JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 27, No. 3, May–June 2004

Adaptive Neural Network Inverse Controller for General Aviation Safety

Urpo J. Pesonen,* James E. Steck,† and Kamran Rokhsaz‡
Wichita State University, Wichita, Kansas 67260
and
Hugh Samuel Bruner§ and Noel Duerksen¶
Raytheon Aircraft Company, Wichita, Kansas 67206

An advanced flight control system developed for and demonstrated to compensate for unanticipated failures in military aircraft is investigated for use in general aviation. This method uses inverse control to decouple the flight controls and modify the handling qualities of the aircraft. The purpose of the system is to render a general aviation aircraft easier to fly by decoupling its flight control system and making the aircraft handling more natural to a nonpilot. Artificial neural networks are used to counteract the modeling errors in the inverse controller but, more important, to adapt to unanticipated failures during flight, thus, allowing the pilot to continue to control the aircraft safely. Because a decoupled flight control system is software based, it is a fly-by-wire system. For such a system, it is difficult from a cost standpoint for general aviation to incorporate the level of redundancy required in such flight control systems; therefore, the demonstration of this system's capability to handle control system failures is critical to future certification efforts. The system is verified with MATLAB® simulations for longitudinal flight. In simulations, the control system is shown to be able to track pilot velocity and pitch angle and flight-path angle commands. Simulations of changing configurations, payload, and partial control system failures have shown that the controller does rapidly adapt to these changes without a need for a pilot response. A pilot-in-the-loop flight simulator has verified the MATLAB simulations and ongoing work to flight test the control system on a Raytheon Bonanza F33C fly-by-wire testbed is discussed.

Nomenclature

$C_{D_{\delta_e}}$	=	change in drag coefficient with elevator deflection
C_L, C_m, C_D	=	lift, pitching moment, and drag coefficients
$C_{L_{\delta_e}}$	=	change in lift coefficient with elevator deflection
C_{m_α}	=	change in pitching moment coefficient with angle of attack
$C_{m_{\delta_e}}$	=	change in pitching moment coefficient with elevator deflection
$C_{T_{\delta_T}}$	=	change in thrust coefficient with throttle
$\bar{c}, b$	=	aircraft reference chord length and wing span
$F_{A_x}, F_{A_y}, F_{A_z}$	=	aerodynamic forces in forward, side, and vertical directions
$I_{xx}, I_{yy}, I_{zz}, I_{xz}$	=	aircraft mass moment of inertias
L, M, N	=	aircraft rolling, pitching, and yawing moments
m	=	mass of the aircraft
P, Q, R	=	roll, pitch, and yaw aircraft angular velocities
q	=	dynamic pressure
S	=	aircraft reference wing area
T	=	thrust
U, V, W	=	forward, side, and vertical aircraft velocities
V_c, θ_c, γ_c	=	commanded airspeed, pitch angle, and flight-path angle
V_p or V_{true}	=	total aircraft airspeed
α, β	=	aircraft angle of attack and sideslip
γ	=	aircraft flight-path angle
$\delta_a, \delta_e, \delta_r$	=	aileron, elevator, and rudder deflections
ϕ_T, d_T	=	angle and moment arm of the thrust force
Ψ, Θ, Φ	=	aircraft heading, pitch, and roll angles

I. Introduction

ADVANCES in modern fight control design provide the means to design an operationally simplified control system that allows a general aviation aircraft to be flown safely with a lower level of piloting skills. This can increase aviation safety and make personal air transport available to a larger group of people who are not aviation enthusiasts. NASA's Small Aircraft Transportation System (SATS) Program aims to provide reliable personal air travel for a wider audience, with the objective of a flight control system that provides a reduced pilot workload through decoupled control modes and stability augmentation. The envisioned SATS aircraft also enables pilots with low experience to operate the aircraft in most weather conditions, has an emergency auto-land capability, and adapts to changes in aircraft behavior due to unanticipated actuator/sensor failures or structural damage.

Modern flight control technology, including fly by wire, has been used in commercial aviation for over a dozen years (Airbus A320, A330, and A340, and Boeing 777). Reliability of these types of systems is typically established through high-integrity design including redundancy of various independent parts of the control system and is validated through a certification process requiring careful documentation of the design process and extensive demonstration, which includes failure scenarios. This method of establishing reliability can be cost prohibitive when applied to general aviation aircraft. This is one of the reasons most general aviation aircraft have mechanical control systems. General aviation (GA) can also benefit from modern flight control technology by using it to provide a huge reduction of flight accidents commonly referred to as pilot error.

Presented as Paper 2003-578 at the 41st Aerospace Sciences Meeting and Exhibit, Reno, NV, 6 January 2003; received 15 April 2003; revision received 16 September 2003; accepted for publication 10 October 2003.

*Research Associate, Aerospace Engineering. Member AIAA.
†Associate Professor, Aerospace Engineering. Senior Member AIAA.
‡Professor, Aerospace Engineering. Associate Fellow AIAA.
§Director, Advanced Design Grade. Senior Member AIAA.
¶Engineering Fellow.

This can be accomplished by incorporating low-cost flight control systems designed specifically for GA aircraft that have enhanced safety features to reduce the impact of pilot errors and low piloting skills on accident probabilities, as well as their severity. In addition, the nonlinear adaptive control method investigated in this project provides an alternative to redundancy for achieving a required level of safety by adapting the flight control system in-flight to accommodate for failures of different subsystems. Also, failure tolerance, or the ability to adapt to a failure in the flight control system, goes beyond redundancy by providing a flexible means to deal with unforeseen failures, thereby improving survivability. Nonlinear adaptive control provides the following important benefits: 1) consistency in response without the need for expensive gain scheduled design, 2) the ability to take advantage of inherent control redundancy without a priori knowledge of the failure, and 3) a required guarantee of boundedness with traditional analytical tools. This establishes finite bounds on the response of the aircraft with the control system in place. The disadvantages are as follows: 1) The supporting theories are complex. 2) Hardware, software, and their integration require investigation. 3) The adaptive nature requires careful application. 4) It may be difficult to prove determinism of the algorithms and to prove their stability and their performance robustness under all operating conditions.

The work presented in this paper is the beginning of an investigation and flight demonstration of the usefulness of adaptive control techniques in general aviation software-based fly-by-wire EZ-fly systems. Adaptive nonlinear control techniques have been investigated and demonstrated by several research groups for military and commercial aviation. Most are based on linearizing the equations of motion and using feedback. Brinker and Wise[1] found that a dynamic inverse controller could be used for aircraft and that the longitudinal stability and flying qualities were robust to parameter uncertainties, but the lateral directional flying qualities were sensitive to uncertainty in stability derivatives. For longitudinal flight control of a missile, McFarland and Calise[2] proposed a method where neural networks and direct adaptive control are used to compensate for unknown nonlinearities, whereas dynamic nonlinear damping provides robustness to unmodeled dynamics. In a later study,[3] the same authors used a similar methodology for a bank-in-turn control of a missile. In both studies, a neural network was used to cancel linearization errors adaptively through on-line training. Artificial neural networks (ANN) are capable of approximating continuous nonlinear functions with very little memory and computational time required, but because of the empirical character of these methods, it has previously been difficult to guarantee sufficient reliability for such a high-risk application as flight control. Using neural networks for nonlinear inverse control of the XV-15 tilt-rotor, Rysdyk et al.[4] showed theoretically, as well as by simulation, that the ANN weights remain bounded during online training. This is an important step toward certifying aircraft control systems that use ANN. The guaranteed boundedness of signals and tracking error is discussed in detail in Ref. 5. Rysdyk et al.[6] also obtained consistent response characteristics throughout the operating envelope of a tilt-rotor aircraft. Furthermore, in a recent study,[7] Rysdyk and Agarwal applied ANN to a total energy control system for longitudinal flight control.

Soloway and Haley at NASA studied reconfigurable aircraft control.[8] Their model is capable of real-time control law reconfiguration, model adaptation, and identification of failures in model effectiveness. A full six-degree-of-freedom (6-DOF) model of a conceptual commercial transport aircraft was used to simulate the elevator freezing in flight, and the algorithm reconfigured itself to use symmetric aileron deflections to control pitch rate, thereby stabilizing the aircraft. Again, an ANN was used to learn the changed dynamics of the aircraft with frozen elevators. Kim and Calise[9] developed a direct adaptive tracking control using neural networks to represent the nonlinear inverse transformation needed for feedback linearization. It was shown that the adaptation algorithm ensured uniform boundedness of all signals in the loop and that the weights of the online neural network converged to constant values. In a study concerning a serial-link robot arm control, Lewis et al.[10] found that online training of the neural network, together with a signal that adds robustness, guarantees tracking as well as bounded ANN weights. Standard backpropagation, however, yielded unbounded ANN weights if the network was not able to reconstruct a required control function or there were unknown disturbances in the system dynamics. Alternate backpropagation schemes corrected this problem.

In flight tests, adaptive control designs have been demonstrated for a midsize transport (NASA)[11] and in the X-36 (U.S. Air Force)[12] with unanticipated control failures. The NASA demonstration included a piloted simulation using only propulsion for backup flight control. Applicability of an emergency flight control system greatly increases if it can provide desirable responses over a wide range of unanticipated failures via adaptive control.

Recently, Ferrari and Stengel[13] developed a global controller, using ANN and an adaptive critic, that they have applied to control a full 6-DOF simulation of a business jet aircraft. Their controller was shown to be as conservative as traditional linear designs and as effective as the global controller. Wyeth et al.[14] used neural networks to generate hover command on a model helicopter. In their results, the mapping from training data was adequate but generalization for previously unseen data was not achieved. Sharma and Ward[15] developed a method for flight-path angle control using neuroadaptive backstepping. The simulation results showed that, on a 3-DOF model of a flying-wing unmanned air vehicle (UAV), the algorithms produced good command tracking and the neural networks were able to adapt to parametric errors in the plant dynamics. Idan et al.[16,17] used neural networks and pseudocontrol hedging methodology to design an adaptive flight controller that responds to faults in the system by utilizing redundancy in the controls. This type of approach would work well as a safety retrofit of any existing flight control system. KrishnaKumar and Gundy-Burlet[18] suggest that the recent developments in intelligent control offer the potential for improved handling qualities and significantly increasing survivability rates under various failure conditions. Schumann and Nelson[19] have discussed the certification, verification, and validation of such systems, which is challenging due to the difficulty in ensuring their correct behavior.

An operationally simplified (EZ-fly) flight control system already exists (modifications funded by NASA Advanced GA Transport Experiments [AGATE]) on a Raytheon Aircraft Company Bonanza F33C fly-by-wire testbed. This system is designed to follow pilot input as follows: Longitudinal stick position commands pitch angle or vertical flight-path angle, where centered stick commands level flight. Lateral stick position commands bank angle, and centered stick commands constant heading. A speed command lever commands airspeed, where stall speed plus 5 kn and never-exceed flight speed minus 10 kn is allowed. Plus or minus 7 deg is allowed in vertical flight-path angle, and lateral bank-angle control is limited to plus or minus 60 deg. Eventually, this system includes automatic turn coordination, a yaw damper, angle-of-attack limiter, bank-angle limiter, automatic overspeed limiter, and a load factor limiter. The system will also automatically lock to a differential global positioning system and approach and landing with no pilot action. The aircraft is controlled to hold straight and level if the pilot lets go of the stick, and if the pilot maneuvers into an approach capture zone and lets go of the stick, the airplane will execute the approach and land with no pilot input. The pilot can reestablish direct control of the airplane by moving the stick from the centered position.

In the current project, an adaptive nonlinear inverse controller has been designed for the Beech Bonanza F33C single engine GA aircraft. In the future, the new controller will be added to the existing system on the Bonanza F33C fly-by-wire testbed, which will then be used for test flights. The longitudinal flight controller was designed in the MATLAB®[20] Simulink environment with the ANN developed using the MATLAB Neural Network Toolbox. Nonlinear inverse control is used, and ANN are trained online to counteract modeling error. The ANN are also used to adapt to changing flying characteristics due to unanticipated control failures, or changing configurations of the airplane. Operationally simplified flight controls will allow the pilot to provide full attention to an emergency while the airplane maintains its current path, thus, reducing the cognitive effort required to operate the airplane safely. Of course,

flight control system failures might lead to different types of accidents, and these failures must be reduced to acceptable levels by proper analysis and certification of the new flight control system. Also the reliability and safety of such systems specifically require dependable integration of software and hardware.

II. Adaptive Inverse Longitudinal Control Design

The longitudinal aircraft model is based on the familiar nonlinear equations in the body fixed x and z axis directions and the pitching moment equation as detailed in the Appendix, Eqs. (A1– A3). The aircraft is a system with thrust and elevator deflections as inputs and the longitudinal state variables, pitch angle θ, forward velocity U, and vertical velocity W, as outputs along with the derived variables, angle of attack α and flight-path angle γ. This aircraft block is shown in Fig. 1 along with the inverse controller, the linear controllers, and the neural networks. This is the complete MATLAB Simulink model. The aircraft block uses the Simulink Aerospace 6-DOF block with a nonlinear drag polar and linear lift and pitching moment models. The lateral forces and moments are set to zero.

The two linear output feedback loops track pilot inputs of commanded velocity V_c and flight-path angle γ using a proportional controller for the velocity feedback loop and a proportional derivative controller for the angle feedback loop. The same controller architecture can be used to command the pitch angle θ, if desired, by replacing the gamma feedback signal with theta. The linear controllers output the commanded linear acceleration $\dot{V}_c$ and commanded angular acceleration $\ddot{\gamma}$ or $\ddot{\theta}$. These accelerations are inputs to the inverse controller, which calculates the required thrust and elevator deflections needed to produce these commanded accelerations. The inverse flight controller is derived in the Appendix in what amounts to an algebraic manipulation and inversion of the aircraft equations of motion (A10) and (A11) and gives the following equations describing the inverse controller calculations.

$$T_c = [1/\cos(\alpha + \phi_T)]\left[m\dot{V}_c + mg\sin\gamma + qS(C_D)_{\delta_e=0}\right] \tag{1}$$

$$\delta_{e_c} = \left[C_m(\delta_e)/C_{m_{\delta_e}}\right] = \left(I_{yy}/q S\bar{c} C_{m_{\delta_e}}\right)(\ddot{\gamma}_c \text{ or } \ddot{\theta}_c) - (C_m)_{\delta_e=0}/C_{m_{\delta_e}} + \left[d_T/qS\bar{c}\cos(\alpha+\phi_T)C_{m_{\delta_e}}\right] \times \left[m\dot{V}_c + mg\sin\gamma + qS(C_D)_{\delta_e=0}\right] \tag{2}$$

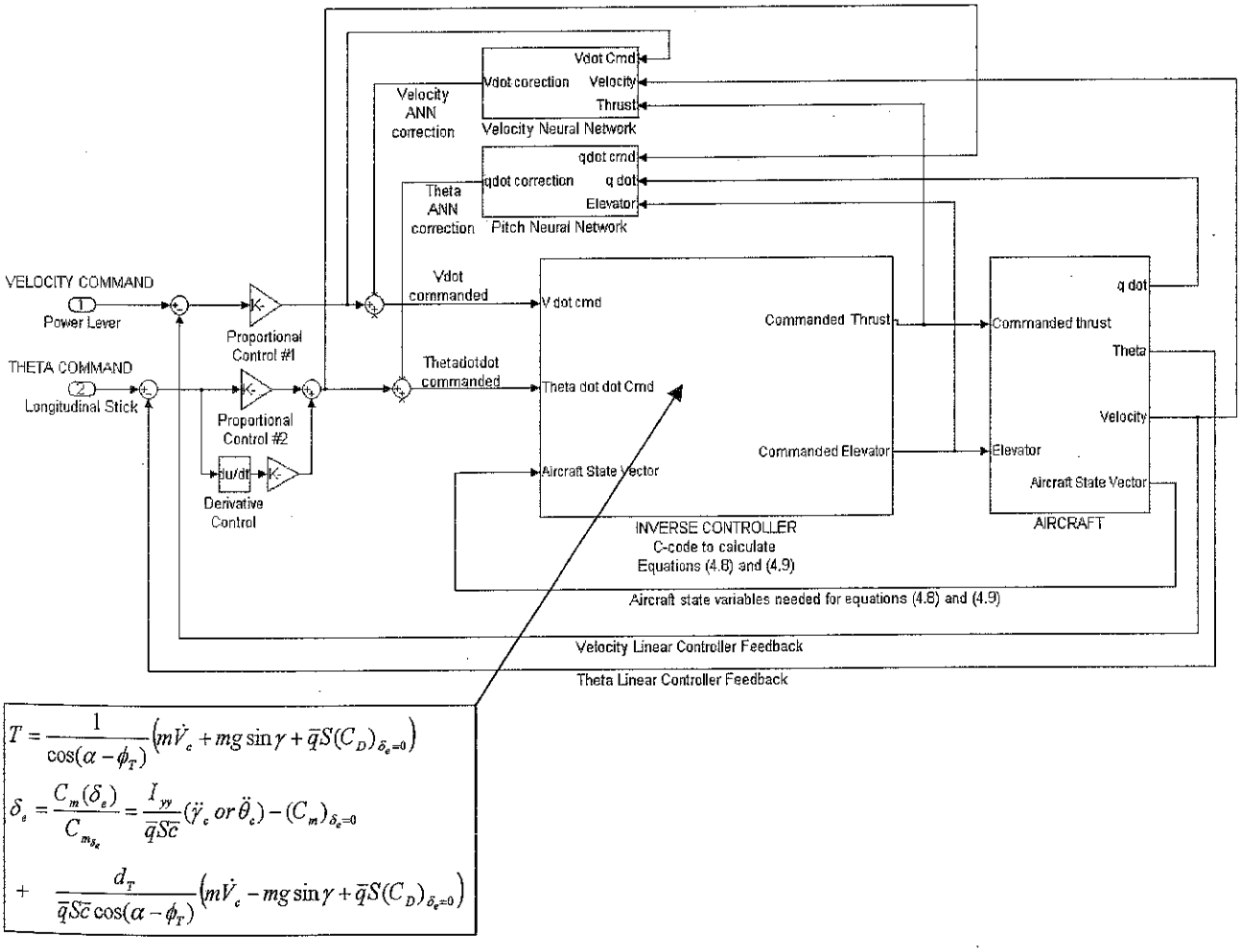
where

$$(C_D)_{\delta_e=0} = C_{D_0} + C_{D_K}\cdot\left(C_{L_0} + C_{L_\alpha}\alpha - C_{L_1}\right)^2$$

$$(C_M)_{\delta_e=0} = C_{M_0} + C_{M_\alpha}\alpha + C_{M_q}\hat{q}$$

In the cockpit, the decoupled controls are as follows. Longitudinal stick position commands the desired flight-path angle, which translates to vertical speed, and a speed control lever replaces the throttle and commands the desired flight speed.

When the inverse controller is added to the aircraft, the combination becomes, if there is no modeling error, 1) for gamma (or theta) tracking, a simple double integrator block ($1/s^2$ in the Laplace domain) that takes a commanded angular acceleration and outputs the resulting aircraft angular displacement, and 2) for the velocity tracking, a simple integrator block ($1/s$ in the Laplace domain) that takes a commanded velocity rate (acceleration) and outputs the resulting aircraft velocity. A proportional derivative (PD) controller is used in the theta or gamma tracking linear feedback loop. Theta tracking is straightforward because the theta acceleration is simply commanded in the inverse controller of Eq. (2). As indicated in this equation,



$$T = \frac{1}{\cos(\alpha-\phi_T)}\left(m\dot{V}_c + mg\sin\gamma + \bar{q}S(C_D)_{\delta_e=0}\right)$$

$$\delta_e = \frac{C_m(\delta_e)}{C_{m_{\delta_e}}} = \frac{I_{yy}}{\bar{q}S\bar{c}}(\ddot{\gamma}_c \text{ or } \ddot{\theta}_c) - (C_m)_{\delta_e=0} + \frac{d_T}{\bar{q}S\bar{c}\cos(\alpha-\phi_T)}\left(m\dot{V}_c - mg\sin\gamma + \bar{q}S(C_D)_{\delta_e=0}\right)$$

Fig. 1 Longitudinal controller block diagram.

to track gamma, theta acceleration is simply replaced by gamma acceleration. Implicit in this is the assumption that angle-of-attack acceleration is small because theta, gamma, and alpha are linearly related for longitudinal flight. For the gentle maneuvers expected of this flight control system, this has proven to be a valid assumption, based on plots of theta, gamma, and alpha accelerations. Also, the experience Noel Duerksen has had with the E-Z fly system on the Bonanza is that gamma tracks theta very closely. The parameters of the PD controller can be chosen to provide any second-order system response characteristics desired, that is, rise time overshoot damping ratio, etc., can be specified. However, because the elevator has deflection limits and its actuator has its own response time, there is a physical limit to how fast a rise time can actually be achieved and how much the faster short period of the aircraft can be damped. The theta or gamma response proportional, integral, derivative (PID) controller parameters were chosen to give a damping ratio of 0.9 and a natural frequency of 0.25 yielding a rise time of 5 s. A proportional controller is used in the velocity feedback loop because this is has a first-order system response. The constant of proportionality can be chosen to give any desired time constant for the aircraft response. Again, because the throttle has limits, there is a physical limit to how fast the aircraft can speed up or slow down. The time constant for the velocity response was chosen to be 5.0 s.

A note of caution should be mentioned if this work is extended to include turbulence, wind, and wind shear: The various terms in the inverse controller must be carefully examined as to whether they are inertial terms or air-mass velocity terms. For example, alpha and airspeed are air-mass referenced, whereas theta and commanded acceleration (velocity change) are inertial referenced.

III. ANN for Flight Controller Error Compensation

In the longitudinal flight controller, two ANN are used to compensate for modeling errors, partial actuator failure, and other changes in flight characteristics. There are separate networks for pitch angle control and for flight velocity control. A neural network is a computational device modeled loosely after biological nervous systems.[21] They are composed of artificial neurons connected to each other by weight values. The network processes input signals through the connectivity of the neurons to produce output values at the output neurons. These networks learn input–output relationships from example data through various training methods that modify the connecting weights to produce an approximation to the input–output mapping represented by the data. The ANN Simulink block containing the velocity ANN is shown in Fig. 2. This neural network takes as input the commanded acceleration $\dot{V}_c(t)$ from the linear controller and outputs the error correction $e(t)$ that will be needed to be added to $\dot{V}_c(t)$ to make the actual aircraft acceleration $\dot{V}(t)$ match the commanded $\dot{V}_c(t)$. Note, if the inverse controller is an exact inverse of the aircraft, the neural network will learn to output an error correction signal of zero. At every time step, the ANN Simulink block receives the following information: The current commanded acceleration from the PD controller, $\dot{V}_c(t)$, the error compensation signal $e(t-1)$ used one time step ago, and the actual acceleration of the aircraft one time step ago, $\dot{V}(t-1)$. The ANN Simulink block contains two functions: 1) training the network and 2) computing the error compensation signal. The ANN contained in this block has one input and one output. For training, $\dot{V}(t-1)$ is used as the input and the target output is $e(t-1)$. There is a correlation between these two signals because the aircraft velocity response $\dot{V}(t-1)$ is a result of using that error compensation signal at $e(t-1)$. Then, the ANN that is continually trained to perform this mapping is used to calculate the error compensation signal used in the controller for the current time step. The current commanded acceleration $\dot{V}_c(t)$ is assigned as the input to the ANN, and the network output is the compensation signal needed for this time step. The architecture of the neural network was developed by testing numerous different designs and choosing one that was sufficiently robust and that provided fast adaptation to modeling errors and control failures. The need for the derivative of airspeed used as an input to the network may cause some difficulty in the event of turbulence or sensor noise. If the disturbance is random with zero mean, then, in general, learning algorithms have been shown to be robust in filtering this out, when the learning rate is set properly. Of course, this remains to be shown for this application.

This neural network contains two hidden layers with five neurons in each hidden layer. The first hidden layer consists of neurons with tan–sigmoid activation functions. The second hidden layer and the output neuron have linear activation functions. The training technique used is the Levenberg–Marquardt algorithm with a learning rate of 0.7. The ANN used for pitch angle control is identical in structure to the velocity controller network. The linear acceleration (velocity rate) values are simply replaced with pitch angle acceleration. Figure 3 shows the details of the neural network structure. Each layer takes an input vector, multiplies it by a weight vector, adds a bias vector, and then processes each element of the resulting vector through the activation function show. This result is then passed on to the next layer, or if it is the final layer, this becomes the network output. The weight and bias vectors for each layer are the parameters that are updated by the training algorithm. The training algorithm, at each simulation time step, compares the network output with the correct output fed to the network block. This gives an error signal that is used to adjust the weight and bias vectors a small amount in a direction that makes the error smaller. The key is that the weights change only a small amount with each training step, but over many time steps, the weights are changed in directions that makes the output converge to give the correct answer.

IV. Longitudinal Controller Simulation Results

The adaptive inverse controller was tested in MATLAB Simulink with a Bonanza model in trimmed flight and with speed and pitch angle commands, both with modeling errors and partial control system failures. The Simulink model is shown in Fig. 1. The ANN were able to start immediately adapting to sudden changes in flight characteristics and during partial elevator or engine failure.

Figures 4–9 show the response of the Bonanza in trimmed flight, where the aircraft C_{m_α} drops suddenly by 10% at 5 s and the

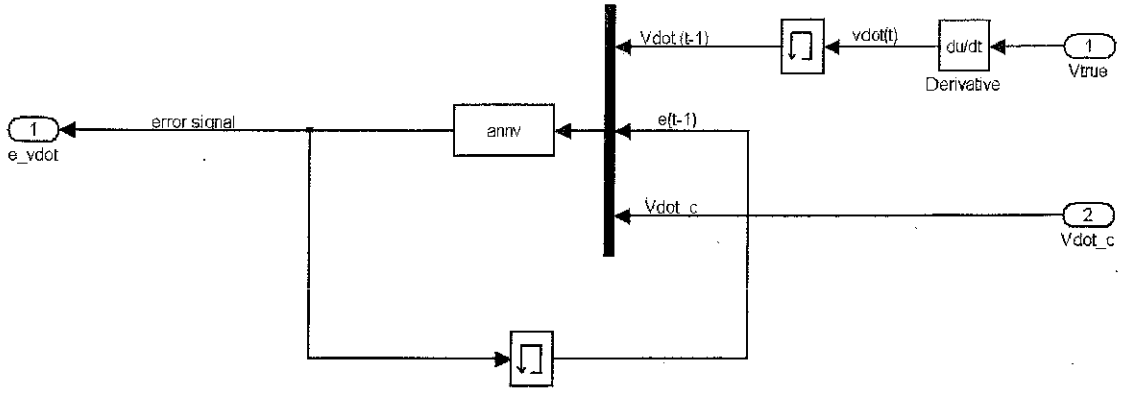


Fig. 2 ANN function for velocity control corrections.

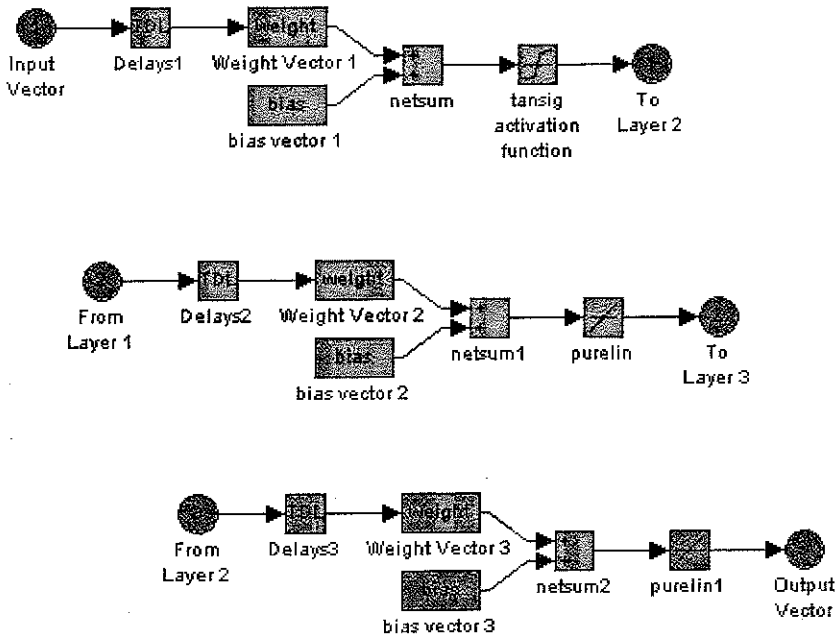


Fig. 3 Neural network architecture details.

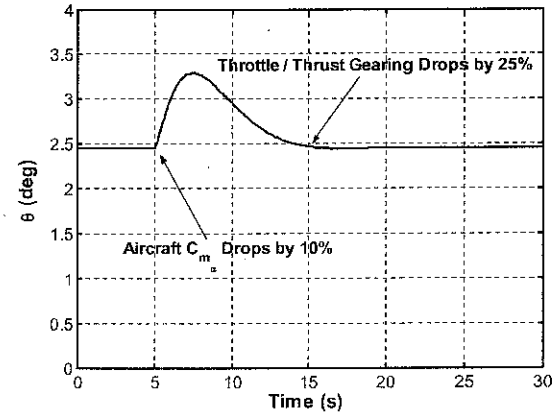


Fig. 4 Pitch angle θ for trimmed flight with unanticipated failures.

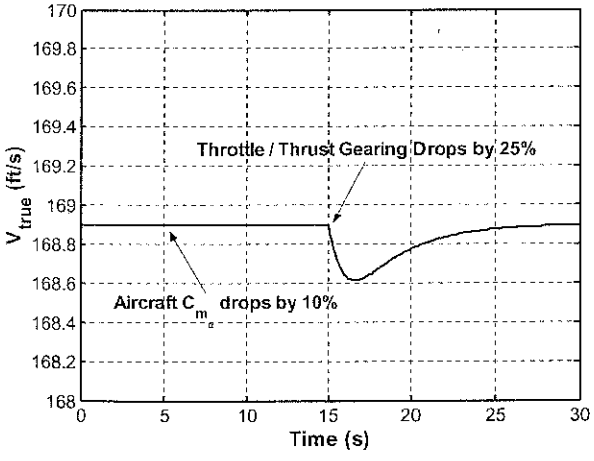


Fig. 5 Velocity V_{true} for trimmed flight with unanticipated failures.

throttle/thrust gearing drops suddenly by 25% at 15 s, that is, the actual engine thrust drops to 75% of the commanded thrust. Note in Fig. 7 and subsequent thrust plots that the commanded thrust shown is the thrust the inverse controller is commanding, not the actual thrust delivered by the engine. After the C_{m_α} change occurs, the pitch angle varies by less than 1 deg during the recovery that takes about 10 s. Figure 8 shows that the neural network starts instantly feeding in an error correction signal when the C_{m_α} change occurs.

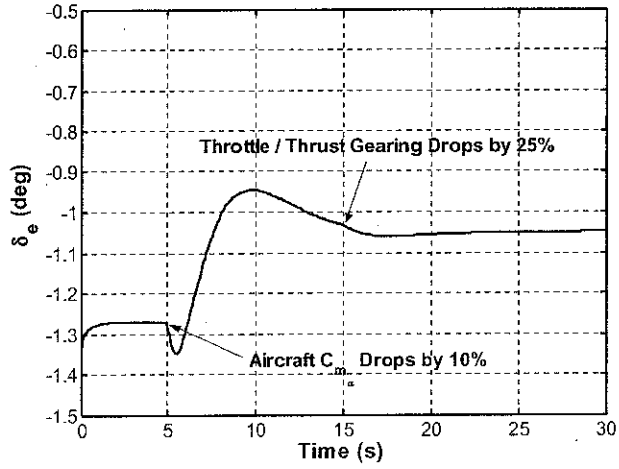


Fig. 6 Commanded elevator position δ_e for trimmed flight with unanticipated failures.

With the aid of the ANN, the aircraft response matches the commanded value again within 1-s time. The necessary change in the elevator command is shown in Fig. 6. Similarly, the throttle gearing change causes only a 0.3 ft/s change in velocity (Fig. 5) with the ANN compensation shown in Fig. 9. Additional tests showed that the ANN allows for a C_{m_α} change of up to 40% with rapid recovery and without notable oscillation.

Figures 10–13 show the response of the Bonanza in controlled flight, with the same failures as earlier (where the aircraft C_{m_α} drops suddenly by 10% at 5 s and the throttle/thrust gearing drops suddenly by 25% at 15 s). While these changes are occurring, the pilot commands a 5-deg increase in pitch angle at 8 s and a 20-ft/s increase in flight speed at 20 s. The ANN make the necessary corrections with similar robustness and accuracy as in the test for the trimmed flight condition.

Figures 14–17 show the response of the Bonanza, when half of the elevator becomes ineffective (breaks off) at 0 s, that is $C_{m_{\delta_e}}$ and $C_{L_{\delta_e}}$ are reduced by 50%. Here, any lateral directional effects of an asymmetrical elevator are neglected. Figures 14–17 show that the ANN make the necessary corrections and that the pitch angle varies by less than 3 deg. The aircraft returns to the trimmed state in 60 s. Some initial oscillation of other variables occurs, but the amplitudes are very small. The lack of velocity perturbation with the elevator failure is because a change in elevator effectiveness does not significantly

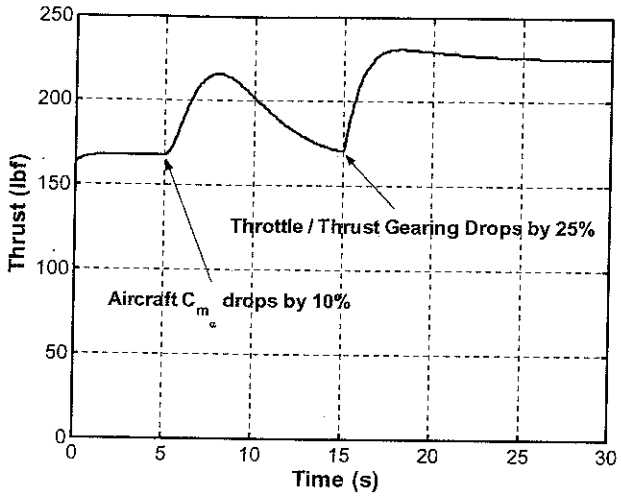


Fig. 7 Commanded thrust (throttle) for trimmed flight with unanticipated failures.

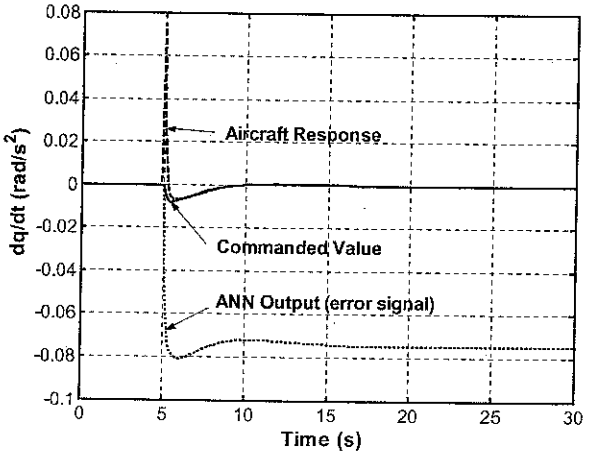


Fig. 8 Pitch acceleration dq/dt for trimmed flight with unanticipated failures.

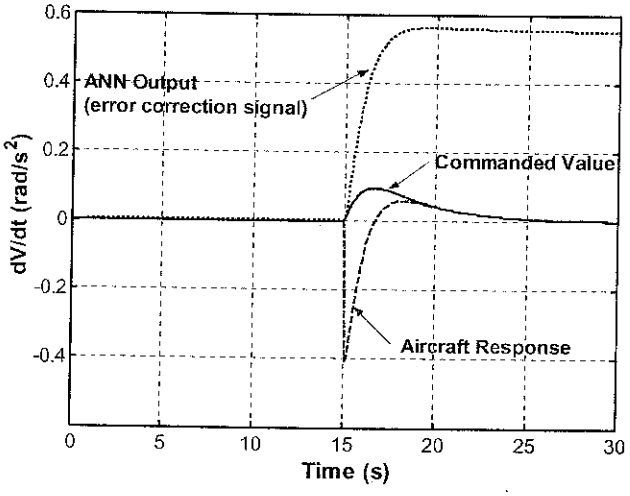


Fig. 9 Aircraft acceleration dV/dt for trimmed flight with unanticipated failures.

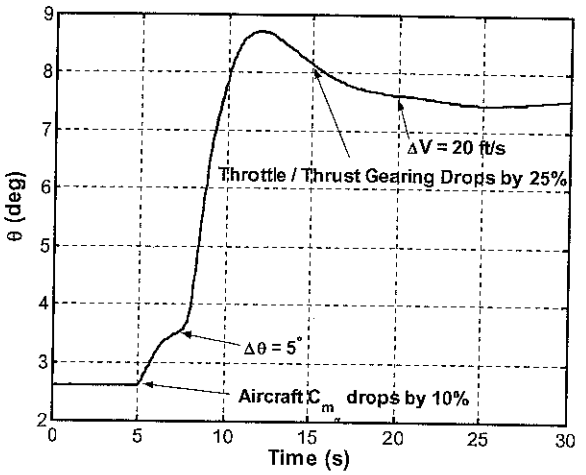


Fig. 10 Pitch angle θ for pilot inputs with unanticipated failures.

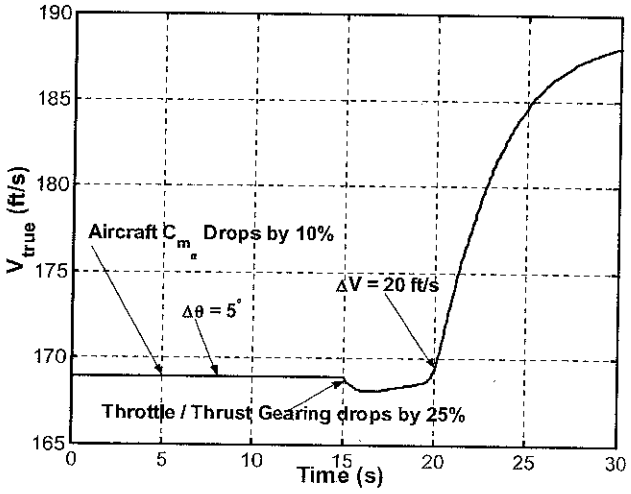


Fig. 11 Velocity V_{true} for pilot inputs with unanticipated failures.

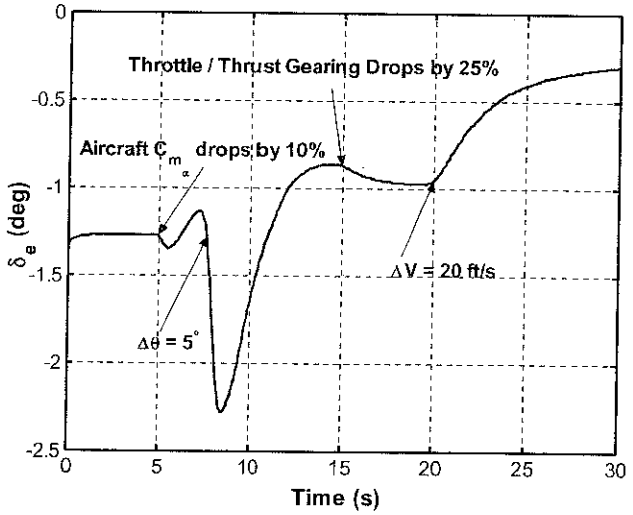


Fig. 12 Commanded elevator position, δ_e, for pilot inputs with unanticipated failures.

affect the drag equation, which is solved for thrust so there is no significant modeling error in the velocity control loop. The pitch equation is greatly affected; therefore, a large pitch transient occurs because there is a large modeling error in the inverse control calculated for the elevator. During the pitch transient, the neural network is learning to compensate for this modeling error successfully.

Gamma tracking is achieved with as much success as theta tracking. Figures 18–22 show the response to a commanded level flight-path angle of 0 deg and a commanded decrease in flight velocity of 70 ft/s at 3 s into the simulation with the following failures occurring: The aircraft C_{m_α} drops suddenly by 10% at 20 s and the throttle/thrust gearing drops suddenly by 25% at 25 s. Notice that the neural networks learn to compensate for the failures while tracking the commanded flight path and velocity. When the throttle saturates at its minimum value, the elevator controller continues to maintain the pitch angle. The velocity controller is calculating negative

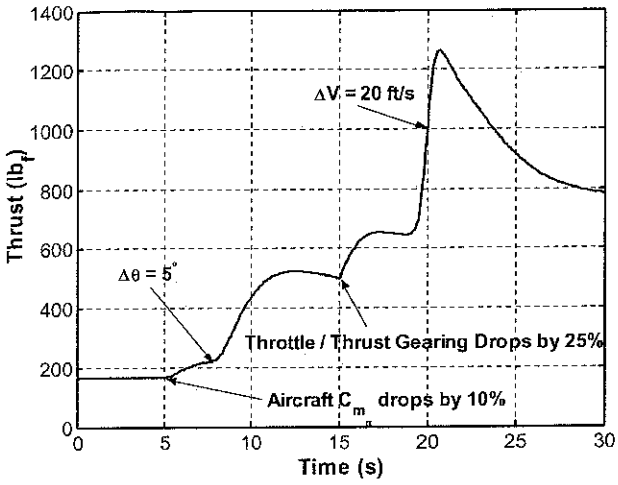


Fig. 13 Commanded thrust (throttle) for pilot inputs with unanticipated failures.

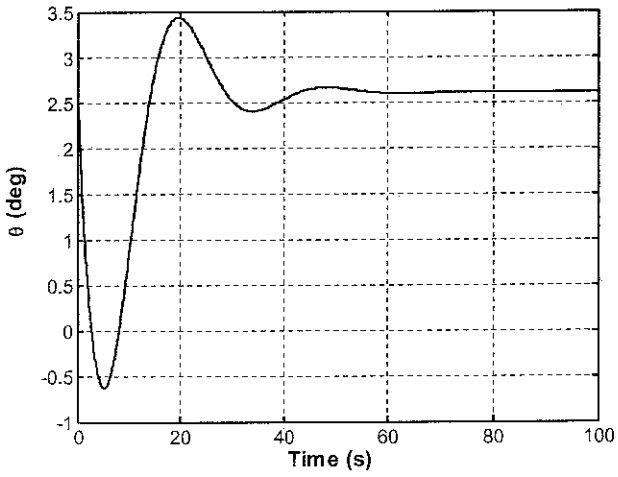


Fig. 14 Pitch angle θ for 50% loss of elevator effectiveness.

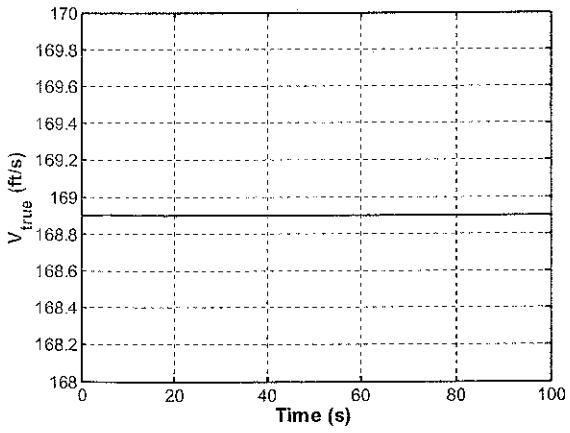


Fig. 15 Velocity V_{true} for 50% loss of elevator effectiveness.

commanded thrust correctly, but because the aircraft cannot respond, due to the thrust saturation, the response is essentially open loop in velocity control. As soon as the thrust becomes unsaturated, however, the velocity control becomes effective again and tracks the velocity command.

Windup effects due to the throttle saturation have been observed in the early design without windup compensation. When the throttle saturates, the neural network sees the aircraft not responding to the commanded throttle. (The controller is commanding a negative

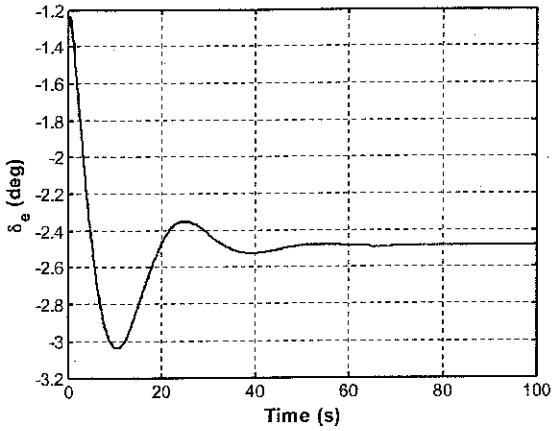


Fig. 16 Commanded elevator position, δ_e, for 50% loss of elevator effectiveness.

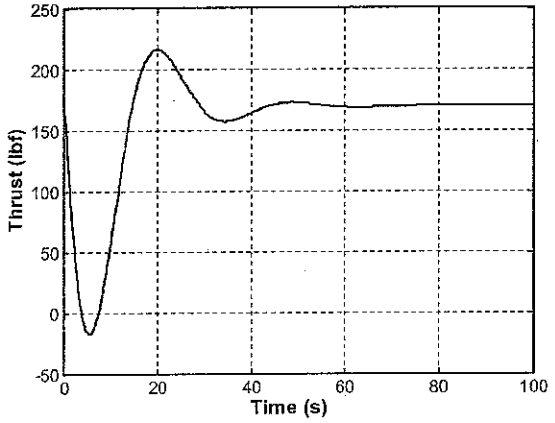


Fig. 17 Commanded thrust (throttle) for 50% loss of elevator effectiveness.

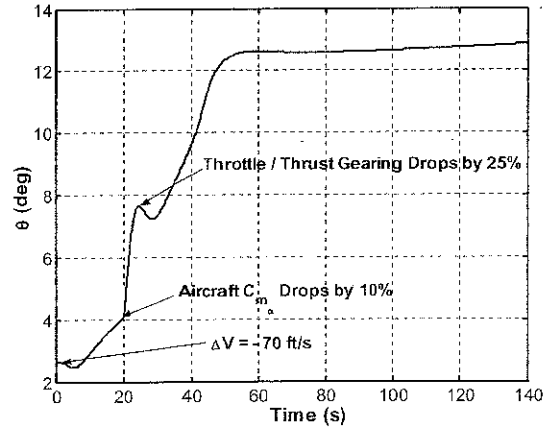


Fig. 18 Pitch angle θ for gamma tracking with throttle and C_{m_α} failures.

throttle to slow the aircraft down.) The network sees this as a modeling error and adapts to try to compensate for it, in much the same way as traditional integral control will windup trying to compensate for a steady-state error in the presence of control saturation. The network cannot compensate because the throttle is saturated, and the network weights train to erroneously huge values (similar to integral windup) that cause large speed changes when the throttle does finally unsaturate. Therefore, an antiwindup method was implemented in the final design. The antiwindup method used for the neural network was to stop training when the control associated with that neural network was saturated. When the control became unsaturated, then the neural network began to train again from where it left

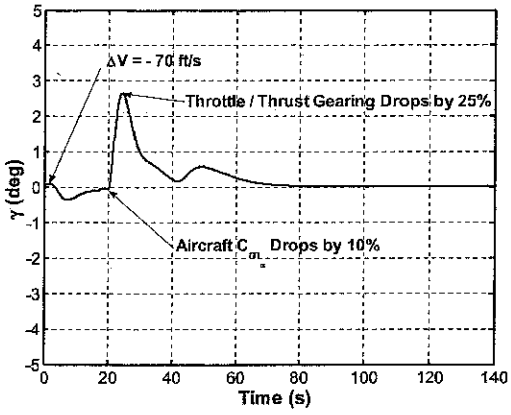


Fig. 19 Flight-path angle γ for gamma tracking with throttle and C_{m_α} failures.

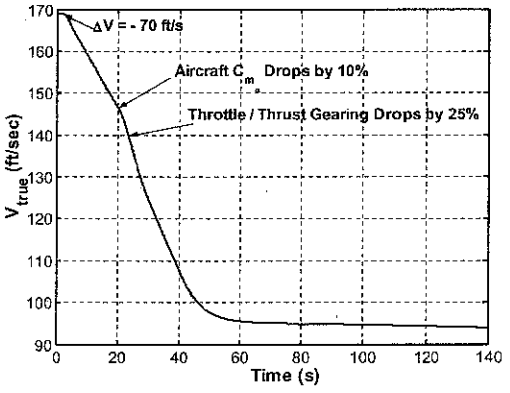


Fig. 20 Velocity V_{true} for gamma tracking with throttle and C_{m_α} failures.

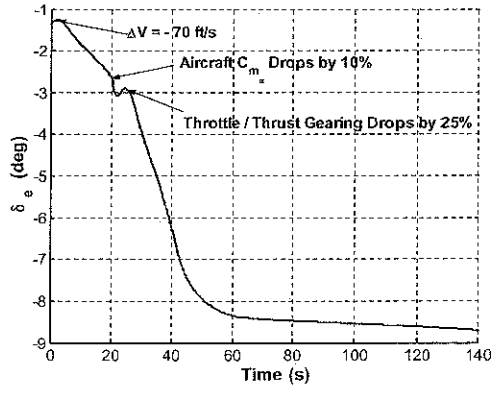


Fig. 21 Commanded elevator position, δ_e, for gamma tracking with throttle and C_{m_α} failures.

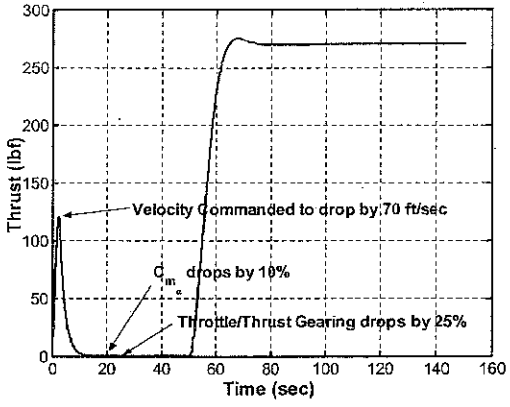


Fig. 22 Commanded thrust (throttle) for gamma tracking with throttle and C_{m_α} failures.

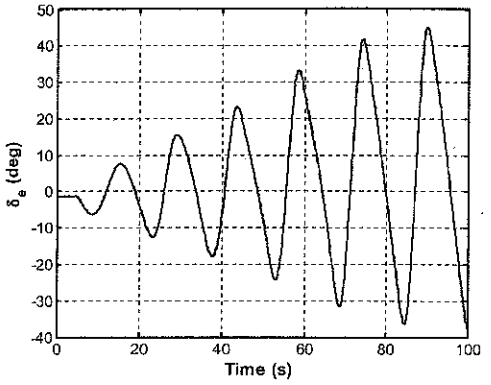


Fig. 23 Commanded elevator angle δ_e with throttle and C_{m_α} failures without the aid of ANN.

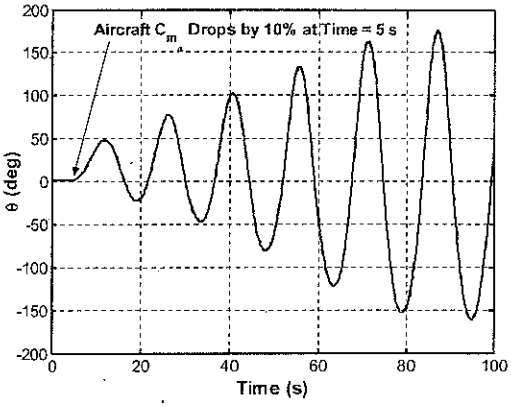


Fig. 24 Aircraft pitch angle θ with throttle and C_{m_α} failures without the aid of ANN.

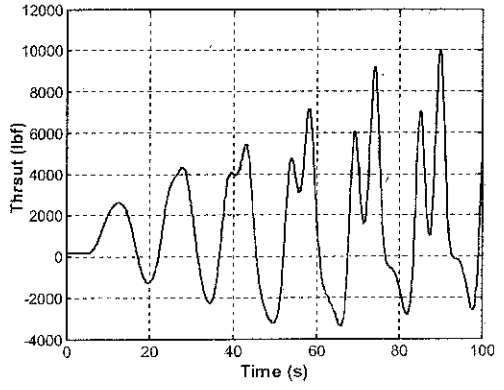


Fig. 25 Commanded thrust (throttle) with throttle and C_{m_α} failures without the aid of ANN.

off with no reinitialization. Rysdyk and Agarwal[7] have investigated a more elegant antiwindup method called pseudocontrol hedging, which limits the commanded accelerations to prevent control saturation using a model-following controller. This has not been implemented in our controller, but should be investigated extensively in the near future. For comparison, Figs. 23–26 show the response of the aircraft, with failures occurring, but, without the assistance of the neural networks. Figures 23–26 show clearly that without the ANN the inverse controller is not capable of handling such failures or changes in aircraft characteristics.

V. Longitudinal Controller Simulation and Flight Test

The longitudinal controller has been tested initially in MATLAB environment. The controller is currently in the process of being implemented as C-code (ANSI-C format) on the actual aircraft. The nonlinear adaptive control architecture will be incorporated into the

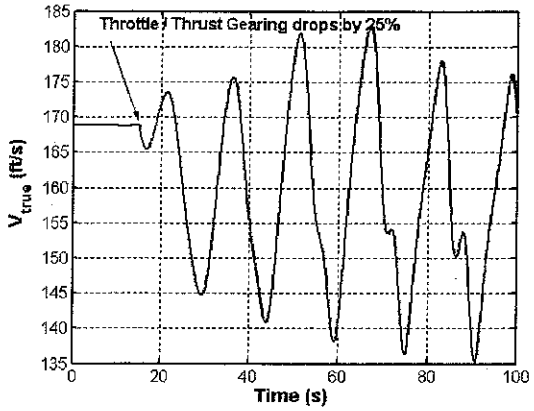


Fig. 26 Aircraft velocity V_{true} with throttle and C_{m_α} failures without the aid of ANN.

existing Raytheon Bonanza Velocity Vector Command with Envelope Protection control system (EZ-fly developed by Lambregts[22,23] and modified and implemented on the Bonanza by Duerksen). The Beech Bonanza F33C fly-by-wire testbed is a development platform for advanced controls, advanced displays, flight controls integrated with pilot fault display and multifunction display, attitude and heading reference system/air data computer, and Datalink.

The MATLAB Simulink system development environment was used to develop and test the adaptive inverse controller. The Real-Time-Workshop toolbox will be used to dump embedded system-C code directly from the Simulink control system model. This code will then be compiled as a subroutine embedded in the existing Bonanza F33C flight computer EZ-fly software. This system replaces portions of two subroutines in the EZ-fly software that calculate engine throttle position and elevator position that will be output from the flight computer to command engine throttle and elevator actuators.

Flight testing on the Raytheon Bonanza is currently underway. The following goals are set for flight testing: Demonstrate velocity vector command with nonlinear adaptive control architecture, demonstrate adaptive control effectiveness with changing configuration and payloads, and investigate the system performance in the presence of external disturbances such as atmospheric turbulence.

VI. Conclusions

The presented ANN approach was proven to be very effective in designing an adaptive inverse controller. Changes in C_{m_α} of up to 40% and even the loss of half of the elevator effectiveness were successfully compensated for and would allow the pilot to continue safely controlling the aircraft without any adjustments. The next phase of the project is to demonstrate the extension of this method to lateral directional control by implementing in Simulink the inverse controller for the full 6-DOF model of the aircraft. Flight testing on the Raytheon fly-by-wire testbed Bonanza F33C will follow this. Although actuator failure has been demonstrated, more work is required to demonstrate compensation for sensor failure, as well as switching to backup redundant control when total failure of a primary control surface occurs.

When this type of a safe and operationally simplified control system is eventually applied to GA aircraft, the following issues will have to be considered: A different type of pilot license is envisioned that takes advantage of the increased accessibility afforded by an operationally simplified control system combined with the appropriate display. This type of control system lends itself to automatic approach and landing in conditions as low as zero ceiling and just enough visibility to taxi off of the runway to parking. Speed limits can be imposed through the flight control system to limit aircraft performance in areas where airplanes are typically faster than pilots think, for example, around airports. Instrument-flight rules clearances with tighter tolerances in space and time can be issued and expected to be followed, which will allow tighter spacing of aircraft around airports while improving safety. An Equivalent Level of Safety standard needs to be developed that can replace the current regulations for operationally simplified flight control airplanes. The following issues also need to be examined: 1) software development/assurance, 2) acceptable command tracking characteristics (rise time, overshoot, etc.), 3) acceptable ranges of control motion and forces, 4) minimum aircraft response capabilities, 5) degraded performance requirements after failures, 6) reliability requirements, 7) envelope protection guidelines, 8) operating rules, and 9) piloting requirements.

Appendix: Longitudinal Inverse Controller Derivation

The derivation of the inverse controller for the longitudinal motion involves specifying the commanded aircraft accelerations and solving for the control settings required for achieving these accelerations. This is opposite of the normal forward aircraft model where the control settings are specified and the resulting motion of the aircraft is solved for.

In body coordinates, the equations of motion for longitudinal maneuvers with the wings level are[24]

$$\dot{U} + g\sin(\theta) = -WQ + (1/m)F_{A_x} + (1/m)T\cos(\varphi_T) + (1/m)F_{X_{\delta_e}}\delta_e \tag{A1}$$

$$\dot{W} - g\cos(\theta) = +UQ + (1/m)F_{A_z} - (1/m)T\sin(\varphi_T) + (1/m)F_{Z_{\delta_e}}\delta_e \tag{A2}$$

$$\ddot{\theta} = (1/I_{yy})M_A - (1/I_{yy})Td_T + (1/I_{yy})M_{\delta_e}\delta_e \tag{A3}$$

where d_T is the thrust moment arm and φ_T is the thrust angle relative to the fuselage axis. In addition, the true airspeed and flight path angle γ are defined as

$$V_p = \sqrt{U^2 + W^2} \tag{A4}$$

$$\theta = \gamma + \tan^{-1}(W/U) \tag{A5}$$

An output feedback controller is set up to track the flight-path angle (or pitch angle) and the flight speed. The tracking error for each is fed to a PID controller that outputs commanded accelerations $\dot{V}_p$ and $\ddot{\gamma}$ that are inputs to the inverse controller. Therefore, given $\dot{V}_p$ and $\ddot{\gamma}$ or $\ddot{\theta}$ find θ, T, δ_e, U, and W. This constitutes a mathematically well-posed problem with five equations and five unknown parameters. The following three statements constitute auxiliary equations that go along with the preceding equations:

$$Q = \dot{\theta} \tag{A6}$$

$$\dot{\theta} = \dot{\gamma} + \dot{\alpha} = \dot{\gamma} + (\dot{W}U - \dot{U}W)/V_p^2 \tag{A7}$$

$$\dot{V}_p V_p = \dot{U}U + \dot{W}W \tag{A8}$$

Equations (A1–A3) are in the body-fixed axis system (X,Z). The commanded forward acceleration $\dot{V}_p$ and commanded normal acceleration resulting in a flight-path angular acceleration $\ddot{\gamma}$ are in an axis system parallel and normal to the velocity vector (stability coordinates); therefore, Eqs. (A1) and (A2) are transformed into the V_p and γ coordinate system.

Multiply Eq. (A1) by (W/V_p^2), Eq. (A2) by (U/V_p^2), and subtract Eq. (A1) from Eq. (A2):

$$(U\dot{W} - W\dot{U})/V_p^2 - (g/V_p)\cos(\theta - \alpha) = Q + (1/mV_p)\left[F_{A_z}\cos(\alpha) - F_{A_x}\sin(\alpha)\right] - (T/mV_p)\sin(\alpha + \varphi_T) + (1/mV_p) \times \left[F_{Z_{\delta_e}}\cos(\alpha) - F_{X_{\delta_e}}\sin(\alpha)\right]\delta_e$$

Note that

$$\cos(\theta - \alpha) = \cos(\gamma), \qquad Q = \dot{\gamma} + \dot{\alpha}$$

$$F_{A_z}\cos(\alpha) - F_{A_x}\sin(\alpha) = -qSC_L|_{\delta_e=0}$$

$$F_{Z_{\delta_e}}\cos(\alpha) - F_{X_{\delta_e}}\sin(\alpha) = -qSC_{L_{\delta_e}}$$

Therefore, rearranging and simplifying results in

$$\dot{\gamma} + (g/V_p)\cos(\gamma) - (qS/mV_p)C_L|_{\delta_e=0} - (T/mV_p)\sin(\alpha+\varphi_T) - (qS/mV_p)C_{L_{\delta_e}}\delta_e = 0 \tag{A9}$$

In the inverse problem, Eq. (A9) involves three unknowns, α, T, and δ_e. We need two other equations with the same unknowns.

Now, multiply Eq. (A1) by (U/V_p^2) and Eq. (A2) by (W/V_p^2) and add

$$(\dot{V}_p/V_p) - (g/V_p)\sin(\theta-\alpha) = (1/mV_p)\left[F_{A_z}\sin(\alpha) + F_{A_x}\cos(\alpha)\right] + (T/mV_p)\cos(\alpha+\varphi_T) + (1/mV_p) \times \left[F_{Z_{\delta_e}}\sin(\alpha) + F_{X_{\delta_e}}\cos(\alpha)\right]\delta_e$$

Note that

$$F_{A_z}\sin(\alpha) + F_{A_x}\cos(\alpha) = -qSC_D|_{\delta_e=0}$$

$$F_{Z_{\delta_e}}\sin(\alpha) + F_{X_{\delta_e}}\cos(\alpha) = -qSC_{D_{\delta_e}}$$

Then, again rearranging and simplifying results in

$$-\dot{V}_p - g\sin(\gamma) - (qS/m)C_D|_{\delta_e=0} + (T/m)\cos(\alpha+\varphi_T) - (qS/m)C_{D_{\delta_e}}\delta_e = 0 \tag{A10}$$

From Eq. (A3), because $\ddot{\theta} = \ddot{\gamma} + \ddot{\alpha}$,

$$\ddot{\alpha} + \ddot{\gamma} = \ddot{\theta} = (qS\bar{c}/I_{yy})\left[C_m - C_T(d_T/\bar{c}) + C_{m_{\delta_e}}\delta_e\right] \tag{A11}$$

A speed control lever (replacing the cockpit throttle) is set to command airspeed V_F. When this is compared with the feedback of actual airspeed, a linear proportional controller that outputs a commanded acceleration processes the resulting error $\dot{V}_p$. Similarly a longitudinal stick can be set up to commands flight-path angle γ or pitch angle θ. The tracking error of either of these variables is fed to a PD controller that outputs a commanded angular acceleration $\ddot{\theta}$ or $\ddot{\gamma}$. Then Eqs. (A10) and (A11) can be solved simultaneously for the control variables T and δ_e. Other variables can be determined from the auxiliary equations.

An approximate solution can be obtained if one assumes $C_{D_{\delta_e}} \approx 0$. Then Eq. (A10) can be solved for T, from which Eq. (A11) can be solved for δ_e. Note that these equations are linear in terms of the controls, as long as control saturation does not occur.

As mentioned in Sec. IV, a note of caution must be inserted related to when this work is extended to include turbulence, wind, and wind shear. The various terms in the inverse controller must be carefully examined as to whether they are inertial terms or air mass velocity terms. For example, alpha and airspeed are air mass referenced, whereas theta and commanded acceleration (velocity change) are inertial referenced.

Acknowledgment

This work was sponsored in part by the Federal Aviation Administration, Grant Request 00-C-WSU-00-14.

References

[1]Brinker, J., and Wise, K., "Stability and Flying Qualities Robustness of a Dynamic Inversion Aircraft Control Law," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 6, 1996, pp. 1270–1277.

[2]McFarland, M., and Calise, A., "Robust Adaptive Control of Uncertain Nonlinear Systems Using Neural Networks," *IEEE Transactions on Automatic Control*, July 1997, pp. 1996–2000.

[3]McFarland, M., and Calise, A., "Adaptive Nonlinear Control of Agile Antiair Missiles Using Neural Networks," *IEEE Transactions on Control Systems Technology*, Vol. 8, No. 5, 2000, pp. 749–756.

[4]Rysdyk, R., Nardi, F., and Calise, A., "Robust Adaptive Nonlinear Flight Control Applications Using Neural Networks," *Proceedings of the American Control Conference*, American Automatic Control Council, Dayton, OH, pp. 2595–2599.

[5]Rysdyk, R., "Adaptive Nonlinear Flight Control," Ph.D. Dissertation, Dept. of Aerospace Engineering, Georgia Inst. of Technology, Atlanta, GA, Nov. 1998.

[6]Rysdyk, R., Calise, A., and Chen, R., "Nonlinear Adaptive Control of Tiltrotor Aircraft Using Neural Networks," World Aviation Congress, Paper 975613, Oct. 1997.

[7]Rysdyk, R., and Agarwal, R., "Nonlinear Adaptive Flight Path and Speed Control Using Energy Principles," AIAA 2002-4440, Aug. 2002.

[8]Soloway, D., and Haley, P., "Aircraft Reconfiguration Using Neural Generalized Predictive Control," *American Control Conference Proceedings*, Vol. 4, American Automatic Control Council, Dayton, OH, pp. 2924–2929.

[9]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.

[10]Lewis, F., Liu, K., and Yesildirek, A., "Neural Net Robot Controller with Guaranteed Tracking Performance," *IEEE Transactions on Neural Networks*, Vol. 6, No. 3, 1995, pp. 703–715.

[11]Kaneshige, J., and Gundy-Burlet, K., "Integrated Neural Flight and Propulsion Control System," AIAA Paper 2001-4386, Aug. 2001.

[12]Sharma, M., Calise, P., and Seungjae, L., "Development of a Reconfigurable Flight Control Law for the X-36 Tailless Fighter Aircraft," AIAA Paper 2001-3940, Aug. 2001.

[13]Ferrari, S., and Stengel, R. F. "An Adaptive Critic Global Controller," *Proceedings of the American Control Conference*, American Automatic Control Council, Dayton, OH, 2002.

[14]Wyeth, G. F., Buskey, G., and Roberts, J., "Flight Control Using an Artificial Neural Network," *Proceedings of the Australian Conference on Robotics and Automation* (ACRA 2000), Australian Robotics and Automation Association, Sydney, Australia, 2000, pp. 65–70.

[15]Sharma, M., and Ward, D., "Flight-Path Angle Control via Neuro-Adaptive Backstepping," *Proceedings of the American Control Conference*, American Automatic Control Council, Dayton, OH, 2002.

[16]Idan, M., Johnson, M., and Calise, A. J., "A Hierarchical Approach to Adaptive Control for Improved Flight Safety," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 6, 2002, p. 1012.

[17]Idan, M., Johnson, M. D., Calise, A. J., and Kaneshige, J., "Intelligent Aerodynamic/Propulsion Flight Control For Flight Safety: A Nonlinear Adaptive Approach," *Proceedings of the American Control Conference*, American Automatic Control Council, Dayton, OH, 2001.

[18]Gundy-Burlet, K., KrishnaKumar, K., Limes, G., and Bryant, D., "Control Reallocation Strategies for Damage Adaptation in Transport Class Aircraft," AIAA Paper 2003-5642, Aug. 2003.

[19]Schumann, J., and Nelson, S., "Toward V&V of Neural Network Based Controllers," *Proceedings of the Workshop on Self-Healing Systems*, Assoc. of Computing Machinery, New York, 2002, pp. 67–72.

[20]*MATLAB Reference Guide*, Math Works, Inc., Natick, MA, Aug. 1992.

[21]Faussett, L., *Fundamentals of Neural Networks*, Prentice–Hall, Upper Saddle River, NJ, 1994.

[22]Lambregts, T., "Vertical Flight Path and Speed Autopilot Design Using Total Energy Principles," AIAA Paper 83-2239, Aug. 1983.

[23]Lambregts, T., "Operational Aspects of the Integrated Vertical Flight Path and Speed Control System," Society of Automotive Engineers, SAE Paper 831420, Warrendale, PA, Oct. 1983.

[24]Roskam, J., *Airplane Flight Dynamics and Automatic Flight Controls*, 3rd ed., Pt. 1 DARcorp., Lawrence, KS, 2001.