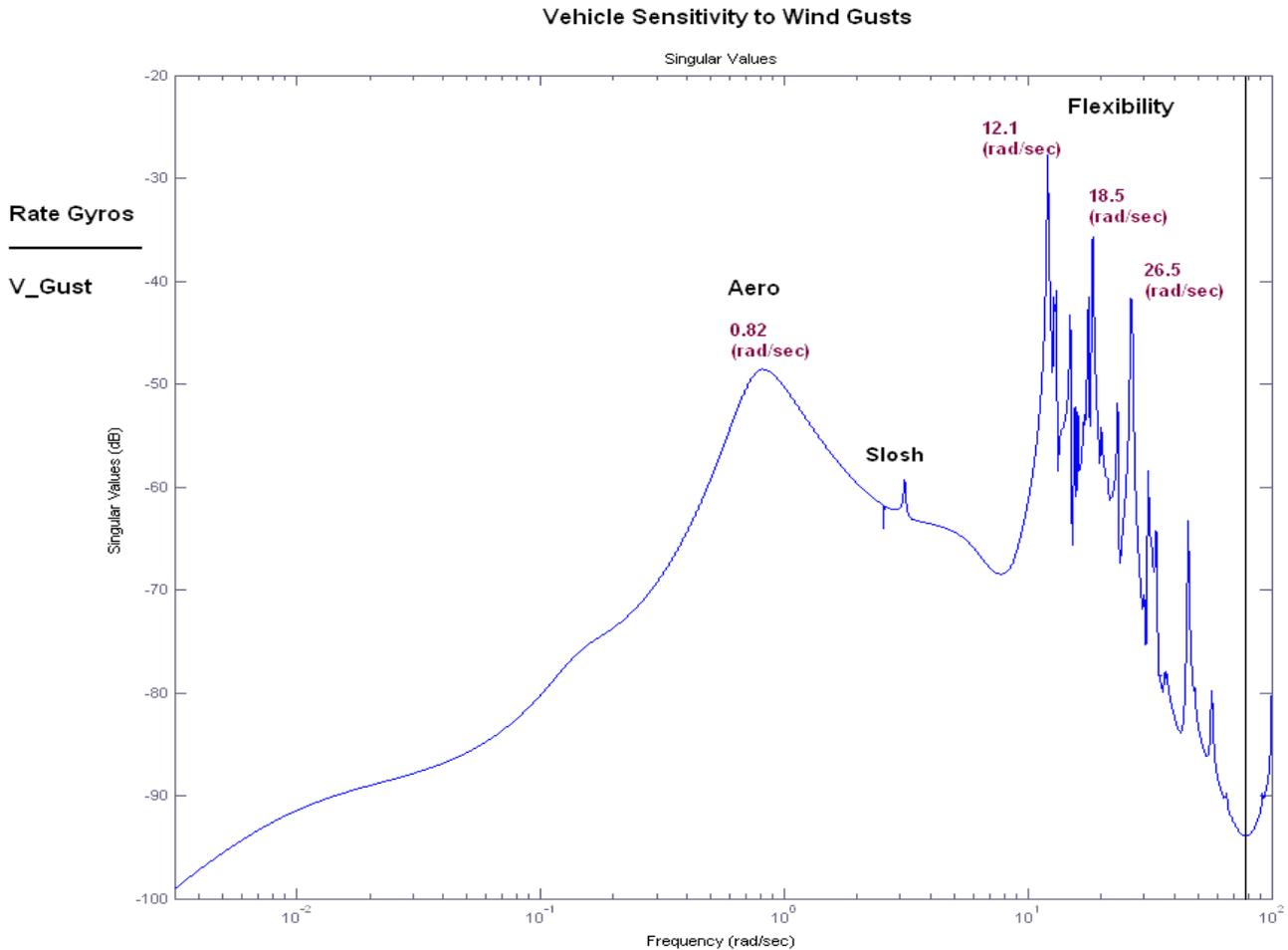


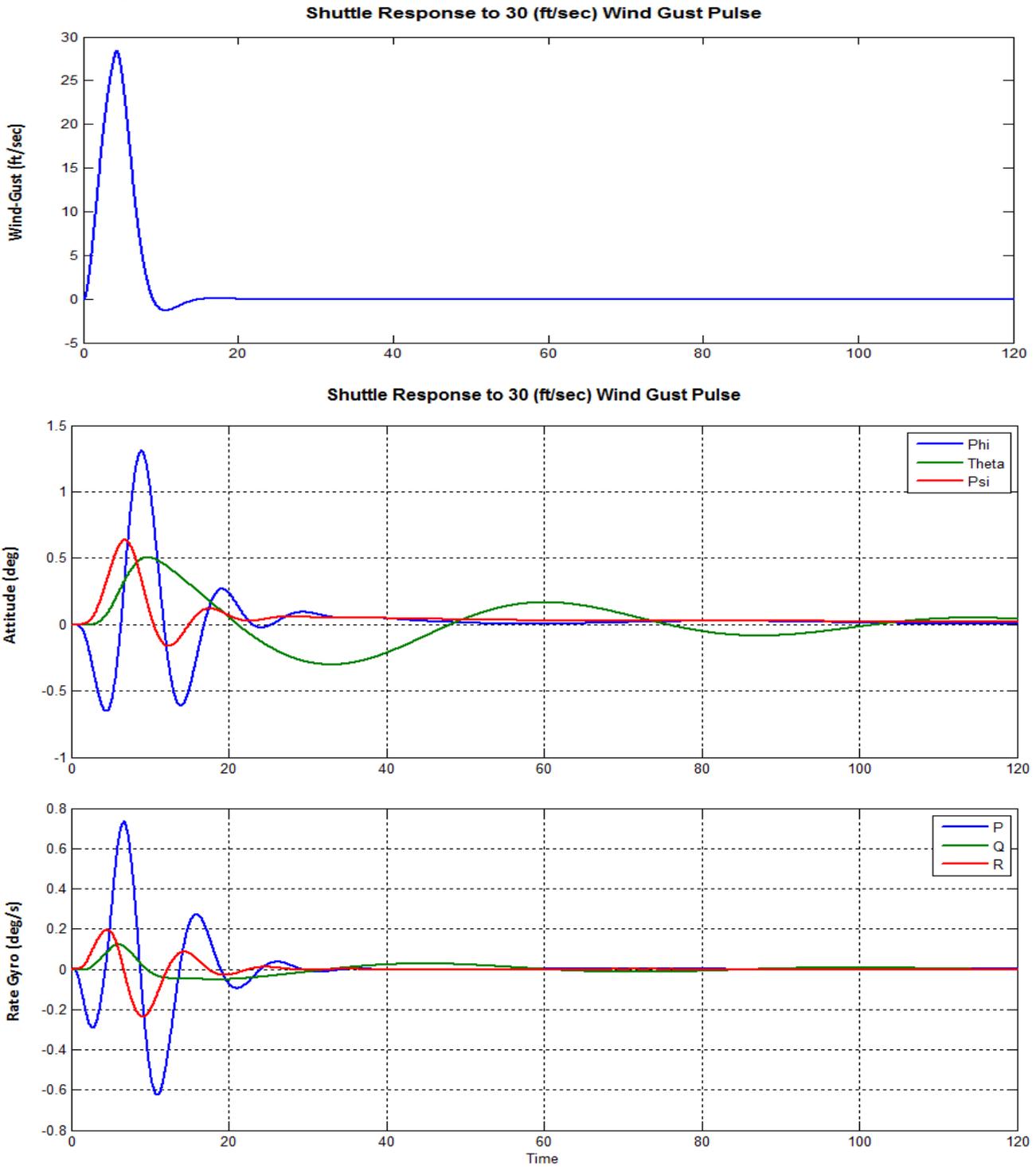
Figure 2.5.2 The Closed-Loop system is sensitive to Aero Disturbances at frequency 0.8 (rad/sec)

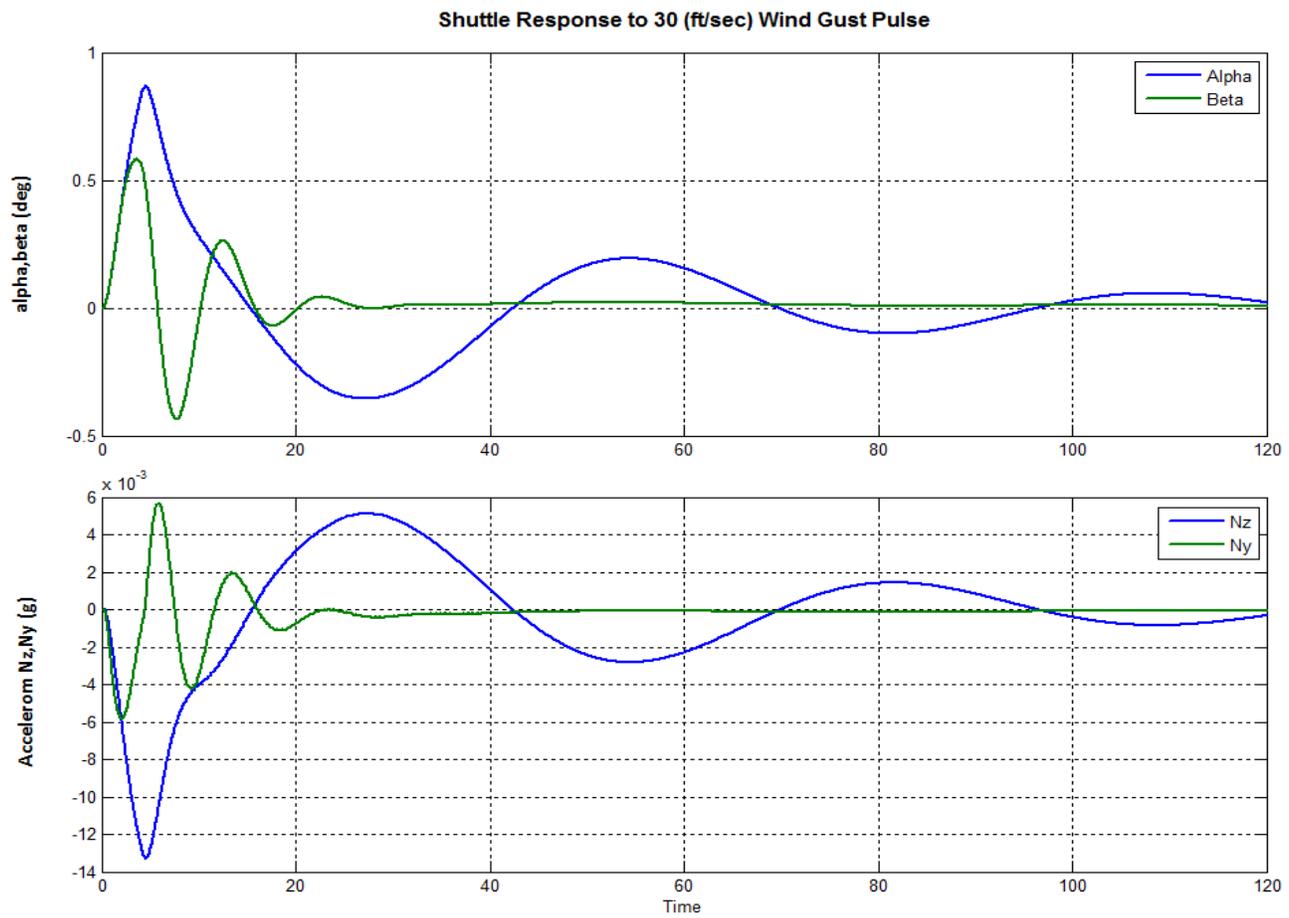
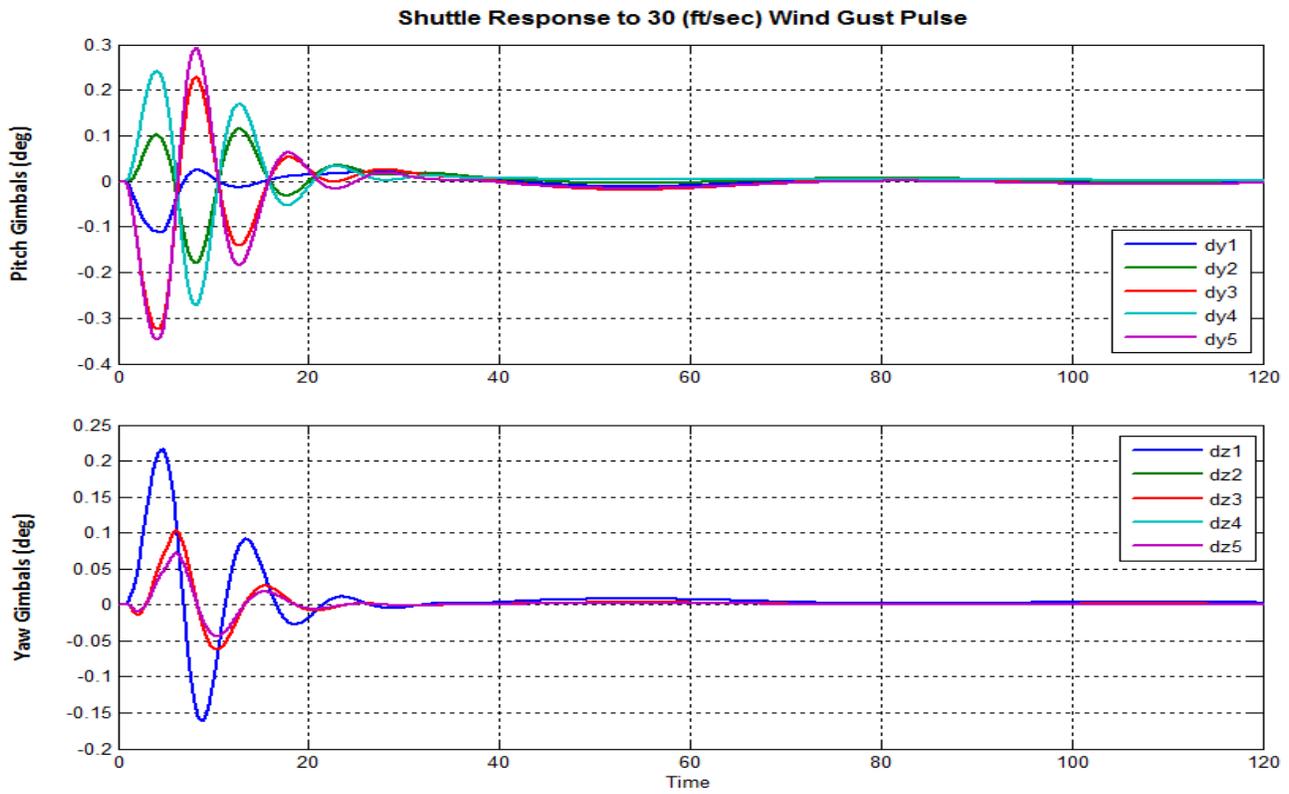
2.6 Coupled Vehicle Simulations

Similar to the stability analysis models, three Simulink models were created for simulation purposes in directory "Shuttle Ascent\Coupled Vehicle". They are dynamically equivalent, producing the same results, but they were synthesized differently. The first Simulink model is “*Sim_Flex1.mdl*”, shown in Figure (2.6.1), and it uses separate subsystems generated by Flixan and combined together in Simulink. The second implementation is “*Sim_Flex2.mdl*” consisting of a plant model (vehicle plus actuators) combined in Flixan, and the third model “*Sim_Flex3.mdl*” which is entirely combined and discretized using the Flixan program. A Matlab file “Pl.m” is used for plotting the simulation results.

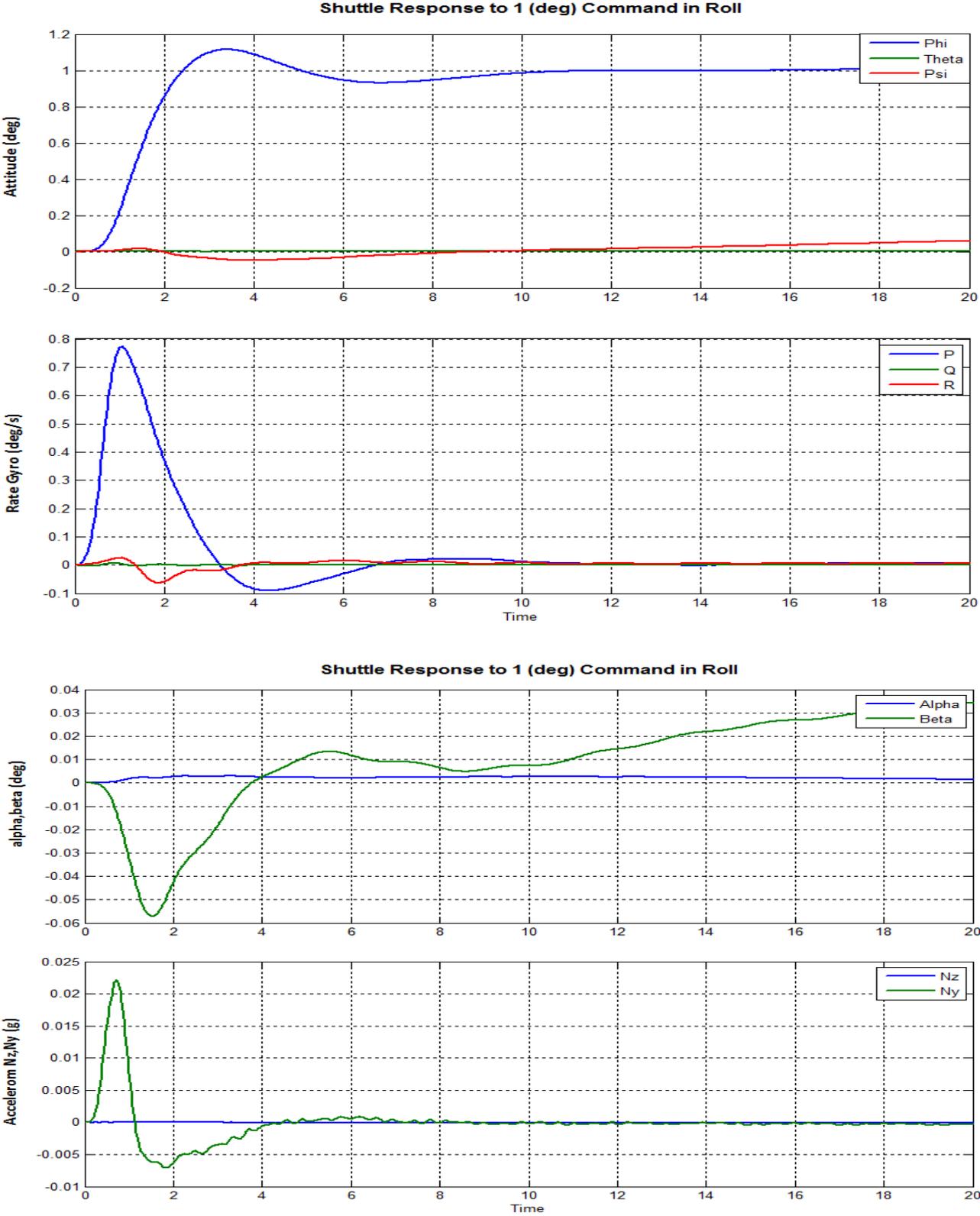
Closed-Loop Simulation Results

Three sets of simulation results were obtained from separate runs using the simulation models. In the first case, a wind-gust pulse is applied perpendicular to the vehicle x axis, at 45° between the +Y and the +Z axes. The gust causes simultaneously alpha and beta transient angles which the flight control system attempts to correct.

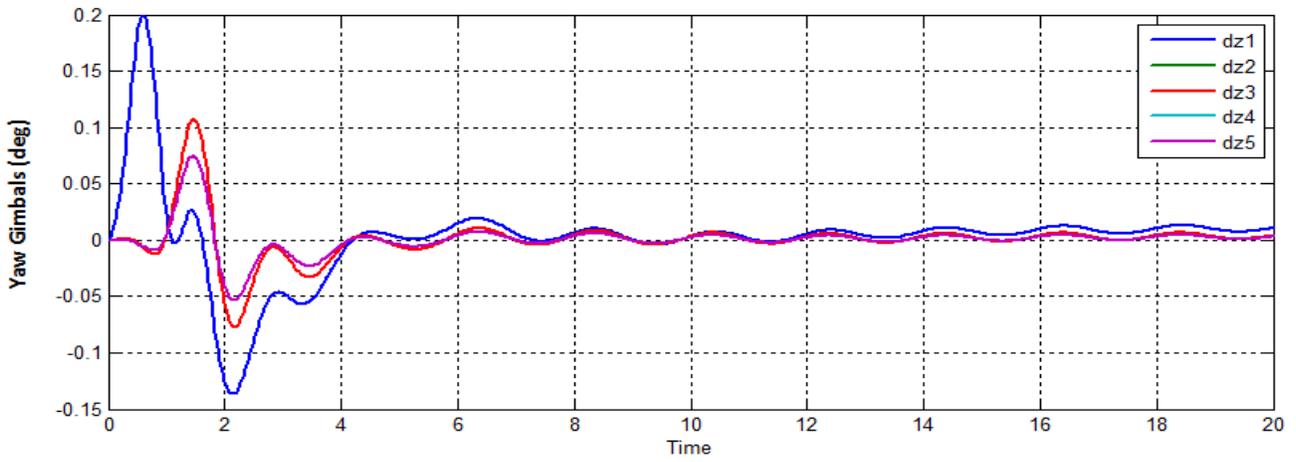
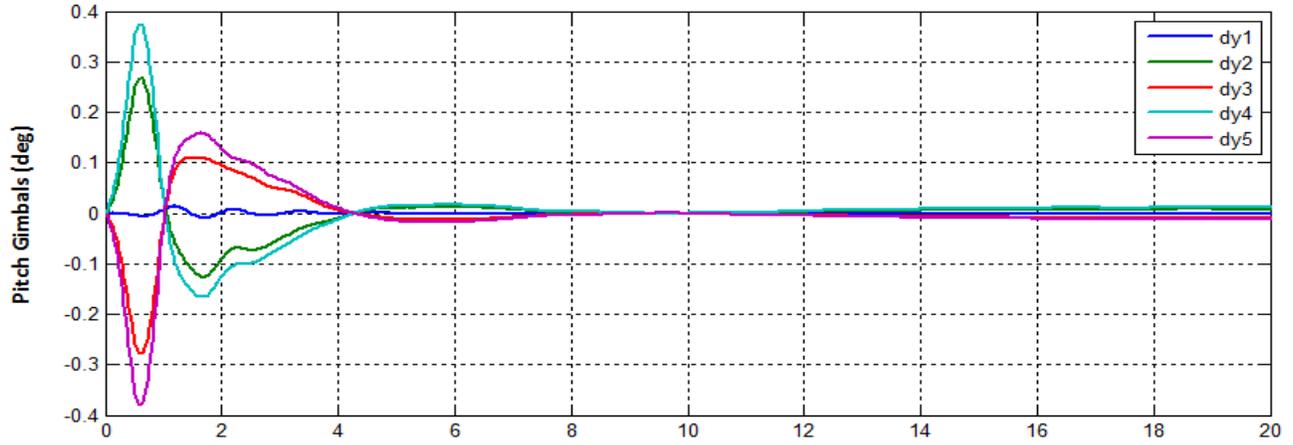




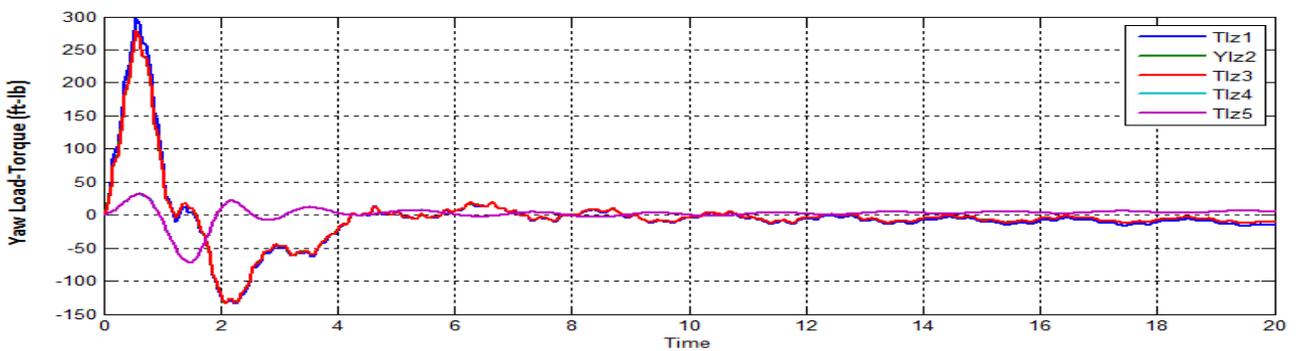
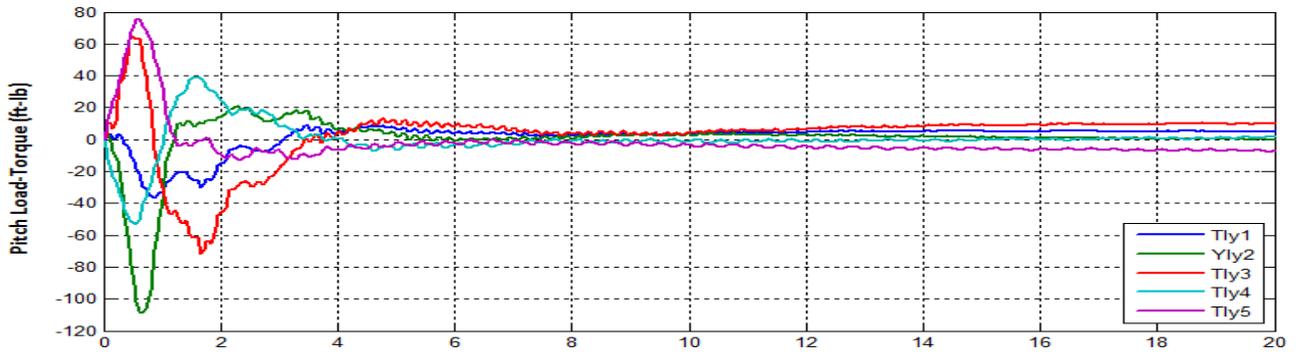
In the third simulation the vehicle is commanded to rotate 1° in roll. The left and right SSMEs and the left and right SRBs deflect differentially in pitch to generate the rolling torque on the vehicle. The roll attitude (ϕ) responds to the command after 2.5 sec. The maneuver causes transients in (β) and in the (N_y) acceleration.



Shuttle Response to 1 (deg) Command in Roll



Shuttle Response to 1 (deg) Command in Roll



3.0 Shuttle Ascent Redesign Using H-Infinity Control

The Space Shuttle flight control system was designed in the mid 70's using classical methods before the robust control design methods were available. It was designed based entirely on stability using phase and gain margins as control criteria without taking into consideration directly the system sensitivity to wind-gusts. The sensitivity analysis results show that the lateral system at high dynamic pressures is very vulnerable to wind-gust disturbances with a peak resonance occurring at frequencies around 1 (rad/sec), see Figure (3.1). This sensitivity is further aggravated by the fact that the disturbance itself is cyclic and it occurs on the average at approximately the same frequency. The reason why the disturbance is cyclic is because the vehicle is ascending almost vertically and it encounters various atmospheric layers that have different horizontal speeds. The air-speed variations occur at an average frequency of approximately 1 (rad/sec), the same frequency as the sensitivity resonance. This sensitivity is mostly due to the vertical stabilizer and it causes undesirable roll oscillations.

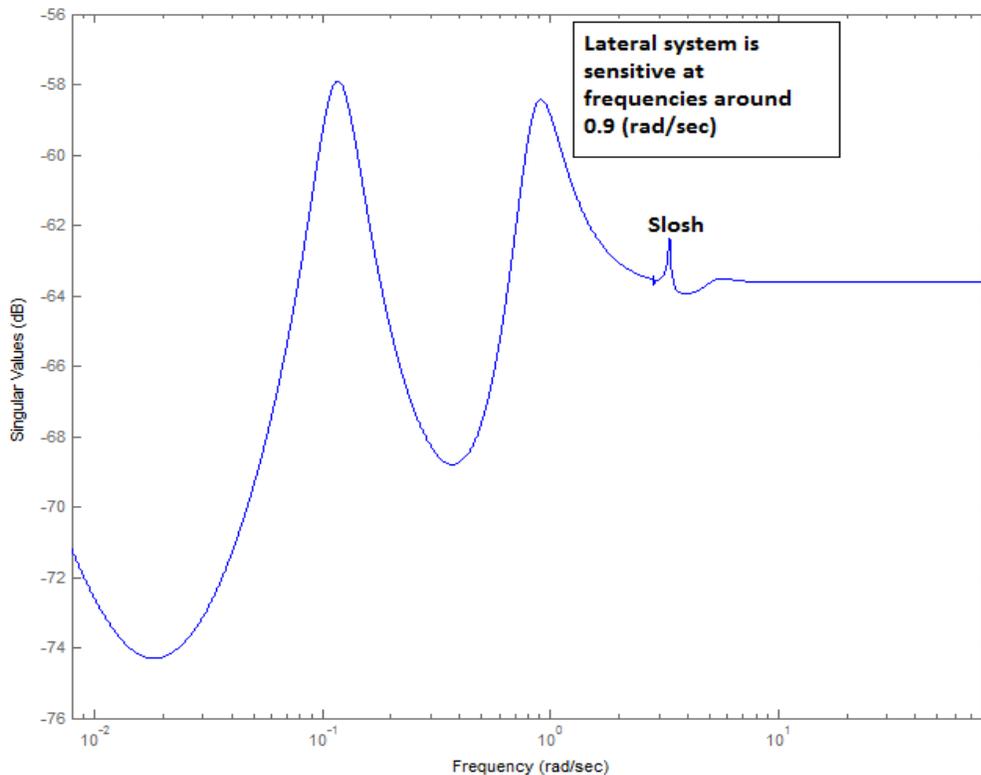


Figure (3.1) The Sensitivity Function from Gust Velocity to Alpha/ Beta response has an undesirable resonance

This classical analysis and the sensitivity function calculation for the original Shuttle control design at Max-Q is in folder "C:\Flixan\ Examples\Shuttle Ascent\Hinfinity Design\Original Shuttle FCS". The file "run.m" loads the already discretized vehicle and flight control systems and calculates the sensitivity function using the closed-loop model "Sensitivity.Mdl", in Figure (3.2). The green vehicle block consists of the system "Discrete Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)" from file "veh_i_plant_z.m" that already includes the actuators, sensors, and TWD. The FCS subsystems are loaded from files "pitch_fcs.m" and "lateral_fcs.m". In addition to the sensitivity model, a closed-loop simulation model "Simz.mdl", and an open-loop analysis model "Open_Loop.mdl" are also included in this folder.

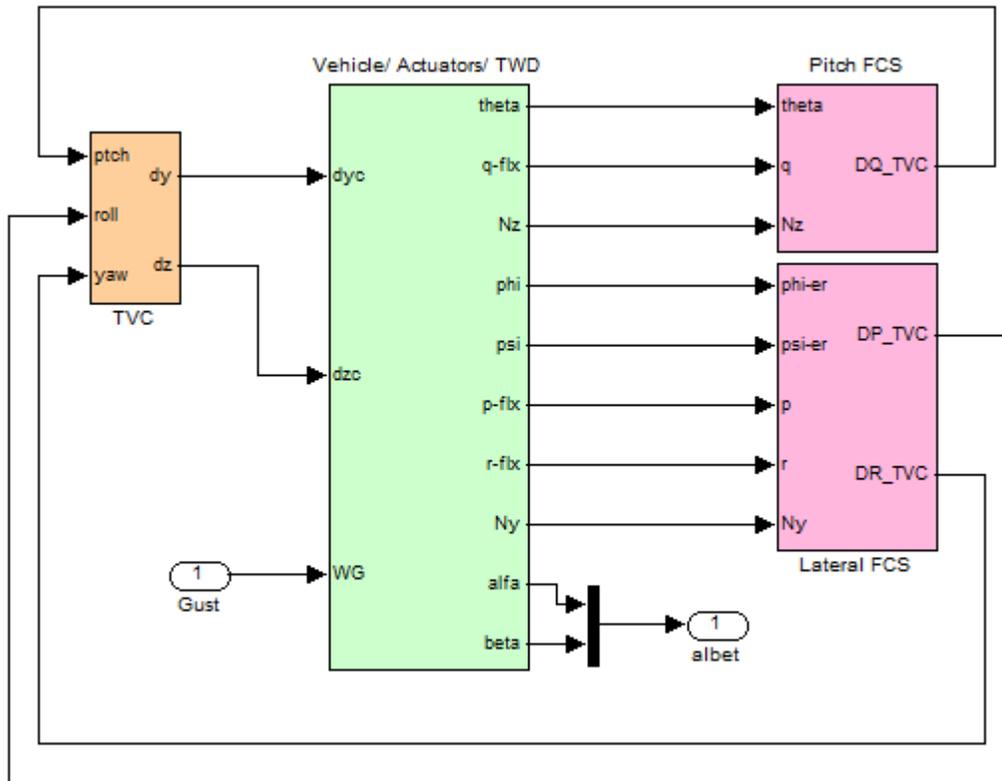


Figure (3.2) Simulink System "Sensitivity.Mdl" which calculates the Sensitivity Transfer-Function between the Gust and the Angles of Attack and Sideslip

The question arises, how would we re-design the Shuttle ascent control system if the robust control design methods were available? At high dynamic pressures the control emphasis is in minimizing the normal loading on the vehicle due to winds and the command following performance is compromised. The H-infinity method is very suitable in defining this performance versus sensitivity trade-off, because in addition to stability it also takes into consideration the transfer-function between the disturbance input and the vehicle response criteria. In this case we shall attempt to minimize the transfer function between the gust input and the angles of attack and sideslip (α & β) which is equivalent to the classical load-relief action applied in launch vehicles.

3.1 Pitch Axis H-infinity Design

At Max-Q the vehicle response to attitude commands is not a major issue and its response to pitch and yaw commands is not expected to be very efficient due to an increased load-relief, which is feedback from the accelerometers. The command response performance is sacrificed in order to gain sensitivity reduction against winds. The load-relief system causes the vehicle to turn towards the wind, as soon as the control bandwidth permits, in order to minimize the normal and lateral wind loading rather than maintaining its commanded direction. This slows-down the command performance and the original Shuttle design exhibits the same characteristic at high dynamic pressures.

The H-infinity method is based on a synthesis model (SM) of the vehicle plant, typically consisting only of the rigid-body dynamics. The SM has two sets of input vectors and two sets of output vectors, see Figure (3.3). The inputs are disturbances (w) and controls (u). The outputs are the criteria to be

optimized (z) and measurements (y) which are processed by the control system to generate the controls (u). The criteria to be minimized (z) are not necessarily measurable variables but they can be defined by equations. The H-infinity methodology synthesizes a controller $K(s)$ that closes the control loop between the measurements and the controls. This controller minimizes the infinity norm of the transfer function between the disturbance and the criteria. In this case, however, we know that the disturbance is likely to occur at around 1 (rad/sec). It is possible, therefore, to introduce the disturbance as a 2nd order resonance inside the SM, as shown in Figure (3.4). This will entice the H-infinity algorithm to provide additional control at that frequency in order to further attenuate the disturbance, as we shall see.

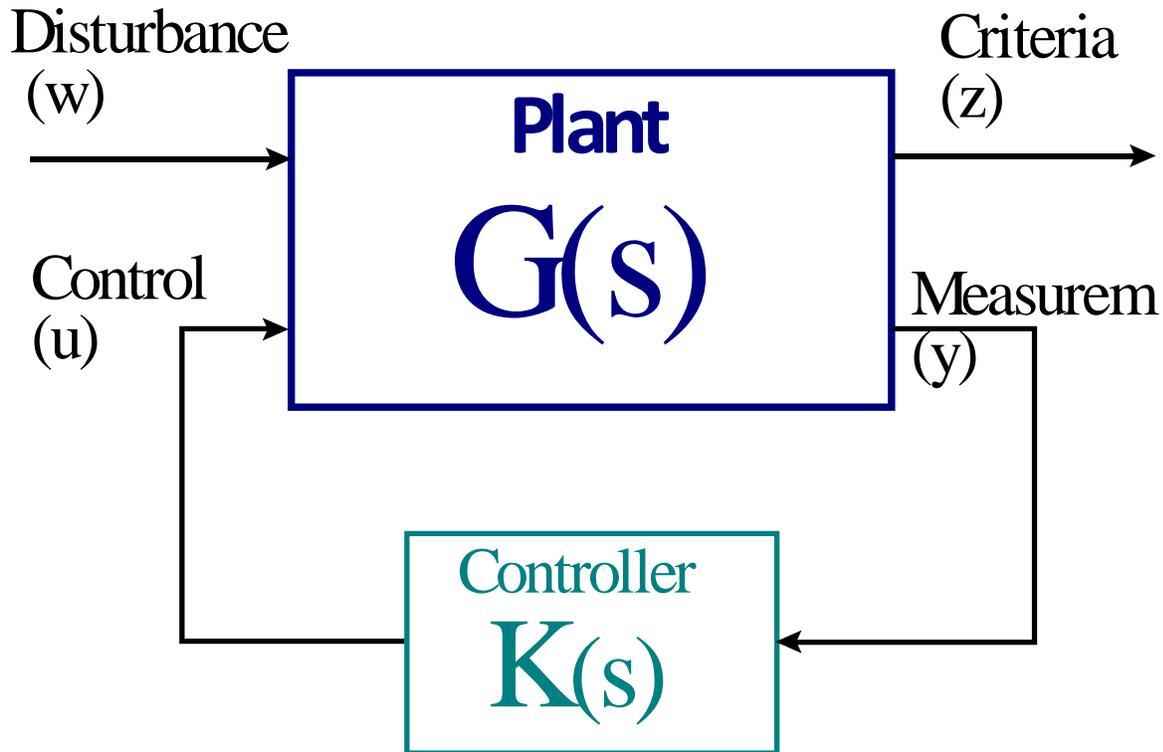
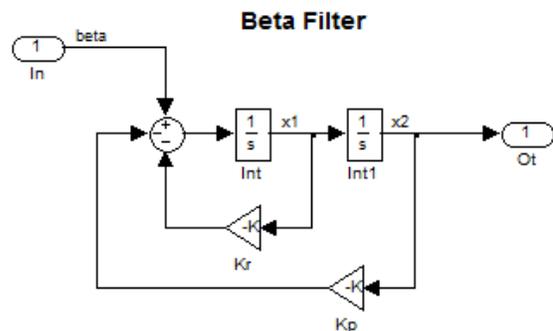
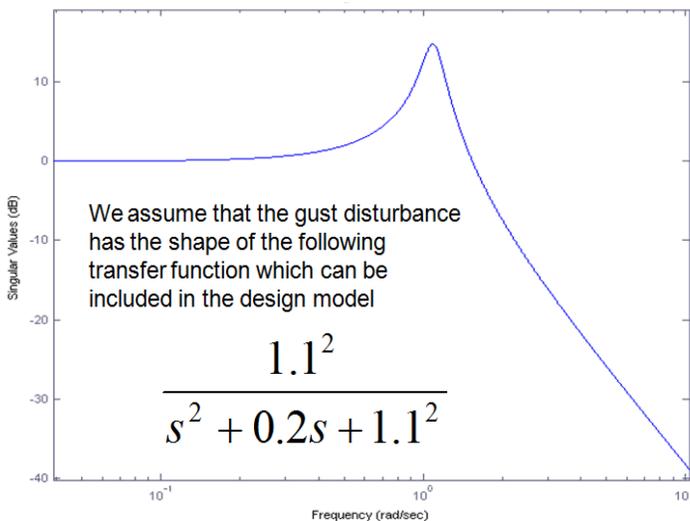


Figure (3.3) Synthesis Model $G(s)$ used in H-infinity Control Design. $K(s)$ minimizes the Infinity-Norm of the Transfer-Function between (w) and (z)

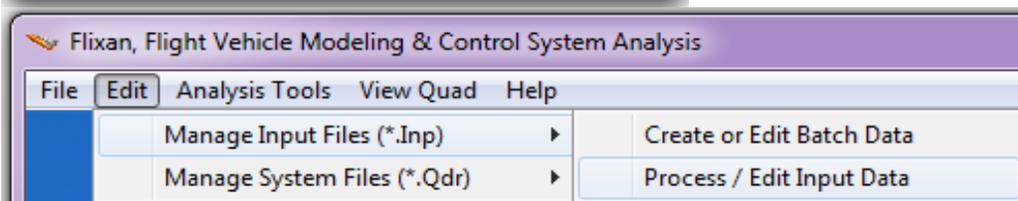
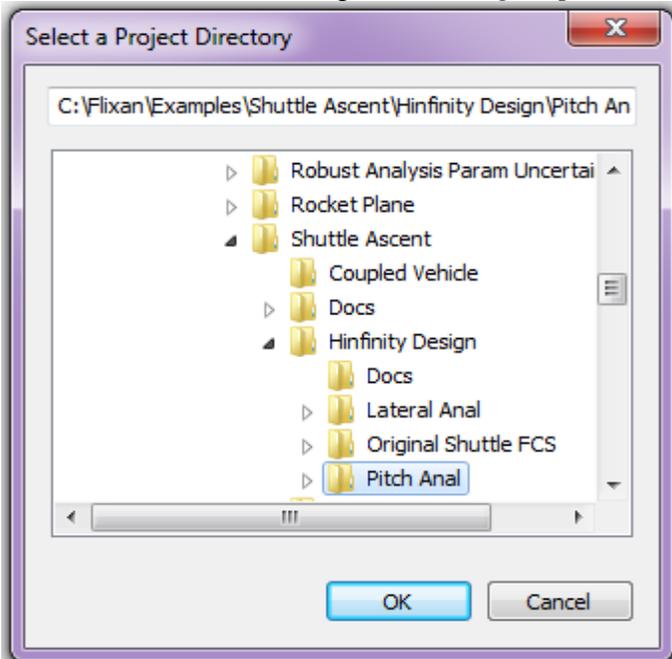


Processing the Input Data in Batch Mode

Let us first create the systems and matrices that will be used in the pitch H-infinity design and analysis. The files are in folder "C:\Flixan\Examples\Shuttle Ascent\Hinfinitiy Design\Pitch Anal". The systems are generated by Flixan in file "Pitch_MaxQ.Qdr" by processing the input data file "Pitch_MaxQ.Inp". The batch set in file "Pitch_MaxQ.Inp" is used for processing the data-sets in batch mode. Its title is "Batch for preparing the Pitch Shuttle Ascent Models at First Stage Max-Q, T=61 sec". It creates the following systems.

- A pitch design system "Shuttle Ascent, Max_Q, Pitch Hinf Design Model" in file "pitch_des.m". This system includes the TVC matrix and it is used to create the H-infinity SM. Its inputs are pitch FCS demand and wind disturbance in (ft/sec).
- A second pitch system "Shuttle Ascent, Max_Q, Pitch Analysis with Slosh & TVC" in file "pitch_anal.m". This system includes slosh and the TVC matrix. It is used for preliminary stability and sensitivity analysis using Simulink files "Anal3.Mdl" and "Sensit3.Mdl".
- A plant system "Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)" in file "plant.m". This system includes the actuator and sensor dynamics and it is combined together using Flixan. It is used for the final (complete) analysis in Simulink systems "Anal4.Mdl", "Sensit4.Mdl", and "Sim4.Mdl" which perform stability and sensitivity analysis, and time simulations.

To process the input file in batch mode using Flixan you must first select the project directory "C:\Flixan\Examples\Shuttle Ascent\Hinfinitiy Design\Pitch Anal", then go to the Flixan main menu and click on "Edit". Then go to "Manage Input Files", and then "Process/ Edit Input Data".



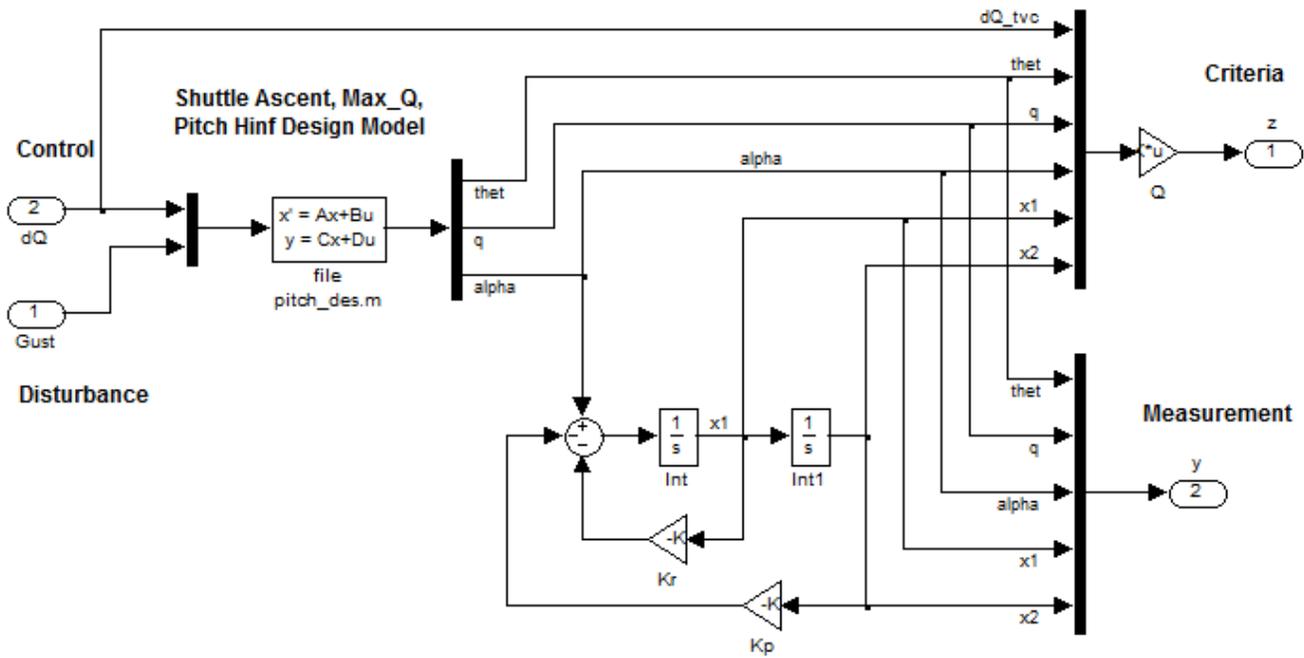


Figure (3.4) Pitch Synthesis Model captured in Simulink file "Design.Mdl"

Notice that the 1 (rad/sec) resonance is included in the SM and it is excited by the angle of attack (α) which is the key variable to minimize. It introduces two additional filter states x_1 and x_2 which become part of the measurement and also the criteria output vectors. The criteria consist of mainly α related states because the angle of attack is directly related to the normal load and this is what we desire to relieve at high Q . The criteria variables also include the pitch attitude and rate, and also the control (dQ). The criteria penalties are attached in the diagonal matrix Q . Notice that the pitch attitude (θ) is not penalized much because, as we said, the command following is compromised at high pressures. The weight on the control (dQ) prevents it from becoming too big under closed-loop control. Increasing the dQ weight in matrix Q decreases the control bandwidth. On the other hand, increasing the penalties that correspond to (α , x_1 , and x_2) it improves sensitivity but it increases bandwidth. The measurements vector consists of the five states, 3 vehicle and 2 filter. Notice that only the α -state is not measurable and it must be estimated, mainly from N_z and the gimbal deflections (δy).

The Matlab file "run.m" loads the systems and the TVC matrix and performs the design and control analysis. It uses the H-infinity method to design a (1x5) state-feedback gain K_{p3} from the 5-states vector (θ , q , α , x_1 , x_2) that appear in the measurements, although (α) is not directly measurable. The H-infinity derived state-feedback matrix K_{p3} minimizes sensitivity between the gust disturbance and the criteria defined. Having designed the state-feedback controller the next step is to analyze the (α) sensitivity to the gust disturbance. This is performed in the frequency domain by calculating the singular-value response between the gust input and the angle of attack using the closed-loop Simulink system "Sensit4.mdl" in Figure (3.5). Notice that the actual angle of attack (α) is used for output in the sensitivity calculation. In the state-feedback, however, (α) is estimated using an α -estimator block, shown in detail in Figure (3.6). Also the input to the disturbance filter (which calculates states x_1 and x_2) is α -estimate rather than real α .

```

k2d=180/pi; d2r=pi/180;
[Ad, Bd, Cd, Dd]= pitch_des;           % Pitch design plant
[An, Bn, Cn, Dn]= pitch_anal;         % Simple analysis plant
[Ap, Bp, Cp, Dp]= plant;              % Plant w actuators, sensors, TWD
load TVC.mat -ascii                   % TVC Matrix

%... Beta Estimator parameters
Mass=93215; Sref= 2690; Qbar=745.4; Cza=-0.0574;
Thr=[470000, 470000, 470000, 0.245e7, 0.245e7]';

%... Hinfinitiy Design ...
% [DQtvc, thet, q,  alfa, x1,  x2]
Q=diag([0.06, 0.005, 0.2, 0.03, 0.03, 0.03]);
om=1.0; zt=0.14; Kp=om^2; Kr=2*zt*om;
[Ad, Bd, Cd, Dd]= linmod('Design'); sys= SS(Ad,Bd,Cd,Dd);
sysd=MKTITO(sys,5,1);
[K,CL,GAM,INFO] = hinfsyn(sysd);
Kp3=INFO.KFI(:,1:5);
%Kp3 = [0.3042  3.4539  0.5105  0.0711 -0.2670];

[Ao, Bo, Co, Do]= linmod('Anal4');
[As, Bs, Cs, Ds]= linmod('Sensit4');
w=logspace(-3, 2, 10000);             % Define Frequ Range
syso= SS(Ao,Bo,Co,Do);                % Create SS System
syss= SS(As,Bs,Cs,Ds);                % Create SS System
figure(1); Nichols(syso,w)             % Plot Nichol's Chart
figure(2); Bode(syso,w)                % Plot Bode
figure(3); Sigma(syss,w)               % Plot Sensitivity

```

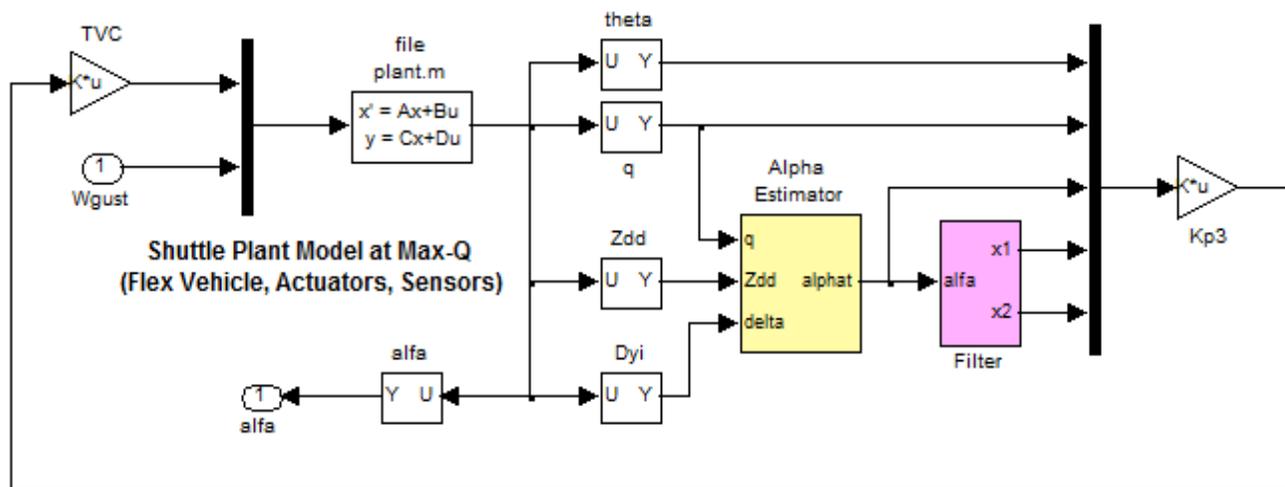


Figure (3.5) Simulink System "Sensit4.mdl" used for Sensitivity Analysis

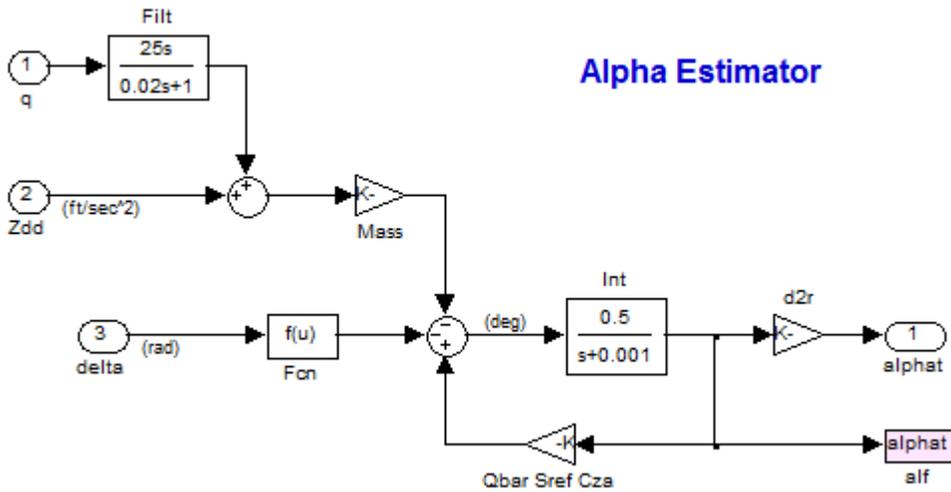


Figure (3.6) Estimator of the Angle of Attack is using measurements from the Normal-accelerometer, Gimbal Deflections, and Pitch Rate. It solves the following equation for (α) . The pitch rate derivative compensates the accelerometer measurement because it is not located at the vehicle CG.

$$M \ddot{z} = \bar{Q} S_{ref} C_{z\alpha} \alpha + \sum T_i \delta_{yi}$$

Notice that the big plant model "Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)" from file "plant.m" is used in the sensitivity analysis model "Sensit4.mdl". This system does not include the TVC matrix and it is added in the Simulink diagram (3.5). Figure (3.8) is a SV frequency response plot that shows the alpha-sensitivity of the closed-loop system to wind-gust disturbances. A similar Simulink model "Anal4.Mdl" is used for the pitch axis stability analysis, see Figure (3.7). The Bode and Nichols plots in Figure (3.9) show the pitch axis phase and gain margins.

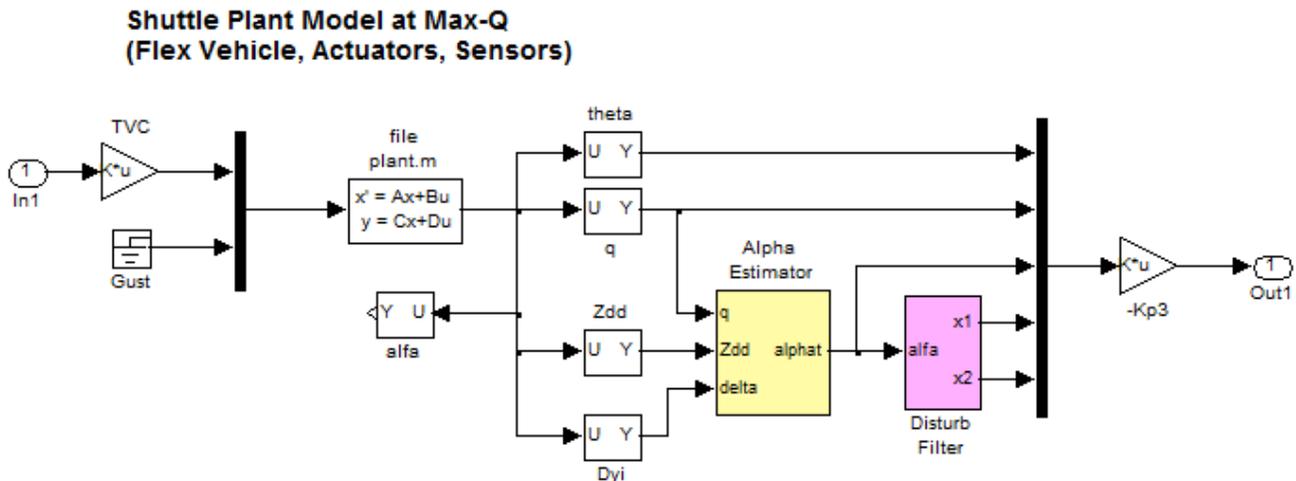


Figure (3.7) Simulink System "Anal4.mdl" used for Pitch Stability Analysis

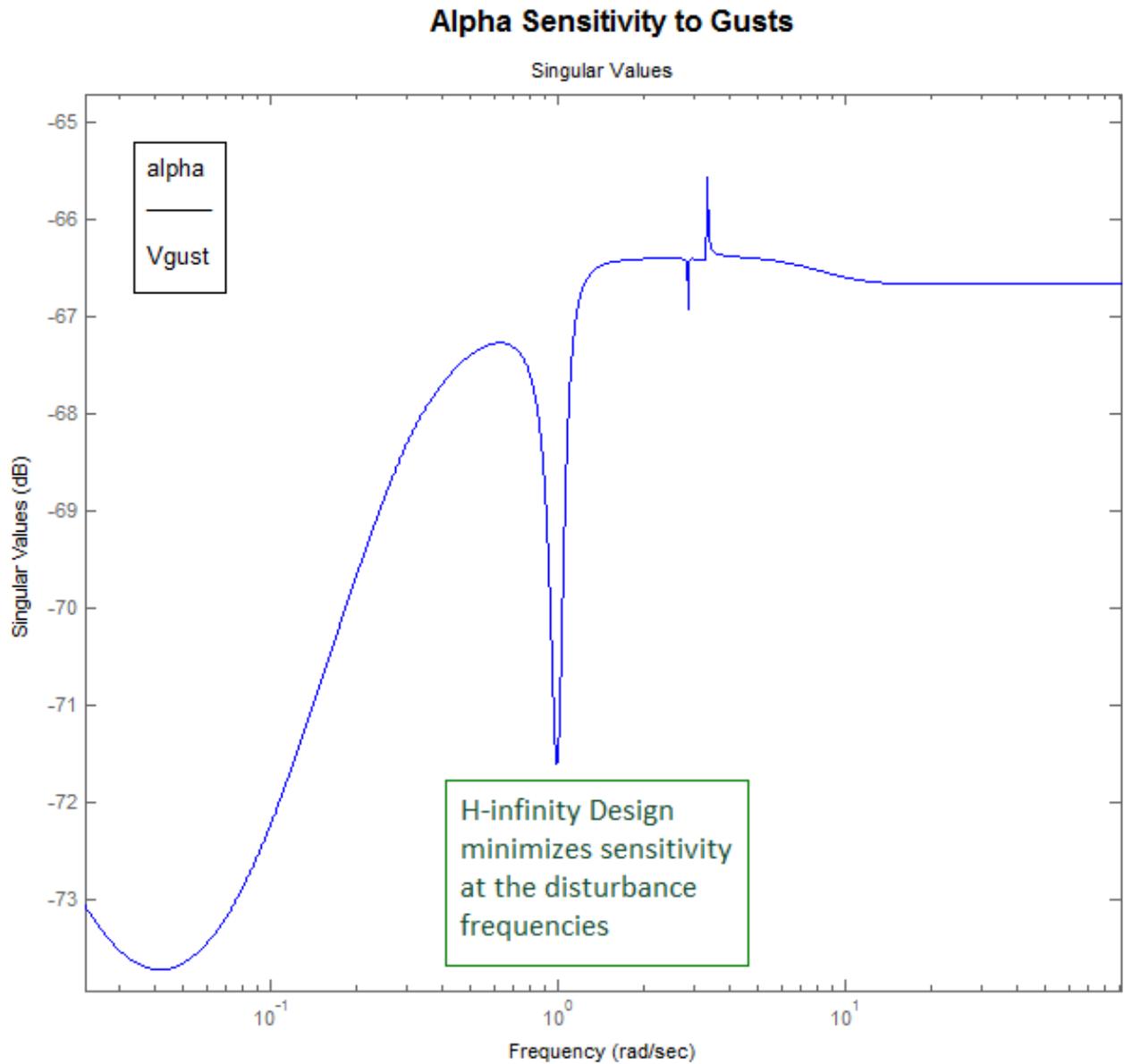


Figure (3.8) Sensitivity function between the gust velocity disturbance and the angle of attack. It shows how the disturbance filter provides additional attenuation at the expected disturbance frequency which is around 1 (rad/sec).

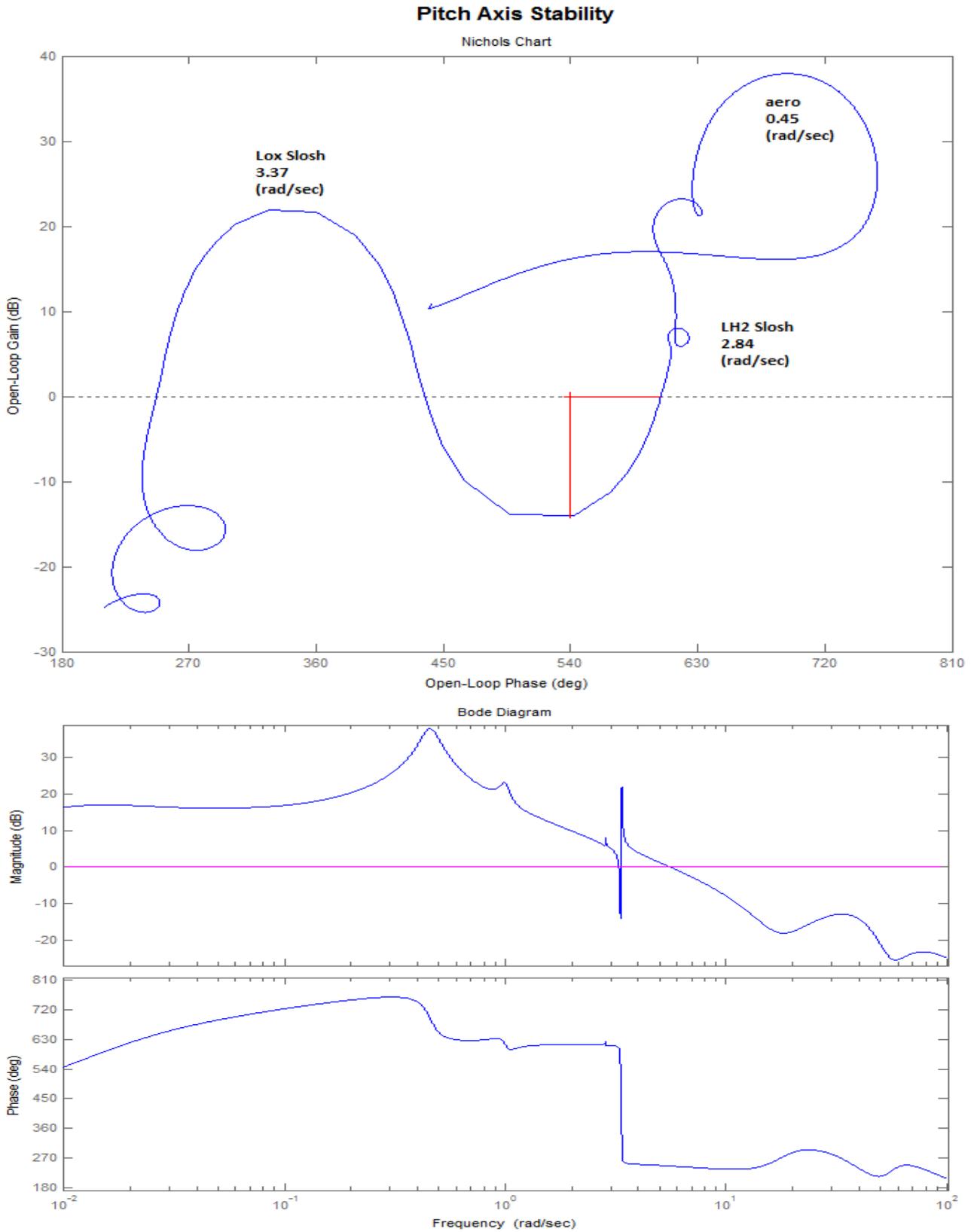


Figure (3.9) Open-Loop Nichols and Bode plots for the pitch axis system showing the phase and gain margins

3.2 Lateral Axes H-infinity Design

A very similar approach is used in the lateral flight control system design where the original Shuttle design demonstrated poor sideslip sensitivity to gust disturbances. The control system architecture is shown in Figure (3.10). The flight control system, in addition to attitude step commands, it receives also a load-relief feedback signal from the sideslip angle (β) which is estimated from the lateral accelerometer (N_y).

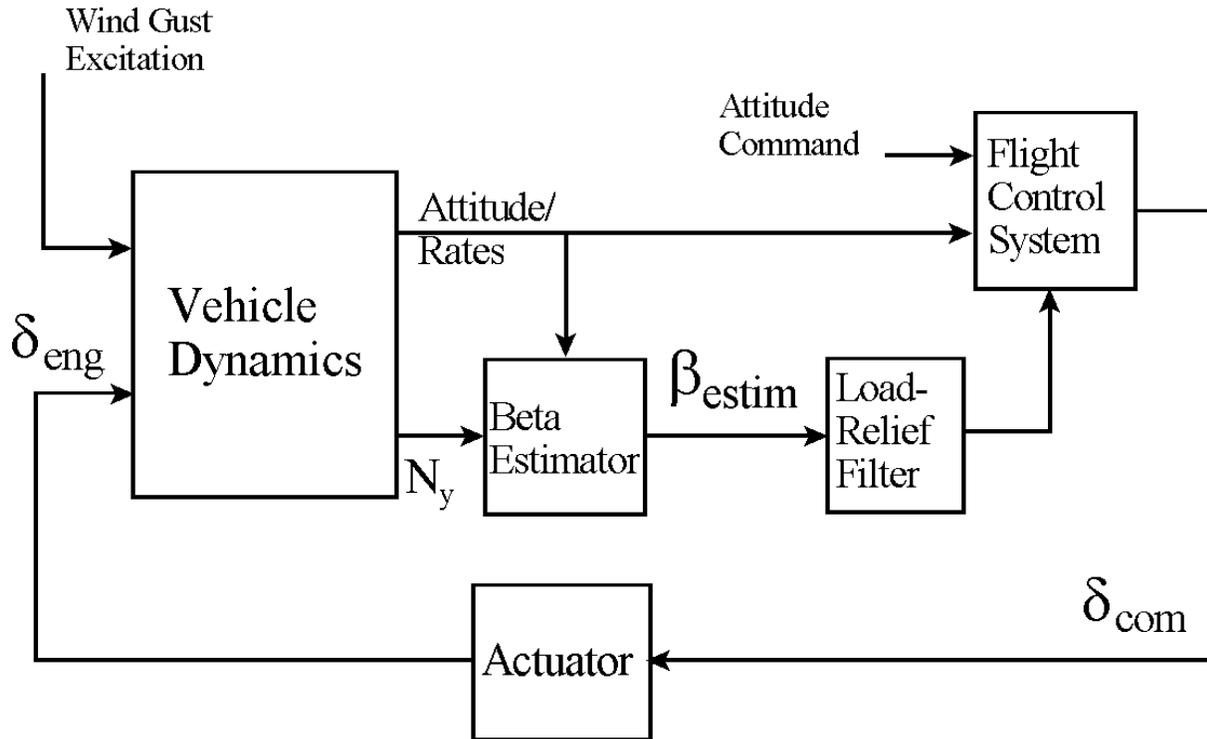


Figure (3.10) Lateral Control System Architecture has a load-relief feedback via a beta-estimator.

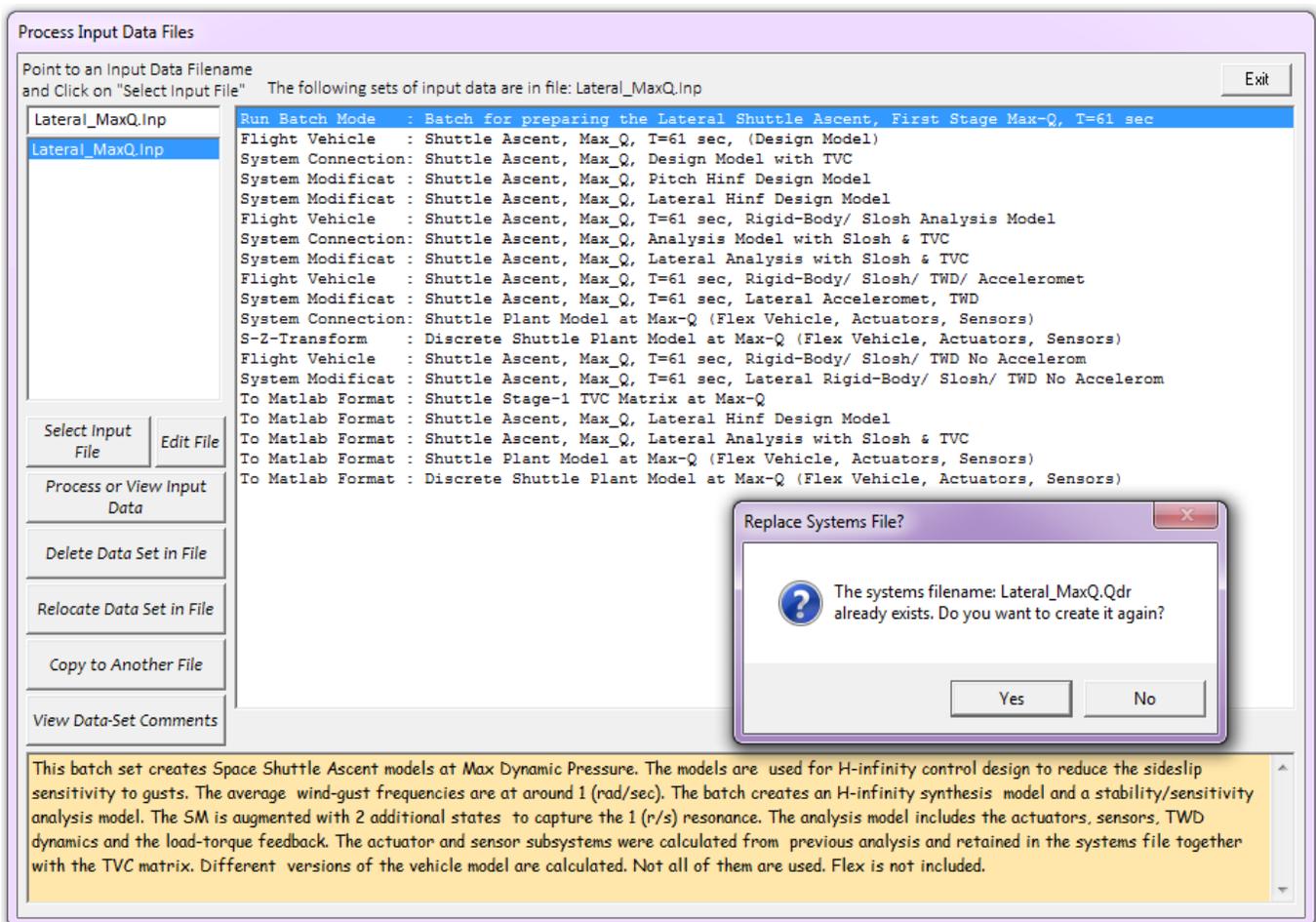
Processing the Input Data in Batch Mode

The analysis is similar to the pitch axis. The analysis files are in folder "C:\Flixan\Examples\Shuttle Ascent\Hinfinitiy Design\ Lateral Anal". The input file that creates the systems is "Lateral_MaxQ.Inp". This file is processed by Flixan and the systems generated are saved in "Lateral_MaxQ.Qdr". There is a batch set on the top of file "Lateral_MaxQ.Inp" that processes the data-sets in batch mode. Its title is "Batch for preparing the Lateral Shuttle Ascent Models at First Stage Max-Q, T=61 sec". It creates the following systems.

- A lateral design system "Shuttle Ascent, Max_Q, Lateral Hinf Design Model", saved in file "later_des.m". This system includes the TVC matrix and it is used to create the H-infinity synthesis model "design.mdl". Its inputs are roll and yaw FCS demands, and wind disturbance in (ft/sec).
- A second lateral system "Shuttle Ascent, Max_Q, Lateral Analysis with Slosh & TVC", saved in file "later_anal.m". This system includes slosh and also the TVC matrix. It is used for preliminary stability and sensitivity analysis using the Simulink files "Anal2.Mdl" and "Sensitiv2.Mdl".

- A plant system "Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)" in file "plant.m". This system includes slosh and the actuator/ sensor dynamics. It is combined together using Flixan, and used for the final (complete) analysis in the Simulink systems "Anal4.Mdl", "Anal5.Mdl", "Sensitiv4.Mdl", and "Sim4.Mdl" which perform stability and sensitivity analysis, and time simulations.

To process the input file in batch mode using Flixan you must first select the project directory "C:\Flixan\Examples\Shuttle Ascent\Hinfinity Design\Lateral Anal", then go to the Flixan main menu and click on "Edit". Then go to "Manage Input Files", and then "Process/ Edit Input Data". This will open the utility dialog for processing input data, as shown below. The menu in the upper left hand side of the dialog displays the input data files (having an extension .inp) existing in the project directory. In this case there is only one input data file "Lateral_MaxQ.Inp" and we select it by clicking on "Select Input File". The menu on the right side shows the data-sets which are included in file "Lateral_MaxQ.Inp". It shows the utility program that will be used for processing each set followed by the title of the data set in one line. Select the batch set on the top to process the data in batch mode and click on "Process or View Input Data". Flixan will process the input file and create the systems in file "Lateral_MaxQ.Qdr". It will also create the Matlab system files. Answer "Yes" in the next dialog to overwrite the previous systems data. However, some of the data, such as: TVC matrix, actuators, and sensors, will be preserved in the systems file "Lateral_MaxQ.Qdr" after processing, because "Retain" statements are included in the batch.



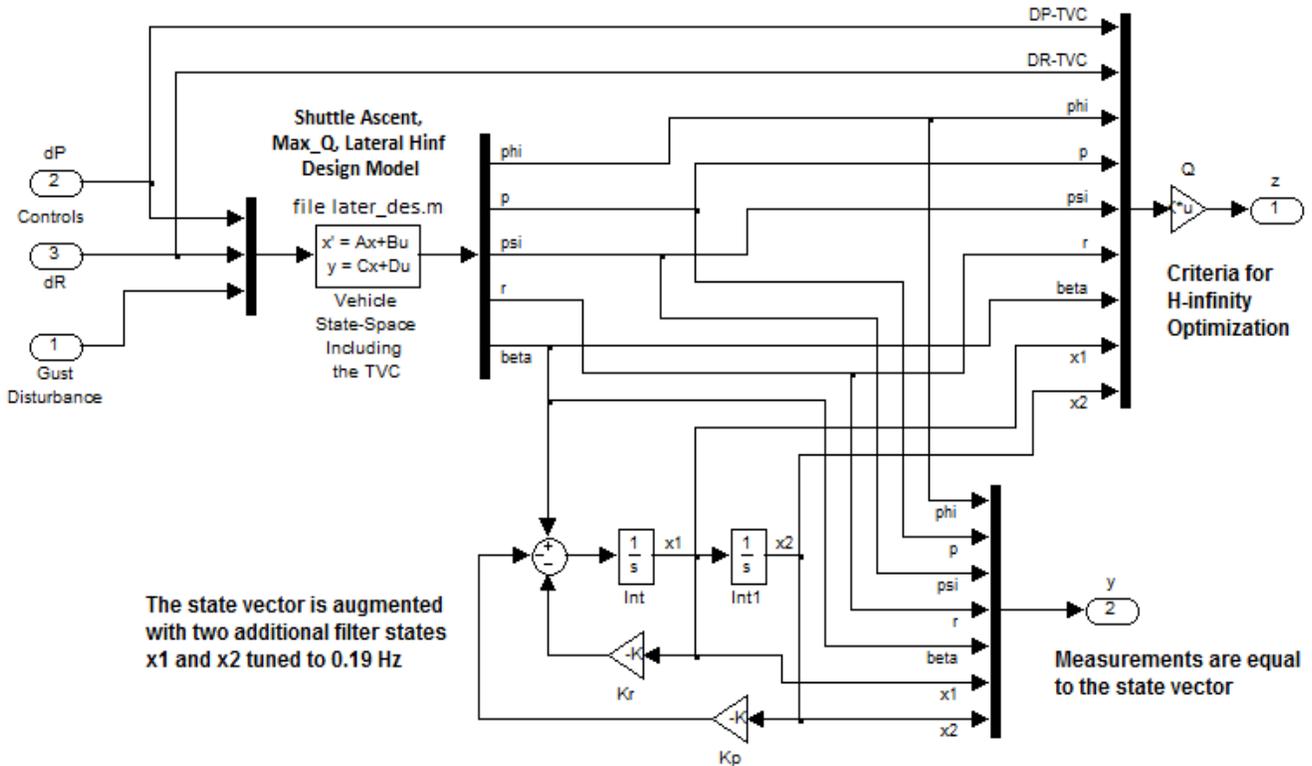


Figure (3.11) Lateral H-infinity Synthesis Model included in Simulink file "Design.Mdl"

Figure (3.11) shows the H-infinity synthesis model for the lateral axes. It uses the Flixan generated vehicle system "Shuttle Ascent, Max_Q, Lateral Hinf Design Model" from file "later_des.m". The TVC matrix is already included in this system and the control inputs are roll and yaw demands (dP , dR). The disturbance input (w) is the wind-gust velocity in (ft/sec). The wind-gust direction relative to the vehicle body is defined in the vehicle data to excite both pitch and lateral axes. The 1 (rad/sec) disturbance resonance is included in the SM and it is excited by the angle of sideslip (β) which is the key variable to minimize by the H-infinity design. It introduces two additional resonance states x_1 and x_2 which become part of the measurement and also the criteria output vectors. The criteria consist of mainly beta related states because the angle of sideslip is directly related to the lateral load and this is what we desire to relieve at high Q. The criteria variables also include the roll and yaw attitudes and rates, and also the controls (dP , dR). The criteria penalties are attached in the diagonal matrix Q. Notice that the attitudes (ϕ , ψ) are not penalized much because command following is compromised at high pressures. The control penalties prevent them from becoming too big under closed-loop control. Increasing the Q elements that correspond to the controls (dP , dR) decreases the control bandwidth. On the other hand, increasing the penalties that correspond to (β , x_1 , and x_2) it improves sensitivity but it increases the control bandwidth. The measurements vector consists of the 7 states, 5 vehicle and 2 filter states. Notice that only the β -state is not directly measurable and it must be estimated, mainly from N_y and the gimbal deflections (δz).

The Matlab file "run.m" loads the systems and the TVC matrix and performs the design and control analysis. It uses the H-infinity method to design a (2×7) state-feedback gain K_{I3} from the 7-states vector (ϕ , p , ψ , r , β , x_1 , x_2) that appear in the measurements, although (β) is not directly measurable. The H-infinity derived state-feedback matrix K_{I3} minimizes sensitivity between the gust disturbance

and the criteria defined. Having designed the state-feedback controller the next step is to analyze the (β) sensitivity to the gust disturbance. This is performed in the frequency domain by calculating the singular-value frequency response between the gust input and the angle of sideslip using the closed-loop Simulink system "Sensitiv4.mdl" shown in Figure (3.12). Notice that the actual angle of sideslip (β) is used for output in the sensitivity calculation. In the state-feedback, however, (β) is estimated using an β -estimator block, shown in detail in Figure (3.14). Also the input to the disturbance filter (which calculates states x_1 and x_2) is β -estimate rather than real β .

```

r2d=180/pi; d2r=pi/180;
om=1.2; zt=0.14; Kp=om^2; Kr=2*zt*om; %frq; % Disturb Filter Characteristics
[Ad, Bd, Cd, Dd]= later_des; % Shuttle Ascent, Max_Q, Lateral Hinf Design Model
[An, Bn, Cn, Dn]= later_anal; % Shuttle Ascent, Max_Q, Lateral Analysis with Slosh & TVC
[Ap, Bp, Cp, Dp]= plant; % Shuttle Plant Model at Max-Q (Flex Vehicle, Actuators, Sensors)
load TVC.mat -ascii % Shuttle Stage-1 TVC Matrix at Max-Q
load orig.mat % from Classical Design

%... Alpha/Beta Estimator parameters
Mass=93215; Sref= 2690; Qbar=745.4; Cyb=-0.0353; Cza=-0.0574;
Thr=[470000, 470000, 470000, 0.245e7, 0.245e7]';

%... Hinfinity Design ...
% [DPtvc,DRtvc, phi, p, psi, r, beta, x1, x2]
Q=diag([0.09, 0.12, 0.001, 0.1, 0.001, 0.1, 0.24, 0.13, 0.07]);
[Ad, Bd, Cd, Dd]= linmod('design');
sys = SS(Ad,Bd,Cd,Dd);
sysd=MKTITO(sys,7,2);
[K,CL,GAM,INFO] = hinfsyn(sysd,7,2);
K13=INFO.KFI(:,1:7);
Kp3 = [0.3042 3.4539 0.5105 0.0711 -0.2670];

% [Ao, Bo, Co, Do]= linmod('Anal2');
% [As, Bs, Cs, Ds]= linmod('Sensitiv2');
[Ao, Bo, Co, Do]= linmod('Anal5');
[As, Bs, Cs, Ds]= linmod('Sensitiv4');
%w=logspace(-3, 2.5, 10000); % Define Frequ Range
syso= SS(Ao,Bo,Co,Do); % Create SS System
sys2= SS(As,Bs,Cs,Ds); % Create SS System
figure(1); Nichols(syso,w) % Plot Nichol's Chart
figure(2); Bode(syso,w) % Plot Bode
sig2=Sigma(sys2,w); % SV Bode
figure(3); loglog(w,sig1,'r',w,sig2,'b') % Plot SV Bode

```

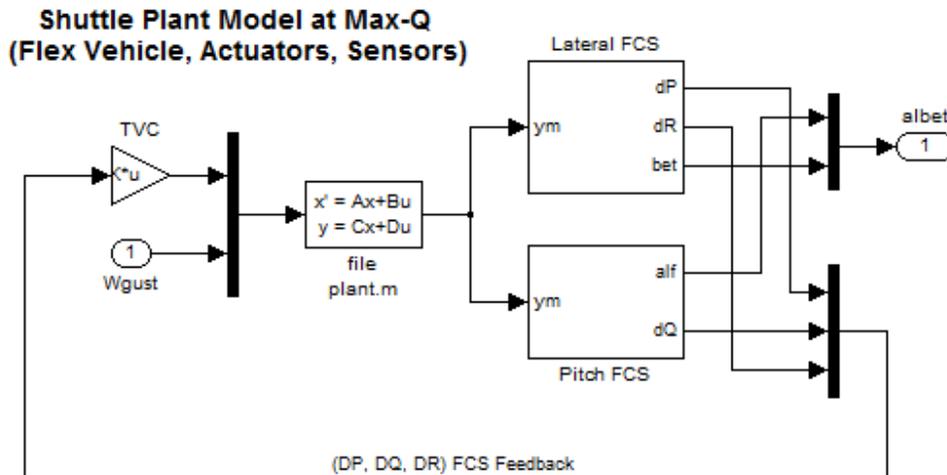


Figure (3.12a) Simulink System "Sensitiv4.mdl" used for Sensitivity Analysis

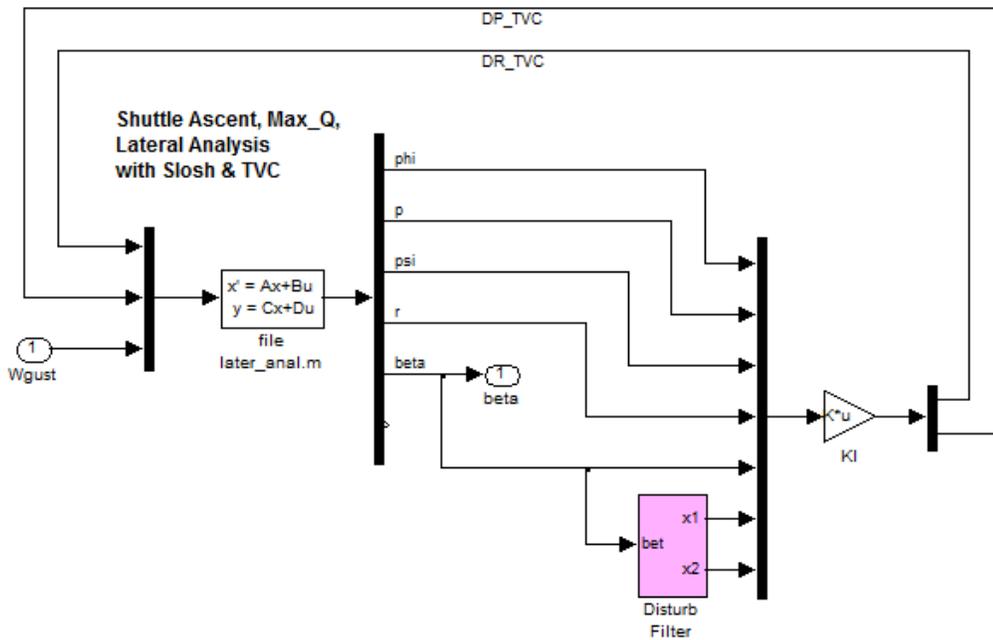


Figure (3.13) Simulink System "Sensitiv2.mdl" used for Preliminary Sensitivity Analysis. It has a direct measurement of Beta (not estimate)

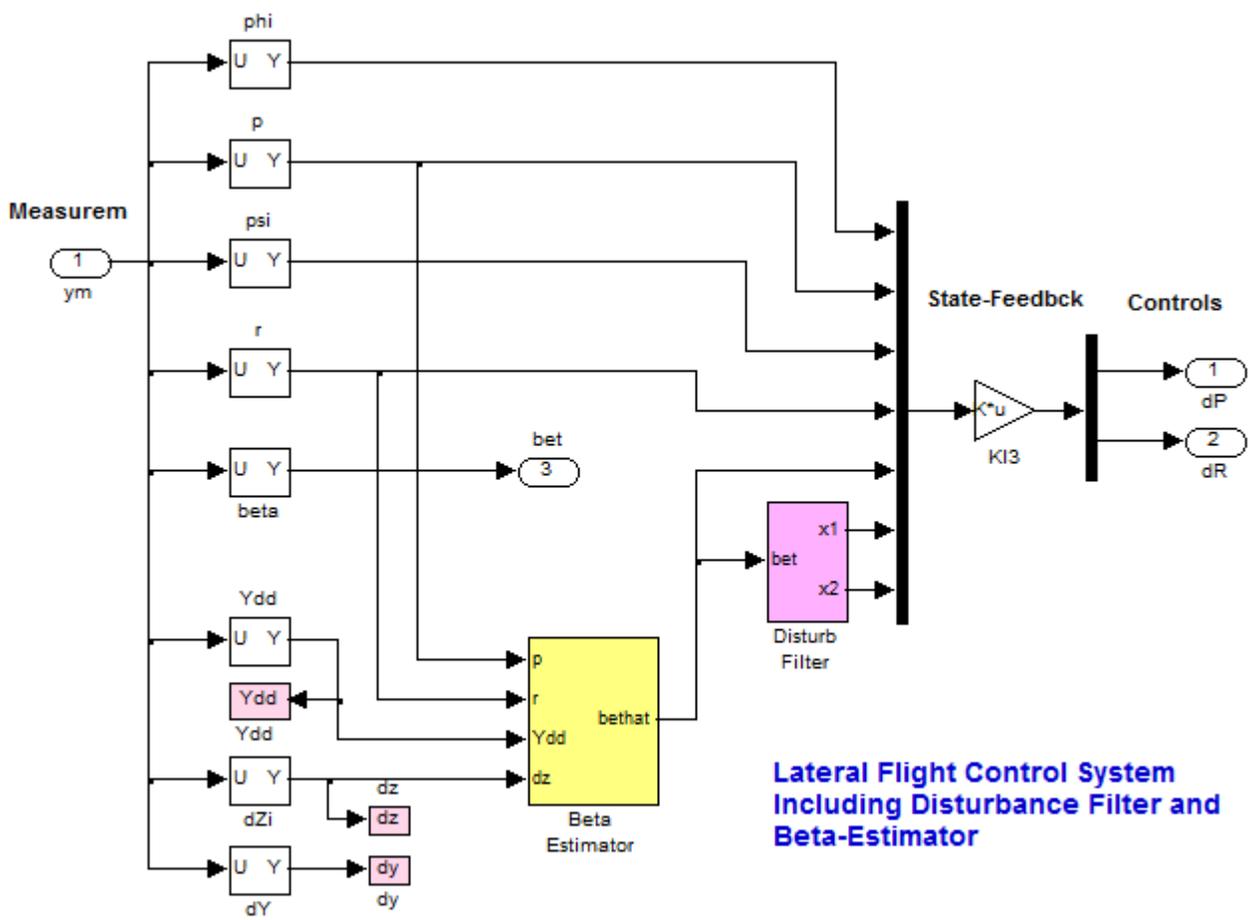


Figure (3.12b) Lateral Flight Control System Used in "Sensitiv4.mdl"

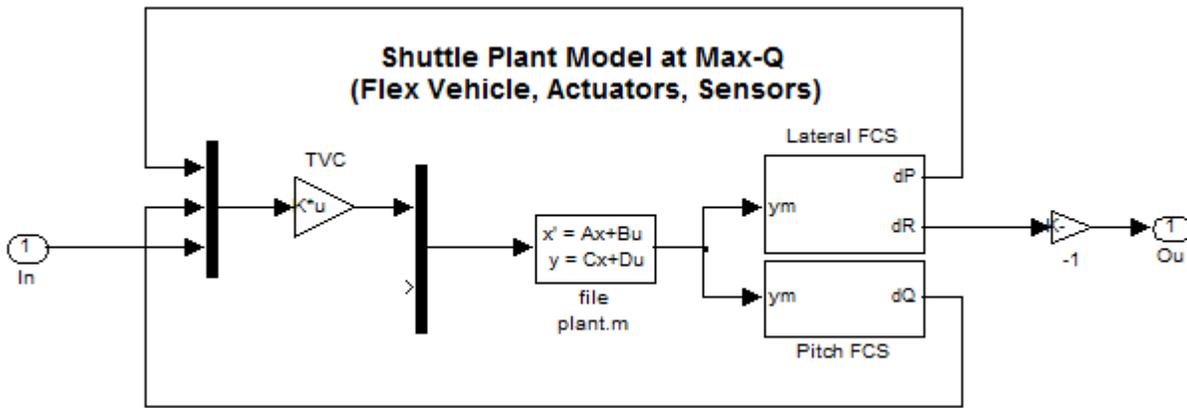
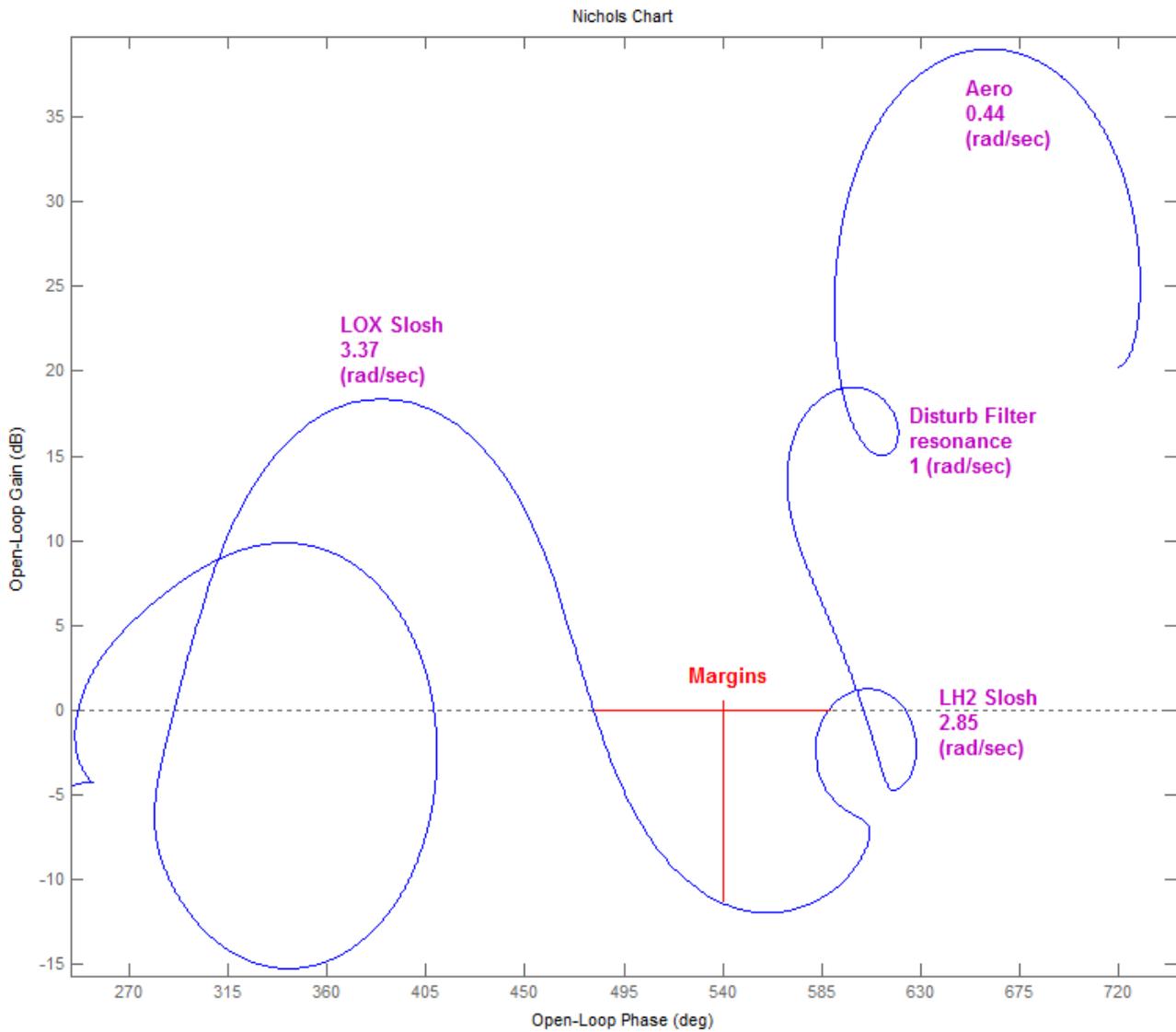


Figure (3.15) Simulink System "Anal5.mdl" used for Yaw Stability Analysis

Yaw Axis Stability (H-infinity design)



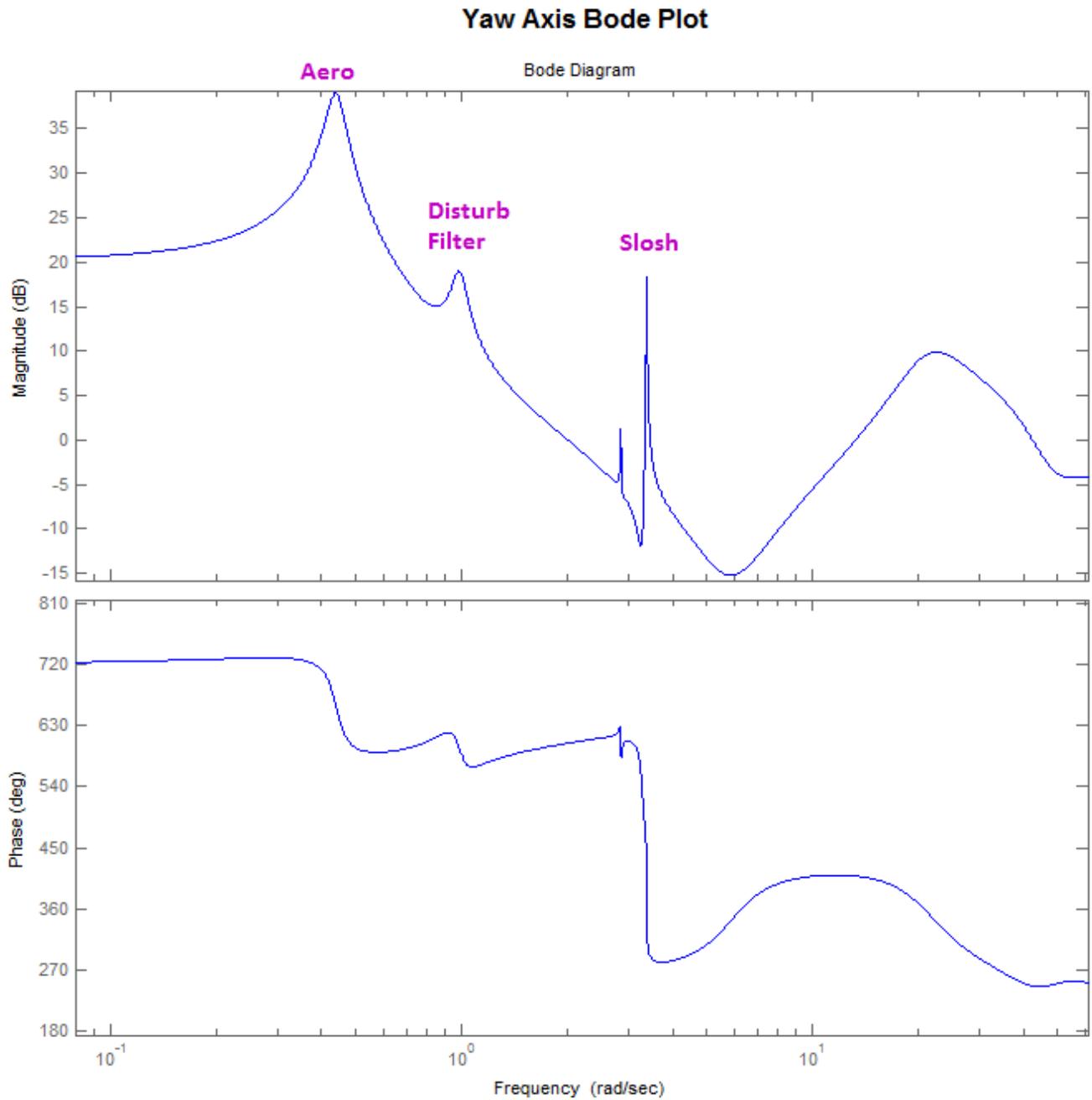


Figure (3.16) Nichols and Bode Open-Loop frequency response plots for the Yaw axis obtained using the Simulink System "Anal5.mdl". It shows an additional resonance at 1 (rad/sec) introduced by the disturbance filter. The high frequency resonance is typically attenuated with low-pass filters which are not included in this analysis.

Alpha/ Beta Sensitivity to Gusts

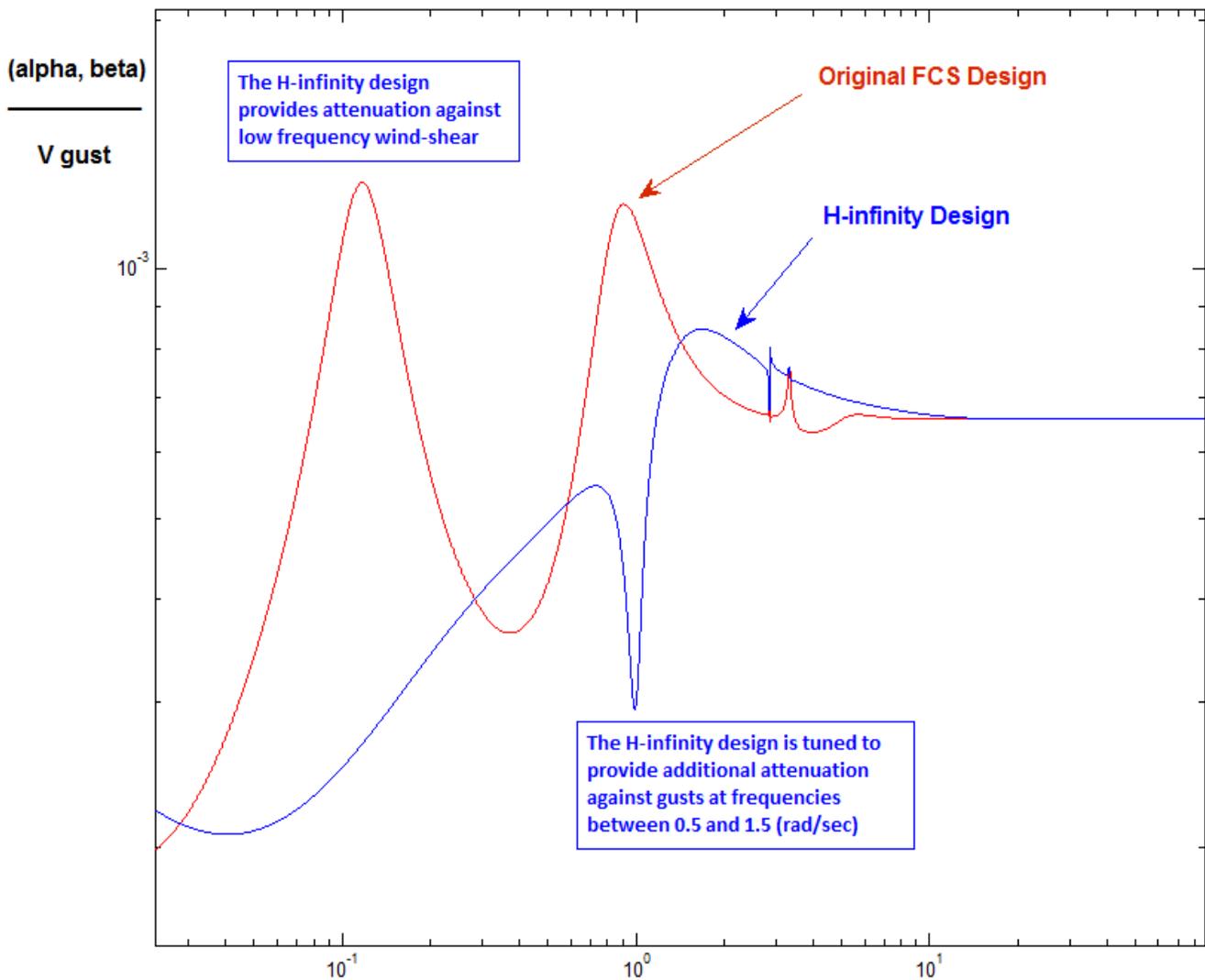


Figure (3.17) Sensitivity analysis plot showing the singular values for the transfer-function between the Gust input and (α, β) together. The frequency is in (rad/sec). It is comparing the sensitivity to gust disturbances between the original Shuttle FCS design and the H-infinity design. The blue curve is obtained using Simulink model "Sensitiv4.mdl" in Figure (3.12). The red curve was obtained from the original classical FCS design in section 3.0, Figure (3.1), as already described. The H-infinity design demonstrates superiority in the sensitivity both: at low frequencies against wind-shear, and also at 1 (rad/sec) which is the anticipated frequency of wind-gust disturbances.