

International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

Control OF DC Motor Using Artificial Bee Colony based PID Controller

Prakash Varma prakash18885@gmail.com

Abstract – DC drive systems are often used in many industrial applications such as robotics, actuation and manipulators. In the first two, a wide range of position control is required. Tuning method for PID controller is very important for the process industries. Proportional Integral Derivative controllers have the advantage of simple structure, good stability, and high reliability. Accordingly, PID controllers are widely used to control system outputs, especially for systems with accurate mathematical models. The key issue for PID controllers is the accurate and efficient tuning of parameters. The aim of this paper is to study the Position control of DC motor using Artificial Bee Colony Algorithm. In order to solve this problem a PID controller under Artificial Bee Colony Algorithm with self-tuning is applied, which will perform high efficiency position control. The efficiency of Control Algorithm is presented through a simulation and compared with the quality of PID controller. The proposed method is compared with Ziegler Nichols method. It is found that the proposed PID parameters adjustment by the Artificial Bee Colony Algorithm is better than the Ziegler & Nichols' method.

Keywords – Artificial Bee Colony Algorithm DC drive systems, PID controller, and Ziegler Nichols method.

I. INTRODUCTION

The direct current (DC) motor is a device that used in many industries in order to convert electrical energy into mechanical energy. This is all result from the availability of speed controllers is wide range, easily and many ways. In most applications, speed control is very important. For example, if we have DC motor in radio controller car, if we just apply a constant power to the motor, it is impossible to maintain the desired speed. It will go slower over rocky road, slower uphill, faster downhill and so on. So, it is important to make a controller to control the speed of DC motor in desired speed.

DC motor plays a significant role in modern industry. The purpose of a motor speed controller is to take a signal representing the B. Anjanee Kumar anjanee_kumar@rediffmail.com

demanded speed, and to drive a motor at that speed. There are numerous applications where control of speed is required, as in rolling mills, cranes, hoists, elevators, machine tools, transit system and locomotive drives. These applications may demand high-speed control accuracy and good dynamic responses.

The control of DC motor uses the digital signal processing system. Proportional Integral Derivative (PID) controller has been widely used for processes and motion control system in industry. Now more than 90% of control systems are still with PID controllers. The most critical step in the application of PID controller is parameters tuning. The main objective of the work is to design a position controller of a DC motor by selection of PID parameters using Artificial Bee Colony algorithm.

II. BACKGROUND

D.C. Motor

At the most basic level, electric motors exist to convert electrical energy into mechanical energy. This is done by way of two interacting magnetic fields – one stationary, and another attached to a part that can move. A number of types of electric motors exist, but mostly used DC motors in some form or another. DC motors have the potential for very high torque capabilities (although this is generally a function of the physical size of the motor), are easy to miniaturize, and can be "throttled" via adjusting their supply voltage. DC motors are also not only the simplest, but the oldest electric motors.

The basic principles of electromagnetic induction were discovered in the early 1800's by Oersted, Gauss, and Faraday. By 1820, Hans Christian Oersted and Andre Marie Ampere had discovered that an electric current produces a magnetic field. The next 15 years saw a flurry of cross-Atlantic experimentation and innovation, leading finally to a simple DC rotary motor. A

International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

number of men were involved in the work, so proper credit for the first DC motor is really a function of just how broadly we choose to define the word "motor". Figure 1 shows the basic parts of a DC motor.



Figure 1: Parts of a DC motor

Position Control of a DC motor

The position of the motor is the rotation of the motor shaft or the degree of the rotation which is to be controlled by giving the feedback to the controller which rectifies the controlled output to achieve the desired position.

Proportional Integral Derivative (PID) controller has been widely used for processes and motion control system in industry.

PID Controller



Figure 2: Schematic of the PID Controller

PID controller consists of Proportional Action, Integral Action and Derivative Action. It is commonly refer to Ziegler-Nichols PID tuning parameters. It is by far the most common control algorithm. Under this heading, the basic concept of the PID controls will be explained. PID controller's algorithm are mostly used in feedback loops. PID controllers can be implemented in many forms. It can be implemented as a stand-alone controller or as part of Direct Digital Control (DDC) package or even Distributed Control System (DCS).

A diagram illustrating the schematic of the PID controller is shown in Figure 2. Such set up is known as non-interacting form or parallel form.

In proportional control,

$$P_{\text{term}} = K_P * \text{Error}$$

(1)

(2)

It uses proportion of the system error to control the system. In this action an offset is introduced in the system.

In Integral control,

$$I_{term} = K_I * \int Error \, dt$$

It is proportional to the amount of error in the system. In this action, the I-action will introduce a lag in the system. This will eliminate the offset that was introduced earlier on by the P-action.

In Derivative control,

$$D_{term} = K_D * \frac{d(Error)}{dt}$$
(3)

It is proportional to the rate of change of the error. In this action, the D-action will introduce a lead in the system. This will eliminate the lag in the system that was introduced by the I-action earlier on.

Continuous PID

The three controllers when combined together can be represented by the following transfer function.

$$G_{C}(s) = K (1 + 1/sT_{i} + sT_{d})$$

(4)

This can be illustrated below in the following block diagram:



Figure 3: Block diagram of Continuous PID Controller

What the PID controller does is basically is to act on the variable to be manipulated through a proper combination of the three control actions that is the P control action, I- control action and D control action.

The P action is the control action that is proportional to the actuating error signal which is the

(5)

O IJDACR International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

difference between the input and the feedback signal. The, I-action is the control action which is proportional to the integral of the actuating error signal. Finally the D action is the control action which is proportional to the derivative of the actuating error signal. With the integration of all the three actions, the continuous PID can be realized.

This type of controller is widely used in industries all over the world. In fact a lot of research, studies and application have been discovered in the recent years.

Optimization of PID Controller

Zeigler & Nichols Method

For the system under study, Zeigler-Nichols tuning rule based on critical gain *Ker* and critical period *Per* will be used. In this method, the integral time T_i will be set to infinity and the derivative time T_d to zero. This is used to get the initial PID setting of the system. This PID setting will then be further optimized using the Basic Design mode and Extended Design mode.

In this method, only the proportional control action will be used. The K_p will be increased to a critical value *Ker* at which the system output will exhibit sustained oscillations. In this method, if the system output does not exhibit the sustained oscillations hence this method does not apply.

From the response below, the system under study is indeed oscillatory and hence the Z-N tuning rule based on critical gain *Ker* and critical period *Per* can be applied.



Figure 4: Illustration of Sustained oscillation with period Per

The transfer function of the PID controller is

$$G_{C}(s) = K (1 + 1/sT_{i} + sT_{d})$$

The objective is to achieve a unit-step response curve of the designed system that exhibits a maximum overshoot of 25 %. If the maximum overshoot is excessive says about greater than 40%, fine tuning should be done to reduce it to less than 25%. The system under study above has a following block diagram.



Figure 5: Block diagram of Controller and Plant

Since the $T_i = \infty$ and $T_d = 0$, this can be reduced to the transfer function of

$$R(s)/C(s) = K_p/s(s+1)(s+5) + K_p$$
(6)

The value of K_p that makes the system marginally stable so that sustained oscillation occurs can be obtained by using the Routh's stability criterion. Since the characteristic equation for the closed-loop system is

$$s^{3} + 6s^{2} + 5s + K_{p} = 0$$
 (7)

From the Routh's Stability Criterion, the value of K_p that makes the system marginally stable can be determined.

The table below illustrates the Routh array.

s ³	1	5
s ²	6	K _p
s ¹	$(30 - K_p)/6$	0
s ⁰	Kp	0

By observing the coefficient of the first column, the sustained oscillation will occur if $K_p = 30$.

Hence the critical gain Ker = 30

Thus with K_p set equal to *Ker*, the characteristic equation becomes

$$s^3 + 6s^2 + 5s + 30 = 0 \tag{9}$$



International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

The frequency of the sustained oscillation can be determined by substituting the *s* terms with $j\omega$ term. Hence the new equation becomes:

$$(j\omega)^3 + 6(j\omega)^2 + 5(j\omega) + 30 = 0$$
(10)

This can be simplified to

$$6(5 - \omega)^{2} + j\omega(5 - \omega) = 0$$
(11)

From the above simplification, the sustained oscillation can be reduced to

$$\omega^2 = 5 \text{ or } \omega = \sqrt{5}$$

(12)

The period of the sustained oscillation can be calculated as

$$Per = 2\pi/\sqrt{5}$$
$$= 2.8099$$

(13)

From Ziegler-Nichols frequency method of the second method, Table 2 suggested tuning rule according to the formula shown. From these we are able to estimate the parameters of K_p , T_i and T_d .

Table 2: Recommended PID value Setting	Table 2:	Recommended	PID	Value	Setting
--	----------	-------------	-----	-------	---------

Type of controller	K _p	T _i	T _d
Р	0.5 Ker	∞	0
PI	0.45 Ker	(1/1.2) <i>Per</i>	0
PID	0.6 Ker	0.5 Per	0.125
			Per

Hence from the above table, the values of the PID parameters K_p , T_i and T_d will be:

$$K_p = 30$$

 $T_i = 0.5 \times 2.8099$
 $= 1.405$
 $T_d = 0.125 \times 2.8099$
 $= 0.351$

(14)

(15)

The transfer function of the PID controller with all the parameters putting in below equation

$$G_{C}(s) = K (1 + 1/sT_{i} + sT_{d})$$

From the above transfer function, we can see that the PID controller has pole at the origin and double zero at s = -1.4235. The block diagram of the control system with PID controller is as follows:



Figure 6: The Closed Loop Transfer Function

III. METHODOLOGY

The Ziegler-Nichols formulation is a classical tuning method which found a wide range of applications in the controller design process. However, computing the gains does not always give best results because the tuning criteria presume a one-fourth reduction in the first two-peaks that's why system stability here is matter of unreliable stability that's why we are adopting ABC (Artificial Bee Colony algorithm) here.

Artificial Bee Colony (ABC) Algorithm

The ABC algorithm is a swarm based, metaheuristic method based on the model first proposed by [15] on the foraging behaviour of honey bee colonies. The model is composed of three important elements: employed and unemployed foragers, and food sources. The employed and unemployed foragers are the first two elements, while the third element is the rich food sources close to their hive. The two leading modes of behaviour are also described by the model. These behaviours are necessary for self -organization and collective intelligence: recruitment of forager bees to rich food sources, resulting in positive feedback and simultaneously, the abandonment of poor sources by foragers, which causes negative feedback [16].

The ABC consists of three groups of artificial bees: employed foragers, onlookers and scouts. The employed bees comprise the first half of the colony whereas the second half consists of the onlookers. The employed bees link to particular food sources. In other words, the number of employed bees is equal to the number of food sources for the hive. The onlookers observe the dance of the employed bees within the hive, to select a food source, whereas scouts search randomly for new food sources. Analogously in the optimization context, the number of food sources (that is the employed or onlooker bees) in ABC algorithm, is equivalent to the number of solutions in the population. Moreover, the position of a food source



International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

signifies the position of a promising solution to the optimization problem, whereas the value of nectar of

a food source represents the fitness cost (quality) of the associated solution.



Figure 7: Flow diagram of ABC algorithm

The search cycle of ABC consists of three rules:

- 1. Sending the employed bees to a food source and evaluating the nectar quality.
- 2. Onlookers choosing the food sources after obtaining information from employing bees and calculating the nectar quality
- 3. Determining the scout bees and sending them on to possible food sources.

The procedure of ABC could be described in the following seven steps:



International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

1. Initialization of ABC and optimization problem parameters

In general, optimization problem could be formulated as follows:

Where f(x) is the objective function to be minimized; x is the set of decision variables $\{x_i | i = 1, ..., N\}$. X is the possible range for each decision variable, where $X = \{X_1, X_2, ..., X_N\}$ and $X_i \in (LB_i, UB_i)$ and LB_i and UB_i are the lower and upper bound values for the variable x_i . N represents the number of decision variables and, g(x) and h(x)are the inequality and equality constraints, correspondingly.

Additionally, ABC consists of three control parameters:

- 1. Population size (SN) is the number of food sources (or solutions) in the population. SN is equal to the number of employed bees or onlooker bees.
- 2. Maximum Cycle Number (MCN) refers to the maximum number of generations.
- 3. The limit is used to diversify the search, to determine the number of allowable generations for which each non-improved food source is to be abandoned.

2. Initialization of the Food Source Memory (FSM)

The Food Source Memory (FSM) is an augmented matrix of size $SN \times N$ comprised in each row, a vector representing a food source as in equation (16). Note that the vectors in the FSM are sorted in ascending order, according to proximity cost function values.

$$FSM = \begin{bmatrix} x_1(1) & x_1(1) & . & x_1(1) \\ x_2(2) & x_2(2) & . & x_2(2) \\ \vdots & \vdots & \ddots & \vdots \\ x_{SN}(1) & x_{SN}(1) & . & x_{SN}(1) \end{bmatrix} \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{SN}) \end{bmatrix}$$
(16)

Generally, each vector is generated as follows:

$$x_{j}(i) = LB_{i} + (UB_{i} - LB_{i}) \times r$$

$$\forall_{j} \in (1, 2, \dots, SN), \forall_{i} \in (1, 2, \dots, N)$$
(17)

Note that $r \sim (0,1)$ generates a uniform random number between 0 and 1.

3. Assigning employed bees to the food sources

In this step, each employee bee is assigned to its food source and in turn, a new one is generated from its neighbouring solution, using equation (18) as is shown in algorithm (1):

$$\begin{aligned} x'(i) &= x_j(i) \pm r\left(x_j(i) - x_k(i)\right) \\ \forall k \in (1, 2, \dots, SN) \ k \neq jand \ r \sim (0, 1) \end{aligned}$$
(18)

Algorithm (1): Employed Bee Phase 1. for j = 1, 2, ..., SN do 2. for i = 1, 2, ..., SN do 3. $x'(i) = x_j(i) \pm r(x_j(i) - x_k(i))$ $\forall k \in (1, 2, ..., SN) \ k \neq jand \ r \sim (0, 1)$ 4. end 5. calculate $f(x_i)$ 6. if $(f(x') \leq f(x_i))$ then 7. $x_i = x'$ 8. $f(x_i) = f(x')$ 9. end if 10. end for

4. Sending the onlooker bees

The onlooker bee has the same number of food sources as the employed. It initially calculates the selection probability of each food source generated by the employed bee in the previous step. The fittest food source is selected by the onlooker, using Roulette Wheel selection mechanism. The process of selection at the onlooker phase works as follows:

1. Assign for each employed bee a selection probability p_j as follows:

 $p_j = \frac{f(x_j)}{\sum_{k=1}^{SN} f(x_k)}$

2. The food source of the employed bee with the highest fitness is selected by the onlooker bee, based on its selection probability and adjusted as shown in the algorithm (2).

In the algorithm, *sum_prob* is the accumulated probability of all the employed bees; where the *sum_prob* of solution x_j ; $\{j = 1, 2, ..., SN\}$ is unity *Algorithm* (2): *Onlooker Bee Phase*

1. for i = 1, 2, ..., SN do

2.
$$r \sim (0,1)$$



International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

 $sum_prob = 0$ 3. 4. j = 05. while $(sum_prob \leq r)$ do 6. $sum_prob = sum_prob + p_i$ 7. j = j + 1for i = 1, 2, ..., SN do8. $x'(i) = x_j(i) \pm r\left(x_j(i) - x_j(m)\right)$ 9. $\forall m \in$ (1, 2, ..., SN)10. end for11. calculate $f(x_i)$ 12. if $(f(x_i) \leq f(x_i))$ then $x_i = x_{ij}$ 13. 14. $f(x_i) \leq f(x_i)$ then 15. end if 16. end for

5. Sending the Scout to search for possible new food sources

The scout bee carries out a random search to replace the abandoned food sources, using equation (17). The abundant food source is one that cannot be improved upon after a certain number of cycles, as determined by the limit parameter. Algorithm (3) describes the process of the scout bee;

In algorithm (3), Scout (i) is a vector of size (SN), which contain information related to the improvement in any of the food sources during search.

Algorithm (3): Scout Bee Phase

1. for i = 1, 2, ..., SN do

- 2. *if* (scout(i) = limit) *then*
- 3. generate x_i using equation (17)
- 4. end if
- 5. end for

6. Memorizing the best food source

This involves memorizing the fitness and position of the best food source, x^{best} found so far in FSM.

7. Stop condition

Steps 3 to 6 are repeated until a stop criterion is met. This is originally determined by the MCN value.



Figure 8: Response of DC Motor by Zeigler Nichols method



Figure 9: Response of DC Motor by Artificial Bee Colony (ABC) Algorithm



Figure 10: Comparison of DC Motor Response by Zeigler Nichols method and Artificial Bee Colony (ABC) algorithm

International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

Table 3: Comparison of result					
	Z-N	ABC			
Rise Time (Sec.)	0.038557	0.031884			
Settling Time (Sec.)	0.736871	0.299073			
Overshoot	53.671479	33.148262			
(Amplitude Unit)					

From Table 3, it is found that the overall system gives better response (in terms of Rise Time, Settling Time and Overshoot) when tuned by Artificial Bee Colony Algorithm compared to classical Ziegler Nichols Method (See Figure 10).

V. CONCLUSION

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

REFERENCE

- [1] Math Works, 2001, Introduction to MATLAB, the Math Works, Inc.
- [2] J. J. D'Azzo and C. H. Houpis, "Linear control system analysis and design", McGraw-Hill, New York, 1995.
- [3] S. J. Chapman, Electric machinery fundamentals, 3rd ed., WCB/McGraw-Hill, New York, 1998.
- [4] Astrom, K., T. Hagglund, "PID Controllers; Theory, Design and Tuning", Instrument Society of America, Research Triangle Park, 2005.
- [5] G. A. Gurzadyan, Space Dynamics, Taylor & Francis Inc, London, 2002.
- [6] O. Dwyer, "PI And PID Controller Tuning Rules for Time Delay Process: A Summary. Part 1: PI Controller Tuning Rules", Proceedings of Irish Signals and Systems Conference, June1999.
- [7] M. Chow and A. Menozzi, "On the comparison of emerging and conventional techniques for DC motor control", Proc. IECON, pp. 1008- 1013, 1992.

- [8] S. Li and R. Challoo, "Restructuring electric machinery course with an integrative approach and computer- assisted teaching methodology", IEEE Trans Educ. (49), 2006.
- [9] P. C. Sen, "Electric Motor Drives and Control: Past, Present and Future", IEEE Transaction on Industrial Electronics, Vol. IE37, No. 6, pp. 562-575, 1990.
- [10] S. Skogestad, "Probably The Best Simple PID Tuning Rules In The World" Journal Of Process Control, September 2000.
- [11] Ang K., Chong G., Li Y., "PID control system analysis, design, and technology", IEEE Trans. Control System Technology, vol. 13, p. 559 – 576, Jul. 2005.
- [12] Anguluri Rajasekhar, Ajith Abraham, Pratap Kunathi, Millie Pant, "Fractinal Order Speed Control of DC Motor Using Levy Mutated Artificial Bee Colony Algorithm", IEEE, World Congress on Information and Communication Technologies (WICT), pp. 7-13, December 2011.
- [13] Mohammed E. El-Telbany, "Tuning PID Controller for DC Motor: An Artificial Bees Optimization Approach", International Journal of Computer Applications, ISSN: 0975 – 8887, Volume 77– No.15, September 2013.
- [14] Akhilesh Kumar Mishra, Vineet Kumar Tiwari, Rohit Kumar, Tushar Verma, "Speed Control of DC Motor Using Artificial Bee Colony Optimization Technique", Universal Journal of Electrical and Electronic Engineering, pp. 68-75, October 2013.
- [15] V. Tereshko and A. Loengarov. "Collective decision making in honey-bee foraging dynamics", Computing and Information Systems, 9(3):1, 2005.
- [16] D. Karaboga and B. Akay. "Artificial bee colony algorithm on training artificial neural networks", In Signal Processing and Communications Applications, 2007. SIU 2007. IEEE 15th, pages 1–4. IEEE, 2007.
- [17] Capolino G.A., Cirrincione G., Cirrincione M., Henao H., Grisel R., "Digital Signal Processing for Electrical Machines", Invited paper, Proceedings of ACEMP'01 (Aegan International Conference on Electrical Machines and Power Electronics), Kusadasi, Turkey, June 2001, p. 211 – 219.
- [18] B.C. Kuo "Automatic Control Sytems" (sixth edition), Prentice Hall, 1991.
- [19] K Ogata, "Modern Control Systems", University of Minnesota, Prentice Hall, 1987.
- [20] Lancet Mit "Introduction to D.C. Motors" User conference, Austin texas Vol 11 2004.
- [21] Matt Krass, PID Control Theory, Matt Krass 2nd Edition 2006.
- [22] Saffet Ayasun "DC Motor Speed Control Methods Using MATLAB/Simulink" Department of Electrical and Electronics Engineering, Nigde University, Nigde 51100, Turkey 2006.
- [23] S. J. Chapman, "Electric machinery fundamentals", 3rd edition, WCB/McGraw-Hill, New York, 1999.
- [24] Kamaraj "Tuning Algorithms for PID Controller Using Soft Computing Techniques" 2001.



International Journal of Digital Application & Contemporary research Website: www.ijdacr.com (Volume 2, Issue 3, October 2013)

- [25] Aigner, P., Pythian, M., Wen, P. and Black, J, "Computer Controlled System", Distance Education Centre, USQ, 2003.
- [26] P.E. Wellstead and M.B Zarrop "Self Tuning Systems, Control and Signal Processing", (Publishers (WILEY)), pp. 140-150, 1991.
- [27] Dorf, R.C and Bishop, R.H. "Modern Control Systems". 11ed. Prentice Hall. 887, 2007.
- [28] A. Carriere, J. Perriere: "A Two-Step automatic Tuning Procedure for PID controllers" IEEE Conference on control applications, Sep-13-16, 1993.
- [29] Qing-Guo Wang, Tong-Heng Lee: "PID Tuning for Improved Performance" IEEE Transactions On Control Systems Technology, VOL-7, NO.4 JULY 1999, pp:457-465.
- [30] Microchip Technology, Inc.2001, PIC16F84A Data Sheet, www.microchip.com.
- [31] K. Peney and G. Littlefair, "Free search comparative analysis", Information Sciences 172 (1–2) (2005), pp. 173–193.