A Multiobjective G.A./Fuzzy Logic augmented flight controller for an F16 aircraft.

P. Stewart *Senior Member, IEEE*, D. Gladwin, M. Parr, J. Stewart

Abstract— **An investigation is made in this paper of the possibility of enhancing the performance of controllers of unstable systems while retaining safety critical function. In this case, a General Dynamics F16 fighter is considered in simulation. A fuzzy logic controller is designed and its membership functions tuned by Multiobjective Genetic Algorithms in order to design an augmented flight controller with enhanced manouverability which still retains safety critical operation. The controller is assessed in terms of pilot effort and thus reduction of pilot fatigue. The controller is incorporated into a six degree of freedom real-time flight simulator, and flight tested by a qualified pilot instructor.**

I. Introduction

The General Dynamics F16 Fighting Falcon jet fighter was one of the first widespread implementations of fly-by-wire and represented at the time of its introduction a step change in terms of its inherently unstable airframe and computer flight controller. Of particular interest in terms of pilot efficiency is the amount of pilot effort in terms of corrections which have to be input to the flight controls to achieve trajectory tracking. The F16 flight controller is by its intrinsic nature conservative, operating the airframe within a relatively modest envelope when compared to the potential capabilities of the airframe. In this paper, the standard controller is augmented with an auxiliary controller designed to increase tracking performance while staying within the reasonable bounds of stability and safety critical operation. Flight control and fighter manoeuvre have previously been considered using Radial Basis Function Networks and Lyapunov functions [1],[2]. However, the simulations used relatively simple nonlinear models and flight trajectories. In this paper, a fuzzy logic controller is developed on a real-time authentic aircraft model, and is tested by a qualified pilot instructor against a realistic flight plan.

II. Fuzzy Logic controller

Upon consideration of the number of flight performance variables related to the calculation of the final flight surface position for any pilot input demands, a Fuzzy Logic Flight Management Controller was developed [3]. This management controller would assess the pilot's demands and apply rules to create an output that utilizes the full potential of the F-16's 'relaxed static stability' [4]. The final output would be 'de-fuzzified' by a real-time updated output membership set that is calculated from the flight performance variables to give optimum flight surface deployment values for that instance.

The fuzzy logic controller was designed through a process of five steps:

- Define the input and control variables.
- Define the condition interface. ('Fuzzification')
- Design the rule-base.
- Design the computational unit.
- Determine rules according to which fuzzy control statement can be transformed into crisp action. ('De-Fuzzification')

A. Input, Control Variables and Condition Interface

The input variables were selected as the pilot pitch, roll and yaw rate commands as well as the corresponding demand rate-of change. The control variables were the aileron, elevator and rudder output demands to the physical control section of the F-16 model. The Condition Interface is the method in which observations of the processes are expressed as members of the fuzzy linguistic sets. When creating the membership sets for the fuzzy logic management controller the following points were considered:-

- It was decided that five classes would be defined for the input variables and that the output membership set would comprise of seven classes.
- The use of narrower membership functions results in faster responses. However large oscillations overshoot and settling-times appear when narrow membership functions are used. Steady-state may also never be achieved when using very narrow membership sets.
- Piecewise linear functions are popular because they are simple to construct but for smoothness of response, Gaussian functions are represented.
- The final membership sets were constructed using Gaussian functions.

The pitch, roll and yaw membership sets were constructed in an identical manner, although the range the membership sets covered was altered according to the relative range of the input signals. Representative function plots for variable 'roll' are shown in figures 1,2 and 3.

B. Rule Base

The aim of this project was the analysis in simulation of the fly-by-wire flight controller of an F-16 Fighting Falcon Fighter Jet and through this analysis identification of areas for improved performance. The objectives for the new flight controller were defined as:-

- The aircraft should have acceptable flying qualities as defined using the Cooper-Harper Pilot Rating Scale [5].

Paul Stewart (Corresponding Author) is with the Department of Electronic and Electrical Engineering, University of Sheffield, Mappin St., Sheffield S1 3JD, UK (phone: +44 114 2225841; email: p.stewart@sheffield.ac.uk).

Dan Gladwin, Matt Parr and Jill Stewart are also members of the University of Sheffield UK.

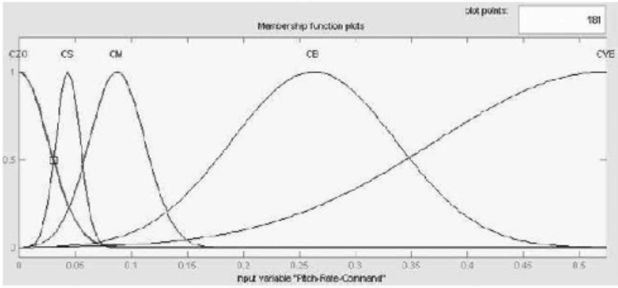

Fig. 1. Roll rate command function

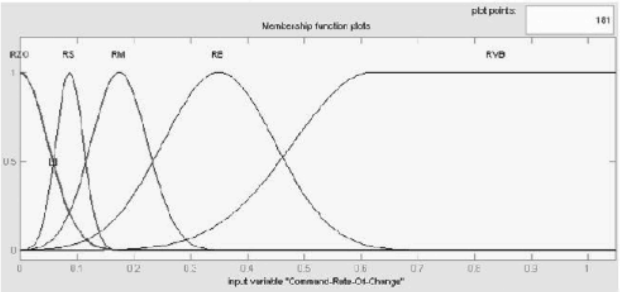

Fig. 2. Rate of change of roll rate function

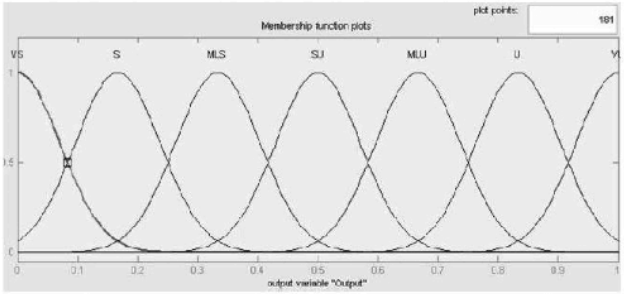

Fig. 3. Roll output function

Output Flight Surface Response		Input Rate Demand				
		Zero Order	Small	Medium	Large	Very Large
Input Acceleration Rate	Zero Order	VS	VS	S	SU	SU
	Small	VS	S	S	MLS	MLU
	Medium	S	S	MLS	MLU	U
	Large	SU	MLS	MLU	U	U
	Very Large	SU	MLU	U	U	VU

Designed rule base

where VS – Very Stable; VU – Very Unstable

S Stable; U - Unstable

MLS More or Less Stable; MLU More or Less Unstable

SU – Stable-Unstable

Fig. 4. Rule base

- The quality of the controller must be an improvement on the original controller.
- The aircraft should respond precisely to small absolute stick movements.
- The aircraft should respond rapidly to large rate of change stick demands.
- Low phase lag between cockpit controller and flight control surface
- The controller must not allow the aircraft to become unstable.
- The controller must not react faster than the pilot can think.
- The controller must not allow the aircraft to exceed its characteristics (i.e. Max G)

A number of these objectives can be considered to be subjective and therefore a trained pilot was used in consultation to help decide, in the final assessment of the controller, whether they had been met. The rule base uses the linguistic sets defined by the inputs to determine the optimum output. The design of the rule base consists of determining which rule should be applied under which circumstance. To help decide in this matter a controller specification was developed. The rule base was designed to operate to produce output results in the manner shown in figure 4. The relational surface for the motion controllers is shown in figure 5

C. De-fuzzification

The computational unit utilised Mamdani type fuzzy processing [13]. The computational unit uses the Centre of Area (COG) method [14] to 'de-fuzzify' the results and provide a scale factor which will be applied to the raw pilot input demands and the original controller action to give the desired action.

The COG method has both continuity and un-ambiguity which were considered more desirable than the simpler computational complexity of the mean of maxima method. The scale factor produced by the fuzzy logic flight management controller ranged between one and zero. An output of zero was considered very stable and equivalent to the original controller. An output of one was considered very un-stable and equivalent to there being no flight controller present at all; this was simulated by passing the pilot inputs directly through as surface demands to the physical model.

This new fuzzy logic flight management controller was tested on the Explorer RD and appeared to provide improved manoeuvrability however the performance of the system still needed to be verified.

The testing on the Explorer RD demonstrated the possibility of G-Forces being placed on the aircraft which exceeded the maximum the plane structure could accommodate. The specification specifically did not allow this and for this reason management of the G-Forces experienced was included in the new flight controller. If the aircraft started to approach the maximum or minimum G-Forces allowed, the flight controller would demand the aircraft to respond in a completely stable manner to prevent any structural damage to the aircraft. The controller was tested by inputting small (10% of full demand) and large (95% of full demand) disturbances at low (Mach 0.25) and high (Mach 1.75) speeds to the system with and without the new flight management controller and analysing the results.

The results demonstrated that while the aircraft manoeuvrability may have been increased, its performance in tracking pilot demands was poor and in most cases poorer than

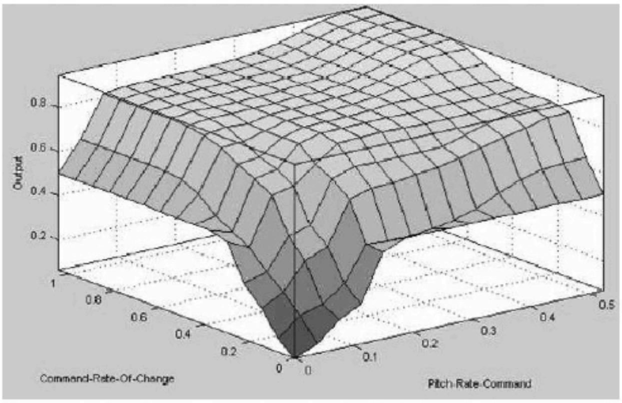

Fig. 5. Relational surface

the original F-16 flight controller; showing evidence of high overshoot and large settling times.

As stated earlier, a fuzzy logic controller represents an expert system and therefore only the expertise of the user who developed it. To improve the performance of the controller it was decided to use Genetic Algorithms to tune the system [6],[7],[8]. Genetic Algorithms (GAs) are a sto-

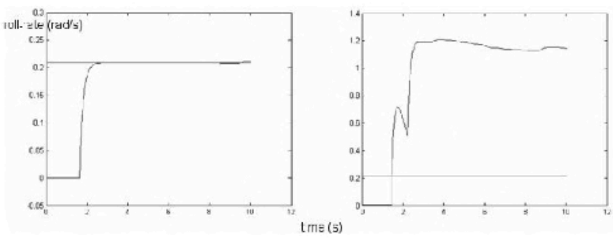

Fig. 6. Rerpresentative roll-rate comparison

chastic global search and optimization methods that mimic natural biological evolution through the principles of natural selection, genetic modification and selective breeding. GAs operate on a population of individuals with each individual representing a possible solution to the optimization problem.

GAs begin with the random initiation of a population. With each generation of a GA, a new set of approximations is created by a process of selection, crossover and mutation. The selection process determines the fittest individuals to go on to the next population. Crossover exchanges the genetic material of two of individuals, creating two new individuals. Mutation changes, at random, the genetic material of an individual. This process leads to the evolution of a population that is better suited to its environment than the individuals from which it was created, just as with natural adaptation.

The solution is usually achieved when a certain number of generations have been reached. GAs are applicable to nonlinear optimization problems which make them ideal for optimally tuning the Fuzzy Logic Flight Management Controller [9].

The genetic algorithm was written in Matlab incorporating the Multi-Objective Genetic Algorithm (MOGA) toolbox [10],[11] and integrating the full F-16 system model including the new fuzzy logic flight management controller [12].

The fuzzy logic controllers for pitch, roll and yaw were each optimised individually using MOGA techniques. It was decided to optimise the membership sets of each controller to attempt to improve the overall system performance. The initial population of individuals was selected at random but within boundaries stipulated by the pilot demands. The population was limited to a hundred individuals to allow for a large spread of values but not increase the processing time significantly. The variables were defined as the parameters of each class in the membership sets. Each class required two parameters to define its Gaussian shape; a location point for the peak value and a width. With 17 classes contained in the three membership sets (*the G-Force membership set was not optimised as this was considered safety critical*) thus the MOGA optimised 34 variables. This phase of the algorithm applies the individuals to the objective functions and measures how well they complete the task. The level to which the individual successfully completed the task is known as its fitness.

The individuals were assigned a fitness based upon the following four objective functions:

- Overshoot
- Rise-Time
- Target Constraint
- Integrated Time Absolute Error (ITAE)

The overshoot and rise time were calculated as shown below. The target constraint was a check to ensure that the controller constructed using the individual's phenotypes (variables) was possible of a response equal to that of the demand. ITAE is a method of analysing the error between the demand signal and the controllers. The integration of time means this objective function especially penalises large settling times as well as large deviations from the demanded signal. Assigning the fitness required ranking the values based on how well they minimised the objective functions. These functions returned a column vector containing the relevant individual fitness values ranked upon the goals and priorities declared in the MOGA's initialisation. The simulation was run using a modified form of the F-16 Simulink model including the new fuzzy logic flight management controller. The model was modified to record all the required data for objective function analysis and to input step commands of varying magnitude. The pitch, roll and yaw axis were each tested independently using small, medium and large pilot rate demands. The initialisation files of the model were modified to place the aircraft level at 25,000 ft and airspeed of 500 knots. The F-16 autopilot functions were appropriately used to hold the aircraft in this position except in the axis being tested. The model was run in response to varying step inputs as discussed for a simulation time of 25 seconds. The individuals in the MOGA were to be used as fuzzy logic controllers in the Simulink model. This portion of the GA chose which values from the population were to be used in the crossover phase. This was achieved based upon the fitness of an individual and the population as a whole. There are a number of different methods to accomplish this; however stochastic universal sampling was selected for this GA.

Individuals are selected based upon their fitness. The better suited they are to the task, the higher the probability they are to be selected for cross over to produce the next generation.

A generation gap of eighty percent was used to allow twenty percent of the population to continue through to the next generation with no modification. This allows preservation of optimum values from the previous generation to ensure that the new generation's performance does not deteriorate when compared to the previous generation. The MOGA created was used to produce optimum controllers for pitch, roll and yaw responses to small, medium and large pilot rate inputs. The results were used to develop optimum pitch, roll and yaw controllers for all pilot rate demands.

The response surfaces produced by the various fuzzy logic flight management controllers were analysed at their significant areas when considering the pilot demand. This analysis allowed the correct shape of surface required for an optimum controller to be determined and these are shown in figures 7, 8 and 9.

The analysis of the surfaces produced raised some interesting results. Most notably that the optimisation did not recommend a highly unstable response to large pilot demands. The MOGA results did not recommend that the system ever experience instability greater than 60% of the maximum available to prevent large overshoot of the required rate demand. Also the results suggested that instability should be used to improve the system performance to small pilot demands. However this was against the initial specification and it was ensured that the final surfaces did not include this attribute. These developed surfaces were tested using step

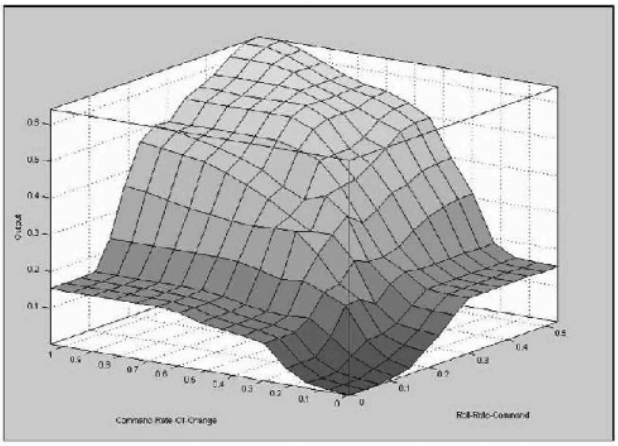

Fig. 7. Pitch controller optimal surface

inputs with the results shown in figures 10, 11 and 12. These results demonstrate that the new optimum fuzzy logic flight management controllers represent an improvement not only on their initial design but also on the initial system in terms of performance in response to pilot rate demands.

III. Rule Base Optimisation

It was decided to investigate the optimisation of the established rule base incorporating the memberships sets recently derived. This involved the creation of a new MOGA to utilise individuals that would construct new rules instead of membership classes. This and the development of a new

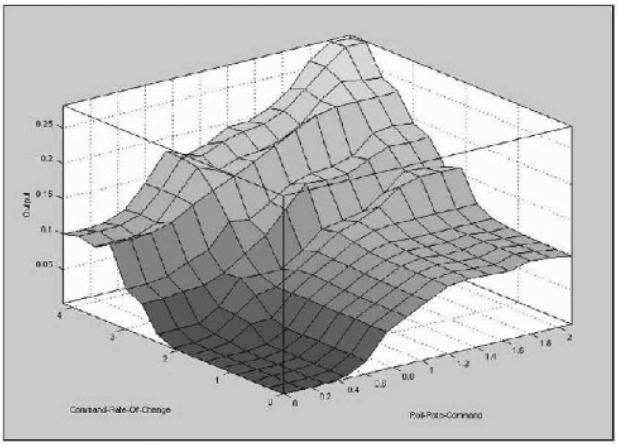

Fig. 8. Roll controller optimal surface

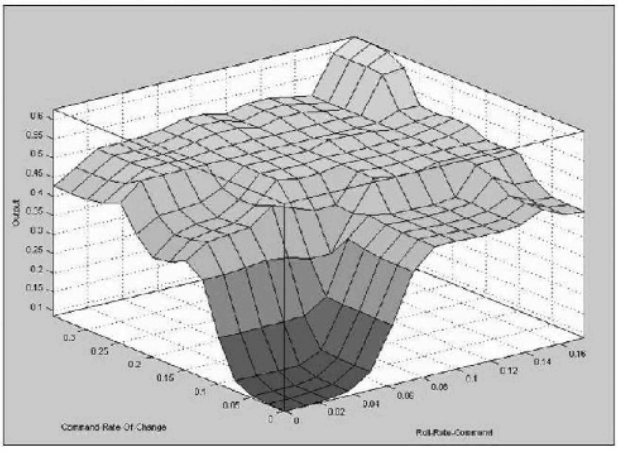

Fig. 9. Yaw controller optimal surface

routine to convert the rules into a suitable format for forming a fuzzy logic controller within Simulink were successfully achieved.

The final results of this analysis returned the original constructed rule-base even when starting from a set of random rules. Though this goes a small way to verifying that the original rule-base does represent an expert system, the results can also be explained by remembering that the membership functions were optimised to the original rule-base in the previously constructed MOGA. This would make it highly likely that when attempting to optimise a rule-base to these membership functions that the originally constructed rule-base would be returned.

IV. Test Flight

In the designing of this improved flight controller it has always been known that the aim was to improve aircraft manoeuvrability. This could be considered to be tested by assessing the following two points:-

- Tracking of pilot rate demands and actual rates achieved for all control surfaces while performing set manoeuvres.
- Total pilot effort required to perform these set manoeuvres.

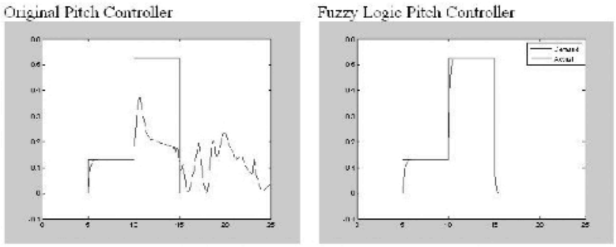

Fig. 10. Pitch controller step response

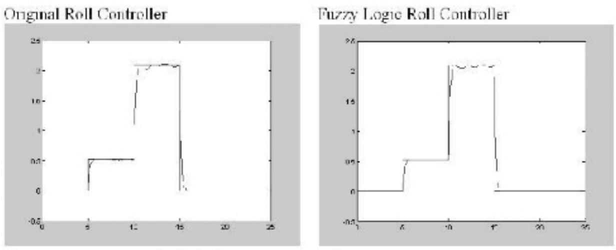

Fig. 11. Roll controller step response

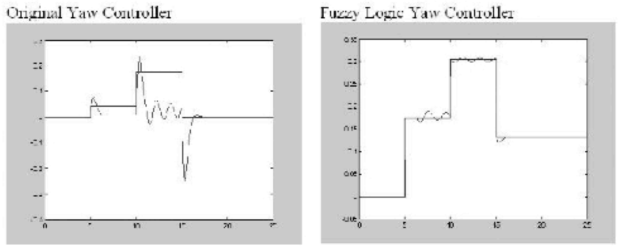

Fig. 12. Yaw controller step response

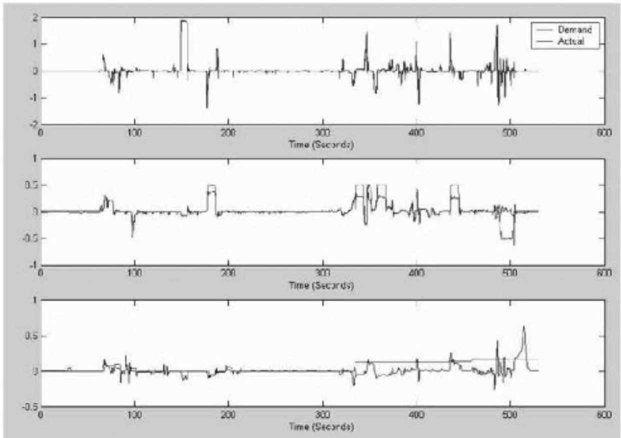

Fig. 13. Test flight performance of the original controller

Correspondence was undertaken with large aeronautical firms to determine if there exists an industry standard set of manoeuvres for assessing aircraft stability. It was discovered that no such standard exists and therefore a test flight would have to be defined.

The designed test flight was constructed as follows:-

- Take-off and turn to a heading of due north while maintaining a 10^o climb.
- Increase climb to 45^0 until a height if 10,000ft is achieved.
- Level off aircraft and accelerate to Mach 1.5.
- Perform two barrel rolls.
- Reduce speed to Mach 0.75 and turn to a heading of due south.
- Intercept drone for one rotation of the airfield.
- Depart from interception on heading of 220^o or 40^o (NB. Runway in Cranfield Model runs from 40^o to 220^0).
- Climb to 1000ft.
- Return to airfield and land safely.

This test flight procedure was performed for analysis only after suitable familiarization with the procedure had been achieved to prevent any influence from pilot skill affecting the results taken. The pilot rate demands and aircraft responses were recorded through use of an inbuilt function of the real-time vehicle flight software. This allowed analysis of recorded real-time flight data with a pilot in the control loop; as apposed to the analysis of data produced through set inputs into a model.

V. Flight Controller Tracking

The relative tracking ability of each system was assessed through the calculation of the integral absolute error between pilot rate demands and aircraft rate responses. The rate demand used for this analysis can be seen for the original controller in figure 13 and the augmented controller in figure 14. Analysing by ITAE, the new controller has a ratio of 0.3842, a reduction of 61.58%.

VI. Pilot Effort

The mental effort required to fly an aeroplane, while meeting its mission objectives, are usually referred to as pilot compensation. If an aeroplane reacts either too slow or too fast to a pilot command, the pilot must compensate for the behaviour by 'adjusting his own gain' or 'leading' the airplane. Clearly a pilot should not have to excessively lead an aeroplane nor should they have too much or too little gain.

The pilot effort was analysed by both considering the absolute integral of pilot demands placed on the aircraft as well as the sum of the differential of these commands. The sum of the differential will provide an indication of the scale of commands made on the system; an indication of whether the pilot is forced to make large corrections due to over/under steer or many small corrections to keep the aircraft correctly trimmed. In which case the ratio of the new controller to old controller is 0.6215, an improvement of 37.85% An analysis of the differential of the pilot inputs to both systems to carry out the same test flight shows a 23.5% reduction in the new controller when compared to the original F-16 flight controller. This would appear to suggest that the new controller required smaller corrections by the pilot to maintain the desired heading.

A. Cooper-Harper Rating Scale

This is a set of criteria used by test pilots and flight test engineers to evaluate the handling qualities of aircraft during

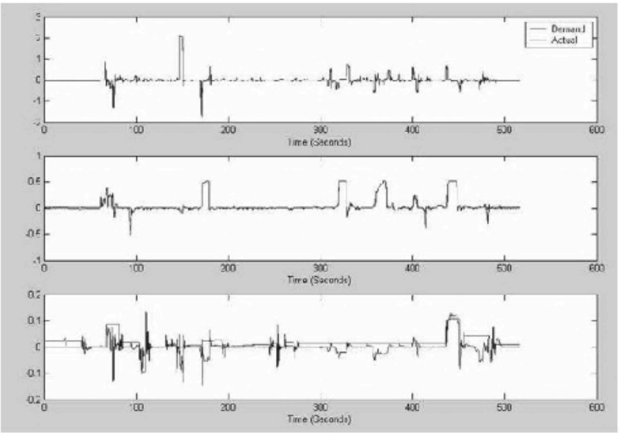

Fig. 14. Test flight performance of the Fuzzy Logic Flight Management Controller

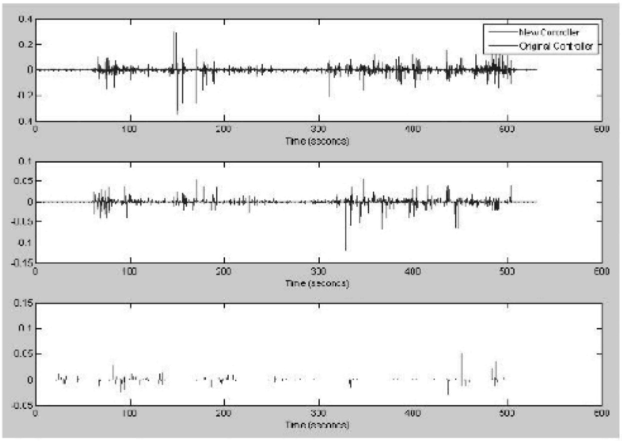

Fig. 15. Differential pilot inputs

flight tests. The Cooper-Harper scale ranges from 1 to 10, with 1 indicating the best handling characteristics for an aircraft, and 10 the worst. The criteria are evaluative and thus the scale is considered subjective. Therefore a rating is only valid when the aircraft is evaluated by an expert. The F-16 flight controller was evaluated by a qualified pilot instructor. Although this individual was not familiar with the aircraft they were an expert in flying a number of aircraft types and their opinion was considered to be authoritative. The pilot rated the aircraft with the new Fuzzy Logic Flight Management Controller as having the highest Cooper-Harper rating of 1.

Appendix: Experimental Hardware

The augmented controller was developed in Simulink offline, then integrated into the real-time operating system of a six-degree-of-freedom flight simulator in the Faculty of Engineering at the University of Sheffield (figure 16). The Simulator is an *Explorer RD* manufactured by CueSim Ltd. (www.cuesim.co.uk).

References

[1] No T.S., Min B.M., Stone R.H. and Wong K.C., 'Control and simulation of arbitrary flight trajectory-tracking', *Control Engineering Practice*, Vol. 13, pp. 601-612, 2005.

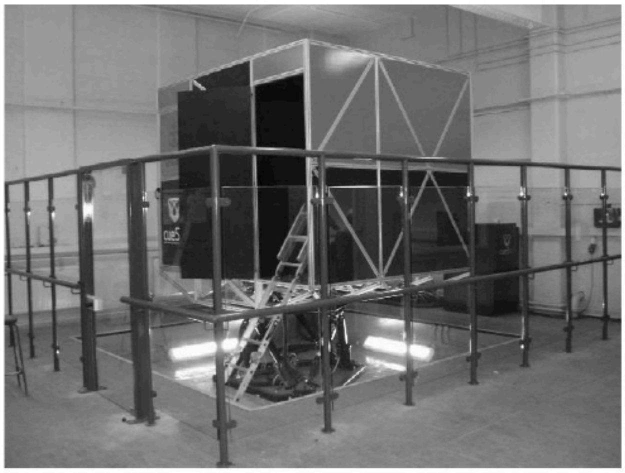

Fig. 16. University of Sheffield Faculty of Engineering Flight Simulator

[2] Li Y., Sundararajan N. and Saratchandran P., 'Neuro-controller design for nonlinear fighter aircraft maneuver using fully tuned RBF networks', *Automatica*, Vol. 37, pp. 1293-1301, 2001.

[3] Oosterom M., Babuska R. and Verbruggen H.B., 'Soft computing applications in aircraft sensor management and flight control law reconfiguration', *IEEE Transactions on Systems, Mand and Cybernetics*, Part C, Vol. 32, no. 2, pp. 125-139, May 2002.

[4] Tischler M.B. (ed.), 'Advances in aircraft flight control' *Taylor and Francis* London, UK, ISBN 0-7484-0479-1, 1996

[5] Cooper G.E. and Harper R.P., 'The use of pilot rating in the evaluation of aircraft handling qualities' *N.A.S.A.* Washington D.C., USA, Technical Note TN-D5153, 1969

[6] Belarbi K., Titel F., Bourebia W. and Benmahammed K., 'Design of Mamdani fuzzy logic controllers with rule base minimisation using genetic algorithms', *Engineering Applications of Artificial Intelligence* Vol 18, No 7, pp 875-880, October 2005

[7] Cordon O., Herrera F., Hoffmann F. and Magdalena L., 'Genetic fuzzy systems: Evolutionary tuning and learning of fuzzy knowledge bases' *Advances in Fuzzy Systems*, World Scientific, Singapore, ISBN 981-02-4016-3, 2001.

[8] Deb K., 'Multi-objective optimisation using evolutionary algorithms', *John Wiley and Sons*, London, 2001.

[9] Blumel A.L., Hughes E.J. and White B.A., 'Fuzzy autopilot design using a multiobjective evolutionary algorithm', *Proceedings of the 2000 Congress on Evolutionary Computation*, Vol. 1, pp. 54-61, 16-19 July 2000.

[10] Fonseca C.M. and Fleming P.J., 'Genetic algorithms for multiobjective optimisation: formulation, discussion and generalisation", *Genetic Algorithms: Proceedings of the Fifth International Conference*, Morgan Kaufmann, San Mateo, CA, pp.416-423, 1993.

[11] Fonseca C.M. and Fleming P.J., 'Multiobjective optimisation and multiple constraint handling with evolutionary algorithms - Part 1: A unified formulation and Part 2: Application example", *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, vol.28, no.1, pp.26-37, 38-47, 1998.

[12] Stewart P., Stone D.A. and Fleming P.J. 'Design of robust fuzzy-logic control systems by multi-objective evolutionary methods with hardware in the loop' *IFAC Journal of Engineering Applications of Artificial Intelligence* Vol.70, no.3, pp.275-284, May 2004.

[13] Ying H., 'A general technique for deriving analytical structure of fuzzy controllers using arbitrary trapezoidal input fuzzy sets' *Automatica* Vol 39, No 7, pp 1171-1184, July 2004

[14] van Broekhouen B. and de Baets B, 'Fast and accurate centre of gravity defuzzification of fuzzy system outputs defined on trapezoidal fuzzy partitions' *Fuzzy Sets and Systems* vol 157, no 7, April 2006