JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 24, No. 4, July–August 2001

Comparison of Intelligent, Adaptive, and Nonlinear Flight Control Laws

Marc L. Steinberg*
Naval Air Systems Command, Patuxent River, Maryland 20670

Seven different nonlinear control laws for multiaxis control of a high-performance aircraft are compared in simulation. The control law approaches are fuzzy logic control, backstepping adaptive control, neural network augmented control, variable structure control, and indirect adaptive versions of model predictive control and dynamic inversion. In addition, a more conventional scheduled dynamic inversion control law is used as a baseline. In some of the cases, a stochastic genetic algorithm was used to optimize fixed parameters during design. The control laws are demonstrated on a six-degree-of-freedom simulation with nonlinear aerodynamic and engine models, actuator models with position and rate saturations, and turbulence. Simulation results include a variety of single- and multiple-axis maneuvers in normal operation and with failures or damage. The specific failure and damage cases that are examined include single and multiple lost surfaces, actuator hardovers, and an oscillating stabilator case. There are also substantial differences between the control law design and simulation models, which are used to demonstrate some robustness aspects of the different control laws.

Introduction

THE last decade has seen substantial advances in nonlinear control due to both theoretical achievements and the availability of powerful computer hardware and user-friendly nonlinear simulation software. Whereas nonlinear control approaches other than gain scheduling have not been commonly used on aircraft, there have been many research efforts that have produced simulation results. In a few noteworthy cases, nonlinear control algorithms have been flown on test aircraft[1,2] or used in limited ways on production aircraft.[3] Despite all of this work, it can be difficult to judge the relative value of different nonlinear control approaches for any specific flight control problem. Even for a single vehicle, nonlinear flight control laws may have widely varying performance depending on the class of inputs and the desired flight envelope. Nonlinear controllers have been known to demonstrate spectacular results on simplified aircraft models, but then display pathological responses when applied to higher fidelity simulation models. Actuator nonlinearities have been a significant problem because many nonlinear control approaches tend to generate large effector commands or rates and have poor performance when effectors become saturated. Even without actuator saturations, the issue of control power requirements is of importance for the acceptance of nonlinear flight control. Given the penalties involved with increasing control power or rate requirements on new aircraft designs, there is some reluctance to use any control law where it is not reasonably clear that every bit of effector command serves a useful and well-understood purpose. Other key areas of concern with nonlinear control laws include the ease with which designs can be tuned to incorporate pilot feedback, configuration changes, and so on, and the ease with which designs can be analyzed and validated for safety of flight.

The purpose of the work described in this paper is to compare seven different nonlinear control approaches on an aircraft problem with some of the complexities of a real flight control law design. A scheduled dynamic inversion (DI) controller was used as a baseline because it is a relatively mature technique that has been successfully used on the X-36 and demonstrated on a wide variety of complex nonlinear aircraft simulations, such as the F-18 High-Alpha Research Vehicle,[4] an advanced tailless fighter configuration,[5] and the F-117 (Ref. 6). DI also is very closely related to most of the other controllers in this paper. The basic concept behind DI is to cancel out the aircraft's natural dynamics so that it will follow desired dynamics inserted by a designer. A DI control law is shown in Fig. 1. The command generator outputs desired values of the controlled variables. The outputs of the command generator are combined with the sensed values of the controlled variables to create desired dynamics for the aircraft to follow. The desired dynamics take into account tracking error and integrated error to give the control law some robustness to uncertainty. The next step is the DI block, which inverts a state-space model of the aircraft to choose actuator commands that will make the aircraft follow the desired dynamics. The state-space model has parameters that vary across the envelope and need to be updated based on flight condition. Ideally, this control law should make the aircraft behave like an integrator so that it tracks the desired dynamics precisely. In reality, control power limitations and modeling error prevent this from happening perfectly, and so control allocation and an integrator antiwindup approach have to be used to determine acceptable actuator commands and to keep integrated error from unrestrained growth when the aircraft cannot achieve the desired performance.

One simple adaptive modification to a DI controller is to replace the parameter scheduling block of Fig. 1 with online parameter estimation. The indirect adaptive controller (IAC) uses the modified sequential least squares (MSLS) approach to identify the major stability and control parameters. MSLS has been successfully flight tested on an F-16 as part of the U.S. Air Force's self-designing control program[1] and demonstrated in nonlinear simulation on an advanced tailless configuration[7] and a vertical takeoff and landing uninhabited air vehicle.[8] Although this indirect adaptive approach has many practical benefits, proving stability and convergence of the total system is very challenging. An approach more focused on theoretical proofs of total system stability and convergence is the backstepping adaptive controller (BAC). The BAC also performs online estimation of parameters in a DI framework, but does so using parameter update laws chosen with a Lyapunov approach to ensure the nominal system's stability and convergence to zero tracking error. Another potential benefit of the backstepping approach is that actuator saturations can be dealt with to some extent without violating the stability proof. However, like many stability-oriented approaches, achieving acceptable transient properties can be difficult. Currently, there have been only a fairly limited number of flight control simulation applications of this approach.[9,10]

Another modification of a DI controller that can be proven to be stable with a Lyapunov approach is to include a nonlinear adaptive term in the desired dynamics block of Fig. 1. This approach attempts to compensate for uncertainty without explicitly identifying changes in the aircraft model. The neural network controller (NNC) and the

Received 18 October 1999; revision received 20 April 2000; accepted for publication 20 July 2000.

*Flight Controls Engineer, Air Vehicle Department, Aeromechanics Division; steinbergml@navair.navy.mil. Senior Member AIAA.

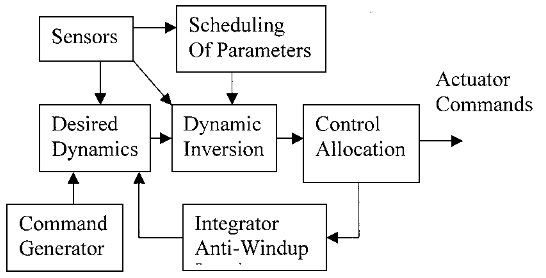

Fig. 1 DI controller.

variable structure controller (VSC) both adopt this approach. The NNC uses a type of adaptive nonlinearity very loosely related to the parallel distributed way the brain processes and stores information. This approach has been flight tested under a joint U.S. Navy, U.S. Air Force, and NASA program on the X-36 (Ref. 2) and applied to several different aircraft simulation models, such as advanced tailless configuration[11] and the XV-15 (Ref. 12). Alternatively, the VSC uses an approximation of a discontinuous nonlinearity that has been proven to have considerable robustness properties in theory. This approach has been demonstrated for flight control on numerous simulation models, such as an F-14 (Ref. 13), an AV-8B (Ref. 14), and a tailless aircraft configuration.[15] However, most of this work has used either linear or fairly simple nonlinear models.

A sixth controller, the model predictive controller (MPC) is related to DI, but provides lead action by calculating a sequence of commands that optimizes a quadratic cost function over a short time interval. When the optimization problem is solved using sequential quadratic programming (SQP), it is also possible to directly incorporate a variety of useful state and control constraints at a cost of much greater controller computational complexity. The MPC controller used in this paper is also another indirect adaptive controller that uses MSLS parameter estimation for adaptation. Predictive control is an approach that has been very successful in the process control industry and has been demonstrated for flight control in numerous simulation studies, such as a nonlinear F-16 simulation,[16] a simple nonlinear aircraft model,[17] and an agile interceptor missile.[18]

The final controller is the only one that does not explicitly use a model of the plant and is quite different from DI. Fuzzy logic is a machine intelligence approach in which desired behavior can be specified in rules, such as "if roll error is large and roll rate is medium then aileron position is large." This allows incorporation of complex nonlinear strategies based on pilot or engineer intelligence within the control law. Although the fuzzy logic controller is a nonadaptive controller, the use of pilot strategies can allow considerable ability to respond to failures. However, because of the use of highly nonlinear heuristic strategies, it can be very difficult to analyze fuzzy logic control laws. Fuzzy logic is currently used on the E-6A aircraft in a limited way[3] and has been demonstrated in simulation for an F-18 automatic carrier landing system,[19,20] and inner-loop control of aircraft.[21,22]

Because of the difficulty of choosing values for the fixed parameters in some of the preceding control laws, a genetic algorithm (GA) was used during the control law design process for all of the controllers except for the baseline DI and the neural network control laws. GAs are global search algorithms based on Darwin's theory of natural selection and a simple model of genetics. The main advantages of using GA are that they do not get trapped in local minima, and they can use any cost function that can be computed in a reasonable amount of time. The primary disadvantages are that they take considerable computational time, there is no guarantee that they will find an optimal solution, and that there are not currently good stopping rules. GAs have been applied in research to a number of flight control optimization problems, such as design of nonlinear guidance and control laws,[23] H-infinity control laws,[24] and fuzzy logic control laws.[21] The particular GA that is used in this paper is a stochastic GA[25] that divides the search into a local and global phase.

Note that this paper is not meant to pick winners and losers, but only to provide empirical data to show some potential strengths and weaknesses of each approach on a problem with some aspects of the complexity of a real aircraft design. All of the control laws examined display features that might make them a good choice for certain types of design problems. There are also numerous variations of each approach that could not be tried within the scope of this effort that might yield better results. Furthermore, changes in the design problem or rating criteria could certainly yield different answers in the relative performance of each controller. For example, the use of a linear desired performance model may penalize the fuzzy logic or MPC, which have been demonstrated elsewhere to be particularly good with certain types of nonlinear performance criteria. Alternatively, the fixed controllers may have a disadvantage compared to the adaptive controllers due to the lack of any supervisory control like a pilot (though it is also possible a pilot might interact more unpredictably given the potential for pilot–vehicle coupling with some of these approaches).

Design Problem

The design problem examined in this paper includes the following elements:

1) Track a linear desired performance model for different types of single- and multiple-axis maneuvers. The controlled variables are roll angle, angle of attack, and sideslip angle.

2) Achieve this performance despite fairly restrictive actuator position and rate saturations. The control allocation technique was a simple ganging of ailerons, rudders, and stabilators into three pseudoeffectors. As a result, the controllers have to deal with much less control power than would be available with a more sophisticated control allocation approach. An additional complexity is that the saturation rates are substantially different for the different actuators. Ailerons, for example, are more than twice as fast as stabilators.

3) Minimize control effector usage. It is typical in many research papers to use the maximum capability of an existing aircraft. However, for a new aircraft design, reduced control power requirements translate into lower penalties in areas such as weight and drag.

4) Demonstrate robustness to uncertainty in stability and control parameters and in the structure of the aircraft model. Those approaches that were model based used a simplified nonlinear model described in the third section. Approaches that use a priori values of stability and control parameters also had to deal with substantial parametric error.

5) Maintain stability and restore maximum tracking capability following single actuator hardover and oscillating failure cases and single or multiple control surface damage cases. The oscillating failure is similar to one caused by a detached linear variable differential transducer package. The damage cases were simulated by negating the effect of the control surface on the force and moment buildups in the model.

The aircraft simulation that was used to generate all results in this paper is a high-performance aircraft based on the F-18 with two engines, two stabilators, two ailerons, two rudders, two leading-edge flaps, and two trailing-edge flaps. The simulation uses the standard equations of motion and kinematic relations found in a variety of standard references:

$$\dot{u} = \left(F_{X_A} + F_{X_T}\right)/m - g\sin\Theta + rv - qw$$

$$\dot{v} = \left(F_{Y_A} + F_{Y_T}\right)/m + g\cos\Theta\sin\Phi + pw - ru$$

$$\dot{w} = \left(F_{Z_A} + F_{Z_T}\right)/m + g\cos\Theta\cos\Phi + qu - pv$$

$$\dot{p}\left(I_{XX}I_{ZZ} - I_{XZ}^2\right) = I_{ZZ}[l_A + l_T - qr(I_{ZZ} - I_{YY}) + qpI_{XZ}] + I_{XZ}[n_a + n_T + qp(I_{XX} - I_{YY}) - qrI_{XZ}]$$

$$\dot{q}(I_{YY}) = m_A + m_T + (r^2 - p^2)(I_{XZ}) + pr(I_{ZZ} - I_{XX})$$

$$\dot{r}\left(I_{XX}I_{ZZ} - I_{XZ}^2\right) = I_{XX}[n_A + n_T + qp(I_{XX} - I_{YY}) - qrI_{XZ}] + I_{XZ}[l_a + l_T - qr(I_{ZZ} - I_{YY}) + qpI_{XZ}]$$

$$P = \dot{\Phi} - \dot{\Psi}\sin\Theta, \qquad Q = \dot{\Theta}\cos\Phi + \dot{\Psi}\cos\Theta\sin\Phi$$

$$R = \dot{\Psi}\cos\Theta\cos\Phi - \dot{\Theta}\sin\Phi$$

The components of the aerodynamic forces (F_{XA}, F_{YA}, and F_{ZA}) and moments (l_A, m_A, and n_A) are calculated from table lookups. Gross thrust T is calculated from the following equation:

$$T = \left[1 + a_1\alpha + a_2\alpha^2\right]F_T(h, M, P_{\mathrm{LT}})[kP_{\mathrm{LT}} + c]$$

where a_1, a_2, c, and k are constants; F_T is calculated from a table lookup; and P_{LT} is lagged throttle position. The throttle model is a first-order linear system with a variable time constant and variable rate limit based on the value of P_{LT}. The actuator models are second-order linear systems (except for stabilators, which are fourth order) with rate and position limits.

Controller Descriptions

For purposes of design, the full simulation model would yield a control law that was far too complex to be practically implemented for some of the model-based approaches. As a result, the following design model was used:

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\phi} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} l_\beta\beta + l_q q + l_r r + (l_{\beta\alpha}\beta + l_{r\alpha}r)\Delta\alpha + l_p p - i_1 qr \\ \bar{m}_\alpha\Delta\alpha + \bar{m}_q q + i_2 pr - m_{\dot{\alpha}} p\beta + m_{\dot{\alpha}}(g_0/V)(\cos\theta\cos\phi - \cos\theta_0) \\ n_\beta\beta + n_r r + n_p p + n_{p\alpha} p\Delta\alpha - i_3 pq + n_q q \\ q - p\beta + z_\alpha\Delta\alpha + (g_0/V)(\cos\theta\cos\phi - \cos\theta_0) \\ y_\beta\beta + p(\sin\alpha_0 + \Delta\alpha) - r\cos\alpha_0 + (g_0/V)\cos\theta\sin\phi \\ p + q\tan\theta\sin\phi + r\tan\theta\cos\phi \\ q\cos\phi - r\sin\theta \end{pmatrix} + \begin{pmatrix} l_{\delta a} & l_{\delta r} & 0 \\ 0 & 0 & \bar{m}_{\delta e} \\ n_{\delta a} & n_{\delta r} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \delta_a \\ \delta_r \\ \delta_e \end{pmatrix}$$

Some of the key simplifications made in this model are constant velocity (a separate autothrottle will attempt to maintain this in the full simulation model), no lift and drag effects of the control surfaces, and none of the higher-frequency dynamics, for example, actuators. Also, the stabilators, ailerons, and rudders will only be used collectively, and the effects of flaps, which are scheduled with Mach and angle of attack, are ignored. Note again that this model is only used for controller design, and the full nonlinear simulation will be used to generate all results in this paper for GA optimization and for final tuning of all of the control laws.

The design model can be put in the form

$$\dot{\boldsymbol{y}} = \Phi_0(\boldsymbol{x}_1) + \Phi(\boldsymbol{x}_1)\boldsymbol{w}_1 + B(\boldsymbol{x}_1)\boldsymbol{\omega}$$

$$\dot{\boldsymbol{\omega}} = \psi_0(\boldsymbol{x}) + \psi_1(\boldsymbol{x})\boldsymbol{w}_2 + D(\boldsymbol{w}_u)\boldsymbol{u}$$

$$\dot{\eta} = q_0(\boldsymbol{x}) + q_1(\boldsymbol{x})\boldsymbol{w} + q_2(\boldsymbol{w}_u)\boldsymbol{u}$$

where $\boldsymbol{y}$ is a vector of the outputs that will be controlled; $\boldsymbol{w}_1$, $\boldsymbol{w}_2$, and $\boldsymbol{w}_u$ are vectors of parameters that vary over the flight envelope; $\boldsymbol{x}$ is the state vector; $\boldsymbol{x}_1$ is a subset of the state vector; $\boldsymbol{u}$ is the vector of control effector commands; and

$$\boldsymbol{y} = (\phi, \alpha, \beta)^T, \qquad \boldsymbol{\omega} = (p, q, r)^T, \qquad \eta = \theta$$

$$\boldsymbol{x} = (p, q, r, \alpha, \beta, \phi, \theta), \qquad \boldsymbol{x}_1 = (\alpha, \beta, \phi, \theta), \qquad \boldsymbol{w}_1 = (z_\alpha, y_\beta)^T$$

$$\boldsymbol{w}_2 = (l_\beta, l_p, l_q, l_r, l_{\beta\alpha}, l_{r\alpha}, \bar{m}_\alpha, \bar{m}_q, m_{\dot{\alpha}}, n_\beta, n_r, n_p, n_{p\alpha}, n_q)^T$$

$$\boldsymbol{w}_u = (l_{\delta a}, l_{\delta e}, l_{\delta r}, \bar{m}_{\delta e}, n_{\delta a}, n_{\delta e}, n_{\delta r})^T$$

Note again that this form is used only for initial design purposes. The full equations of motion of the preceding section will be used for simulation and final tuning of the control laws. An error will be defined as

$$\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{y}_c$$

where $\boldsymbol{y}_c$ is the output of a command generator that is a linear, stable third-order system. Finally, all of the controllers were designed assuming an update rate of 100 Hz.

DI

The DI controller had separate inner- and outer-loop control laws. The outer loop used desired values of $\boldsymbol{\omega}$ as virtual effectors to track the commanded values of the outputs. The virtual effector commands were calculated by inverting the design model

$$\boldsymbol{\omega}_d = B^{-1}\{\dot{\boldsymbol{y}}_d - [\Phi_0(\boldsymbol{x}_1) + \Phi(\boldsymbol{x}_1)\boldsymbol{w}_1]\}$$

Because it is possible that a commanded $\boldsymbol{\omega}_d$ will far exceed the capabilities of the aircraft, limits were set on $\boldsymbol{\omega}_d$ that vary with flight condition. When the commanded values of $\boldsymbol{\omega}_d$ did not exceed the limits, the desired dynamics were

$$\dot{\boldsymbol{y}}_d = \dot{\boldsymbol{y}}_c + K_{\mathrm{DI}_P}\boldsymbol{e} + K_{\mathrm{DI}_I}\int \boldsymbol{e}$$

where K_{DI_P} and K_{DI_I} are positive diagonal gain matrices. When the ω_d limits were exceeded, a control allocation approach was used to calculate the output of the control law, and integrator windup protection modified the preceding desired dynamics equation following the approach of Ref. 4. The inner-loop DI control law works similarly to the outer loop to track $\boldsymbol{\omega}_d$ with control limits based on actuator position and rate limits. The control parameter vector $\boldsymbol{w}_u$ was scheduled during maneuvering using a coarse linear interpolation based on Mach number, angle of attack, and dynamic pressure. Scheduling of the stability parameter vectors $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ was done only for the trim condition at the beginning of a maneuver and was then kept fixed. As a result, the DI control law was required to have considerable robustness to parametric errors because the flight condition could change substantially during some maneuvers. Following detection of actuator hardover failures, the relevant control effectiveness parameters were updated, and the $\boldsymbol{\omega}_d$ limits were set at a lower degraded setting. For other failure cases, the DI control law was not modified because it was assumed that the system has no knowledge of the failure.

Overall, the DI controller was relatively easy to design. The primary fixed parameters all had a relatively clear relationship to the aircraft's performance. The integrator antiwindup protection required some trial-and-error tuning, but this was not prohibitive. Furthermore, the performance of the system could be easily understood using fairly conventional analysis tools.

IAC

An indirect adaptive version of the preceding DI controller was created by replacing the parameter scheduling block of Fig. 1 with online parameter estimation. Parameter estimation was only used as part of the inner-loop DI controller to identify the 21 parameters in the vectors $\boldsymbol{w}_2$ and $\boldsymbol{w}_u$, as well as additional terms in the control vector to deal with damage and failure conditions. Parameter estimation was not used with the outer loop because the most important varying parameters have a limited effect and are very hard to estimate. The parameter identification approach used was MSLS. MSLS attempts to optimize a cost function that includes both the more conventional predicted squared error of the estimate over a weighted window of data and a term that penalizes the estimate for deviations from a constraint of the form

$$k = M\boldsymbol{\theta}$$

where $\boldsymbol{\theta}$ is a vector of parameter estimates, M is a positive weighting matrix, and k is a matrix of constraints. The constraints penalize the estimate for large deviations from a weighted blending of earlier and a priori estimates of the parameters. Finding good values for the weighting of these constraints along with the forgetting factor required considerable trial-and-error experimentation. For that reason, the stochastic GA was used for initial determination of these fixed parameters.

BAC

The BAC is essentially another adaptive variant of the DI control law that attempts to estimate parameters online. However, unlike the IAC, the parameter update laws were not chosen to minimize the predicted error of the estimate. Instead, the parameter update laws were chosen using a Lyapunov approach to ensure system stability and convergence to zero tracking error. The BAC uses the approach of Ref. 9 exactly, except for the addition of a projection algorithm to keep the parameter estimates within reasonable bounds. The parameter update laws are as follows:

$$\dot{\hat{\boldsymbol{w}}}_1 = f_c L_1^{-1}\left(\Phi^T \boldsymbol{s} + \psi_{1a}^T \tilde{\boldsymbol{\omega}}\right), \qquad \dot{\hat{\boldsymbol{w}}}_2 = f_c L_2^{-1} \psi_{2a}^T \tilde{\boldsymbol{\omega}}$$

$$\dot{\hat{\boldsymbol{w}}}_2 = f_c L_3^{-1} \Psi_u^T \tilde{\boldsymbol{\omega}}$$

where L_1, L_2, and L_3 are all positive-definite diagonal matrices; $\tilde{\boldsymbol{\omega}}$ is the difference between the desired and actual values of $\boldsymbol{\omega}$; and the ψ_{ia} and Ψ_u matrices are functions of the states and effector positions. Also, f_c is a positive scalar function with bounded derivatives that normally equals 1, but can be decreased to reduce the growth of integrated error and rate of change of parameters when the actuators are approaching saturation. As described in Ref. 9, f_c was made up of two components. The first was a fuzzy logic component and the second was a third-order linear stable system, chosen such that the derivative of f_c meets requirements for system stability.

The connection between fixed parameters and aircraft response in this control law is very complex, although partially understandable with effort. As a result, the fixed parameters in the control law were initially chosen using a stochastic GA. Following this, much simulation and tuning of the parameters was still necessary to get acceptable performance throughout the flight envelope.

NNC

The NNC is another modification of a DI controller based on the approach of Refs. 11 and 12. The neural network is placed in the desired dynamics block of Fig. 1 so that

$$\dot{\boldsymbol{y}}_d = \dot{\boldsymbol{y}}_c + K_{\mathrm{DI}_P}\boldsymbol{e} + w_{\mathrm{NN}}\varsigma$$

where w_{NN} is a matrix of neural network weights and ς is a vector of the neural network basis functions. The neural network had 172 basis functions, and its inputs were aircraft states and the past output of the desired dynamics block. Adaptation of weights in the neural network was done using a slightly modified form of

$$\dot{w}_{\mathrm{NN}} = -\gamma(\boldsymbol{e}\varsigma + \eta|\boldsymbol{e}|w_{\mathrm{NN}})$$

where γ and η are positive constants. The first term is derived from a Lyapunov stability approach, and the second term ensures the boundedness of the neural network weights. Choosing acceptable values for the fixed parameters in the neural network required considerable trial-and-error experimentation across the flight envelope because the parameters can effect the aircraft's behavior in complicated ways.

VSC

Basic VSCs use a discontinuous control law to compensate for uncertainty. In theory, this approach has impressive robustness properties. In reality, the discontinuity can create oscillations due to imperfect sensors and actuators. As a result, practical VSC designs often use a continuous approximation of the discontinuity around a boundary region, such as

$$\mathrm{sat}(e) = \begin{cases} e/\varepsilon & \text{if} \quad \varepsilon \geq \mathrm{abs}(e) \\ \mathrm{sign}(e) & \text{if} \quad \varepsilon \prec \mathrm{abs}(e) \end{cases}$$

where ε is a positive scalar value. This term was added to the desired dynamics block in Fig. 1 so that the desired dynamics were

$$\dot{\boldsymbol{y}}_d = \dot{\boldsymbol{y}}_c + K_{\mathrm{DI}_P}\boldsymbol{e} + K_{\mathrm{VSC}}\,\mathrm{sat}(\boldsymbol{s})$$

where K_{VSC} is an adaptive gain. An adaptive gain is used because choosing values of K_{VSC} and ε to ensure stability despite the worst-case uncertainty can lead to a very high gain controller. Adaptation of K_{VSC} was done using a supervisory set of fuzzy logic rules that includes actuator usage similar to the approach of Ref. 9. The choice of fixed parameters in this control law was fairly challenging because the parameters can affect the system response in unpredictable ways for different inputs and flight conditions. In particular, trying to choose parameters to get good performance robustness was very difficult.

MPC

The basic concept behind MPCs is to choose a sequence of control commands that optimize a quadratic cost function over a finite period of time. At each time step, only the first control value of the sequence is used. At the next time step, a new optimal sequence is computed. By solving this optimization problem using SQP, it is possible to incorporate directly a variety of constraints, such as actuator position and rate limits. To keep the optimization problem readily solvable, the nonlinear aircraft model was linearized around the current operating point during each control calculation. There are a variety of adaptive approaches that can be used with MPC. Because of time constraints, it was decided to create an indirect adaptive version using the MSLS parameter estimation approach described earlier. Also, MPC was only used to replace the outer loop of the DI controller due to convergence problems with the constrained MPC approach in situations when the aircraft was unstable. The fixed parameters in this control law all have a somewhat understandable effect on system performance individually. However, the effect of modifying multiple parameters was not always clear, particularly for cases of constrained optimization. As a result, the stochastic GA was used to determine initial parameter values.

Fuzzy Logic Controller (FLC)

Fuzzy logic control is a machine intelligence approach that can be used to incorporate aspects of pilot intelligence with more conventional control approaches. This can, to a limited extent, duplicate some of the ways a pilot might respond to an aircraft that was not behaving as expected due to damage or failures. The fuzzy logic controller (FLC) used in this paper was based on the control rule bases of the automatic carrier landing system of Refs. 19 and 20. There were three rule bases that control roll, angle of attack, and sideslip. Separate rule bases were necessary because FLCs can become very unmanageable if there are more than a few important inputs. The main inputs were error, derivative of error, and integrated error of the controlled variable. The rules that use these inputs make up the majority of the rules and are used essentially to create a nonlinear response with low damping for large errors and high damping for small errors. In addition, a small number of rules used some aircraft states and past commands. These rules were designed to deal with extreme damage or failure cases and are of the form "if the aircraft is doing something substantially different from what was commanded, then perform this compensation." The membership functions were Gaussian to allow smooth transition between rules. Initial values of the membership functions were determined using the stochastic GA, although much further tuning was required. Each rule base had between 40–55 rules and outputted commands of decoupled pseudocontrols. The outputs were multiplied by the inverse of the control matrix to distribute commands among the three control pseudoeffectors. Further scheduling was done by scaling the inputs to the rule bases using the control effectiveness at each point of the envelope. For hardover failures, a pseudoinverse is used to reallocate the commands following loss of the actuator. For other failures, no changes are made.

Results

Given the large number of controllers, and the literally thousands of maneuvers that were done with each controller, it was extremely

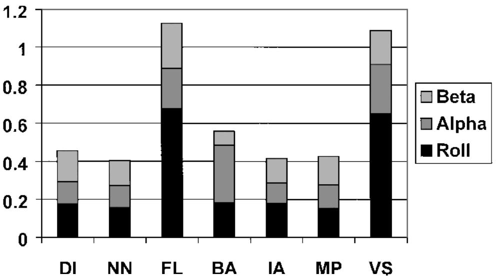

Fig. 2 Average absolute error for no failures and medium commands (degrees).

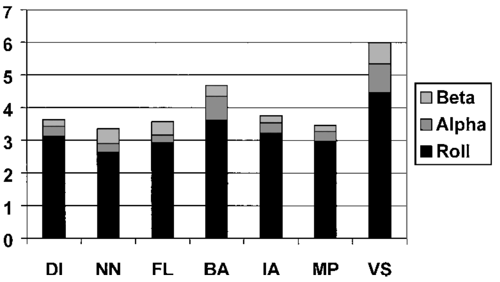

Fig. 3 Average absolute error for no failures and large commands (degrees).

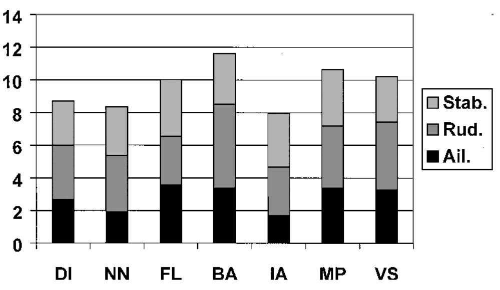

Fig. 4 Average absolute actuator position for no-failure cases (degrees).

difficult to reduce the data into a meaningful form that could fit into a short paper. The following graphs show average absolute error over a 15-s time window for a set of maneuvers at four flight conditions. The flight conditions are $0.9M$, 10,000-ft altitude; $0.8M$, 20,000 ft; $0.7M$, 30,000 ft; and $0.6M$, 40,000 ft. The maneuvers are 4-s single- and multiple-axis pulses and triangular doublets from a trim condition. The graphs are divided between small, medium, and large maneuvers. The three sizes of roll maneuvers were 5, 60, and 180 deg across the envelope. The three sizes of angle-of-attack maneuvers varied throughout the envelope and included small and medium negative maneuvers as well. At low dynamic pressure they were -15, -5, 5, 15, and 30 deg from trim. At high dynamic pressure, the alpha maneuvers were -2, -1, 1, 2, and 5 deg from trim. This adds up to a total of 26 maneuvers per flight condition.

Figures 2 and 3 show the average absolute error for medium and large single- and multiaxis maneuvers with no failures. The controllers all have somewhat comparable performance for medium maneuvers, except for the FLC and VSC, which do noticeably worse. The larger error by the FLC is due to a large extent to its slower convergence to zero steady-state error. This was necessary to provide the robustness to deal with failure cases. To a lesser extent, the BAC also has more error. For large maneuvers, FLC does relatively better, and only the VSC has substantially higher error than the rest. The best result for large maneuvers is the MPC. The MPC did particularly well on large multiaxis maneuvers. Figures 4 and 5 show average absolute actuator position and rate for the preceding set of maneuvers. There were sometimes substantial increases in actuator usage over the DI controller by some of the nonlinear controllers, particularly the FLC, MPC, BAC, and VSC.

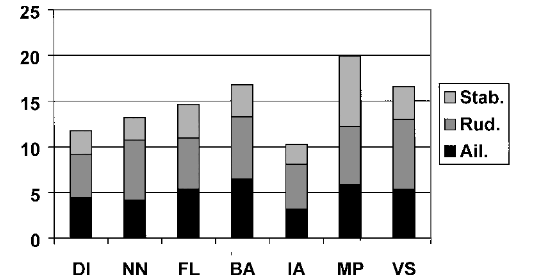

Fig. 5 Average absolute actuator rate for no-failure cases (degrees).

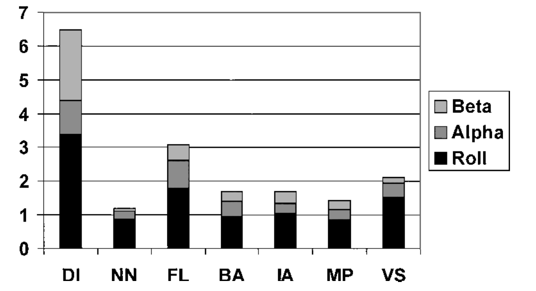

Fig. 6 Average absolute error for damaged surfaces and medium commands (degrees).

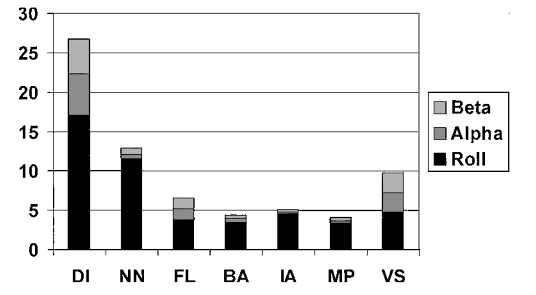

Fig. 7 Average absolute error for hardover surfaces and small commands (degrees).

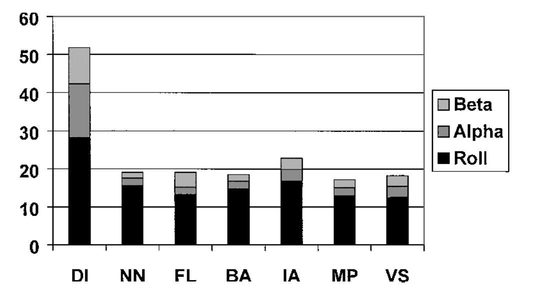

Fig. 8 Average absolute error for hardover surfaces and medium commands (degrees).

Figure 6 shows average absolute error for damaged control surfaces on the same set of medium maneuvers. The specific failure cases are 100% lost stabilator, rudder, aileron, and a combination of rudder and stabilator (which might reasonably be lost together because they are located in close physical proximity). All failures were implemented at 1.5 s into the maneuver. The main direct and indirect adaptive approaches of MPC, IAC, NNC, and BAC do the best, with the other approaches having substantially larger error. The NNC has the lowest error followed closely by the MPC. The DI control law does the worst. This is due primarily to a few cases with much worse error, rather than the DI having consistently poorer performance on all maneuvers and failure cases.

Figures 7 and 8 show average absolute error for small and medium maneuvers with aileron, rudder, and stabilator hardover failures.

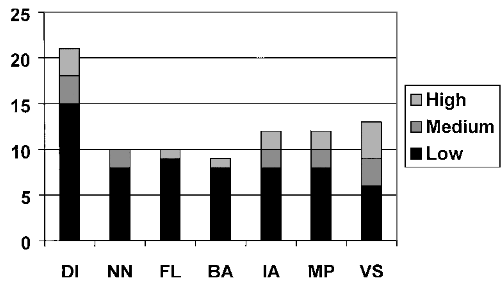

Fig. 9 Number of large-error cases for medium-sized maneuvers and hardover failures.

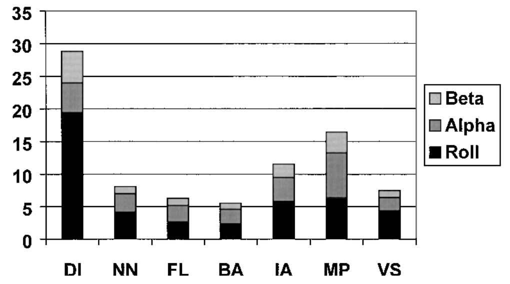

Fig. 10 Average absolute error for oscillating stab failure and small commands (degrees).

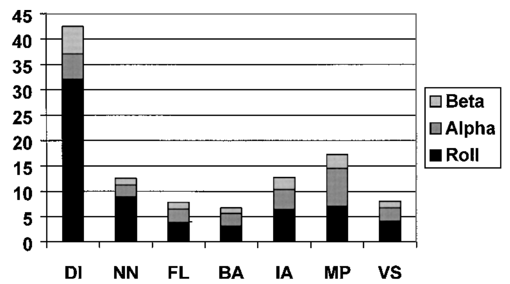

Fig. 11 Average absolute error for oscillating stab failure and medium commands (degrees).

For small maneuvers, the MPC leads, followed fairly closely by the BAC and the IAC. The NNC does poorly in roll error with stabilator and rudder failures, but does as well in the other axes and cases. The DI controller has the largest error, with the other approaches falling somewhere in between. For medium maneuvers, the MPC again does the best, followed by the FLC and the VSC, which both demonstrate admirable ability not to depart. The approaches other than DI fall closely behind. Again poor performance by DI is due to some specific cases more than generally poor results. Figure 9 shows the number of maneuvers where average absolute roll error was greater than 20 deg or average absolute alpha or beta was greater than 10 deg for just medium pulse maneuvers at high, medium, and low dynamic pressure. DI has the largest number of maneuvers with large errors. Most of the other control laws have problems with stabilator and rudder failures at low dynamic pressure. The VSC does best at this condition, but has more problems at other flight conditions.

Figures 10 and 11 show the average absolute error for small and medium maneuvers with an oscillating stabilator failure. In this case, the least adaptive controllers FLC and VSC do the best, followed closely by the direct adaptive NNC and BAC. The indirect adaptive approaches do not do as well because they have convergence difficulties in this case. Figure 12 shows the number of large error cases for medium-sized pulse maneuvers with this failure.

In addition to the general robustness demonstrated in the full simulation, some results were generated for robustness to specific types of uncertainty. For these tests, a perfect model of the aircraft was used as a starting point. One test was for robustness to parameter variations using Monte Carlo analyses. For this test, 200 runs were done at each of the 4 flight conditions. Parameter variations were

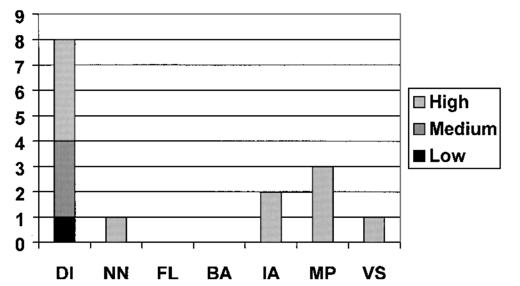

Fig. 12 Number of large-error cases for medium-sized maneuvers and oscillating failures.

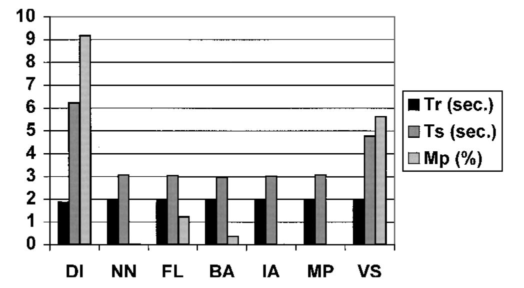

Fig. 13 Mean values of time-domain statistics for Monte Carlo analysis.

determined using a zero mean normally distributed random number generator with a standard deviation of 50% of the value of the stability and control parameters. Figure 13 shows the mean values for 10–90% rise time, settling time to 5%, and percent overshoot of 180-deg roll steps. The desired model has values of 2 s, 3 s, and 0%, respectively. Note that it may not seem to make much sense to test adaptive control laws this way but that all of the control laws used some knowledge of parameters in the design process and most used a priori values of parameters explicitly in the control law. As can be seen, all of the controllers did fairly well. The largest variations were for the DI controller followed by the VSC controller.

Conclusions

Seven different nonlinear controllers were applied to a complex aircraft design problem. Each of the controllers demonstrated some ability to achieve the design criteria. The baseline DI controller was fairly robust, and it was quite challenging to develop other approaches that could consistently improve on its performance. Because the other controllers were all more complex computationally, more labor intensive to design, and would probably require new approaches for validation, this raises questions about the value of the nonlinear approaches for fairly stable, conventional aircraft. Furthermore the adaptive and intelligent approaches also required ad hoc modifications to work best, such as gain scheduling of fixed parameters, which increased the complexity of the designs. Nonetheless, there were some cases where the nonlinear approaches greatly outperformed the baseline DI controller. In damage/failure cases where excess control power was highly limited, the approaches that used explicit identification of parameters tended to do the best, particularly MPC due to its ability to optimize with nonlinear constraints. MPC also demonstrated excellent performance for large multiaxis maneuvers without any failures or damage. However, MPC also required over an order of magnitude more processing power than the other six controllers did and tended to command relatively large actuator position and rates. It also was a more difficult approach to design than the DI-based approaches. For cases where faster adaptation was needed or control power limits were less important, the direct adaptive neural network and backstepping approaches did very well. These were the only approaches that often converged in less than 1 s. The neural network approach also did extremely well in the nominal case because it had the least negative impact on the system when no large errors requiring adaptation were present. In contrast, the backstepping approach often had somewhat problematic performance, even without failures. The fuzzy logic control system had remarkably stable performance in damage and failure scenarios for a fixed controller, though it was rarely the best performer due to its slow convergence to zero steady-state error. The variable structure control law demonstrated some ability to remain stable in cases where other controllers had much more difficulty, but was the most likely to have problematic performance for a variety of different cases across the envelope.

References

[1]Monaco, J., Ward, D., Barron, R., and Bird, R., "Implementation and Flight Test Assessment of an Adaptive Reconfigurable Flight Control System," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1999, pp. 1443–1454.

[2]Brinker, J., and Wise, K., "Flight Testing of a Reconfigurable Flight Control Law on the X-36 Tailless Fighter Aircraft," *2000 AIAA Meeting Papers on Disc* [CD-ROM], Vol. 5, No. 4, AIAA, Reston, VA, 2000.

[3]Brushwood, D., Olson, A., and Smyth, J., "The E-6A Orbit Improvement System and Its Effect upon LTWA Verticality," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998, pp. 1453–1458.

[4]Honeywell and Lockheed Martin, "Multivariable Flight Control Design Guidelines," U.S. Air Force Rept. WL-TR-96-3099, Wright–Patterson AFB, OH, May 1996.

[5]Buffington, J., "Modular Control Law Design for the Innovative Control Effectors (ICE) Tailless Fighter Aircraft Configuration 101-3," U.S. Air Force Rept. AFRL-VA-WP-TR-1999-3057, Wright–Patterson AFB, OH, June 1999.

[6]Colgren, R., and Enns, D., "Dynamic Inversion Applied to the F-117," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1997, pp. 275–284.

[7]Eberhardt, R., and Ward, D., "Indirect Adaptive Flight Control of a Tailless Fighter Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1999, pp. 466–476.

[8]Ward, D., Monaco, J., and Schierman, D., "Reconfigurable Control for VTOL UAV Shipboard Landing," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1999, pp. 499–509.

[9]Steinberg, M., and Page, A., "Nonlinear Adaptive Flight Control with Genetic Algorithm Design Optimization," *International Journal of Robust and Nonlinear Control*, Vol. 9, No. 15, 1999, pp. 1097–1115.

[10]Krstic, M., Kanellakopoulos, I., and Kokotovic, P., *Nonlinear and Adaptive Control Design*, 1st ed., Wiley, New York, 1995.

[11]Calise, A., Lee, S., and Sharma, M., "Direct Adaptive Reconfigurable Control of a Tailless Fighter Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998, pp. 88–97.

[12]Rysdyk, R., and Calise, A., "Fault-Tolerant Flight Control via Adaptive Neural Network Augmentation," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998, pp. 1722–1728.

[13]Singh, S., Steinberg, M., and DiGirolamo, R., "Variable Structure Robust Flight Control System for the F-14," *IEEE Transactions on Aerospace Electronics Systems*, Vol. 33, No. 1, 1997, pp. 77–84.

[14]Calise, A., and Kramer, F., "A Variable Structure Approach to Robust Control of VTOL Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 5, 1984, pp. 620–626.

[15]Shtessel, Y., Buffington, J., Pachter, M., Chandler, P., and Banda, S., "Reconfigurable Flight Control on Sliding Modes Addressing Actuator Deflection and Deflection Rate Saturation," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998, pp. 127–137.

[16]Shearer, C., and Heise, S., "Constrained Model Predictive Control of a Nonlinear Aerospace System," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998, pp. 772–785.

[17]Singh, S., Steinberg, M., and DiGirolamo, R., "Nonlinear Predictive Control of Feedback Linearizable Systems and Flight Control System Design," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 5, 1995, pp. 1023–1028.

[18]Mehra, R., Gopinathan, M., and Sistu, P., "Robust Nonlinear Model Predictive Control for Agile Interceptor Missiles," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998, pp. 201–211.

[19]Steinberg, M., "Development and Simulation of an F/A-18 Fuzzy Logic Automatic Carrier Landing System," *Proceedings of the Fuzzy Systems Conference*, Inst. of Electrical and Electronics Engineers, New York, 1993, pp. 797–802.

[20]Steinberg, M., "A Fuzzy Logic-Based F/A-18 Automatic Carrier Landing System," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1992, pp. 407–417.

[21]KrishnaKumar, K., Gonsalves, P., Satyadas, A., and Zacharias, G., "Hybrid Fuzzy Logic Flight Controller Synthesis via Pilot Modeling," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1995, pp. 494–504.

[22]Ho, H., and Balakrishnan, S., "Hierarchical Fuzzy Logic Controller with a Velocity Modifier for Restructurable Flight Control Systems, *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1995, pp. 505–515.

[23]Sweriduk, G., Menon, P., and Steinberg, M., "Genetic Programming for Nonlinear Flight Control System Synthesis," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1995, pp. 461–470.

[24]Menon, P., Yousefpor, M., Lam, T., and Steinberg, M., "Robust Command Augmentation Design Using Genetic Programming," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1998 pp. 296–304.

[25]KrishnaKumar, K., Swaminathan, R., Garg, S., and Narayanaswamy, S., "Solving Large Optimization Problems Using Genetic Algorithms," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1995 pp. 449–460.