Modified Interactive Evolutionary Computing for Speed Control of an Electric DC Motor

M. B. Anandaraju
Professor, Dept of ECE,
BGSIT B.G.Nagara, Mandya, Dist

P. S. Puttaswamy
Professor, Dept of E&E,
PESCE, Mandya

ABSTRACT

DC motors are important components of most of the process control industries. PID controllers are extensively used in DC motors for speed as well as position control. Tuning of PID controller parameters is an iterative process and needs an optimization to achieve the desired performance. In this paper a modified form of Interactive Evolutionary Computing (IEC) is used as the tool for achieving optimization of PID controller parameters for the speed control of DC motor. Different error models are used for optimization and a comparison with Genetic Algorithm based approach is presented.

Keywords
DC motor, PID, IEC, GA, optimization

1. INTRODUCTION

Mechanical applications based on electric motors demand high speed accuracies [1]. Drive design for such systems is a complicated and process defines the number of parameter optimizations involved. Many modern control methodologies like adaptive control [2], nonlinear control [3], optimal control [4] have been studied and attempted to deal with different types of motor controls. These approaches are mathematically complex and complicated to implement. Motor control based on proportional, integral and derivative (PID) controllers has been largely studied and implemented with success. The present work deals with controlling of time domain performance parameters of a DC motor to achieve speed control. Determination of the controller gains which maximizes or minimizes the performance specification is a complex, constrained optimization problem.

Modified form of Interactive Evolutionary Computing (IEC) is used as the optimization approach to minimize the errors in the time domain performance s-parameters of DC motor.

Modeling and simulation studies are carried out using MATLAB and SIMULINK.

2. DC MOTOR MODEL

2.1 Electrical Equivalent Circuit

Electric DC motor coverts electrical energy into mechanical energy based on performance of electromagnetic induction. To achieve velocity control for the DC motor, the primary requirement is to model the DC motor system itself, in control system terminology it is called as plant modeling. The equivalent circuit of DC motor is shown in Fig1.
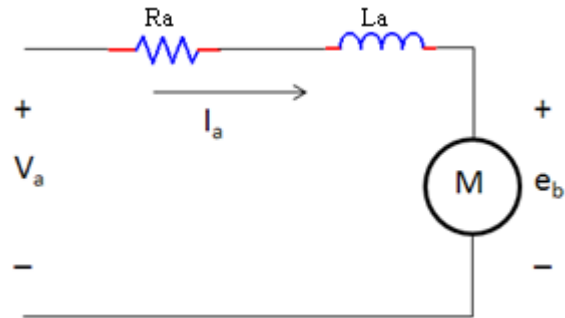

Fig 1: DC motor armature equivalent circuit

R_a, L_a and e_b represent the armature resistance, inductance and the back-emf, respectively. The motor torque T is related to the armature current, i_a by torque constant K

$$T = Ki$$

The back emf e_b, is related to angular velocity by;

$$e_b = K\omega = K\frac{d\theta}{dt}$$

In mechanical domain, if J is the rotational moment of inertia of motor, ω the motor angular velocity, $(T_m - T_l)$ the net torque with τ_m the electromagnetic torque and τ_l load torque,

$$J\frac{d\omega}{dt} = (T_m - T_l) = Ki$$

Combining Newton's law and Kirchhoff's law we can write:

$$J\frac{d^2\theta}{dt^2} = Ki$$
$$L\frac{di}{dt} + Ri = V - K\frac{d\theta}{dt}$$

From the above equations, we can derive the s-domain transfer function of the motor as:

$$Js\,\omega(s) = KI(s)$$
$$LsI(s) + RI(s) = V(s) - K\omega(s)$$

Combining the two equations:

$$\frac{\omega(s)}{V(s)} = \frac{\frac{K}{Js}\cdot\frac{1/R_a}{\left(1+s^{(L_a}/R_a)\right)}}{1+\frac{K^2}{Js}\cdot\frac{1/R_a}{\left(1+s^{(L_a}/R_a)\right)}}$$

The s-domain model of the DC motor derived can be represented as shown in figure2.
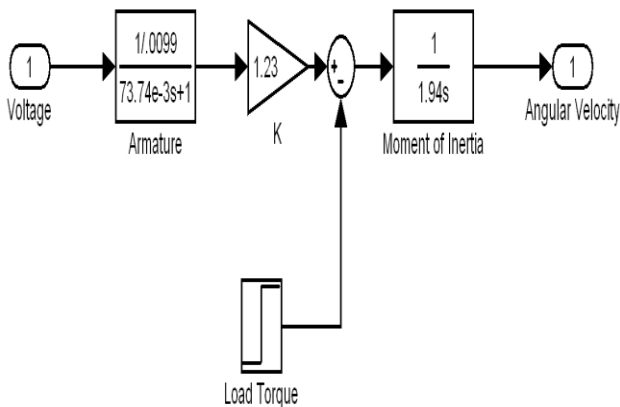
Fig 2: DC Motor SIMULINK Model

3. INTERACTIVE EVOLUTIONARY COMPUTING

Interactive evolutionary computing is a technique from the class of Evolutionary algorithms (EA) whose fitness function is replaced by a human. A human plays the role of the fitness function and selects one or more individual(s) which survive(s) and reproduces to constitute a new generation. Interactive Evolutionary Computing (IEC) takes the advantage of EA and human knowledge and intuition in selecting the individual(s). Optimization process using human intervention has presented in [5].

3.1 Evolution Strategy

Search procedures which mimic the natural evolution of the species in the natural system are coined as Evolutionary strategies [6]. Like Genetic algorithm (GA), require data based on objective function and constraints, and not the derivatives or other auxiliary knowledge. Evolution Strategies (ESs) were developed by [7] and discussed the selection, mutation and a population of size one.

3.2 Selection

For a given generation, parents and offsprings are generated by recombination and mutation. They are sorted according to the user defined criteria and the user himself selects the best to make parents of the next generation.

3.2 Reproduction

Next set of population for the successive generation is generated by a process called reproduction and other involves crossover (recombination) and mutation. The result is a new set of population derived from the selected solutions of the previous population. Generally the average fitness of the population is increased as compared to the population of the previous population.

3.3 Termination

The process of optimization is halted once a required criterion is achieved. The termination condition can be either the number of generations or the solution satisfying an optimum criterion.

3.4 Modification to IEC

As in automated evolutionary optimization algorithms, (IEC) cannot use many individuals and cannot be iterated over many generations. This limitation is because the decision maker is the human and due to human fatigue, the process cannot be iterated over many generations. Hence, EA developed efficiently searches with a few individuals and few generations. Moreover, the selection of individuals done by a human is also incorporated in the software by selecting the best four individuals from the population to participate in the next generation. The fitness function results from a generation are sorted out and the four individuals with the least value of the fitness function are allowed to participate in the next generation. This overcomes the limitation of the human fatigue and iterations over a few more generations can be performed and much better optimizations can be achieved.

4. MATLAB MODELING

4.1 Motor model

Motor model is made using SIMULINK/MATLAB. The parameters for the PID controller are the individuals in the IEC population. For each individual, the SIMULINK model is run and the error is computed. Four individuals with the minimum errors are chosen for representing the next generation population. Different error models are used for the evaluation purpose. The feedback loop consists of a current sensor in the current loop (inner loop) and the tacho-generator in the velocity loop (outer loop). Generally the reference voltage to the motor V_a is fed through a PWM controller. The PWM controller consists of a set of comparators that compares a reference signal with the tacho-feedback signal and generate PWM pulses for an H-bridge converter. The H-bridge converter has been modeled as a single pole function as shown in Fig4.
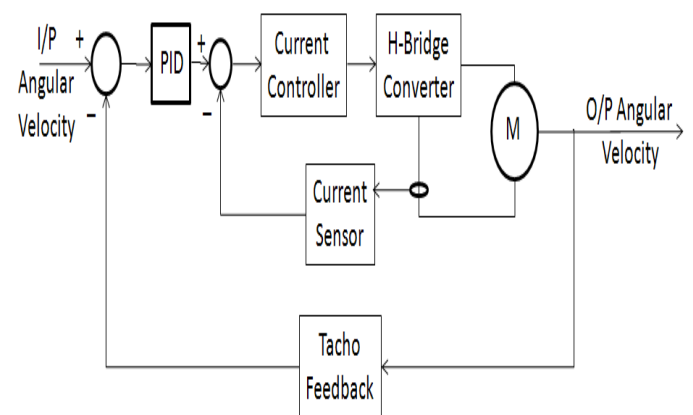

Fig3: Block diagram for motor speed control

Motor parameters used in the modeling are listed in Table1.

Parameter	Value
Armature Resistance (R_a)	0.0099Ω
Armature Inductance (L_a)	mH
Motor Inertia (J)	1.94 kg-m^2
K	1.23 V/rad/s
Tacho DC gain	00273 V/rad/s
PWM frequency	1kHz

Table1: DC Motor Parameters

4.2 PID Controller

PID controller transfer function used in the model is shown below:

$$\frac{K_D s^2 + K_P s + K_i}{s(s + \omega_d)}$$

PID controller takes the error input and computes the proportional, derivative and integral of the error. The derivative component improves the system response, whereas the integral and proportional components shape the steady state performance.

4.3 Optimization paramter

Initial population of the parameters used in the simulation is listed in Table 2.

Initial Population Range	
Kp	[0 10]
Kd	[0 10]
Ki	[0 100]
Tau	[1 50]
Maximum Generations	100
Fitness function	Various error functions

Table2: IEC parameters

Different error functions, (ITSE) Integral Time Squared Error, (ITAE) Integral Time Absolute Error, Peak Overshoot, Absolute Error and Squared Error are computed at every generation and four individuals with the minimum value of the error are selected to participate for forming the next generation. The definitions of different error functions used are given below:

$$ITSE = \int_0^T |output - command|^2 \times t \, dt$$

$$ITAE = \int_0^T |output - command| \times t \, dt$$

$$Absolute\ Error = \frac{|\max(output) - command|}{command} \times 100$$

$$Squared\ Error = (\max(output) - command)^2$$

4.4 Simulation Model

The complete simulation model is built in SIMULINK and shown in Figure4.

Table3 lists the transfer functions of the various blocks in the model.

Motor	$\frac{\omega(s)}{V(s)} = \frac{G(s)}{1+G(s)H(s)}$ where $G(s) = \frac{K}{Js} \cdot \frac{1/R_a}{\left(1 + s L_a/R_a\right)} ; H(s) = K$
Current Sensor	$\frac{5/523}{(1 + 0.8 \times 10^{-3}s)}$
Tacho Generator	$\frac{5/183.26}{(1 + 25 \times 10^{-3}s)}$
H-Bridge Converter	$\frac{240/5}{(1 + 0.5 \times 10^{-3}s)}$

Current Controller	$\frac{6.222 \times 10^{-3}s + 0.61}{(73.74 \times 10^{-3}s + 1)}$
PID Controller	$\frac{K_D s^2 + K_P s + K_i}{s(s + \omega_d)}$

Table3: Subsystem transfer functions

4.5 Comparative Study with Genetic Algorithm

Speed control of electric DC motor using GA was demonstrated in [8]. A comparative study of optimization using modified IEC and GA has been carried out in the present work. The modified IEC targeted to achieve the fitness function values achieved in [8] in lesser number of generations.

5. RESULTS

The different optimization functions and the resulting PID controller transfer functions are listed below. The results for PID and GA parameters for different error models are complied and are presented in Table 4. Figure6-9 shows the time domain response of the DC motor for different optimizations.

A. ITSE (Integral Time Square Error)

$$PID = \frac{4.195s^2 + 4.576s + 0.397}{s(s + 1)}$$

B. ITAE (Integral Time Absolute Error)

$$PID = \frac{9.206s^2 + 9.811s + 5.943}{s(s + 1)}$$

C. Absolute Error

$$PID = \frac{2.823}{(s + 3.147)}$$

D. Squared Error

$$PID = \frac{6.519s + 7.180}{(s + 1)}$$

Parameter	K_d	K_p	K_i	ω_d	Generations
ITSE	4.195	4.576	0.397	1	29
ITAE	9.206	9.811	5.943	1	16
Abs Error	0	2.823	0	3.147	10
Squared Error	6.519	7.180	0	1	3

Table4: Compiled Results for modified IEC

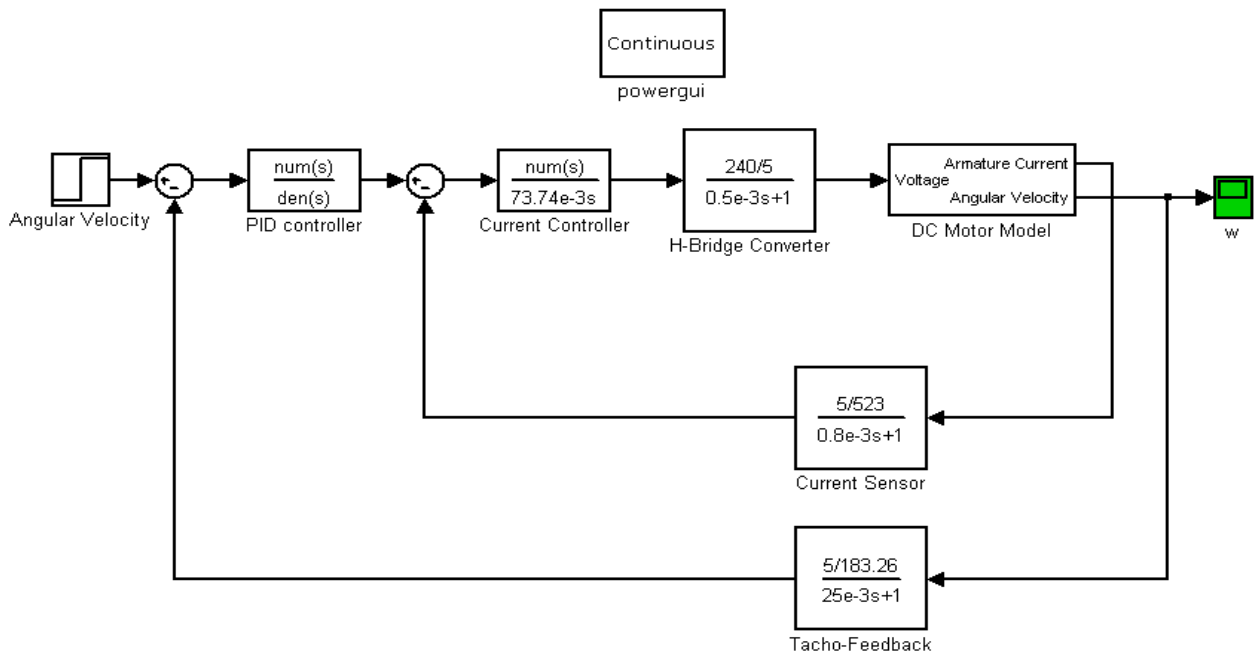| Parameter	Number of generation	
	IEC	GA [8]
ITSE	29	62
ITAE	16	67
Abs Error	10	24
Squared Error	3	35

Table5: Comparison with GA approach

Fig4: SIMULINK model of complete system
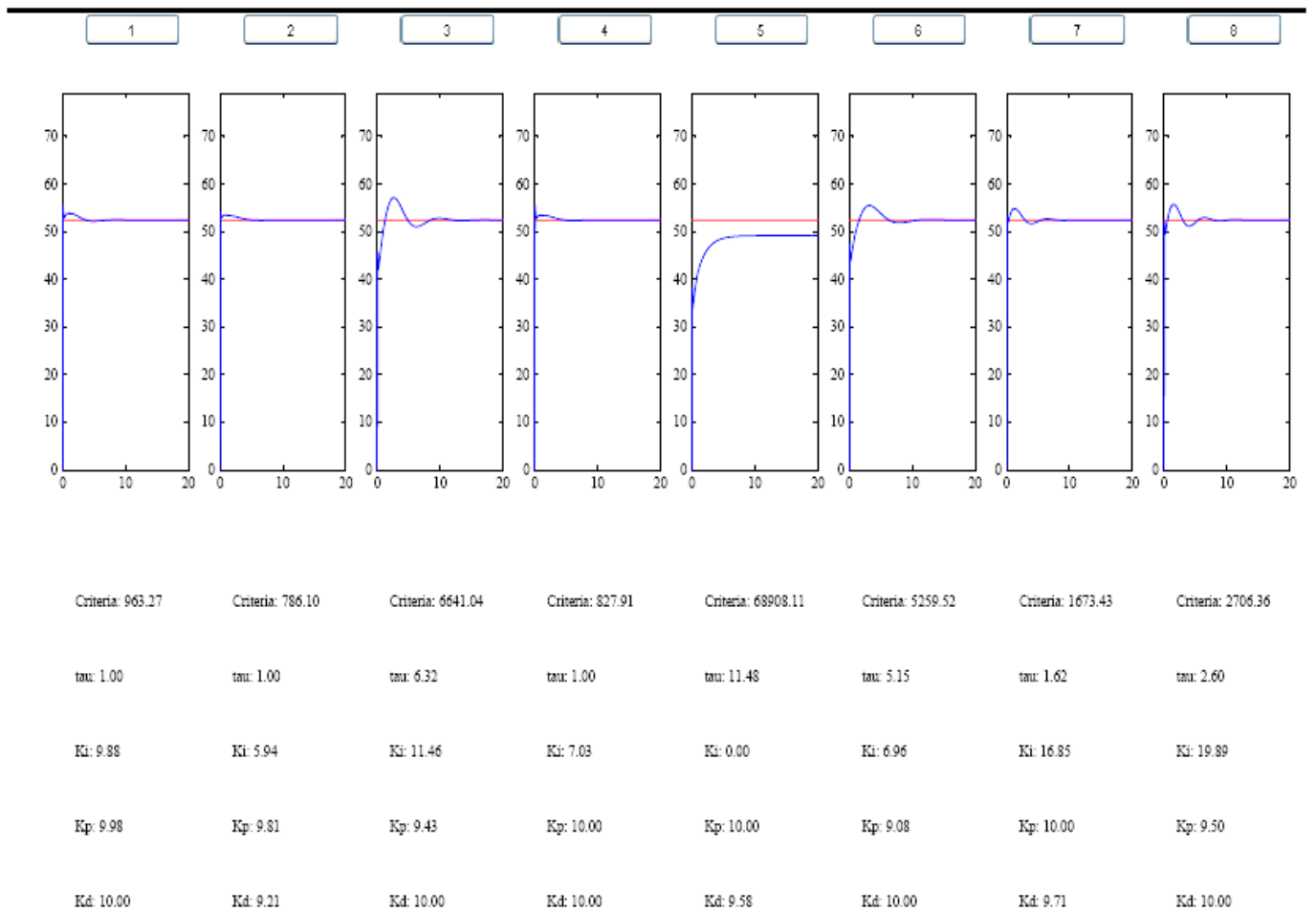

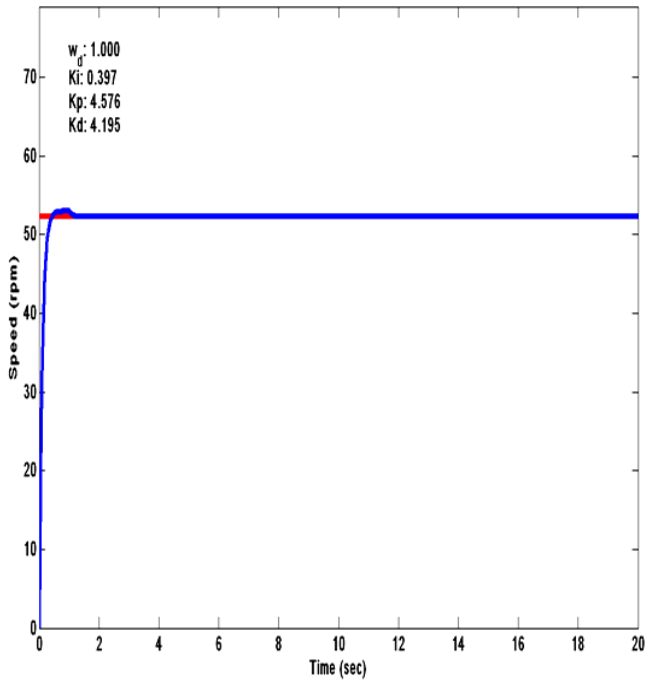**Fig5: Human- Machine interface for IEC algorithm**

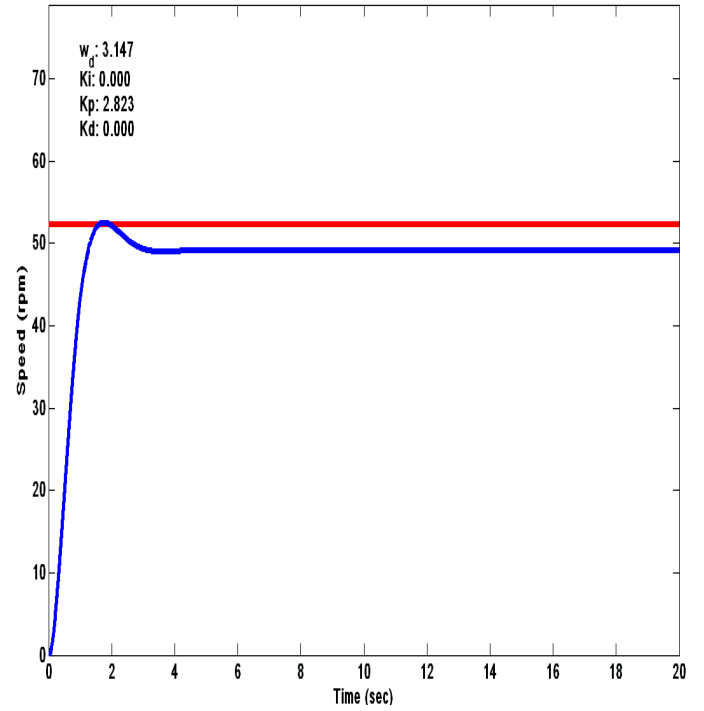**Fig6: Motor Speed optimized for ITSE**

Fig8: Motor Speed optimized for Absolute Error
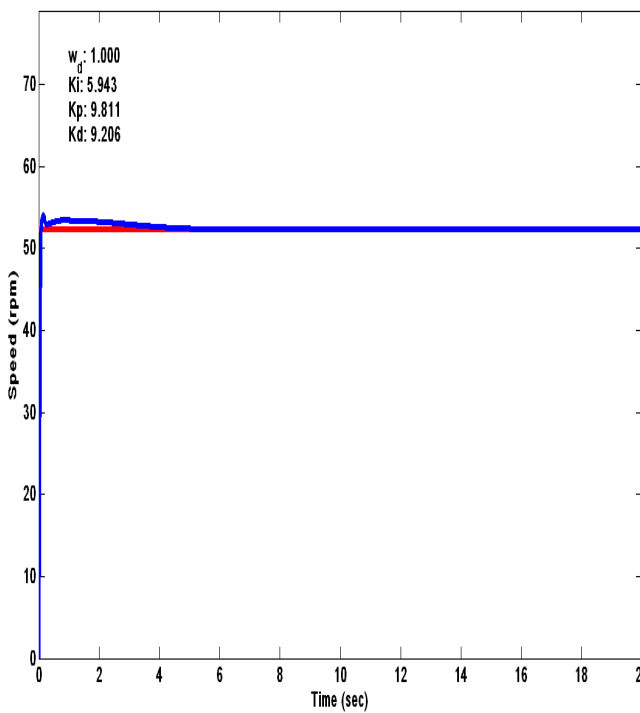

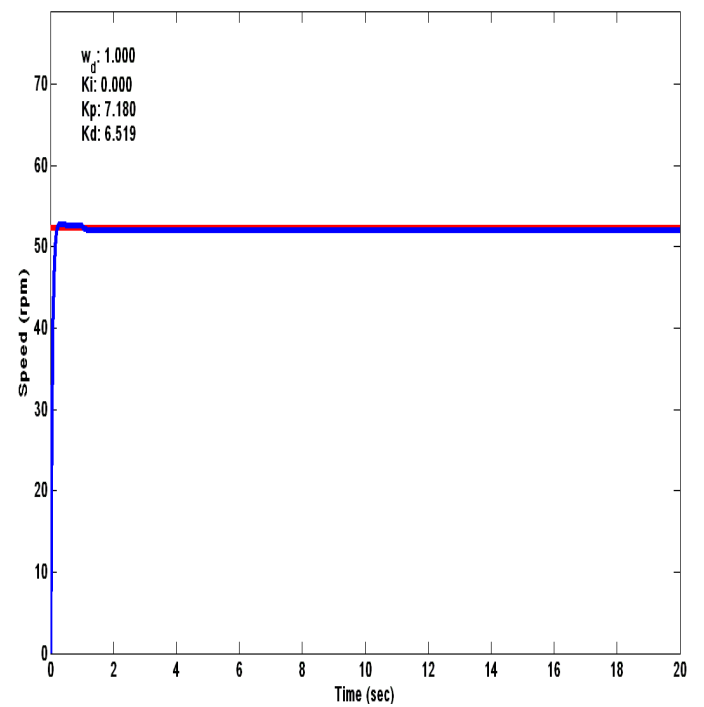**Fig7: Motor Speed optimized for ITAE**

Fig9: Motor Speed optimized for Squared Error

6. CONCLUSION & FUTURE WORK

Interactive Evolutionary Computing approach is used for optimizing the PID controller parameters for the speed control loop of a DC motor. Major limitation of IEC due to the human intervention was overcome by automatic selection of individuals to form the population of next generation.

Different error models used in [8] are considered for optimization and the PID controller parameters are evaluated.

It is found that the modified IEC capability in the minimization criteria to same as with GA with lesser number of generations of iteration. Hence IEC can be presented to achieve faster approach for optimization as compared to GA.

The optimization can be further improved by incorporating a weighted mean of different performance parameters as the return value of fitness function.

7. ACKNOWLEDGMENT

8. REFERENCES

[1] Boldea and S.A. Nasar. *Linear electric actuators and generators*. Cambridge university press,1997

[2] E. Cerruto, A. Consoli, A. Raciti, and A. Testa. A robust adaptive controller for PM motor drives in robotic applications. *Power Electronics, IEEE Transactions on,* 10(1):62–71, 1995.

[3] N. Hemati, J.S. Thorp, and M.C. Leu. Robust nonlinear control of brushless DC motors for direct drive robotic applications. Industrial Electronics, IEEE Transactions on, 37(6):460–468, 1990.

[4] PM Pelczewski and U.H. Kunz. The optimal control of a constrained drive system with brushless dc motor. Industrial Electronics, IEEE Transactions on, 37(5):342–348, 1990.

[5] I. Parmee, D. Cvetkovic, C. Bonham, and I. Packham. Intoducing prototype interactive evolutionary systems for ill-defined multi-objective design environments. *Advances in Engineering Software*, 32:429–441, 2001

[6] H.P. Schwefel. Kybernetische evolution als strategie der experimentellen forschung in der tromungstechnik. 1965

[7] I. Rechenberg. Case studies in evolutionary experimentation and computation. *Comput. Methods Appl. Mech. Engrg.*, 186:125–140, 2000

[8] M B Anandaraju, Dr. P S Puttaswamy and Jaswant Singh Rajpurohit. Article: Genetic Algorithm: An approach to Velocity Control of an Electric DC Motor. *International Journal of Computer Applications* 26(1):37-43, July 2011.