

Speed Control of a Real Time D.C. Shunt Motor Using SA Based Tuning of a PID Controller

S.M.Girirajkumar
Assistant Professor
SEEE, SASTRA University
Thanjavur-613402,India

Atal.A.Kumar
Student
Mechatronics,SASTRA University
Thanjavur-613402,India

N.Anantharaman
Professor
NIT
Tiruchirapalli-620015,India

ABSTRACT:

The proposed work deals with optimal tuning of a Proportional-Integral-Derivative (PID) controller for speed control of a DC shunt motor. PID controllers are widely used in industrial plants because of their simplicity and robustness. Industrial processes are subjected to variation in parameters and parameter perturbations, which when significant makes the system unstable. So the control engineers are on look for automatic tuning procedures. The performance of Ziegler-Nichols method, one of the widely accepted conventional methods has been compared and analyzed with the intelligent tuning technique called the Simulated Annealing method (SA). The results establishes that tuning the PID controller using SA technique which comes under evolutionary programming has proved its excellence in giving better results by improving the steady state characteristics and performance indices.

Keywords: PID, Robust, Conventional techniques, SA, evolutionary programming, D.C shunt motor.

1. INTRODUCTION

Most industrial process are controlled by PID [4, 24]. The success of PID controllers is mainly due to their functional simplicity, reliability and cost effectiveness. The basic function of the controller is to execute an algorithm based on input and hence to maintain the output at or around the set point. A PID controller improves the transient response of a system by reducing the overshoot and by shortening the settling time of a system [22]. Although the number of parameters to adjust a PID controller is very small a great deal of tuning rules can be found in literature [16]. The earliest known technique for tuning of PID was proposed by Ziegler and Nichols [25] in the year 1942 and this technique is still very popular among the conventional techniques. Since then till now nearly 650 sources of tuning rules have been recorded. For a recent study, it has been experimentally checked that more than 30% of installed controller are operating in manual mode and 65% operating in automatic mode are poorly tuned [10]. The main point to note is that the use of conventional technique does not guarantee optimal control of the system with complex dynamics such as large dead time, inverse response and highly nonlinear characteristics. These techniques lack the intelligence and flexibility which would increase the performance rate and also improve the stability and error criterion [19, 14]. Since the PID controllers are usually poorly tuned for the systems with above said features, a higher degree of experience and technology are required for tuning [13].

During the last 25 years there has been significant developments in methods for model based control [11,17].A recent survey of evolutionary algorithms for evaluation of improved learning algorithm, control system can be found in [12,2]. Among the techniques found out, intelligent techniques and computational optimization techniques have found themselves a place in tuning of the parameters. The intelligent techniques include Artificial neural networks (ANN), Fuzzy logic (FL) which have developed over the last ten years [22,1]. Neural and fuzzy logic mimic the functioning of human intelligence process [3]. But their real time implementation is quite difficult [18], hence as a result of the above said problems optimization algorithms have received increasing attention by research community [23]. In recent years, there have extensive research on heuristic stochastic search techniques for optimization of the PID gains [9, 21].

An optimization algorithm is a numerical method or algorithm for finding the maxima or the minima of a function operating with certain constraints [26]. An optimal control is a set of differential equations describing the paths of the control variables that minimize the cost function [6,7]. Computational intelligence was the way in which optimisation was done. Computational intelligence (CI) is a successor of artificial intelligence relying on evolutionary computation, which is a famous optimization technique. Computational intelligence combines elements of learning; adaptation and evolution to create programs that are, in some sense, intelligent. Computational intelligence research does not reject statistical methods, but often gives a complementary view [19].The importance of CI lies in the fact that these techniques often find optima in complicated optimization problems more quickly than the traditional optimization methods [14].

Simulated Annealing (SA) is a derivative-free stochastic search method for determining the optimum solution in an optimization problem. The method was proposed by Kirkpatrick in 1983 and has since been used extensively to solve large-scale problems of combinatorial optimization. The SA evolves a single solution in the parameter space with certain guiding principles that imitate the random behavior of molecules during annealing process. It is similar to the physical process of heating up a solid until it melts, followed by cooling it down slowly until it crystallizes into a perfect lattice. The objective function here corresponds to the energy of the states of a solid [20]. The SA algorithm requires the definition of the neighborhood structure as well as the parameters for the cooling schedule. The temperature parameter distinguishes between large and small changes

in the objective function. An attractive feature of SA is that it is very easy to program and the algorithm typically has few parameters that require tuning.

In the proposed work we compare the time domain specifications, the values of the performance measures like integral of absolute error (IAE), the integral of time weighted square error (ITSE) and the integral of square time error (ISTE) [8,15] obtained by using the conventional technique i.e the Ziegler-Nichols method and our proposed method using SA to prove that our method is better than the conventional methods. In the section that follows we have given the explanation of our set up, a view of the conventional methods used, our proposed algorithm, the values obtained, the results and graphs and finally the conclusion.

2. EXPERIMENTAL SET UP:

The experimental set up consists of a DC shunt motor, rectifier module, tachogenerator, voltage divider, Analog and Digital Interface card(ADI Card) and a Personal Computer(PC).The DC motor speed is obtained by measuring the coupled tachogenerator output voltage which has a sensitivity of 5V/100rpm.The tachogenerator output is higher than the voltage range of the ADI card, which is in the range of 0 to 10V.The voltage divider converts the tachogenerator output voltage to a proportional voltage below 10V. This conditioned voltage is given as the feedback to the PC through the analog channel 1 of the ADI card .The rectifier module is controlled by the output voltage of the digital to analog converter (DAC)1 of the ADI card which is in the range of 0 to 10V.The ESA PCI DAS Card is a high speed PCI Bus Data acquisition add-on card which consists of 16 channel 12- bit analog to digital converter (ADC) with 10µs conversion time, Software programmable gain for analog inputs, two 12-bit DAC. The controller is realized with the C program with the analog channel of ADI card as the feedback and the DAC1 of the ADI card as the output to drive the motor to a required speed. The block diagram is given the figure1 and the technical details are provided in the table 1.The proposed controller could be made real time in future depending on the hardware available.

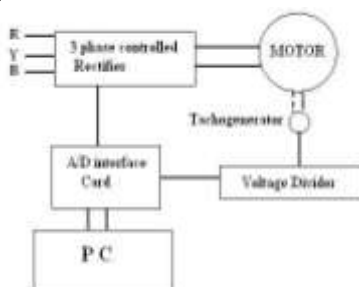


Figure1: Block diagram of the system considered.
Technical description of the setup

Table 1: Technical details of the experimental setup

PART NAME	DETAILS
DC motor	Shunt winding, power rating:3.5KW,speed:1500rpm, Voltage :220V,Current: 18.5A
ADI Card	ESAP DAS card, 16 Analog channels and 2 DAC with 12 bit

	recision,12microsecond conversion time. Range:0 to 10V
Tachogenerator	Sensitivity coefficient:5V/100rpm
Rectifier module	Three phase full controlled SCR bridge

2.1 Evaluation of transfer function:

The transfer function of the armature controlled DC motor is derived experimentally by calculating the values of the back electromotive force (emf) constant (K_b), Inertia constant(J),Armature resistance (R_a),inductance of the armature winding(L_a).

The transfer function of the armature controlled DC motor is of the form

$$\frac{\theta(s)}{E(s)} = \frac{K_t}{s((K_t * K_b / R_a F) + (1 + sT_a)(1 + sT_m))} \dots(1)$$

Where, $K_t' = \frac{K_b}{R_a F}$, $K_t = K_b$

K_b = back emf constant

J= inertia constant (Kg^m²)

T_a =Electrical Time constant

T_m = mechanical Time constant

2.2 Measurement of Armature resistance (R_a):

The armature current is varied by using the rheostat and the corresponding DC voltage is measured. The same procedure is repeated for different values of armature current and the readings are tabulated which is given in table2. By applying the relation between voltage(v), current(I) and resistance(R), ($v=I*R$) the resistance is calculated.

Table2: Measurement of Armature resistance.

Voltage(V)	Current(A)	R=V/I (Ω)
0.8	0.9	0.88
2	2	1
2.4	2.5	0.96
4.0	3.9	1.0256

The value of armature resistance is found to be $R_a=0.9686 \Omega$.

2.3 Measurement of Armature inductance (L_a):

The AC voltage applied across the armature of DC motor is varied by adjusting the variac and the corresponding current values are tabulated. The impedance is first measured with the relationship between current (I), voltage(V) and impedance(Z). The inductive reactance is calculated from the following relationship

$$X_l = \sqrt{Z^2 - Rac^2} \dots\dots\dots(2)$$

The armature inductance (L_a) is calculated from its relationship with the inductive reactance(X_l)

$$L_a = \frac{X_l}{2\pi f} \text{ H} \dots\dots\dots(3)$$

Table3: Measurement of Armature inductance

Voltage(V)	Current(A)	Impedance (Ω)
5	1.05	4.76
7.9	1.7	4.647
10.6	2.2	4.818
13.3	2.7	4.925

2.4 Measurement of Back emf constant (K_b):

The DC motor is made to run in the rated speed (N) by varying the motor field constant. When the motor has reached its rated speed(N)the single pole single through switch connected across the ammeter connected to the armature is closed. Now the values of the current(I_a) and voltage(V) is noted down. The value of back emf constant (K_b) is calculated from the equation4 given below.

$$K_b = \frac{v - I_a R_a}{2\pi N / 60} \dots\dots\dots(4)$$

The values obtained are as

For the open condition of the SPDT switch
 $V=210$ volts
 $I_a=1.2$ amps
 $N=1500$ rpm

For the closed condition of the SPDT switch
 $V=210$ volts
 $I_a=1$ amps
 $N=1500$ rpm

From the equation(4), the value of back emf constant(K_b) is evaluated as $K_b=1.3274$

2.5 Measurement of Inertia constant(J):

The DC motor is first made to run at the rated speed of 1500 rpm by varying the field and armature rheostat. When the motor has reached its rated speed the supply to the motor is disconnected and the time taken for the speed to fall from 1500 rpm (N_1) to 1000 rpm (N_2) is noted. The above explained procedure is carried out for conditions with 500W load and without load. The values obtained are as

Without load

Time for speed to fall from 1500rpm to 1000rpm without load, $t_1=3.69$ (s)

V_1 and I_1 at 1500 rpm are 0 and 0 respectively.

V_2 and I_2 at 1000 rpm are 0 and 0 respectively.

With load

Time for speed to fall from 1500rpm to 1000rpm with load, $t_2=1.78$ (s)

V_1 and I_1 at 1500 rpm are 220v and 1.5amps respectively.

V_2 and I_2 at 1000 rpm are 146v and 1.2 amps respectively

The inertia constant is evaluated from the following expression:

$$J = V_{av} I_{av} \left(\frac{t_1 t_2}{t_1 - t_2} \right) * \frac{1}{N_{av}} \left(\frac{60}{2\pi} \right)^2 * \frac{1}{\Delta N} \dots\dots\dots(5)$$

Where, $V_{av} = (V_1 + V_2) / 2 = 183V$

$N_{av} = (N_1 + N_2) / 2 = 1250$ rpm

$\Delta N = N_1 - N_2 = 500$ rpm

$I_{av} = (I_1 + I_2) / 2 = 1.35$ A

By substituting the values in equation(5), the value of inertia constant (J) is calculated as 0.12396.The

mechanical time constant T_m is evaluated from the following expression

$$T_m = \frac{t_2}{-\ln\left(\frac{N_2}{N_1}\right)} \dots\dots\dots(6)$$

The value of T_m is calculated as 4.39. Hence by substituting the calculated values in equation(1) the transfer function of armature controlled DC motor can be evaluated as follows.

$$\frac{\theta(s)}{E(s)} = \frac{34.71}{s \cdot 46.074 + (1 + 0.01074 s)(4.39 s + 1)} = \frac{34.71}{0.0471s^3 + 4.40s^2 + 47.074s} \dots\dots\dots(7)$$

Equation(7) gives the transfer function of the armature controlled DC motor.

3. CONVENTIONAL TUNING TECHNIQUES:

“Tuning” is the engineering work to adjust the parameters of the controller so that the control system exhibits desired property. In our paper we take up the oldest known tuning method proposed by Ziegler and Nichols[3]in the year 1942 in which a set of formulae is given to find out the controller parameters. The ultimate gain value for the above mentioned system has been calculated. Based on Ziegler-Nichols tuning method the tuning parameters has been calculated to be:

$K_u = 118.6266, P_u = 1.994$

Where, K_u = ultimate gain
 P_u = ultimate time period

Ziegler-Nichols ultimate sensitivity method

$$K_p = 0.6 K_u, T_i = 0.5 P_u, T_d = 0.125 P_u \dots\dots\dots(8)$$

From the above values the PID parameters are

$$K_p = 71.176 \quad K_i = 1209.38 \quad K_d = 3.02653$$

The frequency response of the system with PID tuned with Ziegler-Nichols method is compared with the proposed method of tuning in the forthcoming paragraphs.

4. SA BASED CONTROLLER:

4.1 Simulated Annealing :

SA is a numerical optimization technique based on the principles of thermodynamics. The Simulated Annealing method resembles the cooling process of molten metals through annealing . At high temperature, the atoms in the molten metal can move freely with respect to each another. But, as the temperature is reduced, the movement of the atoms gets reduced. The atoms start to get ordered and finally form crystals having the minimum possible energy. However, the formation of the crystal depends on the cooling rate. If the temperature is reduced at a very fast rate, the crystalline state may not be achieved at all; instead the system may end up in a polycrystalline state, which may have a higher energy state than the crystalline state. Therefore in order to achieve the absolute minimum state, the temperature needs to be reduced at a slow rate. The process of slow cooling is known as annealing in metallurgical parlance. SA simulates this process of slow cooling of molten metal to achieve the minimum function value in a minimization problem. The cooling phenomenon is simulated by controlling a temperature-like parameter introduced with the concept of the Boltzmann probability distribution. According to the Boltzmann probability distribution, a system in thermal equilibrium at a temperature T has its energy distributed probabilistically according to

$$P(E) = \exp(-\Delta E / kT),$$

where k is the Boltzmann constant. This expression suggests that a system at a high temperature has almost uniform probability of being at any energy state, but at a low temperature it has a small probability of being at a high energy state. Therefore, by controlling the temperature T and assuming that the search process follows the Boltzmann probability distribution, the convergence of an algorithm can be controlled using the Metropolis algorithm.

At any instant the current point is $x^{(t)}$ and the function value at that point is $E(t) = f(x^{(t)})$. Using the Metropolis algorithm, the probability of the next point being at $x^{(t+1)}$ depends on the difference in the function values at these two points or on $\Delta E = E(t+1) - E(t)$ and is calculated using the Boltzmann probability distribution:

$$P(E(t+1)) = \min[1, \exp(-\Delta E/kT)].$$

If $\Delta E \leq 0$, this probability is one and the point $x^{(t+1)}$ is always accepted. In the function minimization context, this makes sense because if the function value at $x^{(t+1)}$ is better than that at $x^{(t)}$, the point $x^{(t+1)}$ must be accepted. When $\Delta E > 0$, which implies that the function value at $x^{(t+1)}$ is worse than that at $x^{(t)}$. According to the Metropolis algorithm, there is some finite probability of selecting the point $x^{(t+1)}$ even though it is a worse than the point $x^{(t)}$. The principle is represented in figure.2. The new state $K1$ is accepted, but the new state $K2$ is only accepted with a certain probability.

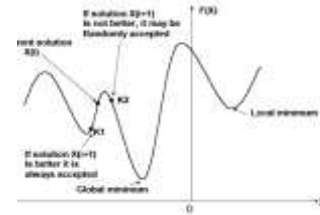


Figure 2. Selection of new states in SA

The probability of accepting a worse state is high at the beginning and decreases as the temperature decreases. For each temperature, the system must reach an equilibrium i.e., a number of new states must be tried before the temperature is reduced typically by 10 %. It can be shown that the algorithm will find, under certain condition, the global minimum and not get stuck in local minima. Figure 3.illustrates the flowchart of SA algorithm.

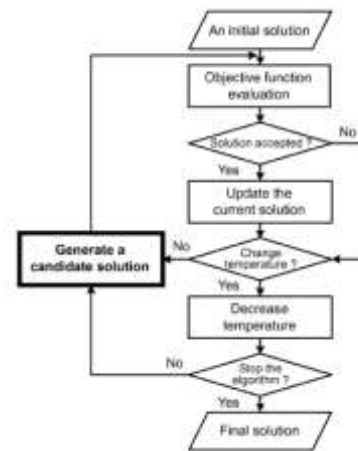


Figure 3. Flowchart for SA algorithm

5. IMPLEMENTATION OF SA ALGORITHM:

The optimal values of the conventional function PI controller parameters K_p and K_i , is found using SA. All possible sets of controller parameter values are particles whose values are adjusted so as to minimize the objective function, which in this case is the error criterion, which is discussed in detail. For the PI controller design, it is ensured the controller settings estimated results in a stable closed loop system.

5.1 Selection of SA parameters:

To start up with SA, certain parameters need to be defined. It includes initial temperature (T_i), decrement temperature (T_d), population size, terminating temperature (T_t), multiplication factor etc. Selection of these parameters decides to a great extent the ability of global minimization. The initial temperature and decrement factor, through the multiplication factor decides the number of iterations that affect the ability of escaping from local optimization and refining global optimization. In this work the parameters are so selected to have 100 iterations. The population size balances the requirement of global optimization and computational time. The range of the tuning parameters is considered in the range of 0-10. Initializing the values of the parameters for this paper is as follows:

Population size – 100
 Initial Temperature- 1500
 Decrement Temperature- 10
 Termination temperature- 0.001
 Multiplication Factor – $(T_i - T_d) / T_i = 0.995$

5.2. Objective function for Simulated Annealing

The objective functions considered are based on the error criterion. A number of such criteria are available and in this paper controller’s performance is evaluated in terms of Integral time absolute error (ITAE) error criteria. The error criterion is given as a measure of performance index given by the equation:

$$ITAE = \int_0^T t|e(t)|dt \dots \dots \dots \theta$$

5.3. Termination criteria

Termination of optimization algorithm can take place either when the maximum number of iterations gets over or with the attainment of satisfactory fitness value. Fitness value, in this case is nothing but reciprocal of the magnitude of the objective function, since we consider for a minimization of objective function. In this paper the termination criteria is considered to be the attainment of satisfactory fitness value which occurs with the maximum number of iterations as 100. For each iteration the best among the 100 particles considered as potential solution are chosen Therefore the best values for 100 iterations is sketched with respect to iterations, and are as shown in Figures. 4,5 and 6.

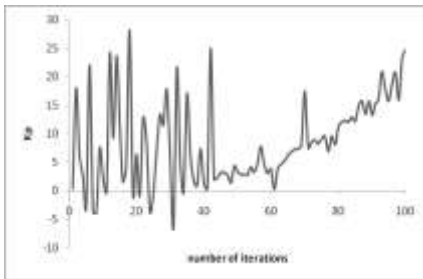


Figure. 4. Iteration progress for K_p

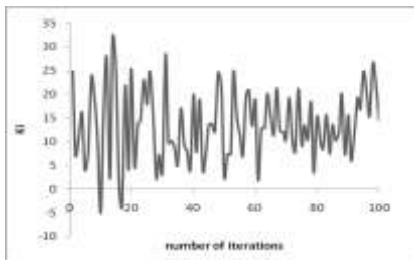


Figure.5. Iteration progress for K_i

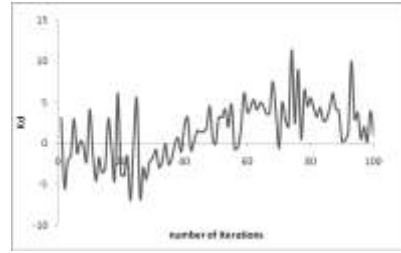


Figure.6. Iteration progress for K_d

PID controller was formed based upon the respective parameters for 100 iterations, and the global best solution was selected for the set of parameters, which had the minimum error. A sketch of the error based on ITAE criterion for 100 iterations is as given in Figure. 7.

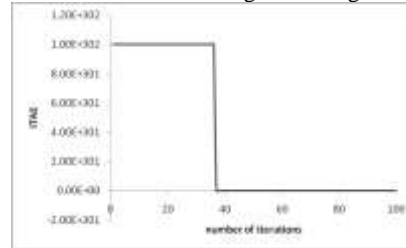


Figure.7. ITAE values for 100 iterations.

It was seen that the error value tends to decrease for a larger number of iterations. As such the algorithm was restricted to 100 iterations for beyond which there was only a negligible improvement. Based on SA for the application of the PID tuning we get the PID tuning parameters for the model as

$$K_p = 24.706 \quad K_i = 14.546 \quad K_d = 0.92894$$

6. RESULTS AND COMPARISON

Analysis shows that the design of proposed controller gives a better robustness, and, the performance is satisfactory over a wide range of process operations. Simulation results show performance improvement in time domain specifications for a step response. All the simulations were implemented using MATLAB.

6.1. Performance related to steady state conditions

A reference input of unit step was given to the closed loop system. The above procedure was implemented for the controller for which the PID values were tuned by Zeigler-Nichols as well as SA algorithm. The response curves obtained are as shown in the Figure.8.

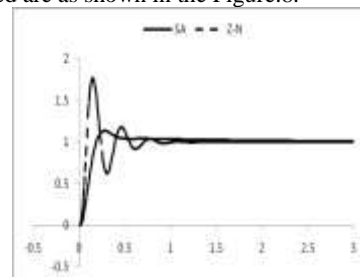


Figure.8: Unit step response with Z-N and SA based controller

A comparison of time domain specifications peak overshoot, peak time, rise time and settling time are tabulated as given in table (4).

6.2. Robustness Investigation:

The PID controllers tuned by the SA based method should not be compared only with their time domain responses but also with its performance index from the four major error criterion techniques of Integral Time of Absolute Error (ITAE) ,Integral of Absolute Error(IAE) ,Integral Square of Error(ISE)and Mean Square Error (MSE). For the proposed model the comparison of performance index were done and are listed as per the given Table4.

Table4: Comparison of performance index, time domain specifications obtained for Z-N and SA

	Z-N	SA
IAE	2.1064	1.9220
ISE	1.3375	1.0224
ITAE	0.3320	0.9860
MSE	0.0027	0.0022
OVERSHOOT(%)	52.3	14.14
RISE TIME(S)	0.0367	0.1397
SETTLING TIME(S)	0.924	0.74625
STEADY STATE	1.2	1.0026

From these values obtained it is clearly visible that the error magnitude obtained for Z-N is far too high as compared to the proposed tuning method based on SA algorithm.

7. CONCLUSION

In the proposed work, a systematic design method aiming at enhancing PID control for a real time process is proposed. It is shown analytically and graphically that there is a substantial improvement in the time domain specification in terms of lesser rise time, peak time, settling time as well as a lower overshoot. The performance index for various error criterions for the proposed controller using Simulated Annealing is proved to be less than the controller tuned by Ziegler Nichols method.SA presents multiple advantages to a designer by operating with a reduced number of design methods to establish the type of the controller, giving a possibility of configuring the dynamic behavior of the control system with ease, starting the design with a reduced amount of information about the controller (type and allowable range of the parameters), but keeping sight of the behavior of the control system. The performance of the above said method of tuning a PID controller can even be proved to be better than the method of tuning the controller after approximating the system to a first order process with time delay(FOPTD) model, and then using the traditional techniques, regarding which a rich literature is available. Since the above obtained values using the conventional method is found to be in a range that is difficult to be set for electrical components, the values obtained using the proposed method holds an upper hand as these values are within the range of the components. So this method of tuning can be applied to any system irrespective of its order and can be proved to be better than the existing traditional techniques of tuning the controller.

8. REFERENCES

- [1] Afzalian.A,D.A.Linkens,2000,Training of neuro fuzzy power system stabilisers using genetic algorithms,"International Journal of Electrical Power and Energy Systems,vol.22,no.2,pp.93-102.
- [2] Angeline, P.J.,1998: Using Selection to Improve Particle Swarm Optimization, IEEE int. Conf. 4- 9, pp. 84-89.
- [3] Asriel U. Levin and Kumpati S. Narendra,1996, Control of nonlinear dynamical systems using Neural Networks- Part II : observability, identification and control, IEEE Transactions on Neural Networks, Vol. 7, No. 1.
- [4]Astrom,K.J.,Hagglund,T.,1995:PIDcontrollers:Theory , Design, and Tuning. Instruments Society of America. 2edn.
- [5] Astrom K.J, T.Hagglund,2001, The future of PID control, ControlEng.Pract.9(11)pp.1163-1175.
- [6] Chen.S, R.H. Istepanian and J.Wu,1999, " Optimizing stability bounds of finite-precision PID controllers using adaptive simulated annealing", Proceedings of the American control conference ,San Diego, California .
- [7] Chen.S and B.L.Luk,1999, "Adaptive simulated annealing for optimization in signal processing applications", Journal of Engineering and electronics, Vol. 79,pp. 117-128.
- [8]DorigoM,ManiezzoV,ColomiA.,1996The ant system:optimization by a colony of cooperating agents.IEEETransSystManCybernetB;26(1):29-41.
- [9] Dubey.M,P.Gupta,2005 ,Design of Genetic Algorithm Based Robust Power System Stabilizer,"International Journal of Computational Intelligence,vol.2,no.1,pp.48-52.
- [10]Ender,D.B.,1993,Processcontrolperformance:Control.Eng. pp180-190.
- [11]GarciaC.E,D.M.Prett,M.Morari,1989,Model predictive control;Theory and practice-A survey," Automatica,vol.25,no.3,pp.333-348.
- [12] Kennedy,J. and Eberhart,R.,1995: Particle Swarm Optimization in Proc. IEEE int. Conf. Neural Networks, vol. IV, Perth, Australia
- [13] Matsummura,S.,1998: Adaptive Control for the SteamTemperature of Thermal Power Plants,Proceedings the 1993 IEEE on Control applications pp.1105-1109.
- [14] Mehrdad Salami and Greg Cain,1995, An adaptive PID controller based on Genetic algorithm processor, Genetic algorithms in engineering systems: Innovations and applications, 1214, Conference publication No. 414, IEEE.
- [15]Mwembeshia.M.M,C.A.Kenta,S.Salhi,2004,A genetic algorithm based approach to intelligent modelling and control of pH in reactors,"Computers and Chemical Engineering,vol.28,pp.1743-1757.
- [16]O'Dwyer,A.,2000,PIandPIDControllerTuningRulesforTimeDelayProcesses:aSummary.Tech.Rep.AOD00-01 Ver.1,DublinInstituteofTechnology,Ireland.

- [17] Qin S.J,T.A.Badgwell,1997,An over view of industrial model predictive control technology," AICHESymposium Series.
- [18] Simon Fabri and Visakan Kadiramanathan,1996 Dynamic structure neural networks for stable adaptive control of nonlinear systems, IEEE Transactions on Neural Networks, Vol. 7, No. 5, September.
- [19] Su Whan Sung, In-Beum Lee and Jitae Lee,1995, Modified Proportional-Integral Derivative (PID) Controller and a New Tuning Method for the PID Controller, Ind. Eng. Chem. Res., 34, pp. 4127-4132.
- [20] Varela L.R, R.A.Ribeiro and F.M.Pires,2002, "Simulated annealing and fuzzy optimization", Proceedings of the 10th Mediterranean conference on control and automation- MED2002,Portugal, July 9-12.
- [21] Vlachos .C,D.Williams,J.B.Gomm,2002,Solution to the Shell standard control problem using genetically tuned PID controllers," Control Engineering Practice,vol.10,no.2,pp.151-163,2002.
- [22]Vieira,J.F.MorgadoDias,A.Mota,2004,Artificial Neural Networks and Neuro fuzzy Systems for Modeling and Controlling Real Systems: A Comparative Study,"Engineering Applications of Artificial Intelligence,vol.17,pp.265-273.
- [23] Wah.B and Y. Chen,2000, Constrained genetic algorithms and their applications in nonlinear constrained optimization, In Proceedings of International conference on tools with artificial intelligence, IEEE, pp. 286-293.
- [24] Yu,C.C.,1999 :Auto-tuning of PID Controllers. Berlin: Springer 7edn .
- [25]Ziegler J.G,N.B.Nichols,1942,Optimum setting for AutomaticControllers," Trans.ASME,vol.64,pp.759-768.
- [26] Zwe-Lee Gaing,2002, " A particle swarm optimization approach for optimum design of PID controller in AVR system", IEEE transactions on energy conversion, November 6.