IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

D.C. Motor Control by Genetic Algorithm

Tushar Chouhan
er.tusharchouhan@gmail.com

Arkaja Agrawal
arkaja.agrawal@yahoo.com

Abstract — **The aim of this paper is to design a position controller of a DC motor by selection of a PID parameters using genetic algorithm. The model of a DC motor is considered as a third order system. PID parameters are obtained using genetic algorithm. Result is compared with conventional Ziegler Nichols Method.**

Keywords — **DC motor, GA, PID, Motor control**.

I. Introduction

Due to its excellent speed control characteristics, the DC motor has been widely used in industry even though its maintenance costs are higher than the induction motor. As a result, position control of DC motor has attracted considerable research and several methods have evolved. Proportional-Integral Derivative (PID) controllers have been widely used for speed and position control of DC motor [1]. The speed of DC motors can be adjusted within wide boundaries so that this provides easy controllability and high performance. DC motors used in many applications such as still rolling mills, electric vehicles, electric cranes and robotic manipulators require speed controllers to perform their tasks. Speed controller of DC motors is carried out by means of voltage control in 1981 firstly by Ward Leonard. The regulated voltage sources used for DC motor speed control have gained more importance after the introduction of thyristor as switching devices in power electronics. Then semiconductor components such as MOSFET, IGBT and GTO have been used as electric switching devices [2].

II. Genetic Algorithm

Genetic Algorithm (GA) is a search heuristic that mimics the process of natural evolution. This heuristic is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. In a genetic algorithm, a population of strings (called chromosomes or the genotype of the genome), which encode candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem, evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

A typical genetic algorithm requires:

1. A genetic representation of the solution domain,
2. A fitness function to evaluate the solution domain.

A standard representation of the solution is as an array of bits. Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in genetic programming and graph-form representations are explored in evolutionary programming; a mix of both linear chromosomes and trees is explored in gene expression programming.

The fitness function is defined over the genetic representation and measures the *quality* of the represented solution. The fitness function is always problem dependent. Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions (usually randomly) and then to improve it through

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

repetitive application of the mutation, crossover, inversion and selection operators [3].

III. Simulation

The system is evaluated by Genetic algorithm toolbox of MATLAB R2009a. The DC motor under study has the following specifications and parameters:

A. Specification

2hp, 230 volts, 8.5 amperes, 1500rpm

B. Parameters

$Ra = 2.45$ ohms
$La = 0.035$ H
$Kb = 1.2$ volts/(rad/sec)
$J = 0.022$ Kg-m^2 /rad
$B = 0.5*10^{-3}$ N-m/(rad/sec)

Based on above parameter the transfer Function of DC Motor:

$$TF = Kb / [J\, La\, S^3 + (Ra\, J+B\, La)\, S^2 + (Kb^2+Ra\, B)\, S]$$

IV. Results

Step response of DC motor controlled by conventional Z-N method and by GA is shown below:

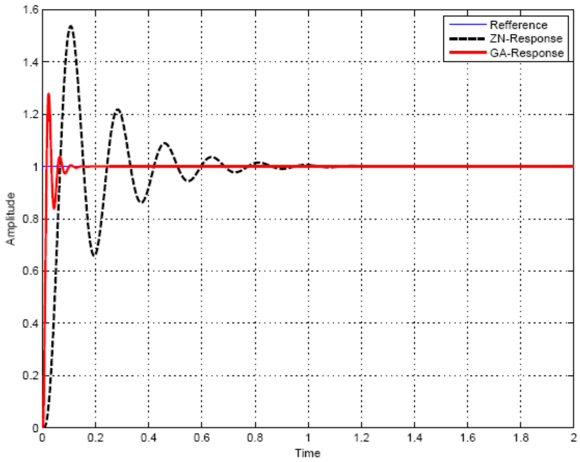

Figure1: Step response of system with control

IV. Conclusions

The designed PID with GA has much faster response than response of the classical method. The classical method is good for giving us as the starting point of what are the PID values. However the GA designed PID is much better in terms of the rise time and the settling time than the conventional method. Finally the genetic algorithm provides much better results compared to the conventional methods. And also the error associated with the genetic based PID is much lesser than the error calculated in the conventional scheme. In this paper, implementation of the genetic algorithm based PID controller for the DC motor position control system is covered.

References

[1] David E. Goldberg, “Genetic Algorithms in Search, Optimization and Machine Learning.” The University of Alabama, Addison-Wesley Publishing Company Inc, 1989.
[2] K Ogata, Modern Control Systems, University of Minnesota, rentice Hall, 1987.
[3] T. O..Mahony, C J Downing and K Fatla, “Genetic Algorithm for PID Parameter Optimization: Minimizing Error Criteria”, Process Control and Instrumentation 2000 26-28 July 2000, University of Stracthclyde, pg 148- 153.
[4] Neenu Thomas, Dr. P. Poongodi, “Position Control of DC Motor Using Genetic Algorithm Based PID Controller”. Proceedings of the World Congress on Engineering 2009 Vol II WCE 2009, July 1 - 3, 2009, London, U.K.