Application Research of CFD-MOEA/D Optimization Algorithm in Large-Scale Reservoir Flood Control Scheduling

Hongbo Jiao, Huaibin Wei *, Qi Yang and Min Li

School of Water Resources, North China University of Water Resources and Electric Power, Zhengzhou 450046, China
* Correspondence: whbncwu@126.com

Abstract: Reservoir flood control has an important impact on flood protection and plays an important role in reducing the loss of people’s lives and property. In order to play an important role in flood control operation of large-scale reservoirs, a control flood dispatching multi-objective evolutionary algorithm based on decomposition (CFD-MOEA/D) is proposed. The same type of multi-objective optimization algorithm (non-dominated sorting genetic algorithm II (NSGA-II)) is introduced, and CFD-MOEA/D, NSGA-II, and traditional MOEA/D algorithms are compared. The research results show that the CFD-MOEA/D algorithm can obtain the non-dominated solution of the higher water level in the upstream, and the solution obtained by the CFD-MOEA/D algorithm is more sufficient than the NSGA-II algorithm and the MOEA/D algorithm. When analyzing the HV value curve, the uniformity and convergence of the optimal solution obtained by the CFD-MOEA/D algorithm are better than those of the other two algorithms. The optimal dispatching scheme of the CFD-MOEA/D algorithm is compared with the actual dispatching scheme of the reservoir, and it is found that the maximum upstream water level and the final water level obtained by the CFD-MOEA/D algorithm are both kept at approximately 325 m, which is consistent with the actual dispatching scheme. The new feature of the algorithm is that it uses a decomposition method from coarse to fine and improves the hourly scheduling scheme to obtain higher scheduling efficiency.

Keywords: CFD-MOEA/D; optimization algorithm; pareto optimal solution; reservoir flood control scheduling

1. Introduction

As a frequent natural disaster, floods pose a great threat to the safety of people’s property and lives. As floods affect a large area and the factors causing floods are complex, it is difficult to prevent and relieve flooding. As an effective flood control method, reservoir flood control can play a role in flood relief and control [1]. For example, in the early stages of a flood, the reservoir can discharge in advance to leave space for flood storage to ease the flow of the flood. In the main flood season of flood disaster, it mainly guarantees the safety of flood control and reduces the damage degree of flood disaster. In the later stages, the flood stored in the reservoir is properly discharged, and the original water storage capacity before the flood is maintained. In the whole reservoir flood control scheduling, there are optimization problems with multiple objectives, such as the need to ensure the safety of the reservoir and the safety of the people downstream, so it requires a multi-objective optimization problem (MOP) [2]. For the reservoir flood control operation problem, the optimization algorithm can be used to solve it. The multi-objective evolutionary algorithm based on decomposition (multi-objective evolutionary algorithm based on decomposition, MOEA/D) has certain advantages in solving such problems [3]. The principle of the large-scale multi-objective reservoir flood control operation algorithm is analyzed, and the algorithm performance is deeply analyzed and verified in order to find an effective solution to reservoir flood control operation.

The main contents of the study are as follows:
(1) Firstly, domestic and foreign research on scheduling problems and multi-objective optimization problems is analyzed.

(2) The problems in multi-objective reservoir flood control operation and the method of model construction are introduced. The principle of the traditional MOEA/D algorithm and the improved CFD-MOEA/D algorithm are introduced.

(3) The NSGA-II algorithm is introduced and compared with traditional MOEA/D algorithm and CFD-MOEA/D algorithm, and the advantages of the CFD-MOEA/D algorithm are verified through experiments.

2. Related Work

Large scale reservoir flood control operation is a common operation problem. At present, there is little research on algorithms in reservoir flood control operation at home and abroad. However, for other kinds of scheduling problems, such as job shop scheduling, power scheduling, and so on, scholars have undertaken considerable research. For example, Zhao et al. proposed a scheduling algorithm based on a deep Q network (DQN) for the production shop scheduling problem, and simulation results show that the algorithm has better applicability and performance than a single scheduling rule or the traditional Q learning algorithm [4]. For the job shop scheduling problem in the energy production department, Grosch et al. proposed an energy adaptive production scheduling algorithm using a non-dominated sorting genetic algorithm, and research shows that the algorithm can achieve the efficiency of energy production scheduling, thereby reducing production costs [5]. Aiming at the operation scheduling problem of shipborne aircraft flight decks, Cui et al. established a comprehensive scheduling problem model for multi-aircraft turnaround time and applied the dual population multi-operator genetic algorithm to solve the problem. The simulation time results show that the algorithm has good performance [6]. For the coordinated serial batch processing scheduling problem, Pei et al. proposed a hybrid BA-VNS algorithm combining the bat algorithm (BA) and the variable neighborhood algorithm (VNS). The experimental results show that the combined algorithm has good performance, high convergence speed, and computational stability [7]. For the scheduling problem of flexible equipment systems, Li et al. proposed a heuristic beam search algorithm to minimize the manufacturing time, and the experiment proved that the algorithm has a certain effectiveness and superiority [8]. Wang et al. proposed an intrusion tolerance scheduling algorithm in a cloud-based scientific workflow system to enhance its security when researching the security problems of cloud computing, and research shows that this method has a high efficiency and success rate [9]. Iqbal et al. introduced a genetic algorithm to optimize their power system when studying the power system scheduling problem, and the results show that this method can improve the load curve by reducing the overall energy cost and peak-to-average ratio, thus effectively solving the load scheduling problem [10]. Niu et al. applied a scheduling optimization algorithm to improve the efficiency of cloud platform research on a big data-driven network physical system [11]. The research proved that the algorithm is feasible, and it has higher accuracy and efficiency than other algorithms.

The CFD-MOEA/D optimization algorithm applied in the research belongs to a multi-objective optimization algorithm, which is widely used in various fields by different scholars. For example, in view of the wavelength selection and shortest path planning problems in network data transmission, Hamsaveni et al. proposed a hunger locust optimization algorithm model based on multi-objective optimization to solve this problem, and the research results show that the algorithm can effectively analyze the availability of the best routing path and wavelength at each time node [12]. Bose et al. proposed a new optimization algorithm based on an ideal grey relational analysis and applied it to the statistics and experiments of new titanium-based hybrid composites. The results show that the method has high accuracy and performance [13]. For multi-label classification, Aab et al. proposed a multi-target feature selection algorithm to solve the multi-label feature selection problem, and the experimental results show that this method can obtain a set
of well-distributed trade-off solutions, and higher classification accuracy [14]. Bao et al. applied a multifunctional optimization algorithm based on the biological immune system to virtual reality (VR) image segmentation, and the study showed that it can demonstrate the diversity of Pareto-oriented methods and solutions. In addition, the algorithm also has high performance, prediction accuracy, and robustness [15]. Kaveh et al. proposed a ground motion record selection method based on three different multi-objective optimization algorithms in order to better conduct a time–history dynamic analysis of ground motion records at specific sites. The results show that NSGA II outperforms other algorithms in the case of GMR selection [16]. To improve the service efficiency of the agricultural Internet of Things, Yang et al. designed a hierarchical co-evolutionary multi-objective optimization algorithm inspired by the immune endocrine system and applied it to the field of agricultural Internet of Things services. Simulation experiments verified that the algorithm has strong search ability and excellent performance [17]. In order to effectively obtain the corpus features of neural machine translation, Song improved the multi-objective optimization algorithm and built an intelligent English translation system. It has been proven by experiments that it has good performance and can meet the translation needs [18].

In order to optimize the characteristics of the synchronous condenser and improve the condenser’s ability to support AC system voltage, Shi et al. proposed a fuzzy control method based on MOEA/D to optimize the proportional adjustment coefficient in the condenser reactive power and AC system voltage outer loop control. The simulation results show that the design improves the condenser performance and solves the AC system overvoltage problem [19].

From the above analysis, it can be seen that there are many studies on scheduling problems, such as workshop production scheduling, load scheduling, etc., and different researchers have applied various machine learning and deep learning methods to scheduling problems and have achieved good results. However, there are few studies on the application of related algorithm technology on reservoir flood control and its research value needs to be further explored. The research uses the multi-objective optimization algorithm based on decomposition technology and analyzes its effect in the flood control of the reservoir through the relevant models, in order to find an effective means to solve the flood control problem.

3. Model Building

3.1. Construction of Multi-Objective Reservoir Flood Control Scheduling Model

The multi-objective reservoir flood control scheduling problem belongs to a multi-objective optimization problem (MOP), which can be expressed by certain mathematical models and formulas, as shown in Formula (1).

\[
\begin{align*}
\text{minimize} & \quad F(x) = (f_1(x), f_2(x), \cdots, f_m(x)) \\
\text{subject to} & \quad x \in \Omega
\end{align*}
\]

In Formula (1), \( \Omega \) is the feasible region of the decision space, which meets the conditions: \( \Omega \in \mathbb{R}^n \). \( x = \{x_1, x_2, \cdots, x_n\} \in \Omega \) represents the decision variable, the dimension corresponding to its space is \( n \). The number of objective functions is \( m \), and \( F(x) : \Omega \rightarrow \mathbb{R}^m \) represents the objective vector function, which consists of \( m \) objective functions that need to be optimized at the same time. In the reservoir flood control scheduling problem, \( m = 2, n \geq 100 \), so it is a large-scale multi-objective optimization problem. For multi-objective problems, Pareto dominance and Pareto optimal solutions are usually introduced. To define the two, Pareto is dominant: one decision variable \( x_u \in \Omega \) dominates the other decision variable \( x_v \in \Omega \) (recorded as \( x_u \prec x_v \)), only if \( \forall i = 1, 2, \cdots, m, f_i(x_u) \leq f_i(x_v) \), and at least one \( j \in \{1, 2, \cdots, m\} \) makes \( f_j(x_u) \neq f_j(x_v) \). Pareto optimal solution: if there is no decision vector \( x \in \Omega \) that makes \( x \prec x^* \) true, that is, \( x^* \) is always Pareto dominant, then \( x^* \in \Omega \) is called Pareto optimal solution. For all objectives, there is no unique solution to make them all optimal, so Pareto solutions usually appear in the form of a set, that is, the Pareto optimal solution set [20]. The Pareto optimal preamble belongs to the set in the
target space, which is mapped from the elements in the optimal solution set. An example graph of both is shown in Figure 1.

Figure 1. Pareto optimal solution set and Pareto optimal preface example diagram.

Figure 1 shows the relationship between Pareto optimal solution set and optimal frontier. Point A, B and C in the figure are Pareto advantages. Point Q and point E are called Pareto optimal solutions of multi-objective optimization problems, F represents mapping relationship, and D represents feasible region. For the convenience of calculation, the D area is represented by Ω. In the Pareto optimal solution, there is a related definition. Among them, Pareto dominates as shown in Formula (2).

\[
\forall i \in \{1, 2, \cdots, m\} : f_i(x_u) \leq f_i(x_v) \\
\exists j \in \{1, 2, \cdots, m\} : f_j(x_u) \neq f_j(x_v)
\]

In Formula (2), \( u \) is the solution, and \( v \) is the dominant solution, denoted \( u \prec v \). The Pareto optimal solution is defined as Formula (3).

\[
\exists v \in \Omega | v \prec u
\]

In Formula (3), the definite solution \( u \) is called the Pareto optimal solution. Its optimal solution set is shown in Formula (4).

\[
PS = \{ u \in \Omega | \exists v \in \Omega | v \prec u \}
\]

In Formula (4), \( PS \) is the Pareto optimal solution set. The Pareto front is shown in Formula (5).

\[
PF = \{ F(x) | x \in PS \}
\]

In Formula (5), \( PF \) is the Pareto front. There are two optimization objectives in reservoir flood control scheduling, which correspond to two tasks: the safety of upstream reservoirs and downstream residents. The first task corresponds to the objective function that minimizes the highest upstream water level, and its expression is shown in Formula (6), and \( Q \) represents the discharge flow.

\[
\text{Min} f_1(Q) = \max f_1(Z_t), t = 1, 2, \cdots, T
\]

In Formula (6), \( t \) represents the scheduling time, and \( Z_t \) represents the water level corresponding to this time, which is the period during which the flood lasts. The second
task corresponds to the objective function of minimizing the maximum discharge of the reservoir, whose expression is shown in Formula (7).

\[
\text{Min} f_2(Q) = \max f_2(Q_t), t = 1, 2, \cdots, T
\]  

(7)

In Formula (7), \(t\) represents the scheduling time, and \(Q_t\) represents the leakage flow corresponding to this time. In the scheduling model, there are several different constraints. For the upstream water level, the constraints are shown in Formula (8).

\[
Z_{\text{min}} \leq Z_t \leq Z_{\text{max}}
\]  

(8)

In Formula (8), \(Z_{\text{min}}\) is the minimum water level and \(Z_{\text{max}}\) is the maximum water level. For the leakage flow, its constraints are shown in Formula (9).

\[
0 \leq Q_t \leq Q_{\text{max}}
\]  

(9)

In Formula (9), \(Q_{\text{max}}\) is the maximum drain flow. The water balance expression is shown in Formula (10).

\[
V_t = V_{t-1} + I_t - Q_t
\]  

(10)

In Formula (10), \(V_t\) and \(V_{t-1}\) are the reservoir capacity at the corresponding dispatch time, and \(I_t\) is the inflow flow corresponding to this time. The reservoir flood control scheduling model has a certain complexity, and it also has different characteristics in decision space and target space. In the arrangement of flood scheduling time, set the total scheduling time as \(T\), and the scheduling interval as \(M\), then the number of decision variables is \(n\), as shown in Formula (11).

\[
n = \frac{T}{M}
\]  

(11)

It can be seen from Formula (11) that when the total scheduling time is determined, the number of decision variables is inversely related to the scheduling time interval. The study sets the time interval to one hour, so it belongs to a large-scale scheduling problem. It can be seen from Formula (10) that the discharge flow at time \(t\) will have an impact on both the current reservoir capacity and the discharge flow at time \(t - 1\), so the model belongs to a chain-related time series problem. The two objective functions determine that the Pareto optimal frontier has a more complex shape. In addition, many decision variables and strong correlations determine the complexity of the reservoir flood control scheduling model.

3.2. MOEA/D Algorithm Model Framework

For complex multi-objective optimization problems, the MOEA/D algorithm can solve them effectively. The principle of the MOEA/D algorithm is to transform the multi-objective optimization problem into a series of single objective optimization sub-problems. The decomposed problem is a linear or non-linear weighted set of the original MOP objective, so relevant algorithms can be applied to co-optimize it. The MOEA/D algorithm has many characteristics. First, the decomposition method is introduced in the process of running the algorithm [20]. Secondly, the MOEA/D algorithm does not directly process the MOP at runtime but optimizes the scalar sub-problem after its decomposition [21]. The MOEA/D algorithm uses a set of uniform fixed-weight vectors to aggregate each sub-problem, and the weight of the weight vector reflects the importance of the sub-problem. Each weight vector corresponds to a solution of the multi-objective optimization problem, and the distribution of the weight vector determines the distribution of the solution set [22]. In the MOEA/D algorithm, the Euclidean distance between weight vectors is used to reflect the distance between individuals in the population, and the non-dominated solution approaching Pareto is obtained by using the co-evolution of adjacent individuals [23]. Certain decomposition methods are required to decompose the MOP, and the common ones are weighted sum (WS), Tchebycheff (TCH), and boundary intersection (BI). Among them, the weighted sum
method mainly solves the MOP by weighting, that is, using a non-negative weight vector to weight the sub-objective function, so the problem is transformed into the solution of a single-objective sub-problem [24]. The expression is shown in Formula (12).

\[
\begin{cases}
\max \ g^{ws}(x|\lambda) = \sum_{i=1}^{m} \lambda_i f_i(x) \\
\text{subject to } x \in \Omega
\end{cases}
\]

In Formula (12), \( \lambda = (\lambda_1, \cdots, \lambda_m)^T \) is a set of uniform weight vectors, and for \( \lambda_i \), all need to satisfy the conditions: \( \lambda_i \geq 0 \) and \( \sum_{i=1}^{m} \lambda_i = 1 \). Weighted sum is different from the traditional weighting method. Although its calculation process is simple, it also has certain limitations. For example, it can only obtain the Pareto optimal solution in the convex case by adjusting the weight vector, but it cannot solve the non-convex case. Another method is the Tchebycheff method, whose expression is shown in Formula (13) [25].

\[
\begin{cases}
\min \ g^{tc}(x|\lambda, z^*) = \max_{1 \leq i \leq m} \{ \lambda_i | f_i(x) - z_i^* | \} \\
\text{subject to } x \in \Omega
\end{cases}
\]

In Formula (13), \( z^* = (z^*_1, \cdots, z^*_m)^T \) denotes the reference point, and \( z_i^* = \max \{ f_i(x) | x \in \Omega \} \). Compared with the weighted sum method, the Tchebycheff method is a great improvement, and it can effectively solve the problem that the shape of the front surface is non-convex. Its main advantage is that, regardless of the shape of the Pareto front, the weight can be modified to obtain the unique Pareto optimal solution corresponding to the weight. When this method deals with MOP, the curve of the front surface of the Pareto optimal solution set obtained is usually not smooth. The last decomposition method is the boundary intersection method, the schematic diagram of which is shown in Figure 2.

![Figure 2](image_url)

**Figure 2.** Schematic diagram of the boundary intersection.

As a common decomposition method, the boundary intersection method can effectively deal with continuous MOP [26]. It can be seen from Figure 2 that the upper right part
of the target space corresponds to the Pareto optimal frontier of the MOP. Its mathematical definition is shown in Formula (14).

\[
\begin{align*}
\text{minimize} & \quad g^b_i(x|\lambda, z^*) = d \\
\text{subject to} & \quad z^* - M(x) = d\lambda, x \in \Omega 
\end{align*}
\] (14)

In Formula (14), \(\lambda\) represents the weight vector and \(z^*\) represents the reference point. \(z^* - M(x) = d\lambda\) is the constraint condition of the method, which constrains the vector \(z^* - M(x)\) and the weight vector \(\lambda\) so that they are on the same straight line \(L\). Its main purpose is to let \(M(x)\) approach \(z^*\), so that the solution of the algorithm converges to the Pareto optimal solution. However, this constraint has certain shortcomings, that is, it will reduce the diversity of algorithms. To solve this problem, a penalty factor can be added, as shown in Formula (15).

\[
\begin{align*}
\text{minimize} & \quad g^{bip}(x|\lambda, z^*) = d_1 + \theta d_2 \\
\text{subject to} & \quad x \in \Omega \\
\text{where} & \quad d_1 = \frac{\|z^* - M(x)^T\|}{\|\lambda\|} \\
\text{and} & \quad d_2 = \|M(x) - (z^* - d_1\lambda)\|
\end{align*}
\] (15)

In Formula (15), \(\theta\) is the penalty coefficient, which has a great influence on the performance of the boundary crossing method. When the value is appropriate, the solution with uniform distribution can be obtained to increase the diversity of the algorithm.

3.3. CFD-MOEA/D Algorithm Model Construction

For the multi-objective problem in reservoir flood control, it is usually decomposed “from coarse to fine,” that is, it is decomposed into multiple sub-problems to achieve the purpose of optimization. This method achieves the purpose of simplification by dividing the large-scale flood scheduling problem into a number of sub-problems according to a certain time interval, and then converting it from a coarse-grained problem to a fine-grained problem step by step. The proposed coarse-to-fine decomposition method combined with the MOEA/D algorithm can be used to solve large-scale reservoir flood control operation problems, namely the CFD-MOEA/D algorithm. The process is shown in Figure 3.

![Figure 3](image)

Figure 3. Schematic diagram of decomposition method from coarse to fine.

It can be seen from Figure 3 that the scheduling time interval determines the complexity of the reservoir flood scheduling sub-problem, and the scheduling time interval for the coarse particle sub-problem is larger. For the first sub-problem, initialize its initial population in a random way. The optimal solution obtained by this sub-problem is expanded to
the initial population of the next sub-problem, and the optimal solution obtained is better than the previous sub-problem. The optimal solution is obtained. Each sub-problem is optimized accordingly, so the optimal solution corresponding to the last sub-problem is the optimal scheduling scheme. In the transformation process of the coarse- and fine-particle sub-problems, the number of decision variables contained in them is not the same, so the sub-problem needs to be expanded. This extension method is to extend the optimal solution of the coarse-grained sub-problem into the initial solution of the fine-grained sub-problem. The study takes flood scheduling with a duration of 18 h as an example, as shown in Figure 4.

![Figure 4. Schematic diagram of sub-problem expansion method.](image)

It can be seen from Figure 4 that the total duration of the flood is 18 h, the scheduling interval of sub-problem one is five hours, and the corresponding decision variables are four. After expansion, the scheduling time interval of sub-problem two becomes three hours, the corresponding decision variables are six, and the leakage flow remains stable during the expansion process. It can be seen from the above analysis that the CFD-MOEA/D algorithm mainly decomposes the multi-objective problem into multiple sub-problems, and then optimizes it gradually. The optimization process is the process from obtaining a rough approximate solution to an optimal solution, that is, for the previous sub-problems, the accuracy of the corresponding solutions is not high, and the algorithm is used to gradually reduce the search space to continuously improve the accuracy of the solution and the efficiency of the algorithm.
4. Experimental Performance Analysis of the CFD-MOEA/D Algorithm

The study uses the Ankang Reservoir in Shaanxi Province, China, as an example, and takes it as the research object of the actual reservoir flood control and scheduling problem, and takes the floods with different characteristics of the reservoir in 2016 and 2021 as the test problem. In order to effectively analyze the performance of the CFD-MOEA/D algorithm, the MOEA/D algorithm is used in the study, and another reservoir of the same type with the same flood control scheduling algorithm is introduced (the non-dominated sorting genetic algorithm-II (NSGA-II)) for comparative analysis. The NSGA-II algorithm is one of the classic multi-objective optimization algorithms. Compared with the first generation of non-dominated sorting genetic algorithm NSGA, it proposes a fast non-dominated sorting method with a computational complexity of $O(MN^2)$ [27]. In addition, it also provides a selection operator, which creates a mating pool by combining the parent population and the child population and selecting some solutions with the best fitness and sparsity values, this can ensure that the obtained optimal solution is not lost. The NSGA-II algorithm has been verified to be effective in quickly finding the Pareto frontier and maintaining population diversity. It is the benchmark of performance of other multiple optimization algorithms, so it is selected as one of the comparative algorithms in this paper.

In the performance evaluation of flood scheduling algorithms, there are different evaluation indicators. For example, PF is an evaluation of the degree of convergence of the algorithm, and hypervolume (HV) is a comprehensive evaluation index. Given the approximation solution set $P$ and reference point $R$ obtained by the multi-objective optimization algorithm, the HV index represents the volume of the hypercube surrounded by an approximation solution set $P$ and reference point $R$. The larger the value, the better the convergence and consistency of the solution. The optimal non-dominated solution distributions of the three algorithms are shown in Figure 5.

![Figure 5](image-url)

**Figure 5.** Comparison of three algorithms to obtain the best non-dominant solution in 2016 flood and 2021 flood respectively.

Figure 5a shows the optimal non-dominated solutions obtained by the three algorithms in the reservoir flood test problem in 2016, and Figure 5b shows the optimal non-dominated solutions obtained by the three algorithms in the reservoir flood test problem in 2021. It can be seen from Figure 5 that both the NSGA-II algorithm and the CFD-MOEA/D algorithm have good convergence in the two flood test problems, but the optimal non-dominated solutions obtained by the NSGA-II algorithm are all in the lower water level. For the upstream part, for the Pareto optimal front, it cannot be completely covered. For the CFD-MOEA/D algorithm, it can obtain a non-dominated solution with a higher upstream water level, so the solution obtained is more sufficient and can cover the entire Pareto
optimal front. As for the MOEA/D algorithm, its convergence in the flood test problem in 2016 is poor. Although it can obtain the optimal non-dominated solution for the higher upstream water level, the optimal non-dominated solution for the lower upstream water level is less likely to be obtained. Therefore, in general, the CFD-MOEA/D algorithm has the best effect in obtaining the optimal non-dominated solution. The HV comparison of the three algorithms is shown in Figure 6.

![Figure 6. Comparison of HV curves of three algorithms in flood regulation in 2016 and 2021.](image)

It can be seen from Figure 6 that, compared with the NSGA-II algorithm and the MOEA/D algorithm, the CFD-MOEA/D algorithm has a faster convergence speed and the best convergence effect. In Figure 6a, when the convergence state is reached, the HV value of the CFD-MOEA/D algorithm is approximately $3.68 \times 10^4$, while the HV value of NSGA-II algorithm and MOEA/D algorithm is approximately $2.0 \times 10^4$. In Figure 6b, when the convergence state is reached, the HV value of the CFD-MOEA/D algorithm is approximately $7.21 \times 10^4$, while the HV value of the NSGA-II algorithm is approximately $4.1 \times 10^4$, and the HV value of MOEA/D algorithm is approximately $3.45 \times 10^4$. The HV values obtained by the CFD-MOEA/D algorithm are significantly higher than those of the other two algorithms, indicating that the solution has better uniformity and convergence. In addition, the optimal dispatching scheme of the CFD-MOEA/D algorithm selected when the water level is 325 m is compared with the actual dispatching scheme of the reservoir, and the results are shown in Table 1.

**Table 1.** Comparison table of scheduling schemes generated by the CFD-MOEA/D algorithm and scheduling rules.

| Flood Problem | 2008          | 2016          |
|---------------|---------------|---------------|
| Qmax          | 18,100        | 13,250        |
| Zmax          | 325.11        | 324.557       |
| Zend          | 318.034       | 321.142       |
| Qmax          | 18,000        | 12,500        |

In Table 1, $Q_{\text{max}}$, $Q_{\text{max}}$, $Z_{\text{max}}$, and $Z_{\text{end}}$ are the maximum inbound flow, the maximum discharge flow, the maximum upstream water level, and the end-of-period water level, respectively. It can be seen from Table 1 that in different years, the maximum leakage flow obtained by the CFD-MOEA/D algorithm is significantly lower than that of the actual
scheduling scheme, indicating that the algorithm can obtain a stable and small leakage flow sequence. In addition, the maximum upstream water level and the final water level obtained by the CFD-MOEA/D algorithm are kept at approximately 325 m, which is consistent with the actual dispatching plan, indicating that the algorithm can reduce the discharge volume while ensuring the water level of the reservoir to meet the needs of many. The solution of a goal demonstrates certain advantages. From the characteristics of the flood control scheduling problem of the reservoir, it can be seen that the scheduling time interval will also have an impact on its performance. Therefore, taking the flood in 2021 as an example, the CFD-MOEA/D algorithm is used to solve the scheduling problem with different scheduling time intervals. The intervals are set to eight, six, four, and three hours, respectively, and the optimal drain flow sequence obtained is shown in Figure 7.

![Figure 7](image-url)

**Figure 7.** The sequence diagram of optimal drainage flow obtained when the dispatching interval is 8 h, 6 h, 4 h and 3 h respectively.

It can be seen from Figure 7 that although the set scheduling time intervals are different, the trend of the optimal scheduling scheme for each interval is consistent. For example, in Figure 7a, when the interval is eight hours, there is a clear turning point at the eighth variable, which corresponds to a time of 64 h. In Figure 7b, when the interval is six hours, there is an obvious turning point on the right side of the 10th variable, and the corresponding time is also 64 h. In Figure 7c, when the interval is four hours, there is an obvious turning point at the 16th variable, and its corresponding time is also 64 h. In Figure 7d, when the interval is 3 hours, there is an obvious turning point on the right side
of the 21st variable, and the corresponding time is also 64 h. Therefore, the result shows that the size of the scheduling time interval does not affect the overall trend of the optimal scheduling scheme. Different scheduling time intervals are set for the two floods, and the CFD-MOEA/D algorithm is used to solve and analyze, and the number and difficulty of decision variables are counted, as shown in Table 2.

Table 2. Difficulty analysis of flood problem with different dispatching time intervals.

| Scheduling Interval | 2016 | 2021 |
|---------------------|------|------|
|                     | Number of Decision Variables | Convergence or Not | Number of Decision Variables | Convergence or Not |
| 1 h                 | 92   | ×    | 144   | ×    |
| 3 h                 | 33   | √    | 49    | ×    |
| 4 h                 | 25   | √    | 35    | ×    |
| 6 h                 | 15   | √    | 25    | √    |
| 8 h                 | 12   | √    | 18    | √    |
| 12                  | 8    | √    | 12    | √    |

In Table 2, √ means successful convergence, × means convergence failure. It can be seen from the table that the smaller the scheduling time interval, the greater the number of corresponding variables, and the greater the difficulty of convergence. In addition, the number of decision variables also determines whether the algorithm can converge into the true Pareto front, that is, when the number is less than 30, the convergence and effect are the best, and the scheduling time interval is approximately three to six hours. In the performance evaluation of flood scheduling, in addition to the influence of the number of function evaluations on the performance, the number of sub-problems also has a certain influence on it. Therefore, the influence effect is examined by setting different numbers of sub-problems, as shown in Figure 8.

Figure 8. Parameter sensitivity analysis of different subproblems in two floods in 2016 and 2021.

It can be seen from Figure 8 that in Figure 8b, when the number of sub-problems is two, the fluctuation of the HV value is large. When the number of sub-problems is greater than two, the HV value is less affected, and the HV values obtained by the CFD-MOEA/D algorithm have little difference, indicating that after the algorithm evolves the first coarse-grained sub-problem, the search space can be effectively narrowed. After setting
the appropriate number of evaluations and the number of sub-questions, it can obtain better performance.

5. Conclusions

Floods pose a serious threat to people’s lives and property safety, and with the change of the global climate, floods occur more and more frequently. Therefore, a large-scale multi-objective reservoir flood control scheduling algorithm is designed to solve the flood scheduling problem. When evaluating its performance, the same type of non-dominated sorting genetic algorithm (NSGA-II) was introduced for comparative analysis. The research results show that, compared with the NSGA-II algorithm and the MOEA/D algorithm, the CFD-MOEA/D algorithm can obtain a non-dominated solution with a higher upstream water level, and the obtained solution is more sufficient and can cover the entire Pareto optimal front, so it has a better effect. When analyzing the HV value curve, the uniformity and convergence of the optimal solution obtained by the CFD-MOEA/D algorithm are better than those of the other two algorithms. For example, in the simulation experiment for the flood problem in 2016, the HV value of the CFD-MOEA/D algorithm when it reaches a state of convergence is approximately $3.68 \times 10^4$, which is significantly higher than that of the other two algorithms. The maximum upstream water level and the end-of-period water level obtained by the optimal dispatching scheme of the CFD-MOEA/D algorithm are both kept at approximately 325 m, which is consistent with the actual dispatching scheme. In addition, it is found that different scheduling time intervals have little effect on the optimal scheduling scheme, and better algorithm performance can be obtained by setting appropriate evaluation times and sub-problems. However, there is still room for improvement in the research. For example, when setting parameters, an adaptive method can be adopted to improve the flexibility of the algorithm.

Author Contributions: Conceptualization, H.J. and H.W.; methodology, H.W.; software, H.J.; validation, H.W., Q.Y. and M.L.; formal analysis, H.J.; investigation, Q.Y.; re-source, H.W.; data curation, M.L.; writing—original draft preparation, H.J.; writing—review and editing, H.W.; visualization, Q.Y.; supervision, H.J.; project administration, H.J.; funding acquisition, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Y.; Wu, M.; Ye, X.; Li, W.; Xue, R.; Wang, D.; Zhang, H.; Fan, D. An efficient scheduling algorithm for dataflow architecture using loop-pipelining. Inf. Sci. 2021, 547, 1136–1153. [CrossRef]
2. León, J.; Chullo-Llave, B.; Enciso-Rodas, L.; Soncco-Álvarez, J.L. A Multi-Objective Optimization Algorithm for Center-Based Clustering. Electron. Notes Theor. Comput. Sci. 2020, 349, 49–67. [CrossRef]
3. Dong, Z.; Wang, X.; Tang, L. Color-coating scheduling with a multi-objective evolutionary algorithm based on decomposition and dynamic local search. IEEE Trans. Autom. Sci. Eng. 2020, 18, 1590–1601. [CrossRef]
4. Zhao, Y.; Wang, Y.; Tan, Y.; Zhang, J.; Yu, H. Dynamic Jobshop Scheduling Algorithm Based on Deep Q Network. IEEE Access 2021, 9, 122995–123011. [CrossRef]
5. Grosch, B.; Kohne, T.; Weigold, M. Multi-objective hybrid genetic algorithm for energy adaptive production scheduling in job shops. Procedia CIRP 2021, 98, 294–299. [CrossRef]
6. Rongwei, C.U.I.; Wei, H.A.N.; Xichao, S.U.; Hongyu, L.I.A.N.G.; Zhengyang, L.I. A dual population multi-operator genetic algorithm for flight deck operations scheduling problem. J. Syst. Eng. Electron. 2021, 32, 331–346. [CrossRef]
7. Pet, J.; Liu, X.; Fan, W.; Pardalos, P.M.; Lu, S. A hybrid BA-VNS algorithm for coordinated serial-batching scheduling with deteriorating jobs, financial budget, and resource constraint in multiple manufacturers. Omega 2019, 82, 55–69. [CrossRef]
8. Li, X.; Xing, K. Iterative Widen Heuristic Beam Search Algorithm for Scheduling Problem of Flexible Assembly Systems. IEEE Trans. Ind. Inform. 2021, 17, 7348–7358. [CrossRef]
9. Wang, Y.; Guo, Y.; Wang, W.; Liang, H.; Huo, S. INHIBITOR: An intrusion tolerant scheduling algorithm in cloud-based scientific workflow system. Futur. Gener. Comput. Syst. 2021, 114, 272–284. [CrossRef]
10. Iqbal, F.; Iqbal, K. Optimal load scheduling using genetic algorithm to improve the load profile. arXiv 2021, arXiv:2111.14634.
11. Niu, C.; Wang, L. Big data-driven scheduling optimization algorithm for Cyber–Physical Systems based on a cloud platform. Comput. Commun. 2022, 181, 173–181. [CrossRef]
12. Hamsaveni, M.; Choudhary, S. A multi-objective optimization algorithm for routing path selection and wavelength allocation for dynamic WDM network using MO-HLO. *Int. J. Eng. Adv. Technol.* 2021, 10, 111–118.

13. Bose, S.; Nandi, T. Statistical and experimental investigation using a novel multi-objective optimization algorithm on a novel titanium hybrid composite developed by lens process. *Proc. Inst. Mech. Eng. Part C Mech. Eng. Sci.* 2021, 235, 2911–2933. [CrossRef]

14. Bidgoli, A.A.; Ebrahimpour-Komleh, H.; Rahnamayan, S. Reference-point-based multi-objective optimization algorithm with opposition-based voting scheme for multi-label feature selection. *Inf. Sci.* 2021, 547, 1–17. [CrossRef]

15. Bao, J.; Liu, X.; Xiang, Z.; Wei, G. Multi-Objective Optimization Algorithm and Preference Multi-Objective Decision-Making Based on Artificial Intelligence Biological Immune System. *IEEE Access* 2020, 8, 160221–160230. [CrossRef]

16. Kaveh, A.; Moghanni, R.M.; Javadi, S.M. Ground motion record selection using multi-objective optimization algorithms: A comparative study. *J. Neurosurg. Sci.* 2019, 63, 812–822. [CrossRef]

17. Yang, Z.; Ding, Y.; Jin, Y.; Hao, K. Immune-endocrine system inspired hierarchical coevolutionary multiobjective optimization algorithm for IoT service. *IEEE Trans. Cybern.* 2018, 50, 164–177.

18. Song, X. Intelligent English translation system based on evolutionary multi-objective optimization algorithm. *J. Intell. Fuzzy Syst.* 2020, 40, 6327–6337. [CrossRef]

19. Shi, F.; Wang, H.; Lu, T.; Wang, C. Multi-Objective Optimal Design of Excitation Systems of Synchronous Condensers for HVDC Systems Based on MOEA/D. In Proceedings of the 13th International Conference on Machine Learning and Computing, Shenzhen, China, 26 February–1 March 2021; pp. 575–581.

20. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* 1999, 3, 257–271. [CrossRef]

21. Gu, F.; Liu, H.L.; Cheung, Y.M.; Zheng, M. A rough-to-fine evolutionary multiobjective optimization algorithm. *IEEE Trans. Cybern.* 2021, 7, 1–14.

22. Sharifi, M.R.; Akbarifard, S.; Qaderi, K.; Madadi, M.R. A new optimization algorithm to solve multi-objective problems. *Sci. Rep.* 2021, 11, 20326. [CrossRef] [PubMed]

23. Hu, R.; Zhang, Z.; Ma, X.; Jin, Y. Dynamic rebalancing optimization for bike-sharing system using priority-based MOEA/D algorithm. *IEEE Access* 2021, 9, 27067–27084. [CrossRef]

24. Van Quang, N.; Duyen, H.T. Convergence of weighted sums and strong law of large numbers for convex compact integrable random sets and fuzzy random sets. *J. Convex Anal.* 2017, 24, 213–238.

25. Mandal, W.A. Weighted Tchebycheff optimization technique under uncertainty. *Ann. Data Sci.* 2021, 8, 709–731. [CrossRef]

26. Datta, S.; Ghosh, A.; Sanyal, K.; Das, S. A Radial Boundary Intersection aided interior point method for multi-objective optimization. *Inf. Sci.* 2017, 377, 1–16. [CrossRef]

27. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T.A.M.T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* 2002, 6, 182–197. [CrossRef]