Abstract- In this paper, position control of servo motor using PID controller with soft computing optimization techniques is discussed. PID controllers widely used in the industry. Different methods are available for tuning the PID controller. In this paper conventional tuning method Z-N method and soft computing methods like Genetic algorithm (GA) and Particle swarm optimization (PSO) are used for the position control of the DC servo motor. The results obtained from soft computing methods (GA, PSO) are compared with conventional tuning method (Z-N) found that the soft computing techniques gives better results compared to the conventional PID tuning method.

Key Words: DC servo motor, position control, tuning methods, ZN, GA and PSO methods.

INTRODUCTION

Now a day’s PID controllers are widely used in the industry. About 85-90% of the controllers are used in the industry are of PID type. Position control systems are normally unstable when they are implemented in closed loop configuration. PID controllers tuning for positional control systems is a time consuming task, therefore much effort has been given to analyse the servo systems.

The main aim of this paper is to analyse the soft computing methods and enumerate their advantages over conventional PID tuning methodologies. In this paper Position control of a 3rd ordered plant (Servo motor) using Conventional PID tuning and soft computing methods with their comparisons is analysed. Conventional PID tuning method Ziegler-Nichols, soft computing methods like genetic algorithm and PSO is used in this paper for the position control of servo systems.

Except for minor difference in constructional features a dc servo motor is essentially an ordinary dc motor. Physical requirements of DC servo motor are Low inertia and High starting torque. Low inertia is attained with reduced armature diameter with consequent in armature length such that the desired power output is reached.

SYSTEM MODELLING:

In this dc servo motor can be consider as a linear SISO system having 3rd order transfer function. Relation between shaft position and armature voltage is derived from the physical laws.

The air gap flux is given by

\[ \Phi = k_j i_f \]

Torque is proportional to product of Flux and Armature current

\[ T = k_\phi I_a(t) \]

Or

\[ T = k_i k_j I_f(t) I_a(t) \]

The motor torque when the constant flux established in the field coil is given by

\[ T = K_m I_a(t) \]

Back EMF of the motor is given by

\[ V_b = k_b \omega \]

By apply Laplace transform to the armature loop

\[ V_a(s) = R_a I_a(s) + L_a s I_a(s) + V_b(s) \]

Where \( V_b(s) \) is back EMF voltage proportional to the motor speed. Therefore, we have
The armature current is expressed as
\[ I_a(s) = \frac{V_a(s) - k_b w(s)}{R_a + sL_a} \]

The motor torque is expressed as
\[ T_m(s) = T_i(s) + T_L(s) \]

Here, \( T_i \) is the load torque
\[ T_i(s) = js^2 \theta(s) + Bs \theta(s) \]

The relation between speed and position is given by
\[ w(s) = s \theta(s) \]

The above equations can be represented in a block diagram as

\[ V_a(s) = k_b w(s) \]

From the above block diagram the relation between shaft position and armature obtained as
\[ \theta(s) = \frac{K_m}{m^2} \frac{V_a(s)}{S[(S L_a + R_a)(S^2 + B) + (K_m k_b)]} \]

J=0.01kg/m², B=0.1n.m.s, \( k_b=0.01 \) v/rad/sec, km=0.01N.m/amp, Ra=1 ohm, L=0.5H

Substitute above values in the above equation,
\[ \theta(s) = 0.01 \]

The output of The PID controller \((U(t))\) is given by
\[ U(t) = K_p e(t) + K_i \int e(t) + K_d \frac{d}{dt} e(t) \]

Where \( K_p, K_i, K_d \) are proportional, Integral and derivative gains and \( e(t) \) error= set point-output

The PID output in Frequency domain can be represented as
\[ \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s \]

The closed loop Transfer Function is given by
\[ Y(s) = \frac{G_c(s) G(s)}{1 + G_c(s) G(s)} \]

\( Y(s) = \text{Output response}(s) \), \( G(s) = \text{plant} \)

And \( G_c(s) = \text{controller} \)

Ziegler-Nichols (ZN) method is a conventional PID tuning method. This method is widely used for design of various controllers. Ziegler-Nichols presented two methods 1. Step response method and 2. Frequency response method. In this paper frequency response method is discussed for tuning the PID controller

PROCEDURE:

In this method derivative time \((T_d)\) is set to zero and integral time \((T_i)\) set to infinity. This is used to get the initial PID setting of the systems. The critical gain \((K_u)\) and periodic oscillations \((P_u)\) are determined by using R-H criteria. Ku is determined by equating the row containing ‘s’ in R-H row to zero. Pu is determined by equating the row containing ‘s-1’ in R-H row to zero. Evaluate parameters described by Z-N method. Values of \( K_p, K_i, K_d \) are determined using the formulas \( K_p=0.6*K_u \), \( K_i = K_p/T_i \) and \( K_d = K_p/T_d \).

\( K_p, T_i, T_d \) are calculated using the formulas given in below table, \( T_e = \frac{2\pi}{\omega_c} \)

<table>
<thead>
<tr>
<th>Control type</th>
<th>( K_p )</th>
<th>( T_i )</th>
<th>( T_d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>0.5 ( K_u )</td>
<td>inf</td>
<td>0</td>
</tr>
<tr>
<td>PI</td>
<td>0.45 ( K_u )</td>
<td>0.831c</td>
<td>0</td>
</tr>
<tr>
<td>PID</td>
<td>0.6 ( K_u )</td>
<td>0.51c</td>
<td>0.125c</td>
</tr>
</tbody>
</table>

Table 1: ZN PID tuning parameters

(This work is licensed under a Creative Commons Attribution 4.0 International License.)
The advantage of this method is applying easy rules to simple mathematical models. But the disadvantage of this method does not provide as good results as expected.

**GENETIC ALGORITHM:**

A genetic algorithm is a powerful searching capabilities and heuristic characteristics. GA has also been used in control tuning applications, being shown to obtain better results than classical techniques. Genetic algorithms are inspired from phenomena found in living organisms (nature). In Genetic algorithms they choose the next generations based on genetic operators like cross over, mutation selection and survival of fittest.

*The components of GA are*

- A problem definition as input, and encoding principles (gene, chromosome), initialization procedure followed by cross over, mutation and selection operators for reproduction with the help of an objective function.

*Simple Genetic Algorithm:*

```
{ Initialize population;
  Evaluate population;
  While Termination Criteria Not Satisfied
    { Select parents for reproduction;
      Perform recombination and mutation;
      Evaluate population;
    }
  }
```

*GA PARAMETERS:*

In this paper the following genetic algorithm parameters are used

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lower bounds</td>
<td>[0 0 0]</td>
</tr>
<tr>
<td>Upper bounds</td>
<td>[100 100 100]</td>
</tr>
<tr>
<td>Stopping criteria</td>
<td>100</td>
</tr>
<tr>
<td>Population size</td>
<td>40</td>
</tr>
<tr>
<td>Cross over fraction</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Table 2: The parameters of the genetic algorithms.

**PSO**

PSO is a robust stochastic optimization technique based on the movement and intelligence of swarms. The components of PSO are Swarm Size, Velocity, position components and maximum no of iteration. Here I have consider the following objective function

\[
F=(1-\exp(-0.5))^\star(M_p)+\exp(-0.5)^\star(t_s-t_r)
\]

\[M_p = \text{peak overshoot, } t_r = \text{rise time, } t_s = \text{settling time}\]

**Algorithm of PSO**

1. Create an initial population of particles with random positions and velocities within the solution space.
2. For each particle, calculate the value of the fitness function.
3. Compare the fitness of each particle with *local-best*. If current solution is better than its *local-best*, then replace its *local best* by the current solution.
4. Compare the fitness of all the particles with global *best*. If the fitness of any particle is better than global *best*, then replace global *best*.
5. Update the velocity and positions of all particles using velocity update equations.
6. Repeat steps (2)-(5) until a stopping criterion is met.

**FLOW CHART OF PSO**

![Flow chart of PSO](image)
**Block Diagram Of Dc Servo Motor With Pid Controller**: 

![Block Diagram Of Dc Servo Motor With Pid Controller](image)

**Fig4**: Block diagram of servo motor

**STEP RESPONSE OF Z-N METHOD**: 

![Step response of Z-N method](image)

**Fig5**: Step response of Z-N method

**STEP RESPONSE OF GA**

![Step response of GA](image)

**FIG 6**: STEP RESPONSE OF GA

**STEP RESPONSE OF PSO**

![Step response of PSO](image)

**FIG 7**: STEP RESPONSE OF PSO

**COMPARISONS OF ALL WAVE FORMS**

![Comparisons of all Wave form](image)

**Fig8**: Comparisons of all Wave form

### Table3: comparison of all methods

<table>
<thead>
<tr>
<th>parameters</th>
<th>ZN</th>
<th>GA</th>
<th>PSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Settling time(sec)</td>
<td>5.0139</td>
<td>1.6</td>
<td>0.56</td>
</tr>
<tr>
<td>Rise time(Sec)</td>
<td>0.2901</td>
<td>0.25</td>
<td>0.35</td>
</tr>
<tr>
<td>Peak over shoot (%)</td>
<td>61.74</td>
<td>30</td>
<td>3</td>
</tr>
<tr>
<td>Fitness fun value</td>
<td>17.57</td>
<td>0.7225</td>
<td>0.4668</td>
</tr>
</tbody>
</table>

**CONCLUSION**: 

In this paper conventional and soft computing methods for position control of DC servo motor is used. Soft computing techniques to the optimum tuning of PID controllers led to a satisfactory close loop response. By comparing the all methods PSO gives better response in terms of performance indices. The draw backs associated with GAs may have a tendency to converge towards local optima or even arbitrary points rather than the global optimum of the problem is over come in PSO .This work may be extending by using advanced genetic algorithm and also using evolutionary algorithms.
REFERENCES:

8. DC motor angular position control using PID controller for the purpose of controlling the hydraulic pump, “International conference on control, Engineering &information technology CEIT-13proceedings Engineering& Technology-vol.1, pp.22, 26, 2013)”.
10. Tuning of PID Controller Based on Fruit Fly Optimization Algorithm. “Proceedings of 2012 IEEE International Conference on Mechatronics and Automation August 5 - 8, Chengdu, China”.