

Ant colony optimization-based adjusted PID parameters: a proposed method

Long Wang^{1,2}, Yiqun Luo² and Hongyan Yan³

¹ Department of Energy Electrical Engineering, Graduate School, Woosuk University, Jincheon-gun, Chungbuk-do, South Korea

² College of Physics and Electronic and Electrical Engineering, Xiangnan University, Hunan, Chenzhou, China

³ School of Water Conservancy and Hydroelectric Power, Hebei University of Engineering, Handan, Hebei, China

ABSTRACT

The ant colony algorithm (ACA) is a heuristic algorithm that resolves the optimality problem by simulating an ant's foraging process, which finds the shortest path. The connotation of the ACA is to find the optimal solution. The Proportional Integral Derivative (PID) parameter tuning is an essential tool in the control field and includes three parameters, K_p , K_i , and K_d , to achieve the best control effect. Besides, tuning the PID parameters is closely related to finding the "optimal" solution that can be attained based on the feasible combination of the two. This article transforms the PID parameter tuning problem into an ACA that finds the optimal solution called ACA-based PID parameters tuning. Furthermore, PID control is simulated by setting the parameters of ACA, such as ant colony size, iteration times, nodes, paths, path evaluation criteria, pheromone concentration, heuristic function, weight factor, and decision function. Eventually, the two PID controller parameter tuning strategies are compared and analyzed, and the advantages and disadvantages of each are obtained. Compared with the 4:1 attenuation curve method, the proposed method can significantly reduce the MP score of the overshoot of the system, increase the time, and improve the dynamic and steady-state performance of the system, but reduce the steady-state error of the system. Therefore, the feasibility and effectiveness of the proposed method is verified.

Submitted 25 August 2023
Accepted 2 October 2023
Published 16 November 2023

Corresponding author
Hongyan Yan, yhyan118@163.com

Academic editor
Muhammad Asif

Additional Information and
Declarations can be found on
page 13

DOI 10.7717/peerj-cs.1660

© Copyright
2023 Wang et al.

Distributed under
Creative Commons CC-BY 4.0

OPEN ACCESS

Subjects Adaptive and Self-Organizing Systems, Algorithms and Analysis of Algorithms, Computer Networks and Communications, Social Computing

Keywords Ant colony algorithm, PID, Pheromone, Decay curve method

INTRODUCTION

Proportional Integral Derivative (PID) controllers are the most commonly employed controllers in implementations and get more attention from researchers (*Dhanasekarana, Siddhanb & Kaliannan, 2020*). A PID controller is omnipresent in a diverse industrial framework and operations (*Aström & Hägglund, 2005*). Its design depends on choosing controller parameters to realize closed-loop system needs. Many algorithms and tuning approaches have been suggested, notably for standard PID controllers. Even though customary tuning approaches exist, such as Ziegler-Nichols, pole placement, and Cohen-Coon, they depend on approximated models whose order is low, barely realize high dynamic performance, and rarely manage constraints related to processes, which are frequent and substantial in real-life cases (*Padula & Visioli, 2012; Reynoso-Meza et al.,*

2013). To work both uncertainties and frequent external factors, control methods employing iterative learning are effective. Low transient tracking errors are realized. However, both single and repeated operations can employ them (Bristow, Tharayil & Alleyne, 2006). Thus, since systems used in industries face swift and volatile alterations, different smart control approaches, such as neural network (NN) controllers, can be implemented. For instance, an autotuning controller approach using smart NN and a relay feedback method was suggested (Nguyen, Shin & Kim, 2015; Sento & Kitjaidure, 2016). For systems with unknown parameters or uncertainties, NN based on PID utilizing an extended Kalman filter method (An-Hua, 2013) and a grey relational analysis-based practice (Jensi & Jiji, 2016) could efficiently enhance the controller's performance. Intrinsic particularities and operating requirements are unique in each controlled system. The adjustment approach of the controller must be considered. Furthermore, contemporary heuristic optimization (CHO) methods, such as Particle Swarm Optimization (PSO) (Mariajayaprakash, Senthilvelan & Gnanadass, 2016), genetic algorithms (GA) (Song, Yan & Zhao, 2017), and the artificial bee colony method (ABC) (Lee & El-Sharkawi, 2008), among others, have been advanced and have been implemented diverse problems of the control systems (Fleming & Purshouse, 2002) handling performance evaluation, design specification, and other features as multimodality and non-linearities (Zhao et al., 2011).

Additionally, multiple system requirements could be met during the adjustment of the PID controller (Reynoso-Meza et al., 2013). Many improvements within CHO methods have been suggested to advance the performance of PID tuning. For instance, an improved type of PSO provides higher performance for the robust PID controller design of two MIMO systems, a distillation column plant, and an aircraft control system (Puri & Ghosh, 2013). In Blondin & Sicard (2013), a combination of a stochastic population-based optimization technique and a pattern search-based method has been suggested for tuning the PID controller, where an effective combination method utilizes each optimization algorithm's advantage. As Dorigo & Stützle (2004) suggests, the tuning of the PID concerning a motion system with flexible transmission is conducted with a combination of ACA (Lagarias et al., 1998) and the Nelder–Mead approach (NM) (Tützle & Hoos, 2000).

As one of the heuristic optimization methods, ACA has several good properties and is applied to various problems in diverse fields with promising outcomes. Several methods are suggested to advance the ACA's performance. Tützle & Hoos (2000) proposed the Max and Min Any System (MMAS) model, which can effectively inhibit the production of pheromones and enhance the positive feedback effect. Baojiang & Shiyong (2007) adopted dynamic information updating rules and variation strategies to speed up the algorithm. Meng, You & Liu (2020) suggested a method to remedy the problem that the ACA easily falls into local optima and has a slow convergence. A multi-colony collaborative ACA employing a cooperative game approach (CCACO) was suggested. Qin, Huang & Suganthan (2009) developed an adaptive rule to overcome the ACA's precocious stagnation and slower convergence. Dhanasekaran, Siddhan & Kaliannan (2020) suggested that the load frequency control (LFC) of nuclear power systems was researched by applying the PID controller as a secondary controller. The controller gain scores were optimized by employing ACA. Blondina et al. (2018) claimed that an optimal gain tuning

approach for the PID controller was suggested to combine the simplified ACA algorithm and Nelder–Mead approach (ACO-NM) containing a novel operation to constrain NM. [Hai-bin, Dao-bo & Xiu-fen \(2006\)](#) proposed that an implementation of the ACA is presented to optimize the design parameters of the nonlinear PID controllers whose characteristics are higher precision of control and swift reaction. [Al-Amyal, Számel & Hamouda \(2023\)](#) suggested that an upgraded approach of the ACA is presented based on multistage ACA that advances the standard ACA's capability in searching. Several investigations have been run to improve the parameters of the PID controllers in the literature. More detailed discussions can be found in [Mahfoud et al. \(2022\)](#), [He & Tong \(2020\)](#), [Li & Peng \(2020\)](#), [Zeng et al. \(2019\)](#), [Narayana et al. \(2015\)](#), [Sun & Zheng \(2017\)](#), [Juyoung, Minje & Jonghwan \(2019\)](#), [Li et al. \(2021\)](#), [Dufek, Xiao & Murphy \(2021\)](#), [Ajeil et al. \(2020\)](#), [Xiaoming, Sheng & Chen \(2018\)](#), [Dai et al. \(2019\)](#), [Soyguder & Alli \(2010\)](#), [Lv, Duan & Jia \(2008\)](#), [Wei & Dong \(2018\)](#), [Ahn \(2020\)](#), [Shen & Yan \(2017\)](#), [Yu, Chang & Yu \(2005\)](#), [Ni \(2011\)](#), [Ming-tao et al. \(2016\)](#), [Yasunobu \(1985\)](#), [Chu et al. \(2020\)](#).

In this article, the parameter adjustment problem of PID is investigated based on using ACA. The Proportional Integral Derivative (PID) parameter tuning with three K_p , K_i , and K_d parameters is optimized to achieve the best control effect. It is found that finding the “optimal” solution could be reached based on the feasible combination of the two. The PID parameter tuning problem is transformed into an ACA that finds the optimal solution called ACA-based PID parameters tuning.

The other sections of the manuscript are outlined as follows: The optimization notion of the PID controllers is presented in “The Optimization of PID Controllers”. “Ant Colony Algorithm” introduces the full explanation of the standard ACA. The suggested algorithm is presented in “The Proposed Method”. “Results and Analysis” runs experiments and presents the outcomes and the analysis. The conclusion is given in “Discussion and Conclusion”.

THE OPTIMIZATION OF PID CONTROLLERS

This article first presents the relevant knowledge of PID, whose control mode has a series of advantages such as simple principle, convenient use, strong adaptability, strong robustness, and high reliability. Both process and motion controls use PID.

[Figure 1](#) depicts that Z denotes the input, e represents the deviation, C indicates the control quantity, O shows the output quantity and $e = O - C$. The working mechanism of the system is to continuously give feedback to the PID control system through the output of the controlled object. Then, the PID control system calculates the control quantity C again through the obtained deviation and modifies the strength of the controlled object successively to make the controlled object stable, which makes deviation $e = 0$. This is an effective negative feedback regulation mechanism that is expressed by

$$C(t) = K_p \left(e(t) + \frac{1}{K_i} \left(\int_0^t e(t) \right) dt + K_d \left(\frac{de(t)}{dt} \right) \right) \quad (1)$$

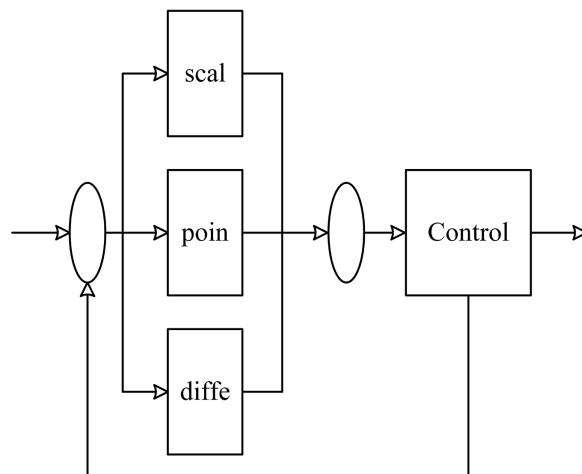


Figure 1 PID control schematic diagram.

Full-size  DOI: [10.7717/peerj-cs.1660/fig-1](https://doi.org/10.7717/peerj-cs.1660/fig-1)

The optimization design of the PID control system depends entirely on the tuning and optimization of the PID parameters, which K_p , K_i , and K_d denote. However, K_p , K_i , and K_d have no obvious rules in practical implementations, making it necessary to find those by running a trial-and-error approach in most production operations. Even though this method is feasible, it also comes with some disadvantages. For example, it has randomness and unpredictability issues; the selected K_p , K_i , and K_d may not be the optimal choices and will also need cumbersome computation and a long time. Therefore, the ACA will automatically find the optimal PID values of K_p , K_i , and K_d parameters to solve these problems.

ANT COLONY ALGORITHM

Preliminary

The ant colony algorithm (ACA) is a new heuristic algorithm proposed (*Colorni, Dorigo & Maniezzo, 1991*), which can reasonably simulate the foraging behavior of ants in nature. ACA does not need to describe the problem and has the characteristics of global optimization, internal parallelism, positive feedback, and robustness. Besides, compared with other algorithms, the ACA has higher reliability, faster, robust searchability, and easy-to-use implementation in solving combinatorial optimization problems, so it has attracted diverse attention that results in wide use in various fields. Moreover, ACA currently has become a hot topic in the field of multi-disciplines. For example, it is implemented to solve the traveling salesman problem (TSP) with good results. The existing research shows that it can effectively solve some NP-class problems with strong computational power, such as quadratic assignment, vehicle path planning, and job shop scheduling since its enhanced learning system with the characteristics of distributed computing, strong robustness, and easy-to-integrate structure can fit into other optimization algorithms. However, it has some problems, such as the long search time and appearing to be a pause phenomenon.

The implementation of the ant colony algorithm

The ACA is composed of components as follows:

(1) Node establishment: In the upper node, the ant foraging is a process from the starting point to the halfway node and finally to the food source node. Therefore, the practical problems studied should be analogically abstracted with ant colony foraging. This is the first step to establishing the algorithm, and the node establishment itself is the modeling of the actual problem, which is related to the feasibility and advantages of the algorithm. Unreasonable node establishment will lead to the actual problem not being solved and result in low efficiency. Therefore, it is necessary to collect relevant information and establish influential nodes for practical problems in the link of node establishment.

(2) Colony size and total foraging times: “Ant colony” generally expresses the number of ants and describes the total number of foraging times of the entire ant colony. However, due to the limited characteristics of the algorithm, the specific number of ants must be given in practical applications to represent the scale of the ant colony and the total number of foraging times of the entire ant colony. The size of the ant colony, showing the number of ants and the total number of foraging times of the ant colony showing the number of iterations, are extremely important parameters to the algorithm. If the number of ants and iterations is picked too small, the algorithm may be inefficient and eventually lead to the termination of the run before the algorithm convergence. Thus, the result would not be the optimal solution. On the contrary, if the number of ants and iterations is chosen too large, the algorithm will eventually converge. Still, subsequent calculations will be redundant since the number of iterations required for algorithm convergence is exceeded, which will lead to a very long execution time and eventually lose the advantages of ACA. Therefore, when the algorithm is run, it is necessary to set the appropriate number of ants and iterations based on the problem. Additionally, timely adjustments can be made according to the execution effect of the algorithm.

(3) Path evaluation criteria: In the process of ant foraging, the quality of the path can be intuitively expressed by the length of the path, which is an evaluation criterion of the path. However, in the practical application of the ACA, it may be necessary to specify a reasonable index according to the actual demand to replace the path length, which is the formulation used for evaluating path evaluation criteria. The path evaluation criterion is often closely related to the actual intent and represents the definition of the good or bad solution obtained by the final algorithm. Therefore, the formulation of the path evaluation criteria is very related to the final solution of the algorithm representing the solution state of the actual problem.

(4) Transfer probability rule: Transfer probability is an essential basis for ants to transfer from the current node to another node, which is expressed by

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{s \in J(i)} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, j \in J_i \\ 0, j \notin J_i \end{cases} \quad (2)$$

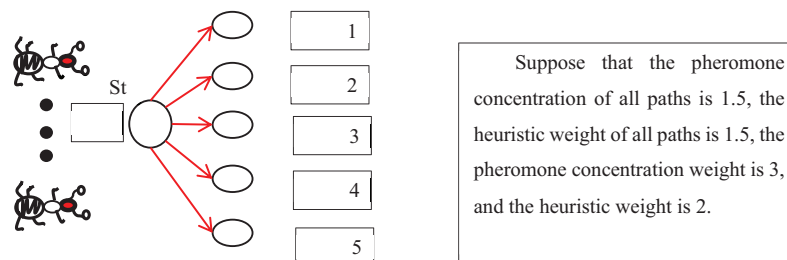


Figure 2 Schematic diagram of transfer probability calculation.

Full-size DOI: 10.7717/peerj-cs.1660/fig-2

p_{ij}^k represents the probability of ants moving from node i to node j when the number of ant colony iterations equals k generation, τ_{ij} denotes the pheromone concentration from node i to node j , η_{ij} represents the heuristic function from node i to node j , which can be set according to the actual performance requirements, α designates the weight factor of the pheromone, and β represents the weight factor of the heuristic function. The weight factor represents the degree of influence of each factor on the transition probability. When $\alpha > \beta$, the transfer probability mainly depends on the pheromone concentration. When $\alpha < \beta$, the transfer probability depends primarily on the heuristic provided that α and β are greater than 1. The transfer probability of the current node to each adjacent node is equal to the ratio of the excitation of each alternative path time, the concentration of the pheromone, and the excitation of the total alternative path times, the attention of the pheromone.

Figure 2 depicts it.

Figure 2 depicts that the transfer probability of node 0 to node one is computed by

$$P_{01} = \frac{(1.5^2) * (1.5^3)}{(1.5^2) * (1.5^3) + (1.5^2) * (1.5^3) + (1.5^2)(1.5^3) + (1.5^2) * (1.5^3) + (1.5^2) * (1.5^3)} = 0.2$$

Similarly, the transfer probabilities of other paths can be calculated and obtained as follows: $p_{02} = 0.2$, $p_{03} = 0.2$, $p_{04} = 0.2$, $p_{05} = 0.2$. When the transfer probability is obtained, the ant's movement has a basis, and the algorithm establishes certain rules.

(5) Pheromone update rule: The pheromone update rule directly affects the algorithm's convergence. If the pheromone concentration increases too fast, the ants will not have time to search for other paths and nodes. Due to the high proportion of pheromone growth in some paths, the ants will have gathered at a part of the paths and nodes, and the final optimal solution obtained by the algorithm would not be a globally optimal solution.

Figure 3 depicts it.

Figure 3 depicts that the activity range of the ant colony is limited within the blue coil, which leads to left nodes and paths that are not searched. Although the "optimal solution" can be obtained by premature local search, the optimal herein is not a real optimal in the real sense, but a local optimal that is a pathological manifestation of the ACA because it is not certain that the path formed by the combination of missing nodes and local nodes must be worse than the local optimal solution. Therefore, a reasonable pheromone growth rule must be adopted when designing the updating rule of information number to avoid

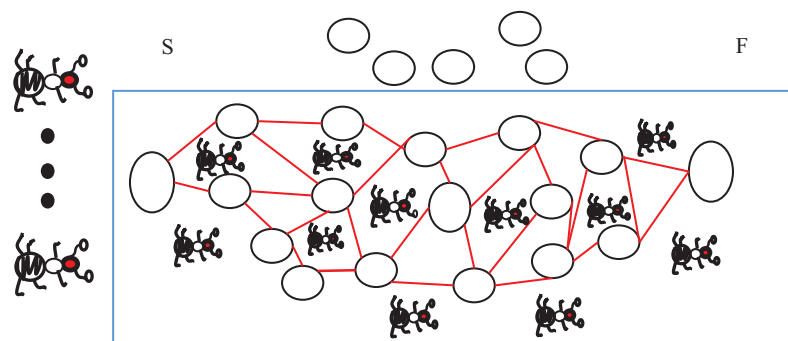


Figure 3 Schematic diagram of ant colony trapped in local search.

Full-size DOI: 10.7717/peerj-cs.1660/fig-3

the algorithm falling into the local optimum prematurely and causing the loss of the solution.

THE PROPOSED METHOD

The architecture of the proposed method

The first step is to set the ACA to adjust the PID parameters as the experimental group. MATLAB 7.9 version is used to write the program and three parameters, K_p , K_i , and K_d , are obtained by the path formed by each ant in each generation, and then K_p , K_i , and K_d are loaded into the PID simulation model written in the Simulink in MATLAB 7.9. Afterward, a simulation is run to import the output data of the PID simulation system into MATLAB 7.9 through the output module. Then, the output data is analyzed to obtain the performance value of the PID system formed by K_p , K_i , and K_d values at this time. The next step is to update the pheromone with the set pheromone rules. The steps above are repeated until the algorithm is complete and the relevant data is recorded.

The second step is to set the engineering tuning method to adjust the PID parameters as the control group. The PID simulation model uses the engineering tuning method to calculate the values of K_p , K_i , and K_d , and then the simulation output formed by these three parameters is imported into the program, and the data is analyzed by the same method in the experimental group to obtain the relevant performance values.

The third step is to draw the output waveform diagram of the two groups of experiments, compare and analyze the waveform and related performance values, and draw a conclusion. Figure 4 depicts it.

PID simulation system model

The purpose of building a PID simulation system with Simulink is to simulate the most real PID application effect through an effective model. Second, this article needs to analyze the data through the simulation output to get the pros and cons of the relevant performance index judgment algorithm.

A controlled system function is defined by $G(s) = \frac{130}{s^2 + 25s}$, which can be set according to the actual situation. Figure 5 shows that the system architecture is roughly the same as the PID schematic diagram. This model passes the PID control system through the step input

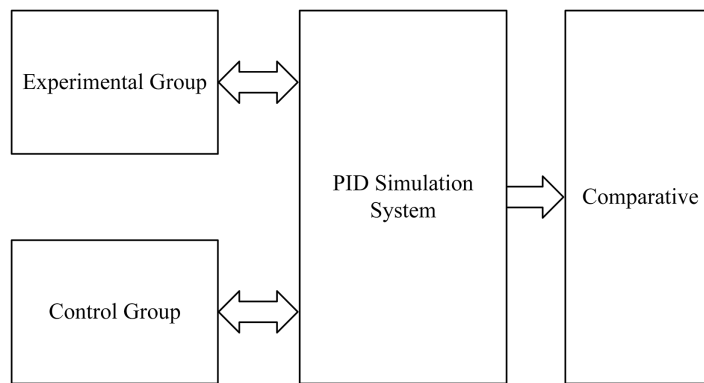


Figure 4 Design architecture diagram.

Full-size DOI: 10.7717/peerj-cs.1660/fig-4

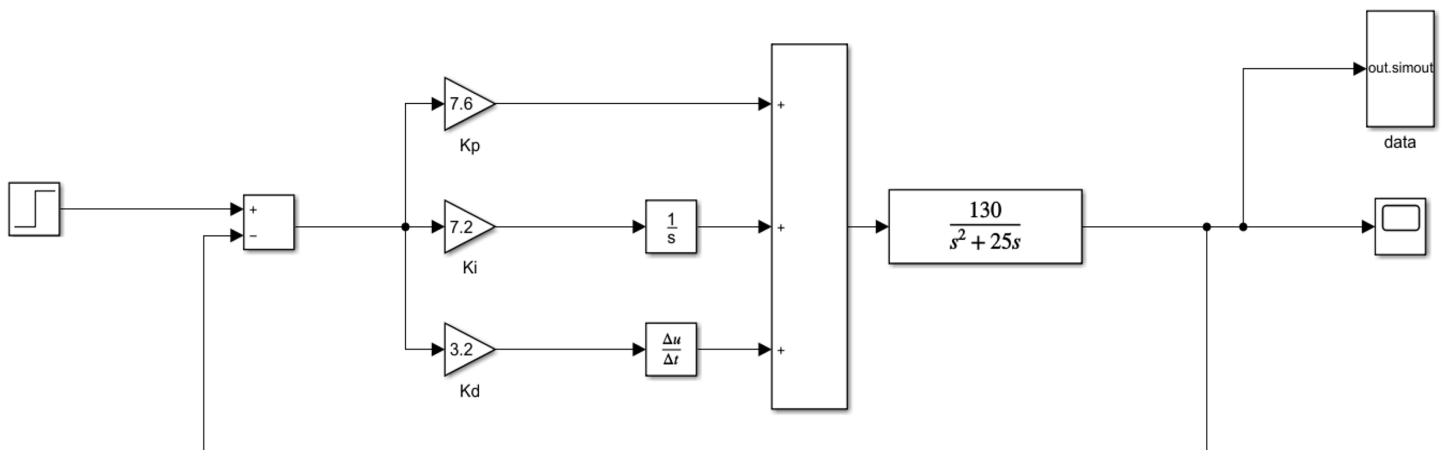


Figure 5 PID simulation system.

Full-size DOI: 10.7717/peerj-cs.1660/fig-5

(target temperature) and then acts on the controlled system. Finally, the output of the controlled system is divided into three ways: one is fed back to the PID control system, the other is connected to the oscilloscope (display waveform), and the final is connected to the output module (output data to the work area).

ACA-based PID parameters tuning: experimental group

The construction of the experimental group uses the implementation steps of the ACA. The main steps are given as follows: The first step is the node and path establishment. K_p , K_i , and K_d are found and transformed into a node and path problem. In this experiment, K_p , K_i , and K_d are set as nodes on different axes of different three-dimensional coordinate axes.

K_p , K_i , and K_d correspond to the x-axis, the y-axis, and the z-axis, respectively, and are set with 50 nodes from 0.2 to 10 with a step value of 0.2. Figure 6 depicts that a line of the same color and a node connection represents a valid path, and the sequence is the K_p connected to K_i , and K_i is then related to K_d . Therefore, the rules of nodes and paths are

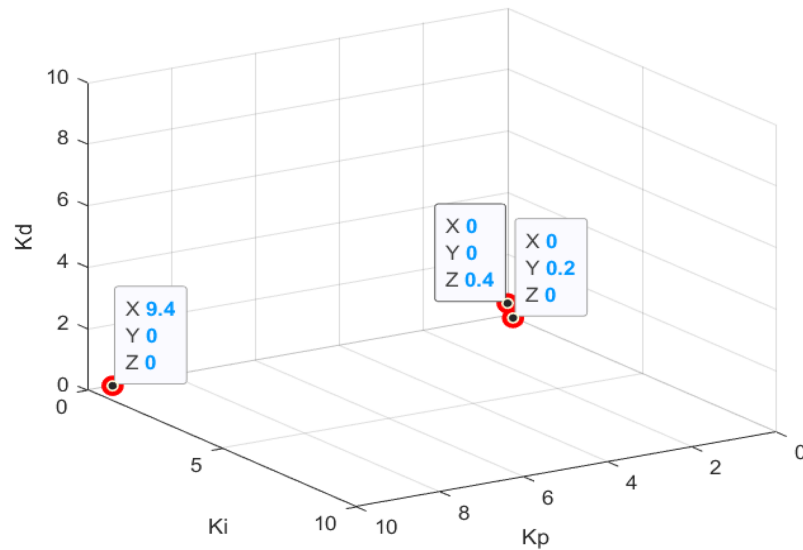


Figure 6 Optimal solution of PID parameters adjusted by ant colony algorithm.

Full-size DOI: [10.7717/peerj-cs.1660/fig-6](https://doi.org/10.7717/peerj-cs.1660/fig-6)

established, and when the ant selects a node on a different axis, the ant's K_p , K_i , and K_d values are obtained.

The second step determines ant colony size and total foraging times. In the program, the number of ants (count) represents the colony size, the number of iterations (N.C.) means the total foraging times and the count is set to 50. N.C. is set to 200; the number of ants in one generation is 50, and the total number is 200 generations. At the end of every 50 ants searching the path, the iteration counter is increased by one until the number of iterations reaches 200 and the entire program is finalized.

The third step is path evaluation criteria. In this experiment, the comprehensive performance value (CPV) is set to replace the path length as the path evaluation criterion. First, the performance indicators related to step response are introduced (the output of the controlled object of the simulation system is step response).

- (1) Overshoot (MP): the maximum value of the simulation result.
- (2) Rise time (tr): the time from the start to the first time for the specified stable value.
- (3) Adjustment time (ts): the time of transient process experienced when the deviation between the actual output and the specified stable value reaches the allowable range (1% in this experiment) (timing from $t = 0$)
- (4) Steady-state error (err): specifies the difference between the stable value and the steady-state output state.

Comprehensive performance value (CPV) calculation formula:

$$CPV = \frac{10^{32}}{MP^{20} * tr^3 * ts^3 * err^4} \quad (3)$$

Equation (3) implies that the higher the CPV, the better the performance of the control system, and the CPV is more biased to the performance value of overshoot (M.P.).

Step 4: Transfer probability rules and transfer implementation.

We mainly establish the correlation function and parameters. To pursue the speed of the control system, the enlightening function in this design is set as follows:

- (1) Path heuristic function from start point to K_p node: $\eta = 1 + 0.1 \times K_p$ (min = 1.02, max = 2)
- (2) Path heuristic function from K_p node to K_i node: $\eta = 1 + 0.1 \times K_i$ (min = 1.02, max = 2)
- (3) Path heuristic function from K_i node to K_d node: $\eta = 1 + 0.2/K_d$ (min = 1.02, max = 2)

The pheromone concentration function is set as follows: the pheromone concentration of each path is initialized to 1 and changes as the pheromone concentration is updated. The greater the inspiration is equivalent to the larger the K_p value, the larger the K_i value, the smaller the K_d value, the faster the system recovery, which can effectively reduce the rise time and adjustment time, but the corresponding overshoot and stability error will increase, which is equivalent to a trend of one part of the performance, away from the other part of the performance. This setting will make the control system faster. However, stability is not guaranteed, so the weight of the inspiration function should not be too large and less than the weight of the pheromone concentration function because the inspiration is very limited. The pheromone is focused on the global. The hope is to focus on the control system's speed to ensure comprehensive performance rather than mindlessly pursuing the control system's speed. Therefore, in this design, the pheromone concentration weight α is set to 10, and the heuristic weight β is set to 1.

The fifth step is the pheromone update rule: the update frequency is set to tune the algorithm once every iteration occurs, and each generation's optimal path (the path with the largest CPV) is selected for update. To prevent local search, the update of pheromone concentration should not be too large, so it is set to increase CPV/100 each time.

At this point, the experimental group was completed. Since K.P., K.I., and K.D. are subdivided into 50 pieces of data from 0.2 to 10 with a step value of 0.2, the composed path reaches $50 \times 50 \times 50 = 125,000$ kinds, and the amount of data is too large to be displayed. The algorithm is a dynamic process that cannot exhaust each state's specific state. Therefore, the following K.P., K.I., and K.D. are intercepted from 0.2 to 1, and the initial stage of the algorithm is used as a program flow demonstration.

Adjusted PID parameters by engineering setting method: control group

The methods of tuning PID parameters in engineering mainly include critical proportion, response curve, and attenuation. This article uses the 4:1 attenuation curve method to adjust PID parameters to construct a control group. The steps are given as follows:

- (1) The regulator integral time is set to infinity, the differential time is set to zero ($K_i = \infty$, $K_d = 0$), K_p is appropriate, and the control system is input by pure K_p action. After stability, the proper proportion is reduced until the adjustment process changes reach the

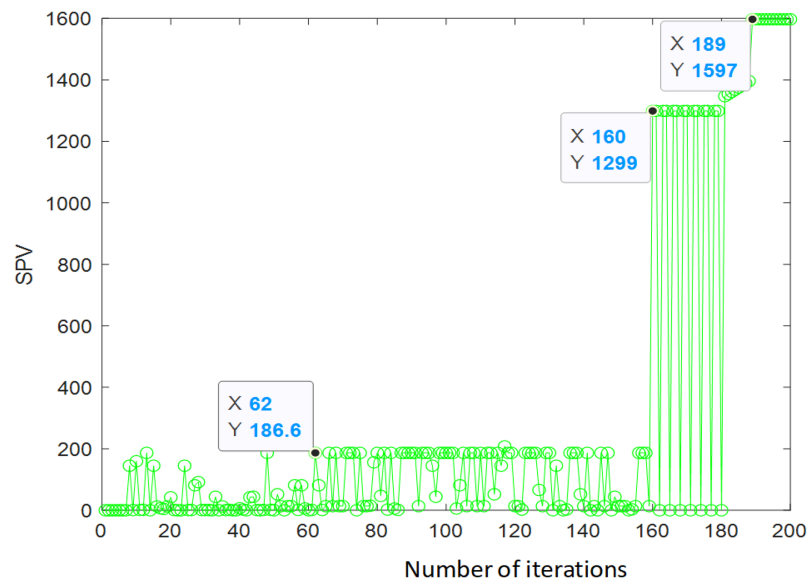


Figure 7 Iterative result of ant colony algorithm.

Full-size  DOI: [10.7717/peerj-cs.1660/fig-7](https://doi.org/10.7717/peerj-cs.1660/fig-7)

specified 4:1 attenuation ratio. The relevant parameters of proportionality δs and attenuation operation period T.S. in the case of 4:1 attenuation are obtained.

(2) According to the equation, critical scale δs and the calculation of the critical period T_k is defined by

$$K_p = 1/\delta s; K_i = K_p/(0.3 * TS); K_d = K_p * (0.1 * TS).$$

(3) According to the actual effect correction.

It is measured that when $\delta s = 0.37$ (K_p value in step 1), the two adjacent peak values are 74.73 and 56.22, respectively, and the peak time is 1.057 and 1.167 s, respectively. $(74.73 - 50)/(56.22 - 50) = 3.96$, which is close to the 4:1 attenuation ratio, so it meets the conditions. Then T.S. = $1.207 - 1.057 = 0.15$.

$K_p = 2.7$, $K_i = 0.045$ and $K_d = 0.015$ can be obtained from the step equation and final correction.

RESULTS AND ANALYSIS

Figure 6 depicts that the optimal result of the PID parameter tuning by the ACA is $K_p = 9.4$, $K_i = 0.2$, $K_d = 0.4$.

Figure 7 depicts that the algorithm has completed convergence in about 190 iterations, and the comprehensive performance value (CPV) is finally 1,597.

Figure 8 depicts that the blue curve represents the adjusted PID parameter simulation results by the ACA, and the red curve represents the adjusted PID parameter simulation results by the attenuation curve method. Both strategies provide effective parameters, and the output table of the controlled system shows strong stability. Still, it is evident that the blue curve is smoother, and it can be roughly estimated that the ACA is better than the decay curve strategy.

Table 1 suggests that the output of the control system of the ACA is better than that of the attenuation curve method except for rise time, overshoot, adjustment time, and

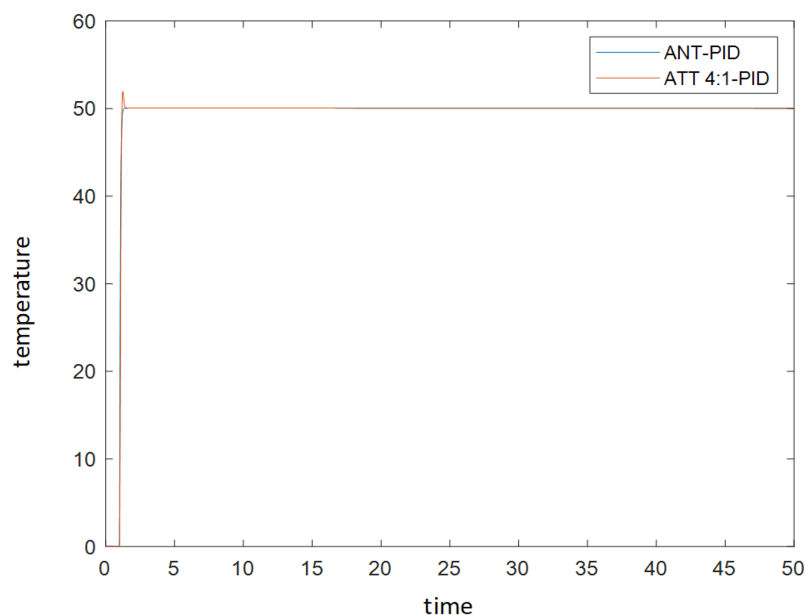


Figure 8 Comparison of simulation results of PID parameter tuning by ant colony algorithm and PID parameter tuning by attenuation curve method. [Full-size](#) DOI: 10.7717/peerj-cs.1660/fig-8

Table 1 Comparison of two different optimization strategies.

Algorithm	Kp	Ki	Kd	MP	TR	TS	ERR	SPV
ATT4:1-PID	2.7	0.045	0.015	51.9501	1.1872	1.3376	0.0476	22.0467
ANT-PID	9.4	0.2	0.4	50.0612	1.3298	1.2109	0.0352	1597.1

stability error. The comprehensive performance value (CPV) demonstrates that the ACA's tuning strategy is better than the decay curve method. Based on theoretical analysis and experimental research, the PID parameter tuning strategy based on the ACA is proposed in this article and compared with the traditional attenuation curve method. It is found that this strategy is effective and feasible and has the following advantages:

- (1) The algorithm implementation is more straightforward;
- (2) It has good stability;
- (3) For the PID control parameter optimization problem, the global optimal can be found;
- (4) At the same time, the advantages are more significant in the case of large data volumes and complex modeling.

DISCUSSION AND CONCLUSION

The Proportional Integral Derivative (PID) parameter tuning is an essential tool in the control field and includes three parameters, Kp, Ki, and Kd, to achieve the best control effect. Besides, tuning the PID parameters is closely related to finding the "optimal" solution that can be attained based on the feasible combination of the two. In this article,

the PID parameter tuning problem is transformed into an ACA that finds the optimal solution called ACA-based PID parameters tuning. Furthermore, PID control is simulated by setting the parameters of ACA, such as ant colony size, iteration times, nodes, paths, path evaluation criteria, pheromone concentration, heuristic function, weight factor, and decision function. Eventually, the two PID controller parameter tuning strategies are compared and analyzed, and the advantages and disadvantages of each are obtained. Compared with the 4:1 attenuation curve method, the proposed method can significantly reduce the M.P. score of the overshoot of the system, increase the time, and improve the dynamic and steady-state performance of the system, but reduce the steady-state error of the system.

The corresponding control group was constructed by the engineering setting method to make the experiment more convincing and scientific. Then, the two groups of experimental results utilizing K_p , K_i , and K_d values, overshoot, rise time, adjustment time, stability error, and self-set comprehensive performance value (CPV) and other related performance indexes were compared and analyzed. Finally, it is concluded that the ACA-based tuning of the PID parameters has proved feasible and better than the traditional PID method implemented in engineering tuning applications. Therefore, the feasibility and effectiveness of the proposed method is verified.

Future research will employ heuristic methods to optimize the same parameters.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This study was supported by the Hunan Province Social Science Achievement Evaluation Committee Project, for the research on the Service System Based on the Internet of Things + Home Care for the Aged (No. XSP2023GLC002), the Science and Technology Project of Chen Zhou, for the research on Design and Research of COVID-19 Prevention and Detection Bracelet Based on ZigBee Wireless Sensor (No. ZDYF2020156), and the Hunan Province College Students Innovation and Entrepreneurship Training Program Project, for the research on Wearable Temperature Measurement and Positioning Bracelet Based on Bluetooth Design (No. Xiangjiaotong (2022) No.174, No. 4308). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:

Hunan Province Social Science Achievement Evaluation Committee Project:
XSP2023GLC002.

Science and Technology Project of Chen Zhou: ZDYF2020156.

Hunan Province College Students Innovation and Entrepreneurship Training Program Project: No. Xiangjiaotong [2022] No.174, No. 4308.

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Long Wang conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the article, and approved the final draft.
- Yiqun Luo conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the article, and approved the final draft.
- Hongyan Yan conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the article, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The Matlab code is available in the [Supplemental File](#).

Supplemental Information

Supplemental information for this article can be found online at <http://dx.doi.org/10.7717/peerj-cs.1660#supplemental-information>.

REFERENCES

- Ahn DS. 2020.** Control of humanoid robots using time-delay-estimation and fuzzy logic systems. *Journal of Drive and Control* **17**(1):44–50.
- Ajeil FH, Ibraheem IK, Azar AT, Humaidi AJ. 2020.** Grid-based mobile robot path planning using aging-based ant colony optimization algorithm in static and dynamic environments. *Sensors* **20**(7):1880 DOI [10.3390/s20071880](https://doi.org/10.3390/s20071880).
- Al-Amyal F, Számel L, Hamouda M. 2023.** An enhanced direct instantaneous torque control of switched reluctance motor drives using ant colony optimization. *Ain Shams Engineering Journal* **14**(5):101967 DOI [10.1016/j.asej.2022.101967](https://doi.org/10.1016/j.asej.2022.101967).
- An-Hua P. 2013.** GRA-based approach to PID parameter tuning for closed-loop servo systems. In: *Chinese Control and Decision Conference*. 1628–1633 DOI [10.1109/CCDC.2013.6561191](https://doi.org/10.1109/CCDC.2013.6561191).
- Aström K, Hägglund T. 2005.** *Advanced PID control*. Research Triangle Park: ISA.
- Baojiang Z, Shiyong L. 2007.** Ant colony optimization algorithm and its application to Neuro-Fuzzy controller design. *Journal of Systems Engineering and Electronics* **18**(3):603–610 DOI [10.1016/S1004-4132\(07\)60135-2](https://doi.org/10.1016/S1004-4132(07)60135-2).
- Blondin MJ, Sicard P. 2013.** Combined, ACO algorithm—Nelder–Mead simplex search for controller and anti-windup tuning for a motion system with flexible transmission. In: *IECON 2013*. Piscataway: IEEE, 3364–3369.
- Blondina MJ, Sanchis J, Sicarda P, Herrero JM. 2018.** New optimal controller tuning method for an AVR system using a simplified Ant Colony Optimization with a new constrained Nelder–Mead algorithm. *Applied Soft Computing* **62**(6):216–229 DOI [10.1016/j.asoc.2017.10.007](https://doi.org/10.1016/j.asoc.2017.10.007).
- Bristow DA, Tharayil M, Alleyne AG. 2006.** A survey of iterative learning control. *IEEE Control Systems* **26**(3):96–114 DOI [10.1109/MCS.2006.1636313](https://doi.org/10.1109/MCS.2006.1636313).

- Chu P, Yu Y, Dong D, Lin H, Yuan J. 2020.** NSGA-II-based parameter tuning method and GM (1, 1)-based development of fuzzy immune PID controller for automatic train operation system. *Mathematical Problems in Engineering* 2020(6):1–20 DOI 10.1155/2020/3731749.
- Coloni A, Dorigo M, Maniezzo V. 1991.** Distributed optimization by ant colonies. In: *European Conference of Artificial Life*. 134–142.
- Dai X, Long S, Zhang Z, Gong D. 2019.** Mobile robot path planning based on ant colony algorithm with A* heuristic method. *Frontiers in Neurorobotics* 13:15 DOI 10.3389/fnbot.2019.00015.
- Dhanasekaran B, Siddhan S, Kaliannan J. 2020.** Ant colony optimization technique tuned controller for frequency regulation of single area nuclear power generating system. *Microprocessors and Microsystems* 73(2):102953 DOI 10.1016/j.micpro.2019.102953.
- Dhanasekaran B, Siddhan S, Kaliannan J. 2020.** Ant colony optimization technique tuned controller for frequency regulation of single area nuclear power generating system. *Microprocessors and Microsystems* 73(2):102953 DOI 10.1016/j.micpro.2019.102953.
- Dorigo M, Stützle T. 2004.** *Ant colony optimization*. Cambridge: MIT Press.
- Dufek J, Xiao X, Murphy RR. 2021.** Best viewpoints for external robots or sensors assisting other robots. *IEEE Transactions on Human-Machine Systems* 51(4):324–334 DOI 10.1109/THMS.2021.3090765.
- Fleming PJ, Purshouse RC. 2002.** Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice* 10(11):1223–1241 DOI 10.1016/S0967-0661(02)00081-3.
- Hai-bin' D, Dao-bo W, Xiu-fen Y. 2006.** Novel approach to nonlinear PID parameter optimization using ant colony optimization algorithm. *Journal of Bionic Engineering* 3(2):73–78 DOI 10.1016/S1672-6529(06)60010-3.
- He TD, Tong H. 2020.** Remote sensing image classification based on adaptive ant colony algorithm. *Arabian Journal of Geosciences* 13(14):1–13 DOI 10.1007/s12517-020-05717-9.
- Jensi R, Jiji GW. 2016.** An enhanced particle swarm optimization with levy flight for global optimization. *Applied Soft Computing* 43(23):248–261 DOI 10.1016/j.asoc.2016.02.018.
- Juyoung J, Minje K, Jonghwan L. 2019.** Improvement of ant colony optimization algorithm to solve traveling salesman problem. *Journal of the Society of Korea Industrial and Systems Engineering* 42(3):1–7 DOI 10.11627/jkise.2019.42.3.001.
- Lagarias JC, Reeds JA, Wright MH, Wright PE. 1998.** Convergence properties of the Nelder–Mead simplex method in low dimension. *SIAM Journal on Optimization Society for Industrial and Applied Mathematics* 9(1):112–147 DOI 10.1137/S1052623496303470.
- Lee KY, El-Sharkawi MA. 2008.** *Modern heuristic optimization techniques: theory and applications to power systems*. Hoboken: Wiley Press.
- Li QQ, Peng Y. 2020.** A wireless mesh multipath routing protocol based on sorting ant colony algorithm. *Procedia Computer Science* 166:570–575 DOI 10.1016/j.procs.2020.02.018.
- Li J, Wang J, Peng H, Hu Y, Su H. 2021.** Fuzzy-torque approximation-enhanced sliding mode control for lateral stability of mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 52(4):2491–2500 DOI 10.1109/TSMC.2021.3050616.
- Lv H, Duan P, Jia L. 2008.** A novel fuzzy controller design based on PID gains for HVAC systems. In: *Proceedings of 7th World Congress on Intelligent Control and Automation (WCICA 2008)*. Piscataway: IEEE, 736–739.
- Mahfoud S, Derouich A, Iqbal A, El Ouanjli N. 2022.** ANT-colony optimization-direct torque control for a doubly fed induction motor: an experimental validation. *Energy Reports* 8(10):81–98 DOI 10.1016/j.egyr.2021.11.239.

- Mariajayaprakash A, Senthilvelan T, Gnanadass R. 2016.** Optimization of process parameters through fuzzy logic and genetic algorithm—a case study in a process industry. *Applied Soft Computing* 30(5):94–103 DOI 10.1016/j.asoc.2015.01.042.
- Meng L, You X, Liu S. 2020.** Multi-colony collaborative ant optimization algorithm based on cooperative game mechanism. *IEEE Access* 8:154153–154165 DOI 10.1109/ACCESS.2020.3011936.
- Ming-tao Y, Feng C, Rui-yi Q, Jun L, Yun C, Nai-li H, Fu-lin Z. 2016.** Fuzzy PID control considering vehicular suspension dynamic deflection. *Transactions of Beijing Institute of Technology* 36(9):929–934 DOI 10.15918/j.tbit1001-0645.2016.09.009.
- Narayana KL, Kumar VN, Dhivya M, Raj RP. 2015.** Application of ant colony optimization in tuning a PID controller to a conical tank. *Indian Journal of Science and Technology* 8(S2):217–223 DOI 10.17485/ijst/2015/v8iS2/61635.
- Nguyen GH, Shin JH, Kim WH. 2015.** Autotuning controller for motion control system based on intelligent neural network and relay feedback approach. *IEEE/ASME Transactions on Mechatronics* 20(3):1138–1148 DOI 10.1109/TMECH.2014.2344692.
- Ni Z. 2011.** Fuzzy PID control in train ATO system’s simulation research. *Friend of Science Amateurs* 4(2):741–750.
- Padula F, Visioli A. 2012.** Optimal tuning rules for proportional-integral-derivative and fractional-order proportional-integral-derivative controllers for integral and unstable processes. *IET Control Theory & Applications* 6(6):776–786 DOI 10.1049/iet-cta.2011.0419.
- Puri P, Ghosh S. 2013.** A hybrid optimization approach for P.I. controller tuning based on gain and phase margin specifications. *Swarm and Evolutionary Computation* 8:69–78 DOI 10.1016/j.swevo.2012.09.001.
- Qin AK, Huang VL, Suganthan PN. 2009.** Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation* 13(2):398–417 DOI 10.1109/TEVC.2008.927706.
- Reynoso-Meza G, Sanchis J, Blasco X, Martínez M. 2013.** Evolutionary algorithms for PID controller tuning: current trends and perspectives. *Revista Iberoamericana de Automática e Informática Industrial* 10(3):251–268 DOI 10.1016/j.riai.2013.04.001.
- Sento A, Kitjaidure Y. 2016.** Neural network controller based on PID using an extended Kalman filter algorithm for multi-variable nonlinear control system. In: *International Conference on Advanced Computational Intelligence*. 302–309.
- Shen H, Yan J. 2017.** Optimal control of rail transportation associated automatic train operation based on fuzzy control algorithm and PID algorithm. *Automatic Control & Computer Sciences* 51(6):435–441 DOI 10.3103/S0146411617060086.
- Song X, Yan Q, Zhao M. 2017.** An adaptive artificial bee colony algorithm based on objective function value information. *Applied Soft Computing* 55:384–401 DOI 10.1016/j.asoc.2017.01.031.
- Soyguder S, Alli H. 2010.** Fuzzy adaptive control for the actuators position control and modeling of an expert system. *Expert Systems with Applications* 37(3):2072–2080 DOI 10.1016/j.eswa.2009.06.071.
- Sun L, Zheng Z. 2017.** Adaptive relative pose control for autonomous spacecraft rendezvous and proximity operations with thrust misalignment and model uncertainties. *Advances in Space Research* 59(7):1861–1871 DOI 10.1016/j.asr.2017.01.005.
- Tützle T, Hoos HH. 2000.** MAX-MIN ant system. *Future Generation and Computer Systems* 16(8):889–914 DOI 10.1016/S0167-739X(00)00043-1.

- Wei H, Dong Y. 2018.** Adaptive fuzzy neural network control for a constrained robot using impedance learning. *IEEE Transactions on Neural Networks and Learning Systems* **2018(99)**:1–13 DOI [10.1109/TNNLS.2017.2665581](https://doi.org/10.1109/TNNLS.2017.2665581).
- Xiaoming Y, Sheng L, Chen Z. 2018.** An improved ant colony system algorithm for robot path planning and performance analysis. *International Journal of Robotics and Automation* **33(5)**:527–533 DOI [10.2316/Journal.206.2018.5.206-0071](https://doi.org/10.2316/Journal.206.2018.5.206-0071).
- Yasunobu S. 1985.** Automatic train operation system by predictive fuzzy control. *Industrial Applications of Fuzzy Control* **1(3)**:432–438.
- Yu DL, Chang TK, Yu DW. 2005.** Fault tolerant control of multivariable processes using auto tuning PID controller. *IEEE Transactions on Systems Man & Cybernetics Part B* **35(1)**:32–43 DOI [10.1109/TSMCB.2004.839247](https://doi.org/10.1109/TSMCB.2004.839247).
- Zeng GQ, Xie XQ, Chen MR, Weng J. 2019.** Adaptive population extremal optimization based PID neural network for multivariable nonlinear control systems. *Swarm and Evolutionary Computation* **44(4)**:320–334 DOI [10.1016/j.swevo.2018.04.008](https://doi.org/10.1016/j.swevo.2018.04.008).
- Zhao S-Z, Iruthayarajan MW, Baskar S, Suganthan P. 2011.** Multiobjective robust PID controller tuning using two lbests multi-objective particle swarm optimization. *Information Sciences* **181(16)**:3323–3335 DOI [10.1016/j.ins.2011.04.003](https://doi.org/10.1016/j.ins.2011.04.003).