ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Speed Control of DC Motor using Genetic Algorithm.

Jagan Kumar M*, and P Aadaleesan.

SASTRA University, Thanjavur, Tamil Nadu, India.

ABSTRACT

The aim of the project is to design a PID controller for the linear time varying DC motor system. The PID controller is tuned using online using Genetic Algorithm (GA). DC motor drive systems are often used in many applications such as robotics, lathes, drills and electric traction etc. All these applications require good speed control of motors. Speed control of DC drives are generally achieved using Proportional, Integral and Derivative (PID) controller. Tuning of PID controller is highly crucial for the speed control of motor. The PID controllers are tuned mostly according to the system's parameters and mathematical model.In this work Genetic Algorithm (GA), a soft computing tuning method is used to tune the PID parameters, Kp, Ki and Kd, for the regulatory speed control of a linear time varying (LTV) DC motor. The adopted method, GA, is a popular stochastic optimization technique based on the natural process of evolution. When applied to complex engineering problems, GA searches for the optimal solution in a wider space using diversity and crossover operators. This feature of GA distinguishes it from other classical optimization technique. The LTV mathematical model of the DC motor is whose denominator parameters are slowly and continuously changing with time is considered in the present work. A suitable stable PID controller parameters are found using GA. The tuned parameters are used in the controller online so as to control the speed of the DC motor. Simulations results are obtained using MATLAB are found to give satisfactory results.

Keywords: speed control, dc shunt motor, PID controller, Genetic algorithm, armature voltage control and Performance indexes

**Corresponding author*

INTRODUCTION

The speed control of DC motor is very important in all of its applications. In this proposed work, Genetic algorithm (GA) based PID controller is used to regulate the speed of the DC motor. The denominator of transfer function's coefficient is constantly changing according to the time. So the GA technique is used to optimize the PID controller for each of the iterations. In each iteration the GA is performed and based on the GA the PID controller get its gain value. According to the gain value obtained the PID controller performs its controlling action for the control of speed in the DC motor. Performance index criteria is used to here to create the objective function for the DC motor model. The objective function is taken for GA optimization. ITAE, IAE, MSE and ISE is performed in the following work for the optimization of gain values of PID controller to control the speed of the DC motor.

SPEED CONTROL OF DC MOTOR

DC Motor is most commonly used electrical device in all places. Speed control is very crucial step in DC motor. It is easy to do so. But getting optimal valued speed is such a difficult process. The speed control of DC motor can be done using many methods. But in this project the armature control is used. To perform armature control in DC motor the transfer function for the control have to be derived at first. In armature control of the DC motor the voltage which is applied to the armature of the motor is adjusted without applying any change to field voltage.

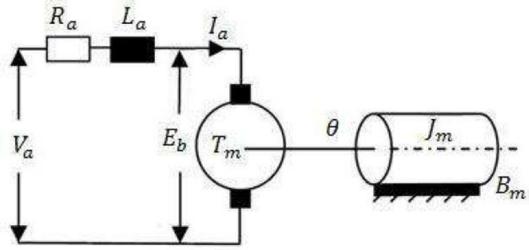


Figure 1: DC Motor Model

Some relations are given below,

$$T_m(t) - T_L(t) = J_m \frac{d\omega(t)}{dt} + B_m \omega(t) \qquad (1)$$

Figure 3.2 showing the basic block diagram of DC motor model including their transfer functions. Va is the input supply, TL is load torque and   is angular speed.

The speed control can be done using the transfer function of the motor model which is created using the above relation. The transfer function of ω(s) with respect to V_a(s) is given below. So relation of speed and voltage is given by the transfer function,

$$\frac{\omega(s)}{T_L(s)} = \frac{K_t}{L_a J_m s^2 + (R_a J_m + L_a B_m)s \ + (R_a B_m + K_b K_t)} \qquad (2)$$

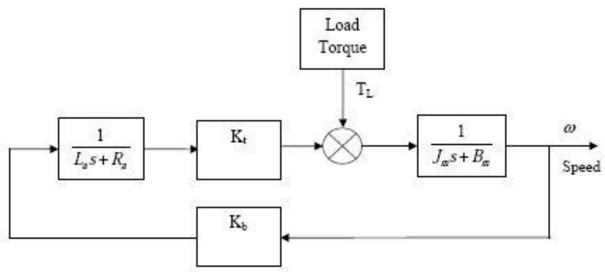


Figure 2: Block diagram of DC Motor Model when V_a = 0

GENETIC ALGORITM

Genetic algorithm falls under the stochastic global search method which mimics the natural evolution. It begins with or without the solution which is correct belong to environment. It also does or doesn't have the evolution operators which is operated to obtain the best solution. It can be started in several points and can be searched in parallel, local minima and the sub optimal points in the solution will be avoided by GA. Because of this GA has the high performance area in complex domains without any difficulty were the other techniques have. GA have to be initialized with the randomly generated population of desired population sites. This is also called as the mating pool it can be a binary number or real valued number. Which is also called as the chromosomes. The objective function which is created by the performance indices can be checked by the gain value obtained from the population.

Reproduction

Reproduction is the process where the fitness value of each chromosome is assessed. The value is biased towards fittest or stronger individuals. This is the process happen in the natural evolution, where a chromosome which is fitter has been selected for reproduction.

There are many methods available for selection process in reproduction, they are

- Roulette wheel selection
- Stochastic universal sampling
- Normalized geometric selection
- Tournament selection

But the most common method is roulette wheel selection. A pointer is spun and the probability of individual being selected is related to its fitness fig 3 represents roulette wheel selection method. Many number of same strings are selected for reproduction and which is fitter will dominate the others.

Crossover

After the selection process the next step is cross over. Which swaps some parts of selected strings to capture the best data of the chromosomes and create new best ones. If the probability of the off spring is 0% then it is the exact replica.

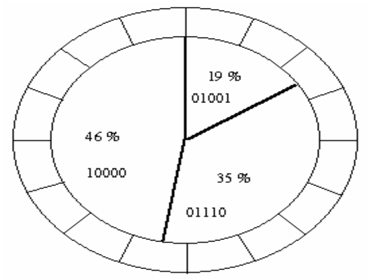


Figure 3: Depiction of roulette wheel selection

If probability is 100% then the offspring doesn't match the parents. There is also some methods in crossover like

- Single point crossover
- Multipoint crossover
- Uniform crossover

In single point crossover members of new string in the pool are paired. To the each pair crossover technique is done.

Figure 4: Illustration of single point crossover

There are also a multi-point crossover which is extended version of the single point cross over. In multi-point crossover new offspring in mating pool are mated in random and multiple point is selected with no repetition. Then the bits are changed successively.

Figure 5: Illustration of a Multi-point crossover

In uniform crossover mask is created and placed over the string. If the mask bit is 0 then the bit is changed and if the bit is 1 the same bit is placed underneath.

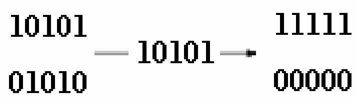


Figure 5: Illustration of a Uniform Crossover

Mutation

Mutation is the process which is used to regain the data which is loosed in the selection and crossover process. The probability of the mutation should be low in between 0.1% to 0.01%. In mutation random bit is selected and the bit is changed to other.

Figure 6: Illustration of mutation operation

Genetic algorithm process

For genetic algorithm the steps have to be followed are

- Random, initial population of individuals of fixed size are generated.
- Fitness is evaluated for each individuals
- Fittest member of the population is selected
- Reproduction is done by probabilistic method (e.g., roulette wheel).

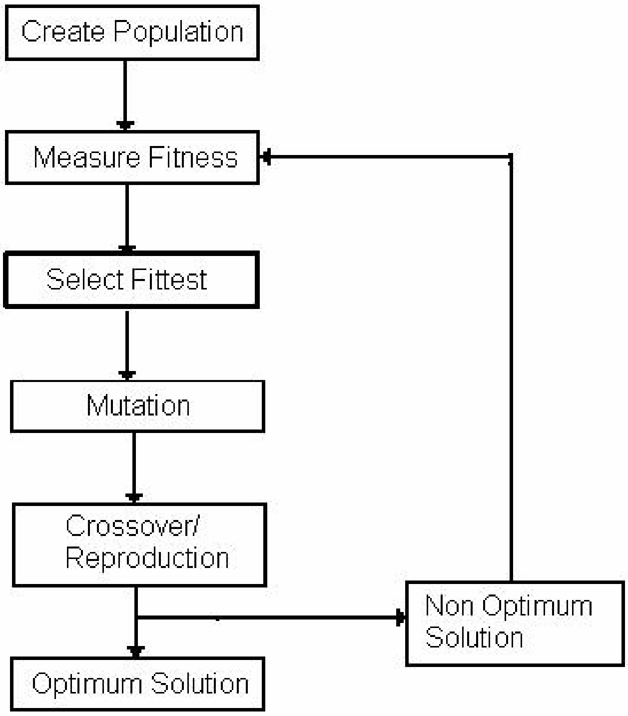


Figure 7: Flow chart of the genetic algorithm technique

- Crossover is implemented over the reproduced chromosomes (Crossover sites and the mates are chosen probabilistically).
- Mutation operation is executed with very low probability say 0.01-0.01%.
- Iterate the process again and again until the predefined convergence criteria is achieved

PERFORMANCE INDICES

In this project based on the performance index objective function is generated for the algorithm and implemented in the controller.

For the good and optimized performance of the controller which is based on PID performance index is chosen. The gains are adjusted to make the optimized performance for best controller.

Integral of time multiplied by absolute error(ITAE)

$$I_{ITAE} = \int_0^T t|e(t)|dt \tag{3}$$

The extra weight is given to the error to some extent is given by the ITAE function.

Integral Absolute Magnitude of the error (IAE)

$$I_{ISE} = \int_0^T |e(t)|dt \tag{4}$$

Negative error components are removed by the IAE factor.

Integral of the square error (ISE)

$$I_{ISE} = \int_0^T e^2(t)dt \tag{5}$$

Squaring of the error will remove the negative error component which is done by ISE. Over and under damped components are distinguished by ISE.

Mean square error (MSE)

$$I_{MSE} = \frac{1}{n}\sum_{i-1}^{n}(e(t))^2 \qquad (6)$$

It reflects each variation and deviation from the target value.

RESULTS

The controller output for the LTV system is obtained using various parameters of the GA. The comparison is shown below.

Results obtained from ITAE

Iteration	K_d	K_p	K_i	Rise time(ms)	Settling time(ms)	overshoot	Peak time
1	7.82657	25.56480	31.1354	0.258	0.2811	41.0728	0.0650
2	7.81474	26.66068	30.0269	0.0248	0.3382	0.7546	0.0633
3	8.50127	27.70416	31.1403	0.0241	0.3980	53.1843	0.0664
4	9.20112	34.19715	40.0867	0.0241	0.3980	53.1843	0.0664
5	8.85094	27.7994	31.8226	0.0256	0.4235	52.5820	0.0703
6	10.52379	32.11566	35.2603	0.0241	0.4090	57.1415	0.0650
7	16.09126	43.47338	47.8202	0.0245	0.5393	61.0160	0.0636

Table 1: PID gains and time domain characteristics for ITAE

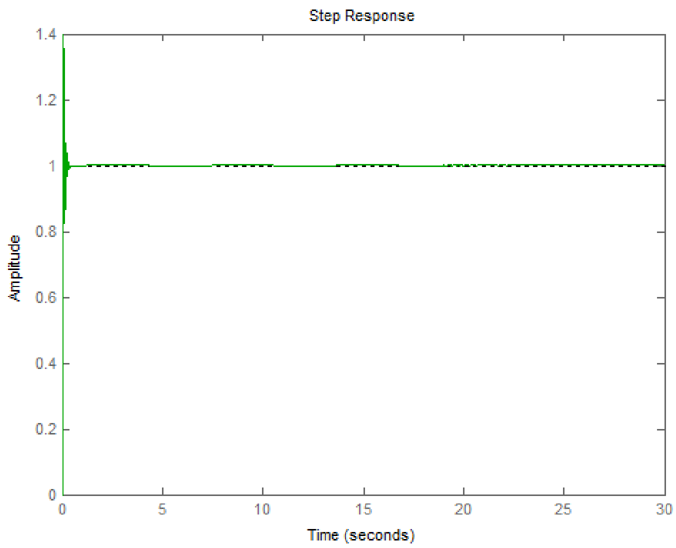


Figure 8: System's response using ITAE in 1st iteration

The Figure 8 gives the system's response with the PID controller with the gain values below in the 1st iteration.

ISSN: 0975-8585

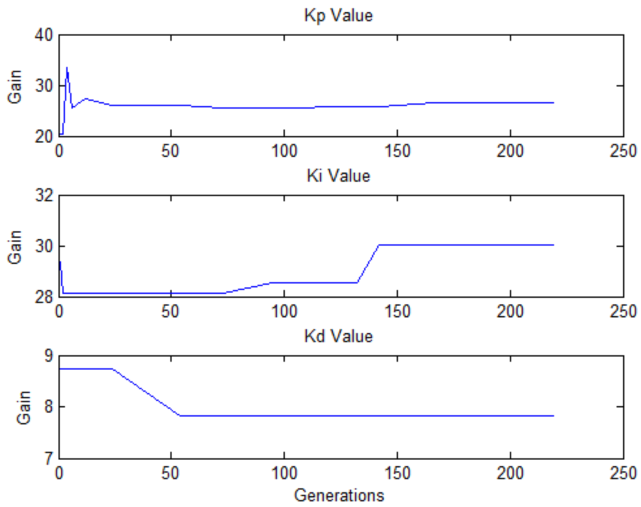


Figure 9: Optimized gain values in each generation in 1st iteration

Kp=25.56480, Kd=31.1354 and Ki=7.826 were obtained as gain values

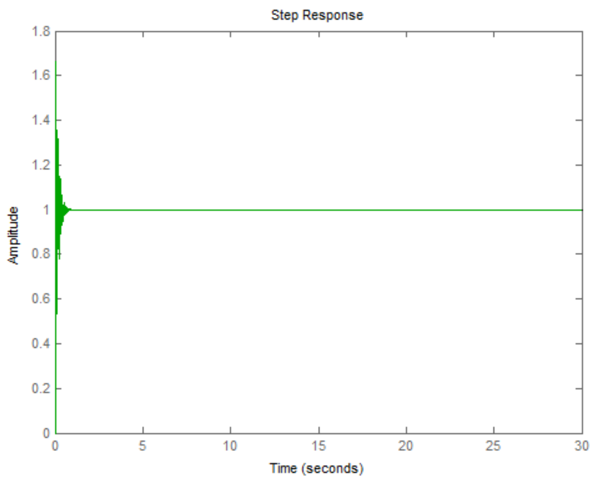


Figure 10: Controller output of ITAE in 10th iteration

The Fig 10 gives the system's response with the PID controller with the gain values below in the 10th iteration.

Results obtained from IAE

Iteration no.	k_d	k_p	k_i	Rise time	Settling time	Overshoot	Peak time
1	18.90382	54.53915	67.1126	0.01446	0.2792	57.8478	0.0405
2	64.17197	99.68197	98.4764	0.0078	0.3102	74.7736	0.0220
3	42.67227	83.34215	99.4279	0.0101	0.3431	71.5793	0.0281
4	9.75492	26.22301	33.6230	0.0236	0.3877	51.7323	0.0630
5	25.80446	50.76503	68.0011	0.0143	0.4014	67.2996	0.0391
6	10.35318	22.81930	28.9489	0.0244	0.4120	54.2034	0.0671
7	56.21156	71.21679	99.9765	0.0100	0.4574	77.4039	0.0283
8	59.82150	73.35650	99.8210	0.0100	0.4610	78.6367	1.7864
9	63.10628	71.07387	99.8475	0.0100	0.5146	79.6125	0.0284
10	34.47854	45.53605	84.7382	0.0141	0.5566	74.2297	0.0394

Table 2: PID gains and time domain characteristics for IAE

The Fig 11 gives the system's response with the PID controller with the gain values below in the 1st iteration.

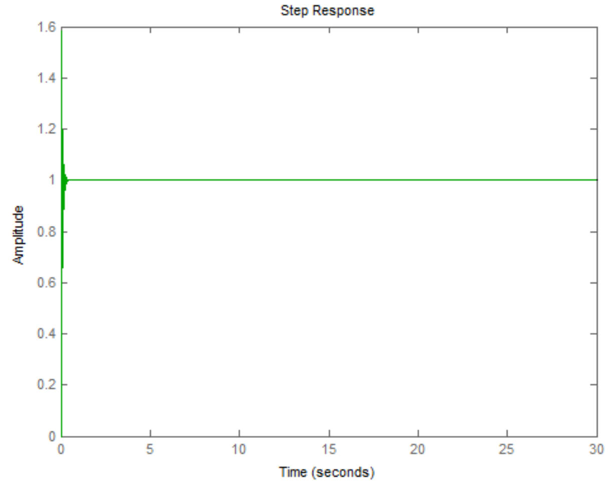


Figure 11: Controller output of IAE in 1st iteration

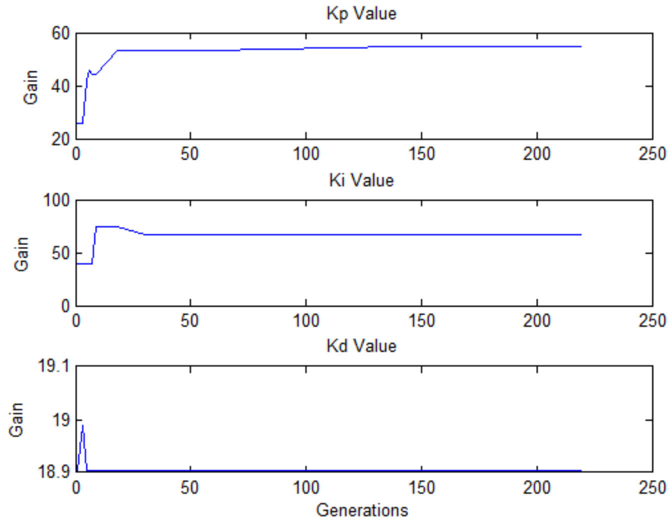


Figure 12: Optimized gain values in each generation in 1st iteration of IAE

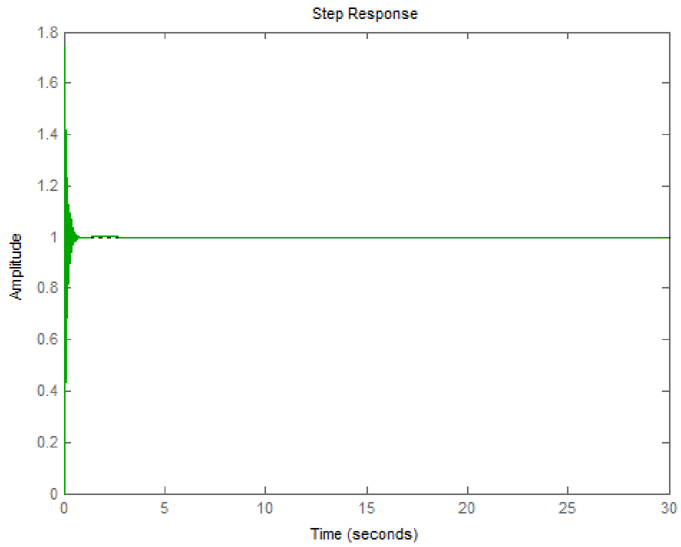


Figure 13: controller output of IAE in 10th iteration

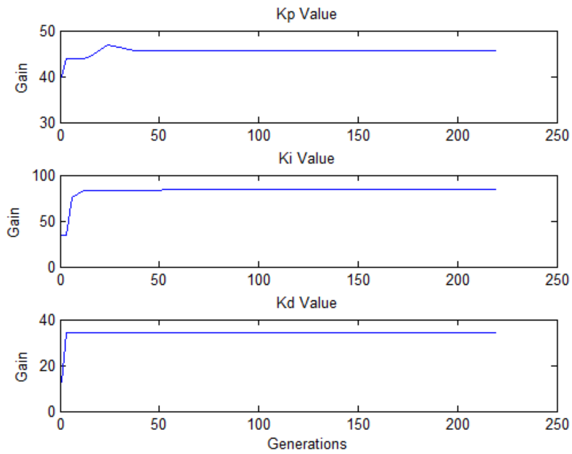


Figure 14: optimized gain values in each generation in 10th iteration of IAE

Fig 12 gives Optimized gain values in each generation in 1st iteration of IAE 1st iteration Kp=54.53915, Ki=67.1126 and Kd=18.90382 were obtained as gain values.

The Fig 13 gives the system's response with the PID controller with the gain values below in the 10th iteration.

Fig 14 gives Optimized gain values in each generation in 10th iteration of IAE in 10th iteration Kp=45.53605, Ki=84.7382 and Kd=34.47854 were obtained as gain value.

Results obtained from ISE

The Fig 15 gives the system's response with the PID controller with the gain values below in the 1st iteration.

Fig 16 gives Optimized gain values in each generation in 1st iteration of IAE 1st iteration Kp=23.0897, Ki=92.8278 and Kd=35.81057 were obtained as gain values.

The Fig 17 gives the system's response with the PID controller with the gain values below in the 10th iteration.

Fig 14 gives Optimized gain values in each generation in 10th iteration of IAE in 10th iteration Kp=5.61819, Ki=38.7938 and Kd=13.28641 were obtained as gain values were obtained as gain value.

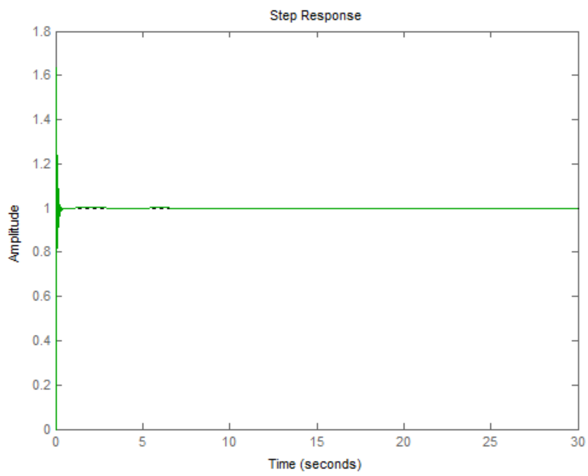


Figure 15: Controller output of ISE in 1st iteration

ISSN: 0975-8585

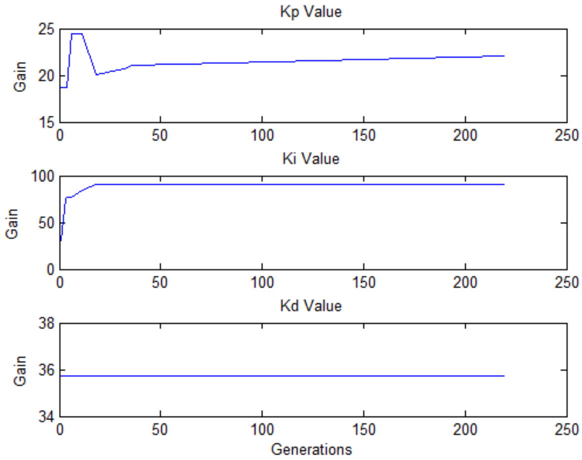


Figure 16: Optimized gain values in each generation in 1st iteration of ISE

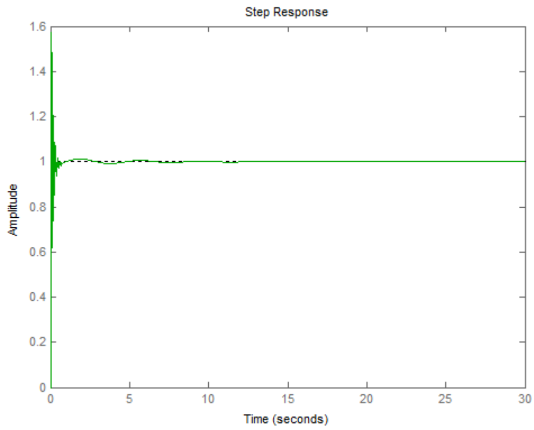
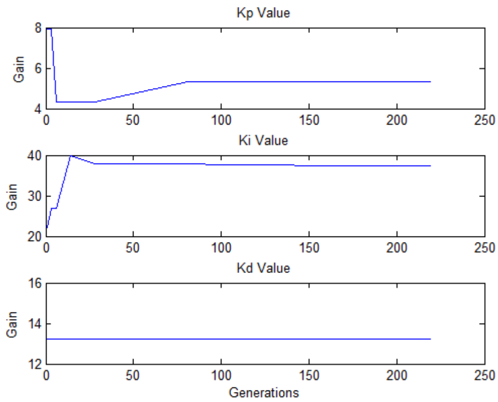


Figure 17: Controller output of ISE in 10th iteration



Figures 18: optimized gain values in each generation in 10th iteration of ISE

The armature voltage speed control method is for a DC shunt motor is studied in the present work. The DC motor modeled as Linear Time Variant (LTV) system is used for scrutinizing the system for first 10 instants of time. To gain a good speed control over the DC motor, the classical PID controller is employed. Genetic Algorithm, being the most predominant optimization technique to obtain global optima is utilized to adapt the PID controller values such that efficient speed control is achieved. Also, the performance indexes such as ITAE, IAE, ISE and MSE are used to analyze the efficiency of the closed loop system with the application of genetic algorithm to adjust the controller parameters. Each of these performance indexes has their own merits and demerits. GA has given best results with individual performance indexes and found that the proportional, integral and derivative gains are optimized to their fullest extent, resulting in a better speed control of DC motor for the given armature voltage.

ACKNOWLEDGMENT

I express my gratitude to our respected Vice Chancellor Prof. R. Sethuraman, SASTRA University for having given me an opportunity to undergo the M.Tech course in Instrumentation & Control at SASTRA University and carry out this project work.

I thank Dr. G.Bhalachandran, Registrar, SASTRA University for granting me the permission and extending facilities to carry out this project.

I express my sincere thanks and gratitude to Dr. B.Viswanathan, Dean, School of Electrical & Electronics Engineering, SASTRA University, Thanjavur for his constant support during constant support during project work.

I record my sincere thanks to Dr. S.Jayalalitha, Associate Dean, Electronics & Instrumentation Engineering Department, for her support in the accomplishment of this project work. With deep sense of gratitude, I express my sincere and heartfelt thanks to my guide Dr.P.Aadaleesan, Senior Assistant Professor, Department of Electronics & Instrumentation, School of Electrical & Electronics Engineering, SASTRA University, Thanjavur for his innovative ideas and continued encouragement for successful completion of this project My sincere thanks to our project coordinator Dr. K.Ramkumar, Associate Professor, Electronics & Instrumentation Engineering Department for his valuable suggestions that helped me to complete this project.

I extend my sincere thanks to my parents and friends for their moral support and the almighty for his manifold mercies in carrying out this project successfully.

REFERENCES

[1] Megha Jaiswal, Mohna Phadnis. IJARCSSE 2013;3:247-253.
[2] Bindu R, Mini K Namboothiripad. IJETAE 2012;2:310-314.
[3] Tushar Chouhan, Arkaja Agrawal. IJDACR 2012;1.
[4] Ayman A Aly. Intelligent Control and Automation 2011;2:69-76.
[5] Atulit Patel, Kapil Parikh. IOSR Journal of Electrical and Electronics Engineering 2014;9:4-8.
[6] Roberto Gutierrez-Guerra, Rodolfo Murrieta-Due˜nas, Jazm´ın Cort´ez-Gonz´alez, Arturo Hern´andez-Aguirre and JG Segovia-Hern´andez IEEE Congress on Evolutionary Computation, 2013;2267-2274.