A Nonlinear Dynamic Inversion L_1 Adaptive Controller for a Generic Transport Model

Stefan F. Campbell and John T. Kaneshige

***Abstract*—Presented here is an L_1 adaptive control architecture for a generic transport aircraft. At its core, this architecture features a three-axis, non-linear, dynamic-inversion controller. Command inputs for this baseline controller are provided by pilot roll-rate, pitch-rate, and side-slip commands. This paper will first thoroughly present the baseline controller followed by a description of the L_1 adaptive augmentation to this control system. Results are presented via a full-scale, non-linear simulation of NASA's Generic Transport Model (GTM).**

I. Introduction

The Integrated Resilient Aircraft Control project (IRAC) is a part of the NASA Aviation Safety program. A key focus of this project is to investigate adaptive control systems as a risk-mitigating technology for off-nominal conditions. In this work, an L_1 adaptive control architecture is investigated as just such a technology.

To begin, section II presents the baseline control architecture and justifies the design selection. Section III then presents the L_1 adaptive augmentation to this system. Here we briefly show some of the fundamental stability proofs for the architecture, but we refer the interested reader to the large body of L_1 literature for further reference. Section IV presents results from a full non-linear simulation of the control system and provides a discussion of the qualitative meaning of these results. Finally, section V presents a brief summation of this paper's contribution to the community.

The primary intentions of this paper are two fold: to present the L_1 architecture for a complete nonlinear dynamic inversion controller (a novel contribution to the community in as far as the authors are aware) and to demonstrate the architecture on a complete nonlinear flight simulation. Future work will be aimed at more thoroughly assessing the strengths and weaknesses of this control strategy.

Manuscript received September 16, 2009.

S. F. Campbell is with SGT Inc., Greenbelt, MD 20770 USA (e-mail: stefan.f.campbell@nasa.gov).

J. T. Kaneshige is with NASA, Moffett Field, CA 94035 USA. (e-mail: john.t.kaneshige@nasa.gov).

II. Baseline Controller: Nonlinear Dynamic Inversion

A. Control Selection Justification

The baseline controller for this work is a full non-linear dynamic inversion controller based in large part on [1], [2], and [3]. This control architecture is chosen for several key reasons. First, this control approach can be effectively implemented as evinced by the recent selection of a dynamic inversion controller for the F-35 aircraft. Second, and more importantly, a dynamic inversion controller offers a cost and time effective way to develop a control system; with appropriate modeling, a full-flight control system can be quickly and efficiently developed for research studies, as contrasted with a more time intensive traditional gain-scheduled controller. This ultimately facilitates the rapid evaluation and testing of multiple adaptive control systems over a large range of flight conditions.

B. Control Architecture

The general (and well known) rigid body dynamics for an aircraft are presented below:

$$\tau = I\dot{\omega} + \omega \times I\omega \,. \tag{1}$$

Here, $\omega \in \Re^{3X1}$ and represents a vector of the roll (p), pitch (q), and yaw (r) rates. By modeling the torques on the aircraft using traditional aerodynamic stability derivatives (includes p, q, and r derivatives), (1) may be decomposed into the following convenient form:

$$\dot{\omega} = A\omega + Gz + Bu \tag{2}$$

For this decomposition, the matrices A ($\in \Re^{3X3}$), G ($\in \Re^{3X7}$), and B ($\in \Re^{3X3}$) are time-varying. Moreover, the aircraft's control allocation tables are incorporated into (1) such that the control vector u is $\in \Re^{3X1}$ and represents the three, non-dimensional lateral, longitudinal, and directional control signals (limited from -1 to 1). The vector z represents a non-linear combination of ω, specifically pq, qr, rp, (p^2-q^2), (r^2- p^2), and (q^2 - r^2), as well as a bias term (which accounts for the τ dependence on slowly changing variables such as angle of attack and side-slip).

A general non-linear dynamic inversion control law is then

$$u = \hat{B}^{-1}(\dot{\omega}_d - \hat{A}\omega - \hat{G}z)\,. \tag{3}$$

For clarity, over-hats have been used to denote that parameters in the dynamic inversion control law are estimated; look-up tables provide these matrices in real-time operation. The reader should be careful not to confuse the dynamic inversion parameters with the real-time, adaptively estimated parameters that will be introduced in later sections. Finally, the subscript *d* denotes a *desired* value.

To generate the desired values in (3), each of the three directional axes is treated independently. For the longitudinal and lateral axes, the pilot's stick commands are interpreted as pitch and roll *rate* commands. These commands are then filtered to produce a tracking signal for a linear PI control law. The control signal is further augmented by a feedforward term from the filter states. The resulting control signals are the desired pitch and roll rates for the dynamic inversion in (3).

For the directional axis, the pilot's pedal inputs are not interpreted directly as yaw rate commands, but instead as sideslip commands. As a result, an additional dynamic inversion is employed to generate the necessary yaw rate command; this is similar to the work done in [4] in which a two stage *slow* and *fast* dynamic inversion architecture was proposed (in this work the fast dynamic inversion is similar to the dynamic inversion shown in (3)). More precisely, the pilot's sideslip command is filtered to produce a tracking command for a proportional controller. This proportional control signal is augmented with a feedforward term from the filter states to generate a *desired sideslip* command. From this, a dynamic inversion is performed to generate a yaw rate command for the *fast* dynamic inversion in (3). The *desired* yaw rate command is then generated identically to the lateral and longitudinal axes, i.e. the *desired* yaw rate is generated from a linear PI controller augmented with a feedforward term.

For the slow dynamic inversion, the following well-known relationship is used as a starting point for derivation:

$$\begin{aligned}\dot{\beta} = \frac{1}{mV}[\, & D\sin(\beta) + Y\cos(\beta) - X_T\cos(\alpha)\sin(\beta) + \\ & mg(\cos\alpha\sin\beta\sin\theta + \cos\beta\sin\phi\cos\theta - \\ & \sin\alpha\sin\beta\cos\phi\cos\theta + p\sin\alpha - r\cos\alpha)]\end{aligned} \tag{4}$$

This expression is simplified to the more tractable relationship shown in (5).

$$\dot{\beta} \approx p\sin(\alpha) - r\cos(\alpha) + (g/V_t)\cos(\theta)\sin(\phi). \tag{5}$$

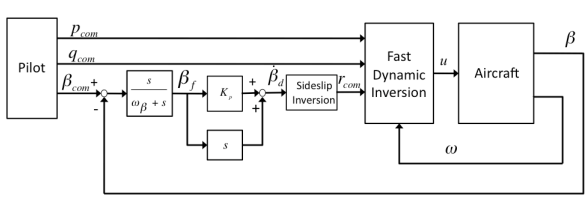

Fig. 1. High-level base-line control architecture. Here the dependency of the sideslip inversion on additional states (α, *θ*, *ϕ*, *p*, and V_t) is not illustrated.

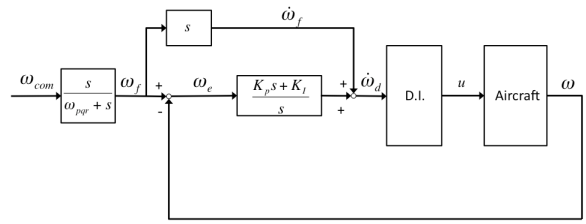

Fig. 2. Presented here is the inner-loop fast dynamics controller. The *D.I.* block contains the equation presented in (3).

The yaw rate command is then determined from (5), as shown in (6).

$$r_{com} = -(\dot{\beta}_d - p\sin(\alpha) - (g/V_t)\cos(\theta)\sin(\phi))/\cos(\alpha) \tag{6}$$

The complete architecture is presented in both Fig. 1 and Fig. 2. Specifically, Fig. 1 illustrates the outer loop control architecture while Fig. 2 shows the fast dynamic inversion using all three of the pitch, roll, and yaw rate commands. It is worth noting that the desired angular accelerations for this architecture (as diagramed in Fig. 2) are given below in (7).

$$\dot{\omega}_d = K_p(\omega_f - \omega) + K_I\int_0^t(\omega_f - \omega)d\tau + \dot{\omega}_f - u_{ad} \tag{7}$$

Here, the output of the command filter is expressly represented with a subscript *f*. Moreover, it should be noted that both K_p and K_I are *constant* diagonal, 3*x*3 matrices. Though not yet explicitly defined or illustrated in Fig. 2, the adaptive compensation term u_{ad} has been introduced in (7).

As a final note on the general architecture, the pilot's commanded pitch-rate is augmented by the level turn-compensation term $g\sin^2\phi/V\cos\phi$. Qualitatively, this helps keep the nose of the aircraft up during bank and roll maneuvers.

III. *L1* Augmentation

The following *L1* development follows as closely as possible the approach outlined in [5-9]. This work is different in that the baseline controller is a full nonlinear dynamic inversion controller, as discussed in the previous sections.

A. Preliminaries

To begin defining the L_1 adaptive augmentation, we manipulate the general system model of (2) by incorporating the true matrix values *A*, *B*, and *G* with their controller estimates.

$$\dot{\omega} = \hat{A}\omega + \hat{G}z + \hat{B}u + \Delta A\omega + \Delta Gz + \Delta Bu \tag{8}$$

In (8), Δ denotes the difference between the true matrices and the estimated values used by the controller. To further develop the controller, the result in (8) is transformed into the form presented in (9).

$$\dot{\omega} = \hat{A}\omega + \hat{B}(u + f) \tag{9}$$

The term f is here introduced for convenience and comprises the modeling uncertainty; this is the uncertainty for which the L_1 adaptive augmentation is intended to compensate. It should be further noted that $\hat{B}$ is assumed for a nominal aircraft and is thus taken as full-rank throughout the entirety of this paper. Comparing (8) and (9), the term f is then defined as

$$f = \hat{B}^{-1}(\Delta A\omega + \Delta Bu + \Delta Gz) + \hat{B}^{-1}\hat{G}z. \tag{10}$$

In much of the adaptive control literature an additional term is added to (10) to approximate non-parametric uncertainty. This term is significant from a stability proof and analysis perspective, but is ignored here for the purposes of developing the correct control laws and L_1 predictor system.

To facilitate further development, the command filter in Fig. 2 is represented in state space form as below in (11).

$$\dot{\omega}_f = A_m\omega_f + B_m r \tag{11}$$

Here the *reference input* r is provided from the pilot commanded roll and pitch rates and the outer-loop side-slip controller. For this work, the filter is assumed first order, as illustrated in Fig. 2.

B. L_1 Predictor System Development

For the $L1$ development presented here, the predictor system is defined using the following state vector:

$$x = \begin{bmatrix} \left(\int_0^t \omega_f d\tau\right)^T & \omega_f^T & \left(\int_0^t \omega d\tau\right)^T & \omega^T \end{bmatrix}^T. \tag{12}$$

Combining (3), (7), and (11) the control law is rewritten below as

$$u = K_x^T x + K_r^T r - K_z^T z - \hat{B}^{-1}u_{ad}, \tag{13}$$

where

$$K_x^T = \begin{bmatrix} \hat{B}^{-1}K_I & \hat{B}^{-1}(K_p + A_m) & -\hat{B}^{-1}K_I & -\hat{B}^{-1}(K_p + \hat{A}) \end{bmatrix}, \tag{14}$$

$$K_r^T = \hat{B}^{-1}B_m, \tag{15}$$

and

$$K_z^T = \hat{B}^{-1}\hat{G}. \tag{16}$$

It should be observed that $x \in \Re^{12X1}$, $K_x \in \Re^{12X3}$, $K_r \in \Re^{3X3}$, $K_z \in \Re^{7X3}$, $u_{ad} \in \Re^{3X1}$, and $u \in \Re^{3X1}$.

The system in (9), given the above definitions, is easily written as

$$\begin{aligned} \dot{x} = \bar{A}x + \bar{B}_m r + \bar{B}[& K_x^T x + K_r^T r - \hat{B}^{-1}u_{ad} \\ & + \hat{B}^{-1}(\Delta\tilde{A}x + \Delta Bu + \Delta Gz)] \end{aligned} \tag{17}$$

In (17), the following definitions have been introduced:

$$\bar{A} = \begin{bmatrix} 0 & I & 0 & 0 \\ 0 & A_m & 0 & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & \hat{A} \end{bmatrix}, \tag{18}$$

$$\bar{B}_m = \begin{bmatrix} 0 & B_m^T & 0 & 0 \end{bmatrix}^T, \tag{19}$$

$$\bar{B} = \begin{bmatrix} 0 & 0 & 0 & \hat{B}^T \end{bmatrix}^T, \tag{20}$$

and

$$\Delta\tilde{A} = \begin{bmatrix} 0 & 0 & 0 & \Delta A \end{bmatrix}. \tag{21}$$

To complete the manipulation of the system dynamics, the Δ terms are expanded, and (17) can be once again rewritten as

$$\begin{aligned} \dot{x} = & (\bar{A} + \bar{B}K_x^T)x + (\bar{B}_m + \bar{B}K_r^T)r + B_l[(\Delta BK_x^T + \Delta\tilde{A})x \\ & + (\Delta BK_r^T)r - B\hat{B}^{-1}u_{ad} + (G - BK_z^T)z] \end{aligned}, \tag{22}$$

where

$$B_l^T = \begin{bmatrix} 0 & 0 & 0 & I \end{bmatrix},\ B_l^T \in \Re^{3X12}. \tag{23}$$

For further simplicity, (22) is rewritten with a simpler notation.

$$\begin{aligned} \dot{x} = & A_r x + B_r r + \\ & B_l(W_1^T x + W_2^T r - W_3^T u_{ad} + W_4^T z) \end{aligned} \tag{24}$$

The above result employs the proceeding definitions:

$$\begin{aligned} A_r &= (\bar{A} + \bar{B}K_x^T) \\ B_r &= (\bar{B}_m + \bar{B}K_r^T) \\ W_1 &= \Delta BK_x^T + \Delta\tilde{A} \\ W_2 &= \Delta BK_r \\ W_3 &= B\hat{B}^{-1} \\ W_4 &= (G - BK_z^T) \end{aligned} \tag{25}$$

In order to be precise on matrix dimensions, we note that $W_1 \in \Re^{12X3}$, $W_2 \in \Re^{3X3}$, $W_3 \in \Re^{3X3}$, and $W_4 \in \Re^{7X3}$. Moreover, each of the W parameters in (25) is time varying. Conversely, the terms A_r and B_r are time invariant. Finally, A_r is a Hurwitz matrix.

Though seemingly excessive, the system manipulations of (2) in order to achieve (24) have placed the system in a more

amenable form for defining the L_1 predictor system. Specifically, the L_1 predictor system is then simply manipulated from (24) to be the following:

$$\begin{aligned}\dot{\hat{x}} &= A_r\hat{x} + B_r r + \\ &B_l(\hat{W}_1^T x + \hat{W}_2^T r - \hat{W}_3^T u_{ad} + \hat{W}_4^T z)\end{aligned} \quad (26)$$

C. Adaptive Control Laws

To complete the adaptive architecture, we here define the adaptive control laws. The L_1 adaptive control signal is given as:

$$\dot{u}_{ad} = k_{L1}(\hat{W}_1^T x + \hat{W}_2^T r - \hat{W}_3^T u_{ad} + \hat{W}_4^T z) \quad (27)$$

In (27) the gain k_{L1} is a tunable parameter selected by the designer. For this application, k_{L1} is a diagonal matrix with dimensions $3X3$.

The adaptive laws for estimating the unknown weights W require an examination of the error dynamics

$$\dot{e}_r = \dot{\hat{x}} - \dot{x}. \quad (28)$$

Investigation of the dynamics in (28) reveals that the filter commands (ω_f) are perfectly reproduced by the predictor states, assuming identical initial conditions (a reasonable assumption). This is obvious from (24) as the predicted values of the filter states are not a function of the control signal or the system uncertainty. As a consequence, the error dynamics can be reduced to

$$\dot{e}_r = A_e e_r + B_e(\Delta W_1^T x + \Delta W_2^T r - \Delta W_3^T u_{ad} + \Delta W_4^T z) \quad (29)$$

where

$$e_r = \begin{bmatrix}\left(\int_0^t (\hat{\omega} - \omega)d\tau\right)^T & (\hat{\omega} - \omega)^T\end{bmatrix}^T, \quad (30)$$

$$A_e = \begin{bmatrix} 0 & I \\ -K_I & -K_p \end{bmatrix}, \quad (31)$$

and

$$B_e^T = [0 \quad I]. \quad (32)$$

To further elucidate the notation in (29), we note that Δ denotes the difference between the true weights W^* and their estimated values. Finally, $\hat{\omega}$ represents the $3X1$ vector of the predicted pitch, roll, and yaw rates. Given (29 – 32), the adaptive laws are then defined as in (33).

$$\begin{aligned}\dot{\hat{W}}_1 &= \Gamma_1 \mathrm{Proj}(\hat{W}_1, -x e_r^T P B_e) \\ \dot{\hat{W}}_2 &= \Gamma_2 \mathrm{Proj}(\hat{W}_2, -r e_r^T P B_e) \\ \dot{\hat{W}}_3 &= \Gamma_3 \mathrm{Proj}(\hat{W}_3, -u_{ad} e_r^T P B_e) \\ \dot{\hat{W}}_4 &= \Gamma_4 \mathrm{Proj}(\hat{W}_4, -z e_r^T P B_e)\end{aligned} \quad (33)$$

As is consistent with most model adaptive control formulations, P is the solution to the Lyapunov equation $A_e^T P + P A_e = -Q$ where both P and Q are positive definite, symmetric matrices and Q is a tunable parameter. The "Γ" terms are the respective adaptive learning rates for each unknown parameter. It is here worth noting that the matrix in (31) is constant for the error dynamics irrespective of the fact that the system is actually time varying.

D. Theoretical Analysis

To show the correctness of the development, three theoretical issues are addressed here: the boundedness of the prediction error, the boundedness of the predictor system states, and the strictly proper nature of the L_1 filter. As aforementioned, all uncertainty has been treated as parametric, thereby simplifying the presentation here. If non-parametric uncertainty were considered, the theoretical development would be further complicated. However, the results would still match the existing body of L_1 literature.

1) Bounded Error Dynamics

The error dynamics are already presented in (29). To show that these errors are bounded, we choose the Lyapunov function below.

$$\begin{aligned}V &= e_r^T P e_r + trace(\Delta W_1^T \Gamma_1^{-1} \Delta W_1) + trace(\Delta W_2^T \Gamma_2^{-1} \Delta W_2) \\ &+ trace(\Delta W_3^T \Gamma_3^{-1} \Delta W_3) + trace(\Delta W_4^T \Gamma_4^{-1} \Delta W_4)\end{aligned}. \quad (34)$$

Taking the derivative of (34) and substituting the error dynamics in (29) yields the following:

$$\begin{aligned}\dot{V} &= -e_r^T Q e_r - \hat{e}_p^T Q_p \hat{e}_p - 2trace(\Delta W_1^T \Gamma_1^{-1} \dot{W}_1) \\ &- 2trace(\Delta W_2^T \Gamma_2^{-1} \dot{W}_2) - 2trace(\Delta W_3^T \Gamma_3^{-1} \dot{W}_3) \\ &- 2trace(\Delta W_4^T \Gamma_4^{-1} \dot{W}_4)\end{aligned}. \quad (35)$$

Because the weights W are time varying, the stability proof requires bounds on each of the weights as well as bounds on the derivatives of the weights. More precisely,

$$W_1(t) \in \overline{W}_1,\ W_2(t) \in \overline{W}_2,\ W_3(t) \in \overline{W}_3,\ W_4(t) \in \overline{W}_4 \quad \forall t, \quad (36)$$

where $\overline{W}$ denotes a known compact set and

$$\|W_1(t)\| \le d_1,\ \|W_2(t)\| \le d_2,\ \|W_3(t)\| \le d_3,\ \|W_4(t)\| \le d_4 \quad \forall t. \quad (37)$$

If these exist, the stability proof outlined in [16] establishes that the tracking error in (28) is *uniformly ultimately bounded*. To establish this result, the use of the projection operator (in (33)) is required to ensure that each adaptive weight error is bounded. The result in [16] is somewhat simplified here by the nature of the error dynamics in (29); because A_e is a constant matrix, the Lyapunov matrix P is a constant for all time.

2) Bounded Predictor System

As noted in [7], it is necessary to establish that the true system is tracking a stable predictor. Given that the tracking error is bounded, the true system states can be shown bounded if the prediction model is shown bounded. To establish this, we inspect the closed loop reference system in which fast adaptation (high-gain adaptation) ensures that the adaptive weights are known instantaneously. With this assumption, the closed loop reference system can be formulated as below.

$$\begin{aligned}\dot{x}_{ref} &= A_r x_{ref} + \\ &B_I(B_r r + B_l(W_1^T x_{ref} + W_2^T r - W_3^T u_{ad} + W_4^T z))\end{aligned} \tag{38}$$

In (38), B_I is a 12x12 identity matrix. Solving (27) for u_{ad} and substituting the result into (38) yields the following system:

$$x_{ref} = HB_r r + H\overline{C}B_l(W_1^T x_{ref} + W_2^T r + Gx_{ref}) + H_{in}x_o. \tag{39}$$

In (39) H is the input-output map of the system

$$\dot{x}_{ref} = A_r x_{ref} + B_I u_{ref}. \tag{40}$$

Additionally, G is the input-output map for non-linear terms $W_4^T z$, more precisely

$$(W_4^T z) = Gx_{ref}. \tag{41}$$

Finally, H_{in} is the input output map for the initial condition x_o and $\overline{C}$ is the input output map for $(1-C(s))$, where $C(s)$ is the adaptive filter defined below.

$$C(s) = (sI + k_{L1}W_3^T)^{-1}k_{L1}W_3^T \tag{42}$$

The states x_{ref} are then bounded if the following condition is satisfied:

$$\left\|H\overline{C}\right\|_1 < \frac{1}{L_1 + L_2}, \tag{43}$$

where L_1 and L_2 are the £$_1$ norms of the terms W_1^T and G, respectively. The interested reader will find much greater detail on this theoretical analysis in [7] and [16]. It should be noted, however, that the filter parameters ultimately shape the performance of the reference model in (38).

3) Strictly Proper Filter C(s)

To ensure that the filter $C(s)$ is strictly proper, several assumptions are made about the parameter W_3. From (25), W_3 is defined as below:

$$W_3 = B\hat{B}^{-1}. \tag{44}$$

The true B matrix and the dynamic inversion estimate of the B matrix are then both assumed to be diagonally dominant, a reasonable assumption for a transport aircraft. The uncertainty in $\hat{B}$ may then be approximated by the following relationship:

$$B \approx \hat{B}\Lambda. \tag{45}$$

In (45), Λ is a purely diagonal matrix. Given (45) and (44) and the original assumption that both B and $\hat{B}$ are diagonally dominant, then W_3 can be approximated by the relationship in (46).

$$W_3 \approx \Lambda \tag{46}$$

Treating W_3 as a diagonal matrix then ensures that $C(s)$ (in (43)) is a diagonal matrix of strictly proper first order filters.

IV. Simulation Results

To investigate the functionality of the system, we use a full-scale version of NASA's Generic Transport Model (GTM - developed at NASA Langley Research Center); this model is intended to represent a scaled version of a two engine commercial transport aircraft [14]. Using real aerodynamic data, the above controller is simulated in a full-nonlinear simulation environment. As an added caveat, data for the dynamic inversion is collected using the vortex-lattice code base VorView [15]; this ensures a separation between the true aircraft model and the dynamic inversion (i.e. we do no circularly use the simulation aerodynamic data for the dynamic inversion controller). For the purposes of this paper, we consider an aircraft performing a doublet maneuver in roll, pitch, and sideslip. The results for a nominal aircraft operating with the baseline controller are presented in Fig. 3.

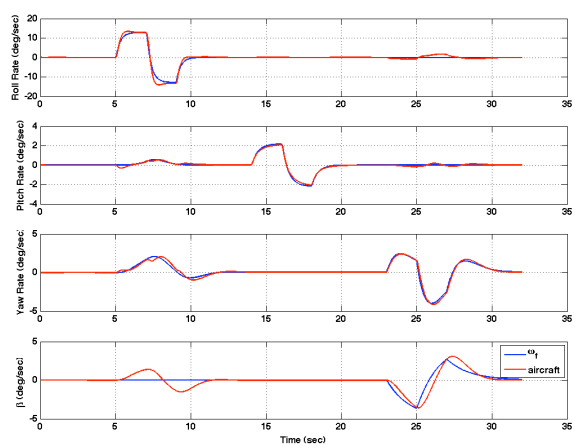

Fig. 3. Nonlinear dynamic inversion controller for GTM performing lateral, longitudinal, and directional doublet tracking filtered command signal.

The results in Fig. 3 illustrate the effectiveness of the baseline controller. To further exercise the system, however, we repeat the simulation by failing the stab and the left horizontal elevator. As shown in Fig. 4., the L_1 Adaptive controller reduces the initial pitch rate oscillation and significantly improves tracking performance. The baseline controller, in this example, also benefits from its initial poor

performance in that the aircraft will lose altitude and increase speed, allowing the control system to gain additional actuator effectiveness; the L_1 architecture does not benefit from a similar phenomenon. The L_1 overshoot in Fig. 5 also illustrates a key point: the L_1 system will track the reference model in (38), not the filtered states of (11). As a result, the choice of learning rates and k_{L1} directly impact the target performance by shaping the closed loop reference model. As a final point, the close-loop reference system used to establish stability in (38) requires that the L_1 architecture be implemented with a gain significantly larger than may be typically considered for an MRAC based adaptive controller (i.e. as compared to the MRAC system in [2]); the large L_1 adaptive gain ensures that the predictor system behaves as the closed loop reference system in (38).

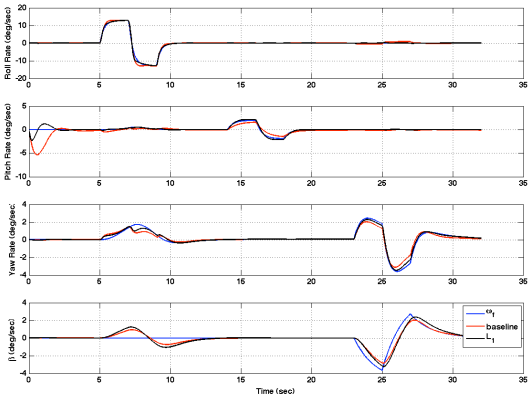

Fig. 4. Nonlinear dynamic inversion controller and L_1 for GTM aircraft and failed stabilator and left elevator.

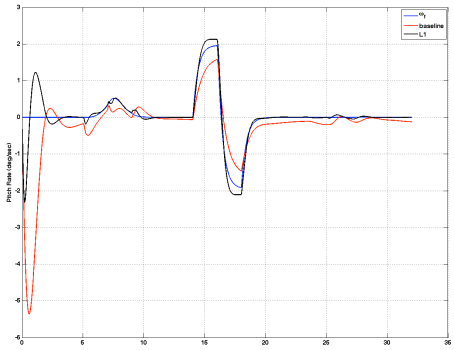

Fig. 5. Enlarged view of pitch performance in Fig. 4.

During development, experimentation revealed that actuator saturation can be destabilizing to the L_1 system if the predictor model did not include any saturation compensation (whether it be adaptive or as a fixed gain). Techniques for this may be found in [10-13].

V. Conclusion

This paper has presented a baseline nonlinear dynamic inversion controller with an L_1 adaptive augmentation component incorporating a notion of actuator saturation. Through the presentation of simulation results, we have shown that this architecture can improve tracking performance as compared to the un-augmented controller.

Acknowledgment

Thanks must be given to Enric Xargay (U. of Illinois) for assistance identifying errors in our L_1 development. Additional thanks must be given to NASA Adaptive Control and Evolvable Systems group members Maryam Bakhtiari-Nejad, Susan Frost, Kalmanje Krishnakumar, Greg Larchev, Nhan Nguyen, Shivanjli Sharma, and Vahram Stepanyan.

References

[1] R. T. Rysdyk and A. J. Calise, "Fault Tolerant Flight Control via Adaptive Neural Augmentation", *AIAA Guidance, Navigation, and Control Conference*, Aug. 1998.

[2] J. T. Kaneshige, J. Bull, and J. J. Totah, "Generic Neural Flight Control and Autopilot System," *AIAA Guidance, Navigation, and Control Conference*, Aug. 2000.

[3] N. Nguyen, K. Krishnakumar, J. T. Kaneshige, and P. Nespeca, "Flight Dynamics and Hybrid Adaptive Control of Damaged Aircraft", AIAA Journal of Guidance, Control, and Dynamics, Vol. 31, No. 3, pp. 751-764, 2008.

[4] S. A. Snell, D. F. Enns, and W. L. Garrard Jr., "Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, July-August 1992.

[5] C. Cao, and N. Hovakimyan, "Design and Analysis of a Novel L1 Adaptive Controller, Part I: Control Signal and Asymptotic Stability," *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, June 14-16, 2006.

[6] C. Cao, and N. Hovakimyan, "Design and Analysis of a Novel L1 Adaptive Controller, Part II: Guaranteed Transient Performance," *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, June 14-16, 2006.

[7] C. Cao and N. Hovakimyan, "Novel L1 Neural Network Adaptive Control Architecture with Guaranteed Transient Performance," *IEEE Transactions on Neural Networks*, Vol. 18, No. 4, July 2007.

[8] V. V. Patel, C. Chengyu, N. Hovakimyan, K. A. Wise, and E. Lavretsky, "L1 Adaptive Controller for Tailless Unstable Aircraft," *Proceedings of the 2007 American Control Conference*, New York City, NY, July 11-13, 2007.

[9] J. Wang, V. V. Patel, C. Chengyu, and N. Hovakimyan, "L1 Adaptive Neural Network Controller for Autonomous Aerial Refueling with Guaranteed Transient Performance", *AIAA Guidance, Navigation, and Control Conference*, Keystone, CO, Aug. 21-24, 2006.

[10] S. P. Karason and A. M. Annaswamy, "Adaptive Control in the Presence of Input Constraints", *IEEE Transactions on Autmoatic Control*, Vol. 39, No. 11, November 1994.

[11] E. Lavretsky and N. Hovakimyan, "Stable Adaptation in the Presence of Input Contraints", *Systems & Control Letters 56,* 2007, pp. 722 - 729.

[12] E. Lavretsky and N. Hovakimyan, "Stable Adaptation in the Presence of Actuator Constraints with Flight Control Applications," *Journal of Guidance, Control, and Dyanmics,* Vol. 30, No. 2, March – April, 2007.

[13] D. Li, N. Hovakimyan, and C. Cao, "L1 Adaptive Controller in the Presence of Input Saturation," *AIAA Guidance, Navigation, and Control Conference*, Chicago, IL, Aug. 10-13, 2009.

[14] T.L. Jordan, W. M. Langford, et al., "Development of a Dynamically Scaled Generic Transport Model Testbed for Flight Research Experiments," *AUVSI Unmannded Systems North America 2004,* Arlington, VA, 2004.

[15] J. J. Totah, D. J. Kinney, J. T. Kaneshige, and S. Agabon, "An Integrated Vehicle Modeling Environment," *AIAA Atmospheric Flight Mechanics Conference and Exhibit,* Portland, OR, Aug 9-11, 1999.

[16] J. Wang, N. Hovakimyan, C. Cao, "L_1 Adaptive Augmentation of Gain-Scheduled Controller for Racetrack Maneuver in Aerial Refueling," *AIAA Guidance, Navigation, and Control Conference,* Aug. 2009, Chicago, IL.