

## Research Article

# Analysis of College Course Scheduling Problem Based on Ant Colony Algorithm

Ruqun Ge  and Jingyi Chen

HoHai University, Nanjing 211100, China

Correspondence should be addressed to Ruqun Ge; geruqun@163.com

Received 14 June 2022; Revised 2 July 2022; Accepted 7 July 2022; Published 25 August 2022

Academic Editor: D. Plewczynski

Copyright © 2022 Ruqun Ge and Jingyi Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Ant colony algorithm is a new evolutionary algorithm, which is gradually applied due to its easy robustness with other methods and excellent distributed computing mechanism. Currently, the application field of ant colony algorithm has been infiltrated by a single TSP problem. Ant colony algorithm moves the algorithm toward the optimal solution through the combination of positive feedback and negative feedback. This paper briefly analyzes the basic characteristics of the basic idea and principle of the ant colony algorithm to apply it to the ant colony problem. Abstract the course scheduling problem, transform the course arrangement problem into the maximum matching problem of solving the bipartite diagram, and discharge the high-quality curriculum that basically meet the needs.

## 1. Introduction

*1.1. Background.* Class scheduling problem is a typical multi-objective combination optimization problem involving a variety of factors [1–4]. We should not only consider the learning effect of students, but also take care of the reasonable work arrangement of teachers and make full use of teaching equipment [5]. Scheduling the class schedule is a very important and very complicated task for the university academic affairs office. Due to the large amount of courses in each school, manual scheduling will inevitably appear all kinds of conflicts, and its shortcomings are more and more prominent. With the development of computer science and technology, the use of computer scheduling will solve the disadvantages of manual scheduling [1, 6, 7]. It is the computer method of simulating the human brain with the computer, discussing the thinking law and the choice method of arranging the class schedule. Automatic class scheduling has the advantages of short class scheduling time, high quality, and manpower saving [8]. At present, the methods to solve the scheduling problem include genetic algorithm, expert system, binary graph and coloring theory, simulated annealing algorithm, backtracking method, and so on.

However, from the actual use situation, the developed class scheduling software is not generally practical. On the one hand, the class scheduling system is a very complex system, and it is very difficult to solve all the constraints and aspects. On the other hand, because every kind of school or even every school has its own particularity, it is difficult to use the class scheduling software widely. Therefore, the current practice is still to develop a different class scheduling software for different types of schools and then take a separate manual adjustment method for inappropriate scheduling methods.

### 1.2. Analysis of Class Scheduling Problems

*1.2.1. Five Elements of the Lesson Scheduling Problem.* The scheduling process involves the five elements of curriculum, teacher, class, classroom, and time, which are defined as the following set [9, 10]:

- (1) Curriculum collection:  $O = \{O_1, O_2, O_3, \dots, O_L\}$ , where  $L$  is the number of courses offered.
- (2) Teaching teacher collection:  $T = \{T_1, T_2, T_3, \dots, T_K\}$ , where  $K$  is the number of teachers taught throughout the school.

- (3) Class collection:  $L = \{L_1, L_2, L_3, \dots, L_N\}$ , where  $N$  is the number of classes in the whole school.
- (4) Classroom collection:  $R = \{R_1, R_2, R_3, \dots, R_P\}$ , where  $P$  is the total number of different types of classrooms.
- (5) Time collection:  $I = \{I_1, I_2, I_3, \dots, I_Q\}$ , where  $Q$  represents the number of time slices.

1.2.2. *Constraints for Class Scheduling Problems.* The hard and soft constraints are as follows [11]:

Hard constraints

- (1) The classroom can only have one course at the same time.
- (2) The class can only have one course at the same time.
- (3) The teacher can only arrange one course at the same time.
- (4) The teacher can only teach in one class at the same time, except for joint classes.
- (5) The teaching classroom capacity must be greater than or equal to the number of the class.
- (6) The classroom type should match the type of course requirements for the classroom.
- (7) The total number of school classrooms must be greater than or equal to the total number of classes taught at the same time.

Soft constraint

- (1) Classes in the same class should be evenly distributed over a week [12].
- (2) Mathematics courses or professional courses are arranged in the morning, and elective courses are installed in the evening or on weekends.
- (3) In continuous classes or teaching, the distance between classrooms should not be too far.
- (4) Physical education class is generally arranged in the afternoon, and no other classes are arranged after the physical education class.
- (5) The same class is arranged in the same class, at least one day apart, but not more than two days.
- (6) The same class is arranged in the same class, at least one day apart, but not more than two days.
- (7) Teachers' teaching tasks are evenly distributed within a week.
- (8) The same teacher teaches the same course and different classes, and on the same day, each class arranges the teaching content to ensure the continuity of teaching.

## 2. An Overview of the Colony Algorithm

### 2.1. Basic Idea of the Algorithm

2.1.1. *Basic Thought.* When an ant goes to  $m$  city for food, it randomly selects one city as the starting point, and then goes to all other cities one by one. The ant volatilizes pheromones

in the process of foraging, and finally returns to the starting city. The ant can only arrive in each city once [13], What order does the ant enter the city to make the foraging journey shortest?

2.1.2. *Building a Mathematical Model of the Ant Colony Algorithm.* Assuming the scale of the solving problem, where the foraging city number uses  $m$  instead, in the colony, the number of ants was  $n$ . Then get the city set  $V = \{v_1, v_2, \dots, v_m\}$ . Then, the resulting set of paths between the cities is  $E = \{(r,s) | r,s \in V\}$ . At time  $t$ , if  $c_i(t)$  ( $i = 1, \dots, m$ ) is the number of ants located in the city  $i$ , the total number of ants is represented by  $n$ , and then  $n$  is calculated as below:

$$n = \sum_{i=1}^m c_i(t). \quad (1)$$

The concepts of taboo list, pheromone, and visibility emerged in the ant colony algorithm. Taboo list is a data structure that records the city that ants pass through in order to avoid ants entering the same city many times. Tubek is the taboo table of ant  $k$ , ant  $k$  passes through city  $i$  and adds city  $i$  to the taboo table Tubek. Tubek( $q$ ) is the  $q$  element in the taboo table of the ant  $k$ , indicating the  $q$  city through which the ant  $k$  passes. Pheromones are a substance released by ants whose concentrations are volatile over time. Visibility, also known as enlightening information, is defined as the reciprocal of the distance between two cities. The calculation formula is as follows:

$$\sigma_{ij}(t) = \frac{1}{d_{ij}}, \quad (2)$$

$\sigma_{ij}(t)$  is the enlightening information between cities  $i$  and cities  $j$ , and  $d_{ij}$  is the distance between the two cities. The closer the distance between cities, the greater the enlightening information, and the more likely the path is to be selected. Enlightening information is fixed.  $p_{ij}^k(t)$  represents the probability of ant  $k$  moving from city  $i$  to city  $j$  at time  $t$ , and  $p_{ij}^k(t)$  is defined by the following formula:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\sigma_{ij}(t)]^\beta}{\sum_{j \in J_k(i)} [\tau_{is}(t)]^\alpha [\sigma_{is}(t)]^\beta}, & j \in J_k(i), \\ 0, & j \in J_K(i). \end{cases} \quad (3)$$

In the above formula,  $\tau_{ij}(t)$  is the residual pheromone on the two interurban paths ( $i, j$ ) at time  $t$ ,  $\alpha$  is the information heuristic factor, the importance of the residual information, and  $\beta$  is the expected heuristic factor, which is the relative importance of the information.  $J_k(i)$  represents the set of cities to which the next step allows transfer when ant  $k$  is in city  $i$ , defined as the following formula:

$$J_k(i) = V - \text{Tubek}_k. \quad (4)$$

After each ant foraging visits all cities, the pheromone concentration of the interurban path will change. The pheromone on the path should be updated, and the pheromone update is calculated as below:

$$\begin{aligned}
\tau_{ij}(t+1) &= (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t), \\
\Delta\tau_{ij}(t) &= \sum_{k=1}^n \Delta\tau_{ij}^k(t), \\
\Delta\tau_{ij}^k(t) &= \begin{cases} \frac{Q}{L_k}, & \text{Ant } K \text{ passes through the path between the two cities } (i, j), \\ 0, & \text{If not,} \end{cases}
\end{aligned} \tag{5}$$

where  $\Delta\tau_{ij}(t)$  is the added pheromone on the path at time  $t$  ( $i, j$ ) and  $\Delta\tau_{ij}^k(t)$  is the added pheromone of the ant  $k$  at time  $t$  on the path ( $i, j$ ).  $\rho$  is the pheromone evaporation coefficient, and  $1 - \rho$  is the persistence coefficient of the pheromone. The amount of pheromone released by the ants is  $Q$ . The  $L_k$  is the total path length that the ant  $k$  passes through all cities.

## 2.2. Model of Ant Group Algorithm

**2.2.1. Abstraction of the Individual Ants.** Ant colony algorithm is a computer model of real ant foraging behavior, which is a mechanistic behavior. Therefore, the “ants” in the ant colony algorithm are an abstraction of the real ants, and it is impossible and unnecessary to completely reproduce the individual real ants. Ant abstracted out is known as artificial ant. Artificial ant can be regarded as a simple agent, which can complete the construction process of the simple solution to the problem and can talk with other agents through a means.

Artificial ants and real ants have the following points in common:

- (1) There is a mechanism in which individuals communicate in a group [14]: pheromone.
- (2) Complete the same task: to find the shortest path [15].
- (3) Random selection strategies that use current information for path selection: probabilistic selection strategies.

The abstract artificial ants have their own personality.

- (1) Artificial ants are present in a discrete space [16].
- (2) Artificial ants have the memory ability.
- (3) Artificial ants have the corresponding enlightening information.

It is precisely because of these characteristics that artificial ants make ant colony algorithms have higher intelligence and make them more efficient [17].

**2.2.2. Description of the Problem Space.** Although real ants in nature exist in a three-dimensional environment, we can completely abstract the ant foraging route on a plane or curved surface [18]. In other words, such a three-dimensional environment can be regarded as a special two-dimensional plane. In addition, if you want to use the ant colony algorithm on the computer, the abstract artificial ant must move in a discrete two-dimensional plane [19, 20]. Real ants forage on continuous paths, so the ant colony algorithm must abstract such continuity

into discrete, and this abstraction is understandable. If the actual problem can be described by graphs, you can consider whether the problem can be solved with the ant colony algorithm [21].

**2.2.3. Find an Abstraction of the Path.** Artificial ants move between nodes in a discrete two-dimensional plane, so through which path does the current node reach the next node? What to choose? In this case, the size of the pheromone on the path can be abstracted into the weights on the edge of the graph [22]. At each node, the artificial ant selects the next node based on the size of the weights on the path. By using this method, reaching the target node from the initial node obtains a feasible solution to the problem.

**2.2.4. Abstraction of the Pheromone Volatilization.** Ants in nature always leave pheromones continuously along the path, which will gradually evaporate over time. Since the computer handles disengagement events, it can consider the discrete time of the pheromones (i.e., the weight (the pheromone) of the artificial ant) [23].

**2.2.5. Introduction of Enlightening Information.** The above abstraction is just a computerized representation of the real ant colony foraging behavior, reflecting the very high self-organization of ant colony foraging behavior [24]. However, this self-organization takes a lot of time, often this time-consuming is unacceptable, and we must improve the time utilization. The ant colony algorithm is given a guide according to the specific features of the problem space. Introduce heuristic information within the probabilistic strategy that determines the direction of an ant walk, giving all ants a prior knowledge about the length of the path.

## 3. Design of the Ant Colony Algorithm

To solve the scheduling problem, the ant colony algorithm [25–31] needs to establish a special graphical structure. From the analysis of the scheduling problem, the relationship between the five elements of curriculum, teacher, class, classroom, and time in the scheduling process turns into the relationship between OTL “Courses, Teachers, Classes” and RI “Classroom, Time.” Using the binary graph model can solve the maximum matching problem of two mutually disjoint OTL sets and RI sets. The structural course scheduling problem satisfies the hard constraints of the binary graph model, and can fulfill the basic requirements of the course scheduling problem, that is, the class plan is at the

right vertex, the classroom meets the needs of all courses, and does not conflict with the teacher, class, and course time. However, to optimize the course scheduling, we need to use ant colony algorithm for further dynamic adjustment.

**3.1. Bipartite Model.** The bipartite graph [32–35] is that for an undirected graph, the set of vertices can be divided into two disjoint subsets, and the two vertices attached to each edge belong to these two different subsets.

**3.1.1. Set of Vertices of the Model.** On the basis of analyzing the problem of class scheduling, it is found that before class scheduling, each department should provide <course, class and teacher> relationship to the general academic affairs office, and the academic affairs office of the general school should arrange the class <time and classroom>. Here, we reduce the scheduling problem to be searched in the solution space of  $O \times T \times L \times R \times I$  Cartesian product to solve the maximum matching problem formed by <course, class, teacher> relationship and <time, classroom> relationship.

Suppose  $\langle R_{OTL}$  representative for course, class, teacher> relationship,  $\langle R_{RI}$  representative for time, classroom> relationship.

All tuples in the  $R_{OTL}$  relationship are regarded as all vertices on the left of the bipartite graph, actually the constituent vertex set  $G_{OTL}$ , and all tuples in the  $R_{RI}$  relationship are regarded as all vertices on the right of the bipartite graph, actually the constituent vertex set  $G_{RI}$ . So, any two vertices belonging to set  $G_{OTL}$  are independent between them. Similarly, any two vertices belonging to set  $G_{RI}$  are also independent. In addition, the quantitative relationship between sets  $|G_{OTL}|$  and  $|G_{RI}|$  must meet  $|G_{OTL}| \leq |G_{RI}|$ ; otherwise, the scheduling problem must not be solved.  $|G_{OTL}|$  represents the number of courses to be scheduled, and  $|G_{RI}|$  represents the number of time-space resources that can be provided. Not only that, but  $|G_{OTL}| < |G_{RI}|$  should be guaranteed, so that enough <time-space> for students to self-study. Let  $N$  be a set of all vertices in the bipartite graph model of the scheduling problem with the following formula:

$$N = G_{OTL} \cup G_{RI}. \quad (6)$$

**3.1.2. The Set of Edges of the Model.** First, because the curriculum must be arranged in the desired type of classroom, not all vertices in  $G_{OTL}$  are full mapping to all vertices in  $G_{RI}$ , and only classrooms that meet the requirements can be connected to the corresponding curriculum. So, the edges in the bipartite graph are the connections of all nodes in  $G_{OTL}$  and all nodes in which matching the classroom type matches  $G_{RI}$ . Secondly, the classroom capacity allocated during the schedule must meet the number of students in the class, so the second part diagram should not include the classroom capacity that does not meet the class size. Thirdly, the course of a class must be arranged in the campus where the class is located, so the edges of the two diagram must meet the same campus attributes of the left and right vertices. Assuming that  $S$  is a set of edges in the bipartite graph model of the scheduling problem, then  $S = \{(n_1, n_2) | n_1 \in G_{OTL}, n_2 \in G_{RI}\}$ . The type of classrooms required in course  $n_1$  matches the type of classrooms provided by  $n_2$ , the number

of classes taught in course  $n_1$  is <the number of seats provided by  $n_2$ , and the campus attributes of the class in course  $n_1$  match the campus where the classroom provided by  $n_2$  is located.

**3.1.3. The Weights of Each Side of the Bipartite Graph.** The weights on each edge can be constructed according to the expected degree of a specific course on the correlation <time, classroom> in the scheduling question. If in order to ensure the teaching effect, higher mathematics class had better be arranged in 1-2 morning. In this way, the weights (cost  $c_{ij}$ ) of all the higher math classes in the left set  $G_{OTL}$  in the bipartite graph and all the nodes in the 1-2 am set  $G_{RI}$  are less than the weights of the nodes in the other time periods. In this way, these edges are selected to the better solution.

**3.1.4. Bipartite Model.** The bipartite model for the scheduling problem,  $G = \{N, S, C\}$ , is shown in Figure 1.

**3.2. Determine the Bipartite Graph.** The bipartite graph model of the scheduling problem is constructed into a graph as shown in Figure 2.

### 3.3. Ant Group Algorithm Design in Scheduling

**3.3.1. Structure of the Individual Ants.** Every ant must have a memory ability

- (1) Each ant can remember which <time, classroom> nodes in  $G_{RI}$  have completed the schedule and which nodes are still in the waiting stage. The specific method is to assign each ant an RI allocation table, which records which < $R, I$ > nodes the ant passes through during this tour, and updates every node. Also, the < $R, I$ > allocation table needs to be emptied in time before the ant starts its next trip.
- (2) In the process of a weekly tour, the distribution of all courses <time and classroom> can be used to compare with other ant results to select the optimal class scheduling method. The implementation was made by assigning a temporary schedule to each ant. For each course, a match is written into a temporary schedule. When you end the tour, compare the cost of all the ants' temporary schedules and find the one with the least cost. Clear out again as all the ants begin a new round of travel.
- (3) Ability to remember the closeness of course nodes and <time, classroom> nodes. All ants used the same weight information table, used to record this closeness.

Each ant should have a priori ability when choosing a path.

Ants have a full understanding that it is inappropriate to arrange a class in a certain classroom or in a certain period of time. The method is to make the generation value of the edge in the bipartite graph play a full role. Moreover, this generation value directly reflects the expected degree of correlation between the class nodes and <time and classroom> nodes.

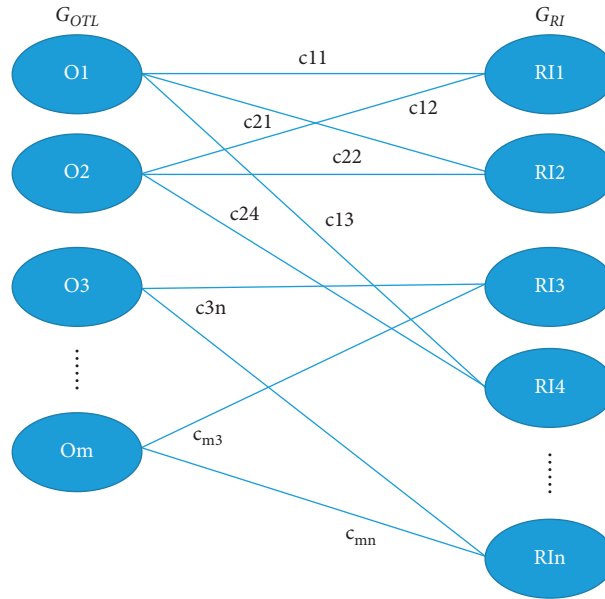


FIGURE 1: Bipartite graph model of lesson scheduling problems.

Each ant communicates with each other by the pheromone left on the path passing through.

With the process of ant optimization, the amount of information on some paths becomes more informative than others. The amount of information can guide the ants to find the optimal solution.

The ants follow a probabilistic selection strategy to perform path selection.

The probabilistic selection strategy also increases the possibility of random selection of paths and avoids the premature ending of algorithms in bad solutions.

**3.3.2. A Weekly Tour by the Ants.** Ants complete a search for all the nodes in the graph, called a weekly walk. The global optimal route is the route that results the best of all the searches you have done. In the classic TSP problem, ants start from the initial city node and pass through each city node in the graph and only once. When all the nodes finish, the ant's one tour ends. The definition of ant travel in the scheduling problem is different in the bipartite model of the scheduling problem, and all ants start from a node in the left  $G_{OTL}$  set of the bipartite graph to find the corresponding matching node in the right  $G_{RI}$  set according to the state transition function. Then, all the ants return to the next node in the  $G_{OTL}$  set to start the next search for matches, logically regarded as a node at the beginning of the next one and the previous one. It was not until the ants matched all the courses in set  $G_{OTL}$  that the weekly tour was over (of course, there must be some knots in  $G_{RI}$  at the end of an ant's tour).

**3.3.3. The Pheromone Strategy—The MMAS Pheromone Strategy Was Used**

- (1) Only the ants with the least cost perform pheromone updates in each iteration, a practice that enables the algorithm faster convergence to the better solution.

- (2) The premature stagnation of the algorithm appears because pheromones are too concentrated on several better paths, while other paths are because no ants pass by for a long time, and pheromones are gradually volatile, resulting in the pheromone concentration on the path to 0. Exploring new paths is almost impossible. To reduce the possibility of stagnation in the early algorithm search, pheromones are restricted to range  $[\tau_{min}, \tau_{max}]$ , which can easily avoid pheromones becoming too large or too small in the course of the algorithm running. Clearly point out  $\tau_{ij} \in [\tau_{min}, \tau_{max}]$ . In the process of ant optimization, if  $\tau_{ij} \geq \tau_{max}$ , then make  $\tau_{ij} = \tau_{max}$ ; if over  $\tau_{ij} \leq \tau_{min}$ , the order  $\tau_{ij} = \tau_{min}$ . This effectively avoids premature stagnation and expands the search space of the algorithm.
- (3) Each path initializes the pheromone concentration to  $\tau_{max}$ , and doing so can increase the search range of the solution in the beginning of the algorithm.
- (4) The simultaneous introduction of a pheromone smoothing processing mechanism, namely,  $\Delta\tau \propto (\tau_{max} - \tau_{ij}(t))$ , the pheromone enhancement obtained by paths with already high pheromone concentration, will facilitate efficient path search for paths smaller than the pheromone concentration.

**3.3.4. Scheduling Steps**

- (1) Refer to the course schedule, divide the five elements of the scheduling problem into two vertex sets OTL and RI, then connect the vertices in the OTL and RI sets that meet the conditions, build the common weight information table of vertices on both sides, and establish a binary graph model.

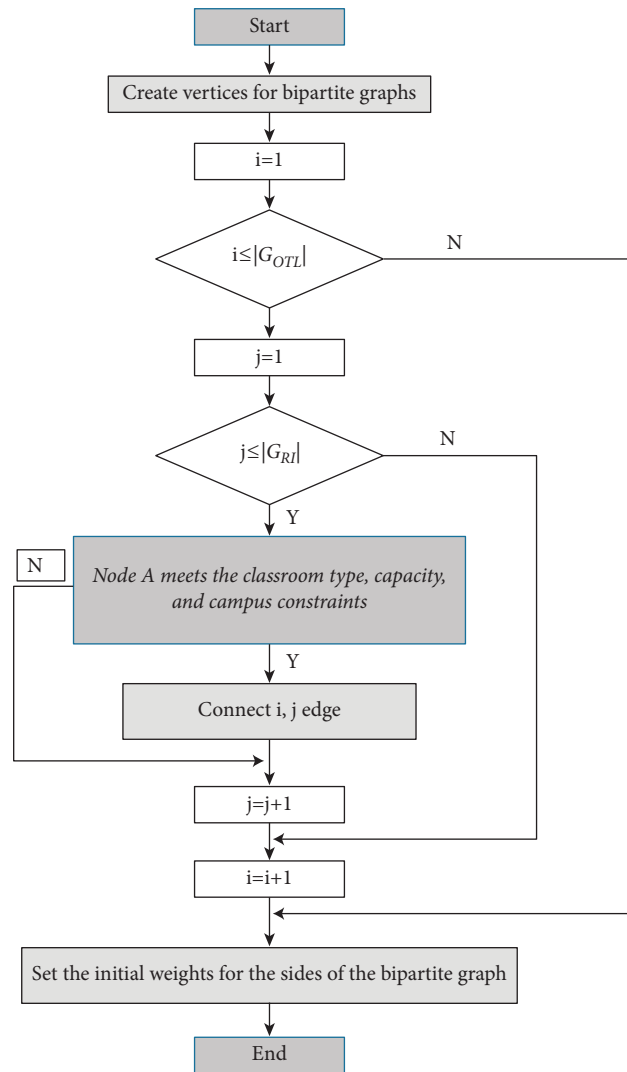


FIGURE 2: The bipartite graph model of the scheduling problem is constructed into a graph.

- (2) The parameters are information heuristic factor, expected heuristic factor, pheromone evaporation coefficient, cycle iteration number, and pheromone concentration recording table in the initialized ant colony algorithm.
- (3) Initialize all teacher time allocation table, class time allocation table, and classroom time allocation table; initialize the temporary class table, individual weights, taboo table, visited vertex set, and classroom and time allocation table in RI.
- (4) Determine whether the iterative optimization is over. If the  $CNT \leq MAXITER$  result is false, jump to step 11.
- (5) All ants will randomly assign a vertex in the OTL set, initializing the individual weights and schedules for each ant.
- (6) Determine whether each ant has visited all vertices in the OTL set; if all vertices are visited, jump to step 10; otherwise, randomly select a vertex that is not visited.
- (7) The transition probability  $A$  was calculated for each ant.
- (8) Determine whether the hard constraint is met. If there is a conflict, jump to step 7 to select the path.
- (9) Record the selected path, modify the ant individual class table, taboo table, visited vertex set, and individual weights, and jump to step 6.
- (10) Update the pheromone concentration value of the binary ad model and jump to step 3.
- (11) Output individual weights with the least cost.

3.4. *Flowchart of Automatic Class Scheduling.* Flowchart of automatic class scheduling under the ant colony algorithm is shown in Figure 3.

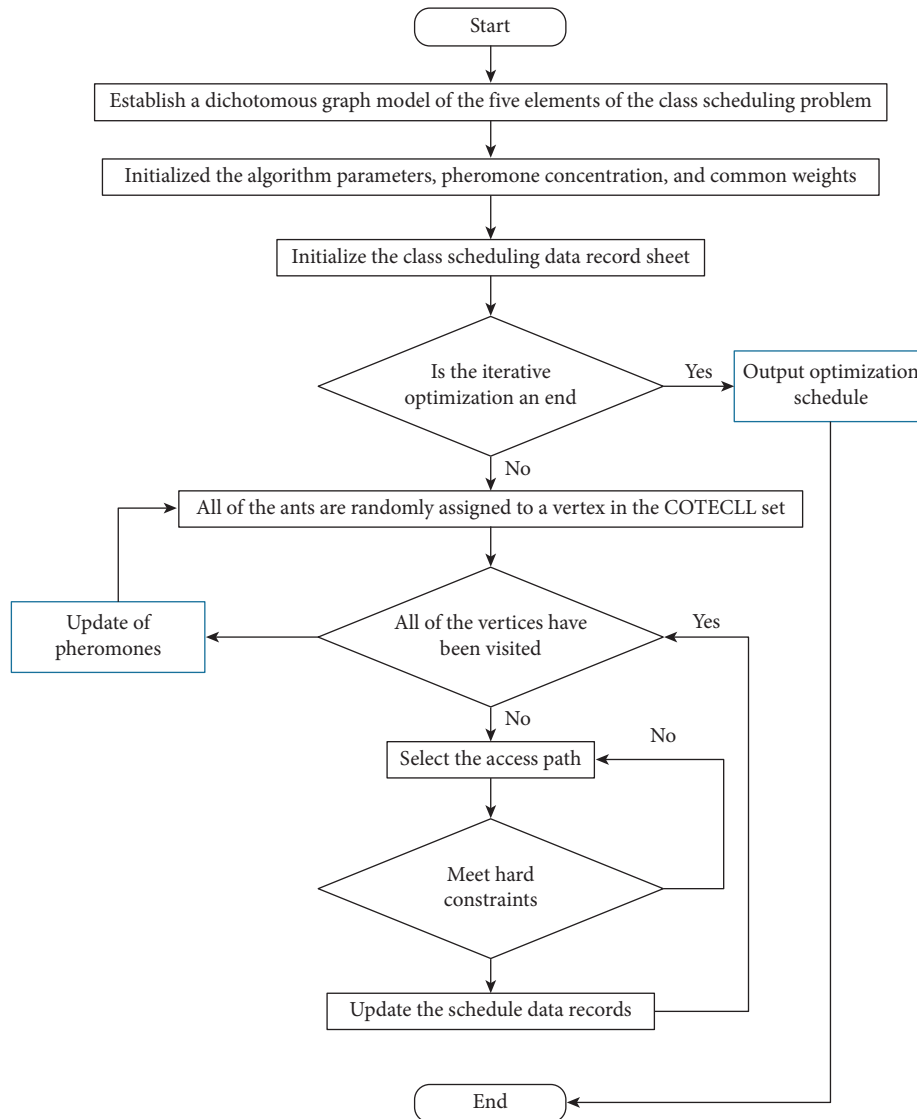


FIGURE 3: Flowchart of automatic class scheduling of ant colony algorithm.

#### 4. Conclusion

With the gradual expansion of college enrollment scale and the more and more popularization of computer in teaching, it has become one of the urgent research topics to be solved in teaching management to replace the manual class scheduling with high labor intensity and low work efficiency in teaching management. According to the characteristics of the scheduling problem, the paper abstracts it into a bipartite graph model and applies the ant colony algorithm to match the bipartite graph to find the better arrangement method of the course. The main achievements are as follows:

- (1) The constraints of the scheduling problem are discussed systematically, and we describe the scheduling problem and its constraints with a mathematical model.
- (2) The scheduling problem is transformed into a bipartite graph model, and the memory ability, ability

to master prior knowledge, adaptation ability, and collaboration ability are applied to move back and forth in the two sets of the bipartite graph to complete the schedule arrangement with higher quality.

- (3) Adaptive selection strategy combining deterministic selection and random selection, dynamically adjusting the probability of deterministic selection and random selection use, maximum and minimalistic pheromone strategy, and pheromone smoothing processing mechanism are applied to the scheduling ant colony algorithm. Experimental results show that this method guarantees the quality of the efficiency and reconciliation of the algorithm.
- (4) When solving the constraints of the scheduling problem, the ants are endowed with a lot of memory ability and necessary prior knowledge mastery ability, effectively satisfying most of the individual

inspired information strategies introduced at the same time, and well solving the problem of a certain time interval in the course arrangement of more than 2 weeks.

The scheduling problem is a complete NP problem, and this paper establishes a bipartite graph model for the scheduling problem. The proposed model is able to greatly reduce the complexity of the algorithm and reduce the solution time from exponential growth to polynomial growth. The experimental results show that the scheduling results using this paper method are ideal, solving all the hard conflicts mentioned here, and also satisfying some soft constraints to the maximum. However, many practical problems are combined classes, special weeks, elective courses, and time constraints on different campuses. Therefore, the algorithm needs to be further improved. I hope my research can provide a new idea for the solution of class scheduling algorithm and contribute to the application of ant colony algorithm.

### Data Availability

The dataset can be accessed from the corresponding author upon request.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### References

- [1] Z. Zheng, W. Zhao, and W. Jian, "Model and solution of college course scheduling problem," *Journal of Shandong University of Science and Technology:Natural Science Edition*, vol. 23, no. 4, p. 3, 2004.
- [2] K. Ren and G. Zhao, "Research on college course scheduling problem based on constraint satisfaction," *Journal of Jiangxi University of Science and Technology*, vol. 27, no. 6, pp. 4-5, 2006.
- [3] J. Li, "An evolution algorithm to solve the problem of college course scheduling," *Microcomputer Development*, vol. 13, no. 10, pp. 3-4, 2003.
- [4] H. Li, G. Hu, and H. Zhu, "Research on college course scheduling based on population advantage genetic algorithm," *Computer Engineering and Application*, vol. 47, no. 10, pp. 233-236, 2011.
- [5] H. Li and H. Zhu, "The decimal immunogenetic algorithm was used to solve the college course scheduling problem," *System Engineering Theory and Practice*, vol. 32, no. 9, pp. 6-7, 2012.
- [6] Xu Qi, *Research on College Course Scheduling Problem Based on Genetic algorithm*, pp. 10-11, Institutes Of Technology Of South China, Guangzhou, China, 2021.
- [7] H. Zhang and S. Zhang, "A modified mixed genetic algorithm was used to solve college course scheduling problems," *Computer Engineering and Application*, vol. 20, 2015.
- [8] K. Ren, G. Zhao, and G. Zhang, "Model and algorithm," *Computer and Modernization*, vol. 000, no. 010, pp. 80-82, 2007.
- [9] H. Zhu and H. Li, "The decimal best individual replacement genetic algorithm is used to solve the college course scheduling problem," *Computer Engineering and Science*, vol. 33, no. 6, pp. 5-7, 2011.
- [10] T. Liu, R. Wu, and Li Yi, "Discussion on genescheduling algorithm," *Journal of Chongqing Electronic Engineering Vocational College*, vol. 27, no. 3, pp. 3-5, 2018.
- [11] Y. Gu and X. Hu, "Application of decimal chaotic genetic algorithm in college course scheduling," *Computer engineering and Science*, vol. 11, pp. 118-120, 2014.
- [12] M. Sun, "Analysis of college course scheduling problems based on genetic algorithm," *Source stream:Education edition*, vol. 10, no. 12, pp. 5-8, 2012.
- [13] Z. Zheng, W. Zhao, and W. Jian, "Model and solution of college course scheduling problem," *Journal of Shandong University of Science and Technology:Natural Science Edition*, vol. 15, no. 4, pp. 115-117, 2004.
- [14] C. Hui, J. He, and Chen, "Mathematical model study of class scheduling problems," *Journal of Shenyang Institute of Engineering(Natural Science Edition)*, vol. 01, no. 05, pp. 101-102, 2017.
- [15] X. Zhang, H. Zhu, and H. Wu, "Research on college scheduling based on 3 D immunogenetic algorithm," *Computer Engineering and Application*, vol. 48, no. 5, pp. 217-220, 2012.
- [16] H. Zhang and S. Zhang, "A modified mixed genetic algorithm was used to solve college course scheduling problems," *Computer Engineering and Application*, vol. 51, no. 005, pp. 240-246, 2015.
- [17] X. Li, Luffy, and C. Tian, "Application of artificial fish algorithm for combinatorial optimization problems," *Journal of Shandong University Engineering edition*, vol. 34, no. 5, pp. 64-67, 2004.
- [18] Y. J. Noble, "Group intelligent algorithm and its application beijing China water resources and," *Hydropower Press*, vol. 13, no. 2, pp. 110-112, 2006.
- [19] Y. Lv, "Zhao guangzhong ant group optimization algorithm and its application in electric power system," *Electrical Technology Journal*, vol. 18, no. 4, pp. 71-74, 2003.
- [20] F. Wu, "Application of ant group Algorithm," *Journal of Kaifeng University*, vol. 24, no. 01, pp. 88-90, 2010.
- [21] C. Tang, "Model design of class scheduling system based on ant colony algorithm," *Journal of Anshan Normal University*, vol. 20, no. 04, pp. 39-45, 2018.
- [22] *The Algorithm Principle and Application of Section*, Science Press, Beijing, China, 2005.
- [23] Yi Sun and H. Hu, "Lesson scheduling system based on the genetic-ant colony hybrid algorithm," *Computer System Application*, vol. 28, no. 02, pp. 81-86, 2019.
- [24] Thorns-Stitzle and H. Hoos, "Max-min ant system and local search for combinatorial optimization problem," in *Proc.of 2d Int.Conf.on Metaheuristics.teen*, pp. 11-12, no. 12, Springer-Verlag, New York, NY, USA, 1997.
- [25] L. Li, C. Mao, H. Sun, Y. Yuan, and B. Lei, "Digital twin driven green performance evaluation methodology of intelligent manufacturing: hybrid model based on fuzzy rough-sets AHP, multistage weight synthesis, and PROMETHEE II," *Complexity*, vol. 2020, no. 6, Article ID 3853925, 24 pages, 2020.
- [26] Q. Zhou, "Research on optimization algorithm of cloud computing resource allocation for internet of things engineering based on improved ant colony algorithm," *Mathematical Problems in Engineering*, vol. 2022, Article ID 5632117, 6 pages, 2022.
- [27] L. Li and C. Mao, "Big data supported PSS evaluation decision in service-oriented manufacturing," *IEEE Access*, vol. 8, no. 99, Article ID 154663, 2020.



- [28] W. Yue and P. Dai, "Research on urban national sports fitness demand prediction method based on ant colony algorithm," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 5872643, 9 pages, 2022.
- [29] L. Li, T. Qu, Y. Liu et al., "Sustainability assessment of intelligent manufacturing supported by digital twin," *IEEE Access*, vol. 8, Article ID 174988, 2020.
- [30] S. Liao, "An ant colony algorithm model construction on the impact of urban real estate value and national economic changes," *Scientific Programming*, vol. 2022, Article ID 9616177, 7 pages, 2022.
- [31] F. Wu, "Contactless distribution path optimization based on improved ant colony algorithm," *Mathematical Problems in Engineering*, vol. 2021, Article ID 5517778, 11 pages, 2021.
- [32] T.-H. Jeng, Yi-M. Chen, C.-C. Chen, C. Huang, and P. Miner, "Interaction profiling bipartite graph mining for malware-control domain detection," *Security and Communication Networks*, vol. 2020, Article ID 8841544, 20 pages, 2020.
- [33] L. Li, B. Lei, and C. Mao, "Digital twin in smart manufacturing," *Journal of Industrial Information Integration*, vol. 26, no. 9, Article ID 100289, January 2022.
- [34] H. Feng, Z. Yu, J. Guan, and G. Lin, "A hybrid spectrum combinational auction mechanism based on a weighted bipartite graph for energy internet in smart cities," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8829602, 13 pages, 2020.
- [35] J. Hu, J. Liang, and S. Dong, "A bipartite graph propagation approach for mobile advertising fraud detection," *Mobile Information Systems*, vol. 2017, Article ID 6412521, 2017.