# Development of Intelligent Learning Model Based on Ant Colony Optimization Algorithm

Xiaojing Guo[1]*, Xiaoying Zhu[2], Lei Liu[3]
College of Technology and Data, Yantai Nanshan University, Yantai 265700, China

***Abstract*—In the process of the gradual popularization of online courses, learners are increasingly dissatisfied with the recommendation mechanism of imprecise courses when faced with a large number of course choices. How to better recommend relevant courses to targeted users has become a current research hotspot. An intelligent learning model based on ant colony optimization algorithm is introduced, which can accurately calculate the similarity between courses and learners. After structured classification, the model recommends courses to learners in the optimal way. The results showed that the accuracy of this method reached 10-20 when tested in Sphere and Ellipse functions, and the optimal solution for problem Ulysses21 was 27, which was better than Advanced Sorting Ant System (ASrank), Maximum Minimum Ant System (MMAS), and Ant System (AS) based on optimization sorting. The proposed ant colony optimization algorithm had better convergence performance than ASrank, MMAS, and AS algorithms, with a shortest path of 53.5. After reaching Root Mean Square Error (RMSE) and Relative Deviation (RD) distributions of 6% and 8%, the stability of the proposed method no longer decreased with increasing RMSE. The accuracy did not vary significantly with changes in the dataset, and the reproducibility performance was better than other comparison models. In the scenarios of path Block and path Naive, the proposed algorithm had an average computation time of only 1011, which was better than the Ant Colony Optimization (ACO) and Massive Multilingual Speech (MMS) models. Therefore, the proposed algorithm improves the performance of intelligent learning models, solves the problem of local optima while enhancing the convergence efficiency of the model, and provides new solutions and directions for increasing the recommendation performance of online learning platforms.**

***Keywords*—*Online courses; ant colony optimization algorithm; intelligent learning model; path planning; local optimum***

## I. INTRODUCTION

With the development of the Internet, online courses have become more and more popular, which has greatly promoted the development of education in China [1-2]. The booming development of online courses has led to a rapid increase in the number of online courses. However, with the explosive growth of online courses, students are facing unprecedented information overload problems. Traditional recommendation systems often rely on simple user behavior or content features for course matching, making it difficult to accurately associate personalized student needs with corresponding courses, resulting in the dilemma of "rich information but difficult selection" [3-4]. The mismatch between students and courses not only reduces students' learning efficiency, but also restricts the further improvement of the service quality of online education platforms [5]. Therefore, there is an urgent need for an intelligent course recommendation system to solve the above problems.

To address the above challenges, more and more experts and scholars are combining machine learning with path planning, aiming to achieve more efficient personalized recommendation strategies. Liu Y and other researchers proposed a Levy-based ACO algorithm. This algorithm utilized the Levy flight model to expand its search range while searching for paths. The results indicated that its search performance was superior to existing travel agent path planning algorithms [6]. Gao W and other researchers proposed a new ACO algorithm. This algorithm utilized the combination search function to solve the problem of spatial complexity. The results showed that the algorithm outperformed existing common ant colony algorithms in terms of convergence speed and search efficiency [7]. Liu Y and other researchers proposed an ant colony algorithm based on greed and Levy flight improvement. This algorithm adopted pseudo randomness and balanced speed and space to solve the local optimal problem of path search. The results showed that the algorithm performed better than other algorithms when applied in travel sales scenarios [8]. Stodola P and other scholars proposed a car path planning algorithm based on ant colony algorithm. This algorithm used the Cordreau benchmark instance to solve the problems of other methods. The results showed that this algorithm had higher accuracy and precision than other related algorithms [9]. Zhang G et al. proposed an ACO algorithm based on genetic variation, which introduced methods such as crossover, recombination, and mutation in genetics to solve the problem of ship meteorological route planning. The results indicated that this algorithm had higher accuracy and applicability than other algorithms [10].

In recent years, many scholars have devoted themselves to building intelligent learning models to improve the existing performance of algorithms in data similarity calculation research. Intended to enhance the ability of intelligent learning models in processing large-scale complex data, improving classification accuracy, and accelerating computational efficiency. Li W and other researchers proposed a learning classification method based on intelligent optimization algorithms. This method combined operators to assign learning mechanisms to specific learning abilities, achieving analysis and classification of complex learning scenarios. As a result, this algorithm achieved better classification accuracy and reliability than other algorithms [11]. Ge Q and other scholars proposed an intelligent learning model for path

*Corresponding Author

aggregation through an efficient single source VecSim algorithm. This model used a threshold filtering algorithm to remove items with low similarity using a set threshold, and used a sampling algorithm to estimate the probability of encountering additional paths. The results indicated that the proposed method could effectively accelerate the calculation speed, with a query speed of 0.1 seconds and an error of only $10^{-4}$ [12]. Aggarwal K et al. developed an intelligent learning model based on artificial intelligence, machine learning, and deep learning. This model could effectively identify, analyze, and made decisions when facing large amounts of complex data, increasing the automation of data processing. The results indicated that the algorithm achieved good data analysis performance in the healthcare field [13]. Janiesch C et al. proposed an intelligent machine learning model based on neural networks, which could effectively distinguish relevant concepts in the fields of electronics and network business. The results indicated that the model outperformed traditional data analysis methods and shallow machine learning models in terms of performance. Yang K et al. proposed a federated machine learning model based on aerial computing. This model utilized multiple access channels combined with stacked waveforms to overcome the bottleneck of machine learning model aggregation. The results indicated that the model increased signal propagation strength and reduced model aggregation error [14]. Mousavinasab E et al. proposed a machine learning model based on action condition rule reasoning, which could redefine learners, classify and aggregate them. The results indicated that this method could achieve personalized learning recommendations and promote the application of learning in physics, chemistry, and clinical fields [15].

The existing models still face many challenges in dealing with the complex user behavior patterns and course features unique to online education platforms, such as the dynamic changes in user interests, the diversity of course content, and the real-time nature of recommendation results. Therefore, the study introduces the Negative Feedback Ant Colony Algorithm (NFACA) and the construction of intelligent learning models to be applied to the platform course recommendation platform, helping to improve the performance of platform recommendation algorithms. The innovation of this research lies in the development of an intelligent learning model based on NFACA mechanism, which improves the efficiency and accuracy of path planning and provides learners with more targeted course recommendations. The research contribution lies in proposing an intelligent recommendation model based on NFACA, which provides a new idea and method for course recommendation on online education platforms, solves the problems of low accuracy and efficiency of traditional recommendation algorithms, and provides reference and inspiration for future recommendation system development.

## II. Methods and Materials

Firstly, an NFACA is proposed to explore the updating effects of the worst and best solutions in the negative feedback mechanism on pheromone concentration. After defining the learner, a course classification model is constructed based on the calculation method of pheromone concentration, and the intelligent learning model is optimized to improve its performance and find the optimal solution in a shorter time.

### A. Design of NFACA Mechanism

Due to the rapid development of society, machine intelligence learning-related technologies have also been further improved. Introducing path planning to find the optimal solution on online learning platforms will become a new driving force for promoting educational progress. The ACO algorithm is inspired by the foraging process of ants, where food always appears randomly. Therefore, the way it selects the optimal path when searching for food has reference significance for model construction in various fields [16]. The use of the ant colony algorithm for path planning mainly solves two problems: optimal path selection and pheromone updating [17]. The selection of the optimal path involves the probability of choosing the next location from one location, as shown in Eq. (1).

$$P_{ij}^{k}(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}\times[\eta_{ij}(t)]^{\beta}}{\sum\limits_{k\in allowed_k}[\tau_{ij}(t)]^{\alpha}\times[\eta_{ij}(t)]^{\beta}}, \text{if jeallowed}_k \\ 0, \text{otherwise}\end{cases} \tag{1}$$

In Eq. (1), represents the concentration of pheromones; $t$ is the time for selecting the next location; $i$ is the current position; $j$ is the next selected location $\tau_{ij}(t)$ at this moment; $\eta_{ij}(t)$ is the visible range of the ant's next location selection; represents the weight of information concentration; $\beta$ represents f $\alpha$ actors that affect information concentration; $\text{allowed}_k$ is the total number of locations that the $k$ th ant has not yet reached; $P_{ij}^{k}(t)$ represents the probability of ants choosing the next location at this moment. After passing through ants multiple times on a certain road, the concentration of pheromones updates as shown in Eq. (2).

$$\tau_{ij}(t+1)=(1-\rho)\times\tau_{ij}(t)+\sum_{k=1}^{m}\Delta\tau_{ij}^{k}(t) \tag{2}$$

In Eq. (2), $\tau_{ij}(t+1)$ is the pheromone concentration of the $ij$ channel at time $\tau_{ij}(t+1)$ ; $\Delta\tau_{ij}^{k}(t)$ represents the concentration of pheromones left by ant $k$ on the $ij$ road; $m$ represents the total number of ants passing through $ij$ road; $\rho$ represents the concentration of effective information. The overall process of ant colony algorithm is shown in Fig. 1 [18].

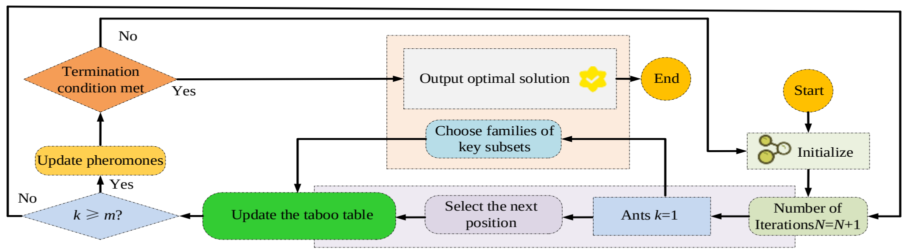


Fig. 1. General flow of ant colony algorithm.

The basic ant colony algorithm takes all the pheromones released by ants as references, which can lead to the algorithm quickly getting stuck in local optima. Therefore, the pheromones released by ants are weighted to avoid their limited field of view and see more possible paths. The weighting formula is shown in Eq. (3).

$$\tau_{ij}(t)=(1-\rho)\tau_{ij}(t)+\sum_{k=1}^{\omega-1}(\omega-k)\Delta\tau_k(i,j)+\omega\Delta\tau_b(i,j) \tag{3}$$

In Eq. (3), $w$ represents the number of ants selected to release pheromones based on their path planning distance sorting; $\Delta\tau_b(i,j)$ represents the concentration of pheromones released by ants that have planned the optimal path; $\Delta\tau_k(i,j)$ represents the concentration of pheromones released by the total number of ants that passed through before $k$. Only ants that find the optimal path are selected to release pheromones, which can avoid the algorithm's "premature" mechanism in selecting the optimal path. The pheromone optimization formula is shown in Eq. (4).

$$\tau_{ij}(t)=(1-\rho)\tau_{ij}(t)+\Delta\tau_{ij}^{best} \tag{4}$$

In Eq. (4), $\Delta\tau_{ij}^{best}$ is the total concentration of pheromones on the optimal path chosen by ants. The above introduces thinking about optimal path planning from the perspective of the optimal solution, and there is also a reverse thinking entry point: the worst solution, also known as negative feedback. Negative feedback can be used to avoid falling back into the worst solution next time, just like guiding ants to the optimal solution. The formula for the negative feedback model is shown in Eq. (5).

$$\mathbf{Y}=[\delta-\varphi_{ij}(t)]^{\gamma} \tag{5}$$

In Eq. (5), $\delta$ represents the upper limit of negative feedback pheromones released on the worst planned path; $\varphi_{ij}(t)$ represents the worst path pheromone matrix; $\gamma$ represents the weight of negative feedback pheromones on the path; $Y$ represents the probability of negative feedback. The calculation steps of the negative feedback algorithm are shown in Fig. 2.

In the process of combining the worst and best solutions for path planning, different pheromones are used to select paths, and trajectories that are not within the optimal and worst ranges are updated, as shown in Eq. (6) [19].

$$\xi=\frac{L_{\text{best}}}{L_{\text{better}}} \tag{6}$$

In Eq. (6), $L_{best}$ represents the optimal path set that conforms to the concentration of pheromones; $L_{better}$ represents the optimal path set that conforms to the information concentration; $\xi$ represents the relationship and similarity between the optimal and optimal paths. The paths that are not within the optimal and better range are updated, as shown in Eq. (7).

$$\chi_{ij}(t+1)=(1-\rho(t))\times\chi_{ij}(t)+\Delta\chi_{ij} \tag{7}$$

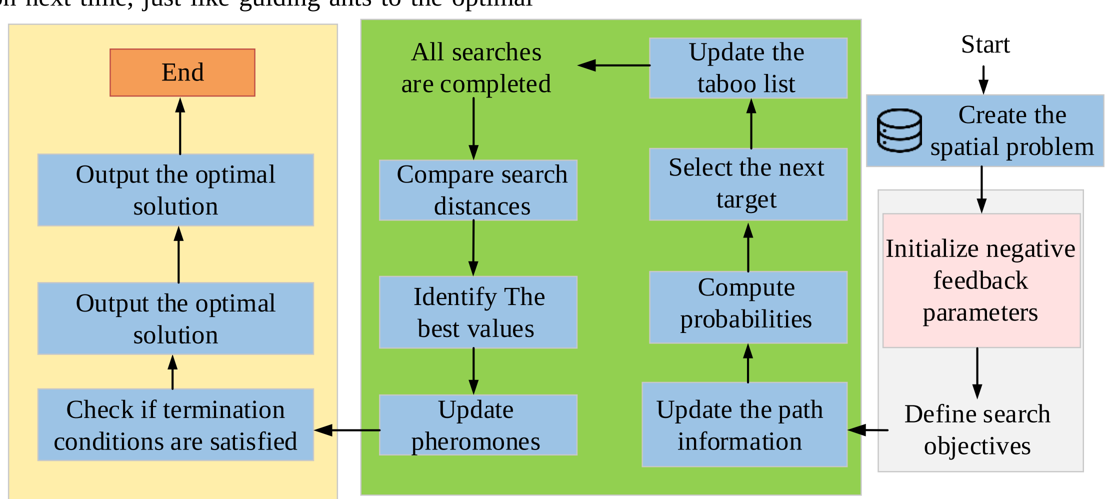


Fig. 2. Computational steps of the negative feedback algorithm.

In Eq. (7), $\chi_{ij}(t+1)$ is the total penalty amount on the channel during $t+1$ time; $\chi_{ij}(t)$ is the total amount of punishment on the ij channel within time t; $\Delta\chi_{ij}$ is the total amount of punishment that varies per unit time; $\rho(t)$ is the amount of remaining pheromone loss within time t [20].

## *B. Development of Intelligent Learning Path Recommendation Model Based on ACO Algorithm*

The primary issue to be addressed in the construction of intelligent learning path recommendation models is the psychological burden that learners face when dealing with a large number of courses. Because a large number of unsorted or unclassified courses are directly pushed without being classified and organized by algorithms, it can cause difficulties for learners to make choices. To structurally classify massive courses and push them along the optimal path, the first step is to classify learners, which can be achieved through accurate descriptions of their learning status [21]. The description of learning state is defined based on the learner's mastery of the course, as shown in Fig. 3.

As can be seen from Fig. 3, the learner's state is divided into start, adaption, and adjustment according to its definition. After defining learners, it is necessary to clarify the calculation of pheromone concentration in ant colony algorithm, which is of great significance for the probability of selecting which path in the path recommendation process. The concentration of pheromones is calculated based on the learner's historical learning curriculum set. The concentration of pheromones on this path not only needs to divide learners into corresponding structural models based on their learning status, but also needs to divide the course into modules corresponding to learners. Therefore, the intelligent learning model requires three datasets for course recommendation: learner dataset, learning course dataset, and pheromone concentration database. The recommendation process is shown in Fig. 4.

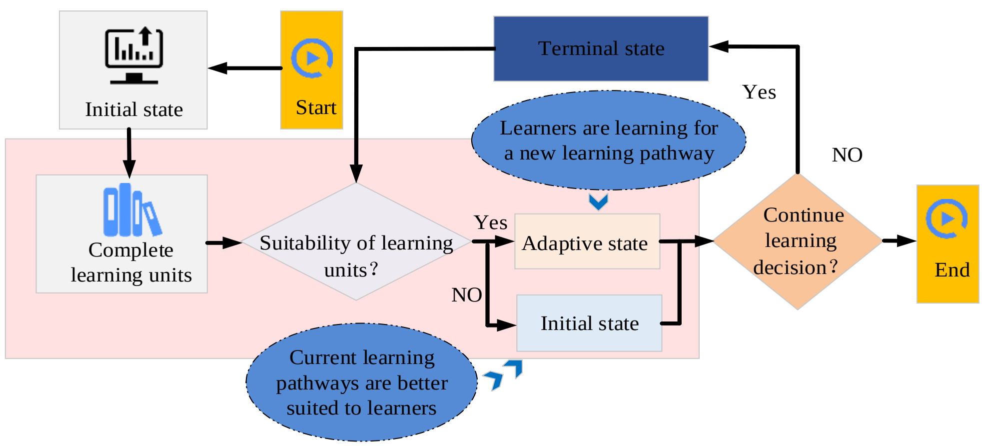


Fig. 3. The transfer diagram of learning states.

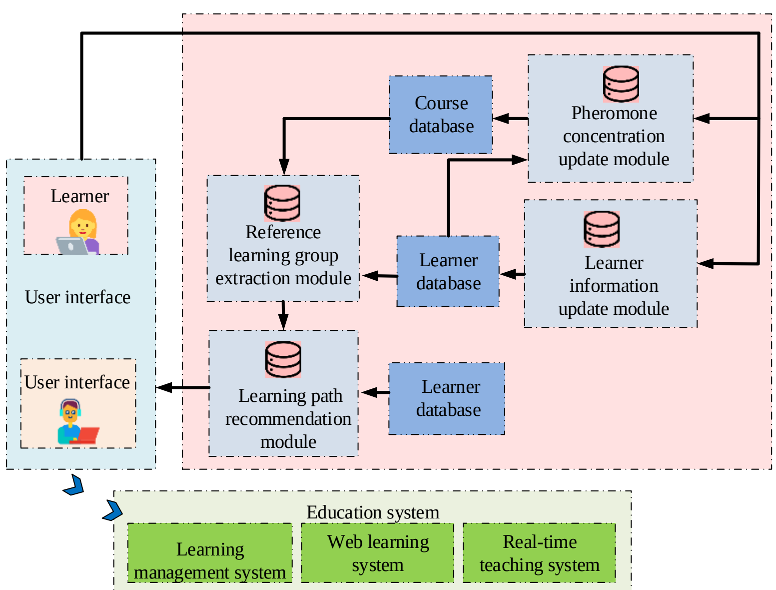


Fig. 4. Study program recommended path.

Learner data information needs to be kept updated, which is the basis for selecting the latest path based on positive and negative feedback. This ensures that each recommended path is not dependent on historical judgments and can be adaptively adjusted based on data updates. This data update requires continuous similarity calculation of learner information, as shown in Eq. (8).

$$\mathrm{Dis}\left(\mathrm{LA}_i, \mathrm{LA}_j\right) = \mathrm{Dis}\left(\mathrm{LD}_i, \mathrm{LD}_j\right) \times \mathrm{Dis}\left(\mathrm{E}_i, \mathrm{E}_j\right) \tag{8}$$

In Eq. (8), $\mathrm{LA}_i$ and $\mathrm{LA}_j$ respectively represent two different learners; $\mathrm{LD}_i$ and $\mathrm{LD}_j$ respectively represent the areas that two different learners are interested in learning; $\mathrm{E}_i$ and $\mathrm{E}_j$ respectively represent the current level of knowledge that two different learners have in the field they are interested in learning [22]. The relationship between learners and learning domains belongs to a one-to-one correspondence, as calculated in Eq. (9).

$$\eta_{ik}(t) = \sum_{\mathrm{LUA}_j \in \mathrm{Sim}(\mathrm{LA}_i)} \mathrm{Dis}(\mathrm{LA}_i, \mathrm{LA}_j) \times \mathrm{Count}(\mathrm{LA}_j, \mathrm{LUA}_k) \tag{9}$$

In Eq. (9), $\eta_{ik}(t)$ represents similarity; $\mathrm{LUA}_k$ represents the overall learning area; $\mathrm{LUA}_j$ represents the current learning field; $\mathrm{Sim}(\mathrm{LA}_i)$ represents the learning group. The probability that the learner has previously learned knowledge in the learning field is shown in Eq. (10) [23].

$$P_{ik}(t) = \begin{cases} \dfrac{[\eta_{ik}(t)]^a}{\sum_k [\eta_{ik}(t)]^a} \\ 0 \end{cases} \tag{10}$$

In Eq. (10), $P_{ik}(t)$ represents the probability that the learner has learned the selected learning area within time $t$, the probability of not learning is 0, and the probability of learning is $\dfrac{[\eta_{ik}(t)]^a}{\sum_k [\eta_{ik}(t)]^a}$; $\alpha$ represents the weight of similarity; $k*$ represents the number of learners and learning domains. The similarity estimation between learning units is shown in Eq. (11) [24].

$$\Phi_{jk} = \mathrm{Dis}(\mathrm{UA}_j, \mathrm{UA}_k) \times \mathrm{Dis}(\mathrm{C}_j, \mathrm{C}_k) \tag{11}$$

In Eq. (11), $UA_j$ and $UA_k$ respectively represent the number of similar contents present in the learning unit, while $C_j$ and $C_k$ respectively represent the degree of correlations between learning neighborhoods. The representation of pheromone concentration is the core of path planning, and the calculation method is shown in Eq. (12) [25].

$$\varepsilon = (\Delta\tau_{jk}(t) - 60) / (90 - 60) \tag{12}$$

In Eq. (12), when $\Delta\tau_{jk}(t)$ represents the pheromone variation value, and its range is between [60, 90]. $\varepsilon$ represents the adjustment factor. According to the change value of pheromones, learners are scored, and the current learning path is adjusted using adjustment factors. The process of judging the current level of knowledge mastery of learners is shown in Fig. 5.

When recommending learning paths based on the initial learning state of learners, due to their low or zero knowledge level and inability to rely on the calculation of the previous path or pheromone, it is necessary to choose based on the degree of matching with the course database in the model. The recommendation for learners in the initial state is shown in Eq. (13).

$$p_{oi} = \sum_{r \neq o} \lambda_{or} \cdot \theta_{ri} \tag{13}$$

In Eq. (13), $p_{oi}$ represents the similarity between the learner in the starting learning state and the learning unit; $\lambda_{or}$ is the similarity between learner $S_o$ and learner $S_r$; $\theta_{ri}$ is the similarity between learner $S_r$ and learning unit $L_i$. When recommending to learners who are in an adaptive state and are satisfied with the current course, it is only necessary to recommend units based on the adjustment factor, as shown in Eq. (14).

$$p_{ij}^{S_o}(t) = \begin{cases} \dfrac{\left[\eta_{ij}\right]^{\alpha^*} \times \left[\tau_{ij}(t)^*\right]^{\beta^*}}{\sum\left[\eta_{ij}\right]^{\alpha^*} \times \left[\tau_{ij}(t)^*\right]^{\beta^*}}, L_j | \ \phi_{ij}, \ \varepsilon \\ 0, L_j | \ \phi_{ij} > \varepsilon \end{cases} \tag{14}$$

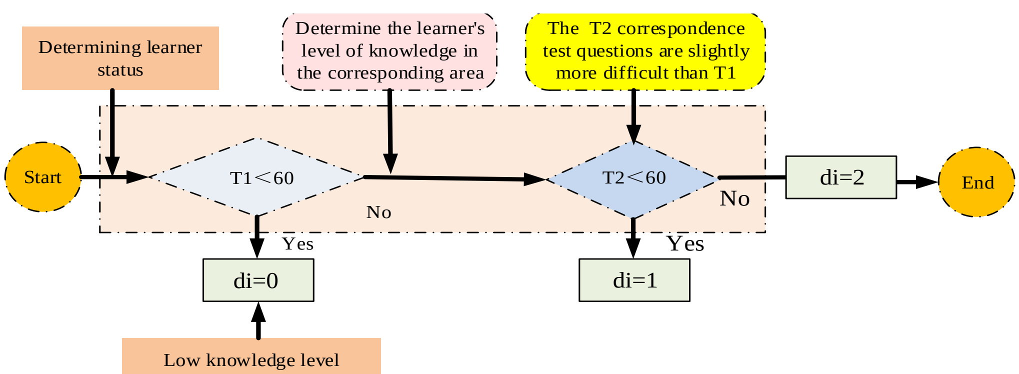


Fig. 5. Flowchart for judging the level of knowledgeable people.

In Eq. (14), $p_{ij}^{S_o}(t)$ represents the probability of $S_o$ learners learning from $L_i$ unit to $L_j$ unit during $t$ time; $\eta_{ij}$ represents the pheromone between two units $ij$ at time $t$; $\tau_{ij}(t)^*$ represents the pheromone density of the path connecting two units $ij$ at $t$ time; $\alpha^*$ and $\beta^*$ respectively represent the influencing factors of path pheromone density and learners' level of knowledge mastery. $\phi_{ij}$ represents the degree of overlap between the feature values of two learning courses in $ij$. When recommending learners who have adjusted their status, it indicates that the learner is not satisfied with their current level of knowledge in the learning course and needs to revise their learning path, as shown in Eq. (15).

$$p_{ij}^{S_o}(t)=\begin{cases}\dfrac{\left[\eta_{ij}\right]^{\alpha^*}\times\left[\tau_{ij}(t)\right]^{\beta^*}}{\Sigma\left[\eta_{ij}\right]^{\alpha^*}\times\left[\tau_{ij}(t)\right]^{\beta^*}},L_j|\ d_m^o=C_m^j\\ 0,L_j|\ d_m^o\neq C_m^j\end{cases}\tag{15}$$

In Eq. (15), $d_m^o$ represents the degree of knowledge mastery of learner $S_o$ in the knowledge domain $n$; $C_m^j$ represents the breadth of knowledge of learning unit $L_j$ within the knowledge domain $n$.

## III. RESULTS

To validate the performance of the proposed algorithm model, the dataset source and experimental platform were first introduced. Then relevant performance tests were conducted on the algorithms before and after optimization, as well as other algorithms, in different scenarios. Secondly, different functions were used to compare the proposed algorithm with other algorithms. Finally, the proposed algorithm was compared with other algorithms in terms of practical application effects.

### A. *Performance Testing of Intelligent Learning Model Based on ACO Algorithm*

The experimental platform adopted the hardware configuration of Genuine Intel equipment ® CPU 2140 (dual core) 1.6GHz, 512MB memory. The experiment was conducted on the MATLAB 7.0 platform. The data used in the study was the TSPLIB dataset, and the related problems of this dataset achieved optimal solutions and were widely recognized. It selected relevant questions from this dataset to train and test the performance of the proposed algorithm. Combined with the comparison of three classic models in the past, the superiority of this method could be clarified. Classic algorithms included AS, MMAS, and ASrank. The study adopted the four core issues present in the TSPLIB dataset, as shown in Table I.

To verify the search capability of the proposed algorithm, four out of the seven shutdown key questions were selected for performance testing, and to compare it with three classic algorithms, the optimization results are shown in Fig. 6. From the figure, the proposed AS-N algorithm had better search ability and could search for the shortest path in a shorter time. The four algorithms ASrank, MMAS, AS, and NFACA converged to the optimal solution of problem Ulysses21 at 40, 32, 36, and 27, respectively, and the proposed results performed the best. The proposed NFACA algorithm reduced the optimal solution by 10% compared to the original AS algorithm, and reduced it by 12% and 24% compared to MMAS and ASrank. The performance of the four algorithms on datasets of four different problems is not significantly different, therefore the proposed algorithm has better reliability and convergence speed.

To verify the reliability of the proposed optimization, the actual application effects of the proposed NFACA algorithm were compared and validated with 10 dimensional test functions of ASrank, MMAS, and AS algorithms. The test functions were Sphere, Ellipse, Ackley, and Griebank, respectively. From Fig. 7, the proposed algorithm had the fastest convergence speed when tested in Sphere and Ellipse functions, with an average function value of 10-20, while the average function values of the other three algorithms were around 103. Under the influence of effective and critical subsets, its solution accuracy was higher, and when the adaptive function was estimated around 4000 times, it was already more pronounced on the average of the function than other algorithms. In the Ackley case, all four algorithms achieved good function averages, but NFACA still reached 104 before one-third of the optimization process. During the Griewankf test, only the optimized NFACA algorithm avoided the problems of "premature convergence" and getting stuck in local optima. When the number of adaptive evaluations was between 0-3000, the convergence speed was significantly faster than other algorithms.

TABLE I. FOUR PROBLEMS WITH THE TSPLIB DATASET

| Question title | Problem description | Optimal solution |
|---|---|---|
| Ulysses21 | Odyssey of ulysses | 78 |
| Att44 | 44 cities of the US | 33117 |
| Berlin50 | 50 location in Berlin | 7033 |
| Eril21 | 21 problem of city | 519 |
| KroA100 | 100 cities in Kroatien | 21,282 |
| Lin105 | 105 cities with global optimal | 14,379 |
| Ch130 | 130 cities in Switzerland | 6434 |

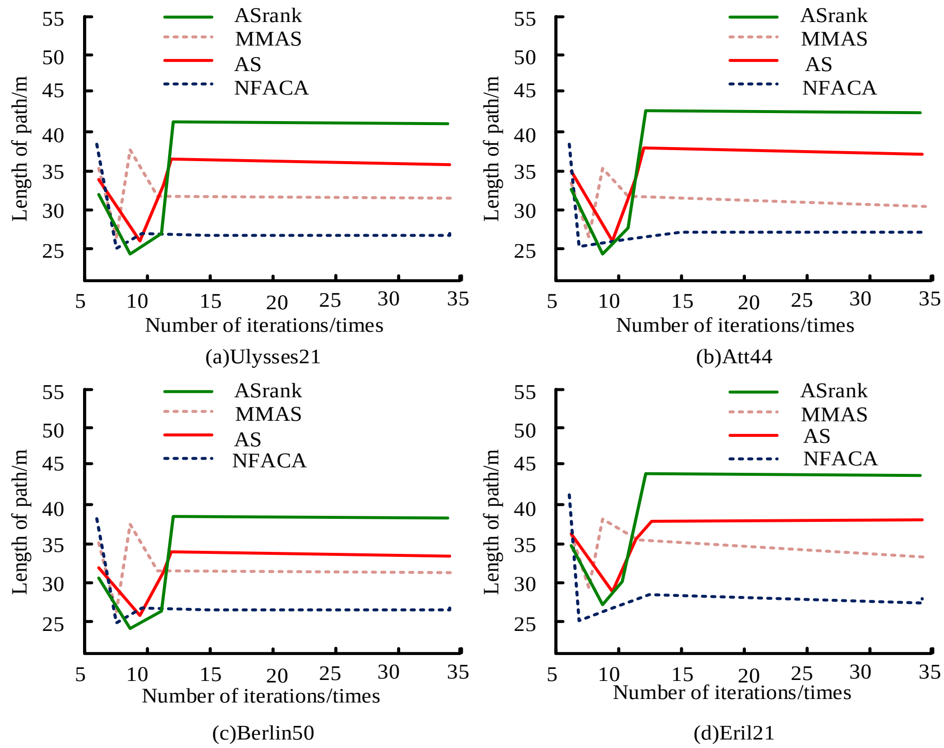


Fig. 6. Comparison of search ability of four algorithms in different problems.

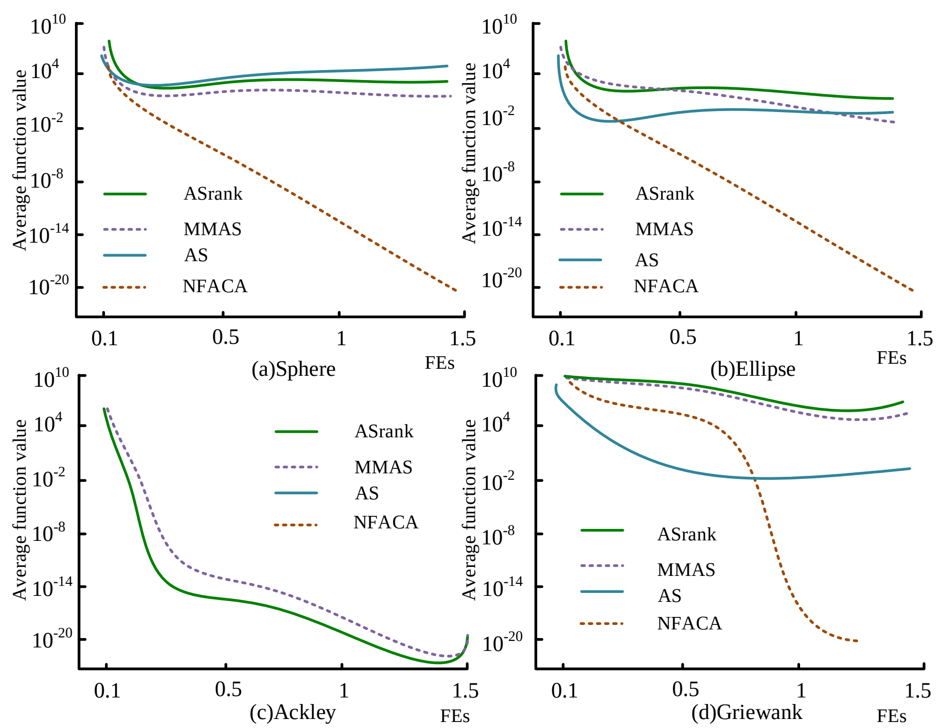


Fig. 7. Comparison of convergence curves of each algorithm in four functions.

## B. *Analysis of the Actual Effect of the Intelligent Learning Model Based on the ACO Algorithm*

The convergence process curve of the proposed model with increasing iteration times was analyzed, the application effect of the model in four different real environments was evaluated, and it was compared with MMAS and AS models. From Fig. 8, all four models began to converge after more than eight searches. The ACO algorithm had a better convergence effect, with the shortest path of 53.5. The AS model had the worst convergence effect, with a difference of 5% compared to the ACO algorithm. The convergence degree of the MMAS model and the proposed ACO algorithm was not significantly different, but they began to converge at the 6th search, showing a phenomenon of “premature convergence”. The proposed algorithm achieved the best convergence effect in all four environments, indicating that it can stably and effectively search for the optimal solution in different complex environments.

To verify the stability of the constructed intelligent learning model in the application of real learning course recommendation scenarios, the stability of the proposed method, ACO and MMS algorithms were compared under different RMSE and RD. The results are shown in Fig. 9. The stability of all three methods decreased with the increase of RMSE, but when RMSE reached 6%, the stability of the proposed method no longer decreased but tended to stabilize. Among them, the ACO algorithm showed the greatest decrease, with a total decrease of 12%. The stability decline of MMS algorithm slowed down in the later stage, with a total decrease of 10%. As the absolute value of the maximum RD increased, the stability of ACO and MMS algorithms has been on a downward trend, decreasing by 12% and 15% respectively. However, the stability of the proposed algorithm was the best, and when RD increased to 8%, the stability did not continue to decrease but increases, with a total decrease of only 5%. The results indicate that the proposed method has higher stability and is more reliable for clustering classification in practical employment recommendation scenarios.

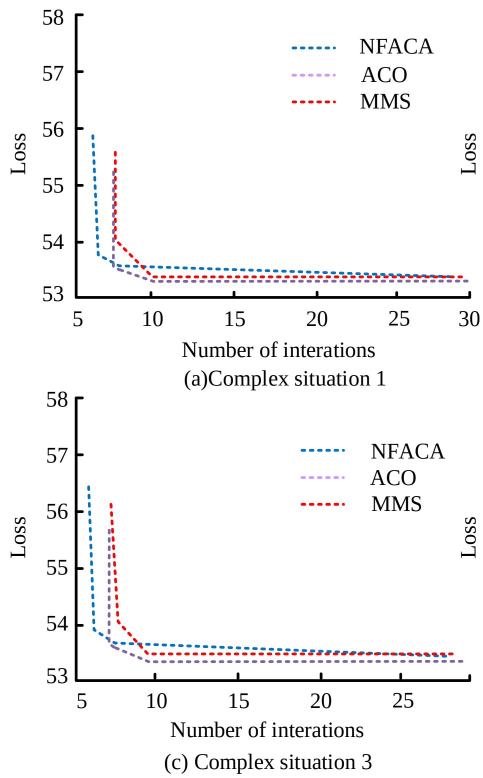


Fig. 8. Comparison of different algorithms for path search.

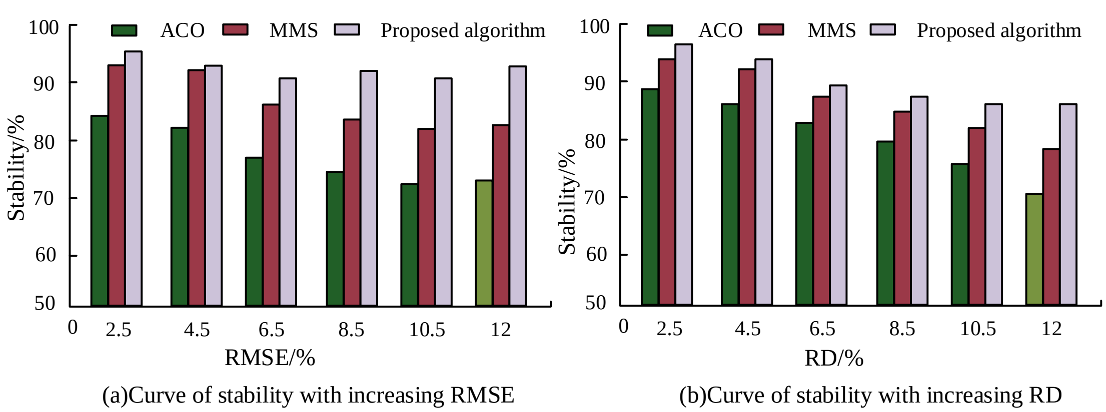


Fig. 9. Changes in stability by method.

To test the accuracy of calculating similarity between students using NFACA model, it was applied to three different university course learning platforms in real scenarios for simulation experiments, and the effects were compared using AS, TaSimRank, and RD3 algorithms. The results are shown in Fig. 10. From Fig. 10, in the datasets of the three scenarios, the accuracy of all four models increased with the increase of k. Among them, the accuracy of the proposed model achieved the best results in three scenarios, with differences within 0.2 in different datasets, and was on average 10%, 11%, and 13% higher than the accuracy of AS, TaSimRank, and RD algorithms, respectively. The RD algorithm had the lowest variation in calculation accuracy among different university datasets, ranging from 0.18 to 0.7. So, the proposed model has higher accuracy and can achieve good repeatability when applied in different course recommendation scenarios.

To test the similarity calculation efficiency of the intelligent learning model proposed by the research in recommending learning units to learners, the ACO and MMS algorithms were compared in three different university learning course recommendation scenarios under two different paths, Naive and Block. The results are shown in Fig. 11. From the figure, the proposed model used the least amount of computation time in both the path Block and path Naive scenarios. Among them, the Naive scenario took the least amount of time, with an average of only 1025s in three different scenarios, which was 16% and 23% lower than the ACO and MMS algorithms, respectively. In the path Block scenario, the proposed model still took the least amount of time, averaging only 1302 seconds, which was 12% and 19% lower than the ACO and MMS algorithms, respectively. From the results, the proposed algorithm performs the best in terms of computation time under different path comparison times. Therefore, this method is applicable in practical environments.

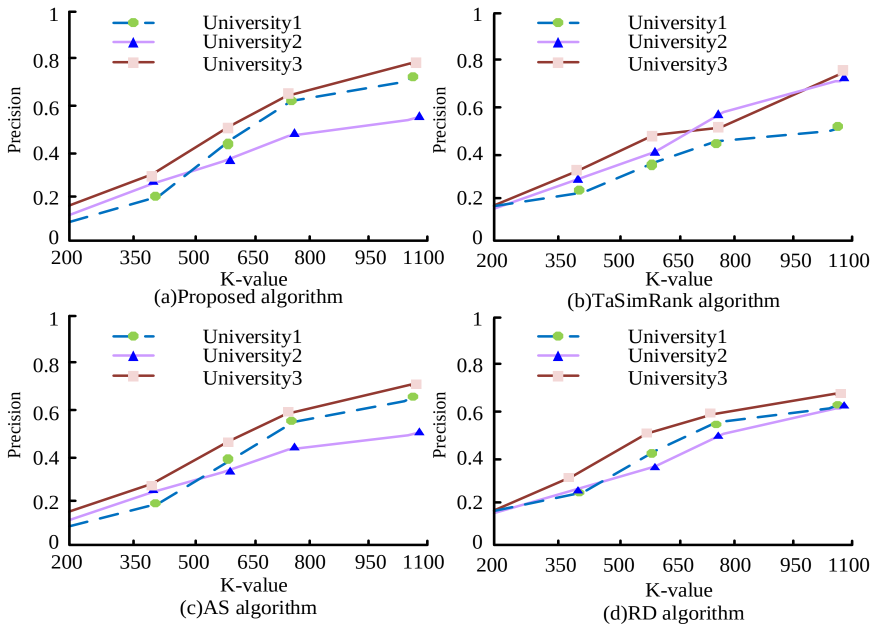


Fig. 10. Comparison of computational accuracy of different algorithms in different scenarios.

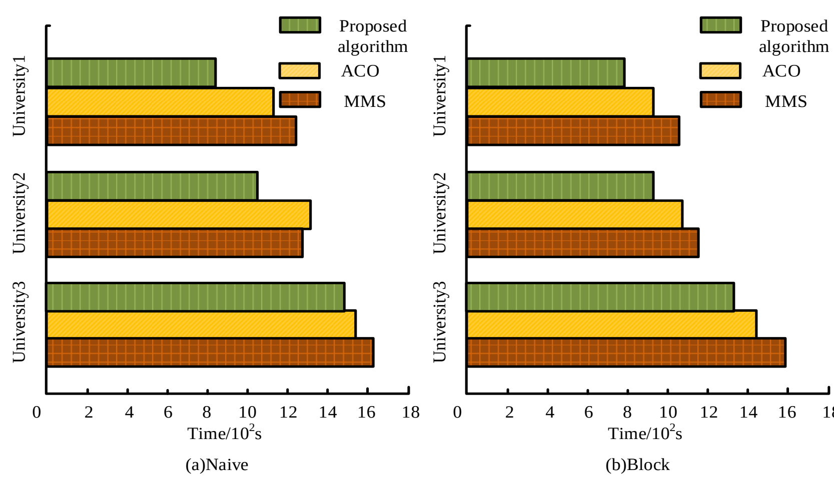


Fig. 11. Comparison of computational accuracy of different algorithms in different scenarios.

## IV. Discussion and Conclusion

The rapid development of online learning has brought many benefits, but it is also accompanied by problems that need to be solved. In the field of course recommendation, although existing intelligent models have shown some effectiveness, they still face limitations in terms of real-time and accuracy. Liu Y and other researchers proposed a pseudo-random algorithm for path planning, which solves the problem of local optima. However, when facing large-scale datasets, there are limitations to their computational efficiency and scalability. Researchers such as Zhang G have developed an ant colony algorithm based on a genetic mutation algorithm to improve the accuracy and applicability of meteorological route planning. However, when directly applied to the field of course recommendation, the complex parameter adjustment and domain specificity issues limit its universality. Mousavinasab et al. proposed a machine learning model that better classifies learners, although it has shown good personalized recommendation performance in physics, chemistry, and clinical fields. However, how to accurately capture learners' interest changes and dynamic needs in course recommendations is still an unsolved problem. Given the limitations of the aforementioned research, an innovative intelligent learning model based on a negative feedback ant colony algorithm is proposed. Compared to traditional methods, the model proposed by the research institute has shown significant advantages in reducing the time cost for learners to choose courses, improving recommendation accuracy, and enhancing user experience.

The results showed that the proposed model achieved a convergence speed accuracy of $10^{-20}$ when tested in Sphere and Ellipse functions, and the optimal solution for problem Ulysses21 was 27, all of which were better than ASrank, MMAS, and AS. The NFACA mechanism had better convergence effect than ASrank, MMAS, and AS algorithms, with a shortest path of 53.5. After reaching RMSE and RD distributions of 6% and 8%, the stability of the proposed method no longer decreased with increasing RMSE. In the scenarios of path Block and path Naive, the proposed algorithm had an average computation time of only 1011, which was better than the ACO and MMS models. In the path Block scenario, the computation time of the three algorithms was slightly reduced compared to the Naive path scenario. This was because the Naive path had more comparisons, which greatly slowed down the algorithm’s running speed and increased computation time. In datasets from three different university scenarios, the accuracy of all four models increased with the increase of k. Among them, the accuracy of the proposed model achieved the best results in three scenarios, and the difference in accuracy between different datasets was within 0.2, which was 10%, 11%, and 13% higher than the average accuracy of AS, TaSimRank, and RD algorithms, respectively. Therefore, the proposed method provides a new solution for path planning in the field of online course recommendation, increasing convergence speed while conducting global search, and improving the accuracy, stability, and applicability of course recommendation platform algorithms. However, there are certain limitations in the methodology, as there are multiple parameters in the negative feedback ant colony algorithm, including pheromone heuristic factor, expected heuristic factor, and pheromone evaporation coefficient. Adapting these parameters to further enhance the algorithm's ability to adaptively adjust the environment can be a future research direction. It can also be combined with a differential particle optimization algorithm to improve the diversity of solutions.

## References

[1] Labrague LJ, De los Santos JA. Transition shock and newly graduated nurses' job outcomes and select patient outcomes: A cross-sectional study. Journal of nursing management, 2020 ,28(5):1070-1079.

[2] Li J, Netessine S. Higher market thickness reduces matching rate in online platforms: Evidence from a quasiexperiment. Management Science, 2020 ,66(1):271-289.

[3] Simone II CB, Bogart JA, Cabrera AR, Daly ME, DeNunzio NJ, Detterbeck F, Faivre-Finn C, Gatschet N, Gore E, Jabbour SK, Kruser TJ. Radiation therapy for small cell lung cancer: an ASTRO clinical practice guideline. Practical radiation oncology, 2020 ,10(3):158-173.

[4] Hebbi C, Mamatha H. Comprehensive Dataset Building and Recognition of Isolated Handwritten Kannada Characters Using Machine Learning Models. Artificial Intelligence and Applications, 2023, 1(3):179-190.

[5] Bandewad G, Datta K P, Gawali B W, & Pawar, S. N.Review on Discrimination of Hazardous Gases by Smart Sensing Technology.Artificial Intelligence and Applications,2023, 1(2): 86-97.

[6] Gao W. New ant colony optimization algorithm for the traveling salesman problem. International Journal of Computational Intelligence Systems, 2020 ,13(1):44-55.

[7] Liu Y, Cao B, Li H. Improving ant colony optimization algorithm with epsilon greedy and Levy flight. Complex & Intelligent Systems,2021 ,7(4):1711-1722.

[8] Stodola P. Hybrid ant colony optimization algorithm applied to the multi-depot vehicle routing problem. Natural computing, 2020 ,19(2):463-475.

[9] Zhang G, Wang H, Zhao W, Guan Z, Li P. Application of improved multi-objective ant colony optimization algorithm in ship weather routing. Journal of Ocean University of China, 2021(6) ,20:45-55.

[10] Li W, Wang GG, Gandomi AH. A survey of learning-based intelligent optimization algorithms. Archives of Computational Methods in Engineering, 2021 ,28(5):3781-3799.

[11] Patibandla RS, Veeranjaneyulu N. A SimRank based ensemble method for resolving challenges of partition clustering methods. Journal of Scientific & Industrial Research,2022 ,79(4):323-327.

[12] Aggarwal K, Mijwil MM, Al-Mistarehi AH, Alomari S, Gök M, Alaabdin AM, Abdulrhman SH. Has the future started? The current growth of artificial intelligence, machine learning, and deep learning. Iraqi Journal for Computer Science and Mathematics,2022 ,3(1):115-123.

[13] Janiesch C, Zschech P, Heinrich K. Machine learning and deep learning. Electronic Markets,2021 ,31(3):685-695.

[14] Yang K, Shi Y, Zhou Y, Yang Z, Fu L, Chen W. Federated machine learning for intelligent IoT via reconfigurable intelligent surface. IEEE network, 2020 ,34(5):16-22.

[15] Mousavinasab E, Zarifsanaiey N, R. Niakan Kalhori S, Rakhshan M, Keikha L, Ghazi Saeedi M. Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods. Interactive Learning Environments,2021 ,29(1):142-163.

[16] Fang X, Heldring A, Rius J M, Chen X, Cao Q. Nested fast adaptive cross approximation algorithm for solving electromagnetic scattering problems. IEEE Transactions on Microwave Theory and Techniques, 2020, 68(12): 4995-5003.

[17] Deng W, Xu J, Song Y, et al. An effective improved co-evolution ant colony optimisation algorithm with multi-strategies and its application. International Journal of Bio-Inspired Computation, 2020, 16(3): 158-170.

[18] Sadiq A T, Raheem F A, Abbas N. Ant colony algorithm improvement for robot arm path planning optimization based on D strategy.

International Journal of Mechanical & Mechatronics Engineering, 2021, 21(1): 196-111.

[19] Kanso B, Kansou A, Yassine A. Open capacitated ARC routing problem by hybridized ant colony algorithm. RAIRO-Operations Research, 2021, 55(2): 639-652.

[20] Wang Z, Ding H, Li B, Bao L, Yang Z, Liu Q. Energy efficient cluster based routing protocol for WSN using firefly algorithm and ant colony optimization. Wireless Personal Communications, 2022, 125(3): 2167-2200.

[21] Lv Z, Chen D, Feng H,Wei W, Lv H. Artificial intelligence in underwater digital twins sensor networks. ACM Transactions on Sensor Networks (TOSN), 2022, 18(3): 1-27.

[22] Liu Y, Cao B. A novel ant colony optimization algorithm with Levy flight. Ieee Access, 2020 ,8:67205-67213.

[23] Too J, Abdullah A R. A new and fast rival genetic algorithm for feature selection. The Journal of Supercomputing, 2021, 77(3): 2844-2874.

[24] Wang B, Liu Z, Li Q, Prorok A. Mobile robot path planning in dynamic environments through globally guided reinforcement learning. IEEE Robotics and Automation Letters, 2020, 5(4): 6932-6939.

[25] Yang Y, Juntao L, Lingling P. Multi-robot path planning based on a deep reinforcement learning DQN algorithm. CAAI Transactions on Intelligence Technology, 2020, 5(3): 177-183.