Implementation of Nonlinear Reconfigurable Controllers for Autonomous Unmanned Vehicles

Derek Schmitz
Vijayakumar Janardhan
S. N. Balakrishnan
Department of Mechanical and Aerospace engineering and Engineering Mechanics
University of Missouri - Rolla

Abstract
Reconfigurable control will play a major role in the advancement of aerospace control especially in the realm of autonomous air vehicles. This project consists of the development and implementation of three nonlinear controllers. A modified dynamic inversion controller will be used for validation of the flight hardware. A Single Network Adaptive Critic (SNAC) controller will then be implemented to test its viability and robustness. Lastly, an outer loop controller in the form of an online learning neural network will be developed and implemented to supply extra control to the actual aircraft if a change in system dynamics is detected. To aid in system identification, parameter estimation will be performed before any autonomous flights are performed. The results of this project will validate the use of a SNAC controller for autonomous flight as well as prove the viability of an outer loop extra controller to account for changes in system dynamics.

1. Introduction
Loss of lives in airplane crashes and the recent spurt in interest in using unmanned air vehicles for many civilian and military missions has caused a great deal of interest in the study of reconfigurable control of manned aircraft in general and autonomous systems. A reconfigurable control system installed in an aircraft can protect it under stressed conditions whether it is a loss of an engine or a frozen control surface or battle-damage. Furthermore, in an unmanned vehicle, where uncertainty is a prominent concern, reconfigurable control has become a building block of future control systems.

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 a dynamic inversion control techniques with direct adaptive control from pre-trained and online neural networks[2].

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. The aircraft will rely on feedback information from a gyroscope and an air data boom that will be mounted on the aircraft. Along with the feedback sensors, a microcontroller and a radio modem will also be located on the aircraft to act as the airplane controller and 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 group3 will be implemented on the aircraft to validate the control hardware. A more sophisticated optimal control based neural network controller design of our group4 will be implemented next to test its performance under mildly stressed conditions. Furthermore, analytical formulations underway now 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, which has hardly been done elsewhere.

2. The Autonomous Unmanned Aerial Vehicle
A thirty percent scale model of a Cessna 150, shown in Fig. 1 is used as the autonomous unmanned test vehicle. It has a 10 ft. wingspan, weighs 35 lbs, and utilizes a Moki 2.1 cu. in. engine for power. Because the Cessna 150 is a stable airplane, a scale model of the same is used for implementation. 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. The 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 can be switched between the microcontroller and the RC pilot commands.

The onboard data acquisition (ODAQ) system comprises of a PC-104 486 DX4 at 100MHz with 32 MB of RAM and 32 MB of Flash RAM. The ODAQ runs MSDOS and has 16 12-bit analog inputs, 4 serial ports, Ethernet and parallel port and 4 12-bit analog outputs. The ODAQ communicates with the base station through a 115.2 Kbps RS232 radio modem from Cirronet Inc. A Pentium III 1GHz laptop is used as the base station computer which interfaces to the 115.2 Kbps RS232 radio modem from Cirronet Inc.

The roll, pitch and yaw accelerations and rates and the roll and pitch angles are provided by an Inertial Measurement Unit (IMU) VG-400CA from Crossbow. The airspeed, altitude, angle of attack and the side slip angle is 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 ODAQ and the Crossbow IMU. A separate battery source is used for the ODAQ and the inputs are properly shielded to prevent noise. After the ODAQ is turned on, the system is initialized from the base station and data acquisition and logging can be performed. The data is stored in .dat files on the base station computer and can be retrieved for further analysis later on.

3. System Identification
To model the aircraft for analytical computations, standard 6 degree of freedom nonlinear aircraft equations of motion will be used during the study. Using telemetry data from flights, stability parameters will be estimated and compared with the simulation based estimates from Advanced Aircraft Analysis (DARcorp). If the equations of motion prove unusable as a system model, a system model derived from the input/output telemetry data. To accomplish this system identification, step inputs to the system will be given and the response of the aircraft will be logged.

4. Controller Designs
After a proper and sufficient system model has been validated, synthesis of the intended controllers may proceed. The types of controllers that will be implemented are as follows:

**Dynamic Inversion Technique**
Dynamic inversion, 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. For our example, define a nonlinear system like that of an aircraft

\[ \dot{X} = f(X) + g(X) \cdot U_C \]  \hspace{1cm} (1)

The error dynamics is desired to have the following form

\[ \dot{X} + K \cdot \dot{X} = 0 \]  \hspace{1cm} (2)

where the error between current and desired values is given as

\[ \dot{X} = X - X^* \]  \hspace{1cm} (3)
where $K$ represents the inverse error dynamics time constant.

Substituting Equation (3) and (1) into Equation (2) and assuming step commands we get

$$A \cdot U_C = b$$

where

$$A = g(X)$$

and

$$b = -K(X - X^*) - f(X)$$

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

$$U_C = A^{-1} \cdot b$$

Using commands such as roll rate, normal acceleration, lateral acceleration, and forward speed, a longitudinal mode dynamic inversion controller is used to output four control variables: elevator, aileron, and rudder deflection as well as throttle. A second controller is used for lateral maneuvers to control roll rate, altitude, lateral acceleration, and forward speed.

**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 architecture. The SNAC is very powerful with its origins in approximate dynamic programming, which offers comprehensive solutions to optimal control and its development is given in this section.

In a discrete form the aircraft equations can be written as

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

The goal is to find a controller minimizing a cost function $J$ given by

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

where $k$ denotes the time step. $X_k$ and $U_k$ represent the states and control respectively. $\Psi_k$ is assumed to be convex (e.g. a quadratic function in $X_k$ and $U_k$).

By rewriting Eq.(9) to start from time step $k$ as

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

$J_k$ can be split into

$$J_k = \Psi_k + J_{k+1}$$

where $\Psi_k$ and $J_{k+1} = \sum_{k'=k+1}^{N-1} \Psi_{k'}$ 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

$$\lambda_k = \frac{\partial J_k}{\partial X_k}$$
Optimality condition is given by
\[ \frac{\partial J_k}{\partial U_k} = 0 \] (13)
and further reduced to
\[ \left( \frac{\partial \Psi_k}{\partial U_k} \right) + \left( \frac{\partial X_{k+1}}{\partial U_k} \right)^T \lambda_{k+1} = 0 \] (14)

The costate equation is derived in the following way
\[ \lambda_k = \frac{\partial J_k}{\partial X_k} = \left( \frac{\partial \Psi_k}{\partial X_k} \right) + \left( \frac{\partial J_{k+1}}{\partial X_k} \right) \]
\[ = \left[ \left( \frac{\partial \Psi_k}{\partial X_k} \right) + \left( \frac{\partial X_{k+1}}{\partial X_k} \right)^T \lambda_{k+1} \right] + \left( \frac{\partial U_k}{\partial X_k} \right)^T \left( \frac{\partial \Psi_k}{\partial U_k} \right) + \left( \frac{\partial X_{k+1}}{\partial U_k} \right)^T \lambda_{k+1} \] (15)

By using Equation (14), in (15), we get
\[ \lambda_k = \left( \frac{\partial \Psi_k}{\partial X_k} \right) + \left( \frac{\partial X_{k+1}}{\partial X_k} \right)^T \lambda_{k+1} \] (16)

The steps in SNAC network training are as follows (Figure 2):
1. Generate a set of training points. For each point in the training set:
   a. Input \( X_k \) to the critic network to obtain \( \lambda_{k+1} = \lambda_{k+1}^o \)
   b. Calculate \( U_k \) (Eq.14) with known \( X_k \) and \( \lambda_{k+1} \).
   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 \( \lambda_{k+2} \)
   e. Using \( X_{k+1} \) and \( \lambda_{k+2} \), calculate \( \lambda_{k+1}^t \) from costate equation (16)
2. Train the critic network for all \( X_k \) in the training set to output \( \lambda_{k+1}^w \).
3. Check convergence of the critic network. 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 2 - SNAC Scheme](image_url)
Neural networks are widely known for their ability to handle nonlinearities in control systems. This study will determine how well the network will have the ability to successfully control 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 frozen controller 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 (Figure 3).

![Figure 3 – Outer Loop Extra Control](image)

The extra control neural network will be a learning neural network that updates its weights based on a training algorithm that feeds off the inputs, the state vector and the error between the actual aircraft and the model.

**5. Testing Procedures**

After the above controllers are verified to work on the system model 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 to be able to take over the system. This step will be completed with both controllers with commands like a simple autopilot (steady state, non-turning flight). After this is accomplished specific task 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.

**6. Preliminary Results and Discussion**
Computer simulation will be completed on all of the controllers before they are flight tested. Figure 2, 3, and 4 show the results from the dynamic inversion controller operating in the lateral mode on the 30% scale Cessna 150. Figure 2 shows the commanded variables while Figure 3 shows the control usage. Figure 4 displays the flight trajectory in three dimensions.

![Figure 2 – Commanded Variables, Lateral Mode](image)

![Figure 3 – Control Inputs](image)
The SNAC controller synthesis is nearly complete and will be simulated for the Cessna 150 soon. A SNAC control architecture has been simulated with a Boeing 747. The extra control neural network has been proven to work on a nonlinear helicopter model but has not been applied to an aircraft simulation. Work left to do for the project is as follows:

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.
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.

7. Conclusions

This project consists of the implementation of nonlinear flight controllers via a 30% scale Cessna 150. The scale model is fitted with full state feedback equipment. Also it carries a radio modem for communication with a base station. The control technique that was chosen for hardware validation was a modified dynamic inversion technique. After the dynamic inversion controller and the flight hardware has been proven a SNAC controller will be implemented to test the viability and robustness of such a controller. Next, an extra control controller will be implemented around the SNAC controller to account for changes in the system dynamics. We expect to show the results of these tests at the conference.

Many studies have been done to implement adaptive control in pilot controlled aircraft. This study would be one of the first to implement a nonlinear, optimal, and reconfigurable controller for an autonomous aircraft.
References


