Hindawi
Security and Communication Networks
Volume 2022, Article ID 9862870, 1 page
https://doi.org/10.1155/2022/9862870

Retraction

Retracted: Simulation Research on the Palm Mechanism of Volleyball Robot Based on Artificial Intelligence and Ant Colony Optimization Algorithm

Security and Communication Networks

Received 18 November 2022; Accepted 18 November 2022; Published 4 December 2022

Security and Communication Networks has retracted the article titled "Simulation Research on the Palm Mechanism of Volleyball Robot Based on Artificial Intelligence and Ant Colony Optimization Algorithm" [1] due to concerns that the peer review process has been compromised.

Following an investigation conducted by the Hindawi Research Integrity team [2], significant concerns were identified with the peer reviewers assigned to this article; the investigation has concluded that the peer review process was compromised. We therefore can no longer trust the peer review process, and the article is being retracted with the agreement of the editorial board.

References

[1] G. Jiao, "Simulation Research on the Palm Mechanism of Volleyball Robot Based on Artificial Intelligence and Ant Colony Optimization Algorithm," *Security and Communication Networks*, vol. 2022, Article ID 6030545, 10 pages, 2022.

[2] L. Ferguson, "Advancing Research Integrity Collaboratively and with Vigour," 2022, https://www.hindawi.com/post/advancing-research-integrity-collaboratively-and-vigour/.

Hindawi
Security and Communication Networks
Volume 2022, Article ID 6030545, 10 pages
https://doi.org/10.1155/2022/6030545

Research Article

Simulation Research on the Palm Mechanism of Volleyball Robot Based on Artificial Intelligence and Ant Colony Optimization Algorithm

Guangshi Jiao

College of Sports Science, Shenyang Normal University, Shenyang 110034, Liaoning, China

Correspondence should be addressed to Guangshi Jiao; 13889112552@synu.edu.cn

Received 8 January 2022; Revised 24 January 2022; Accepted 14 February 2022; Published 10 March 2022

Academic Editor: Muhammad Arif

The study of mobile robots, which began in the late 1960s, is the most dramatic development in human history in the twentieth century, and the invention has undergone radical changes in just over 50 years. The robot body is developing in the direction of flexibility and miniaturization. This is because the robot application is mostly oriented to the family and service industries, and it needs to adapt to a more complex environment. This manuscript aims to improve further the ant colony optimization algorithm by using rough set theory to improve the convergence speed and accuracy of the algorithm in robot path planning on the basis of an in-depth diagnosis on the shortcomings and its causes of development of the ant colony optimization algorithm. It overcomes the drawbacks of the algorithm that easily get trapped in partial optimality solution, the search time is much slower and the search effect is not good. In this paper, the CMA-ES algorithm, the modified ant colony, and the BK method are proposed, which have high theoretical value and exploration significance. In addition, simulation experiments are conducted to obtain the stage results on the basis of artificial information. The results of the present paper indicate the GWO algorithm performs more stable in the optimization results of each experiment when there are 14 robots and the communication range is 1.6, compared with the PSO algorithm and BA algorithm.

1. Introduction

1.1. Background. People knew the word "robot" in the early 1920s and gave it a rich meaning. Since the birth of industrial robots in the 1960s, and the continuous expansion and expansion of application fields, widely used in industry, agriculture, military, medical and service industries. The requirements for intelligent robots in various fields have become higher and higher. At present, whether autonomous mobile robots can achieve intelligence to a certain extent is one of the main research contents of this subject. Among them, path planning is the core of mobile robot research [1–3]. This research direction has a wide range of implementation prospects and has attracted great attention. Typical applications include robotic cleaning, self-contained vacuum cleaners, binoculars, and self-contained mineral soil detectors. After entering the new century, the practicability, ease of use, and technical performance of mobile robots have reached a higher level. In order to build a higher level of environmental detection sensors, we develop more advanced environmental information processing technologies and explore higher adaptability. Intelligent robots can be more widely used in the real environment, and appropriate design technology can be used as an opportunity to carry out a higher level of robot research.

1.2. Significance. Although multirobots are currently being used in an increasingly wide range of applications, there are still obvious shortcomings. In the process of multirobot systems completing tasks, path conflicts, blockages, and collisions can occur due to the complexity of the system and environment, resulting in an inability to complete tasks or low work efficiency [4–6]. Due to these problems, the study of multirobot path planning techniques should become the focus of current robotics research, which requires further research by related scholars. At present, there are many intelligent algorithms used to solve robot path planning problems, but each algorithm has its shortcomings and cannot solve the path planning problem well. The ant colony algorithm (ACO) is an intelligent planning algorithm with better performance. It is a random search operation that emulates the feeding process of ants in nature. The information interaction of the ant colony algorithm is mainly completed by pheromone. When ants are looking for food, they can release a special secretion pheromone on the path they traveled. Over time, the substance will gradually volatilize, and later ants will choose to go according to the strength of the pheromone route. Compared with other algorithms, ACO is a cluster intelligence algorithm that can search for better solutions. Because simple modifications to the algorithm model can be used to solve other problems, the ACO has good robustness; it is a colony-based evolutionary algorithm with parallelism in that multiple ants can search in parallel during the algorithm operation. This algorithm is easy to combine with other algorithms, and by combining them together to form other methods, its efficiency could improve dramatically. Although there are so many advantages of the algorithm, the proposed method works as a stochastic search algorithm. Due to various randomness, if the parameter is not set properly, the solution can easily drop to a partial optimum. Therefore, the research of the ant colony algorithm needs subsequent improvements.

1.3. Related Work. Research on mobile robots began in the late 1960s and was among some of the more ambitious discoveries in human history throughout the twentieth century. This invention has undergone radical changes in a very short period of more than 50 years. Liu Yu proposed a model-based regression algorithm, constructed a scoring algorithm, and an integrated model combining clustering and regression was proposed according to the actual situation. In order to study the effectiveness of the proposed algorithm in terms of models, performance simulation experiments were designed to compare and contrast proposed algorithm-based computer algorithm models to traditional algorithm-based computer models, collected relevant empirical results, and plotted appropriate static graphs [7]. In the past decades, evolutive as well as other meta-heuristic-based methods were widely used to solve optimization problems in many different fields. On this basis, Zheng F proposed an innovative parameter adaptation policy of the ant colony optimal (ACO) algorithm, which is designed to control trajectories of convergence in the decision space based on making them follow any prespecified path [8]. Hajara I proposed fault-tolerant job scheduling using ACO in grid computing. The technique used by the authors is to employ a combination that uses server error rates and a rollback recovery strategy based on checkpoints. The purpose of checkpoints is to reduce job losses in case of system failure by immediately preserving the system state [9].

1.4. Innovation. There are three innovations in this paper. (1) It is an innovation in selecting the angle of the topic. Currently, the research that integrates artificial intelligence, ant colony optimization algorithm, volleyball robot, and palm mechanism into one is not much. This is of experimental importance. (2) It is an innovation in terms of research method. The article presents the CMA-ES algorithm, the improved ant colony algorithm, the BK algorithm, etc., which have great theory value as well as explore meaning. (3) It is an innovation in practical aspects, which overcomes the drawback that the rough set theory can only find the approximate optimal solution and can make up for the defect that the ant colony algorithm is prone to local optimal paths. It improves the search speed and avoids the duplication of the ant colony algorithm in the initial value selection to a certain extent.

2. Optimization of Motion Parameters Based on Ant Colony Algorithm

2.1. CMA-ES Algorithm. The CMA-ES algorithm is also called covariance matrix adaptation evolutionary strategies. It is mainly used to solve continuous optimization problems, especially continuous optimization problems under ill-conditioned conditions. The algorithm continuously selects the partially best individuals in a population to improve the population fitness. A comparison of the algorithm against alternative optimization algorithms shows that CMA-ES has a faster convergence rate and constant rotation. Most importantly, a smaller population size in the system can lead to an optimized system efficiently. The exact procedure of the CMA-ES optimal operation algorithm [10] includes three main parts. Initial, iterative, as well as until.

(1) Initial.

Procedure 1: randomly generate ∂ variable, $\partial = 1, 2, \ldots\ldots, \partial$ to form the initial population;

Procedure 2: initialize population mean value B, step size $\beta \in R^{+}$, and evolutionary algebra $H = 0$;

Procedure 3: initialize evolution path W_C and conjugate evolution path W_α as zero vectors, respectively;

(2) Repeat:

Procedure 4: generate the search population [11], where 7 is the i-th individual in the k-th generation population, and the function of CMA-ES to generate offspring individuals is expressed as

$$s_i^{(k)} = m^{(g)} + \mathrm{M}^{(g)} Z_i^{(g)}, \tag{1}$$

where $D_i^{(g)} - G(0, C_i^{(g)}), i = 1, 2, \ldots, \chi, F_i^{(g)}$ is an r-dimensional random vector, generated by a Gaussian function with zero mean and variance of $V_i^{(g)} \in Q^{n\times n}$.

∂^g is the step size factor [12], and m_g is the best-weighted average of the individual offspring [13, 14].

Procedure 5: select and reorganize the population, and update the mean of the search population.

$$x^{(g+1)} = \sum_{i=1}^{\beta} T_i y_i^{(g)}. \tag{2}$$

Reorganize 1, select the first N 2 in turn, let

$$k_{sel}^{(g)} = \sum_{i=1}^{\varepsilon} L_i \times k_i^{(g)}. \tag{3}$$

Procedure 6: update search paths W_C and W_α.

$$\begin{aligned} W_c^{(g+1)} &= \left(1 - j_\beta^\partial\right) W_i^{(g)} + \sqrt{n_o^{(g)} \mu_{eff}} D^{\frac{1}{2}} \langle z \rangle_{sel}^g, \\ W_n^{(g+1)} &= \left(1 - j_c\right) W_c^{(g)} + l_\partial \sqrt{n_c \mu_{eff}} \langle z \rangle_{sel}^g. \end{aligned} \tag{4}$$

Procedure 7: update step size ν and covariance matrix G.

$$\begin{aligned} \nu^{(g+1)} &= \nu^g \times erp\left(\frac{c_\partial}{d_\partial}\left(\frac{\|P_\partial\|}{W\|N(0,1)\|}\right)2\right), \\ \nu^{(g+1)} &= \left(1 - \nu_{\text{cov}}\right) + \nu_{\text{cov}}\left(1 - \frac{1}{\nu_{\text{cov}}}\right) \sum_{i=1}^{\alpha} z_i^T. \end{aligned} \tag{5}$$

(3) Until:

Procedure 8: determine the objective function (i.e., fitness function) [15] to determine the convergence conditions

$$\max\left(T\left(h_i^{(g)}\right)\right) - \min\left(T\left(h_i^{(g)}\right)\right) \leq \beta, i = 1, 2 \ldots \eta. \tag{6}$$

In other words, the difference between the minimum as well as the maxima in a population's objective function at the population sample points is less than a predetermined minimum threshold.

2.2. Revised Ant Colonies Computer System. In the elementary colony algorithm, the difference in the Euclidean function generated by the distance between adjacent plots is not obvious. The meaning of Euclid's algorithm is to quickly find the greatest common divisor of two numbers, and the final result is that the search performance becomes low. To improve this problem, borrowing the idea of adaptive construction of the evaluation function for termite swarm algorithms [16], an objective for constructing the function of termite swarm algorithms is added to the Euclidean function, and the algorithm, therefore, has a faster rate of conversion. A Euclidean search engine is just one such algorithm. The construction of a customized evaluation function using the position relationship among the present node, optional junction, and objective junction is a basic principle. A path chosen for further steps is a path having the lowest value of the evaluation function. The evaluation function is the product that is the product sum of the cost from the current T and optional M nodes and the cost from the optional M to R knots. The specific expression is as follows:

$$f(m) = g(m) + h(m). \tag{7}$$

The heuristic function of the constructed ant colony algorithm is

$$\mu_{ii}(t) = \frac{1}{g+h} = \frac{1}{d_{ii} + d_{jE}}. \tag{8}$$

The formula represents the distance from fence i to i. Suppose the ant colony is initialized [17] pheromone $\ell_{ii}(d) = c$. After n times, the ant completes a cycle, and the amount of pheromone on each path will change due to the passage of the ant and the change of time. The change rules are as follows:

$$\begin{aligned} \ell_{ii}(d+n) &= p \bullet \ell_{ii}(d) + \nabla \ell_{ii}(d, d+n), \\ \nabla \ell_{ii}(d, d+n) &= \sum_{k=1}^{m} \nabla \ell_{ii}^k (t, t+n). \end{aligned} \tag{9}$$

2.3. Basic Structure of BC Algorithm. The method of using group robots [18] for target search and hunting proposed by the BC algorithm can be divided into the following steps:

Step 1: establish an initial coordinate system according to the initial position of the robot in the target area;

Step 2: using Voronoi diagrams, it consists of a set of continuous polygons consisting of vertical bisectors of straight lines connecting two adjacent points. The N distinct points on the plane, divide the plane according to the nearest neighbor principle; each point is associated with its nearest neighbor. Divide the target area into corresponding units according to the number of robots;

Step 3: use the BC algorithm to search for each Voronoi unit in the target area [19];

Step 4: use the BC algorithm [20] to acutely round up the detected target.

The basic structure figure is shown in Figure 1.

3. Simulation Experiment Based on Artificial Intelligence

3.1. NSGA-II Optimized Parameters. The NSGA-II method is used to optimize the parameters in the swarm robot control algorithm. The NSGA-II algorithm, a fast nondominated multiobjective optimization algorithm with an elite retention strategy, is a multiobjective optimization algorithm based on the Pareto optimal solution. The concept of fast nondominated sorting is proposed, which effectively reduces the computational complexity compared with the first-generation algorithm. However, the experiment of Guo

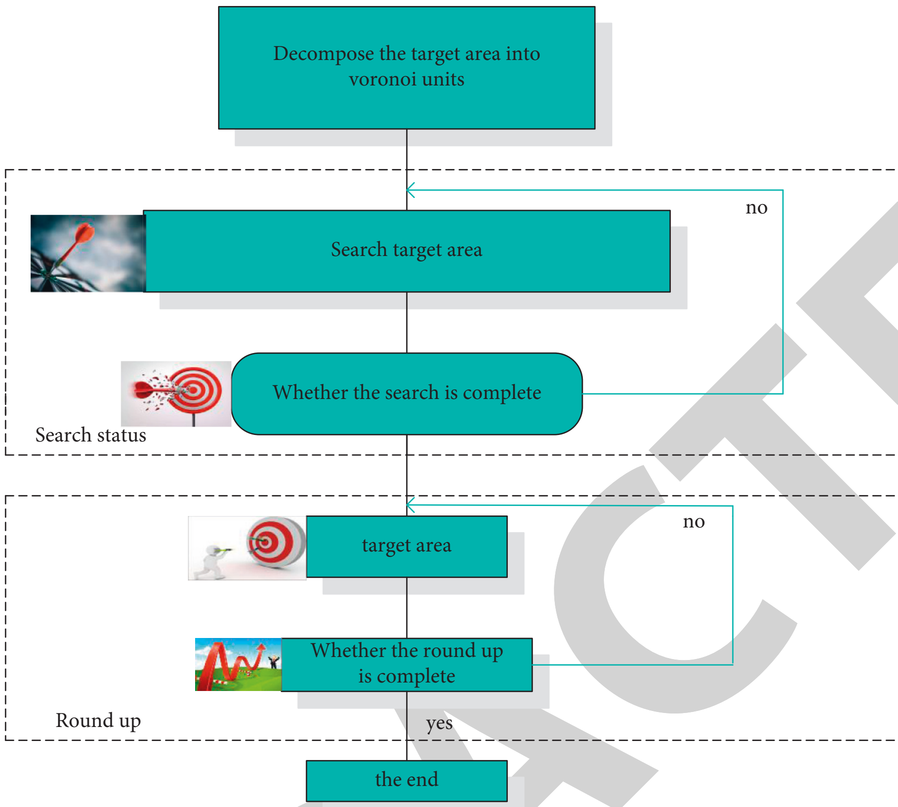

Figure 1: Flow chart of group robot target search and hunting based on BC algorithm.

et al. in the literature proves that in the corresponding swarm robot control system, when it takes more time, it does not have much to reduce the total running distance help [21–24]. The optimal path of the standard genetic algorithm will continue to be optimized as the population evolution algebra increases so that the optimal value of the path searched by the algorithm is constantly approaching the global optimal value. However, as the number of iterations [25] increases, the algorithm's calculation time will show a linear increasing trend. Although the length of the optimal path searched by the algorithm is showing a decreasing trend, there are still some shocks in this trend. This phenomenon is mainly due to the optimal path being searched by the algorithm and determined by the random rules of the algorithm. The value is more stable than in other cases, and in other cases, the trend of the optimal value searched by the algorithm shows a certain oscillating phenomenon. When the crossover probability is low, excellent genes cannot be fully utilized; when the crossover probability is high, the algorithm is easy to fall into the trap of local extremum [26], which is not conducive to the optimization of the algorithm. We use the NSGA-II [27] to GRN [28] method to solve the group robot convergence time and total travel distance, as shown in Figure 2.

In the simulation experiment, the shape of the robots we defined is a unit circle with the center at (0, 0). All robots should move toward the target unit circle and avoid collisions with each other. In the experiment, noise and delay are added to the robot's movement, so there is always a certain error between the position of the robot and the final predetermined target shape. Therefore, we can define a position error threshold constant to judge whether the robot converges. In the following research, we set the threshold to 0.125 and we define that the error between the actual distance and the theoretical distance of the neighboring robot should be less than 0.1. Let NSGA-II solve the filter parameters, and the results obtained are shown in Figure 3. The experimental results show that the problem of multiobjective optimization has become a problem of single-objective optimization. We simulated 30 independent experiments to find the best solution. At the same time, we always get similar results. The simulation results also show that the error between the best results of each experiment is very small, as shown in Figure 3 and as shown in Figure 4.

3.2. Test Experimental Data. Based on the IABC-K-means algorithm proposed by the combination of the improved artificial bee colony algorithm and the KMC algorithm, to check the optimization capability for the IABC-K-means technique, that is, for the improvement of clustering performance, the dataset in the UCI database is used for the clustering test, and the traditional K-means algorithm is compared longitudinally. First, we introduce the test experiment data. The UCI dataset is an online dataset with rich resources and dynamic development. It provides researchers with open source download and user permissions for various data, and users can also upload and enrich the dataset.

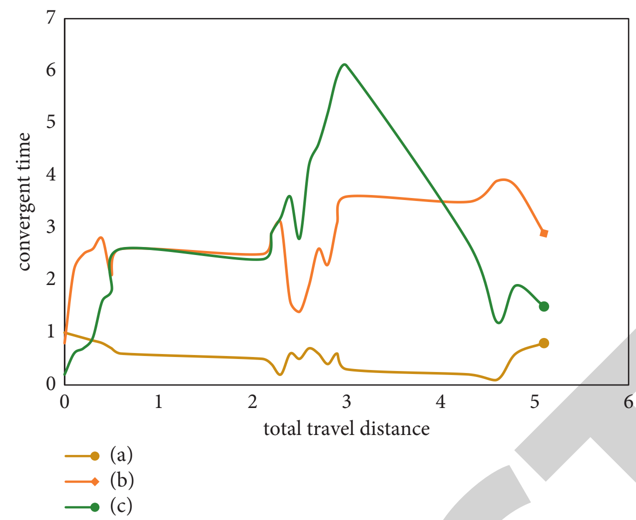

Figure 2: Using NSGA-II to optimize GRN algorithm parameters.

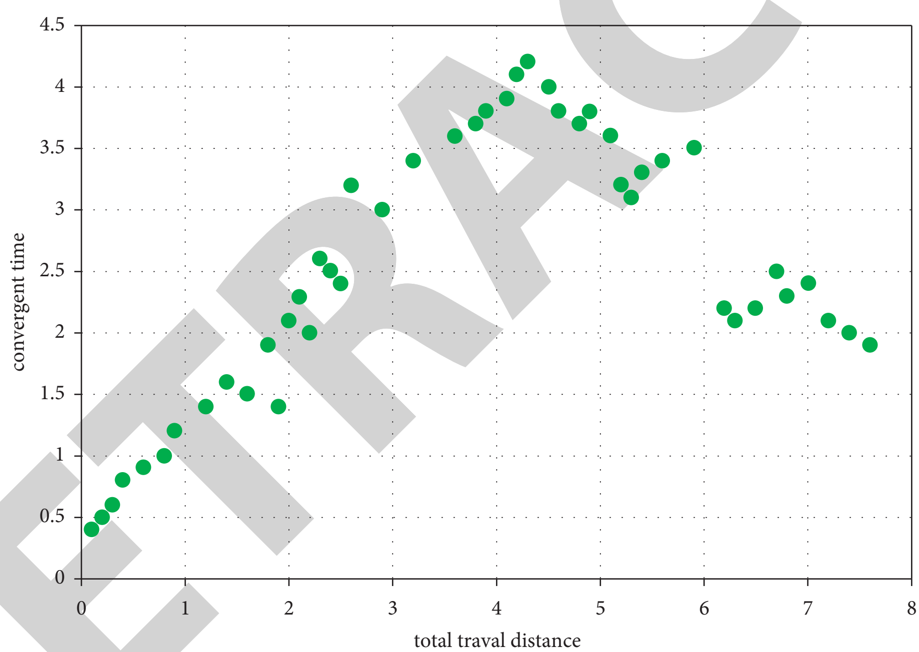

Figure 3: Using NSGA-II to optimize filter parameters.

Antenna community analysis (ACA) as well as general acceptance analysis (GAA) are similar to biology's evolutionary process. Both algorithms have implicit parallel search capabilities. The ant colony algorithm can make full advantage out of pheromone's correct response system. This speeds up the speed of convergence to the best solution with strong robustness; the genetic algorithm imitates the genetic, crossover, and mutation operations of biology, preserves the genes of excellent individuals in the evolution process, and finally converges to the optimal solution, but crosses when there are too many individuals and the mutation will become complicated. Therefore, integrating the strengths from both methods for addressing route planning problems is a popular research direction. The UCI dataset provides data in various fields, including clustering, classification, machine learning, and deep learning. The algorithm in this paper makes full use of the advantages of the ant colony algorithm and the genetic algorithm based on the better performance of the max-min ant system (MMAS), using the pheromone update of the local optimal path and the global optimal path. By adding weight factors, introducing dual ant colony thinking and cross factor, iterating the optimal path and the global optimal path, if the same part is passed, the two will be crossed to generate a new path. When there are many newly generated paths, it will be used for a more global optimal path in the short term. This paper conducts clustering tests on three datasets of iris, red wine, and new red wine, which have different sample sizes, attribute dimensions, and the number of clusters. The description of the detailed information of the dataset used is shown in Table 1.

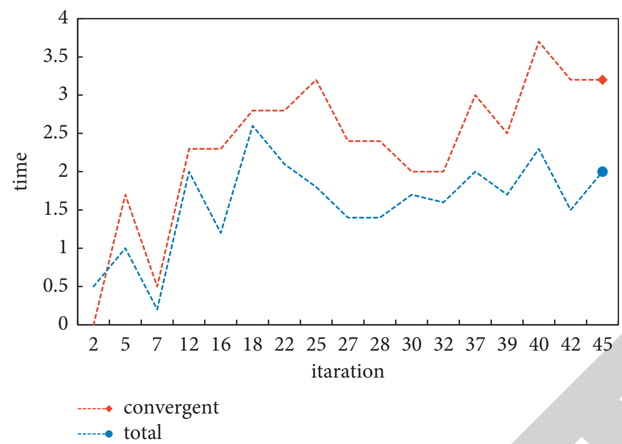

Figure 4: The results of 30 experiments using NSGA-II to optimize filter parameters.

Table 1: Dataset information used in the experiment.

Dataset name	Number of samples included	Number of sample attributes	Number of clusters
Iris	145	5	4
Red wine	174	12	4
New red wine	1688	15	7

We run the IABC-K-means algorithm and the K-means algorithm 30 times, respectively, and count the maximum, minimum, average, and standard deviation of the E value of the clustering measurement criterion and the average time consumed by the algorithm to run. The statistical data is shown in Tables 2–4:

By comparing and analyzing the data of Tables 2–4, it can be seen that the original K-means algorithm can converge quickly and output the cluster results in a short time, but the E value of the cluster measurement standard has changed a lot. Choosing different initial cluster centers for other experiments will have a greater impact on the results. In other words, the original K-average algorithm is susceptible to the influence of the initial cluster center, and it is easy to fall into the regional optimal value. The instability reflected in the standard deviation of the measurement function is greater. The IABC algorithm overcomes the disadvantage that the original algorithm is easy to fall into the local optimal solution; the IABC-K-means algorithm has better clustering quality and comprehensive performance.

3.3. Establishing the Control Panel. On the DSP control board, the main control components, communication interface, and control program of the six-degree-of-freedom welding robot are integrated. In the DSP control board experiment, the main test is to test the serial communication between the experiment board and the host PC to send control signals to the host computer software. The joint motor is a driving motor applied to the intelligent robot, which has the functions of deceleration, transmission, and torque enhancement. For the joint motors of the robot model, mainly receives the PWM control signals from the DSP. Among them, DSP refers to digital signal processing technology; the purpose is to measure or filter continuous analog signals in the real world; and PWM control signal refers to pulse width modulation, which is also an analog control method, which can make the output voltage of the power supply remain constant as operating conditions change. As the mutation rate increases, the length of the optimal path searched by the algorithm becomes smaller and smaller. However, it is observed in the experiment that as the mutation rate increases, more and more individuals participate in the mutation, and the calculation time of the algorithm will gradually increase. So, for the algorithm, although increasing the mutation rate will increase the diversity of the population, the calculation cost and calculation time of the algorithm need to be considered. The joint values of the robot's circular interpolation around the Z-axis are sent to the robot model built in the upper computer simulation software in real-time. MATLAB is a commercial mathematical software used for computer vision and signal processing. The following is the PWM control waveform generated by the DSP received by MATLAB, as shown in Figure 5.

4. Analysis of Experimental Results

4.1. Comparison and Analysis of the Optimization Process and Optimization Results of the Three Algorithms. We use the LMI toolbox in MATLAB to solve the equation, showing the theoretical trajectory of the group robot, and the trajectory affected by discrete random noise is shown in the figure. The amount of data obtained in this experiment is very large. In order to extract effective information from the experimental results for analysis, we need to look back at the purpose of this experiment in order to determine which algorithm is more widely applicable to obstacle avoidance learning of robots in the field of intelligence. That is, the overall analysis

Table 2: Comparison of clustering results of iris dataset.

Algorithm	Convergence time (s)	E minimum	E max	E average	E standard deviation
K-means	0.236	79.526	145.223	95.63	26.58
IABC-K-means	2.56	63.52	72.56	73.25	0.263

Table 3: Comparison of clustering results of red wine dataset.

Algorithm	Convergence time (s)	E minimum	E max	E average	E standard deviation
K-means	0.0052	$3.256e+6$	$2.4707e+6$	$3.3807e+6$	$2.3632e+6$
IABC-K-means	3.36	$3.365e+6$	$2.4536e+6$	$3.5207e+6$	$2.2563e+6$

Table 4: Comparison of clustering results of new red wine dataset.

Algorithm	Convergence time (s)	E minimum	E max	E average	E standard deviation
K-means	0.236	$1.235e+4$	$1.2365e+4$	$1.5369e+4$	420
IABC-K-means	12.235	$1.6350e+4$	$1.0236e+4$	$1.2365e+4$	32.56

of the results of the three swarm intelligence algorithms is carried out, and from a macro perspective, which algorithm has the better performance as a whole is compared. That is to say, for each algorithm, we compare the learning performance of the algorithm in robot group obstacle avoidance under different parameter settings.

To escape the effect that contingency has on the analysis results, from the optimal solutions of the 30 experiments of each algorithm, we arrange the optimal solutions in order from largest to smallest, and select the experiment ranked 15th. The 50 iterations of the optimization process of the population intelligence algorithm are compared, and the results are displayed in the histogram of the following figure, as shown in Figure 6.

As shown in the figure, the GWO algorithm has the problem of premature convergence. Although GWO has a very good initial solution, it lasts for more than 12 iterations of the initial solution, resulting in unsatisfactory final experimental results. The BA algorithm has not improved after the 25th iteration, so the final experimental results are not as ideal as PSO. Obviously, the PSO algorithm performs more efficiently and better in the optimization process. In addition, the line graph also very intuitively illustrates the performance of the three swarm intelligence algorithms in the optimization process.

The mean, standard deviation, and minimum values are higher than the BA and GWO algorithms, except for the maximum value of the PSO algorithm, which is slightly smaller than that of the BA algorithm. Based on the GWO algorithm, the PSO algorithm performed better overall than the BA algorithm and the GWO algorithm, while the BA algorithm was superior. In addition, the BA algorithm achieved the maximum value, which was the best optimization result. Compared with the PSO algorithm and the BA algorithm, the GWO algorithm was more stable in the optimization results across all experiments.

4.2. Comparative Analysis of the Performance Results of the Three Algorithms under Different Parameter Settings. In this section, we want to compare the performance of the three algorithms in more detail under different NR and CR. To this end, we took out the final solution of each experiment from 26 sets of experimental data for data analysis. The abscissa is the BA algorithm, the GWO algorithm, and the PSO algorithm for comparison, and the ordinate is the optimal objective function of each experiment. Graph 7 is drawn, where triangles represent the average of thirty repeated experiments, and dots represent abnormal values. Comparing the experimental results of the three algorithms under different combinations of NR and CR, it is actually to compare the data of the red box, blue box, and green box in each group, as shown in Figure 7.

Through the above analysis and comparison of the experimental data, we can summarize the analysis of the experimental results as follows: in general, the performance of the PSO algorithm is better than the BA algorithm, and the performance of the BA algorithm is better than the GWO algorithm, but when there are 14 robots and the communication range is 0.7, the performance of GWO algorithm is better than BA algorithm. When there are 14 robots and the communication range is 1.6, the performance of the GWO algorithm is better than that of the PSO and BA algorithms. Since the PSO algorithm stands out in the overall performance of the three swarm intelligence algorithms and the GWO algorithm performs mediocrely, we can conclude that (1) the swarm intelligence algorithm with social learning ability and self-awareness is good at avoiding obstacles in the robot population. It will provide better learning performance. (2) When the number of robots is insufficient and the communication range is not large, the swarm intelligence algorithm with only social learning ability does not perform as well as the group with mixed learning ability in obstacle avoidance in the robot population.

4.3. Experimental Results. From previous studies, it can be found that the basic ant colony algorithm is particularly prone to falling into local traps in the middle of the path search process. The reason for this is usually due to the influence of objective heuristic functions in the path search process by ants, which prematurely focuses most of the

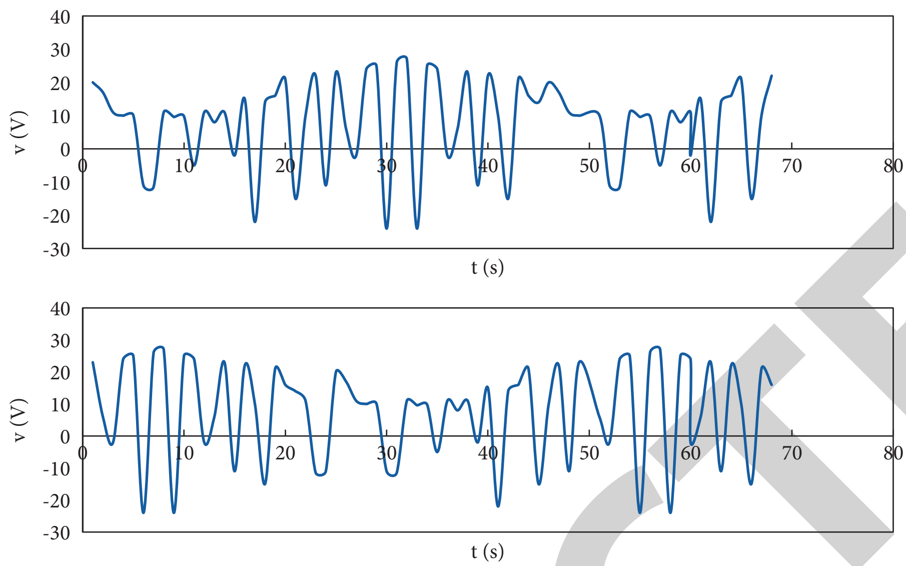

Figure 5: PWM control waveform.

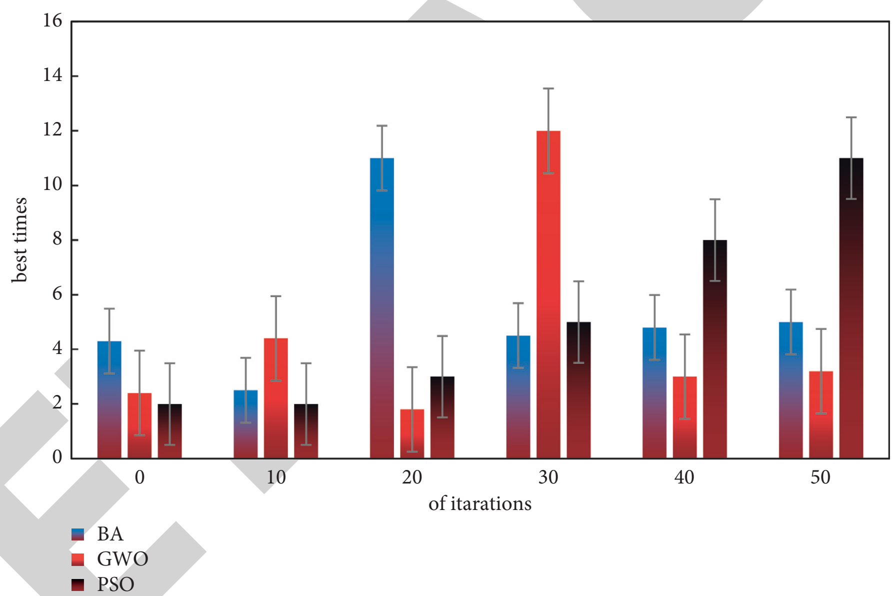

Figure 6: Data diagram of BA, PSO, and GWO.

algorithm on a few suboptimal solutions due to the emphasis of pheromones, and the randomness of the algorithm is greatly affected in this case. The classification ant colony algorithm, in order to cope with this situation, divides the ant colony into two classes and designs different heuristic functions for each class to ensure the convergence speed of the algorithm while considering the objectivity and randomness of the algorithm, so that the diversity of the algorithm solutions is guaranteed and the optimization ability of the algorithm is improved. As mentioned above, the algorithm is simulated and validated in a relatively large-scale, complex grid environment. The convergence curve of the basic ant colony algorithm shows that the ant colony is prone to stagnation in the path search process, resulting in the final solution of the algorithm often being nonoptimal and the convergence speed of the algorithm being affected. The convergence curve of the improved ant colony algorithm clearly shows that the stagnation phenomenon of the colony in the path search process has been effectively eliminated and the convergence speed of the algorithm has been significantly improved. Its path planning diagram is shown in Figure 8.

A path planning simulation study was conducted for the improved and basic colony arithmetic in various grid environments. The simulation results show that the basic colony-based approach converges faster than the categorical ant colony search algorithm, which is mainly due to the improved colony approach. As a result, a dual-feedback

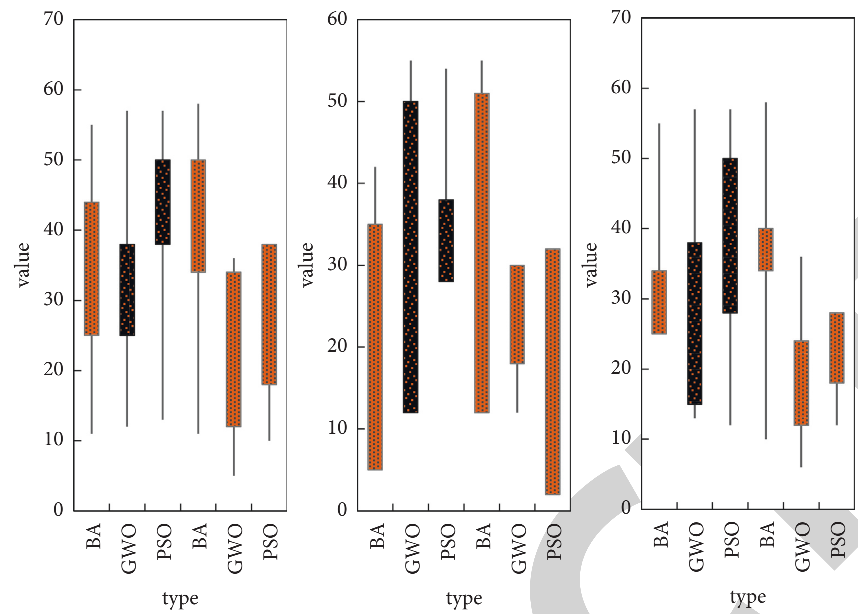

Figure 7: Comparative analysis diagram of BA, GWO, and PSO.

Figure 8: Path planning diagram.

pheromone update strategy is obtained. The new fiducial renewal policy enhances the current solution of the algorithm obtained by ants during the search process rather than the optimal path so far, but also enhances the algorithm that is inferior to the optimal path so far obtained by the ant during the search process. The current solution attenuates the pheromone so that even in a very complex grid environment, the ant colony algorithm can quickly search for a better path. This point can be clearly seen in the simulation of a larger, complex grid environment. The path planning research carried out by the improved algorithm that introduces the classification ant colony search strategy ensures that the algorithm results are excellent.

5. Conclusions

In this paper, based on the theoretical basis of the two algorithms, the two algorithms are combined with each other, the principles and basic concepts of the artificial immunity-ant colony algorithm are clarified in the original concept, and the detailed design is carried out to obtain the artificial immunity-ant colony (AIAC) algorithm. This algorithm uses the ant colony algorithm to find the optimal path in the population obtained by artificial immunity. By simulating a different number of multiple groups of nodes, and comparing other algorithms under the same conditions, it proves the artificial immunity-ant group algorithm can effectively optimize the path of a wireless sensor network and has a considerable improvement in energy and delay compared with other algorithms. In this paper, the ant colony algorithm is improved when the global information is known. Comparative experiments of the algorithm were conducted using MATLAB, and the improved algorithm was effectively proven to be effective by the experimental results. However, the effectiveness of the algorithm in practical scenarios needs to be further explored. It is necessary to conduct experiments in the real environment on the robot platform. In recent years, the improvement of algorithms has become more and more inclined to the mutual combination of intelligent algorithms. The mutual combination of intelligent algorithms can achieve the result of maximizing strengths and avoiding weaknesses to avoid the shortcomings of the algorithm.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares that there are no conflicts of interest with this article.

References

[1] Y. Qiao, Y. Li, D. Chen, S. Serikawa, M. Guizani, and Z. Lv, "A survey on 5G/6G, AI, and Robotics," *Computers & Electrical Engineering*, vol. 95, p. 107372, 2021.

[2] M. Pan, Y. Liu, J. Cao, Y. Li, C. Li, and C.-H. Chen, "Visual recognition based on deep learning for navigation mark classification," *IEEE Access*, vol. 8, pp. 32767–32775, 2020.

[3] Q. Wang, Z. Mu, and L. Jin, "Control method of robot detour obstacle based on eeg," *Neural Computing & Applications*, pp. 1–8, 2021.

[4] Z. Lv and L. Qiao, "Deep belief network and linear perceptron based cognitive computing for collaborative robots," *Applied Soft Computing*, vol. 92, p. 106300, 2020.

[5] S. Wan, Z. Gu, and Q. Ni, "Cognitive computing and wireless communications on the edge for healthcare service robots," *Computer Communications*, https://doi.org/10.1016/j.comcom.2019.10.012, 2019.

[6] S. Wan and S. Goudos, "Faster R-CNN for multi-class fruit detection using a robotic vision system," *Computer Networks*, vol. 168, p. 107036, 2019.

[7] Y. Liu and J. Wang, "Research on post-match score mechanism of players based on artificial intelligence and clustering regression model," *Journal of Intelligent and Fuzzy Systems*, vol. 39, no. 4, pp. 1–11, 2020.

[8] F. Zheng, A. C. Zecchin, J. P. Newman, H. R. Maier, and G. C. Dandy, "An adaptive convergence-trajectory controlled ant colony optimization algorithm with application to water distribution system design problems," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 5, pp. 773–791, 2017.

[9] I. Hajara, A. E. Ezugwu, S. B. Junaidu et al., "An improved ant colony optimization algorithm with fault tolerance for job scheduling in grid computing systems," *Plos One*, vol. 12, p. e0177567, 2017.

[10] L. Jia and L. Li, "Research on core strength training of aerobics based on artificial intelligence and sensor network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, pp. 1–16, 2020.

[11] W. Hu, K. Wu, P. P. Shum, N. I. Zheludev, and C. Soci, "All-optical implementation of the ant colony optimization algorithm," *Scientific Reports*, vol. 6, no. 1, p. 26283, 2016.

[12] V. Sharma and A. Grover, "A modified ant colony optimization algorithm (mACO) for energy efficient wireless sensor networks," *Optik*, vol. 127, no. 4, pp. 2169–2172, 2016.

[13] Y. Moon, H. Yu, J.-M. Gil, and J. Lim, "A slave ants based ant colony optimization algorithm for task scheduling in cloud computing environments," *Human-centric Computing and Information Sciences*, vol. 7, no. 1, p. 28, 2017.

[14] D. Ping and A. I. Yong, "Research on an improved ant colony optimization algorithm and its application," *International Journal of Hospitality Information Technology*, vol. 9, no. 4, pp. 223–234, 2016.

[15] I. Saouane, A. Chaker, B. Zaidi, and C. Shekhar, "Optimal angle of polycrystalline silicon solar panels placed in a building using the ant colony optimization algorithm," *The European Physical Journal Plus*, vol. 132, no. 3, p. 106, 2017.

[16] K. Singhal and S. Sahu, "Fire evacuation using ant colony optimization algorithm," *International Journal of Computer Application*, vol. 139, no. 8, pp. 17–20, 2016.

[17] B. Song, H. Miao, and L. Xu, "Path planning for coal mine robot via improved ant colony optimization algorithm," *Systems Science & Control Engineering*, vol. 9, no. 1, pp. 283–289, 2021.

[18] P. Voosen, "Safely settled, InSight gets ready to look inside Mars," *Science*, vol. 362, no. 6418, pp. 979-980, 2018.

[19] A. Youssef, M. El-Telbany, and A. Zekry, "The role of artificial intelligence in photo-voltaic systems design and control: a review," *Renewable and Sustainable Energy Reviews*, vol. 78, no. oct., pp. 72–79, 2017.

[20] Osamah Ibrahim Khalaf, "Carlos andrés tavera romero, A. Azhagu jaisudhan pazhani, G. Vinuja, "VLSI implementation of a high-performance nonlinear image scaling algorithm"," *Journal of Healthcare Engineering*, vol. 2021, p. 10, 2021, https://doi.org/10.1155/2021/6297856 6297856.

[21] O. I. Khalaf, G. M. Abdulsahib, H. D. Kasmaei, and K. A. Ogudo, "A new algorithm on application of blockchain technology in live stream video transmissions and telecommunications," *International Journal of E-Collaboration*, vol. 16, no. 1, pp. 16–32, 2020.

[22] O. I. Khalaf, G. M. Abdulsahib, and B. M. Sabbar, "Optimization of wireless sensor network coverage using the bee algorithm," *Journal of Information Science and Engineering*, vol. 36, no. 2, pp. 377–386, 2020.

[23] M. Rajalakshmi, V. Saravanan, V. A. C. Arunprasad, and O. I. Khalaf, "Machine learning for modeling and control of industrial clarifier process," *Intelligent Automation & Soft Computing*, vol. 32, no. 1, pp. 339–359, 2022.

[24] N. Syam and A. Sharma, "Waiting for a sales renaissance in the fourth industrial revolution: machine learning and artificial intelligence in sales research and practice," *Industrial Marketing Management*, vol. 69, pp. 135–146, 2018.

[25] H. Lu, Y. Li, C. Min et al., "Brain intelligence: go beyond artificial intelligence," *Mobile Networks and Applications*, vol. 23, no. 7553, pp. 368–375, 2017.

[26] M. Moravík, M. Schmid, N. Burch et al., "DeepStack: expert-level artificial intelligence in No-limit poker," *Science*, vol. 356, no. 6337, p. 508, 2017.

[27] D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick, "Neuroscience-inspired artificial intelligence," *Neuron*, vol. 95, no. 2, pp. 245–258, 2017.