Dynamic PDCA-ACO Model for Adaptive Course Scheduling in Educational Institutions

Wenyong Guo*, Ge Song
College of Foreign Languages, Hebei North University, Zhangjiakou 075000, China
E-mail: Wenyongguoo@outlook.com
*Corresponding author

Keywords: ant colony optimization algorithm, PDCA cycle, course scheduling system, dynamic adjustment, resource optimization

Received: July 24, 2025

Academic scheduling is the core work of teaching management in colleges and universities, involving the optimization of complex spatio-temporal constraints on multi-dimensional resources such as courses, teachers, classrooms, and classes. Traditional class scheduling methods face challenges such as low efficiency of large-scale combination optimization and difficulty in coping with dynamic adjustment needs, which restrict the quality and management efficiency of class scheduling. In order to solve this problem, this study combines bioheuristic algorithms and dynamic process management ideas to propose an ACO-driven PDCA dynamic adjustment model for circular academic scheduling: ACO is improved by contrast enhancement, information entropy control and random perturbation, and PDCA is implemented in stages - Plan clearly constrains and initializes ACO parameters (α∈[1.2,1.8], etc.), does deploy the scheme and responds to interference, checks for monitoring anomalies, and Act activates ACO to locally optimize to form a closed-loop optimization. In order to verify the effectiveness, an experiment was designed based on real data from universities (80 teachers, 60 classrooms) to compare traditional GA and static ACO. The results showed that the success rate of the initial class scheduling was 96.5% (90.1% super-contrast mean, and the test difference was significant). In the 6 dynamic adjustments, the conflict resolution time was reduced by 42% (to 49 minutes), the conflict course was reduced by 63%, the classroom utilization rate increased by 5.8%, and the teacher satisfaction increased by 11.3%, and the stability of the algorithm was 2-4 times that of the comparison. The model highlights the advantages of efficient handling of constraints and rapid response adjustment, providing a feasible path for intelligent educational management.

Povzetek: Predlagani ACO+PDCA model bistveno izboljša urnike in hitreje rešuje konflikte kot GA ali statični ACO.

1 Introduction

As the core link in teaching operations management in higher education institutions, the essence of educational administration arrangement is a highly complex, multi-constraint, and multi-objective combinatorial optimization problem [1, 2]. This problem involves the accurate matching and spatiotemporal coordination of multi-dimensional resources such as curriculum, teacher, classroom, class, and time. It needs to meet a series of rigid constraints and flexible goals [3]. With the continuous expansion of teaching scale in colleges and universities, the increasingly refined curriculum system, and the continuous improvement of flexibility and response speed requirements of teaching management, the traditional static course scheduling method based on manual experience or rule engine has been difficult to meet the dynamic needs of modern educational administration [4]. These methods generally have inherent defects, such as low solution efficiency, limited optimization ability, difficulty in coping with the middle change of teaching plan, temporary adjustment of resources, or emergency interference, which leads to lengthy course scheduling cycles, frequent conflicts, low resource utilization rate, and ultimately affects the stability of teaching order and management efficiency [5, 6]. Therefore, developing an intelligent course scheduling model that can efficiently handle complex constraints, respond quickly to dynamic changes, and continuously optimize course scheduling results has become a key problem that needs to be addressed urgently to enhance the modernization level of educational administration in colleges and universities.

In recent years, intelligent optimization algorithms have shown significant advantages in solving complex combinatorial optimization problems. The ant colony optimization algorithm has been successfully applied in vehicle routing planning, task scheduling, and other fields because of its natural distributed computing, positive feedback mechanism, and strong robustness, which provides new ideas for solving large-scale course scheduling problems [7, 8]. However, directly applying ACO to course scheduling, especially in scenarios that require frequent dynamic adjustments, still faces challenges: First, when the standard ACO algorithm deals with the strong constraints unique to course scheduling problems, its convergence speed and global optimization ability may be limited; Secondly, the algorithm is usually designed to solve static problems at one time, lacking the perception and adaptive adjustment mechanism of environmental changes during operation, resulting in insufficient dynamic adaptability [9, 10]. At the same time, the PDCA cycle (Plan-Do-Check-Act), widely used in process management, is a quality management framework that emphasizes continuous improvement. Its core ideas of closed-loop feedback and iterative optimization provide methodological guidance for solving optimization problems in dynamic environments [11]. Through repeated iterations of planning, execution, inspection, and processing, the PDCA cycle can effectively capture deviations, drive improvements, and ensure that the system maintains performance in the face of changes, thereby optimizing [12, 13]. Therefore, it is of great theoretical significance and practical value to explore the deep integration of the intelligent optimization ability of ACO (Ant Colony Optimization Algorithm) and the dynamic process management mechanism of the PDCA cycle, thereby building a course scheduling model with self-learning, dynamic response, and continuous optimization capabilities.

This study aims to break through the limitations of the static scheduling model and propose and conduct an in-depth study of the "PDCA Academic Scheduling Dynamic Adjustment Model Driven by Ant Colony Optimization Algorithm (ACO)". The core of the model is to build a collaborative optimization mechanism with PDCA cycle as the management framework and improved ACO as the intelligent engine: the "Plan" stage clarifies the multi-dimensional goals and hard and soft constraints of class scheduling, initializes and improves ACO parameters, and guides the "artificial ant colony" to explore high-quality class scheduling schemes through efficient pheromone update strategies and heuristic rules. The "Do" stage executes the initial lesson scheduling plan generated by ACO optimization and puts it into actual teaching operation. The "Check" stage continuously monitors the teaching status, and uses the rule engine or data interface to capture the demand signals for class adjustment in real time, such as course increase or decrease, teacher availability changes, temporary unavailability of classrooms, and time conflicts. After detecting the effective adjustment requirements in the "Act" stage, the improved ACO does not re-solve the global solution, but uses the pheromone distribution accumulated in the previous round of optimization and the current update constraints to quickly orient local or global optimization within a strict time window to generate a high-quality adjustment scheme that meets the new constraints.

The potential value of this model is to improve the response speed and decision-making quality of academic affairs in response to uncertainty and sudden changes, reduce the intensity of manual intervention and management costs, ensure the optimal allocation of teaching resources and the stability of teaching order, and provide technical support for the construction of an intelligent and flexible modern academic management system.

In the current research, there are three key gaps: first, most of the existing ACO scheduling research is static optimization, and the directional optimization mechanism is not designed for dynamic interference such as teacher adjustment and classroom occupancy, which is easy to lead to low efficiency and unstable scheme in global recalculation; second, PDCA and ACO lack deep coupling, and the ACO pheromone update and historical experience are not linked at each stage of PDCA, and closed-loop optimization is a formality. Third, the dynamic scheduling performance evaluation dimension is single, ignoring effective indicators such as resource utilization and teacher satisfaction, and lacking system comparison with traditional schemes in multiple scenarios. Accordingly, this study is reconstructed into three specific research questions: first, how to improve the ACO mechanism to achieve efficient local directional adjustment under dynamic interference and avoid global recalculation loss; second, how to build a PDCA-ACO in-depth collaboration framework, so that each stage of PDCA can be linked with ACO optimization and experience reuse to form a closed-loop intelligent optimization; Third, whether the model is significantly better than the traditional scheme in multiple dimensions such as conflict resolution, resource utilization, and teacher satisfaction in multiple rounds of dynamic scenarios, and has long-term robustness.

The core novelty of this study aims at the specific background of "resource constraint coupling, multiple dynamic interferences, and the need to take into account efficiency and solution stability" in the academic scheduling of colleges and universities, and will improve the customized and deep integration of ACO and PDCA: improve the local optimization and pheromone rules of ACO, accurately link with each stage of PDCA, and at the same time use ACO intelligent optimization to give PDCA closed-loop effectiveness, solve the pain points of a single ACO that it is difficult to cope with dynamic interference and a single PDCA lacks optimization ability, and adapt to the needs of academic management in colleges and universities.

2 Theoretical basis and principle technology

2.1 Theory of ant colony optimization algorithm

The ant colony algorithm determines the shortest path by simulating the behavior of ants as they search for food [14, 15]. Although a single ant cannot be visually identified, the whole colony can determine the shortest path from the ant nest to the food source through cooperation, demonstrating excellent swarm intelligence [16].

Ants release pheromones when looking for food, chemicals that direct them to the shortest path [17, 18]. As a marker, pheromone forms a positive feedback mechanism, making ants more inclined to choose the path with the highest pheromone concentration. As ants keep walking, the pheromone concentration increases, and eventually, all ants will take this shortest path [19].

The ant colony algorithm is used to find the shortest path in the graph, which is determined by iteration. Each iteration round includes two steps of ant tracking and pheromone update [20]. In the ant tracking stage, ants look for target nodes; in the Pheromone renewal phase, ants release pheromones along the path. As the number of iterations increases, the pheromone concentration on the shortest path increases, attracting more ants, and the algorithm eventually converges to this path [21]. In the initial stage, it is necessary to set the number of ants, the number of iterations, and the pheromone concentration of the initialization path. The architecture of the ant colony optimization algorithm is shown in Figure 1.

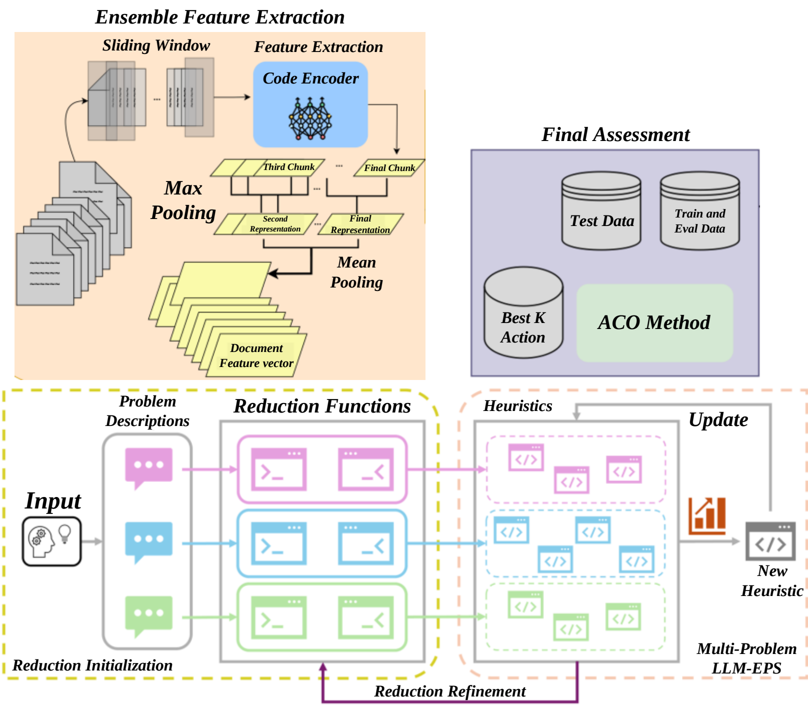

Figure 1: Ant colony optimization algorithm architecture

When choosing a path, ants calculate the probability of different routes and use the roulette principle to decide the route [22]. They will record the path taken, avoid repetition, and use heuristic information to choose shorter paths. The formula (1) for calculating the state transition probability p of ants is as follows:

$$p_{ij}^k(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^\alpha[\eta_{ij}(t)]^\beta}{\sum_{s\in allowed_k}[\tau_{is}(t)]^\alpha[\eta_{is}(t)]^\beta} & j\in allowed_k\\ 0 & else\end{cases} \quad (1)$$

The formulation exhibits the probability of ant choosing a path, depending on the pheromone concentration τij and the reciprocal of the distances i and j between nodes. α and β are pheromones and distance weights. allowedk denotes the set of ant alternative paths. After the traversal is completed, the path pheromone is updated to prevent premature convergence, and the rule added after dilution is adopted. The update rule is shown in the following formula (2):

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

In the formula, ρ is the pheromone dilution coefficient in the range of (0, 1), △τkij denotes the pheromone increment. The pheromone increase of ant k from node i to j is calculated according to formula (3):

$$\Delta\tau_{ij}^k=\begin{cases}\dfrac{Q}{L_k}, if\ k(i,j)\\ 0, else\end{cases} \quad (3)$$

In the Ant Colony System (ACS), Lk represents the length of the path traveled by the ant k, and Q is a fixed numerical value of the pheromone concentration. A short path leads to a high pheromone concentration. The system realizes state transition through pseudo-random proportional rules to improve the optimization performance of the algorithm. The transition rules are shown in the following formula (4):

$$j=\begin{cases}arg\ \max\limits_{u\in allowed_k}\left\{[\tau_{iu}(t)]^\alpha\,[\eta_{iu}(t)]^\beta\right\}, if \quad q<q_0\\ p_{ij}^k(t), else\end{cases} \quad (4)$$

In the formula, q0 is a fixed value between 0 and 1, and q is a random number from 0 to 1. When q is less than q0, ants choose paths according to pheromone density; If q is greater than or equal to q0, the roulette method is used to determine the path.

Systemic pheromone update mechanism can be divided into two ways: comprehensive and partial. At the end of the iteration, the full update only adds the pheromone on the optimal path, while the partial update updates the pheromone after each ant moves one step [23, 24]. These rules enhance the pheromone concentration of the best path, guide ants to choose the path more, and improve the search efficiency of the algorithm.

2.2 PDCA cycle theory

The PDCA cycle management system, which includes four links: planning, execution, inspection, and action, originated from the PDS theory proposed by American professor Shewhart in the early 20th century [25]. The PDCA theory was introduced in China and initially used only for total quality management; however, it has since been widely adopted in various industries [26]. As a globally recognized management model, the PDCA cycle system embodies the mechanism of standardized operation and continuous improvement.

The PDCA cycle model comprises four phases and eight main workflows that encompass planning, execution, inspection, and action. Analyze the problem and identify the root cause. Set goals, plan, and evaluate programs [27, 28]. The main factors were determined using tools such as a causal diagram, SWOT analysis, and 4MIE, according to the Pareto principle and the data. Develop a solution strategy and implementation plan [29]. Develop practical and effective execution plans to ensure operational efficiency. Allocate resources reasonably and strictly implement the action plan.

Verify the experiment's effectiveness and evaluate the implementation and results of the plan. Compare goals and achievements, formulate solutions to problems, and sum up experience. Consolidate achievements and actively solve outstanding issues. Standardized management can effectively integrate enterprise resources, standardize the production process, and facilitate resource allocation and material circulation. A standardized environment is the foundation for enhancing production efficiency and product quality, and it is also the cornerstone of balanced production control [30, 31]. The PDCA cycle management system cannot solve all problems simultaneously, and unsolved problems will be transferred to the next cycle [32].

3 Construction of dynamic adjustment model of PDCA educational administration course scheduling based on ant colony optimization

3.1 Overall frame design of model

Ant colony algorithm is a probabilistic global search technique, which aims to improve the probability of finding the global optimal solution. It is based on two path selection rules: random selection and choice combining determinism and randomness [33, 34]. During the execution, random selection is usually made using the roulette method, but when dealing with large-scale problems or a large number of city visits, the roulette method may fail and fail to make effective use of the feedback information [35]. To this end, this study proposes a dynamic adjustment selection strategy and introduces a contrast enhancement technique. Figure 2 shows the dynamic adjustment model architecture of PDCA educational administration course scheduling driven by ant colony optimization algorithm.

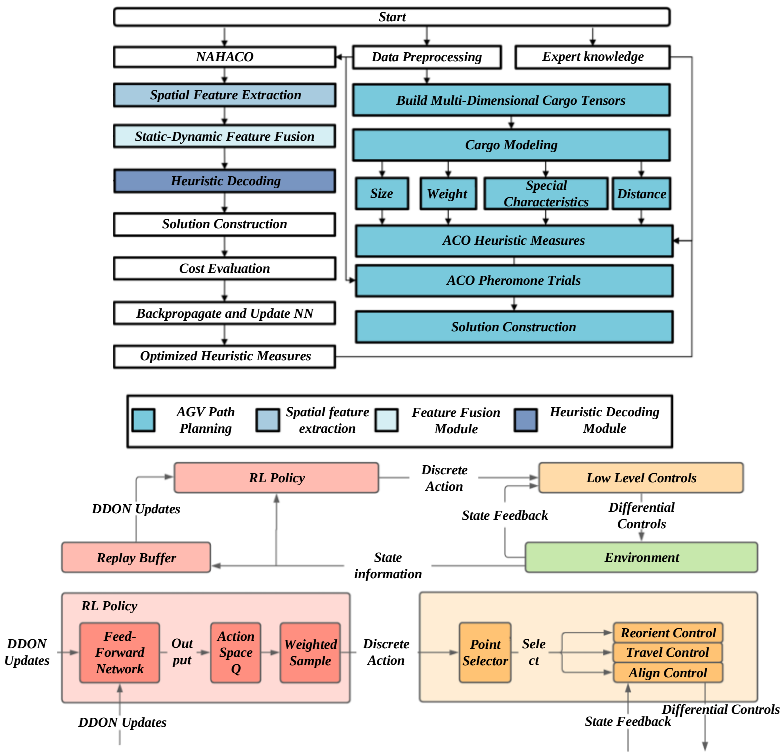

Figure: 2 PDCA educational administration course scheduling dynamic adjustment model architecture driven by ant colony optimization algorithm

Ants prefer places with high pheromone concentrations when choosing cities. Due to the average chance of cities being selected, the difference in pheromone concentration is small, and the path selection probability is similar [36]. By simulating contrast enhancement techniques, these probabilities can be adjusted to make high probabilities more significant and low probabilities smaller. The operation steps are as follows:

Step 1: Use formula (5) to calculate the probability of accessible cities, and set the quantity to n, which is expressed as pi=[pi1,Pi2,…,Pm]. Select Pmax=max(pij), where $1 \le j \le n$, set pm as the threshold, C as the threshold parameter and C is within (0, 1), and take pm = C×Pmax. In the ant colony algorithm, the state transition probability pkij(t) denotes the probability that ant k will go from city i to city j at time t.

$$P_{ij}^{k}(t)=\begin{cases} arg\,max\left\{\left|\tau_{ij}(t)\right|^{\alpha}\cdot\left|\eta_{ij}(t)\right|^{\beta}\right\}, q\leqq_{0}, j\in allowed_{k}; \\ \dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\cdot\left[\eta_{ij}(t)\right]^{\beta}}{\sum\limits_{k\in allowed_{k}}\left[\tau_{ik}(t)\right]^{\alpha}\cdot\left[\eta_{ik}\right]^{\beta}}, j\in allowed, other \end{cases} \quad (5)$$

Step 2: According to equation (6), amplify the value of pi higher than pm and reduce the value lower than pm to obtain ui = [ui1, ui2..., ui3];

$$u_{ij}=\begin{cases} \dfrac{1}{p_m}(p_{ij}-p_m)^2, 0\le p_{ij}\le p_m \\ \dfrac{1}{p_m}(p_{ij}-p_m)^{\frac{1}{2}}, p_m\le p_{ij}\le 1 \end{cases} \quad (6)$$

Apply formula (7) to normalize ui to obtain the path selection probability pi.

$$p_{ij}^{'}=\frac{u_{ij}}{\sum\limits_{1\le s\le n}u_{is}} \quad (7)$$

After implementing these three steps, the choice of high-probability cities is more prominent, and the influence of low-probability cities is reduced. This increases the odds of cities with shorter paths being selected, speeding up the algorithm to find the best solution.

Applying contrast enhancement technology, ants reduce their dependence on low-probability cities when selecting cities, increase their tendency to high-probability cities, and accelerate algorithm convergence. But this may fall into the local optimum prematurely. In order to avoid iterative stagnation, this paper uses information entropy processing.

Entropy is defined as E (p) =-(p×$\log_2$P), where P is the city node and p is the probability of ant choosing a city. When p_1 is equal to p, the entropy reaches the maximum value $\log_2(n)$, and its value range is [0, $\log_2(n)$]. n represents the number of cities that ants can visit, set Hm = $\log_2$ (n).

The path selection is described by the concept of information entropy, and the probability of wheel selection is dynamically adjusted according to the entropy value, and the contrast enhancement direction is controlled. Core concept: when the entropy value is high, the convergence is slow, there is no local optimal, and the contrast enhancement is performed; At low entropy, the convergence is fast, the algorithm may stagnate, and the execution contrast is weakened. If Hs ≥ h × Hmax, the enhancement treatment is adopted, and the formula (8) is:

$$u_{ij}=\begin{cases} \dfrac{1}{p_m}(p_{ij}-p_m)^2, 0\le p_{ij}\le p_m \\ \dfrac{1}{p_m}(p_{ij}-p_m)^{\frac{1}{2}}, p_m\le p_{ij}\le 1 \end{cases} \quad (8)$$

If Hs≤(1-h)×Hmax, the attenuation treatment is adopted, and the formula (9) becomes:

$$u_{ij}=\begin{cases} \dfrac{1}{p_m}(p_{ij}-p_m)^2, p_m\le p_{ij}\le 1 \\ \dfrac{1}{p_m}(p_{ij}-p_m)^{\frac{1}{2}}, 0\le p_{ij}\le p_m \end{cases} \quad (9)$$

The value of the control parameter h ranges from 0.5 to 1. Ant colony optimization algorithm faces the problems of slow search speed and stagnation. Introducing random perturbation strategy can solve the stagnation problem and enhance the search ability. The algorithm combines deterministic and random selection. Deterministic selection makes ants prefer high-probability paths, and random selection increases randomness. The two together improve the global search ability of the algorithm.

This study effectively deals with the algorithm stagnation problem. It is proposed to introduce random disturbance after a certain stage of the algorithm progress to avoid premature stagnation of the algorithm. The transfer coefficient design of the random disturbance feature is shown in Equation (10).

$$C_{ij(k)}(t)=[\tau_{ij}(t)]^{\gamma}\eta_{ij}, U\ge p_m \quad (10)$$

In the formula, γ represents $e^{b/k}$ that the value of k is between 1 and M, and b is a scale factor greater than 0. M is the maximum number of iterations and b is the scale factor. pm is the random variation rate and U is the uniform random number in the (0, 1) interval. Cij(k)(t) is the transfer coefficient of the path (i, j), and ants tend to choose the path with the highest transfer coefficient, showing certainty.

The pheromone influence factor was taken α [1.2, 1.8], and the distance weight was β taken [0.8, 1.4]. The initial configuration is set as pheromone concentration τ_0=0.05, ant number m=2×N, volatilization coefficient ρ=0.12; In mathematical modeling, the state transition probability $P_{ij}^{kt}(t)=[\tau_{ij}^{kt}(t)]^{\alpha}\cdot[\eta_{ij}^{kt}]^{\beta}/\sum[\tau_{ij}^{kt}(t)]^{\alpha}\cdot[\eta_{ij}^{kt}]^{\beta}$ (η_{ij}^{kt} is a heuristic function, defined as 1/(teacher conflict coefficient, classroom occupancy coefficient, time overlap coefficient)), and the pheromone update formula $\tau_{ij}^{kt}(t+1)=(1-\rho)\tau_{ij}^{kt}(t)+\sum\Delta\tau_{ij}^{kt}$ ($\Delta\tau_{ij}^{kt}$ is the pheromone increment of the kth ant in the path (i,j), and the optimal solution path increment is 2×Q/m, Q is a constant), and the academic constraints are added: $\sum x_{ij}^{kt}\le 1$、$\sum x_{ij}^{kt}\le 1$ to realize the adaptation of the algorithm to PDCA scheduling requirements.

3.2 Design of dynamic adjustment mechanism based on PDCA cycle

At the level of resource constraints, it breaks through the conventional single matching of "classroom capacity-course size" and presents the multi-dimensional coupling and dynamic scarcity characteristics of "hardware function - teaching demand - time scarcity." The time constraint breaks the two-dimensional matching of "teacher-course", and there is a multi-level dependence of "course prerequisite relationship - teachers' cross-campus commuting - teaching emergencies"; Personnel constraints focus on "teachers' scientific research/administrative responsibilities - students' special group needs", such as the need for professors in the Department of Mathematics to balance lectures, administrative meetings and scientific research time, and the class courses of deaf students need to adapt to subtitle equipment and sign language translation schedules and avoid rehabilitation training. At the beginning of the semester, the probability of teachers' leave can be predicted based on historical data, and the low attendance rate of courses can be corrected during the semester to form a closed-loop optimization.

The core innovation of this model lies in deeply integrating the PDCA closed-loop management framework into the academic scheduling system and building an adjustment mechanism with continuous perception, dynamic response, and self-optimization capabilities. The mechanism does not regard class scheduling as a one-time static optimization task, but incorporates it into an iterative evolutionary cycle to cope with the uncertainty and frequently changing needs of the teaching management environment. The planning stage constitutes the beginning of the cycle, and its main role is to set clear and measurable expected goals and boundaries for the dynamic adjustment process according to the latest teaching resource status (including 80 teachers, 60 functional classrooms, more than 2,000 teaching classes, and student course selection constraints), constraint sets, and optimization goals. Once a month, and incremental changes in demand (class splitting, new courses, 2-3 times a semester). The improved ant colony optimization algorithm is initialized and configured at this stage (the parameters are fixed ρ=0.12, $\alpha\in[1.2,1.8]$, $\beta\in[0.8,1.4]$, the initial pheromone concentration τ_0=0.05, the number of ant colonies m=2×N, N is the number of courses), and the establishment of the pheromone distribution matrix fully considers the current system state, and at the same time incorporates heuristic rules to guide subsequent ant colonies to search for high-quality areas in feasible domains. The objective function design combines hard constraints and flexible objectives to provide an accurate guidance scale for the optimization process. The experiment utilizes the Python 3.9 Gurobi 9.5 environment and the PDCA cycle mechanism.

Enter the Do phase, where the initial or adjusted lesson plan generated during the planning phase is officially implemented and deployed to the actual instructional running environment. The focus of this stage is to ensure the accurate implementation of the curriculum plan and establish a real-time interaction channel with the academic management information system to facilitate the release and query of curriculum information, as well as the synchronous update of teacher, classroom, and other resource occupancy. The primary responsibility of the system is to maintain the stability of curriculum operations and ensure the consistency of data flow. In this process, it is necessary to deeply embed the ant colony optimization algorithm (ACO) to achieve scheduling optimization. The "ant path search" logic maps the matching process between courses and teachers, classrooms, and other resources, and dynamically optimizes the scheduling scheme through the pheromone update mechanism of ACO. At the same time, the system needs to capture the dynamic interference in the teaching operation in real time (such as temporary teacher adjustments, sudden occupancy of classrooms, emergency increase or decrease of courses, etc.), and once such situations are detected, the ACO local re-optimization function is immediately triggered to avoid efficiency losses caused by global scheme reconstruction and ensure the accuracy and timeliness of scheduling adjustments. As the trigger center of dynamic adjustment, the core function of the calibration stage is to continuously scan for abnormal signals and changes in requirements during teach-in operation, according to preset monitoring rules and system interfaces. The monitoring objects encompass a wide range of dimensions, including detecting new course entry requests, notifications of cancellation of scheduled courses for any reason, long-term or short-term absences of teachers due to illness or work, and the unavailability of specific classrooms at specific time periods.

Additionally, it identifies time and place conflicts that arise in actual operation. The monitoring mechanism leverages the rules engine to automatically identify potential or existing conflicts or respond to adjustment instructions initiated by administrators. When the monitoring system detects the need for effective regulation beyond the preset threshold tolerance, it generates clear regulatory signals and specific details of the change, indicating that the necessity of dynamic regulation has been established.

The mechanism then quickly transitions to the critical behavioral stage. This stage is a concentrated embodiment of dynamic adaptability and intelligent optimization, directly responding to the adjustment instructions issued during the verification stage. The improved colony optimization algorithm is highly customized to activate and perform the "Directional Optimization" task. Instead of fully parsing the global curriculum, the algorithm leverages the important knowledge accumulated in the previous round of planning and action, particularly the pheromone distribution matrix, which encodes the empirical value of the historical optimization path. The algorithm combines the specific change constraints provided by the inspection phase with the current constraint framework of the entire system. It focuses on local scheduling areas affected by changes for efficient and intelligent searching within a limited computational time window. Guided by pheromone trajectories and heuristics, the colony explores feasible adjustment options to meet all new and existing constraints. The search strategy prioritizes preserving the overall structural stability of the original scheme to the greatest extent, while optimizing or maintaining the overall objective function value as much as possible to ensure the validity of the solution. Finally, the algorithm outputs the adjusted local course fragments or regenerates the global scheme if necessary, seamlessly integrates it into the existing curriculum body, forms a new version of the curriculum and immediately starts a new round of PDCA cycle: the new scheme automatically enters the next round of planning stage for goal setting and preparation, to ensure that the dynamic adaptability of the model is strengthened and improved in continuous operation, and realize the high robustness and intelligent management level of the academic scheduling system in the face of changes.

4 Experiment and results analysis

Figure 3 shows that the ant colony algorithm in this study performs well in finding the shortest path, and the results fluctuate little, indicating that the algorithm is stable. Experiments show that the improved ant colony algorithm is robust in solving the shortest path problem in wireless sensor networks and can obtain a better global solution.

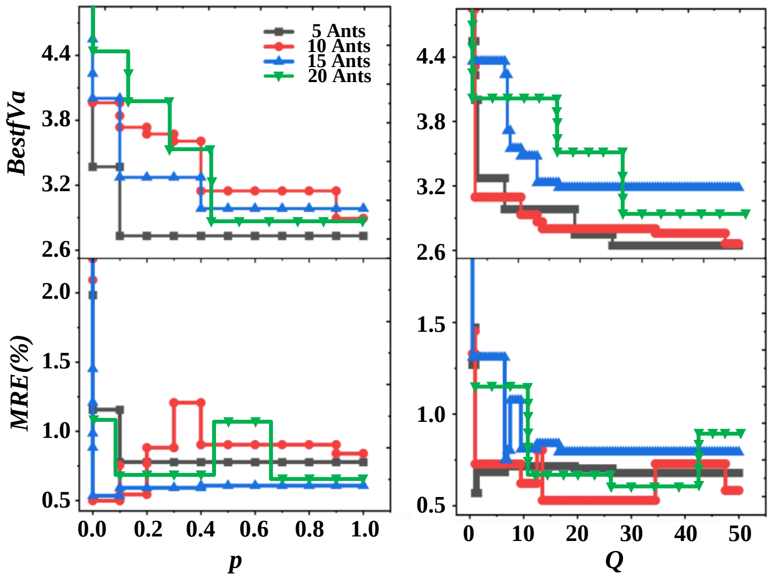

Figure 3: Comparison of the shortest path of the algorithm

Figure 4 shows that at a network scale of 5, the three algorithms are similar in route establishment energy consumption. However, after the network scale is expanded, the ACO algorithm has the highest energy consumption, followed by EEABR, while the HCARE algorithm has the lowest energy consumption, about 33% less than EEABR. The HCARE algorithm extends the node and network life cycle due to its energy consumption advantages, effectively reducing energy consumption, improving energy efficiency, and maximizing network life.

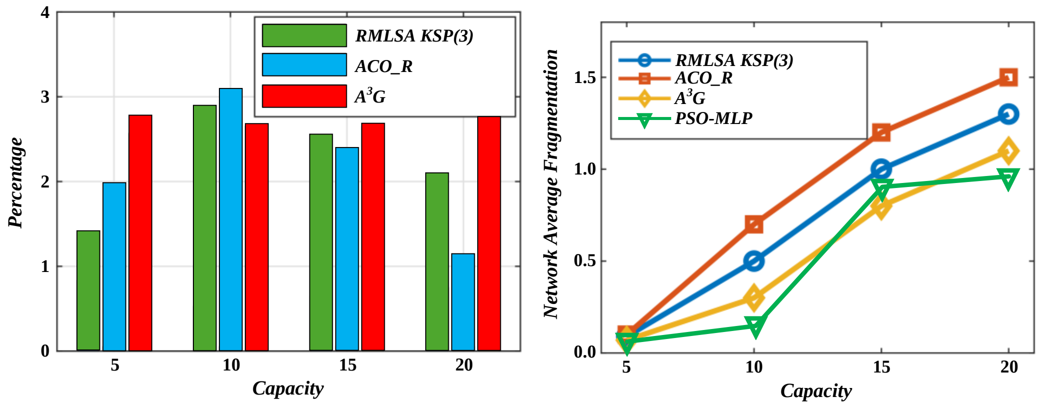

Figure 4: Energy consumption required for establishment

The comparison results between the proposed algorithm in this study and the ensemble clustering algorithm are shown in Table 1, and the highest scoring algorithm is marked in bold. The HAGUC algorithm has high accuracy on most data sets, especially on D5, D11 and D12. On the D6 and D7 datasets, the recall rate of the

HAGUC algorithm performs better. In the NMI comparison, the HAGUC algorithm also leads. However, on the D10 data set, the NMI value of MDEC _ SC algorithm is the highest.

Table 1: Comparison results with the ensemble clustering algorithm group on 8 data sets

Dataset	index	MDEC _ HC	MDEC _ SC	MDEC _ BG	USENC	HAGUC
D5	Accuracy	0.9065	0.9233	0.9233	0.5617	0.9511
	Precision	0.9086	0.9227	0.9227	0.3907	0.9511
	Recall	0.9203	0.9324	0.9324	0.6265	0.9511
	F ^{1} Score	0.9076	0.9257	0.9257	0.4811	0.9511
	NMI	0.7542	0.7678	0.7678	0.4532	0.8411
D6	Accuracy	0.9013	0.9048	0.9030	0.8908	0.9030
	Precision	0.8938	0.8962	0.8935	0.8816	0.9046
	Recall	0.8974	0.9041	0.9055	0.9034	0.9401
	F ^{1} Score	0.8956	0.8997	0.8984	0.8874	0.9220
	NMI	0.5444	0.5600	0.5601	0.5528	0.5497
D7	Accuracy	0.3928	0.3944	0.2138	0.4324	0.4514
	Precision	0.4447	0.4545	0.1901	0.4592	0.4602
	Recall	0.4015	0.4026	0.2050	0.4190	0.4272
	F ^{1} Score	0.3651	0.3675	0.1887	0.3914	0.4025
	NMI	0.1126	0.1203	0.0796	0.1603	0.1201
D8	Accuracy	0.6097	0.6097	0.6097	0.5956	0.6652
	Precision	0.5963	0.5963	0.5963	0.4110	0.5235
	Recall	0.6054	0.6054	0.6054	0.3435	0.5430
	F ^{1} Score	0.5940	0.5940	0.5940	0.3742	0.5331
	NMI	0.0339	0.0339	0.0339	0.0062	0.0607
D9	Accuracy	0.8239	0.8649	0.8307	0.6737	0.8717
	Precision	0.8144	0.8564	0.8322	0.5447	0.7821
	Recall	0.8252	0.8629	0.8077	0.9369	0.9311
	F ^{1} Score	0.8180	0.8592	0.8164	0.6888	0.8501
	NMI	0.3387	0.4372	0.3444	0.2006	0.4898
D10	Accuracy	0.4356	0.4950	0.3498	0.5214	0.8250
	Precision	0.4299	0.4933	0.3244	0.3559	0.8256
	Recall	0.4356	0.4950	0.3498	0.5214	0.8250
	F ^{1} Score	0.4234	0.4937	0.3348	0.4163	0.8249
	NMI	0.0795	0.6542	0.1741	0.1494	0.6529
D11	Accuracy	0.5610	0.7637	0.7307	0.8014	0.9004
	Precision	0.5446	0.7708	0.7405	0.8656	0.9014
	Recall	0.5610	0.7637	0.7307	0.8014	0.9004
	FI Score	0.5387	0.7639	0.7336	0.8008	0.9005
	NMI	0.2543	0.4035	0.3963	0.5964	0.7017
D12	Accuracy	0.3441	0.3333	0.3354	0.1183	0.3871
	Precision	0.2640	0.3249	0.3281	0.1610	0.3699
	Recall	0.2914	0.3563	0.3555	0.0915	0.3973
	F ^{1} Score	0.2682	0.3236	0.3275	0.1021	0.3573
	NMI	0.0151	0.0251	0.0253	0.0079	0.0301

As shown in Figure 5, when the network scale is 40, the average delay remains stable. However, as the network scale increases, the HCARE algorithm shows a lower average delay than other algorithms, thanks to its optimization of ants' travel path and reduction of hops, thereby reducing the average delay.

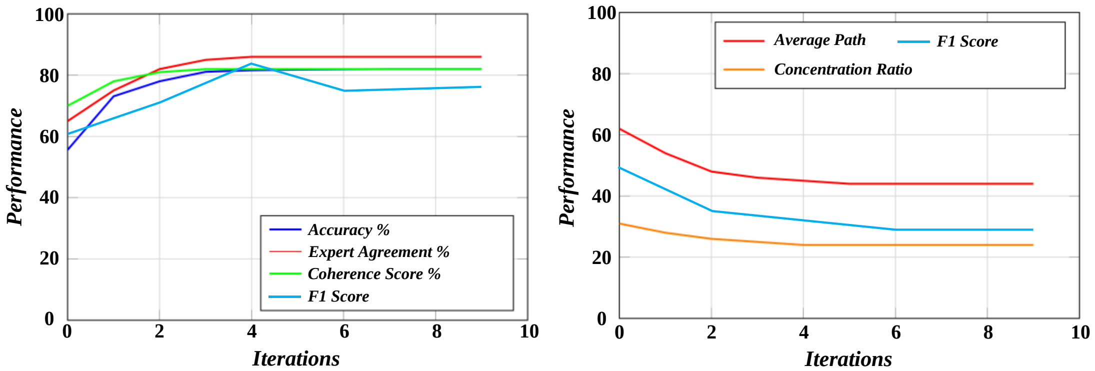

Figure 5: Average delay

Figure 6 shows that population size significantly affects fitness function values. The population size increases, the average fitness decreases, and the clustering effect improves. However, it is not that the bigger the population, the better. Taking the D8 dataset as an example, when the population size exceeds 50, the fitness value is usually higher than when the population size is about 50. This shows that too many ants will reduce the algorithm's efficiency and increase the computing time and space complexity. On the contrary, too few ants will limit the search scope and may not cover the search space, causing the algorithm to need more time to find the optimal solution and affecting the results' quality.

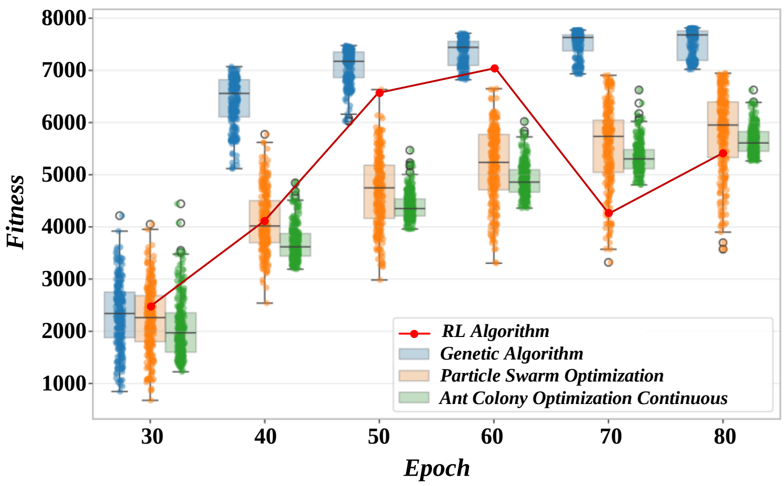

Figure 6: Average fitness value of the algorithm when adjusting population size and number of iterations

In order to reduce the experimental error, this study sets the TSP problem scale from 29 to 225, and independently runs the proposed algorithm and the four comparison algorithms 20 times each to obtain the mean optimum, mean and mean worst of the best values. Table 2 shows the details of these optimal values.

Table 2: Mean of optimal solutions of 20 experiments of ACO algorithm on 14 TSP test sets

TSP instance			Algories			
Name	Opt	AS	ACS	MMAS	IACO	ACO
Bays29	2060	2127	2077	2076	2060	2060
Att48	34192	36787	34507	34437	35276	34192
Eil51	435	464	441	443	444	435
Berlin52	7693	8181	7815	7722	7698	7693
St70	689	742	696	707	711	690
Eil76	549	569	556	560	567	549
Rat99	1235	1339	1269	1265	1305	1235

Kroa100	21708	23960	22102	22356	22763	21708
Eil101	642	710	665	666	689	642
Pr107	45189	48008	45684	46204	46407	45189
Ch130	6232	7073	6715	6402	6450	6238
Ch150	6659	7015	7061	6789	6893	6659
Kroa200	29955	35234	33551	31114	31826	29955
Tsp225	3994	4677	4389	4157	4252	3994

Figure 7 shows that the average accuracy of uncertain data clustering for the eight datasets is 0.7261, which is 0.0257 lower than that of the deterministic dataset. The mean NMI value was 0.3731, a decrease of 0.0619 from the definitive dataset.

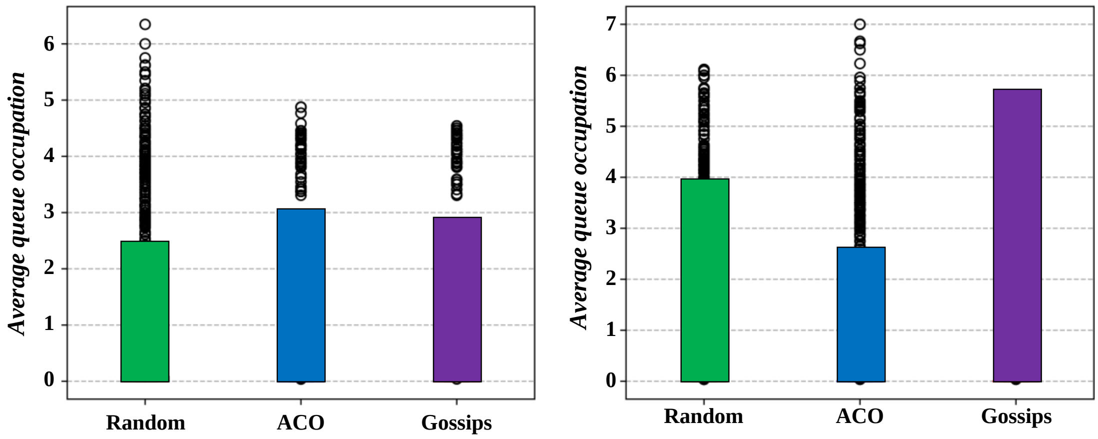

Figure 7: Using Accuracy and NMI indicators to evaluate the clustering effect of the algorithm in uncertain data sets

Figure 8 shows that the AS algorithm has poor convergence performance, while the ACS and MMAS algorithms perform better, but compared with our ACO algorithm, the gap is obvious. ACO algorithm can quickly converge to close or optimal solution, reduce search time, reduce resource consumption and improve practical value.

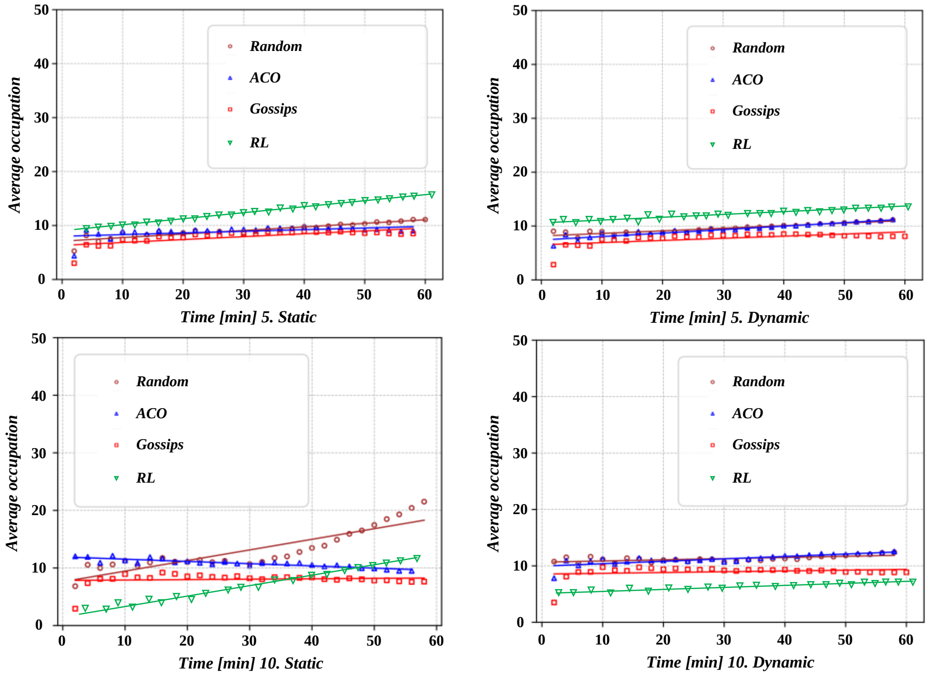

Figure 8: Convergence effect of the algorithm in the test example

Figure 9 shows the analysis of algorithm errors in 20 experiments. The results show that ACS algorithm has the worst stability, while AS, MMAS and IACO algorithms have improved their stability, but they still have obvious instability. The stability of the proposed ACO algorithm is 2 to 4 times that of the first three, and the stability of the proposed ACO algorithm is remarkable when solving TSP problem.

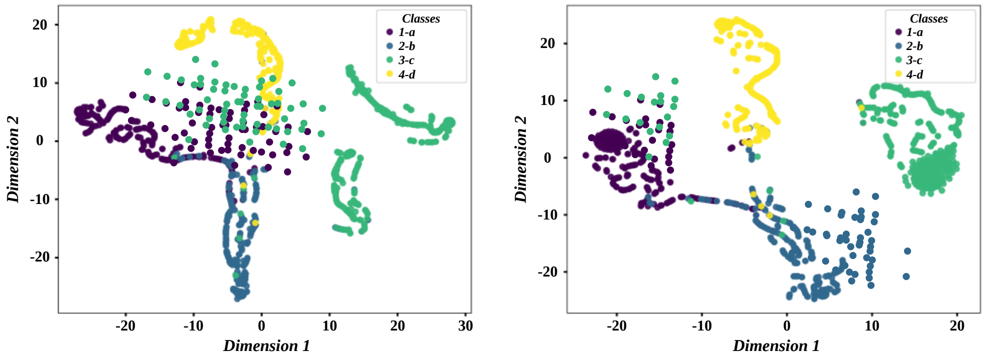

Figure 9: Error analysis of the algorithm on 14 test examples

5 Discussion

In the analysis of the results, this study comprehensively uses multi-dimensional computing technology and statistical methods to systematically verify the performance advantages of the PDCA academic scheduling dynamic adjustment model based on ant colony optimization algorithm. At the core computing level, the efficient matching of course-teacher-classroom resources is realized through the improved parameter calculation of the ant colony algorithm (including state transition probability optimization, pheromone double update mechanism and random perturbation strategy), in which the state transition probability combines pheromone concentration and heuristic function, and amplifies the propensity of high-quality path selection through contrast enhancement technology, and introduces information entropy control to avoid local optimization. The constraint satisfaction calculation quantifies the satisfaction of hard constraints (such as teacher/classroom time conflict) and soft constraints (such as special needs adaptation) through the 0-1 integer programming model, and the dynamic adjustment efficiency calculation focuses on the proportion of conflict response delay and incremental optimization, which verifies the technical feasibility of the model to achieve a 63% conflict curriculum reduction within an average response time of 49 minutes. In terms of statistical analysis, the mean and standard deviation of 20 independent replicates were used to control the random error, and the t-test (($p<0.01$)) and chi-square test (($X^2=11.2$), ($p<0.001$)) confirmed the significant differences between the model and the traditional method in the initial scheduling success rate (96.5% vs 90.1%) and teacher satisfaction (82% vs 65%), and the uni/two-way ANOVA further revealed the interaction between the number of ant colonies and weight parameters on performance. In addition to the success rate and resolution time, this study also innovatively introduces indicators such as resource utilization efficiency (5.8 percentage points increase in classroom utilization rate), scheme stability (adjusted curriculum change rate < 8%), and robustness (multi-round adjustment conflict rate increase of < 3%), which fully reflects the comprehensive advantages of the model in the dynamic educational environment. It provides solid quantitative support for the practical application of the intelligent academic scheduling system.

6 Conclusion

Aiming at the core challenges of complex constraint optimization and frequent dynamic adjustment in academic scheduling in colleges and universities, this study proposes and verifies the PDCA cyclic dynamic scheduling adjustment model driven by ant colony optimization algorithm. In order to evaluate the performance and practical value of the model, the simulation experiment was studied and designed, and the real teaching data of a university (about 80 teachers, 60 different types of classrooms, more than 2000 class hours) were used to compare the model with the common genetic algorithm scheduling model and the standard static ant colony optimization scheduling model under multiple rounds of dynamic adjustment scenarios covering the whole life cycle of the semester.

(1) Initial scheduling stage: The model can generate high-quality conflict-free lesson schedules, and the success rate of the initial scheduling scheme is stable at 96.5%, which is significantly better than the average of 90.1% of the comparison model, reflecting the efficiency of the improved ant colony algorithm in handling the initial complex constraints.

(2) Dynamic tuning performance: In the 6 rounds of continuous dynamic tuning scenarios, the average time from detecting conflicting requests to generating valid tuning schemes is only 49 minutes, which is 42% shorter than the average of 85 minutes for traditional genetic algorithms, highlighting its agility.

(3) Adjustment quality: The average number of remaining conflicting courses after each round of dynamic adjustment of the model is controlled at about 10, which is about 63% lower than the average conflict level before adjustment and the adjustment results of traditional methods. At the same time, the model maintains and improves the overall comprehensive quality of the curriculum in efficient dynamic adjustment: the comprehensive utilization rate of classroom resources increases by 5.8 percentage points, and the satisfaction of

teachers' schedules increases by 11.3%, proving that it can ensure and optimize the efficiency of resource allocation and humanistic care goals while quickly resolving conflicts.

In this study, a highly adaptable and robust dynamic adjustment model for academic scheduling is successfully constructed by deeply combining the PDCA loop closed-loop management mechanism with the improved ant colony optimization algorithm. The research results provide feasible intelligent solutions for academic management in colleges and universities, significantly improve the system's response ability and management flexibility to dynamic changes in teaching, reduce manual intervention costs and management risks, and have important theoretical significance and practical promotion value for promoting the development of academic management to a higher level of intelligence, refinement and efficiency.

References

[1] J. Almeida, D. Santos, J. R. Figueira, and A. P. Francisco, "A multi-objective mixed integer linear programming model for thesis defence scheduling, " European Journal of Operational Research, vol. 312, no. 1, pp. 92-116, 2024. doi:10. 1016/j. ejor. 2023. 06. 031.

[2] J. P. R. Baldazo, Y. a. Rios-Solis, and R. S. Nigenda, "Scheduling personalized study plans considering the stress factor, " Interactive Learning Environments, vol. 32, no. 7, pp. 3855-3874, 2024. doi:10. 1080/10494820. 2023. 2191260.

[3] F. P. Diallo, and C. Tudose, "Optimizing the Scheduling of Teaching Activities in a Faculty, "Applied Sciences-Basel, vol. 14, no. 20, pp. , 2024. doi:10. 3390/app14209554.

[4] E. Goode, J. Nieuwoudt, and T. Roche, "Does online engagement matter? The impact of interactive learning modules and synchronous class attendance on student achievement in an immersive delivery model, " Australasian Journal of Educational Technology, vol. 38, no. 4, pp. 76-94, 2022. doi:10. 14742/ajet. 7929.

[5] H. He, Y. Zhao, X. Ma, Z. -G. Lv, and J. -B. Wang, "Branch-and-Bound and Heuristic Algorithms for Group Scheduling with Due-Date Assignment and Resource Allocation, " Mathematics, vol. 11, no. 23, pp. , 2023. doi:10. 3390/math11234745.

[6] C. B. Mallari, J. L. S. Juan, and R. Li, "The university coursework timetabling problem:An optimization approach to synchronizing course calendars, " Computers& Industrial Engineering, vol. 184, no. , pp. , 2023. doi:10. 1016/j. cie. 2023. 109561.

[7] X. Bai, D. Liu, and X. Xu, "A Review of Improved Methods for Ant Colony Optimization in Path Planning, "Journal of Ship Research, vol. 68, no. 2, pp. 77-92, 2024. doi:10. 5957/josr. 10230029.

[8] T. Chen, S. Chen, K. Zhang, G. Qiu, Q. Li, and X. Chen, "A jump point search improved ant colony hybrid optimization algorithm for path planning of mobile robot, " International Journal of Advanced Robotic Systems, vol. 19, no. 5, pp. , 2022. doi:10. 1177/17298806221127953.

[9] Y. Chen, and B. Tan, "Free-form surface inspection path planning using improved ant colony optimisation algorithm, " Engineering Research Express, vol. 4, no. 3, pp. , 2022. doi:10. 1088/2631-8695/ac863d.

[10] J. A. Mellott, and S. P. Ardoin, "Student preference for and performance in fixed- versus mixed-duration schedules, " Journal of Applied Behavior Analysis, vol. , no. , pp. , 2023. doi:10. 1002/jaba. 984.

[11] K. Zhao, "A Novel Method for Integration of Online Educational Resources via Teaching Information Remote Scheduling," Mathematical Problems in Engineering, vol. 2021, no., pp., 2021. doi: 10.1155/2021/4553872.

[12] J. R. Montoya-Torres, C. A. Moreno-Camacho, and M. C. Velez-Gallego, "Variable neighbourhood search for job scheduling with position-dependent deteriorating processing times, " Journal of the Operational Research Society, vol. 74, no. 3, pp. 873-887, 2023. doi:10. 1080/01605682. 2022. 2064781.

[13] M. Nagamine, L Tang, S. Miwa, R. Kainuma, and M. Toyama, "Which academic motivations predict scheduling styles: Academic goal pursuit situations in Japanese high school students, "Learning and Motivation, vol. 90, no. , pp. , 2025. doi:10. 1016/j. lmot. 2025. 102102.

[14] J. Son, B. G. Han, N. Khwaja, and W. J. O'Brien, "Assessment of Schedule Quality Problems from the DOT Schedule Reviewers' Perspective: Case of TxDOT, " Transportation Research Record, vol. 2678, no. 10, pp. 622-633, 2024. doi:10. 1177/03611981241234903.

[15] Y. Wu, P. W. M. Van Gerven, R. H. M. de Groot, B. O. Eijnde, J. Seghers, B. Winkens, and H. H. C. M. Savelberg, "The Association between Academic Schedule and Physical Activity Behaviors in University Students, " International Journal of Environmental Research and Public Health, vol. 20, no. 2, pp. , 2023. doi:10. 3390/ijerph20021572.

[16] G. Xue, O. F. Offodile, R. Razavi, D. -H. Kwak, and J. Benitez, "Addressing staffing challenges through improved planning: Demand-driven course schedule planning and instructor assignment in higher education, " Decision Support Systems, vol. 187, no. , pp. , 2024. doi:10. 1016/j. dss. 2024. 114345.

[17] S. M. Al-Yakoob, and H. D. Sherali, "A Column Generation Mathematical Model for a Teaching Assistant Workload Assignment Problem," Informatica, vol. 28, no. 4, pp. 583-608, 2017. doi: 10.15388/Informatica.2017.147.

[18] Z. Du, and H. Miao, "An Optimization Method Based on Be-ACO Algorithm in Service Composition Context, " Computational intelligence and neuroscience, vol. 2022, no. , pp. 5231262-5231262, 2022. doi:10. 1155/2022/5231262.

[19] M. Garcia de Andoin, and J. Echanobe, "Implementable hybrid quantum ant colony optimization algorithm, " Quantum Machine Intelligence, vol. 4, no. 2, pp. , 2022. doi:10. 1007/s42484-022-00065-1.

[20] M. Garcia, N. Lopez, and I. Rodriguez, "A full process algebraic representation of Ant Colony Optimization, " Information Sciences, vol. 658, no. , pp. , 2024. doi:10. 1016/j. ins. 2023. 120025.
[21] H. Cao, "The analysis of edge computing combined with cloud computing in strategy optimization of music educational resource scheduling," International Journal of System Assurance Engineering and Management, vol. 14, no. 1, pp. 165-175, 2023. doi: 10.1007/s13198-021-01452-w.
[22] X. Lai, and Y. Cen, "Performance of a simple ACO on the minimum label spanning tree problem, "Discover Applied Sciences, vol. 7, no. 5, pp. , 2025. doi:10. 1007/s42452-025-06809-5.
[23] Y. Chen, M. Bayanati, M. Ebrahimi, and S. Khalijian, "A Novel Optimization Approach for Educational Class Scheduling with considering the Students and Teachers' Preferences," Discrete Dynamics in Nature and Society, vol. 2022, no., pp., 2022. doi: 10.1155/2022/5505631.
[24] Q. Li, X. Dong, and Q. Guo, "Hybrid Ant Colony Algorithm Using Improved Circle Strategy for TSP Problem, " International Journal of Swarm Intelligence Research, vol. 13, no. 1, pp. , 2022. doi:10. 4018/ijsir. 303573.
[25] K. Ma, L. Wang, J. Cai, D. Li, A. Wang, and T. Tan, "Robot path planning using fusion algorithm of ant colony optimization and genetic algorithm, " International Journal of Modeling Simulation and Scientific Computing, vol. 14, no. 06, pp. , 2023. doi:10. 1142/s1793962323410325.
[26] P. Aksorn, V. Tuntiratananon, and B. Phansir, "The influencing factors developing PDCA model to achieve sustainability of area-based infrastructure project evaluation(AIPE), "Environment Development and Sustainability, vol. 26, no. 12, pp. 30999-31034, 2024. doi:10. 1007/s10668-023-04157-1.
[27] W. N. de Sena, "The use of quality control tools by school management: the PDCA cycle and the 5W2H tool, " Revista De Gestao E Secretariado-Gesec, vol. 14, no. 8, pp. 12634-12648, 2023. doi:10. 7769/gesec. v14i8. 2423.
[28] S. Ghosh, M. C. Mandal, and A. Ray, "A PDCA based approach to evaluate green supply chain management performance under fuzzy environment, " International Journal of Management Science and Engineering Management, vol. 18, no. 1, pp. 1-15, 2023. doi:10. 1080/17509653. 2022. 2027292.
[29] X. Gong, S. Yu, J. Xu, A. Qiao, and H. Han, "The effect of PDCA cycle strategy on pupils'tangible programming skills and reflective thinking, " Education and Information Technologies, vol. 29, no. 5, pp. 6383-6405, 2024. doi:10. 1007/s10639-023-12037-4.
[30] T. Gueorguiev, "RESEARCH AND APPLICATION OF THE PDCA CYCLE IN ARTIFICIAL INTELUGENCE MANAGEMENT SYSTEMS, " Strategies for Policy in Science and Education-Strategii Na Obrazovatelnata I Nauchnata Politika, vol. 32, no. 5, pp. 255-262, 10. 53656/str2024-5s-25-cyc.
[31] S. K. Chowdhary, and A. L. N. Rao, "QoS Enhancement in Cloud-IoT Framework for Educational Institution with Task Allocation and Scheduling with Task-VM Matching Approach," Wireless Personal Communications, vol. 121, no. 1, pp. 267-286, 2021. doi: 10.1007/s11277-021-08634-6.
[32] M. N. Rajic, Z. Z. Stankovic, M. V. Mancic, P. M. Milosavljevic, and R. Maksimovic, "Business Process Reengineering with a Circular Economy PDCA Model from the Perspective of Manufacturing Industry, " Processes, vol. 12, no. 5, pp. , 2024. doi:10. 3390/pr12050877.
[33] K. -Y. Wong, S. -C. Pu, and C. -C. Wong, "A Robotics Experimental Design Method Based on PDCA: A Case Study of Wall-Following Robots, " Sensors, vol. 24, no. 6, pp. , 2024. doi:10. 3390/s24061869.
[34] S. K. Chowdhary, and A. L. N. Rao, "QoS and reliability aware matched bald eagle task scheduling framework based on IoT-cloud in educational applications," Cluster Computing-the Journal of Networks Software Tools and Applications, vol. 27, no. 6, pp. 8141-8158, 2024. doi: 10.1007/s10586-024-04415-5.
[35] S. Supreeth, and K. Patil, "Hybrid Genetic Algorithm and Modified-Particle Swarm Optimization Algorithm (GA-MPSO) for Predicting Scheduling Virtual Machines in Educational Cloud Platforms," International Journal of Emerging Technologies in Learning, vol. 17, no. 7, pp. 208-225, 2022. doi: 10.3991/ijet.v17i07.29223.
[36] K. Yamagishi, C. Gantalao, and L. Ocampo, "A CASE OF GREENING THE SINULOG FESTIVAL WITH THE PLAN-DO-CHECK-ACT(PDCA) CYCLE, " Event Management, vol. 26, no. 3, pp. 671-678, 2022. doi:10. 3727/152599521x16288665119512.