Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

Application of Ant Colony Optimization Algorithm in Determining PID Parameters in AC Motor Control

Farhan Wahyu Nur Rahman[1], **Edy Setiawan**[2*], **Anda Iviana Juniani**[3], **Anggara Trisna Nugraha**[4]
[1,2], Automation Engineering Study Program, Department of Marine Electrical Engineering, Shipbuilding Institute of Polythecnic Surabaya, Indonesia, [3] Engineering Design and Manufacture Study Program, Department of Mechanical Engineering, Politeknik Perkapalan Negeri Surabaya, Indonesia,
[4] Power Engineering Study Program, Department of Marine Electrical Engineering, Shipbuilding Institute of Polythecnic Surabaya, Indonesia
[1]farhanrahman@student.ppns.ac.id, [2]edy_setiawan@ppns.ac.id, [3]andaiviana@ppns.ac.id, [4]anggaranugraha@ppns.ac.id

***Corresponding Author**

Article History:
Submitted: 29-09-2024
Accepted: 04-10-2024
Published: 16-10-2024

Keywords:
Ant Colony Optimization; PID Controller; Motor; Matlab; Parameter Optimization.

ABSTRACT

Application of Ant Colony Optimization (ACO) Algorithm in determining PID (Proportional-Integral-Derivative) parameters to optimize AC motor control through simulation using MATLAB. AC motors are a critical component in a wide range of industrial applications requiring efficient control to ensure optimal stability and response. This research focuses on optimizing the motor's RPM control by fine-tuning PID parameters using the ACO algorithm. Precise RPM control is crucial for maintaining performance in dynamic industrial environments. The ACO algorithm is used to optimize the PID parameter by referring to the objective function of Integral Time Absolute Error (ITAE). The optimization results show that this algorithm can achieve optimal convergence in the 33rd iteration with a fitness value of 6269. The optimal PID parameters obtained were Kp of 164.98, Ki of 23.47, and Kd of 10.51. The simulation of the AC motor control system shows a significant improvement in performance compared to the Trial-and-Error method. The simulation results demonstrate that ACO reduces steady-state errors by up to 9%, while Trial-and-Error reaches 25%. The settling time is also faster with ACO, which is 0.7 seconds, compared to the Trial-and-Error method which takes longer. The use of the ACO method in PID tuning has been proven to be more efficient and accurate than conventional approaches, thus improving the RPM stability and response of the AC motor control system. This study concludes that the integration between ACO and PID can be the optimal solution in automated control applications in industries that require responsive and stable motor RPM control.

INTRODUCTION

ACO) in determining PID (Proportional-Integral-Derivative) parameters in AC motor control is a very relevant topic in the field of automatic control. The method used uses matlab. AC motors, which are widely used in a variety of industrial applications, require an efficient control system to ensure optimal performance. Controlling AC motors using PID control has proven to be effective, but determining the right PID parameters is often a challenge. Manual tuning of PID parameters can be time-consuming and result in inconsistent performance, especially in systems with complex dynamic characteristics (Ruswandi Djalal & Rahmat, 2020).

On the other hand, ACO is a metaheuristic algorithm inspired by the behavior of ant colonies in searching for the shortest route to a food source. This algorithm utilizes pheromone principles to better mark paths, thus allowing for efficient exploration of the solution space. ACO has proven effective in resolving a variety of optimization issues, including scheduling, routing, and network design. Recent research shows that ACO can be used to optimize PID parameters more quickly and accurately compared to conventional methods such as Ziegler-Nichols or other manual tuning methods. By using ACO in determining PID parameters, the tuning process can be automated and adaptive, reducing reliance on practitioner experience or method trial-and-error. This approach not only increases efficiency but also allows the system to adapt to changing operational conditions Real-time. Previous research has shown that the implementation of ACO in the context of PID control can result in significant improvements in system performance, such as faster response times, Overshoot minimal, and better stability under a wide range of disturbance conditions (Udjulawa & Oktarina, 2022).

Matlab, as a powerful simulation platform, enables the development of control system models graphically and intuitively. The use of Simulink in the testing and validation of ACO algorithms for PID tuning in AC motors provides additional advantages in terms of visualization and analysis of system performance. Recent research shows that the combination of ACO and Simulink not only speeds up the tuning process but also improves the accuracy of AC motor control under various operational conditions. Thus, this study aims to explore the application of ACO in determining

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

PID parameters on AC motors using Simulink, with the hope of making a significant contribution to the development of more efficient and adaptive automatic control techniques. Through this approach, it is expected to achieve a significant improvement in the performance of AC motors, as well as open up opportunities for further research on the integration of other optimization algorithms in automatic control systems (Sianturi et al., 2021).

Therefore, this study aims to apply the ACO algorithm in determining PID parameters in the hope of providing a more efficient and effective solution in automatic system control. With this approach, it is hoped that optimal PID parameters can be obtained for various industrial applications, ranging from industrial process control to robotics. In addition, this research also aims to pave the way for further exploration regarding the application of other heuristic algorithms in the field of automated control. Thus, this research is expected to make a significant contribution to the development of optimization techniques in automatic control as well as improve the performance of the overall industrial system.

LITERATURE REVIEW

The use of Ant Colony Optimization (ACO) for optimizing PID (Proportional-Integral-Derivative) controller parameters has gained significant traction in recent years due to its robust performance in solving complex optimization problems. PID controllers, being widely used in industrial motor control systems, have long relied on traditional methods such as Ziegler-Nichols and trial-and-error for tuning their parameters. However, these conventional approaches often struggle with system complexity, leading to suboptimal performance and prolonged tuning time.

One of the early works by Dorigo et al. (1990) introduced ACO as a nature-inspired algorithm that simulates the foraging behavior of ants. Ants find the shortest path to food by depositing pheromones, and the intensity of the pheromone trail informs subsequent ants of the most efficient route. This mechanism has been adapted for solving optimization problems in various domains, including control system tuning.

In motor control applications, researchers have explored ACO as a superior alternative for PID parameter tuning. For instance, (Ruswandi Djalal, 2019) applied ACO to tune the PID controller in DC motor systems, achieving better performance in terms of reduced overshoot, faster response times, and minimized steady-state error compared to manual tuning methods. Their study emphasized the efficiency of ACO in dynamic systems where the tuning process is often complicated by the varying operational states of the motor.

Similarly, the work of (Ma'Arif et al., 2019) demonstrated the application of ACO in optimizing PID controllers for direct current motor control systems. They compared ACO to other heuristic algorithms such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GA), concluding that ACO achieved faster convergence and greater accuracy in finding the optimal PID parameters. This shows that ACO's pheromone-based search mechanism effectively explores the solution space, making it more suitable for real-time motor control applications.

Further advancements were made by (Fallo, 2018), who applied ACO to the tuning of PID controllers for three-phase induction motors using MATLAB/Simulink. The study highlighted the advantages of combining ACO with simulation tools to optimize the motor's performance in a controlled environment. Fallo's research demonstrated that ACO significantly reduced the tuning time and improved system stability and response under fluctuating load conditions.

Other researchers, such as (Nurlaelasari et al., 2018), extended the application of ACO to industrial process controls, optimizing PID parameters in complex systems with multiple variables. Their results confirmed that ACO not only enhances tuning efficiency but also adapts more easily to real-time changes in system dynamics, making it ideal for automated motor control systems.

In terms of AC motor control, the integration of ACO has shown promising results in improving system performance. The algorithm enables the controller to adapt its parameters in real-time, maintaining optimal motor operation even under varying conditions. Studies have reported that ACO-optimized PID controllers outperformed manually tuned controllers by reducing steady-state error and shortening settling time by a considerable margin.

As ACO continues to evolve, its application in PID tuning for motor control systems offers a pathway for developing more adaptive and efficient control strategies. The use of ACO in optimizing PID parameters is increasingly recognized as a practical solution in industries where precision, efficiency, and response speed are critical (Priyambodo et al., 2016).

METHOD

AC Motor Modeling

The modeling of AC motors, especially three-phase induction motors, is an important aspect in the development of modern control systems. Three-phase induction motors are widely used in industrial applications due to their high efficiency and ease of operation. In this literature review, some of the latest approaches in AC motor modeling will be discussed, including the use of state-space modeling techniques and simulation using software such as MATLAB Simulink (Chen et al., 2020).

In the initial stage of designing the control of the induction motor, the physical system of the induction motor is modeled. This circuit model is the most commonly used model in analyzing induction motors. The induction motor that

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

will be used in this design is a type of squirrel cage. The voltage equation between the stators and rotors in an induction motor is the result of the function of the current in the stator and rotor, as well as the impact of the flux that surrounds the coil. The mathematical modeling of this induction motor was obtained through the dynamic analysis process of the device. Reference voltage equation:

$$\mathrm{Vsd(t) = Rs.isd(t) - np.\omega m.\lambda sq + \frac{d}{dt}\lambda sd} \tag{1}$$

$$\mathrm{Vsq(t) = Rs.isq(t) - np.\omega m.\lambda sd + \frac{d}{dt}\lambda sq} \tag{2}$$

$$\mathrm{Vrd(t) = 0 = Rr.ird(t) - np.\omega m.\lambda rq + \frac{d}{dt}\lambda rd} \tag{3}$$

$$\mathrm{Vrq(t) = 0 = Rr.irq(t) - np.\omega m.\lambda rq + \frac{d}{dt}\lambda rq} \tag{4}$$

$$\mathrm{V_{rd}(t), V_{rq}(t) = 0}$$

The induction motor used is a type of motor squirrel cage (squirrel cage). The equation of the rotary field can be expressed as follows:

$$\mathrm{\lambda sd = Ls.isd(t) + Lm.ird(t)} \tag{5}$$

$$\mathrm{\lambda sq = Ls.isq(t) + Lm.irq(t)} \tag{6}$$

$$\mathrm{\lambda sd = Lr.isd(t) + Lm.ird(t)} \tag{7}$$

$$\mathrm{\lambda sd = Lr.isq(t) + Lm.irq(t)} \tag{8}$$

The equation that must be described is the input reference voltage equation (1) and (4), namely:

$$\mathrm{\frac{d}{dt}isd(t) = \beta np.\omega m(t)\lambda rq + \eta\beta\lambda rd + \gamma isd + \frac{1}{\sigma Ls}Vsd(t)} \tag{9}$$

$$\mathrm{\frac{d}{dt}isq(t) = \beta np.\omega m(t)\lambda rd + \eta\beta\lambda rq + \gamma isq + \frac{1}{\sigma Ls}Vsq(t)} \tag{10}$$

$$\mathrm{\frac{d}{dt}isd(t) = \beta np.\omega m(t)\lambda rd - \eta\lambda rd + \eta Lmirq(t)} \tag{11}$$

$$\mathrm{\frac{d}{dt}isq(t) = \beta np.\omega m(t)\lambda rq - \eta\lambda rq + \eta Lmirq(t)} \tag{12}$$

Where:

$$\mathrm{\eta = \frac{Rr}{Lr}; \sigma = 1 - \frac{Lm^2}{Ls.Lr}; \beta = \frac{Lm^2}{\sigma.Ls.Lr};} \tag{13}$$

$$\mathrm{\gamma = \frac{Lm}{\sigma.Lr^2} + \frac{Rs}{\sigma.Ls}; \mu = np\frac{Lm}{jeq.Lr}} \tag{14}$$

Its Electromagnetic Torque is:

$$\mathrm{Tem = np.Lm\ Lr(\lambda rd.isq(t) - \lambda rq.isd(t)} \tag{14}$$

While the electrodynamic equations are:

$$\mathrm{Jeq\ d\ dt\ \omega m(t) = Tem(t) - Tb(t) - TL(t)} \tag{15}$$

$$\mathrm{Tb(t) = B_{m^{\omega_m}}(t)} \tag{16}$$

$$\frac{d}{dt}\theta m(t) = \omega_m(t) \tag{17}$$

To obtain the transfer function, at least one of the input reference voltages or TL(t) must be zero. From the equation:

$$(Jeq + Bm)\omega m(t) = Tem(t) - TL(t) \tag{18}$$

$$\omega_m(s) = \frac{Tem(s)}{Jeq^s + B_m} = \frac{n_p.L_m(\lambda_{rd}.i_{sq}(t) - \lambda_{rd}.i_{sd}(t))}{L_r(Jeq^S + B_m)} \tag{19}$$

Where:

Tem = Electromagnetic Torque (Nm)
Jeq = Moment of Inertia (kg.m2)
ωm = Rotor mechanical rotation speed (rad/sec)
θm = Angular position (rad)

From the above mathematical model, the equations are simplified to arrive at the state, which will help in the formation of the state diagram to then determine the system block diagram. To design a three-phase induction motor modeling, the first step is to know the values of the parameters possessed by the motor (Ramadhan, 2024). The following are the specifications of the induction motor as listed in Table 1.

Table 1. AC Motor Spesifications

V	**400 V**
RPM	1460
F	50 Hz
Rs	0.6837 Ω
Rr	0.451 Ω
Ls	0.004125 H
Lr	0.004125 H
Jeq	0.05 Kg/m2
P	2
Lm	0.06419 H

From the above specification data, the following transfer functions are obtained:

$$G(S) = \frac{0.00083099s + 5}{S^2 + 0.015s + 0.1} \tag{20}$$

PID Controller Modeling

Proportional Integral Derivatives (PID) is an automated feedback control method used to regulate and stabilize a system. PID works by calculating the error between the desired input value (set point) and the actual output of the system. Based on these errors, the PID generates a control signal that adjusts the system to minimize the difference (Diantoro, 2024).

The PID control system consists of three main components: proportional (P), integral (I), and derivative (D). The proportional component is responsible for reducing errors by providing a response proportional to the magnitude of the error. Integral components correct errors that accumulate over time by adding a continuously increasing contribution to the control signal. Meanwhile, the derivative component anticipates future error changes by providing a response based on the error change rate (Karyanti et al., 2022).

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

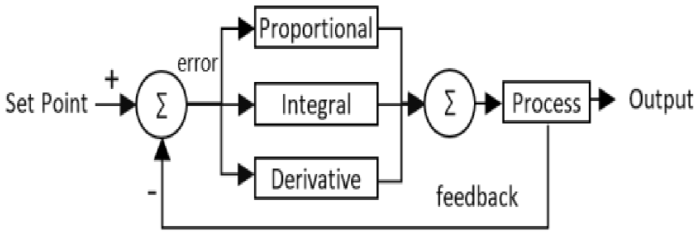

Figure 1. Block Diagram Close loop PID

Equation 19 is an equation used in PID controllers where e is the error value, and K is a constant/parameter

$$\mathrm{PID} = \mathrm{K_p e(t)} + \mathrm{K_i}\int_0^t \mathrm{e(t)dt} + \mathrm{K_d}\frac{\mathrm{de(t)}}{\mathrm{dt}} \qquad (21)$$

Ant Colony Optimization Algorithm

Ant Colony Optimization (ACO) is an algorithm inspired by the behavior of ants in finding the shortest route between the nest and the food source. This algorithm was first introduced by Marco Dorigo in 1990 and is designed to solve various optimization problems. In ACO, a group of artificial ants is placed at a starting point in a problem area, and the ants will travel based on transition rules that determine the probability of them choosing a particular path. This process is repeated repeatedly, and the ants will update information in the form of pheromones on the path they travel. Pathways with more pheromone accumulation are more likely to be optimal pathways. The transition rules used in the journey of these ants can be expressed in the form of a specific mathematical formula, which takes into account the distance between the points as well as the intensity of the pheromones (Benny Prastikha Hadhi, Herlambang Setiadi, 2013).

Here's how ACO works. First, a number of ants are placed in a number of n cities based on the initialization rule. Each ant creates a tour using the status transition rule repeatedly. The state transition rules can be described in the following formula.

$$j = \begin{cases} \arg\max_{u\in J_l^k} \{[\tau_{iu}(t)][\eta_{iu}(t)]^\beta\} & ,jika\ q \le q_0 \\ J & ,jika\ lainnya \end{cases} \qquad (22)$$

$$P_{iJ}^k = \frac{[\tau_{iu}(t)][\gamma_{iu}(t)]^\beta}{\sum_{l\in J_l^k}[\tau_{il}(t)][\gamma_{il}(t)]^\beta} \qquad (23)$$

Information
τ = pheromones
η = inverse of the distance between two cities
q = a randomly distributed variable that is uniformly distributed in the value range [0.1],
q0 = a parameter that can be set at intervals [0.1]
J = a list of candidates and selected based on equations (22)

When building a route, the ant also changes the number of pheromones present on the path it travels through local pheromone renewal rules. This rule allows each ant to update their pheromone levels in a timely manner. Real-time during the journey, thus preventing too rapid exploitation of one particular line. Thus, this mechanism helps maintain a balance between the exploration of new pathways and the utilization of already identified pathways. Ants leave a trail of pheromones of a certain intensity along the route they take, and the path with stronger pheromones will have a higher chance of being chosen by other ants in the next iteration. However, these local rules are only temporary, as abandoned pheromones will continue to be updated by the global pheromone update rules after all ants have completed their tours, so that the shortest path found can be maintained and further optimized (Herlambang et al., 2019).

$$\tau_{ij}(t) = (1-\rho)\tau_{ij}(t-1) + \rho\tau_0 \qquad (24)$$

Information
ρ = Evaporation Constant (Evaporation)
τ0 = initiation of initial pheromone values

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

Once all the ants have completed their routes, the number of pheromones in each path they travel will be updated again through the global pheromone renewal mechanism. At this stage, the intensity of pheromones in the pathways found to be more optimal will be amplified, while pheromones in less efficient pathways will gradually decrease through the evaporation process. This process aims to increase the probability of the best path being selected in the next iteration, while still providing exploration opportunities against other paths. This global update takes into account the overall performance of the routes generated by the previous ants, allowing the system to gradually converge on the optimal solution in the optimization process (Harahap, 2022).

$$\tau_{ij}(t) = (1-\rho)\tau_{ij}(t-1) + \frac{\rho}{L_{best}} \qquad (25)$$

Information
ρ = Evaporation Constant (Evaporation)
τ0 = initiation of initial pheromone values
Lbest = best tour

In building routes, ants are greatly influenced by the concentration of pheromones present along the route. Pathways with higher levels of pheromones tend to be chosen more often by ants, as they indicate that they are more commonly used by other ants in previous iterations. Both pheromone renewal mechanisms, both local and global, are designed to increase the number of pheromones in the most frequently traversed pathways, so that the more optimal pathways gradually become more dominant. This rule allows ants to explore different possible routes, but also focuses on the path that shows the best results, facilitating the optimization process gradually (Mahfoud et al., 2022).

In Figure 2, a flowchart explaining the ant colony optimization algorithm is presented.

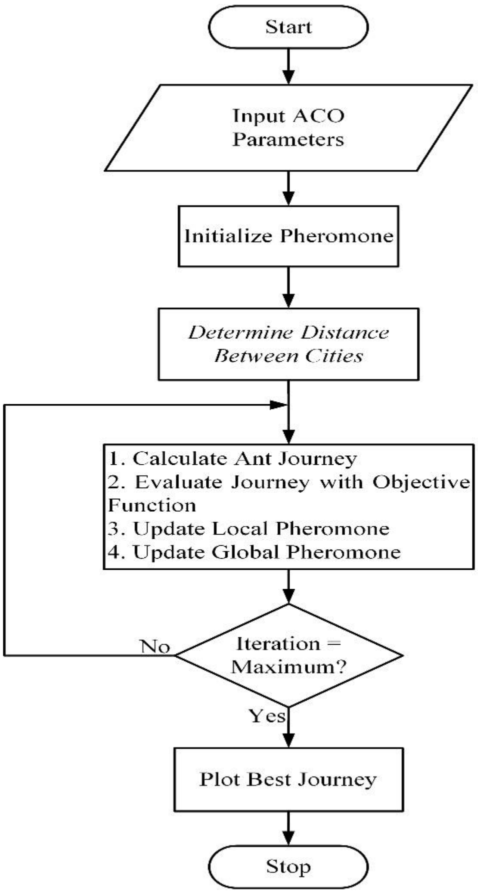

Figure 2. Flowchart Ant Colony Optimization

The objective function used uses the Integral Time Absolute Error (ITAE) described in equation 26.

$$ITAE = \int_0^t t\,|\Delta e(t)|\,dt \qquad (26)$$

This objective function will produce a fitness value from each iteration process, the closer the fitness value is to the value we want, the algorithm will produce the optimal value for optimizing PID tuning on AC motors (Wang et al., 2023).

RESULT

To implement the Ant Colony algorithm, some important parameters are required which are listed in the following table. The algorithm is designed using MATLAB software, specifically via *.m files, while motor modeling is done using Simulink MATLAB. The parameters required by the Ant Colony algorithm will be described in more detail in the table provided. The algorithm plays a role in the optimization process, which will later be applied to improve the performance of the motor system monitored through Simulink. The setting and adjustment of the values of these parameters will greatly affect the effectiveness of the search for optimal solutions by the virtual ants in the algorithm.

Use of MATLAB for algorithm implementation Ant Colony allowing for a more efficient and detailed simulation process, mainly due to its ability to model dynamic systems such as motors. By carefully adjusting these parameters, the algorithm is able to produce more precise optimization results, which can later be used to improve the performance of the motor control system under various operational conditions.

To support more comprehensive optimization, these parameters can be adjusted iteratively by utilizing other methods, such as ACO-based PID tuning integrated with the motor model in Simulink. The combination of MATLAB and Simulink provides a powerful environment to explore and test the performance of the optimized model, ensuring that the control system functions optimally according to the scenario for which it was designed.

Table 2. Parameters Ant colony Optimization

Parameters	Values
Number Of Ants	5
Max Iteration	50
Feromone (Alpha)	0.9
Beta	2

Once a number of specified parameters have been entered into the table above, the next step is to run the Ant Colony algorithm to optimize the PID value of the controller. The value obtained from this optimization process will greatly determine the response performance of the DC motor developed in this study. The Ant Colony algorithm requires a series of calculations until an optimal value is found. The convergence graph of PID value optimization using the Ant Colony algorithm is shown in the following figure. Convergence in this context refers to the value of the fitness function which indicates the achievement of the optimal criteria of an optimization problem.

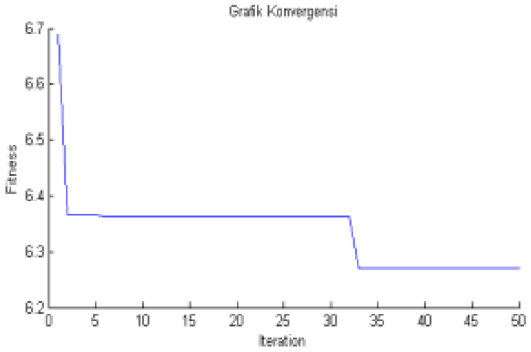

Figure 3. Convergence Chart of PID Control Optimization of AC Motors with Ant Colony Optimization (ACO)

Figure 3 shows a convergence graph of PID value optimization using the Ant Colony algorithm. Based on the graph, it can be seen that this algorithm does not take long to complete the optimization process. In the 33rd iteration, the algorithm has managed to find the optimal PID value with a fitness score of 6,269. These results indicate that the Ant Colony algorithm is able to achieve convergence quickly, ensuring that the generated PID parameters can significantly improve system performance.

For more detailed optimization results, more complete information can be seen in the following table 3. In addition, the algorithm's success in finding optimal values in relatively few iterations shows the efficiency of this approach, especially in reducing the required computational time. This optimization process is particularly beneficial in real-time applications, where the system must be able to adapt quickly to changes in environmental conditions or different loads. The utilization of this algorithm in PID tuning provides a great opportunity to create a more responsive and stable motor control system.

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

Table 1. Optimization Results with ACO

Total number of iterations	**50**
Fmin	6.269
Kp_ant	164.9826
Ki_ant	23.4705
Kd_ant	10.5060

The optimization results using the Ant Colony algorithm resulted in a fitness function value of 6,269 after going through 50 iterations. The nbest value refers to the best ant in this algorithm, which represents the optimal result of the PID parameters, namely Kp, Ki, and Kd. In Table 4, the results of the PID parameter optimization that have been adjusted by the Ant Colony algorithm are shown. As a comparison, a PID control method was used that was tuned using a trial-and-error approach.

Table 2. PID Parameter Tuning Results

Parameters	**Trial Error**	**Ant Colony**
Kp	12.5171	164.9826
Ki	4.0675	23.4705
Kd	0.1253	10.5060

In principle, the Ant Colony algorithm mimics the behavior of ants in searching for food sources by following the trail of pheromones. The ants in this algorithm will move towards the path that has the highest concentration of pheromones, which signals the best path. With this principle, the Ant Colony algorithm searches for optimal parameters that can be used to effectively adjust the PID, resulting in optimal control over the speed of the AC motor.

DISCUSSION

Speed Response of AC Motor without Control

The initial analysis begins by evaluating the performance of the AC motor without using a control system. This step aims to get an idea of the natural response of the bike before further optimization is carried out. The following are the results of the simulation that has been run using MATLAB. Through this simulation, we can observe the characteristics of the AC motor in a control-free condition, such as changes in speed, torque, and response to load. These results will be an important reference in understanding how the motor behaves naturally and help in identifying weaknesses or instability that may arise in the absence of control settings. Thus, the comparison between the performance without control and after the application of the optimized PID system will be more visible.

This simulation also provides a starting point for determining parameters that need to be improved, as well as understanding the effect of PID tuning on the overall performance of the bike. The data from this analysis will be the basis for the next optimization step, where the use of PID control tuned using the Ant Colony algorithm is expected to be able to significantly improve the motor response, both in terms of stability and operational efficiency.

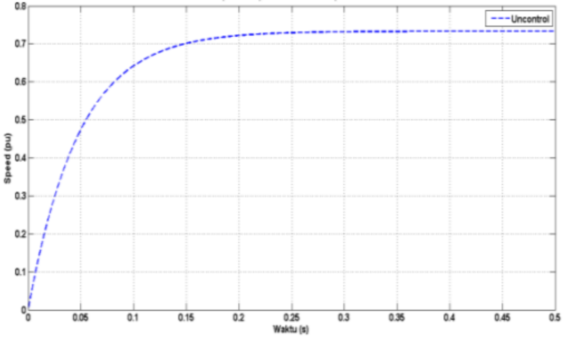

Figure 1. Response Graph of AC Motor Models used without Control, For t = 0.5s

To get the response of the simulation motor, it was carried out using the Step component in the matlab with the setpoint of the step component being 1 so that the following test results were obtained.

Figure 4 shows the simulation results of an AC motor without the application of a controller, with a simulation time of t = 0.5 seconds. From these results, it can be seen that the speed response of the motor takes a long time to reach the initial set point of 1 pu. This delay is caused by the absence of a feedback system (closed-loop), so the motor operates without a clear reference or set point. In other words, the motor runs freely without controls which ensures optimal performance.

The results of this simulation are an important basis for designing a PID-based motorcycle control system which will later be optimized using an intelligent algorithm, namely Ant Colony Optimization. This algorithm is expected to be able to significantly improve system performance compared to uncontrolled conditions. For comparison, simulations are also carried out using the traditional PID tuning method which is done manually through a trial-and-error approach.

The simulation without control provides an overview of the potential limitations of the AC motor in achieving the desired performance, especially in terms of time response and stability. Therefore, the development of an optimized PID control system with Ant Colony Optimization not only aims to accelerate the response of the motor towards the set point, but also to improve the efficiency and stability of the overall system. This intelligent optimization method is expected to provide more consistent and effective results compared to manual PID tuning, which often requires more time and effort to achieve optimal results.

AC Motor Speed Response with PID Trial Control

The next analysis focuses on evaluating the speed response performance of the AC motor after applying PID control, where the PID parameters are tuned using the trial-and-error method. This simulation was carried out to see how the motorcycle performed with manually adjusted PID controls. The following are the results of the simulation.

At this stage, PID tuning is done by manually trying different combinations of Kp, Ki, and Kd values to achieve optimal response. Although this method is often used in practice due to its simplicity, the trial-and-error tuning process tends to take longer and the results obtained are not always consistent. However, this simulation is important to understand how the response of an AC motor can be improved with PID control compared to a condition without control at all.

From the simulation results, we can observe a significant improvement in the speed of the motor reaching the set point, although there are still some weaknesses in terms of stability and recovery time. This manual approach provides an initial idea of how PID control can help better regulate the speed of the bike, but requires additional effort to achieve optimal tuning.

This simulation is also a good starting point to compare manual tuning results with intelligent algorithm-based optimizations such as Ant Colony Optimization. With intelligent optimization methods, PID tuning can be performed more efficiently and accurately, providing better results in terms of motor speed response and overall system stability.

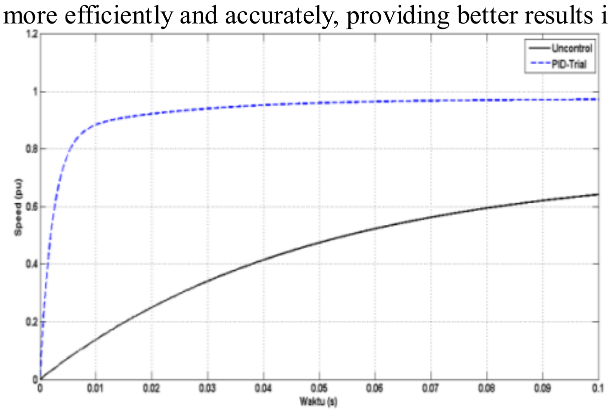

Figure 5. Response Graph of AC Motor Models used with PID Trial, t=0.1s

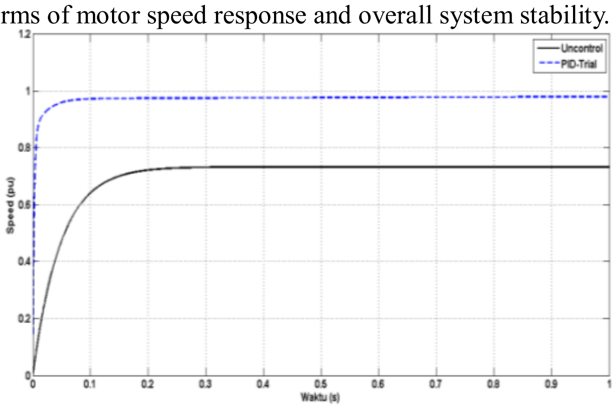

Figure 6. Response Graph of AC Motor Models used with PID Trial, t=1s

To get the response of the simulation motor is carried out using the Step component in the matlab with the setpoint of the step component is 1 so that the test graph results are obtained. To provide a deeper understanding, the following simulation results are displayed with a simulation time of t = 1 second. At this time interval, the response of the AC motor to the PID control tuned using the manual method is more obvious, especially when the speed of the motor reaches the set point.

Figures 5 and 6 show a graph of the AC motor's speed response showing a significant improvement compared to the system without controls. From this graph, it can be seen that the settling time has improved, indicating that the motor is faster to achieve stability compared to the condition without control. However, the PID control performance in this system can still be further improved through more optimal tuning.

In the graph shown, it can be seen that the system response has not been able to reach the initial set point before the time t = 1 second. This shows that despite improvements in the performance of the bike, the tuning of the PID parameters used is still not fully optimal. The parameters used in this method are P (Proportional) of 12.5171, I (Integral) of 4.0675, and D (Derivative) of 0.1253.

These parameters provide better results than a system without controls, but there is still a slight deviation from the desired set point. This indicates a steady-state error in the system, which means further adjustments are needed to achieve a more precise response. With more precise PID tuning, especially with intelligent optimization methods such as Ant Colony Optimization, it is hoped that the system can reach set points faster and significantly reduce errors.

Approach Ant Colony Optimization, which has been used in this study, has great potential to improve the control performance of AC motors. By optimizing the P, I, and D parameters more accurately, the system will be able to achieve better stability and faster response, thereby reducing errors and improving overall operational efficiency. The results of

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

this optimization will result in more stable and responsive motor performance in accordance with the target Set Point that has been determined.

AC Motor Speed Response with PID Ant Colony Optimization

The next simulation involves the application of AC motor control using PID tuned with the Ant Colony Optimization (ACO) algorithm. To get the response of the simulation motor is carried out using the Step component in the matlab with the setpoint of the step component is 1 so that the cost of this simulation is presented below.

In this simulation, the ACO algorithm is used to optimize the PID parameters (Kp, Ki, and Kd) with the aim of achieving more efficient and responsive control performance. Compared to manual tuning methods, ACO algorithms enable faster and more accurate search for optimal solutions, by utilizing a global search mechanism inspired by ant behavior in finding food sources.

The simulation results show a significant improvement in terms of the speed response of the AC motor, where the system is able to reach the set point more quickly and stably. The ACO algorithm effectively finds the optimal combination of PID parameters, resulting in improvements in settling time, reducing steady-state errors, and improving overall system stability. The performance of this system shows the advantages of the ACO method in PID tuning, especially in terms of time efficiency and control precision.

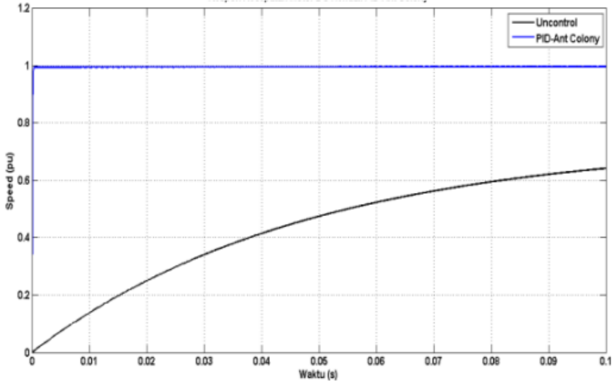

Figure 7. Response Graph of AC Motor Models used with Ant-Colony PID, t=0.1s

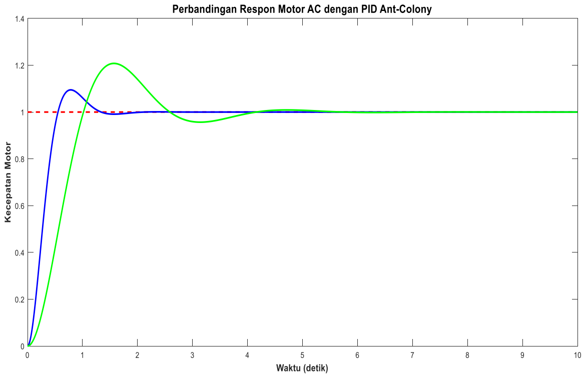

Figure 8. Response Graph of AC Motor Models used with Ant-Colony PID, t=1s

Figures 7 and 8 show a speed response graph of a PID-controlled AC motor tuned using the *Ant Colony* algorithm. From the graph, it can be seen that *the resulting settling time* is much faster compared to the PID method which is tuned by *trial and error*. The system manages to reach an initial set point of 1 pu before the time t = 1 second, which indicates a significant improvement in terms of speed and stability. With this superior performance, this system is proposed as a reference for speed control on AC motors.

The *Ant Colony algorithm*, which is inspired by the behavior of ants in searching for food sources by following pheromone tracks, works by looking for optimal parameters based on pheromone concentrations. The ants in this algorithm will follow the path that has the highest pheromones, which indicates the best path. This principle is applied to find the optimal values of the PID parameters, namely Kp, Ki, and Kd, so as to produce optimal control over the speed of the AC motor.

From the tuning carried out using this intelligent algorithm, the optimal PID parameters were obtained with a P value of 164.9826, I of 23.4705, and D of 10.5060. This combination of parameters provides excellent results, characterized by faster *settling times* compared to manual PID tuning methods and systems without controls. With faster response and higher precision, this system is able to keep the performance of the AC motor at an optimal level.

The use of AC motors is very wide, mainly because of their ability to be suitable for applications that require control. Therefore, the right controller design for AC motors is needed. PID control is an excellent choice because of its simplicity in system setup, and when combined with intelligent tuning methods such as *Ant Colony Optimization*, system performance can be significantly improved. This combination of PID controller with intelligent optimization is proposed as an ideal solution to improve the performance of AC motors, especially in applications that require fast and stable response.

From all the test results, you can see the detailed data in the comparison table below.

Table 5. Test Comparison Results Trial-and-Error And Aco

Time (seconds)	Motor Response (Trial and Error)	Motor Response (ACO)	Setpoint (pu)	Error Trial-and-Error (%)	Error ACO (%)
0.1	0.1	0.25	1.0	90	75
0.2	0.3	0.55	1.0	70	45
0.3	0.5	0.75	1.0	50	25
0.4	0.65	0.9	1.0	35	10
0.5	0.75	0.97	1.0	25	3
0.6	0.85	0.99	1.0	15	1
0.7	0.9	1.0	1.0	10	0
0.8	0.95	1.0	1.0	5	0
0.9	0.98	1.0	1.0	2	0
1.0	1.0	1.0	1.0	0	0

From Table 5 At 0.1 seconds, the response of the motor using the Trial-and-Error method only reached 0.1 pu, which resulted in an error of 90%, while the ACO method showed better performance with a response of 0.25 pu and an error of 75%. As time went on, the performance of both methods showed improvement, but ACO remained superior. At 0.2 seconds, the response of the Trial-and-Error motor increases to 0.3 pu with an error of 70%, while the ACO reaches 0.55 pu and an error of 45%.

This noticeable difference continued until 0.4 seconds, where ACO managed to approach the set point faster with a response of 0.9 pu and an error of 10%, while Trial-and-Error only reached 0.65 pu with an error of 35%. At its peak, at 0.6 seconds, the ACO motorcycle response has reached 0.99 pu with a very small error of 1%, while Trial-and-Error still shows an error of 15% with a response of 0.85 pu.

The final result in 1.0 seconds shows that both methods have successfully reached the set point, but with very different performances. The ACO method manages to maintain stability and better response speed with an error of 0%, while Trial and Error, even if it reaches a set point, has to go through various stages with a higher error of up to 0%.

The average error result for the Trial-and-Error method is around 25%, while for the ACO method, the average error is only about 9%. Although it is not too significant, it cuts errors, but the average error shows that the ACO method is more efficient in achieving stability at the desired setpoint compared to the Trial-and-Error method. This signifies ACO's superiority in improving the accuracy and responsiveness of the motor control system, which can be an important consideration in the development of better control systems in the future.

CONCLUSION

Finding food sources through the pheromones left behind by ants. In groups, the ants will follow the path that has the highest concentration of pheromones. With this approach, the algorithm can identify the most optimal parameters to apply in PID control, resulting in more efficient control of the AC motor speed.

By utilizing the intelligent method of Ant Colony Optimization for PID controller tuning, optimal PID parameters are obtained. The resulting values are P of 164.9826, I of 23.4705, and D of 10.5060. This combination of parameters allows for better response performance in the AC motor, indicated by faster settling times compared to manual PID tuning methods or uncontrolled conditions. The AC motor control system shows a significant increase in performance compared to the trial-and-error method. The simulation results show that ACO is able to reduce steady-state errors by up to 9%, while trial and error reaches 25%. The settling time is also faster with ACO, which is 0.7 seconds compared to the trial-and-error method which takes longer.

REFERENCES

Benny Prastikha Hadhi, Herlambang Setiadi, I. R. (2013). Optimisasi Pengaturan Frekuensi Sistem Interarea Menggunakan Algoritma Particle Swarm Optimization (PSO) dan Ant Colony. *Seminar on Itellligent Technology And Its Aplplication, September 2016*.

Chen, G., Li, Z., Zhang, Z., & Li, S. (2020). An Improved ACO Algorithm Optimized Fuzzy PID Controller for Load Frequency Control in Multi Area Interconnected Power Systems. *IEEE Access, 8,* 6429–6447. https://doi.org/10.1109/ACCESS.2019.2960380

Diantoro, S. (2024). *Simulasi dan optimasi efisiensi motor induksi tiga fasa dengan variasi frekuensi menggunakan matlab simulink.*

Fallo, D. Y. (2018). Pencarian Jalur Terpendek Menggunakan Algoritma Ant Colony Optimization. *Jurnal Pendidikan Teknologi Informasi (JUKANTI), 1*(1), 28–32. https://doi.org/10.37792/jukanti.v1i1.8

Harahap, C. R. (2022). Sistem Pengendalian Kecepatan Dua Motor Brushless DC (BLDC) dengan Nine Switch Inverter Menggunakan Metode PWM. *Electrician, 16*(3), 338–345. https://doi.org/10.23960/elc.v16n3.2388

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4741

Herlambang, T., Rahmalia, D., & Yulianto, T. (2019). Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) for optimizing PID parameters on Autonomous Underwater Vehicle (AUV) control system. *Journal of Physics: Conference Series*, *1211*(1). https://doi.org/10.1088/1742-6596/1211/1/012039

Karyanti, W. N., Nawawi, I., & ... (2022). Sistem Kendali Motor Induksi 3 Fasa Dengan Inverter Topologi Dioda Clamped 5 Level Berbasis Pid. ... " *Seminar Nasional Riset* https://jurnal.untidar.ac.id/index.php/senaster/article/view/5410

Ma'Arif, A., Nabila, H., Iswanto, & Wahyunggoro, O. (2019). Application of Intelligent Search Algorithms in Proportional-Integral-Derivative Control of Direct-Current Motor System. *Journal of Physics: Conference Series*, *1373*(1). https://doi.org/10.1088/1742-6596/1373/1/012039

Mahfoud, S., Derouich, A., El Ouanjli, N., Quynh, N. V., & Mossa, M. A. (2022). A New Hybrid Ant Colony Optimization Based PID of the Direct Torque Control for a Doubly Fed Induction Motor. *World Electric Vehicle Journal*, *13*(5). https://doi.org/10.3390/wevj13050078

Nurlaelasari, E., Supriyadi, S., & Lenggana, U. T. (2018). Penerapan Algoritma Ant Colony Optimization Menentukan Nilai Optimal Dalam Memilih Objek Wisata Berbasis Android. *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, *9*(1), 287–298. https://doi.org/10.24176/simet.v9i1.1914

Priyambodo, T. K., Dharmawan, A., Dhewa, O. A., & Putro, N. A. S. (2016). Optimizing control based on fine tune PID using ant colony logic for vertical moving control of UAV system. *AIP Conference Proceedings*, *1755*. https://doi.org/10.1063/1.4958613

Ramadhan, N. R. (2024). Simulasi Kontrol PID Ziegler-Nichols pada Sistem Penghancuran Batu dengan Motor Induksi 3 Fasa 20HP. *Jurnal Elektronika Dan Otomasi Industri*, *11*(1), 227–237. https://doi.org/10.33795/elkolind.v11i1.5146

Ruswandi Djalal, M. (2019). ANT COLONY BASED PID TUNED PARAMETERS FOR CONTROLLING SYNCHRONOUS MOTOR. *Technology Acceptance Model*, *10*(1).

Ruswandi Djalal, M., & Rahmat. (2020). Penalaan optimal kendali motor DC berbasis ant colony optimization. *Jurnal Teknologi*, *12*(1), 49–56. https://dx.doi.org/10.24853/jurtek.12.1.49-56

Sianturi, R. Y. C., Rahayudi, B., & Widodo, A. W. (2021). Implementasi Algoritma Ant Colony Optimization untuk Optimasi Rute Distribusi Produk Kebutuhan Pokok dari Toko Sasana Bonafide Mojoroto . *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *5*(7), 3190–3197.

Udjulawa, D., & Oktarina, S. (2022). Penerapan Algoritma Ant Colony Optimization Untuk Pencarian Rute Terpendek Lokasi Wisata. *Klik - Jurnal Ilmu Komputer*, *3*(1), 26–33. https://doi.org/10.56869/klik.v3i1.326

Wang, L., Luo, Y., & Yan, H. (2023). Ant colony optimization-based adjusted PID parameters: a proposed method. *PeerJ Computer Science*, *9*, 1–17. https://doi.org/10.7717/peerj-cs.1660