MSFC-442

Rapid Contingency Simulation Modeling of the NASA Crew Launch Vehicle

Kevin M. Betts[1], and R. Chad Rutherford[2], James McDuffie[3], and Matthew D. Johnson[4]
bd Systems, a Subsidiary of SAIC, Huntsville, AL, 35802

The NASA Crew Launch Vehicle is a two-stage orbital launcher designed to meet NASA's current as well as future needs for human space flight. In order to free the designers to explore more possibilities during the design phase, a need exists for the ability to quickly perform simulation on both the baseline vehicle as well as the vehicle after proposed changes due to mission planning, vehicle configuration and avionics changes, proposed new guidance and control algorithms, and any other contingencies the designers may wish to consider. Further, after the vehicle is designed and built, the need will remain for such analysis in the event of future mission planning. An easily reconfigurable, modular, nonlinear six-degree-of-freedom simulation matching NASA Marshall's in-house high-fidelity simulator is created with the ability to quickly perform simulation and analysis of the Crew Launch Vehicle throughout the entire launch profile. Simulation results are presented and discussed, and an example comparison fly-off between two candidate controllers is presented.

I. Introduction

THE United States National Aeronautics and Space Administration (NASA) has committed to building the ARES I Crew Launch Vehicle as the man-rated launcher to support the implementation of the Vision for Space Exploration [1]. Preliminary Guidance, Navigation, and Control (GN&C) architectures are being developed by Marshall Space Flight Center (MSFC) to evaluate the ability of the vehicle to meet requirements. An important component of the flight control process is the frequency domain stability analysis, used to calculate the control system margins as a measure of the robustness of control algorithms. The need exists for a tool that can be used to model the medium to high fidelity aspects of the GN&C and plant characteristics of the ARES I vehicle. It is desired that this tool be able to simulate the performance of the GN&C during flight and be capable of generating results quickly due to modifications in software algorithms, sensor and actuator hardware, etc. To this end, bd Systems has developed SAVANT (Stability Aerospace Vehicle ANalysis Tool), an easily re-configurable, modular, nonlinear six degree of freedom simulation of the ARES I CLV in Matlab/Simulink that can be used to rapidly support changes to both vehicle software and hardware configurations. SAVANT is capable of generating both time domain and frequency domain information of both the vehicle integrated stack and Upper Stage flight during ascent flight phases. The tool is used to provide vehicle system and component analysis involving system data for multiple Design Analysis Cycles (DAC) and is able to simulate the effects of component and algorithm modifications in terms of flight performance and robustness requirements.

This paper provides a description of the simulation tool and preliminary results from a variety of analyses conducted. Section II provides an overview of the tool and, for frequency domain analysis, the linearization methodology used. Section III discusses the subsystem models in detail in terms of model algorithms and implementation into SAVANT. Section IV gives time-domain results of the system for simulation variables of interest in a nominal trajectory. Section V discusses the conclusions and potential future applications of the tool.

---

[1] Avionics Manager, Advanced Technology Division, Address/Mail Stop, and AIAA Member.
[2] Program Manager, Advanced Technology Division, Address/Mail Stop, and AIAA Member.
[3] Avionics Engineer, Advanced Technology Division, Address/Mail Stop, and AIAA Member.
[4] Aerospace Engineer II, Advanced Technology Division, Address/Mail Stop, and AIAA Member.

II. SAVANT Overview

The initial purpose of the SAVANT program developed by bd Systems was to provide frequency-domain information about the preliminary CLV control system design. The tool has been developed in Matlab/Simulink. A representation of the CLV control system algorithms has been coded in Simulink to match the C algorithms used in MSFC's 6DOF time-domain simulation tool, MAVERIC II. The simulation supports the complete flight of the vehicle and switches the control modes and gains based on the flight regime. Models have been developed to represent the forces and moments generated on the vehicle by aerodynamics, propulsion, and gravity. Engine inertia ("tail-wags-dog") effects and liquid propellant slosh models are also included. The equations of motion for the vehicle include both rigid body (six-degrees-of-freedom) and flexible body effects. The user can select the number of flexible modes that are included in a particular run. TVC actuator dynamics models are used to simulate the response of the actual nozzle position when given a certain commanded nozzle position. The simulation reads values for guidance commands directly from a MAVERIC II output data file and interpolates these values as a function of the simulation time. The simulation can be run in a variety of configurations (rigid body with or without flex, slosh, and engine inertia) by making changes to the simulation initialization file. Figure 1 shows the top-level Matlab/Simulink diagram of the CLV Stability Analysis Tool.

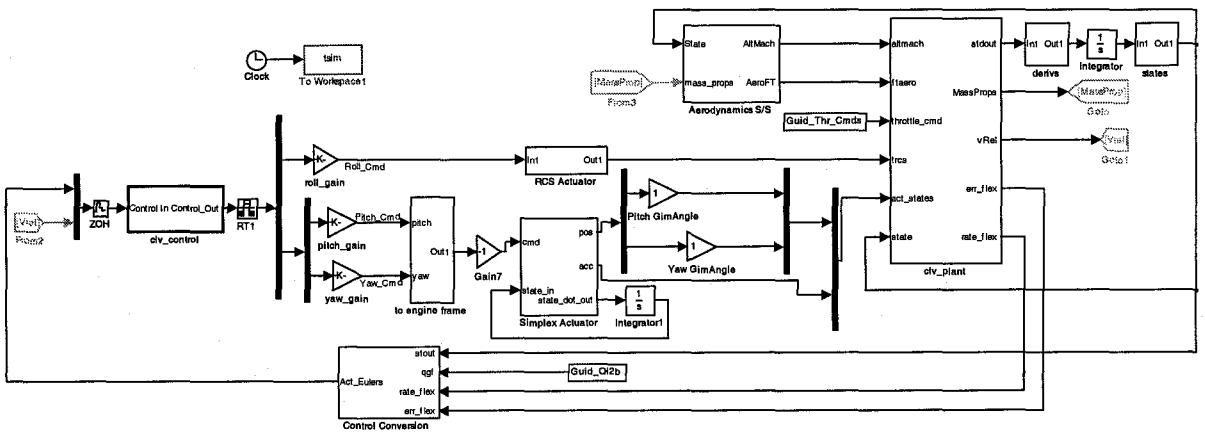

Figure 1. Top-level Matlab/Simulink diagram of SAVANT

The simulation can be run in both the time-domain and the frequency-domain. Time-domain results allow the simulation output to be compared to other simulations' time-domain output data to verify that the controller has been implemented correctly and to serve as an independent check of the other models used in multiple tools. In the frequency-domain analysis mode the simulation makes use of the Simulink Control Design toolbox functionality to generate linearized state-space representations of both the open and closed-loop systems. The user specifies the time points in the trajectory when linearized models are required. The simulation is first run in the time-domain to establish the operating points about which to linearize the system. The tool then linearizes the system about the operating points, generating separate A, B, C, and D matrices to represent the open and closed-loop system dynamics. The tool currently uses Simulink's block-by-block analytic linearization capability; in the future results can also be generated using the alternative numerical perturbation option available in Simulink Control Design.

Once the state-space representations are generated for the desired operating points, the tool has access to the built-in Matlab Control Toolbox frequency domain analysis capabilities to analyze the linearized systems. A variety of analysis methods are available {Bode plots, Nyquist plots, Root Locus, Pole/Zero Maps, and linear system time-domain responses (step, impulse, ramp)}. In the default mode, gain and phase margins, Bode plots, and closed-loop system bandwidths are captured for each operating point specified by the user, and the state-space representations are saved Matlab Linear Time Invariant (LTI) objects. This allows the user to run a variety of operating points in batch mode, then go back and examine specific operating points in more detail after the run if the initial results reveal anything unexpected.

SAVANT also has the capability to analyze dispersion data via Monte Carlo runs. In this case the user specifies a list of variables to be dispersed in the simulation and the ways in which those variables should be dispersed (normal distributions, uniform distributions, etc.). The simulation is then run a number of times (user-dependent)

and both time-domain and frequency-domain trends can be analyzed. In the time-domain, maximums and minimums of variables of interest are displayed with accompanying run number. In the frequency-domain the gain and phase margin spread is available for the number of runs selected.

III. Model Description

Models of varying levels of fidelity are employed in SAVANT. A description of each model is given in the subsections below.

A. Aerodynamic Model

The aerodynamics model consists of three elements: the environments model, the aerodynamic coefficients lookups, and the force and moment generation algorithm. The environments model is responsible for calculating, from the integrated state vector, those inputs needed by the rest of the model. These include the angles of attack and sideslip, mach, wind-relative velocity, altitude, and dynamic pressure. A winds model is employed in this element to bias the Earth-relative velocity vector. KSC mean annual winds are used for this purpose. Figure 2 is the Simulink block diagram algorithm for this model, which outputs horizontal winds in the North East Down frame, based upon altitude. The wind vector is then transformed to the ECI frame and subtracted from the Earth-relative velocity vector so that a wind-based mach number can be calculated. The winds model is contained in an enabled subsystem so that it can easily be turned off by use of an input flag.

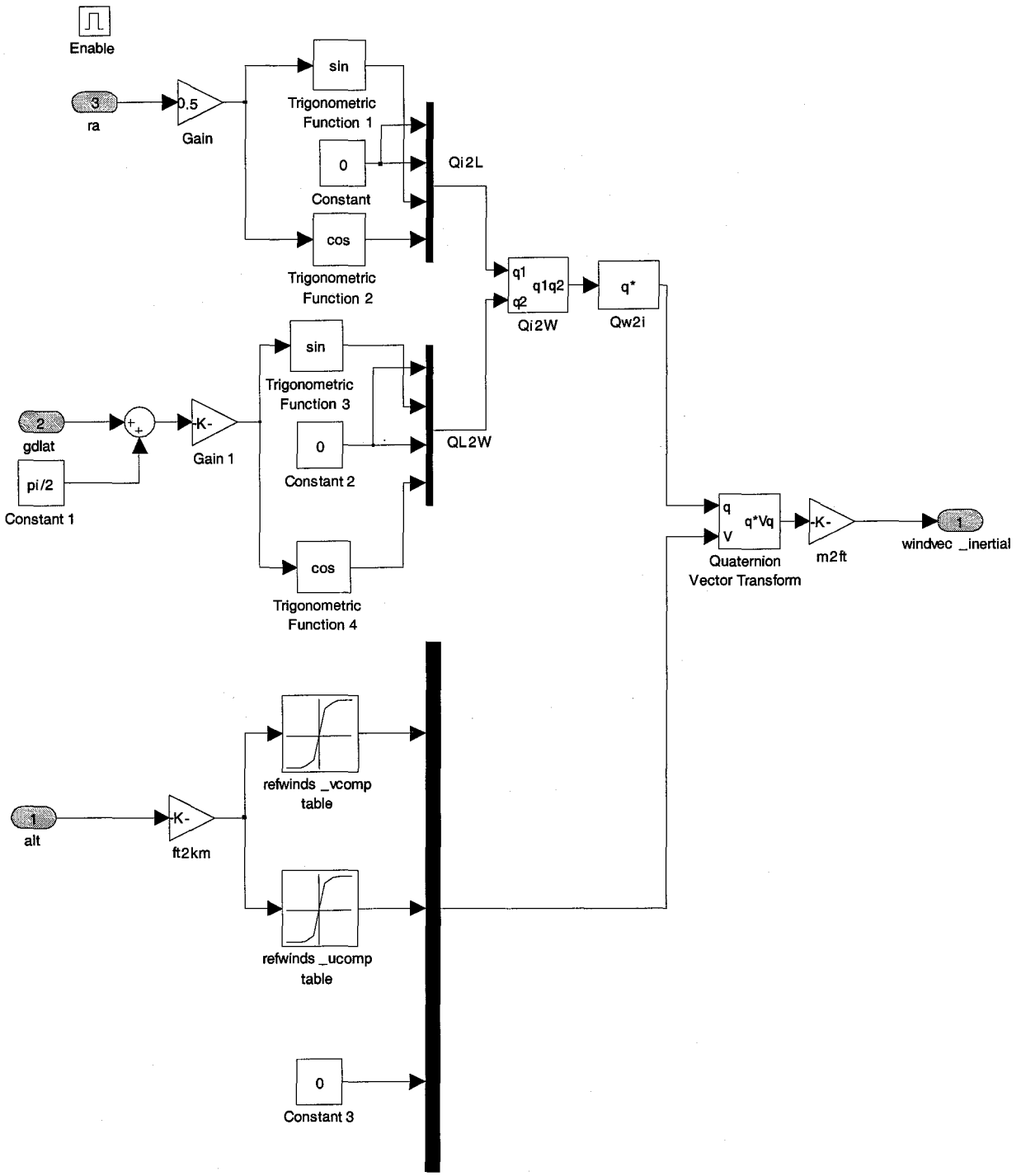

Figure 2. KSC reference winds model block diagram

The aerodynamic coefficients model includes 6DOF tabular coefficient data based upon DAC-2 CFD and wind-tunnel data. These coefficients are functions of combinations of mach, angle of attack, and sideslip angle. A base force is also included as a function of time and acting in the body X axis. The aerodynamic coefficients apply to a specific vehicle configuration. For ascent, three such configurations are taken into account: the integrated stack, the Upper Stage, and the Upper Stage after the jettison of the Launch Escape System (LES). Each configuration's model is contained in its own enabled subsystem that is triggered by a vehicle event flag or event time history that is

input to the model. The block diagram representation of the integrated stack's aerodynamic look-up tables is given in Figure 3.

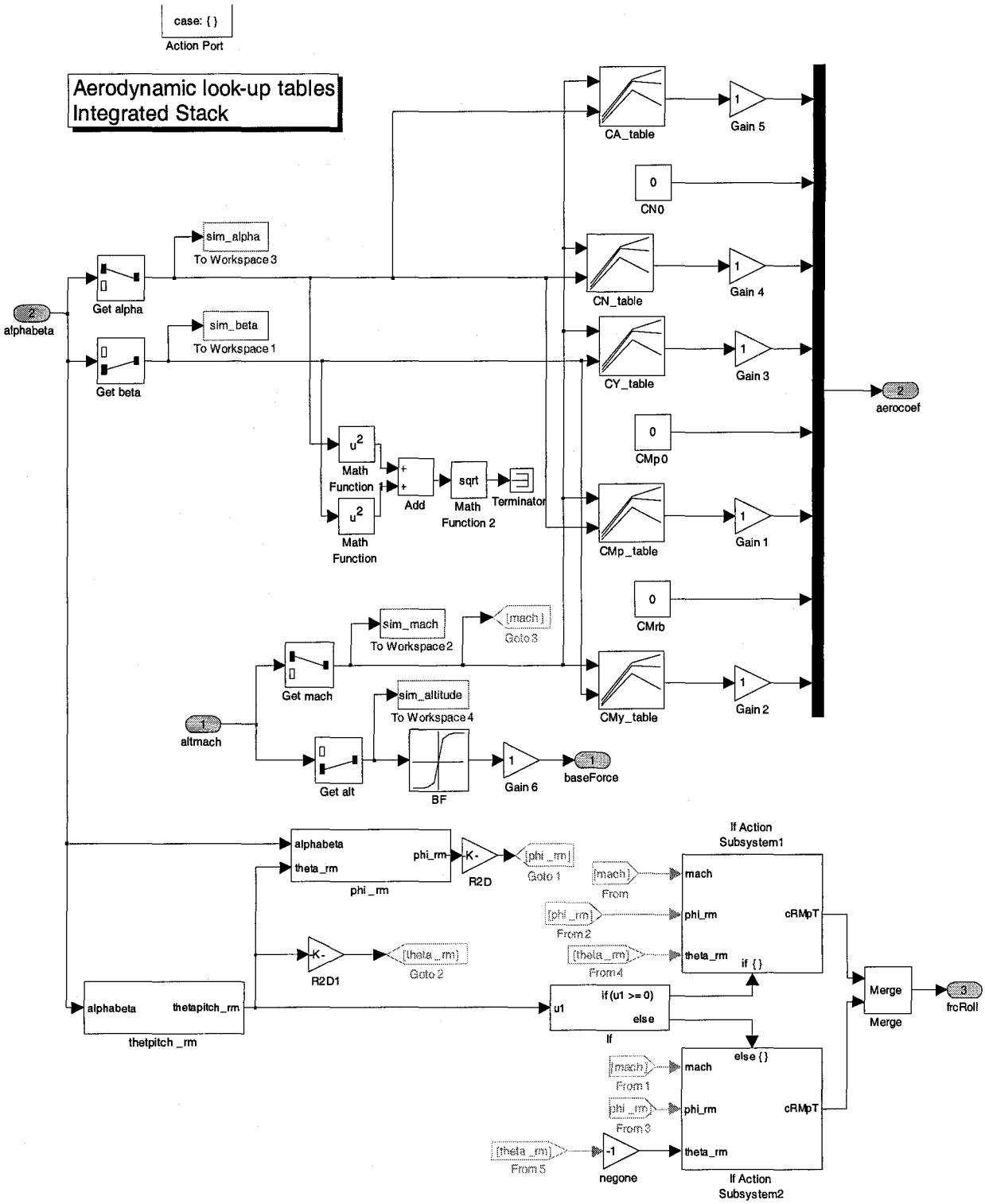

Figure 3. Aerodynamic look-up tables for the integrated stack

The force and moment generation algorithm uses the computed coefficients to calculate the aerodynamic force and moment of the vehicle for a given state:

$$\overline{F}_a = qs_{REF}\overline{C}_F + \overline{F}_{BASE}$$

where q represents the dynamic pressure, S_{REF} is the reference area, $\overline{C}_F$ is a vector of coefficients in the body frame (axial, side, normal), and $\overline{F}_{BASE}$ is the base force vector, whose only nonzero element is in the body X direction.

Similarly, the moments are computed as:

$$\overline{M}_a = qs_{REF}l_{REF}\overline{C}_M + \overline{r}_{CG/MRP} \times \overline{F}_a$$

where l_{REF} represents a vector whose elements are the so-called "wingspan" in the body X and Z axes and the aerodynamic chord in the body Y axis, $\overline{C}_M$ is a vector of roll coeffients in the body frame, $\overline{r}_{CG/MRP}$ is the position vector from the vehicle cg to the moment reference point, and $\overline{F}_a$ the computed aerodynamic force vector.

B. Mass Properties

A time-history of mass properties (vehicle weight, inertia, and center of gravity location) is implemented in SAVANT as lookup tables with propellant remaining as the independent variable. If the slosh model is enabled, these parameters are adjusted to account for the absence of the slosh masses in the total vehicle. This is necessary because the slosh model employed in SAVANT models the slosh masses as separate vehicles connected to the ARES I by springs and dampers. These idealized connections provide the slosh forces and moments on the vehicle due to the displacements and displacement rates of the slosh masses.

Three sets of mass properties tables are used in the tool, coinciding with three flight phases of the vehicle: the integrated stack, the Upper stage with the LES, and the Upper stage without the LES.

C. Actuator Models

There are two gimbal actuator models available for simulation –a 3rd-order model, and a higher-fidelity 'simplex'[ref] model. Figure 4 represents the block diagram for the 3rd-order model, which outputs the actuator acceleration and position to the rest of the plant.

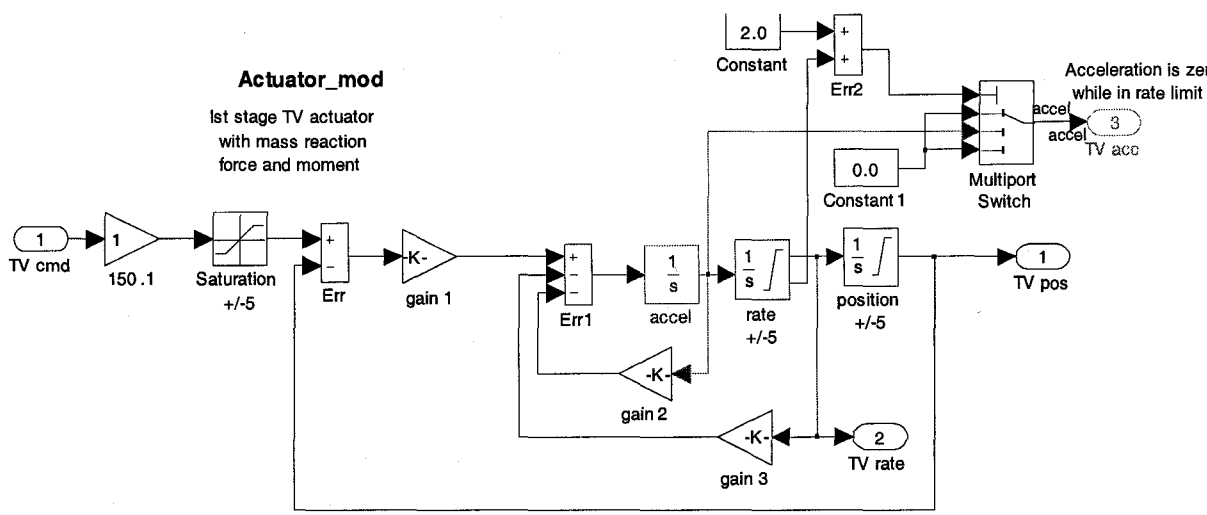

Figure 4. 3rd Order TVC actuator model

Figure 5 is a block diagram of the simplex actuator model, which incorporates the power valve dynamics as well as the engine motor parameters that make up the 3rd-order model.

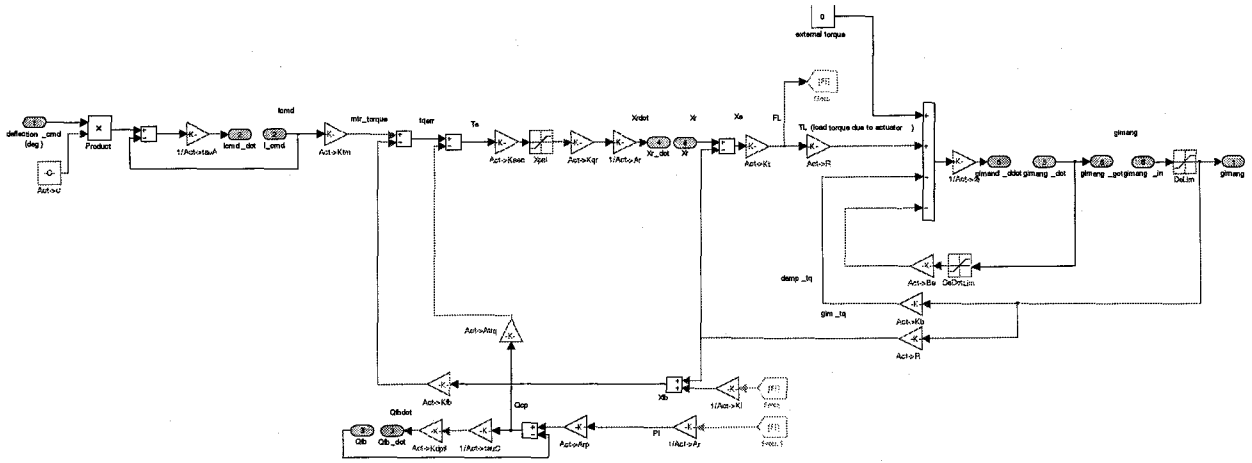

Figure 5. Simplex TVC Actuator model

D. Reaction Control System (RCS) Jets

The RCS system is modeled as a standard on/off actuation system. The commanded thrust value is always either 'on' or 'off' over the 0.02 sec control cycle. The RCS thrust is modeled as a ramp up to a maximum value that is altitude dependant (tabular data input to the model based upon a specific Design Analysis Cycle) and a subsequent ramp down when the valve closes. Back-pressure effects are also modeled in the system. For the ARES I vehicle the RCS is used to control the body roll axis only.

E. Guidance Commands

Guidance commands are generated from a MAVERIC II run and are read into the SAVANT simulation. Guidance commands are given in the form of a commanded angular velocity (commanded rate) and an inertial-to-commanded body quaternion. This data is generated every 1 Hz and is passed through a smoothing algorithm before being used in the flight control system. The smoother operates by assuming a constant rate eigenaxis maneuver about the error quaternion between the old and new quaternion command. The smoothed rate commands are generated by integrating computed angular acceleration commands at the 50 Hz rate. This acceleration vector is computed in a low rate block by the equation:

$$\overline{\alpha} = -2\xi\omega\left(\overline{v} - \overline{v}_{eig}\right) - 2\omega^2 \overline{q}_{err}$$

where $\overline{\alpha}$ is the acceleration vector, ξ is the damping amount, ω is the undamped natural frequency, $\overline{v}$ is the previous rate command, $\overline{v}_{eig}$ the (assumed) constant angular velocity of the maneuver, and $\overline{q}_{err}$ the maneuver eigenaxis. The smoothed quaternion command is created by multiplying the previous smoother output by the delta quaternion, or the quaternion between the old desired body axes to the updated desired body axes. This delta quaternion is based upon the commanded rate and the guidance timestep.

F. Flight Control System

The Flight control system is uncoupled in all three axes (pitch, yaw, roll), although the same architecture is present for pitch and yaw. The FCS block diagram is shown in Figure 6. Here it is seen that the pitch and yaw controller channels consist of flex filters (two fourth order discrete transfer functions in series) followed by a gain-scheduled PID controller. The filter coefficients change from the integrated stack phase to the Upper stage flight phases, and the PID gain tables change based upon the event of three flight phases: integrated stack, coast, and Upper stage from J2X ignition. In the PID gain tables' case, not only the table data but the input to the tables themselves changes, from altitude to relative velocity.

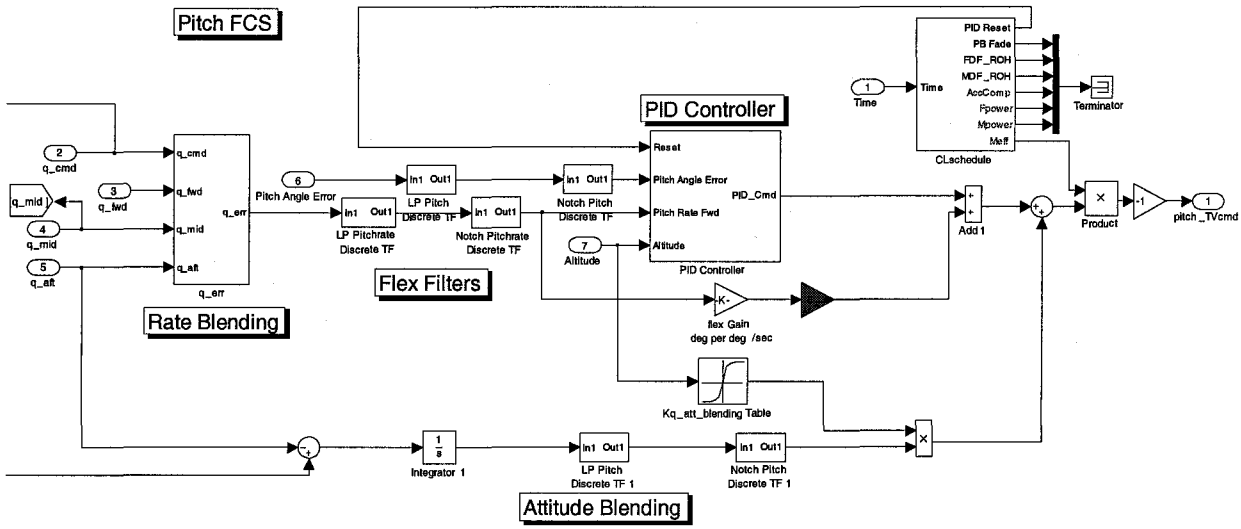

Figure 6. Pitch FCS block

The inputs to the FCS are the smoothed rate commands, the sensed rates at each sensor, the error angles, and the inputs necessary for the lookup tables (altitude, relative velocity). The error angles are extracted from the error quaternion, which is formed by multiplying the sensed body-to-inertial quaternion with the smoothed inertial-to-commanded body quaternion. Flex angular displacements are added to these angles after extraction.

Blending is accomplished in the block of the same name). This algorithm allows the FCS to use certain percentages of each rate signal combined with each other for a "unified" rate error. In this scheme the total percentages will amount to 1. For example, the blending may be [0.8 0 0.2] for the Upper Stage rate gyro, the IMU, and the First Stage rate gyro, respectively. The blended rate errors pass through two fourth order flex filters before they are input as the derivative element to the PID. The final contribution to the control signal comes from the integrated sensed rate gyro (First stage rate gyro for the integrated stack, Upper stage rate gyro after separation). This (unblended) signal is passed through a set of flex filters and then added to the output of the PID.

The roll controller makes use of the flex filtered rate and angle errors. This form of control uses a phase plane (roll error vs. roll rate error) to determine the appropriate condition for commands. Both attitude and rate deadbands are employed in the switching logic, which moves from a positive torque command, to zero command (drift), to a negative torque command, or vice-versa.

G. Vehicle Separation

Staging is modeled as an instantaneous loss of mass at the commanded time step. There is no modeling of the separation dynamics between the two vehicles.

H. Navigation Sensors

Currently, perfect navigation sensors are used for the FTINU, while transfer functions are employed for the rate gyros. The transfer functions are discretized to the sensor update rate of 200HZ.

I. Atmosphere Model

The US76 atmosphere model was used in the simulation. The US76 atmosphere model is used to calculate air density and Mach number at a certain altitude and speed.

J. Gravity Model

The J4 gravity model was used in the simulation. The gravity model includes first- and second-order Earth oblateness effects, but does not include local high-frequency gravity anomalies or vertical deflection data specific to the launch site.

K. Flexible Body Dynamics Model

The CLV body flexibility portion of the bd Systems Stability Analysis Model is based on several assumptions. The modal data set being used for this analysis was provided by NASA/MSFC. The data was obtained from Charles

Hall/MSFC after it had been developed by Dave McGhee/MSFC. This modal data set contains mode shapes and frequencies for 1st stage liftoff and burnout and 2nd stage ignition and burnout. The modal displacements and rotations are given at several nodal points along the centerline of the vehicle in structural coordinates. The units of the displacement data are inches and the rotations are radians. The assumption made for this design cycle analysis is that changes of mass properties of the vehicle are negligible except for the change from 1st stage lift off modes to 2nd stage ignition modes. This assumption needs to be re-examined for later analyses.

The units of the stability model are feet and the Body reference frame is rotated about the body/structure Y axis by 180 degrees from the Structure frame. The origin of the body frame is the empty vehicle center of mass. Since the modal units as given are in inches, they must be changed to feet. This is done while changing the modal normalization to be relative to the new feet units. The net effect of this is that the modal displacements are divided by sqrt(12) and the modal rotations are multiplied by sqrt(12). The modal frequencies are unchanged. Since the modal components are given in structural frame coordinates, the Structure to Body frame transformation is BS = diag([-1 1 -1]).

It is necessary to select modes to be retained in the analysis model using the available data from the existing modal set. Our current methodology is to base modal retention on response. Modes to be retained are selected by starting at the lowest frequency beyond the rigid body modes and adding modes one by one until no further changes in the closed loop response are noted.

The key elements of the dynamic model are the sensors, actuators, and propellant slosh mass attach points. These are identified with nodes of the structural model as stated in the following:

```
Nav sensors    Node ID 6001 (not currently available in model)
4021 (co-located) used instead
1st stg RCS    Node ID 6030
1st stg gmbl   Node ID 718 (closest available node)
2nd stg RCS    Node ID 6023
2nd stg gmbl   Node ID 6226 (6025 is closest available)
LOX slosh att pt 1st stage, average of Nodes 6021 and 6023
LH2 slosh att pt 1st state, average of Nodes 6004 and 6003
LOX slosh att pt 2nd stage, average of Nodes 6021 and 6023
LH2 slosh att pt 2nd stage, average of Nodes 6004 and 6003
```

The basic development of the flexible body dynamics model proceeds from several linearizing assumptions. These are summarized as small translation and rotation rates and accelerations, small deformations and deformation rates. These assumptions allow linearization of the dynamics model beyond the rigid body.

Double counting of slosh masses, propellant masses, and rocket motor mass are ignored for this analysis cycle. Their effects are expected to be negligible, at least for first stage stability analysis. These effects may be more significant for 2nd stage studies and will be reconsidered for future analyses.

L. Slosh Dynamics Model

The slosh model in SAVANT consists of two architecturally identical blocks, one each for the LOX and LH2 Upper stage tanks. For each block the assumption is made that slosh forces can be generated by modeling an effective slosh point mass connected to the tank centerline by a spring and damper acting in the lateral (body Y and Z) direction. The frequency, damping, slosh mass, and slosh mass position are all functions of the liquid level in the tank, which is assumed to not be affected by rotations of the vehicle. These functions are manifested in SAVANT as one dimensional lookup tables. The slosh mass vertical position together with the previous lateral position from the state vector are used to calculate the position vector from the cg of the vehicle to the slosh point mass. The slosh mass acceleration is then calculated as

$$\ddot{x}_s = 2\xi_s \omega_s \dot{x}_s + \omega_s^{\,2} x_s + \ddot{r} + \alpha \times r + 2\omega \times \dot{r} + \omega \times (\omega \times r)$$

where the s subscript denotes a slosh state or parameter, r is the position to the slosh mass from the vehicle cg, ω is the angular velocity of the vehicle at the slosh mass location, and α is the angular acceleration of the vehicle at the slosh mass location. The position, velocity, acceleration, angular velocity, and angular acceleration mentioned are made up of rigid and flexible elements in the SAVANT slosh model, although at present the flex additions are nulled out. One concern that exists with the flex components is the need for a relatively fine node layout in the flex tables. This is because the slosh mass axial location changes with time as the fluid drains from the tank, and the flex components of each input depend upon this node location. Once flex data is available for a fine enough mesh along the tank centerlines, these inputs can be incorporated into the simulation.

Because of the assumption of lateral spring and dampers, the axial acceleration is computed as the axial vehicle acceleration at the slosh point mass. This is equivalent to the equation above with the slosh axial state equal to zero.

To obtain the slosh force the slosh acceleration is multiplied by the effective slosh mass and negated to create the force applied on the vehicle. The moment is computed as the cross product of the moment arm (cg to slosh mass) with the slosh force. The slosh accelerations and velocities are integrated with the rest of the SAVANT state vector to obtain new velocities and positions.

Figure 7 is the block diagram top level of the LOX slosh model (LH2 is identical).

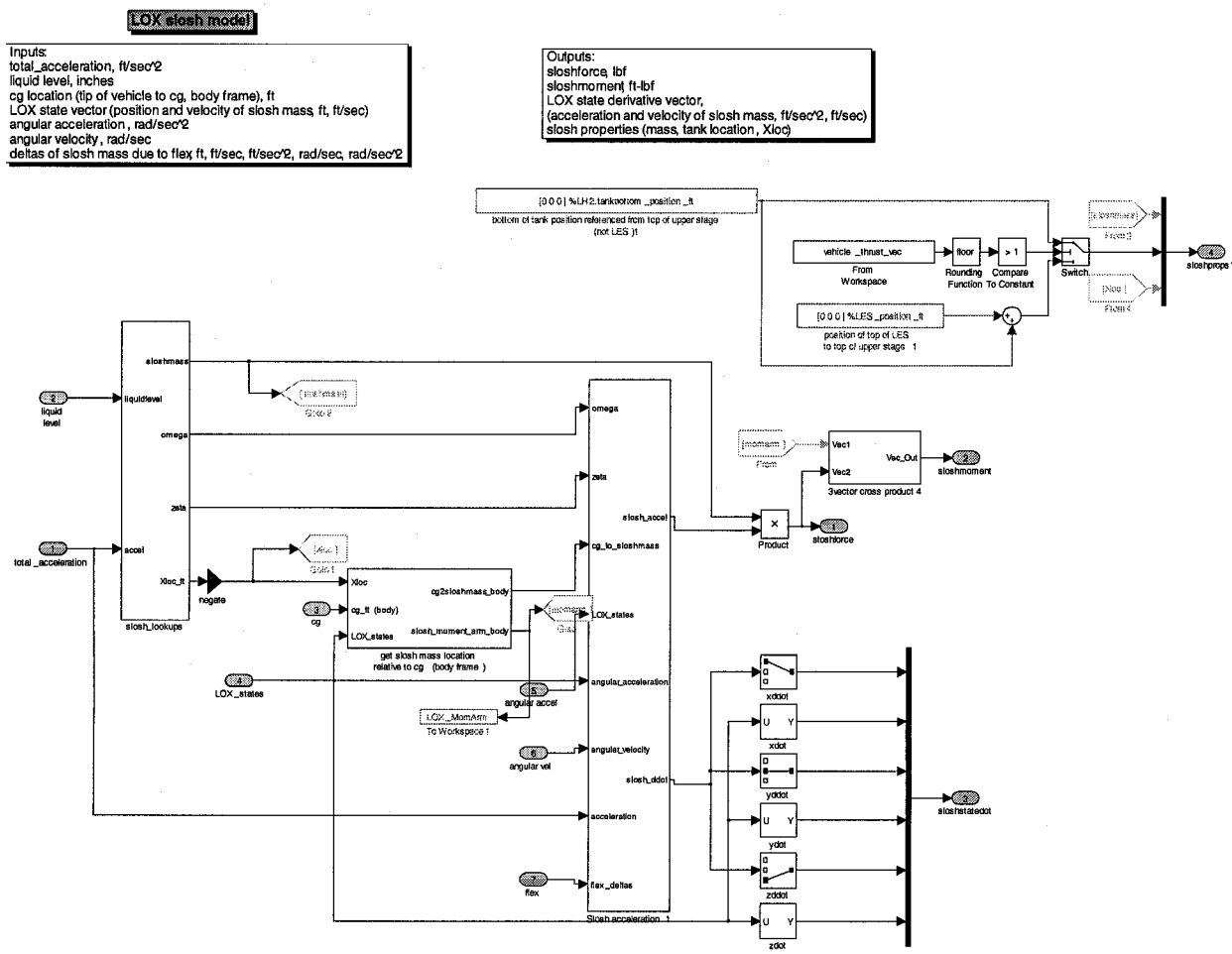

Figure 7. Slosh LOX top-level block diagram

Other assumptions made with this slosh model include:

- In the expression for acceleration of the rigid body, it is assumed that the velocity and acceleration of the rigid body's CG relative to the rigid body itself is negligible. That is, the model accounts for the location of the CG over time, but does not include the dynamic effects due to the motion of the CG.
- Similarly, the model assumes that the angular velocity effect caused by the rotation of the body is negligible.
- The position of the slosh masses is bounded by the tank walls, but no further effects of tank wall / slosh mass interactions are modeled.
- Tank liquid level is found by calculating the mass of liquid propellant used and assuming that the mixture ratio achieved is equal to the mixture ratio commanded in the MAVERIC II simulation.
- The frequency of oscillation of the slosh mass is assumed to vary with the square root of the magnitude of the vehicle acceleration.

- **Nozzle Engine Inertia Effects (Tail-Wags-Dog)**

The following assumptions were made in developing the nozzle engine inertia effects model (commonly known as 'tail-wags-dog'); the fidelity of the model can be increased in the future to remove these assumptions:

- The CLV was kept at small steady-state pitch angle for the majority of the trajectory.
- The CLV gimbal angles were small throughout the trajectory.
- Forces exerted on the stages due to the angular acceleration of the nozzles are negligible.
- Moments exerted on the stages due to terms other than the angular acceleration of the nozzles are negligible.
- Higher order terms in the coupled equations of motion are neglected.

IV. Results

This section shows results for two cases. In section A, the simulation tool is compared to MAVERIC II, NASA-Marshall's in-house high fidelity vehicle simulator. In section B, the tool is used to host a simulated fly-off between NASA-Marshall's PID pitch and yaw controller, and NASA-Langley's PD pitch and yaw controller. In Section C, results are given that show frequency-domain gain and phase margins for tests where flex frequencies were varied within realistic uncertainty bands to give a measure of system robustness.

A. Time Domain Results

In order to demonstrate that the tool yields valid results as a CLV simulation, the time history of the simulation is shown, with a comparison of the inertial tracking commands to the inertial tracking performance. Figure 8 - Figure 14 show the time history of a typical launch to the International Space Station. Figure 8 shows that the body rates are generally small, except during the large commanded roll maneuver shortly after launch. Figure 10 shows that the commanded trajectory is tracked well by the simulation.

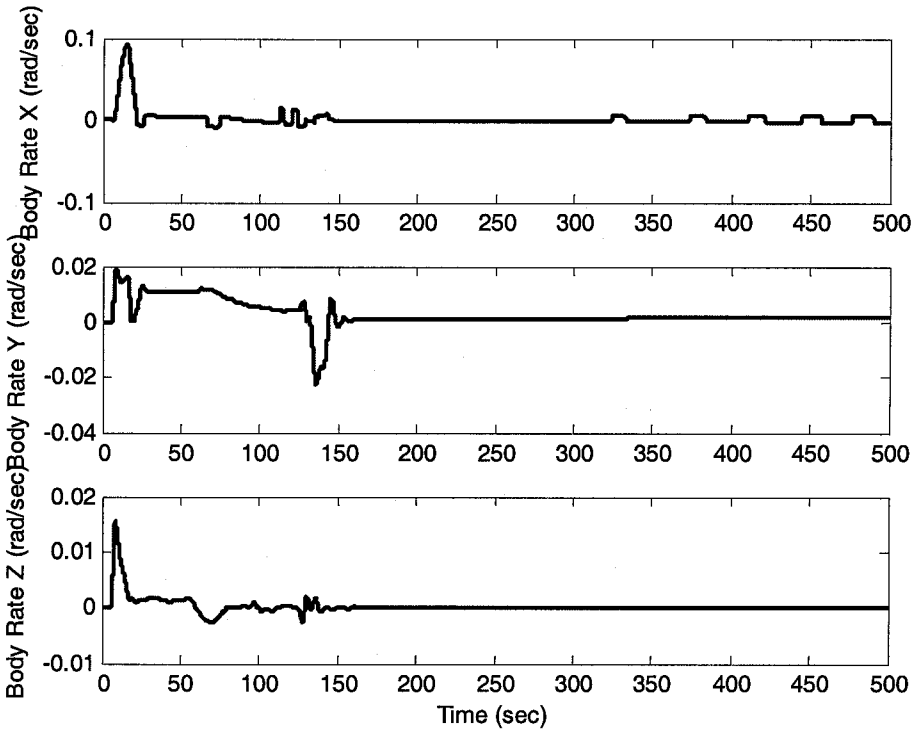

Figure 8. CLV Body Angular Rate in Radians per Second.

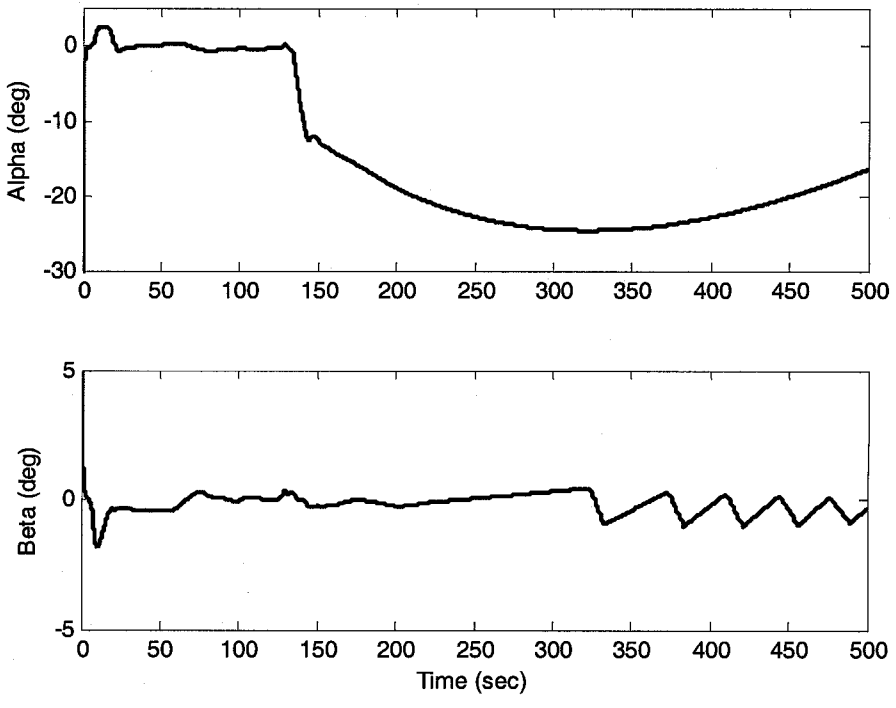

Figure 9. Angle of Attack and Sideslip Angle.

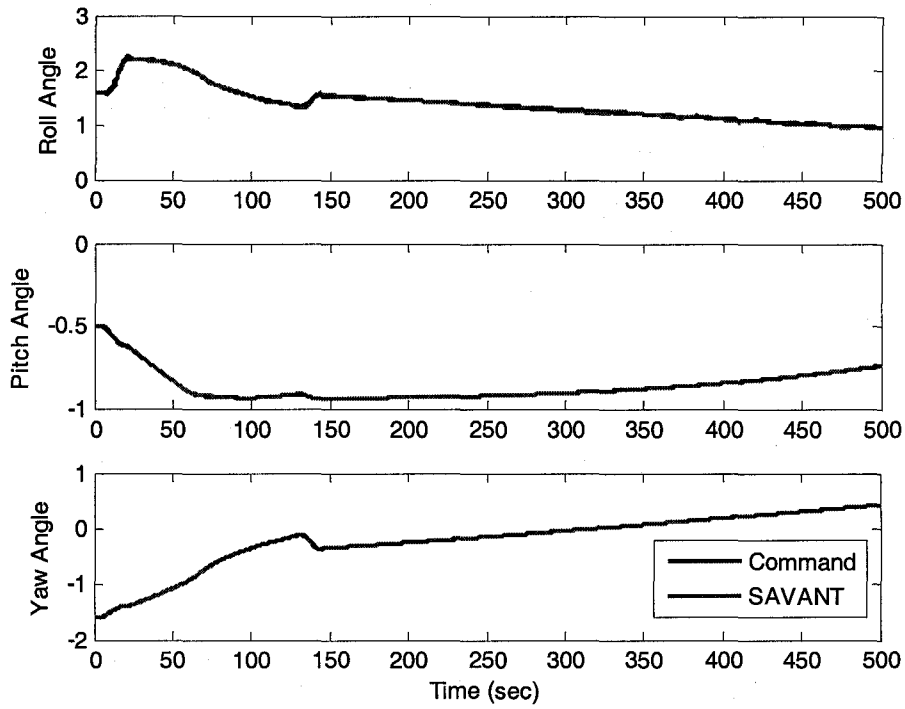

Figure 10. CLV Attitude Command Tracking.

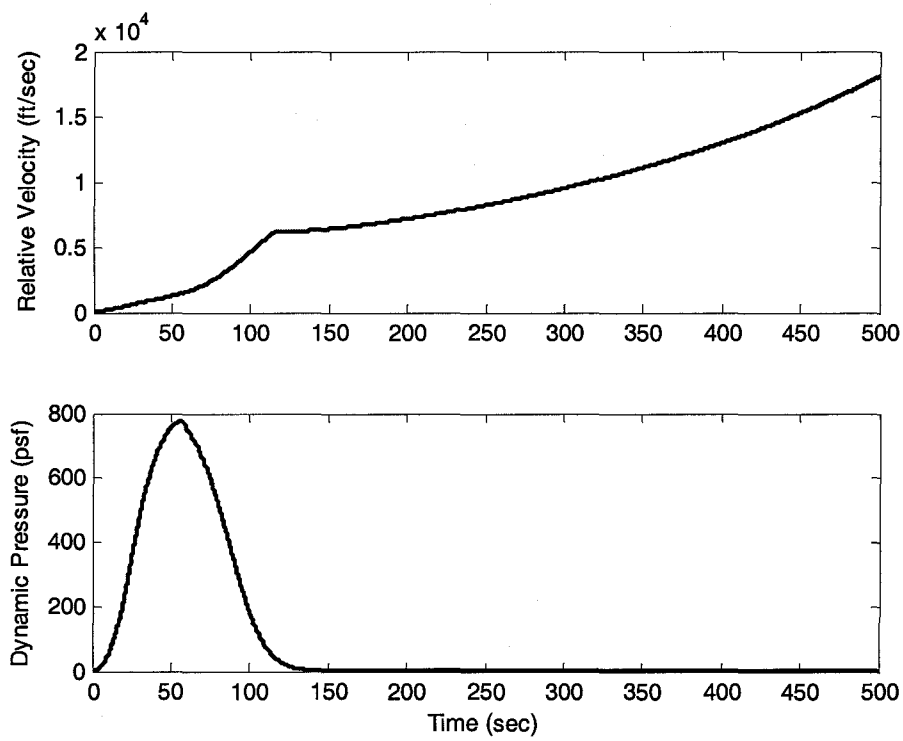

Figure 11. Relative Velocity and Dynamic Pressure.

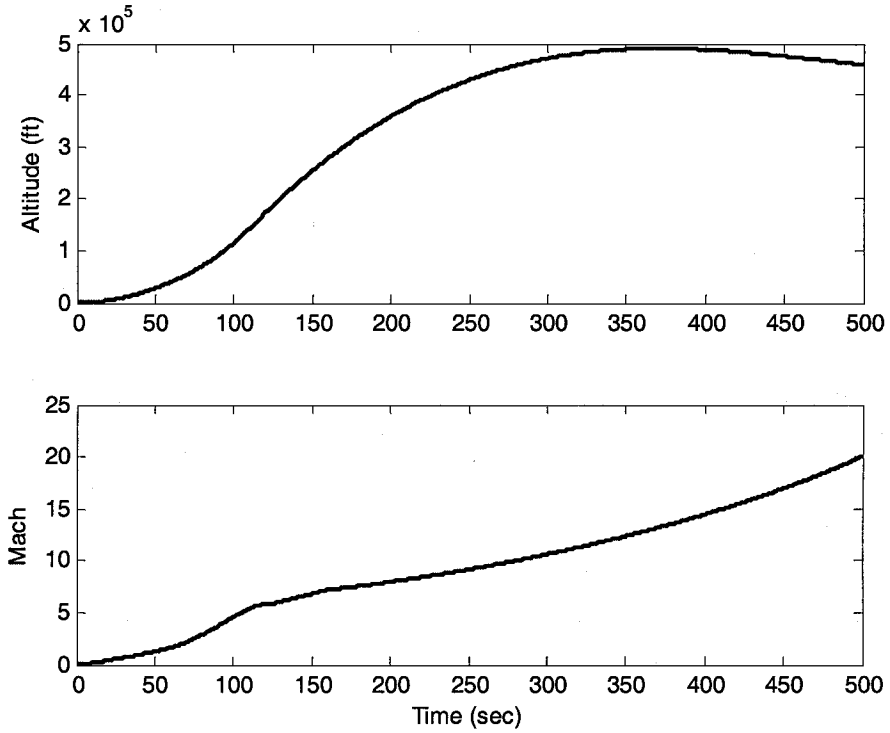

Figure 12. Altitude and Mach.

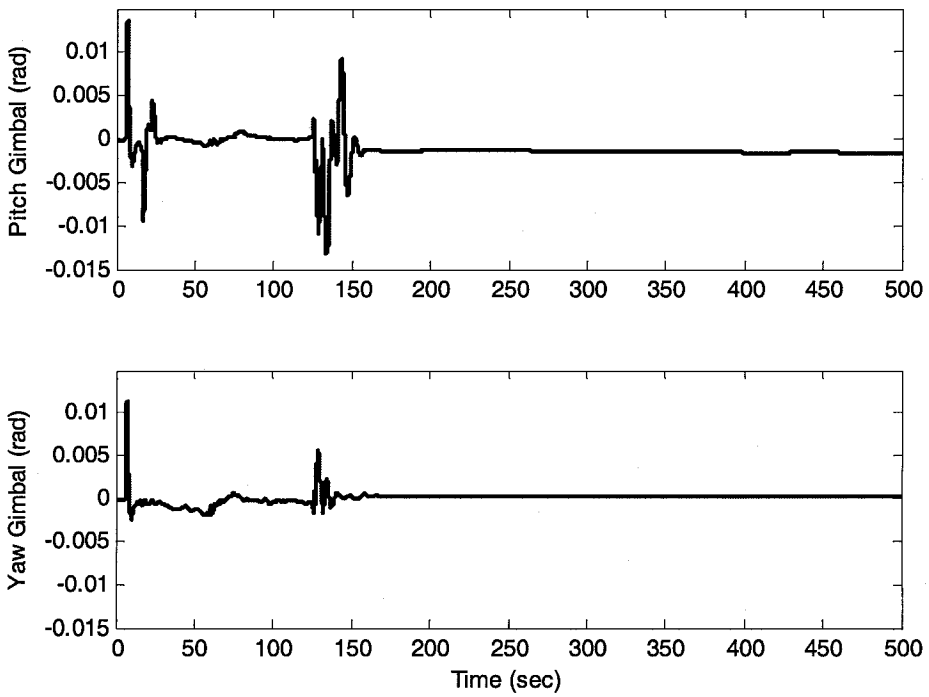

Figure 13. Pitch and Yaw Gimbal Angles.

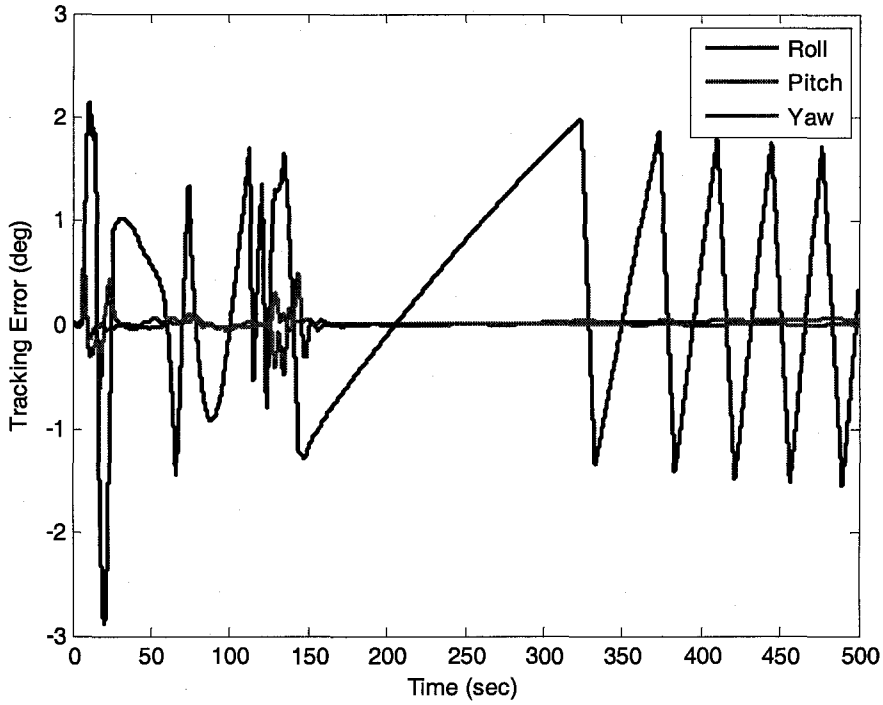

Figure 14. Attitude Error Angles.

B. Flex Parameter Study for GVT

A rapid-turnaround study to determine the dependence of the stability parameters on the flexible modes of the vehicle was conducted in order to decide whether GVT (Ground Vibration Testing) would be performed. The vehicle simulation was flown to several representative points in the trajectory and the system was linearized at those operating points in order to show the stability. In separate tests, flex frequency and the magnitude of the flexures was varied from 70% to 130% of the nominal values. The pitch Nichols chart results are shown in Figures 14-21.

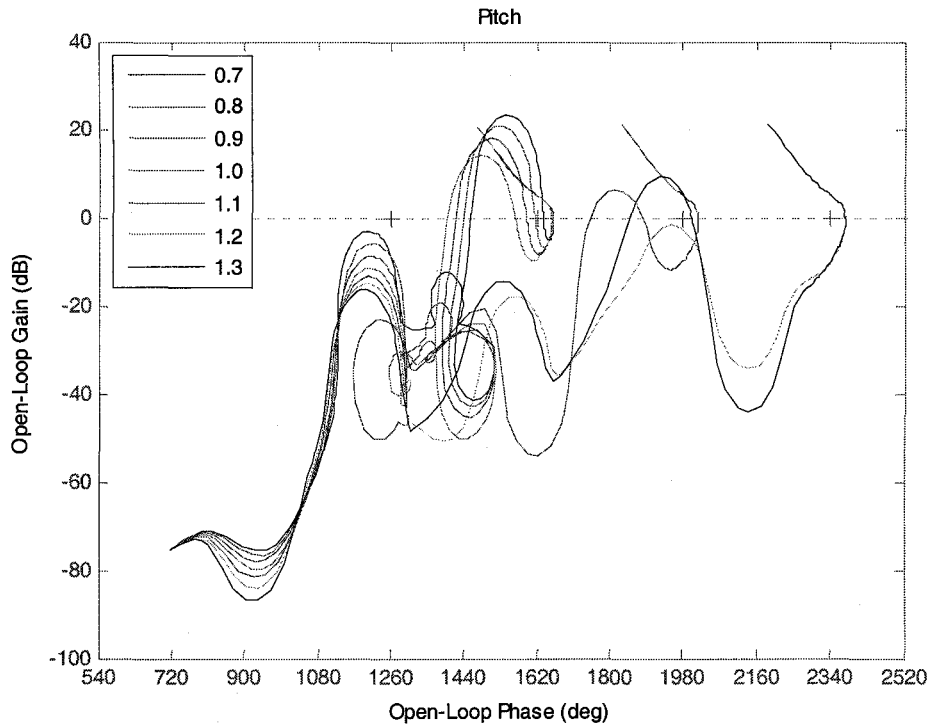

Figure 15. Nichols Results of the Linearized Pitch Channel at 55 sec. with Flex Frequency varied

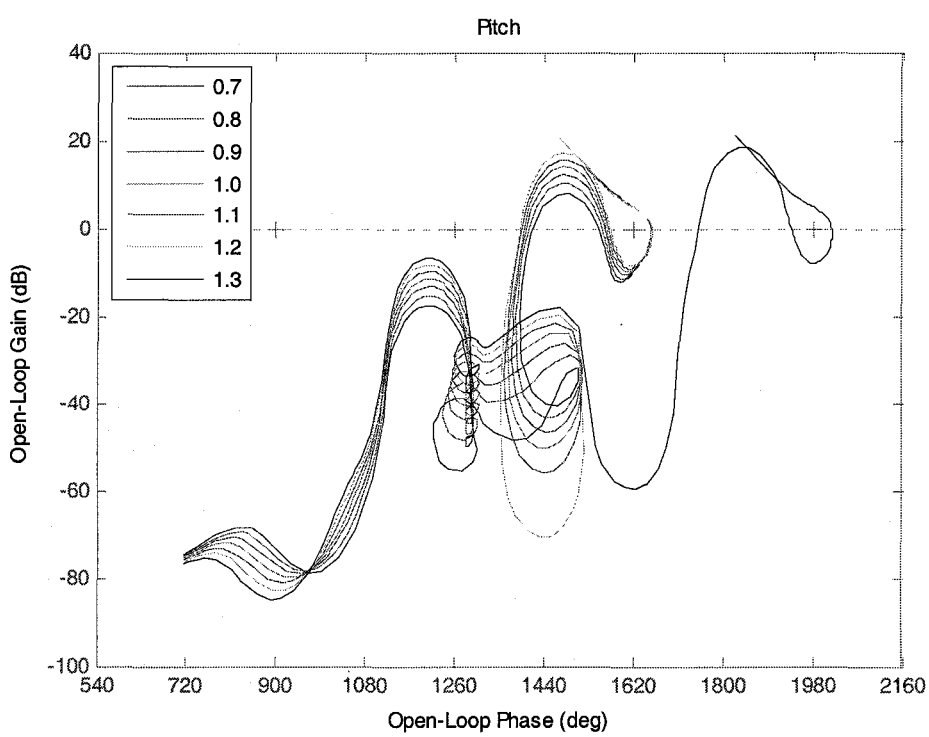

Figure 16. Nichols Results of the Linearized Pitch Channel at 55 sec. with Flex Magnitude varied

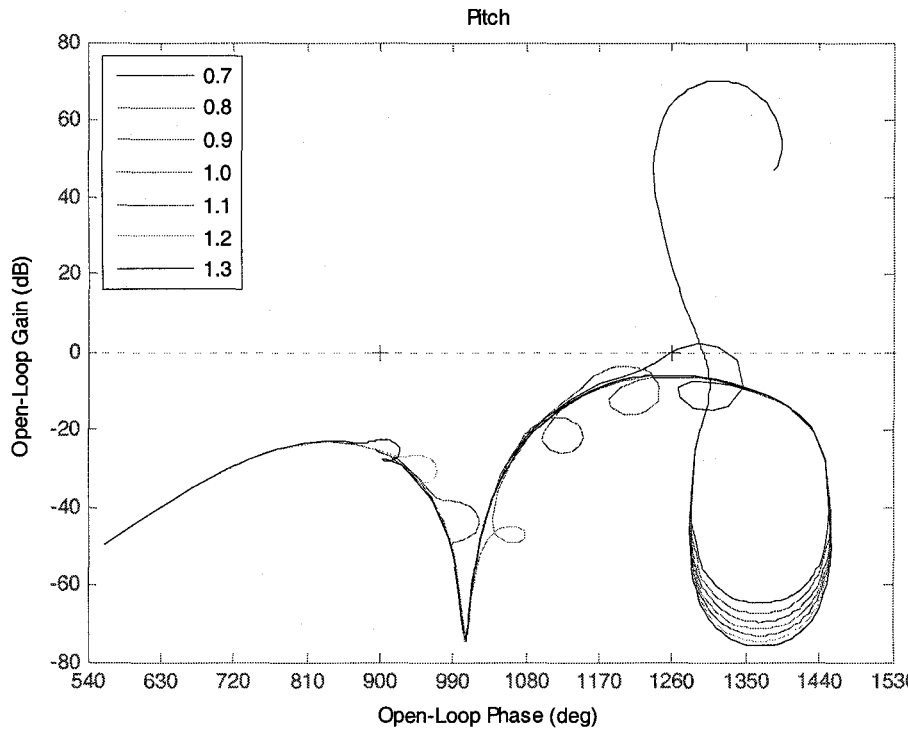

Figure 17. Nichols Results of the Linearized Pitch Channel at 130 sec. with Flex Frequency varied

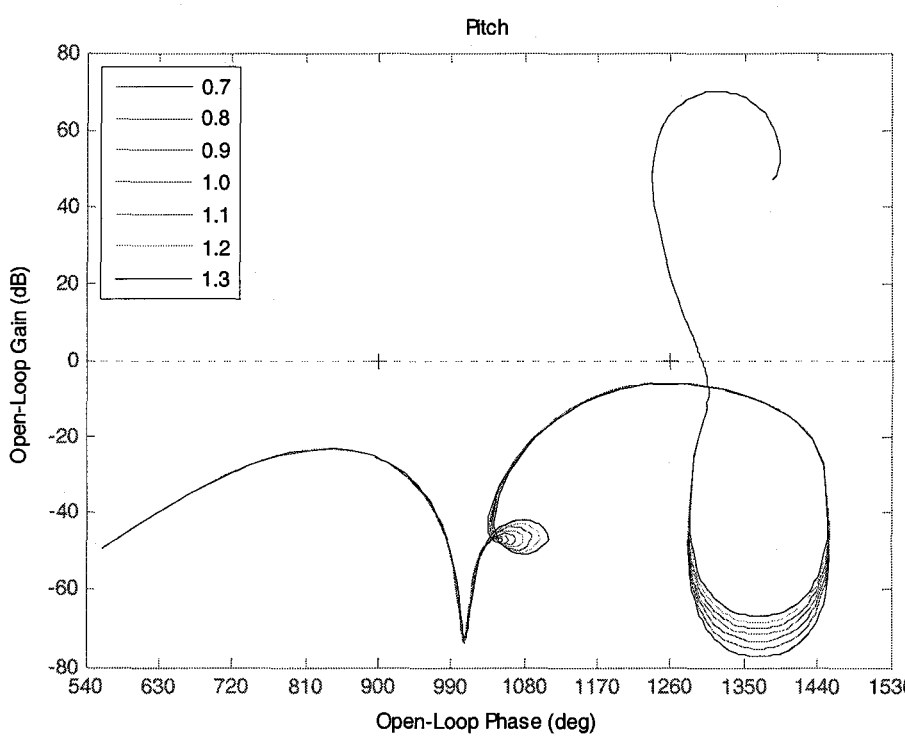

Figure 18. Nichols Results of the Linearized Pitch Channel at 130 sec. with Flex Magnitude varied

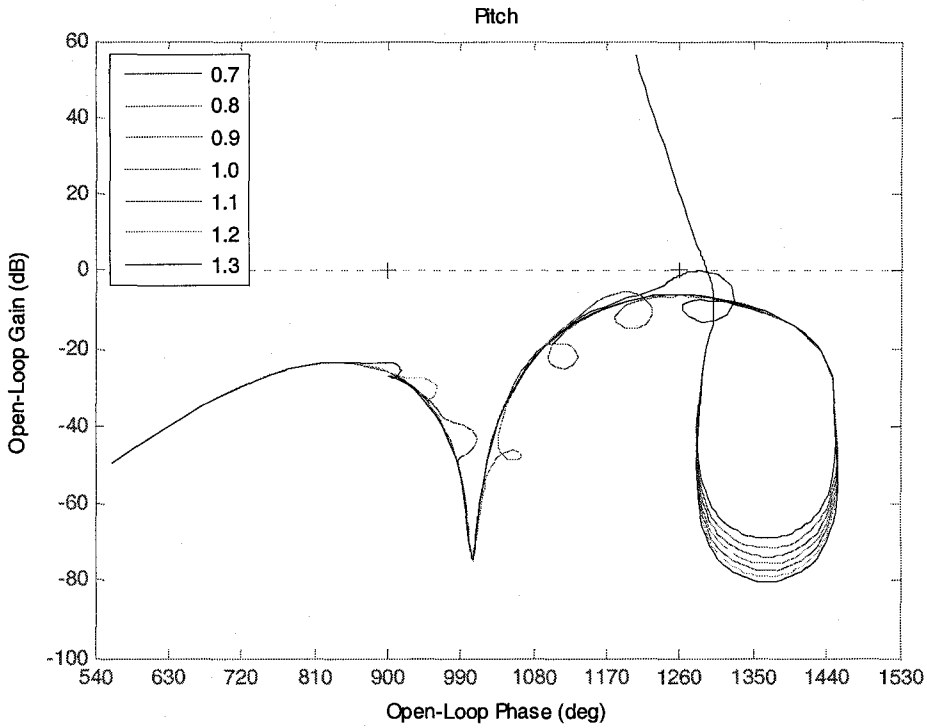

Figure 19. Nichols Results of the Linearized Pitch Channel at 180 sec. with Flex Frequency varied

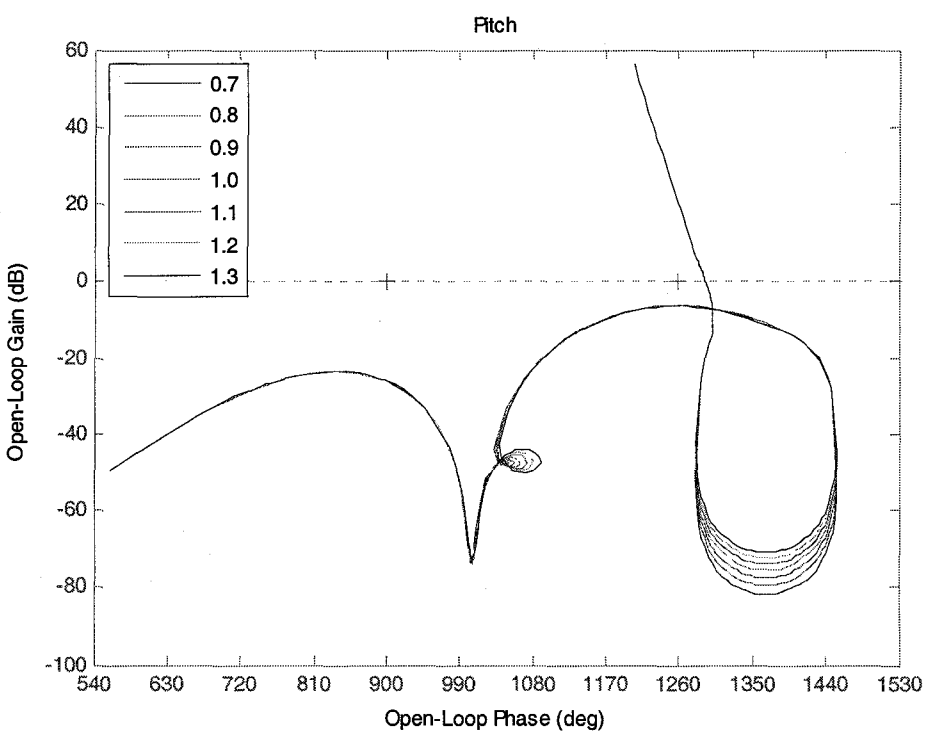

Figure 20. Nichols Results of the Linearized Pitch Channel at 180 sec. with Flex Magnitude varied

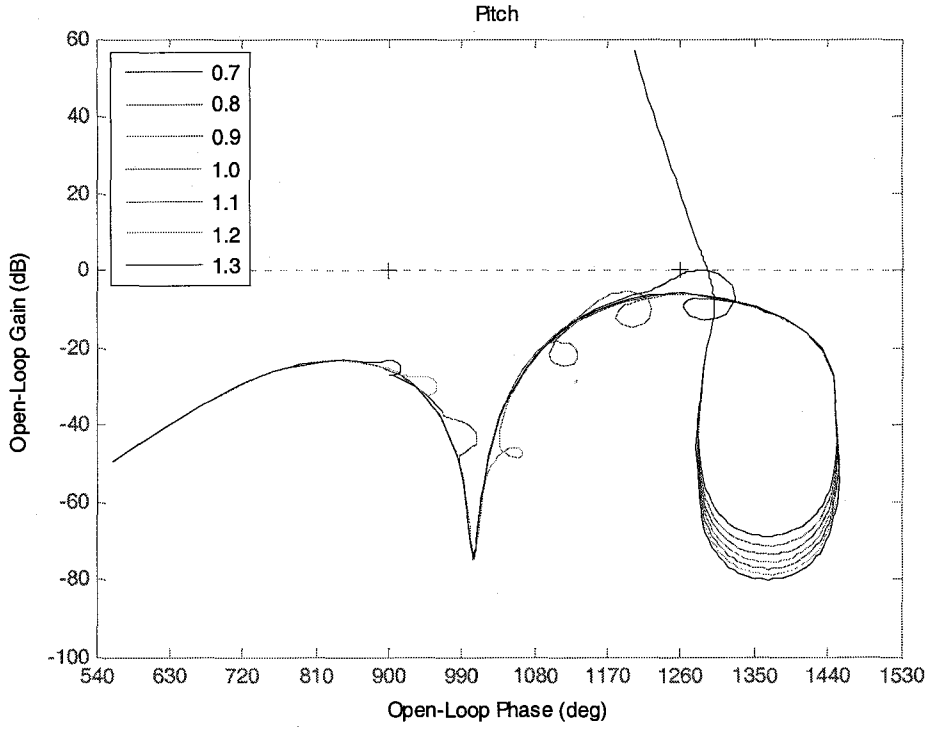

Figure 21. Nichols Results of the Linearized Pitch Channel at 230 sec. with Flex Frequency varied

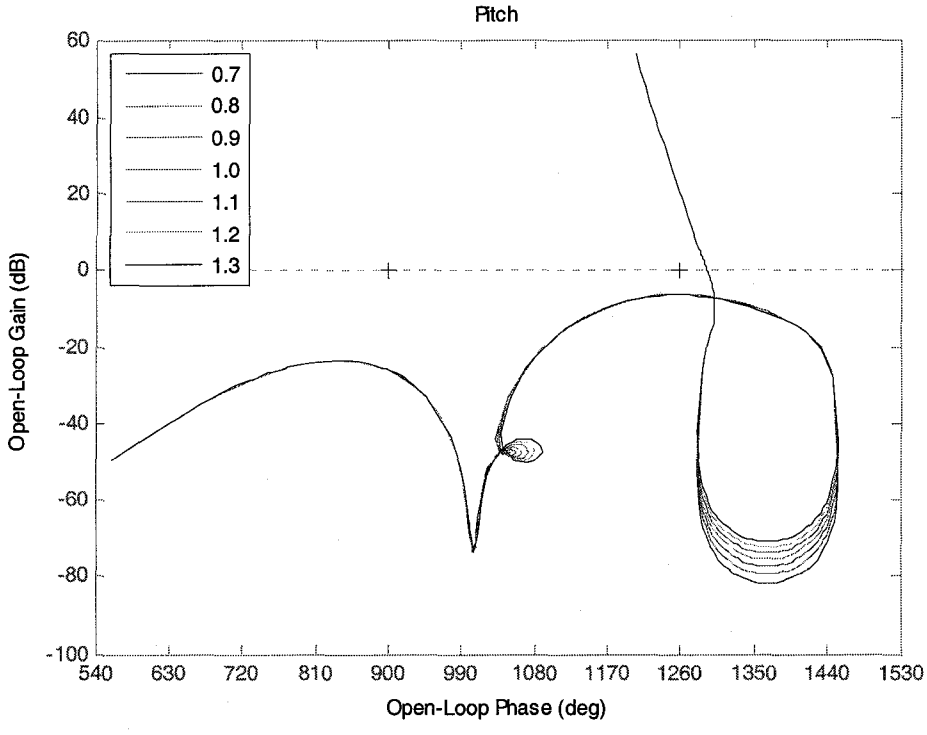

Figure 22. Nichols Results of the Linearized Pitch Channel at 230 sec. with Flex Magnitude varied

The results show that variations in flex frequency result in less stability, and even significant instabilities for the case at 55 seconds, corresponding to maximum dynamic pressure.

Conclusion

This paper presents a tool to quickly analyze time domain behavior of launch vehicles, and results are shown for the NASA CLV for both first and second stage. The versatility of the tool in enabling quick modifications and analysis on both the entire vehicle as well as the vehicle with or without any of a number of various submodels or combinations of submodels is demonstrated with the example of a quick-turnaround comparison performed to evaluate two possible control algorithms and an analysis of GVT support for Upper Stage.

Acknowledgments

This work was prepared by bd Systems, a Subsidiary of SAIC, for the NASA Marshall Space Flight Center under contract to MSFC (NNM06AA01Z). The authors would like to thank Mr. Mark West, Mr. Mark Jackson, Mr. Charles Hall, Mr. Joey Powers, and Mr. Mike Hannan from NASA MSFC for their insight and recommendations during the analyses.

References

[1] NASA's Exploration Systems Architecture Study; NASA-TM-2005-214062

[2] Greensite, A. L., Analysis and Design of Space Vehicle Flight Control Systems, Spartan Book, New York, 1970

[3] Frosch, J.A., Vallely, D.P., "Saturn AS-501/S-IC Flight Control System Design", Journal of Spacecraft, Vol. 4, No. 8, August 1967.

[4] Betts, K.M. and Johnson M.D., "CLV Control Systems Analyses", bd Systems Technical Memo TCD20060263; July 14, 2006.

[1]Betts, Kevin M., Rutherford, Chad, McDuffie, James, Johnson, Matthew D., Jackson, Mark, and Hall, Charles. "Stability Analysis of the NASA ARES I Crew Launch Vehicle Control System" *AIAA Guidance, Navigation, and Control Conference*, AIAA-2007, Hilton Head, South Carolina, Aug. 2007.