



Missouri University of Science and Technology
Scholars' Mine

Mechanical and Aerospace Engineering Faculty
Research & Creative Works

Mechanical and Aerospace Engineering

01 Jan 2006

Nonlinear Control Concepts for a UA

Vijayakumar Janardhan

Derek Schmitz

S. N. Balakrishnan

Missouri University of Science and Technology, bala@mst.edu

Follow this and additional works at: https://scholarsmine.mst.edu/mec_aereng_facwork

 Part of the [Aerospace Engineering Commons](#), and the [Mechanical Engineering Commons](#)

Recommended Citation

V. Janardhan et al., "Nonlinear Control Concepts for a UA," *IEEE Aerospace and Electronic Systems Magazine*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2006.

This Article - Journal is brought to you for free and open access by Scholars' Mine. It has been accepted for inclusion in Mechanical and Aerospace Engineering Faculty Research & Creative Works by an authorized administrator of Scholars' Mine. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

DEVELOPMENT AND IMPLEMENTATION OF NEW NONLINEAR CONTROL CONCEPTS FOR A UA

Vijayakumar Janardhan, Derek Schmitz, and S. N. Balakrishnan

University of Missouri – Rolla, Rolla, MO - 65409

Abstract

A reconfigurable flight control method is developed to be implemented on an Unmanned Aircraft (UA), a thirty percent scale model of the Cessna 150. This paper presents the details of the UAV platform, system identification, reconfigurable controller design, development, and implementation on the UA to analyze the performance metrics. A Crossbow Inertial Measurement Unit provides the roll, pitch and yaw accelerations and rates along with the roll and pitch. The 100400 mini-air data boom from SpaceAge Control provides the airspeed, altitude, angle of attack and the side slip angles. System identification is accomplished by commanding preprogrammed inputs to the control surfaces and correlating the corresponding variations at the outputs. A Single Network Adaptive Critic, which is a neural network based optimal controller, is developed as part of a nonlinear flight control system. An online learning neural network is augmented to form an outer loop to reconfigure and supplement the optimal controller to guarantee a "practical stability" for the airplane. This paper also presents some simulations from the hardware-in-the-loop testing and concludes with an analysis of the flight performance metrics for the controller under investigation.

Introduction

With the advancement of electronic technologies along with modern control theory, totally autonomous unmanned aircraft (UA) have taken to the sky. Unmanned aircraft can be used for objectives such as data link stations, weather observers, and reconnaissance and attack platforms. Perhaps in the future UA's will bear the responsibility of trafficking passengers around the world.

As costs rise for defense budgeting, UA's appear to be a simple way to spend less and get the same accomplishment from a manned vehicle

without the risk of a human life. In the battlefield or in the cockpit of a passenger airplane, the adaptability of the controller to unknown or unpredicted scenarios is key for mission success. It is this necessity that pushes control experts to develop not only autonomous flight controllers, but reconfigurable flight controllers. Reconfigurable controllers have the ability to adapt to situations that they were not explicitly designed for as in actuator, structural, or engine failures. This increases the survivability of the combat UA with battle damage as well as increases the safety of a passenger aircraft.

Research in the area of reconfigurable control via neural networks has been undertaken by many. Reference model adaptation [1] showed the ability to match the reference model to an actual aircraft in the event of damage. Further along these lines, this reference model adaptation was incorporated into a neural flight control system that combined dynamic inversion control techniques with direct adaptive control from pre-trained and online neural networks [2]. Calise and Rysdyk also show the applicability of supplementing dynamic inversion control with a neural network to achieve constant handling characteristics and consistent aircraft response during flight [3].

The objective of this project is to successfully implement a reconfigurable control system for autonomous control of a 30% scale model of a Cessna 150. This paper covers the current progress of the project and its future objectives. The aircraft will rely on feedback information from a gyroscope and an air data boom mounted on the aircraft. Along with the feedback sensors, a microcontroller and a radio modem are also installed on the aircraft to act as the airplane controller and to send information back to a ground station. After proper parameter estimation of the aircraft system has been accomplished, a modified dynamic inversion controller based on a design from our group [4] will be implemented on the aircraft to validate the

control hardware. Next, a more sophisticated optimal control based neural network controller design of our group [5] will be implemented to test its performance under mildly stressed conditions. Furthermore, analytical formulations underway will be implemented in an outer loop to the basic controller structure to test the abilities of the reconfigurable controller in highly stressed conditions such as non-operative actuators. Note that all these tests will be conducted in an autonomous mode.

The Autonomous UAV

Because the Cessna 150 is a stable airplane, a 30% scale model of the same is used for implementation (Figure 1). Although aerodynamics are not scalable, a similar layout was thought to offer similar stability characteristics. The aircraft has a 10 foot wingspan, weighs 35 lbs, and utilizes a Moki 2.1 in³ engine for power. The size of the aircraft allows ample space in the “cockpit” area for the onboard telemetry and control equipment. Ailerons, elevators, a rudder, and retractable flaps provide the control surfaces for the airplane. The control surfaces along with a throttle control provide the inputs to the test vehicle. Inputs are actuated by commercially available digital servos and the position information from the servos is fed to the data acquisition system onboard the airplane. The inputs to the servos will be able to be switched between the microcontroller and the RC pilot commands.

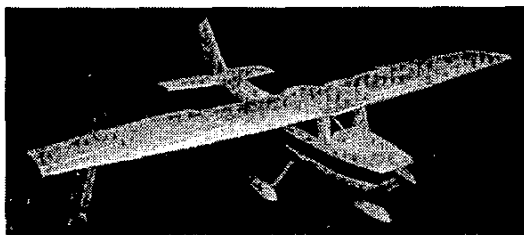


Figure 1. 30% Scale Cessna 150

The onboard data acquisition and control (ODAC) system is comprised of a PC-104 486 DX4 at 100MHz with 32 MB of RAM and 32 MB of Flash RAM. The ODAC runs MSDOS and has 16 12-bit analog inputs, four serial ports, Ethernet and parallel port, four 12-bit analog outputs and eight servo control ports. The ODAC communicates with

the base station through a 115.2 Kbps RS232 radio modem from Cirronet, Inc. The functional diagram of the ODAC is shown in Figure 2.

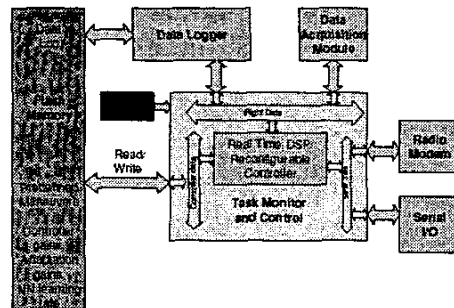


Figure 2. Functional Diagram of ODAC

Roll, pitch, and yaw rates, the roll and pitch angles, and axial, normal, and lateral accelerations are provided by an Inertial Measurement Unit (IMU) VG-400CA from Crossbow. The IMU is unable to interpret a yaw angle because magnetic north, a reference for yaw angle, is unavailable to the IMU. Airspeed, altitude, angle of attack, and side slip angle are provided by the 100400 mini-air data boom (MADM) from SpaceAge Control. Honeywell precision pressure transducers (PPT), are connected to the MADM pressure ports to determine the altitude and the airspeed.

Passive vibration isolation is provided for the ODAC and the Crossbow IMU through the use of neoprene rubber as a cushion as well as rubber mounting pads, for the engine mount, which will eliminate most of the noise from the source. A separate battery source is used for the ODAC and the inputs are properly shielded to prevent noise. An active second-order Butterworth filter was also added to filter the incoming signals before they are sent to the base station for recording. The component interface of the ODAC is shown in Figure 2a. After the ODAC is turned on, the system is initialized from the base station and data acquisition and logging are performed. A Pentium III 1GHz laptop is used as the base station computer which interfaces with the radio modem. A variety of maneuvers maybe commanded from the base station. The data logging feature maybe turned on at the base station to store the data in .dat files on the base station computer and can be retrieved for

further analysis. The functional diagram of the base station system (BSS) is shown in Figure 3.

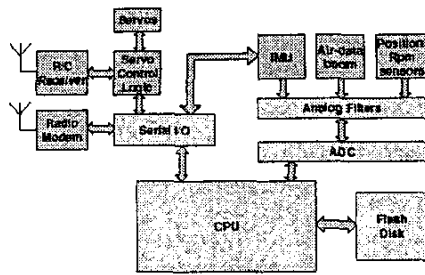


Figure 2a. Component Interface of ODAC

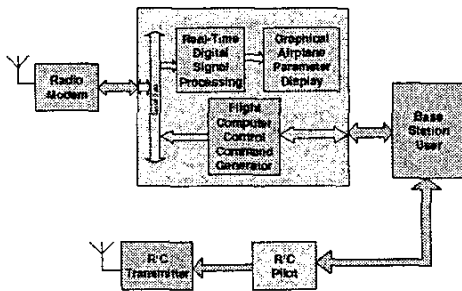


Figure 3. Functional Diagram of BSS

System Identification

To model the aircraft for analytical computations, standard six degree of freedom (DOF) nonlinear aircraft equations of motion will be used during the study [6]. Using telemetry data from flights and System Identification Programs for Aircraft [7] (SIDPAC), stability parameters will be estimated and compared to estimates from Advanced Aircraft Analysis [8] (AAA). The SIDPAC program used to compute stability derivatives utilizes an equation error approach solved via a least mean squares algorithm. AAA takes physical parameters of an aircraft as inputs and outputs stability derivatives based on the physical aircraft characteristics.

The flight maneuvers designed for the system identification was a 3-2-1-1 longitudinal maneuver utilizing the elevator to provide estimates of the lift and moment derivatives. A small impulse maneuver to record the Phugoid mode of the aircraft to extract drag coefficients, and simple doublet maneuvers for the ailerons and rudder to estimate the lateral-directional derivatives will also be executed. Some of the results obtained using the 3-2-1-1 maneuver is shown in the results and discussion section.

The stability derivatives that are needed for the six DOF equations of motion along with major aircraft parameters are listed in Table 1. A simulation of the six DOF equations of motion will be compared to the response of the actual aircraft model; the equations of motion will be tailored to account for the errors in the response.

Table 1. Parameters for 30% Scale Cessna 150

Weight	36 lbs	Max Thrust	19 lbs	C_{mih}	-1.4411 1/rad
I_{xx}	1.8 slug/ft ²	C_{D0}	0.0415	$C_{m\delta e}$	-0.4380 1/rad
I_{yy}	2.4 slug/ft ²	C_{Da}	0.3141 1/rad	$C_{l\beta}$	-0.0471 1/rad
I_{zz}	3.7 slug/ft ²	C_{Dih}	0.0126 1/rad	$C_{l\delta a}$	0.2875 1/rad
I_{xz}	0 slug/ft ²	$C_{D\delta e}$	0.0097 1/rad	$C_{l\delta r}$	0.0291 1/rad
b	10 ft	C_{L0}	0.2011	$C_{y\beta}$	-0.3628 1/rad
S	15.33 ft ²	C_{La}	4.9282 1/rad	$C_{y\delta a}$	0 1/rad
c _{bar}	1.533 ft	C_{Lih}	0.5868 1/rad	$C_{y\delta r}$	0.1596 1/rad
i_w	1 deg	$C_{L\delta e}$	0.3803 1/rad	$C_{n\beta}$	0.0004 1/rad
I_h	-1.5 deg	C_{m0}	0.0212	$C_{n\delta a}$	-0.0270 1/rad
Engine	Moki 2.1 in ³	C_{ma}	-0.1066 1/rad	$C_{n\delta r}$	-0.0758 1/rad

Controller Designs

After proper parameter estimates and a sufficient system model have been developed, synthesis of the intended controllers may proceed. The types of controllers that will be implemented are as follows:

Dynamic Inversion Technique

Dynamic inversion [4], a form of feedback linearization that derives its control from an equation that describes the dynamics of the error, was chosen to be the first controller in order to verify the flight hardware. For an example of the process, define a nonlinear system like that of an aircraft

$$\dot{X} = f(X) + g(X) \cdot U_C \quad (1)$$

where X is an $n \times 1$ state vector containing n states and U_C is an $n \times 1$ control vector. Note that for the given aircraft problem, $f(X)$ and $g(X)$ are square matrices. The error dynamics is desired to have the following form

$$\dot{\hat{X}} + K \cdot \hat{X} = 0 \quad (2)$$

where the error between current and desired values is given as

$$\hat{X} = X - X^* \quad (3)$$

and K represents the inverse error dynamics time constant. Substituting Equation (3) and (1) into Equation (2) and assuming step commands we get

$$A \cdot U_C = b \quad (4)$$

where

$$A = g(X) \quad (5)$$

and

$$b = -K(X - X^*) - f(X) \quad (6)$$

By multiplying both sides by the inverse of A (assuming it exists), a control solution is computed as

$$U_C = A^{-1} \cdot b \quad (7)$$

Using commands such as roll angle, normal acceleration, lateral acceleration, and forward speed, a longitudinal mode dynamic inversion controller is used to output four control variables: elevator, aileron, and rudder deflections as well as throttle percentage. A second controller is used for lateral maneuvers to control roll angle, altitude, lateral acceleration, and forward speed while utilizing the same control variables. Both controllers have the ability to control both lateral and longitudinal motion but the commanded longitudinal state variable changes. In the longitudinal mode controller, normal acceleration is commanded while in the lateral mode controller, altitude is directly commanded. This was done to create more precise altitude control (hold) in a lateral, directional maneuver. The tracking commands are generated from the errors between the current aircraft states and the commanded states which are given by the user. Lastly, the two controllers are individual and will not operate the aircraft at the same time.

Single Network Adaptive Critic (SNAC)

The second controller to be implemented on the aircraft will be a neural network based optimal controller in the form of a Single Network Adaptive Critic (SNAC) architecture [5]. The SNAC is very powerful with its origins in approximate dynamic programming, which offers solutions to optimal control problems. Its development is given in this section.

In a discrete form the previously mentioned aircraft equations of motion can be written as

$$X_{k+1} = F_k(X_k, U_k) \quad (8)$$

with the state and control vectors the same as previously mentioned. The goal is to find a controller minimizing a cost function J given by

$$J = \sum_{k=1}^{N-1} \Psi_k(X_k, U_k) \quad (9)$$

which minimizes the error and error rate between the actual and commanded states. These states are the same as the dynamic inversion controllers. Here, k denotes the time step while X_k and U_k

represent the states and control respectively. Ψ_k is assumed to be convex (e.g. a quadratic function in X_k and U_k).

By rewriting Equation (9) to start from time step k as

$$J_k = \sum_{\bar{k}=k}^{N-1} \Psi_{\bar{k}}(X_{\bar{k}}, U_{\bar{k}}) \quad (10)$$

J_k can be split into

$$J_k = \Psi_k + J_{k+1} \quad (11)$$

where Ψ_k and $J_{k+1} = \sum_{\bar{k}=k+1}^{N-1} \Psi_{\bar{k}}$ represent the

“utility function” at time step k and the cost-to-go from time step $k+1$ to N , respectively. The costate vector at time step k is defined as

$$\lambda_k = \frac{\partial J_k}{\partial X_k} \quad (12)$$

The optimality condition for the cost function is given by

$$\frac{\partial J_k}{\partial U_k} = 0 \quad (13)$$

and further reduced to

$$\left(\frac{\partial \Psi_k}{\partial U_k} \right) + \left(\frac{\partial X_{k+1}}{\partial U_k} \right)^T \lambda_{k+1} = 0 \quad (14)$$

The costate equation is derived in the following way

$$\begin{aligned} \lambda_k &= \frac{\partial J_k}{\partial X_k} = \left(\frac{\partial \Psi_k}{\partial X_k} \right) + \left(\frac{\partial J_{k+1}}{\partial X_k} \right) \\ &= \left[\left(\frac{\partial \Psi_k}{\partial X_k} \right) + \left(\frac{\partial X_{k+1}}{\partial X_k} \right)^T \lambda_{k+1} \right] + \left(\frac{\partial J_{k+1}}{\partial X_k} \right) \end{aligned} \quad (15)$$

By using Equation (14), in (15), we get

$$\lambda_k = \left(\frac{\partial \Psi_k}{\partial X_k} \right) + \left(\frac{\partial X_{k+1}}{\partial X_k} \right)^T \lambda_{k+1} \quad (16)$$

The steps in SNAC network training are as follows (Figure 4):

1. Generate a set of training points. For each point in the training set:
 - a. Input X_k to the critic network to obtain $\lambda_{k+1}^a = \lambda_{k+1}^a$
 - b. Calculate U_k from the optimal control equation (14) with known X_k and λ_{k+1}^a .
 - c. Get X_{k+1} from the state equation (8) using X_k and U_k
 - d. Input X_{k+1} to the critic network to get λ_{k+2}
 - e. Using X_{k+1} and λ_{k+2} , calculate λ_{k+1}^a from the costate equation (16)
2. Train the critic network for all X_k in the training set to output λ_{k+1}^a .
3. Check convergence of the critic network. Convergence is defined as minimal change in the critic network between subsequent critic network trainings. If convergence is achieved, revert to step 1 with the next element of the training set. Otherwise, repeat steps 1-2.
4. Continue steps 1-3 until finished with the training set.

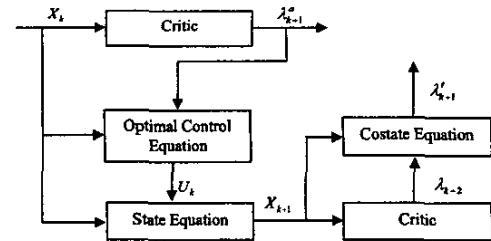


Figure 4. SNAC Training Procedure

The SNAC network is broken into four separate networks, one for each control output. Each network is a feed forward network with 13 or 14 neurons in the first layer, 18 neurons in the hidden layer, and 1 neuron in the output layer. The activation function for each of the layers is tangent sigmoid, tangent sigmoid, and linear combination, respectively. During network training the inputs (states) are normalized with respect to a predetermined value specified by the user. The

output of the network (control/costate) is not normalized. The SNAC network is trained offline with a representative training set of possible inputs (state combinations). The Levenberg-Marquardt training algorithm is used to train the networks with a learning rate of 0.5.

Neural networks are widely known for their ability to handle nonlinearities in control systems. This study will determine the network's ability to successfully control a nonlinear aircraft as well as an aircraft with mildly simulated damage.

Outer Loop Extra Control

As a third step, we plan to append the analytical work under way to have an online learning neural network to account for the highly stressed situations such as a frozen control surface, etc. This neural network would monitor the errors between the aircraft model and the actual flight data and output extra control to bring the error between the aircraft and the model to zero. The diagram for the extra control process can be seen below in Figure 5.

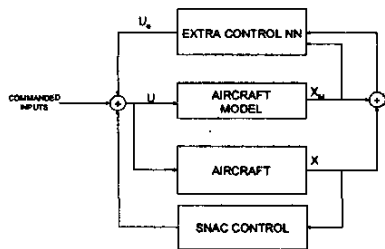


Figure 5. Outer Loop Extra Control Scheme

The extra control neural network will be a learning neural network that updates its weights based on a training algorithm that feeds off its inputs: the model state vector and the error between the actual aircraft and the model. This work would be similar to previous work done at UMR in which a radial basis function neural network was used to add extra control based on errors in the system due to uncertainties [9].

Testing Procedures

After the above controllers are verified to work on the system model during computer simulation,

they will be implemented in the aircraft. Testing procedures specific to the type of controller being implemented will be followed.

Dynamic Inversion

The two dynamic inversion controllers (longitudinal and lateral) will be implemented in much the same way. First a transfer between the normal R/C system and the microcontroller must be validated. This is essential to having the dynamic inverse controller able to take over the aircraft during trim flight. This step will be completed with both controllers using commands similar to a simple autopilot (steady state, non-turning flight). After this is accomplished, specific tasks will be tested for both controllers. For the longitudinal controller, simple altitude changes will be commanded. For the lateral controller, simple turns will be performed with possible altitude changes incorporated later. Once the initial maneuvers are carried out, more commands can be given over time.

Single Network Adaptive Critic

The SNAC controller is trained via a certain maneuver for the aircraft such as straight and level flight then a turn to the left, then a turn to the right, and finally straight and level flight. Once implemented, the trained maneuver would be commanded. After this maneuver was complete, the extents of the network's capabilities will be tested via maneuvers modified from the original trained maneuver. The network will also be tested with respect to control surfaces that may have restricted ranges or even hard coded offsets or biases that would simulate a change in the system model. The testing procedure for the Outer Loop Extra Control SNAC controller will be similar to the regular SNAC controller testing procedure and should exhibit more robust characteristics than the single SNAC network.

Results and Discussion

Telemetry flights of the aircraft have been successfully performed for data collection, verification, and parameter estimation, but currently, only the moment derivatives have been identified. Figure 6 shows a graph of a 3-2-1-1

maneuver for the elevator. For our designed 3-2-1-1 maneuver, the 2 pulse was determined to last 2 seconds and the 3 and 1 pulse were scaled accordingly. Three of the recorded variables excited by this maneuver: Theta, a_z , and Q are displayed in Figure 7. Theta and Phi are recorded through the use of a Kalman filter hard coded into the IMU. The roll rate, P, is computed by the IMU by taking the derivative of the pitch angle, Theta. Using SIDPAC the moment derivatives have been estimated and are shown in Table 1.

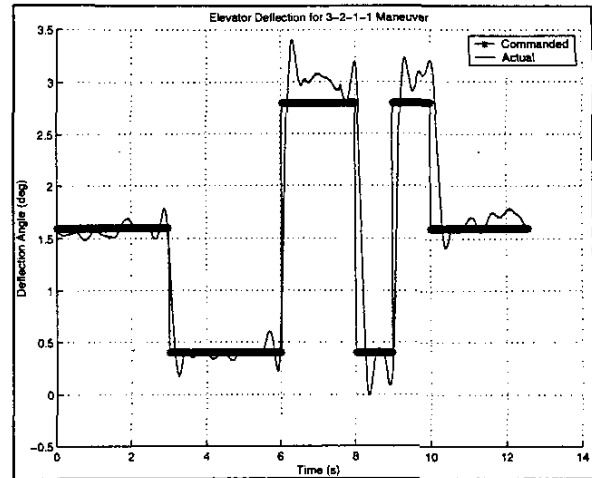


Figure 6. 3-2-1-1 Elevator Maneuver

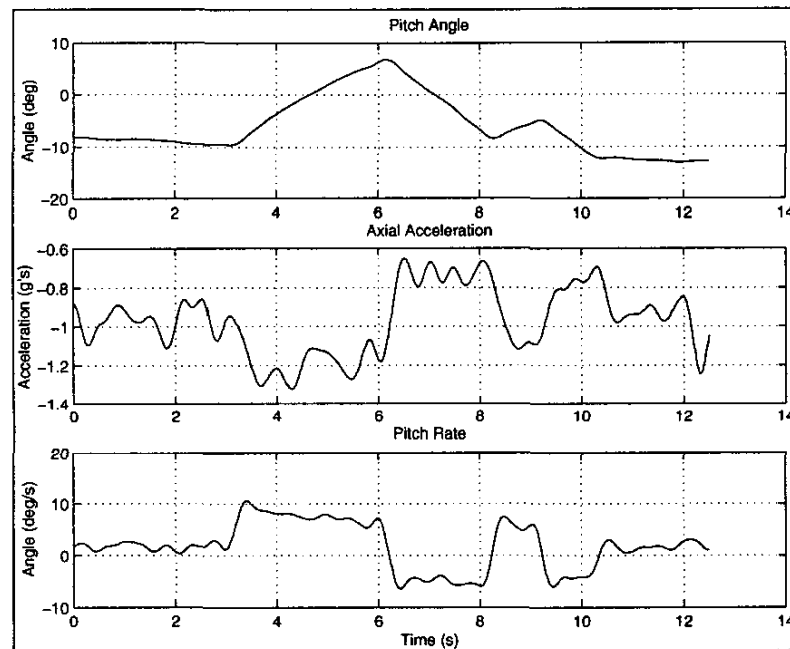


Figure 7. Recorded States

Computer simulation will be completed on all of the controllers before they are flight tested. Figures 8, 9, and 10 show the results from the dynamic inversion controller operating in the lateral mode on the 30% scale Cessna 150. Figure 8 shows the commanded variables as a solid line and the commands plotted as a dashed line while Figure 9 shows the control usage. The commanded variables are as follows: bank angle, Phi, is

commanded to -5 deg for a minute and then to 5 deg for another minute before returning to a value of 0 deg, altitude was commanded to increase by 200 feet for the first minute and then to decrease by 400 feet for the next minute and then to return to the initial altitude, lateral acceleration was commanded to 0 for coordinated flight, and forward speed, U, was commanded to be constant at the trim velocity before the maneuver.

This maneuver was selected to show the capabilities of the lateral mode controller when commanded changes in the longitudinal and lateral modes of motion. The longitudinal mode controller has the same cross mode capabilities but, instead of

commanding altitude, normal acceleration is commanded. Figure 10 displays the flight trajectory in three dimensions and verifies the successful simulation of the aircraft.

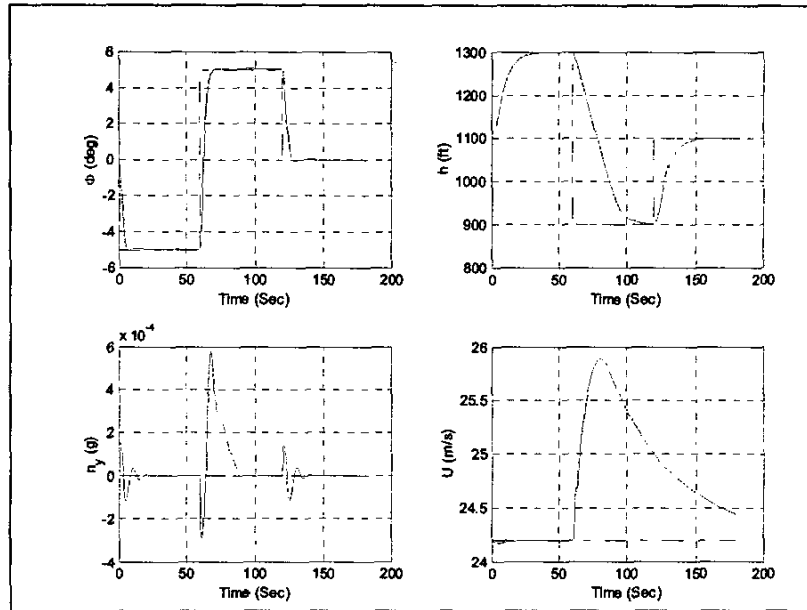


Figure 8. Commanded Variables, Lateral Mode

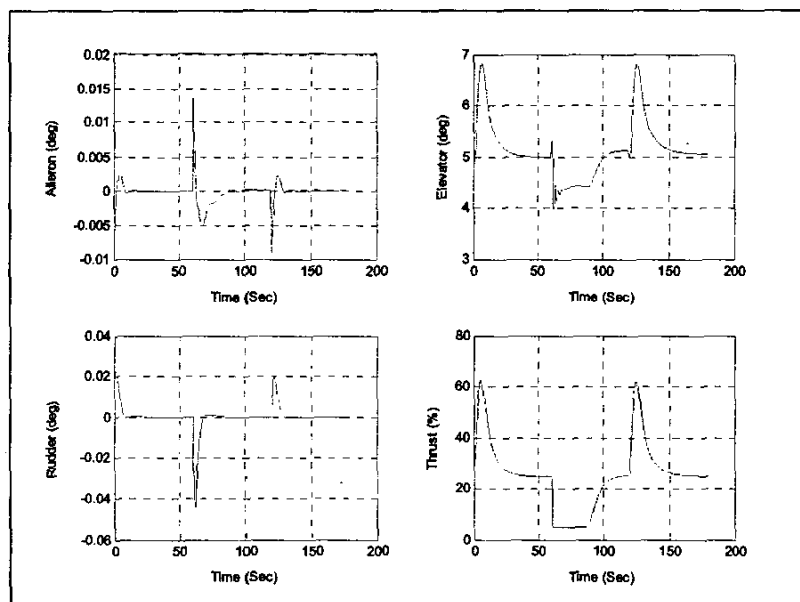


Figure 9. Control Inputs

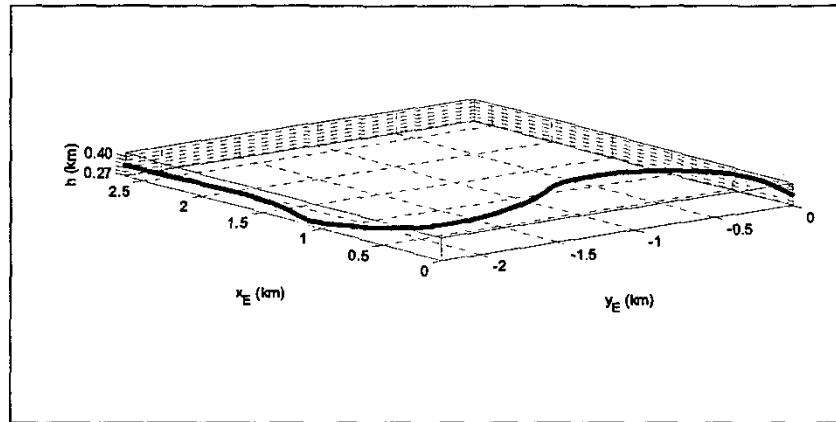


Figure 10. Three Dimensional Aircraft Trajectory

The total simulation lasted three minutes and was run on a time step of 0.02s. The equations of motion are solved using a fourth order Runge-Kutta solver with fixed step size developed for use in Matlab. The simulation tracked very well as can be seen by the commanded variables plot. The error in bank angle decreased at a rate of about 5 deg per 10s. Error in altitude decreased at a rate of 200 feet per about 30 seconds. Coordinated flight was achieved with maximum lateral acceleration near $6 \cdot 10^{-4}$ g's. Forward speed, commanded to remain constant, only showed a departure from the initial velocity when the aircraft began the descent, during which the throttle control saturated to the least amount of thrust. All other controls throughout the simulation remained at very acceptable magnitudes of 7 deg or less.

The SNAC controller synthesis is nearly complete and will be simulated for the Cessna 150 soon. A similar extra control neural network has been proven to work on a nonlinear helicopter model[9] but has not been applied to an aircraft simulation.

Work on the aircraft is continuing and work left for the project is as follows, but not necessarily in the order given:

- 1 Simulate the SNAC controller with the Cessna 150.
- 2 Simulate the outer loop control with the Cessna 150.
- 3 Perform telemetry flight with Cessna 150.
- 4 Complete parameter estimation of stability and control derivatives.

- 5 Implement dynamic inversion controller and flight test.
- 6 Implement SNAC controller and flight test.
- 7 Implement outer loop NN with SNAC controller and flight test.

Conclusion

Implementation of non-linear flight controllers in autonomous air vehicles is becoming a very important aspect of aerospace engineering. This paper consists of the system identification and the plan to implement nonlinear flight controllers via a 30% scale Cessna 150 that is fitted with full state feedback equipment. The first control technique chosen for hardware validation was a modified dynamic inversion technique. After the dynamic inversion controller and the flight hardware have been proven, a SNAC controller will be implemented to test the viability and robustness of such a controller. Next, an extra control controller via an online learning neural network will be implemented around the SNAC controller to account for changes in the system dynamics.

Many studies have been done to implement adaptive control in pilot controlled aircraft. When complete, this project will have validated the use of a SNAC controller for use in aircraft as well as studied its robustness in the presence of system uncertainties and actuator failures. This study will further the knowledge in nonlinear control implementation by being one of the first to implement a nonlinear, optimal, and reconfigurable controller for an autonomous aircraft.

References

- [1] Krishnakumar, K., G. Limes, K. Gundy-Burlet, and D. Bryant, "An Adaptive Critic Approach to Reference Model Adaptation," AIAA-2003-5790.
- [2] Gundy-Burlet, K., K. Krishnakumar, G. Limes, and D. Bryant, "Control Reallocation Strategies for Damage Adaptation in Transport Class Aircraft," AIAA 2003-5642.
- [3] Calise, Anthony J., and Rysdyk, Rolf T., "Adaptive Model Inversion Flight Control for Tiltrotor Aircraft," AIAA-97-3758.
- [4] Padhi, R., and Balakrishnan, S.N., "Implementation of Pilot Commands in Aircraft Control: A Modified Dynamic Inversion Based Approach," AIAA-2003-5505.
- [5] Padhi, R., Unnikrishnan, N., Balakrishnan, S.N., Optimal Control Synthesis of a Class of Nonlinear Systems Using Single Network Adaptive Critics, Accepted, 2004 American Control Conference, Submission Number 1479.
- [6] Roskam, J., Airplane Flight Dynamics and Automatic Flight Controls, DARcorporation, Lawrence, KS, 1995
- [7] Morelli, E.A., "System Identification Programs for Aircraft (SIDPAC)," AIAA-2002-4704.
- [8] Advanced Aircraft Analysis (AAA), Software Package, Ver. 2.4, DARcorporation, Lawrence, KS.
- [9] Huang, Z., and Balakrishnan, S.N., "Robust Neurocontrollers for Systems with Model Uncertainties – A Helicopter Application," Journal of Guidance, Navigation, and Control (to be published).

Email Addresses

Vijayakumar Janardhan, vijay@umr.edu

Derek Schmitz, dts@umr.edu

S. N. Balakrishnan, bala@umr.edu