Projection Operator: A Step towards Certification of Adaptive Controllers

Gregory V. Larchev[1]
UARC, NASA Ames Research Center, Moffett Field, CA 94035

Stefan F. Campbell[2]
SGT Inc., NASA Ames Research Center, Moffett Field, CA 94035

and

John T. Kaneshige[3]
NASA Ames Research Center, Moffett Field, CA 94035

One of the major barriers to wider use of adaptive controllers in commercial aviation is the lack of appropriate certification procedures. In order to be certified by the Federal Aviation Administration (FAA), an aircraft controller is expected to meet a set of guidelines on functionality and reliability while not negatively impacting other systems or safety of aircraft operations. Due to their inherent time-variant and non-linear behavior, adaptive controllers cannot be certified via the metrics used for linear conventional controllers, such as gain and phase margin. Projection Operator is a robustness augmentation technique that bounds the output of a non-linear adaptive controller while conforming to the Lyapunov stability rules. It can also be used to limit the control authority of the adaptive component so that the said control authority can be arbitrarily close to that of a linear controller. In this paper we will present the results of applying the Projection Operator to a Model-Reference Adaptive Controller (MRAC), varying the amount of control authority, and comparing controller's performance and stability characteristics with those of a linear controller. We will also show how adjusting Projection Operator parameters can make it easier for the controller to satisfy the certification guidelines by enabling a tradeoff between controller's performance and robustness.

I. Introduction

Adaptive controllers have long been of much interest to the aerospace research community. Many such controllers have been implemented on simulated aircraft with promising results (Ref. 1-3). Adaptive controllers have also been successfully flight-tested on a number of flight vehicles, such as X-36 Tailless fighter (Ref. 10), and JDAM guided munitions (Ref. 11). Some of the main benefits of adaptive controllers are their ability to incorporate system uncertainties without requiring explicit system identification. These benefits are particularly noticeable when such controllers are applied to poorly-modeled systems, such as damaged aircraft. Nevertheless, certain FAA certification guidelines are more difficult to achieve for adaptive flight control software than for conventional flight control software. At this time, no adaptive flight control systems have been certified by the FAA for use in the national air space (Ref. 8).

This paper is organized as follows. Section 2 describes the general design of an adaptive controller used in the experiments. Section 3 presents an overview of the challenges adaptive controllers have traditionally faced in obtaining FAA certification. Section 4 introduces the Projection Operator. Section 5 illustrates the application of limiting adaptive control authority to a simple control problem, and discusses the effect on stability margins. Section

[1] Computer Scientist, Intelligent Systems Division, MS 269-3
[2] Controls Engineer, Intelligent Systems Division, MS 269-1
[3] Computer Engineer, Intelligent Systems Division, MS 269-1

6 presents experimental results for a simulated aircraft, and analyzes them in the context of limited control authority. Section 7 summarizes the results and discusses the potential direction of future work.

II. Adaptive Controller Design

The dynamics of a generic system are described as

$$\dot{x} = Ax + Bu + Gz + f(x,u,z) \tag{1}$$

where z represents the linearized non-linear terms. The unknown terms in the equation are represented by $f(x,u,z)$. The reference model is

$$\dot{x}_m = A_m x_m + B_m r \tag{2}$$

where r is the commanded pilot input.

It is then possible to define the desired acceleration as

$$\dot{x}_d = \dot{x}_m + u_{bl} - u_{ad} \tag{3}$$

where u_{bl} is the baseline control input defined as

$$u_{bl} = K_p(x_m - x) + K_i \int_0^t (x_m - x) d\tau \tag{4}$$

and u_{ad} is the adaptive control input intended to cancel the uncertainty.

Once the desired acceleration is computed, the final control input is determined via dynamic inversion

$$u = B^{-1}(\dot{x}_d - Ax - Gz) \tag{5}$$

The error dynamics for the system are

$$x_e = x_m - x \tag{6}$$

$$\dot{x}_e = -u_{bl} + u_{ad} - f(x,u,z) \tag{7}$$

which can be rewritten in a state-space form

$$e = \begin{bmatrix} \int_0^t x_e d\tau \\ x_e \end{bmatrix} \tag{8}$$

$$\Delta = u_{ad} - f(x,u,z) \tag{9}$$

$$A_c = \begin{bmatrix} 0 & I \\ -K_i & -K_p \end{bmatrix} \tag{10}$$

$$b = \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{11}$$

$$\dot{e} = A_c e + b(\Delta) \tag{12}$$

For a Lyapunov function described by the above equations, it is possible to prove the convergence of the system error to zero using the Barbalat's Lemma, if the following condition is satisfied

$$A_c^T P + P A_c = -Q_c \tag{13}$$

where P and Q_c are symmetric and positive definite.

The adaptation signal for a standard MRAC is defined as

$$u_{ad} = W_{ad}^T \beta_{ad}(x,u,z) \tag{14}$$

$$\dot{W}_{ad} = -\Gamma \beta_{ad}(x,u,z) e^T P b \tag{15}$$

where W_{ad} represents estimated weights, β_{ad} denotes selected basis functions, and Γ is the adaptive gain.

Figure 1 shows the generic MRAC structure. The baseline controller is linear, and based on the dynamic inversion control law. The adaptive portions provide augmentation to the baseline controller. Depending on the particular controller type used, adaptive methods can modify either the control input or the system parameters used by the dynamic inversion.

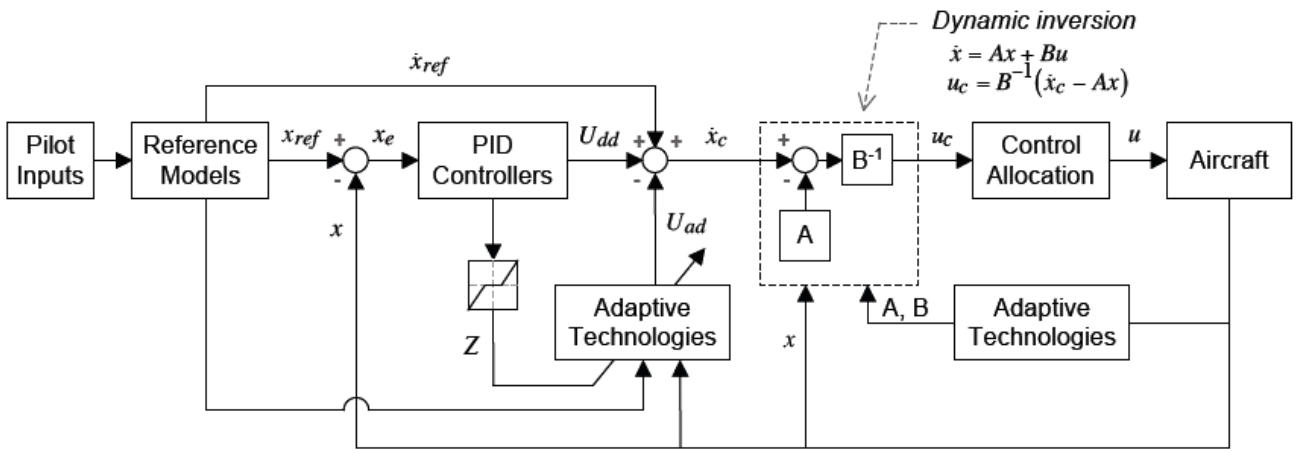

Figure 1. Generic MRAC structure.

The adaptive contribution is denoted by signal $U_{ad} \in R^n$, where u_{ad}^i is the *i*-th element of U_{ad}. While the exact implementation may vary depending on the type of the adaptive controller used, the adaptive signal will include the term

$$u_{ad}^i = \ldots W_{ad}^T(i) \cdot \beta_{ad}(i) \ldots \tag{16}$$

where $\beta_{ad}(i) \in R^m$ is a vector of system parameters or basis functions used to compute the adaptive signal, and $W_{ad}(i) \in R^m$ is a set of weights assigned to those parameters. The Projection Operator is designed to adjust $W_{ad}(i)$ (for simplicity subsequently referred to as W_{ad}).

III. Certification Challenges for Adaptive Controllers

The authority responsible for certifying flight control software in the United States is the Federal Aviation Administration (FAA). The FAA has stated that all flight-critical software must be developed according to guidance provided in RTCA DO-178B (Ref.9), or show compliance to airworthiness standards using alternative means. The statements in DO-178B are generally regarded as certification requirements, rather than guidelines. The DO-178B does not provide the metrics to assess the adequacy of the verification and validation plans (Ref. 8). This function is given to the Designated Engineering Representative (DER), who negotiates the certification process between the FAA and the software vendor.

In Ref. 8, Jacklin has identified a number of certification guidelines which are more difficult to achieve for adaptive controllers than for traditional controllers. These guidelines are:

- Define the software performance requirements
- Provide a software verification plan
- Define software requirements and all derived requirements
- Provide software verification test cases and procedures
- Provide a plan for software aspects of certification
- Provide software lifecycle data

Jacklin then grouped the above certification guidelines into 5 major gap areas in the certification process of adaptive controllers. The following section will discuss these gaps and the potential of closing them through the use of the Projection Operator.

Controller requirements. A necessary certification step would be developing procedures and methodologies which completely and correctly specify the design requirements of adaptive flight controllers. DO-178B recommends that the requirements be written in a manner that allows them to be tested. Performance requirements for non-adaptive controllers are usually specified via the traditional metrics, such as gain and phase margins. With the emergence of aircraft simulation environments and models, the general movement has been toward using simulation to describe and verify the behavior of adaptive controllers. While such initiatives are important, the introduction of the Projection Operator would allow traditional metrics (gain and phase margin) to be used to evaluate the performance of adaptive controllers by defining the boundary conditions for the adaptive signal's contribution. An example of such a derivation is provided in Section 5.

Controller Verification and Simulation. While the traditional methods of controller certification rely on rigorous mathematical analysis, DO-178B also allows certification credit to be obtained for high-fidelity simulation testing. It is generally believed that, due to their complexity, adaptive controller behavior can only effectively be evaluated in simulation. However, simulation techniques used for verifying controllers' behavior are more challenging to apply to the adaptive controllers. Since adaptive controllers are generally more complex, and have a larger range of parameter variation, using Monte Carlo analysis to simulate a variety of possible flight conditions can be very time consuming. By using Projection Operator, some of the traditional mathematical methods used for verifying linear controllers can be used for adaptive controllers as well. In addition, by limiting the maximum size of adaptive weights, Projection Operator would help reduce the total number of Monte Carlo simulations performed.

Proving Learning Stability and Convergence. Another gap in adaptive controller verification is the lack of methods that can validate that the learning algorithm is functioning properly, and converges to the correct solution in acceptable time. Learning is the main component that separates adaptive controllers from traditional ones, so proving that the learning process is stable and convergent for all flight conditions is imperative. Mathematical proofs of adaptive controllers' stability typically rely on Lyapunov's second method. One of the weaknesses of Lyapunov's method, however, is the fact that it does not provide a measure of how far away the system is from instability (the type of information derived from gain and phase margins). Thus, it is difficult to assess the robustness of the system in the presence of large disturbances (such as aircraft damage). While the stability proof for the system augmented with Projection Operator also relies on Lyapunov's techniques (as described in Section 4 of this paper), some traditional controller performance metrics can also be applied to the Projection Operator-enhanced adaptive controllers.

On-line Controller Monitoring Tools. While on-line monitoring tools are not a necessary part of controller certification, it is believed that their inclusion may make the certification process easier. If the analytical examination or simulation testing of a controller is not deemed to be sufficient for the certification purposes, on-line monitoring tools could be used to evaluate controller performance and stability during flight. The method of applying Projection Operator relies on the calculation of certain parameters of the adaptive controller (the complete process is described in Section 4). These calculated parameters can potentially be used for the purposes of monitoring real-time controller performance.

Controller Certification Plans. The DO-178B document suggests that the controller verification and validation (V&V) plans are to be developed before any code is written. V&V plans should provide a test matrix, as well as a description of test conditions and instructions, and the fail/pass criteria for each test. Even for non-adaptive controllers the development process often lacks formal software verification component. There is an on-going effort within the research community to define methodologies and test procedures for adaptive controllers. By looking at the Projection Operator as a certification-friendly adaptive controller augmentation, this paper is a part of that research effort.

IV. Projection Operator

Projection Operator and its application to adaptive control are described in detail in (Ref. 12 and 6). Suppose $f(x): R^n \rightarrow R$ is a continuously differentiable convex function. The gradient vector of f evaluated at θ is defined as

$$\nabla f(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1} \cdots \frac{\partial f(\theta)}{\partial \theta_n}\right)^T \in R^n \tag{17}$$

For a constant δ > 0, consider subset $\Omega_\delta = \{\theta \in R^n \mid f(\theta) \le \delta\} \subset R^n$. Let $\theta^* \in \Omega_\delta$ and assume that $f(\theta^*) < \delta$ (θ^* is not on the boundary of the subset). Also let $\theta \in \Omega_\delta$ and assume that $f(\theta) = \delta$ (θ is on the boundary of the subset). Then the following inequality takes place:

$$(\theta^* - \theta)^T \nabla f(\theta) \le 0 \tag{18}$$

In other words, the gradient vector evaluated at the boundary of a convex set always points away from the set.

Projection Operator is then defined as

$$\text{Proj}(\theta,y) = y - \frac{\nabla f(\theta)(\nabla f(\theta))^T}{\|\nabla f(\theta)\|^2} y f(\theta), \text{ if } f(\theta) > 0 \text{ and } y^T \nabla f(\theta) > 0 \tag{19}$$

$$\text{Proj}(\theta,y) = y, \text{ if not}$$

Projection Operator does not modify vector *y* as long as *y* belongs to the specified convex set. At the same time, Projection Operator provides a smooth transformation for any vector *y* that approaches the boundary of the set to vector Proj(θ,y), which is tangent to the boundary of the set.

We consider a special case of the Projection Operator, where variable *y* is the derivative of *θ*. Projection Operator then becomes

$$\text{Proj}(\theta,\dot{\theta}) = \dot{\theta} - \frac{\nabla f(\theta)(\nabla f(\theta))^T}{\|\nabla f(\theta)\|^2} \dot{\theta} f(\theta), \text{ if } f(\theta) > 0 \text{ and } \dot{\theta}^T \nabla f(\theta) > 0 \tag{20}$$

$$\text{Proj}(\theta,\dot{\theta}) = \dot{\theta}, \text{ if not}$$

The only requirement for function *f(θ)* is that it is continuously differentiable and convex. Function *f(θ)* is then defined as:

$$f(\theta) = \frac{\|\theta\|^2 - \theta_{\max}^2}{\varepsilon_\theta \theta_{\max}^2} \tag{21}$$

where ε_θ is projection tolerance which defines Projection Operator's region of operation within the subset Ω ($0 < \varepsilon_\theta \le 1$).

The variable *θ* to which we apply Projection Operator is the vector of adaptive weights W_{ad}, described in Section 2. Variable $\theta_{\max}$ represents the maximum allowed adaptive weight value, thus defining a hyper-sphere within which the weight vector must remain. Whenever commanded derivative $\dot{\theta}$ attempts to push the weights out of the hyper-sphere, Projection Operator smoothly adjusts the derivative to bring the weights back to the specified bounds.

Figure 2 illustrates the Projection Operator mechanism.

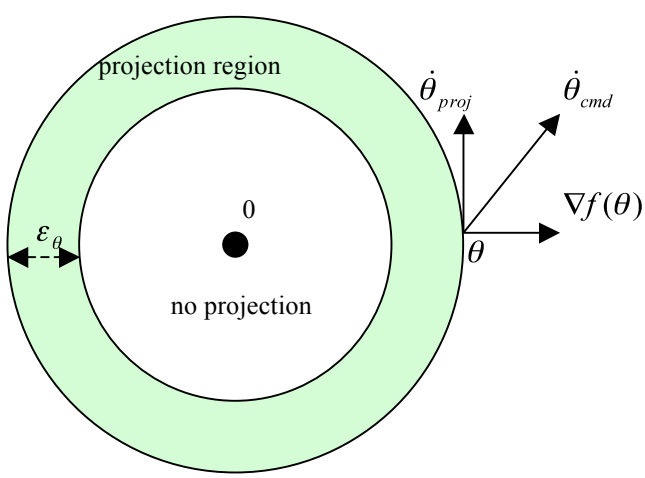

Figure 2.

It can be seen that Projection Operator limits the values of the adaptive weights. It can also be shown that the result of applying Projection Operator to a Lyapunov-stable system is also Lyapunov-stable (proof available in Ref. 12). In Ref. 7, Lavretsky et al has applied Projection Operator to constrain the weights of an adaptive predictor. Such application of Projection Operator guarantees that the prediction error dynamics are ultimately bounded with respect to the estimated error (proof available in Ref. 7). Since the equation for computing weight derivatives used in Ref. 7 is identical to Eq. 15, we assert that Projection Operator application to the adaptive controller weights will yield a similar result.

V. Simple Projection Operator Application

We consider a simplified adaptive system to illustrate the effect of limiting adaptive weights on system's stability margins. The simplified systems is shown in Figure 3.

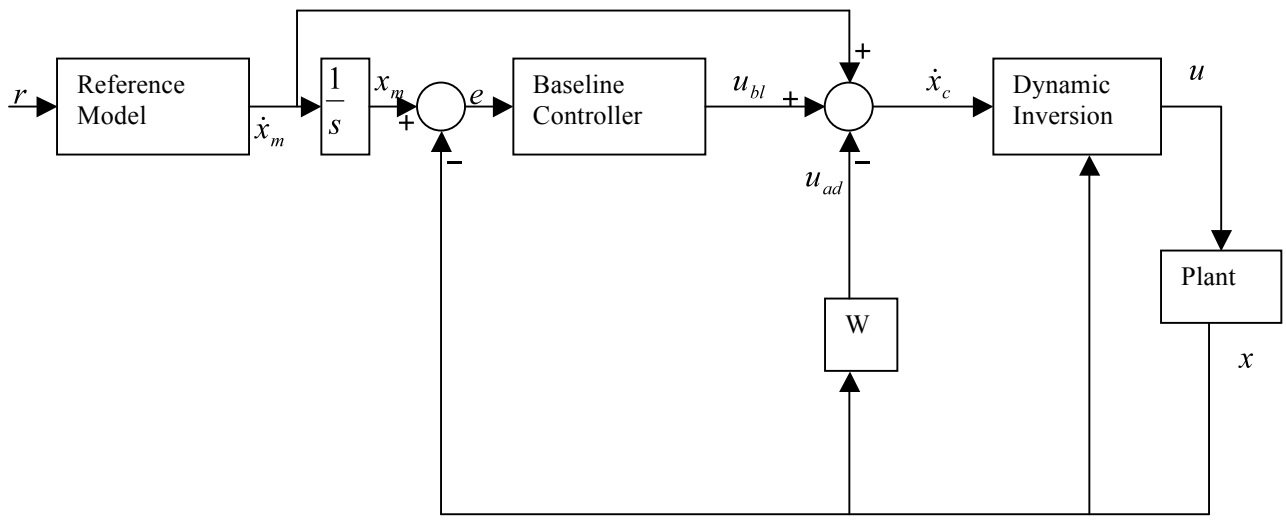

Figure 3.

The reference model of the system is

$$\dot{x}_m = A_r x_m + B_r r \tag{22}$$

The baseline controller is a PI controller, implemented as

$$u_{bl} = K_p e + K_i \int e \text{, where } e = x_m - x \tag{23}$$

The Dynamic Inversion block computes the desired control signal u

$$u = B_m^{-1}(\dot{x}_c - A_m x) \tag{24}$$

while the actual system is represented as

$$\dot{x} = Ax + Bu \tag{25}$$

From these equations we can derive the transfer function from *r(s)* to *x(s)* (full derivation is available in Appendix A). Over the course of the derivation, it is assumed that the system states are the input parameters to generate the adaptive control signal, such that

$$u_{ad} = Wx \tag{26}$$

Since our goal is to determine the behavior of the system for various values of W, we treat W as a constant during the transfer function analysis.

The derived transfer function is

$$\frac{x(s)}{r(s)} = \frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{s^3 + s^2(BB_m^{-1}(K_p + A_m + W) - A - A_R) + s(BB_m^{-1}K_i - A_R(BB_m^{-1}(K_p + A_m + W) - A) - A_R BB_m^{-1}K_i} \tag{27}$$

It can be seen that W affects the location of the transfer function poles. We can use the root locus method to compute the effect of varying W on the response of the system.

When working with a real system, we would attempt to derive the actual values of the system matrices, and then use these values during the Dynamic Inversion computation. Thus, we assume that A_m=A, and B_m=B. Eq. (27) then simplifies to

$$\frac{x(s)}{r(s)} = \frac{s^2 B_R + sK_p B_R + K_i B_R}{s^3 + s^2(K_p + W - A_R) + s(K_i - A_R(K_p + W)) - A_R K_i} \tag{28}$$

As an example, we consider a simple SISO system of order 1, with the following parameters:

$$A_R = -0.5, B_R = 0.5, K_p = 1.5, K_i = 6$$

The transfer function in Eq. (26) then becomes

$$\frac{x(s)}{r(s)} = \frac{2s^2 + 3s + 12}{4s^3 + 8s^2 + 4s^2 W + 27s + 2sW + 12} \tag{29}$$

To determine the effect of varying W on the system, we construct a root-locus plot with respect to W (shown in Fig. 4).

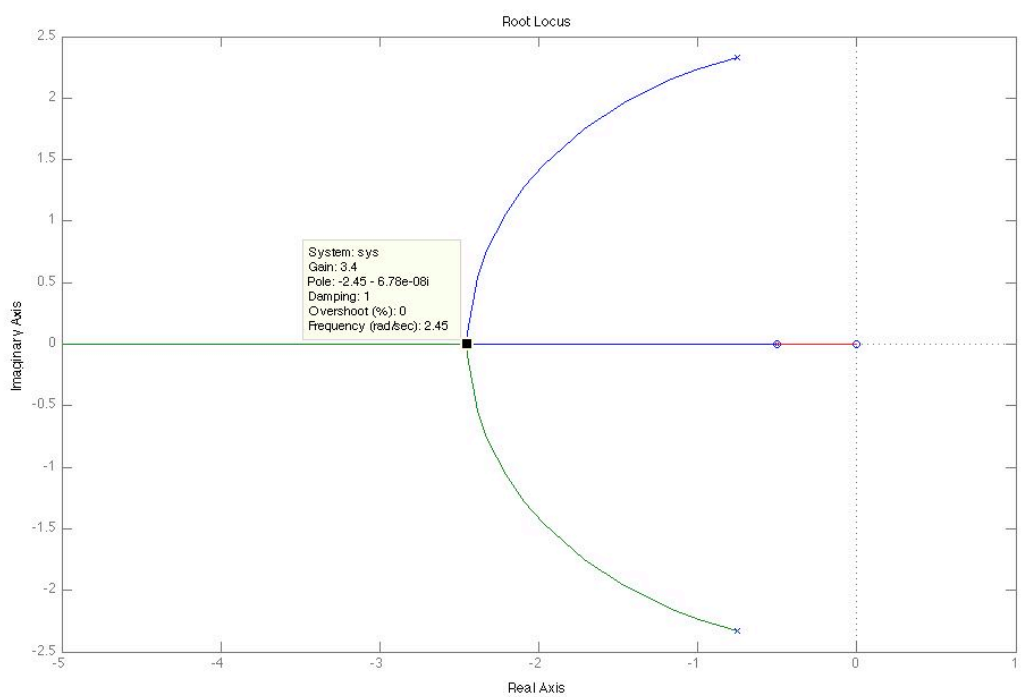

Figure 4. Root locus plot of the system in Eg. 27 with respect to W.

The system has a pair of complex closed-loop poles, and a pole-zero pair at -0.5. The complex poles converge to the real axis at the gain value of approximately 3.4. At higher gains, the system response is primarily determined by the single pole approaching 0. Figure 5 shows the frequency response of the system with an overall system gain of 10 for the values of W equal to 0, 1, 10, and 100. The system has a larger phase margin for W=1 than for W=0 (which is expected based on the root locus diagram). The phase margin decreases for larger values of W. As W continues to increase, the bandwidth of the system decreases as well.

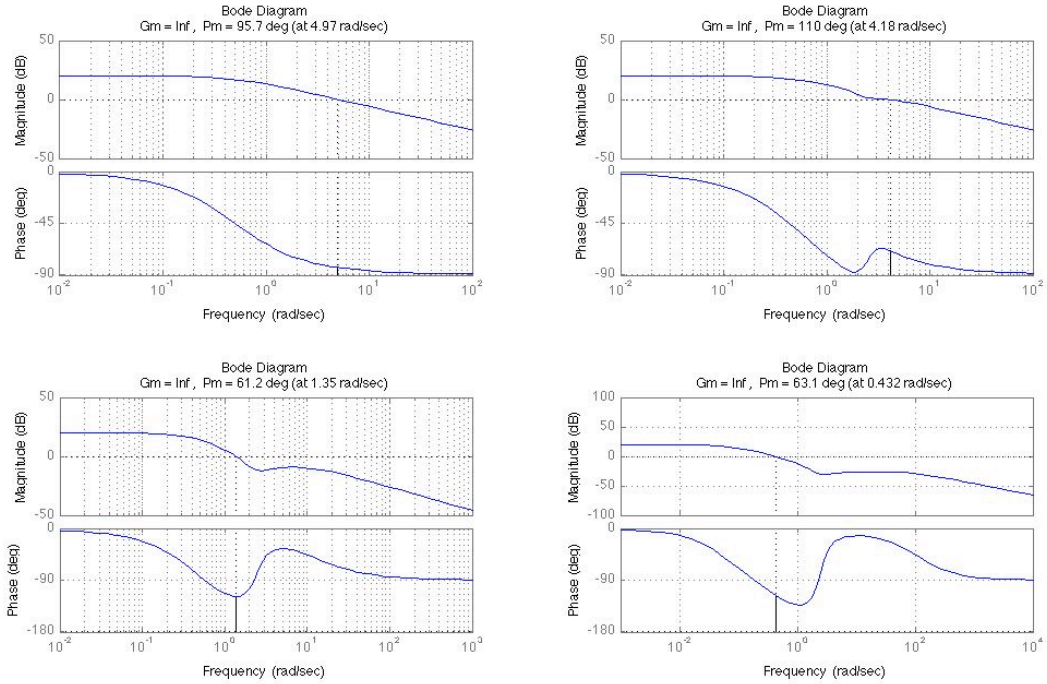

Figure 5. Frequency response for W=0 (upper left), W=1 (upper right), W=10 (lower left), and W=100 (lower right).

The goal of this exercise was to show that when the adaptive weights are limited to a maximum possible value (by a technique such as Projection Operator), the overall system can be analyzed through methods normally reserved for traditional controllers, namely root locus and phase and gain margins. An adaptive system that can meet the requirements imposed by the traditional certification techniques would be more likely to successfully navigate the certification process.

VI. Flight Simulation Experiment

The test vehicle used for simulation is a twin-engine transport-class aircraft model developed by NASA Langley Research Center (Ref. 4). The model allows for introduction of physical aircraft damage, as well as controller failure scenarios. Physical damage simulation is accomplished via the use of the vortex-lattice code developed at NASA Ames Research Center (Ref. 5) to estimate the aerodynamic coefficients, and the stability and control derivatives of the damaged model.

A number of Model-Reference-based adaptive controllers (along with the baseline dynamic inversion controller) were tested in a simulated flight scenario. The simulated pilot input was comprised of 4 LLD (Longitude-Latitude-Directional) doublets. Each doublet was 4 seconds long, and there was a wait of 5 seconds between the doublets. The goal of the controller was to have the aircraft track the reference input as closely as possible. All of the aircraft parameters, except for the adaptive controller gains, were reset prior to the start of each doublet. This was done to ensure that the controller performance improvement or decrement was the result of controller adaptation (as opposed to a change in flight condition).

The controllers were evaluated on their ability to reduce the tracking error between the reference input and the aircraft state. The so-called M5 metric was used, which represented the L_2 norm of the tracking error:

$$M_5 = \sqrt{\int_{t_0}^{t_f} \left[x(t) - x_m(t)\right]^2 d\tau} \tag{30}$$

The M5 metrics is described in further detail in Ref. 13. The metric was computed individually for each aircraft axis, and, for some tests, the three resulting metric values were combined into one.

For some tests, controller failures were introduced into the simulation. The controller failure consisted of scaling the B-matrix of the dynamic inversion block by a certain factor (between 0.125 and 8). Such a failure represents the condition when the dynamic inversion controller does not have the accurate information about the vehicle model, resulting in an improper input-to-state mapping. Depending on the B-matrix gain value, such controller would produce a response that is either underdamped or overdamped.

Figure 6 shows the simulation results for various adaptive controllers (plotted in different colors), normalized against the M5 value for the baseline dynamic inversion controller (dashed line). The x-axis represents the maximum limit imposed by Projection Operator on the adaptive weights. This particular simulation was performed with the B-matrix gain of 4. For the low projection limits, the adaptive controllers' performance converges to that of the dynamic inversion controller. Performance generally improves as the projection limit is increased, and stabilizes for the projection limit values above 4, thereby indicating the maximum natural adaptive weight limit for this simulation.

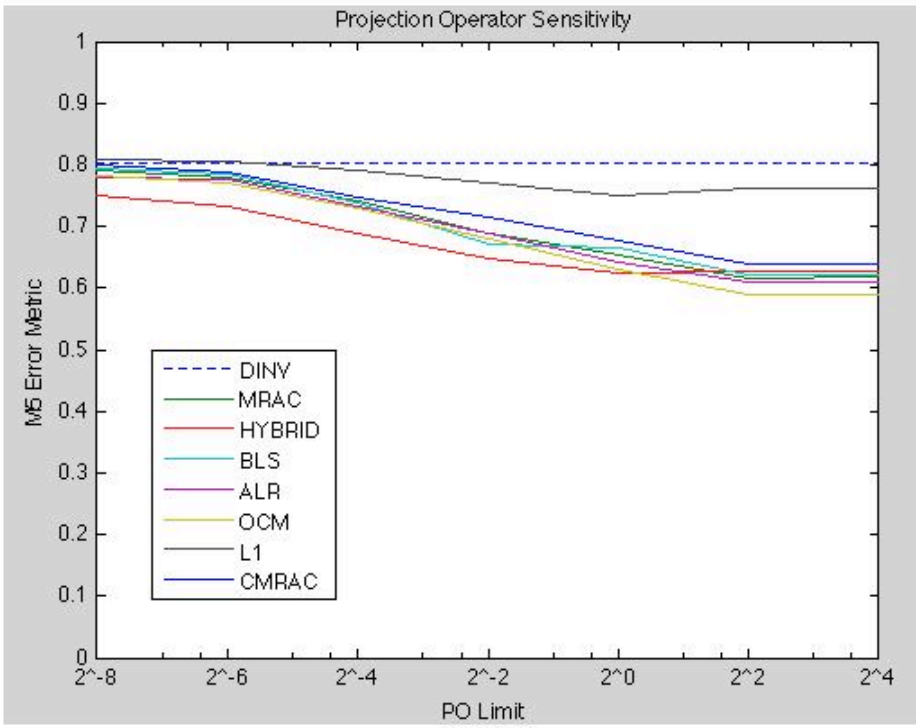

Figure 6.

Figure 7 illustrates a similar trend. In Fig. 7, different colors represent various maximum projection limits, as applied to a standard MRAC (the exception is the blue line which represents the dynamic inversion controller). The controllers are evaluated for different B-matrix gains (plotted along the x-axis). Only the plots for the roll axis are shown (the plots for pitch and yaw axes have a similar overall shape, although the differences between individual plots are not as pronounced). Here the dynamic inversion controller generally has the worst performance, while MRAC's performance with low projection limits is very similar to that of the dynamic inversion controller. As the projection limits are relaxed, the MRAC's performance generally improves.

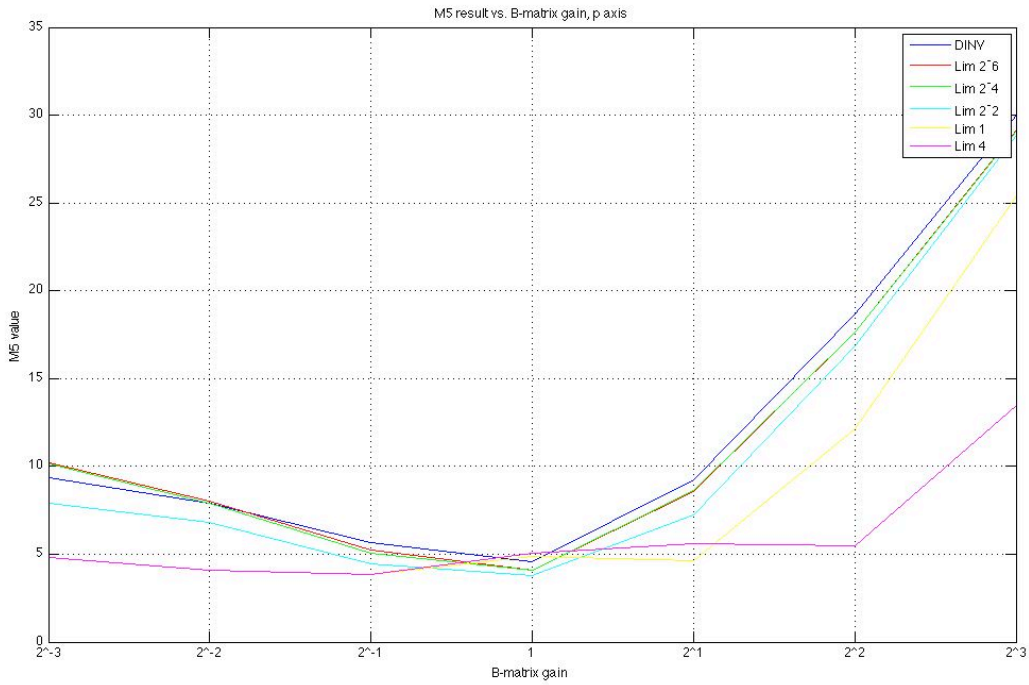

Figure 7

When projection limits are set to a low value, the performance of the adaptive controller approaches that of a linear dynamic inversion controller. Such result is expected: if the adaptive weights are set to zero, the adaptive augmentation term is zero as well, so the controller relies fully on dynamic inversion. Thus, such curtailed version of the adaptive controller should theoretically be certified just as easily as a linear controller. Moreover, it would be expected that the certification process for a linear controller would allow for a certain level of tolerance in controller's performance. An adaptive controller could be designed to stay within the specified tolerance limits by using Projection Operator to limit the adaptive weights. Projection Operator thus represents a progressive compromise between the improved performance of the adaptive controller, and the predictability of the linear controller.

Figure 8 represents the time-domain performance comparison of the dynamic inversion controller, MRAC, and MRAC with Projection Operator enabled, and projection limit set to 0.5. Here the aircraft's roll rate (blue line) attempts to track the roll rate commanded by the reference model (red line). Although the differences between controllers are small, in the dynamic inversion case the roll response remains the same from one doublet to another, while with regular MRAC the roll rate response changes as the controller adapts. In case of the MRAC with Projection Operator enabled, the roll rate response changes initially, but then the adaptation rate appears to slow down. This can be explained by adaptive weights reaching their preset limit. Again, it can be seen that Projection Operator-enabled MRAC represents a compromise between the regular MRAC and the dynamic inversion controller.

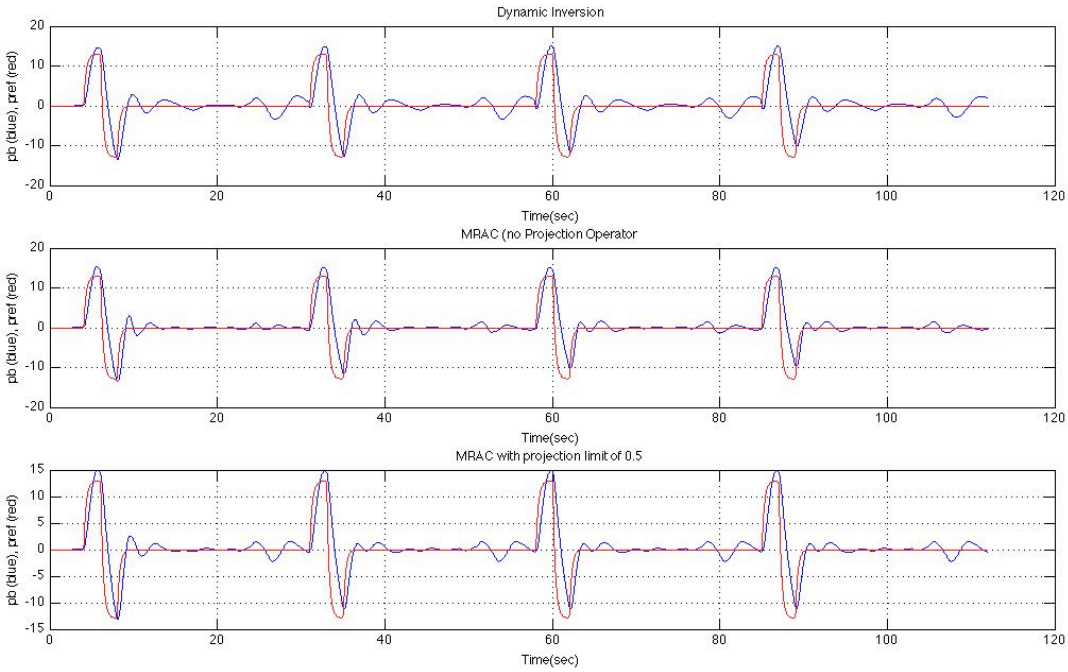

Figure 8.

VII. Conclusion and Future Work

Projection Operator is a technique that we use to limit adaptive weights of a controller. The control authority of an adaptive controller can thus be bound arbitrarily close to that of a linear controller. While certain certification requirements may be more difficult to achieve for adaptive controllers than they are for traditional controllers, the application of Projection Operator can help close the gap. A Projection Operator-enhanced MRAC can be evaluated by the metrics normally reserved for traditional controllers (such as gain and phase margins). Future Projection Operator research may explore the effects of modifying the algorithm to allow different limits along separate directions of the adaptive weight vector (thereby transforming the hyper-sphere of weight limits into a hyper-ellipsoid). As far as the adaptive controller certification goes, many challenges remain. For instance, it is expected that the process of applying traditional stability metrics (i.e. root locus and phase and gain margins) to a multi-order adaptive system will be more complicated than the simple example presented in Section 5. However, the exact nature of the certification challenges will be much better understood once the certification process is attempted.

Appendix A

This section derives the transfer function for the system shown in Fig. 3. Equations (22) through (25) of the main paper can be represented in the frequency domain as the following:

$$x_m s = A_R x_m + B_R r \tag{1}$$

$$u_{bl} = e(K_p + \frac{1}{s} K_i) \tag{2}$$

$$u = B_m^{-1}(x_c s - A_m x) \tag{3}$$

$$xs = Ax + Bu \tag{4}$$

Combining these equations, we can proceed to derive the transfer function from *r(s)* to *x(s)*:

$$x_m(s - A_R) = B_R r$$

$$x_m = \frac{B_R r}{s - A_R}$$

$$u_{bl} = (K_p + \frac{1}{s}K_i)(\frac{B_R r}{s - A_R} - x)$$

$$sx_c = (K_p + \frac{1}{s}K_i)(\frac{B_R r}{s - A_R} - x) + \frac{sB_R r}{s - A_R} - u_{ad}$$

$$u = B_m^{-1}((K_p + \frac{1}{s}K_i)(\frac{B_R r}{s - A_R} - x) + \frac{sB_R r}{s - A_R} - u_{ad} - A_m x)$$

$$sx = Ax + BB_m^{-1}((K_p + \frac{1}{s}K_i)(\frac{B_R r}{s - A_R} - x) + \frac{sB_R r}{s - A_R} - u_{ad} - A_m x) =$$

$$= Ax + BB_m^{-1}(\frac{K_p B_R r}{s - A_R} + \frac{K_i B_R r}{s(s - A_R)} - K_p x - \frac{K_i x}{s} + \frac{sB_R r}{s - A_R} - u_{ad} - A_m x) =$$

$$= r(\frac{BB_m^{-1}K_p B_R}{s - A_R} + \frac{BB_m^{-1}K_i B_R}{s(s - A_R)} + \frac{sB_R}{s - A_R}) + x(A - BB_m^{-1}K_p - \frac{BB_m^{-1}K_i}{s} - BB_m^{-1}A_m) - BB_m^{-1}u_{ad}$$

$$x(\frac{s^2 + s(BB_m^{-1}(K_p + A_m) - A) + BB_m^{-1}K_i}{s}) = r(\frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{s(s - A_R)}) - BB_m^{-1}u_{ad}$$

From the last equation, we can derive the transfer function from *r(s)* and $u_{ad}(s)$ to *x(s)*. We consider the case where the system states are used as input parameters to generate the adaptive control signal, such that

$$u_{ad} = Wx$$

Then the transfer function equation becomes

$$x(s - A + BB_m^{-1}(K_p + A_m) + BB_m^{-1}W + \frac{BB_m^{-1}K_i}{s}) = r(\frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{s(s - A_R)})$$

$$x(\frac{s^2 + s(BB_m^{-1}(K_p + A_m + W) - A) + BB_m^{-1}K_i}{s}) = r(\frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{s(s - A_R)})$$

$$\frac{x}{r} = \frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{(s - A_R)(s^2 + s(BB_m^{-1}(K_p + A_m + W) - A) + BB_m^{-1}K_i)} =$$

$$= \frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{s^3 + s^2(BB_m^{-1}(K_p + A_m + W) - A) + sBB_m^{-1}K_i - A_R s^2 - sA_R(BB_m^{-1}(K_p + A_m + W) - A) - A_R BB_m^{-1}K_i}$$

$$\frac{x(s)}{r(s)} = \frac{s^2 B_R + sBB_m^{-1}K_p B_R + BB_m^{-1}K_i B_R}{s^3 + s^2(BB_m^{-1}(K_p + A_m + W) - A - A_R) + s(BB_m^{-1}K_i - A_R(BB_m^{-1}(K_p + A_m + W) - A) - A_R BB_m^{-1}K_i} \quad (5)$$

References

[1]Nguyen,N.T., Krishnakumar,K., and Boscovic,J., "An Optimal Control Modification to Model-Reference Adaptive Control for Fast Adaptation", AIAA Guidance, Navigation and Control Conference and Exhibit, 18 - 21 August 2008, Honolulu, HI.

[2]Kaneshige, J., Bull, J., and Totah, J., "Generic Neural Flight Control and Autopilot System", AIAA Guidance, Navigation, and Control Conference, AIAA-2000-4281, 2000.

[3]Nguyen, N., Krishnakumar, K., Kaneshige, J., and Nespeca, P., "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]Bailey, R., Hostetler, R., Barnes, K., Belcastro, C., and Belcastro,C., "Experimental validation: subscale aircraft ground facilities and integrated test capability", Proceedings of the AIAA Guidance Navigation, and Control Conference and Exhibit 2005, San Francisco, California, 2005.

[5]Totah, J. J., Kinney, D.J., Kaneshige, J.T., and Agabon,S., "An Integrated Vehicle Modeling Environment", AIAA 99-4106, August 1999.

[6]Lavretsky,E., "Lecture Notes for Control and Dynamical Systems (CDS 270), Lecture 6, Spring 2007", California Institute of Technology, Pasadena, CA, 2007.

[7]Lavretsky,E., and Hovakimyan,N., "Adaptive compensation of control dependent modeling uncertainties using time-scale separation", in Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference, (Seville, Spain), pp. 2230 – 2235, December 2005.

[8]Jacklin,S.A., "Closing the Certification Gaps in Adaptive Flight Control Software", AIAA Guidance, Navigation and Control Conference and Exhibit, 18 - 21 August 2008, Honolulu, HI.

[9]Software Considerations in Airborne Systems and Equipment Certification, Document No RTCA (Requirements and Technical Concepts for Aviation)/DO-178B, December 1, 1992

[10]Calise, A.J., Lee, S., and Sharma, M., "Development of a Reconfigurable Flight Control Law for Tailless Aircraft", AIAA Journal of Guidance, Control, and Dynamics, Vol. 24, No. 5, 2001, pp. 896-902.

[11]Sharma, M., Lavretsky, E., and Wise, K.A., "Application and Flight Testing of an Adaptive Autopilot on Precision Guided Munitions", AIAA Guidance, Navigation and Control Conference and Exhibit, 21 - 24 August 2006, Keystone, CO.

[12]Ioannou, P.A, Sun, J, "Robust Adaptive Control", Prentice Hall, Englewood Cliffs, NJ, 1995.

[13]Stepanyan, V., Krishnakumar, K., Nguyen, N, and Van Eykeren, L., "Stability and Performance Metrics for Adaptive Flight Control", AIAA Guidance, Navigation and Control Conference, Chicago, Il, 2009