Cooperative scheduling method for harvester and grain transport vehicle based on IMPA algorithm

Huanyu Liu*, Jiahao Luo, Lihan Zhang, Hao Yu, Shun Zou, Shuang Wang
(*Institute of Modern Agricultural Equipment, Xihua University, Chengdu 610039, China*)

Abstract: To address the issues of low response efficiency, poor timeliness, and high scheduling costs in the collaborative operation of harvesting and transportation, a cooperative scheduling model was constructed in this paper with the objectives of minimizing both scheduling costs and time. The model aimed to solve the challenge of coordinating harvesters and grain transport vehicles of different capacities under time window constraints, thereby completing the harvesting and transportation tasks across multiple fields efficiently. First, to improve the response efficiency and timeliness of harvesters and grain transport vehicles in complex operating environments, a task allocation method based on priority strategy and a response scheme were designed to enhance the efficiency of cooperative operations. Secondly, an Improved Marine Predators Algorithm (IMPA) was proposed by incorporating a radial transformation matrix, a dynamic search mechanism, and an enhanced opposition-based learning strategy to strengthen the overall optimization capability of the scheduling scheme. Additionally, non-dominated sorting and crowding distance calculation were integrated into the algorithm to optimize the solution set, leading to a more optimal scheduling plan. Simulation results demonstrated that, compared to the Non-dominated Sorting Genetic Algorithms-II (NSGA-II) and the Multi-objective Harris Hawks Optimization (MOHHO) presented in the literature, the proposed model significantly reduced scheduling costs and time, with a reduction of 7% and 2.9% in scheduling costs and a reduction of 2.6% and 8.3% in scheduling time, respectively. The results indicated that IMPA not only reduced scheduling costs and time but also enhanced the response capability and operational efficiency of cooperative scheduling, offering significant practical value in solving the collaborative scheduling problem for multi-field harvesting and transportation.
Keywords: cooperative operation, scheduling, harvester, grain transport vehicle, marine predators algorithm
DOI: 10.25165/j.ijabe.20261901.9373

Citation: Liu H Y, Luo J H, Zhang L H, Yu H, Zou S, Wang S. Cooperative scheduling method for harvester and grain transport vehicle based on IMPA algorithm. Int J Agric & Biol Eng, 2026; 19(1): 226–240.

1 Introduction

The multi-machine cooperation technology is an important component of the intelligent agricultural machinery field[1]. Scientifically and effectively planning multi-machine cooperative operation methods plays a crucial role in enhancing agricultural production efficiency and reducing costs[2]. However, with the scaling up of agricultural production and shifts in management models, rural farmland is increasingly being managed in a centralized manner, and the harvesting and transportation operations of crops are highly time-sensitive[3], increasing the demand for efficient agricultural machinery allocation and scheduling[4]. In addition to the independent scheduling mode of harvesters, there is an urgent need for an accurate and rational response scheme for grain transport vehicles to reduce scheduling time and avoid the waste of agricultural machinery resources. Therefore, to complete the complex harvesting and transportation tasks across multiple fields, it is essential to properly design the response schemes for harvesters and grain transport vehicles and scientifically optimize scheduling routes, which is of great significance for saving agricultural machinery resources and improving the efficiency of field harvesting and transportation[5].

To address the issue of cooperative scheduling between harvesters and grain transport vehicles, much research has been conducted. Many studies[6,7] have discussed the harvester harvesting and grain transport vehicles transportation problem as a multi-trip vehicle path problem with dynamic satellites, and a heuristic algorithm has been proposed to solve it based on the dynamics of the specific location that the grain transport vehicles need to respond to. Reference [8] transformed the response scheduling problem of harvesters and grain transport vehicles into an in-field logistics vehicle routing problem. However, while the model is applicable to large-scale field in-field route planning, it does not address the out-field route planning for grain transport vehicles. In terms of agricultural machinery resource allocation, Reference [9] proposed an improved immune reinforcement algorithm focused on achieving efficient and intelligent optimization for the collaborative scheduling of agricultural resource allocation and distribution vehicles, which enhanced the adaptability and global search capability of the algorithm. Reference [10,11] proposed a method for planning in-field and inter-field routes for grain transport vehicles unloading from multiple harvesters and established a route planning model, realizing remote scheduling management of multi-machine cooperative operations. Many studies have discussed the collaborative task allocation methods for multiple agricultural

Received date: 2024-09-13 **Accepted date:** 2026-01-06
Biographies: Jiahao Luo, MS, research interest: collaborative scheduling and path planning for multi-agricultural machines, Email: luojiahao@stu.xhu.edu.cn; **Lihan Zhang**, MS, research interest: scheduling and route planning for agricultural machines, Email: zhanglihan1@stu.xhu.edu.cn; **Hao Yu**, MS, research interest: multimodal sensing for agricultural applications, Email: lancit0387@gmail.com; **Shun Zou**, MS, research interest: path tracking control for agricultural machinery, Email: zs15532696790@163.com; **Shuang Wang**, Professor, research interest: intelligent agricultural machinery and equipment key technology, Email: wsh@mail.xhu.edu.cn.
***Corresponding author: Huanyu Liu**, Professor, research interest: intelligent agricultural machinery and equipment key technology. Institute of Modern Agricultural Equipment, Xihua University, Chengdu 610039, China. Tel: +86-28-87723323, Email: liuhy0528@163.com.

machines[12-16] , which can solve the task allocation problem by improving the heuristic optimization algorithm. In the study of emergency scheduling with multi-machine coordination, much literature has investigated the problem of scheduling agricultural machines in emergency situations[17,18], which effectively reduces the loss of scheduling time. References [19] focused on the cooperative scheduling of harvesters and grain transport vehicles, primarily concentrating on theoretical analysis and the development of digital systems for agricultural machinery, but failed to provide effective scheduling strategies. The response problem between harvesters and grain transport vehicles is a complex multi-objective combinatorial optimization problem, and existing cooperative scheduling studies remain relatively weak in this area.

In conclusion, existing research has rarely considered the complex cooperative scheduling problem between harvesters and grain transport vehicles of varying capacities under time window constraints to complete the harvesting and transportation tasks across multiple fields. This is particularly problematic during peak farming seasons, where low scheduling efficiency leads to the wastage of agricultural machinery resources and prolonged harvest times, posing significant challenges to optimizing scheduling costs and time. To ensure that the harvesting and transportation tasks across multiple fields are effectively completed while meeting timeliness requirements, it is essential to comprehensively consider key factors such as scheduling distance, time windows, and machinery capacity. Therefore, to address the issues of low response efficiency, poor timeliness, and high scheduling costs in multi-field harvesting and transportation tasks, a cooperative scheduling model was constructed in this paper, taking into account the relevant performance parameters of harvesters and grain transport vehicles, starting from real-world production scenarios. The goal is to minimize scheduling costs and reduce scheduling time to the greatest extent possible. To solve the complex response and matching problems between harvesters and grain transport vehicles, a task allocation method and response scheme based on a priority strategy was designed. To solve this model, the Marine Predators Algorithm (MPA) was employed as the framework. MPA is one of the most effective metaheuristic (MH) and swarm intelligence (SI) algorithms for solving various optimization problems, known for its fast convergence, high flexibility, and accurate computation[20]. However, the algorithm suffers from issues such as insufficient late-stage search capability and a tendency to fall into local optima. To enhance the overall optimization ability, a radial transformation matrix, dynamic search mechanism, and improved opposition-based learning strategy were introduced. Furthermore, non-dominated sorting and crowding distance calculation are incorporated to balance multiple objective functions, thereby maintaining population diversity. Finally, simulations with real field data were conducted, and statistical analysis of the algorithm was performed, demonstrating that the IMPA offers advantages in solving the cooperative scheduling problem between harvesters and grain transport vehicles. This approach holds significant theoretical and practical value for improving harvesting and transportation efficiency, saving agricultural machinery resources, and reducing costs.

2 Cooperative scheduling model for harvesters and grain transport vehicles

This paper primarily focuses on the cooperative scheduling problem of harvesters and grain transport vehicles with different capacities under time window constraints. The harvester, as the primary machine, performs the harvesting operations for each field, while the grain transport vehicle, as the secondary machine, responds based on factors such as the grain unloading efficiency, capacity, idle time, and travel time of the secondary machine. The response order of the secondary machines is determined accordingly. The unloading time of the primary machine and the time it reaches the unloading state are estimated. When unloading is required by the primary machine, a request command is sent to the secondary machines, and the optimal secondary machine is selected to proceed to the unloading point. The field is entered by the secondary machine through the entrance, and the unloading point is reached, where the unloading task is completed in cooperation with the primary machine. After unloading, the secondary machine waits for the next response. If fully loaded, the secondary machine is returned to the agricultural machinery depot to wait for further instructions, while the harvesting operation is continued by the primary machine. This process continues until all harvesting tasks are completed, at which point the primary machine also returns to the agricultural machinery depot.

2.1 Demand response mode for harvesting and transportation based on priority strategy

The grain transport vehicle's response target is the harvester, and the harvester's unloading time and the frequency of grain transport vehicle responses are related to the remaining harvestable volume in the field. To maximize the efficiency of the harvester, it is necessary to predict the time required for the harvester to complete the harvesting of the remaining fields and the time needed for the grain transport vehicle to travel to the unloading point. Optimizing the response scheme of the grain transport vehicle is crucial for reducing the harvester's waiting time and improving harvesting efficiency.

The response strategy for the harvester and grain transport vehicle in this paper can be described as: how to effectively reduce waiting time, simplify the response complexity between the harvester and the grain transport vehicle, and achieve optimal matching of the grain transport vehicle to complete the harvesting and transportation tasks. The demand response model for harvesting and transportation is shown in Figure 1.

2.2 Related assumptions and conditions

Variable symbols are defined as follows:

$F=\{f_1, f_2, \ldots f_k\}$—farmland set.

$H=\{h_1, h_2, \ldots h_k\}$—harvester set.

$G=\{g_1, g_2, \ldots g_k\}$—grain transport vehicles set.

$P=\{p_1, p_2, p_3, p_4\}$—tasks set. Where p_1 represents transportation, p_2 represents loading/unloading, p_3 represents harvesting operations, and p_4 represents returning. The variable k represents the maximum value.

A—farm machinery depot.

The problem can be described as follows: Multiple harvesters (h) and multiple grain transport vehicles (g) are required to perform various tasks (p). The objective of this study is to minimize both the scheduling cost and the scheduling time. Prior to conducting the study, several assumptions must be established.

(1) Each field can only be accessed by one harvester and at least zero grain transport vehicles.

(2) Both the harvesters and grain transport vehicles depart from the machinery depot and return to the depot after completing all tasks. The location of the machinery depot is known.

(3) The capacities and travel speeds of the harvesters and grain transport vehicles are known.

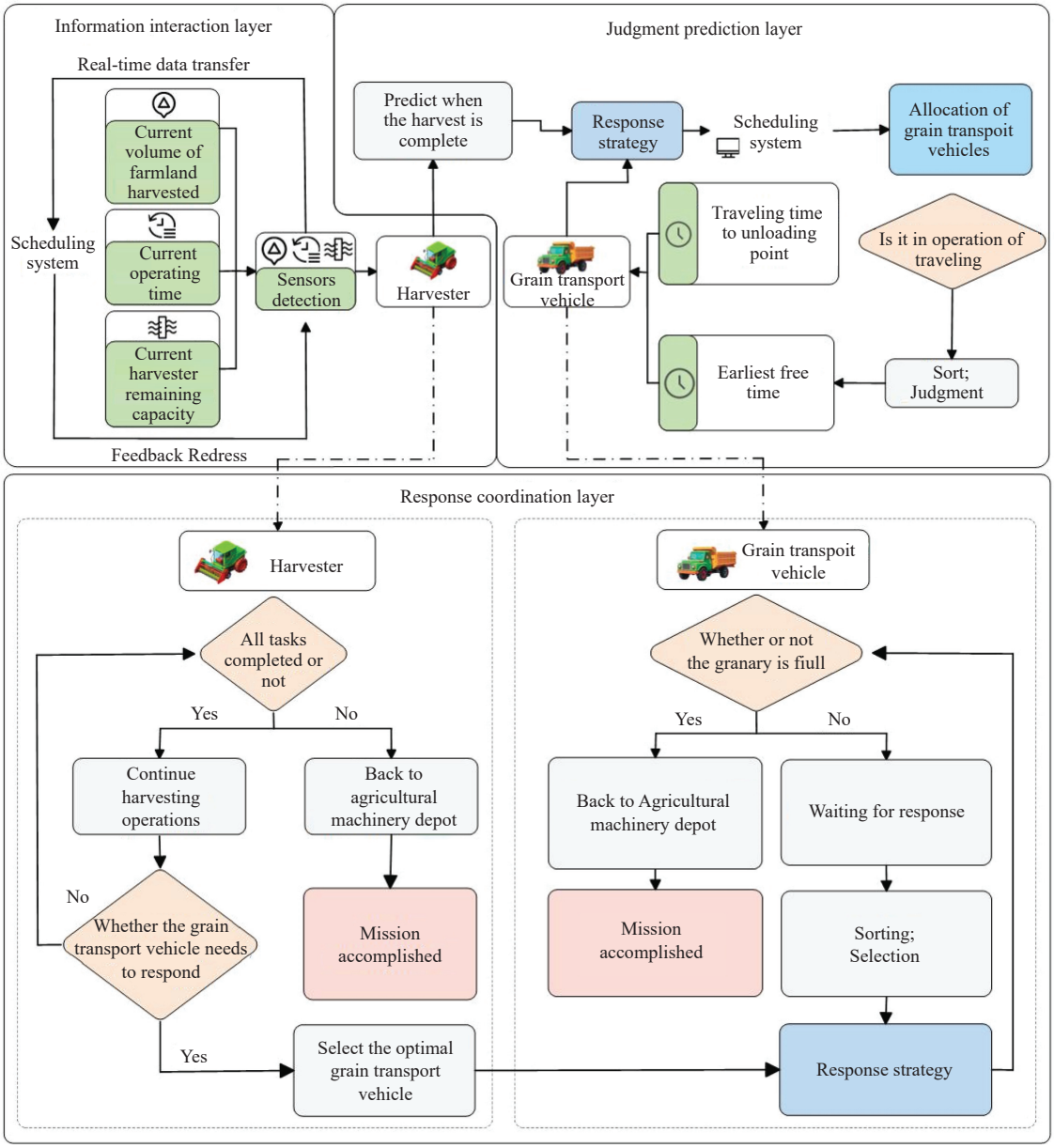

Figure 1 Demand response strategy

(4) The coordinates, time windows, and harvestable volumes of each field are known.

(5) The maximum load capacity of the harvesters and grain transport vehicles cannot exceed their respective capacities, and the grain transport vehicle must return to the machinery depot when fully loaded.

(6) The grain transport vehicle can only enter the field from the designated entrance and move to the unloading point for grain loading.

(7) The harvester will stop harvesting when fully loaded, and that location will serve as the unloading point.

(8) The time and distance for the grain transport vehicle to move within the field are ignored.

(9) It is assumed that no machinery failures or obstacles exist in the field during operations.

2.3 Cooperative scheduling model

In this paper, the cooperative scheduling problem for multiple machines is addressed by considering various factors that affect the operations of harvesters and grain transport vehicles. The objective functions are to minimize scheduling costs and time, as shown in Equation (1).

$$\begin{cases} \min T = \max(T_h, T_g) \\ \min C = C_1 + C_2 + C_3 \end{cases} \tag{1}$$

The shortest scheduling time is determined by the later of two times: T_h, the latest time for the harvester to complete all tasks, or T_g, the latest time for the grain transport vehicle to complete all tasks along the full path, as illustrated in Equation (2). Where, T_h includes the sum of transportation time, unloading time, and operation time. T_g includes the sum of beginning time, transportation time, and loading time. The lowest scheduling cost is the lowest sum of the harvester's distance cost from the grain transport vehicle, C_1; schedule, maintenance, and fuel cost, C_2; and the penalty cost for violating the time window, C_3, as shown in Equation (3).

$$\begin{cases} T_h = \max(Tr_h + U_h + O_h) \\ T_g = \max(B_g + Tr_g + L_g) \end{cases} \tag{2}$$

where, Tr_h denotes the transportation time of each path harvester, U_h denotes the unloading time of each path harvester, O_h denotes the operation time of each path harvester, B_g denotes the beginning time of each path grain transport vehicle, Tr_g denotes the

transportation time of each path grain transport vehicle, and L_g denotes the loading time of each path grain transport vehicle, where the transport time of the harvester is related to the traveling speed V_h and the transport time of the grain transport vehicle is related to the traveling speed V_g.

$$\begin{cases} C_1 = \sum_{i=1}^{F+A}\sum_{j=1}^{F+A} x_{ijh}x_{ijg}d_{ij}d_c \\ C_2 = \sum_{h=1}^{H} H_c + \sum_{g=1}^{G} G_c \\ C_3 = \sum_{i=1}^{F+H+G} P_{thg} \end{cases} \tag{3}$$

where, d_{ij} represents the distance from field i to field j, and d_c represents the travel cost per unit distance. H_c and G_c represent the schedule, maintenance, and fuel costs for harvester and grain transport vehicles, respectively. P_{thg} represents the time window penalty cost: the harvester and grain transport vehicle are only effective if they operate within the time window, where $[B_i, E_i]$ is the time window for field i. If the machinery arrives before the start of the time window, waiting time costs will be incurred, and if the machinery arrives after the start of the time window, penalty time costs will be incurred.

The constraints for the cooperative scheduling are as follows:

$$\begin{cases} \sum_{i=1}^{F}\sum_{h=1}^{H} x_{ijh} = 1 \\ \sum_{i=1}^{F}\sum_{g=1}^{G} x_{ijg} \geq 0 \end{cases} \tag{4}$$

$$\begin{cases} \sum_{i=1}^{A}\sum_{j=1}^{F}\sum_{h=1}^{H} x_{ijh} = H \\ \sum_{i=1}^{A}\sum_{j=1}^{F}\sum_{g=1}^{G} x_{ijg} = G \end{cases} \tag{5}$$

$$\begin{cases} \sum_{i=1}^{F} q_i x_{ijh} \leq H_v \\ \sum_{i=1}^{F} q_i x_{ijg} \leq G_v \end{cases} \tag{6}$$

$$u_h = l_g = D_i H_v / U_{eh} \tag{7}$$

$$P_{thg} = \begin{cases} C_w\left(B_i - t_{hgi}\right); \\ C_l\left(t_{hgi} - E_i\right); (i \in F) \\ 0, \text{ others}; \end{cases} \tag{8}$$

$$X_{ijh} = \begin{cases} 1, \text{ Harvester } h \text{ travels from field } i \text{ to field } j \\ 0, \text{ Harvester } h \text{ does not travel from field } i \text{ to field } j. \end{cases} \tag{9}$$

$$X_{ijg} = \begin{cases} 1, \text{ Grain transport vehicle } g \text{ travels from field } i \text{ to field } j \\ 0, \text{ Grain transport vehicle } g \text{ does not travel from field } i \text{ to field } j \end{cases} \tag{10}$$

Equation (4) states that each field can only be accessed by one harvester and at least zero grain transport vehicles. Equation (5) indicates that all harvesters and grain transport vehicles depart from and eventually return to the machinery depot. Equation (6) specifies that the maximum load capacity of the harvesters and grain transport vehicles cannot exceed their respective capacities; H_v, G_v denote harvester capacity and grain transport vehicle capacity, respectively. q_i represents the harvestable volume of field i. Equation (7) implies that the unloading time u_h and loading time l_g are equal, where D_i represents the operational density of field i, and U_{eh} represents the unloading efficiency of harvester h. Equation (8) refers to the soft time window constraint, where P_{thg} represents the cost of violating the time window, C_w represents the unit time waiting cost, C_l represents the unit time penalty cost, and B_i、E_i are the allowed start and latest times for tasks at field i, while t_{hgi} represents the time when the harvester or grain transport vehicle arrives at field i. X_{ijh} in Equation (9) and X_{ijg} in Equation (10) both represent decision variables, where a binary 0-1 variable is used to indicate whether the harvester or grain transport vehicle travels from field i to field j.

2.4 Cooperative unloading process

The current operational state of the harvester in the field is selected: waiting for a call to enter the unloading state W_t, operating time O_t, return time B_t, grain transport vehicle travel time G_t, loading/unloading time LU_t, time required to complete the remaining field harvest H_{ht}, and the state of completing the current field harvest operation F_h. The operational states of the harvester and the grain transport vehicle can be represented using time slots, with each time slot representing the duration of a specific task. Before unloading, the harvester sends a command, after which the grain transport vehicle responds in time H_{ht}, and the grain transport vehicle takes G_t time to travel. At time F_h, the grain transport vehicle reaches the unloading state. After time LU_t, the loading/unloading is completed. The cooperative unloading process can be divided into three scenarios, as shown in Figure 2. First, the harvester finishes harvesting and continues harvesting operations, and the grain transport vehicle loads grain and waits for a response; second, the harvester finishes harvesting and continues harvesting operations on the next farm field, and the grain transport vehicle responds to go to the unloading point; and third, the harvester and the grain transport vehicle return to the farm depot after completing all the harvesting and transportation tasks. For cases where the capacity of the harvester is greater than the harvested volume, the cooperative unloading step is not considered.

$$U_t = \frac{d_f \sum_{i=1}^{f}\sum_{j=1}^{f} x_{ijh}d_{ij}}{V_h} + D_i H_v / V_h g_n O_w \tag{11}$$

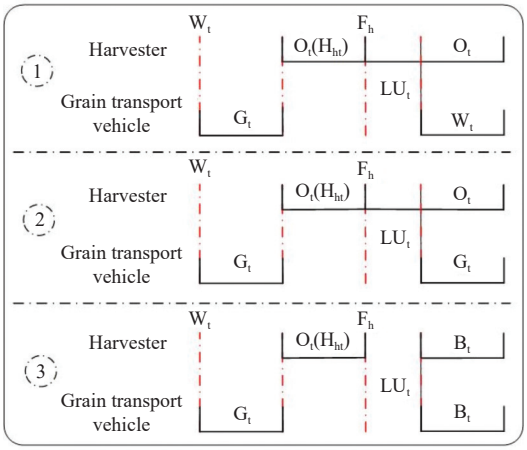

Figure 2 Unloading process

$$H_{ht} = U_t h_v / H_v \tag{12}$$

$$F_h = G_t + H_{ht} \tag{13}$$

where, U_t represents the time at which the harvester reaches the unloading state, d_f represents the turning distance coefficient, g_n represents the unit harvesting volume of the harvester, and O_w refers to the working width of the harvester; H_{ht} represents the time required for the harvester to complete the remaining field harvest, and h_v represents the current harvesting capacity of the harvester; F_h represents the time when the harvester completes the harvest operation of the field, and G_t represents the travel time of grain transport vehicle g to reach field i.

By establishing a mathematical model of the entire loading and unloading process as well as the cooperative scheduling, the operational states of the harvester and grain transport vehicle during the cooperative scheduling and harvesting/transportation process can be clearly expressed. This allows the harvester to effectively reduce waiting time, and fundamentally to reduce the overall scheduling time.

3 IMPA-based cooperative scheduling algorithm

3.1 Algorithm design for model solution according to the scheduling model

The cooperative scheduling problem of harvesters and grain transport vehicles studied in this paper is an NP-hard problem in a multi-objective complex environment. The multi-objective MPA has been proven to be one of the most effective methods for solving multi-objective optimization problems in recent years[21]. This algorithm is inspired by the extensive foraging mechanisms of marine predators based on Lévy and Brownian motion, as well as the optimal encounter rate strategy between predators and prey. Additionally, it considers environmental factors such as vortex formation and fish aggregating devices (FADs), simulating the theory of survival of the fittest in the ocean. Its powerful optimization capabilities and fast search speed have been widely applied to solve various optimization problems. The technical roadmap is shown in Figure 3.

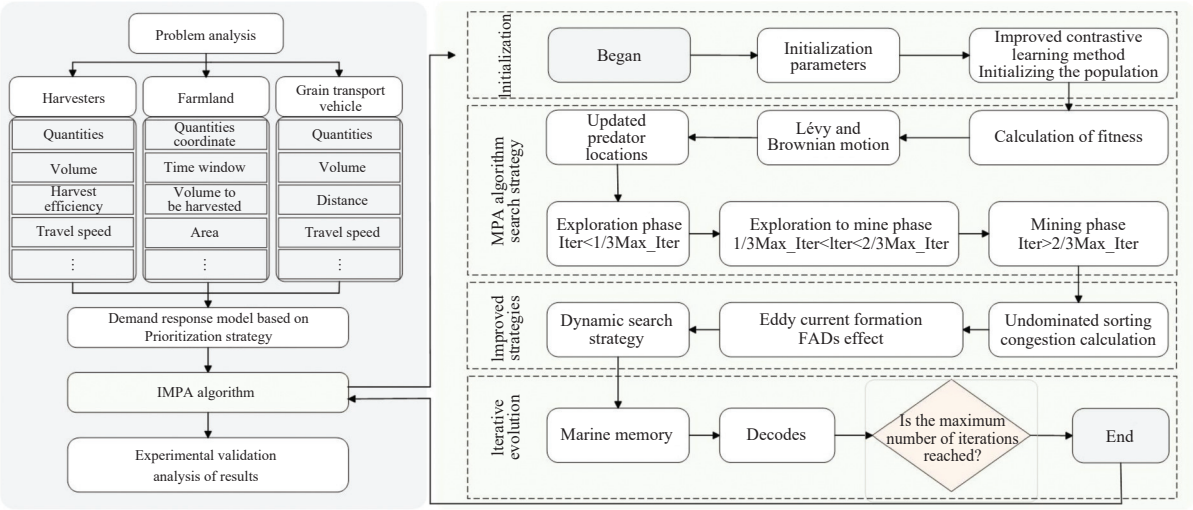

Figure 3 Technical roadmap

3.2 Model solution process based on the cooperative scheduling model

The specific steps of the IMPA are as follows:

(1) Parameter Setting: Set the basic parameters for fields, harvesters, and grain transport vehicles, as well as the relevant parameters for the algorithm.

(2) Chromosome Encoding and Decoding: In this paper, a real number encoding strategy is used to represent the fitness and behavior patterns of predators in the IMPA. Each chromosome consists of a string of floating-point numbers between 0 and 1, which correspond to the fitness values of different predators during the prey capture process. These fitness values represent the priority of a harvester to harvest the pending fields. The change in fitness is influenced by several factors, such as the distance between predator and prey (i.e., the distance between harvester and field) and the current state and strategic choices of the predator (i.e., the harvester's assigned task load). The decoding process converts the chromosome into a specific solution for the predator to capture prey (i.e., the strategy for assigning fields to harvesters). Through the iterative optimization of the improved MPA, the optimal field allocation scheme can be effectively found, thereby improving harvesting efficiency and field utilization. Figure 4 illustrates the encoding and decoding method, where the tasks of the harvester and grain transport vehicle are represented by the numbers 1, 2, 3, and 4, corresponding to different task states such as transportation, loading/unloading, operation, and return, respectively.

(3) Population Initialization: During the optimization process, the quality of the initial solutions significantly affects both the convergence speed and the quality of the final solution. However, the MPA suffers from insufficient diversity and coverage in the initial solutions, especially in high-dimensional and complex search spaces. Randomly generated solutions may fail to effectively cover critical areas, causing the algorithm to miss potential optimal regions from the beginning. Therefore, this section improves the initialization of the prey population and the fitness and elite matrices through an enhanced opposition-based learning (OBL) strategy. This method improves the accuracy and effectiveness of the initial solutions and expands the search range for feasible solutions. However, the traditional opposition-based learning strategy[22] often falls short when dealing with complex and combinatorial optimization problems. To address these shortcomings, an innovative "Improved Opposition-Based Learning Strategy," introduced in references [23, 24], is incorporated to enhance and extend the algorithm's search capability and accelerate the convergence of the MPA. The algorithm starts with a specific initialization function to generate the initial population. This

population consists of NP independent individuals, each representing a potential solution in the solution space. The calculation method for the initial solution is shown below:

$$X = lb + \mathrm{rd}(\mathrm{group}_{\mathrm{size}}, \mathrm{dim}) \times (ub - lb) \tag{14}$$

where, ub and lb are the upper and lower boundaries, and $\mathrm{rd}(\mathrm{group}_{\mathrm{size}}, \mathrm{dim})$ generates a random number matrix uniformly distributed between [0,1], ensuring that the initial individuals are uniformly distributed within the solution space, providing a comprehensive and diverse starting point for the algorithm's search.

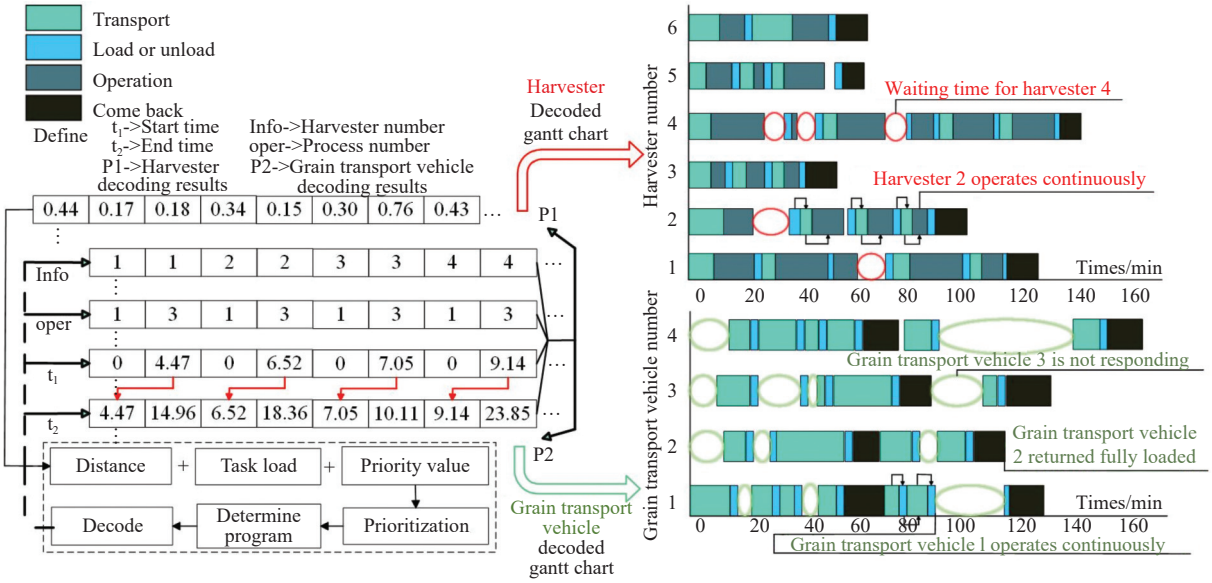

Figure 4 Encoding and decoding

After generating the initial solution, in order to improve search efficiency and promote the rapid convergence of the algorithm, an improved opposition-based learning strategy is introduced. This strategy enhances population diversity and generates opposite individuals (OPrey) for each initial individual.

$$\mathrm{OPrey} = 1 + \mathrm{size}(\mathrm{dim}, 2) - X \tag{15}$$

where, the dimensionality of the problem is expressed as (dim,2), ensuring the effective generation of opposite individuals in a multidimensional solution space. This allows for a comprehensive exploration of the dimensions and significantly improves the diversity of the population during the search process. The algorithm evaluates the fitness of both the individual and its opposite counterpart, and when the opposite individual demonstrates better fitness, it replaces the original individual. This update mechanism accelerates the algorithm's convergence and enhances its ability to escape local optima.

(4) Fitness Calculation: The fitness value of the prey matrix (Prey) needs to be calculated, the optimal position recorded, and an elite matrix (Elite) constructed. In the MPA, predators and prey interact with each other. Predators search for prey, while prey also search for food. The algorithm uses two key matrices to simulate this process, namely Elite and Prey. According to the theory of survival of the fittest, top predators in nature are more adept at hunting, so the most suitable solutions are referred to as top predators. These are used to construct a matrix called Elite, which guides the search operations and adjusts the position of the predators based on the movement of the prey. See Equation (16) for details.

$$Elite = \begin{bmatrix} X_{1,1}^{l} & X_{1,2}^{l} & \cdots & X_{1,d}^{l} \\ X_{2,1}^{l} & X_{2,2}^{l} & \cdots & X_{2,d}^{l} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ X_{n,1}^{l} & X_{n,2}^{l} & \cdots & X_{n,d}^{l} \end{bmatrix}_{n \times d} \tag{16}$$

where, $\overrightarrow{X^{l}}$ represents the top predator vector, which is replicated n times to construct the Elite matrix, and d is the number of dimensions. Additionally, when predators are searching for prey, the prey are also searching for food. At the end of each iteration, if a top predator is replaced by a better predator, the Elite matrix is updated accordingly.

Another matrix with the same dimensions as Elite is called Prey, and predators update their positions based on it. The Prey matrix is defined in Equation (17):

$$Prey = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,d} \\ X_{3,1} & X_{3,2} & \cdots & X_{3,d} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,d} \end{bmatrix}_{n \times d} \tag{17}$$

where, $X_{i,j}$ represents the j-th dimension of the i-th prey. In summary, the initialization process creates the initial prey, and the individuals with the highest fitness form the Elite matrix. The entire optimization process is primarily driven by the interaction between these two matrices.

(5) Lévy and Brownian Motion: In the early and late stages of the MPA, when exploring new search areas or when trapped in local optima, the properties of Lévy flights allow the algorithm to quickly cover a larger search space. This enhances the algorithm's search capabilities in complex environments, improves the quality of the optimization results, and achieves an effective balance between global and local search. In the mid-phase, Brownian motion enables more refined searches within local regions, as described below.

Lévy flight is a type of random walk that allows for long-distance jumps, with step lengths determined by a probability function:

$$L(x_j) \approx |x_j|^{1-\alpha} \tag{18}$$

where, x_j represents the flight length, and $1 < \alpha \le 2$ is the power-law exponent. The integral form of the probability density function for the Lévy stable process is shown as follows:

$$f_L(x;\alpha,\gamma) = \frac{1}{\pi}\int_0^{\infty} \exp(-\gamma q^{\alpha})\cos(qx)dq \tag{19}$$

where, α is defined as the distribution index and controls the scaling properties of the process, while γ selects the unit of scale. The integral in Equation (19) has an analytical solution only in a few cases. When α equals 2, it represents a Gaussian distribution, and when α equals 1, it represents a Cauchy distribution. Generally, the integral can only be solved using a series expansion when the value of x is very large, as shown in Equation (20):

$$f_L(x;\alpha,\gamma) \approx \frac{\gamma\Gamma(1+\alpha)\sin\left(\frac{\pi\alpha}{2}\right)}{\pi x^{(1+\alpha)}}, x \to \infty \tag{20}$$

where, Γ represents the Gamma function, and for integers, $\Gamma(1+\alpha)$ equals α.

In reference [25], a fast and accurate algorithm was proposed for generating a Lévy stable process for any value of the exponential distribution (α) within the range of 0.3-1.99. This paper uses the Magneta method from that reference to generate random numbers based on the Lévy distribution, as shown in Equation (21), where x and y are two normally distributed variables:

$$\text{Lévy}(\alpha) = 0.05 \times \frac{x}{|y|^{1/\alpha}} \tag{21}$$

$$\begin{cases} x = Normal\left(0,\sigma_x^2\right) \\ y = Normal\left(0,\sigma_y^2\right) \end{cases} \tag{22}$$

$$\sigma_x = \left[\frac{\Gamma(1+\alpha)\sin\left(\frac{\pi\alpha}{2}\right)}{\Gamma\left(\frac{(1+\alpha)}{2}\right)\alpha\times 2^{\frac{(\alpha-1)}{2}}}\right]^{1/\alpha} \tag{23}$$

where, σ_x and σ_y represent the standard deviation, where the calculation method for σ_x is shown in Equation (23). σ_y equals 1, α equals 1.5.

In the mid-phase of the MPA, when the speed ratio between the predator and the prey is close, Brownian motion helps the algorithm perform more refined searches within the local area. This motion is a random process, with step lengths determined by a probability function defined by a normal (Gaussian) distribution, where the mean is zero ($\mu = 0$) and the unit variance ($\sigma^2 = 1$). The control probability density function at point x is shown in Equation (24):

$$f_B(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2}\right) \tag{24}$$

(6) Predator Position Update: Based on the speeds of the prey and predators, the optimization phase of the MPA is divided into three stages. The corresponding update method is selected according to these stages, and the predator positions are updated accordingly. The pseudocode for this optimization phase is shown in Table 1.

Stage 1: Exploration Phase (High Speed Ratio): This occurs when the predator moves faster than the prey, typically during the initial iterations of the optimization process. When $Iter < \frac{1}{3}Max\_Iter$, the mathematical model for this rule is shown in Equation (25):

$$\begin{cases} \overrightarrow{stepsize_i} = \vec{R}_B \otimes \left(\overrightarrow{Elite_i} - \vec{R}_B \otimes \overrightarrow{Prey_i}\right), \ i = 1,\ldots,n \\ \overrightarrow{Prey_i} = \overrightarrow{Prey_i} + 0.5 * \vec{R} \otimes \overrightarrow{stepsize_i} \end{cases} \tag{25}$$

where, $Iter$ represents the current iteration number, Max_Iter represents the maximum number of iterations, $\overrightarrow{stepsize_i}$ represents the step size, and $\overrightarrow{Prey_i}$ represents the prey. $\vec{R}_B$ represents Brownian motion, which is a vector containing random numbers created based on a normal distribution. $\vec{R}_B$ is multiplied by the prey to simulate the prey's movement. $\vec{R}$ is a vector of uniformly distributed random numbers in the range [0,1].

Table 1 Pseudocode of optimization phases

If *Iter*<*Max_Iter*/3 then
Update prey based on Equation (25)
Else if *Max_Iter*/3<*Iter*<2**Max_Iter*/3 then
For the first half of the solutions (i = 1, ..., n/2).
Update prey based on Equation (26)
For the second half of the solutions (i = 1, ..., n/2).
Update prey based on Equation (27)
Else if *Iter*>2**Max_Iter*/3
Update prey based on Equation (29)

Stage 2: Transition Between Exploration and Exploitation (Unit Speed Ratio): In this stage, the predator and prey move at the same speed or at a unit speed ratio. The prey is responsible for exploitation, while the predator focuses on exploration. According to this rule, under the condition of a unit speed ratio ($v \approx 1$), this study employs both Lévy and Brownian strategies. The prey moves using the Lévy strategy, while the predator follows the Brownian strategy. When $\frac{1}{3}Max_{Iter} < Iter < \frac{2}{3}Max_{Iter}$, the mathematical model for this rule, for the first half of the population, is shown in Equation (26):

$$\begin{cases} \overrightarrow{stepsize_i} = \vec{R}_L \otimes \left(\overrightarrow{Elite_i} - \vec{R}_L \otimes \overrightarrow{Prey_i}\right), \ i = 1,\ldots,n/2 \\ \overrightarrow{Prey_i} = \overrightarrow{Prey_i} + 0.5 * \vec{R} \otimes \overrightarrow{stepsize_i} \end{cases} \tag{26}$$

where, $\vec{R}_L$ represents the random number vector for Lévy motion. $\vec{R}_L$ is multiplied by Prey to simulate the movement of prey using the Lévy strategy, while adding a step size to the prey's position further simulates its movement.

Since most step lengths in the Lévy distribution are related to small steps, to improve applicability, the assumption for the other half of the population is shown in Equation (27):

$$\begin{cases} \overrightarrow{stepsize_i} = \vec{R}_B \otimes \left(\vec{R}_B \otimes \overrightarrow{Elite_i} - \overrightarrow{Prey_i}\right), \ i = n/2,\ldots,n \\ \overrightarrow{Prey_i} = \overrightarrow{Elite_i} + 0.5 * CF \otimes \overrightarrow{stepsize_i} \end{cases} \tag{27}$$

$$\text{CF} = \left(1 - \frac{Iter}{Max\_Iter}\right)^{\left(2\frac{Iter}{Max\_Iter}\right)} \tag{28}$$

where, $\vec{R}_B$ is multiplied by Elite to simulate the Brownian motion of the predator, while the prey updates its position based on the predator's Brownian motion. CF is an adaptive parameter that controls the step size of the predator's movement, and its calculation is shown in Equation (28).

Stage 3: Exploitation Phase (Low Speed Ratio): In this phase, the predator moves slower than the prey, with a low speed ratio (v = 0.1). Under this condition, the best strategy for the predator is to use the Lévy motion. When $Iter > \frac{2}{3}Max_{Iter}$, the mathematical model for this phase is shown in Equation (29):

$$\begin{cases} \overrightarrow{stepsize_i} = \vec{R}_L \otimes \left(\vec{R}_L \otimes \overrightarrow{Elite_i} - \overrightarrow{Prey_i}\right), \ i = 1,\ldots,n \\ \overrightarrow{Prey_i} = \overrightarrow{Elite_i} + 0.5 * CF \otimes \overrightarrow{stepsize_i} \end{cases} \tag{29}$$

where, $\vec{R}_L$ is multiplied by Elite to simulate the predator's movement in the Lévy strategy, and adding a step size to the Elite position simulates the predator's movement, which helps update the prey's position.

(7) Non-dominated Sorting and Crowding Distance Calculation: The MPA is typically designed for single-objective optimization problems, but this study addresses a complex multi-objective optimization problem, requiring an extension of the MPA. Often, conflicts arise between multiple objective functions and multi-objective optimization problems, making it necessary to find a set of good solutions in the solution space. By introducing the concept of Pareto dominance, constructing a non-dominated solution set can effectively balance multiple objective functions. To maintain population diversity, crowding distance calculation is introduced to assess the density between individuals, thereby optimizing the solution set.

$$\begin{cases} D(ind,aim)=\dfrac{f_{aim}(ind+1)-f_{aim}(ind-1)}{f_{aim}^{\max}-f_{aim}^{\min}} \\ D(ind)=\sum\limits_{aim=1}^{M}D(ind,aim) \end{cases} \tag{30}$$

where, $D(ind,aim)$ represents the distance of individual ind in objective aim; $f_{aim}^{\max}$ and $f_{aim}^{\min}$ represent the maximum and minimum values of objective aim, respectively; $f_{aim}(ind+1)$ and $f_{aim}(ind-1)$ are the neighboring values of individual ind in objective aim. $D(ind)$ is the crowding distance of individual ind, which is the sum of the distances in all objectives. Define the letter M as the total number of objective functions. The crowding distance of the boundary points in each front is set to infinity.

(8) Vortex Formation and Fish Aggregating Device (FADs) Effect: Another factor that influences marine predator behavior is environmental changes, such as vortex formation and the FADs effect. These effects alter the predator's strategy, helping the algorithm effectively escape local optima. According to the study in reference [26], sharks spend more than 80% of their time near FADs, while the remaining 20% is spent making longer jumps across different dimensions, likely to search for another environment where prey is distributed. FADs are considered local optima, acting as traps within the search space. By considering these longer jumps during the simulation process, the algorithm can avoid stagnation in local optima. The mathematical expression of the FADs effect is shown in Equation (31):

$$\overrightarrow{Prey_i}=\begin{cases} \overrightarrow{Prey_i}+CF\left[\vec{X}_{\min}+\vec{R}\otimes\left(\vec{X}_{\max}-\vec{X}_{\min}\right)\right]\otimes\vec{U}, & if\ rd\le \text{FADs} \\ \overrightarrow{Prey_i}+[\text{FADs}(1-rd)+rd]\left(\overrightarrow{Prey_{r1}}-\overrightarrow{Prey_{r2}}\right), & if\ rd>\text{FADs} \end{cases} \tag{31}$$

where, FADs=0.2 is the probability of the FADs effect influencing the optimization process. $\vec{U}$ is a binary vector whose array contains 0s and 1s. This is constructed by generating a random vector within [0,1], with FADs=0.2, and rd is a uniformly distributed random number in [0,1]. $X_{\max}$ and $X_{\min}$ are vectors containing the lower and upper bounds of the dimensions. The subscripts r1 and r2 represent random indices of the prey matrix.

(9) Dynamic Local Search Strategy: During the optimization process, the MPA tends to overly concentrate search activities around local optimal regions. While this behavior aids in thoroughly exploring advantageous areas, it also increases the risk of being trapped in a local optimum, making it difficult to escape. In MPA, predators often operate near FADs, which helps in finding prey but overlooks broader potential regions. When applied to optimization problems, MPA mimics the tracking strategy of top marine predators. Although these strategies can effectively and quickly identify and exploit local environmental advantages, they also carry the risk of the algorithm prematurely focusing on local optima while ignoring other potentially better solutions. This issue is particularly evident in complex optimization applications. Moreover, the search capability of the MPA diminishes during the later stages of the optimization process. In the initial stages of the algorithm, Lévy and Brownian motions provide large step sizes, facilitating broad search activities. However, as the number of iterations increases, the step sizes gradually decrease, reducing the search range and exploration capability, potentially leading to slower convergence or stagnation, thus hindering further improvement in solution quality. Therefore, this section introduces a radial transformation matrix and a dynamic search strategy, allowing the algorithm to seek local optima while improving its robustness. The specific method is outlined as follows.

An affine transformation-like method[27] is employed to guide the particle swarm's search behavior, constructing a bio-inspired transformation matrix.

$$\begin{cases} Pray\rightarrow C\otimes Pray+\bar{C}\otimes Pray_B \\ Pray_B=\text{Update } Pray \text{ using Equations (25)-(27) and (29)} \end{cases} \tag{32}$$

where, the operation $\otimes$ represents the random combination of corresponding elements from two matrices in the formula. $Pray$ represents the current position of the particle swarm, while C and $\bar{C}$ represent the transformation matrix and translation vector in the affine transformation, respectively. C, as a cooperative search matrix, not only guides cooperation and information sharing among individuals, but also introduces randomness and diversity into the search process through $\bar{C}$. $\bar{C}$ is the inverse binary operation of the elements in matrix C. If all elements are 0, the inverse element is set to 1, significantly enhancing the algorithm's global search capability. Matrix C starts as a lower triangular matrix with all elements equal to 1 and is cleverly transformed into matrix C through two consecutive operations, as shown in Equation (33):

$$C_{inf}=\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}\sim\begin{bmatrix} 0 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 0 \end{bmatrix}=C \tag{33}$$

This transformation method not only embodies the essence of traditional affine transformations but also incorporates random elements, ensuring that each individual's search path is unique and innovative.

By utilizing the precision and efficiency of the dynamic search strategy in global search, and combining it with the FADs effect, the aim is to improve the algorithm's overall performance in handling complex optimization and scheduling problems. This approach effectively balances exploration and exploitation, enabling the algorithm to seek local optima while retaining the ability to escape these regions and explore broader search spaces. This design not only enhances the flexibility and adaptability of the algorithm in multidimensional search spaces, but also improves the overall robustness of the algorithm when dealing with diverse and complex problems. The strategy is shown in Equation (34):

$$\begin{cases} M=Pray_{\text{gbest}}+t*(Pray_{r1}-Pray_{r2}) \\ Pray\rightarrow C\otimes Pray+\bar{C}\otimes M \end{cases} \tag{34}$$

where, $Pray_i$ is considered an entity that explores the multidimensional search space. The positions of these particles form a coordinate matrix $Pray$, providing a global representation of the algorithm's search space. This strategy not only allows each particle to retain the memory of its own individual best solution $Pray_{gbest}$, but also optimizes the search through collective intelligence guided by the global best solution $Pray_{gbest}$. The global best solution is determined by evaluating the performance of all particles and selecting the optimal solution from them, which provides a common objective for the entire population. t is the coefficient factor (step size) of the differential matrix, controlling the step size for particle position updates during each iteration. Specifically, the differential matrix is used through $Pray_{r1}$ and $Pray_{r2}$ to dynamically adjust the update direction for each particle. $Pray_{r1}$-$Pray_{r2}$ not only provides the update direction for each particle but also adds randomness and diversity to the algorithm.

(10) Ocean Memory: Marine predators have excellent memory, allowing them to recall locations where they successfully foraged. This ability is simulated by memory savings. After updating Prey and implementing the FADs effect, this matrix is evaluated for fitness to update the Elite. The fitness of each solution in the current iteration is compared with the corresponding solution from the previous iteration, and if the current solution is better, it replaces the previous solution. This process improves the quality of solutions as the iterations progress[28], simulating the predator's return to prey-rich areas where it successfully foraged. This step involves calculating the fitness value of each individual in the population, and if the fitness is better than the corresponding position in the Elite matrix, that individual is updated as an Elite, enhancing the adaptability of the population.

(11) Iterative Evolution: The algorithm checks whether the specified number of iterations has been reached. If the condition is satisfied, the algorithm terminates and decodes the optimal result. If the iteration limit has not been reached, the algorithm continues iterating. The pseudocode for the IMPA is shown in Table 2.

Table 2 Pseudocode of IMPA

Input: Initialize the population based on Equations (14) and (15).
% Improving Adversarial Learning Strategies
while termination criteria have not been satisfied **do**
  Compute fitness, build Elite matrix, and achieve memory optimization
  Generate cooperative search matrix C, shown in Equation (34)
  % Matrix transformation update strategy
  Execute the pseudo-code program in Table 1.
  End if
  Memory protection and elite upgrades based on Equation (32)
  % Dynamic search strategy
  Apply dynamic update strategy based on Equation (34)
  for i=1 **to** populations **do**
    if $f(Prey_i)$ **optimal than** $f(Prey_{pbest,i})$ **then**
    $Prey_{pbest,i} \leftarrow Pray_i$
    end if
  end for
  $Prey_i = Prey_{pbest,i}$
  $Prey_{gbest} = opt(Prey_{pbest})$
end while
Output

4 Experiment

4.1 Experimental data

In Longxing Village, Huaziku District, Huagai Town, Mianyang City, Sichuan Province, data from 20 fields were selected. The light green shapes represent the field boundaries, the green markers indicate the locations of the fields, and the numbers 1-20 correspond to the field IDs. The red marker represents the agricultural machinery depot, with the letter "A" denoting the depot's ID. These data were obtained using the LocaSpace Viewer software (version 4.4.8.0), which integrates advanced technologies such as GPS and GIS, as shown in Figure 5. To validate the subsequent analysis, the experiment was conducted using the MATLAB 2022b platform, running on a computer equipped with an i5-8250U CPU (1.60GHz) and 8GB of memory, with Windows 10 as the operating system. For the algorithm section, the parameter settings of the IMPA are detailed in Table 3.

Figure 5 Experimental area

Table 3 Algorithm parameter settings

Algorithm parameter setting	
$Max\_Iter \leftarrow 50$	% Number of iterations
Popsize ← 20	% Population size
$ub \leftarrow 20$	% Upper bound ← Number is farmland
$lb \leftarrow 1$	% Lower bound
$W \leftarrow$ [1 0.5 0.1 0.01]	% Vector of weight coefficients
FADs ← 0.2	% Fish Aggregating Devices (FADs) effect

Taking corn harvesting and transportation as an example, this experiment requires the coordinated operation of harvesters and grain transport vehicles to complete the harvesting and transportation tasks for 20 fields. The experimental parameters are shown in Table 4, and the basic information of the harvesters and grain transport vehicles is provided in Table 5, including model, capacity, power, and maximum usage quantity.

Table 4 Experimental parameter settings

Experimental parameter setting	
$V_h \leftarrow 10$	% Harvester travel speed
$V_g \leftarrow 10$	% Grain transport vehicle travel speed
$H_v \leftarrow 1000$	% Harvester capacity
$G_v \leftarrow$ [3000,4000]	% Capacity of different types of grain transport vehicle
$U_{eh} \leftarrow 1600$	% Harvester unit unloading capacity
$g_n \leftarrow 10$	% Harvest per unit area
$H_c \leftarrow 600$	% Harvester transfer, maintenance, and fuel costs
$G_c \leftarrow$ [800,1000]	% Transfers, maintenance, and fuel costs for grain transport vehicle
$d_c \leftarrow 1$	% Cost per unit distance traveled
$C_W \leftarrow 10$	% Cost of waiting per unit of time
$C_1 \leftarrow 10$	% Unit time penalty cost
$O_w \leftarrow 5$	% Harvester operating width
$d_f \leftarrow 3$	% Turning distance factor

Table 5 Basic information of agricultural machinery

Name	Volume/m³	Model	Watt/kW	Maximum usage
Henan Songke 4YD-2A corn harvester	1000	4YB-2	14.71	6
Jinan VT-YZ3000 crawler grain transport vehicle	3000	VT-YZ3000	17.6	3
Jinan Jinwang JW-4t crawler grain transport vehicle	4000	JW-4t	38	3

The data required to validate the algorithm are based on real field data and field operation parameters. As shown in Table 6, the data mainly include the coordinates of the agricultural machinery depot, the latitude and longitude coordinates of the field entrances and exits, the field area, the harvestable volume, and the time windows.

4.2 Experimental results

In recent years, multi-objective optimization algorithms have achieved significant success in addressing complex scheduling problems. Notably, the NSGA-II and the MOHHO have proven effective in solving complex optimization problems due to their excellent solution quality and efficiency. Therefore, this paper uses the NSGA-II from reference [29] and the MOHHO from reference [30] as comparison benchmarks for the experiment. Gantt charts and route maps for the three algorithms are plotted, and the iterative curves and results for cost and time are compared to validate the effectiveness and superiority of the proposed algorithm. The Gantt charts for the IMPA algorithm, the algorithm in reference [29], and the algorithm in reference [30], corresponding to the harvester and grain transport vehicles, are shown in Figures 6, 8, and 10, respectively. The route maps for the harvester and grain transport vehicles are shown in Figures 7, 9, and 11, respectively, where the letters "en" indicates the entrance, "ex" indicates the exit, numbers 1-20 correspond to field IDs, and letter "A" represents the agricultural machinery depot.

Table 6 Basic information of fields

No.	Entrance		Exit		Area/m²	Demand/m³	Time window
	Longitude	Latitude	Longitude	Latitude			
A	104°33′23.11″	31°33′17.61″	/	/	/	/	[0,1000]
1	104°33′10.79″	31°33′21.57″	104°33′09.67″	31°33′22.48″	842	1684	[50,380]
2	104°33′12.93″	31°33′20.65″	104°33′13.14″	31°33′21.10″	960	1920	[109,239]
3	104°33′14.01″	31°33′19.98″	104°33′14.01″	31°33′20.78″	613	1226	[60,771]
4	104°33′15.73″	31°33′19.21″	104°33′16.67″	31°33′19.75″	381	762	[141,571]
5	104°33′11.60″	31°33′19.54″	104°33′11.53″	31°33′21.01″	273	546	[41,271]
6	104°33′12.36″	31°33′18.98″	104°33′13.40″	31°33′19.73″	645	1290	[95,625]
7	104°33′15.86″	31°33′17.72″	104°33′16.32″	31°33′18.66″	506	1012	[91,321]
8	104°33′09.09″	31°33′19.16″	104°33′09.34″	31°33′20.64″	1562	3124	[21,521]
9	104°33′14.24″	31°33′17.71″	104°33′15.19″	31°33′18.40″	280	560	[119,749]
10	104°33′09.96″	31°33′17.48″	104°33′10.61″	31°33′18.82″	659	1318	[59,389]
11	104°33′12.80″	31°33′16.15″	104°33′14.53″	31°33′17.14″	1094	2188	[64,294]
12	104°33′14.56″	31°33′15.45″	104°33′15.34″	31°33′16.76″	966	1932	[142,772]
13	104°33′06.69″	31°33′17.29″	104°33′07.49″	31°33′18.37″	988	1976	[35,655]
14	104°33′07.19″	31°33′13.36″	104°33′12.79″	31°33′15.93″	1441	2882	[58,888]
15	104°33′05.74″	31°33′16.11″	104°33′06.80″	31°33′17.10″	605	1210	[72,702]
16	104°33′09.37″	31°33′15.57″	104°33′09.80″	31°33′16.68″	513	1026	[149,979]
17	104°33′10.62″	31°33′14.52″	104°33′11.19″	31°33′15.54″	563	1126	[388,911]
18	104°33′09.31″	31°33′14.65″	104°33′07.77″	31°33′15.47″	815	1630	[30,546]
19	104°33′11.32″	31°33′13.71″	104°33′11.27″	31°33′14.24″	340	680	[353,708]
20	104°33′13.12″	31°33′12.77″	104°33′14.43″	31°33′13.74″	886	1772	[425,913]

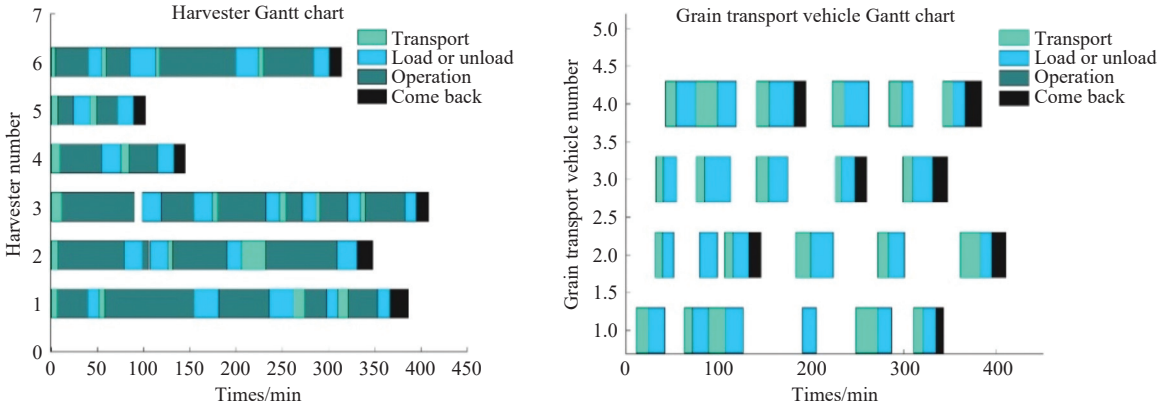

Figure 6 Gantt chart of the IMPA algorithm

The route plans for harvesters and grain transport vehicles corresponding to the three algorithms are listed in Table 7. In the table, the letter "h" represents the harvester, and "g" represents the grain transport vehicle. The numeric suffixes indicate the machine's ID number, while the values in parentheses, such as 3000 and 4000, represent grain transport vehicles with capacities of 3000 and 4000, respectively.

To analyze the performance of the three algorithms in terms of

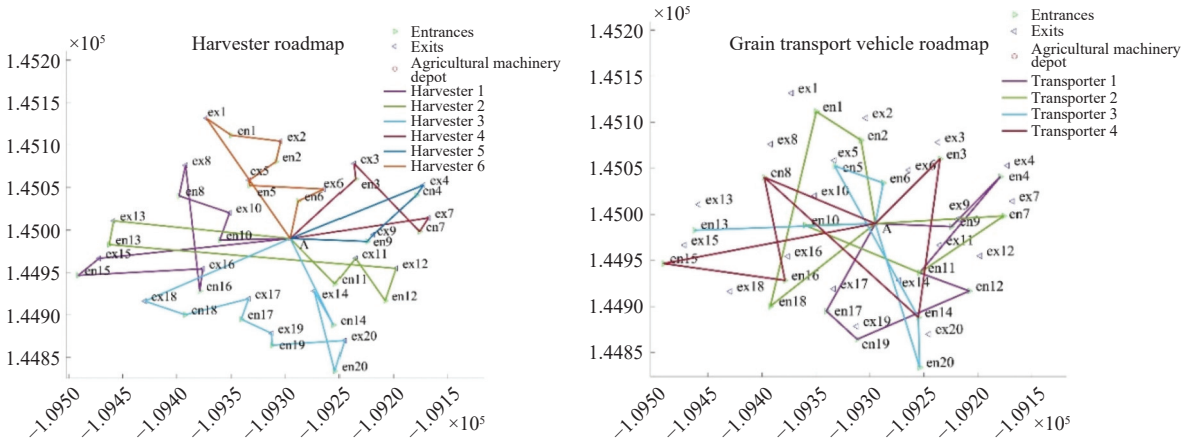

Figure 7 Route map of the IMPA algorithm

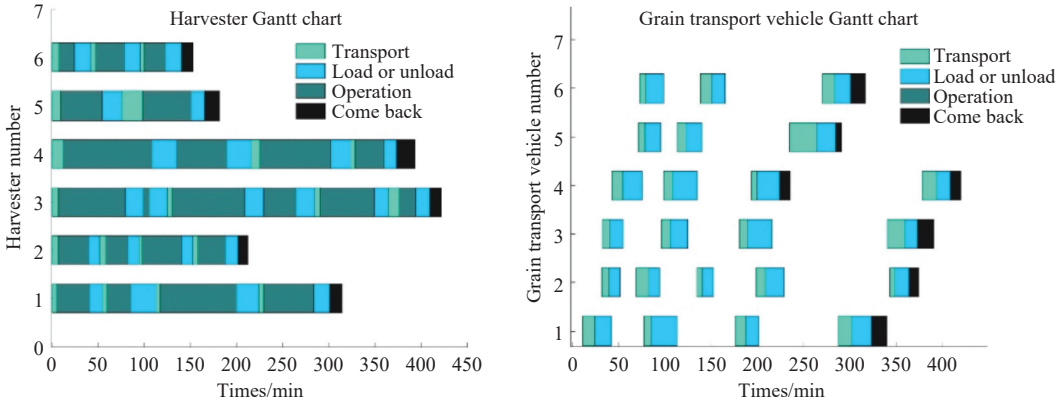

Figure 8 Gantt chart of the algorithm in reference [29]

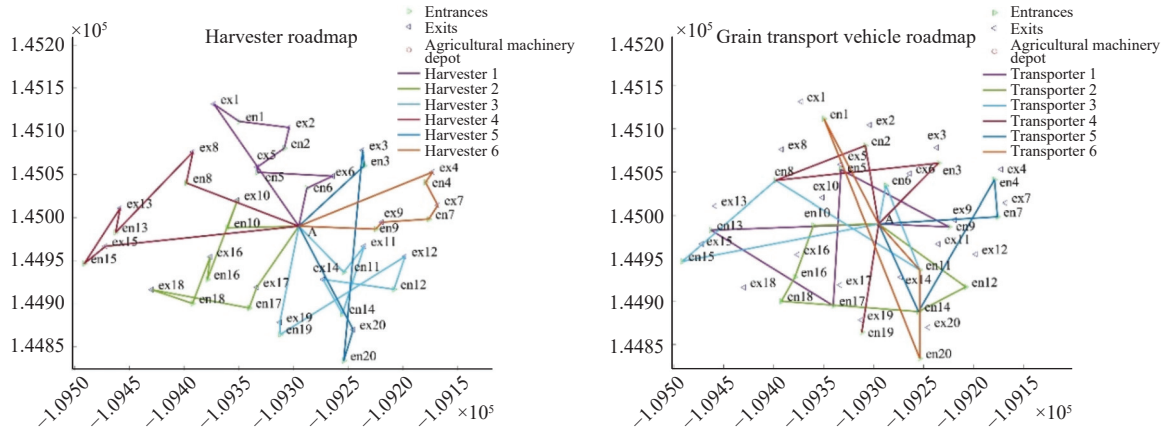

Figure 9 Route map of the algorithm in reference [29]

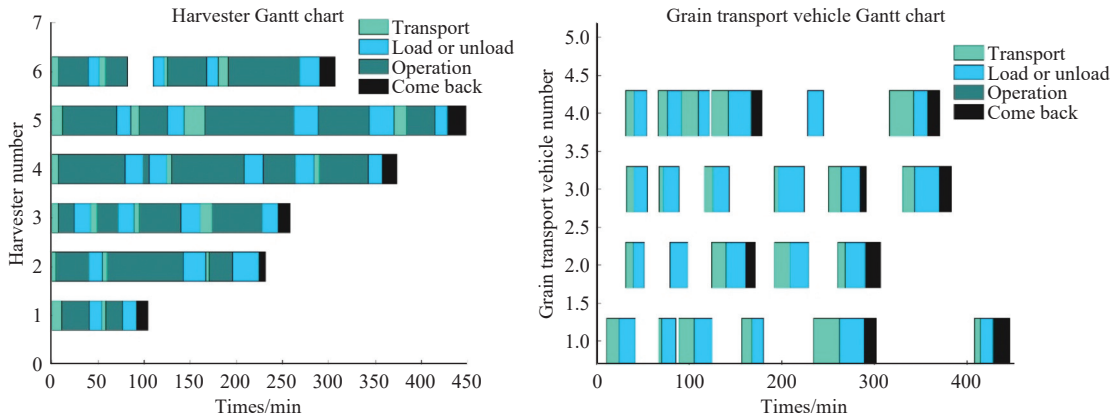

Figure 10 Gantt chart of the algorithm in reference [30]

cost and time, 50 iterations of experiments were conducted for each algorithm. The comparison of the iterative curves is shown in Figure 12, and the data comparison is presented in Table 8.

As analyzed in Figure 12, compared to reference [29] and [30], the IMPA achieves better solutions in both scheduling cost and scheduling time, demonstrating its superiority in solution accuracy and optimization capability. The final iteration data in Table 7 indicate that, compared to reference [29] and [30], the scheduling cost of the IMPA is reduced by 7% and 2.9%, respectively, while the scheduling time is reduced by 2.6% and 8.3%, respectively, with the shortest solution time. Therefore, the IMPA proposed in this paper effectively solves the multi-machine cooperative scheduling problem for harvesting and transportation operations and provides high-quality solutions. This translation maintains the formal academic tone and accurately conveys the comparative analysis of the algorithms.

Table 7 Comparison of route plans

Source	Harvester routes	Grain transport vehicle routes
IMPA algorithm	h_1:A->10->8->16->15->A h_2: A->11->12->13->A h_3:A->14->20->19->17->18->A h_4:A->3->7->A h_5:A->9->4->A h_6:A->6->5->2->1->A	g_1(4000):A->9->4->11->12->19->17->A g_2(3000):A->10->11->7->0->2->1->18->A g_3(3000):A->6->5->14->20->0->13->A g_4(3000):A->3->14->8->0->8->16->15->A
Literature [29]	h_1:A->6->5->2->1->A h_2:A->10->16->18->17->A h_3:A->11->14->12->19->A h_4:A->8->13->15->A h_5:A->3->20->A h_6:A->9->7->4->A	g_1(4000):A->9->5->17->13->A g_2(4000):A->10->16->18->14->12->A g_3(4000):A->6->11->8->15->A g_4(3000):A->3->8->2->0->19->A g_5(3000):A->7->4->14->A g_6(4000):A->11->20->1->A
Literature [30]	h_1:A->17->19->A h_2:A->6->2->5->A h_3:A->9->4->3->1->A h_4:A->11->14->20->A h_5:A->12->7->8->15->A h_6:A->10->16->18->13->A	g_1(4000):A->9->12->11->18->8->0->15->A g_2(3000):A->10->11->3->0->14->13->A g_3(3000):A->6->4->7->5->14->0->8->A g_4(3000):A->17->19->16->2->0->1->20->A

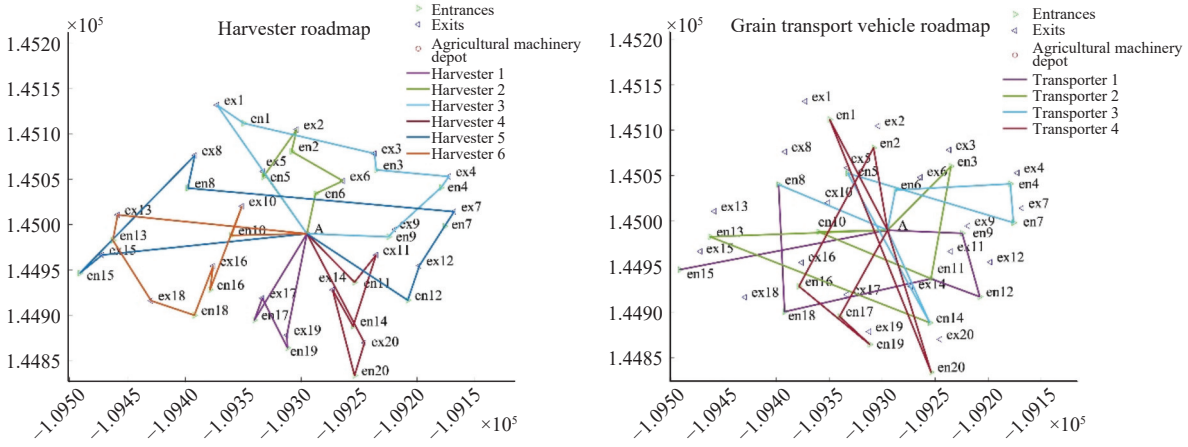

Figure 11 Route map of the algorithm in reference [30]

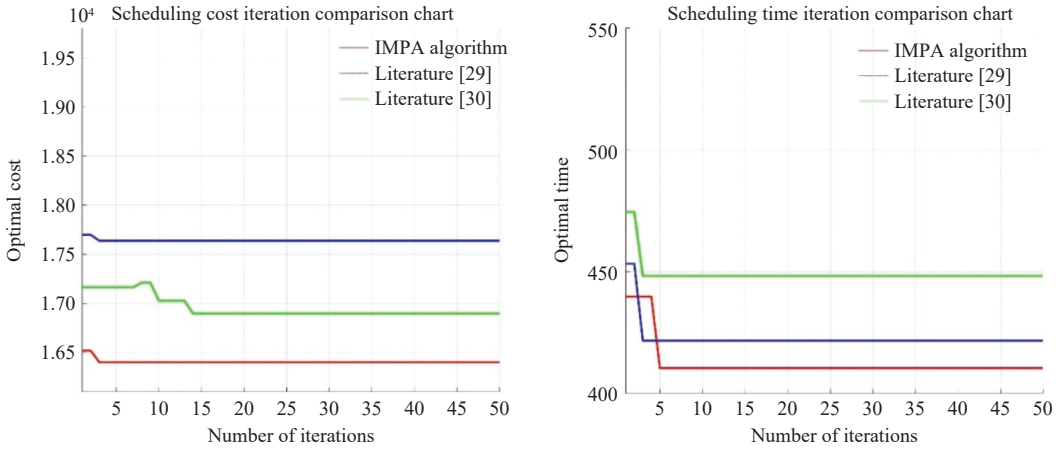

Figure 12 Iterative comparison of cost and time for the three algorithms

Table 8 Comparison of cost and time

Source	Scheduling costs/CNY	Scheduling time/min	Period/s
IMPA algorithm	16 401	411	7.1055
Reference [29]	17 637	422	8.2489
Reference [30]	16 896	448	7.9686

5 Statistical analysis of the IMPA

To comprehensively evaluate the performance of the IMPA proposed in this paper, 50 independent experiments were conducted for each of the three algorithms. The experimental results for the IMPA and for references [29, 30] are listed in Tables 9-11, respectively. The data comparison and analysis are illustrated in Figure 13.

Fifty experiments were conducted, and the results are shown in the box plot[31,32] in Figure 13. In terms of scheduling cost, compared to references [29, 30], the IMPA has fewer outliers[33], lower fluctuations, and a smaller interquartile range (IQR)[34], making the cost data significantly better than those of references [29, 30]. Regarding scheduling time, references [29, 30] exhibit larger IQRs,

Table 9 Results of 50 experiments with IMPA

No.	Scheduling costs/CNY	Scheduling time/min	No	Scheduling costs/CNY	Scheduling time/min	No	Scheduling costs/CNY	Scheduling time/min
1	16 401	411	18	16 433	413	35	16 348	411
2	16 407	404	19	16 530	402	36	16 426	412
3	16 713	403	20	16 565	404	37	16 563	403
4	16 331	415	21	16 720	413	38	16 519	412
5	16 740	404	22	16 695	407	39	16 315	406
6	16 323	408	23	16 724	411	40	16 308	410
7	16 313	412	24	16 308	408	41	16 760	413
8	16 755	405	25	16 483	415	42	16 448	408
9	16 249	405	26	16 349	404	43	16 247	403
10	16 478	402	27	16 526	415	44	16 530	407
11	16 729	405	28	16 614	411	45	16 508	405
12	16 769	405	29	16 602	409	46	16 777	403
13	16 803	410	30	16 331	413	47	16 660	412
14	16 304	405	31	16 441	405	48	16 233	413
15	16 418	408	32	16 498	406	49	16 202	412
16	16 204	407	33	16 360	413	50	16 697	413
17	16 380	411	34	16 468	412			

Table 10 Results of 50 experiments from reference [29]

No.	Scheduling costs/CNY	Scheduling time/min	No	Scheduling costs/CNY	Scheduling time/min	No	Scheduling costs/CNY	Scheduling time/min
1	17 637	422	18	17 804	429	35	17 479	416
2	17 727	429	19	17 303	418	36	17 805	420
3	17 810	426	20	17 283	416	37	17 810	426
4	17 826	429	21	17 618	422	38	17 712	418
5	17 643	418	22	17 720	422	39	17 633	424
6	17 716	420	23	17 498	423	40	17 573	431
7	17 324	430	24	17 828	417	41	17 629	420
8	17 281	425	25	17 572	425	42	17 699	428
9	17 547	428	26	17 690	416	43	17 375	431
10	17 421	422	27	17 387	421	44	17 580	420
11	17 535	420	28	17 640	417	45	17 304	431
12	17 748	426	29	17 369	425	46	17 320	417
13	17 298	426	30	17 397	428	47	17 300	430
14	17 342	416	31	17 777	430	48	17 445	422
15	17 458	432	32	17 694	416	49	17 336	423
16	17 585	417	33	17 259	426	50	17 532	423
17	17 325	415	34	17 471	431			

Table 11 Results of 50 experiments from reference [30]

No.	Scheduling costs/CNY	Scheduling time/min	No	Scheduling costs/CNY	Scheduling time/min	No	Scheduling costs/CNY	Scheduling time/min
1	16 896	448	18	16 552	449	35	16 924	451
2	16 926	453	19	17 133	449	36	16 684	436
3	16 878	449	20	16 894	446	37	17 197	445
4	16 798	445	21	16 700	437	38	17 045	436
5	16 543	452	22	17 078	450	39	16 571	444
6	16 848	447	23	16 761	445	40	16 785	445
7	16 962	446	24	16 603	436	41	16 794	450
8	17 177	450	25	17 065	436	42	16 649	439
9	16 980	446	26	16 741	454	43	16 903	447
10	16 939	441	27	17 014	438	44	16 600	439
11	16 740	434	28	17 211	451	45	17 098	449
12	16 664	433	29	16 817	451	46	16 688	435
13	16 942	434	30	16 797	448	47	16 827	451
14	17 104	437	31	17 068	443	48	16 550	453
15	16 545	441	32	17 210	435	49	17 062	441
16	16 673	445	33	16 589	446	50	16 955	442
17	17 177	453	34	16 891	441			

greater fluctuations, and more outliers than the algorithm proposed in this paper. In both scheduling cost and time, the IMPA has the lowest average values. Furthermore, as shown in Tables 12 and 13, the Kruskal-Wallis ANOVA test results show *p*-values less than 0.05, indicating significant differences between the groups. In summary, the IMPA in this paper demonstrates superior and more stable performance in terms of scheduling cost and time.

Table 12 Kruskal-Wallis ANOVA table for scheduling costs

	SS	*df*	*MS*	*F*	*p*
Intergroup	230 307.6	2	115 153.8	122.02	3.19156e-27
Intra-group	50 926.9	147	346.4		
Total	281 234.5	149			

Table 13 Kruskal-Wallis ANOVA table for scheduling time

	SS	*df*	*MS*	*F*	*p*
Intergroup	249 850.1	2	124 925	132.47	1.7157e-29
Intra-group	31 176.4	147	212.1		
Total	281 026.5	149			

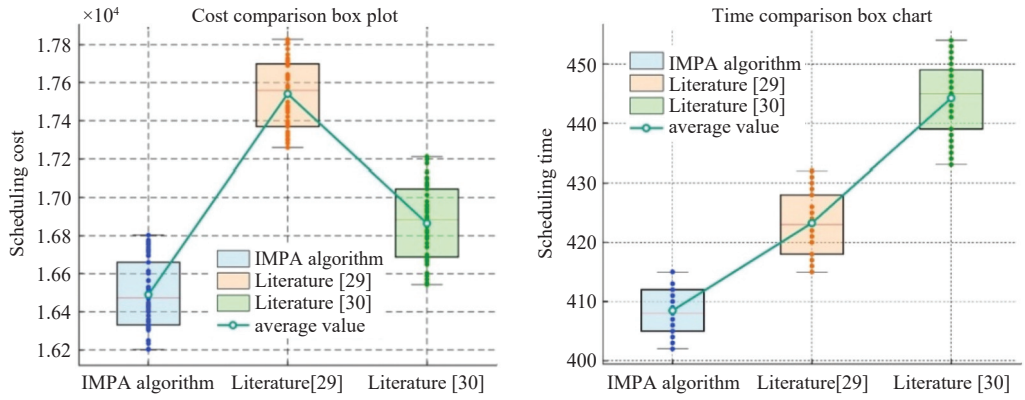

Figure 13 Box plot comparison

6 Conclusions

This paper studied the complex cooperative scheduling problem of harvesters and grain transport vehicles with different capacities under time window constraints. First, in response to the issues of poor timeliness, low coordination efficiency, and high scheduling costs in multi-field harvesting and transportation tasks, a cooperative scheduling model was established with the objectives of

minimizing scheduling costs and time. An IMPA was then designed to solve the model. Finally, compared to scheduling schemes from other studies, this research found a more optimal cooperative scheduling solution, improving the efficiency and timeliness of the coordinated operations between harvesters and grain transport vehicles. The main conclusions of this study are as follows:

(1) To address the issue of low response efficiency between harvesters and grain transport vehicles in complex operational environments, a task allocation method and response scheme based on a priority strategy were designed. A cooperative scheduling model for harvesters and grain transport vehicles was constructed, considering the complex influencing factors in the coordination process, with the goal of minimizing scheduling costs and time to complete the harvesting and transportation tasks across multiple fields.

(2) An IMPA was designed to solve the cooperative scheduling model. By introducing a radial transformation matrix, dynamic search, and an enhanced opposition-based learning strategy, the optimization capability of the overall algorithm was improved. Non-dominated sorting and crowding distance calculations were incorporated to optimize the solution set, enhancing solution quality and diversity. Experimental results showed that scheduling costs were reduced by 7% and 2.9%, and scheduling times were reduced by 2.6% and 8.3%, respectively, compared to other algorithms in the literature. Finally, statistical analysis confirmed that the IMPA demonstrated superior optimization capability and effectiveness.

(3) While this study has demonstrated the effectiveness of the IMPA in solving the cooperative scheduling problem, several areas warrant further investigation. The current model assumes ideal operational conditions and does not account for uncertainties such as weather variations, mechanical breakdowns, and dynamic field condition changes. Future research could incorporate stochastic optimization techniques to address these real-world uncertainties and enhance model robustness. Additionally, integrating real-time monitoring technologies such as IoT sensors and GPS tracking would enable dynamic rescheduling capabilities during operations. Finally, extending the model to consider additional objectives such as fuel consumption minimization and carbon emission reduction would further enhance its practical value for sustainable precision agriculture.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 32501805), the Sichuan Provincial Natural Science Foundation (Grant No. 2025ZNSFSC1004), and the Sichuan Provincial Science and Technology Program (Grant No. 2025NZZ10009).

[References]

[1] Wang N, Li S D, Xiao J X, Wang T H, Han Y X, Wang H, et al. A collaborative scheduling and planning method for multiple machines in harvesting and transportation operations-Part Ⅰ: Harvester task allocation and sequence optimization. Computers and Electronics in Agriculture. 2025; 232: 110060. DOI: 10.1016/j.compag.2025.110060

[2] Guo Y Q, Zhang F, Chang S H, Li Z, Li Z K. Research on a multiobjective cooperative operation scheduling method for agricultural machinery across regions with time windows. Computers and Electronics in Agriculture, 2024; 224: 109121.

[3] Wang N, Li S D, Xiao J X, Wang T H, Han Y X, Wang H, et al. A collaborative scheduling and planning method for multiple machines in harvesting and transportation operations-part Ⅱ: Scheduling and planning of harvesters and grain trucks. Computers and Electronics in Agriculture. 2025; 235: 110344. DOI: 10.1016/j.compag.2025.110344

[4] Liu H Y, Zhang L H, Zhao B D, Tang J C, Wang F L, Wang S. Research on agricultural machine scheduling in hilly areas based on improved non-dominated sorting genetic algorithm-III. IEEE Access. 2024; 12: 32584–32596. DOI: 10.1109/ACCESS.2024.3371176

[5] Zhu S J, Wang B, Pan S Q, Ye Y T, Wang E G, Mao H P. Task allocation of multi-machine collaborative operation for agricultural machinery based on the improved fireworks algorithm. Agronomy, 2024; 14(4): 710.

[6] He P F, Li J. The two-echelon multi-trip vehicle routing problem with dynamic satellites for crop harvesting and transportation. Applied Soft Computing, 2019; 77: 387–398.

[7] Zhang D Z, Zhou S Q, Ji B, Li S Y. A two-echelon capacitated vehicle routing problem with sharing satellite resources. IEEE Transactions on Intelligent Transportation Systems, 2024; 25(9): 12216–12227.

[8] Zhao L S, Liu G S, Lu Z L, Xiao Y, Nie J Q, Yang L Y, et al. A new framework for delineating farmland consolidation priority areas for promoting agricultural mechanization in hilly and mountainous areas. Computers and Electronics in Agriculture, 2024; 218: 108681.

[9] Liu X Y, Zhu X M, Hao K R. Dynamic immune cooperative scheduling of agricultural machineries. Complex & Intelligent Systems, 2021; 7(6): 2871–2884.

[10] Liu H Y, Luo J H, Yu H, Tang J C, Wang F L, Wang S. Intelligent cooperative scheduling and path planning for tracked maize harvesters and grain trucks using an enhanced hybrid MOEA/D-LSA algorithm. Computers and Electronics in Agriculture, 2025; 239: 110952.

[11] Liu H Y, Luo J H, Zhang L H, Yu H, Liu X N, Wang S. Research on traversal path planning and collaborative scheduling for corn harvesting and transportation in hilly areas based on Dijkstra's algorithm and improved Harris Hawk optimization. Agriculture, 2025; 15(3): 233.

[12] Liu H Y, Luo J H, Zhang L H, Wang F L, Wang S. Optimal scheduling of agricultural machines in hilly mountainous areas based on NSGA-Ⅱ-SA hybrid algorithm with applications. Int J Agric & Biol Eng, 2025; 18(5): 234–245.

[13] Liu H Y, Zhang L H, Zhao B D, Tang J C, Luo J H, Wang S. Research on emergency scheduling based on improved genetic algorithm in harvester failure scenarios. Frontiers in Plant Science, 2024; 15: 1413595.

[14] Zhao B D, Zheng D K, Yang C H, Wang S, Mansurova M, Jomartova S, et al. Design and optimization of an internet of things-based cloud platform for autonomous agricultural machinery using narrowband internet of things and 5G dual-channel communication. Electronics, 2025; 14(8): 1672.

[15] Khalid Q S, Azim S, Abas M, Babar A R, Ahmad I. Modified particle swarm algorithm for scheduling agricultural products. Engineering Science and Technology, 2021; 24(3): 818–828.

[16] Li S C, Zhang M, Wang N, Cao R Y, Zhang Z Q, Ji Y H, et al. Intelligent scheduling method for multi-machine cooperative operation based on NSGA-III and improved ant colony algorithm. Computers and Electronics in Agriculture, 2023; 204: 107532.

[17] Rodias E C, Sopegno A, Berruto R, Bochtis D D, Cavallo E, Busato P. A combined simulation and linear programming method for scheduling organic fertiliser application. Biosystems Engineering. 2019; 178: 233–243. DOI: 10.1016/j.biosystemseng.2018.11.002

[18] He P F, Li J, Wang X. Wheat harvest schedule model for agricultural machinery cooperatives considering fragmental farmlands. Computers and Electronics in Agriculture, 2018; 145: 226–234.

[19] Cao R Y, Guo Y A, Zhang Z Q, Li S C, Zhang M, Li H, et al. Global path conflict detection algorithm of multiple agricultural machinery cooperation based on topographic map and time window. Computers and Electronics in Agriculture, 2023; 208: 107773.

[20] Worasan K, Sethanan K, Pitakaso R, Moonsri K, Nitisiri K. Hybrid particle swarm optimization and neighborhood strategy search for scheduling machines and equipment and routing of tractors in sugarcane field preparation. Computers and Electronics in Agriculture, 2020; 178: 105733.

[21] Ding C B, Wang L M, Chen X Y, Yang H T, Huang L X, Song X M. A blockchain-based wide-area agricultural machinery resource scheduling system. Applied Engineering in Agriculture, 2023; 39(1): 1–2.

[22] Borodin V, Bourtembourg J, Hnaien F, Labadie N. A quality risk management problem: case of annual crop harvest scheduling. International Journal of Production Research, 2014; 52(9): 2682–2695.

[23] Foulds L R, Wilson J M. Scheduling operations for the harvesting of renewable resources. Journal of Food Engineering, 2005; 70(3): 281–292.

[24] Pakawanich P, Udomsakdigool A, Khompatraporn C. Crop production scheduling for revenue inequality reduction among smallholder farmers in

an agricultural cooperative. Journal of the Operational Research Society, 2022; 73(12): 2614–2625.

[25] Ma L, Wang Y D, Ma M Q, Bai J Y. Research on multi-cooperative combine-integrated scheduling based on improved NSGA-II algorithm. International Journal of Agricultural and Environmental Information Systems (IJAEIS), 2021; 12(4): 1–21.

[26] Tan B, Çömden N. Agricultural planning of annual plants under demand, maturation, harvest, and yield risk. European Journal of Operational Research, 2012; 220(2): 539–549.

[27] Bochtis D D, Sørensen C G, Busato P. Advances in agricultural machinery management: A review. Biosystems Engineering, 2014; 126: 69–81.

[28] Ospina R, Noguchi N. Real-time work progress estimation based on GIS remote monitoring system for agricultural robot vehicles. Computers and Electronics in Agriculture, 2025; 234: 110313.

[29] Han Y L, Shao M, Wu Y Z, Zhang X M. An improved complete coverage path planning method for intelligent agricultural machinery based on backtracking method. Information, 2022; 13(7): 313.

[30] Zhang J, Li D. Research on path tracking algorithm of green agricultural machinery for sustainable development. Sustainable Energy Technologies and Assessments, 2023; 55: 102917.

[31] Oksanen T, Visala A. Coverage path planning algorithms for agricultural field machines. Journal of Field Robotics, 2009; 26(8): 651–668.

[32] Martínez-Vargas A, Domínguez-Guerrero J, Andrade A G, Sepúlveda R, Montiel-Ross O. Application of NSGA-II algorithm to the spectrum assignment problem in spectrum sharing networks. Applied Soft Computing, 2016; 39: 188–198.

[33] Yuan M H, Li Y D, Zhang L Z, Pei F Q. Research on intelligent workshop resource scheduling method based on improved NSGA-II algorithm. Robotics and Computer-Integrated Manufacturing, 2021; 71: 102141.

[34] Khoshahval F, Zolfaghari A, Minuchehr H, Abbasi M R. A new hybrid method for multi-objective fuel management optimization using parallel PSO-SA. Progress in Nuclear Energy, 2014; 76: 112–121.