Reducing mineral processing water costs and freshwater consumption is a challenging task in the mineral processing water distribution (MPWD). The work presented in this paper focuses on two aspects of the MPWD optimization model and the MPWD optimization method. To achieve MPWD optimization effectively, a nonlinear constrained multiobjective model is built. The problem is formulated with two objectives of minimizing the mineral processing water costs and maximizing the amount of recycled water. In this paper, an optimization method named enhancing the multiobjective artificial bee colony (EMOABC) algorithm is proposed to solve this model. The EMOABC algorithm uses four strategies to obtain the Pareto-optimal solutions and to achieve the MPWD optimal solutions. With the three benchmark functions, the EMOABC algorithm outperforms the other two widely used algorithms in solving complex multiobjective optimization problems. The EMOABC algorithm is then applied to two cases. Results have shown that the proposed algorithm has the ability to solve the MPWD optimization model. The developed model and the proposed algorithm provide decision support for the actual MPWD problem.

1. Introduction

Mineral processing is the process of obtaining the raw materials for smelting and consumes large amounts of water. For water distribution, the decision-makers in mineral processing plants need to make decisions concerning different objectives, such as water costs and freshwater minimization, to maximize the comprehensive benefits of mineral processing plants. The solution to the multiobjective optimization problem often results from both an optimization model and an optimization algorithm. Thus, this paper focuses on solving mineral processing water distribution (MPWD) optimization problems from these two perspectives.

Over the years, many works have examined the water distribution optimization problem and presented many mathematical models and conventional methods such as linear programming [1, 2], nonlinear programming [3], and integer linear programming [4], which have been applied to solve the problem. However, these methods suffer from severe limitations in handling discreteness, nonlinearity, complex constraints, and local convergence [5]. To overcome the limitations of classical optimization methods, the heuristic methods have been proposed to solve water distribution optimization, such as simulated annealing (SA) algorithms [6, 7], genetic algorithms (GA) [5, 8, 9], differential evolution (DE) algorithms [10–12], particle swarm optimization (PSO) [13–15], ant colony optimization (ACO) [16–18], strength Pareto evolutionary algorithm (SPEA) [19], and shuffled frog leaping algorithm (SFLA) [20, 21]. Despite the success of these heuristic methods in certain aspects, they have not been applied to solve MPWD problems. Currently, MPWD problems are mainly based on manual scheduling. Therefore, building MPWD model and applying heuristic methods to solve the model represent an important issue of our current concern.

Heuristic methods have been widely used in various fields [22–25]. The artificial bee colony (ABC) algorithm is a relatively new heuristic method that has been proven to be an excellent competitor in multiobjective problems [26]. Several multiobjective ABC algorithms (MOABCs) have been proposed for multiobjective problems [27–29].
However, compared to the huge in-depth studies of other heuristic methods, such as nondominated sorting genetic algorithm II (NSGA-II) [30] and multiobjective particle swarm optimization (MOPSO) [31], on multiobjective problems, improving the performance of MOABCs is still a challenge.

MPWD is a complex process in which interrelated factors are multiple. In addition to meeting the total amount of mineral processing water, many factors, such as the flow of the water source, the pressure of the water source, and maximizing the use of recycled water, need to be considered. The traditional linear model is difficult to describe the relationship between these complex factors for MPWD with the above characteristics and cannot treat multiple constraints appropriately.

The purpose of this paper is to build a nonlinear and multiobjective model for MPWD and to develop an MOABC algorithm for solving the MPWD optimization problem. The MPWD optimization model has two objectives: one objective is to minimize mineral processing water costs, and the other objective is to maximize the amount of recycled water. The proposed MOABC algorithm called multiobjective artificial bee colony (EMOABC) algorithm integrated multiple strategies to improve its performance. Therefore, some of the contributions of this paper are as follows:

1. The developed model for MPWD is the nonlinear model that has two objective functions
2. The proposed EMOABC algorithm can improve the population diversity and obtain better global convergence

The rest of the paper is organized as follows: In Section 2, the MPWD optimization model is built. Section 3 first gives a review of the basic ABC algorithm and then proposes the EMOABC and presents the results of the EMOABC algorithm compared with other algorithms on three widely used benchmark functions. In Section 4, the implementation of the EMOABC on the MPWD optimization problem is presented. Section 5 outlines the conclusions.

2. MPWD Model

According to categories of water source, the mineral processing water can be divided into fresh water, tailings water, clean water, and in-plant water, and they are from rivers and lakes, tailings reservoirs, wastewater treatment plants, and in-plant recycled systems, respectively (Figure 1). The cost of using the tailings water, the clean water, and in-plant water is relatively high for proper treatment and long-distance transmission before entering into the mineral processing plant. Although the cost of using fresh water is very low due to no treatment required, it is important to minimize the use of fresh water for environmental reasons. Mineral processing water becomes wastewater after mineral processing. The MPWD problem in this paper is to optimize the following competing objective functions simultaneously, minimizing the mineral processing costs and maximizing the amount of recycled water including tailings water, clean water, and in-plant water, while satisfying several equality and inequality constraints. Generally, the problem is formulated as follows:

\[
Q = (q_1, q_2, \ldots, q_n),
\]
\[
Q_1 = (q_1, q_2, \ldots, q_n),
\]
\[
Q_2 = (q_{a+1}, q_{a+2}, \ldots, q_b),
\]
\[
Q_3 = (q_{b+1}, q_{b+2}, \ldots, q_c),
\]
\[
Q_4 = (q_{c+1}, q_{c+2}, \ldots, q_n).
\]

2.1. Decision Variable. The vector of the mineral processing water is expressed as \( Q \), including the freshwater as \( Q_1 \), tailings water as \( Q_2 \), clean water as \( Q_3 \), and in-plant water as \( Q_4 \). The mineral processing water except freshwater is called the recycled water. The vector of the recycled water is expressed as \( Q' = [Q_1, Q_2, Q_3, Q_4] \). \( q_i \) is a flow of the \( i \)th water source; \( a, b, c, \) and \( n \) are positive integers, and \( n \) is the number of water sources. Decision variables are defined as follows:

\[
C(Q) = \sum_{i=1}^{n} c_i q_i (100 p_i + h_i),
\]

where \( h_i \) is the elevation difference of the \( i \)th water source; the integer 100 is used for unit conversion. \( p_i \) is the pressure of the \( i \)th water source. Equation (2) also can be transformed into

\[
C(Q) = \sum_{i=1}^{n} c_i q_i \left[ 100 f(q_{\text{sum}}, q_i, q_j) + h_i \right],
\]

where \( f(q_{\text{sum}}, q_i, q_j) \) is the macro model of the water supply network. This \( q_{\text{sum}} \) is the total flow of mineral processing water required in the mineral processing plant.

The yields of freshwater are low, but its prices are high. So it is important for cost reduction that the recycled water is used in mineral processing. While the total water consumption of mineral processing is met, the recycled water should be used in mineral processing as much as possible. The amount of the recycled water used in mineral processing can be expressed as

\[
M(Q') = M(q_{a+1}, q_{a+2}, \ldots, q_b) = \sum_{i=a+1}^{b} \alpha_i q_i + \sum_{i=b+1}^{c} \beta_i q_i + \sum_{i=c+1}^{n} \chi q_i,
\]

where \( \alpha \) is the penalty factor of tailings water; \( \beta \) is the penalty factor of clean water; and \( \chi \) is the penalty factor of in-plant water.
2.3. Constraint Condition. The sum of the flow from the water source should be equal to the total flow of mineral processing water required in mineral processing:

\[ \sum_{i=1}^{n} q_i = q_{\text{sum}}. \]  

(5)

Each water source should be between the maximum and minimum water supply capacity:

\[ q_{i_{\text{min}}} \leq q_i \leq q_{i_{\text{max}}} \quad i = 1, 2, \ldots, n, \]  

(6)

where \( q_{i_{\text{max}}} \) and \( q_{i_{\text{min}}} \) are upper and lower limits for the flow of the \( i \)th water source.

To ensure efficient transmission, the output pressure of each water source should be restricted as follows:

\[ p_{i_{\text{min}}} \leq p_i \leq p_{i_{\text{max}}} \quad i = 1, 2, \ldots, n, \]  

(7)

where \( p_{i_{\text{max}}} \) and \( p_{i_{\text{min}}} \) are upper and lower limits for the pressure of the \( i \)th water source. Equation (7) also can be transformed into:

\[ p_{i_{\text{min}}} \leq f(q_{\text{sum}}, q_i, q_i') \leq p_{i_{\text{max}}} \quad i = 1, 2, \ldots, n. \]  

(8)

To ensure the pressure requirements of the mineral processing plant, there should be a lower limit for the pressure at the control point:

\[ c_{p_{i}} \geq c_{p_{i_{\text{min}}}} \quad i = 1, 2, \ldots, n, \]  

(9)

where \( c_{p_{i}} \) is the pressure at the control point of the \( i \)th water source; \( c_{p_{i_{\text{min}}}} \) is lower limit for the pressure at the control point of the \( i \)th water source. Equation (9) also can be transformed into:

\[ f'(q_{\text{sum}}, q_i) \geq c_{p_{i_{\text{min}}}} , \quad i = 1, 2, \ldots, n, \]  

(10)

where \( f'(q_{\text{sum}}, q_i) \) is the macro model of the pressure control point.

2.4. Final Model. In summary, the water optimization model for the mineral processing plant is a two-objective optimization model and its final model is expressed as follows:

\[
\begin{align*}
\text{min} & \{C(Q), \max \{M(Q')\}, \\
\text{s.t.} & \begin{cases} \\
\sum_{i=1}^{n} q_i = q_{\text{sum}}, \\
q_{i_{\text{min}}} \leq f_j \leq q_{i_{\text{max}}} , \quad i = 1, 2, \ldots, n, \\
p_{i_{\text{min}}} \leq f(q_{\text{sum}}, q_i, q_i') \leq p_{i_{\text{max}}} , \quad i = 1, 2, \ldots, n, \\
f'(q_{\text{sum}}, q_i) \leq c_{p_{i_{\text{min}}}} , \quad i = 1, 2, \ldots, n. \\
\end{cases}
\end{align*}
\]  

(11)

3. Enhancing Multiobjective Artificial Bee Colony Algorithm

With the above model and constraints, the appropriate optimization algorithm is applied to optimize the MPWD problem. In this study, an EMOABC algorithm was proposed to solve the developed MPWD model.

3.1. Basic ABC Algorithm. ABC algorithm is a swarm intelligence optimization method proposed by Karaboga to simulate the process of bees searching for nectar [32]. In the framework of the artificial bee colony algorithm, the location of the food source represents the possible candidate solutions of the problem to be optimized, the number of nectars in the food source represents the fitness (objective function value) corresponding to the candidate solutions, and the size of the bee colony is equal to the number of solutions. The basic structure of the ABC algorithm can be divided into the four following stages.

The first is initial stage, where ABC algorithm randomly generates a matrix \( X = [x_i]_{N \times D} \), \( N \) is the population size and represents the number of candidate solutions, and \( D \) is the number of solution parameters, \( i \in (1, 2, \ldots, N) \), \( j \in (1, 2, \ldots, D) \). \( D \)-dimensional vector \( x_i = [x_{i1}, x_{i2}, \ldots, x_{iD}] \) is one of the candidate solutions. \( x_i \) are randomly generated by:

\[ x_{ij} = x_{ij}^{\text{min}} + \text{rand} \left( x_{ij}^{\text{max}} - x_{ij}^{\text{min}} \right), \]  

(12)

where \( x_{ij}^{\text{max}} \) and \( x_{ij}^{\text{min}} \) are the upper and lower bound constraints of the \( j \)th variable of \( X_i \), respectively, and \( \text{rand} \) is a uniformly distributed random number within the range \([0, 1]\).

The second is employed bee stage, where an employed bee is to modify and generate a new candidate solution based on the currently collected information, as well as to calculate its fitness. If the fitness of the new candidate solution is higher than the fitness of the original candidate solution, the new candidate solution replaces the original candidate solution; otherwise, the original candidate solution is retained. After all employed bees complete the search task, they share the information on the location of the food source (candidate solution) and the amount of nectar (fitness) they found with onlooker bees. The new candidate solution is generated by:

\[ v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}), \]  

(13)

where \( k \) and \( j \) are randomly chosen indexes, \( k \neq i \), \( \varphi_{ij} \) is a random number in the range \([-1, 1]\). The fitness of a solution can be defined as:

\[
\begin{align*}
\text{fit}(X_i) &= \begin{cases} \\
\frac{1}{1 + f(X_i)}, & \text{if } f(X_i) > 0, \\
1 + |f(X_i)|, & \text{if } f(X_i) \leq 0.
\end{cases}
\end{align*}
\]  

(14)

The third is onlooker bee stage, where onlooker bees assess the fitness of candidate solutions passed from the employed bee and then select food sources based on the probability corresponding to the amount of nectar. In this case, the onlooker bee updates the candidate solution based on the original candidate solution and calculates its fitness; if the fitness of the new candidate solution is higher than the original candidate solution, the new candidate solution replaces the original candidate solution; otherwise, the original candidate solution is retained. The probability of a food source chosen by an onlooker bee can be calculated by
\begin{equation}
    P_i = \frac{\text{fit}(X_i)}{\sum_{j=1}^{N} \text{fit}(X_j)} \tag{15}
\end{equation}

The fourth is scout bee stage; if the nectar of the discovered food source is too inferior to find better solutions several times in a row ("limit" is denoted as the maximum number of times), it will be replaced by a new food source that a scout bee randomly found, and then the new food source is generated by (12).

3.2. Proposed Method. This section presents a detailed description of the proposed EMOABC algorithm. It can be observed from the literature that this approach to mineral processing water has not yet been extensively explored.

Inspired by the idea of NSGA-II algorithm [30] and DNSPSO algorithm [33], in this paper, the EMOABC is integrated with a series of multiobjective optimization strategies. It includes four important multiobjective strategies: noninferior ranking strategy, Pareto-optimal strategy, crowding distance strategy, and search strategy to evaluate the food source location multiobjectively and select the nondominated solution, so they are presented as follows:

1) Noninferior ranking strategy: all individuals in the population that are not dominated by other solutions (noninferior solutions) are defined as rank 1; that is, the virtual fitness of the individual is 1; then these individuals are removed from the population, and new noninferior solutions are identified from the remaining individuals, which are defined as rank 2; that is, the virtual fitness of the individual is 2; the above process is repeated until all individuals in the population have been assigned the corresponding rank. By storing the current solution and all the individuals in the population in a hierarchical manner, the high-performance individuals can have a higher probability of survival and the population rank can be improved rapidly.

2) Pareto-optimal strategy: the Pareto set is multiple viable solutions, which can provide multiple viable solutions in the supply of water to the mineral processing plant and can also reduce supply efficiency. Manual Pareto selection suffers from a variety of subjective uncertainties, which can make the selected solution less than optimal.

Define a function \( \eta_i \) representing the proportion of the \( i \)th objective function value of a solution in the Pareto set to the total range along the direction of the \( i \)th objective function. The function \( \eta_i \) is expressed as
\begin{equation}
    \eta_i = \begin{cases} 
    1, & V_i \leq V_{i}^{\text{min}} \\
    \frac{V_i^{\text{max}} - V_i^{\text{min}}}{V_i^{\text{max}} - V_i^{\text{min}}} & V_i^{\text{min}} \leq V_i \leq V_i^{\text{max}} \\
    0, & V_i \geq V_i^{\text{max}}
    \end{cases} \tag{16}
\end{equation}

where \( V_i^{\text{max}} \) and \( V_i^{\text{min}} \) are the maximum and minimum values of the \( i \)th objective function in the Pareto set, respectively; \( V_i \) is the function value of the \( i \)th objective function. For each noninferior solution \( k \), the normalized membership function \( \eta_k \) is calculated by the following equation:
\begin{equation}
    \eta_k = \frac{\sum_{i=1}^{N_{\text{obj}}} \eta_i^k}{\sum_{j=1}^{M} \sum_{i=1}^{N_{\text{obj}}} \eta_i^j}, \tag{17}
\end{equation}

where \( M \) is the number of noninferior solutions in the Pareto set, and \( N_{\text{obj}} \) is the number of objective functions. The larger the value of \( \eta_k \), the better the performance of \( k \) in coordinating multiple objective functions. Sorting the Pareto set by the \( \eta_k \) value gives a priority sequence of noninferior solutions.
(3) Crowding distance strategy: the crowding distance is used to estimate how dense a solution is with other solutions around it. For each objective function, the set of noninferior solutions is sorted by the objective function value; then, for each solution \( x \), the sum of the two sides of the rectangle formed by the solutions \( x + 1 \) and \( x - 1 \) is calculated; the final result is the crowding distance \( x_{\text{dist}} \) of the solution \( x \). The crowding distance of the boundary solution is infinite. When two solutions have the same rank (i.e., the same virtual fitness), the superiority of the two solutions can be compared by the crowding distance. It can be seen that \( x \) individual is superior to \( y \) individual when and only when \( x_{\text{rank}} < y_{\text{rank}} \) or \( x_{\text{rank}} = y_{\text{rank}} \) and \( x_{\text{dist}} < y_{\text{dist}} \).

(4) Search strategy: as one of the most representative variants, DNSPSO algorithm is characterized by introducing a global NS operator [33]. Therefore, the search strategy in this paper also adds that operator, and it can be expressed as follows:

\[
v_{ij} = r_1 x_{ij} + r_2 x_{\text{best}} + r_3 (x_{\text{aj}} - x_{\text{bj}}),
\]

where \( r_1, r_2, \) and \( r_3 \) are three mutually exclusive numbers which are randomly chosen from \((0, 1)\), and they have to meet another condition: \( r_1 + r_2 + r_3 = 1 \); \( x_{\text{best}} \) is the global best solution of the entire population; the indices \( a \) and \( b \) are mutually exclusive integers randomly chosen from \((1, 2, \ldots, N)\), and they are different from the base index \( i \). It is necessary to point out that \( r_1, r_2, \) and \( r_3 \) are generated anew in each generation, but they are kept the same for all dimensions in each generation. Once the trial solution is generated, its associated food source has to compete with it for entering the next generation. It implies that the one with a better fitness value has the chance to survive.

Combining the above multiobjective strategies and the ABC algorithm, the basic flow of the EMOABC algorithm is shown in the following:

Step 1: set the number of solution parameters \( D \), the number of populations \( N \), the maximum number of cycles \( MNC \), and "limit" value; randomly generate candidate solutions \( \mathbf{X} = \{x_{ij} \}_{N \times D} \).

Step 2: calculate the fitness of each objective function, noninferior rank, and the crowding distance.

Step 3: randomly select two solutions and choose the better storage solution. In this way, \( N/2 \) better solutions are selected.

Step 4: search in the manner of equation (10) to produce \( N/2 \) solution, and if there is a solution beyond the boundary, it is placed on the boundary.

Step 5: identify the better \( N \) solutions from the population of steps 3 and 4 according to noninferior rank and the crowding distance.

Step 6: generate \( N \) new solutions by equation (15), and if there are any solutions beyond the boundary, they are placed on the tenth boundary.

Step 7: select the better \( N \) solutions to form a new population according to noninferior rank and the crowding distance from the solutions in steps 1, 3, and 6.

Step 8: determine if the termination condition is met; if so, output the Pareto front; otherwise, go to step 2.

Step 9: arrange all solutions in the Pareto-optimal set, using equations (13) and (14), and then export a priority list of solutions.

3.3. Performance of the Proposed Algorithm. To verify the effectiveness of the EMOABC proposed in this paper, its performance is compared with those of NSGA-II and MOPSO. Convergence and diversity are usually chosen as evaluation criteria, and their specific mathematical expressions can be found in the literature [30, 31]. To ensure the fairness of algorithm testing, the size of the initial population \( N \) is set to 60 and the maximum number of cycles is 1000 for all algorithms on each test function. To achieve the best performance of the other two algorithms, their control parameters were set using the recommended values from the corresponding original literature. For EMOABC, the "limit" was set to 0.6 \( \times \) \( D \times N \). To avoid the adverse effect of randomness on algorithm evaluation, each algorithm was run 30 times independently, and the mean value and the standard deviation of convergence and diversity were calculated, and the results of each algorithm for three benchmark functions ZDT1, ZDT2, and ZDT3 are shown in Table 1. Detailed expressions of ZDT1, ZDT2, and ZDT3 can be found in the literature [30]. Three algorithms have been run in MATLAB R2020a on a personal computer with Core 2 Duo 2.13 GHz processor and 4 GB memory.

From Table 1, the best values have been marked in bold. In terms of diversity, compared to other algorithms, the EMOABC algorithm obtained the most uniform Pareto front for each test problem with good distribution characteristics, indicating that the EMOABC algorithm has obvious advantages in distribution characteristics; in terms of convergence, for the ZDT3 problem, the EMOABC algorithm performs slightly worse than the NSGA-II algorithm but better than the MOPSO algorithm. Besides, the standard deviations of convergence and diversity obtained by the EMOABC algorithm are small, indicating that this algorithm is more stable and can almost always converge to the optimal nondominated frontier within the maximum number of cycles, which proves the effectiveness of the EMOABC algorithm to obtain a wider, uniform, and realistic Pareto solution set frontier.

Table 1 cannot completely reflect the results obtained by three algorithms; for this reason, and Figure 2 shows the noninferior optimal solution sets of three benchmark functions obtained by three algorithms.
As can be seen in Figure 2, the Pareto-optimal solutions of the EMOABC algorithm on ZDT1–3 can approximate the theoretical Pareto-optimal solution of each function very well, which has especially a good distribution. This experimental result once again verifies the effectiveness and advancement of the EMOABC algorithm for the multiobjective optimization problem.

4. Simulation and Discussion

The proposed method has been applied to two cases of mineral processing plants with different water sources and it is compared with MOPSO and NSGA-II to solve this problem. Parameters settings for three algorithms applied to two cases are the same as those set in Section 3.3.

4.1. Case 1. Case 1 is that the proposed algorithm is applied to the MPWD model based on mineral processing supplied by four water sources. The parameters about upper and lower limits for Case 1 are shown in Table 2, where 1 represents freshwater, 2 represents tailings water, 3 represents clean water, and 4 represents in-plant water. For recycled water, the lower cost with water source should be used first. In Case 1, since \( q_2 < q_4 < q_3 \), set the penalty factor \( \alpha = 3 \), \( \beta = 14 \), and \( \chi = 8 \). The macro model, which uses historical data as a sample, is built by a stepwise regression approach [34]. The macro model of the water supply network and the macro model of the pressure control point in Case 1 are given in Tables 3 and 4, respectively. The following macro models (in Tables 3–6) are all simulation models.

To optimize both mineral processing water costs and the amount of recycled water simultaneously, the proposed EMOABC algorithm has been applied in the MPWD model with \( q_{\text{sum}} = 3000 \) and 4000. The five best solutions and best \( \eta k \) are summarized in Tables 7 and 8. The higher the value of \( \eta k \) for a solution, the higher the priority of the solution. From Table 7, we can see that \( C(Q) \) in S2 and S3 is slightly different, but \( q_1 - q_4 \) in S2 and S3 are completely different. Also, the difference between \( q_1 - q_4 \) in S2 and S3 is not significant, but the difference of \( C(Q) \) in S2 and S3 is obvious. The above denotes that a set of optimal solutions are obtained by the proposed EMOABC algorithm. The comparison between Tables 7 and 8 shows that when \( q_{\text{sum}} \) rises from 3000 to 4000, \( q_1 \), \( q_2 \), \( q_3 \), and \( q_4 \) increase accordingly, while \( q_3 \) generally declines. \( q_2 \) increases the most due to its low cost, and the decline in \( q_4 \) is due to its high cost. The results of using EMOABC algorithm to schedule the flow of different water sources are consistent with those of manual regulation.

The MPWD model is solved using EMOABC, NSGA-II, and MOPSO. A comparison of the results obtained by three algorithms is shown in Table 9. It is seen that the EMOABC algorithm yields a lower cost of mineral processing water and a higher amount of recycled water (in bold) than the other two algorithms.

From Figure 3, it can be observed that EMOABC can discover a well-distributed and diverse solution set for this problem, while the other two algorithms only find some sparse distributions.

4.2. Case 2. Case 2 is that the proposed algorithm is applied to the MPWD model based on mineral processing supplied by six water sources. The parameters about upper and lower limits for Case 2 are shown in Table 10, where 1 represents freshwater no. 1, 2 represents freshwater no. 2, 3 represents tailings water no. 1, 4 represents tailings water no. 2, 5 represents clean water, and 6 represents in-plant water. For recycled water, the water source with a lower cost should be used first. In Case 2, since \( q_2 < q_4 < q_6 < q_5 \), set the penalty factor \( \alpha_1 = 3 \), \( \alpha_2 = 4 \), \( \beta = 14 \), and \( \chi = 8 \). The macro model of the water supply network and the macro model of the pressure control point in Case 2 are given in Tables 5 and 6.

In this case, \( C(Q) \) and \( M(Q) \), two competing objectives, are optimized simultaneously by the EMOABC algorithm, and the five best solutions are shown in Tables 11 and 12. From Tables 11 and 12, the convergence of the solutions in Case 2 is as good as that in Case 1. It is worth noting that as the number of water sources increases, the water selection options are diverse. A comparison of the results obtained by MOPSO, NSGA-II, and EMOABC in Case 2 is shown in Table 13. From Table 13, we can observe that the EMOABC algorithm obtains the optimal solution (in bold) in Case 2 similar to Case 1.

Figure 4 shows the distribution of nondominated solutions obtained by NSGA-II, MOPSO, and EMOABC in Case 2. It is seen from them that EMOABC spreads well...
Figure 2: The comparable Pareto front obtained by three algorithms.

### Table 2: Related parameters in Case 1.

| Water sources | $q_i^\text{min}$ (m$^3$/h) | $q_i^\text{max}$ (m$^3$/h) | $p_i^\text{min}$ (MPa) | $p_i^\text{max}$ (MPa) | $h_i$ (m) | $c_i$ ($$/m^3$$) | $c_p^\text{min}$ (MPa) |
|---------------|-----------------|-----------------|-----------------|-----------------|---------|-----------------|-----------------|
| 1             | 100             | 1000            | 0.15            | 0.95            | 15      | 0.11            | 0.13            |
| 2             | 120             | 3000            | 0.18            | 0.86            | 56      | 0.24            | 0.14            |
| 3             | 110             | 2000            | 0.17            | 0.93            | 26      | 0.46            | 0.13            |
| 4             | 95              | 1500            | 0.16            | 0.74            | 5       | 0.32            | 0.11            |

### Table 3: Related macro model of water supply network in Case 1.

| Water sources | $f(q_{i\text{sum}}, q_i, q_j)$ |
|---------------|----------------------------------|
| 1             | $0.145 + 2.618 \times 10^{-9} \times q_i^2 + 3.698 \times 10^{-9} \times q_j^2 + 6.594 \times 10^{-8} \times q_i q_j + 2.365 \times 10^{-8} \times q_{i\text{sum}}$ |
| 2             | $0.128 + 4.513 \times 10^{-9} \times q_i^2 + 6.485 \times 10^{-8} \times q_j^2 + 2.684 \times 10^{-10} \times q_i q_j + 9.264 \times 10^{-10} \times q_{i\text{sum}}$ |
| 3             | $0.117 + 1.561 \times 10^{-9} \times q_i^2 + 1.248 \times 10^{-9} \times q_j^2 + 5.654 \times 10^{-9} \times q_i q_j + 4.589 \times 10^{-8} \times q_{i\text{sum}}$ |
| 4             | $0.124 + 2.415 \times 10^{-9} \times q_i^2 + 8.261 \times 10^{-9} \times q_j q_i + 1.368 \times 10^{-8} \times q_{i\text{sum}}$ |
Table 4: Related macro model of the pressure control point in Case 1.

| Water source | \( f(q_{sum}, q_i) \) |
|--------------|------------------------|
| 1            | 0.109 + 1.529 \times 10^{-9} \times q_1^2 + 3.659 \times 10^{-10} \times q_{sum}^2 |
| 2            | 0.123 + 4.268 \times 10^{-9} \times q_2^2 + 5.364 \times 10^{-9} \times q_{sum}^2 |
| 3            | 0.151 + 7.268 \times 10^{-10} \times q_3^2 + 5.269 \times 10^{-8} \times q_{sum}^2 |
| 4            | 0.124 + 6.324 \times 10^{-8} \times q_4^2 + 8.624 \times 10^{-10} \times q_{sum}^2 |

Table 5: Related macro model of water supply network in Case 2.

| Water sources | \( f(q_{sum}, q_1, q_2) \) |
|---------------|-----------------------------|
| 1             | 0.125 + 3.584 \times 10^{-8} \times q_1^2 + 5.629 \times 10^{-10} \times q_{sum}^2 |
| 2             | 0.114 + 8.268 \times 10^{-10} \times q_1^2 + 3.628 \times 10^{-9} \times q_{sum}^2 |
| 3             | 0.115 + 1.269 \times 10^{-8} \times q_1^2 + 1.562 \times 10^{-8} \times q_{sum}^2 |
| 4             | 0.142 + 7.261 \times 10^{-8} \times q_1^2 + 6.826 \times 10^{-9} \times q_{sum}^2 |
| 5             | 0.138 + 9.261 \times 10^{-10} \times q_1^2 + 5.698 \times 10^{-10} \times q_{sum}^2 |
| 6             | 0.152 + 3.652 \times 10^{-8} \times q_1^2 + 1.268 \times 10^{-8} \times q_{sum}^2 |

Table 6: Related macro model of the pressure control point in Case 2.

| Water source | \( f(q_{sum}, q_i) \) |
|--------------|------------------------|
| 1            | 0.121 + 2.316 \times 10^{-10} \times q_1^2 + 8.362 \times 10^{-9} \times q_{sum}^2 |
| 2            | 0.132 + 5.326 \times 10^{-9} \times q_2^2 + 3.958 \times 10^{-10} \times q_{sum}^2 |
| 3            | 0.142 + 9.635 \times 10^{-10} \times q_3^2 + 4.369 \times 10^{-9} \times q_{sum}^2 |
| 4            | 0.129 + 2.589 \times 10^{-9} \times q_4^2 + 5.241 \times 10^{-10} \times q_{sum}^2 |
| 5            | 0.161 + 7.125 \times 10^{-9} \times q_5^2 + 8.629 \times 10^{-10} \times q_{sum}^2 |
| 6            | 0.135 + 5.264 \times 10^{-7} \times q_6^2 + 1.259 \times 10^{-8} \times q_{sum}^2 |

Table 7: Five best solutions for Case 1 using the EMOABC (\( q_{sum} = 3000 \)).

| Solutions | \( q_1 \) (m³/h) | \( q_2 \) (m³/h) | \( q_3 \) (m³/h) | \( q_4 \) (m³/h) | \( C (Q) \) ($/h) | \( M (Q') \) (m³/h) | \( rjk \) |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------|
| S₁        | 328             | 1156            | 875             | 641             | 61577           | 20084           | 0.0536  |
| S₂        | 396             | 1297            | 801             | 506             | 61683           | 19153           | 0.0534  |
| S₃        | 298             | 1307            | 783             | 612             | 61681           | 19779           | 0.0531  |
| S₄        | 384             | 1376            | 752             | 488             | 61613           | 18560           | 0.0529  |
| S₅        | 395             | 1298            | 803             | 504             | 61756           | 19168           | 0.0528  |

Table 8: Five best solutions for Case 1 using the EMOABC (\( q_{sum} = 4000 \)).

| Solutions | \( q_1 \) (m³/h) | \( q_2 \) (m³/h) | \( q_3 \) (m³/h) | \( q_4 \) (m³/h) | \( C (Q) \) ($/h) | \( M (Q') \) (m³/h) | \( rjk \) |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------|
| S₁        | 422             | 2120            | 804             | 654             | 103820          | 22848           | 0.0546  |
| S₂        | 439             | 2206            | 739             | 616             | 103450          | 21892           | 0.0554  |
| S₃        | 432             | 2072            | 824             | 672             | 103380          | 23128           | 0.0552  |
| S₄        | 398             | 2218            | 730             | 654             | 103580          | 22106           | 0.0558  |
| S₅        | 406             | 2195            | 752             | 647             | 103790          | 22289           | 0.0549  |

Table 9: The best solutions for Case 1 using three algorithms.

| MOPSO \( q_{sum} = 3000 \) | NSGA-II \( q_{sum} = 4000 \) | EMOABC \( q_{sum} = 4000 \) | \( q_{sum} = 3000 \) | \( q_{sum} = 4000 \) |
|-----------------------------|-----------------------------|-----------------------------|-----------------|-----------------|
| \( q_1 \)                  | 395                         | 419                         | 329             | 395             |
| \( q_2 \)                  | 1294                        | 2118                        | 1159            | 2221            |
| \( q_3 \)                  | 802                         | 807                         | 864             | 732             |
| \( q_4 \)                  | 509                         | 656                         | 648             | 652             |
| \( C (Q) \)                | 61668                       | 103900                      | 61641           | 103750          |
| \( M (Q') \)               | 19182                       | 22900                       | 20057           | 22127           |

Bold highlights that the algorithm proposed in this paper achieves better results.
on the optimization problem, the function value obtained by EMOABC is significantly better than that obtained by NSGA-II, and MOPSO and EMOABC can find a better spread and more solutions in the entire Pareto-optimal region than the other two algorithms. It is indicated that the proposed algorithm is more effective than the other algorithms for solving the MPWD problem. Figures 3 and 4 also show that the two objectives of maximizing the amount of recycled water and minimizing mineral processing water costs conflict. As mineral processing water costs increase, the recycled water consumption decreases, and vice versa.
4.3. Algorithm Complexity Analysis. Complexity analysis of the above algorithms was performed using the timeresponse for both cases. The average computation time for all algorithms over 30 runs is given in Table 14. From Table 14, it can be seen that EMOABC takes the most computation time. This can be explained by the fact that the multiple strategies integrated in EMOABC enhance the search capability at the cost of increased computation. In conclusion, EMOABC requires more computation time to achieve better results compared to other algorithms.

5. Conclusions

The main goal of this study was to solve the MPWD problem effectively. In this study, an MPWD optimization model was developed to maximize the amount of recycled water and minimize mineral processing water costs, with consideration of various constraints. To solve this multiobjective model, the EMOABC algorithm was proposed as an optimizer derived from the ABC algorithm. This study has shown that the developed model was theoretically valid and reasonable.
and could be utilized to provide decision support for actual MPWD. The study has also shown that the proposed algorithm is competent to solve the MPWD problem and is better than NSGA-II and MOPSO in both benchmark functions and cases.

A limitation of this study is that the developed model only has two objectives. The MPWD problem described as a three-objective or even four-objective model may be more realistic. Therefore, building a three-objective or even four-objective model for this MPWD problem is a matter of future work. Meanwhile, continued research of the multiobjective artificial bee colony (MOABC) algorithm is also a focus. In particular, aiming at a three-objective or even four-objective model for this MPWD problem, the concept of multihives bee will be introduced into the ABC algorithm to further improve the performance of the MOABC algorithm.

**Data Availability**

No data were used to support this study.

**Conflicts of Interest**

The authors declare that there are no conflicts of interest regarding the publication of this paper.

**Acknowledgments**

This project was fully supported by Liaoning Provincial Department of Education Basic Research Projects for Higher Education Institutions, China (no. LJKZ0301), the Scientific Research Foundation of the Education Department of Liaoning Province, China (no. 2017LNQN22), and Young Teachers Foundation of University of Science and Technology Liaoning, China (no. 2017QN04).

**References**

[1] P. R. Bhave and V. V. Sonak, "A critical study of the linear programming gradient method for optimal design of water supply networks," *Water Resources Research*, vol. 28, no. 6, pp. 1577–1584, 1992.

[2] J. A. Frizzzone, R. D. Coelho, D. Dourado-Neto, and R. Soliani, "Linear programming model to optimize the water resource use in irrigation projects: an application to the Senator Nilo Coelho Project," *Scientia Agricola*, vol. 54, no. spe, pp. 136–148, 1997.

[3] G. Yu, R. S. Powell, and M. J. H. Sterling, "Optimized pump scheduling in water distribution systems," *Journal of Optimization Theory and Applications*, vol. 83, no. 3, pp. 463–488, 1994.

[4] H. M. V. Samani and A. Mottaghi, "Optimization of water distribution networks using integer linear programming," *Journal of Hydraulic Engineering*, vol. 132, no. 5, pp. 501–509, 2006.

[5] A. Haghhighi, H. M. V. Samani, and Z. M. V. Samani, "GA-ILP method for optimization of water distribution networks," *Water Resources Management*, vol. 25, no. 7, pp. 1791–1808, 2011.

[6] C. Zoppou and M. Reed, "Water distribution network design optimization: simulated annealing approach," *Journal of Water Resources Planning and Management*, vol. 127, no. 1, pp. 69–70, 2001.

[7] J. Marques, M. Cunha, and D. A. Savič, "Multi-objective optimization of water distribution systems based on a real options approach," *Environmental Modelling & Software*, vol. 63, pp. 1–13, 2015.

[8] K. Zhang, H. Yan, H. Zeng, K. Xin, and T. Tao, "A practical multi-objective optimization sectorization method for water distribution network," *The Science of the Total Environment*, vol. 656, pp. 1401–1412, 2019.

[9] B. A. Tolson, H. R. Maier, A. R. Simpson, and B. J. Lence, "Genetic algorithms for reliability-based optimization of water distribution systems," *Journal of Water Resources Planning and Management*, vol. 130, no. 1, pp. 63–72, 2004.

[10] N. Moosavian and B. J. Lence, "Nondominated sorting differential evolution algorithms for multiobjective optimization of water distribution systems," *Journal of Water Resources Planning and Management*, vol. 143, no. 4, Article ID 04016082, 2017.

[11] R. Mansouri, H. Torabi, M. Hoseini, and H. Morshedzadeh, "Optimization of the water distribution networks with differential evolution (DE) and mixed integer linear programming (MILP)," *Journal of Water Resource and Protection*, vol. 7, no. 9, pp. 715–729, 2015.

[12] N. Moosavian and B. J. Lence, "Fittest individual referenced differential evolution algorithms for optimization of water distribution networks," *Journal of Computing in Civil Engineering*, vol. 33, no. 6, Article ID 04019036, 2019.

[13] S. Pandey and Y. P. Mathur, "Optimization of water distribution network using particle swarm optimization," *International Journal of Applied Engineering Research*, vol. 8, no. 12, pp. 92–101, 2013.

[14] I. Montalvo, J. Izquierdo, S. Schwarze, and R. Perez Garcia, "Multi-objective particle swarm optimization applied to water distribution systems design: an approach with human interaction," *Mathematical and Computer Modelling*, vol. 52, no. 7–8, pp. 1219–1227, 2010.

[15] X. Qi, K. Li, and W. D. Potter, "Estimation of distribution algorithm enhanced particle swarm optimization for water distribution network optimization," *Frontiers of Environmental Science & Engineering*, vol. 10, no. 2, pp. 341–351, 2016.

[16] A. C. Zecchin, H. R. Maier, A. R. Simpson, M. Leonard, and J. B. Nixon, "Ant colony optimization applied to water distribution system design: comparative study of five algorithms," *Journal of Water Resources Planning and Management*, vol. 133, no. 1, pp. 87–92, 2007.

[17] A. Afshar, M. Massoumi, A. Afshar, and M. A. Mariño, "State of the art review of ant colony optimization applications in water resource management," *Water Resources Management*, vol. 29, no. 11, pp. 3891–3904, 2015.

[18] M. López-Ibáñez, T. D. Prasad, and B. Paechter, "Ant colony optimization for optimal control of pumps in water distribution networks," *Water Resources Planning and Management*, vol. 129, no. 3, pp. 210–225, 2003.

[19] D. Mora-Melia, P. Iglesias-Rey, F. Martínez-Solano, and P. Muñoz-Velasco, "The efficiency of setting parameters in a
modified shuffled frog leaping algorithm applied to optimizing water distribution networks,” *Water*, vol. 8, no. 5, p. 182, 2016.

[22] W. Deng, X. Zhang, Y. Zhou et al., “An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems,” *Information Sciences*, vol. 585, pp. 441–453, 2022.

[23] Z. H. Zhang, F. Min, G. S. Chen, S. P. Shen, Z. C. Wen, and X. B. Zhou, “Tri-partition state alphabet-based sequential pattern for multivariate time series,” *Cognitive Computation*, pp. 1–19, 2021.

[24] H. Cui, Y. Guan, H. Chen, and W. Deng, “A novel advancing signal processing method based on coupled multi-stable stochastic resonance for fault detection,” *Applied Sciences*, vol. 11, no. 12, p. 5385, 2021.

[25] X. Ran, X. Zhou, M. Lei, W. Tepsan, and W. Deng, “A novel k-means clustering algorithm with a noise algorithm for capturing urban hotspots,” *Applied Sciences*, vol. 11, no. 23, Article ID 11202, 2021.

[26] H. Chen, M. L. Bo, and Y. Zhu, “Multi-hive bee foraging algorithm for multi-objective optimal power flow considering the cost, loss, and emission,” *International Journal of Electrical Power & Energy Systems*, vol. 60, pp. 203–220, 2014.

[27] Z. Yong, H. Chun-lin, S. Xian-fang, and S. Xiao-yan, “A multi-strategy integrated multi-objective artificial bee colony for unsupervised band selection of hyperspectral images,” *Swarm and Evolutionary Computation*, vol. 60, Article ID 100806, 2021.

[28] Z. Li, M. N. Janardhanan, and Q. Tang, “Multi-objective migrating bird optimization algorithm for cost-oriented assembly line balancing problem with collaborative robots,” *Neural Computing & Applications*, vol. 33, pp. 1–22, 2021.

[29] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[30] K. Wang, X. Li, L. Gao, P. Li, and J. W. Sutherland, “A discrete artificial bee colony algorithm for multiobjective disassembly line balancing of end-of-life products,” *IEEE Transactions on Cybernetics*, vol. 99, pp. 1–12, 2021.

[31] Z. Li, M. N. Janardhanan, and Q. Tang, “Multi-objective migrating bird optimization algorithm for cost-oriented assembly line balancing problem with collaborative robots,” *Neural Computing & Applications*, vol. 33, pp. 1–22, 2021.

[32] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, “Handling multiple objectives with particle swarm optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.

[33] D. Karaboga and B. Akay, “A comparative study of artificial bee colony algorithm,” *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.

[34] X. Zhou, H. Wang, M. Wang, and J. Wan, “Enhancing the modified artificial bee colony algorithm with neighborhood search,” *Soft Computing*, vol. 21, no. 10, pp. 2733–2743, 2017.