Engineering Optimization

ISSN: 0305-215X (Print) 1029-0273 (Online) Journal homepage: http://www.tandfonline.com/loi/geno20

Optimal reservoir flood operation using a decomposition-based multi-objective evolutionary algorithm

Xiao Zhang, Jungang Luo, Xiaomei Sun & Jiancang Xie

To cite this article: Xiao Zhang, Jungang Luo, Xiaomei Sun & Jiancang Xie (2019) Optimal reservoir flood operation using a decomposition-based multi-objective evolutionary algorithm, Engineering Optimization, 51:1, 42-62, DOI: 10.1080/0305215X.2018.1439942

To link to this article: https://doi.org/10.1080/0305215X.2018.1439942

Published online: 09 Mar 2018.

Submit your article to this journal

Article views: 72

View Crossmark data

Full Terms & Conditions of access and use can be found at
http://www.tandfonline.com/action/journalInformation?journalCode=geno20

ENGINEERING OPTIMIZATION
2019, VOL. 51, NO. 1, 42–62
https://doi.org/10.1080/0305215X.2018.1439942

Taylor & Francis
Taylor & Francis Group

Check for updates

Optimal reservoir flood operation using a decomposition-based multi-objective evolutionary algorithm

Xiao Zhang, Jungang Luo, Xiaomei Sun and Jiancang Xie

State Key Laboratory of Eco-hydraulics in Northwest Arid Region of China, Xi'an University of Technology, Xi'an, Shaanxi Province, People's Republic of China

ABSTRACT

Reservoir flood control operation (RFCO) is a challenging optimization problem with interdependent decision variables and multiple conflicting criteria. By considering safety both upstream and downstream of the dam, a multi-objective optimization model is built for RFCO. To solve this problem, a multi-objective optimizer, the multi-objective evolutionary algorithm based on decomposition–differential evolution (MOEA/D-DE), is developed by introducing a differential evolution-inspired recombination into the algorithmic framework of the decomposition-based multi-objective optimization algorithm, which has been proven to be effective for solving complex multi-objective optimization problems. Experimental results on four typical floods at the Ankang reservoir illustrated that the suggested algorithm outperforms or performs as well as the comparison algorithms. It can significantly reduce the flood peak and also guarantee the dam's safety.

ARTICLE HISTORY

Received 5 December 2016
Accepted 1 February 2018

KEYWORDS

Reservoir flood control operation; multi-objective evolutionary algorithm; decomposition; differential evolution

1. Introduction

Reservoir flood control operation (RFCO) is a scheduling problem which aims to minimize flood peaks, reduce flood damage and prevent floods by providing appropriate scheduling plans for the dams' water release sequences. Since a flood is a damaging natural disaster with a high frequency and devastating force, RFCO is a problem worth investigating which has attracted a large amount of research effort. During floods, the most important target of RFCO is to ensure safety both upstream and downstream of the dam. Given a determined inflow flood sequence, these two optimization goals can be conflict with each other. Therefore, the RFCO problem is a challenging multi-objective optimization problem (MOP) which involves interdependent decision variables (Reddy and Kumar 2007).

In contrast to single-objective RFCO approaches, such as linear programming (Needham *et al.* 2000), dynamic programming (Yakowitz 1982) and nonlinear programming (Unver and Mays 1990), multi-objective RFCO algorithms can optimize multiple conflicting optimization goals simultaneously and thus provide the decision makers with a set of Pareto-optimal scheduling schemes (Qi *et al.* 2012). With the rapid development of multi-objective optimization techniques, more and more research has been conducted on multi-objective water resource planning and management (Liu 2009; Malekmohammadi, Zahraie, and Kerachian 2011; Xu *et al.* 2012), of which the multi-objective RFCO problem is one of the most important branches (Hajkowicz and Collins 2007).

In recent years, many newly developed approaches have been proposed to solve the multi-objective RFCO problem. Hou and Chen (2004) investigated existing decision-making problems with multiple optimization goals, and summarized the theory and applications of multi-criteria decision making

CONTACT Jungang Luo jgluo@xaut.edu.cn

© 2018 Informa UK Limited, trading as Taylor & Francis Group

for RFCO. Using optimum models based on fuzzy theory, some fuzzy decision-making methods concerning multiple objectives were developed for the RFCO problem (Fu 2008; Hou and Chen 2004; Yu *et al*. 2004; Zhou, Zhang, and Wang 2007). Qin and colleagues suggested a bi-objective optimization model for the RFCO problem, and developed two optimizers, based on differential evolution (DE) (Qin *et al*. 2009) and cultured DE (Qin *et al*. 2010), to solve it. Multi-objective evolutionary algorithms (MOEAs), which have been recognized as efficient optimizers for MOPs, were also applied to solve multi-objective RFCO problems. Using the outstanding non-dominated sorting genetic algorithm II (NSGA-II), Kim, Heo, and Jeong (2006) solved a four interconnected reservoir operation problem with two conflicting optimization goals. To optimize the rule curves of a multi-purpose reservoir, Chen, McPhee, and Yeh (2007) developed a specially designed macro-evolutionary multi-objective genetic algorithm. Considering a three-objective multi-reservoir system, Hakimi-Asiabar, Ghodsypour, and Kerachian (2010) designed a self-learning genetic algorithm by improving the self-organizing map-based multi-objective genetic algorithm. By combining NSGA-II with the multi-layer perception neural network, Shokri, Bozorg, and Mariño (2014) developed an efficient multi-objective optimizer to extract the best set of reservoir operation decisions.

In addition to MOEAs, other multi-objective optimization techniques, such as the multi-objective particle swarm optimization algorithm (Baltar and Fontane 2008), multi-objective ant colony optimization algorithm (Afshar, Sharifi, and Jalali 2009), multi-objective frog leaping algorithm (Dumedah *et al*. 2010), multi-objective electromagnetism-like mechanism algorithm (Ouyang *et al*. 2014) and multi-objective immune algorithm with preference-based selection (Luo, Chen, and Xie 2015), have been applied to solve the RFCO problem. In these studies, MOEAs have played a role in solving multi-objective RFCO problems.

MOEAs, which are inspired by Darwin's theory of evolution, have achieved significant success in solving MOPs (Zhou *et al*. 2011). By evolving a population of solutions, MOEAs are able to obtain a set of best trade-off solutions to MOPs in a single run. Zhang and Li (2007) developed an excellent algorithmic framework for solving MOPs by combining traditional decomposition-based methods in mathematics with evolutionary computation. This multi-objective evolutionary algorithm based on decomposition (MOEA/D) decomposes the target MOPs into a set of scalar optimization problems and then optimizes them simultaneously using an evolutionary algorithm. Many studies have been carried out on enhancing the performance of MOEA/D (Ishibuchi *et al*. 2010; Li and Landa-Silva 2011; Qi *et al*. 2014). Owing to its simplicity and outstanding performance, MOEA/D has been widely investigated and successfully applied to various challenging MOPs (Pal *et al*. 2010; Qi, Hou, Li, *et al*. 2015; Waldock and Corne 2011).

A previous study indicated that the RFCO is a challenging MOP which is complex both in the objective space and in the decision space (Qi *et al*. 2016). In the RFCO problem, an improvement in one objective is usually accompanied by an unstable degradation in another. As a result, the RFCO problem usually has an irregularly shaped Pareto front. The complexity of the Pareto front's shape in the objective space is a great challenge to most multi-objective optimization algorithms. The RFCO problem takes a sequence of water release volumes as the decision variables, which have a chain-like interdependence with each other. The complexity of interdependence between decision variables is the major source of complexity in many optimization problem, and it also presents major challenges to multi-objective optimization algorithms.

Considering the complexity of the RFCO problem in both the objective space and the decision space, a powerful multi-objective optimizer is proposed by introducing a DE-inspired recombination (Qi, Hou, Yin, *et al*. 2015) into the algorithmic framework of the MOEA/D. Owing to the good diversity of the decomposition weights in MOEA/D, the MOEA/D algorithm can provide a set of Pareto-optimal solutions with satisfactory coverage and uniformity over the target Pareto front. Therefore, to cope with the complexity of the RFCO problem in the objective space, this work follows the algorithmic framework of the MOEA/D. However, the complexity of the RFCO problem in the decision space prevents the population of the MOEA/D algorithm from converging on to the target Pareto front quickly. To develop an efficient search mechanism in the decision space, the DE-inspired recombination (Qi, Hou, Yin, *et al.* 2015) is employed to generate new offspring with better qualities based on the individuals in the evolving population, giving rise to the proposed multi-objective evolutionary algorithm based on decomposition–differential evolution (MOEA/D-DE) for solving the RFCO problem.

The rest of this article is organized as follows. Section 2 describes the investigated bi-objective optimization model for RFCO, after introducing some related terms on multi-objective optimization. Section 3 presents details of the proposed MOEA/D-DE for solving multi-objective RFCO problems. Section 4 briefly presents and analyses the experimental results of MOEA/D-DE and the compared algorithms on four investigated floods. Section 5 concludes the article.

2. RFCO with multiple conflicting criteria

The RFCO problem is a challenging MOP. In this section, some terms related to MOPs are first introduced, and then the multi-objective optimization model for the RFCO problem is mathematically described.

2.1. MOP and related terms

Many real-world optimization problems, such as the RFCO problem, involve more than one conflicting optimization goal. Such problems are known as MOPs. An MOP, taking a minimization problem for example, can be formally defined as:

$$\begin{array}{ll} \text{Minimize} & \mathbf{F}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})\} \\ \text{Subject to} & \mathbf{x} \in \Omega \in R^n \end{array} \tag{1}$$

where $\Omega \in R^n$ is the n-dimensional decision space; $\mathbf{x}$ is the decision vector, which represents a solution to the target optimization problem; and $F : \Omega \to R^n$ consist of m conflicting objective functions. Owing to the conflicts between optimization goals, there is usually no unique solution that optimizes all the objectives. Instead, an MOP usually has a set of optimal trade-off solutions which are known as the Pareto-optimal solutions.

Supposing $\mathbf{x}_A, \mathbf{x}_B \in \Omega$ are two decision vectors in the decision space, $\mathbf{x}_A$ is said to dominate the other one $\mathbf{x}_B$ (written as $\mathbf{x}_A \prec \mathbf{x}_B$) if and only if $f_i(\mathbf{x}_A) \leq f_i(\mathbf{x}_B)$ for all the objective functions, and there exists an object $j = 1, 2, \ldots, m$ which makes $f_j(\mathbf{x}_A) < f_j(\mathbf{x}_B)$. A given solution $\mathbf{x}^* \in \Omega$ is termed a Pareto-optimal solution if there is no other solution in the decision space that dominates $\mathbf{x}^*$. All the Pareto-optimal solutions from the Pareto-optimal set and the collection of the corresponding objective vectors of all the Pareto-optimal solutions is termed the Pareto-optimal front.

Usually, it is time consuming or even impossible for the multi-objective optimizer to obtain all the Pareto-optimal solutions. Instead, multi-objective optimization algorithms aim to find a finite set of Pareto-optimal solutions which are evenly scattered over the whole Pareto front.

2.2. Bi-objective optimization model for RFCO

Considering the safety both upstream and downstream of the dam during floods, the following multi-objective optimization model is built and investigated for the RFCO problem (Luo, Chen, and Xie 2015; Qi *et al.* 2012; Qin *et al.* 2010):

$$\min \quad F(Q) = \{f_1(Q), f_2(Q)\} \tag{2}$$

where $f_1(Q)$ and $f_2(Q)$ are the two optimization goals; and $Q = (Q_1, Q_2, \ldots, Q_T)$ is the decision vector which represents the dams' discharge volume sequence during T scheduling periods. As the safety of a dam's upstream side depends on the flood water volume stored in the reservoir, the first optimization goal $f_1(Q)$ can be defined as the maximum upstream water level of the dam. Regarding the safety of the dam's downstream side, the discharge volume is the critical issue. Therefore, the second optimization goal $f_2(Q)$ is modelled as the maximum discharge volume of the dam. Both optimization goals should be minimized to guarantee the safety of the dam during floods. The objectives are expressed as:

$$\min f_1(Q) = \min\{\max(Z_t)\} \quad t = 1, 2, \cdots, T \tag{3}$$

$$\min f_2(Q) = \min\{\max(Q_t)\} \quad t = 1, 2, \cdots, T \tag{4}$$

where Z_t denotes the upstream water level of the tth scheduling period; and Q_t represents the discharge volume of the tth scheduling period. According to the water balance equation, the upstream water level Z_t can be obtained.

The optimization model includes four constraints: the upstream water level constraint, discharge volume constraint, water balance constraint and final reservoir water level constraint, which can be described as:

$$Z_t^{\min} \leq Z_t \leq Z_t^{\max} \tag{5}$$

$$Q_t^{\min} \leq Q_t \leq Q_t^{\max} \tag{6}$$

$$V_t = V_{t-1} + (I_t - Q_t)\Delta t \tag{7}$$

where $Z_t^{\min}$, $Z_t^{\max}$ are the minimum and maximum limit of the reservoir upstream water level of the tth period; $Q_t^{\min}$, $Q_t^{\max}$ are the minimum and maximum limit of the reservoir discharge volume of the tth period; V_t, V_{t-1} are the reservoir storage of the tth and $(t-1)$th periods; and I_t, Q_t are the reservoir inflow and discharge volume of the tth period.

3. Decomposition-based MOEA for RFCO

In this work, a DE-inspired recombination operator which performs well on continuous MOPs is introduced into the algorithmic framework of MOEA/D to form the proposed MOEA/D-DE for solving the above-defined multi-objective RFCO problem. In this section, the basic idea of MOEA/D and the DE-inspired recombination operator are first introduced. Then, the workflow of the proposed MOEA/D-DE is described in detail.

3.1. Decomposition-based MOEA

The MOEA/D suggested by Zhang and Li (2007) is a efficient algorithmic framework of a population evolving-based multi-objective optimizer. Many studies have confirmed the effectiveness of this algorithm, especially in solving complex MOPs. MOEA/D transforms the target MOPs into a set of single-objective optimization subproblems using decomposition techniques.

In this work, the Tchebycheff decomposition approach is used to decompose MOPs into subproblems using a set of weight vectors. Given a weight vector $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_m)$, where $\sum_{i=1}^{m} \lambda_i = 1$, $\lambda_i \geq 0$, $i = 1, \cdots, m$, and the reference point $\mathbf{z}^* = (z_1^*, \ldots, z_m^*)^T$, where $z_i^* = \min\{f_i(\mathbf{x}) | \mathbf{x} \in \Omega\}$, $i = 1, \cdots, m$, a single-objective optimization subproblem can be determined as:

$$\min_{\mathbf{x} \in \Omega} g^{tc}(\mathbf{x}|\boldsymbol{\lambda}, \mathbf{z}^*) = \min_{\mathbf{x} \in \Omega} \max_{1 \leq i \leq m} \{\lambda_i \times |f_i(\mathbf{x}) - z_i^*|\} \tag{8}$$

After decomposition, MOEA/D optimizes all the decomposed single-objective optimization subproblems simultaneously using an evolutionary algorithm. At each generation, MOEA/D keeps the optimal solution obtained for each subproblem to form the current population. MOEA/D also maintains a neighbourhood relationship between subproblems. Each subproblem has a neighbourhood list, and it is optimized by collaborations between its neighbouring subproblems.

In this work, the collaborations between subproblems are performed by the DE-inspired recombination operator, which is expected to be able to enhance the performance of MOEA/D in multi-objective RFCO problems.

3.2. DE-inspired recombination

In the original MOEA/D, the recombination operator, which is designed for single-objective optimization problems, is directly employed. Moreover, each single-objective optimization subproblem in the original MOEA/D is optimized using information from its neighbouring subproblems only, which ignores information from its current best solution. Therefore, the performance of MOEA/D could be expected to be enhanced by replacing its recombination operator with one that is specially designed according to the characteristics of the MOP. The current best solution of each subproblem should also be taken into consideration to enhance the efficiency of optimization.

Qi, Hou, Yin, *et al.* (2015) developed a DE-inspired recombination operator which utilizes the regularity of continuous MOPs and provides two types of candidate searching directions by taking three recombination parents. These two types of candidate searching directions are designed to complement each other: one of them guides the search to find new solutions along the current Pareto set, while the other redirects the search from the current Pareto set to another one.

Given the operated solution $\mathbf{x}$ and its two neighbours $\mathbf{x}^1$ and $\mathbf{x}^2$, the workflow of the DE-inspired recombination operator can be summarized by Algorithm 1.

Algorithm 1: The differential evolution-inspired recombination operator

Input: The operated solution $\mathbf{x}$, other two neighbouring solutions $\mathbf{x}^1$ and $\mathbf{x}^2$.
Output: The newly generated solution $\mathbf{x}'$.

Step 1 Reproducing: Generate a random number r between 0 and 1, if $r < 0.5$, then

$$\mathbf{x}' = \begin{cases} \mathbf{x} + 0.5 \times (\mathbf{x}^1 - \mathbf{x}^2) & \text{with probability } 0.9 \\ \mathbf{x} & \text{with probability } 0.1 \end{cases} \tag{9}$$

Otherwise, generate two random numbers $rand^1$ and $rand^2$

$$\mathbf{x}' = \begin{cases} \mathbf{x} + rand^1 \times (\mathbf{x} - \mathbf{x}^1) + rand^2 \times (\mathbf{x} - \mathbf{x}^2) & \text{with probability } 0.9 \\ \mathbf{x} & \text{with probability } 0.1 \end{cases} \tag{10}$$

Step 2 Repair: If any dimension of $\mathbf{x}'$ is outside the boundary of the feasible region, its value is reset to be a randomly selected value inside the boundary.
Step 3 Output: Output $\mathbf{x}'$ as the offspring solution.

In Algorithm 1, three recombination parents are taken into account; they are the operated solution $\mathbf{x}$ and its two neighbours $\mathbf{x}^1$ and $\mathbf{x}^2$. If $\mathbf{x}$ is the current solution of the optimized subproblem and $\mathbf{x}^1$, $\mathbf{x}^2$ are the current solutions of its two neighbouring subproblems, the DE-inspired recombination operator will take $\mathbf{x}$ into consideration when optimizing its corresponding subproblem. It can be seen that Algorithm 1 provides two candidate searching directions in the reproducing step described by Equations (9) and (10). Usually, the two neighbouring solutions of $\mathbf{x}$ locate along the current Pareto set; therefore, Equation (9) will guide the search along the current Pareto set from $\mathbf{x}$. Equation (10) provides a candidate searching direction apart from the current Pareto set, helping to explore new search areas and to escape from local optima.

3.3. The proposed algorithm: MOEA/D-DE

As described in Equation (2), the decision vector of the multi-objective RFCO problem is $Q = (Q_1, Q_2, \ldots, Q_T)$, which represents the dams' water release sequence during the T scheduling periods. In other words, here Q corresponds to $\mathbf{x}$ in Equation (1) and T corresponds to n in Equation (1), which is the dimension of the decision space. The details of the proposed MOEA/D-DE are described in Algorithm 2. At each iteration, MOEA/D-DE maintains the following items:

- an evolving population with N solutions $Q^1, Q^2, \cdots, Q^N$, in which $Q^i (i = 1, \cdots, N)$ is the current solution to the ith subproblem generated by the Tchebycheff decomposition approach described in Equation (9)

- the objective function values $FV^1, FV^2, \cdots, FV^N$, in which $FV^i(i = 1, \cdots, N)$ is the objective function value of solution Q^i in the evolving population
- a reference point $z* = (z_1*, z_2*)$, in which z_1* and z_2* are less than the best values obtained so far for the two optimization goals in Equations (3) and (4)
- an external population (EP) for the storage of non-dominated solutions during the search.

Algorithm 2: The proposed MOEA/D-DE

Input: The evolving population size N, a uniformly scattered set of N weight vectors $\lambda^1, \lambda^2, \cdots, \lambda^N$, the neighbourhood size T and a stopping criterion.
Output: The external population EP.

Step 1 Initialization: Generate an initial population $Q^1, Q^2, \cdots, Q^N$ at random and set $FV^i = F(Q^i)(i = 1, \cdots, N)$ in which function $\boldsymbol{F}$ is the objective function defined in Equation (2). Determine the neighbourhood list with size T for each subproblem. For the ith subproblem with weight vector $\lambda^i(i = 1, 2, \cdots, N)$, select its T closest weight vectors under the Euclidean distance to form the neighbourhood list $B(i) = \{i_1, i_2, \cdots, i_T\}$, where $\lambda^{i_1}, \cdots, \lambda^{i_T}$ are the T closest weight vectors to λ^i. Initialize $z^* = (z_1^*, z_2^*)$, where z_1* and z_2* are less than the smallest values of all the FV^i. Set the external population EP as an empty set.
Step 2 Evolution:
For $i = 1, \cdots, N$ do
Step 2.1 Recombination: Select two indices j and k from $B(i)$ at random, generate recombination offspring Q^i_{new} using three parent solutions Q^i, Q^j and Q^k, as follows. Generate a random number r between 0 and 1, if $r < 0.5$, then produce Q^i_{new} using the simulated binary crossover (SBX) operator (Zhang and Li 2007) taking Q^j and Q^k as recombination parents, as in the original MOEA/D. Otherwise, generate Q^i_{new} by the previously described Algorithm 1, taking Q^i, Q^j and Q^k as the inputs.
Step 2.2 Mutation: Apply the polynomial mutation operator (Zhang and Li 2007) on Q^i_{new}, giving rise to the mutation offspring $Q^i_{\text{new}'}$.
Step 2.3 Repair: If any dimension of $Q^i_{\text{new}'}$ is outside the boundary of the feasible region, its value is reset to be a randomly selected value inside the boundary. Notate the repaired solution as $Q^i_{\text{new}''}$.
Step 2.4 Update reference point: For the two optimization goals defined in Equations (3) and (4), if $f_1(Q^i_{\text{new}''}) < z_1*$ satisfies, then let $z_1* = f_1(Q^i_{\text{new}''}) - 10^{-7}$. Likewise, if $f_2(Q^i_{\text{new}''}) < z_2*$, then set $z_2* = f_2(Q^i_{\text{new}''}) - 10^{-7}$.
Step 2.5 Update neighbouring solutions: For each subproblem index $s \in B(i)$, if $g^{tc}(Q^i_{\text{new}''}|\boldsymbol{\lambda}^s, \mathbf{z}^*) \leq g^{tc}(Q^s|\boldsymbol{\lambda}^s, \mathbf{z}^*)$, then set $Q^s = Q^i_{\text{new}''}$ and $FV^s = F(Q^i_{\text{new}''})$.
Step 2.6 Update EP: Remove all the vectors which are dominated by $F(Q^i_{\text{new}''})$ from EP. If no vectors in EP dominate $F(Q^i_{\text{new}''})$, then add $F(Q^i_{\text{new}''})$ to EP.
Step 3 Stopping criterion: If the stopping criterion is met, then stop and output EP. Otherwise, go to Step 2.

The proposed MOEA/D-DE follows the main framework of MOEA/D and differs in Step 2.1. In Step 2.1, the SBX operator performs a local search combined with a random search near the recombination parents. On the other hand, the DE-inspired recombination operator uses the regularity of continuous MOPs; it provides candidate search directions along the current Pareto-optimal set and towards the ideal Pareto-optimal set to maintain diversity and accelerate the convergence rate, respectively.

4. Experimental studies

In this section, four benchmark problems with irregularly shaped Pareto fronts and four typical floods of different types at the Ankang reservoir in Shaanxi Province, China, are investigated. The performances of the proposed MOEA/D-DE on the investigated floods are compared with the original MOEA/D to illustrate the efficiency of the enhancement made by this work.

4.1. Benchmark problems and RFCO study cases

The following four benchmark problems with irregularly shaped Pareto fronts (Gu, Liu, and Tan 2012) and the UF suite of benchmark problems in the 2009 IEEE Congress on Evolutionary Computation (CEC 2009) competition are investigated in this work to compare the performance of the proposed

Table 1. Definitions of the four investigated benchmark problems.

Problem	Description	Notes
F1	$f_1(x) = (1 + g(x))x_1$ $f_2(x) = (1 + g(x))(2 - x_1 - \text{sign}(\cos(2\pi x_1))$ $g(x) = \sum_{i=2}^{n}\left(x_i - \cos\left(2\pi x_1 + \frac{i\pi}{n}\right)\right)^2$	$\mathbf{x} \in [0,1] \times [-1,1]^{n-1} n = 10$ bi-objective problem
F2	$f_1(x) = (1 + g(x))\left(1 - \cos\left(\frac{x_1\pi}{2}\right)\right)$ $f_2(x) = (1 + g(x))\left(10 - 10\sin\left(\frac{x_1\pi}{2}\right)\right)$ $g(x) = \sum_{i=2}^{n}\left(x_i - \cos\left(2\pi x_1 + \frac{i\pi}{n}\right)\right)^2$	$\mathbf{x} \in [0,1] \times [-1,1]^{n-1} n = 10$ bi-objective problem
F3	$f_1(x) = (1 + g(x))x_1$ $f_2(x) = \begin{cases}(1 + g(x))(1 - 19x_1), f_1 \leq 0.05 \\ (1 + g(x))\left(\frac{1}{19} - \frac{x_1}{19}\right), f_1 > 0.05\end{cases}$ $g(x) = \sum_{i=2}^{n}\left(x_i - \cos\left(2\pi x_1 + \frac{i\pi}{n}\right)\right)^2$	$\mathbf{x} \in [0,1] \times [-1,1]^{n-1} n = 10$ bi-objective problem
F4	$f_1(x) = (1 + g(x))x_1$ $f_2(x) = (1 + g(x))(2 - 2{x_1}^{0.5}\cos^2(2\pi {x_1}^{0.5})$ $g(x) = \sum_{i=2}^{n}\left(x_i - \cos\left(2\pi x_1 + \frac{i\pi}{n}\right)\right)^2$	$\mathbf{x} \in [0,1] \times [-1,1]^{n-1} n = 10$ bi-objective problem

MOEA/D-DE with the original MOEA/D, an improved version of MOEA/D, called dynamic multi-objective evolutionary algorithm based on decomposition (DMOEA/D) (Gu, Liu, and Tan 2012) and the well-known multi-objective optimization algorithm NSGA-II (Deb *et al.* 2002). Definitions of the investigated benchmark problems with irregularly shaped Pareto fronts in Gu, Liu, and Tan (2012) are listed in Table 1.

Ankang reservoir, located on the upper reach of Hanjiang river, is the largest hydro-junction project in Shaanxi Province. It plays important roles in flood control, power generation, shipping and other aspects, among which flood control is the most important task. Ankang reservoir has a maximum storage of 2.3×10^9 m^3 and a dam height of 128 m. Its designed flood water level is 333 m, check flood water level 337.05 m, normal water level 330 m, flood limit water level 325 m and dead water level 300 m. The designed flood peak discharge of the dam at Ankang reservoir is 36,700 m^3/s, the check flood peak discharge is 45,000 m^3/s and the maximum discharge is 37,474 m^3/s.

To validate the effectiveness of the suggested algorithm, four typical floods of different types are chosen as study cases. These floods occurred at Ankang reservoir on 12 October 2000, 28 August 2003, 1 October 2005 and 15 July 2010. Figure 1 illustrates the inflow volumes of the four investigated floods. It can be seen that three of the four floods (the floods on 12 October 2000, 1 October 2005 and 15 July 2010) have one flood peak, and they have different peak values, curve shapes and total inflow water volumes. The flood on 28 August 2003 has two flood peaks, with relatively low peak values.

4.2. Performance metric

In these experimental studies, two comprehensive performance metrics, the inverted generational distance (IGD) metric and the hyper-volume (HV) metric (Zitzler and Thiele 1998), are used to evaluate the performances of the compared algorithms. For benchmark problems for which the ideal Pareto fronts are available, the IGD metric is used, whereas for RFCO problems for which the ideal Pareto fronts are unknown, the HV metric is used.

The IGD is a commonly used comprehensive metric, which can be formally defined as follows. Let $\mathbf{P}^* = \{p_1^*, p_2^*, \cdots, p_{|\mathbf{P}^*|}^*\}$ be a set of evenly distributed solutions on the entire Pareto front. Given

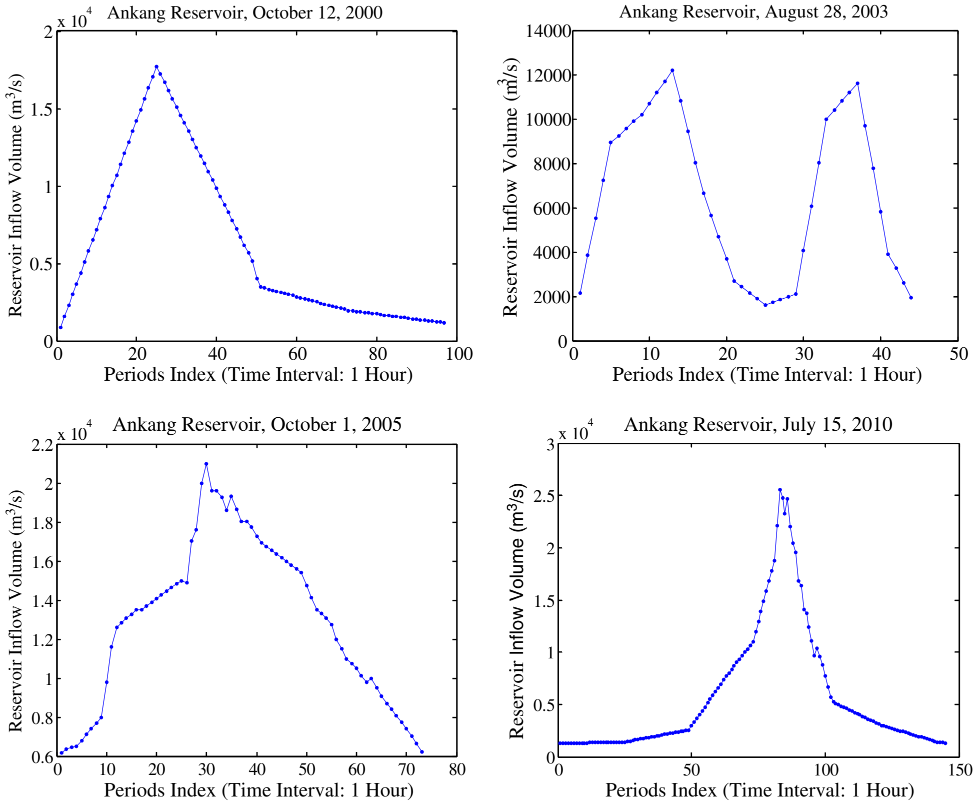

Figure 1. Inflow volumes of the four investigated floods at Ankang reservoir.

a solution set $\mathbf{P} = \{p_1, p_2, \cdots, p_{|\mathbf{P}|}\}$ approximating the Pareto front, the IGD value of $\mathbf{P}$ can be calculated by:

$$\mathrm{IGD}(\mathbf{P}^*, \mathbf{P}) = \frac{1}{|\mathbf{P}^*|} \sum_{i=1}^{|\mathbf{P}^*|} \min_{j=1}^{|\mathbf{P}|} d(p_i^*, p_j) \tag{11}$$

where $d(p_i^*, p_j)$ denotes the Euclidean distance measured from each point in $\mathbf{P}^*$ to its closest point in $\mathbf{P}$ in the objective space.

The HV metric, which is a comprehensive index of convergence, coverage and uniformity, is used to evaluate the performances of the compared algorithms on RFCO problems. The HV metric has been recognized as an objective performance metric for multi-objective optimization algorithms when the ideal Pareto front of the target problem is unknown.

For the minimization multi-objective RFCO problem defined in Equation (2), let $Q^1, Q^2, \cdots, Q^N$ be an approximation to the problem's ideal Pareto front. The HV metric is a measure of the region that is simultaneously dominated by $Q^1, Q^2, \cdots, Q^N$ and bounded above by a reference point $R = (r_1, r_2)$, in which $r_1 > \max\{f_1(Q^i)\}$ and $r_2 > \max\{f_2(Q^i)\}$, $i = 1, 2, \cdots, N$.

Figure 2 is an illustration of the HV metric. The HV value of $Q^1, Q^2, \cdots, Q^N$ with reference point R corresponds to the set of objective vectors within the hatched area in Figure 2, which is dominated by $Q^1, Q^2, \cdots, Q^N$ and enclosed by the reference point R. The larger the HV metric value, the better the algorithm performs.

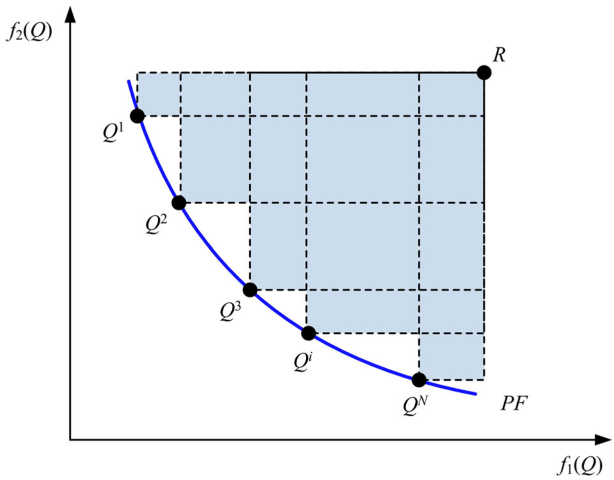

Figure 2. Illustration of the hyper-volume (HV) metric.

4.3. Experimental results and discussion

In the following experimental studies, the parameters of MOEA/D-DE and MOEA/D are consistent with those in Zhang and Li (2007) to enable fair comparison. All algorithms in this study use the same population size of 100 and the same simulated binary crossover (SBX) operator and polynomial mutation operators. In the SBX operator, the crossover probability (pc) is set to 1.0 and the distribution index (ηc) is set to 20. In the polynomial mutation operator, the mutation probability (pm) is set to $1/n$, where n is the number of decision variables. The distribution index (ηm) is set to 20. The maximum number of function evaluations is set to 100,000 for problems F1–F4 and 300,000 for problems UF1–UF4. All the experimental results are averaged over 30 independent runs and the figures in the experimental studies indicate the best result among the 30 runs.

By investigating the four representative floods, the performance of the proposed MOEA/D-DE is compared with MOEA/D (Zhang and Li 2007), DMOEA/D (Gu, Liu, and Tan 2012) and NSGA-II (Deb *et al*. 2002) with the aim of validating the effectiveness of the suggested improvements made in MOEA/D-DE. In the following experimental studies, the parameters of MOEA/D-DE and MOEA/D are consistent with those in Zhang and Li (2007). The parameter settings of the DMOEA/D and NSGA-II algorithms are the same as reported in Gu, Liu, and Tan (2012) and Deb *et al*. (2002), respectively.

4.3.1. Experimental studies on benchmark problems with complex Pareto fronts

Experimental studies on the four investigated benchmark problems with complex Pareto fronts were conducted to make comparisons between the proposed MOEA/D-DE and the other three baseline algorithms, MOEA/D, DMOEA/D and NSGA-II. Figure 3 illustrates the distribution of the best approximation of Pareto sets with the lowest IGD values obtained by the four compared algorithms. It can be seen that MOEA/D-DE obtains solution sets with better convergence than those obtained by MOEA/D and NSGA-II on problem F1. On the other hand, MOEA/D-DE outperforms MOEA/D and NSGA-II on problems F2, F3 and F4 in terms of uniformity. Compared with DMOEA/D, MOEA/D-DE performs as well as DMOEA/D on problems F1, F2 and F4. For problem F3, MOEA/D-DE performs better in terms of uniformity.

Table 2 compares the performance of MOEA/D-DE, MOEA/D, DMOEA/D and NSGA-II on the four benchmark problems in terms of the IGD metric. The mean and standard deviation of IGD values are presented, in which the best results among the compared algorithms are highlighted in bold.

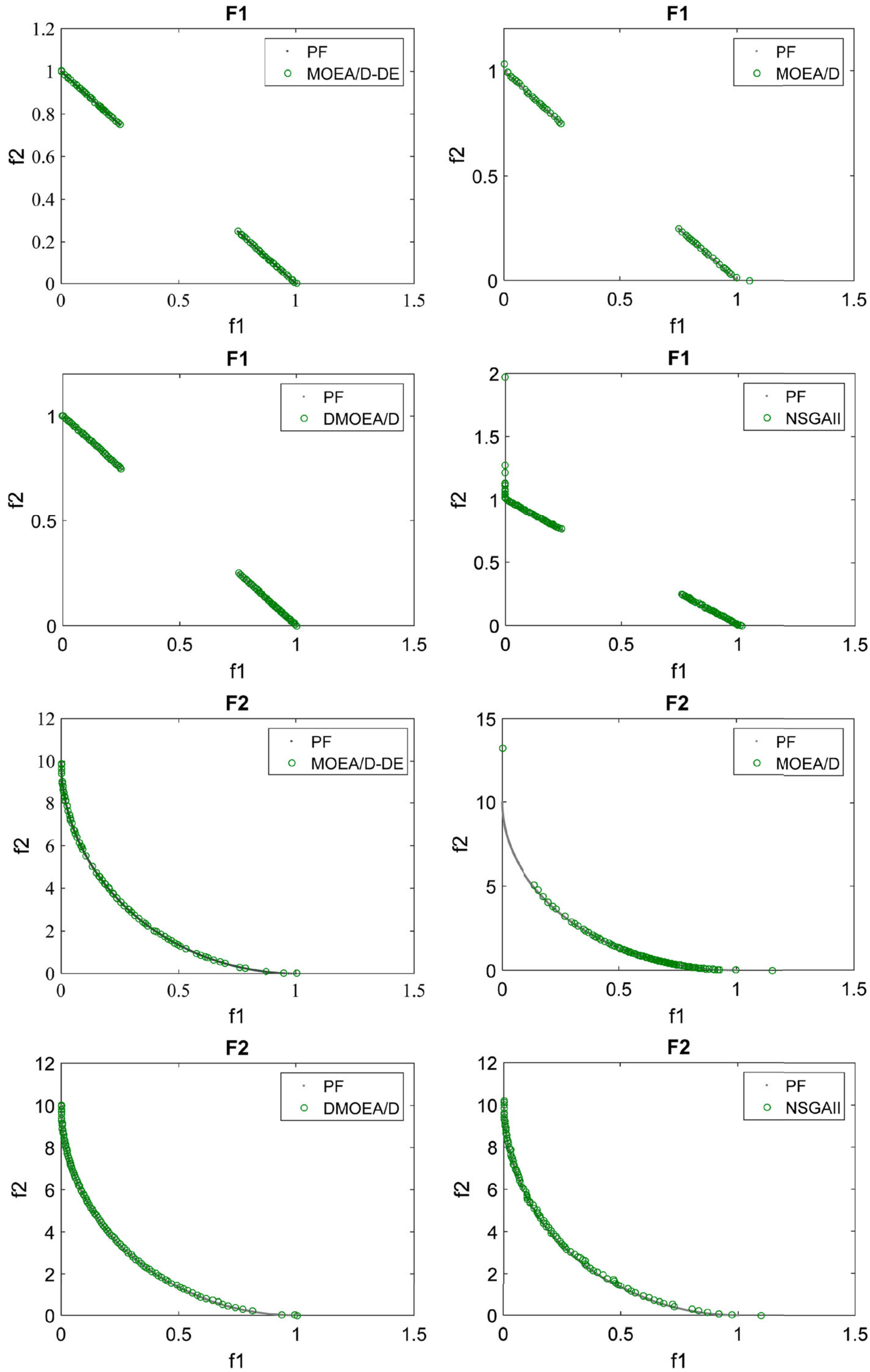

Figure 3. Best approximation of Pareto fronts (PFs) obtained by the compared algorithms on the four benchmark problems with complex Pareto fronts. MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

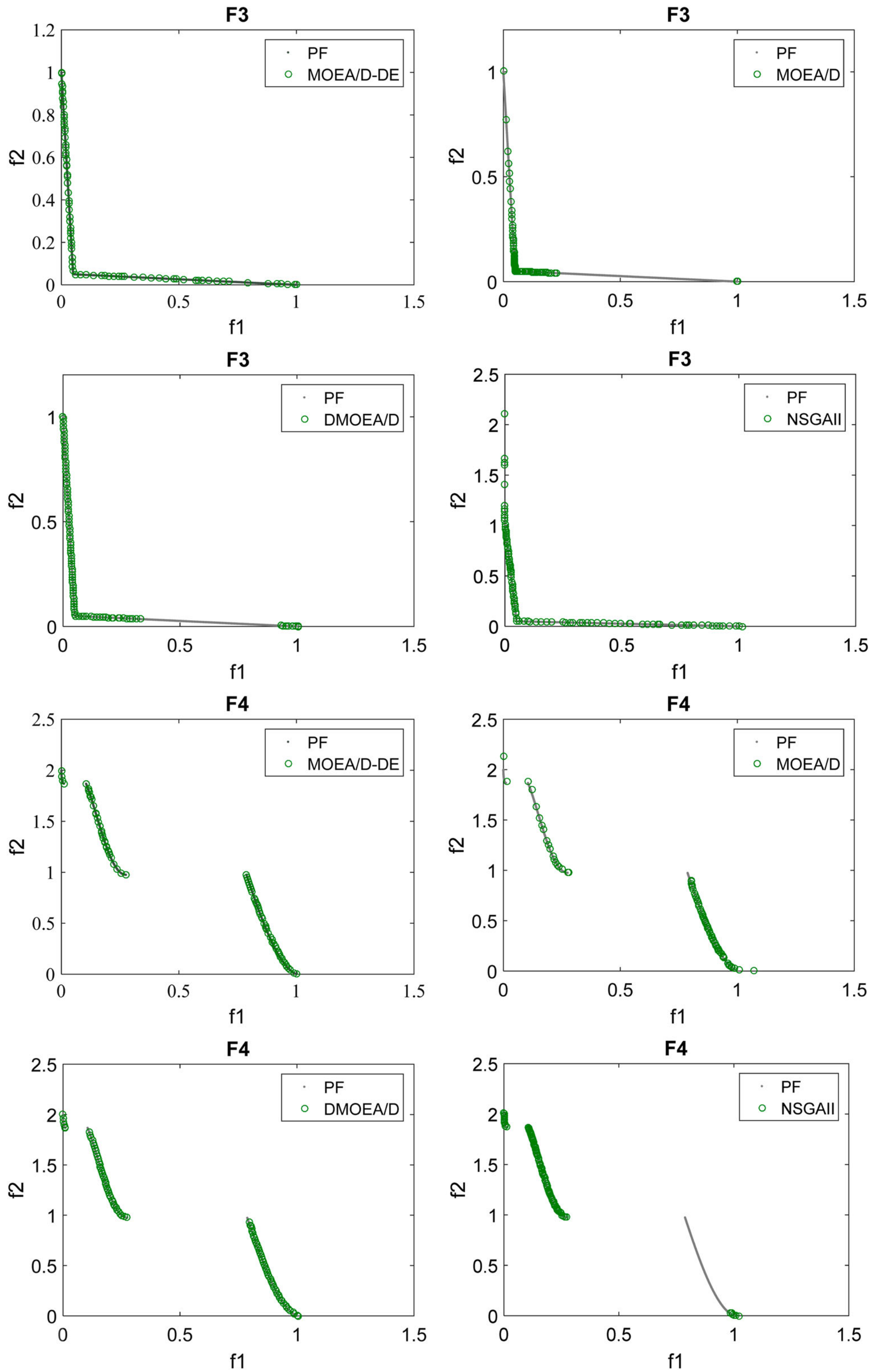

Figure 3. Continued.

Table 2. Comparison of benchmark problems with complex Pareto fronts in terms of the inverted generational distance metric.

Algorithm	Problem F1	F2	F3	F4
MOEA/D-DE	**1.3798e-04**	**2.5587e-04**	**3.5193e-04**	**2.0152e-04**
	(3.5339e-04)	(4.1209e-05)	(5.9543e-07)	(1.2564e-08)
MOEA/D	6.3369e-04+	6.200e-03+	4.600e-03+	4.6662e-04+
	(7.2098e-03)	(6.1539e-04)	(6.4295e-07)	(6.2187e-07)
DMOEA/D	1.3845e-04+	2.5599e-04 =	8.4087e-04+	2.0171e-04 =
	(4.1353e-04)	(6.2297e-05)	(3.23673e-06)	(6.7544e-08)
NSGA-II	1.5124e-04+	2.5691e-04 =	3.5728e-04+	5.800e-03+
NSGA-II	(2.3741e-03)	(4.3519e-05)	(8.0343e-06)	(8.1568e-07)

Note: Data are shown as mean (standard deviation).
MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

The Wilcoxon rank-sum test (Wilcoxon 1945) with a confidence level of 0.95 was conducted based on the IGD values to assess the statistical significance of the results. In Table 2, the symbols '+', '=' and '–', respectively, indicate that the performance of MOEA/D-DE is statistically better than, equivalent to and not as good as the compared algorithm.

As shown in Table 2, the proposed MOEA/D-DE has smaller average IGD values than the original MOEA/D, DMOEA/D and NSGA-II, which illustrates the superiority of MOEA/D-DE over MOEA/D, DMOEA/D and NSGA-II. In addition, MOEA/D-DE has smaller standard deviations of IGD values, which demonstrates that MOEA/D-DE performs more stably.

4.3.2. Experimental studies on benchmark problems with complex Pareto sets

The performances of the four compared algorithms on the suite of UF test instances in the CEC 2009 competition were also investigated to illustrate the superiority of the proposed MOEA/D-DE. Figure 4 illustrates the solution sets with the lowest IGD values obtained by the four compared algorithms. As shown in this figure, MOEA/D-DE performs significantly better than the other three algorithms on all four investigated UF problems in terms of coverage and uniformity. MOEA/D-DE outperforms DMOEA/D on problems UF1, UF2 and UF4 in terms of uniformity. For problem UF3, DMOEA/D fails to achieve as good convergence as the proposed MOEA/D-DE.

Table 3 compares the performance of MOEA/D-DE, MOEA/D, DMOEA/D and NSGA-II on the four UF problems in terms of the IGD metric. It can be seen that MOEA/D-DE has lower mean IGD values than DMOEA/D and NSGA-II, which indicates that the proposed MOEA/D-DE performs better than DMOEA/D and NSGA-II on multi-objective problems with complex Pareto sets. Compared with MOEA/D, the proposed MOEA/D-DE also has lower mean IGD values on three out of the four UF problems, the exception being the UF4 problem. It can be seen from Figure 4 that MOEA/D performs better in terms of convergence; however, MOEA/D-DE has better uniformity over the whole Pareto front.

4.3.3. Experimental results on RFCO problems

The ideal Pareto fronts of the four investigated floods are the non-dominated solutions found by running the original MOEA/D with 5,000,000 function evaluations over 30 independent runs. The total dispatching times of the floods on 12 October 2000, 28 August 2003, 1 October 2005 and 15 July 2010 are 97 h, 44 h, 73 h and 145 h, respectively. To control the number of decision variables, the dispatching time intervals of the above four floods are set as 6 h, 3 h, 4 h and 6 h, respectively.

Figure 5 illustrates the best approximation of the Pareto sets obtained by MOEA/D-DE, MOEA/D and NSGA-II for the four investigated floods. These data are the Pareto-optimal sets with the largest HV metric values obtained by the two comparison algorithms over 30 independent runs. It can be seen that the proposed MOEA/D-DE can obtain Pareto-optimal sets with better uniformity and

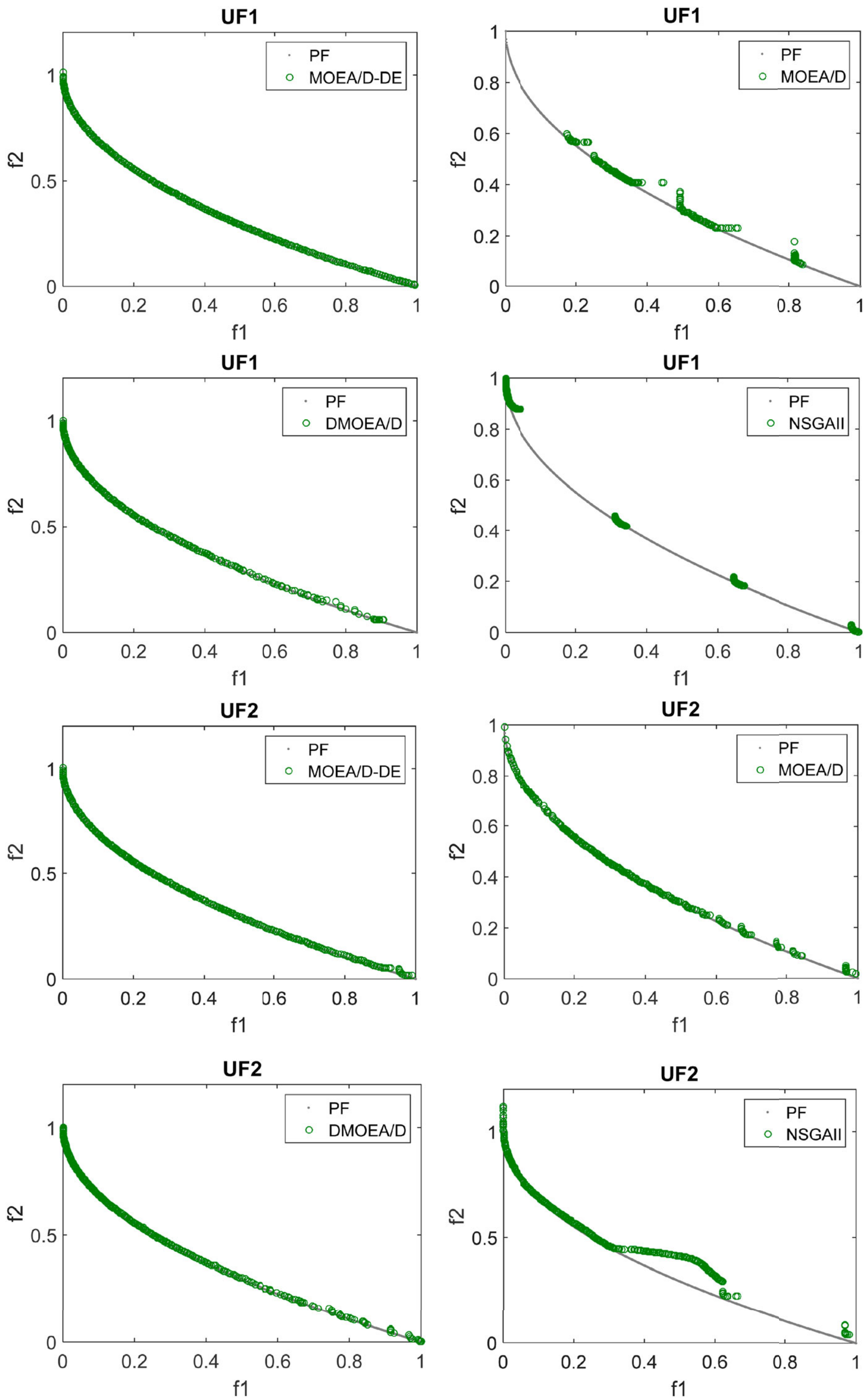

Figure 4. Best approximation of Pareto fronts (PFs) obtained by the compared algorithms on the UF suite of problems with complex Pareto sets. MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

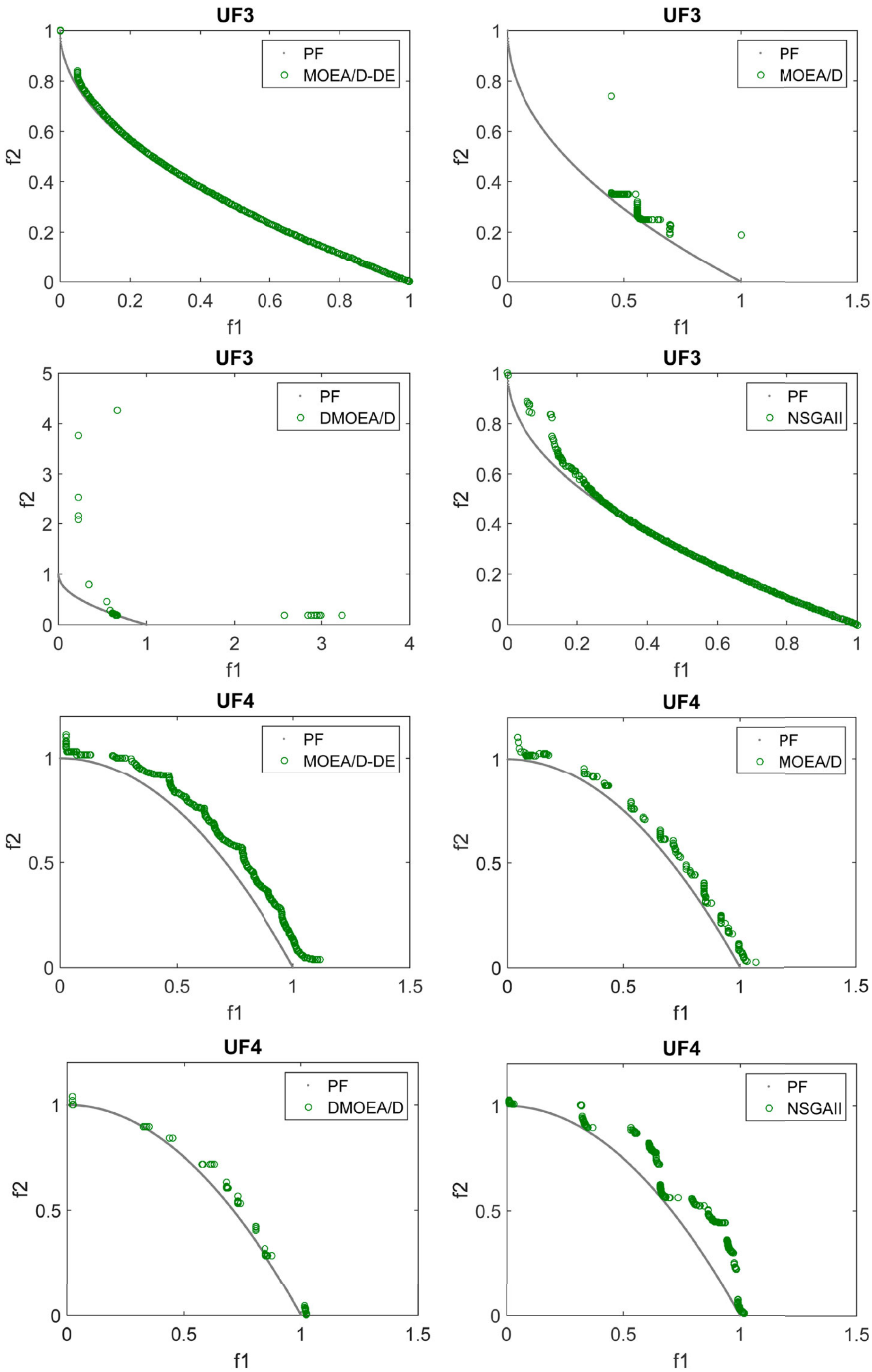

Figure 4. Continued.

Table 3. Comparison of the UF suite of problems with complex Pareto sets in terms of the inverted generational distance metric.

	Problem			
Algorithm	UF1	UF2	UF3	UF4
MOEA/D-DE	7.7177e-05 (1.2312e-04)	1.4980e-04 (2.1487e-05)	3.8395e-04 (9.2168e-06)	1.9762e-03 (4.1879e-06)
MOEA/D	3.3284e-03+ (5.1681e-04)	5.2721e-04+ (1.2485e-05)	6.5489e-03+ (1.4780e-06)	1.5146e-03– (4.1584e-06)
DMOEA/D	6.6047e-04+ (5.8619e-05)	2.3558e-04+ (5.2196e-05)	6.8419e-03+ (3.1468e-06)	2.2268e-03+ (7.2159e-07)
NSGA-II	3.6216e-03+ (1.6518e-04)	2.2864e-03+ (7.1543e-05)	6.4293e-04+ (7.1586e-06)	2.7652e-03+ (8.3549e-06)

Note: Data are shown as mean (standard deviation).
MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

coverage over the ideal Pareto front. This significant improvement results from the DE-inspired recombination operator.

As previously been validated by Qi, Hou, Yin, *et al.* (2015), the DE-inspired recombination operator provides two types of candidate search directions. One of them will guide the search to find new points along the current Pareto set and thus result in the solutions being evenly scattered along the current Pareto set. The other one will lead the algorithm to obtain new points apart from the current Pareto set and explore further search regions to make the Pareto set wider. This is the reason why the proposed MOEA/D-DE has superior performance to the original MOEA/D for the four investigated floods.

Figure 6 shows the discharge volumes and the upstream water levels of the best Pareto-optimal sets obtained by MOEA/D-DE. As shown in this figure, most of the obtained dispatching schemes can significantly reduce the inflow flood peaks and provide stable discharging flows. With regard to the curves of the upstream water level, MOEA/D-DE successfully guarantees the safety of the dam by ensuring that the upstream water levels never get so high that they threaten the dam. In addition, MOEA/D-DE can provide Pareto-optimal solutions with good diversity. Taking the flood on 28 August 2003 as an example, most of the obtained solutions have stable discharge volumes and upstream water levels; however, there are also solutions with relatively high discharge volumes and thus extremely low upstream water levels. These irregular solutions could be selected to be put into practice at the beginning of the flood season, especially when an extraordinarily big flood is anticipated.

Table 4 shows the means and standard deviations of the HV metric values of the solutions obtained by MOEA/D-DE and MOEA/D over 30 independent runs for the four investigated floods. The reference points for calculating the HV values are set as follows: (330, 10,000) for the flood on 12 October 2000, (328, 5000) for the flood on 28 August 2003, (328, 14,000) for the flood on 1 October 2005 and (329, 7000) for the flood on 15 July 2010. It can be seen that the proposed MOEA/D-DE has larger average HV values than the original MOEA/D, DMOEA/D and NSGA-II, which illustrates the superiority of MOEA/D-DE over MOEA/D, DMOEA/D and NSGA-II. In addition, MOEA/D-DE has smaller standard deviations of HV values, which demonstrates that MOEA/D-DE performs more stably.

4.3.4. Experimental results on the running time of all algorithms

The UF suite of problems is taken as an example to compare the running time of the proposed MOEA/D-DE with the other baseline algorithms. The results are reported in Table 5.

It can be seen that NSGA-II, with time complexity of $O(N2)$, is the most time-consuming algorithm, and MOEA/D, with time complexity of $O(TN)$, requires the least running time (N is the population size and T is the neighbourhood size of the decomposition subproblems). The two

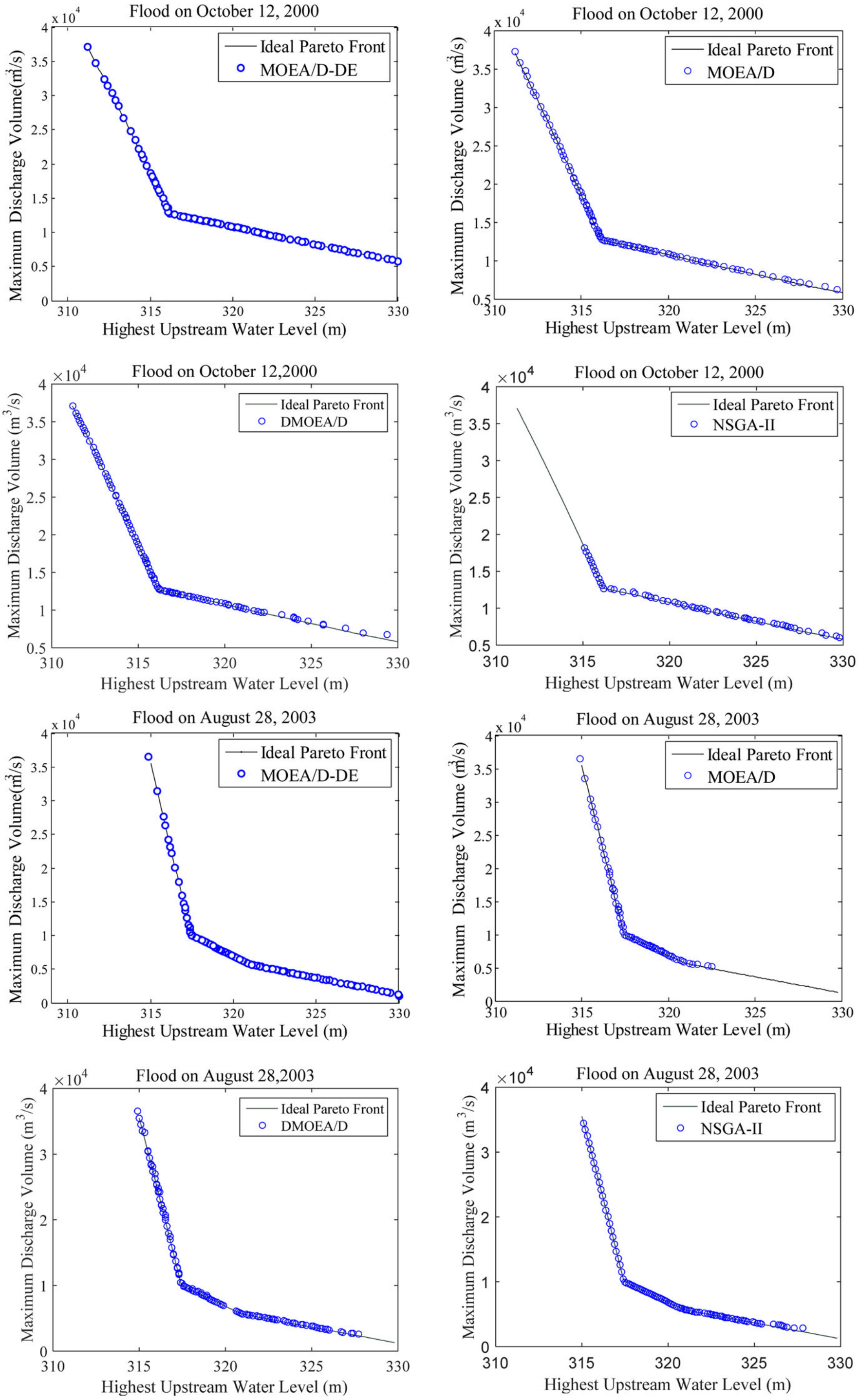

Figure 5. Comparison of the best approximation of Pareto sets obtained by the compared algorithms on the four investigated floods. MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

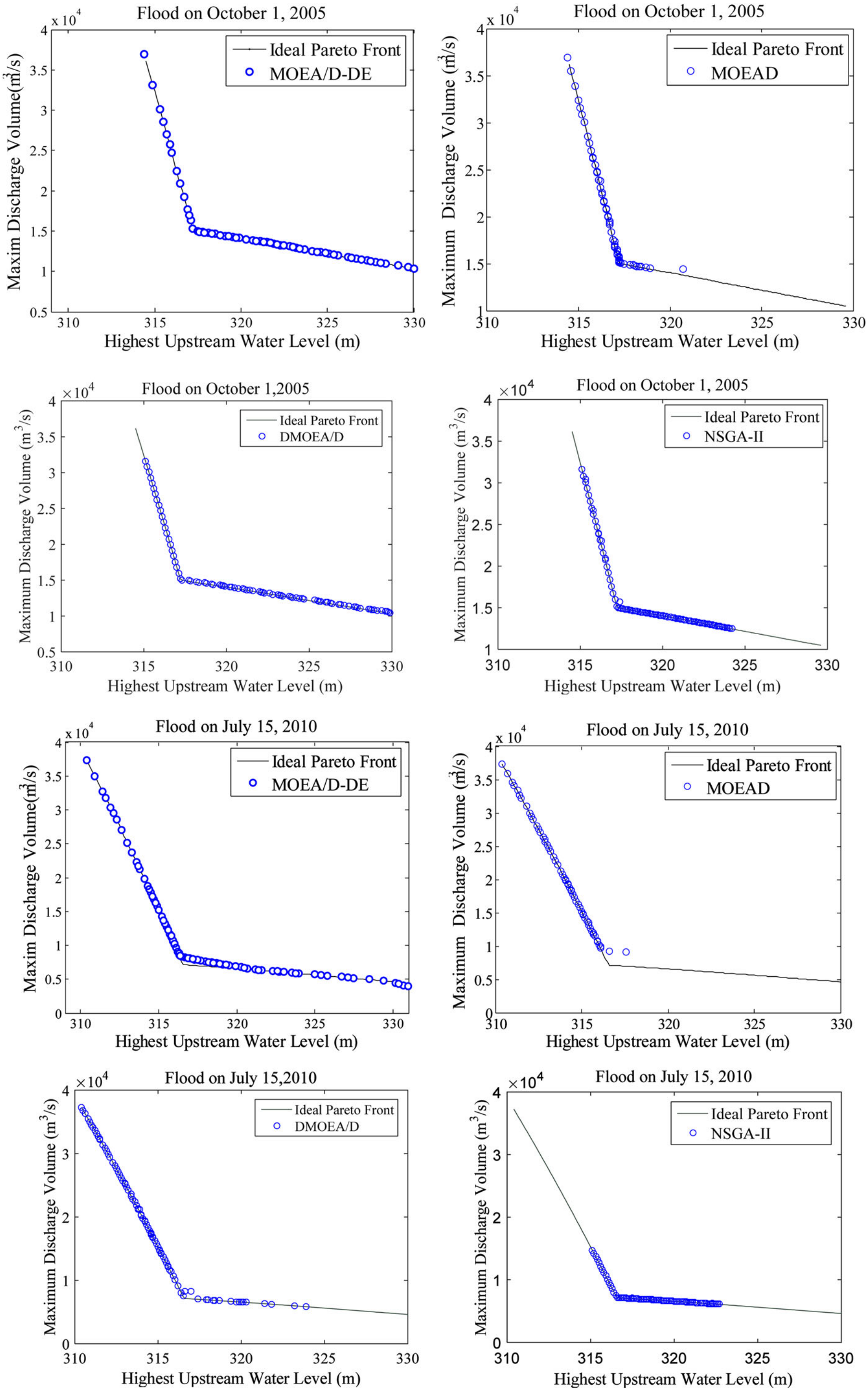

Figure 5. Continued.

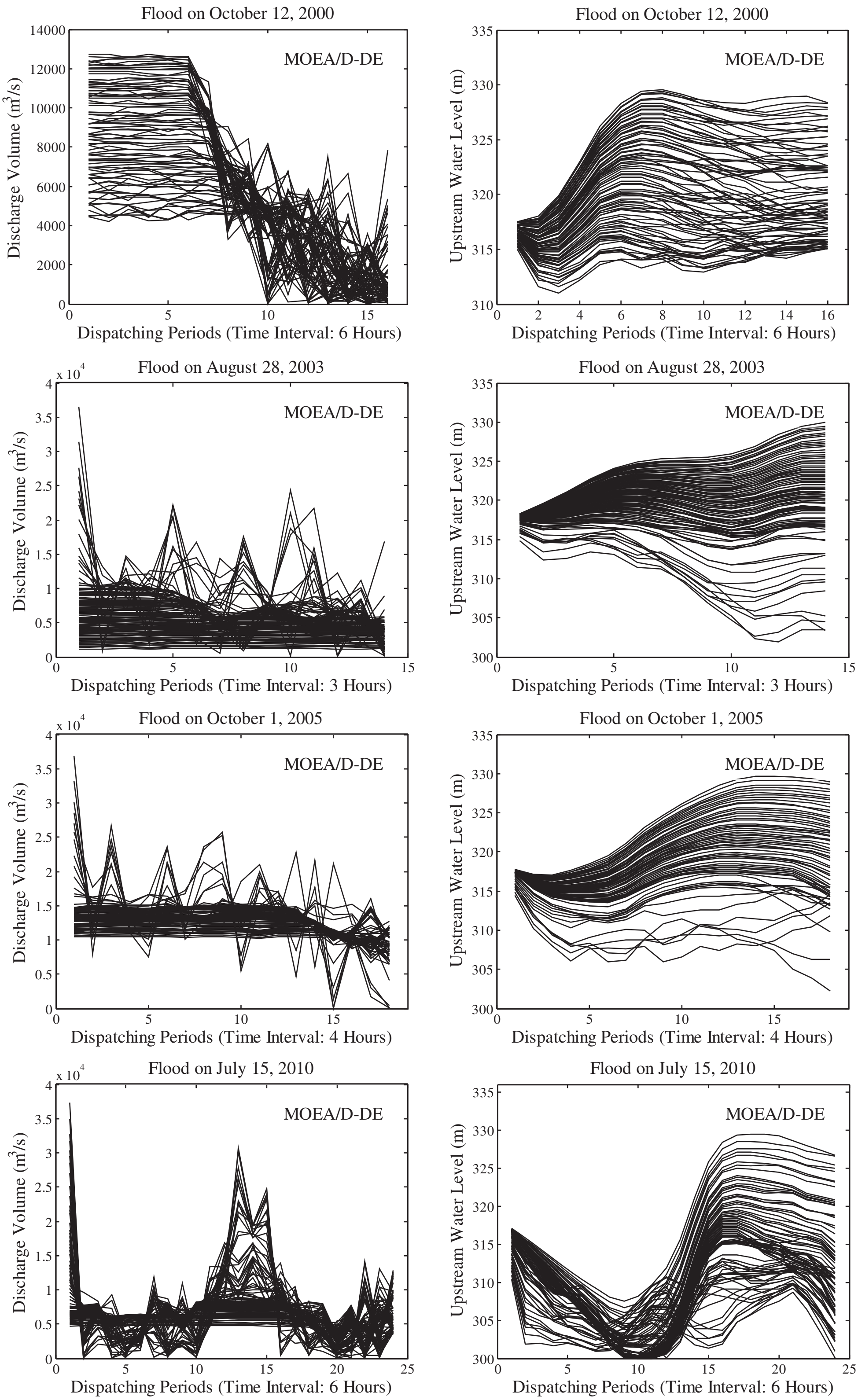

Figure 6. Details of the discharge volumes and upstream water levels of the best Pareto-optimal sets obtained by the multi-objective evolutionary algorithm based on decomposition–differential evolution (MOEA/D-DE).

Table 4. Statistical hyper-volume values of the solutions found by the compared algorithms for the four investigated floods.

Algorithm	Investigated flood			
	12 October 2000	28 August 2003	1 October 2005	15 July 2010
MOEA/D-DE	**1.112e + 09**	**1.236e + 09**	**9.522e + 09**	**1.211e + 09**
	(1.754e + 07)	(4.133e + 06)	(6.635e + 05)	(8.726e + 07)
MOEA/D	1.001e + 09 =	1.103e + 09+	8.065e + 08+	1.051e + 09+
	(7.131e + 07)	(4.249e + 07)	(2.621e + 07)	(3.313e + 08)
DMOEA/D	9.876e + 08+	1.232e + 09 =	9.176e + 09 =	1.117e + 09+
	(3.542e + 07)	(8.249e + 06)	(7.364e + 08)	(5.168e + 08)
NSGA-II	8.621e + 08+	1.229e + 09 =	9.194e + 08+	1.108e + 09+
	(2.479e + 07)	(5.534e + 08)	(7.249e + 07)	(1.572e + 08)

Note: Data are shown as mean (standard deviation). The best results are shown in bold.
MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

Table 5. Running time (in seconds) of the compared algorithms.

Problem	Algorithm			
	MOEA/D-DE	DMOEA/D	MOEA/D	NSGA-II
UF1	10.653	8.259	4.979	27.237
UF2	11.784	10.148	6.05	26.002
UF3	11.215	9.843	5.683	28.931
UF4	10.695	9.416	5.844	24.958

Note: MOEA/D-DE = multi-objective evolutionary algorithm based on decomposition–differential evolution; MOEA/D = multi-objective evolutionary algorithm based on decomposition; DMOEA/D = dynamic multi-objective evolutionary algorithm based on decomposition; NSGA-II = non-dominated sorting genetic algorithm II.

enhanced MOEA/D algorithms, MOEA/D-DE and DMOEA/D, spend twice as much running time as the original MOEA/D algorithm on the four investigated problems; however, both of them run faster than NSGA-II. MOEA/D-DE runs slightly slower than DMOEA/D, owing to the execution of the DE-inspired recombination operator in this algorithm.

5. Conclusions

In this work, a DE-inspired recombination operator, which is specially designed for continuous MOPs, is introduced into the algorithmic framework of the MOEA/D to form the proposed MOEA/D-DE for solving multi-objective RFCO problems. By considering the regularity of continuous MOPs, the DE-inspired recombination operator provides two types of candidate searching directions which are designed to complement each other. One of them guides the search to find new points along the current Pareto set, resulting in the solutions being evenly scattered along current Pareto set. The other leads the algorithm to obtain new points apart from the current Pareto set and explore further search regions to make the Pareto set wider.

Experimental results on benchmark problems and four typical floods at the Ankang reservoir have illustrated that, with the help of the DE-inspired recombination operator, the proposed MOEA/D-DE has superior performance to the original MOEA/D, obtaining Pareto-optimal sets with better uniformity and coverage. Moreover, the dispatching schemes obtained by MOEA/D-DE can significantly reduce the inflow flood peak and guarantee safety both upstream and downstream of the dam.

The proposed algorithm can be combined with existing selection techniques in the objective space to further improve the algorithm's performance, which will be the subject of future work.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the National Natural Science Foundation of China [grant numbers 51679186 and 51679188], the National Key Research and Development Program of China [grant number 2016YFC0401409] and the Science and Technology Program of Shaanxi Province [grant number 2015KJXX-30].

References

Afshar, A., F. Sharifi, and M. R. Jalali. 2009. "Non-dominated Archiving Multi-colony Ant Algorithm for Multi-objective Optimization: Application to Multi-purpose Reservoir Operation." *Engineering Optimization* 41 (4): 313–325. doi:10.1080/03052150802460414.

Baltar, A. M., and D. G. Fontane. 2008. "Use of Multiobjective Particle Swarm Optimization in Water Resources Management." *Journal of Water Resources Planning and Management* 134 (3): 257–265.

Chen, L., J. McPhee, and W. W. G. Yeh. 2007. "A Diversified Multiobjective GA for Optimizing Reservoir Rule Curves." *Advances in Water Resources* 30 (5): 1082–1093. doi:10.1016/j.advwatres.2006.10.001.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II." *IEEE Transactions on Evolutionary Computation* 6 (2): 182–197.

Dumedah, G., A. Berg, M. Wineberg, and R. Collier. 2010. "Selecting Model Parameter Sets From a Trade-off Surface Generated From the Non-Dominated Sorting Genetic Algorithm-II." *Water Resources Management* 24 (15): 4469–4489. doi:10.1007/s11269-010-9668-y.

Fu, G. T. 2008. "A Fuzzy Optimization Method for Multicriteria Decision Making: An Application to Reservoir Flood Control Operation." *Expert Systems with Applications* 34 (1): 145–149. doi:10.1016/j.eswa.2006.08.021.

Gu, F., H. L. Liu, and K. Tan. 2012. "A Multiobjective Evolutionary Algorithm Using Dynamic Weight Design Method." *International Journal of Innovative Computing, Information and Control* 8 (5): 3677–3688.

Hajkowicz, S., and K. Collins. 2007. "A Review of Multiple Criteria Analysis for Water Resource Planning and Management." *Water Resources Management* 21 (9): 1553–1566. doi:10.1007/s11269-006-9112-5.

Hakimi-Asiabar, M., S. H. Ghodsypour, and R. Kerachian. 2010. "Deriving Operating Policies for Multi-objective Reservoir Systems: Application of Self-Learning Genetic Algorithm." *Applied Soft Computing* 10 (4): 1151–1163. doi:10.1016/j.asoc.2009.08.016.

Hou, Z. C., and S. Y. Chen. 2004. "Multi-objective Fuzzy Group Decision-Making Method for Reservoir Flood Control Operation." *Journal of Hydraulic Engineering* 35 (12): 106–111,119.

Ishibuchi, H., Y. Sakane, N. Tsukamoto, and Y. Nojima. 2010. "Simultaneous use of Different Scalarizing Functions in MOEA/D." Article presented at the proceedings of the 12th annual conference on genetic and evolutionary computation, Portland, Oregon, USA.

Kim, T., J. H. Heo, and C. S. Jeong. 2006. "Multireservoir System Optimization in the Han River Basin Using Multi-objective Genetic Algorithms." *Hydrological Processes* 20 (9): 2057–2075. doi:10.1002/hyp.6047.

Li, H., and D. Landa-Silva. 2011. "An Adaptive Evolutionary Multi-objective Approach Based on Simulated Annealing." *Evolutionary Computation* 19 (4): 561–595.

Liu, Y. 2009. "Automatic Calibration of a Rainfall–Runoff Model Using a Fast and Elitist Multi-objective Particle Swarm Algorithm." *Expert Systems with Applications* 36 (5): 9533–9538. doi:10.1016/j.eswa.2008.10.086.

Luo, J., C. Chen, and J. Xie. 2015. "Multi-objective Immune Algorithm with Preference-Based Selection for Reservoir Flood Control Operation." *Water Resources Management* 29 (5): 1447–1466. doi:10.1007/s11269-014-0886-6.

Malekmohammadi, B., B. Zahraie, and R. Kerachian. 2011. "Ranking Solutions of Multi-objective Reservoir Operation Optimization Models Using Multi-criteria Decision Analysis." *Expert Systems with Applications* 38 (6): 7851–7863. doi:10.1016/j.eswa.2010.12.119.

Needham, J., Jr., D. Watkins, J. Lund, and S. Nanda. 2000. "Linear Programming for Flood Control in the Iowa and Des Moines Rivers." *Journal of Water Resources Planning and Management* 126 (3): 118–127. doi:10.1061/(ASCE)0733-9496(2000)126:3(118)

Ouyang, S., J. Z. Zhou, H. Qin, X. Liao, and H. Wang. 2014. "A Novel Multi-objective Electromagnetism-Like Mechanism Algorithm with Applications in Reservoir Flood Control Operation." *Water Science & Technology* 69 (6): 1181–1190. doi:10.2166/wst.2013.812.

Pal, S., B. Y. Qu, S. Das, and P. N. Suganthan. 2010. "Linear Antenna Array Synthesis with Constrained Multi-objective Differential Evolution." Article presented at the Progress in Electromagnetics Research B.

Qi, Y. T., L. Bao, Y. Y. Sun, J. G. Luo, and Q. G. Miao. 2016. "A Memetic Multi-objective Immune Algorithm for Reservoir Flood Control Operation." *Water Resources Management* 30 (9): 2957–2977.

Qi, Y. T., Z. Hou, H. Li, J. B. Huang, and X. D. Li. 2015a. "A Decomposition Based Memetic Algorithm for Multi-objective Vehicle Routing Problem with Time Windows." *Computers & Operations Research* 62: 61–77. doi:10.1016/j.cor.2015.04.009.

Qi, Y., Z. Hou, M. Yin, H. Sun, and J. Huang. 2015b. "An Immune Multi-objective Optimization Algorithm with Differential Evolution Inspired Recombination." *Applied Soft Computing* 29 (0): 395–410. doi:10.1016/j.asoc.2015.01.012.

Qi, Y., F. Liu, M. Liu, M. Gong, and L. Jiao. 2012. “Multi-objective Immune Algorithm with Baldwinian Learning.” *Applied Soft Computing* 12 (8): 2654–2674. doi:10.1016/j.asoc.2012.04.005.
Qi, Y., X. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu. 2014. “MOEA/D with Adaptive Weight Adjustment.” *Evolutionary Computation* 22 (2): 231–264. doi:10.1162/EVCO_a_00109.
Qin, H., J. Z. Zhou, Y. L. Lu, Y. H. Li, and Y. C. Zhang. 2010. “Multi-objective Cultured Differential Evolution for Generating Optimal Trade-offs in Reservoir Flood Control Operation.” *Water Resources Management* 24 (11): 2611–2632. doi:10.1007/s11269-009-9570-7.
Qin, H., J. Z. Zhou, G. Q. Wang, and Y. C. Zhang. 2009. “Multi-objective Optimization of Reservoir Flood Dispatch Based on Multi-objective Differential Evolution Algorithm.” *Journal of Hydraulic Engineering* 40 (5): 513–519. doi:0559-9350(2009)40:5 < 513:jydmbc > 2.0.tx;2-h.
Reddy, M. J., and D. N. Kumar. 2007. “Multiobjective Differential Evolution with Application to Reservoir System Optimization.” *Journal of Computing in Civil Engineering* 21 (2): 136–146.
Shokri, A., H. O. Bozorg, and M. Mariño. 2014. “Multi-objective Quantity–Quality Reservoir Operation in Sudden Pollution.” *Water Resources Management* 28 (2): 567–586. doi:10.1007/s11269-013-0504-z.
Unver, O., and L. Mays. 1990. “Model for Real-Time Optimal Flood Control Operation of a Reservoir System.” *Water Resources Management* 4 (1): 21–46. doi:10.1007/bf00429923.
Waldock, A., and D. Corne. 2011. “Multiple Objective Optimisation Applied to Route Planning.” Article presented at the proceedings of the 13th annual conference on genetic and evolutionary computation, Dublin, Ireland.
Wilcoxon, F. 1945. “Individual Comparisons by Ranking Methods.” *Biometrics Bulletin* 1 (6): 80–83.
Xu, K., J. Zhou, Y. Zhang, and R. Gu. 2012. “Differential Evolution Based on ϵ-Domination and Orthogonal Design Method for Power Environmentally-Friendly Dispatch.” *Expert Systems with Applications* 39 (4): 3956–3963. doi:10.1016/j.eswa.2011.08.145.
Yakowitz, S. 1982. “Dynamic Programming Applications in Water Resources.” *Water Resources Research* 18 (4): 673–696. doi:10.1029/WR018i004p00673.
Yu, Y. B., B. D. Wang, G. L. Wang, and W. Li. 2004. “Multi-person Multiobjective Fuzzy Decision-Making Model for Reservoir Flood Control Operation.” *Water Resources Management* 18 (2): 111–124. doi:10.1023/B:WARM.0000024705.63932.3c.
Zhang, Q., and H. Li. 2007. “MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition.” *IEEE Transactions on Evolutionary Computation* 11 (6): 712–731. doi:10.1109/TEVC.2007.892759.
Zhou, A., B. Y. Qu, H. Li, S. Z. Zhao, P. N. Suganthan, and Q. Zhang. 2011. “Multiobjective Evolutionary Algorithms: A Survey of the State of the Art.” *Swarm and Evolutionary Computation* 1 (1): 32–49. doi:10.1016/j.swevo.2011.03.001.
Zhou, H., G. Zhang, and G. Wang. 2007. “Multi-objective Decision Making Approach Based on Entropy Weights for Reservoir Flood Control Operation.” *Journal of Hydraulic Engineering* 38 (1): 100–106. doi:0559-9350(2007)38:1 < 100:jysqds > 2.0.tx;2-t.
Zitzler, E., and L. Thiele. 1998. “Multiobjective Optimization Using Evolutionary Algorithms – A Comparative Case Study.” In *Parallel Problem Solving From Nature – PPSN V, vol 1498. Lecture Notes in Computer Science*, edited by A. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, 292–301. Berlin: Springer. doi:10.1007/BFb0056872.