MOTION CONTROLLER DESIGN FOR THE SPEED CONTROL OF DC SERVO MOTOR

MEHMET AKAR[1]
Tokat Vocational High School
Gaziosmanpaşa University
Taşlıçiftlik Campus

İSMAİL TEMİZ[2]
Electrical Education
Marmara University
Göztepe Campus

Abstract: **This paper presents a study on speed control problem for Direct Current (DC) servo motor. In this study DC servo motor's mathematical model and equations were extracted and there were three different motion controller designed for controlling the angular velocity. It was created simulation model at the Matlab programme and implemented proportional-integral, fuzzy and adaptive neuro fuzzy inference system controllers respectively while it was working variable conditions. At the servo motor application it is expected that no overshoot, settling reference value quickly and minimum influence on the system stability.**

Key-Words: **DC Servo Motor, Proportional-Integral, Fuzzy, Adaptive Neuro Fuzzy Inference System**

1. Introduction

Electric motors can be classified by their functions as servomotors, gear motors, and so forth, and by their electrical configurations as DC (direct current) and AC (alternating current) motors. A further classification can be made as single phase and polyphase with synchronous and induction motors in terms of their operating principles for AC motors, and PM (permanent magnet) and shunt DC motors for DC's. Although DC motors are preferred dominantly in the variable speed applications, increasing use of AC motors can be seen prior to improvements in solid state components. Servomotor is a motor used for position or speed control in closed loop control systems. The requirement from a servomotor is to turn over a wide range of speeds and also to perform position and speed instructions given. DC and AC servomotors are seen in applications by considering their machine structure in general [1]. DC servo motors have been used generally at the computers, numeric control machines, industrial equipments, weapon industry, and speed control of alternators, control mechanism of full automatic regulators as the first starter, starting systems quickly and correctly [2]. In the field of control of mechanical linkages and robots, research works are mostly found on DC motors.

While some properties of DC servo motors are the same, like inertia, physical structure, shaft resonance and shaft characteristics, their electrical and physical constants are variable. The velocity and position tolerance of servo motors which are used at the control systems are nearly the same. So they must be controlled according to the control system needs.

For this aim; it has implemented proportional-integral, fuzzy logic and adaptive neuro fuzzy inference system respectively at the variable working situations to the simulation model which has prepared at the Matlab programme for improvement the servo motor performance [2].

2. Motion Controllers

2.1. Proportional-Integral Based Controller (PI)

PI controller is unquestionably the most commonly used control algorithm the process control industry. The main reason is its relatively simple structure, which can be easily understood and implemented in practice, and that many sophisticated control strategies, such as model predictive control, are based on it [3].

PI control is math total of integration error and multiplying of error with constant. This situation depicted in equation 1.

$$u(t) = Kp.e(t) + K\imath\int_0^t e(t).dt \qquad (1)$$

Kp, Kı, u(t) and e(t) are proportional gain, integral gain, controller output and error signal, respectively.

There are a lot of methods for calculating the parameters of PI controller. Some of them are; Ziegler-Nichols, Åström-Hägglund, Refined Ziegler-Nichols. PI controller block diagram as illustrated in figure 1.

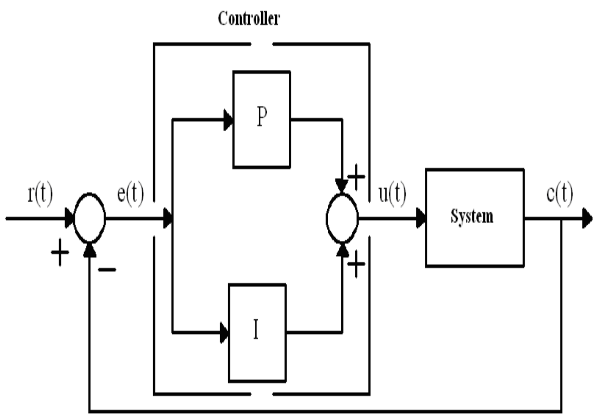


Figure 1: PI controller block diagram

2.2. Fuzzy Logic Based Controller

There is a currently a significant and growing interest in the application of artificial intelligence (AI) type models to the problem of modeling the dynamics of complex, nonlinear processes. By far the most popular type of AI model for these purposes has been the neural network, which attempts to produce 'intelligent' behavior by recreating the hardware involved in the thinking process. Another type of AI model is the fuzzy model, which defines its inputs and outputs as qualitative values (actually fuzzy reference sets) and then defines the strength of the relationships between these input and output reference sets [4].

First of all mathematical model is necessary while making experimentation for control systems but it may not be always possible in practice. Sometimes in spite of creating true model, using of this model makes some complicated problems at the implementation.

When confronted with like these problems, knowledge and experience of expert people is utilized. These people develop flexible control mechanism by using the words frequently in our life ,like "suitable", "not very suitable", "high", "a little high", "more", "very more" which are defined special variable. Fuzz logic control which has experimentation of fuzzy cluster theories and fuzzy logic, has been set up like logical relatives.

A fuzzy logic system is based on 4 main parts which are illustrated in figure 2 [5].

1-) Input units which has been known fuzzifier
2-) Rule base
3-) Inference mechanism
4-) Output units which has been known defuzzifier

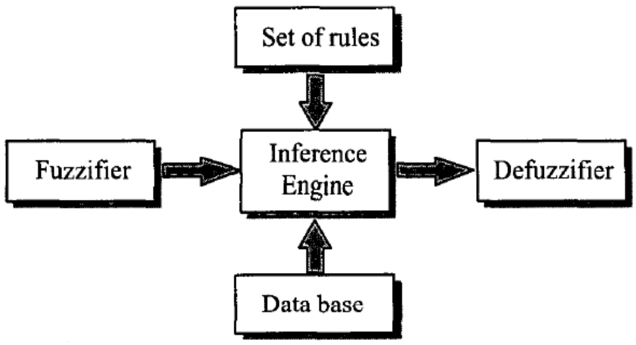


Figure 2: Block diagram of the fuzzy logic controller

2.3. Adaptive -Neuro Fuzzy Inference System Based Controller (Anfis)

Adaptive Neuro-Fuzzy Inference Systems are fuzzy Sugeno models put in the framework of adaptive systems to facilitate learning and adaptation. Such framework makes FLC more systematic and less relying on expert knowledge. To present the ANFIS architecture, let us consider two-fuzzy rules based on a first order Sugeno model:

Rule 1: if (x is A1) and (y is B1) then
(f1 = p1x + q1y + r1)
Rule 2: if (x is A2) and (y is B2) then
(f2 = p2x + q2y + r2)

One possible ANFIS architecture to implement these two rules is shown in Fig. 1. Note that a circle indicates a fixed node whereas a square indicates an adaptive node (the parameters are changed during training). In the following presentation *OLi* denotes the output of node *i* in a layer *L*.

Anfis algorithm is composed of fuzzy logic and neural networks with 5 layers [6].

Properties of layer described for x, y as an input variables.

Layer 1: All the nodes in this layer are adaptive nodes, *I* is the degree of the membership of the input to the fuzzy membership function (MF) represented by the node:

$$O_{1,i} = \mu_{Ai}(x) \qquad i = 1,2$$
$$O_{1,i} = \mu_{Bi-2}(y) \qquad i = 3,4 \qquad (2)$$

Ai and *Bi* can be any appropriate fuzzy sets in parameter form. For example, if bell MF is used then,

$$\mu_{Ai}(x) = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i} \right)^2 \right]^{b_i}} \qquad (3)$$

where *ai* ,*bi* and *ci* are the parameters for the MF.

Layer 2: The nodes in this layer are fixed (not adaptive). These are labelled *M* to indicate that they play the role of a simple multiplier. The outputs of these nodes are given by:

$$O_{2,i} = \omega_i = \mu_{Ai}(x).\mu_{Bi}(y) \qquad i = 1,2 \qquad (4)$$

The output of each node is this layer represents the firing strength of the rule.

Layer 3: Nodes in this layer are also fixed nodes. These are labelled *N* to indicate that these perform a normalization of the firing strength from previous layer. The output of each node in this layer is given by:

$$O_{3,i} = \overline{\omega}_i = \frac{\omega_i}{\omega_1 + \omega_2} \qquad i = 1,2 \qquad (5)$$

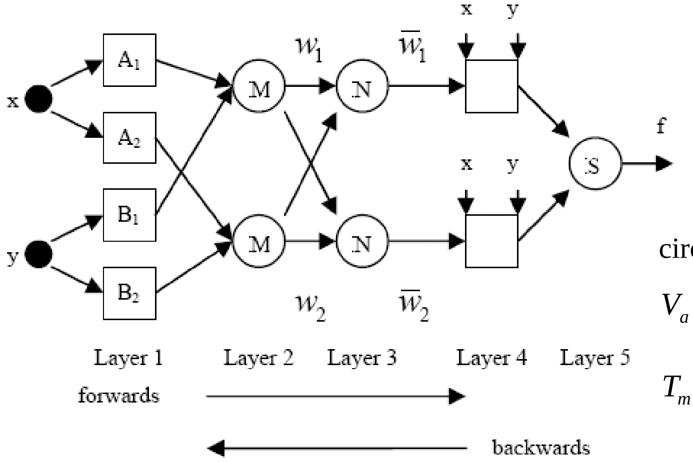


Figure 3: Structure of Anfis with 2 inputs, 1 output

Layer 4: All the nodes in this layer are adaptive nodes. The output of each node is simply the product of the normalized firing strength and a first order polynomial:

$$O_{4,i} = \overline{\omega}_i f_i = \overline{\omega}_i (p_i x + q_i y + r_i \quad i = 1,2 \qquad (6)$$

where *pi*, *qi* and *ri* are design parameters (consequent parameter since they deal with the then-part of the fuzzy rule)

Layer 5: This layer has only one node labelled *S* to indicate that is performs the function of a simple summer. The output of this single node is given by:

$$O_{i,5} = f = \sum_i \overline{\omega}_i f_i = \frac{\sum_i \omega_i f_i}{\sum_i \omega_i} \qquad i = 1,2 \qquad (7)$$

The ANFIS architecture is not unique. Some layers can be combined and still produce the same output. In this ANFIS architecture, there are two adaptive layers [6]. Layer 1 has three modifiable parameters (*ai*, *bi* and *ci*) pertaining to the input MFs. These parameters are called *premise* parameters. Layer 4 has also three modifiable parameters (*pi*,*qi* and *ri*) pertaining to the first order polynomial. These parameters are called *consequent* parameters.

3. DC Servo Motor Mathematical Model

The velocity of the DC servo motor is controlled by changing the supply voltage. According to this theory if rewrites voltage and moment equations;

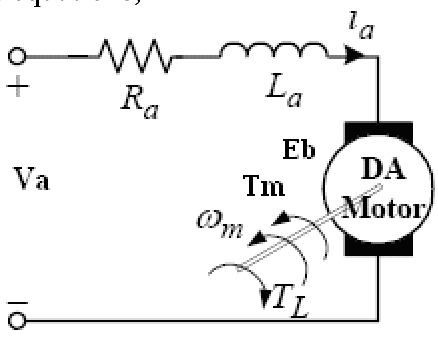


Figure 4: Dc servo motor equivalence circuit

$$V_a(t) = R_a.i_a(t) + L_a \frac{di_a(t)}{dt} + E_b(t) \qquad (8)$$

$$T_m = j \frac{d\varpi_m(t)}{dt} + B.\varpi_m(t) + T_L(t) \qquad (9)$$

In order to create the block diagram of system; initial conditions are acquiescence zero and laplace transform is implemented to the equations.

$$I_a(s) = \frac{E_a(s) - K_b.\varpi_m(s)}{R_a + L_a s} \qquad (10)$$

$$s.\varpi_m(s) = \frac{K}{j_m} I_a(s) - \frac{B}{j_m}\varpi_m(s) - \frac{T_L(s)}{j_m} \qquad (11)$$

$$\varpi_m(s) = \frac{K.I_a(s) - T_L(s)}{B + sj} \qquad (12)$$

$$E_b(s) = K_b.\varpi_m(s) \qquad (13)$$

$$\frac{\omega_m(s)}{E_a(s)} = \frac{K_i}{s^2 J_m L_a + sJ_m R_a + K_i K_b} \qquad (14)$$

PI constants are calculated by these equations.

$$\omega_c T_d - \frac{1}{\omega_c T_i} = \tan\phi_m \qquad (15)$$

$$T_i = \alpha.T_d \qquad (16)$$

$$T_d = \frac{\tan\phi_m + \sqrt{\frac{4}{\alpha} + \tan^2\phi_m}}{2\omega_c} \qquad (17)$$

$$K_p = \frac{g_m}{|G(j\omega_c)|}\cos\phi_m = K_c\cos\phi_m \qquad (18)$$

$i_a(t)$ =Armature current L_a = Armature inductance

R_a = Armature resistance $V_a(t)$ =Input voltage

$E_b(t)$ =Back emf. K_b =Voltage constant

T_L =Load moment T_M =Motor moment

ϖ_m =Motor angular velocity K_i =Moment constant

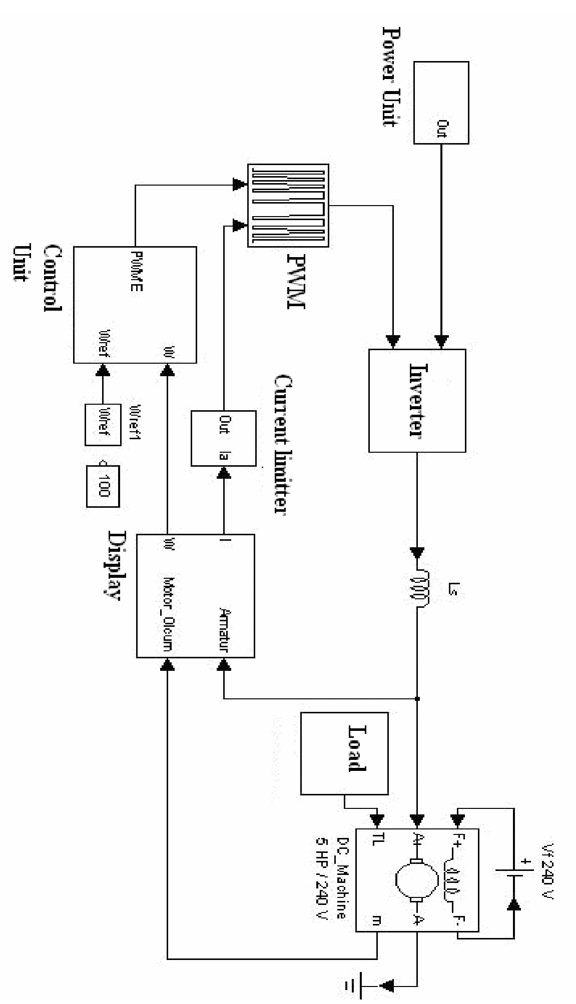
J_m =Motor moment of inertia φ_m =Phase margin

B_m =Friction constant g_m =Gain magrin

Kp, Ti, Td, ω_c = Proportional gain,Integral Time,Derivative Time constant, Critical frequency

4. Application

Proportional-integral controller which is known as a classical controller is created for DC servo motor according to Åström-Hägglund, phase response. DC servo motor's current (I_a), angular velocity (ω_m) is simulated for constant and variable velocity position under the load at the Matlab programme. Then for the same working situations fuzzy and anfis based controllers are created for DC servo motor.



del

V=240 V	I=15 A	n=3000 rpm
M_y=5 Nm	B=0.02 Nms	J=0.05 kg m^2
K=5/15,3Nm/A	Ra=0,5 Ω	La=0,01 H

Table 1: Servo Motor Parameters

In this simulation it is aimed to control motor velocity with the motor current and armature's voltage. $z\omega = \omega - \omega_{ref}$ which is called angular velocity error is used as an input variable. Armature current must be limit for soft start and protect the motor from high current. So that armature current was used as a second input variable.

Output fuzzy cluster is created as an output variable which regulates PWM duty period according to error between motor velocity information and reference value.

Membership functions represents the motor velocity information with the words "very small", "small", "ok", "much", "very much" and people experience is reflected to the system.

Rules used at the controllers which are based on human experience and information are:

1- If ω_{err}= “Very small” => output=”increase”
2- If ω_{err}= “Small” => output=”a little increase”
3- If ω_{err}= “Ok” => output =”Ok”
4- If ω_{err}= “More “=>output =”a little decrease”
5- If ω_{err}= “Most” => output =”decrease”

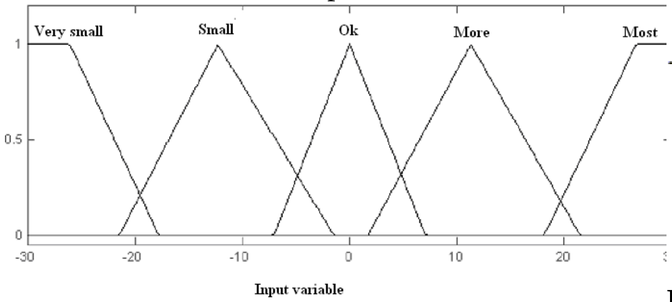


Figure 6: Membership functions relating to $\Delta\omega$ inputs

In figure 6 signal angular velocity error (Δω) range is limited between [-30, 30] values. Negative zone defines motor velocity slower than the reference and positive zone defines faster.

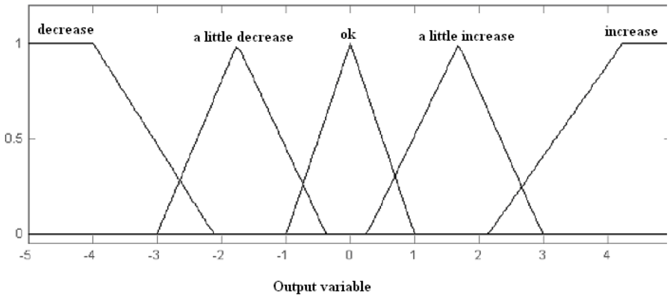


Figure 7: Membership functions belong to PWM Duty output

Output membership function represents how much the period of pwm duty shall increase or decrease. It includes 5 membership functions between [-5, 5] in range. Fuzzy clusters are defined as "decrease", "a little decrease", "ok", "a little increase" and "increase"

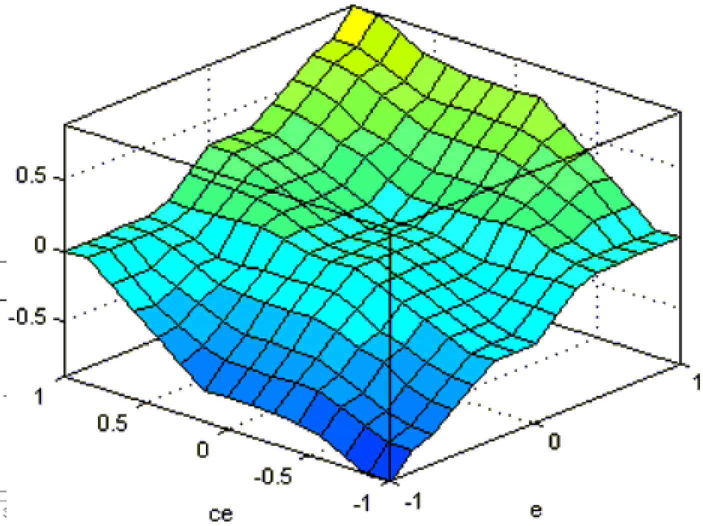


Figure 8: 3-D fuzzy controller surface

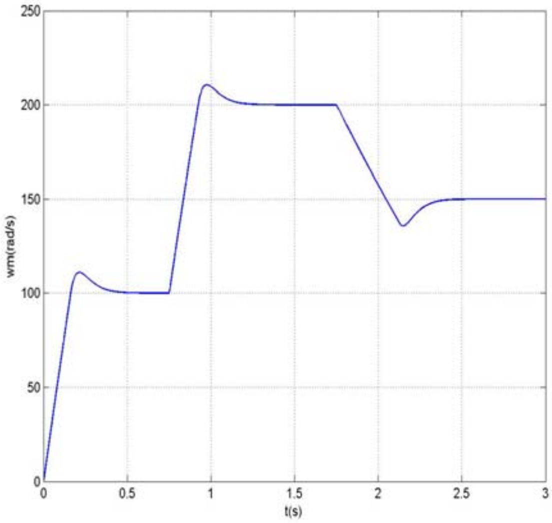


Figure 9: PI controller output under variable load and reference velocity.

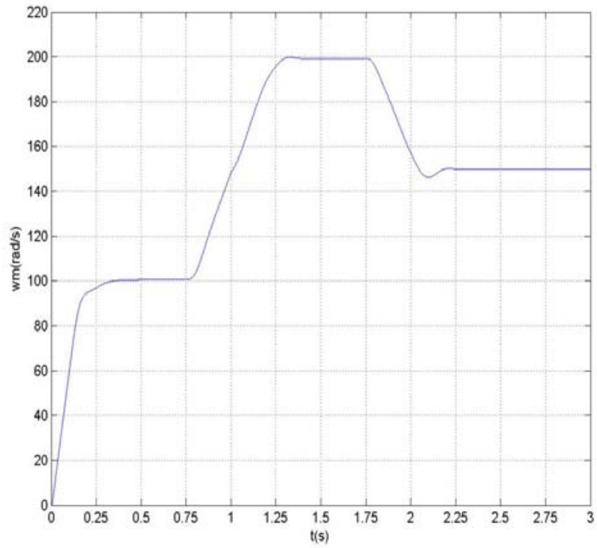


Figure 10: Fuzzy controller output under the variable load and reference velocity.

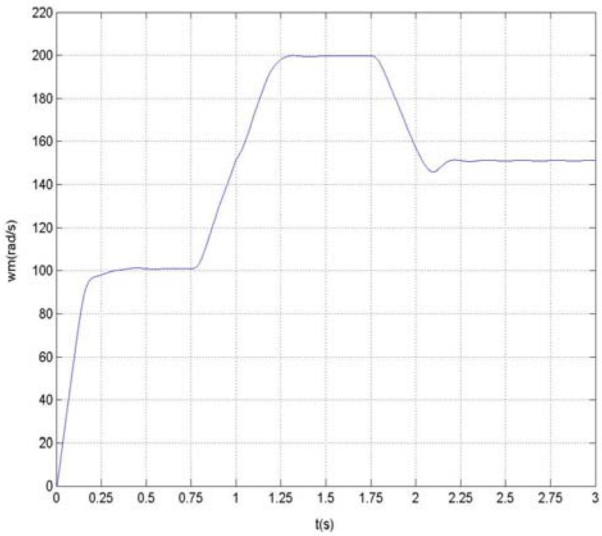


Figure 11: Anfis controller output under the variable load and reference velocity.

At figure 9, 10, 11 variable reference velocity and load response are illustrated for three controllers. When studied carefully these figures according to rising time, settling time, positive and negative overshoot best performance is belong to anfis.

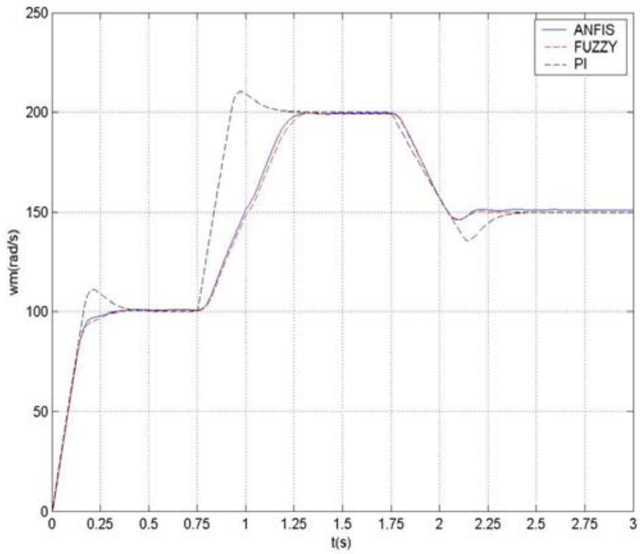


Figure 12: All controller output under the variable load and velocity.

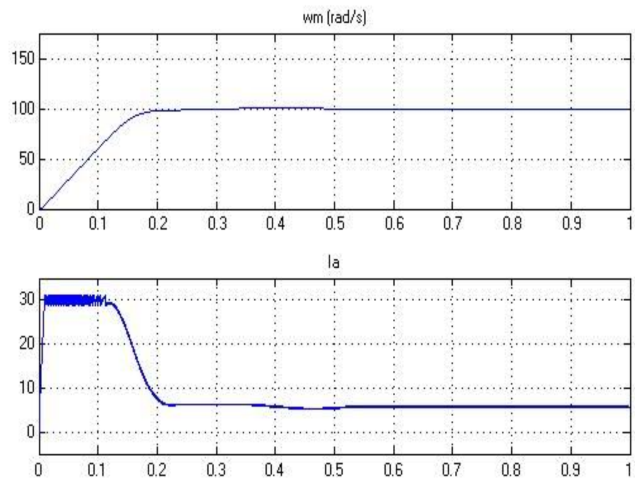


Figure 13: Fuzzy controller velocity-time, current- time response under the 3 Nm load and 100 radian/s reference velocity.

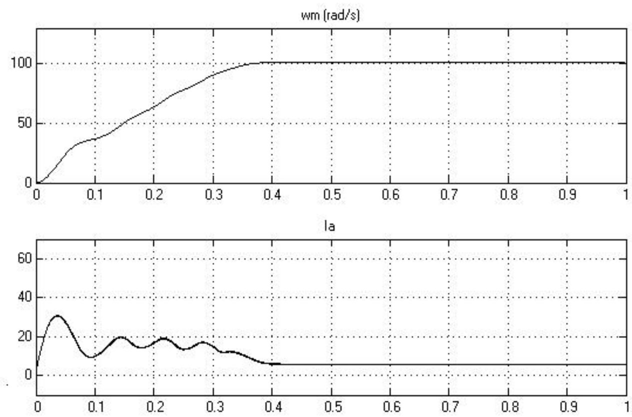


Figure 14: Anfis controller velocity-time, current-time response under the 3 Nm load and 100 radian/s reference velocity.

5. Conclusion

In this study, 3 and 5 Nm load under the variable and reference speed is implemented to the DC servo motor which is used as a model. At the 3 Nm load, although PI controller is stabilized position at 0.4 second overshoot is observed at response of system. At the same working conditions, it is observed no overshoot for fuzzy and anfis controllers, settling time for fuzzy and anfis controller is 0.38 and 0.33 seconds respectively.

When the load moment on DC servo motor's shaft increased 5 Nm and system response tested for variable velocity it is observed that the response of anfis controller is better than the others.

Current's responses are between reasonable values for three controllers at the test situation which is described above. Because of learning characteristic of error the performance of anfis controller is better than the others under variable load and velocity. But fuzzy and anfis controller's response are nearly the same because of the current limit. The control operation is not performed in the limited zone.

References

[1] L. Canan Dülger and Ali Kireçci, "Motion Control and Implementation for an AC Servomotor System," *Modelling and Simulation in Engineering*, vol. 2007, Article ID 50586, 6 pages, 2007. doi:10.1155/2007/50586

[2] Temiz I. and Akar M.," The Speed Control of DC Servo Motor with Proportional Integral, Fuzzy Logic and Adaptive Neuro-Fuzzy Inference Systems" , World Scientific and Engineering Academy and Society,pp.189-194,İstanbul,2007.

[3] Lee, Y., Lee, J., Park, S., (2000), PID Controller Tuning for Integrating and Unstable Processes with Time Delay, *Chem. Eng. Sci.*, vol. **55**, pp. 3481-3493.

[4] Paytra M.J., Mlynek D.M.,1996, Fuzzy logic implementation and Applications, John Wiley pres, New York, 317p.

[5] Zadeh,L.A.: "Bulanık Sets,Conformation and Control 8 ", 339-353

[6] Z. Hímer, V. Wertz, J. Kovács, U. Kortela Neuro- fuzzy model of flue gas oxygen content Proccedings of IASTED International Conference on Modelling Identification and Control, Grindelwald, Switzerland 2004

AUTHOR CONTACT INFORMATION

MEHMET AKAR[1] : Tokat Vocational High School , Gaziosmanpaşa University ,60200 Tokat ,Turkey
mehmetakar@gop.edu.tr

İSMAİL TEMİZ[2]: Faculty of Technical Education, Marmara University Göztepe Campus , İstanbul,Turkey.
itemiz@marmara.edu.tr