water

MDPI

Article

Long-Term Scheduling of Large-Scale Cascade Hydropower Stations Using Improved Differential Evolution Algorithm

Xiaohao Wen [1,2], Jianzhong Zhou [1,3,*], Zhongzheng He [1,3] and Chao Wang [4]

[1] School of Hydropower and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China; wen_xiaohao@ctg.com.cn (X.W.); hezz_hae@hust.edu.cn (Z.H.)
[2] China Three Gorges Corporation, Beijing 100038, China
[3] Hubei Key Laboratory of Digital Valley Science and Technology, Wuhan 430074, China
[4] China Institute of Water Resources and Hydropower Research, Beijing 100038, China; wangchao@iwhr.com
* Correspondence: jz.zhou@mail.hust.edu.cn; Tel.: +86-027-8754-3127

Received: 1 March 2018; Accepted: 20 March 2018; Published: 26 March 2018

Abstract: Long-term scheduling of large cascade hydropower stations (LSLCHS) is a complex problem of high dimension, nonlinearity, coupling and complex constraint. In view of the above problem, we present an improved differential evolution (iLSHADE) algorithm based on LSHADE, a state-of-the-art evolutionary algorithm. iLSHADE uses new mutation strategies "current to pbest/2-rand" to obtain wider search range and accelerate convergence with the preventing individual repeated failure evolution (PIRFE) strategy. The handling of complicated constraints strategy of ε-constrained method is presented to handle outflow, water level and output constraints in the cascade reservoir operation. Numerical experiments of 10 benchmark functions have been done, showing that iLSHADE has stable convergence and high efficiency. Furthermore, we demonstrate the performance of the iLSHADE algorithm by comparing it with other improved differential evolution algorithms for LSLCHS in four large hydropower stations of the Jinsha River. With the applications of iLSHADE in reservoir operation, LSLCHS can obtain more power generation benefit than other alternatives in dry, normal, and wet years. The results of numerical experiments and case studies show that the iLSHADE has a distinct optimization effect and good stability, and it is a valid and reliable tool to solve LSLCHS problem.

Keywords: energy; hydropower stations; differential evolution algorithm; optimal scheduling; ε-constrained method

1. Introduction

Hydropower has a significant share on the total energy consumption as it is renewable, clean, and cheap. Therefore, many countries have been working on the development of the utility of hydropower [1], and many hydropower plants have been put into operation in the past few decades [2–6]. Large cascade hydropower stations (LHS) play an increasingly important role in energy production. Many scholars have conducted a lot of research on the water resources management of LHS. Zhou et al. [7] proposed a joint optimal refill rules for cascade reservoirs to solve the conflict between the flood control and refill operation. The energy storage operation chart combined with discriminant coefficient method was put forward by Jiang [8], which was successfully applied to cascade reservoirs of Li Xianjiang River in southwest China. Regarding the input (e.g., inflow) imprecision and uncertainties, Chen et al. [9–12] analyzed the influence of the uncertainty in water resources management and the distribution of flood forecasting error. Djebou et al. [13,14] presented the interactions between these hydrologic factors that interplay at the watershed scale using the entropy-based index. Aiming to determine the optimal strategy that hedges the risk of energy shortfall, Xu et al. [15] develops a multi-objective stochastic programming model for informing hedging decisions for hydropower operations. Due to complex hydrodynamic relation, various complex constraints and diversified interest demand, the long-term scheduling of large cascade hydropower stations (LSLCHS) has developed into a multi-dimensional, non-convex and non-linear optimization problem. Correspondingly, optimization of LSLCHS has become a fairly challenging theoretical and practical problem, which is urgent for optimization techniques and constraints treatment [16–18]. Over many years, to solve this problem, various optimization algorithms have been applied. Usually there are two categories of methods: traditional optimizers and modern heuristic algorithms. Traditional optimizers include linear programming (LP) [19,20], nonlinear programming (NLP) [21,22], dynamic programs (DP) [23,24], progressive optimality algorithm (POA) [25,26], etc. These algorithms have rigorous mathematical foundations but low convergence efficiency. They suffer from curse of dimensionality. Modern heuristic algorithms use intelligent strategies to guide search to better areas, such as particle swarm optimization (PSO) [27,28], genetic algorithm (GA) [29], cultural algorithm (CA) [30], binary artificial sheep algorithm (BASA) [31], ant colony optimization (ACO) [18,32], etc. Compared with traditional optimizers, modern meta-heuristics are significantly more flexible and have high search efficiency as the meta-heuristics are inspired by different nature principles from biology, ethology, or physics. However, the common heuristic algorithms have some disadvantages such as premature convergence because of local fast convergence, and bad local search capability owing to many global searches. Moreover, they lack effective measures to handle complex constraints, making it difficult to be applied to solve complex optimal problems with high dimensions such as LSLCHS.

Differential evolution (DE) is a simple yet practical modern heuristic algorithm for global optimization over continuous spaces introduced by Price and Storn [33]. The DE algorithm has been used in many practical cases [34,35] and gradually become more popular. Similar to all other modern heuristic algorithms, the evolutionary process of DE uses mutations, crossover, and selection operators at each generation to reach the global optimum. The performance of DE basically depends on the mutation strategy, the crossover operator. Besides, the intrinsic control parameters (population size NP, scaling factor F, the crossover rate Cr) play a vital role in balancing the diversity of population and convergence speed of the algorithm. Therefore, Brest et al. [36] proposed a self-adaptive DE (jDE), in which both F and Cr are applied at random with probability $\tau 1$ and $\tau 2$. SaDE is proposed by Qin et al. [37] adaptively adjusts its trial vector generation strategies and control parameters simultaneously by learning from the previous search. JADE [38] is a well-known, effective DE variant which employs a control parameter adaptation mechanism and puts forward mutation strategy "current-to-pbest/1", differential evolution with composite trial vector generation strategies, control parameters (CoDE) [39], differential evolution with ensemble of parameters and mutation strategies (EPSDE) [40]. Success-History-based Adaptive DE (SHADE) [41] is an improved version of JADE which uses a different parameter adaptation mechanism. LSHADE [42] further extends SHADE with Linear Population Size Reduction (LPSR), which continually decreases the population size according to a linear function. In addition, LSHADE is the best ranked DE algorithm on CEC2014 Competition on Real-Parameter Single Objective Optimization [43] (see http://www.ntu.edu.sg/home/EPNSugan/index_files/CEC2014/-CEC2014.htm).

To avoid premature convergence and to accelerate convergence, we present an improved version of the LSHADE algorithm in this paper, called iLSHADE. The main improvement of iLSHADE is that a new mutation strategy "current to pbest/2-rand" is put forward for wider search range to improve search ability and prevent individual repeated failure evolution (PIRFE) strategy applied in the population evolution process. Finally, iLSHADE is applied to LSLCHS in Jinsha River combined with the improved constraints handling technique. Results of the study demonstrates its superiority in dealing with LSLCHS problem.

The remainder of this paper is organized as follows: Section 2 introduces the formulation of LSLCHS problem. In Section 3, a brief view of DE framework and improvement strategies of iLSHADE is presented. Section 4 presents numerical simulation experiment of iLSHADE. In Section 5, implementation of iLSHADE in LSLCHS is shown in detail. In Section 6, iLSHADE is applied to solve LSLCHS, and the results are analyzed. Finally, conclusions are summarized in Section 7.

2. Optimization Model

The primary objective of LSLCHS problem is to maximize the total power generation of LHS over the whole operation periods, while subjecting to kinds of equality and inequality constraints. The objective formula and constraints are described as follows.

2.1. Objective Function

$$\text{obj} = max\sum_{i=1}^{M}\sum_{t=1}^{T} N_{it}\Delta t,\ N_{it} = A_i H_{it} Q_{it} \tag{1}$$

where obj is the total power generation of LHS over the whole operation periods, M is the number of hydro plants; T is the whole periods; A_i is output coefficient of the i-th hydro plant; Δt shows interval of scheduling term; N_{it}, H_{it} and Q_{it} denote output, pure water head and water discharge through hydro-turbine of the i-th hydro plant in the t-th period, respectively. Moreover, H_{it} is calculated by upstream water level, trail water level and head loss shown in formula (8).

2.2. Constraints

In the process of long-term optimal dispatch, various complex equality and inequality constraints, such as water level, output, and hydraulic connection, should be considered for restricting the total power generation optimization. The constraints of LHS are described as follows:

1. Water balance constraint.

$$\begin{array}{l} V_{i,t+1} = V_{i,t} + (I_{i,t} - Q_{i,t} - S_{i,t})\Delta t, \\ I_{i,t} = q_{i,t} + Q_{i-1,t} + S_{i-1,t} \end{array} \tag{2}$$

$V_{i,t}$ is reservoir storage of the i-th hydropower station at the beginning of period t, $I_{i,t}$ is inflow, $q_{i,t}$ stands for local inflow and $S_{i,t}$ is deserted outflow.

2. Hydraulic connection.

$$Z_{i,t}^{down} = \begin{cases} F(Q_{i,t} + S_{i,t}) & \text{without backwater effect,} \\ F(Q_{i,t} + S_{i,t},\ Z_{i+1,t}) & \text{with backwater effect.} \end{cases} \tag{3}$$

where $Z_{i,t}$ stands for upstream water level, $Z_{i,t}^{down}$ is trail water level. Function F represents the hydraulic connection between upstream and downstream hydropower stations. Generally, the trail water level is a function of outflow. However, when the hydropower station is located at the backwater region of its downstream hydropower station, the upstream water level of the downstream hydropower station must be taken into consideration in the function.

3. Water level constraint.

$$Z_{i,t}^{min} \le Z_{i,t} \le Z_{i,t}^{max} \tag{4}$$

$$|Z_{i,t} - Z_{i,t+1}| \le \Delta Z_i \tag{5}$$

$Z_{i,t}^{min}$ and $Z_{i,t}^{max}$ are the upper and lower water level limits and ΔZ_i is the maximum amplitude of water level variation.

4. Power generating constraint.

$$N_{i,t}^{min} \le N_{i,t} \le N_{i,t}^{max}(H_{i,t}) \tag{6}$$

$N_{i,t}^{max}(H_{i,t})$ represents the maximum output. The maximum output is a function of pure water head. $N_{i,t}^{min}$ is the lower output limit, which is generally called Guaranteed output.

5. Outflow constraint.

$$Q_{i,t}^{min} \leq Q_{i,t} + S_{i,t} \leq Q_{i,t}^{max} \tag{7}$$

$Q_{i,t}^{max}$ is the maximum outflow limit and $Q_{i,t}^{min}$ is the minimum outflow limit.

6. Water head equation.

$$H_{i,t} = (Z_{i,t} + Z_{i,t+1})/2 - Z_{i,t}^{\mathrm{down}} - H_{i,t}^{loss}(Q_{i,t}) \tag{8}$$

$H_{i,t}$ stands for the pure water head. $Z_{i,t}^{\mathrm{down}}$ is trail water level described in formula (3). $H_{i,t}^{loss}(Q_{i,t})$ represents water head loss, which is a function of outflow through hydro-turbines.

7. Boundary condition.

$$Z_{i,0} = Z_{i}^{begin},\ Z_{i,T} = Z_{i}^{end} \tag{9}$$

where Z_{i}^{begin} and Z_{i}^{end} are initial water level and terminal water level of the i-th hydro plant, respectively.

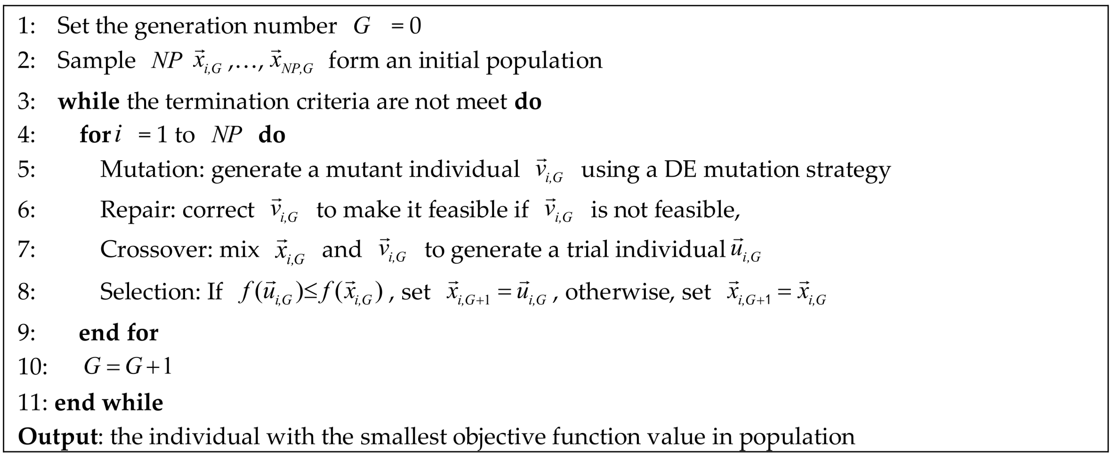
3. Overview of iLSHADE

3.1. DE

DE is a group-based evolutionary algorithm. It is used to solve the following continuous domain global optimization problem:

$$\begin{array}{l} minimize\ f(\vec{x}),\ \vec{x} = (x_1, \ldots, x_D) \\ x_{\mathrm{i}} \in [\overline{x}_i, \underline{x}_i]\ \ \forall i \in \{1, \ldots, D\},\ \ -\infty < \overline{x}_i < \underline{x}_i < +\infty \end{array} \tag{10}$$

where $f(\vec{x})$ is continuous fitness evaluation function, D is the dimension of the problem. DE has three control parameters that need to be set before the calculation: F is scaling factor, CR is crossover control parameter, and NP is population size. The framework of DE is as follows (Figure 1):



```
Set the generation number $G = 0$
Sample $NP$ $\vec{x}_{i,G}, \ldots, \vec{x}_{NP,G}$ form an initial population
while the termination criteria are not meet do
  for $i = 1$ to $NP$ do
    Mutation: generate a mutant individual $\vec{v}_{i,G}$ using a DE mutation strategy
    Repair: correct $\vec{v}_{i,G}$ to make it feasible if $\vec{v}_{i,G}$ is not feasible,
    Crossover: mix $\vec{x}_{i,G}$ and $\vec{v}_{i,G}$ to generate a trial individual $\vec{u}_{i,G}$
    Selection: If $f(\vec{u}_{i,G}) \leq f(\vec{x}_{i,G})$, set $\vec{x}_{i,G+1} = \vec{u}_{i,G}$, otherwise, set $\vec{x}_{i,G+1} = \vec{x}_{i,G}$
  end for
  $G = G + 1$
end while
Output: the individual with the smallest objective function value in population
```

Figure 1. Pseudocode of DE.

The mutation strategy in original DE is "rand/1", which is expressed in formula (11):

$$\vec{v}_{i,\mathrm{G}} = \vec{x}_{r1,\mathrm{G}} + F \cdot (\vec{x}_{r2,\mathrm{G}} - \vec{x}_{r3,\mathrm{G}}) \tag{11}$$

Other common DE mutation strategies are as follows:

- "rand/2":

$$\vec{v}_{i,G} = \vec{x}_{r1,\mathrm{G}} + F \cdot (\vec{x}_{r2,\mathrm{G}} - \vec{x}_{r3,\mathrm{G}}) + F \cdot (\vec{x}_{r4,\mathrm{G}} - \vec{x}_{r5,\mathrm{G}}) \quad (12)$$

- "best/1":

$$\vec{v}_{i,G} = \vec{x}_{best,\mathrm{G}} + F \cdot (\vec{x}_{r1,\mathrm{G}} - \vec{x}_{r2,\mathrm{G}}) \quad (13)$$

- "best/2":

$$\vec{v}_{i,G} = \vec{x}_{i,\mathrm{G}} + F \cdot (\vec{x}_{best,\mathrm{G}} - \vec{x}_{i,\mathrm{G}}) + F \cdot (\vec{x}_{r1,\mathrm{G}} - \vec{x}_{r2,\mathrm{G}}) \quad (14)$$

- "current to best/1":

$$\vec{v}_{i,G} = \vec{x}_{i,\mathrm{G}} + F \cdot (\vec{x}_{best,\mathrm{G}} - \vec{x}_{i,\mathrm{G}}) + F \cdot (\vec{x}_{r1,\mathrm{G}} - \vec{x}_{r2,\mathrm{G}}) \quad (15)$$

where the indexes $r1 - r5$ represent the random and mutually different integers generated within the range $\{1, \mathrm{NP}\}$, and also different from index i. $\vec{x}_{best,\mathrm{G}}$ is the best individual in a current generation. Each strategy has a different ability to maintain the diversity of the population, which may increase/reduce the rate of convergence in the process of evolution.

3.2. *iLSHADE*

An improved LSHADE (iLSHADE) with new mutation strategy "current to best/2-rand" and the PIRFE strategy is proposed. The details of these strategies and algorithm procedure are shown below.

3.2.1. Mutation Strategy "Current to pbest/2-rand"

The mutation strategy "current to pbest/1" was proposed by in the framework of JADE (which is expressed in formula (16)).

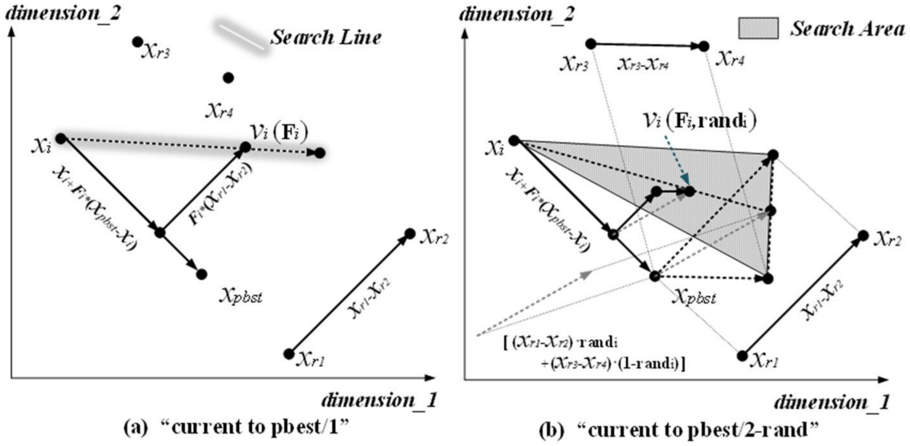
$$\vec{v}_{i,G} = \vec{x}_{i,\mathrm{G}} + F_i \cdot (\vec{x}_{pbest,\mathrm{G}} - \vec{x}_{i,\mathrm{G}}) + F_i \cdot (\vec{x}_{r1,\mathrm{G}} - \vec{x}_{r2,\mathrm{G}}) \quad (16)$$

In Equation (16), the individual $\vec{x}_{pbest,\mathrm{G}}$ is randomly selected from the top $N \times p(p \in [0,1])$ members in the G-th generation. "current to pbest/1" depends on the control parameter p to balance exploitation and exploration (small p behaves more greedily). $\vec{x}_{\mathrm{i,G}}$ and $\vec{x}_{r1,\mathrm{G}}$ are selected from P in the same way as in Equation (12), while $\vec{x}_{r2,\mathrm{G}}$ is randomly chosen from the union $\mathrm{P} \cup \mathrm{A}$, of the current population and the archive. We present an improved mutation strategy "current to pbest/2-rand" to improve the search range based on mutation strategy "current to pbest/1", which is expressed as follows:

$$\vec{v}_{i,G} = \vec{x}_{i,\mathrm{G}} + F_i \cdot (\vec{x}_{pbest,\mathrm{G}} - \vec{x}_{i,\mathrm{G}}) + F_i \cdot [(\vec{x}_{r1,\mathrm{G}} - \vec{x}_{r2,\mathrm{G}}) \cdot \mathrm{rand}_i + (\vec{x}_{r3,\mathrm{G}} - \vec{x}_{r4,\mathrm{G}}) \cdot (1 - \mathrm{rand}_i)] \quad (17)$$

where rand_i is a uniformly distributed random number between [0,1]. $\vec{x}_{\mathrm{i,G}}$, $\vec{x}_{r1,\mathrm{G}}$ and $\vec{x}_{r3,\mathrm{G}}$ are selected randomly and different within the range {1, NP} from P in the same way as in formula (16), while $\vec{x}_{r2,\mathrm{G}}$ and $\vec{x}_{r4,\mathrm{G}}$ is randomly chosen from the union, $\mathrm{P} \cup \mathrm{A}$, of the current population and the archive. The two mutation strategies are illustrated in Figure 2.

As seen in Figure 2a, $\vec{v}_{i,G}$ is the mutation individual generated for individual $\vec{x}_{i,\mathrm{G}}$. According to the principle of vector addition, the position of $\vec{v}_{i,G}$ changes with the associated mutation factor F_i, and its position only exists on this "Search Line". Mutation strategy "current to pbest/2-rand" uses rand_i and linear combination of $(\vec{x}_{r1,\mathrm{G}} - \vec{x}_{r2,\mathrm{G}}) \cdot \mathrm{rand}_i + (\vec{x}_{r3,\mathrm{G}} - \vec{x}_{r4,\mathrm{G}}) \cdot (1 - \mathrm{rand}_i)$ to expand the search range. By the varying value rand_i, $\vec{v}_{i,G}$ can search anywhere in the shaded triangle area with the change of the F_i and rand_i (see Figure 2b). Obviously, the search range of "current to pbest/2-rand" is much larger than that of "current to pbest/1".



(a) "current to pbest/1"

(b) "current to pbest/2-rand"

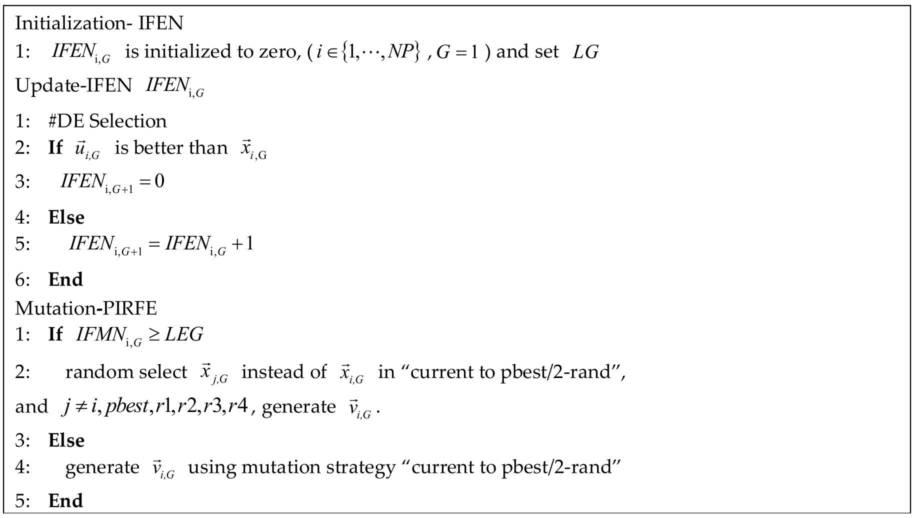
Figure 2. Illustration of the DE mutation strategy in two dimensions.

3.2.2. The PIRFE Strategy

The PIRFE strategy is proposed to avoid individuals in a local optimum lead useless evolution. When an individual falls into a local optimal, it is extremely difficult to produce an effective mutation to jump out of the local optimal. The individual failing evolution number (IFEN) is defined to record the number of individual failing evolution, and it is initialized to zero. $\vec{x}_{i,G}$ generates $\vec{u}_{i,G}$ through mutation and crossover. If $\vec{x}_{i,G+1}$ produces a failing evolution and $\vec{x}_{i,G+1}$ is better than $\vec{u}_{i,G+1}$, $IFEN_{i,G+1}$ is equal to $IFEN_{i,G}$ plus one, otherwise $IFEN_{i,G+1}$ will reset to zero. $IFEN_{i,G+1}$ is illustrated as follows:

$$IFEN_{i,G+1} = \begin{cases} 0 & \text{if } \vec{u}_{i,G+1} \text{ is better than } \vec{x}_{i,G+1} \\ IFEN_{i,G}+1 & \text{otherwise} \end{cases} \tag{18}$$

If $\vec{x}_i$ produces failing evolutions for $IFEN_i$ generations, and $IFEN_i$ is more than limit evolution generations (see formula (18)), $\vec{x}_i$ falls into local optimal. The LEG represents the largest evolution generation allowing failure evolutions. In the next evolutionary process, we should avoid involving $\vec{x}_i$. $\vec{x}_i$ should be replaced by a new individual randomly selected in population. The framework of the PIRFE strategy is as follows (Figure 3):



```
Initialization- IFEN
    $IFEN_{i,G}$ is initialized to zero, ($i \in \{1,\cdots,NP\}$, $G = 1$) and set $LG$
Update-IFEN $IFEN_{i,G}$
    #DE Selection
    If $\vec{u}_{i,G}$ is better than $\vec{x}_{i,G}$
        $IFEN_{i,G+1} = 0$
    Else
        $IFEN_{i,G+1} = IFEN_{i,G} + 1$
    End
Mutation-PIRFE
    If $IFMN_{i,G} \geq LEG$
        random select $\vec{x}_{j,G}$ instead of $\vec{x}_{i,G}$ in "current to pbest/2-rand",
    and $j \neq i, pbest, r1, r2, r3, r4$, generate $\vec{v}_{i,G}$.
    Else
        generate $\vec{v}_{i,G}$ using mutation strategy "current to pbest/2-rand"
    End
```

Figure 3. Pseudocode of PIRFE strategy.

3.2.3. Control Parameters Assignments

SHADE maintains a historical memory with H entries for both DE control parameters CR, F, M_{CR} and M_F. The scaling factor $F \in [0,\ 1]$ controls the magnitude of the differential mutation operator and $CR \in [0,\ 1]$ is the crossover rate. In the beginning, the contents of $M_{CR,k}$, $M_{F,k}$ $(k = 1, \cdots, H)$ are all initialized to 0.5. In each generation G, the control parameters CR_i and F_i used by each individual x_i are generated by randomly selecting an index r_i from $[1,\ H]$, and then applying the formulas (19) and (20):

$$CR_i = \begin{cases} 0 & \text{if } M_{CR,\mathrm{r}_i} = \perp \\ \mathrm{randn}_i\left(M_{CR,\mathrm{r}_i}, 0.1\right) & \text{otherwise} \end{cases} \tag{19}$$

$$F_i = \mathrm{randc}_i\left(M_{F,\mathrm{r}_i}, 0.1\right) \tag{20}$$

In case a value for CR_i outside of [0, 1] is generated, it is replaced by the limit value (0 or 1) closest to the generated value. When $F_i > 1$, is truncated to 1, and when $F_i \leq 0$, formula (20) is repeatedly applied to generate a valid value. These manners are determined according to the procedure for JADE [38]. In formula (19), if M_{CR,r_i} has been assigned the "terminal value" $\perp$, CR_i is set to 0.

In each generation, in formula (17), CR_i and F_i values that succeed in generating a trial individual $\vec{u}_{i,G}$ better than the parent individual $\vec{x}_{i,G}$ are recorded as S_{CR}, S_F. At the end of the generation, the contents of memory are updated as follows:

$$M_{CR,k,G+1} = \begin{cases} \perp & \text{if } M_{CR,k,G} = \perp \text{ or } \max(S_{CR}) = 0 \\ \mathrm{mean}_{WA}(S_{CR}) & \text{if } S_{CR} \neq \varnothing \\ M_{CR,k,G} & \text{otherwise} \end{cases} \tag{21}$$

$$M_{F,k,G+1} = \begin{cases} \mathrm{mean}_{WL}(S_F) & \text{if } S_F \neq \varnothing \\ M_{F,k,G} & \text{otherwise} \end{cases} \tag{22}$$

An index k (1 < k < H) determines the position in the memory to update. At the beginning of the search k is initialized to 1. k is incremented whenever a new element is inserted into the history. If > H, k is set to 1. In generation G, the k-th element in the memory is updated. In the update formula (21) and (22), when all individuals in generation G fail to generate an individual better than the parent, i.e., $S_{CR} = S_F = \varnothing$, the memory is not updated. Also, the weighted mean $mean_{WA}(S_{CR})$ is computed according to formula (23) by Peng et al. [44]. The weighted Lehmer mean $mean_{WL}(S_F)$ is computed using the formula below, and as with $mean_{WA}(S_{CR})$:

$$mean_{WL}(S_F) = \sum\nolimits_k^{|S_F|} w_k \cdot S_{F,k}^2 / \sum\nolimits_k^{|S_F|} w_k \cdot S_{F,k} \tag{23}$$

$$mean_{WA}(S_{CR}) = \sum\nolimits_{k=1}^{|S_{CR}|} w_k \cdot S_{CR,k} \tag{24}$$

$$w_k = \triangle f_k / \sum\nolimits_{k=1}^{|S_{CR}|} \triangle f_k \tag{25}$$

where $\triangle f_k = \left| f(\vec{u}_{i,G}) - f(\vec{x}_{i,G}) \right|$. In the same paper, they also proposed a restart strategy for JADE [38].

LSHADE put forward a new method of NP setting—LPSR which reduces the population linearly. The population size at generation 1 is N_{init}, and the population at the end of the run is N_{min}. After each generation G, the population size in the next generation, N_{G+1} is computed according to formula (26):

$$N_{G+1} = \mathrm{round}[((N_{min} - N_{init})/MAXNFE) \cdot NFE + N_{init}] \tag{26}$$

If N_{min} is set to the smallest possible value, the evolutionary operators can be applied in the case of iLSHADE, N_{min} = 6 because the mutation strategy "current to pbest/2-rand" showed as formula (17) requires 4 individuals. NFE is the current number of fitness evaluations, and $MAXNFE$ is the maximum number of fitness evaluations. Whenever $N_{G+1} < N_G$, the $(N_G - N_{G+1})$ worst-ranking individuals are deleted from the population. Similarly, the external archive size $|\mathrm{A}|$ is set to N_{init} multiplied by a parameter r^{arc}, $|\mathrm{A}| = \mathrm{round}(N_{init} \times r_{arc})$. In addition, external archive A is same as population, whenever $|\mathrm{A}|_{G+1} < |\mathrm{A}|_G$, the $(|\mathrm{A}|_G - |\mathrm{A}|_{G+1})$ worst-ranking individuals are deleted from the A. The p value for "current to pbest/2-rand" in each generation G is computed as follows:

$$p = (p_{max} - p_{min}) \cdot \mathrm{rand}(0,1) + p_{min} \tag{27}$$

where $\mathrm{rand}(0,1)$ returns a uniformly distributed random number in $[0,1]$, p is a random value in $[p_{max}, p_{min}]$. Finally, the pseudo-code of the iLSHADE algorithm is given in Figure 4.

```
$G \leftarrow 1$, $FES \leftarrow 0$ ,Archive $\mathbf{A} \leftarrow \varnothing$;
Initialize population $\mathbf{P}_g = (\vec{x}_{1,G}, \cdots, \vec{x}_{NP,G})$ randomly
Initialization-IFEN // described in Figure 3
Set $M_F[H] = 0.2$, Set other values in $M_F$ to 0.5,
Set $M_{CR}[H] = 0.8$, Set other values in $M_{CR}$ to 0.8,
$k \leftarrow 1$ // index counter
while the termination criteria are not meet do
  $S_{CR} \leftarrow \varnothing$, $S_F \leftarrow \varnothing$,
  for $i = 1$ to $NP$ do
    $r_i \leftarrow$ select from $[1, H]$ randomly
    if $M_{CR,ri} < 0$ then $CR_{i,g} \leftarrow 0$
    else $CR_{i,g} \leftarrow \mathrm{randn}_i(M_{CR,r_i}, 0.1)$
    end if
    $F_{i,g} \leftarrow \mathrm{randc}_i(M_{F,r_i}, 0.1)$
    $\vec{u}_{i,G} \leftarrow$ Mutation-PIRFE with "current to pbest/2-rand" // described in Figure 3
  end for
  for $i = 1$ to $NP$ do
    if $\vec{u}_{i,G}$ is better than $\vec{x}_{i,G}$ then
      $\vec{x}_{i,G+1} \leftarrow \vec{u}_{i,G}$;
      $\vec{x}_{i,G} \rightarrow A$, $CR_{i,g} \rightarrow S_{CR}$, $F_{i,g} \rightarrow S_F$
    else
      $\vec{x}_{i,G+1} \leftarrow \vec{x}_{i,G}$;
    end if
    Update-IFEN $IFEN_{i,G}$ // described in Figure 3
  end for
  Shrink $\mathbf{A}$, if necessary
  Update $M_{CR,i}$ and $M_{F,i}$ except $i$ = H
  Apply LPSR strategy using formula (26)// linear population size reduction
  Update $p$ using formula (27)
  $G \leftarrow G + 1$, $FES \leftarrow FES + NP$
end while
```

Figure 4. Pseudocode of iLSHADE.

4. Numerical Experiment

The iLSHADE algorithm was tested in both low and high dimension on a set of 10 benchmark functions demonstrated in Table 1. Table 1 indicates benchmark problems with different structures and characteristics. In the table, "O-V" means the optimum fitness and "O-S" stands for the optimum solution. f1, f2, f3, f5 and f6 are unimodal optimization problems to evaluate the convergence, while f4, f7, f8,f9 and f10 are multimodal optimization problems with a huge number of local optima to test the convergence precision [18].

Table 1. Details of benchmark problems.

Benchmark Function	Name	Domain	O-V	O-S
$f_1 = \sum_{i=1}^{n} x_i^2$	Sphere	$[-100, 100]^n$	0	$\{0, 0, \cdots, 0\}$
$f_2 = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_i^n \lvert x_i \rvert$	Schwefel (2.2)	$[-100, 100]^n$	0	$\{0, 0, \cdots, 0\}$
$f_3 = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	Schwefel (1.2)	$[-100, 100]^n$	0	$\{0, 0, \cdots, 0\}$
$f_4 = \sum_{i=1}^{n-1} \left[100 (x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right]$	Rosenbrock	$[-30, 30]^n$	0	$\{0, 0, \cdots, 0\}$
$f_5 = \sum_{i=1}^{n} \left(\lfloor x_j + 0.5 \rfloor \right)^2$	Step	$[-100, 100]^n$	0	$\{0, 0, \cdots, 0\}$
$f_6 = \sum_{i=1}^{n} i x_i^4$	Quartic	$[-1.28, 1.28]^n$	0	$\{0, 0, \cdots, 0\}$
$f_7 = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{\lvert x_i \rvert} \right)$	Schwefel (2.26)	$[-500, 500]^n$	−418.9n	*
$f_8 = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	Rastrigin	$[-5.12, 5.12]^n$	0	$\{0, 0, \cdots, 0\}$
$f_9 = -20\exp\left(-0.2\sqrt{1/n \sum_{i=1}^{n} x_i^2}\right) - \exp(1/n \cos(2\pi x_i)) + 20 + e$	Ackley	$[-32, 32]^n$	0	$\{0, 0, \cdots, 0\}$
$f_{10} = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} + \prod_i^n \cos\left(\frac{x_i}{\sqrt{i}}\right)$	Griewank	$[-600, 600]^n$	0	$\{0, 0, \cdots, 0\}$

Note: * (420.9876, 420.9876, . . . , 420.9876).

The iLSHADE is compared to DE and other improved DE like LSHADE, JADE, CoDE and jDE. The number of function evaluations is used to appraise the convergence. These experiments are made on a personal computer, Windows10, Intel(R) Core(TM) i7-5500U CPU@ 2.40GHZ, RAM 8.00 GB. The dimension of benchmark functions is D = 10 in low dimension and 30 in high dimension, and 51 runs of an algorithm were needed for each function. The maximum number of objective function evaluations is D × 10,000. The optimal values are known for all benchmark functions.

In the experiments, the parameters in LSHADE, JADE, CoDE, jDE and DE were kept unchanged refer to [33,36,38,39,42], and the parameter setting in the iLSHADE is same as LSHADE except the following parameters:

- Using "current to pbest/2-rand" mutation strategy,
- The p value for mutation strategy is computed as $p_G = \text{rand}[p_{\min}, p_{\max}]$, where $p_{\min} = 2/NP$ is set such that when $\vec{x}_{pbest,G}$ is selected, at least 2 individuals are needed, and $p_{\max} = 0.25$.
- Initial population size $N_{init} = 15 log(D)\sqrt{D}$, the control parameter of external archive size $r^{arc} = 2$.
- Historical memory size H = 6; set a final pair of parameters $M_F[\text{H}] = 0.2$ and $M_{CR}[\text{H}] = 0.8$, other M_F values are initialized to 0.5 and other M_{CR} are initialized to 0.8.
- PIRFE parameter LEG = 50.

The aggregate results of statistical testing ($+$, $-$, $\approx$) on 10 functions are shown in Tables 2 and 3. The symbols $+$, $-$, $\approx$ indicate that a given algorithm performed significantly better ($+$), significantly worse ($-$), or not significantly different better or worse ($\approx$) compared to iLSHADE using the Wilcoxon rank-sum test [45] (significantly, p < 0.05).

Table 2. Experimental results of 10 test functions in low dimension.

f	**LSHADE** **Mean** **(Std Dev)**	**JADE** **Mean** **(Std Dev)**	**CoDE** **Mean** **(Std Dev)**	**jDE** **Mean** **(Std Dev)**	**DE** **Mean** **(Std Dev)**	**iLSHADE** **Mean** **(Std Dev)**
f_1	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	8.61×10^{-36} $(7.18 \times 10^{-36}) -$	5.93×10^{-38} $(7.34 \times 10^{-38}) -$	1.15×10^{-38} $(1.18 \times 10^{-38}) -$	2.73×10^{-46} $(1.42 \times 10^{-45}) -$	0.00×10^{0} (0.00×10^{0})
f_2	6.49×10^{-49} $(4.38 \times 10^{-48}) -$	8.75×10^{-20} $(4.68 \times 10^{-20}) -$	7.88×10^{-22} $(6.58 \times 10^{-22}) -$	1.38×10^{-22} $(1.01 \times 10^{-22}) -$	8.03×10^{-25} $(1.65 \times 10^{-24}) -$	3.36×10^{-64} (2.05×10^{-63})
f_3	1.10×10^{-91} $(5.90 \times 10^{-91}) -$	2.81×10^{-35} $(2.75 \times 10^{-35}) -$	2.43×10^{-39} $(4.02 \times 10^{-39}) -$	1.24×10^{-40} $(1.79 \times 10^{-40}) -$	4.58×10^{-45} $(2.03 \times 10^{-44}) -$	0.00×10^{0} (0.00×10^{0})
f_4	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	1.47×10^{-05} $(1.96 \times 10^{-05}) -$	7.39×10^{-03} $(9.11 \times 10^{-03}) -$	7.82×10^{-02} $(5.53 \times 10^{-01}) -$	0.00×10^{0} (0.00×10^{0})
f_5	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0})\approx$	0.00×10^{0} (0.00×10^{0})
f_6	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	1.40×10^{-71} $(3.27 \times 10^{-71}) -$	6.47×10^{-71} $(1.52 \times 10^{-70}) -$	1.40×10^{-72} $(3.66 \times 10^{-72}) -$	8.39×10^{-88} $(4.17 \times 10^{-87}) -$	0.00×10^{0} (0.00×10^{0})
f_7	-4189.83 $(2.73 \times 10^{-12}) \approx$	-4189.83 $(2.73 \times 10^{-12}) \approx$	-4189.83 $(2.73 \times 10^{-12}) \approx$	-4189.83 $(2.73 \times 10^{-12}) \approx$	-4189.83 $(2.73 \times 10^{-12}) \approx$	-4189.83 (2.73×10^{-12})
f_8	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	3.06×10^{0} $(2.47 \times 10^{0}) -$	0.00×10^{0} (0.00×10^{0})
f_9	3.72×10^{-15} $(9.55 \times 10^{-16}) \approx$	3.86×10^{-15} $(6.90 \times 10^{-16}) -$	4.00×10^{-15} $(2.37 \times 10^{-30}) -$	3.93×10^{-15} $(4.93 \times 10^{-16}) -$	3.93×10^{-15} $(4.93 \times 10^{-16}) -$	3.72×10^{-15} (9.55×10^{-16})
f_{10}	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	2.42×10^{-12} $(6.51 \times 10^{-12}) -$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	3.22×10^{-04} $(2.28 \times 10^{-03}) -$	8.58×10^{-02} $(6.17 \times 10^{-02}) -$	0.00×10^{0} (0.00×10^{0})
$-$	2	6	6	7	8	
$+$	0	0	0	0	0	
$\approx$	8	4	4	3	2	

Table 3. Experimental results of 10 test functions in high dimension.

f	**LSHADE** **Mean** **(Std Dev)**	**JADE** **Mean** **(Std Dev)**	**CoDE** **Mean** **(Std Dev)**	**jDE** **Mean** **(Std Dev)**	**DE** **Mean** **(Std Dev)**	**iLSHAD** **EMean** **(Std Dev)**
f_1	1.12×10^{-90} $(6.44 \times 10^{-90}) -$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	9.85×10^{-19} $(7.04 \times 10^{-19}) -$	4.31×10^{-41} $(4.35 \times 10^{-41}) -$	9.34×10^{-44} $(2.74 \times 10^{-43}) -$	0.00×10^{0} (0.00×10^{0})
f_2	2.09×10^{-42} $(1.03 \times 10^{-41}) -$	4.11×10^{-27} $(4.89 \times 10^{-27}) -$	4.01×10^{-12} $(1.34 \times 10^{-12}) -$	3.48×10^{-24} $(1.96 \times 10^{-24}) -$	1.16×10^{-05} $(8.17 \times 10^{-05}) -$	4.88×10^{-58} (1.55×10^{-57})
f_3	3.85×10^{-81} $(1.74 \times 10^{-80}) -$	3.58×10^{-49} $(7.53 \times 10^{-49}) -$	4.22×10^{-19} $(3.41 \times 10^{-19}) -$	7.27×10^{-43} $(1.05 \times 10^{-42}) -$	1.16×10^{-46} $(6.85 \times 10^{-46}) -$	0.00×10^{0} (0.00×10^{0})
f_4	1.40×10^{-25} $(9.70 \times 10^{-25}) -$	$1.85 \times 10^{+01}$ $(1.01 \times 10^{+01}) -$	$1.82 \times 10^{+01}$ $(3.29 \times 10^{0}) -$	$1.15 \times 10^{+01}$ $(8.23 \times 10^{0}) -$	2.62×10^{0} $(2.60 \times 10^{0}) -$	0.00×10^{0} (0.00×10^{0})
f_5	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	1.96×10^{-02} $(1.39 \times 10^{-01}) -$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	4.22×10^{0} $(7.61 \times 10^{0}) -$	0.00×10^{0} (0.00×10^{0})
f_6	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	2.35×10^{-33} $(2.93 \times 10^{-33}) -$	1.79×10^{-69} $(3.86 \times 10^{-69}) -$	1.74×10^{-59} $(1.10 \times 10^{-58}) -$	0.00×10^{0} (0.00×10^{0})
f_7	$-12{,}569.49$ $(1.82 \times 10^{-12}) \approx$	$-12{,}567.16$ $(1.64 \times 10^{+01}) -$	$-12{,}569.49$ $(1.82 \times 10^{-12}) \approx$	$-12{,}569.49$ $(1.82 \times 10^{-12}) \approx$	-11552.14 $(3.68 \times 10^{+02}) -$	$-12{,}569.49$ (1.88×10^{-05})
f_8	1.74×10^{-16} $(6.35 \times 10^{-16}) +$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	8.38×10^{-12} $(9.18 \times 10^{-12}) +$	4.83×10^{0} $(3.86 \times 10^{0}) -$	$3.62 \times 10^{+01}$ $(1.44 \times 10^{+01}) -$	3.16×10^{-11} (1.66×10^{-10})
f_9	4.00×10^{-15} $(2.37 \times 10^{-30}) \approx$	4.76×10^{-15} $(1.46 \times 10^{-15}) -$	2.74×10^{-10} $(1.08 \times 10^{-10}) -$	5.60×10^{-15} $(1.77 \times 10^{-15}) -$	2.64×10^{-01} $(5.21 \times 10^{-01}) -$	4.00×10^{-15} (2.37×10^{-30})
f_{10}	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	1.55×10^{-03} $(3.81 \times 10^{-03}) -$	3.05×10^{-17} $(1.08 \times 10^{-16}) -$	0.00×10^{0} $(0.00 \times 10^{0}) \approx$	7.99×10^{-03} $(1.46 \times 10^{-02}) -$	0.00×10^{0} (0.00×10^{0})
$-$	4	6	7	7	10	
$+$	1	1	1	0	0	
$\approx$	5	3	2	3	0	

1 Simulation in low dimension.

Table 2 summarizes the experimental results in low dimension. It shows that iLSHADE eventually converges to optimum for 8 test functions except f_2 and f_9. Moreover, iLSHADE significantly outperforms JADE, CoDE, jDE, DE. Compared to LSHADE, iLSHADE beats LSHADE on two test functions f_2 and f_3. Especially in f_3, only iLSHADE converges to the global optimal value. Overall, iLSHADE performs better than other algorithms on low dimension optimization problems.

2 Simulation in high dimension.

The experimental results in high dimension are summarized in Table 3. iLSHADE eventually converges to optimum for 7 test functions except f_2, f_8 and f_9. iLSHADE performs the best in f_2 and f_9, although it does not converge to the global optimal. While on f_8, JADE performs the best. The proposed algorithm is not good as expected. Above all, iLSHADE has an obvious advantage over other algorithms on high dimension optimization problems.

To sum up, iLSHADE is suitable for both low dimension and high dimension, meaning that the improvement proposed in this paper is effective.

5. Implementation of iLSHADE for LSLCHS

5.1. Solution Structure and Initial Population

To handle constraints and calculate objective function, solution structure for the LSLCHS comprises a group of monthly water levels as the decision variables shown as follows.

$$X = \begin{bmatrix} X^1 \\ X^2 \\ \vdots \\ X^M \end{bmatrix} = \begin{bmatrix} x_1^1, x_2^1, \cdots x_T^1 \\ x_1^2, x_2^2, \cdots x_T^2 \\ \vdots \\ x_1^M, x_2^M, \cdots x_T^M \end{bmatrix} \tag{28}$$

where M is the number of hydropower stations, T (12 month in a year) is the number of intervals. In algorithms relying on heuristic search, initial population is an important issue to convergence speed and population diversity. The iLSHADE has a large initial population size based on LPSR and ensures the diversity of the population by random initialization.

5.2. Constraint Handling

It is multiple and complicate for flow constraint, power generating constraint, amplitude of water level variation and hydraulic connection of cascade. The handling measure currently used for water balance constraint, water level constraint and boundary condition is often corrected to the boundary [18,46]. It has defects such as: (1) The direction of the entering feasible domain is relatively simple and centered on the boundary because of the excessive attention to the rapid into the feasible area; (2) When there are multiple feasible domains, it is easy to ignore small feasible areas;

The ε-constrained method is first proposed by Takahama et al. [47], which relaxes the greed of the feasibility criterion to the constraint conditions. The ε value is set as threshold value in ε-constrained method. In general, constrained optimization maximum problems can be mathematically formulated as follows:

$$\max f\left(\vec{x}\right),\ \vec{x} = (x_1, \ldots, x_D) \tag{29}$$

$$\text{subject to} \begin{cases} g_i\left(\vec{x}\right) \leq 0 \ \ i = 1, 2, \cdots, m \\ h_j\left(\vec{x}\right) = 0 \ \ j = 1, 2, \cdots, n \end{cases} \tag{30}$$

where $g_i\left(\overrightarrow{x}\right)$ is inequality constraints, m is the total number of inequality constraint. $h_j\left(\overrightarrow{x}\right)$ is equality constraints and n is the total number of equality constraints. The value of constraint violation $\varphi\left(\overrightarrow{x}\right)$ can be calculated in the following formula (30), and $\varphi\left(\overrightarrow{x}\right)$ of infeasible solutions is bigger than 0.

$$\varphi\left(\overrightarrow{x}\right) = \sum_{i=0}^{m} \max\left(0, g_i\left(\overrightarrow{x}\right)\right) + \sum_{j=1}^{n}\left(\left|h_j\left(\overrightarrow{x}\right)\right|, 0\right) \tag{31}$$

When the constraint violation values of both solutions are smaller than ε value, the one with better objective function value is selected. Otherwise, the one with smaller constraint violation value is selected. Overall, when any of the following conditions are met, $\overrightarrow{x}_i$ is superior to $\overrightarrow{x}_j$:

$$\begin{cases} f\left(\overrightarrow{x}_i\right) < f\left(\overrightarrow{x}_j\right), \text{ if } g\left(\overrightarrow{x}_i\right) \le \varepsilon \cap g\left(\overrightarrow{x}_j\right) \le \varepsilon \\ f\left(\overrightarrow{x}_i\right) < f\left(\overrightarrow{x}_j\right), \text{ if } g\left(\overrightarrow{x}_i\right) = g\left(\overrightarrow{x}_j\right) \\ g\left(\overrightarrow{x}_i\right) < g\left(\overrightarrow{x}_j\right), \text{ otherwise} \end{cases} \tag{32}$$

and the final effectiveness of e-constrained method strongly depends on the control method of ε value. Takahama et al. [47] proposed the following method,

$$\varepsilon(0) = \varphi\left(P_0^{\theta}\right) \tag{33}$$

$$\varepsilon(t) = \begin{cases} \varepsilon(0)(1 - t/T_c)^{cp} & 0 < t < T_c \\ 0 & t > T_c \end{cases} \tag{34}$$

where P_0^{θ} is the top θ-th individual in the initial population, cp is a control parameter. If the number of iterations t is less than a given threshold value T_c, the ε value declines in an exponential way. Otherwise, ε is set to 0 (see formula (33) and (34)).

The ε-constrained method can expand search space, avoid the constraint correction for unfeasible solutions and enable to search infeasible region that is around feasible region. However, the constraints consist of flow, power generating and amplitude of water level. Operating water level in LSLCHS problem is multiple and complex. Their units are not integrated, and the physical quantities corresponding to the same level of different reservoirs are different. To solve the above problems, we proposed ε-constrained in cascade reservoir operation method (ε-CRO) with unify different physical quantities constraint violation. The ε-CRO chooses water to unify different physical quantities constraint violation because all constraints can be converted to outflow constraint and flow accumulated over time as water. The feasible range of outflow $Q_{i,t}$ is expressed in formula (35) and (36).

$$\overline{Q}_{i,t+1} = \min\begin{cases} Q_{i,t+1}^{\max} \\ \left(V(Z_{i,t}) - V(Z_{i,t+1}^{\min})\right)/\Delta t \\ (V(Z_{i,t}) - V(Z_{i,t} - \Delta Z_i))/\Delta t \end{cases} \tag{35}$$

$$\underline{Q}_{i,t+1} = \max\begin{cases} Q_{i,t+1}^{\min} \\ I_{i,t} + \left(V(Z_{i,t}) - V(Z_{i,t+1}^{\max})\right)/\Delta t \\ I_{i,t} + (V(Z_{i,t}) - V(Z_{i,t} + \Delta Z_i))/\Delta t \\ Q_{i,t+1}^{\min\ N} \end{cases} \tag{36}$$

where $Z_{i,t}$ represents the i-th reservoir water level at the -th period, $Z(V)$ is the relationship between water level and storage. $\overline{Q}_{i,t+1}$ and $\underline{Q}_{i,t+1}$ stand for the minimum and maximum outflow under all constraints. $Q_{i,t+1}^{\min N}$ is the outflow for guaranteed output and Δt stands for time horizon. The value of constraint violation in LSLCHS problem $\varphi_{CRO}\left(\vec{x}\right)$ can be calculated as follows:

$$\varphi_{CRO}\left(\vec{x}\right) = \sum_{i=1}^{M}\sum_{t=1}^{T}\left[\max\left(0, Q_{i,t} - \overline{Q}_{i,t+1}\right) + \max\left(0, \underline{Q}_{i,t+1} - Q_{i,t}\right)\right] \tag{37}$$

$\varepsilon_{CRO}(0)$ and $\varepsilon_{CRO}(t)$ are calculated like in [47] except $\theta = 0.5$, $cp = 1$ and $T_c = 0.5 * MAXNFE$.

6. Case Study

6.1. Description of Case Study

The Jinsha River, the upper stretch of the Yangtze River, is 2290 km long with a 485,000 km^2 basin area flowing through the provinces of Qinghai, Sichuan, and Yunnan in western China (See in Figure 5). Along the river, there are four large hydropower stations with large installed capacity, huge regulating storage and high water head. The total installed capacity of the four large hydropower stations is twice more than the Three Gorges Project (the largest hydropower station in the world). The main parameters of these hydropower stations are listed in Table 4.

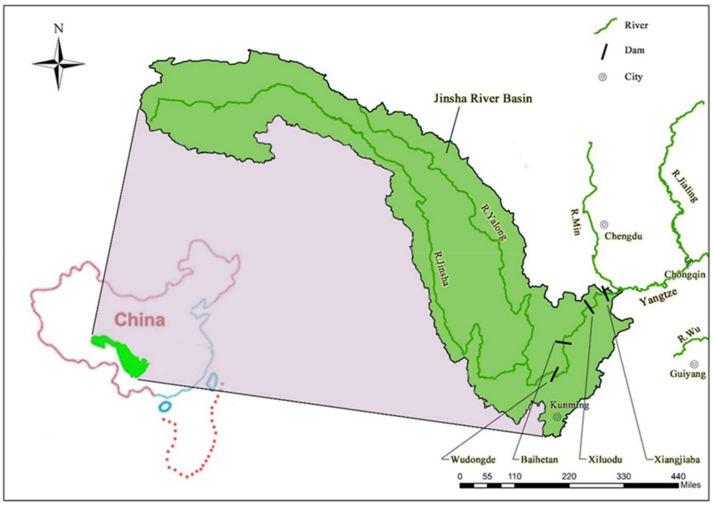


Figure 5. The location of the Jinsha River Basin in China.

Table 4. The main parameters of four large hydropower stations in Jinsha River.

Parameter	Wudongde	Baihetan	Xiluodu	Xiangjiaba
Adjustment ability	Season	Annual	Annual	Season
Regulating storage (billion m^3)	2.60	10.40	6.46	0.90
Hydro plant discharge range (m^3/s)	[49,400, 906]	[49,700, 905]	[43,700, 1500]	[49,800, 1500]
Upriver water level range (m)	[975, 945]	[825, 765]	[600, 540]	[380, 370]
Installed capacity (MW)	12000	16000	13860	6400
Normal water level (m)	975	825	600	380

6.2. Results and Analysis

In the case, the four large hydropower stations are all taken into consideration. According to historical runoff from 1959 to 2014 in the basin, three typical years are chosen to be the inflow conditions: wet year (historical runoff of 1999), normal year (2008) and dry year (1969). Simulation results of iLSHADE are compared to LSHADE, JADE and CoDE in three typical years. The initial water level and terminal water level of all the hydropower stations are set to the normal water level. The schedule period consists of 12 intervals with one month for each interval. In addition, the parameters in all algorithms are the same as those mentioned in Section 4. The maximum evaluation time is set to 40,000.

Table 5 represents that iLSHADE gains the best benefit of power production in three typical years. The convergence process of different algorithms in dry year is shown in Figure 6. Compared to LSHADE, JADE and CoDE on average optimal benefit of 51 independent simulations are illustrated in Table 5, iLSHADE increases the power production by 2.02, 4.04, 2.39 (10^8 KWh) in wet year, 3.03, 7.37, 3.76 (10^8 KWh) in normal year, 3.48, 5.68, 1.96 (10^8 KWh) in dry year. Obviously, the proposed iLSHADE is superior when solving LSLCHS problem by obtaining the maximal benefit of power production efficiently. In particular, the standard deviation of 51 independent simulations in iLSHADE is 0.02 in wet year, 0.01 in normal year, 0.01 in dry year, which shows that the convergence stability of iLSHADE is better than other algorithms. Meanwhile, it can be seen easily from Figure 6a that iLSHADE can avoid premature convergence effectively, at the same evaluation times keep a fast convergence speed compared to LSHADE, CoDE and JADE. Figure 6b depicts that the $\varphi_{CRO}\left(\vec{x}\right)$ of iLSHADE and LSHADE frequent changes and always lower than $\varepsilon_{CRO}(t)$, until evaluation times is greater than T_c, $\varphi_{CRO}\left(\vec{x}\right)$ is limited to 0.

Table 5. Results of 51 independent simulations on generated energy optimization (10^8 KWh).

Method	Wet Year (1999)			Normal Year (2008)			Dry Year (1969)		
	Max	Mean	Std	Max	Mean	Std	Max	Mean	Std
iLSHADE	2425.03	2425.01	0.02	2268.13	2268.11	0.01	1814.36	1814.35	0.01
LSHADE	2423.86	2422.99	0.48	2266.88	2265.08	0.94	1812.64	1810.87	0.99
Diff	1.17	2.02		1.25	3.03		1.72	3.48	
JADE	2424.43	2420.97	1.32	2267.51	2260.74	2.43	1814.00	1808.67	2.234
Diff	0.6	4.04		0.62	7.37		0.36	5.68	
CoDE	2423.39	2422.62	0.30	2265.62	2264.35	0.64	1813.08	1812.39	0.39
Diff	1.64	2.39		2.51	3.76		1.28	1.96	

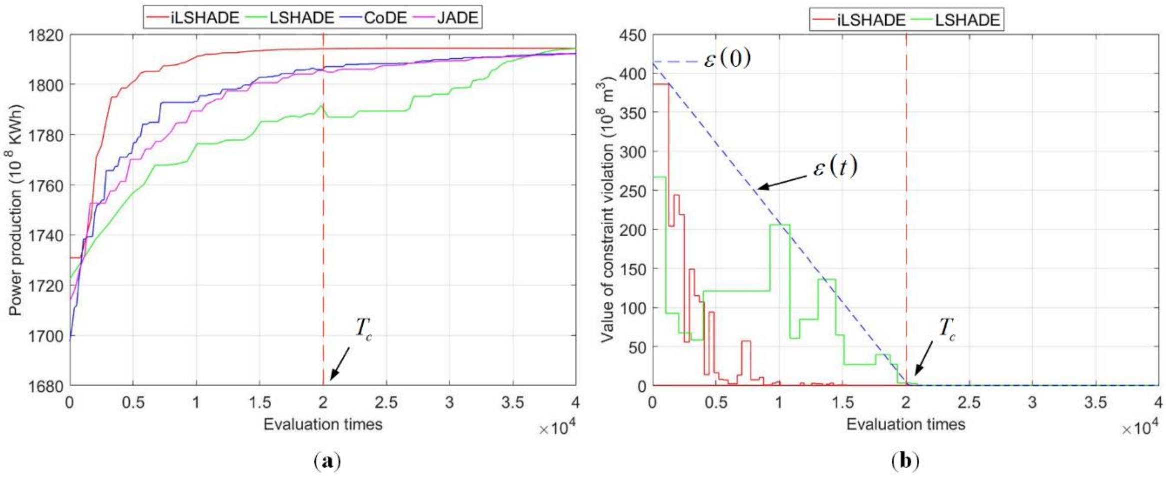
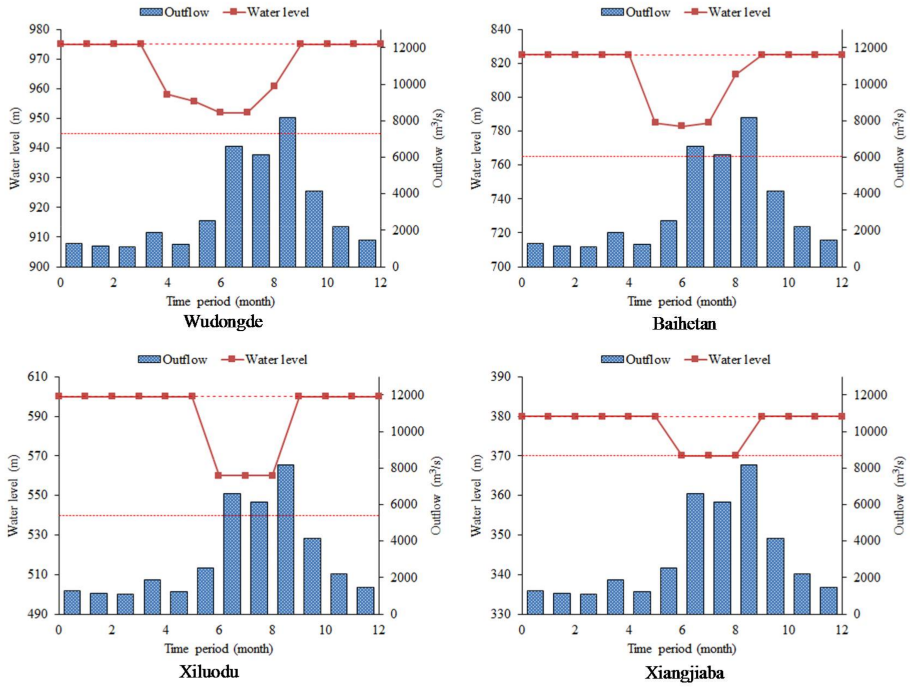


Figure 6. Convergence process of different algorithms.

The monthly reservoir water levels and outflow, as well as the optimal schedule result of Wudongde, Baihetan, Xiluodu and Xiangjiaba result in normal year obtained by iLSHADE are shown in Figure 7. Due to sufficient inflow, the cascade reservoir does not need to release storage capacity to meet the constraints of minimum outflow limit during the dry season from January to March. The inflow of Wudongde is very low in April, so the cascade reservoir needs to release storage capacity to meet the constraint requirements. As one of the upstream reservoirs, Wudongde first lowers the water level. To reduce the water spillage before flood season, Baihetan will lower its water level below the flood control level in advance and impound some water at the last period. Xiluodu and Xiangjiaba lower the water level in the last period. During the impoundment period, Wudongde and Baihetan store water to normal water level before Xiluodu and Xiangjiaba in October. In this way, the water in upstream reservoirs can utilize the downstream high hydraulic head to generate more electric power.



Wudongde

Baihetan

Xiluodu

Xiangjiaba

Figure 7. Optimal results in normal year by month.

Furthermore, the historical runoff data from 1959 to 2014 are selected for long sequence calculation. The parameters in all algorithms are the same as those mentioned in Section 6.2 and the maximum evaluation time is set to 40,000. The average adding annual power production that iLSHADE compares to LSHADE, JADE and CoDE is presented in Figure 8. It can be seen clearly from Figure 8 that iLSHADE is superior compared to other algorithm in solving the LSLCHS problem with different types of historical runoff from 1959 to 2014.

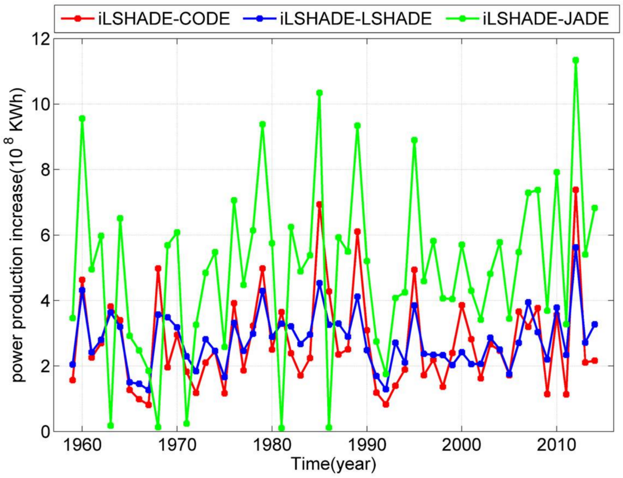


Figure 8. Historical runoff data from 1959 to 2014 for the annual power production increase that iLSHADE compares to LSHADE, JADE and CoDE.

According to the above analysis, proposed iLSHADE is superior when solving LSLCHS problem in different types of runoff by obtaining the maximize benefit of electric power production. All these experiment results fully demonstrate that iLSHADE is a competitive method to solve the LSLCHS problem.

7. Conclusions

An iLSHADE algorithm with new mutation strategy "current to pbest/2-rand" and PIRFE strategy has been developed in this paper to solve the LSLCHS problem. The significant modifications are mainly focused on preventing premature convergence and accelerating convergence. To verify the performance of iLSHADE, numerical simulation in both low and high dimension on a set of 10 benchmark functions has been done. Compared with other improved differential evolution algorithms, iLSHADE obtains better performance with all ten functions in the low dimension and nine functions in the high dimension. This indicates that the proposed new mutation strategy "current to pbest/2-rand" and PIRFE strategy in iLSHADE enhance the performance of original algorithm LSHADE effectively. Then iLSHADE is applied to solve LSLCHS problem for four large hydropower stations in Jinsha River. Compared to LSHADE, JADE and CoDE on average optimal benefit, iLSHADE increases the power production by 3.03, 7.37, 3.76 (10^8 KWh) in a normal year. In particular, the standard deviation of 51 independent simulations in iLSHADE is far lower than other algorithms. Moreover, according to its successful simulation performance with the historical runoff data from 1959 to 2014, iLSHADE can obtain better schedule results with lager generation benefits and better convergence property compared to LSHADE, JADE and CoDE. Above all, iLSHADE is a valid and reliable tool in solving the LSLCHS problem. Future research should consider the iLSHADE algorithm combined with other methods when solving multi objective scheduling problems in LSLCHS problem.

Acknowledgments: This work is supported by the National Natural Science Foundation Key Project of China (No. 91547208), the National Natural Science Foundation of China (No. 51579107), the National Key R&D Program of China (No. 2016YFC0402205) and the Foundation of Huazhong University of Science and Technology (No. 2017KFYXJJ209). Special thanks are given to the anonymous reviewers and editors for their constructive comments.

Author Contributions: Xiaohao Wen and Zhongzheng He designed and conducted the experiments. Xiaohao Wen wrote the draft of the paper. Zhongzheng He prepared the figures for this paper. Jianzhong Zhou proposed the main structure of this study. Chao Wang provided useful advice and made some corrections. All authors read and approved the final manuscript.

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

References

1. Lior, N. Sustainable energy development: The present (2011) situation and possible paths to the future. *Energy* **2012**, *43*, 174–191. [CrossRef]
2. Wu, X.-Y.; Cheng, C.-T.; Shen, J.-J.; Luo, B.; Liao, S.-L.; Li, G. A multi-objective short term hydropower scheduling model for peak shaving. *Int. J. Electr. Power Energy Syst.* **2015**, *68*, 278–293. [CrossRef]
3. Madani, K. Hydropower licensing and climate change: Insights from cooperative game theory. *Adv. Water Resour.* **2011**, *34*, 174–183. [CrossRef]
4. Xu, J.; Ni, T.; Zheng, B. Hydropower development trends from a technological paradigm perspective. *Energy Convers. Manag.* **2015**, *90*, 195–206. [CrossRef]
5. Chang, X.; Liu, X.; Zhou, W. Hydropower in china at present and its further development. *Energy* **2010**, *35*, 4400–4406. [CrossRef]
6. Tasdemiroglu, E. Development of small hydropower in Türkiye. *Energy* **1993**, *18*, 699–702. [CrossRef]
7. Zhou, Y.; Guo, S.; Xu, C.-Y.; Liu, P.; Qin, H. Deriving joint optimal refill rules for cascade reservoirs with multi-objective evaluation. *J. Hydrol.* **2015**, *524*, 166–181. [CrossRef]
8. Jiang, Z.; Li, A.; Ji, C.; Qin, H.; Yu, S.; Li, Y. Research and application of key technologies in drawing energy storage operation chart by discriminant coefficient method. *Energy* **2016**, *114*, 774–786. [CrossRef]

9. Chen, L.; Singh, V.P. Entropy-based derivation of generalized distributions for hydrometeorological frequency analysis. *J. Hydrol.* **2017**. [CrossRef]
10. Chen, L.; Singh, V.P.; Huang, K. Bayesian technique for the selection of probability distributions for frequency analyses of hydrometeorological extremes. *Entropy* **2018**, *20*, 117. [CrossRef]
11. Chen, L.; Singh, V.P.; Guo, S.; Zhou, J.; Zhang, J.; Liu, P. An objective method for partitioning the entire flood season into multiple sub-seasons. *J. Hydrol.* **2015**, *528*, 621–630. [CrossRef]
12. Chen, L.; Singh, V.P.; Lu, W.; Zhang, J.; Zhou, J.; Guo, S. Streamflow forecast uncertainty evolution and its effect on real-time reservoir operation. *J. Hydrol.* **2016**, *540*, 712–726. [CrossRef]
13. Djebou, D.C.S. Spectrum of climate change and streamflow alteration at a watershed scale. *Environ. Earth Sci.* **2017**, *76*, 653. [CrossRef]
14. Sohoulande Djebou, D.C.; Singh, V.P. Entropy-based index for spatiotemporal analysis of streamflow, precipitation, and land-cover. *J. Hydrol. Eng.* **2016**, *21*, 05016024. [CrossRef]
15. Xu, B.; Zhong, P.-A.; Wu, Y.; Fu, F.; Chen, Y.; Zhao, Y. A multiobjective stochastic programming model for hydropower hedging operations under inexact information. *Water Resour. Manag.* **2017**, *31*, 4649–4667. [CrossRef]
16. Mahmoud, M.; Dutton, K.; Denman, M. Dynamical modelling and simulation of a cascaded reserevoirs hydropower plant. *Electr. Power Syst. Res.* **2004**, *70*, 129–139. [CrossRef]
17. Cârdu, M.; Bara, T. Romanian achievement in hydro-power plants. *Energy Convers. Manag.* **1998**, *39*, 1193–1201. [CrossRef]
18. Wang, C.; Zhou, J.; Lu, P.; Yuan, L. Long-term scheduling of large cascade hydropower stations in Jinsha river, China. *Energy Convers. Manag.* **2015**, *90*, 476–487. [CrossRef]
19. Windsor, J.S. Optimization model for the operation of flood control systems. *Water Resour. Res.* **1973**, *9*, 1219–1226. [CrossRef]
20. Yoo, J.-H. Maximization of hydropower generation through the application of a linear programming model. *J. Hydrol.* **2009**, *376*, 182–187. [CrossRef]
21. Brandao, J.L.B. Performance of the equivalent reservoir modelling technique for multi-reservoir hydropower systems. *Water Resour. Manag.* **2010**, *24*, 3101–3114. [CrossRef]
22. Catalão, J.P.D.S.; Pousinho, H.M.I.; Mendes, V.M.F. Scheduling of head-dependent cascaded hydro systems: Mixed-integer quadratic programming approach. *Energy Convers. Manag.* **2010**, *51*, 524–530. [CrossRef]
23. Jiang, Z.; Qin, H.; Ji, C.; Feng, Z.; Zhou, J. Two dimension reduction methods for multi-dimensional dynamic programming and its application in cascade reservoirs operation optimization. *Water* **2017**, *9*, 634. [CrossRef]
24. Lee, J.H.; Labadie, J.W. Stochastic optimization of multireservoir systems via reinforcement learning. *Water Resour. Res.* **2007**, *43*. [CrossRef]
25. Nanda, J.; Bijwe, P.R. Optimal hydrothermal scheduling with cascaded plants using progressive optimality algorithm. *IEEE Trans. Power Appar. Syst.* **1981**, *PAS-100*, 2093–2099. [CrossRef]
26. Jiang, Z.; Ji, C.; Qin, H.; Feng, Z. Multi-stage progressive optimality algorithm and its application in energy storage operation chart optimization of cascade reservoirs. *Energy* **2018**, *148*, 309–323. [CrossRef]
27. Jiang, C.; Bompard, E. A self-adaptive chaotic particle swarm algorithm for short term hydroelectric system scheduling in deregulated environment. *Energy Convers. Manag.* **2005**, *46*, 2689–2696.
28. Cheng, C.T.; Liao, S.L.; Tang, Z.T.; Zhao, M.Y. Comparison of particle swarm optimization and dynamic programming for large scale hydro unit load dispatch. *Energy Convers. Manag.* **2009**, *50*, 3007–3014. [CrossRef]
29. Hınçal, O.; Altan-Sakarya, A.B.; Ger, A.M. Optimization of multireservoir systems by genetic algorithm. *Water Resour. Manag.* **2011**, *25*, 1465–1487. [CrossRef]
30. Yuan, X.; Yuan, Y. Application of cultural algorithm to generation scheduling of hydrothermal systems. *Energy Convers. Manag.* **2006**, *47*, 2192–2201. [CrossRef]
31. Wang, W.; Li, C.; Liao, X.; Qin, H. Study on unit commitment problem considering pumped storage and renewable energy via a novel binary artificial sheep algorithm. *Appl. Energy* **2017**, *187*, 612–626. [CrossRef]
32. Jalali, M.R.; Afshar, A.; Mariño, M.A. Multi-colony ant algorithm for continuous multi-reservoir operation optimization problem. *Water Resour. Manag.* **2007**, *21*, 1429–1447. [CrossRef]
33. Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
34. Lu, Y.; Zhou, J.; Qin, H.; Li, Y.; Zhang, Y. An adaptive hybrid differential evolution algorithm for dynamic economic dispatch with valve-point effects. *Expert Syst. Appl. Int. J.* **2010**, *37*, 4842–4849. [CrossRef]

35. Yuan, X.; Cao, B.; Yang, B.; Yuan, Y. Hydrothermal scheduling using chaotic hybrid differential evolution. *Energy Convers. Manag.* **2008**, *49*, 3627–3633. [CrossRef]
36. Brest, J.; Greiner, S.; Boskovic, B.; Mernik, M.; Zumer, V. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE Trans. Evolut. Comput.* **2006**, *10*, 646–657. [CrossRef]
37. Qin, A.K.; Huang, V.L.; Suganthan, P.N. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Trans. Evolut. Comput.* **2009**, *13*, 398–417. [CrossRef]
38. Zhang, J.; Sanderson, A.C. Jade: Adaptive differential evolution with optional external archive. *IEEE Trans. Evolut. Comput.* **2009**, *13*, 945–958. [CrossRef]
39. Wang, Y.; Cai, Z.; Zhang, Q. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Trans. Evolut. Comput.* **2011**, *15*, 55–66. [CrossRef]
40. Mallipeddi, R.; Suganthan, P.N.; Pan, Q.K.; Tasgetiren, M.F. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl. Soft Comput.* **2011**, *11*, 1679–1696. [CrossRef]
41. Tanabe, R.; Fukunaga, A. Success-History Based Parameter Adaptation for Differential Evolution. In Proceedings of the 2013 IEEE Congress on Evolutionary Computation, Cancun, Mexico, 20–23 June 2013; pp. 71–78.
42. Tanabe, R.; Fukunaga, A.S. Improving the search performance of shade using linear population size reduction. In Proceedings of the 2014 IEEE Congress on Evolutionary Computation, Beijing, China, 6–11 July 2014; pp. 1658–1665.
43. Liang, J.J.; Qu, B.Y.; Suganthan, P.N. *Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization*; Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report; Nanyang Technological University: Singapore, 2013.
44. Peng, F.; Tang, K.; Chen, G.; Yao, X. Multi-start jade with knowledge transfer for numerical optimization. In Proceedings of the Eleventh Conference on Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; pp. 1889–1895.
45. Cortina-Borja, M. *Handbook of Parametric and Nonparametric Statistical Procedures*, 5th ed.; Sheskin, J., Ed.; CRC Press: Boca Raton, FL, USA, 2012; p. 382.
46. Liao, X.; Zhou, J.; Zhang, R.; Zhang, Y. An adaptive artificial bee colony algorithm for long-term economic dispatch in cascaded hydropower systems. *Int. J. Electr. Power Energy Syst.* **2012**, *43*, 1340–1345. [CrossRef]
47. Takahama, T.; Sakai, S.; Iwane, N. Constrained optimization by the ε constrained hybrid algorithm of particle swarm optimization and genetic algorithm. In Proceedings of the Australasian Joint Conference on Artificial Intelligence, Sydney, Australia, 5–9 December 2005; pp. 389–400.