JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 28, No. 6, November–December 2005

Backstepping-Based Flight Control with Adaptive Function Approximation

Jay Farrell*
University of California, Riverside, Riverside, California 92521
Manu Sharma†
Barron Associates, Inc., Charlottesville, Virginia 22901
and
Marios Polycarpou‡
University of Cyprus, Nicosia 1678, Cyprus

A command filtered backstepping approach is presented that uses adaptive function approximation to control unmanned air vehicles. The controller is designed using three feedback loops. The command inputs to the airspeed and flight-path angle controller are χ_c, γ_c, V_c and the bounded first derivatives of these signals. That loop generates comand inputs μ_c, α_c for a wind-axis angle loop. The sideslip angle command β_c is always zero. The wind-axis angle loop generates rate commands P_c, Q_c, R_c for an inner loop that generates surface position commands. The control approach includes adaptive approximation of the aerodynamic force and moment coefficient functions. The approach maintains the stability (in the sense of Lyapunov) of the adaptive function approximation process in the presence of magnitude, rate, and bandwidth limitations on the intermediate states and the surfaces.

I. Introduction

A VARIETY of feedback control approaches have been developed to deal with nonlinear systems.[1–3] Various authors have investigated the applicability of these nonlinear control methodologies to advanced flight vehicles. For flight control, these methods offer both increases in performance as well as reduction of development times by dealing with the complete dynamics of the vehicle rather than local operating point designs. Feedback linearization, in its various forms, is perhaps the most commonly employed nonlinear control method in flight control.[4–9] Backstepping-based approaches are discussed for example in Refs. 10–14. Reference 15 presents a nonlinear model predictive control approach that relies on a Taylor-series approximation to the system's differential equations. Optimal control techniques are applied to control load factor in Ref. 16. Prelinearization theory and singular perturbation theory are applied for the derivation of inner- and outer-loop controllers in Ref. 7. The main drawback to the nonlinear control approaches just mentioned is that, as model-based control methods, they require accurate knowledge of the plant dynamics. This is of significance in flight control because aerodynamic parameters always contain some degree of uncertainty. Although some of these approaches are robust to small modeling errors, they are not intended to accommodate significant unanticipated errors that can occur in the event of failure or battle damage. In such an event, the aerodynamics can change rapidly and deviate significantly from the model used for control design. Uninhabited air vehicles (UAVs) are particularly susceptible to such events because there is no pilot onboard.

To address the issue of uncertainty, several "robustifying" techniques have been developed:

1) Parameter adaptive methods deal with parametric uncertainty,[17] in which the nonlinearities are assumed to be known, but some of the parameters that multiply these nonlinearities are unknown.

2) Robust methods deal with the case in which upper bounds on the unknown nonlinearities are known and the control is devised based on the bounds[18]; therefore, these methods tend to be conservative, sometimes leading to high-gain feedback.

3) Robust adaptive methods combine parametric uncertainty and unknown nonlinearities with partially known bounds.[19]

In many applications, such as the control of high-performance aircraft and UAVs, improved control can be achievable if the unknown nonlinearities are approximated online (i.e., during operation). Examples of such online approximators include sigmoidal neural networks, splines, racial basis functions, and wavelets. The application of online approximation methods to nonlinear systems in a feedback framework yields a complex nonlinear closed-loop system, which is analyzed using Lyapunov stability methods. Typically, the feedback control law and the adaptive law for updating the approximator parameters are derived by using a Lyapunov function, whose time derivative is forced to have some desirable stability properties (for example, negative definiteness). Therefore, the stability of the closed-loop system is obtained during the synthesis of the adaptive control laws. Examples of this type of approach, which is referred to as a Lyapunov synthesis method, include Refs. 10, 14, and 20–31.

From a practical perspective, one of the major issues in feedback control system design is that the signal $u(t)$ generated by the control law might not be implementable because of physical constraints. A common example of such a constraint is actuator saturation, which imposes limitations on the magnitude of the achievable control input. In some applications this problem is crucial, especially in combination with nonlinear online approximation-based control, which tends to be aggressive in seeking the desired tracking performance. In aircraft control applications, input saturation is caused by limitations on control surface deflections. For UAVs, the absence of humans onboard can allow more aggressive maneuvering; however, the feedback control law has to deal both with unknown nonlinearities and input saturation. Another practical issue of significant importance in many applications, especially in backstepping control where states are used as intermediate control variables, is physical constraints on state variables. Such constraints can include magnitude, rate, and bandwidth limitations of the state variables.

Received 25 August 2004; revision received 25 January 2005; accepted for publication 5 March 2005.

*Professor, Department of Electrical Engineering.
†Research Scientist, 1410 Sachem Place, Suite 202.
‡Professor, Department of Electrical and Computer Engineering; also Professor, Department of Electrical and Computer Engineering and Computer Science, University of Cincinnati, Cincinnati, OH 45221-0030.

Control signal rate and amplitude constraints in an adaptive linear control framework are addressed in, for example, Refs. 32–38. One possible approach is to completely stop adaptation during saturation of the control input. Although this ad hoc method does prevent the tracking error induced by actuator constraints from corrupting parameter estimation, the stability properties of the closed-loop system cannot be established. An alternate approach, which we refer to as training signal hedging (TSH), for example, see Refs. 32 and 36, modifies the tracking error definition used in the parameter update laws. Finally, a third approach, referred to as pseudocontrol hedging (PCH), alters the commanded input to the loop.[35,37] The idea behind the PCH approach is to attenuate the command to the loop so that the generated control signal is implementable without exceeding the actuator constraints. Constraints on the intermediate control signals (i.e., state variable and actuator commands) in approximation-based backstepping are addressed by the command filtering approach described for nonlinear scalar systems in Refs. 33, 39, and 40 and for nonlinear block vector systems in Ref. 34. In addition to addressing constraints on the intermediate and actuator control signals, in relation to standard backstepping, the command filtering approach is applicable to a wider class of systems. Finally, it eliminates complications related to computing the analytic derivatives of the intermediate control signals.

This paper presents an online approximation-based backstepping control approach for advanced flight vehicles. The approach uses the block vector command filtered approach described in Ref. 34. The control law is designed using three feedback loops. The inputs to the airspeed and flight-path angle controller are commanded heading χ_c, climb rate γ_c, speed V_c, and the bounded first derivatives of these signals. That flight-path angle controller generates roll angle and angle-of-attack command inputs (μ_c, α_c) for a wind-axis angle loop. The sideslip angle command β_c is always zero. The wind-axis angle loop generates angular rate commands P_c, Q_c, R_c for an inner-loop controller that uses actuator distribution to generate surface position commands. The control approach includes online approximation of the aerodynamic force and moment coefficient functions. The approach maintains the stability (in the sense of Lyapunov) of the online function approximation process in the presence of magnitude, rate, and bandwidth limitations on the intermediate states and the surfaces.

The approach herein develops online approximations to the aircraft force and moment functions. Similar approaches using off-line approximations have a long history in the aerodynamics community. Trankle et al. review off-line system identification methods in Ref. 41. The example of this paper will use splines in the system identification process. An overview, with several examples of off-line, spline-based, system identification based on data partitioning, is presented in Ref. 42. An excellent review of off-line estimation of aircraft model parameters from flight data is contained in Ref. 43. Finally, Ref. 44 performs off-line estimation of additive functional corrections to the aircraft model. The corrections are multidimensional cubic splines (see p. 1294 of Ref. 44), which is interesting relative to the approach herein in which such model error is approximated online.

The main advantages of the approach presented herein include 1) the aerodynamic force and moment models are automatically adjusted to accommodate changes to the aerodynamic properties of the vehicle; 2) Lyapunov stability results are provable; and 3) state and control constraints can be enforced while maintaining Lyapunov stability. The main motivations for this work were to produce a simplified control design that is also more robust to model error, to accommodate large changes in the vehicle dynamics (e.g., damage) online, and to learn the aerodynamic coefficient functions for the vehicle. An anticipated benefit from these properties is that the controller could be applied to an aircraft for which it was not explicitly designed, for example, an aircraft of the same family but different configuration. Additionally, the controller could be developed using a lower-fidelity model than required by current methods, thereby offering a cost savings. This control method is expected to provide significant reduction in design time because the control system design does not depend on a conglomeration of point designs. Reference 45 discusses some of these benefits in the context of adaptive control for guided munitions.

The derivation of the UAV controller in Secs. III–V is based on a theorem presented in Ref. 34. For completeness and ease of reference, that theorem is reviewed in Sec. II. The resulting controller and its stability properties are summarized in Sec. VI. A specific approximator structure that will be used in the subsequent simulation example is defined in Sec. VII. The simulation example is in Sec. VIII. Tables C1–C3 in Appendix C define the notation used in the body of this paper.

II. Review of an Existing Stability Result

Sections III–V will use a stability result that was first presented in Ref. 34 to derive a provably stable online approximation-based backstepping controller suitable for an UAV. Because those stability results will be used iteratively, it is reviewed first in this section.

A. Problem Formulation

Let $x_1 \in \Re^{n_1}$, $x_2 \in \Re^{n_2}$, and $u \in \Re^m$, with $m \geq n_2$. Consider the system

$$\dot{x}_1 = A_1(x) f_1(x) + B_1(x) G_1(x_1) x_2 \tag{1}$$

$$\dot{x}_2 = A_2(x) f_2(x) + B_2(x) G_2(x) u \tag{2}$$

where $x = [x_1^\top, x_2^\top]^\top$, and for $i = 1, 2$, $A_i(x)$ and $B_i(x)$ are known matrices, and $f_i(x)$ and $G_i(x)$ are unknown vector and matrix functions with row dimension denoted by p_i. Assume that $B_2(x)G_2(x)$ has full row rank for any $x \in \mathcal{D}$, where $\mathcal{D}$ is a compact domain of operation. For convenience of notation, we will assume that B_2 is square. This implies that $B_2(x)$ must be invertible on $\mathcal{D}$ for the full rank assumption to hold. Also, assume that the desired trajectory $x_{1c}(t)$ and its derivative $\dot{x}_{1c}(t)$ are known and that $x_{1c}(t) \in \mathcal{D}$ for all $t \geq 0$.

The system (1) and (2) allows consideration of systems where unknown functions affect several states. For example, as described in Secs. III–V, for aircraft control $A_i(x)$ and $B_i(x)$ represent rotation matrices, and f_1 and f_2 represent the vectors of aerodynamic forces and moments. The represented system also allows consideration of applications where a vector of state variables x_2 from an inner loop is used to control a vector of state variables x_1 from an outer loop.

Define the tracking error signals

$$\tilde{x}_i = x_i - x_{ic} \qquad \text{for} \qquad i = 1, 2 \tag{3}$$

where x_{2c} will be defined subsequently. Also, define approximations to the unknown functions:

$$\hat{f}_i(x) = \theta_{f_i}^T \Phi_{F_i}(x), \quad \theta_{f_i}^T \in \Re^{p_i \times N_{f_i}}, \quad \Phi_{f_i}(x) : \Re^n \mapsto \Re^{N_{f_i}}$$

$$\hat{G}_{ij}(x) = \theta_{G_{ij}}^T \Phi_{G_{ij}}(x), \quad \theta_{G_{ij}}^T \in \Re^{p_i \times N_{G_{ij}}}, \quad \Phi_{G_{ij}}(x) : \Re^n \mapsto \Re^{N_{G_{ij}}}$$

where $\hat{G}_{ij}(x)$ represents the jth column of $\hat{G}_i$. We assume that there exist unknown vectors $\theta_{f_i}^*$ and $\theta_{G_{ij}}^*$ such that $f_i(x) = (\theta_{f_i}^*)^T \Phi_{F_i}(x)$ and $G_{ij}(x) = (\theta_{G_{ij}}^*)^T \Phi_{G_{ij}}(x)$. Then the approximator parameter errors are

$$\tilde{\theta}_{f_i} = \theta_{f_i} - \theta_{f_i}^* \tag{4}$$

$$\tilde{\theta}_{G_{ij}} = \theta_{G_{ij}} - \theta_{G_{ij}}^* \tag{5}$$

The procedure and theorem that follow extend backstepping to allow stable online approximation of nonlinear functions while the system being controlled is subject to physical constraints on the intermediate states x_2 and the actuator signals u. Similar to standard backstepping, we will first consider tracking control for the simpler system

$$\dot{x}_1 = A_1(x) f_1(x) + B_1(x) G_1(x_1) \mu_1 \tag{6}$$

Then, we will consider the design of the signal u for the original system of (1) and (2), where x_2 implements μ_1 within a stable

closed-loop system. The following assumption is used to initialize the procedure.

Assumption 1: There exists an online approximation-based control law $\mu_1(x, x_{1c}, \dot{x}_{1c}, \theta_{f1}, \theta_{G_{1j}})$ with positive-definite gain matrix K_1 and parameter update laws

$$\dot{\theta}_{f1} = \alpha_1(x, \tilde{x}_1) \tag{7}$$

$$\dot{\theta}_{G_{ij}} = \beta_{1j}(x, \tilde{x}_1, \mu_1) \tag{8}$$

such that the Lyapunov function

$$\mathcal{V}_1 = \frac{1}{2}\left[\tilde{x}_1^T\tilde{x}_1 + \operatorname{trace}\left(\tilde{\theta}_{f_1}^T\Gamma_{f_1}^{-1}\tilde{\theta}_{f_1}\right) + \sum_{j=1}^{n_2}\operatorname{trace}\left(\tilde{\theta}_{G_{1j}}^T\Gamma_{G_{1j}}^{-1}\tilde{\theta}_{G_{1j}}\right)\right]$$

with Γ_{f_1} and $\Gamma_{G_{1j}}$ positive definite, when evaluated along the trajectories of the closed loop system of Eq. (6), has

$$\dot{\mathcal{V}}_1 = \frac{\partial\mathcal{V}_1}{\partial\tilde{x}_1}(A_1 f_1 + B_1 G_1\mu_1) + \frac{\partial\mathcal{V}_1}{\partial\tilde{\theta}_{f_1}}\alpha_1(x, \tilde{x}_1)$$

$$+ \sum_{j=1}^{n_2}\frac{\partial\mathcal{V}_1}{\partial\theta_{G_{1j}}}\beta_{1j}(x, \tilde{x}_1, \mu_1) \le -\tilde{x}_1^T K_1\tilde{x}_1$$

B. State and Actuator Constraint Compensation Procedure

The purpose of this section is to extend the backstepping approach in two directions. The first direction of extension uses command filters to eliminate the analytic computation of the time derivative of μ_1 that would be required for a standard backstepping implementation. Although the analytic differentiation is possible, it becomes tedious as the number of iterations of the backstepping approach increases and when parameter adaptation is involved. A command filter with outputs x_{2c} and $\dot{x}_{2c}$ will be designed as a linear, stable, low-pass filter with unity gain from its input to x_{2c} at low frequencies. In this command filtered approach, in the absence of physical limitations (i.e., magnitude, rate, and bandwidth constraints on the intermediate state x_2 and control u), we can prove convergence of the tracking errors in the sense of Lyapunov. The second direction of extension is to ensure stability of the online function approximation (i.e., parameter estimation) process even when physical limitations are in effect. Note that when such physical constraints are in effect, the tracking error can increase because the necessary control signal to achieve tracking cannot be implemented within the physical constraints imposed on the system. The requested trajectory is too aggressive for the vehicle. In this circumstance, our goal is to maintain the stability of the online function approximation process. In our approach, we use the command filter to ensure that the commanded state x_{2c} does not leave the safe operating envelope defined a priori for the vehicle.

Assumption 1 is directly applicable to the simplified system of Eq. (6): however, for the actual system of Eqs. (1) and (2), backstepping approaches use the intermediate state x_2 to implement the control signal μ_1. When the control signal μ_1 is implemented by x_2 and the state and actuators have physical limitations, such approaches might not be able to be successfully implemented in a direct fashion. To address magnitude, rate, and bandwidth constraints on the state and control, define the following procedure[34]:

1) Define

$$x_{2c}^o = \mu_1 - \xi_2 \tag{9}$$

where ξ_2 is defined subsequently. Then, filter x_{2c}^o to produce the magnitude, rate, and bandwidth-limited command signal x_{2c} and its derivative $\dot{x}_{2c}$ that are within the vehicle operating envelope $\mathcal{D}$ (see Appendix A). Then define

$$\dot{\xi}_1 = -K_1\xi_1 + B_1\hat{G}_1\left(x_{2c} - x_{2c}^o\right) \tag{10}$$

Note that this is a stable linear filter with input $\hat{G}_1(x_{2c} - x_{2c}^o)$. Theorem 1 provides the boundedness of $\hat{G}_1$, and the boundedness of $(x_{2c} - x_{2c}^o)$ is discussed in Appendix A; therefore, as the output of a stable linear system with a bounded input, the signal $\xi_1(t)$ is bounded.

2) Define the compensated tracking errors as

$$\bar{x}_i = \tilde{x}_i - \xi_i \qquad \text{for} \qquad i = 1, 2 \tag{11}$$

3) Define any continuous signal u_c^o such that

$$B_2\hat{G}_2u_c^o = -K_2\bar{x}_2 + \dot{x}_{2c} - A_2\hat{f}_2(x) - \hat{G}_1^T B_1^T\bar{x}_1 \tag{12}$$

In the case where G_2 has more columns than rows, more than one solution can exist.

4) Filter u_c^o to produce u that is within the magnitude, rate, and bandwidth limitations of the actuation system. Therefore, u is achievable by the actuators. Then define

$$\dot{\xi}_2 = -K_2\xi_2 + B_2\hat{G}_2\left(u - u_c^o\right) \tag{13}$$

By discussion similar to that following Eq. (10), the signal $\xi_2(t)$ is bounded.

5) Define the parameter update laws according to

$$\dot{\theta}_{f_1} = \alpha_1(x_1, \bar{x}_1), \qquad \dot{\theta}_{f_2} = \Gamma_{f_2}\Phi_{f_2}\bar{x}_2^T A_2 \tag{14}$$

$$\dot{\theta}_{G_{1j}} = \beta_1(x_1, \bar{x}_1, x_2), \qquad \dot{\theta}_{G_{2j}} = \Gamma_{G_{2j}}\Phi_{G_{2j}}\bar{x}_2^T u_j B_2 \tag{15}$$

Note that functions α_1, β_1, and μ_1 satisfying assumption 1 are inputs to this procedure. The function μ_1 is used in the computation of x_{2c}^o in Eq. (9). The arguments to the functions α_1 and β_1 are altered in Eqs. (14) and (15) to attain robustness of the parameter estimation process to actuator and state constraints.

The filters in Eqs. (10) and (13) estimate the effect on the tracking error due to implementing the achievable control signal instead of the desired control signal. The compensation of the tracking error in the computation of $\bar{x}_i$ for $i = 1, 2$ is similar to the method of TSH as suggested in Refs. 32 and 36 for actuator constraints. Additionally, the modification of the command μ_1 by ξ_2 to produce x_{2c}^o is similar to the method of PCH as suggested in Refs. 35 and 37 for actuator constraints. The preceding method extends the TSH and PCH methods to fit within a recursive backstepping procedure while enforcing state and actuator constraints. The command filters produce the commands and their first derivatives without differentiation. This method of generating the command derivatives also allows the method to be used for (stabilizable) nontriangular systems.

C. State and Actuator Constraint Stability Result

Given the preceding procedure, the following stability result was proved in Ref. 34.

Theorem 1: Given a system described as Eqs. (1) and (2), let α_1, β_1, and μ_1 be selected such that assumption 1 is true. Then the online approximation-based controller of Eqs. (9–15), with physical constraints, solves the tracking problem with the following properties: 1) $\bar{x}_1$, $\bar{x}_2$, θ_{f_1}, θ_{f_2}, θ_{G_1}, $\theta_{G_2} \in \mathcal{L}_\infty$; 2) $\bar{x}_1$ and $\bar{x}_2 \in \mathcal{L}_2$; and 3) $\lim_{t\to\infty}\bar{x}_1(t) = 0$; and, $\lim_{t\to\infty}\bar{x}_2(t) = 0$. Note that this theorem can be applied recursively $(r-1)$ times to address a system with r state blocks. The objective of this paper is to apply this theorem recursively to derive a controller for an UAV subject to model error, faults, battle damage, and state and/or actuator limitations.

Aircraft dynamics are derived and discussed in several references, for example, Refs. 46 and 47. For convenience of the reader, the final equations (and aircraft notation and parameters) are summarized in Appendix B. The controller will be implemented by viewing the aircraft dynamics as having three sets of states: airspeed and flight-path angles (χ, γ, V), wind-axis angles (μ, α, β), and angular rates (P, Q, R). Online approximation-based control of the airspeed and flight-path angles will be addressed first to establish the control signal μ_1 required to satisfy assumption 1. Then, the wind-axis and angular-rate loops will each be addressed by recursive applications of theorem 1. The control laws, error dynamics, and parameter adaptation laws are derived in Secs. III–V. The control law and its stability properties are summarized in Sec. VI. The structure of the

online approximators are defined in Sec. VII. Section VIII contains a simulation example and discussion of the controller properties.

III. Airspeed and Flight-Path-Angle Control

Let the state vector z_1 be defined by $z_1 = [\chi, \gamma, V]^T$. To initiate the constrained backstepping process of Sec. II, we need a control law that stabilizes the z_1 tracking error dynamics in the sense described by assumption 1. We assume that the command signals (χ_c, γ_c, V_c) and their derivatives are available, bounded, and continuous. [If the external command generation system only provides the bounded command signals $(\chi_c^o, \gamma_c^o, V_c^o)$, but fails to provide the required derivatives, then signals (χ_c, γ_c, V_c) and their derivatives (satisfying the preceding assumptions) can be generated without differentiation by a command filter such as that shown in Appendix A.] The airspeed will be controlled via the thrust. The flight-path angles will be controlled through the wind-axis angles; therefore, $\mu_1^\top = [\mu, \alpha, T]$.

The airspeed and flight-path-angle dynamics of Eqs. (73a–73c) can be represented as

$$\dot{z}_1 = \bar{A}_1 f_1 + F_1 + G_1(\mu_1, x) \tag{16}$$

with

$$\bar{A}_1 = \frac{1}{mV}\begin{bmatrix} \sin\beta\cos\mu/\cos\gamma & \cos\beta\cos\mu/\cos\gamma & 0 \\ -\sin\beta\sin\mu & \cos\beta\sin\mu & 0 \\ -V\cos(\beta) & V\sin(\beta) & 0 \end{bmatrix}$$

$$F_1 = \begin{bmatrix} -T\cos\alpha\sin\beta\cos\mu\dfrac{1}{mV\cos\gamma} \\ (T\cos\alpha\sin\beta\sin\mu - mg\cos\gamma)\dfrac{1}{mV} \\ -g\sin\gamma \end{bmatrix}$$

$$\mu_1 = \begin{bmatrix} \mu_{1_1} \\ \mu_{1_2} \\ \mu_{1_3} \end{bmatrix}, \qquad f_1(x) = \begin{bmatrix} D(x) \\ Y(x) \\ L(x) \end{bmatrix}$$

and

$$G_1(\mu_1, x) = \begin{bmatrix} u_\chi \\ u_\gamma \\ u_V \end{bmatrix} = \begin{bmatrix} \dfrac{g(\mu_{1_2}, x)\sin\mu_{1_1}}{mV\cos\gamma} \\ \dfrac{g(\mu_{1_2}, x)\cos\mu_{1_1}}{mV} \\ \dfrac{\cos\beta\cos\mu_{1_2}}{m}T \end{bmatrix} \tag{17}$$

where

$$g(\mu_{1_2}, x) = L(\mu_{1_2}, x) + T\sin\mu_{1_2} \tag{18}$$

$$L(\mu_{1_2}, x) = L_o(x) + L_\alpha(x)\mu_{1_2} \tag{19}$$

The drag, lift, and side-force functions that are used in the definitions of f_1, $L_o(x)$, and $L_\alpha(x)$ are unknown. The function F_1 is known.

We select the control signal μ_1, with K_1 positive definite, so that the following equation is satisfied:

$$\hat{G}_1(\boldsymbol{\mu}_1, x) = -\bar{A}_1\hat{f}_1 - F_1 + \dot{z}_{1_c} - K_1\tilde{z}_1 \tag{20}$$

where $\hat{f}_1^\top = [\hat{D}(x), \hat{Y}(x), \hat{L}(x)]$ and

$$\hat{G}_1 = \begin{bmatrix} \dfrac{\hat{g}(\mu_{1_2}, x)\sin\mu_{1_1}}{mV\cos\gamma} \\ \dfrac{\hat{g}(\mu_{1_2}, x)\cos\mu_{1_1}}{mV} \\ \dfrac{\cos\beta\cos\mu_{1_2}}{m}T \end{bmatrix} \tag{21}$$

with $\hat{g}(\mu_{1_2}, x) = [\hat{L}(\mu_{1_2}, x) + T\sin\mu_{1_2}]$. The functions $[\hat{D}(x), \hat{Y}(x), \hat{L}(x)]$ are approximations to $[D(x), Y(x), L(x)]$. The effect of the error between these functions is considered in the analysis of Sec. IV.A.

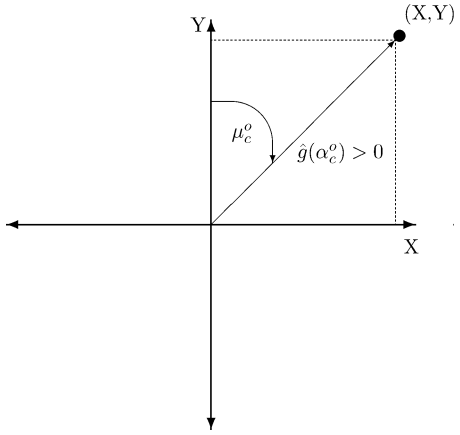
The value of the vector $\boldsymbol{\mu}_1$ in the left-hand side of Eq. (20) must be determined, as it serves as the input to the next loop. Because all quantities in the right-hand side of Eq. (20) are known, the value of $\hat{G}_1(\mu_1, x)$ can be computed at any time instant. The purpose of the following paragraph is to discuss the solution of Eq. (21) for $\boldsymbol{\mu}_1$. Note that $\mu_{1_1} = \mu_c^o$ and $\mu_{1_2} = \alpha_c^o$ are the roll-angle and angle-of-attack commands. Also, to decrease the complexity of the notation, we will use the notation $\hat{g}(\alpha_c^o)$ instead of $\hat{g}(\mu_{1_2}, x)$. Finally, for complete specification of the desired wind-axis state, we will always specify β_c^o as zero.

Defining (X, Y) such that the first two rows of Eq. (21) can be written as

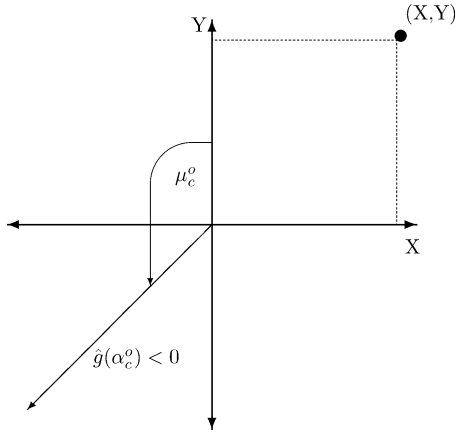
$$X \equiv \cos(\gamma)mVu_\chi = \hat{g}(\alpha_c^o)\sin(\mu_c^o) \tag{22}$$

$$Y \equiv mVu_\gamma = \hat{g}(\alpha_c^o)\cos(u_c^o) \tag{23}$$

we can interpret (X, Y) as rectangular coordinates for a point with (signed) radius $\hat{g}(\alpha_c^o)$ and angle μ_c^o relative to the positive Y axis. Because the force $\hat{g}(\alpha_c^o)$ can be either positive or negative, there are always two possible solutions, as depicted in Figs. 1a and 1b. Switching between the two possible solutions requires μ_c^o to change by 180 deg as $\hat{g}(\alpha_c^o)$ reverses its sign. When $\hat{g}(\alpha_c^o)$ reverses its sign, the point (X, Y) passes through the origin, which is an instantaneous loss of controllability. If $\hat{g}(\alpha_c^o)$ is selected to be positive for a sufficiently aggressive diving turn (i.e., $\dot{\chi}_c$ and $\dot{\gamma}_c$ both large), then the maneuver would be performed with the aircraft inverted (i.e., roll



a) The (α_c^o, μ_c^o) solution with positive lift



b) The (α_c^o, μ_c^o) solution with negative lift

Fig. 1 Two possible choices for α_c^o and μ_c^o to solve the (χ, γ) control.

greater than 90 deg). When choosing (μ_c^o, α_c^o) to satisfy Eq. (21), the designer should only allow $\hat{g}(\alpha_c^o)$ to reverse its sign when $\hat{u}_{\chi c}^o$ is near zero. If the sign of $g(\alpha)$ reversed while $\hat{u}_{\chi c}^o$ was nonzero, then μ_c^o would also need to change so that $[\sin(\mu_c^o), \cos(\mu_c^o)]$ would have the correct signs to attain the desired control signals. This change is a 180-deg roll reversal.

Once μ_c^o and α_c^o have been specified, the third equation of Eq. (21) can be directly solved for T.

IV. Wind-Axis Angle Control

Let z_1 be as defined in Sec. III. Define $z_2 = [\mu, \alpha, \beta]^T$. Then the combined (z_1, z_2) dynamics are

$$\dot{z}_1 = \bar{A}_1(x) f_1 + F_1(x) + G_1(z_2, x, T) \tag{24}$$

$$\dot{z}_2 = A_2(x) f_1 + F_2(x) + B_2 \mu_2 \tag{25}$$

where

$$B_2 = \begin{bmatrix} \dfrac{\cos\alpha}{\cos\beta} & 0 & \dfrac{\sin\alpha}{\cos\beta} \\ -\cos\alpha\tan\beta & 1 & -\sin\alpha\tan\beta \\ \sin\alpha & 0 & -\cos\alpha \end{bmatrix}, \qquad A_2 = \frac{1}{mV}\begin{bmatrix} \sin\beta\cos\mu\tan\gamma & \cos\beta\cos\mu\tan\gamma & (\tan\beta + \tan\gamma\sin\mu) \\ 0 & 0 & -1/\cos\beta \\ \sin\beta & \cos\beta & 0 \end{bmatrix}$$

$$F_2 = \frac{1}{mV}\begin{bmatrix} (\sin\alpha\tan\gamma\sin\mu + \sin\alpha\tan\beta - \cos\alpha\tan\gamma\cos\mu\sin\beta)T - mg\cos\gamma\cos\mu\tan\beta \\ [-T\sin\alpha + mg\cos\gamma\cos\mu]\dfrac{1}{\cos\beta} \\ -T\sin\beta\cos\alpha + mg\cos\gamma\sin\mu \end{bmatrix}$$

are known functions and $\mu_2^\top = [P, Q, R]$. Note that the (z_1, z_2) dynamics are not triangular because $\bar{A}_1$, f_1, F_1 all depend on z_2. Nevertheless, the command filtered backstepping approach is applicable.

Let $z_{2_c}^o = \boldsymbol{\mu}_1$ with $\boldsymbol{\mu}_1$ as defined in Sec. III to satisfy Eqs. (20) and (21). Pass $z_{2_c}^o$ through a command filter (see Appendix A) to produce the magnitude, rate, and bandwidth-limited signals z_{2_c} and $\dot{z}_{2_c}$. Define $\bar{z}_1 = \tilde{z}_1 - \xi_1$, where the variable ξ_1 is the output of the filter

$$\dot{\xi}_1 = -K_1\xi_1 + \left[\hat{G}_1(z_2, x) - \hat{G}_1\left(z_{2_c}^o, x\right)\right] \tag{26}$$

As discussed in Sec. II.B, the signal $\xi_1(t)$ is bounded. Select $\mu_{2_c}^o$ such that

$$B_2\mu_{2_c}^o = -K_2\tilde{z}_2 + \dot{z}_{2c} - A_2\hat{f}_1 - F_2 \tag{27}$$

with K_2 positive definite. Equation (27) can always be satisfied because B_2 is well defined and nonsingular (for $\beta \neq \pm 90$ deg). Pass $\mu_{2_c}^o$ through a filter, such as that shown in Fig. 2, to produce μ_{2_c} and $\dot{\mu}_{2_c}$. Define $\bar{z}_2 = \tilde{z}_2 - \xi_2$, where the variable ξ_2 is the output of the filter

$$\dot{\xi}_2 = -K_2\xi_2 + B_2\left(\mu_{2_c} - \mu_{2_c}^o\right) \tag{28}$$

A. Tracking Error Dynamics

Given the definitions of the preceding section, the dynamics of the z_1 and z_2 tracking errors can be derived. Starting from Eq. (24),

$$\begin{aligned} \dot{\tilde{z}}_1 &= \bar{A}_1 f_1 + F_1 + \hat{G}_1(\mu_1, x) - \dot{z}_{1_c} + [G_1(z_2, x) - \hat{G}_1(z_2, \dot{x})] \\ &\quad + [\hat{G}_1(z_2, x) - \hat{G}_1(\mu_1, x)] \\ &= -K_1\tilde{z}_1 - \bar{A}_1\tilde{f}_1 + [G_1(z_2, x) - \hat{G}_1(z_2, x)] \\ &\quad + [\hat{G}_1(z_2, x) - \hat{G}_1(\mu_1, x)] \\ &= -K_1\tilde{z}_1 - A_1\tilde{f}_1 + [\hat{G}_1(z_2, x) - \hat{G}_1(\mu_1, x)] \end{aligned} \tag{29}$$

where $\tilde{f}(x) = \hat{f}(x) - f(x)$ and algebraic manipulations result in $A_1\tilde{f}_1 = \bar{A}_1\tilde{f}_1 - [G_1(z_2, x) - \hat{G}_1(z_2, x)]$ with

$$A_1 = \frac{1}{mV}\begin{bmatrix} \sin\beta\cos\mu/\cos\gamma & \cos\beta\cos\mu/\cos\gamma & \dfrac{\sin\mu}{\cos\gamma} \\ -\sin\beta\sin\mu & \cos\beta\sin\mu & \cos\mu \\ -V\cos(\beta) & V\sin(\beta) & 0 \end{bmatrix} \tag{30}$$

The tracking error dynamics for z_2 are

$$\begin{aligned} \dot{\tilde{z}}_2 &= A_2 f_1 + F_2 + B_2\mu_{2_c}^o - \dot{z}_{2_c} + B_2\left(\mu_2 - \mu_{2_c}^o\right) \\ &= -K_2\tilde{z}_2 - A_2\tilde{f}_2 + B_2\left(\mu_2 - \mu_{2_c}^o\right) \end{aligned} \tag{31}$$

Combining Eqs. (26) and (28), respectively, with Eqs. (29) and (31), the dynamics of the compensated tracking errors are

$$\begin{aligned} \dot{\bar{z}}_1 &= -K_1\bar{z}_1 - A_1\tilde{f}_1 + [\hat{G}_1(z_2, x) - \hat{G}_1(\mu_1, x)] \\ &\quad - \left\{-K_1\xi_1 + \left[\hat{G}_1(z_2, x) - \hat{G}_1\left(z_{2_c}^o, x\right)\right]\right\} \\ &= -K_1\bar{z}_1 - A_1\tilde{f}_1 \end{aligned} \tag{32}$$

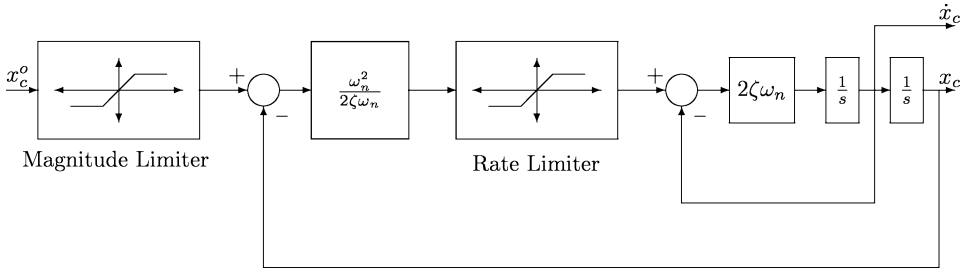


Fig. 2 Filter that generates the command and command derivative while enforcing magnitude, bandwidth, and rate limit constraints.

$$\dot{\tilde{z}}_2 = -K_2\tilde{z}_2 - A_2\tilde{f}_2 + B_2\left(\mu_2 - \mu_{2_c}^o\right) - \left[-K_2\xi_2 + B_2\left(\mu_{2_c} - \mu_{2_c}^o\right)\right] = -K_2\bar{z}_2 - A_2\tilde{f}_2 \tag{33}$$

Equations (32) and (33) are in the form that is required to prove the desired stability properties.

B. Online Approximation and Stability Analysis

Let the parameter update be defined by

$$\dot{\theta}_{f_1} = \Gamma_{f_1}\Phi_{f_1}\left(\bar{z}_1^T A_1 + \bar{z}_2^T A_2\right) \tag{34}$$

Define the Lyapunov function

$$\mathcal{V}_1 = \tfrac{1}{2}\left[\bar{z}_1^T\bar{z}_1 + \bar{z}_2^T\bar{z}_2 + \operatorname{trace}\left(\tilde{\theta}_{f_1}^T\Gamma_{f_1}^{-1}\tilde{\theta}_{f_1}\right)\right] \tag{35}$$

The time derivative of $\mathcal{V}_1$ along solutions of Eqs. (32) and (33) is given by

$$\dot{\mathcal{V}}_1 = \bar{z}_1^T\dot{\bar{z}}_1 + \bar{z}_2^T\dot{\bar{z}}_2 + \operatorname{trace}\left(\tilde{\theta}_{f_1}^T\Gamma_{f_1}^{-1}\dot{\tilde{\theta}}_{f_1}\right) = -\bar{z}_1^T K_1\bar{z}_1 - \bar{z}_2^T K_2\bar{z}_2 \le 0 \tag{36}$$

This completes the first step in the recursive application of the methodology of Sec. II. Minor extensions to the method of that section were required because $\boldsymbol{\mu}_1$ appears in a nonlinear fashion in Eq. (20). Note that the stability analysis is not yet complete, and the next section directly applies Theorem 1, to complete the control design. The stability analysis is concluded in Sec. VI and summarized in Theorem 2.

V. Body-Axis Angular Rate Control

Given the results of the preceding section, we are now in a position to design the angular-rate control law using Theorem 1. To do so, let $x_1 = [z_1^T, z_2^T]^T$, and let $x_2 = [P, Q, R]^T$. The dynamics can be written as

$$\dot{z}_1 = \bar{A}_1(x)f_1 + F_1(x) + G_1(z_2, x, T)$$

$$\dot{z}_2 = A_2(x)f_1 + F_2(x) + B_2(x)z_3$$

$$\dot{z}_3 = A_3 f_3 + F_3(x) + B_3 G_3\delta$$

where

$$B_3 = A_3 = \begin{bmatrix} c_3 & 0 & C_4 \\ 0 & c_7 & 0 \\ c_4 & 0 & c_9 \end{bmatrix}$$

are known matrices,

$$F_3 = \begin{bmatrix} (c_1R + c_2P)Q \\ c_5PR - c_6(P^2 - R^2) \\ (c_8P - c_2R)Q \end{bmatrix}$$

is a known function, and

$$f_3 = \begin{bmatrix} \bar{L}' \\ \bar{M}' \\ \bar{N} \end{bmatrix}$$

and

$$G_3 = \begin{bmatrix} \bar{L}_{\delta_1} & \cdots & \bar{L}_{\delta_G} \\ \bar{M}_{\delta_1} & \cdots & \bar{M}_{\delta_6} \\ \bar{N}_{\delta_1} & \cdots & \bar{N}_{\delta_G} \end{bmatrix}$$

are unknown functions, and

$$\delta = \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_6 \end{bmatrix}$$

is the control signal. For convenience of representation of the control surface effectiveness matrix, the moment functions are decomposed as

$$\bar{L} = \bar{L}' + \sum_{i=1}^{m}\bar{L}_{\delta_i}\delta_i, \qquad \bar{M} = \bar{M}' + \sum_{i=1}^{m}\bar{M}_{\delta_i}\delta_i$$

and

$$\bar{N} = \bar{N}' + \sum_{i=1}^{m}\bar{N}_{\delta_i}\delta_i$$

To specify the angular-rate control signal, we define

$$x_{3c}^o = \mu_{2c}^o - \xi_3 \tag{37}$$

Pass x_{3c}^o through a filter, such as that shown in Fig. 2, to produce x_{3_c} and $\dot{x}_{3_c}$, which are magnitude, rate, and bandwidth limited. To accommodate the additional inner control loop, we change the definition of ξ_2 from Eq. (28) to

$$\dot{\xi}_2 = -K_2\xi_2 + B_2\left(z_{3_c} - z_{3_c}^o\right) \tag{38}$$

To achieve tracking of z_{3c} by z_3, select continuous δ_c^o such that

$$B_3\hat{G}_3\delta_c^o = -A_3\hat{f}_3 - F_3 - K_3\tilde{z}_3 + \dot{z}_{3_c} - B_2^T\bar{z}_2 \tag{39}$$

with K_3 positive definite. When the aircraft is overactuated, the matrix $B_3\hat{G}_3$ will have more columns than rows and will have full row rank. Therefore, many solutions to Eq. (39) exist, and some form of actuator distribution[48,49] is required to select δ_c^o.

We pass δ_c^o through a filter, such as that shown in Fig. 2, to produce δ, which is within the magnitude, rate, and bandwidth limitations of the actuation system. [Alternatively, if the surface deflection is measured, then the signal δ_c^o could be used as the commanded surface positions, and the measured surface deflection vector δ can be used directly to calculate ξ_3. No change is required in the notation of Eq. (40). This is the method that will be used in the simulation example of Sec. VIII.] The signal ξ_3 is the output of the filter

$$\dot{\xi}_3 = -K_3\xi_3 + B_3\hat{G}_3\left(\delta - \delta_c^o\right) \tag{40}$$

Define $\bar{z}_3 = \tilde{z}_3 - \xi_3$.

A. Tracking Error Dynamics

The tracking error and compensated tracking error dynamics for z_1 are still given by Eqs. (29) and (32) because the definitions of z_{2_c}, $\boldsymbol{\mu}_1$, and ξ_1 have not changed.

Because the state z_3 is used to implement μ_{2_c}, the tracking error dynamics for z_2 become

$$\begin{aligned}\dot{\tilde{z}}_2 &= A_2f_1 + F_2(x) + B_2z_{3_c}^o - \dot{z}_{2_c} + B_2\left(z_3 - z_{3_c}\right) + B_2\left(z_{3_c} - z_{3_c}^o\right) \\ &= A_2f_1 + F_2(x) + B_2\mu_{2_c}^o - B_2\xi_3 - \dot{z}_{2_c} \\ &\quad + B_2\left(z_3 - z_{3_c}\right) + B_2\left(z_{3_c} - z_{3_c}^o\right) \\ &= -K_2\tilde{z}_2 + B_2\tilde{z}_3 - B_2\xi_3 - A_2\tilde{f}_1 + B_2\left(z_{3c} - z_{3c}^o\right) \\ &= -K_2\tilde{z}_2 + B_2\tilde{z}_3 - A_2\tilde{f}_1 + B_2\left(z_{3c} - z_{3c}^o\right)\end{aligned} \tag{41}$$

Combining Eqs. (38) and (41), the compensated tracking error dynamics for z_2 are

$$\dot{\bar{z}}_2 = -K_2\bar{z}_2 - A_2\tilde{f}_1 + B_2\bar{z}_3 \tag{42}$$

The tracking error dynamics for z_3 are

$$\begin{aligned}\dot{\tilde{z}}_3 &= A_3f_3 + F_3(x) + B_3\hat{G}_3\delta_c^o - \dot{z}_{3_c} \\ &\quad + B_3\hat{G}_3\left(\delta - \delta_c^o\right) + B_3(G_3 - \hat{G}_3)\delta \\ &= -K_3\tilde{z}_3 - A_3\tilde{f}_3 + B_3\tilde{G}_3\delta + B_3\hat{G}_3\left(\delta - \delta_c^o\right) - B_2^T\bar{z}_2\end{aligned} \tag{43}$$

where $\tilde{f}_3 = \hat{f}_3 - f_3$ and $\tilde{G}_3 = \hat{G}_3 - G_3$. The compensated tracking error dynamics for z_3 are

$$\dot{\bar{z}}_3 = -K_3\bar{z}_3 - A_3\tilde{f}_3 - B_3\tilde{G}_3\delta - B_2^T\bar{z}_2 \tag{44}$$

B. Online Approximation and Stability Analysis

Let the moment function parameter adaptation laws be

$$\dot{\hat{\theta}}_{f_3} = \dot{\theta}_{f_3} = \Gamma_{f_3}\Phi_{f_3}\left(\bar{z}_3^T A_3\right) \tag{45}$$

$$\dot{\hat{\theta}}_{G_{3j}} = \dot{\theta}_{G_{3j}} = \Gamma_{G_{3j}}\Phi_{G_{3j}}\left(\bar{z}_3^T \delta_j B_3\right) \tag{46}$$

Define the Lyapunov function

$$\mathcal{V} = \frac{1}{2}\Bigg[\sum_{i=1}^{3}\bar{z}_i^T\bar{z}_i + \operatorname{trace}\left(\tilde{\theta}_{f_1}^T\Gamma_{f_1}^{-1}\tilde{\theta}_{f_1}\right) + \operatorname{trace}\left(\tilde{\theta}_{f_3}^T\Gamma_{f_3}^{-1}\tilde{\theta}_{f_3}\right) + \sum_{j=1}^{m}\operatorname{trace}\left(\tilde{\theta}_{G_{3j}}^T\Gamma_{G_{3j}}^{-1}\tilde{\theta}_{G_{3j}}\right)\Bigg] \tag{47}$$

The time derivative of $\mathcal{V}$ along solutions of the closed-loop compensated tracking error dynamics defined by Eqs. (32), (42), and (44) is

$$\begin{aligned}\dot{\mathcal{V}} &= \bar{z}_1^T\dot{\bar{z}}_1 + \bar{z}_2^T\dot{\bar{z}}_2 + \bar{z}_3^T\dot{\bar{z}}_3 + \operatorname{trace}\left(\tilde{\theta}_{f_1}^T\Gamma_{f_1}^{-1}\dot{\tilde{\theta}}_{f_1}\right) + \operatorname{trace}\left(\tilde{\theta}_{f_3}^T\Gamma_{f_3}^{-1}\dot{\tilde{\theta}}_{f_3}\right) + \sum_{j=1}^{m}\operatorname{trace}\left(\tilde{\theta}_{G_{3j}}^T\Gamma_{G_{3j}}^{-1}\dot{\tilde{\theta}}_{G_{3j}}\right)\\ &= -\bar{z}_1^T K_1\bar{z}_1 - \bar{z}_2^T K_2\bar{z}_2 - \bar{z}_3^T K_3\bar{z}_3 \le 0.\end{aligned} \tag{48}$$

Equation (48), which shows that $\dot{\mathcal{V}}$ is negative semidefinite, will be used in Sec. VI to prove the stability properties of the UAV online approximation-based controller.

VI. Control Law and Stability Properties

This section summarizes in an organized fashion the control law implementation equations that are distributed throughout the preceding sections and summarizes the stability properties that apply.

For the input signals z_{1_c} and $\dot{z}_{1_c}$ the control law is given by the following:

1) Select the control signals $\boldsymbol{\mu}_1$ so that

$$\hat{G}_1(\boldsymbol{\mu}_1, x) = -K_1\tilde{z}_1 + \dot{z}_{1_c} - \bar{A}_1\hat{f}_1 - F_1 \tag{49}$$

where $\tilde{z}_1 = z_1 - z_{1_c}$. Define $z_{2_c}^o = \boldsymbol{\mu}_1$. Filter $z_{2_c}^o$ to enforce magnitude, rate, and bandwidth constraints appropriate for (μ, α, β) and to produce as outputs the signals z_{2_c} and $\dot{z}_{2_c}$.

2) Select $\mu_{2_c}^o$ such that

$$B_2\mu_{2_c}^o = -K_2\tilde{z}_2 + \dot{z}_{2c} - A_2\hat{f}_1 - F_2 \tag{50}$$

where $\tilde{z}_2 = z_2 - z_{2_c}$. Because B_2 is square and invertible, this solution is unique and straightforward. Define $z_{3_c}^o = \mu_{2_c}^o - \xi_3$. Filter $z_{3_c}^o$ to enforce magnitude, rate, and bandwidth constraints appropriate for the variables (P, Q, R) and to produce as outputs the signals z_{3_c} and $\dot{z}_{3_c}$.

3) Select δ_c^o such that

$$B_3\hat{G}_3\delta_c^o = -K_3\tilde{z}_3 + \dot{z}_{3_c} - A_3\hat{f}_3 - F_3 - B_2^T\bar{z}_2 \tag{51}$$

where $\tilde{z}_3 = z_3 - z_{3_c}$. If $m > 3$, then the system is overactuated, and some form of actuator distribution process will be required. This actuator distribution can be used to limit the extent and rate of the commanded actuator deflections.

4) Implement the following bank of filters to compute ξ_i for $i = 1, 2, 3$:

$$\dot{\xi}_1 = -K_1\xi_1 + \hat{G}_1(z_2, x) - \hat{G}_1\left(z_{2_c}^o, x\right) \tag{52}$$

$$\dot{\xi}_2 = -K_2\xi_2 + B_2\left(z_{3_c} - z_{3_c}^o\right) \tag{53}$$

$$\dot{\xi}_3 = -K_3\xi_3 + B_3\hat{G}_3\left(\delta - \delta_c^o\right) \tag{54}$$

The controller includes online approximation of the unknown force and moment functions using the following parameter estimation equations:

$$\dot{\theta}_{f_1} = \Gamma_{f_1}\phi_{f_1}\left(\bar{z}_1^T A_1 + \bar{z}_2^T A_2\right) \tag{55}$$

$$\dot{\theta}_{f_3} = \Gamma_{f_3}\phi_{f_3}\left(\bar{z}_3^T A_3\right) \tag{56}$$

$$\dot{\theta}_{G_{3j}} = \Gamma_{G_{3j}}\phi_{G_{3j}}\left(\bar{z}_3^T\delta_j B_3\right) \qquad \text{for} \qquad j = 1, \ldots, m \tag{57}$$

Such online approximators are especially useful on UAVs, where the aerodynamics can change during flight, for example, because of battle damage. For the controller just summarized, the following theorem summarizes the stability properties.

Theorem 2: Assuming that the functions ϕ_{f_1}, ϕ_{f_3}, and $\phi_{G_{3j}}$ are bounded, the online approximation-based controller summarized in Eqs. (49–57) has the following properties:

1) The estimated parameters $\theta_{f_1}, \theta_{f_3}, \theta_{G_{3j}}$ and parameter errors $\tilde{\theta}_{f_1}, \tilde{\theta}_{f_3}, \tilde{\theta}_{G_{3j}}$ are bounded.

2) The compensated tracking errors $\bar{z}_1, \bar{z}_2$, and $\bar{z}_3$ are bounded.

3) $\|\bar{z}_i(t)\| \to 0$ as $t \to \infty$ for $i = 1, 2, 3$.

4) $\bar{z}_i(t) \in \mathcal{L}_2$ for $i = 1, 2, 3$.

Proof: Boundedness of the parameter errors is because $\mathcal{V}$ is positive definite in the parameter errors and $\mathrm{d}\mathcal{V}/\mathrm{d}t$ is negative semidefinite [see Eq. (48)]. Therefore, $\mathcal{V}(t) \le \mathcal{V}(0)$ for $t > 0$. This implies that for any $t > 0$,

$$\operatorname{trace}\left(\tilde{\theta}_{f_1}^T\Gamma_{f_1}^{-1}\tilde{\theta}_{f_1}\right) + \operatorname{trace}\left(\tilde{\theta}_{f_3}^T\Gamma_{f_3}^{-1}\tilde{\theta}_{f_3}\right) + \sum_{j=1}^{m}\operatorname{trace}\left(\tilde{\theta}_{G_{3j}}^T\Gamma_{G_{3j}}^{-1}\tilde{\theta}_{G_{3j}}\right) \le \mathcal{V}(0) < \infty$$

This completes the proof of item 1. The boundedness of the compensated tracking errors is shown similarly. The second time derivative of the Lyapunov function is

$$\begin{aligned}\frac{\mathrm{d}^2\mathcal{V}}{\mathrm{d}t^2} &= -\bar{z}_1^T\left(K_1 + K_1^T\right)(-K_1\bar{z}_1 - A_1\tilde{f}_1)\\ &\quad - \bar{z}_2^T\left(K_2 + K_2^T\right)(-K_2\bar{z}_2 - A_2\tilde{f}_1 + B_2\bar{z}_3 - K_3\bar{z}_3)\\ &\quad - \bar{z}_3^T\left(K_3 + K_3^T\right)\left(-K_3\bar{z}_3 - A3\tilde{f}_3 - B_3\hat{G}\delta - B_2^T\bar{z}_2\right)\end{aligned}$$

which is bounded. Therefore, the function $\mathrm{d}\mathcal{V}/\mathrm{d}t$ is uniformly continuous. Barbalat's lemma (p. 123 in Ref. 50) implies that $\mathrm{d}\mathcal{V}/\mathrm{d}t \to 0$ as $t \to \infty$. This requires that $\bar{z}_i^T K_i\bar{z}_i \to 0$ for $i = 1, 2, 3$ as $t \to \infty$, and because $\bar{z}_i^T K_i\bar{z}_i \ge \underline{\lambda}(K_i)\|\bar{z}_i\|^2$, where $\underline{\lambda}(K_i)$ is the minimum eigenvalue of the positive-definite matrix K_i; we see that $\|\bar{z}_i\|^2 \to 0$ as $t \to \infty$ for $i = 1, 2, 3$. This completes the proof of item 3.

Integrating both sides of Eq. (48) yields

$$\mathcal{V}(t) - \mathcal{V}(0) \le \int_0^t -\bar{z}_i^T(\tau)K_i\bar{z}_i(\tau)\,\mathrm{d}\tau \tag{58}$$

$$-\mathcal{V}(0) \le -\int_0^\infty \bar{z}_i^T(\tau)K_i\bar{z}_i(\tau)\,\mathrm{d}\tau \tag{59}$$

$$\mathcal{V}(0) \ge \int_0^\infty \bar{z}_i^T(\tau)K_i\bar{z}_i(\tau)\,\mathrm{d}\tau \tag{60}$$

where $0 \le \mathcal{V}(t) \le \mathcal{V}(0)$ for all $t \ge 0$ and $\mathrm{d}\mathcal{V}/\mathrm{d}t \le 0$ implies that $\lim_{t\to\infty}\mathcal{V}(t)$ is well defined. This completes the proof of item 4. ◇

Remark 1: The third property is particularly interesting. It states that the compensated tracking errors $\bar{z}_i$ approach zero as time approaches infinity, regardless of the input signals $(\chi_c^o, \gamma_c^o, V_c^o)$. No similar statement can be made for the standard tracking errors $\tilde{z}_i$. During periods when the inputs are too aggressive, magnitude, rate, or bandwidth limits can come into effect. With such limits in effect, it is not possible to track the input because the desired control signals are not able to be implemented; therefore, $\tilde{z}_i$ will become nonzero. The control commands not being implementable causes the ξ_i signals to become nonzero. As discussed in the body of the paper, the ξ_i signals do remain bounded. The ξ_i signals compensate the tracking errors so that online approximation can still occur using the $\bar{z}_i$ variables. When the limits are no longer in effect, the ξ_i variables converge to zero, and $\tilde{z}_i$ converges toward $\bar{z}_i$.

Remark 2: If a nominal design model were known and used to define the functions $\hat{f}_1$, $\hat{f}_2$, and $\hat{f}_3$, then the preceding controller can be used without online approximation. The stability and tracking performance would be affected by the errors between the design model and the actual system as indicated in the tracking error equations (32), (42), and (44). In fact, if the command filters were replaced by analytic computation of the command derivatives without rate and magnitude limits, then the ξ_i filters could be removed [i.e., $\xi_i(t) = 0$]. The remaining controller would be a backstepping controller for the aircraft. We mention this only to point out that the approximation-based approach can be considered as a retrofit to a baseline nominal controller designed by the backstepping method. The retrofit would add in command filtering, online approximation, and the ξ filters. The retrofit would ensure that the intermediate commands remain in the specified operating envelope (because of the command filters) and would attain both stability and performance robustness to model error (because of the online approximation).

VII. Approximator Definition

The aircraft dynamics involve three moments ($\bar{L}, \bar{M}, \bar{N}$) and three forces ($D, Y, L$). In the simulation example to follow, the approximations to these functions will be implemented using the nondimensionalized coefficient notation that is standard in the aircraft literature. Each of the coefficient functions C_* is an unknown function that is implemented as $C_*(\alpha, M) = \theta_*^T \phi(\alpha, M)$ [e.g., $C_{D_o}(\alpha, M) = \theta_{D_o}^T \phi(\alpha, M)$], where $\phi(\alpha, M)$ is a regressor vector that is selected by the designer and θ_* is estimated online. M is the Mach number. Note that different regressors can be used for the different functions. This paper uses a single regressor vector $\phi(\alpha, M)$ for all the approximations for notational simplicity. The elements of the regressor vector are third-order cardinal B splines defined on a grid of knot locations. The α knots were spaced every 2 deg over $[-10, 20]$ deg. The Mach knots were spaced every 0.2 over [0.2, 0.8].

The drag approximator is

$$\hat{D} = \bar{q}S\left(C_{D_o} + \sum_{i=1}^{m} C_{D_{\delta_i}}\delta_i\right) = \Psi_D \theta_D^T \phi \tag{61}$$

The matrix $\theta_D = [\theta_{D_o}, \theta_{D_{\delta_1}}, \ldots, \theta_{D_{\delta_\beta}}]$ contains in each column the parameter vector used to approximate one of the coefficient functions, for examples $C_{D_{\delta_i}}(\,) = \theta_{D_{\delta_i}}^T \phi(\,)$, where $\phi(\,)$ is the vector of basis elements. The factor $\Psi_D = \bar{q}S[1, \delta_1, \ldots, \delta_m]$ is a scaling factor $\bar{q}S$ times a signal vector. Similarly, for the other forces and moments,

$$\hat{Y} = \bar{q}S\left(C_{Y_o} + C_{Y_\beta}\beta + C_{Y_P}\frac{bP}{2V} + \sum_{i=1}^{m} C_{Y_{\delta_i}}\delta_i\right) = \Psi_Y \theta_Y^T \phi \tag{62}$$

$$\hat{L} = \bar{q}S\left(C_{L_o} + C_{L_\alpha}\alpha + \sum_{i=1}^{m} C_{L_{\delta_i}}\delta_i\right) = \Psi_L \theta_L^T \phi \tag{63}$$

$$\hat{\bar{L}} = \bar{q}Sb\left(C_{\bar{L}_o} + C_{\bar{L}_P}\frac{bP}{2V} + C_{\bar{L}_R}\frac{bR}{2V} + C_{\bar{L}_\beta}\beta + \sum_{i=1}^{m} C_{\bar{L}_{\delta_i}}\delta_i\right) = \Psi_{\bar{L}} \theta_{\bar{L}}^T \phi \tag{64}$$

$$\hat{\bar{M}} = \bar{q}S\bar{c}\left(C_{\bar{M}_o} + C_{\bar{M}_Q}\frac{\bar{c}Q}{2V} + \sum_{i=1}^{m} C_{\bar{M}_{\delta_i}}\delta_i\right) = \Psi_{\bar{M}} \theta_{\bar{M}}^T \phi \tag{65}$$

$$\hat{\bar{N}} = \bar{q}Sb\left(C_{\bar{N}_o} + C_{\bar{N}_P}\frac{bP}{2V} + C_{\bar{N}_R}\frac{bR}{2V} + C_{\bar{N}_\beta}\beta + \sum_{i=1}^{m} C_{\bar{N}_{\delta_i}}|\delta_i|\right) = \Psi_{\bar{N}} \theta_{\bar{N}}^T \phi \tag{66}$$

with

$$\Psi_Y = \bar{q}S[1, \beta, bp/2V, \delta_1, \ldots, \delta_m]$$

$$\theta_Y = \left[\theta_{Y_o}, \theta_{Y_\beta}, \theta_{Y_P}, \theta_{Y_{\delta_1}}, \ldots, \theta_{Y_{\delta_m}}\right]$$

$$\Psi_L = \bar{q}S[1, \alpha, \delta_1, \ldots, \delta_m], \qquad \theta_L = \left[\theta_{L_o}, \theta_{L_\alpha}, \theta_{L_{\delta_1}}, \ldots, \theta_{L_{\delta_m}}\right]$$

$$\Psi_{\bar{L}} = \bar{q}Sb[1, bP/2V, bR/2V, \beta, |\delta_1|, \ldots, |\delta_m|]$$

$$\theta_{\bar{L}} = \left[\theta_{\bar{L}_o}, \theta_{\bar{L}_P}, \theta_{\bar{L}_R}, \theta_{\bar{L}_\beta}, \theta_{\bar{L}_{\delta_1}}, \ldots, \theta_{\bar{L}_{\delta_m}}\right]$$

$$\Psi_{\bar{M}} = \bar{q}S\bar{c}[1, \bar{c}Q/2V, \delta_1, \ldots, \delta_m]$$

$$\theta_{\bar{M}} = \left[\theta_{\bar{M}_o}, \theta_{\bar{M}_Q}, \theta_{\bar{M}_{\delta_1}}, \ldots, \theta_{\bar{M}_{\delta_m}}\right]$$

$$\Psi_{\bar{N}} = \bar{q}Sb[1, bp/2V, bR/2V, \beta, |\delta_1|, \ldots, |\delta_m|]$$

$$\theta_{\bar{N}} = \left[\theta_{\bar{N}_o}, \theta_{\bar{N}_P}, \theta_{\bar{N}_R}, \theta_{\bar{N}_\beta}, \theta_{\bar{N}_{\delta_1}}, \ldots, \theta_{\bar{N}_{\delta_m}}\right]$$

Each of $\theta_D, \theta_Y, \theta_L, \theta_{\bar{L}}, \theta_{\bar{M}}$, and $\theta_{\bar{N}}$ is a matrix of unknown parameters.

Each of Eqs. (61–66) is linear with respect to the its matrix of unknown parameters; therefore, each approximator can be rewritten into the standard vector form

$$\hat{D} = \Theta_D^T \Phi_D, \qquad \hat{Y} = \Theta_Y^T \Phi_Y, \qquad \hat{L} = \Theta_L^T \Phi_L \tag{67}$$

$$\hat{\bar{L}} = \Theta_{\bar{L}}^T \Phi_{\bar{L}}, \qquad \hat{\bar{M}} = \Theta_{\bar{M}}^T \Phi_{\bar{M}}, \qquad \hat{\bar{N}} = \Theta_{\bar{N}}^T \Phi_{\bar{N}} \tag{68}$$

which is compatible with the approximator form used throughout the preceding sections of this paper. Therefore, the approximator parameters can be adapted according to Eqs. (55–57).

VIII. Simulation Analysis

This section presents simulation results from the application of the control algorithms developed in the preceding sections to the Barron Associates Nonlinear Tailless Aircraft Model (BANTAM), which is a nonlinear six-degree-of-freedom model of a flying-wing UAV. BANTAM was developed primarily using Ref. 51, which contains aerodynamic data from wind-tunnel testing of several flying-wing planforms, but also using analytical estimates of dynamic stability derivatives from DATCOM and HASC-95. The flying-wing airframe is particularly challenging to control because it is statically unstable at low angles of attack and possesses a restricted set of control effectors that provide less yaw authority than the traditional set used on tailed aircraft. The control surfaces consist of two pairs of body flaps mounted on the trailing edge of the wing. Additionally, a pair of spoilers is mounted upstream of the flaps. This configuration generally relies upon the flaps for pitch and roll authority and the spoilers for yaw and drag. The simulation model also contains realistic actuator models for the control effectors with second-order dynamics and both position and rate limits. The body-flap actuators have 40-rad/s bandwidth with ±30-deg position limits and ±90-deg/s rate limits. The spoiler actuators are identical, except that they can only be deflected upward and their motion is limited to 60 deg.

To illustrate the performance and robustness to model error of the control laws developed herein, the aircraft is commanded to perform doublets in both flight-path and heading angles simultaneously while holding constant airspeed and regulating sideslip to zero, that is, coordinated turns. This set of commands is relatively challenging for the autopilot because it induces significant amounts of coupling between all three channels and requires flight at high roll angles. The control parameter settings for this example and design guidelines for their selection are discussed in Sec. IX. The initial values for the approximator parameters $\Theta_D, \Theta_L, \Theta_Y, \Theta_{\bar{L}}, \Theta_{\bar{M}}, \Theta_{\bar{N}}$ do not match the optimal values; therefore, at the start of the simulation online approximation is required to maintain stable flight. The extent of the initial model error is exemplified in Figs. 3 and 4, which will be discussed in greater detail later in this section. The initial model error is significant enough that without online approximation the aircraft is not stable.

To illustrate the effect of constraints on the states and the robustness of the control laws to such constraints, limits on both roll angle and angle of attack are imposed. The roll-angle command is limited to ±70 deg and an upper limit of +10 deg is imposed on the angle of attack. Although this is a rather restrictive limit, it serves to better demonstrate the robustness of the control algorithms to state constraints. Robustness of the control laws to sudden, unforeseen

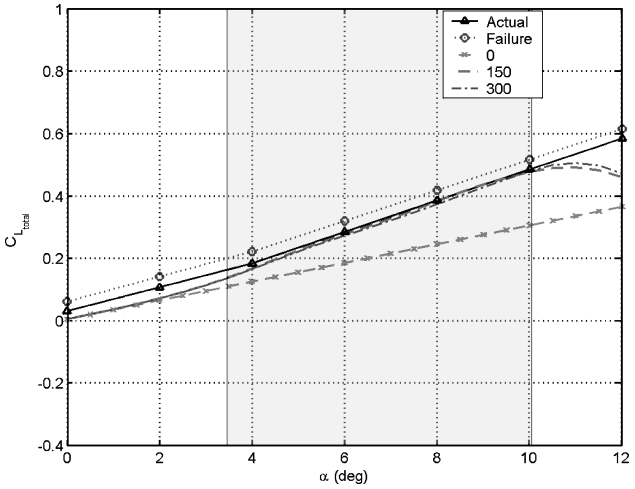


Fig. 3 Learned total lift coefficient.

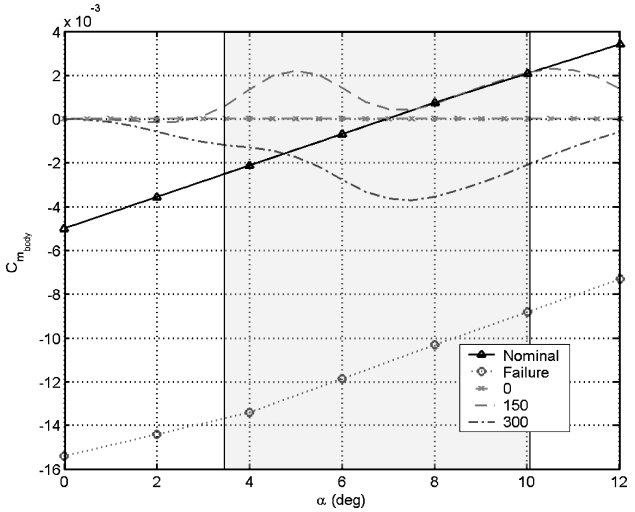


Fig. 4 Learned body pitching-moment coefficient.

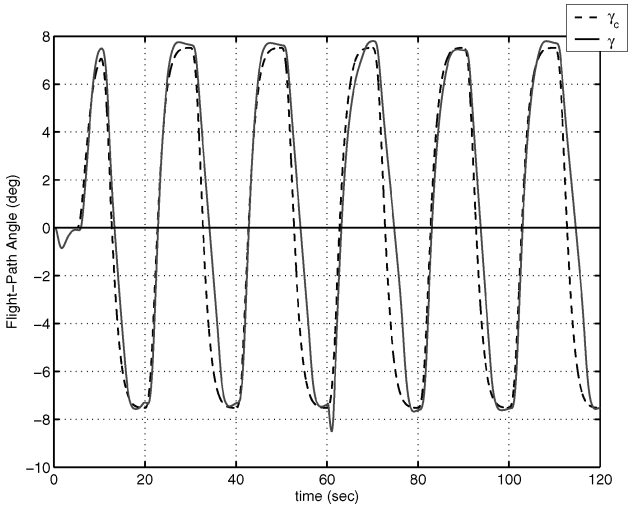


Fig. 5 Flight-path-angle command tracking.

changes in the aircraft dynamics is demonstrated by introduction of a sudden failure at $t = 60$ s in which the left midboard flap moves to and becomes stuck at 15-deg deflection (half of its maximum). This failure introduces a sudden and severe rolling and pitching moment requiring rapid adaptation from the autopilot to maintain stability.

Simulation results given in Figs. 5 and 6 show that the control approach provides good command tracking throughout and that tracking performance is not significantly affected by the midboard flap failure at $t = 60$ s. The initial transient in γ tracking is due to the initial model error; however, accurate tracing is achieved rapidly. The γ tracking error near $t = 25$, 45, 65, and 105 s is caused by α being at its magnitude limit as shown in Fig. 7. The effect of the actuator failure at $t = 60$ s is clearly observed in Fig. 5, but is rapidly removed. Figure 8 shows the flight-path angle pseudocontrols $X \equiv \cos(\gamma) m V u_\chi$ and $Y \equiv m V u_\gamma$ from Eqs. (22) and (23), in polar coordinates as discussed in Sec. III.

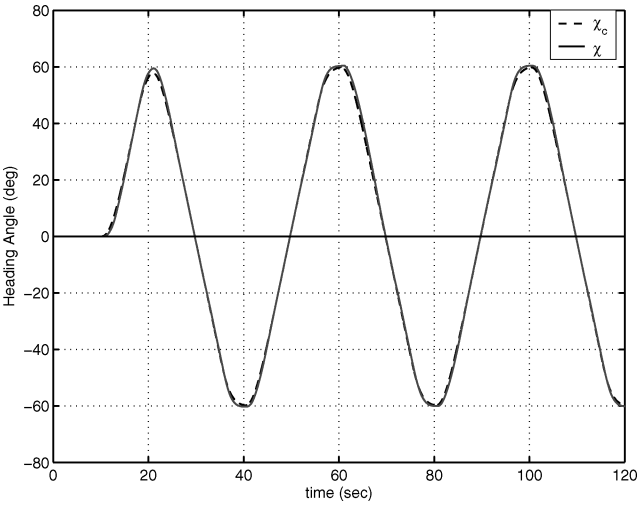


Fig. 6 Heading angle command tracking.

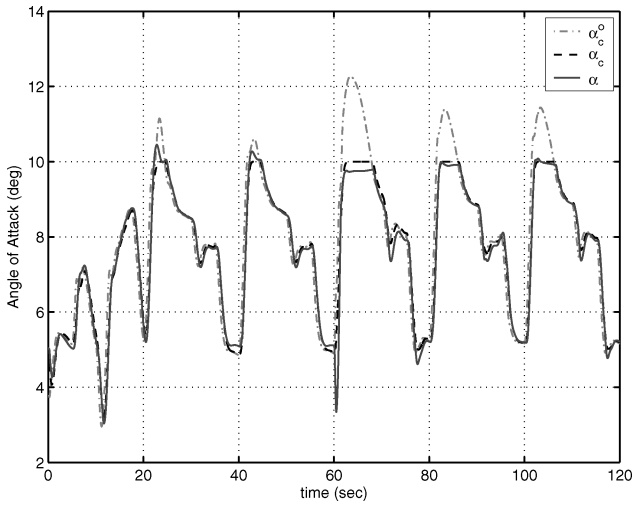


Fig. 7 Angle-of-attack command tracking.

Figures 7 and 9 show the ideal commands (μ_c^o, α_c^o) as dashed-dotted lines, the actual commands (μ_c, α_c) as solid lines, and the states (μ, α) as dashed lines. Inspection of Figs. 7 and 9 more clearly reveals the aggressiveness of the commanded maneuver relative to the $+10$ deg α limit and the ± 70 deg roll-angle limit. Also, although the α command is only at its limit for short instances before the failure, it is much more frequently limited afterward. This is because the failure introduces a nose-down moment and also reduces the lift at a given angle of attack. Thus more angle of attack is needed for the maneuver. Figure 9 shows that the roll-angle tracking is quite good despite the failure. The roll-angle command is also frequently saturated, but this is due primarily to the aggressiveness of the heading command rather than the failure. However, the effects of command saturation of this magnitude would have been detrimental to the learning, often resulting in instability, if it had not been correctly accommodated in the control design by the introduction of the compensated tracking errors.

Figures 10–12 show the commands (P_c, Q_c, R_c) as dashed lines and the states (P, Q, R) as solid lines. The ideal commands (P_c^o, Q_c^o, R_c^o) are not shown as they are essentially the same as

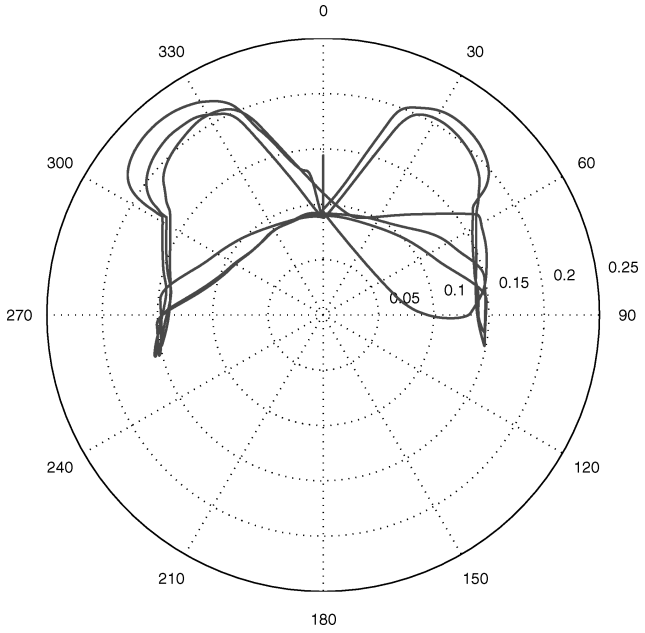


Fig. 8 Flight-path-angle pseudocontrols.

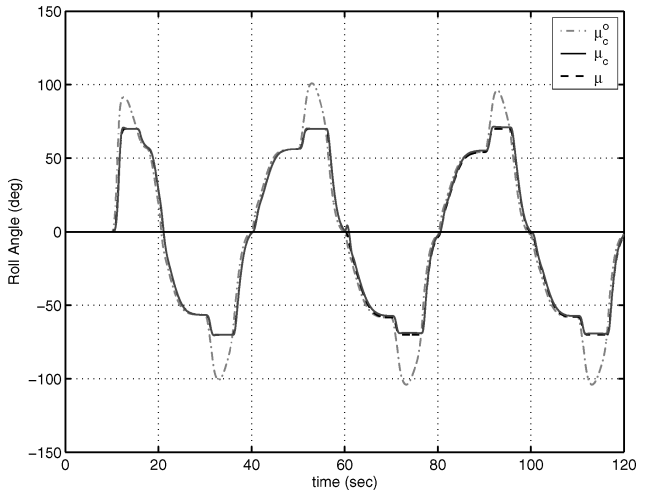


Fig. 9 Roll-angle command tracking.

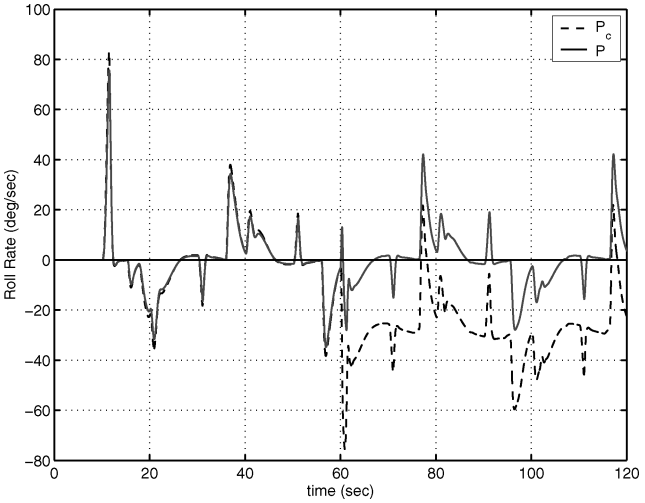


Fig. 10 Body-axis roll-rate command tracking.

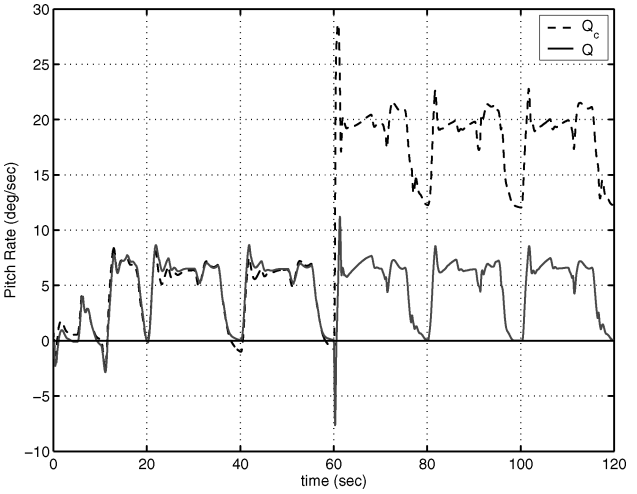


Fig. 11 Body-axis pitch-rate command tracking.

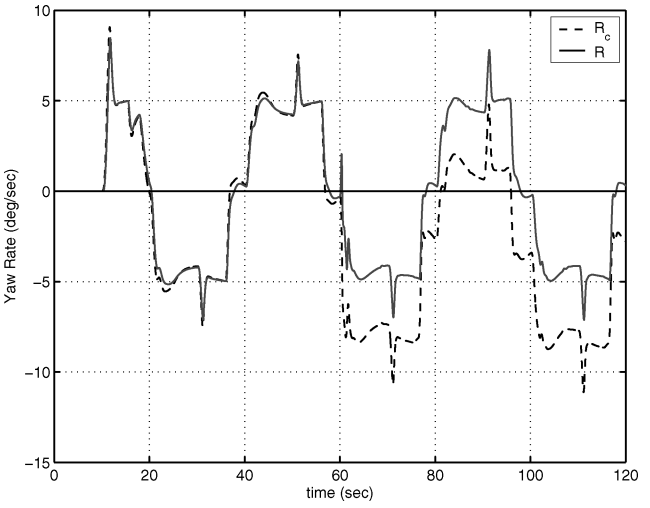


Fig. 12 Body-axis yaw-rate command tracking.

(P_c, Q_c, R_c). These are the control loops that are most affected by the stuck surface. Although these figures seem to indicate the presence of significant postfailure tracking error that should preclude any flight-path angle or wind-axis angle command tracking, it is in fact the commands that are different postfailure and that the (P, Q, R) body-rate responses are virtually unchanged before and after the failure. Note that by Eq. (38) and the fact that we are using the measured surface deflection in Eq. (40), as discussed in the parenthetical note in Sec. V, the signal $\xi_3 \neq 0$ because $\delta \neq \delta_c^o$, because $\xi_3 \neq 0$, $\tilde{z}_3$ and $\bar{z}_3$ cannot both be zero. Theorem 2 proves that $\bar{z}_3 \to 0$; therefore, $\tilde{z}_3 \to \xi_3$. However, the definition of x_{3c}^o in Eq. (37) accommodates for the nonzero ξ_3 to maintain tracking in the outer loops. A comparison of pitch-rate tracking error $\tilde{Q}$ and compensated tracking error $\bar{Q}$ is given in Fig. 13. Comparisons of roll-rate and yaw-rate tracking errors are not given, but are similar.

Next, Figs. 14–16 show the control surface positions over the course of the simulation and clearly show the onset of the surface failure at $t = 60$ s. A stuck-surface failure on the left side of the aircraft is compensated for by the secondary surface on the same side. The reason for this is twofold. First, compensation by the surfaces on the opposite side would create additional pitching/rolling moment and thus result in greater total surface deflection. Second, the yaw moment generated by the flaps in this model is mainly caused by change in the camber of the airfoil rather than to drag from of the surfaces. Therefore, this postfailure surface distribution does not result in significant yawing moment, and what little is created is compensated by the spoilers. These figures also show

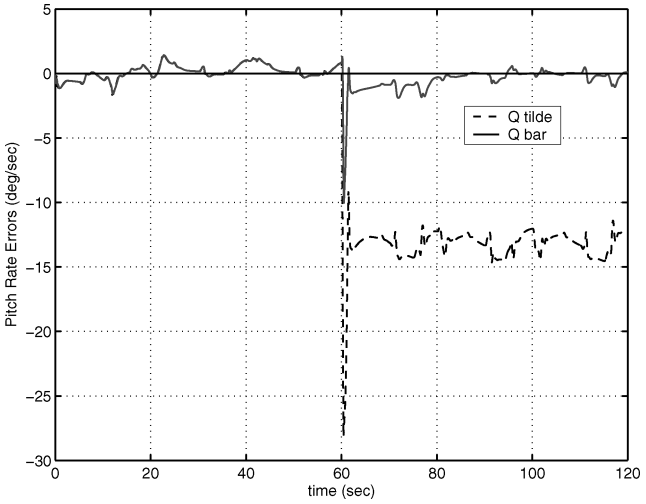


Fig. 13 Pitch-rate command tracking errors.

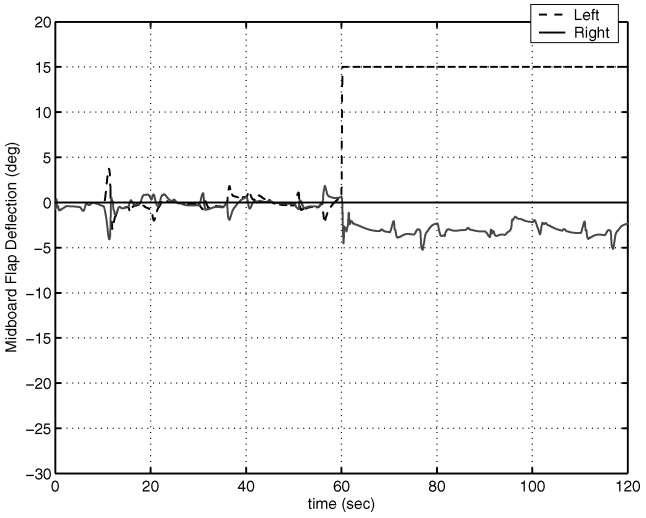


Fig. 14 Midboard flap deflections.

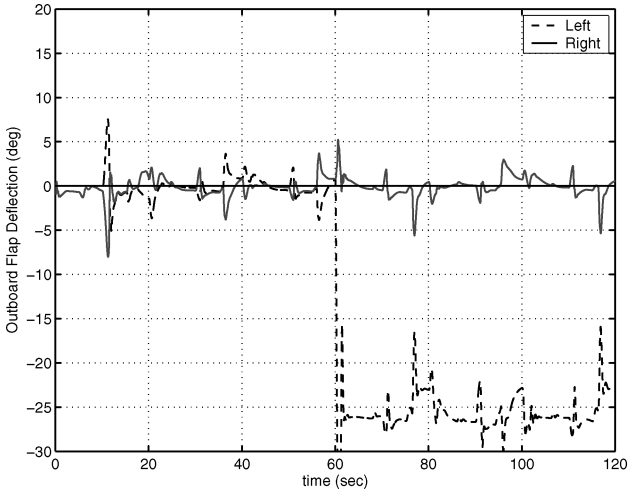


Fig. 15 Outboard flap deflections.

that despite the failure, the surfaces have reasonable motion with no oscillation. This illustrates that the control laws are reasonable in actuation. Note that we have not resorted to high-gain control to meet the control objective in the face of uncertain aerodynamics and state and control limits.

Theorem 2 does not guarantee convergence of the approximated functions. Instead, the approximator parameters can converge to subspaces that allow the compensated tracking error to be zero. As

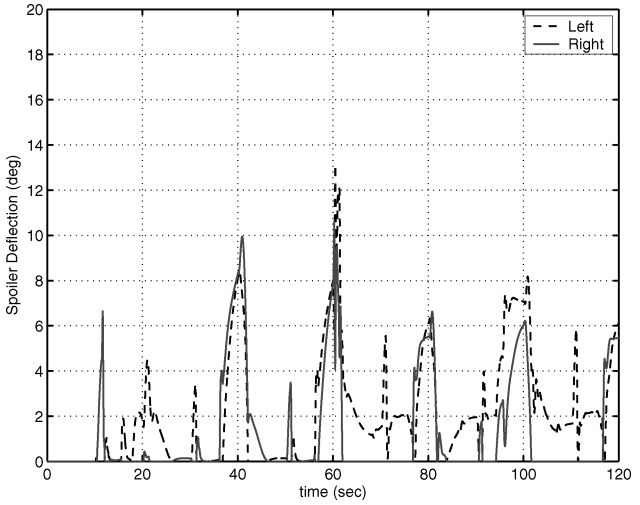


Fig. 16 Spoiler deflections.

the trajectory evolves and the tracking error becomes nonzero, the parameters can adapt, and the Lyapunov function cannot increase. Therefore, the approximated functions can converge towards the true functions. To illustrate that online learning is possible despite the high levels of saturation and sudden changes in aircraft parameters, Figs. 3 and 4 show select time slices of online aerodynamic function learning over the input space (angle of attack). The shaded region represents the range of angle of attack over which the vehicle flew during the simulation. To stress function learning, these results are from a simulation that was run for a longer time. The simulation end time was increased to 300 s, and the failure occurred at 150 s; all other parameters were left unchanged. Although the body lift is learned as two separate components [see Eq. (63)], Fig. 3 presents the complete lift curve ($C_L = C_{L_o} + C_{L_\alpha}\alpha$) rather than the individual components and shows rapid lift learning despite significant initial error and sudden changes in vehicle dynamics. The effect of the initial error in the lift parameters is evidenced in Fig. 5, where γ tracking is initially poor, but improves significantly over time. Next, Fig. 4 shows learning of the body pitching moment C_M. This function is also learned reasonably well and improves with time, albeit not as quickly as lift. An interesting feature of these figures is that function learning is only accomplished over the portion of the input space (marked by the shaded region) over which training samples are accumulated, which stems from the local-support property of the B splines used in the function approximators. The reason for the difference in the pre- and postfailure true values of these functions is that aerodynamic contributions of the stuck surface are subsumed into the body contributions. Convergence of the learned functions to the true functions is not required to meet the control objective, nor is it guaranteed by the analysis herein.[25]

In conclusion, the simulation results presented in this section have demonstrated three main characteristics of the control law developed in the preceding sections: 1) robustness to nonlinear model error, 2) fault tolerance and rapid adaptation to actuator failures, 3) robustness to state and actuator constraints, and 4) excellent tracking of aggressive commands despite the preceding events and limitations.

IX. Parameter Settings and Guidelines

Note that the parameters K_i do not determine the trajectory tracking bandwidth of the ith control law. This is because the time derivative $\dot{x}_{i_c}$ of the reference command x_{i_c} is included in that control law. Instead, the parameters K_i determine the rate of decay of transient errors in $\tilde{x}_i$ caused for example by initial condition errors or disturbances. Note that timescale separation is not required; in fact, the theory of this paper only requires each K_i matrix to be positive definite. However, it it is natural to select $K_3 > K_2 > K_1$. For the simulation of the preceding section, the control gains were $K_1 = \text{diag}(0.8, 0.8, 1.0)$, $K_2 = \text{diag}(4, 4, 2)$, and $K_3 = \text{diag}(20, 20, 10)$.

Table 1 Command filter parameters

Command variable	ω_n, rad/s	Mag limit	Rate limit
χ	0.8	——	——
γ	0.8	±45 deg	10 deg/s
V	0.2	[50, 1000] ft/s	2 ft/s/s
μ	4	±70 deg	120 deg/s
α	4	[−7, 10] deg	——
β	2	±5 deg	15 deg/s
P	20	±120 deg/s	——
Q	20	±30 deg/s	——
R	10	±15 deg/s	——

The specification of each of the command filters is determined by its objective. If the purpose of the command filter is only to compute a command and its derivative (i.e., there are no rate, magnitude, or bandwidth limitations), then the filter could have unity gain and the highest bandwidth consistent with the sampling rate; however, there is usually no reason to select a bandwidth greater than the bandwidth of the command filter of the next inner loop. If there are rate, magnitude, or bandwidth limitations on the signal that is intended to track the output of the command filter, then the command filter must impose those constraints on the command while also outputting the derivative of the command. The command filter parameters used for the simulation in the preceding section are given in Table 1. The damping factor in each filter was 1.0.

X. Conclusions

This paper has been concerned with the problem of designing an aircraft control system capable of tracking ground track, climb rate, and speed commands from a mission planner while being robust to initial model error as well as changes to the nonlinear model that might occur during flight caused by failures and battle damage. This paper derives the controller by extending the command filtering approach presented in Ref. 34. This method uses online approximation to achieve robustness to unmodeled nonlinear effects, even if those effects change during flight. This method also allows the controller to enforce operating envelope constraints on the aircraft states that serve as intermediate control signals in the backstepping controller. The stability properties are proved using Lyapunov methods. The control law and its stability properties are summarized in Sec. VI. Section VIII discusses various aspects of the control approach relative to an unmanned-air-vehicle simulation example.

The presentation of this paper has assumed that the approximators were capable of perfectly fitting the unknown nonlinear functions (i.e., no approximation error). This is, of course, an idealized situation. The effects of approximation error and methods to counter those effects are discussed in Ref. 40.

Appendix A: Magnitude, Rate, and Bandwidth-Limiting Filter

Several places in this paper refer to filtering of a signal x_c^o to produce a magnitude, rate, and bandwidth-limited signal x_c and its derivative $\dot{x}_c$. One such filter is shown in Fig. 2. The state-space representation is

$$\begin{bmatrix}\dot{q}_1(t)\\ \dot{q}_2(t)\end{bmatrix} = \begin{bmatrix} q_2 \\ 2\zeta\omega_n\left(S_R\left\{\frac{\omega_n^2}{2\zeta\omega_n}\left[S_M\left(x_c^o\right)-q_1\right]\right\}-q_2\right)\end{bmatrix} \tag{A1}$$

$$\begin{bmatrix}x_c\\ \dot{x}_c\end{bmatrix} = \begin{bmatrix}q_1\\ q_2\end{bmatrix} \tag{A2}$$

where $S_M(\,)$ and $S_R(\,)$ represent the magnitude and rate limit functions. The function

$$S_M(x) = \begin{cases} M & \text{if} \quad x \geq M \\ x & \text{if} \quad |x| < M \\ -M & \text{if} \quad x \leq -M \end{cases}$$

and S_R is defined similarly. Note that if x_c^o is bounded, then x_c and $\dot{x}_c$ are bounded and continuous. Note also that $\dot{x}_c$ is computed without differentiation. When the state must remain in some operating envelope defined by the magnitude limit M and rate limit R, the command filter ensures that the commanded trajectory and its derivative satisfy these same constraints.

If the only objective in the design of the command filter is to compute x_c and its derivative, then M and R are infinitely large, and the limiters do not need to be included in the filter implementation.

In the linear range of the functions S_M and S_R, the filter dynamics are

$$\begin{bmatrix}\dot{q}_1(t)\\ \dot{q}_2(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\omega_n^2 & -2\zeta\omega_n\end{bmatrix}\begin{bmatrix}q_1\\ q_2\end{bmatrix} + \begin{bmatrix}0\\ \omega_n^2\end{bmatrix}x_c^o \tag{A3}$$

$$\begin{bmatrix}x_c\\ \dot{x}_c\end{bmatrix} = \begin{bmatrix}q_1\\ q_2\end{bmatrix} \tag{A4}$$

with transfer function from the input to the first output defined as

$$\frac{X_c(s)}{X_c^o(s)} = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$$

When command limiting is not in effect, the error $x_c - x_c^o$ can be made arbitrarily small by selecting ω_n sufficiently larger than the bandwidth of the signal x_c^o. When command limiting is in effect, the error $x_c - x_c^o$ will be bounded because both x_c and x_c^o are bounded.

Appendix B: Aircraft Dynamics

This paper used the following representation of the dynamics of a fixed-wing aircraft[46,47]:

$$\dot{\chi} = (1/mV\cos\gamma)[D\sin\beta\cos\mu + Y\cos\beta\cos\mu + L\sin\mu + T(\sin\alpha\sin\mu - \cos\alpha\sin\beta\cos\mu)] \tag{B1a}$$

$$\dot{\gamma} = (1/mV)[-D\sin\beta\sin\mu - Y\cos\beta\sin\mu + L\cos\mu + T(\sin\alpha\cos\mu + \cos\alpha\sin\beta\sin\mu)] - (g/V)\cos\gamma \tag{B1b}$$

$$\dot{V} = (1/m)(T\cos\alpha\cos\beta - D\cos\beta + Y\sin\beta) - g\sin\gamma \tag{B1c}$$

$$\dot{\mu} = (1/mV)[D\sin\beta\tan\gamma\cos\mu + Y\cos\beta\tan\gamma\cos\mu + L(\tan\beta + \tan\gamma\sin\mu) + T(\sin\alpha\tan\gamma\sin\mu + \sin\alpha\tan\beta - \cos\alpha\sin\beta\tan\gamma\cos\mu)] - g\tan\beta\cos\gamma\cos\mu/V + P_s/\cos\beta \tag{B1d}$$

$$\dot{\alpha} = (-1/mV\cos\beta)[L + T\sin\alpha] + g\cos\gamma\cos\mu/V\cos\beta + Q - P_s\tan\beta \tag{B1e}$$

$$\dot{\beta} = (1/mV)[D\sin\beta + Y\cos\beta - T\cos\alpha\sin\beta] + g\cos\gamma\sin\mu/V - R_s \tag{B1f}$$

$$\dot{P} = (c_1R + c_2P)Q + c_3\bar{L} + c_4\bar{N} \tag{B1g}$$

$$\dot{Q} = c_5PR - c_6(P^2 - R^2) + c_7\bar{M} \tag{B1h}$$

$$\dot{R} = (c_8P - c_2R)Q + c_4\bar{L} + c_9\bar{N} \tag{B1i}$$

where the c_i coefficients for $i = 1, \ldots, 9$ are defined on p. 80 in Ref. 47 and

$$\begin{bmatrix} P_s \\ R_s \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} P \\ R \end{bmatrix} \tag{B2}$$

Tables C1, C2, and C3 at the end of this paper define the constants, variable, and functions used in the preceding equations.

Appendix C: Notation

Table C1 Constant definitions

Symbol	Meaning
m	Mass
g	Vertical gravity component
c_i	Rotational inertia parameters defined on p. 80 in Ref. 47
α_i	Weighting parameters in the Lyapunov analysis
K_*	Control gain for variable $*$
L_*	Estimator gain for variable $*$
Γ_*	Parameter adaptation matrix for function $*$
b	Reference wing span
$\bar{c}$	Mean aerodynamic chord

Table C2 Variable definitions

Variable	Definition
χ	Ground track angle
γ	Climb angle
V	Speed
μ	Roll angle
α	Angle of attack
β	Sideslip
P	Body-axis roll rate
Q	Body-axis pitch rate
R	Body-axis yaw rate
M	Mach number
P_s	Stability axis roll rate
R_s	Stability axis yaw rate
u_*	Actual control signal for variable $*$
$\hat{u}^o_{*_c}$	Commanded control signal for variable $*$ computed using $\hat{f}_*$
$\hat{u}_{*_c}$	Achievable commanded control signal for variable $*$ computed using $\hat{f}_*$
δ_i	Deflection of the ith control surface
T	Thrust
ζ_*	Filter control error for variable $*$
y_*	Portion of control signal for variable $*$ that depends on tracking errors from a more inner loop
x_*	Portion of control signal for variable $*$ that depends on tracking errors from a more outer loop

Table C3 Function definitions

Symbol	Definition
D	Body-axis drag force; This function is unknown
Y	Body-axis side force; This function is unknown
L	Body-axis lift force; This function is unknown
$\bar{L}$	Stability-axis roll moment; This function is unknown
$\bar{M}$	Stability-axis pitch moment; This function is unknown
$\bar{N}$	Stability-axis yaw moment; This function is unknown
$\hat{D}$	Approximated body-axis drag
$\hat{Y}$	Approximated body-axis side force
$\hat{L}$	Approximated body-axis lift
$\hat{\bar{L}}$	Approximated stability-axis roll moment
$\hat{\bar{M}}$	Approximated stability-axis pitch moment
$\hat{\bar{N}}$	Approximated stability-axis yaw moment
f_*	Portion of the dynamic equation for variable $*$ that contains at least one of the six unknown functions
F_*	Portion of the dynamic equation for variable $*$ that contains known functions
$\hat{f}_*$	Approximation to f_*

Acknowledgments

This research was supported by Barron Associates, Inc., through U.S. Air Force Contract F33615-01-C-3108, the National Science Foundation through Grant ECS-0322635, the University of California, Riverside, the University of Cyprus, and the University of Cincinnati. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the supporting institutions. The authors gratefully acknowledge the contributions of Nathan Richards for the implementation of the control algorithm software.

References

[1]Isidori, A., *Nonlinear Control Systems*, 3rd ed., Springer-Verlag, Berlin, 1995.

[2]Khalil, H., *Nonlinear Systems*, 2nd ed., Prentice–Hall, Upper Saddle River, NJ, 1996.

[3]Krstic, M., Kanellakopoulos, I., and Kokotovic, P., *Nonlinear and Adaptive Control Design*, Wiley, New York, 1995.

[4]Azam, M., and Singh, S. N., "Invertibility and Trajectory Control for Nonlinear Maneuvers of Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 1, 1994, pp. 192–200.

[5]Bugajski, D. J., Enns, D. F., and Elgersma, M. R., "A Dynamic Inversion Based Control Law with Application to High Angle of Attack Research Vehicle," AIAA Paper 90-3407, 1990, pp. 826–839.

[6]Lane, S. H., and Stengel, R. F., "Flight Control Design Using Nonlinear Inverse Dynamics," *Automatica*, Vol. 31, No. 4, 1988, pp. 781–806.

[7]Menon, P. K. A., Badget, M. E., Walker, R. A., and Duke, E. L., "Nonlinear Flight Test Trajectory Controllers for Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 1, 1987, pp. 67–72.

[8]Meyer, G., Su, R., and Hunt, L. R., "Application of Nonlinear Transformations to Automatic Flight Control," *Automatica*, Vol. 20, No. 1, 1984, pp. 103–107.

[9]Snell, S. A., Enns, D. F., and Garrard, W. L., "Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 4, 1992, pp. 976–984.

[10]Farrell, J. A., Polycarpou, M. M., and Sharma M., "Longitudinal Flight Path Control Using On-Line Function Approximation," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 6, 2003, pp. 885–897.

[11]Glad, S. T., and Härkegård, O., "Backstepping Control of a Rigid Body," *41st IEEE Conference on Decision and Control*, IEEE Publications, Piscataway, NJ, 2002, pp. 3944, 3945.

[12]Härkegård, O., and Glad, S. T., "A Backstepping Design for Flight Path Angle Control," *39th IEEE Conference on Decision and Control*, IEEE Publications, Piscataway, NJ, 2000, pp. 3570–3575.

[13]Härkegård, O., "Backstepping and Control Allocation with Applications to Flight Control," Ph.D. Dissertation 820, Linkoping Univ., 2003.

[14]Sharma, M., Farrell, J. A., Polycarpou, M. M., Richards, N. D., and Ward, D. G., "Backstepping Flight Control Using On-Line Function Approximation," AIAA Paper 2003-5713, 2003.

[15]Khan, M. A., and Lu, P., "New Technique for Nonlinear Control of Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 5, 1994, pp. 1055–1060.

[16]Gerrard, W. L., Enns, D. F., and Snell, A., "Nonlinear Longitudinal Control of a Supermaneuverable Aircraft," *1989 American Control Conference*, Vol. 1, 1989, pp. 142–147.

[17]Kanellakopoulos, I., Kokotovic, P., and Morse. A. S., "Systematic Design of Adaptive Controllers for Feedback Linearizable Systems," *IEEE Transactions on Automatic Control*, Vol. 36, No. 11, 1991, pp. 1241–1253.

[18]Barmish, B. R., Corless, M., and Leitmann, G., "A New Class of Stabilizing Controllers for Uncertain Dynamical Systems," *SIAM Journal on Control and Optimization*, Vol. 21, No. 2, 1983, pp. 246–255.

[19]Polycarpou, M. M., and Ioannou, P., "A Robust Adaptive Nonlinear Control Design," *Automatica*, Vol. 32, No. 3, 1996, pp. 423–427.

[20]Calise, A., and Rysdyk., R., "Nonlinear Adaptive Flight Control Using Neural Networks," *IEEE Control Systems Magazine*, Vol. 18, No. 6, 1998, pp. 14–25.

[21]Chen, F.-C., and Khalil, H. K., "Adaptive Control of a Class of Nonlinear Discrete-Time Systems Using Neural Networks," *IEEE Transactions on Automatic Control*, Vol. 40, No. 5, 1995, pp. 791–801.

[22]Choi, J. Y., and Farrell, J. A., "Nonlinear Adaptive Control Using Networks of Piecewise Linear Approximators," *IEEE Transactions on Neural Networks*, Vol. 11, No. 2, 2000, pp. 390–401.

[23]Farrell, J. A., "Persistence of Excitation Conditions in Passive Learning Control," *Automatica*, Vol. 33, No. 4, 1997, pp. 699–703.

[24]Farrell, J. A., "On Performance Evaluation in On-Line Approximation for Control," *IEEE Transactions on Neural Networks*, Vol. 9, No. 5, 1998, pp. 1001–1007.

[25]Farrell, J. A., "Stability and Approximator Convergence in Nonparametric Nonlinear Adaptive Control," *IEEE Transactions on Neural Networks*, Vol. 9, No. 5, 1998, pp. 1008–1020.

[26]Ge, S. S., Hang, C. C., Lee, T. H., and Zhang, T., *Stable Adaptive Neural Network Control*, Kluwer Academic, Boston, 2001.

[27]Kim, B., and Calise, A., "Nonlinear Flight Control Using Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 26–33.

[28]Polycarpou, M. M., "Stable Adaptive Neural Control Scheme for Nonlinear Systems," *IEEE Transactions on Automatic Control*, Vol. 41, No. 3, 1996, pp. 447–451.

[29]Polycarpou, M. M., and Mears, M. J., "Stable Adaptive Tracking of Uncertain Systems Using Nonlinearly Parameterized On-Line Approximators," *International Journal Control*, Vol. 70, No. 3, 1998, pp. 363–384.

[30]Sanner, R. M., and Slotine, J. E., "Gaussian Networks for Direct Adaptive Control," *IEEE Transactions on Neural Networks*, Vol. 3, No. 6, 1992, pp. 837–863.

[31]Yesildirek, A., and Lewis, F., "Feedback Linearization Using Neural Networks," *Automatica*, Vol. 31, No. 11, 1995, pp. 1659–1664.

[32]Annaswamy, A. M., and Wong, J. E., "Adaptive Control in the Presence of Saturation Nonlinearity," *International Journal of Adaptive Control and Signal Processing*, Vol. 11, No. 1, 1997, pp. 3–19.

[33]Farrell, J. A., Polycarpou, M. M., and Sharma, M., "Adaptive Backstepping with Magnitude, Rate, and Bandwidth Constraints: Aircraft Longitude Control," *American Control Conference*, 2003, pp. 3898–3903.

[34]Farrell, J. A., Polycarpou, M. M., and Sharma, M., "On-Line Approximation Based Control of Uncertain Nonlinear Systems with Magnitude, Rate and Bandwidth Constraints on the States and Actuators," *American Control Conference*, 2004, pp. 2557–2562.

[35]Johnson, E. N., and Calise, A. J., "Neural Network Adaptive Control Systems with Input Saturation," *American Control Conference*, 2001, pp. 3527–3532.

[36]Karason, S. P., and Annaswamy, A. M., "Adaptive Control in the Presence of Input Constraints," *IEEE Transactions on Automatic Control*, Vol. 39, No. 11, 1994, pp. 2325–2330.

[37]Monopoli, R., "Adaptive Control for Systems with Hard Saturation," *IEEE Conference on Decision and Control*, 1975, pp. 841–843.

[38]Wang, H., and Sun, J., "Modified Model Reference Adaptive Control with Saturated Inputs," *IEEE Conference on Decision and Control*, 1992, pp. 3255–3256.

[39]Polycarpou, M. M., Farrell, J. A., Sharma, M., "On-Line Approximation Control of Uncertain Nonlinear Systems: Issues with Control Input Saturation," *American Control Conference*, 2003, pp. 543–548.

[40]Polycarpou, M. M., Farrell, J. A., and M. Sharma, "Robust On-Line Approximation Control of Uncertain Nonlinear Systems Subject to Constraints," *9th IEEE International Conference on Engineering of Complex Computer Systems*, 2004, pp. 66–74.

[41]Trankle, T. L., Vincent, J. H., and Franklin, S. N., "System Identification of Nonlinear Aerodynamic Models," *Advances in the Techniques and Technology of the Application of Nonlinear Filters and Kalman Filters*, AGARDograph No. 256, March 1982.

[42]Batterson, J. G., and Klein, V., "Partitioning of Flight Data for Aerodynamic Modeling of Aircraft at High Angles of Attack," *Journal of Aircraft*, Vol. 26, No. 4, 1989, pp. 334–339.

[43]Iliff, K. W., "Parameter Estimation for Flight Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 12, No. 5, 1989, pp. 609–622.

[44]Trankle, T. L., and Bachner, S. D., "Identification of a Nonlinear Aerodynamic Model of the F-14 Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 6, 1995, pp. 1292–1297.

[45]Sharma, M., Calise, A. J., and Corban, J. E., "Application of an Adaptive Autopilot Design to a Family of Guided Munitions," AIAA Paper 2000-3969, 2000.

[46]"Recommended Practice for Atmospheric and Space Flight Vehicle Coordinate Systems," AIAA/ANSI R-004-1992, AIAA, Reston, VA, Feb. 1992.

[47]Stevens, B., and Lewis, F., *Aircraft Control and Simulation*, Wiley, New York, 1992.

[48]Bodson, M., "Evaluation of Optimization Methods for Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 703–711.

[49]Durham, W. C., "Computationally Efficient Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 3, 2001, pp. 519–524.

[50]Slotine, J.-J., and Li, W., *Applied Nonlinear Control*, Prentice–Hall, Englewood Cliffs, NJ, 1991.

[51]Fears, S. P., Ross, H. M., and Moul, T. M., "Low-Speed Wind-Tunnel Investigation of the Stability and Control Characteristics of a Series of Flying Wings with Sweep Angles of 50°," NASA TM 4640, June 1995.