AIAA-2008-7128

$\mathcal{L}_1$ Adaptive Control Law for Flexible Space Launch Vehicle and Proposed Plan for Flight Test Validation

Evgeny Kharisov[*]

University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

Irene M. Gregory[†]

NASA Langley Research Center, Hampton, VA 23681

Chengyu Cao[‡]

University of Connecticut, Storrs, CT 06269

Naira Hovakimyan[§]

University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

This paper explores application of the $\mathcal{L}_1$ adaptive control architecture to a generic flexible Crew Launch Vehicle (CLV). Adaptive control has the potential to improve performance and enhance safety of space vehicles that often operate in very unforgiving and occasionally highly uncertain environments. NASA's development of the next generation space launch vehicles presents an opportunity for adaptive control to contribute to improved performance of this statically unstable vehicle with low damping and low bending frequency flexible dynamics. In this paper, we consider the $\mathcal{L}_1$ adaptive output feedback controller to control the low frequency structural modes and propose steps to validate the adaptive controller performance utilizing one of the experimental test flights for the CLV Ares-I Program.

I. Introduction

NASA has committed to building the Ares-I Crew Launch Vehicle (CLV) as the man-rated launcher designed to meet its current as well as future needs for human space flight in support of the Vision for Space Exploration.[1] Ares-I is a two stage rocket with a solid propellant first stage derived from the Shuttle Reusable Solid Rocket Motor and an upper stage that uses engines based on the Saturn V. Among numerous technical challenges in building a crew launch vehicle is the ascent flight control system.

The ascent flight control system must accurately track the trajectory guidance commands in order to deliver the payload into its target orbit. Problems in vehicle control arise because long, slender launch vehicles, such as Saturn V and Ares-I, cannot be considered rigid but must be treated as flexible structures. Similar to flexible aircraft, the resulting dynamics are highly coupled with significant interactions between rigid body dynamics and structural modes. Forces acting on the launch vehicle resulting from atmospheric perturbations or active control of the vehicle excite the structure and cause body bending. Since the structure possesses low damping, oscillatory bending modes of considerable amplitude can be produced, thus, subjecting control sensors to these large amplitudes at their particular location. If not properly accounted for, the local sensor readings are interpreted as describing the total vehicle behavior which may cause self-excitation and instability of the control system. A description of the particular challenges associated with the ARES-I Crew Exploration Vehicle and the ascent control system design goals are presented in [2].

Adaptive control has the potential to improve robustness and performance as well as enhance safety of space vehicles that often operate in very unforgiving and occasionally highly uncertain environments. NASA's development of the next generation space launch vehicles presents an opportunity for adaptive

[*] Graduate Student, Department of Aerospace Engineering, AIAA Student Member, evgeny@illinois.edu

[†] Senior Aerospace Research Engineer, Dynamic Systems and Control Branch, MS 308, AIAA Senior Member, irene.m.gregory@nasa.gov

[‡] Assistant Professor, Department of Mechanical Engineering, AIAA Member, ccao@engr.uconn.edu

[§] Professor, Department of Mechanical Science and Engineering, and AIAA Associate Fellow, nhovakim@illinois.edu control to contribute to improved performance of this statically unstable vehicle with low damping and low bending frequency flexible dynamics.

The control challenges associated with an Ares-I CLV and the potential of $\mathcal{L}_1$ control theory motivated the work presented in this paper. We explore the $\mathcal{L}_1$ adaptive output feedback control architecture to achieve the tracking objective and guarantee stability and robustness in the presence of uncertain dynamics, such as changes in flexible mode characteristics, and unexpected failures. This $\mathcal{L}_1$ adaptive control architecture was first proposed by Cao and Hovakimyan in [3–7]. Unlike conventional adaptive controllers, the $\mathcal{L}_1$ controller adapts fast, leading to desired transient and asymptotic tracking with guaranteed, bounded away from zero, time-delay margin. These features of the $\mathcal{L}_1$ control theory make it an ideal candidate for validation and verification (V&V) purposes. We present an architecture to test and validate the $\mathcal{L}_1$ adaptive controller against its theoretical performance guarantees as part of the Ares I flight test series. In addition, we identify potential issues and open problems for flight test part of the V&V process.

The paper is organized as follows. Section II describes a generic flexible crew launch vehicle model and the associated control system. Section III provides an overview of the output feedback $\mathcal{L}_1$ adaptive control theory and quantifies the uniform guaranteed performance bounds. Section IV discusses the specific implementation of the $\mathcal{L}_1$ controller design for the generic CLV. Section V provides simulation results and analysis of the designed control system. Section VI addresses the proposed flight validation test of the $\mathcal{L}_1$ adaptive controller. Conclusions are presented in Section VII.

II. Generic Crew Launch Vehicle Model

A nonlinear complex model of a generic crew launch vehicle, obtained by amalgamation of several legacy vehicles exhibiting the desired characteristics of a flexible space launch vehicle, was obtained from NASA Marshal Space Flight Center. This publicly released generic crew launch vehicle model has been distributed in a SAVANT Matlab/Simulink based tool.[8, 9] The model contains standard environmental dynamics, such as atmospheric and gravity effects, rigid body plant dynamics, flexible dynamics, propulsion system dynamics, and is closed by the $\mathcal{L}_1$ adaptive output feedback controller. Each part is briefly described below, while a top level block diagram of the model is shown in Figure 2.

A simplified schematic of the closed inner-loop system is shown in Figure 1. The control system consists of a control conversion subsystem, which calculates the tracking errors, and the $\mathcal{L}_1$ adaptive output feedback controller.

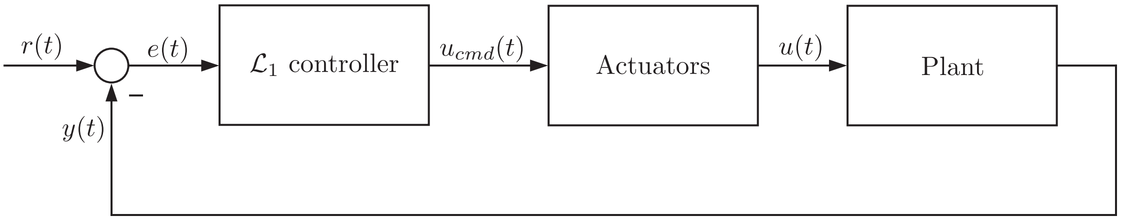


Figure 1: Closed-loop system block diagram

The control system commanded trajectory $r(t)$ is generated by a guidance system and is represented by quaternions that define the desired position of vehicle's body frame with respect to an inertial frame. The guidance system is not modeled in the simulation: instead the commanded trajectory is taken from a file provided with the model. The feedback signal from the plant $y(t)$ is the vehicle's angular position in the body reference frame expressed in quaternions. The input $e(t)$ into the $\mathcal{L}_1$ adaptive controller is the attitude tracking error (roll (ϕ), pitch (θ), and yaw (ψ). A control conversion block is used to compute the three dimensional error vector between the four-dimensional commanded trajectory and the output quaternions. The command signal consists of three components: one is the commanded thrust for the Reaction Control System (RCS), which controls body roll axis only, and the other two components represent the commanded trust vector gimbal angles for the Solid Rocket Booster (SRB) in pitch and yaw directions. The only actuator dynamics present in the model are those associated with the SRB control of the pitch and yaw axis.

The plant model simulates the kinematics and the dynamics of the vehicle and takes into account the

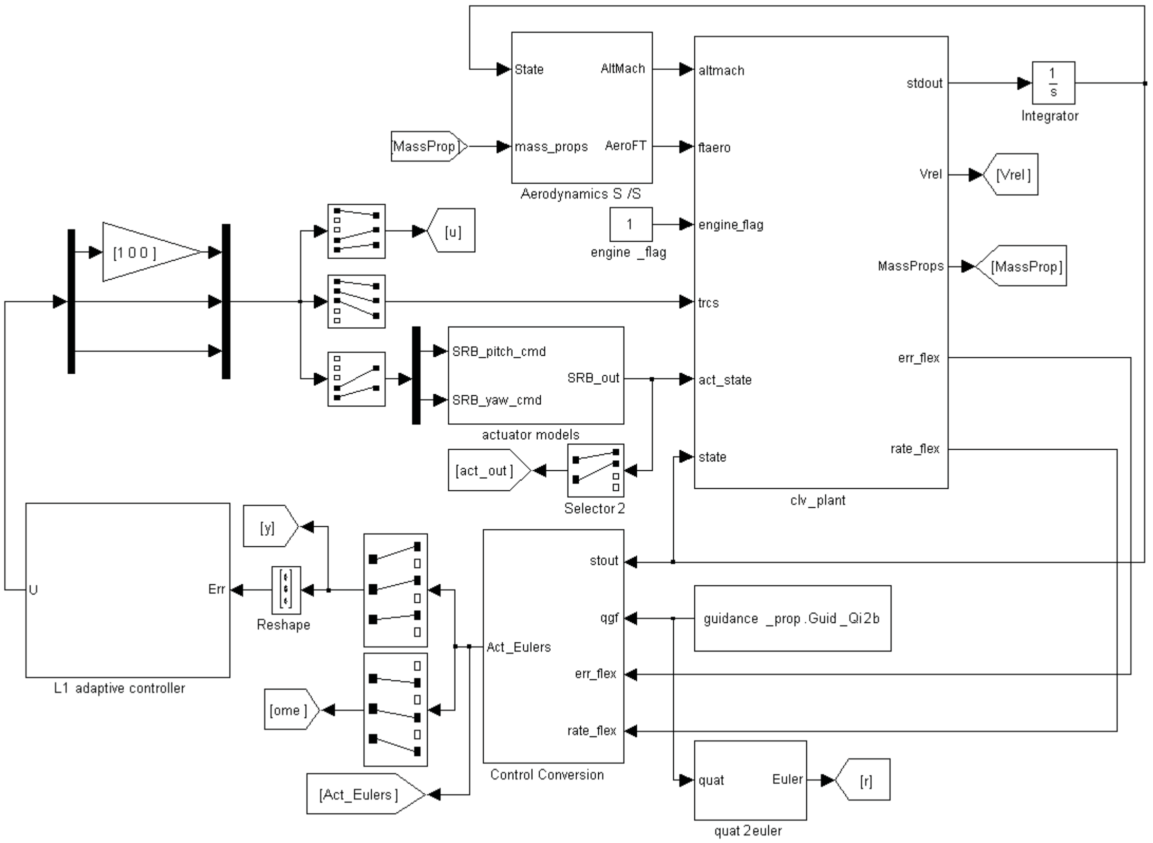


Figure 2: Generic CLV Model with $\mathcal{L}_1$ adaptive controller

following events:

- CLV aerodynamic forces and torques,
- Solid Rocket Booster (SRB) engine properties,
- Gravity model,
- Nozzle engine inertia effects (Tail-Wags-Dog),
- Slosh in fuel tanks of the second stage,
- Flexible body dynamics,
- Actuator dynamics for the SRB control system.

The next section describes the fundamental equations, on which the launch vehicle dynamics are based.

A. Kinematic and dynamic equations for the crew launch vehicle

The simulation model uses three reference frames to define all angular and translational coordinates of the launch vehicle. These frames are shown in Figure 3.

The Υ is a global inertial frame (without considering heliocentric-rotation) connected with the Earth center. The Z axis is directed to the north gyro-pole. The local frame has its origin connected to Earth center and rotates with the Earth. The Z_l axes of the local frame coincides with the Z_Υ. The body frame has its origin at the vehicle center-of-gravity and the X_b axis is directed along the centerline towards the nose of the rocket.

]

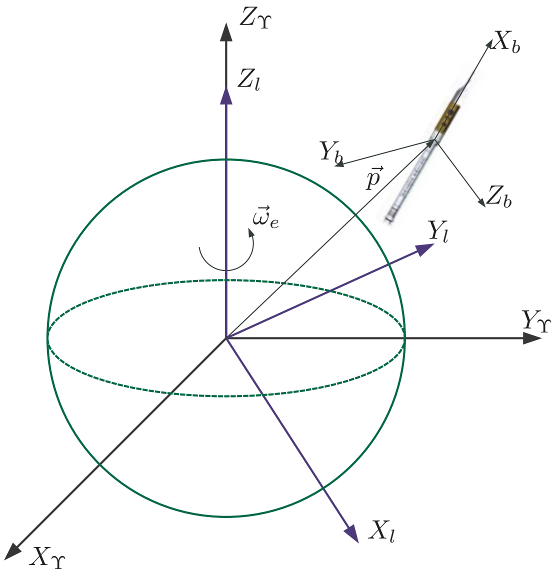


Figure 3: Coordinate frames used in the model

The equations of angular motion are given by

$$\begin{aligned}\dot{\omega}(t) &= \epsilon(t) \\ \dot{Q}_{Ib}(t) &= \frac{1}{2}Q_{Ib}(t)\begin{bmatrix}\omega(t)\\0\end{bmatrix}\end{aligned}$$

where

$$\begin{aligned}\epsilon(t) &= I^{-1}(t)(M_a(P,\rho,v,Q_{Ib},t) + M_{rcs} + M_r(P,\theta_N,\psi_N,t) \\ &+ M_{TWD}(\ddot{\theta}_N,\ddot{\psi}_N) + M_{sl}(a_b,Q_{Ib},\epsilon,\omega,t) - \omega(t)\times(I(t)\omega(t)))\end{aligned}$$

The equations of translational motion are given by

$$\begin{aligned}\dot{v}(t) &= a(t)\\ \dot{p}(t) &= v(t)\end{aligned}$$

where

$$\begin{aligned}a(t) &= Q^*_{Ib}a_b(t)Q_{Ib} + g(p)\\ a_b(t) &= \frac{F_a(P,\rho,v,Q_{Ib},t) + F_r(P,\theta_N,\psi_N,t) + F_{sl}(a_b,Q_{Ib},\epsilon,\omega,t)}{m(t)}\end{aligned}$$

The relative velocity is calculated according to the following equation

$$V_{rel}(t) = \|v(t) - \omega(t)\times p(t)\|_2$$

In the above equations, the system states are given via the following variables:

- $\omega(t)$ is the vector of angular rates of CLV in the body frame,
- $Q_{Ib}(t)$ is the quaternion of translation from the Υ frame to the body frame,
- $v(t)$ is linear velocity vector presented in the Υ frame,
- $p(t)$ is the position of CLV's center of mass in the Υ frame.

The control input variables are:

- θ_N is the nozzle position corresponding to the pitch angle,
- ψ_N is the nozzle position corresponding to the yaw angle,
- M_{rcs} is the torque applied by RCS engine.

The angular acceleration in the body frame and the translational acceleration in the Υ frame are denoted by $\epsilon(t)$ and $a(t)$ respectively. Further, $a_b(t)$ is the translational acceleration, without gravity, in the body frame, and $g(p)$ denotes the gravity acceleration. In the equation of angular motion the following torques are taken into account: $M_a(P, \rho, v, Q_{Ib}, t)$ is the torque inducted by aerodynamic effects, $M_r(P, \theta_N, \psi_N, t)$ is the rocket engine torque, $M_{TWD}(\ddot{\theta}_N, \ddot{\psi}_N)$ is the torque due to engine nozzle inertia effect, $M_{sl}(a_b, Q_{Ib}, \epsilon, \omega, t)$ is the slosh induced torque. In the equations of translational motion the following forces are considered: $F_a(P, \rho, v, Q_{Ib}, t)$ is the aerodynamic force, $F_r(P, \theta_N, \psi_N, t)$ is the main rocket engine force, $F_{sl}(a_b, Q_{Ib}, \epsilon, \omega, t)$ is the slosh induced force. Finally, $I(t)$ denotes the inertia tensor, $m(t)$ is the mass of the vehicle, and P and ρ are the static pressure and the atmospheric density, respectively, at the vehicle's current position.

B. Modeled physics

1. *CLV aerodynamics*

The aerodynamic model consists of three parts: flight conditions model, aerodynamic coefficients lookup tables, and computation of aerodynamic forces and torques. The first part performs calculations of altitude, Mach number, dynamic pressure, angle of attack and sideslip. Then these variables are used to obtain the corresponding information on aerodynamic coefficients and baseline forces from the lookup tables, which are based on wind tunnel data. The computation of forces and moments is done according to the following equations

$$\begin{aligned} F_a &= \bar{q} S C_f + F_{base} \\ M_a &= \bar{q} S c C_m + r_g \times F_a \end{aligned}$$

where $\bar{q}$ is the dynamic pressure, S is the surface area, C_f and C_m are the aerodynamic coefficient matrices, F_{base} is the base force, c is the aerodynamic cord length, and r_g is the position of aerodynamic force center point with respect to the center of mass of the vehicle.

2. *Solid Rocket Booster (SRB) engine*

The engine model computes the propulsive force, $F_r(P, \theta_N, \psi_N, t)$, and the moment, $M_r(P, \theta_N, \psi_N, t)$. First, the thrust in the vacuum corresponding to the current time is read from a table, then it is recalculated for the current value of the static pressure. The rocket engine force and moment are obtained by considering current gimbal angles and the engine location with respect to the center of mass of the vehicle.

3. *Gravity model*

The non-spherical Earth effects are taken into account by the model, which is based on the J4 NASA gravity model.

4. *Nozzle engine inertia effects*

The torque produced by the Tail-Wags-Dog (TWD) effect is calculated according to the following equation:

$$M_{TWD} = \begin{bmatrix} 0 \\ \ddot{\theta}_N \\ \ddot{\psi}_N \end{bmatrix} I_N$$

where I_N is the nozzle's inertia tensor.

5. Slosh model for the fuel tanks of the second stage

The slosh model consists of two similar models for liquid oxygen and hydrogen. The fuel slosh phenomena are modeled via a spring-damper systems. All the parameters are functions of the liquid fuel level in the tanks and are taken from the lookup tables.

6. Flexible body dynamics

Flexible body dynamics are linear and are based on a modal data set that contains mode shapes, modal displacements, rotations, and frequencies. The displacements and rotations are given at several key nodal points along the centerline. These key nodal elements reflect the location of the sensors and the actuators. Flexible dynamics are integrated into the model as an additive component to the rigid body sensor computations of the angular position and the angular rate. The effect of propellant slosh or the flexibility of rocket gimbal dynamics were not included in the provided model.

7. Actuator models

As the roll channel has no actuator model dynamics, the command is directly transformed into thrust that results in the RCS torque applied to the plant. The pitch and yaw channels have the same second order actuator model given by the following transfer function:

$$T_{act}(s) = \frac{a_0}{b_2 s^2 + b_1 s + b_0}$$

The actuator bandwidth in the provided model is roughly 21 rad or 3.3 Hz.

III. $\mathcal{L}_1$ Adaptive Output Feedback Controller

This section presents an overview of the $\mathcal{L}_1$ adaptive output feedback controller and its application to the above presented generic flexible CLV model. The $\mathcal{L}_1$ adaptive control architecture was first presented by Cao and Hovakimyan in [10] for systems constant in unknown parameters using a state feedback approach. The guaranteed time-delay margin of $\mathcal{L}_1$ adaptive control architecture was derived in [6]. Later the paradigm was extended to output feedback in [11] for a class of reference systems with strictly positive real (SPR) transfer functions. Extension to nonlinear time-varying systems in the presence of multiplicative and additive unmodeled dynamics was reported in [7,12,13]. In [14], an output feedback extension is presented for a class of uncertain systems that allows for tracking arbitrary reference systems, without imposing an SPR-type requirement on its input-output transfer function. It is this particular architecture that we employ in this paper to address the control challenge of the generic flexible CLV model. The $\mathcal{L}_1$ adaptive output feedback control architecture is presented in Figure 4 and a brief overview of it is given below.

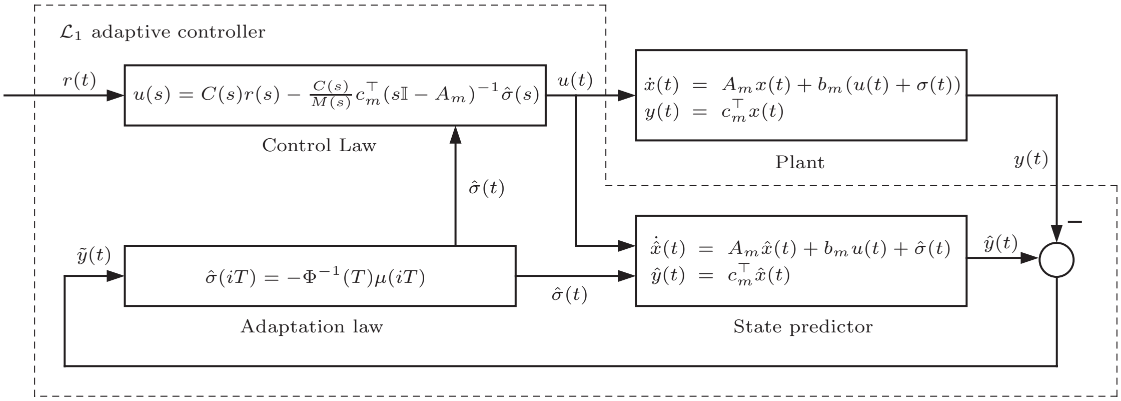

$\mathcal{L}_1$ adaptive controller
$r(t)$
$u(s) = C(s)r(s) - \frac{C(s)}{M(s)} c_m^\top (s\mathbb{I} - A_m)^{-1} \hat{\sigma}(s)$
Control Law
$u(t)$
$\dot{x}(t) = A_m x(t) + b_m(u(t) + \sigma(t))$
$y(t) = c_m^\top x(t)$
Plant
$y(t)$
$\hat{\sigma}(t)$
$\tilde{y}(t)$
$\hat{\sigma}(iT) = -\Phi^{-1}(T)\mu(iT)$
Adaptation law
$\hat{\sigma}(t)$
$\dot{\hat{x}}(t) = A_m \hat{x}(t) + b_m u(t) + \hat{\sigma}(t)$
$\hat{y}(t) = c_m^\top \hat{x}(t)$
State predictor
$\hat{y}(t)$
–

Figure 4: Closed loop system with the L1 adaptive controller

Consider the following single-input single-output (SISO) system:

$$y(s) = A(s)(u(s) + d(s)) \tag{1}$$

where $u(t) \in \mathbb{R}$ is the commanded control, $y(t) \in \mathbb{R}$ is the system output, $A(s)$ is a strictly proper unknown transfer function of unknown relative degree n_{ar} for which only a known lower bound $1 < n_r \leq n_{ar}$ is available, $d(s)$ is the Laplace transform of the time-varying uncertainties and disturbances $d(t) = f(t, y(t))$, where f is an unknown map subject to the following assumptions:

Assumption 1 *There exist constants $L > 0$ and $L_0 > 0$ such that for all $t \geq 0$:*

$$\begin{aligned} |f(t, y_1) - f(t, y_2)| &\leq L|y_1 - y_2| \\ |f(t, y)| &\leq L|y| + L_0. \end{aligned}$$

Assumption 2 *There exist constants $L_1 > 0$, $L_2 > 0$ and $L_3 > 0$ such that for all $t \geq 0$:*

$$|\dot{d}(t)| \leq L_1|\dot{y}(t)| + L_2|y(t)| + L_3\,,$$

where the numbers L, L_0, L_1, L_2, L_3 can be arbitrarily large.

Let $r(t)$ be a given bounded continuous reference input signal. The control objective is to design an adaptive output feedback controller $u(t)$ such that the system output $y(t)$ tracks the reference input $r(t)$ following a *desired model*

$$y_d(s) = M(s)u(s)\,,$$

where $M(s)$ is a minimum-phase stable transfer function of relative degree $n_r > 1$. The system equations in terms of the *desired model* can be rewritten as:

$$y(s) = M(s)(u(s) + \sigma(s))\,,$$

where

$$\sigma(s) = \Big((A(s) - M(s))u(s) + A(s)d(s)\Big)/M(s)$$

Next we introduce the closed-loop *reference system* that defines *an achievable control objective* for the $\mathcal{L}_1$ adaptive controller.

Closed-loop reference system: The reference system is given by

$$\begin{aligned} y_{ref}(s) &= M(s)(u_{ref}(s) + \sigma_{ref}(s)) \\ \sigma_{ref}(s) &= \big((A(s) - M(s))u_{ref}(s) + A(s)d_{ref}(s)\big)/M(s) \\ u_{ref}(s) &= C(s)(r(s) - \sigma_{ref}(s)) \end{aligned} \tag{2}$$

where $C(s)$ is a low pass filter with DC gain $C(0) = 1$ and $d_{ref}(t) = f(t, y_{ref}(t))$.

According to [14, Lemma 1] the selection of $C(s)$ and $M(s)$ must ensure that

$$H(s) = A(s)M(s)/\Big(C(s)A(s) + (1 - C(s))M(s)\Big) \tag{3}$$

is stable and that the $\mathcal{L}_1$-gain of the cascaded system is upper bounded as follows:

$$\|H(s)(1 - C(s))\|_{\mathcal{L}_1} L < 1 \tag{4}$$

Then the reference system in (2) is stable.

The elements of the $\mathcal{L}_1$ adaptive controller are introduced next.

State predictor (passive identifier): Let ($A_m \in \mathbb{R}^{n\times n}$, $b_m \in \mathbb{R}^n$, $c_m \in \mathbb{R}^n$) be the minimal realization of $M(s)$. Hence, (A_m,b_m,c_m) is controllable and observable with A_m being Hurwitz. Then the system in (1) can be rewritten as

$$\begin{aligned} \dot{x}(t) &= A_m x(t) + b_m(u(t) + \sigma(t)) \\ y(t) &= c_m^\top x(t) \end{aligned} \tag{5}$$

The state predictor is given by:

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m\hat{x}(t) + b_m u(t) + \hat{\sigma}(t) \\ \hat{y}(t) &= c_m^\top \hat{x}(t) \end{aligned} \tag{6}$$

where $\hat{\sigma}(t) \in \mathbb{R}^n$ is the vector of adaptive parameters. Notice that in the state predictor equations $\hat{\sigma}(t)$ is not in the span of b_m, while in the equation (5) $\sigma(t)$ is in the span of b_m. Further, let $\tilde{y}(t) = \hat{y}(t) - y(t)$.

Adaptation law: Let P be the solution of the following algebraic Lyapunov equation:

$$A_m^\top P + PA_m = -Q$$

where $Q > 0$. From the properties of P it follows that there always exists a nonsingular $\sqrt{P}$ such that

$$P = \sqrt{P}^\top \sqrt{P}.$$

Given the vector $c_m^\top(\sqrt{P})^{-1}$, let D be the $(n-1)\times n$-dimensional nullspace of $c_m^\top(\sqrt{P})^{-1}$, i.e.

$$D(c_m^\top(\sqrt{P})^{-1})^\top = 0 \tag{7}$$

and let

$$\Lambda = \left[\begin{array}{c} c_m^\top \\ D\sqrt{P} \end{array}\right] \in \mathbb{R}^{n\times n} \tag{8}$$

The update law for $\hat{\sigma}(t)$ is defined via the sampling time $T > 0$[a]:

$$\hat{\sigma}(iT) = -\Phi^{-1}(T)\mu(iT), \quad i = 1, 2, \cdots, \tag{9}$$

where

$$\Phi(T) = \int_0^T \mathrm{e}^{\Lambda A_m \Lambda^{-1}(T-\tau)}\Lambda d\tau \tag{10}$$

and

$$\mu(iT) = \mathrm{e}^{\Lambda A_m \Lambda^{-1} T} 1_1 \tilde{y}(iT), \quad i = 1, 2, \cdots \tag{11}$$

Here $\underline{1}_1$ denotes the basis vector in the space $\mathbb{R}^n$ with its first element equal to 1 and other elements being zero.

Control law: The control law is defined via the output of the low-pass filter:

$$u(s) = C(s)r(s) - \frac{C(s)}{M(s)} c_m^\top (s\mathbb{I} - A_m)^{-1}\hat{\sigma}(s)\,. \tag{12}$$

The complete $\mathcal{L}_1$ adaptive controller consists of the state predictor in (6), the adaptation law in (9), and the control law in (12), subject to the $\mathcal{L}_1$-gain upper bound in (4). The performance bounds of the $\mathcal{L}_1$ adaptive output feedback controller are given by the following theorem.

Theorem 1

$$\begin{aligned} \lim_{T\to 0}\left(\|\tilde{y}\|_{\mathcal{L}_\infty}\right) &= 0 \\ \lim_{T\to 0}\left(\|y - y_{ref}\|_{\mathcal{L}_\infty}\right) &= 0 \\ \lim_{T\to 0}\left(\|u - u_{ref}\|_{\mathcal{L}_\infty}\right) &= 0 \end{aligned}$$

The result in this theorem follows immediately from [14, Theorem 1] and [14, Lemma 3].

[a] T defines the sampling rate of the available CPU.

IV. $\mathcal{L}_1$ Adaptive Control Law for Generic Crew Launch Vehicle Model

The dynamics of the generic CLV model are coupled in all three axis, thus, requiring and extension of the $\mathcal{L}_1$ adaptive output feedback controller to Multi-Input Multi-Output (MIMO) systems. Also, since the tracking error (neither in terms of Euler angles nor in terms of quaternions) is defined by means of conventional subtraction, it was necessary to reformulate the control problem as a stabilization problem by using some parametrization of the rotation matrix from the actual CLV frame to the commanded frame, for which the conventional subtraction is not required for implementing the $\mathcal{L}_1$ controller. Since the control inputs for the plant are in terms of roll, pitch and yaw channels, it was convenient to use the (error) Euler angles as a parametrization for this rotation matrix.

In this framework, the commanded control $r(t)$ in equation (12) becomes $r(t) \equiv 0$, which leads to the following control law

$$u(s) = -\frac{C(s)}{M(s)} c_m^\top (s\mathbb{I} - A_m)^{-1} \hat{\sigma}(s) \tag{13}$$

Also, the structure of the state predictor (6) and the adaptive law (9) remains the same, but they both use the tracking error instead of the actual system output, i.e. .

Next, the components of the designed control system are described in detail.

Desired system. The control system consists of three channels: roll, pitch, yaw. The matrix transfer function of the desired system was selected as follows:

$$M(s) = \begin{bmatrix} M_\phi(s) & 0 & 0 \\ 0 & M_\theta(s) & 0 \\ 0 & 0 & M_\psi(s) \end{bmatrix}$$

where $M_\phi(s)$, $M_\theta(s)$ and $M_\psi(s)$ are the scalar transfer functions for corresponding channels. It is natural that the desired transfer function has decoupled channels, which implies zeros at non-diagonal elements of the matrix transfer function.

In the current design the following transfer functions were selected:

$$M_\phi(s) = K_{M\phi} \frac{1}{1/\omega_{M\phi}^2 s^2 + 2\xi_{M\phi}/\omega_{M\phi} s + 1}$$

for roll and

$$M_{\theta,\psi}(s) = K_M \frac{1/\omega_{Mz}^2 s^2 + 2\xi_{Mz}/\omega_{Mz} s + 1}{1/\omega_{Mp}^2 s^2 + 2\xi_{Mp}/\omega_{Mp} s + 1} \frac{1}{1/\omega_M^2 s^2 + 2\xi_M/\omega_M s + 1}$$

for pitch and yaw. The Bode diagram of $M_{\theta,\psi}(s)$ is shown in Figure 5.

Low-pass filter. The following low-pass filter was selected:

$$C(s) = \begin{bmatrix} C_\phi(s) & 0 & 0 \\ 0 & C_\theta(s) & 0 \\ 0 & 0 & C_\psi(s) \end{bmatrix}$$

Here

$$C_\phi(s) = \frac{1}{1/\omega_{C\phi}^2 s^2 + 2\xi_{C\phi}/\omega_{C\phi} s + 1}$$

and

$$C_{\theta,\psi}(s) = \frac{1/\omega_{Cz} s + 1}{1/\omega_{Cp} s + 1} \frac{1}{1/\omega_C^2 s^2 + 2\xi_C/\omega_C s + 1}$$

The frequency response of $C_{\theta,\psi}(s)$ is shown in Figure 6.

Sample time. The sampling time was set to

$$T = 0.001$$

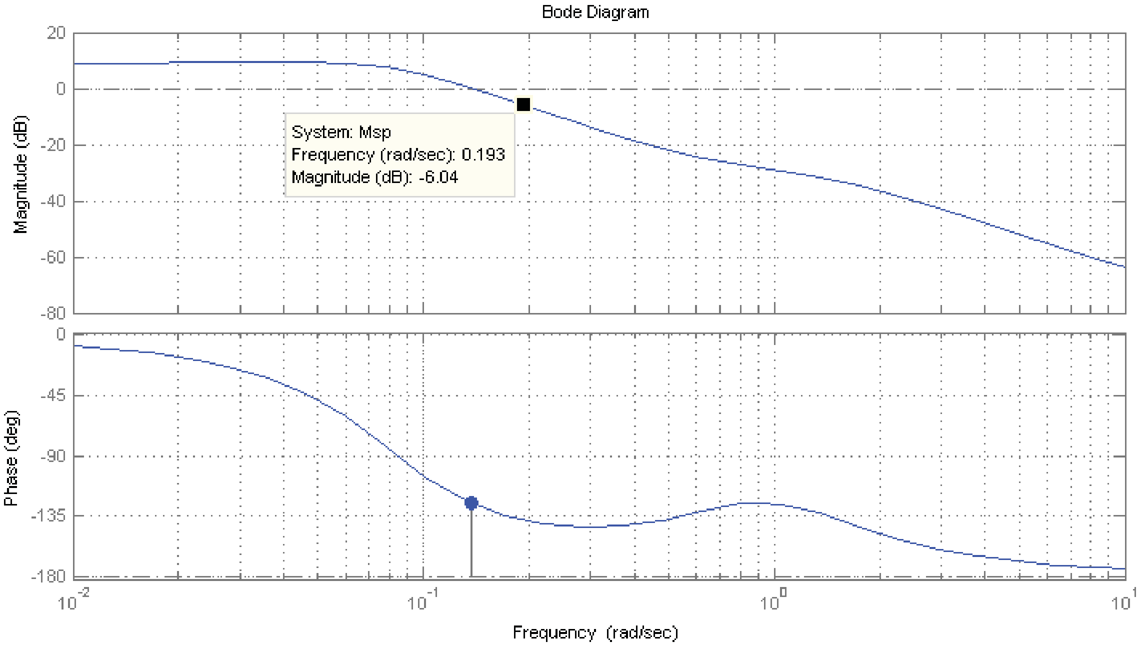


Figure 5: Pitch and roll desired system frequency response

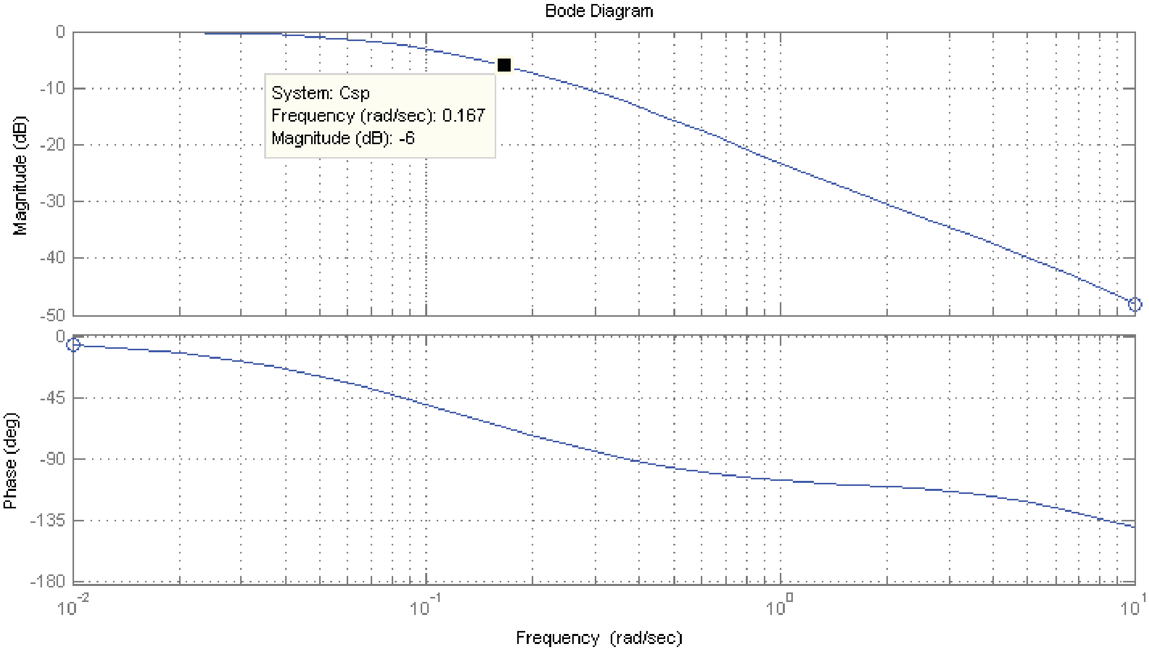


Figure 6: Pitch and roll low pass filter frequency response

V. Simulation Analysis

Full nonlinear simulation for the closed loop system, with all modeled events included, were run to evaluate the performance of the $\mathcal{L}_1$ adaptive controller defined via (6), (9), and (13). The results were obtained for both cases of enabled and absent flexible dynamics. For reference purposes, the guidance commanded ascent trajectory is plotted in figure 7. Note that for the first 10 seconds in all plots in this section, the command is held constant. This is done to verify that the $\mathcal{L}_1$ adaptive controller does not produce spurious signals.

For the purpose of comparison, the results for the baseline controller are reviewed first. The baseline controller that was provided with the model is decoupled and has the same architecture in all three axis (roll, pitch, yaw). The architecture in each axis consists of a low-pass filter and a notch filter on the error signal coming into the controller. The main purpose of the filters is to filter the flexible modes and the high frequencies that appear in the error signals. The filters are followed by a gain scheduled PID controller. The PID controller gains are scheduled on the relative velocity of the vehicle. The generated control command signal (RCS thrust command in roll, SRB gimbal angle command in pitch/yaw) is bounded by a saturation block which introduces the physical control limitations of the plant.

One of the purposes for analyzing the baseline controller was to see how sensitive it was to flexible dynamics. Typically in order for notch filters to be effective, the frequency of the targeted flexible mode needs to be well known. In case of a flexible CLV such an accuracy requirement on modeling flexible dynamics might be very expensive from the testing and control redesign perspective. Furthermore, the frequencies may also changed during flight differently from what has been predicted. Hence, our interest in sensitivity of a PID controller to flexible dynamics. Figure 8 shows results for the generic CLV, with and without flexible dynamics enabled, with the baseline PID controller (notch filters disabled). From Figure 8b it is clear that the PID controller is unable to handle flexible dynamics without the notch filters in the loop. This implies high sensitivity to uncertainty in flexible body dynamics. Furthermore, such a controller requires accurate design with appropriate selection of notches and gain scheduling.

The question we pose is can $\mathcal{L}_1$ adaptive controller provide required tracking performance for a flexible CLV without including notch filters to attenuate flexible dynamics?

Figure 9 shows performance results for the generic flexible CLV, with and without flexible dynamics enabled, with the $\mathcal{L}_1$ adaptive output feedback controller. Note that the time response of the closed-loop system with and without flexible dynamics enabled is almost the same, which implies that the $\mathcal{L}_1$ adaptive controller does not excite the flexible modes. Comparing the two cases, tracking errors do not increase substantially in the presence of flexible dynamics. The $\mathcal{L}_1$ adaptive controller commands in all three axis, with and without flexible dynamics, are very similar in magnitude to those of the baseline PID with rigid body dynamics only. Furthermore, the trajectory following capability of the closed-loop system is illustrated in figure 10b. For completeness we also present, in figure 11, commanded and actual control signals in the pitch and yaw axis.

The results clearly indicate that the system has good tracking performance with small errors and very reasonable angular rates. These results demonstrate that a single design adaptive controller is able to handle statically unstable flexible plant with large parametric variation (mass, velocity, aerodynamics properties, etc.) without addition of notch filters and without re-tuning for different flight conditions along the first stage trajectory.

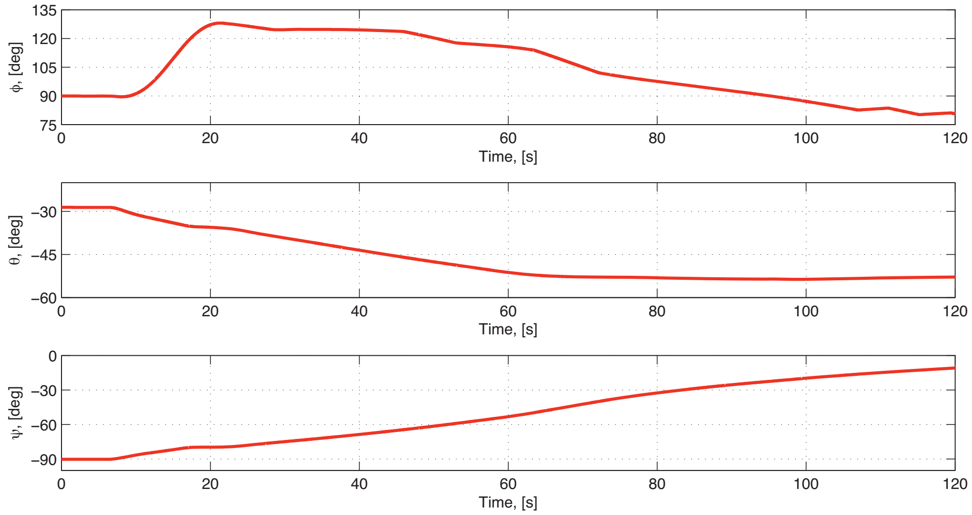
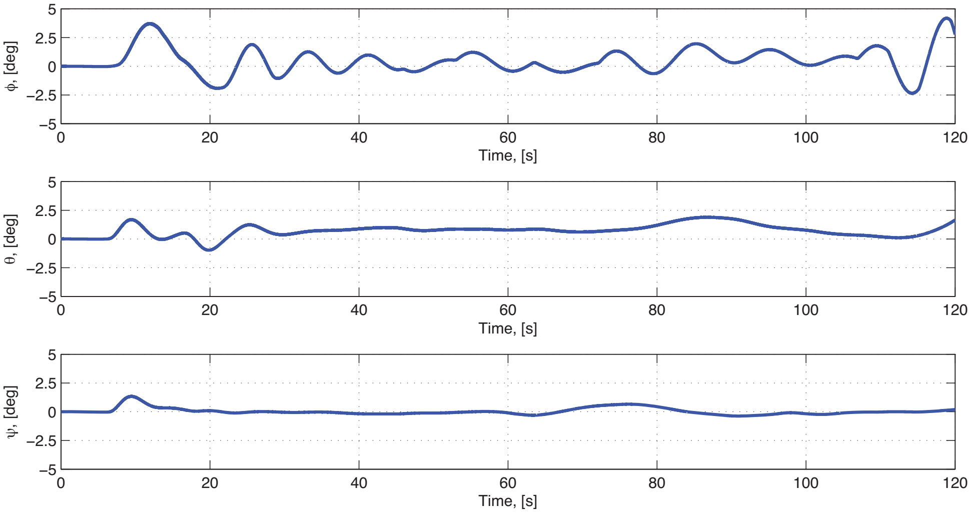
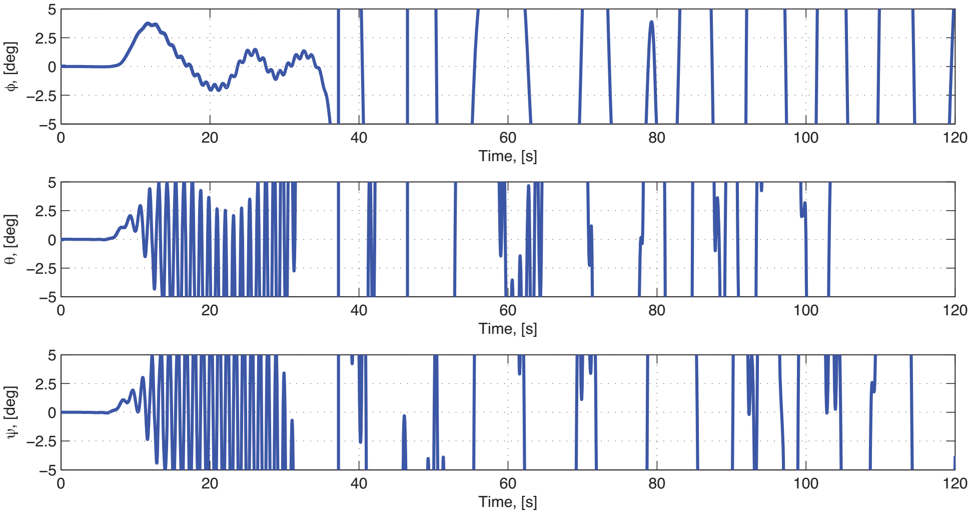


Figure 7: Guidance commanded trajectory

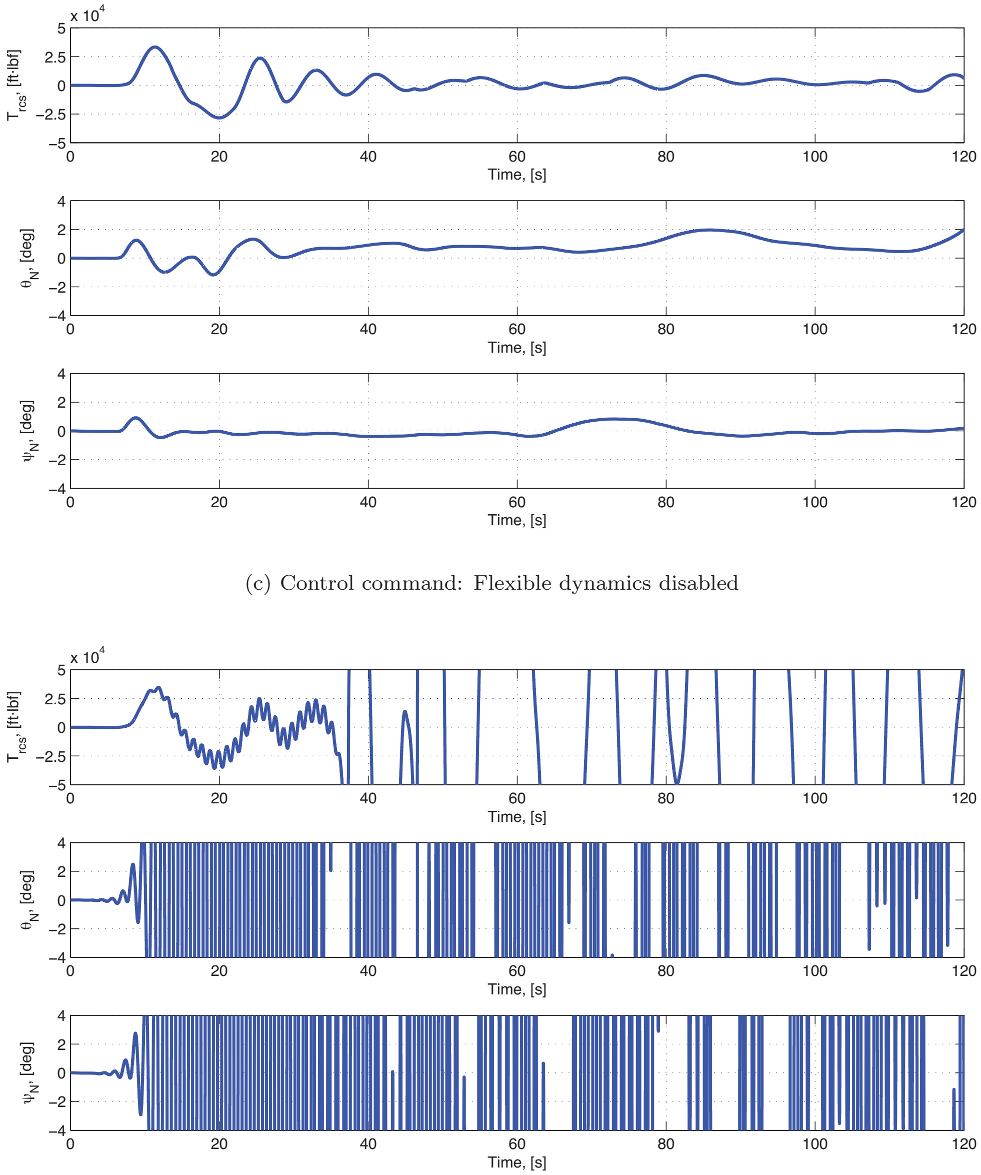


(a) Tracking errors: Flexible dynamics disabled



(b) Tracking errors: Flexible dynamics enabled

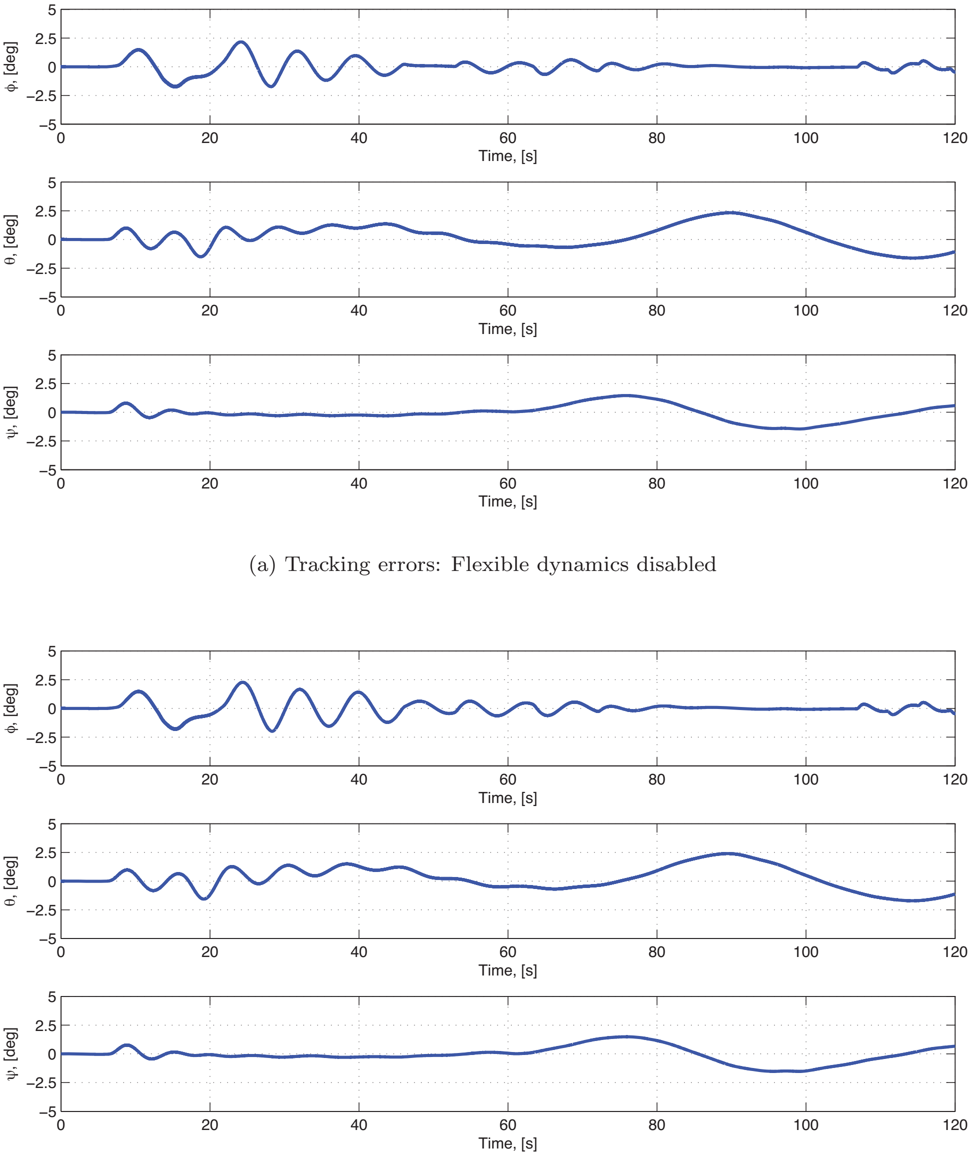
Figure 8: Closed-loop system with baseline PID controller



(c) Control command: Flexible dynamics disabled

(d) Control command: Flexible dynamics enabled

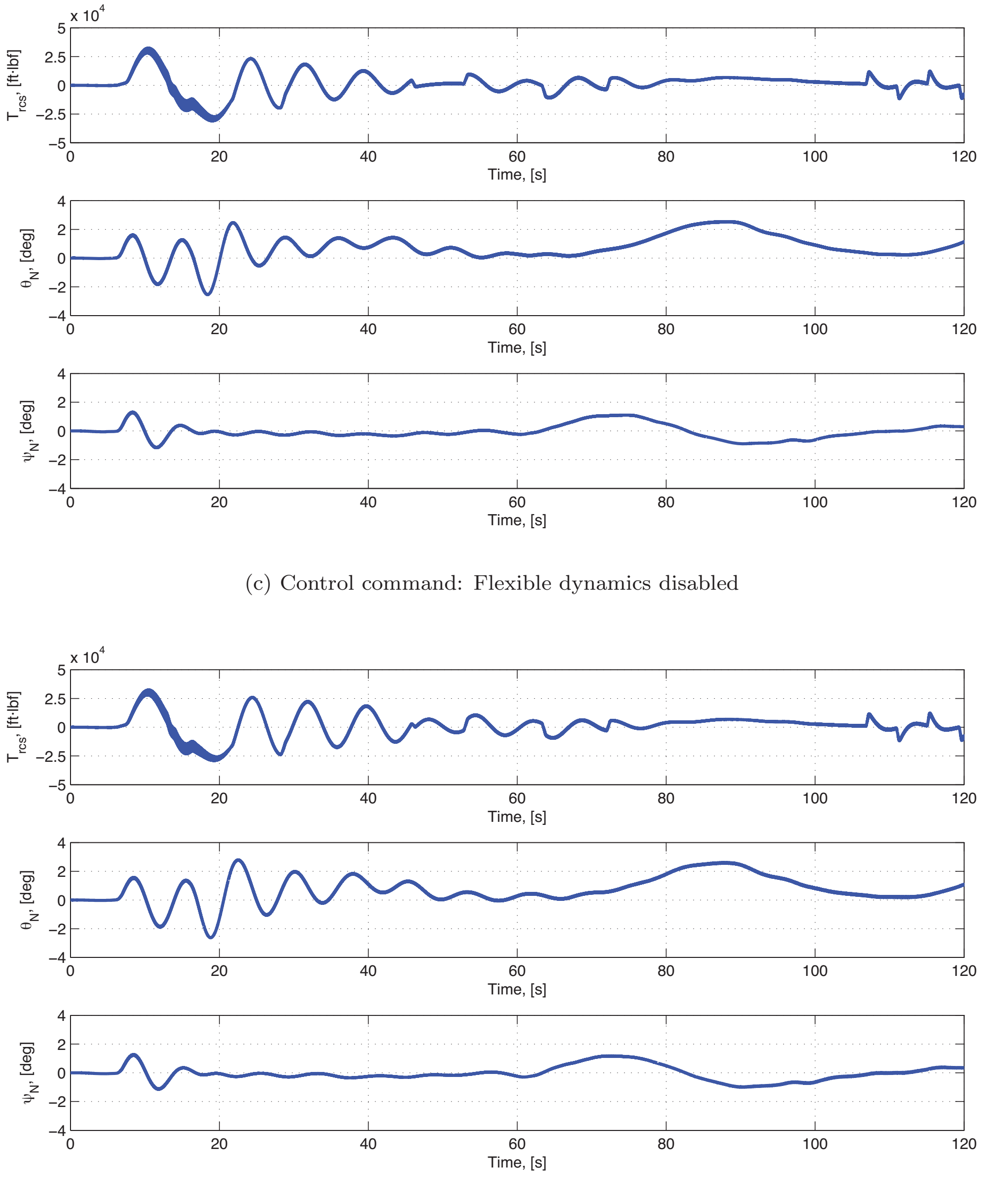
Figure 8: Closed-loop system with baseline PID controller



(a) Tracking errors: Flexible dynamics disabled

(b) Tracking errors: Flexible dynamics enabled

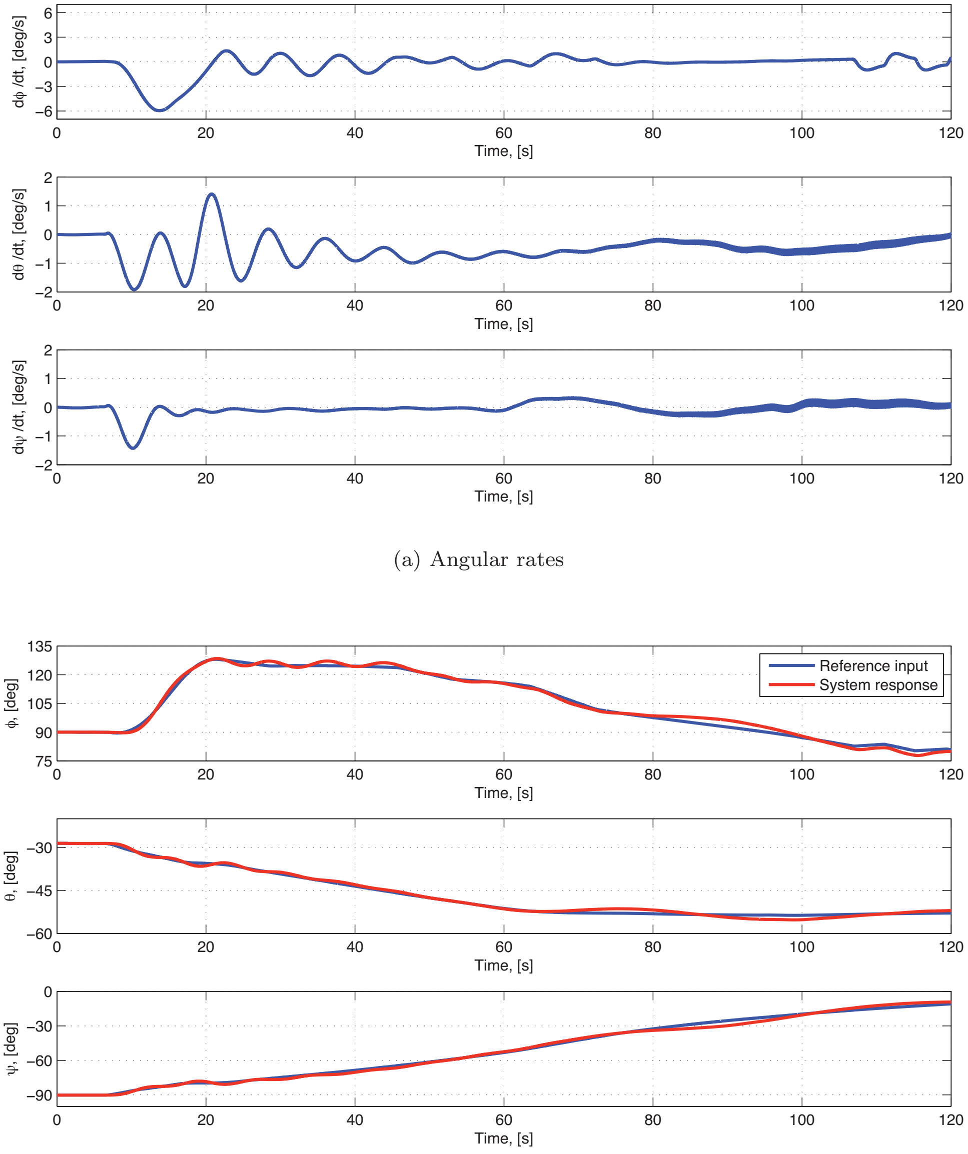
Figure 9: Closed-loop system with $\mathcal{L}_1$ controller



(c) Control command: Flexible dynamics disabled

(d) Control command: Flexible dynamics enabled

Figure 9: Closed-loop system with $\mathcal{L}_1$ controller



(a) Angular rates

(b) Trajectory following capability

Figure 10: Closed-loop system with $\mathcal{L}_1$ controller: flexible dynamics enabled

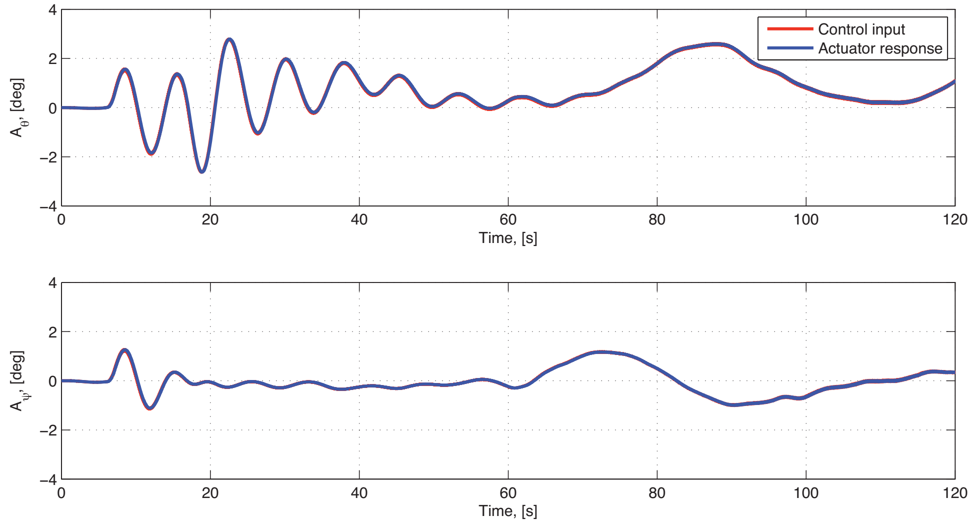


Figure 11: Closed-loop system with $\mathcal{L}_1$ controller: Actuator response

VI. Proposed Validation of $\mathcal{L}_1$ Adaptive Controller

In order for adaptive control to be considered in an operational vehicle, it must undergo a vigorous validation and verification (V&V) process. In this paper, we propose steps to validate adaptive control performance as part of the Ares-I flight test series. We developed an architecture to test and validate the $\mathcal{L}_1$ adaptive controller, and identify potential issues and open problems for flight test part of the V&V process.

A. Proposed architecture for flight test validation

After the entire nominal control system has been validated using the appropriate measures for stability and performance margins prior to flight, we propose a way the adaptive controller itself can be validated in flight. Two identical flight computers would be programmed with the same adaptive flight controller software. One computer would be the flight computer on Ares-I test vehicle the other would remain on the ground. Both of these would be fed with the same sensor data, one directly and the other through telemetry as illustrated in figure 12. The adaptive controller on the ground would receive the same sensor information through telemetry as the one in flight. The command output of the two controllers would be compared, with the output of the ground controller properly adjusted for the telemetry delay. Given the same sensor data both controllers should generate the same output commands. If true, this would provide one measure that the adaptive controller in flight is functioning as predicted. If the controller outputs do not match to within some predefined ϵ , then the interrupt switch, build into the flight architecture as a precaution, would not be closed and the flight adaptive controller would not be allowed into the control loop. If on the other hand everything goes as planned, then the adaptive controller can then be allowed into the loop in the last stages of flight, either as the primary controller or as an augmentation on the baseline controller.

In the next section we propose the type of validation analysis that should be performed prior to flight.

B. Validation procedures

Traditional validation procedures for flight test vehicles include the standard gain and phase/time delay margin analysis, nonlinear Monte Carlo simulations and hardware in the loop simulations for at least a limited test parameter set. In addition to these steps, we propose the following augmentation and redirection. There has been a concerted effort to advance validation techniques for uncertain nonlinear systems. While the problem is by no means solved, there have been some useful advances.

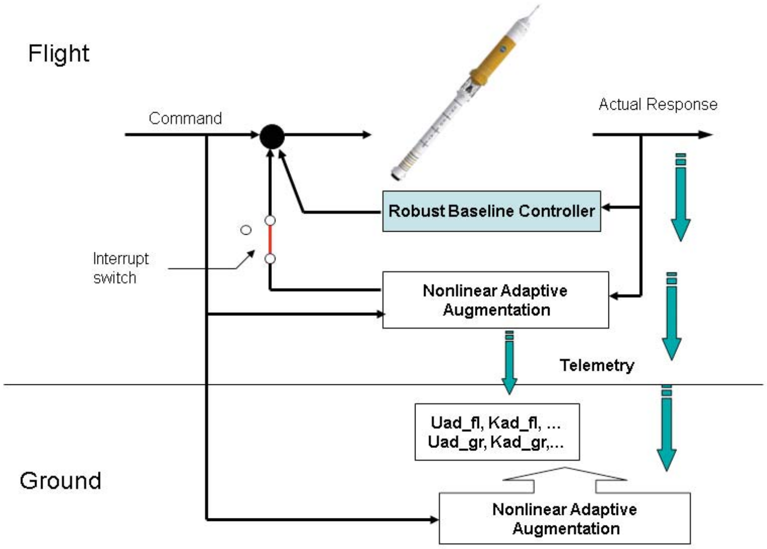


Figure 12: Validation and verification scheme

One of the areas of interest is to uncover, in an efficient way, combinations of environmental conditions, command vectors, and uncertainties that might jeopardize system performance or safety. One of the recently developed tools that helps answer this question is a tool called CAESAR(Control-law Automated Evaluation through Simulation-based and Analytic Routines) software tool for automated testing of complex systems, including systems with intelligence and autonomy in a SIMULINK environment.[9] This tool allows a user to introduce uncertainty blocks in a nonlinear SIMULINK simulation and automatically generate uncertain linear models compatible with the Matlab Robust Control Toolbox analysis tools. This linear uncertainty analysis can be used in a directed Monte Carlo such that the most destabilizing directions and the most sensitive uncertain parameter combinations can be heavily represented in the test set. The tool also allows automated evaluation of batch simulation results based on pre-defined criteria, but can be easily extended with user specified stability and performance metrics. This provides a path for dynamically guiding batch simulation tests.

Another item that we propose to incorporate into pre-flight validation is worst-case analysis for nonlinear trajectories.[15, 16] The work of Tierno et. al. extended robustness analysis techniques for linear systems to nonlinear systems. The method specifically addresses a nonlinear robust performance problem for tracking of aircraft trajectories. A numerical algorithms, which is based on the structured singular value (μ), is used to compute a lower bound on the proposed nonlinear robust performance index. The aircraft tracking problem considered is one of tracking a trajectory in the presence of noise and uncertainty. The objective is to answer the question ”Will the real trajectory, under the worst-case conditions, remain close to the nominal trajectory?” This is considered the robust trajectory tracking problem. The algorithm is similar to the structured singular value (μ) algorithm to compute a lower bound. It uses a power algorithm to compute a lower bound on the performance index associated with the robust trajectory tracking problem has been developed and is well suited to flight certification.

The methods mentioned above would apply to a general flight control validation problem, but there is one additional step we propose for the Ares-I flight test series. Incorporate the pre-flight winds data, supplied by weather balloons, into the full nonlinear simulation and run a pre-defined set of scenarios that include nominal trajectory, maneuvers for adaptive controller demonstration, and errors in vehicle model especially in structural dynamics, to ascertain the likely values for adaptive parameters and gains. These would then be used as a reference set against which the flight adaptive controller gains and parameter estimates would be compared. This would provide yet another safety precaution against the adaptive controller behaving not as expected.

VII. Conclusion

An $\mathcal{L}_1$ adaptive output feedback controller was designed for a generic flexible Crew Launch Vehicle, that is statically unstable with low damping and low bending frequency flexible dynamics. The controller stability and performance were verified through the nonlinear simulations. The results demonstrate that a single design $\mathcal{L}_1$ adaptive output feedback controller is able to handle statically unstable flexible plant with large parametric variation (mass, velocity, aerodynamics properties, etc.) without addition of notch filters and without retuning for different flight conditions along the first stage trajectory. Its feature of fast adaptation achieves the desired performance for the closed-loop system during the entire period of flight.

The paper also offers a set of validation procedures for an adaptive controller in flight test describing several approaches and software tools.

Acknowledgments

The authors would like to thank Dr. Mark Whorton of Marshall Space Flight Center for providing the simulation of a generic flexible space launch vehicle.

This work was sponsored in part by NASA Grants NNX08AB97A and NNX08AC81A, and AFOSR Grant FA9550-08-1-0135.

References

[1] "NASA's Exploration Systems Architecture Study," NASA-TM-2005-214062.

[2] M. Whorton, C. Hall and S. Cook, "Ascent Flight Control and Structural Interaction for the Ares-I Crew Launch Vehicle," *48^{th} AIAA Structures, Structural Dynamics, and Materials Conference*, Honolulu, HI, April 23-26 2007, AIAA-2007-1780.

[3] C. Cao and N. Hovakimyan, "Design and Analysis of a Novel L1 Adaptive Control Architecture, Part I: Control Signal and Asymptotic Stability," *American Control Conference*, 2006, pp. 3397–3402.

[4] C. Cao and N. Hovakimyan, "Design and Analysis of a Novel L1 Adaptive Control Architecture, Part II: Guaranteed Transient Performance," *American Control Conference*, 2006, pp. 3403–3408.

[5] C. Cao and N. Hovakimyan, "Guaranteed Transient Performance with L1 Adaptive Controller for Systems with Unknown Time-varying Parameters and Bounded Disturbances," *American Control Conference*, 2007, pp. 3925–3930.

[6] C. Cao and N. Hovakimyan, "Stability Margins of L1 Adaptive Control Architecture," *American Control Conference*, 2007, pp. 3931–3936.

[7] C. Cao and N. Hovakimyan, "L1 Adaptive Controller for Systems in the Presence of Unmodeled Actuator Dynamics," *IEEE Conference on Decision and Control*, 2007, pp. 891–896.

[8] K. Betts, R. Rutherford, J. McDuffie, M. Johnson, M. Jackson and C. Hall, "Time Domain Simulation of the NASA Crew Launch Vehicle," *AIAA Guidance, Navigation and Control Conference*, Hilton Head, SC, August 20-23 2007, AIAA-2007-6621.

[9] K. Betts, R. Rutherford, J. McDuffie, M. Johnson, M. Jackson and C. Hall, "Stability Analysis of the NASA ARES I Crew launch Vehicle Control System," *AIAA Guidance, Navigation and Control Conference*, Hilton Head, SC, August 20-23 2007, AIAA-2007-6776.

[10] C. Cao and N. Hovakimyan, "Design and Analysis of a Novel $\mathcal{L}_1$ Adaptive Control Architecture with Guaranteed Transient Performance," *IEEE Transactions on Automatic Control*, Vol. 53, No. 3, 2008, pp. 586–591.

[11] C. Cao and N. Hovakimyan, "$\mathcal{L}_1$ Adaptive Output Feedback Controller for Systems of Unknown Dimension," *IEEE Transactions on Automatic Control*, Vol. 53, No. 3, April 2008, pp. 815–821.

[12] Cao, C. and Hovakimyan, N., "Guaranteed Transient Performance with $\mathcal{L}_1$ Adaptive Controller for Systems with Unknown Time-Varying Parameters and Bounded Disturbances: Part I," New York, NY, July 2007, pp. 3925–3930.

[13] Cao, C. and Hovakimyan, N., "Stability Margins of $\mathcal{L}_1$ Adaptive Controller: Part II," New York, NY, July 2007, pp. 3931–3936.

[14] C. Cao, N. Hovakimyan, "L1 Adaptive Output Feedback Controller for Systems of Unknown Relative Degree," *Submited to American Control Conference*, 2009.

[15] A. Bateman, G. Balas, J. Cooper, M. Aiello, "Analytical and Simulation Framework for Performance Validation of Complex Systems." Final technical report contract nnl05aa06c, Barron Associates, December 2006.

[16] J. Tierno. R. Murray, J. Doyle, and I. Gregory, "Numerically efficient robustness analysis of trajectory tracking for nonlinear systems," *J. of Guidance, Control, and Dynamics*, Vol. 20, July-August 1994, pp. 640–647.