

Military Technical College
Kobry El-Kobba
Cairo, Egypt



12-th International Conference
on
Aerospace Sciences &
Aviation Technology

FUZZY LOGIC CONTROLLER TUNING VIA ADAPTIVE GENETIC ALGORITHM APPLIED TO AIRCRAFT LONGITUDINAL MOTION

Ahmed Momtaz*, Han Chao**

ABSTRACT

Design of an efficient fuzzy logic controller involves the optimization of parameters of fuzzy sets, denormalized gains and proper choice of rule base. There are several techniques reported in recent literature that use genetic algorithms to learn and optimize a fuzzy logic controller. This paper develops methodologies to learn and optimize fuzzy logic controller parameters based on genetic algorithm. The strategies developed have been applied to control integration between LQR and nonlinear Fuzzy PID of F16 aircraft pitch motion control and fuzzy controller developed with the help of iterative learning from operator experience. The results show that Genetic-Fuzzy approaches were able to learn rule base and identify membership function parameters accurately.

KEY WORDS

Intelligent control, Fuzzy logic, Genetic algorithm, Linear quadratic regulator

*PHD Candidate, School of astronautics, Beihang University, Beijing, China

**Professor, School of astronautics, Beihang University, Beijing, China

1. INTRODUCTION

GA uses a direct analogy of natural behavior (Fig. 1), They work with a population of individuals, each reprinting a possible solution to a given problem. Each individual is assigned a fitness score according to how good its solution to the problem is. The highly fit individuals are given opportunities to reproduce, by cross breeding with other individuals in the population. This produces new individuals as offspring, who share some features taken from each parent. The least fit members of the population are less likely to get selected for reproduction, and will eventually die out. Adaptive GAs whose parameters, such as the population, size, cross over probability and mutation probability are varied during the GA is running thus converging to the optimum solution faster.

2. APPLICATION OF GA FUNCTIONS FOR TUNING FUZZY PID CONTROLLER PARAMETERS

As shown in the following figure Fig. 2, the fuzzy PID controller consists of output denormalized gains, input-output membership functions and rule bases, [1], [2]. The main objective of the pre-explained GA procedure is to optimize (minimize) the output performance index (maximize fitness function) of the whole integrated control system used in the pitch control motion of F-16 aircraft. Application of GA MATLAB code combined with SIMULINK control model is implemented to minimize the performance index as explained later [3].

Using GA to minimize the value of the output performance index:

$$J = \int_0^{\infty} e(\text{pitch_sim})^2 dt \tag{1}$$

Where $e(\text{pitch_sim})$ is the pitch angle error to workspace. The previous index can be obtained from the nonlinear simulation of the pitch control model consequently GA can use it as the fitness function.

2.1 Optimizing Fuzzy Output Gains

2.1.1 Using binary system coding

Coding : using 10-bit binary genes to express K_p, K_I, K_D . Table 1 shows the upper and lower gain limits. For example (K_p) from bit 1 000000000(0) , to bit 10 is 111111111(1023). Then string K_p, K_I, K_D to 30-bit binary cluster.

$x: 0000110111 1101110001 0000100010$ express a gene , the former 10-bit express K_p , the second portion express K_I and the third one express the K_D .

Decoding : Cut one string of 30-bit binary string to three 10-bit binary string , then converts them to decimal system values: y_1 , y_2 and y_3 . Table 2 shows the decoding equations.

2.1.2 Evaluation of fitness function

Fitness function is the main criterion of the GA algorithm, as it represents how much the system is optimum and stable. The following equation describes the relation between the fitness function and the performance index.

$$f(K_p, K_I, K_D) = (J(\text{pitch_sim}))^{-1} \tag{2}$$

2.1.3 Design operators

Proportion selection operator , single point crossover operator , basic bit mutation operator.

2.1.4 Parameters of GA

Population size is $M = 20$, generation $G = 30$, crossover probability $P_c = 0.60$, mutation probability $P_m = 0.10$.

Adopting the above steps, after 100 steps iteration, we get the best individuals When $K_P=7.190$, $K_D =2.958$ and $K_I= -2.870$ the output performance index has the minimum value , that is 0.9785 shown in Fig. 3.

2.2 Tuning of Fuzzy Sets

It is presented a universal algorithm for solving a very general class of optimization problems applied to fuzzy sets. From studying how such a GA can be applied to the optimization of fuzzy sets. An appropriate coding, genetic operators (in case that the standard variants are not sufficient), and a fitness measure is presented [3], [4].

2.2.1 Coding fuzzy subsets of an interval

A reasonable resolution for encoding the membership degrees is $n = 8$. Such an 8-bit coding is used in several software systems, too. For most problems, however, simpler representations of fuzzy sets are sufficient. Many real-world applications use triangular and trapezoidal membership functions Fig. 4.

Not really surprising, a triangular fuzzy set can be encoded as

$$\left[\begin{array}{|c|c|c|} \hline c_{n,[a,b]}(r) & c_{n,[0,\delta]}(u) & c_{n,[0,\delta]}(v) \\ \hline \end{array} \right],$$

where δ is an upper boundary for the size of the offsets, for example $\delta = (b - a)/2$. The same can be done analogously for trapezoidal fuzzy sets:

$$\left[\begin{array}{|c|c|c|c|} \hline c_{n,[a,b]}(r) & c_{n,[0,\delta]}(q) & c_{n,[0,\delta]}(u) & c_{n,[0,\delta]}(v) \\ \hline \end{array} \right].$$

In specific control applications, where the smoothness of the control surface plays an important role, fuzzy sets of higher differentiability must be used. The most prominent representative is the bell-shaped fuzzy set whose membership function is given by a Gaussian bell function:

$$\mu (x) = e^{-\frac{(x-r)^2}{2u^2}} \tag{3}$$

The “bell-shaped analogue” to trapezoidal fuzzy sets are so-called radial basis functions:

$$\mu(x) = \begin{cases} e^{-\frac{(|x-r|-q)^2}{2u^2}} & \text{if } |x-r| \geq q \\ 1 & \text{if } |x-r| \leq q \end{cases} \tag{4}$$

Fig. 5 shows a typical bell-shaped membership function. Again the coding method is straightforward, i.e.

$c_{n,[a,b]}(r)$	$c_{n,[\varepsilon,\delta]}(u)$
------------------	---------------------------------

where ε is a lower limit for the spread u . Analogously for radial basis functions:

$c_{n,[a,b]}(r)$	$c_{n,[0,\delta]}(q)$	$c_{n,[\varepsilon,\delta]}(u)$
------------------	-----------------------	---------------------------------

The final step is simple and obvious, In order to define a coding of the whole configuration, i.e. of all fuzzy sets involved, it is sufficient to put coding of all relevant fuzzy sets into one large string. This type of membership functions were used in describing the fuzzy **PID** controller denormalized gains (K_p, K_I, K_D) as it has better representation of system uncertainty.

2.2.2 Coding whole fuzzy partitions

In this work a simple example of an increasing sequence of trapezoidal-triangular fuzzy sets has been presented. Such a “fuzzy partition” is uniquely determined by an increasing sequence of $2N$ points, where N is the number of linguistic values we consider. The mathematical formulation is the following:

$$\mu_1(x) = \begin{cases} 1 & \text{if } x \in [x_0, x_1] \\ \frac{x_2 - x}{x_2 - x_1} & \text{if } x \in [x_1, x_2] \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

$$\mu_i(x) = \begin{cases} \frac{x - x_{2i-3}}{x_{2i-2} - x_{2i-3}} & \text{if } x \in [x_{2i-3}, x_{2i-2}] \\ 1 & \text{if } x \in [x_{2i-2}, x_{2i-1}] \\ \frac{x_{2i} - x}{x_{2i} - x_{2i-1}} & \text{if } x \in [x_{2i}, x_{2i-1}] \\ 0 & \text{otherwise} \end{cases} \text{ for } 2 \leq i \leq N - 1 \tag{6}$$

$$\mu_N(x) = \begin{cases} \frac{x - x_{2N-3}}{x_{2N-2} - x_{2N-3}} & \text{if } x \in [x_{2N-3}, x_{2N-2}] \\ 1 & \text{if } x \geq x_{2N-2} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Fig. 6 shows a typical example with **N = 3** that was used to represent (error, error change) in the proposed fuzzy **PID** controller. It is not wise to encode the values x_i as they are, since this requires constraints for ensuring that x_i are non-decreasing. A good alternative is to encode the offsets:

$$\boxed{c_{n,[0,\delta]}(x_1)} \quad \boxed{c_{n,[0,\delta]}(x_2 - x_1)} \quad \cdots \quad \boxed{c_{n,[0,\delta]}(x_{2N-2} - x_{2N-3})}$$

2.2.3 Genetic operators

Since this GA can only deal with binary representations of fuzzy sets and partitions, all the GA operators that mentioned before are also applicable here. However, that the offset encoding of fuzzy partitions is highly epistemic. More specifically, if the first bit encoding x_1 is changed, the whole partition is shifted. If this results in bad convergence, the crossover operator should be modified. A suggestion can be found in [5]. Fig. 6 shows an example what happens if two fuzzy partitions are crossed with normal one-point crossover. Fig. 7 shows the same for bitwise mutation.

2.2.4 Rescaling process (Decoding of all fuzzy partitions)

Using the following simple equation to normalize the membership variables (error, error change) by:

$$x = (2 \times y - 1020) / 1020$$

Therefore the optimum (normalized) membership functions can be obtained for both (error, error change). The following figures (Fig. 8a, Fig. 8b, Fig. 8c, Fig. 8d, Fig. 8e) illustrate the optimum input-output membership functions that obtained from the adaptive genetic algorithm for 40 samples and 20 generations at some operating point of (250 ft/s speed and 1000 ft altitude).

2.3 Rule Base Weight Optimization

The genetic algorithm has been applied to 100 samples of input-output simulation data obtained from fuzzy PID controller. The rules obtained by the algorithm are shown below. The values in parentheses after each rule represent the firing strength of the corresponding rule.

1. if (e is NB) and (ec is NB) then (kp is PB)(Ki is NB)(Kd is PS)(.5811)
 : : : : : : : : : :
49. if (e is PB) and (ec is PB) then (kp is NB)(Ki is PB)(Kd is PB)(.3121)

2. THE LQR INTEGRATED FUZZY CONTROL (LQRIFC)

The LQR integrated fuzzy control utilizes both advantages from the LQR controller and fuzzy logic controller as LQR controller can easily satisfies the flying qualities and pilot rating requirements and fuzzy control can cope with the nonlinearity of the system introducing a smart way to modify the output gains according to the actual performance blending the dynamic response that generating better performance than using LQR alone [6], [7]. Fig. 9 describes the LQR integrated fuzzy control (LQRIFC) for pitch controller.

The states and outputs of the plant plus the compensator are:

$$x = \begin{bmatrix} \alpha \\ q \\ \delta_e \\ \alpha_F \\ \varepsilon \end{bmatrix}, y = \begin{bmatrix} \alpha_F \\ q \\ \varepsilon \end{bmatrix}$$

System dynamics is described by the following state space through LQR design algorithm at several trimmed points for pitch control; Table 3 shows the linearized matrices of F-16 model at different operating points (different altitudes). The LQR closed loop system poles satisfy the flying qualities specifications [8] such as the damping ratio and the natural frequencies. The tuned fuzzy logic controller copes with the LQR output error thus enhancing the net output system performance. Membership functions could be sliding back and forth so that it suppress low frequencies disturbance on the other hand the low pass filter introduced will cope with high frequencies noise that produce a robust and optimal control system, as the integrated control system utilizes both advantages of the LQR and FLC techniques (optimal and robust design). Fig.10 shows the difference between both cases with the optimized fuzzy controller and without fuzzy controller as the additional FLC enhances the performance index (PI) more than 8.5%. Table 4 shows the different optimized LQRIFC gains at different altitude. The following linear equations describe the dynamics of the longitudinal motion of the aircraft at velocity 250 (ft/s).

$$x = Ax + Bu + Gr \tag{8}$$

$$y = Cx + Fr \tag{9}$$

$$z = Hx \tag{10}$$

$$A = \begin{bmatrix} \tilde{A}_{\alpha\alpha} & \tilde{A}_{\alpha q} & \tilde{B}_{\alpha\delta e} & 0 & 0 \\ \tilde{A}_{q\alpha} & \tilde{A}_{qq} & \tilde{B}_{q\delta e} & 0 & 0 \\ 0 & 0 & -20.2 & 0 & 0 \\ 10.0 & 0 & 0 & -10 & 0 \\ 0 & -57.2958 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 20.2 \\ 0 \\ 0 \end{bmatrix}, G = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, F = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 57.2958 & 0 \\ 0 & 57.2958 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, H = [0 \ 57.2958 \ 0 \ 0 \ 0], u = -Ky = -[K_\alpha \ K_q \ K_l]$$

3. APPLICATION METHODOLOGY

Applying the integrated control system of the previous technique on the F-16 nonlinear model during descending slope phase during the landing process with input of $\theta d = -2.5^\circ$, with applying DRYDEN vertical gust model, therefore verifying the output performance with applicable flying qualities [4] that results in satisfied performance for the fuzzy PID controller. Fig. 11 and Fig. 12 illustrate the whole system control effort and the fuzzy controller contribution output to the elevator actuator. Fig. 13 and Fig. 14 illustrate the output performances of the integrated control system with optimized fuzzy PID controller such as the altitude descending rate and the descending angle. It is clear now from these figures that the combination of both LQR and fuzzy logic produces stable and robust control system.

5. CONCLUSION

In this paper an optimization technique (GA) was presented to tune fuzzy logic controller parameters. A new MATLAB code was used to perform the GA operations for the proposed application. The optimized output parameters from the GA algorithm were used to tune the fuzzy logic controller in order to achieve robust and stable output performance of the aircraft control system (longitudinal motion). Moreover the proposed integration utilizes both advantages of LQR and fuzzy PID controller such as the verification of the flying qualities issues and introducing variable gains that can cope with the different uncertainty during the actual flight regimes according to the nature of flexible fuzzy controller output gains. Besides introducing such nonlinear fuzzy PID controller makes up some of the disadvantages of using linear controller (LQR) with aircraft nonlinear modeling such as pre-calculated constant gain matrix at each operating points.

REFERENCES

- [1] Zadeh, L. A, "Fuzzy Sets", *Information and Control*, Vol. 8, 1965, pp. 338-353.
- [2] Takagi, T. and Sugeno, M., "Fuzzy Identification of Systems and its Application to Modeling and Control", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 15, 1985.
- [3] Goldberg, D., *Genetic Algorithms in Search, Optimization & Machine Learning*, Addison Wesley Longman, Inc., 1989.
- [4] Teo, L.S., Bin Khalid, M., and Yusof, R., "Tuning of a Neuro-Fuzzy Controller by Genetic Algorithm", *IEEE Transactions on Systems, Man and Cybernetics*, Part B, Vol. 29, Issue 2, 1999, pp. 226 -236
- [5] Mamdani, E.H. and Assilian, A., "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller", *International Journal of Man Machine Studies*, Vol. 7, No. 1, 1975, pp. 1-13.
- [6] Stevens, B. L., Lewis, F.L., & Al-Sunni, F. (1991). Aircraft flight controls design using output feedback. *Journal of Guidance, Control, and Dynamics*
- [7] Stevens, B. L., Lewis, F.L. (1992). *Aircraft Control And Simulation*. New York: John Wiley & Sons, Inc.
- [8] Flying Qualities of Piloted Airplanes, MIL-F-8785C, 5 Nov. 1980.

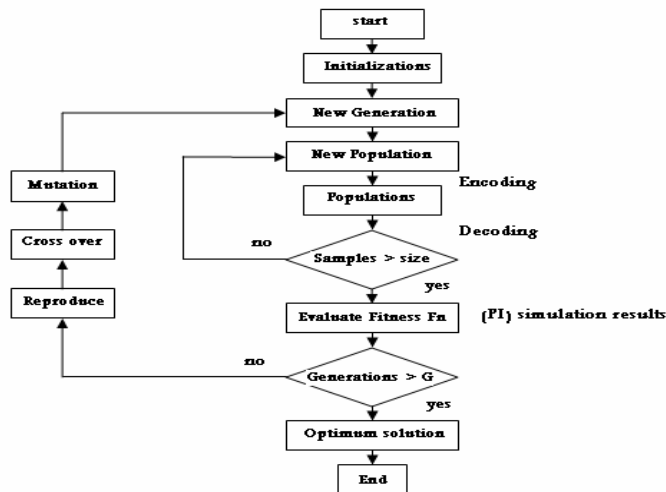


Fig. 1. The flow of GA functions and process

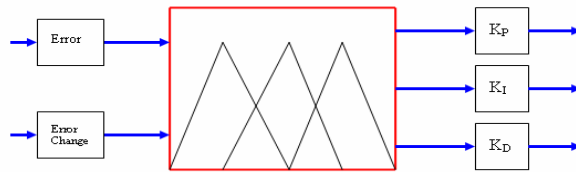


Fig. 2. Nonlinear fuzzy PID controller

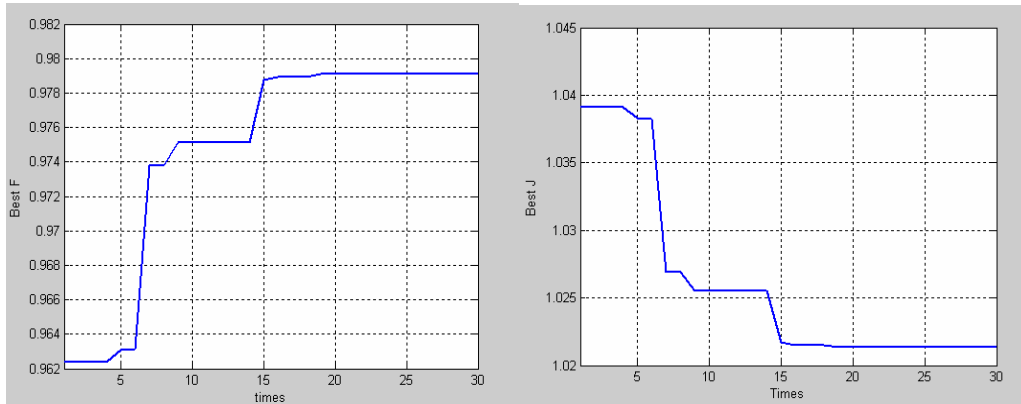


Fig.3. Objective function J and Fitness function F

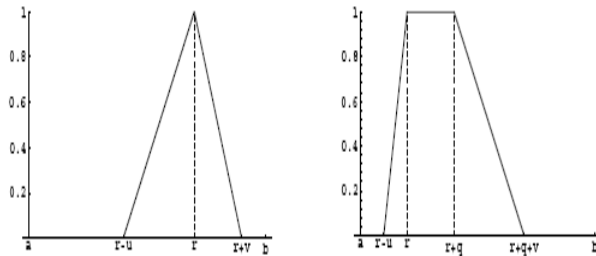


Fig. 4. Triangular and trapezoidal membership functions

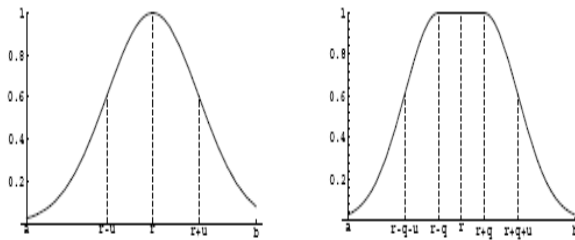


Fig. 5. Bell-shaped membership functions

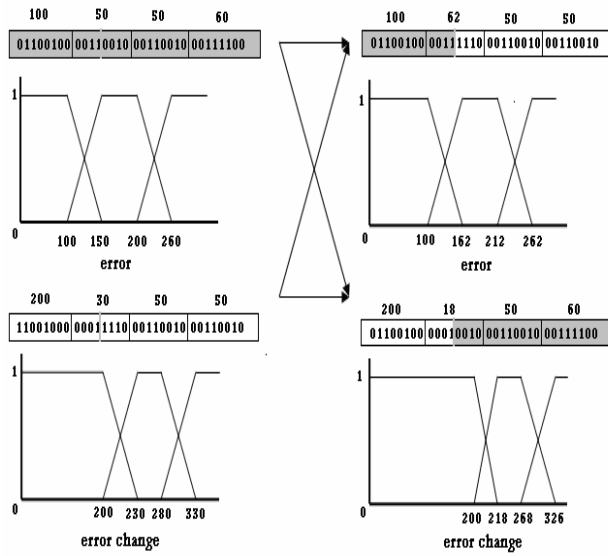


Fig.6. Example for one-point crossover of fuzzy partitions

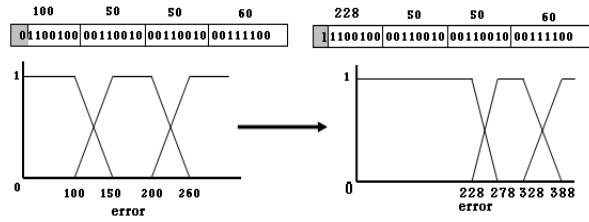


Fig. 7. Mutating a fuzzy partitions

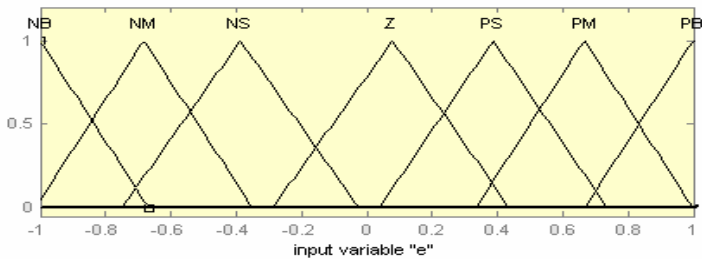


Fig. 8a Input variable "error"

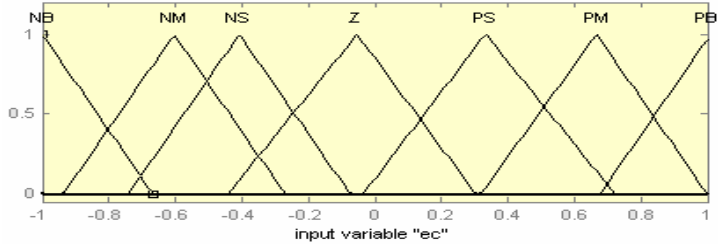


Fig. 8b Input variable "error change"

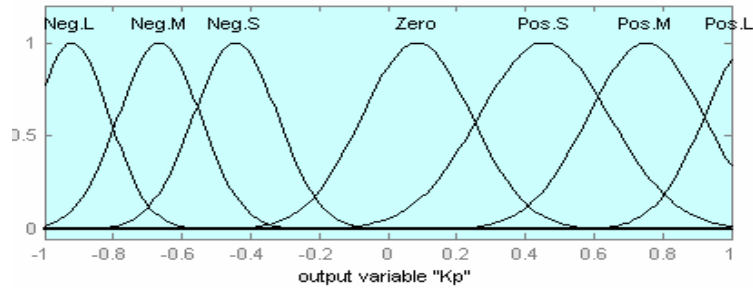


Fig. 8c Output variable "Kp"

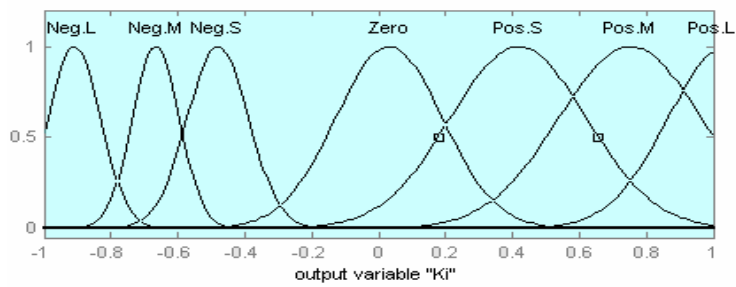


Fig. 8d Output variable "Ki"

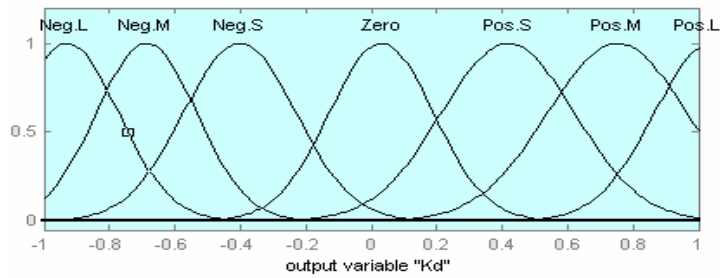


Fig. 8e Output variable "Kd"

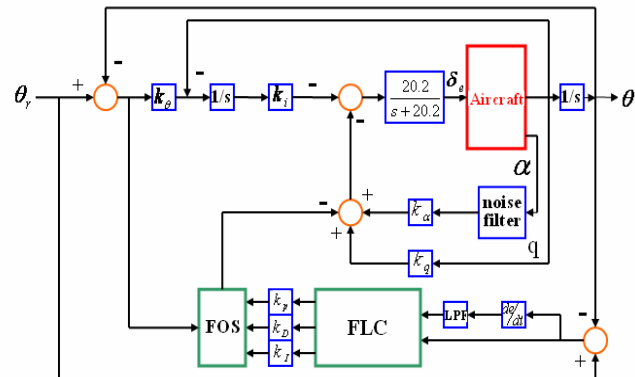


Fig. 9. LQR integrated fuzzy control (LQRIFC) for pitch controller

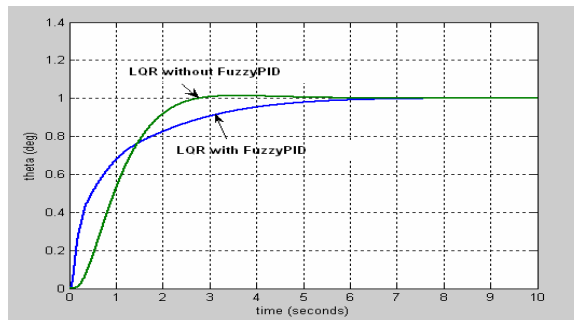


Fig. 10. Integrated system step response in both cases with optimized fuzzy controller and without fuzzy controller

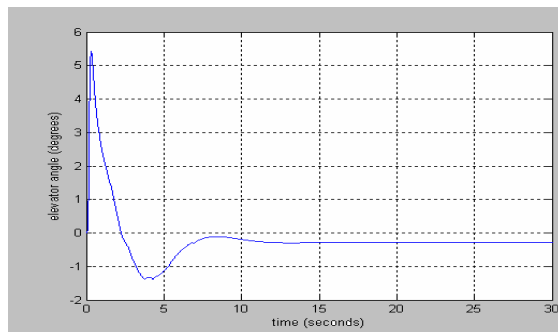


Fig. 11. Total elevator angle (deg)

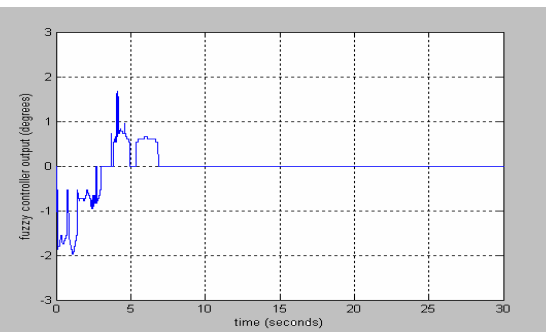


Fig.12. Fuzzy controller contribution output (deg)

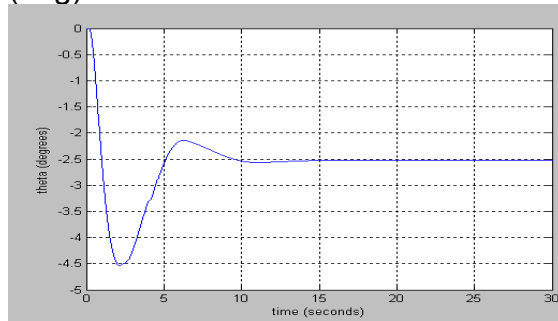


Fig. 13. Descending angel (deg)

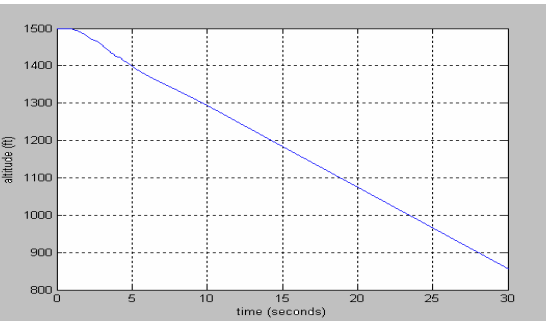


Fig. 14. Descending altitude (ft)

Table 1 Gain Limits

Denormalized Gain	Minimum limit	Maximum limit
K_P	1	10
K_D	-1	-2.5
K_I	.05	1.8

Table 2 Decoding Equation

Denormalized gains	Decoding equation
K_P	$9 \times \frac{y_1}{1023} + 1$

$$\begin{aligned}
 K_D & & -2.4 \times \frac{y_2}{1023} - .1 \\
 K_I & & 1.75 \times \frac{y_3}{1023} + .05
 \end{aligned}$$

Table 3 Linearized Matrices

Alt (ft)	$\tilde{A}_{\alpha\alpha}$	$\tilde{A}_{\alpha q}$	$\tilde{B}_{\alpha\delta\epsilon}$	$\tilde{A}_{q\alpha}$	\tilde{A}_{qq}	$\tilde{B}_{q\delta\epsilon}$
1000	-0.8972	0.9502	-0.0019	-3.6241	-1.2147	-0.2650
1500	-0.8325	0.9533	-0.0017	-3.3307	-1.1265	-0.2438
2000	-0.7716	0.9562	-0.0017	-3.0570	-1.0433	-0.2235

Table 4 Gain values

Alt (ft)	1000	1500	2000
K_α	0.2086	0.2104	0.2127
K_q	-0.7971	-0.8157	-0.9091
K_i	4.6106	4.5618	5.0274
K_p	7.190	8.8725	8.1053
K_D	2.958	2.754	2.333
K_I	-2.870	-2.650	-2.230