Flight Test Results of Adaptive Controllers in Presence of Severe Structural Damage

Girish Chowdhary*, Eric N. Johnson†,
M. Scott Kimbrell ‡, Rajeev Chandramohan§, Anthony Calise ¶
Georgia Institute of Technology, Atlanta, GA, 30318

We present flight test results for adaptive controllers intended to mitigate significant aircraft faults. The adaptive controllers are tested in flight on the Georgia Tech GT Twinstar fixed wing twin engine aircraft with 25% left wing missing. A Model Reference Adaptive Control (MRAC) architecture employing a Neural Network (NN) as adaptive element is used for inner loop attitude control of the aircraft, and is intended to augment a state dependent outer loop guidance logic. Two adaptive control methods are tested. The first is a proven MRAC based method employing a single hidden layer NN. The second is the recently introduced Derivative Free MRAC (DFMRAC) method. The results establish the feasibility of these methods for ensuring safe autonomous flight in presence of severe structural faults.

I. Introduction

Fixed wing aircraft can be rendered inoperable by severe structural damage. Such damage may include battle damage, structural failure, damage due to terrorist attacks, and damage due to high speed impact. Lifting surface loss and control surface loss are examples of failures that make it extremely hard for pilots to maintain stable flight and ensure that the aircraft is capable of approaching a nearby runway and landing successfully. Under such circumstances, the use of proven autonomous control for maintaining stable flight and aiding the pilot in landing can be justified.

Commonly used closed loop autonomous control approaches for aircraft are based on assuming a linearized dynamical model for the dynamics and using proven baseline linear control methods to guarantee stability (see for example 7,24 and the references within). Under severe structural faults however, the dynamics of the aircraft are significantly altered. For example, under an asymmetric wing failure, the aircraft will experience asymmetric lateral forces coupled with longitudinal motion, these are not handled by the standard symmetric wing decoupled aircraft models. For maintaining successful autonomous control, these changes in dynamics, brought about by system faults, must be accounted for by the onboard controller. Controllers that can mitigate such changes automatically and maintain safe flight are known as Fault Tolerant Controllers.[28]

The field of adaptive control presents a rich selection of methods and architectures that can be used to handle the dynamic modeling uncertainties introduced due to significant aircraft faults, and has been proposed as a preferred solution to the problem of Aircraft Fault Tolerant Control.[18,19,23] The purpose of this paper is to present flight test results of adaptive control methods aimed at ensuring safe landing of aircraft under severe structural faults. We use Model Reference Adaptive Control Architecture[10,22] for adaptive inner loop attitude control which augments state dependent linear outer loop Guidance logic. The state dependent control logic uses the knowledge of aircraft air speed to change linear gains accordingly and serves as a baseline for comparing adaptive control performance.[12] The MRAC adaptive control is implemented through approximate model inversion and uses a Neural Network (NN) as the adaptive element.[8,11,21] We present flight test results for MRAC with a proven single hidden layer NN adaptive law,[13,15] and the recently

*Research Assistant, School of Aerospace Engineering, AIAA student member
†Lockheed Martin Professor of Avionics Integration, School of Aerospace Engineering, AIAA member
‡Graduate, School of Aerospace Engineering, AIAA student member
§Research Assistant, School of Aerospace Engineering, AIAA student member
¶Professor, School of Aerospace Engineering, AIAA member

introduced Derivative Free MRAC.[27] These adaptive controllers are tested on the GT Twinstar UAS with 25% of the left wing missing.

II. Model Reference Adaptive Control

This section discusses the formulation of Model Reference Adaptive Control using approximate model inversion. Let $D_x \subset \Re^n$ be compact, and Let $x(t) \in D_x$ be the known state vector, let $\delta \in \Re^k$ denote the control input, and consider the following system that describes the dynamics of an aircraft with damage:

$$\dot{x} = f(x(t), \delta(t)), \tag{1}$$

In the above equation, the function f is assumed to be continuously differentiable in $x \in D_x$, and control input δ is assumed to be bounded and piecewise continuous. The conditions for the existence and the uniqueness of piecewise solutions to 1 are assumed to be met.

Since the exact model 1 is usually not available or not invertible, we choose an approximate inversion model $\hat{f}(x, \delta)$ following the approach of Approximate Model Reference Adaptive Control which can be inverted to determine the control input δ:[11,21]

$$\delta = \hat{f}^{-1}(x, \nu). \tag{2}$$

where $\nu \in \Re^n$ is the pseudo control input, which represents the desired model output $\dot{x}$ and is expected to be approximately achieved by δ. It is assumed that for every pair (x, ν) the chosen inversion model returns a unique δ. Hence, the pseudo control input is the output of the approximate inversion model:

$$\nu = \hat{f}(x, \delta). \tag{3}$$

This approximation results in a model error of the form:

$$\dot{x} = \nu + \Delta(x, \delta) \tag{4}$$

where the model error $\Delta : \Re^{n+k} \to \Re^n$ is given by:

$$\Delta(x, \delta) = f(x, \delta) - \hat{f}(x, \delta). \tag{5}$$

A reference model can be designed that characterizes the desired response of the system:

$$\dot{x}_{rm} = f_{rm}(x_{rm}, r(t)), \tag{6}$$

where $f_{rm}(x_{rm}(t), r(t))$ denote the reference model dynamics which are assumed to be continuously differentiable in x for all $x \in D_x \subset \Re^n$. The command $r(t)$ is assumed to be bounded and piecewise continuous, furthermore, it is assumed that all requirements for guaranteeing the existence of a unique solution to 6 are satisfied. Often a linear reference model is desirable, however the theory allows for a general nonlinear reference model as long as it is bounded-input-bounded-output stable. A tracking control law consisting of a linear feedback part $u_{pd} = Kx$, a linear feedforward part $u_{crm} = \dot{x}_{rm}$, and an adaptive part $u_{ad}(x)$ is proposed to have the following form:

$$u = u_{crm} + u_{pd} - u_{ad}. \tag{7}$$

Define the tracking error e as $e(t) = x_{rm}(t) - x(t)$, then, letting $A = -K$ the tracking error dynamics are found to be:[2,4,11,15]

$$\dot{e} = Ae + [u_{ad}(x, \delta) - \Delta(x, \delta)]. \tag{8}$$

The baseline full state feedback controller $u_{pd} = Kx$ is assumed to be designed such that A is a Hurwitz matrix. Hence for any positive definite matrix $Q \in \Re^{n \times n}$, a positive definite solution $P \in \Re^{n \times n}$ exists to the Lyapunov equation:

$$A^T P + PA + Q = 0. \tag{9}$$

Letting $\bar{x} = [x, \delta] \in \Re^{n+k}$, it is assumed that the uncertainty $\Delta(\bar{x})$ can be approximated using a Neural Network (NN). In this paper we present flight test results from adaptive controllers which are derived from the MRAC framework. These adaptive controllers differ mainly in the types of NN they use and the learning laws for the NN.

A. Model Reference Adaptive Control using a Single Hidden Layer Neural Network

A Single Hidden Layer (SHL) NN is a nonlinearly parameterized map that has also been often used for capturing unstructured uncertainties that are known to be continuous and defined over a compact domain. The output of a SHL NN can be given as:

$$u_{ad}(x) = W^T \sigma(V^T \bar{x}) \in \Re^{n_3}. \tag{10}$$

In the above equation, $W \in \Re^{(n_2+1)\times n_3}$, where n_2 is the number of hidden layer neurons and n_3 is the dimension of the NN output, is the NN synaptic weight matrix connecting the hidden layer with the output layer and has the following form:

$$W = \begin{pmatrix} \Theta_{w,1} & \cdots & \Theta_{w,n_3} \\ w_{1,1} & \cdots & w_{1,n_3} \\ \vdots & \ddots & \vdots \\ w_{n_2,1} & \cdots & w_{n_2,n_3} \end{pmatrix} \in \Re^{(n_2+1)\times n_3}, \tag{11}$$

where $V \in \Re^{(n_1+1)\times n_2}$ is the NN synaptic weight matrix, with n_1 being the number of inputs to the NN, connecting the input layer with the hidden layer and has the following form:

$$V = \begin{pmatrix} \Theta_{v,1} & \cdots & \Theta_{v,n_2} \\ v_{1,1} & \cdots & v_{1,n_2} \\ \vdots & \ddots & \vdots \\ v_{n_1,1} & \cdots & w_{n_1,n_2} \end{pmatrix} \in \Re^{(n_1+1)\times n_2}, \tag{12}$$

$\bar{x} \in D \subset \Re^{n_1+1}$, where D is a compact set, is the NN input which contains the data on which the network trains x_{in} and the constant bias term b_v:

$$\bar{x} = \begin{pmatrix} b_v \\ x_{in} \end{pmatrix} = \begin{pmatrix} b_v \\ x_{in_1} \\ x_{in_2} \\ \vdots \\ x_{in_{n_1}} \end{pmatrix} \in \Re^{n_1+1}. \tag{13}$$

For ease in notation, let $z = V^T \bar{x} \in \Re_2^n$, then the vector function $\sigma(z) \in \Re^{n_2+1}$ is given by:

$$\sigma(z) = \begin{pmatrix} b_w \\ \sigma_1(z_1) \\ \vdots \\ \sigma_{n_2}(z_{n_2}) \end{pmatrix} \in \Re^{n_2+1}. \tag{14}$$

The elements of σ consist of sigmoidal activation functions, which are given by:

$$\sigma_j(z_j) = \frac{1}{1+e^{-a_j z_j}}. \tag{15}$$

Single Hidden Layer (SHL) perceptron NN are known to be universal approximators (see 9 or 25). That is, given an $\bar{\epsilon} > 0$, for all $\bar{x} \in D$, where D is a compact set, there exists a number of hidden layer neurons n_2, and an ideal set of weights (W^*, V^*) that brings the NN output to within an ϵ neighborhood of the function approximation error. The largest such ϵ is given by:

$$\bar{\epsilon} = \sup_{\bar{x}\in D} \left\| {W^*}^T \sigma({V^*}^T \bar{x}) - \Delta(\bar{x}) \right\|. \tag{16}$$

Hence in a similar fashion to RBF NN we have that the following approximation holds for all $x \in D \subset \Re^n$ where D is compact:

$$\Delta(x) = {W^*}^T \sigma({V^*}^T \bar{x}) + \tilde{\epsilon}(x), \tag{17}$$

and $\bar{\epsilon} = \sup_{\bar{x}\in D} \|\tilde{\epsilon}(x)\|$ can be made arbitrarily small given sufficient number of hidden layer neurons.

For this case it has been shown that the following adaptive laws guarantee guarantees uniform ultimate boundedness of the tracking error and the adaptive weights around the origin, and guarantees that the adaptive weights stay bounded within a neighborhood of the ideal weights if the system states are persistently exciting (see for example 17, 20 and the references therein):

$$\dot{W} = -(\sigma(V^T\bar{x}(t)) - \sigma'(V^T\bar{x}(t))V^T\bar{x})r(t)^T\Gamma_w - k\|e\|W \quad (18)$$

$$\dot{V} = -\Gamma_V\bar{x}(t)r(t)^TW^T\sigma'(V^T\bar{x}(t)) - k\|e\|V. \quad (19)$$

Figure 1 depicts the control architecture for MRAC control discussed in this section. Further details on the MRAC implementation used in this paper can be found in references [2, 11, 15, 16, 20, 26].

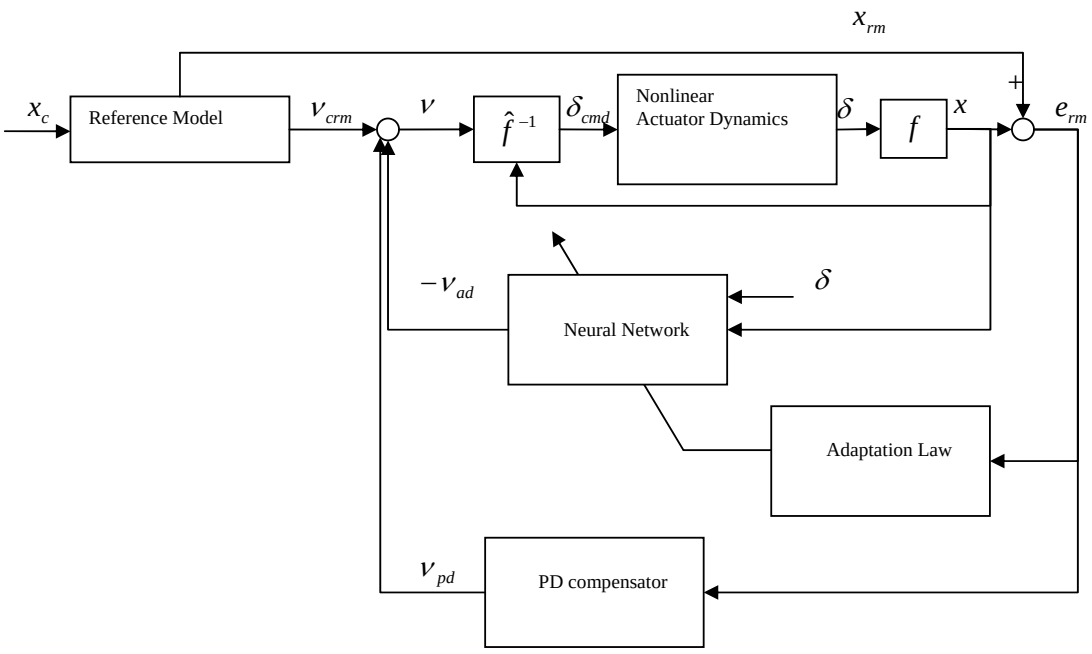


Figure 1. Neural Network Adaptive Control using Approximate Model Inversion

B. Derivative Free Model Reference Adaptive Control

In Derivative Free Model Reference Adaptive Control (DFMRAC) it is assumed that the uncertainty can be linearly parameterized.[27] That is, it is assumed that:

$$\Delta(x(t)) = W^{*^T}(t)\sigma(x(t)) + \epsilon(x(t)). \quad (20)$$

For the results presented in this paper, a linear in the parameter neural network with sigmoidal basis function was used. Hence, $\sigma(x(t)) \in \Re^l$ is such that $\sigma(x(t)) = [\sigma_1(x(t)), \sigma_2(x(t)), ..., \sigma_l(x(t))]$, with $\sigma_j = \frac{1}{1+e^{-a_j z_j}}$. An interesting feature of DFMRAC is that the ideal weights $W^*(t)$ can be time varying. In DFMRAC, the adaptive law is given by:[27]

$$W(t) = \Omega_1 W(t-\tau) + \Omega_2(t). \quad (21)$$

In the above equation τ is selected time delay value, and Ω_1 and $\Omega_2(t)$ satisfy

$$0 < \Omega_1^T \Omega_1 < \kappa_1 I \tag{22}$$

and

$$\Omega_2(t) = \kappa_2[\sigma(x(t))e^T(t)P - \kappa_m S(t)W(t)] \tag{23}$$

where $0 < \kappa_1 < 1$, $\kappa_2 > 0$, $\kappa_m > 0$, and $S(t) \in \Re^{s \times s}$ satisfies $||S(t)|| < s^*$. Further information on DFMRAC can be found in references [1, 27].

III. Flight Test Vehicle

Flight testing of adaptive control algorithms for safe flight in presence of severe structural faults is currently underway at the UAV Research Facility at Georgia Institute of Technology. These flight tests are performed on the GT Twinstar fixed wing Unmanned Aerial System (UAS). The GT Twinstar (Figure 2) is a foam built, twin engine aircraft that has been equipped with the Adaptive Flight Inc. (AFI, www.adaptiveflight.com) FCS 20®. The FCS 20 embedded autopilot system comes with an integrated navigation solution that fuses information using an extended Kalman filter from six degree of freedom inertial measurement sensors, Global Positioning System, air data sensor, and magnetometer to provide accurate state information.[5] The available state information includes velocity and position in global and body reference frames, accelerations along the body x, y, z axes, roll, pitch, yaw rates and attitude, barometric altitude, and air speed information. These measurements can be further used to determine the aircraft's velocity with respect to the air mass, and the flight path angle. The Twinstar can communicate with a Ground Control Station (GCS) using a 900 MHz wireless data link. The GCS serves to display onboard information as well as send commands to the FCS20. Flight measurements of airspeed and throttle setting are used to estimate thrust with this model. An elaborate simulation environment has also been designed for the GT Twinstar. This environment is based on the Georgia Tech UAS Simulation Tool (GUST) environment.[14] A linear model for the Twinstar in nominal configuration (without damage) has been identified using the FTR method by the authors.[6] A linear model with 25% left wing missing has also been identified.[3]

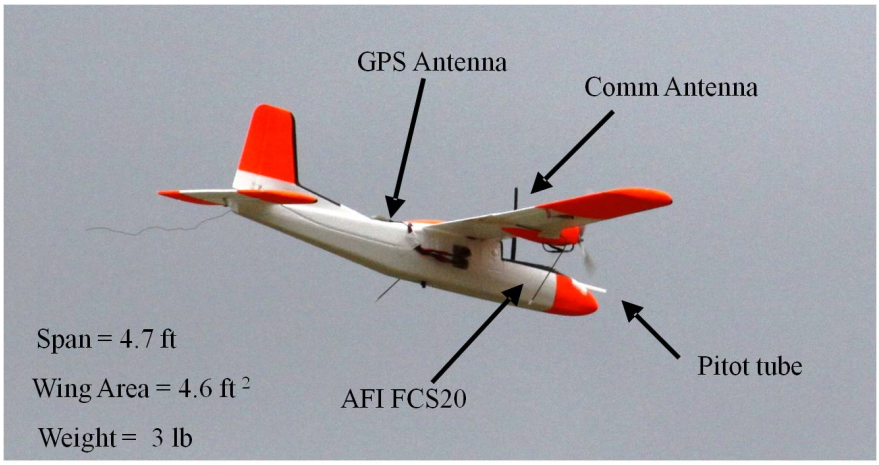


Figure 2. The Georgia Tech Twinstar UAS. The GT Twinstar is a fixed wing foam-built UAS designed for fault tolerant control work.

IV. Flight Test Results

Results from a flight tests are presented as the aircraft tracks an elliptical pattern while holding altitude at 200 ft with 25% of the left wing missing (shown in figure **??**. The guidance algorithm for GT Twinstar is designed to ensure that the aircraft can track feasible trajectories even when it has undergone severe structural damage.[12] The control algorithm has a cascaded inner and outer loop design. The outerloop, which is integrated with the guidance loop, commands the desired roll angle (ϕ), angle of attack (α), and sideslip

angle (β) to achieve desired waypoints. The innerloop ensures that the states of the aircraft track these desired quantities using the control architectures described in section II. The SHL MRAC implementation has 5 hidden layer neurons, while the LIP NN used for the DFMRAC implementation also uses 5 neurons in the input layer. The learning rates in equation 18 are set as $\Gamma_w = 1$ and $\Gamma_V = 5$ for the SHL MRAC implementation while the e-modification constant κ is set to 0.1. This modification is not required for the DFMRAC implementation.[27] The time delay τ is set to 1 time step (0.01 seconds) for the DFMRAC implementation and the training constants were set to $\kappa_1 = 0.1, \kappa_2 = 1$.

Figure 3. The Georgia Tech Twinstar UAS in autonomous flight with 25% of the left wing missing.

A. Flight Test Results Without Damage

In this section flight test results are presented as the aircraft operating in nominal conditions tracks four waypoints arranged in a rectangular pattern. The ground tracks of both controllers are shown in figure 4. In that figure, the circles denote the commanded way points, the dotted line connecting the circles denotes the path the aircraft is expected to take, except while turning at the way points. While turning at the way points, the onboard guidance law smooths the trajectorywith circles of 80 feet radius.[12] Figure 5 shows that the altitude tracking performance of the two controllers. The inner loop tracking error performance of the SHL MRAC adaptive controller is shown in figure 6(a), while the inner loop tracking error performance of the DFMRAC controller is shown in figure 6(b). The actuator input required for the SHL MRAC adaptive controller is shown in figure 7(a), while the actuator input required for the DFMRAC adaptive controller is shown in figure 7(b). The inner loop tracking performance and general inner loop handling was found satisfactory for both adaptive controllers. We noted that the steady state error estimation properties of the DFMRAC controller can be improved by selection of different design parameters. The inclusion of more neurons should improve the trim estimation properties of the linear in the parameter NN used for DFMRAC implementation, thereby improving its cross-tracking and altitude tracking performance.

B. Flight Test Results With Damage

In this section flight test results are presented as the aircraft tracks four waypoints arranged in a rectangular pattern with 25% of the left wing missing. The guidance and control laws remain unaltered in both nominal flight and damaged flight, with the adaptive controller expected to mitigate the effects of the enforced damage. Figure 9 shows that the altitude tracking performance of the two controllers. The inner loop tracking error performance of the SHL MRAC controller is shown in figure 10(a), while the inner loop

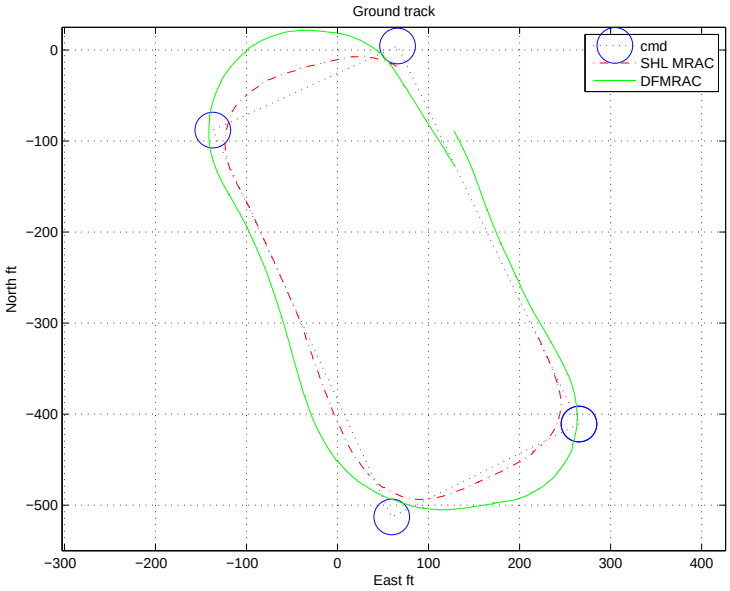


Figure 4. Flight recorded ground track for SHL MRAC and DFMRAC under nominal conditions on the GT Twinstar UAS.

tracking error performance of DFMRAC controller is shown in figure 10(b). The actuator input required for the SHL MRAC controller is shown in figure 11(a), while the actuator input required for the DFMRAC controller is shown in figure 11(b). The inner loop tracking performance and general inner loop handling of the damaged aircraft was found satisfactory for both adaptive controllers. The cross-tracking performance of the SHL MRAC controller when flying with damage remained comparable to controller performance under nominal conditions. We noted that the steady state error estimation properties of the DFMRAC controller can be improved by selection of different design parameters.

V. Conclusion

We presented autonomous flight test results of two adaptive controllers on the GT Twinstar UAS flown with 25% of its left wing missing. These results establish the feasibility of using adaptive controllers for ensuring safe flight of aircraft with severe structural damage. A Model Reference Adaptive Controller (MRAC) with a Single Hidden Layer (SHL) Neural Network (NN) was the first adaptive controller tested in autonomous flight with structural damage. A Derivative Free Model Reference Adaptive Controller (DFMRAC) with a linear in the parameter NN was the second adaptive controller tested in autonomous flight with structural damage. Both controllers were able to sustain autonomous flight with structural damage. The performance of the MRAC with SHL adaptive element was found satisfactory. We believe that the performance of the DFMRAC adaptive controller can be improved by using different design parameters to ensure that the linear in the parameter NN used in the DFMRAC implementation better estimates the steady state trim. Ongoing work includes testing of these and other adaptive flight controllers on the GT Twinstar UAS for various fault cases.

Acknowledgments

This work was supported in part by NSF ECS-0238993 and NASA Cooperative Agreement NNX08AD06A.

References

[1]Rajeev Chandramohan, Tansel Yucelen, Anthony Calise, Girish Chowdhary, and Eric Johnson. Experimental evaluation of derivative free model reference adaptive control. In *AIAA Guidance, Control and Navigation Conference*, Toronto, Canada,

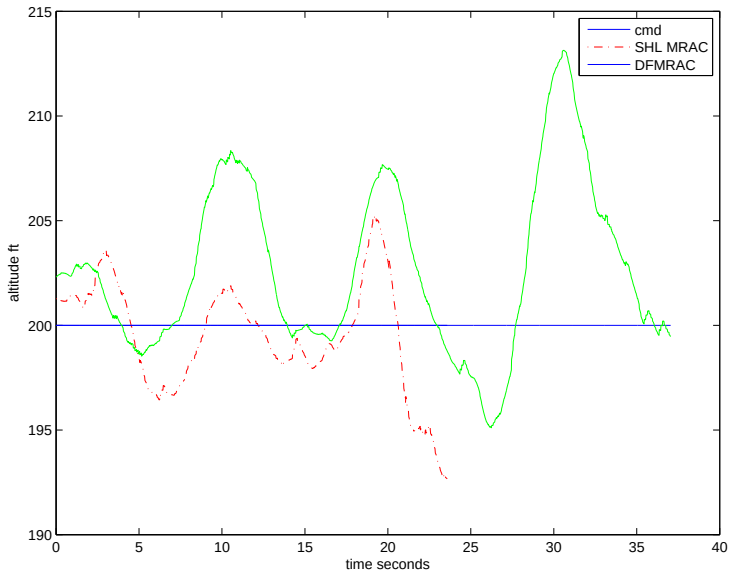


Figure 5. Flight recorded altitude tracking performance for SHL MRAC and DFMRAC under nominal conditions on the GT Twinstar UAS. The commanded altitude is at 200 ft.

August 2010.

[2]Girish Chowdhary. *Concurrent Learning for Convergence in Adaptive Control Without Persistentcy of Excitation*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, 2010. to be submitted in August 2010.

[3]Girish Chowdhary, Wesley Debusk, and Eric Johnson. Real-time system identification of a small multi-engine aircraft with structural damage. In *AIAA Infotech@Aerospace*, 2010.

[4]Girish Chowdhary and Eric Johnson. Concurrent learning for improved parameter convergence in adaptive control. In *AIAA Guidance Navigation and Control Conference*, Toronto, Canada, 2010.

[5]Henrik B. Christophersen, Wayne R. Pickell, James C. Neidoefer, Adrian A. Koller, Suresh K. Kannan, and Eric N. Johnson. A compact guidance, navigation, and control system for unmanned aerial vehicles. *AIAA Journal of Aerospace Computing, Information, and Communication*, 3, May 2006.

[6]Wesely Debusk, Girish Chowdhary, and Johnson Eric. Real-time system identification of a small multi-engine aircraft. In *Proceedings of AIAA AFM*, 2009.

[7]B. Etkin and Reid L. D. *Dynamics of Flight, Stability and Control*. John Wiley and Sons, 1996.

[8]T. Hayakawa, W.M. Haddad, and N. Hovakimyan. Neural network adaptive control for a class of nonlinear uncertain dynamical systems with asymptotic stability guarantees. *IEEE Transactions on Neural Networks*, 19(1):80 –89, jan. 2008.

[9]K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[10]Petros A. Ioannou and Peter V. Kokotovic. *Adaptive Systems with Reduced Models*. Springer Verlag, Secaucus, NJ, 1983.

[11]E. N. Johnson. *Limited Authority Adaptive Flight Control*. PhD thesis, Georgia Institute of Technology, Atlanta Ga, 2000.

[12]Eric Johnson and Girish Chowdhary. Guidance and control of an airplane under severe structural damage. In *AIAA Infotech@Aerospace*, 2010. *Invited*.

[13]Eric Johnson and Suresh Kannan. Adaptive trajectory control for autonomous helicopters. *Journal of Guidance Control and Dynamics*, 28(3):524–538, May 2005.

[14]Eric N. Johnson and Daniel P. Schrage. System integration and operation of a research unmanned aerial vehicle. *AIAA Journal of Aerospace Computing, Information and Communication*, 1(1):5–18, Jan 2004.

[15]Suresh K. Kannan. *Adaptive Control of Systems in Cascade with Saturation*. PhD thesis, Georgia Institute of Technology, Atlanta Ga, 2005.

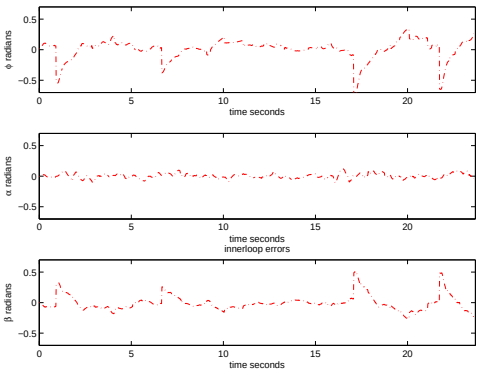
[16]Nakawan Kim. *Improved Methods in Neural Network Based Adaptive Output Feedback Control, with Applications to Flight Control*. PhD thesis, Georgia Institute of Technology, Atlanta Ga, 2003.

[17]Y. H. Kim and F.L. Lewis. *High-Level Feedback Control with Neural Networks*, volume 21 of *Robotics and Intelligent Systems*. World Scientific, Singapore, 1998.

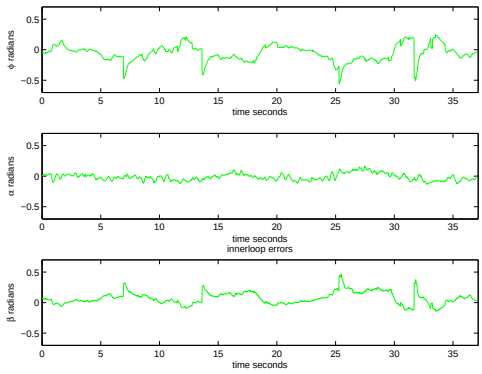
[18]Ali Kutay, Girish V. Chowdhary, A. Calise, and Eric N. Johnson. A comparision of two novel direct adaptive control methods under actuator failure accommodation. In *Proceedings of the AIAA GNC Conference, held at Honolulu, HI*, 2008.

[19]Eugene Lavertsky and Kevin Wise. Flight control of manned/unmanned military aircraft. In *Proceedings of American Control Conference*, 2005.

[20]F. L. Lewis. Nonlinear network structures for feedback control. *Asian Journal of Control*, 1:205–228, 1999. Special Issue on Neural Networks for Feedback Control.

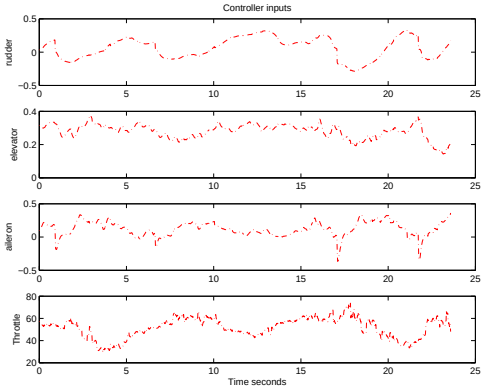


(a) Inner loop tracking errors for SHL MRAC

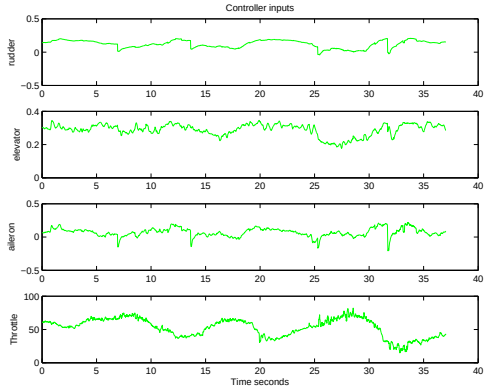


(b) Inner loop tracking errors for DFMRAC

Figure 6. Flight recorded inner loop tracking errors for SHL MRAC and DFMRAC under nominal conditions on the GT Twinstar UAS.



(a) Actuator inputs for SHL MRAC



(b) Actuator inputs for DFMRAC

Figure 7. Flight recorded actuator inputs for SHL MRAC and DFMRAC under nominal conditions on the GT Twinstar UAS. Note that the aileron, rudder, and elevator inputs are normalized between -1 and 1, while the throttle input is given as percentage.

[21]Flavio Nardi. *Neural Network based Adaptive Algorithms for Nonlinear Control*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, Atlanta, GA 30332, nov 2000.

[22]Kumpati S. Narendra and Anuradha M. Annaswamy. *Stable Adaptive Systems*. Prentice-Hall, Englewood Cliffs, 1989.

[23]M. Steinberg. Historical overview of research in reconfigurable flight control. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 219(4):263–275, 2005.

[24]Brian L. Stevens and Frank L. Lewis. *Aircraft Control and Simulation*. John Wiley and Sons, 2003.

[25]Johan A.K. Suykens, Joos P.L. Vandewalle, and Bart L.R. De Moor. *Artificial Neural Networks for Modelling and Control of Non-Linear Systems*. Kluwer, Norwell, 1996.

[26]Gang Tao. *Adaptive Control Design and Analysis*. Wiley, New York, 2003.

[27]Tansel Yucelen and Anthony Calise. A derivative-free model reference adaptive controller for the generic transport model. In *AIAA Guidance, Control and Navigation Conference*, Toronto, Canada, August 2010. invited.

[28]Youmin Zhang and Jin Jang. Bibliographical review on reconfigurable fault-tolerant control systems. *Elsevier Annual Reviews in Control*, 32(1):229–252, 2008.

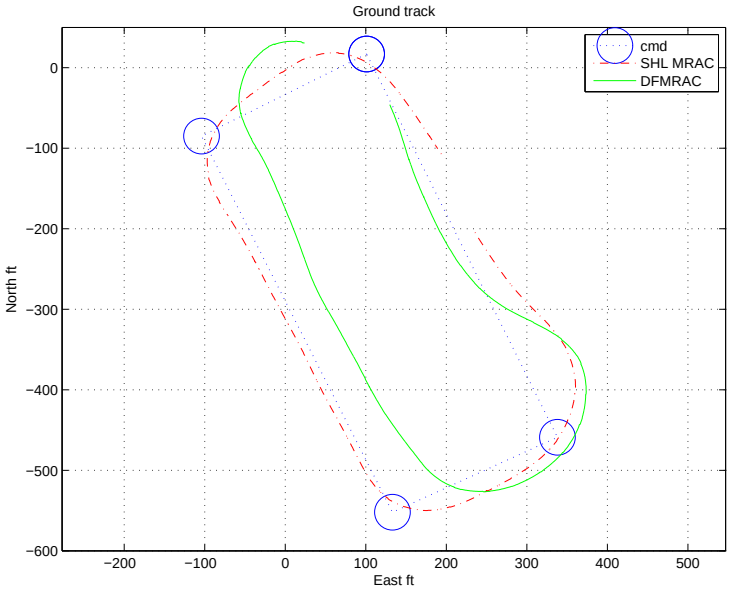


Figure 8. Flight recorded ground track for SHL MRAC and DFMRAC with 25% left wing missing on the GT Twinstar UAS.

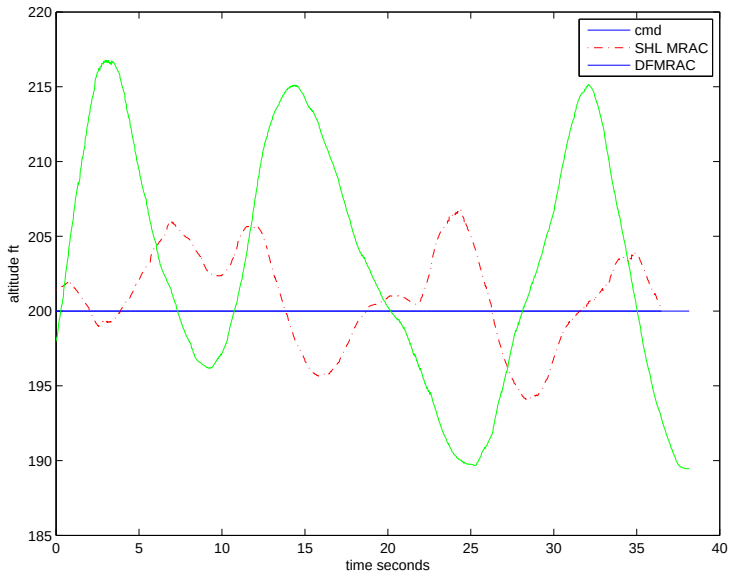
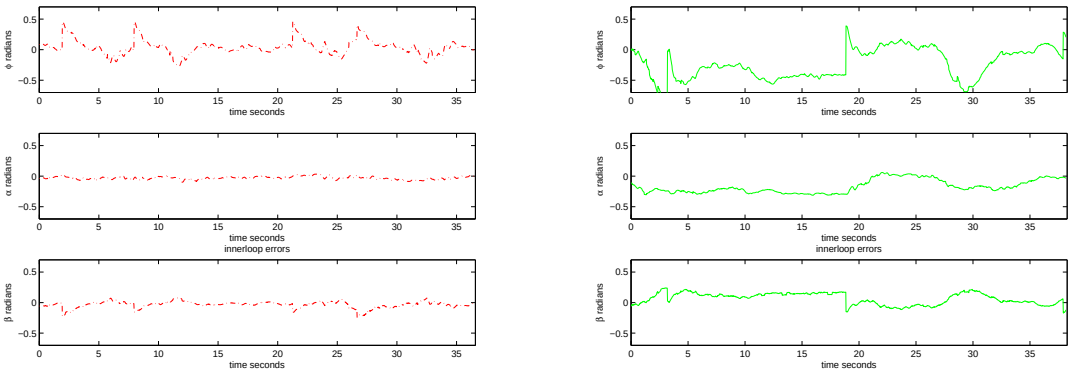


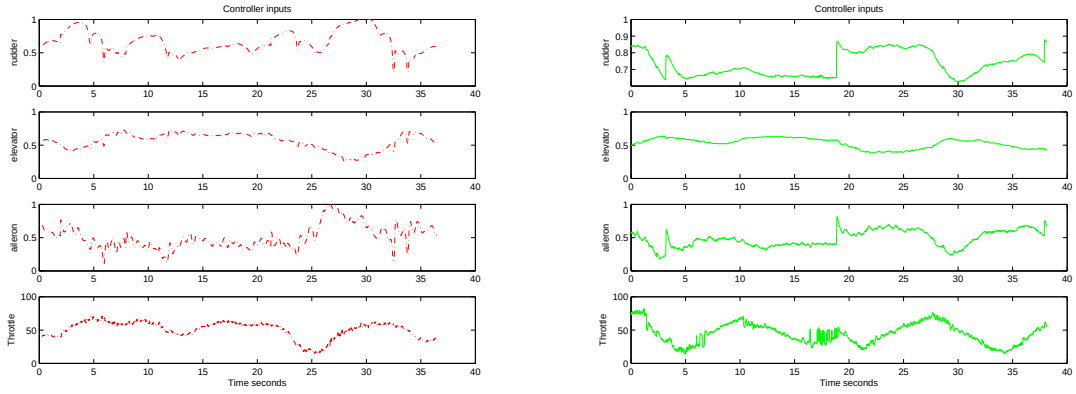
Figure 9. Flight recorded altitude tracking performance for SHL MRAC and DFMRAC with 25% left wing missing on the GT Twinstar UAS. The commanded altitude is at 200 ft.



(a) Inner loop tracking errors for SHL MRAC

(b) Inner loop tracking errors for DFMRAC

Figure 10. Flight recorded inner loop tracking errors for SHL MRAC and DFMRAC with 25% left wing missing on the GT Twinstar UAS.



(a) Actuator inputs for SHL MRAC

(b) Actuator inputs for DFMRAC

Figure 11. Flight recorded actuator inputs for SHL MRAC and DFMRAC with 25% left wing missing on the GT Twinstar UAS. Note that the aileron, rudder, and elevator inputs are normalized between -1 and 1, while the throttle input is given as percentage.