CHAPTER IX

ADAPTIVE ATTITUDE CONTROL OF THE CREW LAUNCH VEHICLE

9.1 Introduction

Classical linear control laws have been developed for the NASA's Crew Launch Vehicle (CLV). Preliminary design of the Ares-I flight control system has shown that classical control theory is sufficient to meet the stability and performance requirements of the CLV[10]. Due to the uncertain nature of a launch vehicle, imprecise knowledge of system parameters and the flexibility of the CLV rocket during its ascent phase creates concern that unforeseen instabilities may develop. Time varying factors such as fuel consumption and mass reduction, combined with a wide range of aerodynamic and dynamic interactions including payloads, propulsion, inertia, and dynamic pressure, can affect the overall dynamics of the vehicle during its flight.

Studies over the past few years have suggested that adaptive control techniques can potentially be beneficial to the flight control system in terms of robustness and safety[122]. Adaptive control may not only provide a higher level of nominal performance, but can also accommodate a greater degree of uncertainty, including active vibration suppression of uncertain flexible modes[129] and accommodation of partial failures in the flight control system. More specifically, it has been shown that adaptive control can increase the robust performance of the CLV[88]. However, the nature of an adaptive control signal raises concern for implementation on a manned launch vehicle. Since model reference adaptive control laws are inherently high bandwidth, high frequency control effort maybe generated that can destabilize unmodeled dynamics and exceed actuator capabilities. Moreover, if the uncertainty does not satisfy matching conditions, the adaptive control law can degrade tracking and even destabilize the plant.

This major disadvantage of adaptive control stems from the fact that it lacks an accepted means of quantifying the behavior of the control signal a priori. These measures are required in order to certify flight control systems of piloted and passenger bearing vehicles. Hence, most adaptive control laws require a more extensive verification and validation process due to the time varying and nonlinear manner in which its gains are adapted since the control effort could be beyond the limits of the system. From this prospective, it is highly desirable to limit the frequency content of an adaptive control signal. Classic and robust control offer natural frameworks for achieving frequency limited signals. The H_∞-NMA Architecture[90] allows one to also achieve frequency limited control signals by combining aspects from robust control theory and adaptive control theory into a single framework. This makes it possible to limit the frequency content of the adaptive control signal in an algorithmic manner.

In this chapter, attitude control of the CLV is accomplished using two decoupled H_∞-NMA state feedback architectures designed to maintain the design level of tracking performance in the presence of disturbances and parametric uncertainties. Emphasis is placed on a minimal order adaptive law in order to see if a low bandwidth, low gain, and reduced order state feedback control law can offer performance improvement. This new control architecture merges ideas from H_∞ control theory and adaptive control theory to achieve band limited control signals. This represents a different approach than previous approaches based on an high order output feedback σ-modification adaptive law[88]. To show the viability of the method, a high fidelity simulation of the CLV(called SAVANT) is used in conjunction with the actual decentralized classical control design used on the CLV to compare the nominal performance against the performance of the CLV with an augmented adaptive law. Rigid body, aerodynamics, gravity, sloshing, engine inertia effects, mass change, actuator, and elastic body models(among other things) are included in the simulation. The nominal controller consists of three independently designed controllers for yaw, pitch and roll attitude. The effect of structural modes are compensated for with using gain and phase stabilization filters in the pitch and yaw channels. The roll control law is a nonlinear bang-zero-bang design. The presence of the bang-zero-bang control law requires a special modification to the adaptive law called control hedging in order to ensure that the nonlinear nature of the roll control law if properly accounted. This has been shown to work well[88]. However, the roll control channel in the work is ignored in the adaptive design due to design restrictions from NASA. In implementation, the original control design is not modified. The adaptive control law simply augments the nominal control law. This facilitates switching the adaptive control law on in case of degraded nominal control performance.

Results examine the degree to which the nominal control design can be improved by adding an adaptive element. To measure the degree of improvement, the *Worst-on-Worst*(WoW) Monte-Carlo dispersion cases are compared. The WoW cases capture the combination of the worst possible uncertainties(i.e. dispersions) occurring simultaneously. Simulation results show that the adaptive control law always improves the performance of all of the stable WoW cases. For the WoW cases that are unstable with only the linear control law, the adaptive control algorithm is able to maintain acceptable tracking performance in almost all of the cases.

9.2 Vehicle Model and Dynamics

The Ares I CLV is a two-staged, serially connected rocket with the Orion crew exploration vehicle located at the top. The launch vehicle's first stage consists of a single, five-segment reusable solid rocket booster, and the second or upper stage is propelled by a main engine fueled with liquid oxygen and liquid hydrogen. The vehicle configuration is shown in figure 9.1.

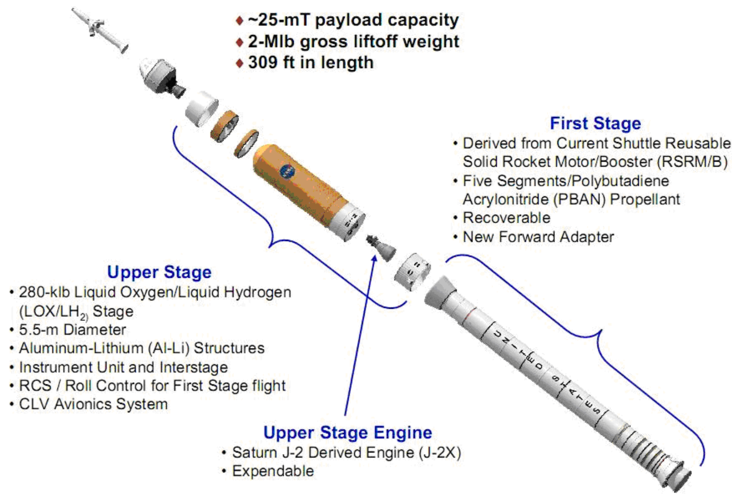


Figure 9.1: Ares I Crew Launch Vehicle

Ares I has two vital missions; lifting astronauts up to the International Space Station and achieving an in orbit rendezvous with the Ares V Earth departure stage at low Earth orbit for a mission to the moon. During the first two and a half minutes of flight, the first stage booster powers the vehicle to an altitude of about 38 miles and a speed of Mach 5.9. After its propellant is consumed, the solid rocket booster separates. The upper stage engine is then ignited and powers the Orion spacecraft. After reaching an altitude of 83 miles, the upper stage separates and the Orion spacecraft completes its trip to a circular orbit of 185 miles above the Earth using its service module propulsion system.

9.2.1 CLV Model Description

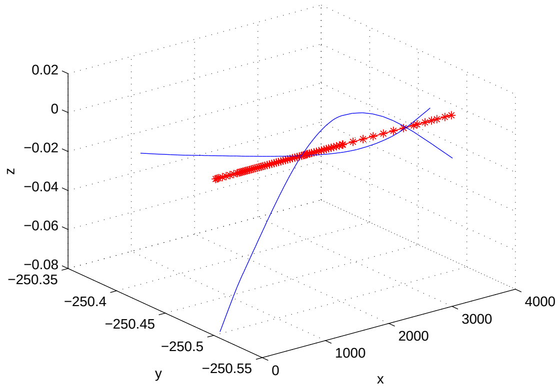
The CLV model employed in this study is called Savant and was developed in a joint effort between bD Systems and NASA Marshall. The model is described in Betts[9] and it contains simulated rigid body, aerodynamics, gravity, sloshing, engine inertia effects, mass change, actuator, and elastic body models. Many of these effects can be turned on or off in the simulation environment.

Since the Ares I CLV possesses the characteristics of a long and slender body, its flexibility should be considered in the control law design. In the structural modeling part, modal frequencies, displacement, and rotation are given from a Nastran (FEM solver) solution and is used to model the interaction effects between the vehicle flexibility and the other dynamic models. Lateral vibration is the dominate vibration mode. It is important to consider the effect of this vibration in control system design due to the modal frequencies being near the control bandwidth. The vehicle's elastic motion can be conveniently expressed in terms of frequencies and mode shapes of a free-free beam structure. Because of the axial symmetry of Ares I launch vehicle, two identical modes exist in the lateral bending. Table 9.1 gives a summary of the approximate dominate bending mode frequencies at launch. The actual modal frequencies are time varying and change significantly through out ascent. Figure 9.2 shows the dominate vehicle mode shapes.

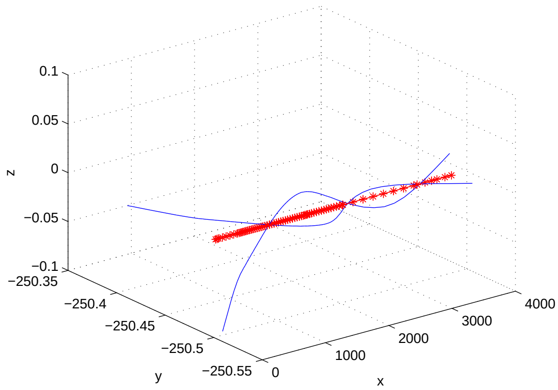
Table 9.1: Bending Frequency Table

1st Bending Frequency	2nd Bending Frequency	3rd Bending Frequency
6.0 [rad/sec]	14.2 [rad/sec]	27.2 [rad/sec]

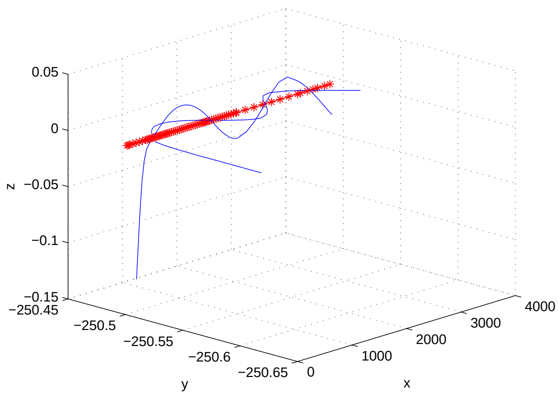
The aerodynamic model contains three parts; the environment model, the aerodynamic coefficients, and the force and moment generation algorithm. The time history of the mass properties including the total vehicle weight and the center of gravity location is stored in the simulation model in the form of look-up tables. Multiple gimbal actuator models are available for simulating actuator limits including a 3rd order model or a high fidelity simplex model. The fuel sloshing model simulates the effect of fuel sloshing as point masses connected to the rocket body as shown in figure 9.3 to the LOX and LH2 upper stage tanks by a spring and damper in the lateral (Y, Z) direction. The vertical position of slosh point mass is function of the liquid level in the tank. Vehicle separation is modeled as an instantaneous loss of mass. The atmosphere is simulated using the US76 atmosphere model and the gravity model



(a) First Bending Mode



(b) Second Bending Mode



(c) Third Bending Mode

Figure 9.2: Visualization of the first three structural mode shapes

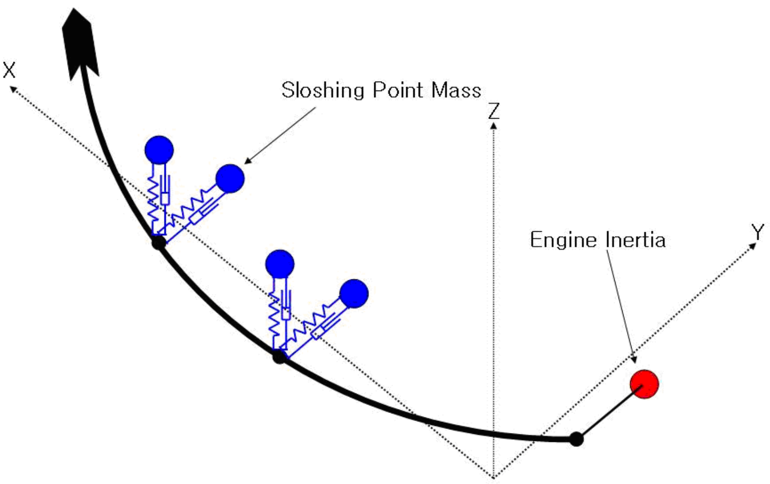


Figure 9.3: Schematic of elastic vehicle with sloshing point mass and engine inertia

used includes Earth oblateness effects without considering abnormality or vertical deflection data.

9.2.2 Nominal Control Law

The nominal control law tracks quaternion guidance commands. At each time instant, the quaternion command and the vehicle's attitude quaternion are used to generate an attitude error signal. These errors are suppressed using two gain scheduled PID control laws for the pitch and yaw degrees of freedom and a phase plane roll controller for the roll degree of freedom. One restriction of the roll RCS is that the actuator only fires when the roll error of the CLV is greater than a defined parameter (a bang-zero-bang control law). Each axis is assumed independent so each control channel is designed separately. The effects of structural modes are gain and phase stabilized using a combination of gain scheduled low pass and notch filters. Gains are scheduled based on the altitude. The nominal control architecture is shown in figure 9.4.

The error signal to the nominal control law is generated based on a quaternion command signal. A comparison between the command and the current vehicle quaternion state generates an error angle state that is used for feedback. This error angle

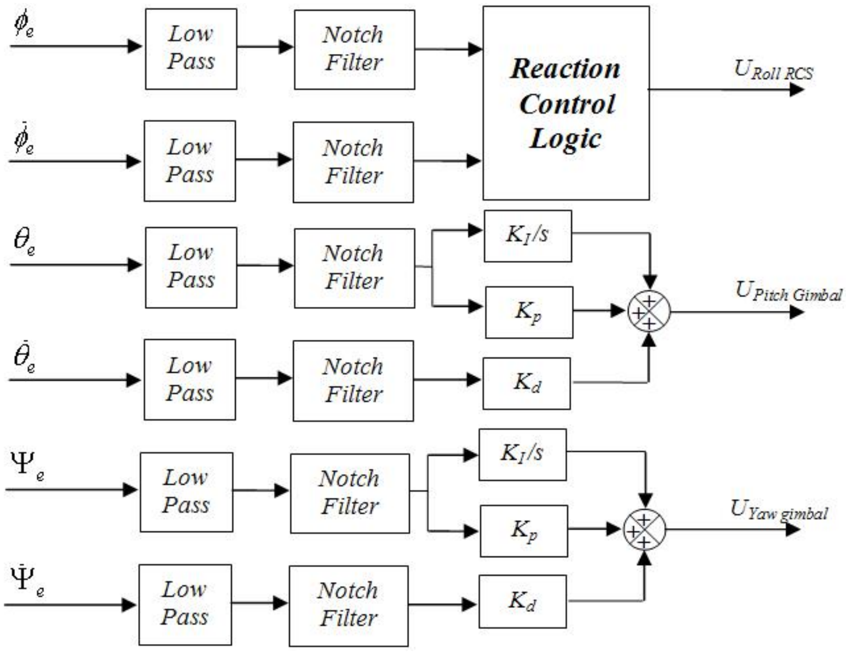


Figure 9.4: Diagram of nominal control system

state is the incremental Euler angle rotation from the current system state to the current command. To define this error angle signal, suppose that there are two frames, frame 1 and frame 2. One can describe the rotation from frame 1 to frame 2 in terms of an axis and an angle to rotate around that axis. This representation can be used to define the quaternions used in this chapter. The following definition of the quaternion vector based on an axis-angle formulation is used.

$$q_{axis-angle} = \begin{bmatrix} cos(\xi/2) \\ a_x sin(\xi/2) \\ a_y sin(\xi/2) \\ a_z sin(\xi/2) \end{bmatrix} \tag{9.1}$$

where a is a unit vector in Cartesian space such that a rotation, ξ, about a will rotate frame 1 to frame 2. It is assumed that the quaternion has unit length such that $\|q_{axis-angle}\| = 1$.

In order to formulate an error signal based on quaternions, an error angle vector based on the deviation between the command and the system state vector is computed. Suppose the quaternion command is represented by q_c and the system quaternion state is given by q. Then the quaternion representing the error between these two quaternion vectors is given by[63]

$$q_e = q_c^{-1} \otimes q = \begin{bmatrix} q_1 q_{c_1} + q_2 q_{c_2} + q_3 q_{c_3} + q_4 q_{c_4} \\ q_2 q_{c_1} - q_1 q_{c_2} - q_4 q_{c_3} + q_3 q_{c_4} \\ q_3 q_{c_1} + q_4 q_{c_2} - q_1 q_{c_3} - q_2 q_{c_4} \\ q_4 q_{c_1} - q_3 q_{c_2} + q_2 q_{c_3} - q_1 q_{c_4} \end{bmatrix} \tag{9.2}$$

Applying a quaternion-to-Euler angle transformation for a z-axis, y-axis, and x-axis sequence, one can compute the this error vector in terms of Euler angles:

$$\begin{aligned} \phi_e &= arctan\left[\frac{2\left(q_{e3}q_{e4} + q_{e1}q_{e2}\right)}{2\left(q_{e1}q_{e1} + q_{e4}q_{e4}\right) - 1}\right] \\ \theta_e &= arcsin\left[2\left(q_{e1}q_{e3} - q_{e2}q_{e4}\right)\right] \\ \psi_e &= arctan\left[\frac{2\left(q_{e2}q_{e3} + q_{e1}q_{e4}\right)}{2\left(q_{e1}q_{e1} + q_{e2}q_{e2}\right) - 1}\right] \end{aligned}$$

If the error between q_c and q is small, the sign of q_{e1} can be taken as positive. In this case, one can apply a small angle approximation. For quaternions, this implies that

$$\begin{aligned} q_{e1} &\approx 1 \\ \|q_{e2}\| &\ll 1 \\ \|q_{e3}\| &\ll 1 \\ \|q_{e4}\| &\ll 1 \end{aligned} \tag{9.3}$$

This can be seen be simply examining the axis-angle formulation of a quaternion. With this assumption, one can approximate the error angles as

$$\begin{aligned} \phi_e &\approx 2 q_{e1} q_{e2} \\ \theta_e &\approx 2 q_{e1} q_{e3} \\ \psi_e &\approx 2 q_{e1} q_{e4} \end{aligned} \tag{9.4}$$

This error is fed into the nominal control law along with the components of the body angular rate vector, ω, to compute the control signals in the roll, pitch, and yaw axes.

9.2.3 H_∞-NMA Architecture Formulation

This section formulates a low-order decoupled H_∞-NMA architecture for the CLV. The dynamics for the adaptive law design can be derived by considering the general attitude control problem. In this problem, the plant dynamics can be approximately captured as

$$\dot{q} = \Gamma(q, \omega)$$

$$\dot{x} = \sigma(q, x, \omega, \delta)$$

$$\dot{\omega} = f(q, x, \omega) + B\delta$$

where q is the quaternion representing the inertial attitude, ω is the angular velocity, x are the position and velocity dynamics, B is a diagonal control effectiveness matrix, δ is the control action, and

$$\dot{q} = \Gamma(q, \omega) = \frac{1}{2} \begin{bmatrix} -q_2 & -q_3 & -q_4 \\ q_1 & -q_4 & q_3 \\ q_4 & q_1 & -q_2 \\ -q_3 & q_2 & q_1 \end{bmatrix} \omega = \frac{1}{2}\Omega_q \omega \tag{9.5}$$

The control effectiveness matrix is assumed diagonal due to the restriction that the control law be decoupled. If one wanted to couple the adaptive control design, this restriction could be removed.

The quaternion guidance command to the attitude control system is open loop[46] and slowly varying. Since, the quaternion is slowly varying, it is assumed that

$$\dot{q}_c \approx 0$$

This assumption allows one to formulate the CLV attitude control problem as a stabilization problem. To this end, the attitude dynamics are reexpressed relative to the slowly varying attitude command, q_c. The quaternion attitude error was expressed in equation (9.2) as

$$q_e = q_c^{-1} \otimes q$$

Since $\dot{q}_c \approx 0$, the time derivative of q_e is given by

$$\begin{aligned}\dot{q}_e &= q_c^{-1} \otimes \dot{q} \\ &= \frac{1}{2} q_c^{-1} \otimes \Omega_q \omega\end{aligned}$$

The quaternion attitude can be expressed as

$$q = q_c \otimes q_e$$

Hence,

$$\dot{q}_e = \frac{1}{2} q_c^{-1} \otimes \Omega_{q_c \otimes q_e} \omega$$

Using quaternion algebra,

$$q_c \otimes q_e = \begin{bmatrix} q_{c_1} q_{e_1} - q_{c_2} q_{e_2} - q_{c_3} q_{e_3} - q_{c_4} q_{e_4} \\ q_{c_1} q_{e_2} + q_{c_2} q_{e_1} + q_{c_3} q_{e_4} - q_{c_4} q_{e_3} \\ q_{c_1} q_{e_3} - q_{c_2} q_{e_4} + q_{c_3} q_{e_1} + q_{c_4} q_{e_2} \\ q_{c_1} q_{e_4} + q_{c_2} q_{e_3} - q_{c_3} q_{e_2} + q_{c_4} q_{e_1} \end{bmatrix}$$

where q_{c_i} is the i^{th} element of q_c and q_{e_i} is the i^{th} element of q_e. From equation (9.5), $\Omega_{q_c \otimes q_e}$ is

$$\Omega_{q_c \otimes q_e} = \begin{bmatrix} -(q_c \otimes q_e)_2 & -(q_c \otimes q_e)_3 & -(q_c \otimes q_e)_4 \\ (q_c \otimes q_e)_1 & -(q_c \otimes q_e)_4 & (q_c \otimes q_e)_3 \\ (q_c \otimes q_e)_4 & (q_c \otimes q_e)_1 & -(q_c \otimes q_e)_2 \\ -(q_c \otimes q_e)_3 & (q_c \otimes q_e)_2 & (q_c \otimes q_e)_1 \end{bmatrix}$$

where $(q_c \otimes q_e)_i$ is the i^{th} element of $q_c \otimes q_e$. Since, q_c is a unit quaternion, q_c^{-1} is given by

$$q_c^{-1} = \begin{bmatrix} q_{c_1} \\ -q_{c_2} \\ -q_{c_3} \\ -q_{c_4} \end{bmatrix}$$

From this, after some algebra, the q_e dynamics in equation (9.6) can be expressed as

$$\dot{q}_e = \frac{1}{2}\begin{bmatrix} -q_{e2} & -q_{e3} & -q_{e4} \\ q_{e1} & -q_{e4} & q_{e3} \\ q_{e4} & q_{e1} & -q_{e2} \\ -q_{e3} & q_{e2} & q_{e1} \end{bmatrix}\omega = \frac{1}{2}\Omega_{q_e}\omega \tag{9.6}$$

Assuming that ω remains *small* and applying the small angle properties of q_e in equation (9.3), the q_e dynamics can be approximated as

$$\dot{q}_e \approx \frac{1}{2}\begin{bmatrix} 0 \\ \omega \end{bmatrix}$$

From the definition of the error angles in equation (9.4),

$$\begin{bmatrix} \dot{\phi}_e \\ \dot{\theta}_e \\ \dot{\psi}_e \end{bmatrix} \approx \omega \tag{9.7}$$

Let the state vector for the H_∞-NMA architecture in figure 9.5 be defined as

$$\bar{e} = \begin{bmatrix} \phi_e \\ \theta_e \\ \psi_e \\ \omega \end{bmatrix} \tag{9.8}$$

Using this definition, the system dynamics can be approximated as

$$\dot{\bar{e}} = \begin{bmatrix} \omega \\ f_c(q, x, \omega) + B\delta \end{bmatrix} \tag{9.9}$$

where the remaining vehicle dynamics (position and velocity) are unmodified in the form as previously defined as

$$\dot{x} = \sigma(q, x, \omega, \delta)$$

With q_c regarded as nearly constant, the error dynamics in equation (9.9) can be rewritten as

$$\dot{\bar{e}} = \begin{bmatrix} \omega \\ f_c(q_e, x, \omega) \end{bmatrix} + \bar{B}\delta \tag{9.10}$$

where $f_c(\cdot)$ is the equivalent form of $f(\cdot)$ as a function of q_e instead of q and

$$\bar{B} = \begin{bmatrix} 0 \\ B \end{bmatrix}$$

The total augmented control effort is defined as

$$\delta(t) = \delta_n(t) - \delta_{ad}(t) \tag{9.11}$$

where $\delta_n(t)$ is the nominal control output and $\delta_{ad}(t)$ is the augmented adaptive signal. It is assumed that the linear control law was designed to achieve a second order response in each control channel. Based on the definition in equation (9.11), it is assumed that the application of the nominal control law, has the effect of creating the following approximate form for the error dynamics in equation (9.10)

$$\dot{\bar{e}}(t) = A_m\bar{e}(t) - \bar{B}\delta_{ad}(t) + \bar{B}\Delta(\bar{e}(t)) \tag{9.12}$$

where $\Delta(\bar{e}(t))$ is the modeling error that exists between the desired dynamics and the actual dynamics and

$$A_m = \begin{bmatrix} 0 & I \\ -\bar{K}_p & -\bar{K}_d \end{bmatrix}$$

$$\bar{K}_p = \begin{bmatrix} k_p & 0 & 0 \\ 0 & k_q & 0 \\ 0 & 0 & k_r \end{bmatrix} \quad \text{and} \quad \bar{K}_d = \begin{bmatrix} b_p & 0 & 0 \\ 0 & b_q & 0 \\ 0 & 0 & b_r \end{bmatrix}$$

k_i and b_i are chosen to match the damping and stiffness associated with the desired second order response of the gain scheduled CLV control system design. These assumed dynamics match the form of the dynamics in Chapter 4 in equation (4.7) with $B_m = 0$. Hence, one could use these dynamics to formulate a coupled H_∞-NMA architecture based on Chapter 4 for the CLV.

In order to maintain a decoupled adaptive design, it is assumed that $\Delta(e_c(t))$ is a diagonal uncertainly of the form

$$\Delta(\bar{e}(t)) = \begin{bmatrix} \Delta_1(\phi_e, p) & 0 & 0 \\ 0 & \Delta_2(\theta_e, q) & 0 \\ 0 & 0 & \Delta_3(\psi_e, r) \end{bmatrix} \tag{9.13}$$

This allows the dynamics of each control channel to be expressed as

$$\dot{x}_i(t) = A_{m_i} x_i(t) - B_i \delta_{ad_i}(t) + B_i \Delta_i(x_i) \tag{9.14}$$

where i represents the i^{th} control channel, $i = 1$ corresponds to the roll channel, $i = 2$ corresponds to the pitch channel, $i = 3$ corresponds to the yaw channel, $x_1 = [\phi_e \quad p]^T$, $x_2 = [\theta_e \quad q]^T$, $x_3 = [\psi_e \quad r]^T$, $B_i = [0 \quad B(i,i)]^T$, and A_{m_i} is defined as

$$A_{m_i} = \begin{bmatrix} 0 & 1 \\ -k_i & -b_i \end{bmatrix} \tag{9.15}$$

Only the pitch and yaw channel are augmented with an adaptive control law. The state emulator for these adaptive laws can be thought of as an error state emulator because the state vector is based on the attitude command error, q_e, and the angular velocity, ω. Note that there is no reference command in the dynamics in equation (9.14). This implies that, unless there is an initial attitude error and/or an angular rate, the reference model does not need to be implemented. In this case, the H_∞-NMA architecture simplifies. Comparing with figure 4.1, the architecture in each channel reduces as shown in figure 9.5. In this case, from the error definitions in Chapter 4, the state emulator error for the i^{th} channel is $\hat{e}_i(t) = x_i(t) - \hat{x}_i(t)$ and the state emulator tracking error for the i^{th} channel is $e_i(t) = -\hat{x}_i(t)$. In this figure, K_x represents the nominal control system for the i^{th} channel which feeds back on the same error state, $x_i(t)$.

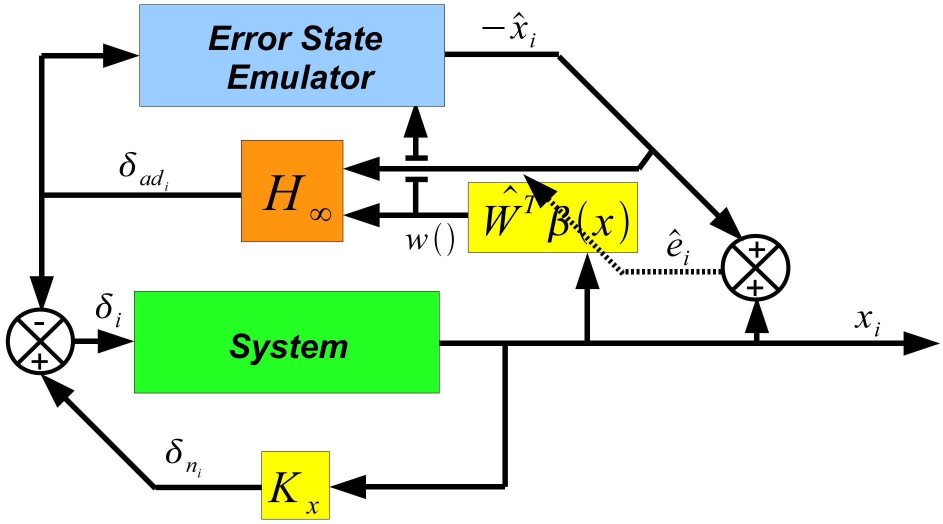


Figure 9.5: H_∞-NMA architecture simulation diagram for the CLV.

9.3 Simulation Results

The adaptive system and the nominal control system is implemented in discrete time with an update rate of 50 Hz. All system latency and actuator limitations are modelled. The actuator command from the control system is in terms of a roll RCS command, a pitch angle command to the gimbal, and a yaw angle command to the gimbal. However, the actual gimbal control signal is a *tilt* and *rock* command. The tilt and rock command is computed based on the following equation.

$$\begin{bmatrix} u_{tilt} \\ u_{rock} \end{bmatrix} = \begin{bmatrix} cos(\phi_{TR}) & sin(\phi_{TR}) \\ -sin(\phi_{TR}) & cos(\phi_{TR}) \end{bmatrix} \begin{bmatrix} \delta_{pitch} \\ \delta_{yaw} \end{bmatrix}$$

where ϕ_{TR} is rotation angle of the tilt-rock gimbal, u_{tilt} is the tilt angle command, u_{rock} is the rock angle command, δ_{pitch} is the control system gimbal pitch command, and δ_{yaw} is the control system gimbal yaw command.

The H_∞-NMA design was meant to be minimum complexity. Therefore the vector $\beta(x_i)$ was chosen as

$$\beta(x_i) = \begin{bmatrix} a \\ x_i \end{bmatrix}$$

where a is a user defined constant and x_i is the corresponding state vector of the i^{th} channel in equation (9.14). It was assumed that the desired response is equivalent to choosing a damping ratio of $\zeta = 0.707$ and an undamped natural frequency of $\omega_n = 5$ rad/sec in both the pitch and yaw control channel. This implies that the stiffness and damping parameters of A_{m_i} should be defined by

$$k_i = \omega_n^2 \quad \text{and} \quad b_i = 2\zeta\omega_n$$

The adaptive gain was simply set to $\Gamma = 30$. The linear H_∞ design was augmented with an integrator. The augmented state vector used for each design was

$$\bar{x} = \begin{bmatrix} x_{int_i} \\ x_i \end{bmatrix}$$

where x_{int_i} is the integrator state. Using this augmented state vector, the design strategy in section 4.4 was followed. In each design, the design matrices $Q^{\frac{1}{2}}$ and $R^{\frac{1}{2}}$ were chosen as

$$Q^{\frac{1}{2}} = I \quad \text{and} \quad R^{\frac{1}{2}} = 0.025$$

This selection of weighting matrices suggests that the tilt-rock actuator should have at least a 3 Hz bandwidth.

The simulation results focus on the mean international space station mission and examine the degree to which the nominal control design can be improved by adding a decoupled state feedback adaptive element described in the previous sections. To measure this degree of improvement, tracking performance is compared between the baseline control performance and the H_∞-NMA architecture performance for the *Worst-on-Worst*(WoW) Monte-Carlo dispersion cases. The WoW cases capture the combination of the worst possible uncertainties(i.e. dispersions) occurring simultaneously. Simulations show that the adaptive control law always improves the performance of all of the stable WoW cases. Results show that system tracking errors and measures of structural stress are reduced. In many examples of adaptive control law implementation, the control laws exhibit increased control effort. This could be detrimental to the performance of a launch vehicle. However, in all of the WoW cases, the total control activity between the adaptive cases and the linear control cases remain approximately the same(as measured by duty cycle and duty cycle rate). Moreover, in the case of rock duty cycle rate, control activity is generally reduced. For the WoW cases that are unstable with only the linear control law, the adaptive control algorithm is able to maintain acceptable tracking performance in almost all of the cases while control effort remains consistent with other dispersion cases.

In comparing performance of the adaptive control system algorithm with the baseline flight control system, parameters that characterize attitude tracking, use of effectors, loads, and errors at the time of first stage separation were captured from the simulations and evaluated. Performance emphasis is on loads and attitude control. These metrics are listed below.

- Total Q-Alpha - Square root of the sum of Q-Alpha (aerodynamic pressure multiplied by angle of attack) and Q-Beta (aerodynamic pressure multiplied by side slip angle). Reduction of this is desirable in all cases since it represents aerodynamic loading on the vehicle, therefore it is heavily weighted.

- Attitude errors - roll, pitch, and yaw errors (command minus sensed) before filtering. This is coupled with total Q-Alpha during high Q since guidance is essentially commanding zero aerodynamic angles in this region of flight. Due to thrust vector dispersions, pitch and yaw couple into roll and affects RCS propellant consumed.

- Total nozzle gimbal angle - The maximum value should be maintained below the specified value due to hardware capability. This metric is heavily weighted since exceeding capability could result in loss of the vehicle.
- Gimbal duty cycle - area under the total nozzle angle curve as defined by

$$\text{Duty Cycle}(t) = \int_0^t (\text{Nozzle Gimbal Angle})\, dt$$

  This is not heavily weighted, recognizing the fact that utilizing the effectors more aggressively may be necessary to achieve better results.
- Gimbal rate duty cycle - number of sign changes in both the rock and tilt actuator rates. This metric is needed to evaluate actuator chatter. It is not heavily weighted.
- Body rates (truth, not measured) at separation. Roll rate is not as important as pitch and yaw. Large rates can cause interference between first stage and interstage hardware. Pitch and yaw are weighted more heavily than roll.

To capture these factors, a scoring metric was developed. Success was judged based upon increasing the maximum values of the performance metrics. Simulations were made with the baseline flight control system and the adaptive control system using the same set of dispersions. For each simulation, a scaling, $S(i)$, for each metric is computed. This value is given by

$$S(i) = 1 + \frac{B(i) - M(i)}{B(i)} \tag{9.16}$$

This expression is based on a percentage difference in the bounding values, $B(i)$, and the maximum value of each metric during a simulation run, $M(i)$. Below are bounding values and the weights for the performance metrics used to score each of the simulations(values were suggested by NASA).

The score for each metric is

$$\text{Score}(i) = W(i)S(i) \tag{9.17}$$

and the total score for each simulation is

$$\text{Total Score} = \sum_i \text{Score}(i) \tag{9.18}$$

The metric score has improved if the total score increases.

The H_∞-NMA adaptive control architecture improves the metric score for all of the Worst-on-Worst cases in which the baseline control law is stable. Figures 9.6 and 9.7 show the raw metric scores for the H_∞-NMA architecture and the baseline control law. There are 100 WoW cases. Case numbers without data represent a case where the baseline control law is unstable (no comparison can be made). A case is considered unstable if the roll error exceeds 10°, the pitch error exceeds 5°, or the yaw error exceeds 5°. For the baseline linear control law, WoW cases 7, 10, 46, 47, and 82 are unstable. For the adaptive control law, only cases 46 and 82 are unstable. Figure 9.8 shows the percentage improvement in the performance metric

Table 9.2: CLV Metric Bounding Values

Metric, $M(i)$	Bounding Value, $B(i)$
Total Q-alpha	5500 $\frac{psf}{\circ}$
Attitude errors (absolute values)	Roll-20°, Pitch-3°, and Yaw-3°
Total nozzle gimbal angle	4°
Gimbal duty cycle	193 $\frac{\circ}{sec}$
Gimbal rate duty cycle	Rock-30 cycles, Tilt-30 cycles
Body rates at separation (absolute values)	Roll-60 $\frac{\circ}{sec}$, Pitch-5 $\frac{\circ}{sec}$, Yaw-5 $\frac{\circ}{sec}$

Table 9.3: CLV Metric Weights

Metric, $M(i)$	Weight, $W(i)$
Total Q-alpha	5
Attitude errors (absolute values)	Roll-2, Pitch-3, and Yaw-3
Total nozzle gimbal angle	5
Gimbal duty cycle	2
Gimbal rate duty cycle	Rock-2, Tilt-2
Body rates at separation (absolute values)	Roll-2, Pitch-5, Yaw-5

for the adaptive control system compared to the corresponding baseline control case. Figure 9.9 shows the percentage improvement in the performance metric when the first 20 seconds is excluded from the metric calculation.

Next the time history of each WoW dispersion is compared. In all of the following fiugres, the unstable cases for both the adaptive and baseline control law are not shown for clarity. Figures 9.10 - 9.12 show the attitude error for each stable baseline and adaptive control case. In each case, roll error between each baseline and adaptive case stays about the same but this isn't surprising because the roll control law is not augmented with an adaptive element. The peaks in the pitch and yaw error is generally reduced using the adaptive controller. Figures 9.13 and 9.14 show the angle of attack and sideslip respectively. These plots show that peaks in both quantities are generally reduced. Figures 9.15 - 9.17 show plots of variables related to structural stress. Once again, in each case, peaks are reduced. Figures 9.18 and 9.19 show the rock and tilt command respectively. In these figures, the total control effort remains approximately the same. This is good because in many applications of adaptive control, control activity tends to increase relative to the linear baseline design. Figures 9.20 - 9.23 show the duty cycle and duty cycle rate for the tilt and rock actuators during each simulation run. While the measures of duty cycle remain approximately the same, it is notable that the tilt and rock duty cycle rate decreases as a general trend.

9.4 Conclusions

An H_∞-NMA architecture for the Crew Launch Vehicle was developed in a state feedback setting. The minimal complexity adaptive law was shown to improve base line performance relative to a performance metric based on Crew Launch Vehicle design requirements for all most all of the Worst-on-Worst dispersion cases. The adaptive law was able to maintain stability for some dispersions that are unstable with the nominal control law. Due to the nature of the H_∞-NMA architecture, the augmented adaptive control signal has low bandwidth which is a great benefit for a manned launch vehicle.

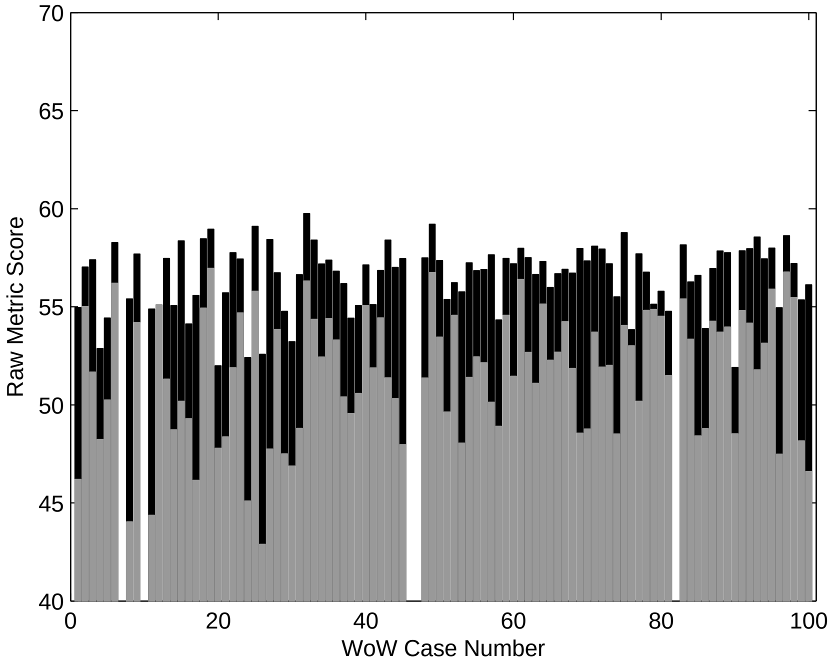


Figure 9.6: Raw metric score values for the stable baseline control WoW cases. Black is the adaptive control law and grey is the baseline control law.

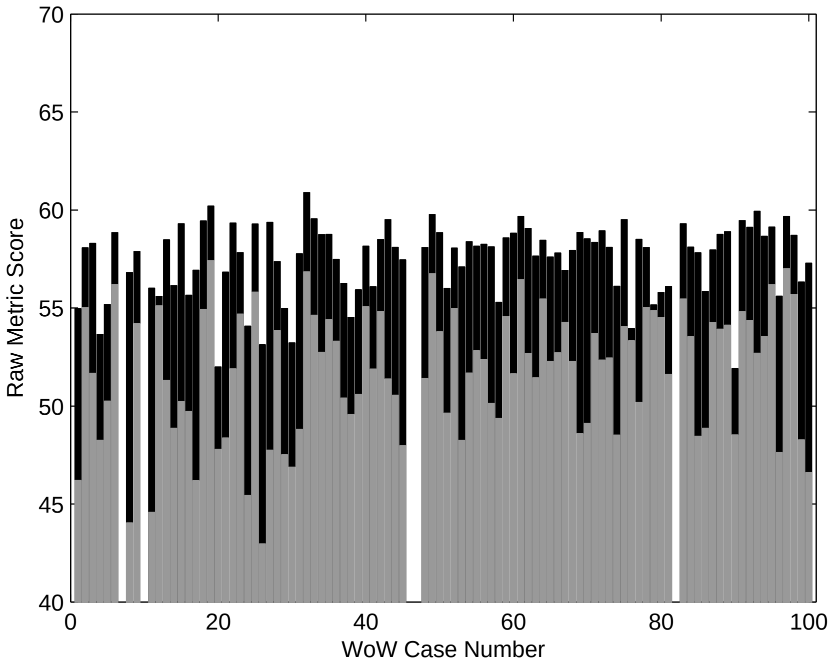


Figure 9.7: Raw metric score values after 20 seconds. Black is the adaptive control law and grey is the baseline control law.

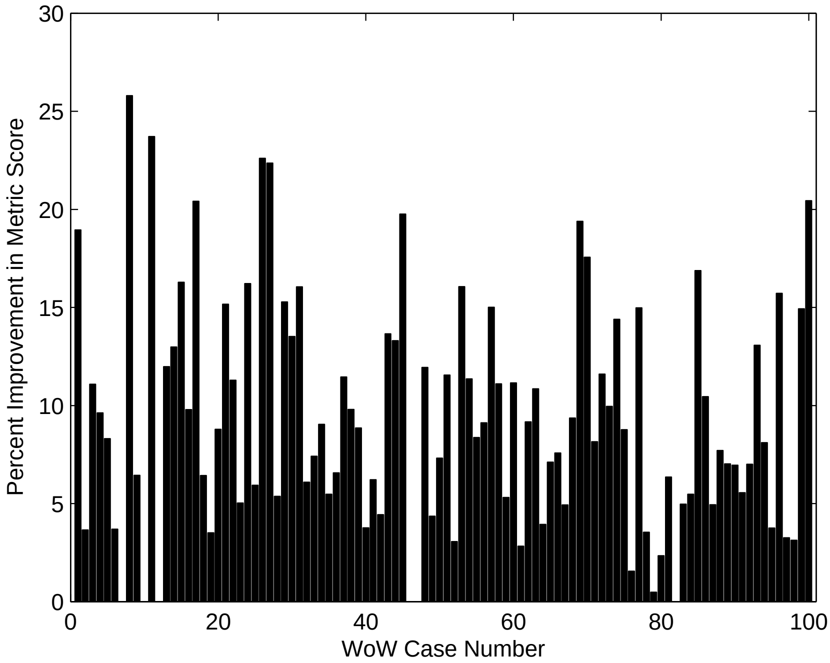


Figure 9.8: Summary of metric score improvements for stable WoW cases.

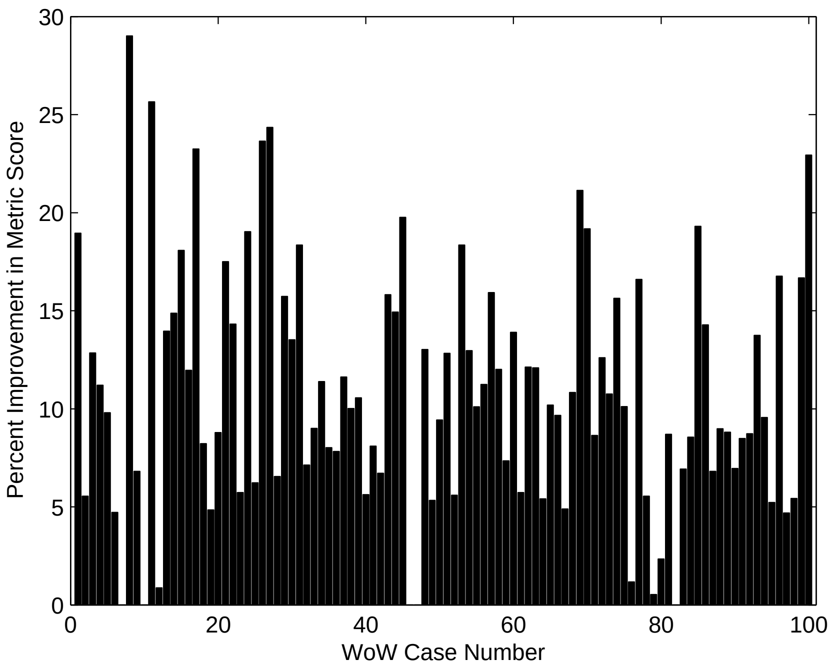


Figure 9.9: Summary of metric score improvements for stable WoW cases after the first 20 seconds.

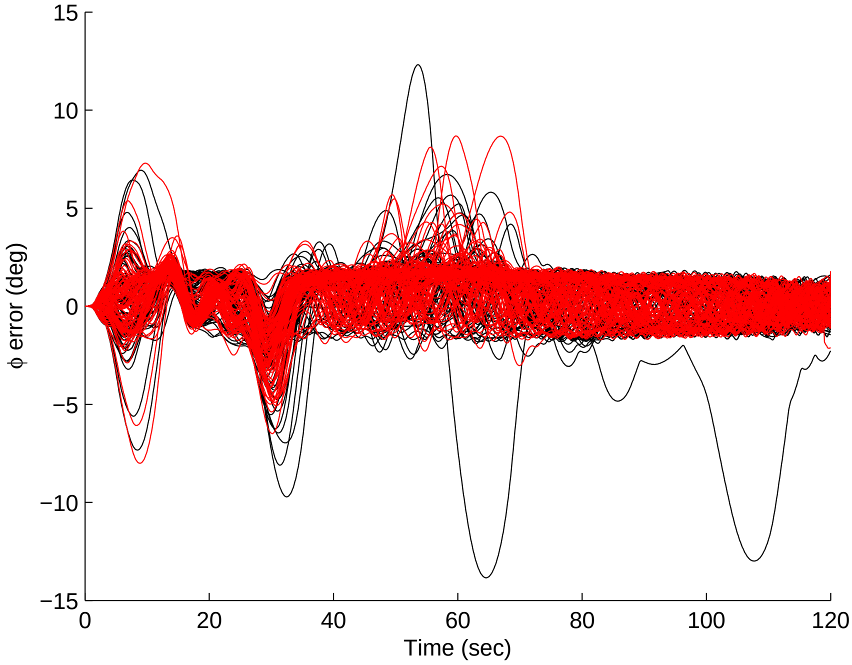


Figure 9.10: CLV roll error comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

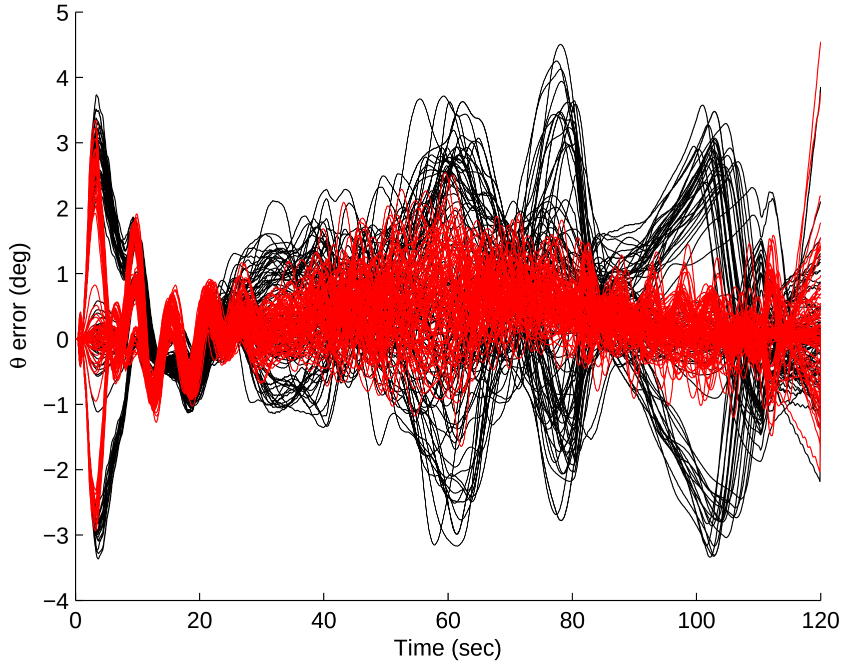


Figure 9.11: CLV pitch error comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

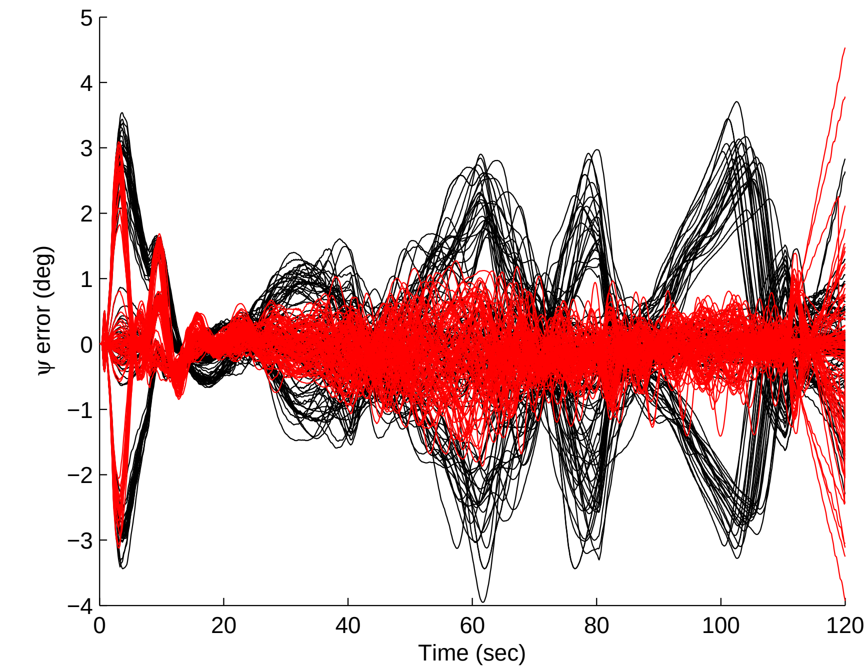


Figure 9.12: CLV yaw error comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

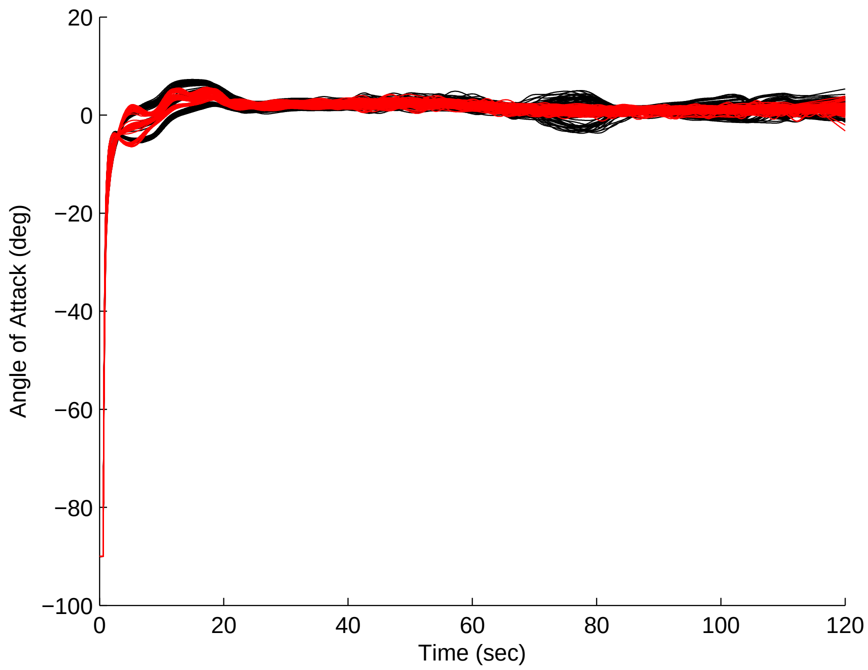


Figure 9.13: CLV angle of attack comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

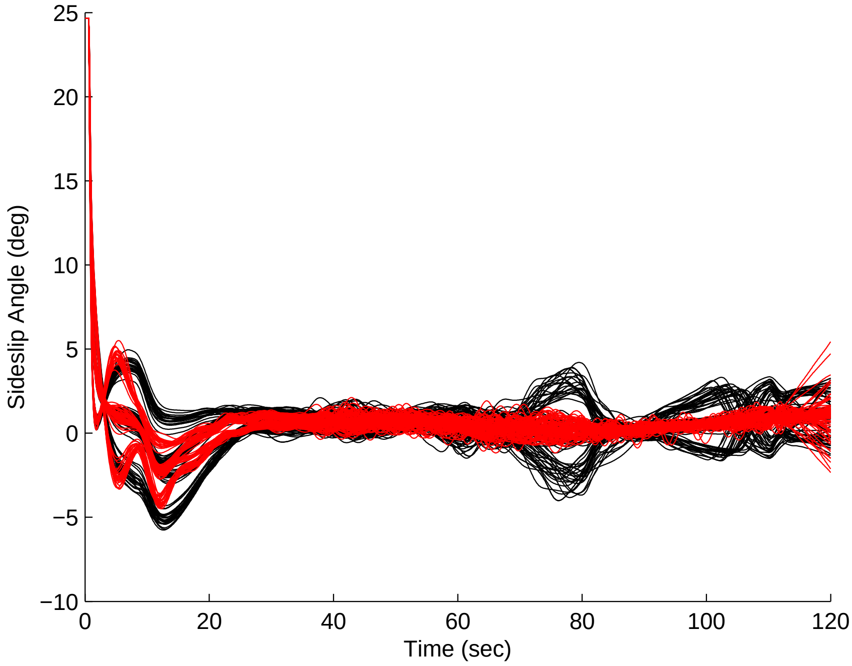


Figure 9.14: CLV sideslip comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

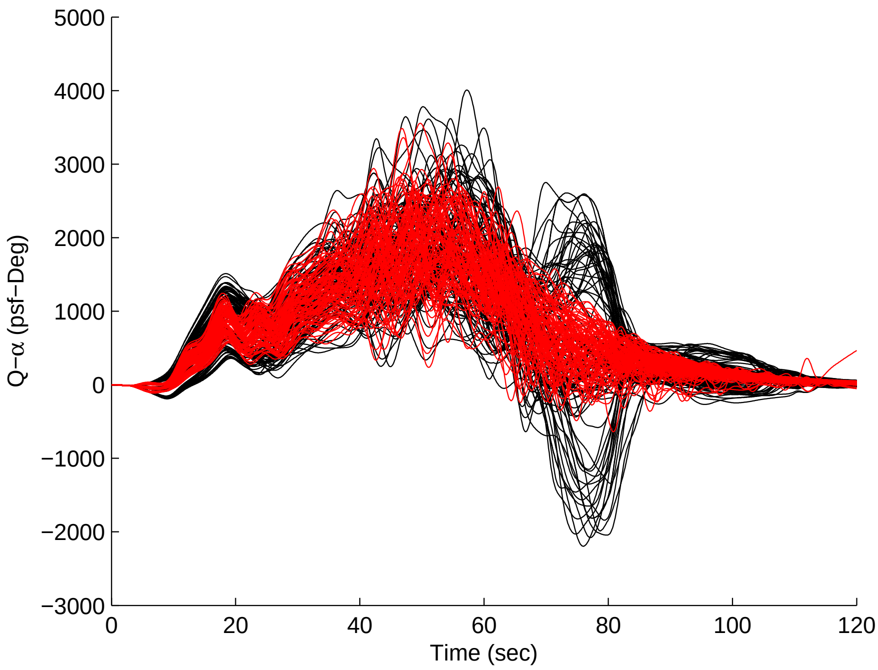


Figure 9.15: CLV $Q - \alpha$ comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

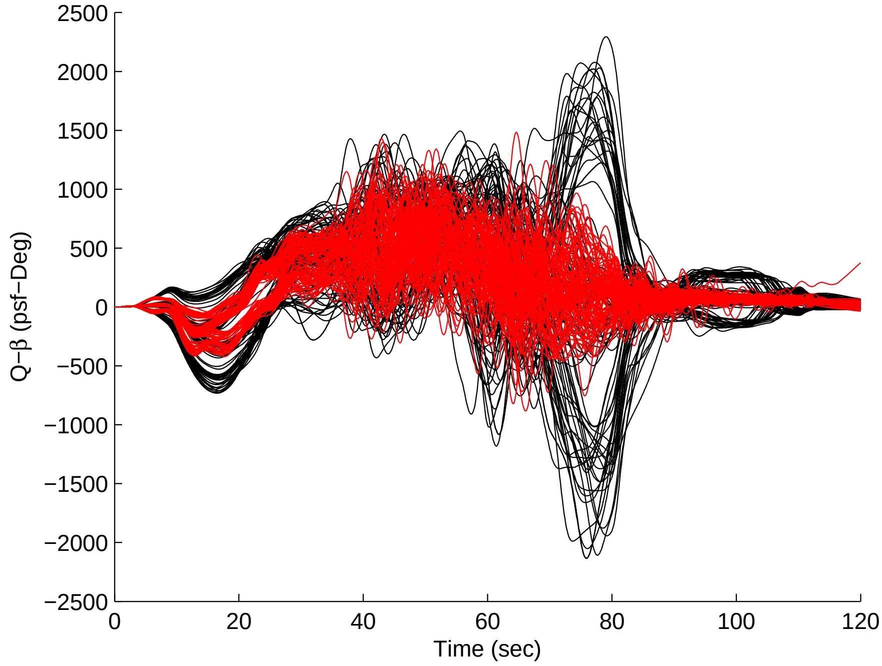


Figure 9.16: CLV $Q-\beta$ comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

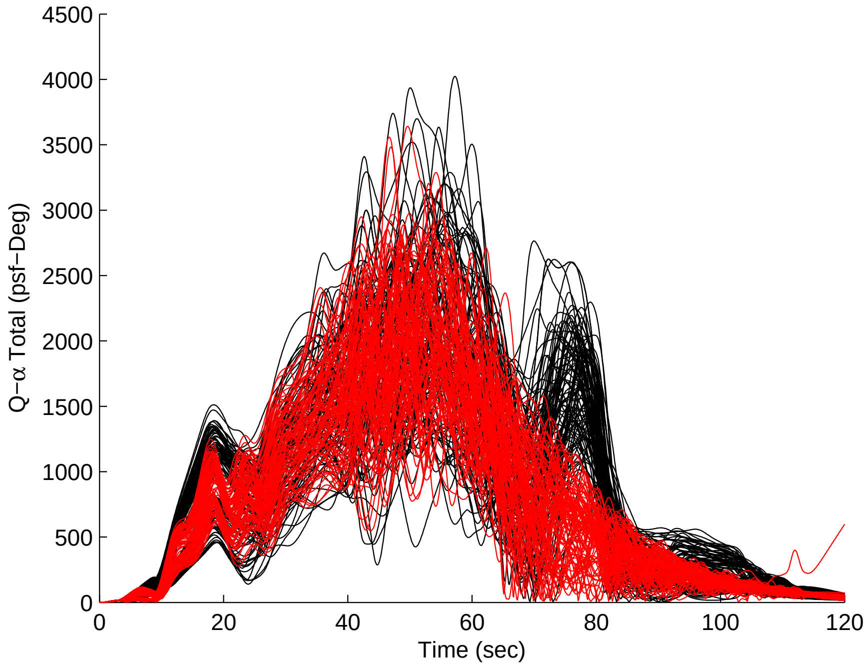


Figure 9.17: CLV $Q-\alpha$ total comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

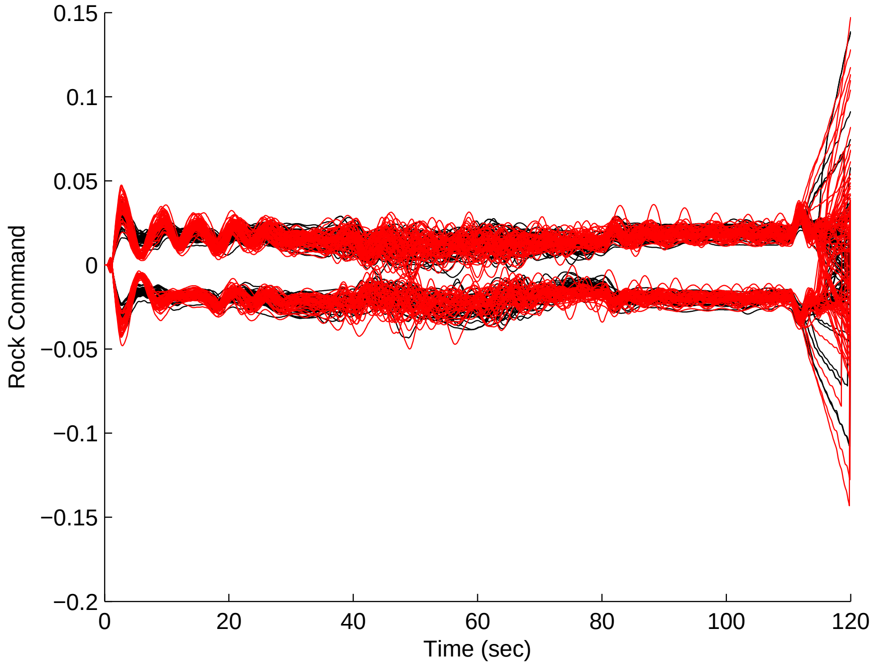


Figure 9.18: CLV rock command comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

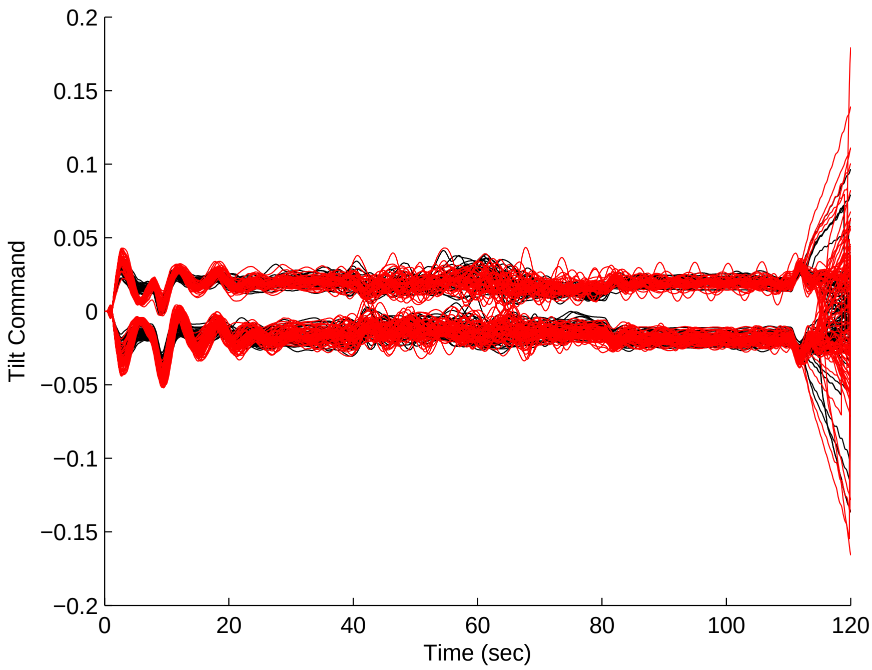


Figure 9.19: CLV tilt command comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

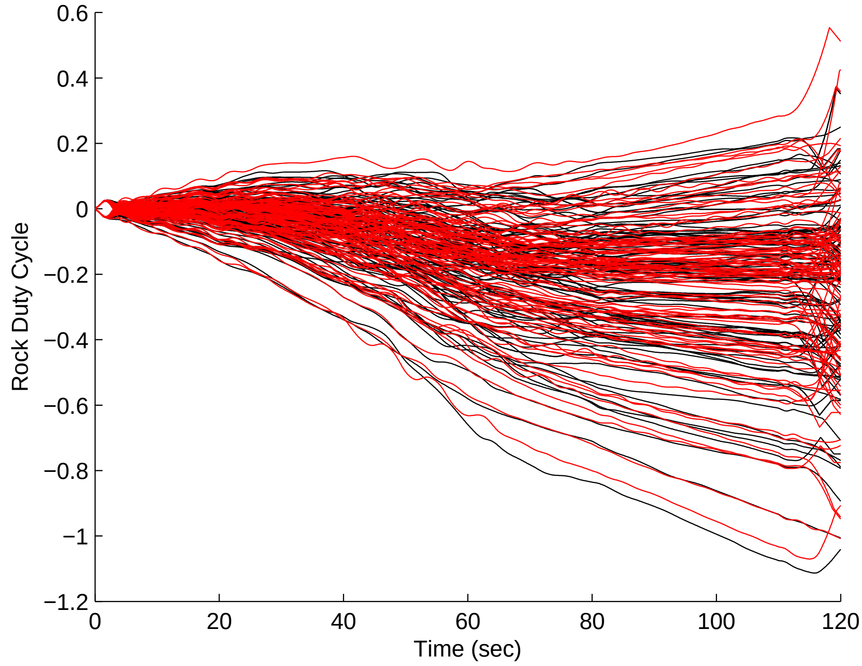


Figure 9.20: CLV rock duty cycle comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

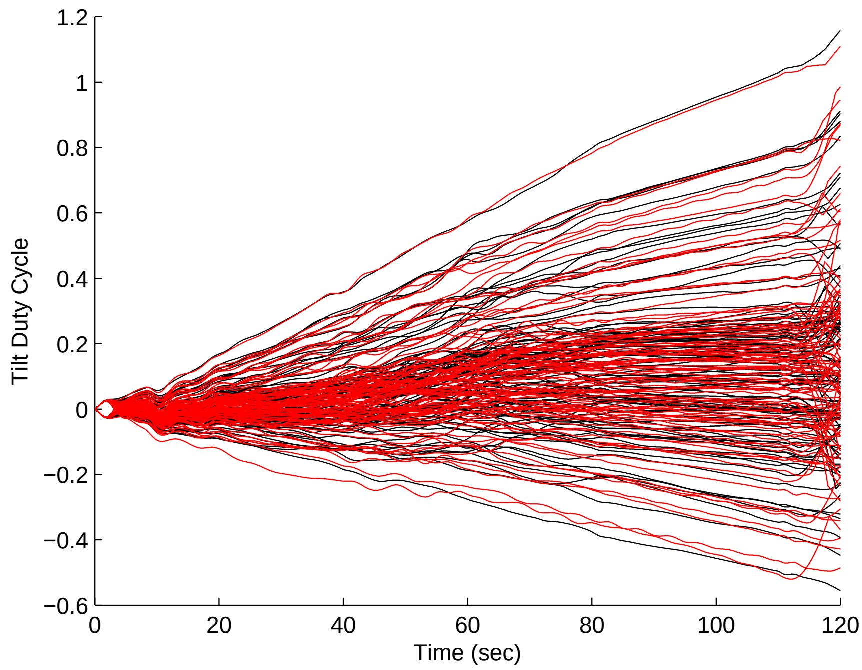


Figure 9.21: CLV tilt duty cycle comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

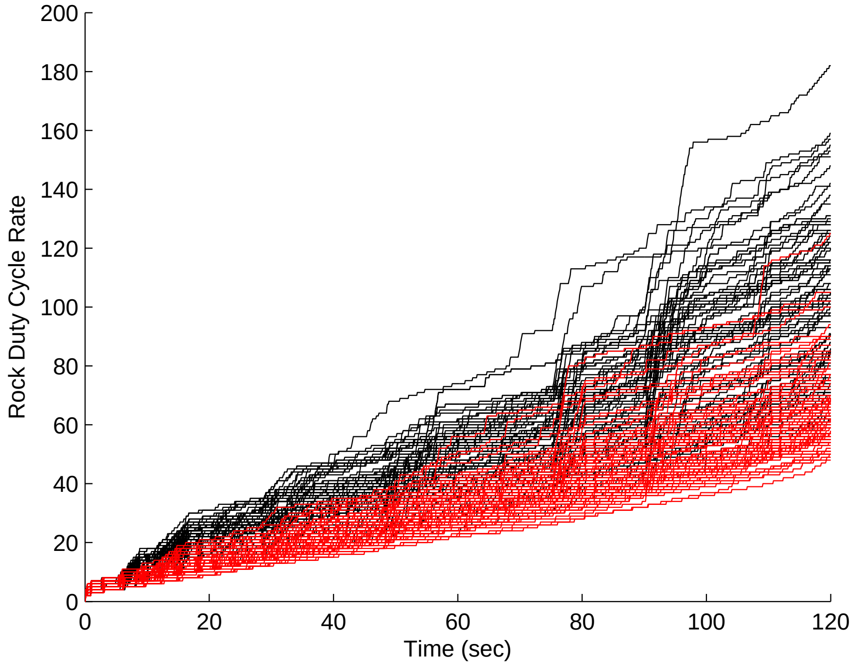


Figure 9.22: CLV rock duty cycle rate comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.

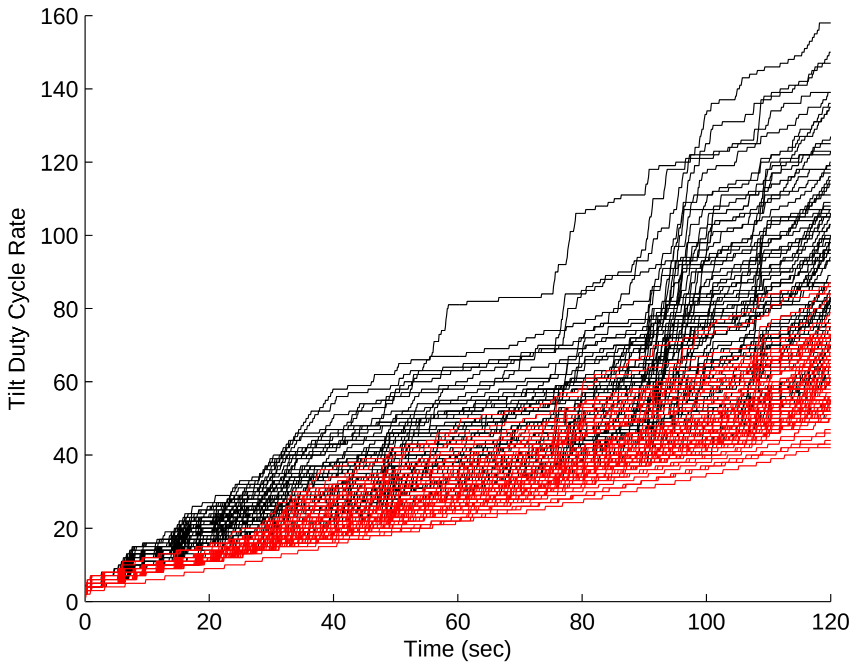


Figure 9.23: CLV tilt duty cycle rate comparison. Black lines represent the baseline responses and red lines represent the H_∞-NMA responses.