Research Article

AI-Driven Multiobjective Scheduling Algorithm of Flood Control Materials Based on Pareto Artificial Bee Colony

Banteng Liu, Junjie Lu, Yourong Chen, Ping Sun, Kehua Zhao, Meng Han, Rengong Zhang, and Zegao Yin

1College of Information Science and Technology, Zhejiang Shuren University, Hangzhou, Zhejiang 310015, China
2School of Computer Science and Artificial Intelligence, Changzhou University, Changzhou, Jiangsu 213164, China
3Data-Driven Intelligence Research (DIR) Lab, Kennesaw State University, Marietta, GA 30060, USA
4Zhejiang Yugong Information Technology Limited Company, Hangzhou, Zhejiang 310051, China
5College of Engineering, Ocean University of China, Qingdao, Shandong 266100, China

Correspondence should be addressed to Meng Han; mhan9@kennesaw.edu

Received 6 January 2021; Accepted 5 June 2021; Published 22 June 2021

Academic Editor: Carles Gomez

Copyright © 2021 Banteng Liu 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.

Considering the competition between rescue points, we use artificial intelligence (AI) driven Internet of Thing (IoT) and regional material storage data to propose a multiobjective scheduling algorithm of flood control materials based on Pareto artificial bee colony (MSA_PABC). To address the scheduling of flood control materials, the multiple types of flood control materials, the multiple disaster sites, and entertain both emergency and fairness of rescue need to be considered comprehensively. The MSA_PABC has the constraints such as storage quantity constraint of warehouse materials, material demand constraint, and maximum transportation distance of flood control materials. We establish the scheduling optimization model of flood control materials for each disaster rescue point and the total scheduling optimization model for all flood control materials. Then, MSA_PABC uses the modified Pareto artificial bee colony algorithm to solve the multiobjective models. Three types of initialization strategies are proposed to calculate the fitness of each rescue point and the overall evaluation value of the food source. We propose the employ bee operations such as niche technology and local search of the variable neighborhood, the onlooker bee operations such as Pareto nondominated sorting and crossover operation, the scout bee operations such as maximum evolutionary threshold, and end elimination mechanism. Finally, our proposed solution obtains the nondominated solution set and its optimal solution. The experimental results show that no matter how the number of rescue points changes, MSA_PABC can find the nondominated solution set and optimal solution quickly. It improves the convergence rate of MSA_PABC and material satisfaction rate. Our solution also reduces the average maximum transportation distance, the standard deviation of maximum transportation distance, and the standard deviation of material satisfaction rate. The evaluation also demonstrates MSA_PABC outperforms the state-of-arts such as ABC (artificial bee colony), NSGA2 (nondominated sorting genetic algorithm 2), and MOPSO (multiobjective particle swarm optimization).

1. Introduction

China has a vast territory, and the precipitation partially concentrates. The precipitations time and space distribution are uneven, and it does not match the distribution of population and cultivated land. Therefore, disasters such as floods and typhoons occur frequently, causing a large number of people to suffer disasters and huge economic losses [1]. For instance, in 2019, 130 million people are suffered from disasters such as typhoons. The direct economic losses were 327.09 billion RMB [2]. Zhejiang Province locates on the southeastern coast of China. Due to its special geographic location and climatic conditions, it is susceptible to typhoons and storms. However, the drainage capacity of Zhejiang Province is limited. Once the flood disasters occur, it will cause serious economic and loss of life to people. Therefore, flood prevention and rescue are particularly important [3]. In the past, rescue dispatching mainly issued dispatching instructions by means of...
of communication. The problem lies in low efficiency and chaotic command. Once the scale of rescue dispatching rises, its order and efficiency will be more difficult to maintain.

With the rapid development of computer technology, sensor technology, and Internet of Things technology, various industries in the market are developing in the direction of high automation and intelligence and hope to improve the service ability of the industry through AI and IOT technology [4, 5]. The future development of flood control and disaster reduction in the direction of intelligence is gradually becoming a consensus in the industry. Internet of Things technology as a basic support means to determine the quantity and availability of materials through a variety of information sensing equipment and electronic tags in the field of flood control, and upload the relevant information to the Internet platform for information interaction to achieve intelligent management of materials. At the same time, it provides data support for the subsequent solution of large-scale orderly and efficient scheduling of materials. The dynamic scheduling problem of rescue materials is NP hard in nature. It is usually solved with various artificial intelligence algorithms based on the existing Internet of Things technology [6, 7].

At present, some scholars have achieved certain results in emergency dispatch, but the characteristics of flood control and rescue scheduling determine that the applicability of artificial intelligence algorithm needs to be analyzed objectively to ensure that the application has a certain quality of service. For instance, Song et al. [8] propose a double mutation improved differential evolution algorithm with the Pareto concept and establish a multisupply point to multirescue point emergency material scheduling model in the case of limited resources to minimize the total cost of distribution expenses and the largest missing loss. Tian et al. [9] take the urgency of different needs into consideration and establish a multiobjective mathematical model for dynamic distribution scheduling. Then, they use a weighted particle swarm optimization algorithm with swarm intelligence to solve the model. However, the above algorithms are more likely to fall into local optimum with the increase of scheduling scale, which makes the reliability of rescue decline. Zhang et al. [10] propose a hybrid intelligent search algorithm based on two-dimensional NSGA-II and ant colony optimization to establish a multiobjective optimization model for concurrent allocation and scheduling of multiple emergency rescue materials. Some scholars [11–14] focus on establishing a material scheduling optimization model which considers multiple parameters and using genetic algorithms (GA), sequential linear programming algorithms (SLP), greedy algorithms, and other algorithms to solve the optimization model and obtain the optimal solution. However, the above algorithms do not consider the dynamic change of material demand in the rescue process to make the material scheduling lag behind.

Some scholars [15–18] focus on establishing multiple target optimization models, such as minimizing the transportation time of materials and maximizing reliability. Then, they use single or multiple hybrid algorithms, such as quick sorting genetic algorithm, harmony algorithm, artificial bee colony algorithm, Monte Carlo algorithm, and stepwise method, to obtain the optimal scheduling schemes. Considering the processing time of emergencies, Wex et al. [19] propose a decision support model for allocating existing rescue units to emergency action centers to improve the efficiency of incident handling and reduce casualties and economic losses in the reaction phase. Considering that the emergency supplies provided to rescue points may be insufficient or excessive, Chen et al. [20] take the minimization of the loss caused by insufficient material distribution and oversupply and the minimization of vehicle scheduling costs as optimization objectives and propose a vehicle scheduling optimization model of disaster emergency logistics based on discrete bee colony. In [8–20], they use intelligent optimization algorithms to solve emergency rescue scheduling problems but do not take account of the urgency and fairness of multiobjective rescue scheduling at the same time, only meet the rescue service requirements of individual rescue points, which makes the overall rescue service quality poor, and only focus on resource allocation, without considering the unexpected factors that affect the service quality such as road damage in the rescue process.

Therefore, based on the algorithms mentioned above, we propose a multiobjective scheduling algorithm of flood control materials based on Pareto artificial bee colony (MSA_PABC). Considering the convergence performance of the algorithm, the feasibility of the actual road transportation in the process of material allocation, and the satisfaction of the allocation between rescue points and the whole rescue, we establish the optimization model of flood control material scheduling for each disaster relief point and the optimization model of all flood control materials scheduling from the time cost, task execution reliability, and service satisfaction. According to the importance and easy measurement of service quality parameters, we select service quality parameters and establish a parameter system including four service quality parameters (transportation efficiency, satisfaction rate of disaster site, transportation reliability, and actual availability of material reserve), which is used as constraint function to guide the optimization objective of an artificial intelligence scheduling algorithm, that is, we propose the modified Pareto artificial bee colony algorithm to solve the multiobjective optimization model. In the initial phase, we propose a variety of food source initialization strategies and calculate the fitness of each rescue point and the overall evaluation value of the food source. In the employed bee phase, we use niche technology and variable neighborhood local search and other employ bee operations to strengthen the local search of the food source. In the onlooker bee phase, we use Pareto dominate sorting, crossover operation, and other onlooker operations to expand the search capabilities of high-quality food sources in areas with small distribution densities in the multidimensional solution space. In the scout bee phase, we use the maximum evolution threshold and end-elimination mechanism to continuously update old food sources to ensure the diversity of solutions. Finally, MSA_PABC can quickly find nondominated sets and optimal solutions, thereby improving the convergence rate and material satisfaction rate and reducing the average maximum transport.
distance, standard deviation of the maximum transport distance, and standard deviation of material satisfaction rate for flood control materials to achieve the overall optimal rescue service quality with practical feasibility.

2. Scheduling Model Establishment

The flood control emergency scheduling is a prerequisite for realizing rapid and accurate handling of dangerous situations. Therefore, it is necessary to take the actual storage of warehouse materials and situations of disasters of each region as influencing factors. According to flood control materials constraints of storage capacity, demand, and transportation, we establish the scheduling optimization model of flood control materials for each disaster rescue point and the total scheduling optimization model of all flood control materials. The details are as follows:

We set \( H_i^k \) to represent the storage capacity of flood control material \( k \) in warehouse \( i \). Considering the number of flood control material \( k \) allocated by the storage warehouse \( i \) to all rescue points does not exceed the total storage capacity of flood control materials \( k \) in the warehouse \( i \), the storage capacity constrains of warehouse materials is

\[
\sum_j W_{ij}^k \leq H_i^k \quad \forall i, k, \tag{1}
\]

where \( W_{ij}^k \) represents the number of flood control material \( k \) allocated by warehouse \( i \) to rescue point \( j \).

To avoid wastage and use the flood control materials effectively, it is required that the number of flood control material \( k \) from all warehouses to rescue point \( j \) does not exceed the required quantity of the flood control material \( k \) in the rescue point \( j \). Thus, the material demand constraint is

\[
\sum_i W_{ij}^k \leq j_i^k \quad \forall j, k, \tag{2}
\]

where \( j_i^k \) represents the required quantity of the flood control material \( k \) in the rescue point \( j \).

According to the value \( W_{ij}^k \), we can know the storage warehouses and rescue points that need to be passed during flood control materials transportation. We assume that the transport capacity is adequate during the transportation of flood control materials. Then, according to the GIS system and the distribution of flood disasters, we determine the shortest transportation distance \( P_{ij}^k \) of the flood control material \( k \) between the storage warehouse \( i \) and the rescue point \( j \) that can avoid flooding inundated roads [21]. Thus, we can obtain the maximum transportation distance for finishing the flood control material scheduling at rescue point \( j \).

\[
D_j = \max \bigg( P_{ij}^k, \forall i, k \bigg), \tag{3}
\]

where \( D_j \) represents the maximum distance required to finish the transportation of flood control materials at rescue point \( j \).

To meet the demand for rescue points for flood control materials, the material satisfaction rate of rescue point \( j \) is

\[
R_j = \frac{\sum_i W_{ij}^k}{\sum_i j_i^k}, \tag{4}
\]

where \( R_j \) represents the material satisfaction rate of rescue point \( j \).

In the aspect of disaster rescue points, each of them hopes that the required flood control materials can meet the fastest transportation time and the maximum satisfaction rate. We establish a scheduling optimization model of flood control materials for each disaster rescue point \( j \).

\[
\min \left( \frac{D_j}{R_j} \right) \quad \text{s.t. } \sum_i W_{ij}^k \leq j_i^k, \forall k
\]

Formulas\(1), (3), (4)\)

\[
W_{ij}^k \geq 0, \forall i, k
\]

In the aspect of flood control material scheduling management, the flood control materials can weigh the principle of urgency and fairness. Therefore, we set the standard deviation of the maximum transportation distance of flood control materials is

\[
\text{Std}_w = \sqrt{\frac{\sum_j \left( D_j - \sum_j D_j/N_j \right)^2}{N_j}}, \tag{6}
\]

where \( \text{Std}_w \) represents the standard deviation of the maximum transportation distances of flood control materials and \( N_j \) represents the number of rescue points. The standard deviation of the satisfaction rate of flood control materials is

\[
\text{Rat}_w = \sum_j \sqrt{\left( R_j - \sum_j R_j/N_j \right)^2/N_j}, \tag{7}
\]

where \( \text{Rat}_w \) represents the standard deviation of the satisfaction rate of flood control materials. Considering the emergency of flood control materials, it is required that the maximum transportation distance and satisfaction rate between rescue points are not much different. The standard deviation of maximum transportation distances and the standard deviation of material satisfaction rates need to be minimized. Therefore, we establish the total scheduling optimization model of all flood control materials.
\[
\min \left( D_{\text{ave}} \times (x_1 \text{Std}_{\text{w}} + x_2 \text{Rat}_{\text{w}})/R_{\text{ave}} \right) \\
\text{s.t. Formulas (1) - (4)} \\
D_{\text{ave}} = \sum_{j} D_j / N_j, \\
R_{\text{ave}} = \sum_{j} R_j / N_j, \\
W_{ij}^k \geq 0, \forall i, j, k
\]

where \( D_{\text{ave}} \) represents the maximum average distance required to finish flood control materials transportation, \( R_{\text{ave}} \) represents the average material satisfaction rate of rescue points, \( x_1 \) represents the distance factor, \( x_2 \) represents the satisfaction rate factor, and \( x_1 + x_2 = 1 \).

3. Scheduling Model Establishment

Since the models (5) and (8) consider the selection of three elements, such as the storage warehouse, rescue point, and flood control material, the optimization model is complicated and involves a lot of calculations. Therefore, we use artificial intelligence to solve the optimization models (5) and (8) and obtain the optimal plan by constantly searching for the distribution scheme of reserve warehouses, rescue points, and flood control materials. The inspiration of the artificial bee colony algorithm (ABC) proposed by Karaboga et al. [22] is based on bee foraging behavior. In ABC, the artificial bee colony comprises three kinds of bees, such as employed bees, onlooker bees, and scout bees. The employed bees are associated with a specific food source. The onlooker bees observe the dance of employed bees in the hive to decide to choose a certain food source and conduct a local search for the selected food source. The scout bees randomly search for food to avoid local optimal solutions. The traditional ABC algorithm can effectively solve the optimization problem of a single objective function. It has the characteristics of a few parameters and fast convergence speed. However, according to the model (5), there are multiple rescue points. Each rescue point needs to make its optimization model as optimal as possible. The problem of flood control material scheduling optimization is a multiobjective optimization problem in which multiple rescue points compete with each other. Therefore, we introduce the Pareto nondominated sorting and propose a multiobjective scheduling algorithm of flood control materials based on Pareto artificial bee colony to solve the scheduling problem of flood control materials between rescue points. Then, we can obtain the nondominated sets and find the optimal solution in the nondominated set based on the model (5). Thus, we obtain the flood control material scheduling schemes that weigh disaster rescue points and flood control scheduling management in the case of multiple types of flood control materials and multiple disaster rescue points.

3.1. Food Source Initialization. Since the traditional ABC algorithm cannot be directly used to solve the scheduling problem, we precode the problem solution and uses a one-dimensional vector coding method with variable length to solve the flood control material scheduling decision problem. The food source code is shown in Figure 1.

Where the storage warehouse represents the storage location of flood control materials in various regions, according to the location of the rescue points and the demands of flood control materials in advance, the algorithm needs to select the storage warehouse set that can meet the flood control materials all rescue points. \( a1 \) represents the storage warehouse number 1. \( b1 \) represents the rescue point number 1. The numbers in the table represent the amount of certain flood control materials transported. A good initial solution set can enable the algorithm to quickly find the areas with great potential in the entire solution space and speed up the algorithm’s convergence. Then, it provides a variety of different types of food sources to avoid the algorithm from falling into a local optimum [23]. In the paper, we use the following three rules in the food source initialization phase.

1. **Large Storage Warehouse First.** Due to the different types of flood control materials stored in each storage warehouse, each flood control material is also different. According to the actual material storage quantity of the warehouse, flood control materials are distributed based on roulette rules in the same set. The warehouse with a large storage quantity can be a priority to assign scheduling tasks [24].

2. **Disaster Demand First.** Different rescue points have different demands for flood control materials. In the same set, according to the demand for flood control materials at each rescue point, flood control materials are preferentially distributed based on roulette rules to rescue points with high demand for materials.

3. **Random Distribution.** Flood control materials are distributed randomly.

The three rules mentioned above generate each initial solution. The solution set ratios are, respectively, set to 30%, 30%, and 40%. In the distribution process, each initial allocation value is the minimum of the warehouse material storage and material demand [25] to meet the constraint (1). Then, the food source is considered whether meets the constraint (2); if the constraint (2) is not met, the overall allocated materials are reduced to avoid repeated scheduling for materials. The food sources that met the constraints (1)–(2) are obtained.

3.2. Fitness Value Calculation. For urgency and fairness of flood control materials’ emergency scheduling, we establish a scheduling optimization model of flood control materials. In the phases of employed bees and onlooker bees, we need to calculate the fitness \( F_j \) of each rescue point and perform the operations such as niche and minimum dominating set selection, which is

\[
F_j = \frac{1}{(D_j/R_j)}. 
\]
When the algorithm is solved, according to the minimum dominating set, we obtained, calculating the overall evaluation value \( F_j \) of all food sources in its set, and select the food source with the maximum value as the optimal solution of MSA_PABC.

\[
F_j = D_{ave} \times \frac{x_1Std_w + x_2Rat_w}{R_{ave}}.
\]

### 3.3. Employed Bee Phase

According to the food sources generated in Section 3.1, employed bees divide the whole food source into subcategories according to the number of rescue points. After that, employed bees search for the unknown solution space near each food source by variable neighborhood local search operation to obtain multiple child food sources. Then, we analyze the relationship between the child’s food source and the father’s food source and obtain the next food source. Finally, we use the elite retention and exclusion strategy in the niche to restore the food source’s size to the original size. The niche operation and variable neighborhood local search operation are as follows.

#### 3.3.1. Niche Operation

Niche is a conception of biology. It refers to a living environment under a specific environment. In the biological evolution process, organisms always live with the same species and breed offspring together. Niche technology divides each generation of individuals into several categories and selects some individuals with greater fitness as an outstanding representative class to form a species, then uses the elite retention and exclusion strategy in the species and between different species to generate a new generation of individual groups.

For individual fitness value \( F_j \) mentioned in Section 3.2, MSA_PABC calculates the fitness value \( F_j \) for each food source and normalizes it. It classifies the food source into the subclass population with the highest fitness value \( F_j \) after normalization and divides the entire food source population into subcategories. For the maximum fitness value \( F_j \) within its child species, MSA_PABC performs the variable neighborhood search operation to enhance algorithm’s local optimization algorithm and obtains multiple new food sources. If the new food source dominates the old food source, the old food source is updated. Otherwise, if the old food source dominates the new food source, it deletes the new food source and keeps the old food sources. Then, it places the new food sources in the temporary food source set. After finishing the variable neighborhood search operation, MSA_PABC merges the food source set and the temporary food source set into the next generation of species.

At last, MSA_PABC calculates the number of food sources in the next generation. Suppose the number of food sources exceeds the initial size. In that case, it randomly selects multiple food sources in the population to form an exclusion member set. It calculates the similarity between other food sources and exclusion members in the next generation. It selects the current food source and calculates the number of the same value at each same code position of each exclusion committee. It chooses the maximum value as the same degree \( A_i \) of the food and calculates the difference sum of the values at the same code position of the food source. Then, it selects the maximum value as the difference degree \( B_i \) of the food source. According to the same degree, MSA_PABC sorts those food sources from small to large. If there are multiple food sources with the same degree, it sorts those food sources from small to large according to the different degree and obtains the sorted food sources. Then, it eliminates front foods in turn until the population returns to its original size. If the eliminated food source is the optimal local solution or the optimal global solution in the offspring population, it skips the food source and reserves the elite solution.

#### 3.3.2. Variable Neighborhood Local Search Operation

A variable neighborhood local search method is an improved local search method. It utilizes a neighborhood structure formed by different actions to perform an alternate search and achieves a good balance between concentration and evacuation. The maximum loop search threshold is set as the termination condition to keep the efficiency and the accuracy of the algorithm. To facilitate the search operation of variable neighborhood, the initial food source has to be transformed into a multidimensional real matrix \( Q_{bjr} \). As shown in Figure 2, where \( i \) represents the storage warehouse (row a). \( j \) represents the rescue point (subordinate columns b). \( v \) represents the type of flood control material. \( Q \) represents the quantity of flood control materials (the value in Figure 2). Suppose some storage warehouses do not have a certain amount of flood control materials. In that case, the \( Q \) value is the Nan identifier; then, MSA_PABC skips the \( N_{an} \) identifier automatically during the calculation to improve search efficiency. The Niche strategy divides the food sources into classes based on the number of rescue points (i.e., \( b_1, b_2, b_3, \ldots, b_n \)). For rescue point \( b_i \) among food sources belonging to the \( b_i \) category, it performs the variable neighborhood search to improve the resource scheduling optimization of the rescue point and the mutual competitiveness between the rescue points. Thus, it obtains some local optimal food sources. As shown in Figure 2, we assume that it belongs to the \( b_1 \) category. Then, MSA_PABC conducts a variable neighborhood search for the flood control material.
distribution at the rescue point \( b_1 \). The specific steps of the variable neighborhood search method are described as follows.

\[
Q_{\text{inv}} = Q_{ijv} + Q_{cwv} Q_{ijv} = 0, \forall i, \tag{11}
\]

where \( Q_{ijv} \) represents quantity \( Q_{ijv} \) of flood control material \( v \) from each storage warehouse to the rescue point \( b_j \) which redistributes them to the rescue point \( b_w \).

**Step 1.** According to the \( b_w \) category to which the food source belongs, the method determines the rescue point \( b_w \) that needs to be searched by the variable neighborhood search operation. Then, it initializes the maximum number of variable neighborhood searches and selects the first column of flood control materials in rescue point \( b_w \). The number of neighborhood searches is 0.

**Step 2.** Getting current flood control material \( v \), the method searches for other rescue points with the flood control materials and selects a rescue point \( b_j \) randomly.

**Step 3.** The method finds the quantity \( Q_{ijv} \) of flood control material \( v \) from each storage warehouse to the rescue point \( b_j \) and redistributes them to the rescue point \( b_w \). The specific formula is as follows:

\[
\text{If } \sum Q_{\text{inv}} > j \text{, the current quantity of flood control materials } v \text{ which are allocated to rescue point } j \text{ is too large, and the excessive flood control materials have to return to rescue point } b_j. \text{ In order to reduce the transportation distance of flood control materials, we set } L = \sum Q_{\text{inv}} - j. \tag{12}
\]

**Step 4.** The method selects a nonzero value randomly among \( Q_{\text{inv}}, \forall i \). If \( Q_{\text{inv}} \geq L \), then \( Q_{\text{inv}} = Q_{\text{inv}} - L, Q_{ijv} = Q_{ijv} + L \) and skip to step 5. Otherwise, \( Q_{\text{inv}} = 0, Q_{ijv} = Q_{cwv} \), \( L = L - Q_{ijv} \) and return to step 4.

**Step 5.** If the method has completed the search for variable neighborhoods of all flood control materials in the rescue point \( b_j \), it skips stepping 6. Otherwise, it selects the next flood control material and skips to step 2.

**Step 6.** If the method obtains a child food source and the number of variable neighborhood searches is smaller than the maximum number of variable neighborhood searches, it reselects the first column of flood control materials in the rescue point \( b_w \); then, the number of variable neighborhood searches adds 1 and skips to step 2. Otherwise, it ends and outputs multiple new food sources.

As shown in Figure 2, we assume that the demands of sandstone \( b_1 \) are 20, and the demands of sandstone \( b_3 \) are 30. Then, \( v = 1 \) and it only represents the sandstone. There is no sandstone in \( b_2 \), and \( b_3 \) has sandstone; thus, \( Q_{011} = [2, 4, 3] \). The subscript 0 represents all of the storage warehouses \( a_1, a_2, \) and \( a_3 \). We randomly select the value of sandstone material \( b_3 \) as \( Q_{032} = [2, 9, 12] \). First, we set \( Q_{111} = 2 + 3 = 5, Q_{211} = 4 + 9 = 13, \) and \( Q_{111} = 3 + 2 = 5 \). Due to the current number of sandstone in \( b_1 \) exceeds its demands, it does not meet the constraints (1) and (2). Then, we randomly select the storage warehouse and return the remaining value to the rescue point \( b_j \). Finally, we operate on each flood control material in \( b_1 \), so as to realize variable neighborhood search operation and output new food sources.

3.4. Onlooker Bees Phase. MSA_PABC calculates the fitness value \( F_i(i) \) of each rescue point for every food source by formula (9) and obtains a \( N_s \times N_j \) data matrix \( M \). Among them, \( F_i(i) \) represents the fitness value of rescue point \( j \) in food source \( i \). \( N_s \) represents the number of food sources. \( N_j \) represents the number of rescue points. According to each rescue point’s fitness value, it sorts each column of data in the matrix \( M \) to obtain the neighbor individuals \( i + 1 \) and \( i - 1 \) of each element in the matrix \( M \) and their fitness values. Then, it calculates the crowdedness through the Euclidean distance value between individual \( i + 1 \) and individual \( i - 1 \). That is, it calculates the sum of the differences in the individual fitness value \( F_j \) for each target. The solution with the maximum and minimum values is specified as the infinite distance, that is, the boundary solution \( d_1 = d_y = \infty \).

\[
d_i = \begin{cases} \infty, & \text{if } y = 1 \text{ and } y \\ \sum_{j=1}^{N_j} \frac{|F_j(i + 1) - F_j(i - 1)|}{F_j^{\text{max}} - F_j^{\text{min}}}, & 1 < i < y \end{cases},
\tag{12}
\]

where \( d_i \) represents the crowdedness of the \( i \)th food source, \( y \) represents the maximum boundary index, \( F_j^{\text{max}} \) represents the maximum fitness value of rescue point \( j \) among all food sources, and \( F_j^{\text{min}} \) represents the minimum fitness value of rescue point \( j \) among all food sources. In the solution set, the part of solutions concentrates in a certain area and distributes sparsely in some areas; therefore, according to the crowdedness of food sources, MSA_PABC sorts the solutions in descending order to obtain the spatial measure of food sources with surrounding neighbors. The selected probability of the food source is calculated by the formula (13).

\[
p_i = \begin{cases} \frac{d_i}{\sum d_i}, & \text{if } d_i \neq \infty \\ 0, & d_i = \infty \end{cases},
\tag{13}
\]
where \( p_{\text{con}}^r \) represents the selected probability of the \( i \)th food source in the solution set. According to the measurement results, the crossover operation is performed on the entire population, the search space is expanded, and the number of solutions at each dominating level and the Pareto curve are increased to provide more choices for decision-making. Namely, MSA_PABC sets the crossover probability \( r \) in advance and cyclically implements the following operations until finishing the crossover of each type of flood control materials: in the range \([0,1]\), it randomly generates a floating-point number \( r \). If \( r \leq r \), it will cross the \( k \)th element of the two selected food sources. Otherwise, it does not change the amount of flood control materials in the food source. After finishing the cross operation, it is necessary to verify whether the newly generated food sources satisfy the constraints (1) and (2). If it does not, the overallocated materials will be cut to obtain the new food sources which meet the constraints. Then, new food sources add to contemporary populations.

MSA_PABC implements the retain Pareto nondominated strategy by formula (12), that is, it compares the fitness value of each rescue point of any two food sources. If the fitness value of each rescue point in one food source is greater than or equal to the fitness value of each rescue point in another food source, and there is at least the fitness value of the corresponding rescue point in another food source, it represents that the food source dominates another food source. Therefore, it can obtain a set of nondominated solution sets \( M_F \) that are not dominated by other solutions.

\[
F_{j}(\kappa) \geq F_{j}(\lambda), \quad \exists F_{j}(\kappa) > F_{j}(\lambda), \quad j = 1, \ldots, N_f, \quad (14)
\]

where \( \kappa \) and \( \lambda \) represent the two food sources in the entire set of solution spaces and \( F_{j}(\cdot) \) represents the fitness value of the \( j \)th rescue point of the food source. If the number of food sources in the solution set exceeds the initial scale, it sorts the food sources in the population in descending order by the overall evaluation value \( F_z \) and eliminates the poor food source solutions to restore the population to its original scale. Otherwise, it will add the initialized food sources to bring it back to the original scale.

3.5. Scout Bee Phase. After completing the above two phases, scout bees select the food source in the last certain proportion of the food source population and implement the initialize operation in Section 3.1 for the same number of times to initialize the food source. Then, it adopts the greedy strategy to update old food sources. Because the entire food source population may have food sources with the same fitness value, it chooses the duplicate and redundant food source and the food source whose fitness value does not change after the maximum number \( U \) of evolution iterations. Then, it uses the food source initialization operation in Section 3.1 to initialize the food source to ensure the diversity of the entire population and prevent it from falling into a local optimum.

4. Algorithm Implementation

As shown in Figure 3, we mainly use the MSA_PABC to solve the model. In the initialization phase of the algorithm, it is necessary to set the model’s data and the initial parameters of the algorithm. The specific steps are as follows:

Step 1. MSA_PABC obtains each warehouse point \( a \) and rescue point \( b \) in the city and obtains the number of types of materials and the number of specific materials in each warehouse. Then, it sets the initial data, such as the total number of food sources in the population, the maximum number of evolution \( U \), the maximum number of local searches \( m1 \) in the variable neighborhood, and the number of onlooker bees \( m2 \).

Step 2. According to the principle of large-scale storage warehouses first, disaster demands first and randomly distribution, MSA_PABC generates the initial food source, and the proportion of food sources is 30%, 30%, and 40%. Then, it modifies the food source by reducing the excessively distributed materials so that all food sources satisfy the constraints (1) and (2).

Step 3. MSA_PABC calculates the individual fitness value \( F_j \) and overall optimal evaluation value \( F_z \) of food sources.

Step 4. MSA_PABC enters the employed bee phase. According to the number of rescue points, it divides the entire food source population into \( N_f \)-independent child populations by the niche technology. It implements the variable neighborhood search operation to find a locally optimal solution for each food source. Thereby, it obtains \( m1 \) food sources. If the new food source dominates the old food source, then it updates the old food source. If the old food source dominates the new food source, then it deletes the new food source. Otherwise, it temporarily stores new food sources without changing the old food source and puts them in temporary food source collections. Completing the variable neighborhood search operation merges the food source set with the temporary food source set. Subsequently, it eliminates similar food sources through the elite retention strategy and the niche exclusion strategy to restore the food source collection’s size to its original size.

Step 5. MSA_PABC enters the onlooker bee phase. Calculating the crowdedness among food sources by formula (11) to determine the selected probability of each food source, it implements the following operation for \( m2 \) times and obtains \( m2 \times 2 \) child food sources: it chooses a food source based on the rules of roulette and selects the neighbor food source with the nearest European distance; then, it performs across the operation to generate two food sources and corrects the new two food sources to obtain two-child food sources satisfying the constraints (1) and (2). The \( m2 \times 2 \) child food sources and the food sources in the current population combine into a food source set, and MSA_PABC performs the Pareto dominance strategy on the food source set to obtain a nondominated solution set. According to the overall evaluation
optimal value $F_z$, it eliminates the poor food source solution so that the population number returns to the original size.

Step 6. MSA_PABC enters the scout bee phase. It implements the final elimination method to rerandomly initialize a certain proportion of the food sources at the end of the ranking. Then, it uses the greedy strategy to update the old food sources. After $U$ iterations, if the food source’s overall value does not change, it reinitializes the food source. Simultaneously, the current population retains and updates the new global optimal solution and historical optimal solution based on the greedy strategy.

Step 7. MSA_PABC determines whether the number of loop iterations is equal to the maximum number of iterations. It outputs a nondominated solution set and outputs the optimal solution for an overall evaluation. Otherwise, it goes to step (3).

5. Algorithm Simulation

5.1. Simulation Parameters Selection. To test and verify the performance of MSA_PABC, as shown in Figure 4, we adopt the factual information of all storage warehouses and flood control materials in the management information system of flood control materials and rescue teams in Zhejiang Province of China developed by Zhejiang Yugong Information Technology Co., Ltd. Then, according to the actual disaster situation, it randomly generates disaster scenarios of different rescue point locations and required flood control materials. Using the simulation parameters shown in Table 1, we carry out simulation experiments in simulated disaster scenarios. We analyze the distribution of nondominated solution sets and the convergence of MSA_PABC, the effects of the number of food sources, crossover probability, the maximum number of evolutions, and end elimination ratio on the optimal solution evaluation value in 10 different disaster scenarios. Then, we calculate the average maximum transportation distance, the average material satisfaction rate of rescue points, the standard deviation of maximum transportation distance, the standard deviation of satisfaction rate, and the number of earliest completed convergence iterations of MSA_PABC, ABC (artificial bee colony), NSGA2 (nondominated sorting genetic algorithm 2), and MOPSO (multiobjective particle swarm optimization) in 10 different disaster scenarios. We take the average value as the simulation result value. Among them, ABC utilizes the traditional artificial bee colony algorithm to solve the optimization model (8) and obtain the optimal solution. NSGA2 carries out the fast nondominated sorting strategy with elite retention. It selects the solution with the most extensive evaluation value as the final solution in the nondominated solution set. MOPSO realizes the scheduling optimization of flood control materials through the second set and adaptive grid method and selects...
the final resolution according to the evaluation value. The average maximum transport distance is the average of the maximum transport distance of all rescue points. The intermediate material satisfaction rate is the average of the material satisfaction rates of all rescue points. The earliest number of iterations to complete convergence is the minimum number of iterations required when the algorithm converges to the optimal value. The standard deviation of the maximum transport distance, the standard deviation of the satisfaction rate, and the evaluation value of the optimal solution are calculated by formulas (6), (7), and (10), respectively.

5.2. Simulation Result Analysis

5.2.1. Nondominated Solution Set and Convergence Analysis.

We choose a disaster scene randomly and selects the number of rescue points as 10 and other parameters in Table 1 to calculate the nondominated solution set of MSA_PABC and analyze its convergence. When the number of rescue points is greater than 2, the rescue points compete, and their fitness values are as optimal as possible. Since the number of rescue points is 10 and it has more spatial dimensions, for the convenience of data display, we divide the rescue points into 2 groups and calculate the normalized fitness value sum of the 2 groups in the nondominated set. Then, it analyzes the Pareto distribution of those 2 groups of rescue points. As shown in Figure 5, the scheduling problem of flood control materials is the distribution problem of rescue points and warehouses. Therefore, it is a combination of multiple distribution schemes and is discrete. Its optimal Pareto nondominated solution set does not present continuous dense data points and only offers a distribution of 10 points. However, its data points achieve the optimal fitness value without reducing the fitness values of another group and still show the Pareto curve's characteristics. As shown in Figure 6, MSA_PABC uses a niche method and variable neighborhood local search method in the employed bee operation to improve the convergence ability and ensure that the offspring can optimally inherit the current optimal food source as far as possible. In the onlooker bee phase, MSA_PABC uses a crossover strategy to enhance global search capability. In the scout bee phase, MSA_PABC sets the bee colony’s maximum evolution threshold and uses an end elimination mechanism to update food sources with low evaluation values and delete food sources that cannot be evolved again. It expands the search capability, and the food source can grow faster in a favorable direction. Simultaneously, through the calculation of congestion and the maintenance of Pareto nondominated sets, MSA_PABC can quickly find and retain the optimal food source in history and improve its convergence accuracy. Therefore, MSA_PABC can discover the optimal evaluation value of the food source after 24 iterations.

5.2.2. The Influence of Parameters on MSA_PABC

(1) The Influence of the Number of Food Sources on MSA_PABC.

We select the number of food sources 20, 30, 40, 50,
The number of rescue points 2, 3, 4, 5, 6, and other parameters in Table 1 calculate the average evaluation value of output solution in 10 disaster scenarios with different materials and analyze the influence of the number of food sources.

As shown in Figure 7, when the number of rescue points is 2, MSA_PABC can converge to the current optimal solution and obtain the optimal solution when solving the low-dimensional answer of flood control material scheduling. The evaluation values of the output solutions under different numbers of food sources remain the same. However, with the increase in the number of rescue points, the data dimension increases. When the number of food sources is small and the number of algorithm iterations is limited, in the process of searching for the optimal solution, the output solution is easy to fall into the optimal local solution, and the maximum fitness value of the output solution is more considerable. Therefore, under different numbers of rescue points, when the number of food sources is 20, the output solution’s evaluation value is the largest. As the number of food sources increases, the evaluation value of the output solution becomes smaller. When the numbers of food sources are 50 and 60, MSA_PABC can obtain the current optimal solution. The evaluation value of the output solution is the smallest, and the difference is not large. Therefore, we choose 50 food sources for a simulation experiment.

(2) The Influence of Crossover Probability on MSA_PABC.

We choose the crossover probability 0.1, 0.2, 0.3, 0.4, 0.5, the number of rescue points 2, 3, 4, 5, 6, and other parameters in Table 1 analyze the crossover probability’s influence on the evaluation value of output solution.

As shown in Figure 8, when the number of rescue points is 2, MSA_PABC has better convergence when solving multi-objective low-dimensional data. They converge to the optimal solution under different crossover probability, and their evaluation values of output solution remain consistent. In the explanation of high-dimensional discrete problems of flood control materials, the low cross probability is difficult to achieve the goal of crossover, and the small number of food sources in the nondominated set and the high cross probability destroy the overall quality of the food source, which makes the algorithm more tend to be random. It also reduces the convergence of MSA_PABC and converges to the optimal local solution within a limited number of iterations. Therefore, under the number of rescue points, when the crossover probability is 0.2, the output solution’s evaluation value is the smallest. When the crossover probability is 0.1, its evaluation value of the output solution is second. As the crossover probability increases after 0.2, the output solution’s evaluation value becomes more extensive. Thus, we choose the crossover probability 0.2 for a simulation experiment.
The Influence of the Maximum Number of Evolutions on MSA_PABC. We select the maximum number of evolutions 1, 2, 3, 4, 5, and the number of rescue points 2, 3, 4, 5, 6, and other parameters in Table 1 analyze the influence of the maximum number of evolutions on the evaluation value of output solution.

As shown in Figure 9, although a minimal evolution threshold will improve the searchability of the solution space, it is a high probability to prevent high-quality solutions from being effectively inherited to the next generation of populations. It destroys the convergence effectiveness of MSA_PABC. However, a considerable evolution threshold will decrease the solution space’s searchability as the number of algorithm iterations increases. Therefore, under different numbers of rescue points, when the maximum number of evolutions is 2, its evaluation value of the output solution is the smallest. As the maximum number of evolutions increases from 2, the output solution’s evaluation value becomes more extensive, so we choose the maximum number of evolutions 2 for the simulation experiment.

The Influence of the End Elimination Ratio on the MSA_PABC. We select the different end elimination ratios 0.05, 0.1, 0.15, 0.2, and 0.25, the number of rescue points 2, 3, 4, 5, 6, and other parameters in Table 1 analyze the influence of different end elimination ratios the evaluation value of output solution.

As shown in Figure 10, the too small end elimination ratio is not conducive to eliminate the part inadequate solutions and reduce the convergence speed. And it converges to the optimal local solution within a limited number of iterations. The too large end elimination ratio will cause many solution sets to be reinitialized, thereby reducing the food source’s overall quality. Therefore, under different numbers of rescue points, when the end elimination ratio is 0.2, the output solution’s evaluation value is the smallest. When the end elimination ratio is less than 0.2, its evaluation value of the output solution becomes larger as the end elimination ratio decreases. When the end elimination ratio is greater than or equal to 0.2, the evaluation value of the output solution becomes larger. So we choose the end elimination ratio of 0.2 for the simulation experiment.

Algorithm Performance Comparison. In Section 5.2.2, when we choose the number of food sources 50, the crossover probability 0.2, the maximum number of evolutions 2, and the end elimination ratio 0.2, we can find the optimal solution of MSA_PABC, and the evaluation value of the output solution is the smallest. Therefore, we choose those parameters, the number of rescue points 2, 3, 4, 5, 6, respectively, the corresponding material types 4, 13, 18, 23, 40, and the parameters in Table 1 calculate the average maximum transportation distance, the average material satisfaction rate of rescue points, the standard deviation of maximum transportation distance, the standard deviation of satisfaction rate, and the number of earliest completed convergence iterations in MSA_PABC, ABC, NSGA2, and MOPSO under 10 randomly generated different disaster scenarios. The average values are used as the simulation results.

As shown in Figures 11 and 12, no matter how the number of rescue points changes, the average maximum distance of MSA_PABC is lower than that of ABC, NSGA2, and MOPSO. The material satisfaction rate of MSA_PABC is larger than that of ABC, NSGA2, and MOPSO. That is because MSA_PABC uses the transportation distance and material satisfaction rate as the fitness parameters for calculating each rescue point and takes the optimal fitness values of multiple rescue points as a multiobjective problem. To solve the multiobjective problem, MSA_PABC uses niche technology to group the offspring in the employing bee
phase. MSA_PABC also maintains the diversity of the population and improves the generation efficiency of Pareto non-dominated solutions. Then, MSA_PABC uses variable neighborhood search operation and crossover operation to improve the algorithm’s global search capabilities and local update capabilities and ensure the solution’s quality. It uses the Pareto control strategy to avoid falling into the optimal local solution. Therefore, MSA_PABC can find the optimal solution within the maximum number of iterations of 50.

As shown in Figures 13 and 14, regardless of the number of rescue points, the standard deviation of the maximum transportation distance and material satisfaction rate of MSA_PABC are lower than those of ABC, NSGA2, and MOPSO. That is because MSA_PABC solves the multiobjective problem of competition among rescue points through a modified Pareto artificial bee colony algorithm to find a non-dominated set covering multiple schemes. In the nondominated location, MSA_PABC takes the standard deviation of the maximum transport distance and the material satisfaction rate as parameters of the overall evaluation value. It balances the maximum transportation distance among rescue points and the secular satisfaction rate and achieves the rescue points’ fairness. However, ABC only considers the comprehensive indicators of flood control material schedule but does not consider the right among rescue points, resulting in a higher standard deviation of the maximum transportation distance and material satisfaction rate. NSGA2 and
MOPSO use the Pareto strategy to solve the multiobjective problems; their standard deviations are smaller than ABC but are significantly worse when solving the high-dimensional issues, resulting in a larger standard deviation than that of MSA_PABC.

As shown in Figure 15, regardless of the number of rescue points, the number of the earliest completed convergence iterations of MSA_PABC is lower than that of ABC, NSGA2, and MOPSO. This is because MSA_PABC uses multistrategy food source initialization in the initialization phase and uses the niche technology, variable neighborhood search operation, and crossover operation in the employed bee phase. It uses the Pareto nondominated sorting, crossover operation, and other operations in the onlooker bee phase. It uses the end elimination mechanism and evolution threshold in the scout bee phase. Therefore, MSA_PABC can find the optimal solution in a short number of iterations, and its number of the earliest completed convergence iterations is the smallest. ABC directly finds better food sources nearby to update the optimal evaluation value. Its search ability is low, and it takes a long time to converge. It is also easy to fall into the optimal local solution. Although NSGA2 and MOPSO can search and approach the Pareto boundary, their traditional crossover, mutation, and other operations have low search efficiency, resulting in slow convergence. Therefore, their numbers of earliest completed convergence iteration are higher than that of ABC.

6. Conclusion

In the paper, we propose a multiobjective scheduling algorithm of flood control materials based on the Pareto artificial bee colony. First of all, considering the constraints, such as distance constraint from multiple storage warehouses to multiple rescue points, storage capacity constraint of warehouse materials, and material requirement constraint, we establish the scheduling optimization model of flood control materials for each disaster rescue point and the total scheduling optimization model of all flood control materials. Then, we use the modified Pareto artificial bee colony algorithm for solving the multiobjective model. That is, we propose a variety of food source initialization strategies, fitness value calculation of food sources, the employed bee operation including niche technology and variable neighborhood local search, onlooker bee operation including Pareto dominates sorting, crossover operations, and scout bee operation.
including maximum evolution threshold and end elimination mechanism. Through the food source initialization strategies, we obtain the nondominated solution set under the scheduling problem of multiple disaster sites and various flood control materials and obtain an optimal solution from the nondominated solution set. Finally, we analyze the distribution of nondominated solution sets and the convergence of MSA_PABC and analyze the influence of the number of food sources, crossover probability, maximum evolution times, and end elimination ratio on the evaluation value of output solution in MSA_PABC. Finally, we compare the average maximum transportation distance, the average material satisfaction rate of rescue points, the standard deviation of maximum transportation distance, the standard deviation of satisfaction rate, and the number of earliest completed convergence iterations in MSA_PABC, ABC, NSGA2, and MOPSO.

The simulation results show that no matter how many rescue points change, MSA_PABC can quickly find the nondominated sets and optimal solutions. It also improves the convergence rate and the material satisfaction rate and reduces the average maximum transportation distance of flood control materials, the standard deviation of the maximum transport distance, and the standard deviation of material satisfaction rate. However, the time complexity of MSA_PABC is high, so in the future, we aim to use the heuristic algorithm to solve the optimal model and find the optimal solution, which reduces the calculation time of the algorithm.

Data Availability

The data used to support the findings are available upon the authors’ reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Public Welfare Technology Application and Research Projects of Zhejiang Province of China under Grant No. LGF19F010006 and Grant No. LGG20F010009, the Natural Science Foundation of Zhejiang Province of China under Grant No. LQ18F030006, and the Project Intelligentization and Digitization for Airline Revolutions #2018R02008.

References


Wireless Communications and Mobile Computing


