IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 10, Issue 4 Ver. I (July – Aug. 2015), PP 29-41
www.iosrjournals.org

A Servo Motor Drives Based On Hybrid Genetic Algorithm Fuzzy Neural Controller

Murad O. Abed Helo
Electrical Engineering Department, University Of Babylon, Babylon, Iraq

***Abstract**: The nonlinearity and time varying characteristics of a servo Motor (SM) make it very difficult to be controlled. Althoughproportional-integral-derivative (PID) controller are widely used in this field but the complex mathematical model of (SM) makes the design procedure of any PID controller very tedious ,in which the time varying behavior of (SM) reduces the accuracy of any PID controller used. The use of Fuzzy Logic Controllers (FLC) in such control problem is widely used too, since Fuzzy Logic does not need any mathematical model and only uses linguistic rules that are based on human expert. However, still checking the parameters of fuzzy logic neural network controller (FLNNC) is a hard task for such a system specially the center and width of the used member ship functions. In this paper a Hybrid Genetic Based (FLNNC) is introduce to control the (SM). The Parameters measurements of (SM) has been implemented based on Genetic Algorithm (GA) tuned (FLNN) with a technique in which only transient speed measurementis required for identification of the parameters (SM).The GAFLNN controller was simulated by MATLAB Simulink using the technique of PID type based onFLNN technique and the scaling gains, fuzzy logic rules, membership function, and coefficients of neural networkare optimized by genetic algorithm technique. Simulated results show a significant improvement in settling time and rising time also reduces overshoot, IAE and ISE and with variations of external load disturbance.*

***Keywords**: Fuzzy Logic,Genetic Algorithm, Neural Networks, PID, Servo motor.*

I. Introduction

The servo motor drives are extensively used in industry all over the world. The outstanding advantages of dc drives, such as ease of control, precise and continuous control of speed over a wide range, and speed of response, will ensure their popularity for more applications. However, dc drives have been used for a long time [1].The development of the Ward-Leonard system, which was introduced in the 1890s, was a significant step in the evolution of dc drives. The system uses a motor-generator (M-G) set to power the dc drive motor. In the late 1940s and early 1950s electronic control brought about a significant improvement in dc drives. In the early stage, industrial-type gas-filled rectifier and controlled rectifier (thyratron) tubes were used in exciters and regulators for the M-G set. This system is of improved response and better accuracy [1]. Later on, these tubes became available in high current capacity and were used in rectifier circuits to convert ac to dc for speed control of the dc motor. In late 1950s solid-state devices, silicon diodes, and silicon controlled rectifiers [2] became available in the market at economic prices. The advent of solid-state devices represents a significant step towards development of dc drives. Thyristorized dc-to-dc and ac-to-dc converters were highly used in dc drives [2].After the advent of microprocessors in the 1970s, dc drives, and control system in general, took other features. Programming a microprocessor to control a servo motor can highly reduce the complexity, and improve rising, and settling times [3,4]. However, many applications of a dc motor require adaptively due to the fact that the inertial load may be variable. So, many schemes utilized microprocessors to do the job of controlling the dc motor.The development of fuzzy logic and neural networks solved many problems of controlling plants even when the exact mathematical model is unknown [5-8]. Non-linear plants have been widely controlled by fuzzy neural controllers [9-11]. Fuzzy, neural, and fuzzy-neural controllers have been widely used to control speed and position of dc motors, was proven have the whole system becomes robust against parameters and load variations [12-13]. Moreover, the fuzzy logic controller had been used to tune the constants of a PID controller for any plant [14].The concept of fuzzy, neural, and fuzzy-neural has been widely utilized to construct adaptive controllers and estimators [15-17]. However, the concept of adaptive control has been, also, combined with fuzzy-neural networks to control the speed of a servo motor [18]. The objective of the paper is to build a fuzzy-neural controller-based drive for an armature controlled dc motor. The overall system would be tested with step loads and reference inputs to evaluate its robustness. A comparative study with fuzzy logic controller-based dc motor drives is also to be considered.

II. Literature Survey

Hybrid systems combining fuzzy logic with neural networks are providing their effectiveness in wide variety of applications [11,12]. Every intelligent technique has particular combinational properties, (e.g. ability to learn, explanation of decisions) that make them suited for particular problems but not for others. For example, while neural networks are good at recognizing patterns, they are not good at explaining how they reach their decisions [13,14]. Fuzzy systems, which can reason with imprecise information, are good at explaining their decisions, but they cannot automatically acquire the rules they used to make those decisions [15]. These limitations are the central driving forces behind the creation of intelligent hybrid systems where the two techniques, say fuzzy logic and neural networks, are combined in a manner that overcomes the limitations of individual techniques. Hybrid systems are also important when considering the varied nature of application domains [16]. Many complex domains have many different component problems, each of which may require different types of processing. If there is a complex application which has two distinct sub-problems, say signal processing task and serial reasoning task, then a neural network and a fuzzy system respectively can be used for solving these separate tasks [17-19]. The use of intelligent hybrid systems is growing rapidly with successful applications in many areas including process control, engineering design, financial trading, credit evaluation, medical diagnosis and cognitive simulation [12].

While fuzzy logic provides an inference mechanism under cognitive uncertainty, computational neural networks offer exciting advantages, such as learning, adaptation, fault-tolerance, parallelism and generalization. To enable a system to deal with cognitive uncertainties in a manner more like human, one may incorporate the concept of fuzzy logic into the neural networks [6].

Fuzzy neural systems witnessed many progresses that made them applicable more efficiently to many applications such as automatic control. An immediate synergy can be found between fuzzy and neural control. The former exploits an important feature of fuzzy systems, which is capable of building the rule base by acquiring the knowledge from human experts (human-friendly approach). On the other hand, neural networks are trained with a suitable set of data samples (computer-friendly approach), although without taking advantage from available human knowledge.Neural networks and fuzzy systems have been unified using the weighted radial basis functions paradigm [20], by means of which fuzzy rules and neurons can immediately be mapped onto each other, and then trained or optimized in the same way as traditional neural networks, using gradient descent methods. This can lead to a noticeable increase in performance and ease of development, since it allows optimizing, with gradient descent methods, a network previously initialized with an approximate solution provided by the human expert as a set of fuzzy rules [14].Recently, neuro-fuzzy processors have been built that are specialized for programming a neuro-fuzzy system, which would simplify realization of neuro-fuzzy systems. A control system have been developed, called Digital Analog Neuro-fuzzy Interface for Enhanced Learning Applications (DANIELA), using a type of neuro-fuzzy processors called (AMINAH) which interacts with a general purpose 68HC11microcontroller and a memory in order to control the plant [14]

The genetic algorithm [16, 20] uses the principles of natural selection and genetics from natural biological systems, in a computer algorithm, to simulate evolution. Essentially the GA is an optimization technique that performs a parallel, stochastic, but direct search that evaluates more than one area of the search space and can discover more than one solution to a problem. A “fitness function” measures the fitness of an individual (possible solution) to survive in a population of individuals.

III. Design of PID Genetic Algorithm Fuzzy Neural Controller Controllers

Fuzzy systems are human-friendly systems, and neural networks are computer-friendly systems [21]. So, fuzzy systems and neural networks can be integrated to get hybrid systems that are both human and computer friendly systems. Another advantage of integrating neural networks with fuzzy systems is to make the fuzzy inference parameters, i.e. membership functions parameters, adjustable, to produce an optimal output [6, 21]. All types of fuzzy inference mechanisms can be integrated with neural networks, and with many approaches.

Fuzzy logic and neural networks can be combined in variety ways. Fuzzy neural networks can be classified into several categories [6]. The classification of a particular fuzzy neural network into one of these five categories is not always easy, and there may be different viewpoints for classifying fuzzy neural networks. The fuzzy neural networks in the first category are fuzzy rule-based systems where fuzzy if-then rules are adjusted by iterative learning algorithms similar to that of neural networks, e.g. the back propagation algorithm. In the second category, a neural network represents fuzzy rule-based systems. Thus, learning algorithms, such as back propagation algorithm, can be applied to the learning of fuzzy rule-based systems. Fuzzy neural networks, in the third category, are neural networks for fuzzy reasoning. Standard feed forward neural networks with special processing procedures are used for fuzzy reasoning. The fourth category of fuzzy neural networks consists of fuzzified neural networks. Using fuzzy numbers as inputs, targets, and connection weights can fuzzify standard feed forward neural network. This category is distinguished from the other categories because it is based on the fuzzy arithmetic, whereas other fuzzy neural networks use real numbers for calculating their outputs.

The structure of SM control system is shown in Fig. (1). The FNNC is fuzzy-neural networks controller [16] which consists of four layers NN and fuzzy logic all are tuned by Genetic algorithms as shown in Fig. (2).e, $e_r$is the input and u* is the output. The fuzzy individual inputs have seven linguistic variables and referred by PS, PM, PB, ZE, NB, NM, NS, respectively. The Triangular type is corresponding membership function of all fuzzy sets and was designed as perfect, uniform, and standard.

The block diagram of the controller with the proportional-integral-derivativebased on Genetic algorithms tuned the Fuzzy Logic Neural network Controller is shown in fig. (2).

The Genetic algorithms (GA) is not only capable of optimized of (SM) system; moreover it is capable of adjusting the membership functions parameters and constants of rules to give an optimized output for a given input. It is important to mention that adjusting the control parameters for a certain controller is normally done with plants of unknown models, i.e. the general input-output mapping is unknown.

Suppose that SM system has n-inputs and one output, and suppose that the training pair to train the GAFLNN:

$y^k = f\left(X^k\right)$Or: $y^k = f\left(x_1^k, x_2^k, \ldots x_n^k\right)$

Where k=1,2…K , and y^k:is the desired output of the k^{th} training pair, X^k is the k^{th} training input vector. K is the number of total training pairs.The approach, that is used to train the ANFIS, is adjusting the parameters to minimize the mean squared error E^k. $E^k = \frac{1}{2}\left(y^k - o^k\right)^2$

WhereO^k is the actual output at the k^{th} training input vector.

The gradient steepest descent method is the approach that can be used to adjust the whole parameters of the system. Suppose that bell-shaped functions are utilized as membership functions, then the control parameters that are to be adjusted: c_{ij} (centers of membership functions).σ_{ij}(width of membership functions).Z_p (rules outputs).

Where (i=1,2…n);(j=1,2…m);(p=1,2…P)

Note that n is the number of linguistic terms assigned for each linguistic variable, m is the number of linguistic variables, and P is the number of rules.

So, the control parameters above can be adjusted using steepest descent method as below:

$$c_{ij}(t+1) = c_{ij}(t) - \frac{\partial E^k\left(c_{ij}(t), \sigma_{ij}(t), z_p(t)\right)}{\partial c_{ij}(t)} * \eta_1$$ and $$\sigma_{ij}(t+1) = \sigma_{ij}(t) - \frac{\partial E^k\left(c_{ij}(t), \sigma_{ij}(t), z_p(t)\right)}{\partial \sigma_{ij}(t)} * \eta_2$$

And $$z_p(t+1) = z_p(t) - \frac{\partial E^k\left(c_{ij}(t), \sigma_{ij}(t), z_p(t)\right)}{\partial z_p(t)} * \eta_3$$

Where $\eta_{1,2,3}$ are the training rates, t is the iteration number.

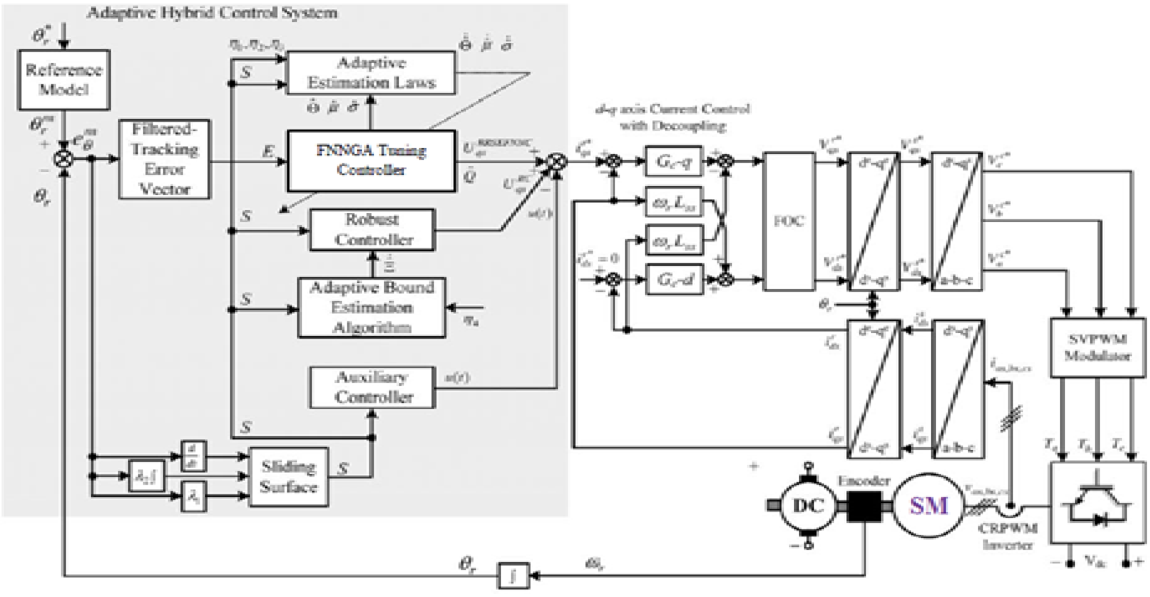


Figure (1) PID GA Fuzzy Neural Network Controller of servo Motor

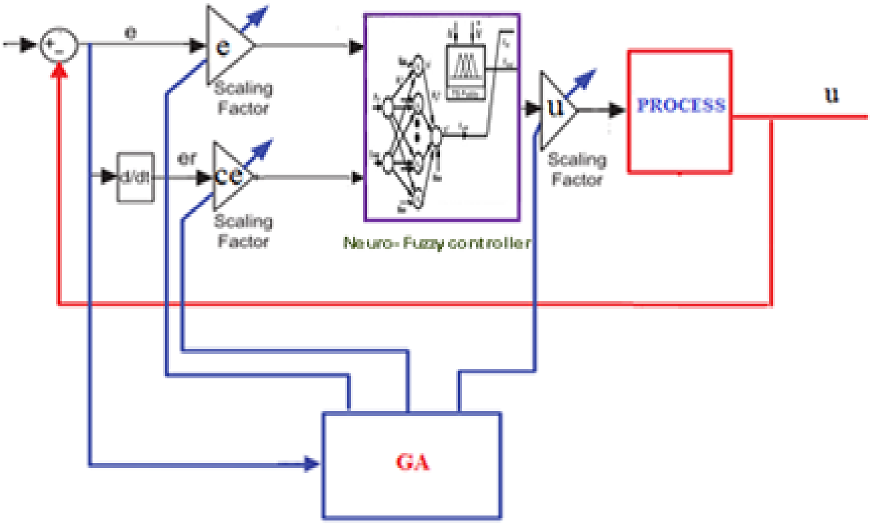


Figure (2) GAFNNC controller structure

The following steps summarize how the PID Genetic algorithms based Fuzzy Logic Neural network controller work:

1. Initiate the parameters of input membership functions, say ci and σj.

Initiate the parameters rules outputs.

Initiate the learning factors (0<η1, 2,3<1).

Initiate the tolerance.

2. However the stop status is false repeat steps 3-11:
3. Calculates the normalized speed error at any time instant kth:

$$e(k)=(\omega_r(k)-\omega(k))/\omega_{base}$$

ω_r(k) is the speed reference and ω(k) is the estimated speed and ω_{base} is the base speed.

4. Calculates the normalized change in speed error by: $\Delta e(k)=(e(k+1)-e(k))$
5. For the two inputs e(k) and Δe(k), do steps 6-10:
6. Compute the output of FLNN; say Δu(k), which corresponds to the inputs e(k) and Δe(k).
7. Adjust by GA the parameters of membership functions and rules outputs such that the cost function E is minimized by optimizing the following equations

$$c_i(t+1)=c_i(t)-\frac{\partial E^k(\sigma_j(t),c_i(t),z_p(t))}{\partial c_i(t)}*\eta_1,\ \sigma_j(t+1)=\sigma_j(t)-\frac{\partial E^k(\sigma_j(t),c_i(t),z_p(t))}{\partial \sigma_j(t)}*\eta_2$$

and $z_p(t+1)=z_p(t)-\frac{\partial E^k(\sigma_j(t),c_i(t),z_p(t))}{\partial z_p(t)}*\eta_3$

8. Check the status of stop:

If the change in parametersin step 7is greater than allowed tolerance and then repeat step 7

Else, go on.

9. After training FLNN, Calculate the output, i.e. Δu(k).
10. Calculate the output of overall system by: $u(k+1)=u(k)+\Delta u(k)$
11. Check the Final status of stop:

If the termination of program is true; then stop.

If False go to step 2.

When designing a Fuzzy Logic Neural networks Controller based on Genetic Algorithms [17], the following issues are to be addressed for representing the optimization by GA:

3.1 Representation

The strategy of representation is how to determine the chromosome code from the variables. The rules used in this paper represent the three sections: rule selection, representation for the input variables and the representation for the output variables. The rule selection bit may be zero or one. One represents the selection of the rule. Depending on the ranges of the input variables and output variable, number of bits has been chosen for representing each rule of the rule set. The input variables of the SM process are error and the rate of change of error and output variable is base flow rate are consider for fuzzy variables. Seven membership functions are allotted for each input and output variables. Input variables are represented by IP1 and IP2 and output variable are represented by OP.

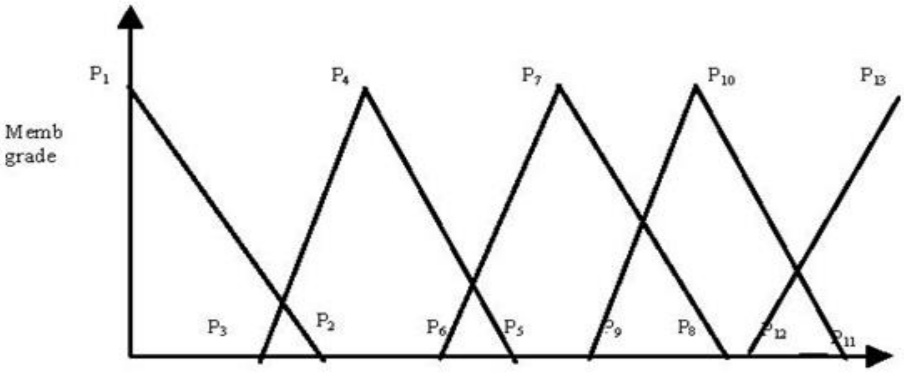


Fig. 3. Fuzzy Space.

Triangular membership function is used in this paper. Each membership function is represented by seven membership points with overlap between each membership function as shown in fig.(3). A total of 13 membership points (P1 to P13) are required for representing each input variable as a fuzzy set. In those thirteen points, first and last points (P1 and P13) are fixed. The remaining eleven membership points are evolved between the dynamic ranges such that P2 has [P3, P13], P3 has [P1, P13], P4 has [P2, P3], P5 has [P6, P10], P6 has [P4, P7], P7 has [P5, P13], P8 has [P9, P13], P9 has [P5, P10], P10 has [P7 P13] ,P11 has [P12 P13] and P12 has [P8 P13]. With the above representation a typical chromosome will look like the following:

$$\underset{IP_1}{\underline{100}} \qquad \underset{IP_2}{\underline{001}} \qquad \underset{OP}{\underline{011}}$$

$$\underbrace{\underset{P_1}{\underline{110}}\ \ \underset{P_2}{\underline{010}}}_{MF_1} \quad \underbrace{\underset{P_3}{\underline{011}}\ \ \underset{P_4}{\underline{111}}\ \ \underset{P_5}{\underline{100}}}_{MF_2} \quad \underbrace{\underset{P_6}{\underline{011}}\ \ \underset{P_7}{\underline{111}}}_{MF_3} \quad \underbrace{\underset{P_8}{\underline{100}}\ \ \underset{P_9}{\underline{011}}\ \ \underset{P_{10}}{\underline{111}}}_{MF_4} \quad \underbrace{\underset{P_{11}}{\underline{100}}\ \ \underset{P_{12}}{\underline{100}}\ \ \underset{P_{13}}{\underline{001}}}_{MF_5}$$

3.2 Fitness Function

The important observance is the choice of fitness function. Evaluation of the individual is accomplished by calculating the objective function value for the problem using the parameter set. The result of the objective function calculation is used to calculate the fitness function of the individuals. The Integral Square Error, settling time and over shoot are taken as performance indices and the objective function is given by minimize

$$f = f_{ISE} + fos + fst \qquad (1)$$

The minimization objective function given by (12) is transformed to fitness function as

$$\text{Fitness}=\frac{k}{1+f} \qquad (2)$$

Where k is a constant. In the denominator a value of 1 is added with f in order to avoid division by zero.

IV. The Simulation Results Of Proposed Driver

The fuzzy membership functions as linguistic values are shown in table (1), the input $\Delta e(k)$ are represented by the linguistic values in the left-most column, and the input $e(k)$ are represented by the linguistic values found in the top row. The linguistic values representing the consequent for each of the rules can be found at the intersections of the row and column of the appropriate places. Fig .4 shows the Simulink model of the proposed servo motor base on GAFLNN controller.

The Results of the PID controller base on GAFLNN described above is implemented for the speed control strategy of a separately excited armature controlled of SM. Response and armature current for step reference input and step load would be carefully measured to make a comparative study genetic algorithm base fuzzy neural controllers.

Table (1) The complete Fuzzy rules setfor input and output

e Δe	NS	NM	NB	ZE	PS	PM	PB
NS	NB	NB	NM	NM	NS	NS	PM
NM	NM	NB	NS	NS	ZE	NS	PM
NB	NM	NB	NS	NS	PS	ZE	PB
ZE	NS	NB	NS	ZE	PS	PS	PB
PS	NS	NB	ZE	PS	PM	PS	PB
PM	ZE	NM	PS	PS	PM	PS	PB
PB	PS	NM	PS	PM	PB	PM	PB

The Results of the PID controller base on GAFLNN described above is implemented for the speed control strategy of a separately excited armature controlled of SM. Response and armature current for step reference input and step load would be carefully measured to make a comparative study genetic algorithm base fuzzy neural controllers.The simulation results ofstep reference input under no-load condition can be obtained by setting the load torque (T_L= zero). Fig.s5 and 6 show the system response and armature current respectively. It is clear that the rising (tr) and the settling (ts) times are: tr=430 msec. ts=0.75 sec. Note that the settling time was obtained for 5% criterion.

It is obvious that the system response is acceptable for some applications and not for others due to large rising and settling times (sees fig.5). On the other hand, the armature current has it highest value at step instant, because of the moment of inertia for the rotating parts of the motor, which would not permit the motor shaft to rise to the required final value at once. Consequently, the back emfE_b would not rise to its final value at once.

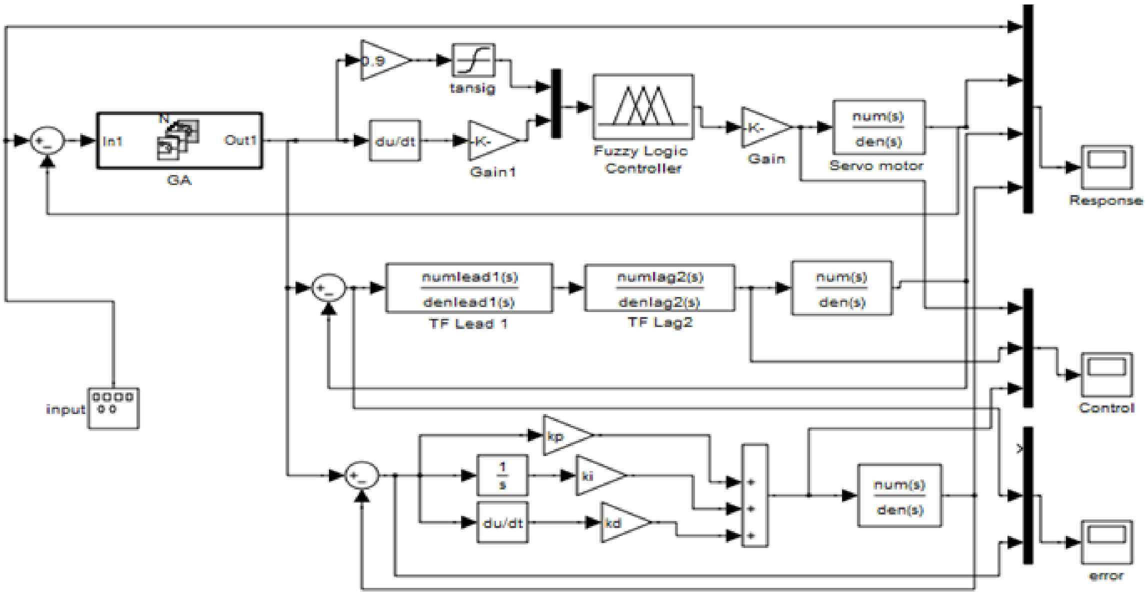


Fig.(4) Simulink model of the proposed servo motor base on GAFLNN controller

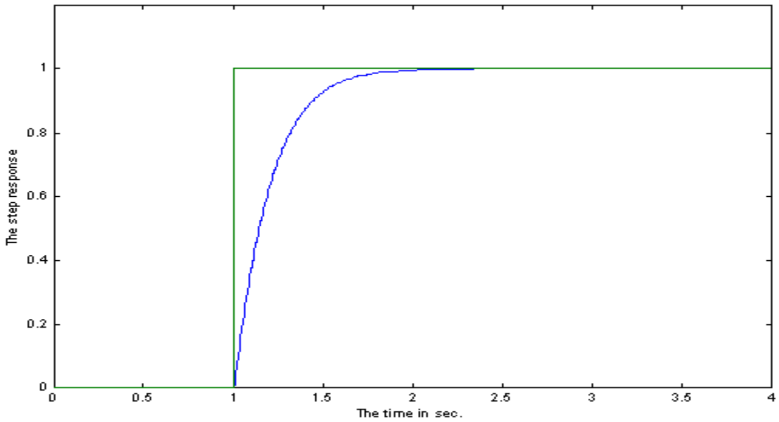


Fig.5 The no-load system step response with GAFLNN controller.

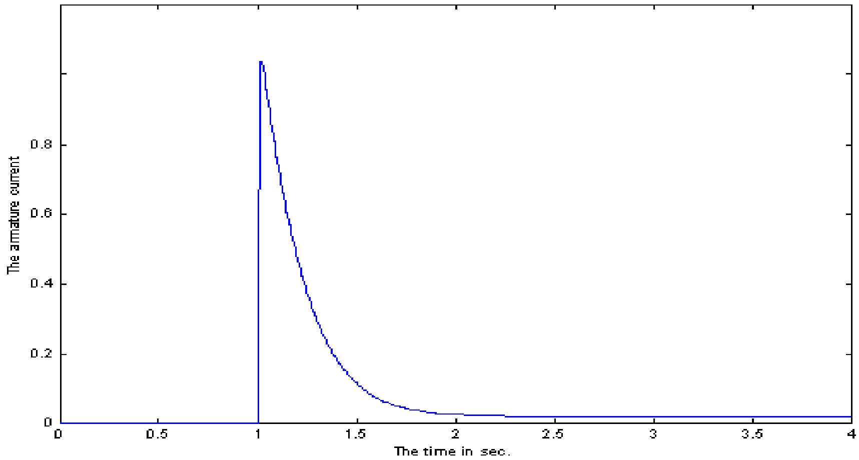


Fig.6 The no-load armature current with GAFLNN controller (A).

Fig.(7 a) shown the SM response under PID based GAFLNNC and fig. (8a) shown the voltage of control action of the GAFLNNC that applied to the SMamplifier stage at sampling time250msec. A step disturbance on the controller output of magnitude (6) was added, so the error is suddenly increases to make a high peak overshoot, as shown in fig.s (7b) and (8b). From these figures, it is clear thatGAFLNNC is Able to deal with this disturbance.In conventional PID controller,the design problem may be includes a proper choice of coefficients of PID controller. But in the GAFLNN controller,the design needed many parameters like number of rules, membership functions,scalar factors fuzzification and defuzzification procedures, numbers of hidden layer of neural network and parameters of genetic algorithm. These different parameters make the GAFLNN controller more robust but much difficult for design.

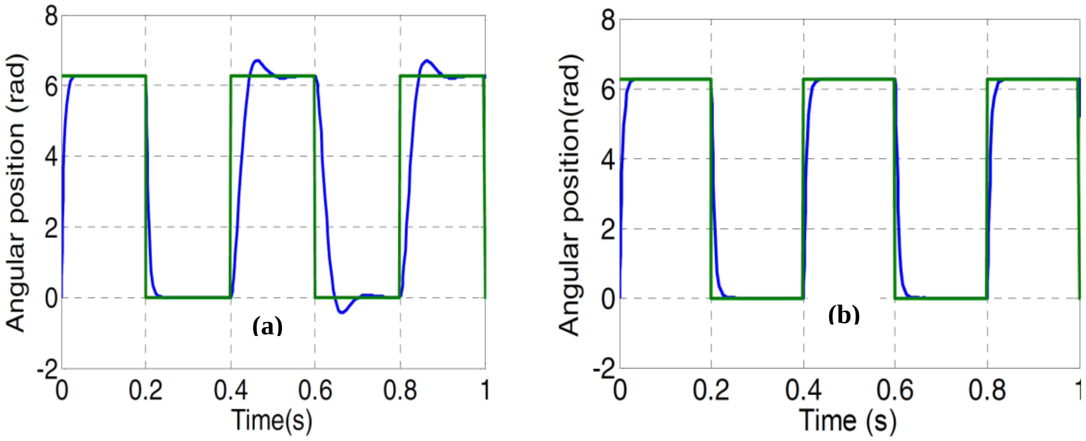


Fig. (7) The output of the SM under GAFLNNC a) without disturbance b) with disturbance

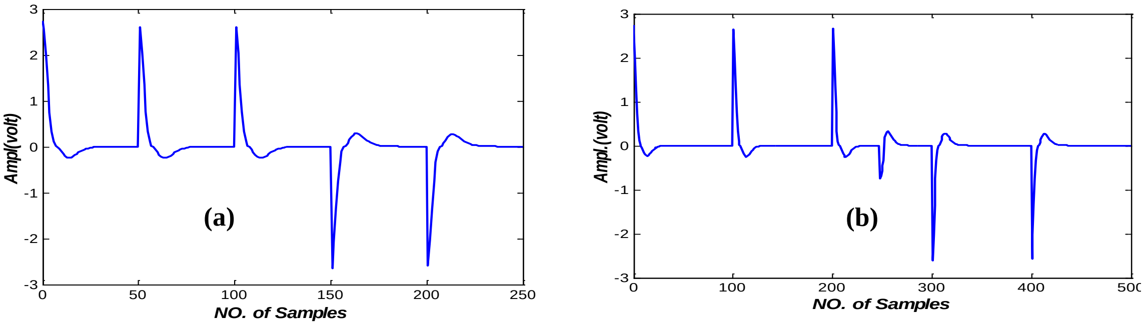


Fig. (8) The Control action of the GAFLNNC a) without disturbance b) with disturbance

The Strong gradient descent is applied as the optimal way to produce optimal output. Settling and Rising times, and peak over shoot are greatly reduced, that lead to follow good behavior. These results had been acquired with various loads that would be more powerful utility by usegenetic algorithms based on fuzzy neural controller for unknowncharacteristics of the load. Due to these high performance of proposed controller, the error signal e(t) of the output of the controller is significantly reducedwhich would result in larger time to get to the final value of the reference input step and sine wave as shown in fig. 9.

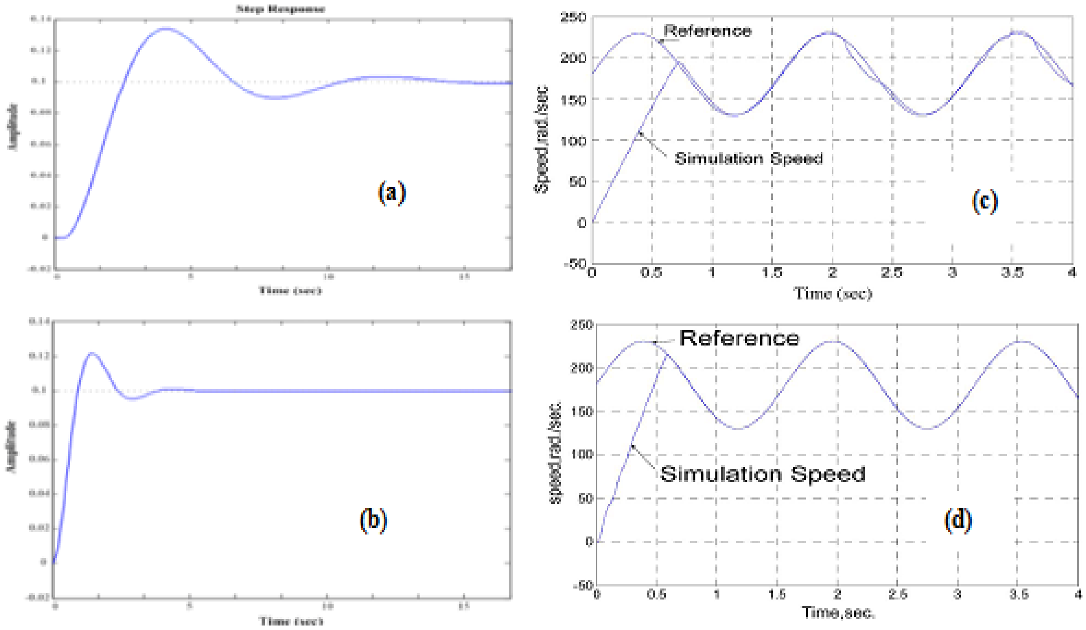
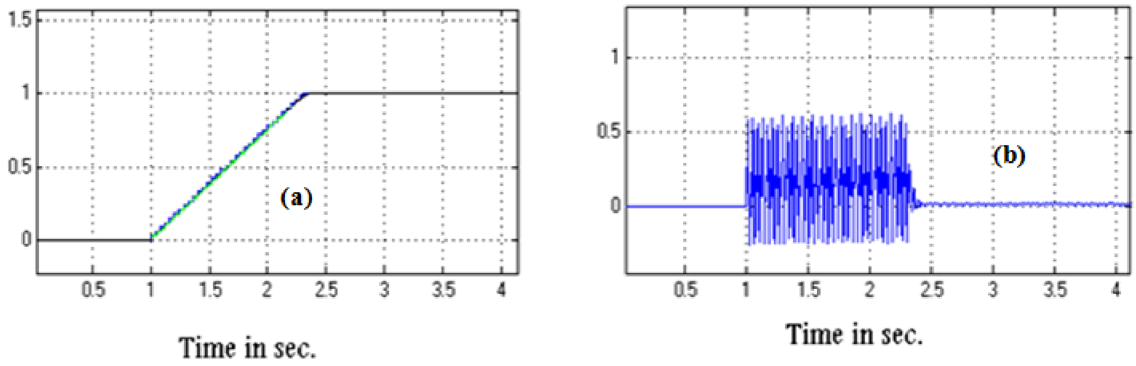


Fig. (9) Simulation results of SM with: a. conventional PID controller to step response. b. GAFLNNC to step response .c. conventional PID controller response to sine input. d. GAFLNNC response to sine input.

After stepping, the input voltage V(t) will be directly stepped, but the speed would not step directly due to the moment of inertia and the viscous of friction of the rotating part of the motor. So, the speed would increase gradually until it reaches the required value. The back emf would also increases gradually in accordance with the speed. As the value of E_b is zero at starting, the armature current, at starting, is maximum. To decrease the armature starting current, the input signal is increased in small steps until it reaches the required value. The SM response, motor torque, armature current, and error signalare shown in fig.10

The soft starting can protect the motor against harmful high starting current. On the other hand, the time required for the response to reach the required value increases (see fig.10.a). As a matter of fact, soft starting is normally utilized when the reference input is increasing or decreasing in relative high steps. As the step size is smaller, the step current becomes smaller which leads to safer operation of the motor.

The stepping size has been increased to 0.08 to approve the high performance of the controller for another stepping size. The system response, normalized system error signal, armature current, and motor torque are shown in fig.11 for stepping size of 0.08. It is obvious that the armature starting current had been raised to 1 A, due to increment in step size.



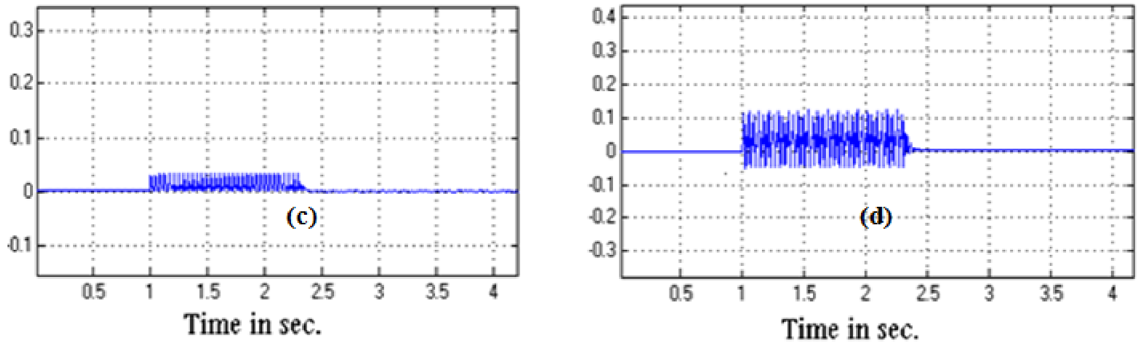


Fig.10 The simulation results for the case of soft starting (0.04 stepping size): a. step response. b. armature current (A). c.normalized error signal of SM . d.The torque (N.m).

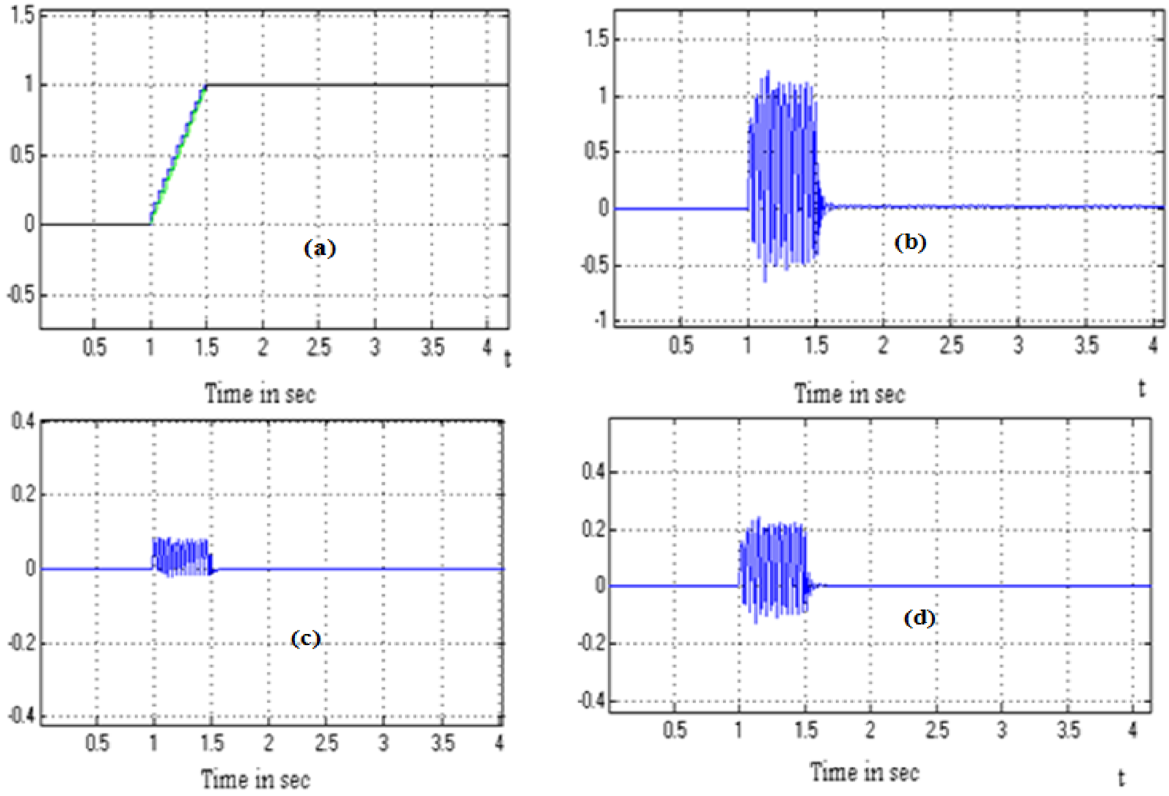


Fig.11 The simulation results for the case of soft starting (0.08 stepping size): a. step response. b. armature current (A). c. normalized error signal of SM. d. torque (N.m).

4.1 Results under Step Load Condition

When a load T_L is applied, the speed would be decreased at the instant of load application. The speed controller would take action and stabilizes the speed to the required value, i.e. to the speed before the load application. Consequently the steady state error is also zero due to the utilization of a PID fuzzy neural based genetic algorithm controller. In many applications, the load is varying which would require a powerful adaptive controller to perform the task of speed stabilization properly. As a matter of fact, load application, normally, take the greater attention due to the nature of motor use, that it is used to rotate a mechanical system that may has a variable inertia. The system response, error signal, armature current, and torque would be investigated. Attention would be paid to parts in which load is applied.

As mentioned previously, SM drives are normally designed for having variable dynamics caused by parameters variations such as inertia changes. Variations in moment of inertia, for the driven load, affect the dynamics of the system. In this section, step load insertion is studied in four Experiments. The load step instant is 2 second, and final value ranges from 0.25 to 1 N.m in step of 0.25.

Experiment 1

A step load of step instant 2-second and final value of 0.25 N.m is applied for the system under study. The SM response, armature current, error signal, and torque for Experiment1 are shown in fig.12. The external load excursion causes a disturbance in the system response, which causes the error signal, at instant of load application, to be increased (see figures 12.a and 12.c).As the error signal e(t) increases at instant of external load excursion, the PID based on GAFLNN controller forces the output to return to its value before the load is inserted. Consequently, the steady state error is forced to be zero. The motor torque is increased after the external load is applied; consequently the armature current is also increased due to linearity relation between motor torque and armature current.

Experiment 2

A step load of step instant 2-second and final value of 0.5 N.m is applied for the system under study. The SM response, armature current, error signal, and torque for Experiment1 are shown in fig.13. The step response is disturbed more than the previous Experiment, so the error signal increases more at time of external load application. Similarly, the PID based on GAFLNN controller stabilizes the speed to the desired value (see figure 13.a and 13.c). The armature current and the motor torque are also increased more than the increment in Experiment 2 after load is inserted.

Experiment 3

The load to be applied in this Experiment is of 2 second of step instant, and 0.75 N.m of final value. The system response, armature current, error signal, and torque for Experiment 3 are shown in fig.14. It is clear that the system response is disturbed more than the previous two Experiments. Consequently, the system error signal is also increased more than the previous Experiments at instant of external load excursion (see figure 14.a and 14.c). The armature current and torque are also increased more than previous two Experiments (see figure 14.b and 14.d).

Experiment 4

In this Experiment, the load that is to be applied has the same step time as that of the three previous Experiments, but with a final value of 1N.m. The SM response, armature current, error signal, and torque for Experiment 4 are shown in fig.14. It is obvious that the system response is also disturbed more than the Experiments before, which make the error signal increases more at instant of external load excursion (see figure 15.a and 15.c). The armature current and torque are also increased more than the three Experiments before (see figure 15.b and 15.d).

The soft starting is implemented by the reduction of step size to 0.04. This would result smaller armature current at step times. But as mentioned previously, the time required getting to the required fixed value, i.e. the steady state is increased. From figure 11.a, it is clear that the time required for getting to the final value is 1.36 sec. So, soft starting is utilized if the time required in reaching the final value is not important, i.e. the system is operated in the region when the response reaches to the required final value. There are some applications that don't require small time to reach to the final value, and only the system external load application is important. So, soft starting would be safer for such applications. Other applications, of dc motors require small time to reach to the final value, so that the soft starting described above is not preferable. So, motors with high withstanding of armature current are required for such applications.

It is apparent that the fuzzy neural controller has the great ability of controlling whatever the plant is, due to the power ability of non-linear mapping of fuzzy logic controllers. But, the parameters of fuzzy logic controllers, such as membership functions parameters and consequent parameters of rules restrict the adaptation of the controller, especially for Experiments of varying dynamic behavior. So, a neural network had been integrated with fuzzy logic to produce an adaptive hybrid fuzzy neural controller in which fuzzy logic parameters are on-line adjusted to produce an optimal output. Another advantage of fuzzy neural controller is the integration of a computer friendly approach, say neural networks and a human friendly approach, say fuzzy logic, which make it applicable with other, especially non-linear and unknown models, applications.

Figures 12 through 15 show the results of the system under study in Experiments of external loads application. It is apparent that external load application causes a disturbance in the system response, which causes the error signal, at the instant of load application, to be increased (see figures 12.a, 12.c, 13.a, 13.c, 14.a, 14.c, 15.a, and 15.c). As the error signal e(t) increases at instant of load application, the PID based on GAFLNN controller forces the output to return to its value before load application. Consequently, the steady state error becomes zero. So it is clear that:

$$T(t) - T_L = J.\frac{d\omega(t)}{dt} + B.\omega(t) \qquad (3)$$

As the steady state error is zero, and then $\frac{d\omega(t)}{dt}$ is also zero at steady state. So, the right hand side of eq.(3) is constant at steady state (B is constant, and ω(t) is also constant at steady state). As a result, the left hand side of eq.(3) is also constant at steady state, which forces T(t) to be increased when T_L is increased (see figures, 12.d, 13.d, 14.dand 15.d). Due to the linearity in relation between the armature current $i_a(t)$ and torque T(t), armature current increases after an external load is applied (see figures, 12.b, 13.b, 14.band 15.b).The system response has a good tracking to the reference input, and the steady state error is zero after external load is applied, this is due to the fact that a PIDGAFLNN controller had been used. There is a small oscillation at step and external load application times due to the integral part of the PIDGAFLNN controller.

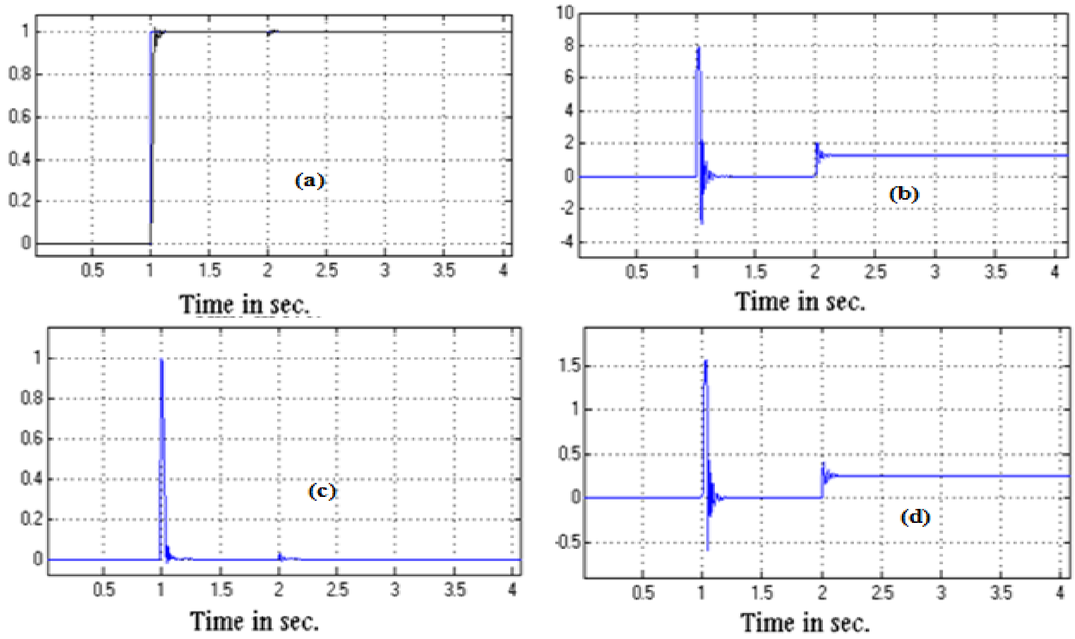


Fig12 The simulation results of Experiment 1: a. step response. b.armature current (A). c. normalized error signal of SM. d.torque (N.m)..

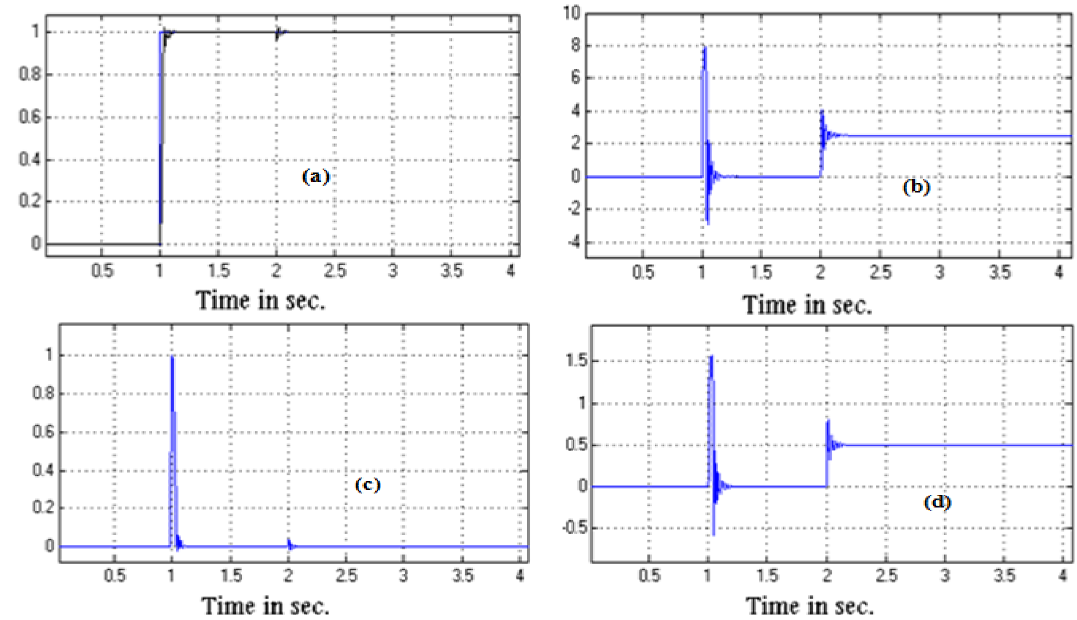


Fig.13 The simulation results of Experiment 2: a. step response. b.armature current (A). c. normalized error signal of SM. d.torque (N.m).

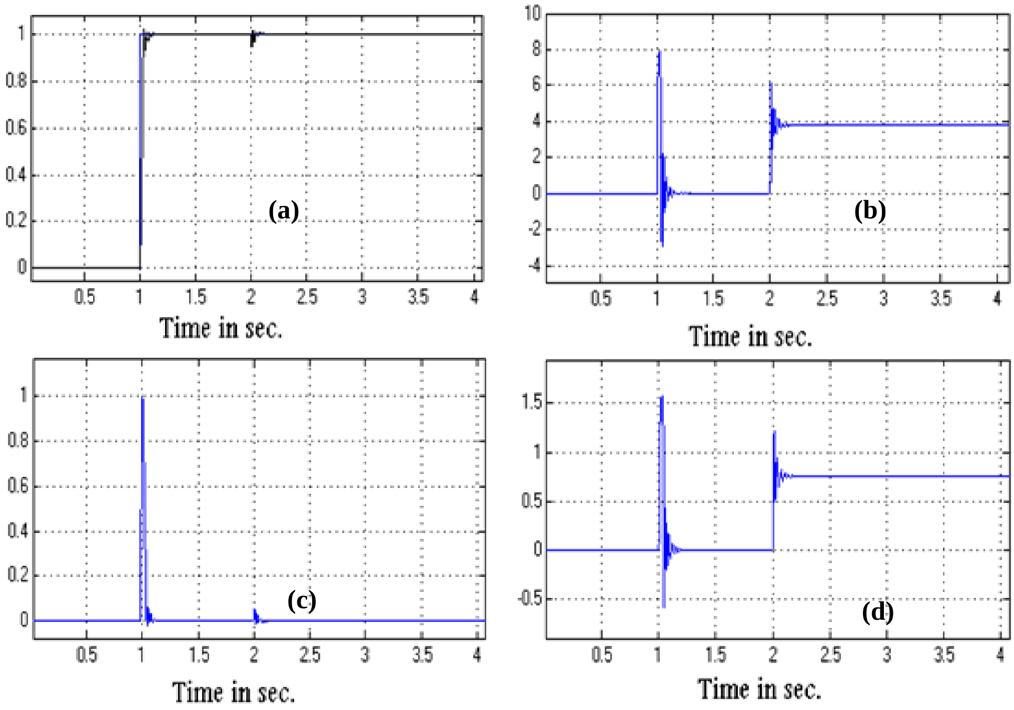


Fig. 14 The simulation results of Experiment 3: a. step response. b.armature current (A). c. normalized error signal of SM. d. torque (N.m).

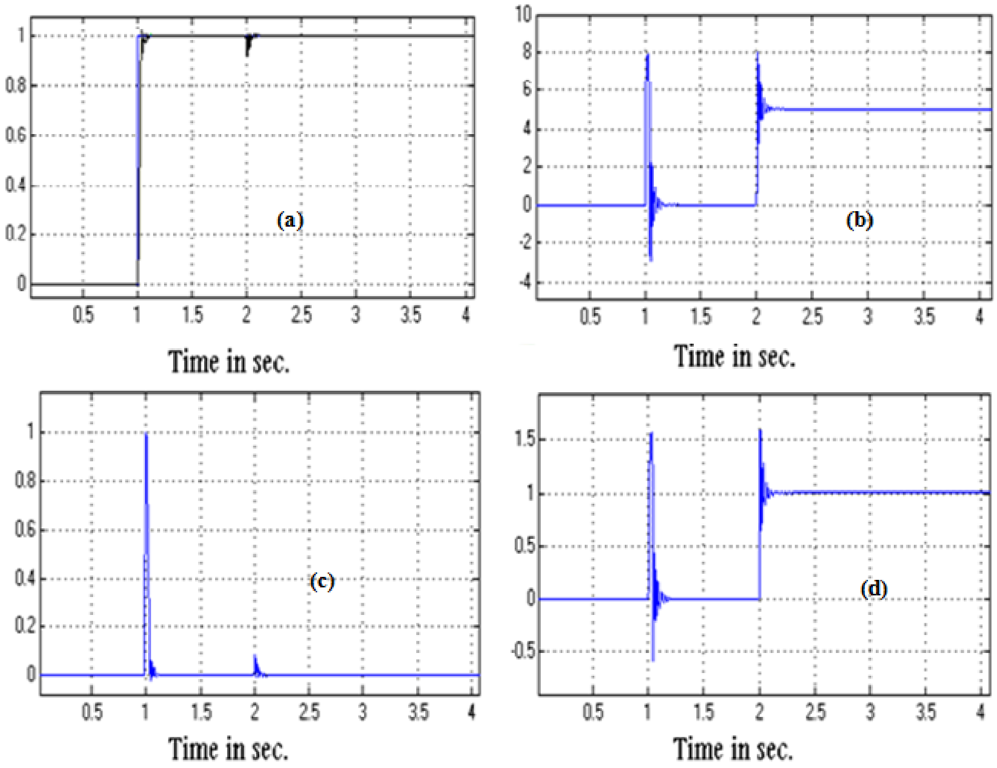


Figure 15 The simulation results of Experiment 4: a. step response. b.armature current (A). c. normalized error signalof SM. d.torque (N.m).

V. Conclusion

A new powerful method was proposed to design PID controller based on fuzzy logic neural network tuned by genetic algorithm (GAFLNN). The proposedGAFLNN controllerneeded many parameters like membership functions, number of rules, scalar factors fuzzification, defuzzification procedures, numbers of hidden layer of neural network and parameters of genetic algorithm. These different parameters make the GAFLNN controller more robust but much difficult for design.The new GAFLNN controllers like human operator to control the SM not needed the accurate mathematical model of the SM to be controlled. The only general behavior of the SM is required to be controlled and the parameters of the proposed controllerare adjusted to make output of SM as near as possible to the reference input, which would make the overall SM system more robust. The adjustment of the parameters in this Strategy was the optimal way to produce optimal output and reduce Settling and Rising times, and peak over shoot even unknown load characteristics has been used. The high performance of the proposed controller leadto highly reduced of the SM system error signal e(t).The GAFLNN controller was simulated by MATLAB Simulink and Simulation results show a considerable improvement in rising time and settling time besides reduces overshoot, IAE and ISE.Simulated results also show high dynamic performance characteristics to (SM) systemwithvariations of external load disturbance.

References

[1] Ahmed R., Abdul R. Ofoli and Donatus C. “DSP-Based Real-Time Implementation of a Hybrid H∞ Adaptive Fuzzy Tracking Controller for Servo-Motor Drives” IEEE Transactions On Industry Applications, VOL. 43, NO. 2, march/april,2007.

[2] Bimal K. Bose “Power Electronics and Motor Drives Advances and Trends” Elsevier Inc.2006.

[3] Cheng Qi-ming, "The Control System of Multi-variable PID Neural Network and Its Application in Deaerator Water Level Control", Journal of Shanghai University of Electric Power, Vol.23, No.1, pp.33-38, January,2007.

[4] Crockett, K , Bandar, Z. Fowdar, J, O'Shea, “On Producing Balanced Fuzzy Decision Tree Classifiers", IEEE World Congress in Computational Intelligence 2006, Vancouver, Canada. IEEE, pp 8415- 8421,2006.

[5] Dongmei Yu, Qing Hu and Fei Zhao" Synchronous Motion Control of Biaxial Driving System with Linear Servo Motors", Proceedings of the IEEE International Conference on Automation and Logistics Shenyang, China,2009.

[6] Feng G “A Survey on analysis and design of model-based fuzzy control systems,” IEEE Trans. Fuzzy Syst., vol. 14, no. 5, pp. 676–697, Oct. ., 2006

[7] Ping Zhou; Tianyou Chai; Jing Sun, "Intelligence-Based Supervisory Control for Optimal Operation of a DCS-Controlled Grinding System," Control Systems Technology, IEEE Transactions on , vol.21, no.1, pp.162,175, Jan. 2013

[8] Grabowski P. Z., Kazmierkowski M. PBose., B. K., and F. Blaabjerg, 2000 “A simple direct-torque neuro-fuzzy control of PWM-inverter-fed induction motor drive,” IEEE Trans. Ind. Electron., vol. 47, pp. 863–870, August.

[9] Mohan B.M., Sinha A., “Analytical structures for fuzzy PID controllers”, IEEE Trans. Fuzzy Syst. 16 (1) 52–60,(2008).

[10] Nicholas B., “A Simple Multi-Chromosome Genetic Algorithm Optimization of a Proportional-plus-Derivative Fuzzy Logic Controller” IEEE Transactions on Fuzzy Systems, vol. 16, pp. 611-616,2008.

[11] Febin, T.P.; Sakthivel, G.; Vinodha, R., "Design of genetic algorithm based optimal fuzzy logic controller for a non linear process," Green Computing Communication and Electrical Engineering (ICGCCEE), 2014 International Conference on , vol., no., pp.1,6, 6-8 March 2014

[12] Sekhar, P.C.; Mishra, S.; Sharma, R., "Data analytics based neuro-fuzzy controller for diesel-photovoltaic hybrid AC microgrid," Generation, Transmission & Distribution, IET , vol.9, no.2, pp.193,207, 1 29 2015

[13] Dogman, A.; Saatchi, R., "Multimedia traffic quality of service management using statistical and artificial intelligence techniques," Circuits, Devices & Systems, IET , vol.8, no.5, pp.367,377, 9 2014

[14] RaduE. P., Stefan P., EmilM P., Jozsef K. T., Marius L. T., Claudiu P. “Generic two-degree-of-freedom linear and fuzzy controllers for integral processes” Journal of the Franklin Institute, Published by Elsevier Ltd, 2009.

[15] Shahnazi R., Shanechi H.M,.Pariz N “Position control of induction and DC servomotors: a novel adaptive fuzzy PI sliding mode control”, IEEE Trans. Energy Conv. 23 (1) 138–147, (2008).

[16] Vengerov, D. Bambos, N. Berenji, H “A fuzzy reinforcement learning approach to power control in wireless transmitters”. IEEE Transactions on Systems, Man, and Cybernetics, Part B 35(4): 768-778, 2005.

[17] Shaowen Lu; Ping Zhou; Tianyou Chai; Wei Dai, "Modeling and Simulation of Whole Ball Mill Grinding Plant for Integrated Control," Automation Science and Engineering, IEEE Transactions on , vol.11, no.4, pp.1004,1019, Oct. 2014

[18] Navid K. D., Aidin S., Seyed H. H. Design of a PID Feed-Forward Controller for Controlling Output Fluid Temperature in Shell and Tube Hear Exchanger. Journal of Electrical and Electronic Engineering. Special Issue: Research and Practices in Electrical and Electronic Engineering in Developing Countries. Vol. 3, No. 2-1, 2015, pp. 30-34. doi: 10.11648/j.jeee.s.2015030201.17

[19] Z.Qiao,T.Shi,Y.Wang,Y.Yan,C.Xia,X.He,New sliding-mode observer for position sensor less control of permanent-magnet synchronous motor, IEEE Trans. Ind. Electron.60(February(2)),pp 710–719, (2013)

[20] Y. Maldonado, O. Castillo, and P. Melin, “Particle swarm optimization of interval type-2 fuzzy systems for FPGA applications,” Appl. SoftComput., vol. 13, pp. 496–508, Jan. 2013.

[21] Stinean, A.-I.; Preitl, S.; Precup, R.-E.; Dragos, C.-A.; Radac, M.-B.; Petriu, E.M., "Low-cost neuro-fuzzy control solution for servo systems with variable parameters," Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2013 IEEE International Conference on , vol., no., pp.156,161, 15-17 July 2013