

Research Article

An Optimization Method for Enterprise Resource Integration Based on Improved Particle Swarm Optimization

Aifang Guo , Lina Zhu, and Lingjie Chang

Department of Economy and Trade, Changzhi Vocational and Technical College, Changzhi 046000, China

Correspondence should be addressed to Aifang Guo; fenling123@163.com

Received 31 March 2022; Revised 26 April 2022; Accepted 11 May 2022; Published 31 May 2022

Academic Editor: Huihua Chen

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

An enterprise's development and growth are inextricably linked to rational and efficient resource integration and optimization. This study focuses on the reorganization and integration of industrial elements inside the firm from the standpoint of resource integration. The ideal resource integration strategy is investigated by integrating the industrial parts of a certain enterprise in order to increase the efficiency of project completion and lower enterprise expenses. The enterprise's internal material and human resources are limited, but it is frequently necessary to execute numerous activities at the same time, and each activity must meet multiple goals. This research investigates how to properly integrate and schedule resources while attaining different goals. This research proposes using an enhanced particle swarm optimization technique (IPSO) to combine firms' internal resources. In order to address the issue of uneven particle dispersion caused by random population initialization, IPSO incorporates chaos theory into particle population initialization. The logistic mapping sequence generates a huge number of particles, and the particles with the highest quality are chosen for initialization. This can increase particle quality, allowing particles to be spread equally during setup. In the late stage, the classic particle swarm optimization algorithm (PSO) has a slow convergence rate, causing the algorithm to readily slip into a local optimal solution. This research proposes a dynamic inertia weight update approach based on fitness value. In the later stages of the algorithm, this strategy can improve the convergence speed and quality of the global optimal solution, allowing the particles to do a global search and eventually identify the population's ideal solution. Furthermore, IPSO creates a fitness function depending on task completion time. IPSO is used to test the performance of an enterprise's resource integration case. Experiments show that the method utilized can swiftly locate the ideal solution, complete the integration, and optimization of enterprise resources in the shortest job completion time, and for the least amount of money.

1. Introduction

Rapid advancements in science and technology have resulted in rapid economic growth. Economic development also poses difficulties for the management and operation of various businesses. To stand firm in today's rapidly changing society, businesses must not only have a long-term development plan on the outside but also a reasonable and efficient resource integration mechanism on the inside. The ability of an enterprise to screen, acquire, and combine various heterogeneous resources in the process of operation, including both the macrolevel that the enterprise obtains resources from the external environment where it is located, and the internal the microlevel of resource allocation. Some academics divide enterprise resource integration capabilities

into macrolevel categories based on this. The macrolevel enterprise resource integration capabilities include enterprises' ability to reconstruct operating rules and predict corporate strategies and risks in advance, whereas the microlevel enterprise resource integration capabilities include enterprises' ability to allocate, stimulate, and combine various existing resources. Existing research has examined the factors that may influence the development of enterprise resource integration capabilities in various contexts. According to the resource-based viewpoint, an enterprise is a collection of resources that can be obtained in a variety of ways. Resources are an important factor in an enterprise's long-term development. If a company wants to gain a competitive advantage and establish a market presence, it must invest in valuable resources and management skills.

Unrepeatable strategic management resources that can assist companies in successfully managing their businesses and obtaining additional compensation. At the moment, many scholars support the theory of resource-based view, and most studies are aimed at the impact of a specific type of enterprise's own resources on enterprise performance. Many of the results of the experimental analyses are consistent with resource-based theory. The uniqueness and nonreplicability of resources can play a role in the operation of businesses, forming competitive advantages and determining enterprise performance. Each enterprise's management resources are unique, and they have varying degrees of importance at each stage of the enterprise's operation. It can be defined as the various human, financial, and material resources that business managers manage in their daily operations. The good operation of the enterprise's resources can achieve the management goals of the enterprise.

Enterprise managers with varying management capabilities deal with their management resources in various ways. Reasonable resource allocation and utilization can result in satisfactory operational results. According to reference [1] the materials owned by the enterprise cannot directly create performance and obtain benefits for the enterprise. Professional resource integration talents, on the other hand, can integrate the enterprise's static resources and generate benefits for the enterprise. Differences in management resources will result in differences in resource allocation efficiency, resulting in differences in corporate performance. According to reference [2], each company has distinct management resources and management employees with different talents. As a result, each business will develop its own management style. This type of management disparity resulting from resource differences will have an impact on the company's performance. Firms with a wealth of management resources outperform their competitors. External mergers and acquisitions by businesses, according to reference [3], might boost a company's ability to manage resources swiftly. The outside offers fresh resources and management strategies to the company, which will boost the company's ability to adapt to changes in the external environment while also promoting its growth. Rich management resources, according to reference [4], can assist businesses optimize and alter their organizational structure to react to changing external market conditions. At the same time, businesses can create long-term plans and alter their management practices as needed. Reference [5] demonstrates that firms can strengthen their market position by fully utilizing management resources. To summarize, resources play an essential role in the development of enterprises, and management resources play a positive function in enhancing firm performance, according to resource-based theory. The better the managerial resources, the better the company's performance.

Enterprise resource management optimization's ultimate goal is to maximize the company's benefits. To optimize earnings, different companies have different criteria. It mostly entails optimizing earnings and reducing the time it takes to execute tasks. To do this, the enterprise's material and human resources must be integrated and dispatched in a

reasonable manner. Enterprise resource integration and scheduling are NP-complete problems [6, 7]. Particle swarm algorithms [8–10], ant colony algorithms [11–13], simulated annealing algorithms [14–16], and genetic algorithms [17–19] are currently available that can solve NP-complete problems. The PSO algorithm has the advantages of fast convergence speed, easy operation, and the algorithm is convenient for global optimization, but it also has disadvantages such as easy to fall into local optimum and low search accuracy. Aiming at these problems, this paper proposes an IPSO algorithm, and applies it to the instance of enterprise resource integration and scheduling. In order to verify the performance of the method used in this paper, the experimental results obtained through simulation experiments show that the IPSO algorithm has a good effect on solving the problem of enterprise resource integration.

2. Information Construction of Enterprise Resource Management

2.1. Enterprise Resource Management Platform. Business resources are not only the most significant aspect of an enterprise's internal environment, but they are also the prerequisites for enterprise resource integration. Enterprise management is now completely reliant on computers, and the advancement of information technology has expedited the reform of internal management and resource management in businesses. As a result, a series of information systems focused on the integration and deployment of enterprise resources have emerged. The system can help with resource integration and give a platform for all employees to quickly acquire tasks and resources. Figure 1 depicts the enterprise resource integration optimization platform's design structure.

Figure 1 depicts a three-tier architecture for an enterprise resource integration platform. The acquisition layer gathers the data that the system requires. The resource integration layer supports various resource release and interface negotiation models, as well as evaluating and classifying the released data before storing it in various databases. To achieve intelligent business resource integration, the resource application layer executes application processing on diverse enterprise information resources.

2.2. Optimization of Enterprise Resource Integration. Optimizing enterprise resource integration mainly starts from four aspects: resource creation, resource use, integration, and effect evaluation. The relationship and interaction of these four aspects are shown in Figure 2.

- (1) Theory is an important basis for guiding actual production. Constructivist theory mainly emphasizes the initiative and situational nature of resource integration. Resource integration is not a stimulus to passively receive information, but to actively process external information according to the corporate background. When the resource content is related to the needs of the enterprise, the enterprise should propose appropriate resource integration and

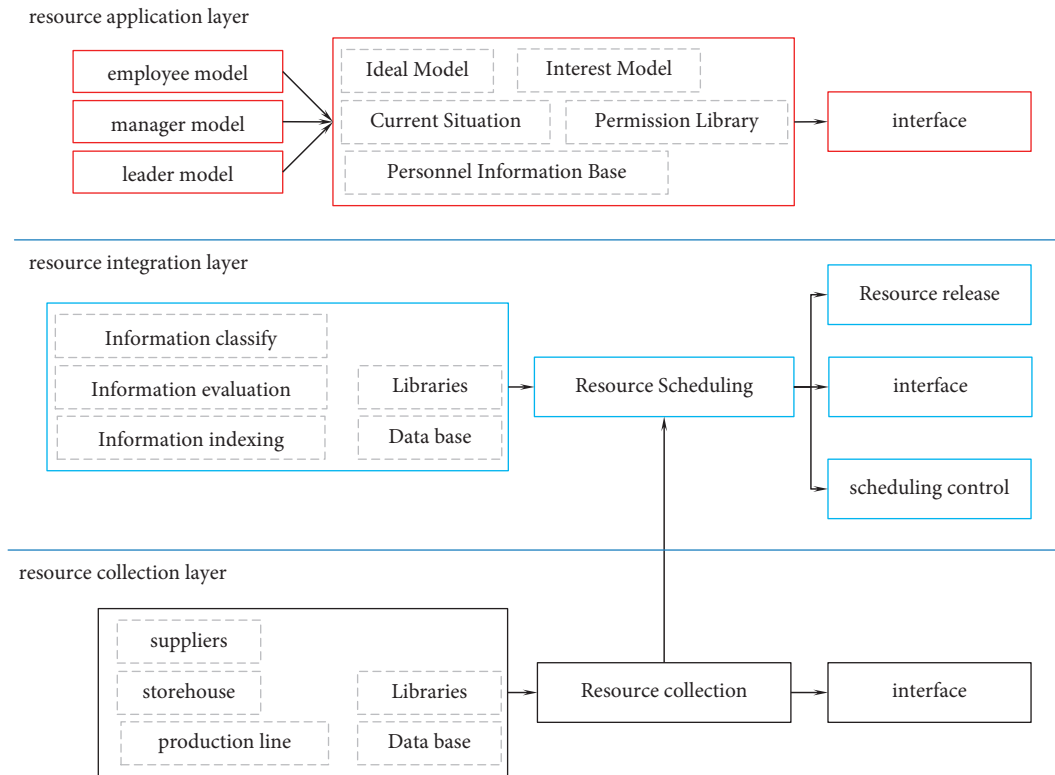


FIGURE 1: Architecture of enterprise resource integration platform.

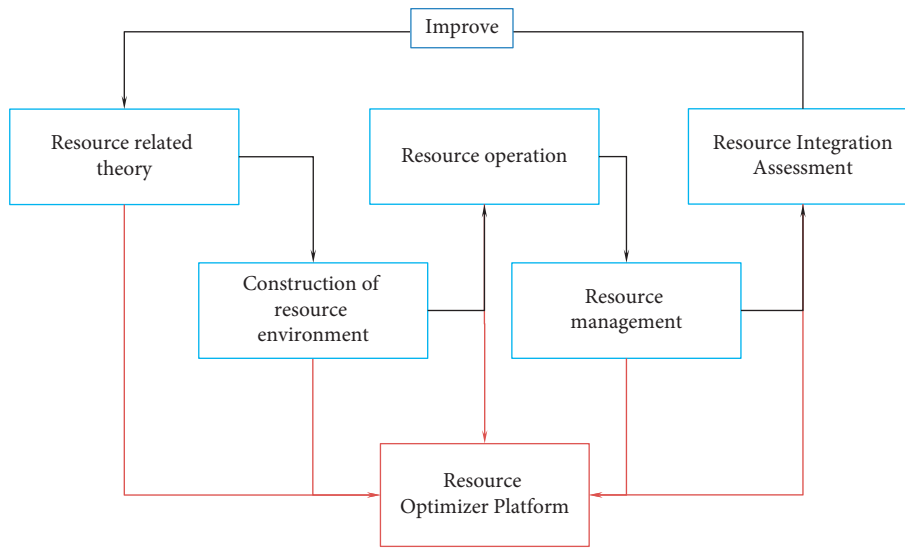


FIGURE 2: Optimization of enterprise resource integration.

allocation strategies to deal with various tasks. The resource integration dependence theory mainly emphasizes that the survival of the entire enterprise organization needs to absorb resources from the surrounding environment, and the interaction between the organization and the control resources in other environments is carried out.

- (2) Resource creation. Resource creation should take enterprise managers as the main body and employees'

needs as the fundamental starting point, so as to encourage employees, managers, and leaders to work together. Build a resource environment dominated by service resources to ensure the reliability of resources, thereby providing guarantees for sustainable development of resources.

- (3) Independent integration management includes user information maintenance, resource information maintenance, and integration strategy. User

information maintenance requires periodic updates and confidentiality of information. Enterprises need to set confidential information goals, formulate information integration plans, determine integration progress, and select integration resources to achieve self-maintenance. Resource information maintenance is to store and update resource information. Integrate and optimize resources based on integration strategies.

- (4) Evaluation is the guarantee for resource integration. With the goal of resource integration and optimization, the design evaluation process is shown in Figure 3. Figure 3 shows that evaluation is an important indicator to measure the rational integration of resources and is the driving force behind the integration of resources and information within an enterprise. To evaluate based on the integration of platform resources, it is not only necessary to analyze each link but also to evaluate the effect of self-integration of employee information in a timely manner.

3. Enterprise Resource Integration and Optimization of IPSO Algorithm

3.1. Introduction to PSO. Suppose a population X is searched in a D -dimensional space, and each population consists of H particles. Each particle in the population can be represented as a solution to the problem. $X_i(t)$ to represent the position of the i -th particle in the t -th iteration of the population. $V_i(t)$ is the velocity of the particle at the t -th iteration in the population. S_i represents the self-optimal solution of the i -th particle, and S_{best} represents the optimal solution of the entire particle population. The mathematical expression formula of each variable is as follows:

$$\begin{aligned} X(t) &= \{X_1(t), X_2(t), \dots, X_H(t)\}, \\ X_i(t) &= (x_{i1}(t), x_{i2}(t) \dots, x_{iD}(t)), \\ V_i(t) &= (v_{i1}(t), v_{i2}(t) \dots, v_{iD}(t)), \\ S_i &= (S_{i1}, S_{i2}, \dots, S_{iD}), \\ S_{\text{best}} &= (S_1, S_2, \dots, S_D). \end{aligned} \quad (1)$$

PSO updates the state of the particle through the particle's velocity and position; equation (2) is the velocity update expression, and equation (3) is the position update expression:

$$v_i(t+1) = wv(t) + e_1g_1(S_i(t) - x_i(t)) + e_2g_2(S_{\text{best}} - x_i(t)), \quad (2)$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \quad (3)$$

where w is the inertia weight coefficient, which is used to determine how much velocity is retained after the last iteration; e_1 , e_2 are the learning factors of the algorithm. Usually, both parameters are set to 2. g_1 , g_2 are random numbers whose values are between $[0, 1]$; t is the number of

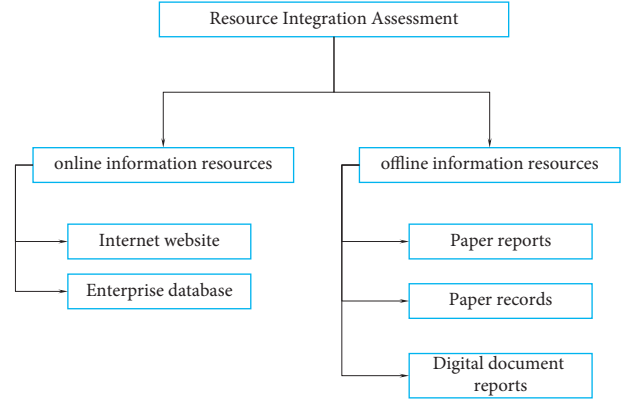


FIGURE 3: Enterprise resource integration assessment process.

iterations variable; and T is the maximum number of iterations. $[X_{\min}, X_{\max}]$ is the value range of position, and $[V_{\min}, V_{\max}]$ is the value range of velocity. The fitness value function f is used to judge the quality of particles. The size of the fitness function value indicates the pros and cons of the problem solution. After the particles are iterated, equation (4) is the update formula of the individual optimal value. When the maximum number of iterations is reached, the algorithm stops iterating:

$$S_i = \begin{cases} X_i(t), & f(X_i(t)) > f(S_i), \\ S_i, & f(X_i(t)) \leq f(S_i). \end{cases} \quad (4)$$

The flow of the PSO algorithm is shown in Figure 4. The specific implementation steps of PSO are as follows:

Step 1: Initialize the particle swarm. Given $t = 0$, bring the value of t into the particle velocity formula and position formula, and determine the initial position of the particle as $X_i(0)$ and the initial velocity as $V_i(0)$. Set all parameter values related to the algorithm, such as H , D , T , w , e_1 , e_2 , g_1 , g_2 .

Step 2: Calculate the fitness value of each particle.

Step 3: Set the fitness value of the current position of the particle as the optimal solution S_i of the particle itself. Set the optimal solution in the initial population as S_{best} .

Step 4: Update the particles.

- ① Update the position of the particle according to equation (3), and update the velocity of the particle according to equation (2).
- ② Judging the velocity and position of the current particle within a given range.
- ③ Calculate the fitness value of each particle, and then update S_i according to equation (4). Compare the fitness value of the individual optimal position and the group optimal position to update S_{best} .

Step 5: Judge whether the end condition is met, if the end condition is met, then step 6 is executed. Otherwise, the number of iterations is increased by 1, and step 4 is executed.

Step 6: Output S_{best} and the algorithm ends.

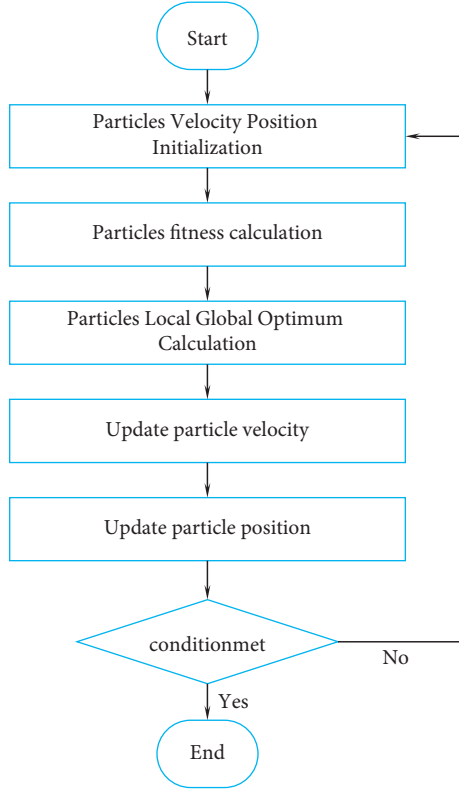


FIGURE 4: PSO flow chart.

3.2. IPSO Algorithm. Due to the slow convergence speed of the traditional PSO late stage, this will result in the inability to perform global search in the particle late stage. Aiming at this problem, this paper improves the update method of inertia weight value, which can improve the convergence ability and search ability of the algorithm. In addition, since the population initialization of PSO is random, this will inevitably lead to a part of the particles far away from the optimal solution and affect the quality of the particles. Therefore, in this paper, chaos theory is added to the population initialization process to improve the particle quality and optimization ability.

Choosing an appropriate inertia weight w is the key to improving the convergence and optimization capabilities of PSO. The size of the inertia weight value determines the optimization ability of the algorithm. When the inertia weight value is small, the local optimization ability of the algorithm is better. When the inertia weight value is large, the global optimization ability of the algorithm is better. However, in the traditional PSO, the inertia weight value is set to a fixed value; so many studies have proposed various improvement strategies for the inertia weight to make the optimization ability of the algorithm the best. Among the many improvement strategies, the improvement ideas of the following strategies are very good. The strategy can update the inertia weight value according to the number of iterations. The ability of particle local optimization and global optimization is balanced by dynamically adjusting the w value. The mathematical expression for this strategy is as follows:

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{T} \times t, \quad (5)$$

where w_{\max} represents the maximum inertia weight value, and w_{\min} represents the minimum inertia weight value. In the early stage of the algorithm search, because the number of iterations is relatively small, the value of w will be large, which is convenient for particles to search globally. In the later stage of the algorithm search, as the value of t increases, the value of w will become smaller, which makes it easier for the particle to find the local optimal solution.

Inspired by the above strategies, this paper proposes a new method to update inertia weights. This method also dynamically updates the inertia weight based on the size of the fitness value. f_{avg} represents the average fitness value of the population. f_i represents the fitness value of the particle currently being iterated, $i = 1, 2, 3, \dots, H$, H is the population size. The inertia weight value ranges from 0.4 to 0.9. The inertia weight expression used in this paper is as follows:

$$v_i(t+1) = wv_i(t) + e_1g_1(S_{\text{best}} - x_i(t)),$$

$$s.Tw = \begin{cases} w_{\min} + \frac{f_{\text{avg}} - f_i}{f_{\text{avg}}}, & f_i \leq f_{\text{avg}}, \\ w_{\max} - \frac{f_i - f_{\text{avg}}}{f_{\text{avg}}}, & f_i \geq f_{\text{avg}}. \end{cases} \quad (6)$$

Chaos theory has different chaotic mapping methods, this paper adopts Logistic mapping. The principle of logistic mapping is the regression equation. Logistic maps are well ergodic and the particles are evenly distributed over their range. Its regression equation is as follows:

$$y_{k+1,j} = 4y_{k,j}(1 - y_{k,j})y_{k,j} \in (0, 1), \quad (7)$$

where k is the number of iterations, and $y_{k,j}$ is the population sequence. This article will use chaos theory to initialize the population. PSO uses the chaotic sequence to produce a large number of particles during initialization and selects the particles with better quality as the initial particles of the population. This ensures that the particles are uniformly distributed in the solution space and the quality of the particles can be guaranteed.

- (1) The steps to initialize using the chaotic sequence are as follows:

- Step 1. Randomly generate a number $y_{0,0}$ in $[0, 1]$.
- Step 2. Bring $y_{0,0}$ into equation (8) for iteration to generate the sequence $y_{k,j}$.
- Step 3. Repeat step 2 until $k = 2^*H$;
- Step 4. Map the generated sequence to the solution space of the population according to equation (8) so that 2^*H particles can be obtained.

$$x_{k,j} = a + y_{k,j}(b - a), \quad (8)$$

where $a = X_{\min}$, $b = X_{\max}$.

- Step 5. Select the optimal H particles from the generated 2^*H particles as the initial population.

(2) The implementation steps of particle swarm optimization based on chaos theory are as follows:

Step 1. Population initialization based on chaos theory. Set the initial iteration number t to 0, and other parameters in the algorithm.

Step 2. Calculate the fitness value of the selected H particles with good quality.

Step 3. Set the fitness value of the current position of the particle as the optimal solution S_i of the particle itself. Set the optimal solution in the initial population as S_{best} .

Step 4. Update the particle's position and velocity:

- ① Update the velocity and position of the particle according to the relevant update formula.
- ② Determine whether the current particle velocity and position are beyond the feasible range.
- ③ Update S_i and S_{best} according to equation (4).

Step 5. Determine whether the end condition is reached. If the end condition is reached, execute step 6. If not, increase the number of iterations by 1, and execute step 4.

Step 6. Output S_{best} , and the algorithm ends.

4. Construction and Solution of Resource Integration Model Based on IPSO

4.1. Model Building. A single project has a clear project goal in the project planning stage and is divided into several tasks through work breakdown, and these tasks are all serving the project goal. Generally speaking, the scope of project objectives includes three dimensions: cost, time, and quality. In the multiproject management environment, the implementation of all projects of the enterprise serves the enterprise strategy. Project resource scheduling meets the project's time requirements for different resources as much as possible and strives to keep the completion time of all projects as short as possible. At the same time, such scheduling will inevitably involve organizational resources, so how to improve the utilization efficiency of organizational resources becomes a problem. In addition, the project leader will also pay attention to cost and quality issues. On the one hand, he hopes to reduce the project budget and improve the project income, and on the other hand, the quality should be as high as possible.

Under the premise of limited total resource supply, the multiproject resource scheduling problem can be regarded as the superposition of resource scheduling problems in multiple tiny time intervals. Therefore, it is possible to take a certain period of time during the execution of the project for analysis. Suppose the multiproject resource scheduling problem involves m mutually independent projects and n resource supply types. Within the time interval ΔT , there are e processes in progress in project i . Where the quantity of the k^{th} resource required by the j^{th} process is denoted as $z_{i,j,k}$, the quantity of the obtained resource is denoted as $n_{i,j,k}$, and the total supply of this type of resource is Z_k . Once the construction period delay in project i causes losses, the total cost of

delaying the project per unit time is recorded as s_i . Δt_i represents the engineering delay time of project i within the time interval ΔT . With the goal of minimizing project losses due to insufficient resource supply, the objective function is established, as follows:

$$Z = \min \sum_{i=1}^q \Delta t_i s_i, \quad (9)$$

where s_i is a known quantity, which can be known from the contract; Δt_i is an unknown quantity, which is determined by the quantity $z_{i,j,k}$ of resources obtained by the process j performed by the project i in the time period ΔT . Under the condition that the total supply of various resources is limited, $n_{i,j,k}$ should meet the restrictive conditions:

$$\sum_{i=1}^q \sum_{j=1}^q n_{ijk} \leq Z_k, k = 1, 2, \dots, w. \quad (10)$$

In the time period ΔT , the number of resources required by the critical path process in project i is z_{izk} , and the number of resources obtained is n_{izk} . Let t_{iz} denote the planned working time of the critical path, $t_{iz} = \Delta T$, and t'_{iz} denote its actual working time. From the principle that the total consumption of each resource in the same process remains unchanged, that is, $z_{izk} t_{iz} = n_{izk} t'_{iz}$, $t'_{iz} = z_{izk} t_{iz} / n_{izk}$, $k = 1, 2, \dots, w$. Therefore, the delay time of the critical path of project i can be expressed and known as

$$\Delta t_i = \Delta t_{iz} = \frac{z_{izk} - n_{izk}}{n_{izk}}. \quad (11)$$

In the same way, for any process on the noncritical path in project i , mark the number of required resources as r_{isk} , and the number of obtained resources as n_{isk} , within the time period ΔT , the planned working time of this path is t_{if} , and the time difference is t_{is} . The delay time is expressed as follows:

$$\Delta t_{is} = \max \left\{ \frac{t_{is} (z_{isk} - n_{isk})}{n_{isk}} \right\}. \quad (12)$$

In order to ensure that the critical path of project i does not change, the difference between the delay time of the noncritical path and the time difference owned by the process should not exceed the delay time of the key process, namely,

$$\Delta t_{is} - T_{is} \leq \Delta t_i. \quad (13)$$

In the actual project implementation, the construction of any process may require multiple resources at the same time, and a certain proportional relationship must be satisfied between different resources to coordinate and promote the process forward. In resource scheduling, under any circumstances, for the ongoing process j in project i , the supply quantity of resources and the demand quantity should satisfy the following proportional relationship:

$$n_{ijk} = \eta_{ij} z_{ijk}, \quad k = 1, 2, \dots, w, 0 < \eta_{ij} \leq 1, \quad (14)$$

where η_{ij} represents the resource allocation coefficient related to each process j under construction within the time interval ΔT .

To sum up, substituting equations (7) and (11) into equations (9), (10) and (13) respectively, we get

$$Z = \min \left[\Delta T \sum_{i=1}^q s_i \left(\frac{1}{\eta_{iz}} - 1 \right) \right], \quad (15)$$

$$\sum_{i=1}^q \sum_{j=1}^q \eta_{ij} z_{ijk} \leq Z_k, \quad k = 1, 2, \dots, w, \quad (16)$$

$$t_{is} \eta_{iz} \leq \Delta T \eta_{is}. \quad (17)$$

Equations (15)–(17) constitute a mathematical model of enterprise resource scheduling in a multiproject environment with limited resources.

4.2. Model Solution. The IPSO algorithm is proposed to be used for enterprise resource integration and scheduling. In order to verify the effectiveness of the proposed algorithm for enterprise resource integration scheduling, this paper introduces the method into a practical case. A company has a total of four projects to be completed within a certain period of time, and there are three types of resources needed and available. Each project has a different number of processes. Every project has strict delivery time requirements. The loss cost M_i ($i = P1, P2, P3, P4$) of each item per delay unit time is 1000 for M_{P1} , $M_{P2} = 800$, $M_{P3} = 1500$, $M_{P4} = 1800$. Table 1 is a breakdown of the resource requirements of the enterprise to complete each project, where ΔT is set to 5.

Substitute the data shown in Table 1 into the resource integration model shown in equation (15), and the obtained formula is as follows:

$$\begin{aligned} \min z = & 5 * 1000 \left(\frac{1}{x_3} - 1 \right) + 5 * 800 \left(\frac{1}{x_5} - 1 \right) + 5 * 1500 \left(\frac{1}{x_6} - 1 \right) + 5 * 1800 \left(\frac{1}{x_{10}} - 1 \right), \\ \text{s.t. } & 18x_1 + 15x_2 + 10x_3 + 29x_4 + 17x_5 + 13x_6 + 21x_7 + 16x_8 + 23x_9 + 18x_{10} \leq 150 \\ & 25x_1 + 27x_2 + 16x_3 + 18x_4 + 23x_5 + 26x_6 + 10x_7 + 18x_8 + 15x_9 + 11x_{10} \leq 190 \\ & 9x_1 + 16x_2 + 35x_3 + 18x_4 + 12x_5 + 11x_6 + 14x_7 + 23x_8 + 14x_9 + 8x_{10} \leq 150 \\ & 4x_3 \leq 5x_1 \\ & 2x_3 \leq 5x_2 \\ & 3.5x_5 \leq 5x_4 \\ & 4x_6 \leq 5x_7 \\ & 4.5x_{10} \leq 5x_8 \\ & 3x_{10} \leq 5x_9 \\ & 0 < x_i < 1, \end{aligned} \quad (18)$$

where x_1, x_2, x_3 represents the resource allocation coefficient of process P11, P12, P13; x_4, x_5 represents the allocation coefficient of process P21, P22, x_6, x_7 represents the resource allocation coefficient of process P31, P32; x_8, x_9, x_{10} represents the resource allocation coefficient of process P41, P42, P43.

For the above mathematical programming model, the standard particle swarm optimization algorithm and the new PSO optimization algorithm proposed in this paper are used to solve the problem, respectively. The parameter settings of the algorithm are evolution times $T=600$, population size $D=300$, and parameter $e_1=1.532$, $e_2=1.541$, the error $\varepsilon=0.0001$.

In addition, Figure 5 shows the simulation results of the two algorithms. It can be seen from the figure that the

average value of the project loss cost obtained by the standard particle swarm optimization method is larger than the result obtained by the new particle swarm optimization algorithm. According to the discrete degree of the objective function, the simulation results of the new particle swarm optimization algorithm are more concentrated, while the simulation results of the standard PSO [20] are more scattered.

After the above comparison and analysis of the simulation results of the two algorithms, it is concluded that compared with the standard particle swarm optimization algorithm, the improved particle swarm optimization algorithm in this paper has higher stability and better convergence.

TABLE 1: Details of enterprise resource requirements.

Project	Process	Required time	Time difference	Resource R1	Resource R2	Resource R2
P1	P11	4	0	18	25	9
	P12	2	2	15	27	16
	P13	5	2.5	10	16	35
P2	P21	3.5	3	29	18	18
	P22	5	2	17	23	12
P3	P31	5	1.5	13	26	11
	P32	4	2	21	10	14
P4	P41	4.5	3	16	18	23
	P42	3	1	23	15	14
	P43	5	3	18	22	8
Resource requirements				180	200	160
Max supplies				150	190	150

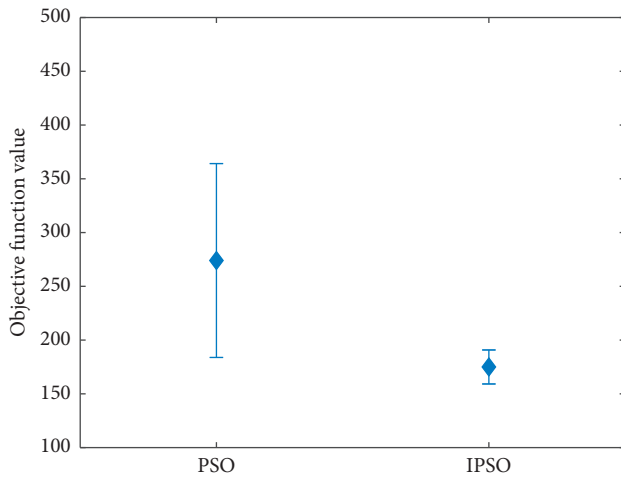


FIGURE 5: Comparison of project loss costs obtained by different algorithms.

According to the resource allocation coefficients obtained by the PSO and IPSO algorithms, the allocation of each resource among different projects is calculated, and the resource allocation of the two algorithms is obtained as shown in Tables 2 and 3.

Observing the data in Table 2, it can be seen that the allocation of resource R1 based on PSO is 136.17, the allocation of resource R2 is 152.63, and the allocation of resource R3 is 121.07. There is a large gap between the calculated allocation and supply of each resource. This shows that the allocation of resources is not reasonable. The delay times for the three operations of project P1 are 1.489, 3.119, and 2.765, respectively. The planned time differences corresponding to these three processes are 0, 2, and 2.5, respectively. The delay time calculated based on PSO is larger than the planned time difference. This shows that resource integration based on PSO is not very effective. The delay times for the two processes of project P2 are 2.120 and 2.221 respectively. The planned time differences corresponding to these two processes are 3 and 2, respectively. It can be seen that the process P21 can be completed within the set time difference, but the process P22 cannot be completed within the set time difference. The delay times for the two processes of project P3 are 1.765 and 0.815

respectively. The planned time differences corresponding to these two processes are 1.5 and 2, respectively. It can be seen that the process P32 can be completed within the set time difference, but the process P31 cannot be completed within the set time difference. The delay times for the three operations of project P4 are 2.642, 0 and 2.812, respectively. The planned time differences corresponding to these three processes are 3, 1, and 3, respectively. From the comparison data, it can be seen that these three processes can be completed within the set time difference. To sum up, in the resource integration scheme based on the PSO algorithm, except for P4, which can be completed according to the set time difference, other projects cannot be completed within the set time difference.

Observing the data in Table 3, it can be seen that the allocation of resource R1 based on IPSO is 159.74, the allocation of resource R2 is 175.89, and the allocation of resource R3 is 137.41. The calculated allocation of each resource is very close to the available Supply amount. This shows that the allocation of resources is more reasonable. In the results calculated based on IPSO, the delay times for the three processes of project P1 are 1.622, 2.120 and 3.235, respectively. The planned time differences corresponding to these three processes are 0, 2, and 2.5, respectively. The delay time calculated based on IPSO is larger than the planned time difference. The delay times for the two processes of project P2 are 1.522 and 1.767, respectively. The planned time differences corresponding to these two processes are 3 and 2, respectively. It can be seen that both steps P21 and P22 can be completed within the set time difference. The delay time for the 2 processes of project P3 is 0.521,0 respectively. The planned time differences corresponding to these two processes are 1.5 and 2, respectively. It can be seen that both steps P31 and P32 can be completed within the set time difference. The delay times for the three processes of project P4 are 2.764, 1.656, and 2.701, respectively. The planned time differences corresponding to these three processes are 3, 1, and 3, respectively. From the comparison data, it can be seen that this process P41 and P43 can be completed within the set time difference, and P42 cannot be completed within the set time difference. To sum up, in the resource integration scheme based on the PSO algorithm, except for P2 and P3, which can be completed according to the set time difference,

TABLE 2: Details of the actual allocation of enterprise resources based on PSO.

Project	Process	Required time	Time difference	Resource R1	Resource R2	Resource R2	Partition coefficient	Delay value
P1	P11	4	0	13.33	18.51	6.66	0.7403	1.489
	P12	2	2	12.38	22.28	13.20	0.8251	3.119
	P13	5	2.5	6.56	10.49	22.95	0.6557	2.765
P2	P21	3.5	3	13.20	8.20	8.20	0.4553	2.120
	P22	5	2	9.29	12.57	6.56	0.5465	2.221
P3	P31	5	1.5	13.00	26.00	11.00	1	1.765
	P32	4	2	19.50	9.29	13.00	0.9287	0.815
P4	P41	4.5	3	14.08	15.84	20.24	0.8798	2.642
	P42	3	1	23.00	15.00	14.00	1	0
	P43	5	3	11.84	14.47	5.26	0.6576	2.812
Original resource requirement				180	200	160		
Actual allocation of each resource				136.17	152.63	121.07		
Max supplies				160	190	150		

TABLE 3: Details of the actual allocation of enterprise resources based on IPSO.

Project	Process	Required time	Time difference	Resource R1	Resource R2	Resource R2	Partition coefficient	Delay value
P1	P11	4	0	17.21	23.90	8.60	0.9561	1.622
	P12	2	2	15.00	27.00	16.00	1	2.120
	P13	5	2.5	7.76	12.41	27.14	0.7755	3.235
P2	P21	3.5	3	24.54	15.23	15.23	0.8462	1.522
	P22	5	2	12.48	16.88	8.81	0.7341	1.767
P3	P31	5	1.5	11.33	22.66	9.59	0.8715	0.521
	P32	4	2	21.00	10.00	14.00	1	0
P4	P41	4.5	3	12.06	13.57	17.34	0.7539	2.764
	P42	3	1	22.20	14.48	13.51	0.9652	1.656
	P43	5	3	16.16	19.75	7.18	0.8978	2.701
Original resource requirement				180	200	160		
Actual allocation of each resource				159.74	175.89	137.41		
Max supplies				160	190	150		

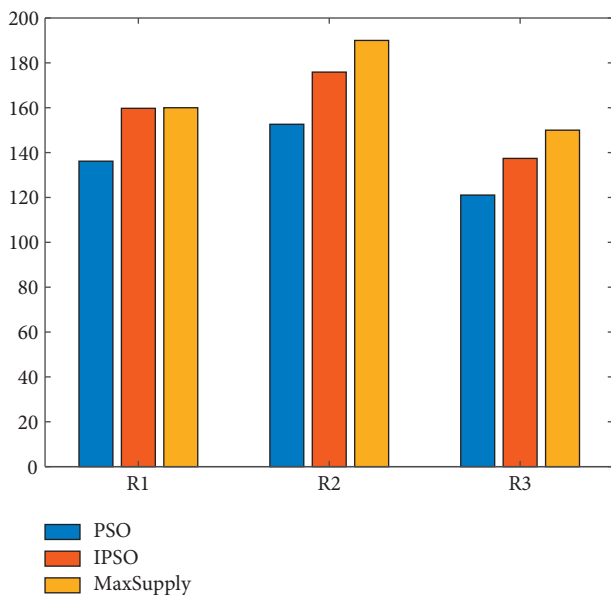


FIGURE 6: Comparison of resource allocation of different algorithms.

other projects cannot be completed within the set time difference.

In order to compare the resource integration and allocation schemes obtained by the two methods more intuitively, screen out the optimal resource allocation scheme, and give a resource allocation comparison chart, as shown in Figure 6. In the figure, MaxSupply represents the amount of resources that the enterprise can supply. From the data in the figure, it can be seen that the allocation of the three resources based on IPSO is the closest to the maximum supply. However, there is still a big gap between the distribution scheme based on PSO and the maximum supply.

5. Conclusion

In the process of continuous operation and development of enterprises, there will definitely be some problems in the use and deployment of internal resources. In order to optimize the integration and allocation of enterprise resources, this paper proposes to use an optimization algorithm to allocate resources. Its purpose is to efficiently complete the delivery of each project within the contracted construction period. Based on the traditional PSO algorithm, this paper proposes

an IPSO algorithm. Firstly, to solve the problem of uneven particle distribution caused by random initialization of the population, the algorithm adds chaos theory to the initialization of particle population. It uses the Logistic mapping sequence to generate a large number of particles, and selects the particles with better quality for initialization, which improves the quality of the particles and enables the particles to be evenly distributed during initialization. Secondly, the late convergence speed of particle swarm optimization algorithm is slow, which makes it easy to fall into the local optimal solution. Aiming at this problem, a dynamic inertia weight update method based on fitness value is designed. This can improve the convergence speed in the later stage of the algorithm, improve the quality of the global optimal solution, and enable the particles to perform a global search and finally find the optimal solution of the population. Finally, a fitness function based on task completion time is designed, and the IPSO algorithm is applied to the integration and optimization of internal resources of the enterprise. Compared with the standard particle swarm optimization algorithm, the IPSO algorithm has higher stability and better convergence. In the experimental part, a set of optimal solutions are selected to solve the resource allocation of each project process in the experimental example. The results show that the IPSO algorithm can produce better results for this kind of resource allocation problem. It can not only allocate the limited enterprise resources reasonably, but also minimize the total project loss caused by the delay of the construction period. The optimization results show that the IPSO algorithm proposed in this paper can effectively integrate and schedule the internal resources of the enterprise and improve the operation efficiency of the enterprise.

Data Availability

The labeled data set used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

This work was supported by the Changzhi Vocational and Technical College.

References

- [1] B. Wernerfelt, "A resource-based view of the firm," *Strategic Management Journal*, vol. 5, no. 2, pp. 171–180, 1984.
- [2] D. J. Teece, G. Pisano, and A. Shuen, "Dynamic capabilities and strategic management," *Strategic Management Journal*, vol. 18, no. 7, pp. 509–533, 1997.
- [3] D. G. Sirmon, "Managing firm resources in dynamic environments to create value: looking inside the black box," *Academy of Management Review*, vol. 32, no. 1, pp. 173–192, 2007.
- [4] D. Hillman and C. Blom, "Optimizing process efficiency in a multi-stream keying environment," *TODAY-The Journal of Work Process Improvement*, vol. 9, no. 6, p. S10, 2009.
- [5] S. C. Lonial and R. E. Carter, "The impact of organizational orientations on medium and small firm performance: a resource-based perspective," *Journal of Small Business Management*, vol. 53, no. 1, pp. 94–113, 2015.
- [6] H. Hoorfar and A. Bagheri, "NP-completeness of chromatic orthogonal art gallery problem," *The Journal of Supercomputing*, vol. 77, no. 3, pp. 3077–3109, 2021.
- [7] A. Kulmburg and M. Althoff, "On the co-NP-completeness of the zonotope containment problem," *European Journal of Control*, vol. 62, pp. 84–91, 2021.
- [8] D. Zhang, R. Chi, J. Li, X. Chi, and T. Zhu, "Cuckoo optimization algorithm visual positioning system based on particle swarm algorithm," *Advances in Intelligent Automation and Soft Computing*, vol. 80, pp. 16–21, 2022.
- [9] M. E. C. Bento, "A hybrid particle swarm optimization algorithm for the wide-area damping control design," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 1, pp. 592–599, 2022.
- [10] M. Quarto, G. D'Urso, and C. Giardini, "Micro-EDM optimization through particle swarm algorithm and artificial neural network," *Precision Engineering*, vol. 73, pp. 63–70, 2022.
- [11] Y.-q. Chen, J.-l. Guo, H. Yang, Z.-q. Wang, and H.-l. Liu, "Research on navigation of bidirectional A* algorithm based on ant colony algorithm," *The Journal of Supercomputing*, vol. 77, no. 2, pp. 1958–1975, 2021.
- [12] L. Yun and H. Lu, "Diagnosis strategy design method based on ant colony algorithm," *Journal of Physics: Conference Series*, vol. 1894, no. 1, Article ID 012081, 2021.
- [13] M. Sharifian, N. Abdolvand, and S. Rajaei Harandi, "Context-based expert finding in online communities using ant colony algorithm," *Journal of Information Systems and Telecommunication (JIST)*, vol. 8, no. 30, pp. 130–139, 2020.
- [14] S. Nayeri, R. Tavakkoli-Moghaddam, Z. Sazvar, and J. Heydari, "A heuristic-based simulated annealing algorithm for the scheduling of relief teams in natural disasters," *Soft Computing*, vol. 26, no. 4, pp. 1825–1843, 2022.
- [15] V. F. Yu, H. Susanto, P. Jodiawan, T.-W. Ho, S.-W. Lin, and Y.-T. Huang, "A simulated annealing algorithm for the vehicle routing problem with parcel lockers," *IEEE Access*, vol. 10, pp. 20764–20782, 2022.
- [16] S. Ren, M. Gao, M. Wang, and Y. Li, "Polarized laser backscattering of atmospheric cloud distribution based on simulated annealing algorithm," *Mathematical Problems in Engineering*, vol. 2021, no. 1, 9 pages, Article ID 6635828, 2021.
- [17] A. Zemliak, "A modified genetic algorithm for system optimization," *COMPEL: The International Journal for Computation & Mathematics in Electrical & Electronic Engineering*, vol. 41, no. 1, pp. 499–516, 2022.
- [18] F. Hajati, A. Rezaee, and S. Gheisari, "Genetic algorithms for scheduling examinations," *Advanced Information Networking and Applications*, vol. 227, pp. 524–532, 2021.
- [19] M. Bortolini, C. Cafarella, E. Ferrari, F. G. Galizia, and M. Gamberi, "Reconfigurable manufacturing system design using a genetic algorithm," *Sustainable Design and Manufacturing*, vol. 262, pp. 130–139, 2022.
- [20] Q. Wang, X.-L. Fu, G.-F. Dong, and T. Li, "Research on cloud computing task scheduling algorithm based on particle swarm optimization," *Journal of Computational Methods in Science and Engineering*, vol. 19, no. 2, pp. 327–335, 2019.