MEKATRONIKA
e-ISSN: 2637-0883
VOL. 5, ISSUE 1, 72 – 78
DOI: https://doi.org/10.15282/mekatronika.v5i1.9414

ORIGINAL ARTICLE

Comparison Performances between MABSA-PI Controller and MABSA-PD Controller for DC Motor Position System

Mohamad Harith Hilmi Fajaffri[1] and Nafrizuan Mat Yahya[1]

[1]Faculty of Manufacturing and Mechatronic Engineering Technolgy, Universiti Malaysia Pahang, 26600 Pahang, Malaysia.

ABSTRACT –The goal of this project is to create two different types of controllers for regulating DC motor position systems which are the MABSA-PI controller and the MABSA-PD controller. The MATLAB controllers are developed using the Simulink toolbox, while the transfer function of a DC motor is obtained from its circuit. Modified adaptive bats sonar algorithm (MABSA) will be implemented in the PI controller and PD controller as the main algorithm. MABSA is one of the swarm intelligences and is usually for an optimization problem. Next, the tuning method for the PI controller and PD controller will be the trial-and-error method. The combination graph of each controller will be shown in Simulink scope as the project is tested for various positions. The controller's ability to regulate the position of a DC motor was studied. The comparision among the system's transient response parameters, including percentage overshoot, rising time (Tr), and settling time (Ts), would be discused. Various toolbox functions available in Simulink software can be employed to repossess these parameter outcomes.

ARTICLE HISTORY
Received: 2nd March 2023
Revised: 5th April 2023
Accepted: 14th April 2023
Published: 28st April 2023

KEYWORDS
MABSA
PI controller
PD controller
DC motor position system
Transient response

INTRODUCTION

A control system encompasses a set of equipment that oversees, directs, guides, or normalises the purposes of other systems or devices, aiming to attain a particular result. The control loops are a type of procedure that keeps a process variable at a predetermined value. These control loops make the control system works. In recent years, control systems have been critical to the creation and growth of advanced technologies and civilization. A control system affects almost every aspect of our everyday life in some way. For instance, there are many products for our daily life such as automated toilets, cruise control, air conditioning, etc.

An closed-loop system and a open-loop system are the two sorts of control systems. The closed-loop system is one in which the output is impacted by the input in such a manner that the input quantity adjusts itself depending on the generated output, whereas an open-loop system controls the action fully independent of the system output. In a control system, there are a few types of controllers which are a proportional derivative controller (D), controller (P) and integral controller (I). Our focus is on the two types of controllers which are the proportional-integral (PI) and proportional-derivative (PD) in order to compare the performances in controlling the DC motor position system.

A proportional-integral controller (PI) is a type of controller that is combined with a proportional controller (P) and an integral controller (I). The PI controller is indisputably the most utilized control algorithm in the process control industry. Its popularity arises from its simple and easily implementable structure, and its role as the foundation for various complex control systems, such as adaptive control. However, despite its widespread usage, there is no universally approved controller design process.

Furthermore, a proportional derivative (PD) is a type of controller that is combined with a proportional controller (P) and a derivative controller (D). The PD controller is the output that varies in proportion to the error signal. In industry, PD controllers are used to regulate temperature, flow, speed, and pressure.

In the present day, the usage of DC motors has been widely introduced in various operations such as manufacturing, defense system, and robotics. This is due to their simplicity yet reliability when used in industries. DC motors also can be cost-effective when developing from simple projects to complex research and development in industries. A DC motor's speed controller can be adjusted by adjusting the terminal voltage, but shaft position control is not possible. When establishing a precise position in a precision control system application, position control of a DC motor is critical.

Embedded electronics make it simple to regulate a DC motor. A microcontroller-based position control system is made up of an electrical component and a microprocessor. DC motor drives are utilized in a range of applications where power electronics regulate the voltage and, as a result, the motor's position or speed. Therefore, the PI controller and PD controller will introduce and use in this project. The reason this research does not use a PID controller is that PID controller has been covered lightly in several previous studies. Burbano and Bernardo, for example, the researchers examined consensus and synchronization in complex networks by employing distributed PID control for first-order linear systems. They also suggested a multiplex PI control approach to investigate consensus challenges across a diverse range of complex networks. Next, according to [1], swarm intelligence (SI) represents a form of artificial intelligence that

*CORRESPONDING AUTHOR | Nafrizuan Mat Yahya | nafrizuanmy@ump.edu.my

leverages the collective behavior of self-organized systems and decentralized. On a local level, these systems are often made up of a population of basic actors that interact with one another and with their environment. In comparison to Evolutionary Algorithm (EA) and other single solution-based techniques, SI is a very recent category of evolutionary computing. SI algorithms employ non-deterministic and approximate methods to efficiently explore and exploit the search space, with the goal of finding near-optimal solutions. Swarm-based approaches are the most widely used nature-inspired meta-heuristics. There are several types of swarm intelligence, for example, Artificial Bee Colony (ABC), Artificial Fish Swarm Optimization (AFSO) and Particle Swarm Optimization (PSO). These algorithms have solved many optimization problems.

Modified adaptive bat sonar algorithm (MABSA) is also a type of swarm intelligence. It is a modified version of the adaptive bat sonar algorithm (ABSA), and it will be used as an application for this project study which is to compare the performances of the MABSA-PI controller and MABSA-PD controller for the DC motor position system. It took its inspiration from the bats that live in caves. The bats will emit the sonar to detect the location of their prey and find the way home due to the blindness.

RELATED WORK

This section clarify a brief summary of optimization algorithms, some of which adopt a population-based methodology where multiple agents partake in the search process. The particle swarm optimization (PSO) is one of the optimization algorithms. PSO involves a swarm of agents that collectively search for the global optimal solution by continuously updating their velocities and positions based on their current positions, personal bests, and the global best of the swarm. They converge towards particles with higher fitness values to ultimately attain the best solution.

Another population-centered optimization algorithm available is the genetic algorithm (GA) [2]. Starting from a starting population that represents the problem-solving answer is the fundamental tenet of a genetic algorithm. Selection, crossover, and mutation operations are carried out in accordance with the fitness of the population's members to create a new generation of populations. The new population's members not only carry the genes of the preceding generation but also outperform it in terms of total performance. Genetic operators have an impact on genetic algorithm convergence in addition to population size. The crossover and mutation probabilities' sizes have a direct impact on the convergence of the genetic algorithm as well as how quickly new individuals are produced by the algorithm. The fundamental genetic algorithm uses mutation probability and fixed crossover probability, which prevents these control parameters from self-adjusting in response to the evolution process and hinders the algorithm's capacity to find the best solution [3].

The Bat Algorithm (BA) was created by [4] and is based on how different bat species use echolocation to locate their prey. By combining the generation of the sound pulse and echo recognition time difference, the time delay between the bat's ears and the variation strength of the sound pulse, bats create a three-dimensional environment. The bat can determine the distance, type, orientation of the prey and rate of movement in this way.

The echolocation method used by a colony of bats to discover food or prey is the basis for the development of the bats sonar algorithm (BSA) by [1]. Bats use their sonar during echolocation to determine the target's velocity, distance, size, azimuth, and elevation. The BSA uses echolocation to hunt out the best solution to a particular issue by modelling the principles of bat sonar. One potential solution is represented by each point (prey location discovered) in the search space (certain constrained area). One unit of sonar is designated as a bat.

SIMULATION OF MABSA-PI AND MABSA-PD CONTROLLER

Development and general equations of DC motor mathematical modelling

There are two parts of this circuit of DC motor mathematical modelling which are the electrical part and mechanical part. The input voltage, resistor and inductor are the parts of in electrical part. However, the rotor is the only part in the mechanical part.

According to Kirchoff's Voltage Law (KVL), the electrical equation can describe as

$$E_a = R_a I_a + L_a \frac{d_{ia}}{dt} + E_b \tag{1}$$

where Ea is the voltage source, Ia is the armature current, and E_b is the back emf voltage. The back emf (E_b) is proportionate to the angle of the rotor of the motor and can be written as

$$E_b = K_b \frac{d\theta}{dt} \tag{2}$$

Where K_b is constant of back emf.

Next, mechanical part equation can be derived as

$$J_m \frac{d^2\theta}{dt} + B_m \frac{d\theta}{dt} = T_m \tag{3}$$

Where Bm is the frictional coefficient and J_m is the rotor moment of inertia. Moreover, the motor will produce torque, T_m, which corresponds to the armature current, I_a, and may be represented as

$$T_m = K_t I_a \tag{4}$$

Where K_t is the torque constant.

All the equations above can be transformed into Laplace transform which is the equation will be in complex variable form. The new equation was generated by substituting, rearranging equation (3-3-2) into (3-3-1) and converting it to Laplace transform, as shown below.

$$E_a(s) - K_b s\theta(s) = (R_a + sL_a)I_a(s) \tag{5}$$

The same mathematical calculation will be performed on equations (2) and (3), and a new equation in Laplace transform will be constructed below.

$$J_m s^2\theta(s) + B_m s\theta(s) = K_t I_a(s) \tag{6}$$

Some mathematical calculations and processes will be required to create a transfer function, such as rearrangement and substitution methods from two equations, (5) and (6). The transfer function may be represented using these two equations:

$$\frac{\theta(s)}{E_a(s)} = \frac{K_t}{J_m L_a s^3 + (J_m R_a + B_m L_a)s^2 + K_t K_b s} \tag{7}$$

Development PI controller and PD controller

The PI controller is composed of an integral gain and proportional gain, whereas the PD controller involves of a derivative gain and a proportional gain. Simulink has provided the PID blocks as known as PID controllers. It can be edited or changed what type of controller that user wants. For this project, PI controller and PD controller are the choices to compare the performance for DC motor position system.

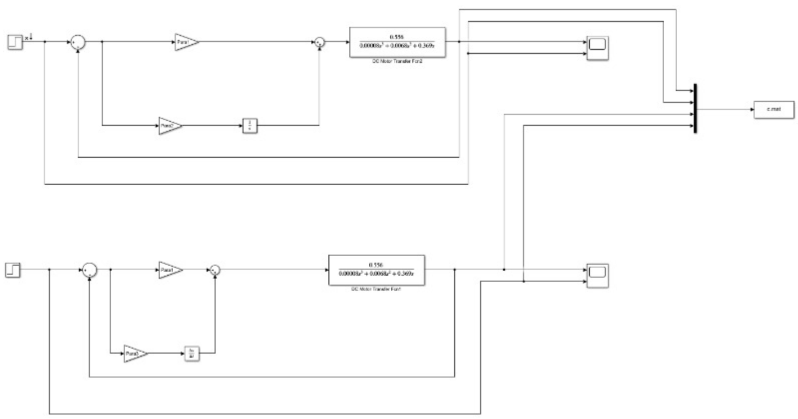

Figure 1. Block diagram of Proportional-Derivative (PD) and Proportional-Integral (PI) controller using Simulink

Gaing's Method Objective Function

This section provided the method to gain the objective function in MABSA-PI and MABSA-PD controller for coding. According to [5], this formula is proposed for evualating the PID controller. However, it can also evaluate the MABSA-PI and MABSA-PD controllers. This is due to the fact that a combination of good control parameters which are k_i, k_p, k_d, can produce a good step response, which will lead to the time domain performance criterion minimization. The steady-state error Ess, rising time tr, overshoot M_p, settling time t_s, are some of the performance benchmarks in the time domain. Furthermore, the following definition for a new performance standard *W(K)* is given:

$$\min K\ stabilizing\ W(K) = \left(1 - e^{-\beta}\right) * (M_P + E_{SS}) + e^{-\beta} * (t_s - t_r) \quad (8)$$

Where K is [k_i, k_p, k_d,], and β is the weighing factor.

The designer can meet their specifications for the performance criterion W(K) by utilizing the weighting factor β. Moreover, to lessen the overshoot and steady-state inaccuracy, we can increase the value β to be greater than 0.7. However, to shorten the rise and settling times, we can set the value β to be smaller than 0.7 and set β between 0.8 to 1.5. For this solely research, the value of β is set at 0.9.

Development of Modified Adaptive Bat Sonar Algorithm (MABSA)

Algorithm 4 Modified adaptive bats sonar algorithm

```
Objective function F(x), x = (x_1, ..., x_d)^T
Initialise: Bats, MaxIter, Dim, SS_Size, NBeam_MAX and NBeam_MIN
for n ← 1 to Bats do
   for d ← 1 to Dim do
      Generate random pos_SP
      Evaluate F_SP value for F(pos_SP)
   end for
end for
Assign the most optimum value as F_GB and its position as pos_GB
while t ≤ MaxIter do
   Define NBeam to transmit by using BNI (Equation 4.4 and Equation 4.5)
   for n ← 1 to Bats do
      for N ← 1 to NBeam do
         for d ← 1 to Dim do
            Set L and limit μ (Equation 5.1 and Equation 4.3)
         end for
      end for
      Generate random θ_m and θ (Equation 4.6)
      Transmit NBeam starting from pos_SP
      for N ← 1 to NBeam do
         for d ← 1 to Dim do
            Determine pos_i for each transmitted beam (Equation 5.2)
            Verify pos_i for each transmitted beam within SS_Size
            if pos_i ≥ SS_Max then
               Update pos_i (Equation 5.3a)
            end if
            if pos_i ≤ SS_Min then
               Update pos_i (Equation 5.3b)
            end if
         end for
         Evaluate F_i value for F(pos_i)
         Assign the optimum value of F_i as F_LB and its position as pos_LB
         if F_LB ≤ F_SP then
            Assign F_LB as F_RB and pos_LB as pos_RB
         else
            Assign F_SP as F_RB and pos_SP as pos_RB
         end if
      end for
   end for
   Select the optimum value among F_RB as current F_GB and its pos_RB as current pos_GB
   if current F_GB ≤ previous F_GB then
      Update current F_GB as new F_GB and current pos_GB as new pos_GB
   else
      Retain previous F_GB and pos_GB
   end if
   for n ← 1 to Bats do
      Determine new pos_SP using (Equation 4.8)
      Evaluate new F_SP value for F(pos_SP)
   end for
end while
Declare F_GB as optimum fitness evaluated and pos_GB as its optimum value(s)
```

Figure 2. MABSA pseudo code [1]

The pseudocode in Figure 2 is the modified version of the ABSA pseudocode. For unconstrained single-objective optimization problems, ABSA was investigated as a superior alternative to BSA. ABSA is used to solve non-constrained single-objective optimization problems. However, in order to address the constraints posed by limited single objective optimization, it is crucial to find a suitable approach for combining equality and inequality constraints with the objective function. As a result, a modified adaptive bat sonar algorithm (MABSA) was developed as a new method. MABSA is investigated to compensate for this difficulty by redefining several aspects in ABSA and reformulating a key component of the bat sonar algorithm (BSA).

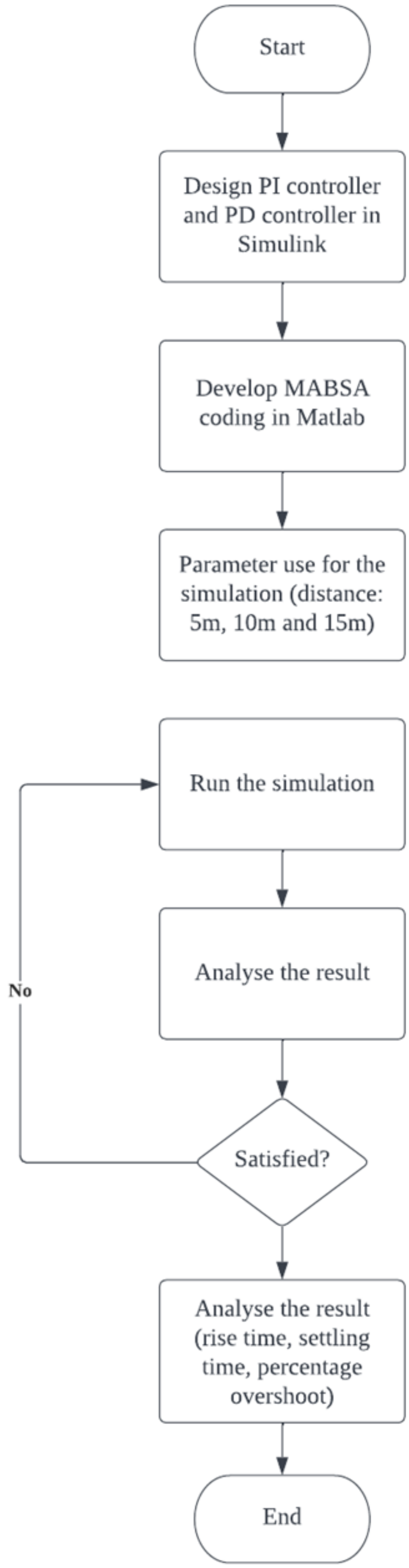

Figure 3. Process flowchart

Performances assessment

In this phase, the assessment of controller performance will involve the comparison of various variables, such as percentage overshoot (%OS), settling time (Ts), and rise time (Tr) for each specific type of controller. Those controllers will be applied with Modified Adaptive Bats Sonar (MABSA) to compare the performances for DC motor position system. Then, the proportional-integral (PI) controller and proportional-derivative (PD) controller would be applied with MABSA to evaluate the performances.

The results of the variables will show in the scope at Simulink. It will show the percentage overshoot, settling time and rise time. Therefore, the comparison of the performances MABSA-PI and MABSA-PD for DC motor position system can be made based on these variables.

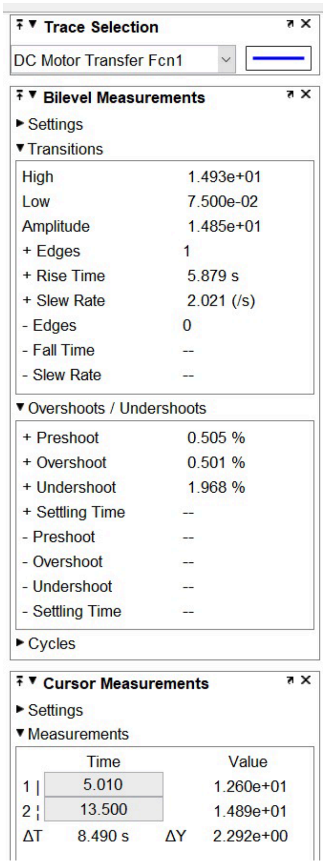

Figure 4. Parameters in scope

EXPERIMENTAL RESULTS

This section uses a step response graph to analyse Simulink simulation of a combination controller and the DC motor transfer function. Furthermore, it would described how MABSA-PI and MABSA-PD controllers compare in terms of percentage overshoot (%OS), settling time (Ts), and rise time (Tr) as in Figure 5 and 6.

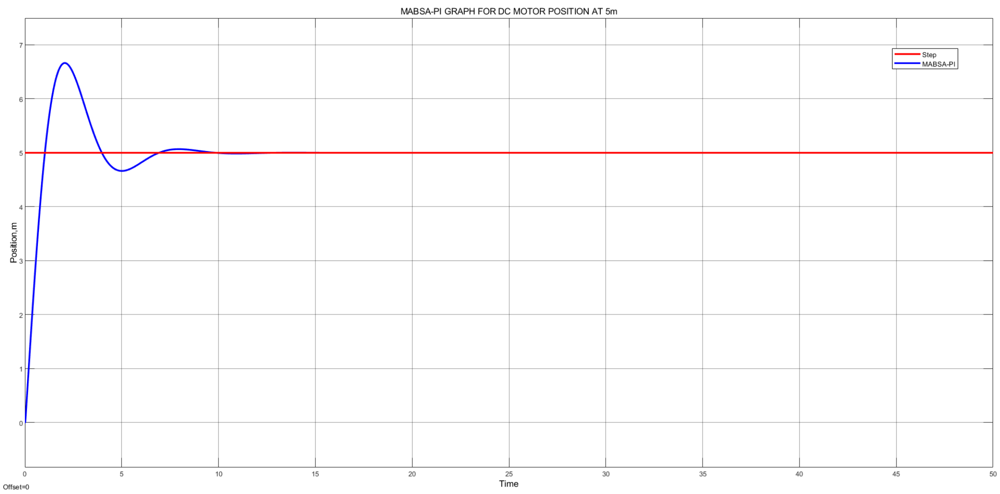

Figure 5. MABSA-PI graph for DC motor position at 5m

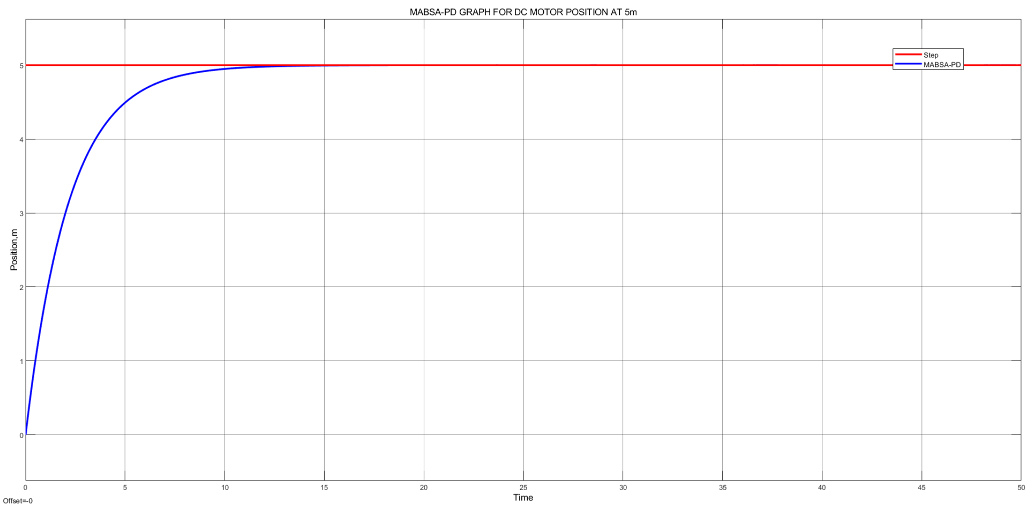

Figure 6. MABSA-PD graph for DC motor position at 5m

Table 1. Transient Response Position of 5m

Transient Response	MABSA-PI	MABSA-PD
Rise time, T_r (s)	0.792	4.7
Settling time, T_s (s)	6.844	12.031
Percentage overshoot, *%OS*	32.667	0.496

CONCLUSION

The initial attempt was proposed by this paper to improve the assessment capability of Proportiona Integral (PI) and Proportional Derivative (PD) controller by applying modified adaptive bats sonar algorithm (MABSA) in both controllers. In addition, the pseudocode of the MABSA is provided as the guideline to develop the coding. The results will be focused on the transient system which is percentage overshoot, rise time and settling time. The graph in Simulink will show the transient system and the performances. There are some aspects of this project that can be enhanced in the future, such its implementation in hardware applications, like the X-Y table position. In addition, this simulation study could be enhanced by adding controller such as PID or fuzzy logic. Last but not least, when it comes to transient reaction, the variables that can be used to compare performances are peak time, time constant, and error.

REFERENCES

[1] Mat Yahya, N., Nor'Azlan, N. A., & Tokhi, M. O. (2019). Parametric Study of Dual-particle Swarm Optimisation-modified Adaptive Bats Sonar Algorithm on Multi-objective Benchmark Test Functions. *MEKATRONIKA*, *1*(2), 72–80. https://doi.org/10.15282/mekatronika.v1i2.4988

[2] Rostami, M., Berahmand, K., Nasiri, E., & Forouzandeh, S. (2021). Review of swarm intelligence-based feature selection methods. *Engineering Applications of Artificial Intelligence*, *100*, 104210. https://doi.org/10.1016/j.engappai.2021.104210

[3] Yahya, N. M., Tokhi, M. O., & Kasdirin, H. A. (2016). A new bats echolocation-based algorithm for single objective optimisation. *Evolutionary Intelligence*, *9*(1–2), 1–20. https://doi.org/10.1007/s12065-016-0134-5

[4] Yang XS. (2010) A New Metaheuristic Bat-Inspired Algorithm. In: González J.R., Pelta D.A., Cruz C., Terrazas G., Krasnogor N. (eds) Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). Studies in Computational Intelligence, vol 284. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-12538-6_6

[5] Zwe-Lee Gaing, "A particle swarm optimization approach for optimum design of PID controller in AVR system," in IEEE Transactions on Energy Conversion, vol. 19, no. 2, pp. 384-391, June 2004, doi: 10.1109/TEC.2003.821821.