CHAPTER 6

COMPARATIVE ANALYSIS OF PI, FUZZY AND ANN CONTROLLERS

6.1 INTRODUCTION

This chapter deals with the comparison of PI, Fuzzy and Artificial Neural network (ANN) controllers for closed loop speed control of DC drive fed by a buck type DC-DC power converter. The performance of the controllers is studied using simulation and then validated using experimental implementation on an embedded system.

A conventional PI controller initially designed for the DC drive system using its small signal modeling is considered. Then, the design of Fuzzy and ANN controllers are considered. The ANN controller is first designed using training data from fuzzy controller. Then another ANN controller is designed using the training data from PI controller. The neuron controllers were trained using the neural network toolbox in MATLAB.

MATLAB/simulink model of the DC motor with the DC-DC converter is developed and simulated. The closed loop operation is simulated with the designed controllers and performance is studied. The parameters selected for the comparison are the steady state error and the rise time of the response.

6.2 COMPUTER SIMULATIONS AND COMPARISON

Using Matlab/ simulink toolbox, the designed PI, fuzzy and neuron controllers are tested. The disadvantage found in the PI controller is its control action can take very big values that are impossible to consider in real implementation. On the other hand, the
fuzzy controller uses real range of values. In the present case, the PI controller gives a high value of change in duty cycle. But the duty cycle has to be limited between 0 and 1. The PI controller output varies between the two extreme high limits and the PI control surface is not smooth. The fuzzy controller surface can be made smooth by proper choice of the fuzzy membership functions and inference rules. The control surface of the fuzzy controller designed is given in Figure 6.1.

![Figure 6.1 - Control surface of the fuzzy controller](image)

The neuron controller surface can be still made smooth and the controller can be made more effective over the given range of the error and change in error. As the training data are taken during the simulation of the motor, the inaccuracies and the limitations of the motor are also considered while creating the training data. The controller surface of the PI implemented neuron controller is given in Figure 6.2. The trained neuron
controller outperforms the original controllers and can deal with the uncertainties as it has

got the different improved control surface.

![Graph showing control surface of the PI implemented ANN controller](image)

**Figure 6.2 - Control surface of the PI implemented ANN controller**

The computer simulation is run for a step change in motor reference speed and
the corresponding change in speed is recorded. The step change in load torque is applied
and the corresponding change in the speed is also recorded.

The simulated graph of speed variation of the motor for PI, fuzzy, Artificial
Neural Network controllers are given in Figures 6.3 to 6.6. A step change in reference
speed from 1000rpm to 1500rpm is applied at 3secs with the no load. A step change in
load torque from zero to rated value is applied at 6secs. These results show that the
neuron controller trained using the data from PI controller has the better performance
compared to conventional and fuzzy controllers.
Figure 6.3 – Simulated speed variation for the step change in reference speed from 1000rpm to 1500rpm and for the change in load torque from zero to rated value at t=6secs with PI controller

Figure 6.4 - Simulated speed variation for the step change in reference speed from 1000rpm to 1500rpm and for the change in load torque from zero to rated value applied at t=6secs with Fuzzy controller
Figure 6.5 – Simulated speed variation for the step change in reference speed from 1000rpm to 1500rpm and for change in load torque from zero to rated value applied at t=6secs with ANN controller trained with fuzzy controller data.

Figure 6.6 – Simulated speed variation for the step change in reference speed from 1000rpm to 1500rpm with load torque from zero to rated value applied at t=6secs with ANN controller trained with PI controller data.

A comparative study of steady state error and rise time for the controllers under study is shown in Tables 6.1 and 6.2 respectively. The study was done for 60% and 100% of rated speed without load torque and with full load torque. The Artificial Neuron
controller trained with data from PI controller is found to provide low steady state error and rise time over the other controllers under comparison. Also from the table, it can be seen that the ANN controller trained with the data from conventional PI controller results in low steady state error and rise time.

Table 6.1 Comparative study of steady state error (in % of rated speed) for controllers under study

<table>
<thead>
<tr>
<th>Speed</th>
<th>Load</th>
<th>PI</th>
<th>Fuzzy</th>
<th>ANN trained with Fuzzy controller data</th>
<th>ANN trained with PI controller data</th>
</tr>
</thead>
<tbody>
<tr>
<td>60% of rated value</td>
<td>0% of rated torque</td>
<td>1.2</td>
<td>0.55</td>
<td>0.44</td>
<td>0.29</td>
</tr>
<tr>
<td></td>
<td>100% of rated torque</td>
<td>1.0</td>
<td>0.75</td>
<td>0.6</td>
<td>0.56</td>
</tr>
<tr>
<td>100% of rated value</td>
<td>0% of rated torque</td>
<td>2.0</td>
<td>1.67</td>
<td>1.0</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>100% of rated torque</td>
<td>8.7</td>
<td>1.86</td>
<td>1.0</td>
<td>0.625</td>
</tr>
</tbody>
</table>

Table 6.2 Comparative study of Rise time (in seconds) for controllers under study

<table>
<thead>
<tr>
<th>Speed</th>
<th>Load</th>
<th>PI</th>
<th>Fuzzy</th>
<th>ANN trained with Fuzzy controller data</th>
<th>ANN trained with PI controller data</th>
</tr>
</thead>
<tbody>
<tr>
<td>60% of rated value</td>
<td>0% of rated torque</td>
<td>1.057</td>
<td>0.8313</td>
<td>0.83</td>
<td>0.814</td>
</tr>
<tr>
<td></td>
<td>100% of rated torque</td>
<td>11.4</td>
<td>5.45</td>
<td>4.89</td>
<td>4.6</td>
</tr>
<tr>
<td>100% of rated value</td>
<td>0% of rated torque</td>
<td>1.9</td>
<td>1.417</td>
<td>1.42</td>
<td>1.417</td>
</tr>
<tr>
<td></td>
<td>100% of rated torque</td>
<td>23.5</td>
<td>10.5</td>
<td>9.95</td>
<td>9.95</td>
</tr>
</tbody>
</table>

6.3 EXPERIMENTAL IMPLEMENTATION

The designed controllers were implemented using Cygnal 8051 based processor (C8051F005) and programming it. A buck converter was built and the controllers were
tested on the 220V DC motor. A tacho-generator was used to sense the speed and to feedback the speed signal. A LEM make current sensor LTS25NP is used to sense the armature current and it is compared with the reference current using a comparator. The AND gate is used to allow the PWM waveform when the actual current is less than the reference current. This PWM waveform is then level amplified and fed to the DC -DC power converter through an isolator and driver chip. The Buck converter output is used as supply to the armature of the DC motor, whose speed is to be controlled. The tacho generator connected to the motor shaft gives a DC voltage proportional to the speed and this DC voltage is fed to the ADC available in the micro-controller. The Figures 6.7 to 6.9 show the experimental graph of response of DC motor for a step change in reference speed applied to a prototype motor. Figure 6.7 shows the response of the Conventional PI controller. While Figures 6.8 shows the response of the Fuzzy controller, Figure 6.9 shows the response of the ANN controller trained using PI controller data. These results confirm the simulation results that the ANN controller reduces the rise time in the response.

![Graph showing experimental speed variation for the step change in reference speed using conventional PI controller](image)

Figure 6.7 - Experimental speed variation for the step change in reference speed using conventional PI controller
Figure 6.8 - Experimental speed variation for the step change in reference speed using Fuzzy controller

Figure 6.9 - Experimental speed variation for the step change in reference speed using ANN controller trained with PI controller data
6.4 CONCLUSION

From the studies, it is found that the ANN controller trained with the training pattern from PI controller has better response in terms of rise time and steady state error. This advantage arises from the fact that the neural network has the property of generalization and neural controller acts independent of the system parameters. Also, the architecture of the neural network controller is simple and uses only two layers with three biased neurons. The neuron controller is easy to implement and reduces the computation burden in embedded systems. The computation rate is high for the neural network controller and so it is best suitable for embedded system implementation.