Expert Supervision of Fuzzy Learning Systems for Fault Tolerant Aircraft Control

WAIHON A. KWONG, KEVIN M. PASSINO, MEMBER, IEEE, ERIC G. LAUKONEN, AND STEPHEN YURKOVICH, SENIOR MEMBER, IEEE

Actuator, sensor, or other aircraft subsystem failures, or structural failures that result from, for example, battle damage can cause catastrophes that may lead to loss of the aircraft. While experienced pilots can often compensate for failures, in certain emergency situations there is the need for computer-assisted or fully computer-automated reconfiguration of the aircraft control laws to save the aircraft. In this paper, we begin by showing that the fuzzy model reference learning controller (FMRLC) [1]–[5] can be used to reconfigure the nominal controller in an F-16 aircraft to compensate for various actuator failures without using explicit failure information (e.g., the time of occurrence of the failure or its magnitude). Next, we show that the performance of the FMRLC can be significantly enhanced by exploiting failure detection and identification (FDI) information to achieve a "performance adaptive" system that seeks an appropriate performance level depending on the type of failure that occurred. We develop an expert supervision strategy for the FMRLC that uses only information about the time at which a failure occurs and show that it achieves higher performance control reconfiguration than an unsupervised FMRLC. In addition, we show that similar performance can be achieved if we only use estimates of the failure time and magnitude obtained from a fuzzy estimator. We close our study with a brief assessment of the advantages and disadvantages of the approaches used in this paper.

I. INTRODUCTION

There are virtually an unlimited number of possible failures that can occur on a sophisticated modern aircraft. While preplanned pilot-executed procedures have been developed for certain anticipated failures (especially catastrophic and high probability failures), certain unanticipated events can occur which complicate successful failure accommodation. Indeed, aircraft accident investigations sometimes find that even with some of the most severe unanticipated failures, there was a way in which the aircraft could have been saved if the pilot had taken proper actions in a timely fashion. Because the time frame during a catastrophic event is typically a few seconds, given the level of stress and confusion during these incidents, it is understandable that a pilot may not find the solution in time to save the aircraft. For instance, American Airlines Flight 191 (a DC-10) crashed at Chicago-O'Hare International Airport on May 29, 1979, because the no. 1 engine and the no. 1 hydraulic system malfunctioned, and the slats were retracted with significant delays at takeoff. Even though later simulator tests showed that the aircraft could have been flown successfully, circumstances of the accident and the lack of available warning systems made it unreasonable to expect the pilots of Flight 191 to save the aircraft. With the recent advances in computing technology and control theory, it appears that the potential exists to implement a computer control strategy that can assist (or replace) the pilot in helping mitigate the consequences of severe failures in aircraft.

Generally speaking on modern aircraft the predominant way to cope with failures is to use "physical redundancy." For example, the F-16 is a high performance fighter aircraft which uses an analog fly-by-wire control system with quadruplex sensor comparison and quadruplex actuator redundancy. However, physical redundancy is expensive in terms of manufacturing, operation and maintenance. In this study, we assume 1) that there is no physical redundancy, or 2) that the redundancy is exhausted (e.g., all redundant actuators have already failed). In this case, the only way to deal with failures in, for example, sensors and actuators, is to use what has been called "analytical redundancy" [6], [7]. A system is said to possess analytical redundancy if, for instance, 1) in response to an actuator failure, it can automatically reconfigure the flight control system so that adequate performance can be obtained using the remaining unfailed actuators, or 2) in response to a sensor failure, it can change the way it processes the remaining sensor information so that the effects of the sensor failure are minimized.

Reconfiguration strategies can generally be classified into two categories: 1) conventional control engineering approaches, and 2) artificial intelligence (AI) approaches. In the conventional control engineering approach, a nominal

Manuscript received April 6, 1994; revised November 4, 1994. This work was supported in part by an Ohio Aerospace Institute Grant, in part by an Ohio State University Interdisciplinary Seed Grant, and in part by National Science Foundation Grant IRI9210332.

The authors are with the Department of Electrical Engineering, The Ohio State University, Columbus, OH 43210-1272 USA. Please address all correspondence to K. Passino.

IEEE Log Number 9407858.

controller is designed using a mathematical model of the aircraft, then various reconfiguration methods are used to alter the nominal control laws in case of a failure; see, for example [8]–[11].[1] The most promising techniques for controller reconfiguration presented to date are, perhaps, the model following based approaches. Huang and Stengel studied the use of an implicit model following strategy for aircraft control reconfiguration [12]. They develop a proportional integral controller using linear-quadratic control techniques, where the cost function includes the difference between reference model and plant output. Morse and Ossman presented a model following reconfigurable flight control system for a modified F-16 fighter, which has flaperons and canards [13]. Failure simulations on the linearized models show that the overall system is successful in maintaining performance of the aircraft in case of failures, except that at some flight conditions, where the structural limits are reached, an adequate compensation cannot be achieved. The AI approaches to reconfigurable control seek to automate the expertise of the pilot and control system designer in reconfiguring the nominal control laws in case there is a failure; see, for example [14]–[16]. Failure detection and identification (FDI) [6], [17]–[20] is of fundamental importance to control reconfiguration since proper failure identification can often significantly enhance our abilities to accommodate for failures [21], [22]. Conventional FDI approaches have been used for a variety of processes with varying degrees of success [17], [20], [21]. Their performance is typically limited due to their dependency on linear deterministic models of the failed and unfailed systems [19], [20]. In this work we investigate the possibility of using fuzzy systems to estimate failure times and magnitudes.

Previous investigations in conventional design approaches have achieved various degrees of success in solving the reconfigurable control problem; however, they are still far from being successfully implemented. One of the most severe limitations is the need for linear perturbation models in the controller designs (an assumption that is made in [8], [9], [11]–[13]). In this study (which is an expansion on and synthesis of the work in [23]–[25]), we avoid relying on the use of a linear model by employing the fuzzy model reference learning controller (FMRLC) introduced in [1]–[5]; instead, we rely on the use of heuristic expertise about how to best reconfigure the control laws. After establishing a failure simulation testbed for the F-16 we overview the FMRLC approach where a learning mechanism automatically synthesizes and tunes an underlying controller so that the closed-loop specifications designed into a reference model are achieved. Then, we introduce a new approach to designing the FMRLC that is tailored to our fault tolerant control application. In particular, since we want to have the performance of the FMRLC match that of the F-16 nominal control laws when there are no failures, we design the FMRLC so that it will emulate the nominal control laws until there is a failure. Our new design technique uses the perspective that the "fuzzy inverse model" [1]–[5] in the learning mechanism of the FMRLC acts as a controller in the adaptation loop; based on this we provide a procedure on how to tune the FMRLC.

The overall goal in the use of expert supervision of the FMRLC is to determine if it can facilitate the exploitation of failure information to: 1) improve performance of the reconfiguration strategy, and 2) provide a "performance adaptive" technique which seeks to obtain the best possible (but realistic) performance depending on what type of failure occurs. The main ideas on supervisory strategies for reconfigurable control presented in this paper can be summarized as follows: 1) if we know only the time at which an actuator failure occurs, (the "limited FDI information" case), then the performance of the FMRLC can be enhanced by using a supervision strategy that tunes the reference model and adjusts the learning mechanism (in [25] it is shown that if perfect FDI information is available, even better performance can be achieved via our supervisory approach); and 2) the performance of our supervisory scheme will not degrade significantly if the fuzzy estimator from [26], [27] is used to estimate the time and magnitude of the failure.

The primary contributions of this work are:

- the introduction of direct and adaptive fuzzy control for the reconfigurable control problem for aircraft (other intelligent systems based approaches are outlined above),
- the introduction of a new supervisory learning control approach for aircraft control law reconfiguration that is "performance adaptive" in the sense that it attempts to recover the best possible performance depending on the type of failure that has occurred (existing techniques in [8]–[13] typically seek to return the performance level to that of an unimpaired aircraft which in many failure scenarios is unreasonable), and
- an approach that exploits the use of failure information in control reconfiguration and does not rely on the availability of models of the failed aircraft to redesign control laws (in [8], [11], [14]–[16], [22] it is assumed perfect FDI information is available and in [8], [9]–[11], [13], [15], [16] it is assumed that models of the failed aircraft are available).

We cannot over emphasize, however, that the main objective of this paper is to investigate the *alternative* of using fuzzy control techniques for aircraft control law reconfiguration rather than conventional or earlier AI-based approaches. While the general claims about the novelty of the results listed above in 1)–3) can be made, it is premature to claim superiority of any one approach to control reconfiguration without a significantly detailed comparative analysis (including theoretical and implementation-based studies). Such an analysis is beyond the scope of this paper, and has not yet appeared in the literature for any approach.

[1] Note that in case of minor failures (e.g., actuator or sensor bias, or some types of actuator or sensor performance degradation), a "robust" flight control system could be sufficient. Of course, it is unlikely that such a robust control system alone would be sufficient to cope with a wide range of drastic failures.

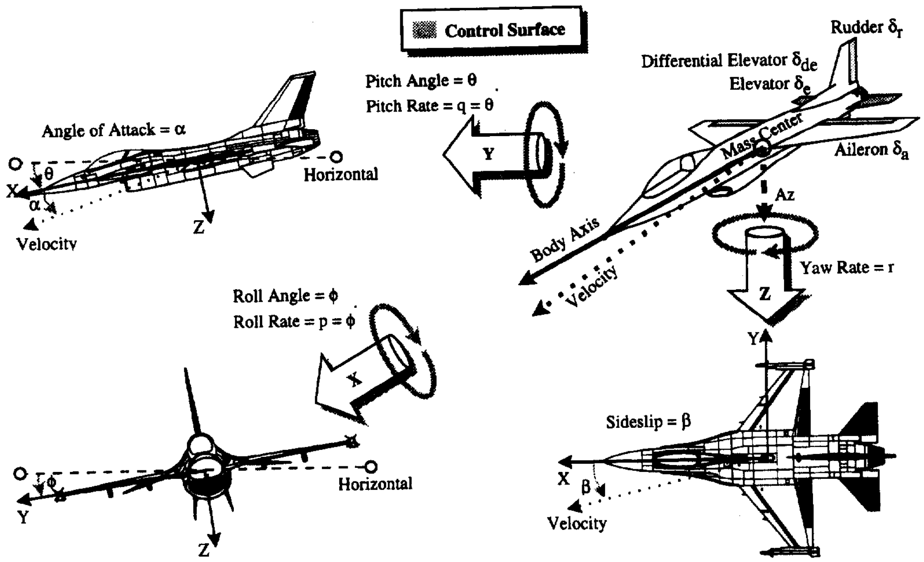


Fig. 1. The F-16 aircraft.

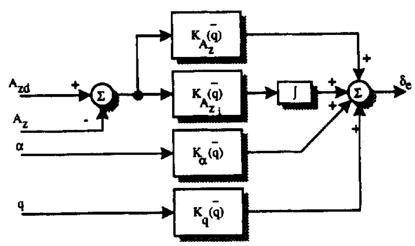


Fig. 2. Nominal longitudinal control system.

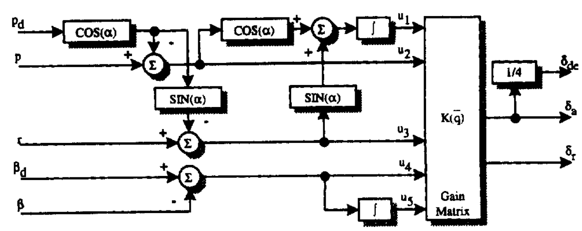


Fig. 3. Nominal lateral control system.

II. Problem Formulation

The F-16 aircraft model used in this research is based on a set of five linear perturbation models (that were extracted from a nonlinear model at five operating conditions)[1] (A_i, B_i, C_i, D_i), $i \in \{1, 2, 3, 4, 5\}$:

$$\begin{aligned}\dot{\underline{x}} &= A_i\underline{x} + B_i\underline{u}\\ \underline{y} &= C_i\underline{x} + D_i\underline{u}\end{aligned} \tag{1}$$

where the variables are defined as shown in Fig. 1 (note that we ignore gravity): 1) Inputs $\underline{u} = [\delta_e \quad \delta_{de} \quad \delta_a \quad \delta_r]'$ (where "$'$" denotes matrix transpose): δ_e = elevator deflection (degrees), δ_{de} = differential elevator deflection (degrees), δ_a = aileron deflection (degrees), δ_r = rudder deflection (degrees); 2) System State $\underline{x} = [\alpha \quad q \quad \phi \quad \beta \quad p \quad r]'$: α = angle of attack (degrees), q = body axis pitch rate (degrees/s), ϕ = Euler roll angle (degrees), β = sideslip angle (degrees), p = body axis roll rate (degrees/s), r = body axis yaw rate (degrees/s); 3) Outputs $\underline{y} = [x^t \quad A_z]'$: A_z = normal acceleration (g's); and 4) system matrices (A_i, B_i, C_i, D_i): (available in report form [28] from the authors on request).

The nominal control laws for the F-16 aircraft used in this study consist of two parts, one for the longitudinal channel as shown in Fig. 2 and the other for the lateral channel as shown in Fig. 3. The inputs to the controller are the pilot commands and the F-16 system feedback signals. For the longitudinal channel, the pilot command is the desired pitch A_{zd}, and the system feedback signals are normal acceleration A_z, angle of attack α, and pitch rate q. Likewise, for the lateral channel, the pilot commands are the desired roll rate p_d as well as the desired sideslip β_d, and the system feedback signals are the roll rate p, yaw angle r, and sideslip β. The controller gains, $K_{A_z}(\bar{q})$, $K_{A_{z_i}}(\bar{q})$, $K_\alpha(\bar{q})$, and $K_q(\bar{q})$ for the longitudinal channel in Fig. 2, and $K(\bar{q})$ for the lateral channel in Fig. 3, are scheduled as a function of different dynamic pressures $\bar{q}$. The dynamic pressure at all five perturbation models is fixed at 499.24 psf, which is based on an assumption that the F-16 aircraft will operate with constant speed and altitude. Hence, a gain schedule table is used to determine the controller gains as follows:

[1] All information about the F-16 aircraft models was provided by Wright Laboratories.

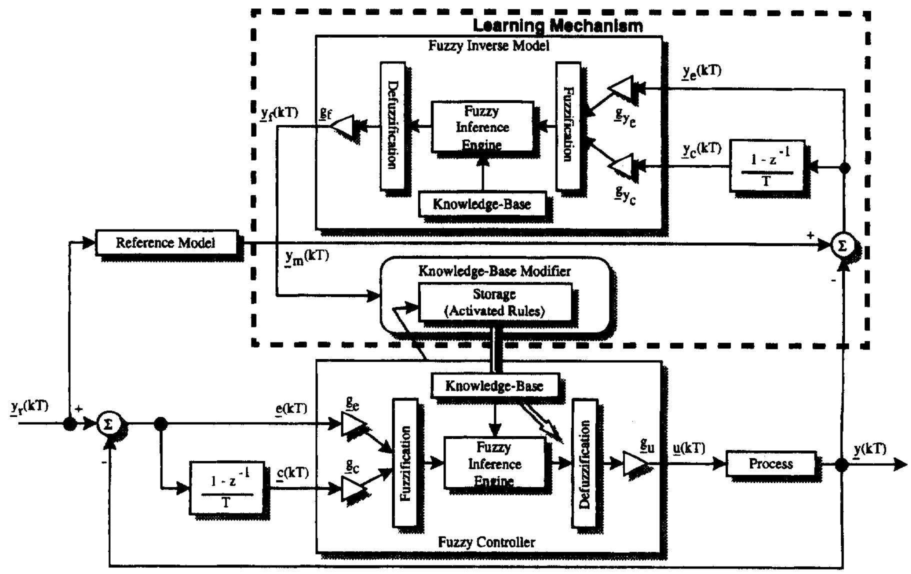


Fig. 4. Architecture for the FMRLC.

$K_{A_z}(499.24) = 0.021$, $K_{A_{z_I}}(499.24) = 0.079$, K_α (499.24) $= 0.53$, K_q (499.24) $=$ 0.37, and

$$K(499.24) = \begin{bmatrix} 0.47 & 0.14 & 0.14 & -0.56 & -0.38 \\ -0.08 & -0.056 & 0.78 & -1.33 & -4.46 \end{bmatrix}. \tag{2}$$

The transfer function $\frac{20}{s+20}$ is used to represent the actuator dynamics for each of the aircraft control surfaces, and the actuators have physical saturation limits so that: $-21° \le \delta_e \le 21°$, $-21° \le \delta_{de} \le 21°$, $-23° \le \delta_a \le 20°$, and $-30° \le \delta_r \le 30°$. The actuator rate saturation is $\pm$ 60°/s for all the actuators. To simulate the closed-loop system we interpolate between the five perturbation models based on the value of α (producing a nonlinear simulation of the F-16). For all the simulations, a special "loaded roll command sequence" is used. For this command sequence: 1) At time $t = 0.0$, a 60°/s roll rate command (p_d) is held for 1 s; 2) At time $t = 1.0$, a 3g pitch command (A_{zd}) is held for 9 s; 3) At time $t = 4.5$, a 60°/s roll rate command (p_d) is held for 1.8 s; and 4) At time $t = 11.5$, a 60°/s roll rate command (p_d) command is held for 1 s. The sideslip command β_d is held at zero throughout the sequence.

Many different failures can occur on a high performance aircraft such as the F-16. For instance, there could be actuator or sensor performance degradation (e.g., a bandwidth decrease), actuators could get stuck at certain angles, actuators could be damaged so that the control surface oscillates in an uncontrollable fashion, or there could be severe structural damage to the aircraft. In this paper we will develop reconfiguration strategies for actuator stuck failures (which are relatively minor considering the severity of failures that could occur).

III. Fuzzy Model Reference Learning Control for the F-16

In this section, we provide a brief overview of the FMRLC technique taken from [2] then introduce a new design procedure for the FMRLC that is tailored to fault tolerant control applications. Following this we evaluate the performance of the FMRLC in accommodating for aileron stuck failures. Due to the applications focus of this special issue, we provide no background information on other fuzzy adaptive control techniques and refer the reader to [1]–[5] and the extensive lists of references therein.

A. An Introduction to Fuzzy Model Reference Learning Control

The FMRLC, which is shown in Fig. 4, utilizes a learning mechanism that 1) observes data from a fuzzy control system, 2) characterizes its current performance, and 3) automatically synthesizes and/or adjusts the fuzzy controller so that some prespecified performance objectives are met.[2] These performance objectives are characterized via the *reference model* shown in Fig. 4. In a manner analogous to conventional model reference adaptive control, where conventional controllers are adjusted, the learning mechanism seeks to adjust the fuzzy controller so that the closed-loop system (the map from $\underline{y}_r(kT)$ to $\underline{y}(kT)$ where T is the sampling period) acts like a prespecified reference model (the map from $\underline{y}_r(kT)$ to $\underline{y}_m(kT)$). Next we describe each component of the FMRLC.

1) The Fuzzy Controller: The process in Fig. 4 is assumed to have r inputs denoted by the r-dimensional vector

[2]Notice that we use sampled versions of all signals as the operation of the FMRLC is easier to explain and visualize in discrete-time. An analogous continuous-time development can be found in [1]–[5].

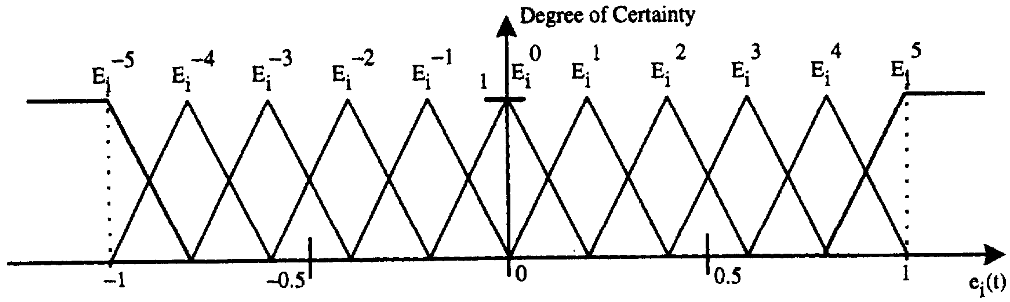


Fig. 5. Fuzzy sets on a universe of discourse.

$\underline{u}(kT) = [u_1(kT) \ldots u_r(kT)]'$ and s outputs denoted by the s-dimensional vector $\underline{y}(kT) = [y_1(kT) \ldots y_s(kT)]'$. Most often the inputs to the fuzzy controller are generated via some function of the plant output $\underline{y}(kT)$ and reference input $\underline{y}_r(kT)$. Fig. 4 shows a special case of such a map that was found useful in many applications. The inputs to the fuzzy controller are the error $\underline{e}(kT) = [e_1(kT) \ldots e_s(kT)]'$ and change in error $\underline{c}(kT) = [c_1(kT) \ldots c_s(kT)]'$, defined as $\underline{e}(kT) = \underline{y}_r(kT) - \underline{y}(kT)$, and $\underline{c}(kT) = \frac{\underline{e}(kT) - \underline{e}(kT-T)}{T}$, respectively, where $\underline{y}_r(kT) = [y_{r_1}(kT) \ldots y_{r_s}(kT)]'$ denotes the desired process output.

In fuzzy control theory, the range of values for a given controller input or output is often called the "universe of discourse" [29]. Often, for greater flexibility in fuzzy controller implementation, the universes of discourse for each process input are "normalized" to the interval [-1, +1] by means of constant scaling factors. For our fuzzy controller design, the gains $\underline{g}_e$, $\underline{g}_c$, and $\underline{g}_u$ are employed to normalize the universe of discourse for the error $\underline{e}(kT)$, change in error $\underline{c}(kT)$, and controller output $\underline{u}(kT)$, respectively (e.g., $\underline{g}_e = [g_{e_1} \cdots g_{e_s}]'$ so that $g_{e_i} e_i(kT)$ is an input to the fuzzy controller). The gains $\underline{g}_e$ are chosen so that the range of values of $g_{e_i} e_i(kT)$ lie on $[-1, 1]$ and $\underline{g}_u$ is chosen by using the allowed range of inputs to the plant in a similar way. The gains $\underline{g}_c$ are determined by experimenting with various inputs to the system to determine the normal range of values that $\underline{c}(kT)$ will take on; then $\underline{g}_c$ is chosen so that this range of values is scaled to [-1, 1]. Thus the choice of the normalizing gains is application dependent; more discussion on the choice of these gains for the particular application of this paper will be given in the next Section.

We utilize r multiple-input, single-output (MISO) fuzzy controllers, one for each process input u_n (equivalent to using one MIMO controller). The knowledge-base for the fuzzy controller associated with the n^{th} process input is generated from **IF-THEN** control rules of the form:

If $\tilde{e}_1$ is $\tilde{E}_1^j$ **and** ... **and** $\tilde{e}_s$ is $\tilde{E}_s^k$ **and** $\tilde{c}_1$ is $\tilde{C}_1^l$ **and** ... **and** $\tilde{c}_s$ is $\tilde{C}_s^m$ **Then** $\tilde{u}_n$ is $\tilde{U}_n^{j,\ldots,k,l,\ldots,m}$,

where $\tilde{e}_a$ and $\tilde{c}_a$ denote the *linguistic variables* associated with controller inputs e_a and c_a, respectively, $\tilde{u}_n$ denotes the linguistic variable associated with the controller output u_n, $\tilde{E}_a^b$ and $\tilde{C}_a^b$ denote the b^{th} *linguistic value* associated with $\tilde{e}_a$ and $\tilde{c}_a$, respectively, and $\tilde{U}_n^{j,\ldots,k,l,\ldots,m}$ denotes the *consequent linguistic value* associated with $\tilde{u}_n$.

As a generic example for the remainder of this section, consider the case where (for example) one fuzzy control rule could be

If *error* is *positive-large* **and** *change-in-error* is *negative-small* **Then** *plant-input is positive-big*

(in this case $\tilde{e}_1$ = "*error*," $\tilde{E}_1^4$ = "*positive-large*," etc.). A set of such rules forms the "rule-base" which characterizes how to control a dynamical system. The above control rule may be quantified by utilizing fuzzy set theory to obtain a fuzzy implication of the form:

If E_1^j **and** ... **and** E_s^k **and** C_1^l **and** ... **and** C_s^m **Then** $U_n^{j,\ldots,k,l,\ldots,m}$,

where E_a^b, C_a^b, and $U_n^{j,\ldots,k,l,\ldots,m}$ denote the fuzzy sets that quantify the *linguistic statements* "$\tilde{e}_a$ is $\tilde{E}_a^b$," "$\tilde{c}_a$ is $\tilde{C}_a^b$," and "$\tilde{u}_n$ is $\tilde{U}_n^{j,\ldots,k,l,\ldots,m}$," respectively. For this example we may use fuzzy sets on the $e_i(t)$ normalized universes of discourse as shown in Fig. 5.

Assume that we use the same fuzzy sets on the $c_i(t)$ normalized universes of discourse (i.e., $C_a^b = E_a^b$). The membership functions on the output universe of discourse are assumed to be unknown; they are what the FMRLC will automatically synthesize. For the example in Fig. 5 we initialize the fuzzy controller knowledge-base with 121 rules (using all possible combinations of rules) where all the right-hand-side membership functions are triangular with base widths of 0.4 and centers at zero. This is done to model the fact that the fuzzy controller initially knows nothing about how to control the plant (of course, one can often make a reasonable best guess at how to specify a fuzzy controller, as is done for the application in this paper). For example, if $s = 1$ then all rules in our controller will take on the form "**If** E_1^j **and** C_1^l **Then** $U_n^{j,l}$," where the membership functions for E_1^j and C_1^l are shown in Fig. 5 and $U_n^{j,l}$ is a fuzzy set with triangular membership function centered at zero with base width 0.4. In conventional direct fuzzy controller development the designer specifies a set of such control rules where $U_n^{j,l}$ are also specified *a priori*; for the FMRLC, the system will automatically specify and/or modify the fuzzy sets $U_n^{j,l}$ to improve/maintain performance. Note that we use "minimum" for representation of the "and" operation in the premise, "minimum" for the inference operation, and the standard center-of-gravity (COG) defuzzification technique [29].

2) The Reference Model: The reference model provides a means for quantifying the desired performance. In general, the reference model may be any type of dynamical system (linear or nonlinear, time-invariant or time-varying, discrete or continuous time, etc.). The performance of the overall system is computed with respect to the reference model by generating an error signal $\underline{y}_e(kT) = [y_{e_1}(kT) \dots y_{e_s}(kT)]'$ where $\underline{y}_e(kT) = \underline{y}_m(kT) - \underline{y}(kT)$. Given that the reference model characterizes design criteria such as rise time and overshoot, and that the input to the reference model is the reference input $\underline{y}_r(kT)$, the desired performance of the controlled process is met if the learning mechanism forces $\underline{y}_e(kT)$ to remain very small for all time; hence, the error $\underline{y}_e(kT)$ provides a characterization of the extent to which the desired performance is met at time kT. If the performance is met ($\underline{y}_e(kT) \approx 0$) then the learning mechanism will not make significant modifications to the fuzzy controller. On the other hand if $\underline{y}_e(kT)$ is big, the desired performance is not achieved and the learning mechanism must adjust the fuzzy controller.

3) The Learning Mechanism: As previously mentioned, the learning mechanism performs the function of modifying the knowledge-base of a direct fuzzy controller so that the closed-loop system behaves like the reference model. These knowledge-base modifications are made by observing data from the controlled process, the reference model, and the fuzzy controller. The learning mechanism consists of two parts: a fuzzy inverse model and a knowledge-base modifier. The fuzzy inverse model performs the function of mapping $\underline{y}_e(kT)$ (representing the deviation from the desired behavior), to changes in the process inputs $\underline{y}_f(kT) = [y_{f_1}(kT) \dots y_{f_r}(kT)]'$ that are necessary to force $\underline{y}_e(kT)$ to zero. The knowledge-base modifier performs the function of modifying the fuzzy controller's knowledge base to affect the necessary changes in the process inputs. More details of this process are discussed next.

The authors in [1]–[5] introduce the idea of using a fuzzy system to map $\underline{y}_e(kT)$ (and possibly functions of $\underline{y}_e(kT)$, or process operating conditions), to the necessary changes in the process inputs $\underline{y}_f(kT)$. This map is called the *fuzzy inverse model* since information about the plant inverse dynamics is used in its specification. Note that similar to the fuzzy controller, the fuzzy inverse model shown in Fig. 4 contains normalizing scaling factors, namely $\underline{g}_{y_e}$, $\underline{g}_{y_c}$, and $\underline{g}_f$, for each universe of discourse. Given that $g_{y_{e_i}} y_{e_i}$ and $g_{y_{c_i}} y_{c_i}$ are inputs to the fuzzy inverse model, the knowledge-base for the fuzzy inverse model associated with the n^{th} process input is generated from fuzzy implications of the form:

If $Y_{e_1}^j$ and ... and $Y_{e_s}^k$ and $Y_{c_1}^l$ and ... and $Y_{c_s}^m$ **Then** $Y_{f_n}^{j,\dots,k,l,\dots,m}$,

where $Y_{e_a}^b$ and $Y_{c_a}^b$ denote the b^{th} fuzzy set for the error y_{e_a} and change in error y_{c_a}, respectively, associated with the a^{th} process output and $Y_{f_n}^{j,\dots,k,l,\dots,m}$ denotes the consequent fuzzy set for this rule describing the necessary change in the n^{th} process input. As with the fuzzy controller, we often utilize membership functions for the normalized input universes of discourse as shown in Fig. 5, triangular membership functions for the output universes of discourse, the minimum operation for inference, and COG defuzzification. Successful design of the fuzzy inverse model has been performed for many applications including a cargo ship steering problem [2]; an inverted pendulum [1]; anti-skid braking systems [3], [4]; a rocket velocity control problem and a rigid robot [5]; and a flexible robot [30]. In the next section we explain how to choose the fuzzy inverse model for our fault tolerant control application.

Given the information about the necessary changes in the input as expressed by the vector $\underline{y}_f(kT)$, the *knowledge-base modifier* changes the knowledge-base of the fuzzy controller so that the previously applied control action will be modified by the amount $\underline{y}_f(kT)$. Therefore, consider the previously computed control action $\underline{u}(kT - T)$, which contributed to the present system performance. Note that $\underline{e}(kT - T)$ and $\underline{c}(kT - T)$ would have been the process error and change in error, respectively, at that time. By modifying the fuzzy controller's knowledge-base we may force the fuzzy controller to produce a desired output $\underline{u}(kT - T) + \underline{y}_f(kT)$. Assume that only symmetric membership functions are defined for the fuzzy controller's output so that $c_n^{j,\dots,k,l,\dots,m}(kT)$ denotes the center value of the membership function at time kT associated with the fuzzy set $U_n^{j,\dots,k,l,\dots,m}$ (initially, all centers are at zero, $c_n^{j,\dots,k,l,\dots,m}(0) = 0$). Knowledge-base modification is performed by shifting centers of the membership functions of the fuzzy sets $U_n^{j,\dots,k,l,\dots,m}$ which are associated with the fuzzy implications that contributed to the previous control action $\underline{u}(kT - T)$. This modification involves shifting these membership functions by an amount specified by $\underline{y}_f(kT) = [y_{f_1} \dots y_{f_r}]'$ so that

$$c_n^{j,\dots,k,l,\dots,m}(kT) = c_n^{j,\dots,k,l,\dots,m}(kT - T) + y_{f_n}(kT). \quad (3)$$

The degree of contribution for a particular fuzzy implication, whose fuzzy relation is denoted $R_n^{j,\dots,k,l,\dots,m}$, is determined by its "activation level," defined by (See (4) at the bottom of the page) where μ_A denotes the membership function of the fuzzy set A and $t = kT$ is the current time. Only those rules with nonzero activation level are modified; all others remain unchanged. It is important to note that our rule-base modification procedure implements a form of *local* learning and hence utilizes memory. In other words, different parts of the rule-base are "filled in" based on different operating conditions for the system, and when one area of the rule-base is updated, other rules are not affected. Hence, the controller adapts to new situations and also remembers how it has adapted to past situations.

$$\bar{\delta}_n^{j,\dots,k,l,\dots,m}(t) = \min\{\mu_{E_1^j}(e_1(t)), \dots, \mu_{E_s^k}(e_s(t)), \mu_{C_1^l}(c_1(t)), \dots, \mu_{C_s^m}(c_s(t))\} \quad (4)$$

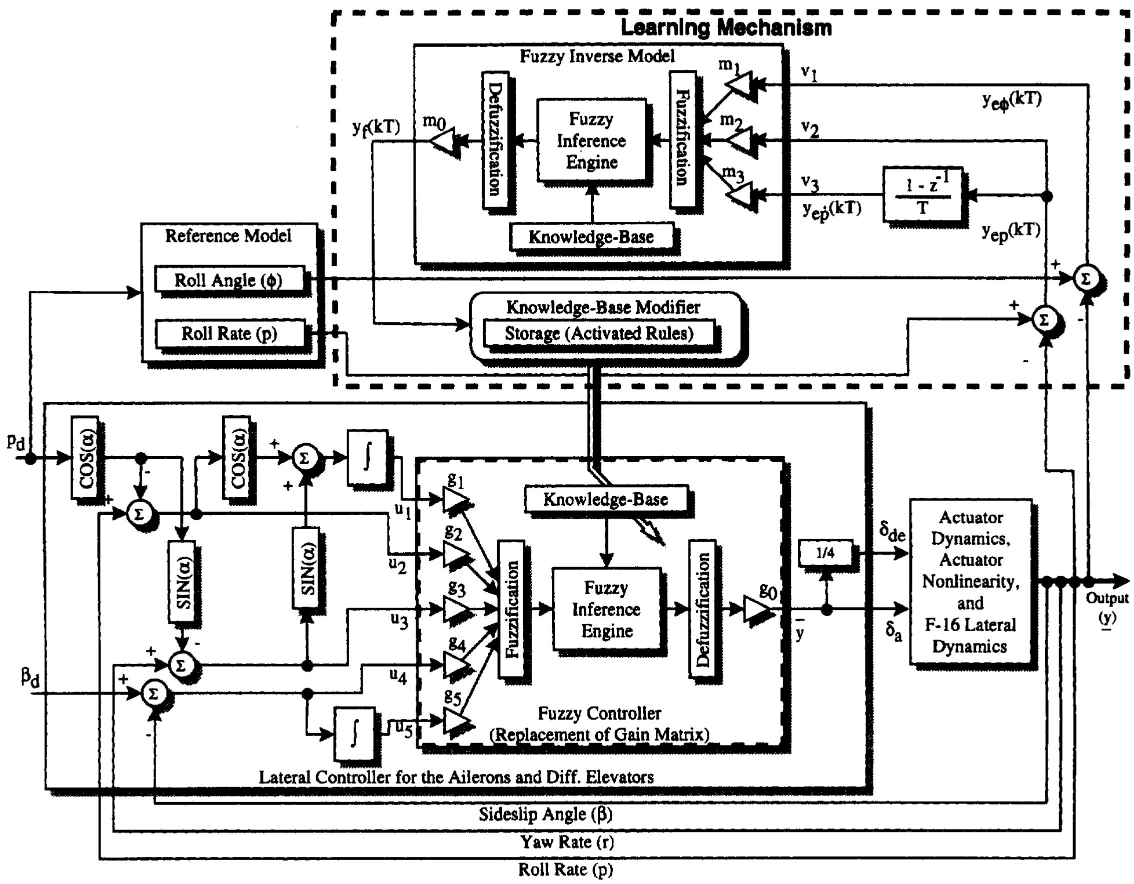


Fig. 6. FMRLC for reconfigurable control in case of aileron or differential elevator failures.

This justifies the use of the term "learning" rather than "adaptive" [1], [2], [31].

Continuing with our example from above, assume that all the normalizing gains for both the direct fuzzy controller and the fuzzy inverse model are unity and that the fuzzy inverse model produces an output $y_{f_n}(kT) = 0.5$ indicating that the value of the output to the plant at time $kT - T$ should have been $u(kT - T)+$ 0.5 to improve performance (i.e., to force $y_{e_1} \approx 0$). Next, suppose that $e_1(kT - T) =$ 0.75 and $c_1(kT - T) =$ -0.2. Then rules **If** E_1^3 **and** C_1^{-1} **Then** $U_n^{3,-1}$ and **If** E_1^4 **and** C_1^{-1} **Then** $U_n^{4,-1}$ are the only rules with nonzero activation levels ($\bar{\delta}_n^{3,-1} = 0.25$ and $\bar{\delta}_n^{4,-1} = 0.75$). Thus these rules will be the only ones that have their consequent fuzzy sets ($U_n^{3,-1}$, $U_n^{4,-1}$) modified (See Fig. 5). To modify these fuzzy sets we simply shift their centers according to (3).

B. An FMRLC for the F-16

Generally, it is not necessary to utilize all the control effectors to compensate for the effects of the failure of a single actuator on the F-16. For instance, if the ailerons in the lateral channel fail, the differential elevators can often be used for compensation, or vice versa. However, the elevators may not aid in reconfiguration for an aileron failure unless they are specially designed to induce moments in the lateral channel. Hence, it is sufficient to only redesign part of the nominal controller to facilitate control reconfiguration. Here, as shown in Fig. 6, we will replace the $K(\bar{q})$ portion of the lateral nominal control laws[3] with a fuzzy controller and let the learning mechanism of the FMRLC tune the fuzzy controller to perform control reconfiguration for an aileron failure. To apply the FMRLC in the F-16 reconfigurable control application, it is of fundamental importance that for an unimpaired aircraft, the FMRLC must behave at least as good as (indeed, the same as) the nominal control laws. In normal operation, the learning mechanism is inactive or used only to maintain the aircraft performance at the level specified in the reference model. In the presence of failures, where the performance becomes different from the specified reference model, the learning mechanism can then tune the fuzzy controller to achieve controller reconfiguration. In the next section, we explain how to pick the initial fuzzy controller shown in Fig. 6 so that it will perform the same as the nominal controller when there is no failure. We did try to use the initialization

[3] In this way we fully utilize the remaining nonlinear portion of the nominal control laws which, intuitively, are applicable whether or not the type of failure we consider has occurred or not.

procedure of [1]–[5] where one simply chooses all the output membership functions to be centered at zero, but found the approach where we initialize the fuzzy controller to match the nominal controller to be more successful.

1) The Fuzzy/Nominal Controller: Notice that the gain matrix block $K(\bar{q})$ in Fig. 3 is replaced by a fuzzy controller in Fig. 6, which will be adjusted by the FMRLC to reconfigure part of the control laws in case there is a failure. Therefore, to copy the nominal control laws, all that is necessary is for the fuzzy controller to simulate the effects of the portion of the gain matrix $K(\bar{q})$ that affects the aileron and differential elevator outputs. In this way, the FMRLC is provided with the very good initial guess of the control strategies (i.e., nominal control laws resulting from years of experience of the designer). In the nominal controller, the gain matrix $K(\bar{q})$ in Fig. 3 takes five inputs u_i (i =1, 2, 3, 4, 5) shown in Fig. 6 and gives δ_a and δ_r. However, only the portion of output corresponding to aileron δ_a and the differential elevator actuator $\delta_{de} = 0.25\delta_a$ is needed (since we found that for the failures investigated effective reconfiguration could be achieved without changing the control law for the rudder), which is calculated from the first row of the gain matrix. In order to implement the effect of a gain matrix by a fuzzy system $\mathcal{F}$, we would like

$$\bar{y} = \mathcal{F}\left(\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix}\right) = \begin{bmatrix} 0.47 \\ 0.14 \\ 0.14 \\ -0.56 \\ -0.38 \end{bmatrix}' \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \\ \vdots \\ K_5 \end{bmatrix}' \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{bmatrix}. \tag{5}$$

Thus the standard fuzzy system $\mathcal{F}$ (i.e., one with singleton fuzzification, symmetric triangular membership functions for input and output universes of discourse, and center of gravity defuzzification) is needed to capture the multiplication and addition operations shown in (5). With a standard fuzzy inference engine the knowledge-base for $\mathcal{F}$ can be constructed so that its output is the weighted sum of all the inputs simply by choosing the values of g_1, g_2, g_3, g_4, and g_5 to make the slope of the nonlinearity induced by $\mathcal{F}$ the same as the K_i gains for small values of the u_i. To do this, we first construct the rule-base for a standard fuzzy system that will sum the values of it inputs. Following this, we choose the gains g_0, and $g_1 - g_5$ so that for small u_i the slopes are K_i as we explain next. Note that the normalizing input gains $g_1 - g_5$ are used to map the symmetric domain interval[4] $[-R_i, R_i]$ of the inputs into a normalized domain interval [-1, 1] by choosing $g_i = 1/R_i$ and the output gain $g_0 = R_0$ is used to map the output of the normalized fuzzy system to the real output to achieve an output domain interval $[-R_0, R_0]$. Assuming the fuzzy system provides the summation operation as it is discussed above, the "net gain" of the fuzzy system for the i^{th} input-output pair can be considered to be $g_i g_0$ (for a fuzzy system of the type we choose above). Hence, it must be true that $g_i g_0 = K_i$. Based on the gains K_i (i = 1,2,3,4,5) in the gain matrix $K(\bar{q})$, the procedure for calculating the input and output gains for the fuzzy controller is: 1) Pick the gain of any one of the inputs (call it the i^{th} input) which has a known domain interval $[-R_i, R_i]$ in the universe of discourse, and set the input gain g_i of the corresponding input in the fuzzy controller to be $g_i = \frac{1}{R_i}$. This operating range of each input must be made as large as possible based on the system responses, so that a good interpolation is ensured within the saturation limits of the fuzzy controller. However, the range should not be too large since there is only a fixed number of membership functions with a limited numerical resolution; 2) given R_i, the output gain g_0 of the fuzzy controller is $g_0 = \frac{K_i}{g_i} = K_i R_i$ where K_i is the corresponding gain in the gain matrix; 3) with the output gain g_0, the rest of the fuzzy controller gains are $g_j = \frac{K_j}{g_0}$ where $j \in \{1,2,3,4,5\}$, $j \neq i$; and 3) Check the range of the rest of the inputs to ensure they all lie in the operating range of the nominal controller. Repeat the first step by choosing a different range or choosing another input to start with. Otherwise the conversion of a gain matrix to the input-output gains of a fuzzy system is completed.

For the F-16, after some simulation-based investigations we chose u_2 (the roll rate error signal, see Fig. 6) to have the domain interval $[-R_2, R_2]$ discussed above. After some investigations we chose the value of the range for this input to be [-100, 100] degrees/s in order to fully cover the typical operating range for the input u_2 (i.e., [-60, 60] degrees/s). The corresponding input gain of the equivalent fuzzy controller is $g_2 = \frac{1}{100}$. For any lower value of this range [-100, 100], the accuracy of $\mathcal{F}$ is distorted because the summation emulated by the fuzzy system is saturated within the range of the input. Using the procedure for gain calculation described above, the gains of the fuzzy system for the F-16 aileron and differential elevator control laws are: $[g_0 \quad g_1 \quad g_2 \quad g_3 \quad g_4 \quad g_5] = [14 \quad \frac{1}{29.79} \quad \frac{1}{100} \quad \frac{1}{100} \quad -\frac{1}{25} \quad -\frac{1}{36.84}]$. While in the interest of space we omit the plots, these choices for the g_i result in nominal and fuzzy/nominal (i.e., Fig. 6 without a learning mechanism) controllers which provide aircraft responses that are quite close for the loaded roll command sequence. The differences between the responses can be reduced by increasing the number of membership functions and rules, but this would create problems with memory and computational complexity for the FMRLC. For our fuzzy controller there are five membership functions in each input universe of discourse so that the total number of rules required is 5^5= 3125. Several test runs showed that a higher number of input membership functions (from 7 to 21 for each input) will definitely make the nominal and fuzzy/nominal controllers behave nearly the same; however, the total number of rules goes from 5^5= 3125 to 21^5= 4 084 101, which can quickly exhaust computational resources.

2) F-16 Reference Model Design: As discussed above, the reference model is used to characterize the closed-loop specifications such as rise time, overshoot, and settling

[4] For example, the symmetric "domain interval" for e_i in Fig. 5 is [-1, 1].

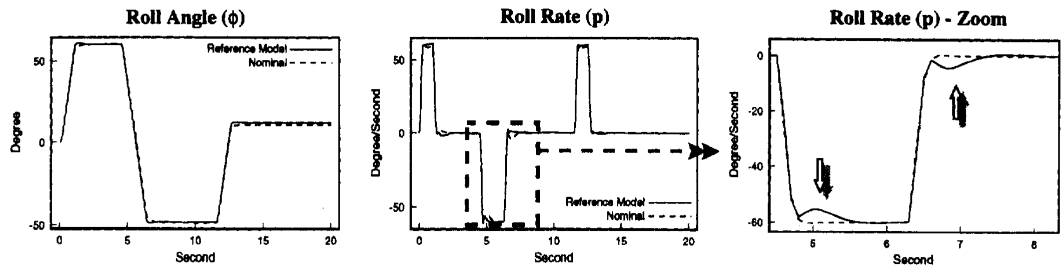


Fig. 7. Comparison of the reference model and the nominal control laws.

time. The performance of the overall system is computed with respect to the reference model by generating error signals between the reference model output and the plant outputs (i.e., $y_{e\phi}(kT)$, $y_{ep}(kT)$, and $y_{e\dot{p}}(kT)$ in Fig. 6).[5] To achieve the desired performance, the learning mechanism must force $y_{e\phi}(kT) \approx 0$, $y_{ep}(kT) \approx 0$, and $y_{e\dot{p}}(kT) \approx 0$ for all $k \geq 0$. For the aircraft, the reference model must be chosen 1) so that the closed-loop system will behave similarly to the unimpaired aircraft when the nominal control laws are used, and 2) so that unreasonable performance requirements are not requested. With these two constraints in mind, we choose a second order transfer function $H(s) = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$, where $\omega_n = \sqrt{200}$ and $\zeta = 0.85$, for the roll rate reference model and $\frac{H(s)}{s}$ for the roll angle reference model. Fig. 7 shows the responses of the reference models and that of the closed-loop system with the nominal control laws. These plots match each other quite well except that there is a small steady state error observed in the roll angle between that of the reference model and the system output (1.1° error in roll angle). Notice that in the enlarged roll rate plot in Fig. 7, the second order model (which is chosen for the reference model of the roll rate) is different at the time instants indicated by arrows. When the roll rate reference response is integrated to obtain the roll angle reference response, there would be a difference in the steady state roll angle between the reference model and the system output as depicted in the roll angle plot in Fig. 7. In general FMRLC design, the relatively simple and low order reference models (e.g., the second/third order models of roll rate/roll angle) are chosen. Choosing more complicated reference models will resolve the tracking error problem, but this is not the original intent of the design for a simple reference system. Hence, from the designer's point of view, these reference models can be used without lost of generality by assuming that they indeed generate the desired responses of the aircraft.

3) Learning Mechanism and FMRLC Design: The learning mechanism consists of two parts: 1) a "fuzzy inverse model" which performs the function of mapping the necessary changes in the process output error $y_{e\phi}(kT)$, $y_{ep}(kT)$, and $y_{e\dot{p}}(kT)$, to the relative changes in the process inputs $y_f(kT)$, so that the process outputs will match the reference model outputs, and 2) a knowledge-base modifier that updates the fuzzy controller's knowledge-base to "memorize" the needed changes in the process inputs. As discussed earlier, from one perspective the "fuzzy inverse model" represents information that the control engineer has about what changes in the plant inputs are needed so that the plant outputs track the reference model outputs. From another point of view that we introduce here, the fuzzy inverse model can be considered as another fuzzy controller in the adaptation loop that is used to monitor the error signals $y_{e\phi}(kT)$, $y_{ep}(kT)$, and $y_{e\dot{p}}(kT)$, and then choose the controller parameters in the main loop (i.e., the lower portion of Fig. 6) in such a way that these errors go to zero. With this concept in mind, we introduce the following design procedure for the FMRLC that we have found to be very useful for our fault tolerant control application.[6]

4) Design Procedure:

1) Initialize the fuzzy controller by designing its rule-base to achieve the highest performance possible when the learning mechanism is disconnected (we completed this step above).[7]
2) Choose a reference model that represents the desired closed-loop system behavior (care must be taken to avoid requesting unreasonable performance). This step was completed above.
3) Choose the rule-base for the fuzzy inverse model in a manner similar to how one would design a standard fuzzy controller (if there are many inputs to the fuzzy inverse model, then follow the approach taken in the application of this procedure below).
4) Find the range in which the i^{th} input to the fuzzy inverse model lies for a typical reference input and denote this by $[-\bar{R}_i, \bar{R}_i]$ ($i = 1, 2, \ldots, n$ where n denotes the number of inputs).
5) Construct the FMRLC with the domain interval of the output universe of discourse $[-\bar{R}_0, \bar{R}_0]$ to be [0,0], which is represented by the output gain m_0 of the fuzzy inverse model set at zero. Then, excite the system with a reference input that would be used in nor-

[5] Note that we use the notation $y_{e\dot{p}}$ to denote the signal that is the approximate derivative of the change in error of the roll rate p. The use of "$\dot{p}$" in the subscript does not denote the use of a continuous time signal.

[6] This procedure was used in the first physical implementation of the FMRLC for vibration suppression in a flexible robot arm [30].

[7] If one wishes to initialize the fuzzy controller rule-base so that all the output membership functions are located at zero (as in [1]–[5]), then this design procedure should be applied iteratively where for each pass through the design procedure the trained fuzzy controller from Steps 5 and 6 is used to initialize the fuzzy controller in Step 1.

mal operation (e.g., a series of step changes).[8] Then, increase the gain m_0 and observe the process response until the desired overall performance is achieved (i.e., the errors between the reference models and system outputs are minimum).

6) If there are difficulties in finding a value of m_0 that improves performance, then check the following three cases:
   a) If there exist unacceptable oscillations in a given process output response about the reference model response, then choose the domain intervals of the input universes of discourse for the fuzzy inverse model to be $[-a_i\bar{R}_i, a_i\bar{R}_i]$ where a_i is a scaling factor that must be selected (typically it lies in the range $0 < a_i \leq 10$),[9] and go back to Step 5.
   b) If a process response is acceptable but there exist unacceptable oscillations in the command input to the plant, then adjust the rule-base of the fuzzy inverse model and go back to Step 4.
   c) If the process output is unable to follow the reference model response, then choose a different reference model (typically at this point, one would want to choose a "slower," i.e., less demanding, reference model), and go back to Step 3.

It is important to note that for Step 5, investigations show that the choice of m_0 significantly affects the learning capabilities and the stability of the system. Generally, the size of m_0 is proportional to the learning rate, and with $m_0 = 0$ learning capabilities are "turned off" completely. Hence, for applications where a good initial guess for the controller is known and only minor variations occur in the plant, one may wish to choose a relatively small value of m_0 in attempting to ensure stability, yet allowing for some learning capabilities. For other applications where significant variations in the plant are expected (e.g., failures), one may want to choose a larger value for m_0 so that the system can quickly learn to accommodate for the variation. In such a situation there is, however, the danger that the larger value of m_0 could lead to an instability. Hence, one generally wants to pick m_0 large enough so that the system can quickly adapt to variations, yet small enough to try to ensure a stable operation. Moreover, we emphasize that if a single step response is used as an evaluation during the tuning procedure, there exists the danger that the resulting system may not be stable for other inputs. Thus, a continuous (or at least a long enough) command sequence is necessary to gain an indication of whether using a specific m_0 will result in a stable overall system. Next, we finish the design of the FMRLC by using the above design procedure to choose the learning mechanism.

In the F-16 aircraft application, starting with Step 3, the rule-base of the fuzzy inverse model is constructed.

[8]Simulations must be run long enough to try to detect possible instabilities.

[9]The value of a_i should not be chosen too small, nor too large, such that the resulting domain interval $[-a_i\bar{R}_i, a_i\bar{R}_i]$ is out of the operating range of the system output; often one would choose to enlarge the input universes of discourse by decreasing a_i.

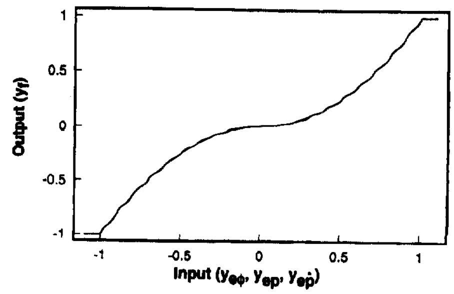


Fig. 8. Input-output relationships for $y_{e\phi}$, y_{ep}, and $y_{e\dot{p}}$ to y_f maps.

To ensure smooth performance, we would like the fuzzy inverse model (viewed as a controller) to provide the capability to correct a large error quickly and adjust more slowly for minor errors; this is indicated in the input-output map for the fuzzy inverse model in Fig. 8. To realize the map in Fig. 8 we use: 1) a similar rule-base initialization procedure as discussed in the fuzzy controller design where we picked a set of uniformly spaced input membership functions for each of the three input universes of discourse, and 2) the centers of the output membership functions c given by $c = -sign(d_i)(\sum_i^n d_i)^2$ where $sign(x) = +1$ if $x \geq 0$ and $sign(x) = -1$ if $x < 0$, d_i is the center of the input membership functions in the i^{th} input universe of discourse, and $i = 1, \ldots, n$, where n is the number of inputs. Note that the negative sign in the expression for c is necessary to represent the inverse relationship in the error between the plant output and the reference model and the change in the fuzzy controller.

According to Step 4, the difference between the reference model responses and the system outputs are measured, as shown in Fig. 9, when $m_0 = 0$. Based on this information, the ranges of all the three inputs to the fuzzy inverse model $y_{e\phi}(kT)$, $y_{ep}(kT)$, and $y_{e\dot{p}}(kT)$ are found to be [-4.4, 4.4] (the maximum deviation of $y_{e\phi}$ in Fig. 9 is +4.4), [-8.4, 8.4] (the maximum deviation of y_{ep} in Fig. 9 is +8.4), and [-97.6, 97.6] (the maximum deviation of $y_{e\dot{p}}$ in Fig. 9 is +97.6). For the first iteration, we will choose $a_i = 1$ (where $i = 1, 2, 3$).

In order to apply Step 5, the loaded roll sequence is repeated several times. In this first iteration, the gain m_0 is found to be 0.02, which is a relatively small value that will not give significant learning capabilities. Therefore, we will proceed to Step 6, and apply Condition 6a where the scaling factors a_i ($i = 1, 2, 3$) are selected to obtain a larger value for m_0. After a few iterations, the scaling factors are found to be $a_1 = 2.273$, $a_2 = 5.952$, and $a_3 = 2.049$ such that the domain intervals for the input universes of discourse for the fuzzy inverse model are [-10, 10], [-50, 50], and [-200, -200], which correspond to $y_{e\phi}(kT)$, $y_{ep}(kT)$, and $y_{e\dot{p}}(kT)$. Then, m_0 is found to be 0.1, and the tuning procedure is completed.

Notice that the actual acceptable m_0, where the difference between the reference models and the system outputs is deemed small enough, is found to be in the range [0.05,

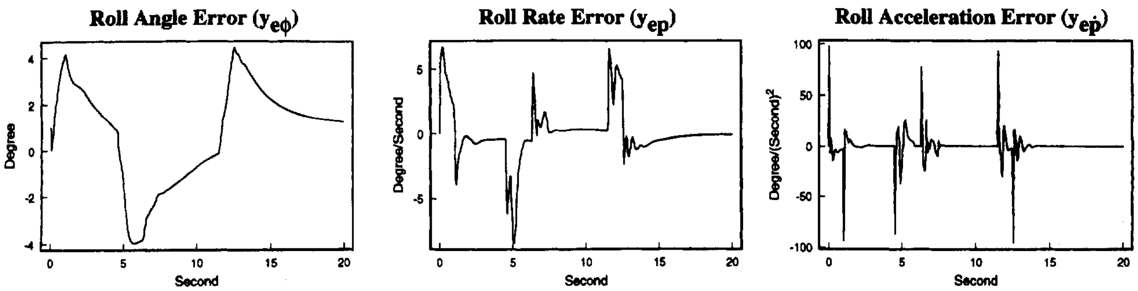


Fig. 9. Errors between the reference model and the aircraft with fuzzy controller.

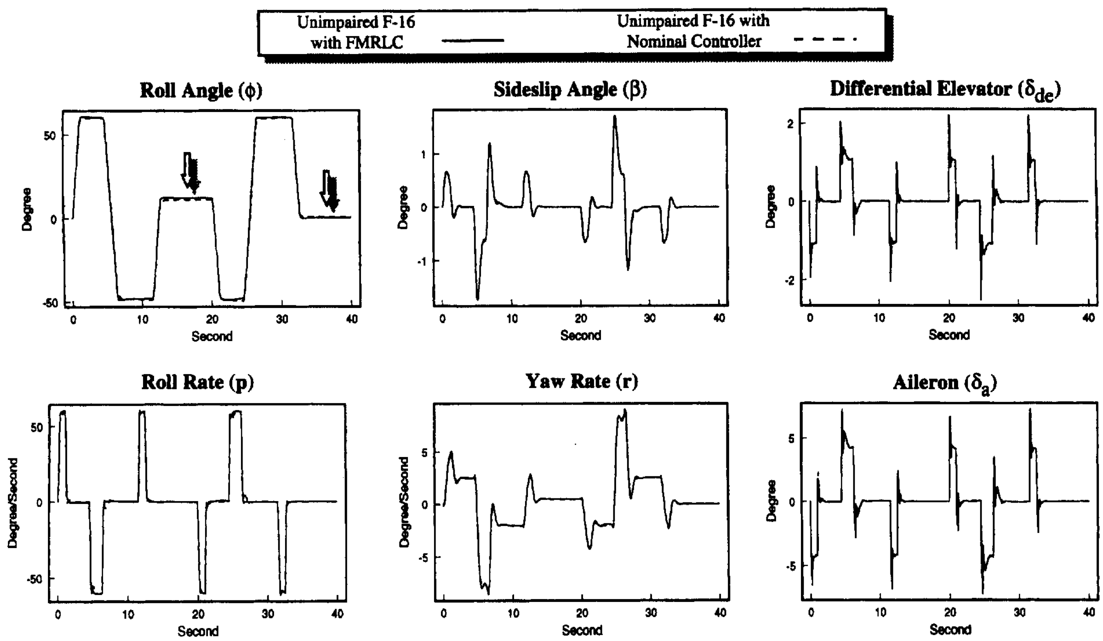


Fig. 10. Unimpaired F-16 system outputs with FMRLC.

0.11] (i.e., a range of m_0 values worked equally well). Due to the fact that we would like the largest possible value of m_0 (i.e., higher learning capabilities) to adapt to failures in the aircraft, and we would like to try to ensure stability of the overall system, we picked the compromise value of $m_0 = 0.1$.

C. Simulation Results

In this section, the F-16 aircraft with the FMRLC is simulated using the sampling time T of 0.02 s, and tested with aileron failure at 1 s.[10] Fig. 10 compares the performance of the FMRLC to the nominal control laws for the case where there is no failure. All six plots show that the FMRLC performs as good, if not better, than the nominal control laws (of course this is only for one command input sequence). Notice that the FMRLC achieves its goal of following the reference models of the roll angle and the roll rate, except for slight steady-state errors (see the portions of the response indicated by the arrows in Fig. 10) where the responses of the FMRLC do not exactly match that of the nominal control laws. As mentioned during the selection of the reference models, these errors are due to the fact that simple, second/third order, zero steady-state error reference models (roll rate/roll angle) are picked for the closed-loop multiple perturbation models of the aircraft. This discrepancy between the nominal controller and FMRLC responses is due to the difficulties one encounters in defining the reference models that can accurately emulate the nominal behavior of a given closed-loop system.

In case of failure, when the ailerons stick at 1s, the responses are shown in Fig. 11. The FMRLC system responses are acceptable since all the responses eventually match that of the unimpaired aircraft with the nominal control laws (note that the nominal control laws were not explicitly designed to accommodate this type of fault so it is not surprising that the FMRLC performs better). However, the performance in the first nine seconds of the

[10] We found that the FMRLC performed equally well when failures were induced at other times. The $t = 1$ s failure time was chosen as it represents the first time maximum aileron deflection is achieved.

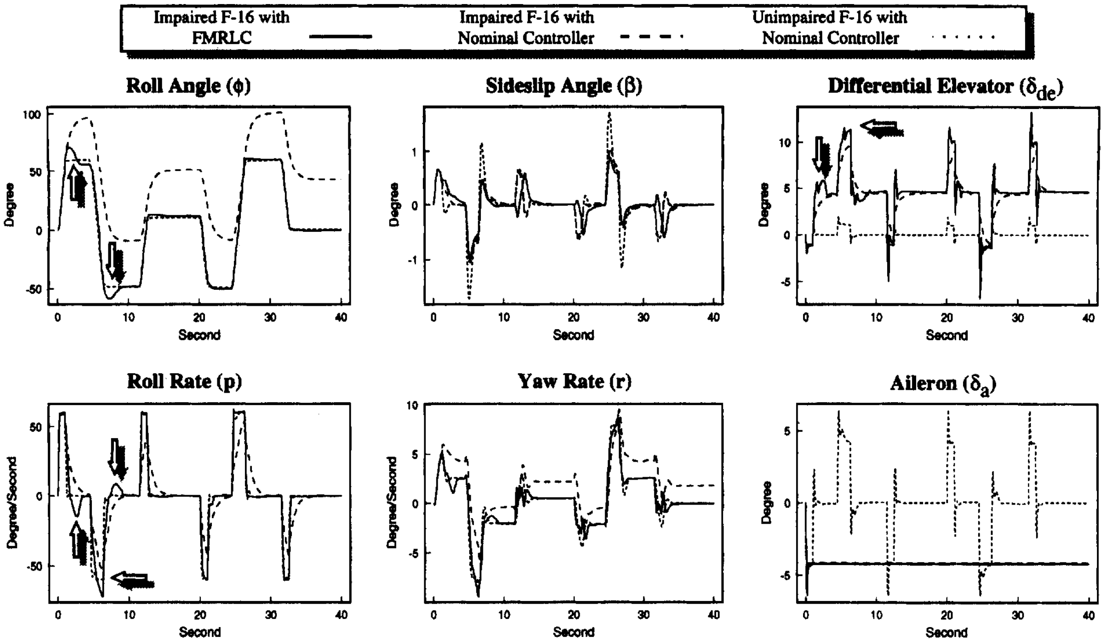


Fig. 11. Impaired F-16 with FMRLC (aileron stuck at 1s).

command sequence is obviously degraded as compared to the unimpaired responses (portions of the roll angle and roll rate responses highlighted with arrows in Fig. 11), but improves as time goes on. As shown in the actuator responses in Fig. 11, the differential elevator (δ_{de}) swings between -1.30 and 10.00 with a bias of about 4.5° for the impaired aircraft with FMRLC. The actuation of the differential elevator replaces the original function of the aileron with the bias so that the effect of the failure is cancelled. In other investigations we showed that if the actuator rate saturation limit was doubled, then performance can be significantly enhanced over what is shown in Fig. 11.

IV. Supervised FMRLC for Reconfigurable Control

The FMRLC designed in the previous Section gives promising results; however, the learning process takes considerable time (approximately nine seconds as shown in Fig. 11) to completely reconfigure the control laws due to the actuator failure. This time requirement arises from the fact that 1) we are forced to use a "slower" learning mechanism (i.e., one with a small value of m_0) to try to ensure stable operation for a wide variety of failures, and 2) the FMRLC does not use special information about the actuator failure (e.g., information about when the failure occurs and where the actuator is stuck). Basically, regardless of what kind of failure occurs, the FMRLC of the previous Section will try to redistribute the remaining control authority to the healthy channels so that the aircraft will behave, if not exactly the same, as closely as possible to its unimpaired condition. It is, however, somewhat unreasonable to expect the FMRLC to achieve performance levels comparable to the unimpaired case if, for example, the aileron becomes stuck at near full deflection. For such failures we would find it acceptable to achieve somewhat lower performance levels.

In this Section, we will show that if FDI information is exploited, a supervisory level can be added to the FMRLC *to tune the reference model characterization* of the desired performance according to the type of failure that occurred. We begin by investigating a supervised FMRLC technique that uses limited FDI information (in [25] we study the case where it is assumed that perfect FDI knowledge is available). Next, utilizing the approach and results in [26], [27] we explain how a fuzzy system can be used for failure estimation and can be coupled with the fuzzy supervisor so that perfect knowledge about failures is not necessary.

A controller is said to be "supervised" if the controller is modified by a set of directives (e.g., a set of heuristic rules) which often results from real-world experiences of a human expert. In the FMRLC, if its components (i.e., the fuzzy controller, the reference model, and the learning mechanism) are changed by a set of heuristic rules, then it is said to be a "supervised" FMRLC. In the F-16 aircraft, an experienced pilot or an aircraft control engineer may have some rule-of-thumb control strategies available to cope with certain failure situations. In the following sections we will investigate if via a supervisory level such heuristic reconfiguration rules can be used to modify the FMRLC to accelerate the learning and ensure that the best possible performance is achieved for the given failure.

A. FMRLC Supervision with Limited FDI Information

We will first consider the case where only limited failure information is available. In particular, we assume that we

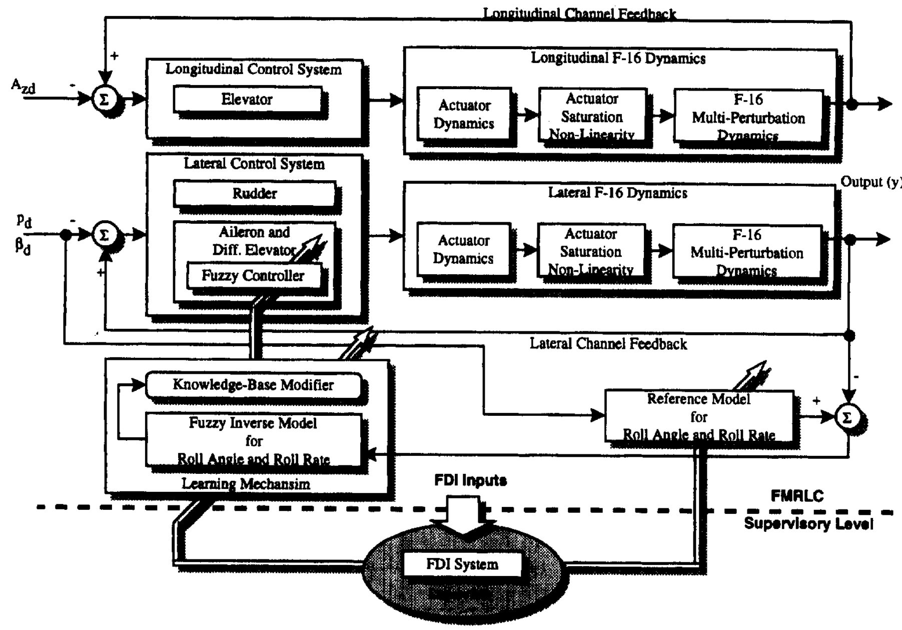


Fig. 12. Supervised FMRLC using FDI information.

have an FDI system that can only indicate whether or not an actuator has failed; hence, while it indicates when an actuator failure occurs it does not indicate the severity of the failure (also we assume it provides no false alarms). A supervised FMRLC that uses such failure information is shown in Fig. 12. The FMRLC supervisor consists of an FDI system, which gives binary information about an actuator failure, and a control supervisor which will tune the reference model and learning mechanism when an actuator failure is detected. It will be assumed that the FDI system can provide its binary failure information within one sampling period after the failure occurs. Simulations (omitted due to space constraints) showed that the performance of the system degraded only to the level obtained by the unsupervised FMRLC when a 1/2 s delay in obtaining FDI information was induced. However, if extremely long delays are induced, then the controller needed re-tuning to maintain adequate performance. Generally, we found that achievable performance levels were directly proportional to the length of the delay in obtaining FDI information.

Due to the fact that only limited knowledge about the nature of the actuator failure is provided by our FDI system, it is not possible to design complex supervision strategies. In general, there are two approaches to supervise the FMRLC:

1) Given limited information, one possible supervision strategy is to increase the learning capabilities of the FMRLC (i.e., increase the output gain m_0 of the fuzzy inverse model) so that the failure recovery can be accelerated. However, in order to obtain an adequate increased output gain for the fuzzy inverse model, the reference model needs to be "slowed down" to try to ensure stability. Therefore, the reference models must be designed much more carefully to represent achievable performance levels of the impaired aircraft. Intuitively, the pilot will not push the performance of an impaired aircraft to the same level as its unimpaired condition. In the case of an aileron actuator malfunction, the pilot loses the primary roll control effector. Even though the differential elevator can often be used to reconfigure for this failure, the pilot will never expect a full recovery of the original performance, realizing that the differential elevators alone do not have enough control authority to overcome the failure and replace all the functions originally designed for the ailerons. In all cases, when an aircraft loses a control surface, it loses some performance. Hence, there is a general decrease in performance expectations from the pilot, which can be mimicked by the change in the functionality provided by the reference model in the FMRLC design. Hence, as soon as a failure is detected, a different set of reference models should be used. The second/third order reference models chosen for the FMRLC have two parameters available for modification; they are ζ and ω_n (which are 0.85 and 14.14, respectively, in Section III-B-2). In the case of actuator failures, a pilot would expect the aircraft to react in a much slower fashion with overdamped responses. Hence, this expectation is translated to a larger ζ (i.e., $\zeta > 0.85$) and a smaller ω_n (i.e., $\omega_n < 14.4$). We find that the values $\zeta = 1.0$ and $\omega_n = 10.0$ are reasonable choices, that is, do not jeopardize performance too much. With this set of slower reference models, the fuzzy inverse model is retuned (using the approach introduced in the previous section), and it is found that the output gain m_0 is 0.6. Here, by

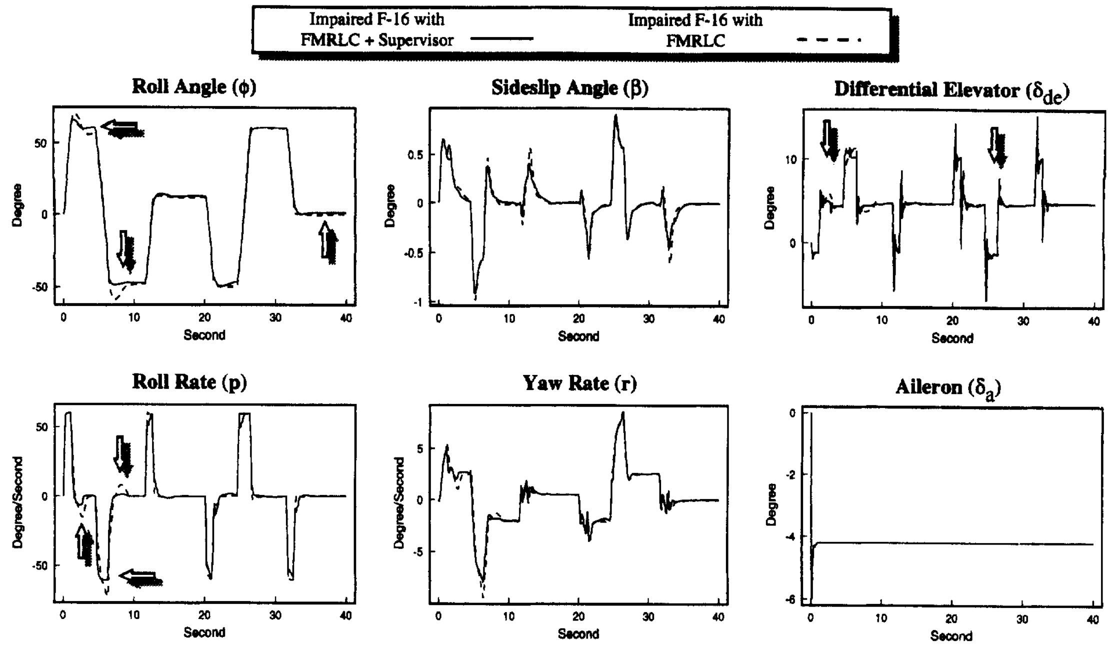


Fig. 13. Impaired F-16 with supervised FMRLC (aileron stuck at 1 s) using limited FDI information.

reducing the performance requested by the reference model, the learning capabilities are increased by a factor of six. This greater learning capacity will then speed up the control reconfiguration process.

2) As an alternative approach, the fuzzy controller can be modified directly in the direction of how the learning mechanism will do the control reconfiguration. Referring to the F-16 nominal controller of the lateral channel as shown in Fig. 3, the effectiveness of the differential elevators as the roll effector is actually depressed by a factor of four compared to the ailerons (see the 0.25 factor between the δ_a and δ_{de} controller outputs). Thus in the control reconfiguration process the learning mechanism of the standard FMRLC needs to bring the effectiveness of the differential elevators up to the level of the ailerons, and then further fine tune the control laws (i.e., the rule-base of the fuzzy controller) to compensate for the actuator failure. With this process in mind, another approach to accelerate the learning process is to assist the learning mechanism in increasing the effectiveness of the differential elevators. This approach can be achieved simply by increasing the output gain of the fuzzy controller for the differential elevators by a factor of (for example) four as soon as the stuck aileron actuator is detected (i.e., increase the domain interval of the output universe of discourse by four times). Using this direct controller modification, instead of using a slower learning mechanism to achieve the same goal, the control reconfiguration process will definitely be accelerated.

Notice that in the first approach, the learning capabilities are increased by reducing the requirements posed by the reference model, whereas the second method allows a direct change in the configuration of the controller itself. The results of applying these two supervision approaches are actually similar and in both cases it is necessary to incorporate them in a fashion that tries to ensure stability. After some simulation studies, it was found that it is best to use a combination of the two methods. The output gain $m_0 = 0.3$ is chosen with the slower reference model given above in approach 1, and the controller output gain of the differential elevators is increased by a factor of two to 0.5. This choice reflects a moderate contribution from each approach, but detailed simulation studies show that virtually any combination of the two approaches will also work with minor differences. Hence, as soon as the FDI system detects the aileron failure, the expert supervisor will switch the reference models, increase the output gain of the fuzzy inverse model, and alter the fuzzy controller as described above.

Fig. 13 shows the responses of the F-16 using the supervised FMRLC. By comparing to the responses where no FDI information is used (see the dotted line in Fig. 13), the results show improvements in that there are less oscillations in the first 9 s (see, e.g., the arrows in the roll rate and the roll angle plots) compared with the unsupervised case in Fig. 11. The supervised FMRLC ensures that the system follows the "slower" reference models in case of failure, which prevents the controller from pushing the aircraft beyond its capabilities. By allowing the FMRLC to learn the failure situation at a higher rate, the actuator response of the differential elevators is more active than that of

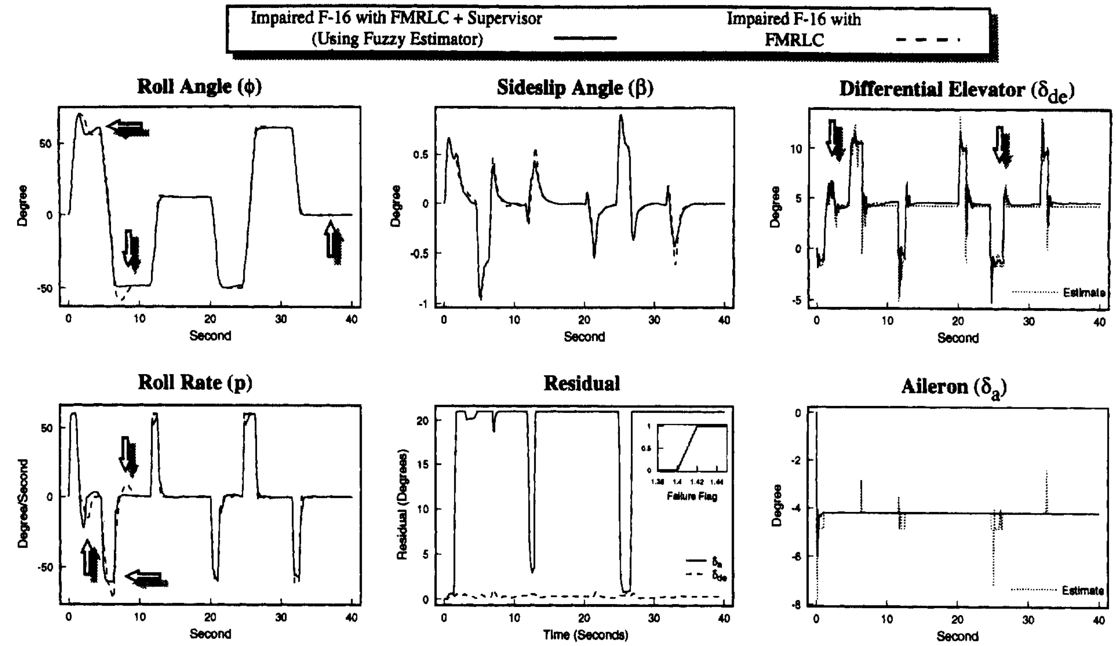


Fig. 14. Impaired F-16 with supervised FMRLC (Aileron stuck at 1 s) that uses an estimate of the failure time and magnitude from the fuzzy estimator.

the response in the FMRLC without the supervisor (see the arrows in the differential elevator plots in Fig. 13). However, the choice of a set of slower reference models results in a larger steady state error in the roll angle after and during the maneuver (see the arrows in the roll angle plot in Fig. 13). This is due to the fact that the slower roll rate model causes the final value of the roll angle to shift. This phenomena is a characteristic of the new set of reference models. From the simulation results, this study shows that a "slower" reference model for FMRLC will often give a less oscillatory overall response, and clearly there exists a trade-off between performance and stability for an impaired aircraft.

B. FMRLC Supervision using FDI Information from a Fuzzy Estimator

Up to this point we have assumed a very specific and idealized form for the FDI subsystem in the implementation of the FMRLC supervisor so that the results would not be limited to a particular type of FDI methodology. In this Section we implement FDI for the F-16 aircraft in conjunction with the FMRLC supervisor using a method for fuzzy failure estimation in [26], [27] (due to the applications focus of this special issue, we provide little background information on fuzzy estimation techniques and refer the interested reader to [26], [27], [32]–[34]). Note that as indicated in the Introduction, the purpose of this section is to investigate the use of fuzzy systems for failure identification as an alternative to conventional FDI techniques. The primary reason for considering fuzzy techniques is due to their lack of dependence on linear mathematical models of the aircraft. Full evaluation of the best approach to FDI that includes theoretical and implementation based comparative analyses with the approaches in, for example [6]–[22], is beyond the scope of this paper since the focus here is on applications of fuzzy logic.

The fuzzy estimator in [26], [27] (which builds on the ideas in [32]–[34]) is constructed by a training algorithm that adjusts the parameters of a fuzzy system so that it approximates a functional mapping represented by a set of M input-output training data pairs that characterize the association between observed aircraft behavior z and failure conditions $\hat{y}$. The input-output training is specified by: 1) inducing an actuator failure (e.g., aileron stuck at $-4.2°$ at $t = 1$ s), 2) gathering measurable aircraft variables and using these to form the input portion of the training data z, and 3) setting the output portion of the training data $\hat{y}$ to the failure condition (e.g., $-4.2°$). Failing the aircraft over a range of M values (e.g., stuck actuator positions) provides a set of input-output training data that provides an association between observed aircraft behavior and the failure which induced this behavior. Hence, training the fuzzy system to approximate this association represented in the input-output training data provides a fuzzy estimator for actuator failures. Basic properties of the fuzzy system, which implement the estimator, ensure that if the observed aircraft behavior is different, then interpolation will provide a reasonable estimate of the failure. Due to space limitations we must refer the reader to [26] for the full details on the development of the fuzzy estimator which provides the estimates of the time and magnitude of the actuator failure.

Fig. 14 shows the results using the fuzzy failure estimator information in the supervisor for an aileron failure (similar results were obtained for a differential elevator failure and for failures which occur at different times). The "residual" plot in the figure shows the residuals generated by using the fuzzy estimator with a windowed-median filter. The companion subview is the decision flag (a value of "0" implies no failure, "1" implies an aileron actuator failure) which indicates the detection of actuator failures based on the selected threshold. Notice that the actuator plots in Fig. 14 (the differential elevator plot and the aileron plot) include the results from the fuzzy estimator. We see that the aileron estimates are accurate enough to detect a failure and that when there is an aileron failure the estimator for the differential elevator performs reasonably well. The results are actually similar to those obtained in the case where we induced a delay in obtaining FDI information.

Although we experienced some success with fuzzy estimation, it is important to note that such a technique is dependent on having access to rich enough input-output training data which contains information for failures we wish to consider. This technique is also very dependent on the controller as we used command inputs from the controller to perform estimation. Our simulation results in [26] actually show that the fuzzy estimator can accurately discriminate between aileron and differential elevator failures. Clearly, there is the need to more fully investigate the use of fuzzy estimation techniques for FDI on aircraft by studying 1) the effect of approximation error tolerance on performance of the estimator and reconfiguration strategy, 2) development of estimators for a wider range of failures, 3) the possibility and effects of false alarms, and 4) the issue of simultaneous and intermittent failures.

V. Concluding Remarks

We have shown that the FMRLC introduced in [1]–[5] can be used to reconfigure the control laws on an aircraft in response to certain actuator failures. We began by discussing the F-16 failure simulation testbed and overviewing the FMRLC technique. Then we introduced a new design procedure for the FMRLC using the aircraft control problem as an example. It is important to note that this design method for the FMRLC is not limited to reconfigurable controller applications. It is a general intuitive design procedure for the FMRLC that 1) uses conventional control laws to initialize a fuzzy controller, 2) tunes the fuzzy inverse model by viewing it as a second controller, and 3) chooses the fuzzy inverse model output gain in terms of the learning capabilities possessed by the fuzzy controller and dictated by the particular application. Following the introduction and application of the design procedure, we showed that the FMRLC was able to compensate for the effects of aileron failures.

Next, we showed that failure information can be effectively used in the supervision of the FMRLC to provide high performance reconfigurable control. In particular, our investigation began by examining the use of limited FDI information in supervision of the FMRLC (in [25] we showed that performance could be improved if perfect FDI information was assumed available). Following this, we employed a fuzzy estimation technique developed in [26], [27] to detect and identify aileron and differential elevator failures. We showed that even with estimated failure information we were able to recover comparable reconfiguration performance as compared to the case when an FDI delay was introduced.

Overall, our results provide: 1) a new supervisory adaptive approach for aircraft control law reconfiguration that is "performance adaptive" in the sense that it tries to recover the best possible performance depending on the type of failure that has occurred, 2) an approach that exploits the use of failure information in control reconfiguration (to date, most often it is assumed that perfect FDI information is always available), and 3) an approach that does not heavily rely on the availability of models of the failed aircraft to reconfigure the control laws (to date, it is often assumed that models of the failed aircraft are available for on-line redesign of the control laws).

While in some respects the supervised FMRLC approach appears promising, there is a significant need to investigate:

- the performance of the supervised FMRLC for a more complete nonlinear simulation model, and for a wider range of failures (we have obtained adequate results for certain actuator failures during the loaded roll command sequence where the actuator suddenly and uncontrollably swings to near maximum deflection) over more aircraft operating conditions;
- stability and convergence issues; and
- robustness issues for the developed techniques including: 1) determination if acceptable plant parameter variations can cause the failure estimation techniques to produce false alarms, and 2) determination if the reconfiguration strategies can accommodate for other types of failures.

In addition, there is a significant need to perform comparative analyses between some of the conventional approaches to reconfigurable control and FDI, and the fuzzy reconfigurable control and fuzzy FDI approaches, in order to carefully address issues associated with computational complexity. Moreover, issues related to proper interface between the pilot and the reconfiguration strategy are critical to the development of a reconfigurable control system.

Acknowledgment

The authors gratefully acknowledge the assistance of Wright Laboratories for providing the aircraft model and nominal control laws. In addition, the authors would like to thank S. Chiu of Rockwell International Science Center, Thousand Oaks, CA, for fruitful discussions on this research.

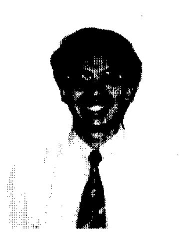
References

[1] J. R. Layne and K. M. Passino, "Fuzzy model reference learning control," in *Proc. IEEE Conf. Control Applications*, pp. 686–691, Dayton, OH, Sept. 1992.

[2] ——, "Fuzzy model reference learning control for cargo ship steering," *IEEE Control Syst. Magazine*, vol. 13, no. 6, pp. 23–34, 1993.

[3] J. R. Layne, K. M. Passino, and S. Yurkovich, "Fuzzy learning control for antiskid braking systems," in *Proc. IEEE Conf. on Decision Control*, Tuscon, AZ, Dec. 1992, pp. 2523–2528.

[4] ——, "Fuzzy learning control for antiskid braking systems," *IEEE Trans. Control Syst. Technol.*, vol. 1, June 1993, pp. 122–129.
[5] J. R. Layne, "Fuzzy model reference learning control," Dept. Elect. Eng., The Ohio State Univ., Mar. 1992.
[6] A. S. Willsky, "A survey of design methods for failure dectection in dynamic systems," *Automatica*, vol. 12, no. 6, pp. 601–611, Nov. 1976.
[7] J. C. Deekert, M. N. Desai, J. J. Deyst, and A. S. Willsky, "Sensor failure identification using analytical redundancy," *IEEE Trans. Automat. Control*, vol. 22, pp. 795–803, Oct. 1977.
[8] J. L. Weiss, D. P. Looze, J. S. Eterno, and D. B. Grunberg, "Initial design and evaluation of automatic restructurable flight control system concepts," NASA Langley Res. Center, NASA Contractor Rep. 178064, ALPHATECH, Inc. Burlington, MA, June 1986.
[9] V. F. Gavito and D. J. Collins, "Application of eignestructure assignment to self reconfiguring aircraft MIMO controllers," in *AIAA, Proc. Guidance, Navigation, and Control Conf.*, Aug. 1987, pp. 1–12.
[10] Y. Ochi and K. Kanai, "Design of restructurable flight control systems using feedback linearization," *J. Guidance, Navigat. Contr.*, vol. 14, no. 5, pp. 903–911, Sept./Oct. 1991.
[11] P. S. Maybeck and R. D. Stevens, "Reconfigurable flight control via multiple model adaptive control method," *IEEE Trans. Aerospace Electron. Syst.*, vol. 27, no. 3, pp. 470–479, May 1991.
[12] C. Y. Huang and R. F. Stengel, "Restructurable control using proportional-integral implicit model following," *J. Guidance, Navigat. Control*, vol. 13, no. 2, pp. 303–309, Mar./Apr. 1990.
[13] W. D. Morse and K. A. Ossman, "Model following reconfigurable flight control system for AFTI/F-16," *J. Guidance, Navigat. Contr.*, vol. 13, no. 6, pp. 969–976, Nov./Dec. 1990.
[14] D. A. Handelman, "An application of artifical intelligence theory to reconfigurable flight control," Dept. Mechanical and Aerospace Eng., Princeton Univ., June 1984.
[15] R. F. Stengel, "Toward intelligent flight control," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 6, pp. 1699–1717, Nov./Dec. 1993.
[16] B. L. Belkin and R. Stengel, "AUTOCREW: A paradigm for intelligent flight control," in *An Introduction to Intelligent and Autonomous Control*, P. J. Antsaklis and K. M. Passino, Eds. New York: Kluwer, 1993, pp. 371–400.
[17] J. J. Gertler, "Survey of model-based failure detection and isolation in complex plants," *IEEE Contr. Syst. Magazine*, vol. 3, no. 6, pp. 3–11, 1988.
[18] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy—A survey and some new results," *Automatica*, vol. 26, no. 3, 1990, pp. 459–474.
[19] R. J. Patton and J. Chen, "Robust fault detection using eigenstructure assignment: A tutorial," in *Proc. 30th Conf. Decision and Control*, Brighton, UK, Dec. 1991, pp. 1143–1147.
[20] R. Patton, P. Frank, and R. Clark, *Fault Diagnosis in Dynamic Systems: Theory and Applications.* Englewood Cliffs, NJ: Prentice-Hall, 1989.
[21] H. E. Rauch, R. J. Kline-Schoder, J. C. Adams, and H. M. Youssef, "Fault detection, isolation and reconfiguration for aircraft using neural networks," in *Proc. AIAA Conf. Guidance, Navigation, and Control*, Monterey, CA, Aug. 1993.
[22] K. M. Passino and P. J. Antsaklis, "Fault detection and identification in an intelligent restructurable controller," *Intelligent and Robotic Syst.*, vol. 1, no. 1, pp. 145–161, June 1988.
[23] W. A. Kwong, "Fuzzy learning control: Techniques and applications," M.S. thesis, Dept. Elect. Eng., The Ohio State Univ., June 1994.
[24] W. A. Kwong, K. M. Passino, and S. Yurkovich, "Fuzzy learning systems for aircraft control law reconfiguration," in *IEEE Int. Symp. on Intell. Control*, Columbus, OH, Aug. 1994, pp. 333–338.
[25] W. A. Kwong, K. M. Passino, E. G. Laukonen, and S. Yurkovich, "Expert supervision of fuzzy learning systems with applications to reconfigurable control for aircraft," in *Proc. IEEE Conf. on Decision and Control*, Orlando, FL, 1994, pp. 4116–4121.
[26] E. G. Laukonen and K. M. Passino, "Fuzzy systems for function approximation with applications to failure estimation," in *Proc. IEEE Int. Symp. Intell. Control*, Columbus, OH, Aug. 1994, pp. 184–189.
[27] E. G. Laukonen, "Training fuzzy systems to perform estimation and identification," M.S. thesis, Dept. Elect. Eng., The Ohio State Univ., 1994.
[28] W. A. Kwong, K. M. Passino, and S. Yurkovich, "Fuzzy supervisory systems for aircraft applications," Dept. Elect. Eng., The Ohio State Univ., Columbus, OH; Ohio Aerospace Institute, Colloborative Core Res. Program Final Rep., Jan. 1994.
[29] C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller—Parts I–II," *IEEE Trans. Systems, Man. Cybern.*, vol. 20, no. 2, Mar./Apr. 1990, pp. 404–435.
[30] V. G. Moudgal, W. A. Kwong, K. M. Passino, and S. Yurkovich, "Fuzzy learning control for a flexible-link robot," to be published in *IEEE Trans. Fuzzy Syst.*, vol. 3, May 1995.
[31] H. Farrell and W. Baker, "Learning control systems," in *An Introduction to Intelligent and Autonomous Control Systems*, P. J. Antsaklis and K. M. Passino, Eds. New York: Kluwer, 1993, pp. 237–262.
[32] L.-X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst., Man. Cybern.,*, vol. 22, no. 6, pp. 1414–1427, Nov. 1992.
[33] ——, "Back-propagation fuzzy system as nonlinear dynamic system identifiers," in *Proc. IEEE Conf. Fuzzy Syst.*, San Diego, CA, 1992, pp. 1409–1418.
[34] ——, "Fuzzy basis functions, universal approximation and orthogonal least-squares learning," *IEEE Trans. Neural Networks*, vol. 3, pp. 1–8, Sept. 1992.



Waihon A. Kwong received the B.S. and M.S. in electrical engineering from The Ohio State University in 1992 and 1994, respectively.

He has worked as a project engineer at Sun's Engineering, Inc., designing quality control systems in paper and pulp industry. He is presently employed by Epsilon Lambda Electronics Corp., Chicago, IL, as a design engineer and is using DSP's for a radar detection and identification system for automotive applications.

Kevin M. Passino (Member, IEEE) received the B.S.E.E. from Tri-State University in 1983 and the M.S. and Ph.D. in electrical engineering from the University of Notre Dame in 1989.

He has worked in the control systems group at Magnavox Electronic Systems Co., Ft. Wayne, IN, on research in missile control and at McDonnell Aircraft Co., St. Louis, MO, on research in flight control. He spent a year at Notre Dame as a Visiting Assistant Professor and is currently an Assistant Professor in the Dept. of Electrical Engineering at The Ohio State University. He is an Associate Editor for the IEEE Transactions on Automatic Control; served as the Guest Editor for the 1993 IEEE Control Systems Magazine Special Issue on Intelligent Control; is currently a Guest Editor for a special track of papers on Intelligent Control for IEEE Expert Magazine; and is on the Editorial Board of the *International. Journal for Engineering Applications of Artificial Intelligence.* He is co-editor (with P. J. Antsaklis) of the book *An Introduction to Intelligent and Autonomous Control* (Kluwer, 1993). His research interests include intelligent and autonomous control techniques, nonlinear analysis of intelligent control systems, failure detection and identification systems, and scheduling and stability analysis of flexible manufacturing systems.

Dr. Passino is a member of the IEEE Control Systems Society Board of Governors. He also serves as the IEEE Control Systems Society Liaison to the IEEE Press and has served in the past as the Chairman of Student Activities for the IEEE Control Systems Society. He is the Publicity Co-Chair for the IEEE Conference on Decision and Control in Japan in 1996; was a Program Chairman for the 8th IEEE International Symposium on Intelligent Control, 1993; and he is serving as the Finance Chair for the 9th IEEE International Symposium on Intelligent Control.

Eric Laukonen received the B.S. and M.S in electrical engineering from The Ohio State University in 1992 and 1994, respectively.

He has worked for General Electric Lighting developing systems to manufacture light bulbs. He has also worked for Battelle Memorial Institute, where he developed computer simulations and computer design tools for ultrasonics and other physical systems. He is presently employed by Motorola Government Systems and Technology Group in Phoenix, AZ, and is part of a team developing satellite based global cellular communications.

Stephen Yurkovich (Senior Member, IEEE) received the B.S. degree in engineering science from Rockhurst College in Kansas City, MO, in 1978, and the M.S. and Ph.D. degrees in electrical engineering from the University of Notre Dame in 1981 and 1984, respectively.

He held teaching and postdoctoral research positions at Notre Dame in 1984, and a Visiting Associate Professor position there in 1992. In 1984 he moved to the Department of Electrical Engineering at The Ohio State University, where he is currently Associate Professor. His research, published in more than 100 technical journal articles and conference papers, includes work in the areas of system identification and parameter set estimation for control, and fuzzy logic for control, in application areas including flexible mechanical structures, intelligent vehicle highway systems, and automotive systems. He has authored the text *Control Systems Laboratory* (Kendall-Hunt, 1992).

In addition to being General Chair for the 1996 Conference on Control Applications, and Program Chair for the 1997 American Control Conference, Dr. Yurkovich has held numerous positions within the IEEE Control Systems Society: He was chairman of the Standing Committees on Student Activities and on Publications, is an elected member of the Board of Governors, was an Executive Officer in 1993, and has been elected as Vice President for Publications for 1995. He is currently Editor-in-Chief of IEEE CONTROL SYSTEMS.