IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 12, 2020, accepted April 28, 2020, date of publication May 4, 2020, date of current version May 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2992116

Multiobjective Grey Prediction Evolution Algorithm for Environmental/Economic Dispatch Problem

ZHONGBO HU[1], ZHENG LI[1], CANYUN DAI[1], XINLIN XU[1], ZENGGANG XIONG[2], AND QINGHUA SU[1]

[1]School of Information and Mathematics, Yangtze University, Jingzhou 434023, China
[2]School of Computer and Information Science, Hubei Engineering University, Xiaogan 432000, China

Corresponding authors: Zhongbo Hu (huzbdd@126.com) and Zenggang Xiong (jkxxzg2003@163.com)

This work was supported in part by the State Key Laboratory of Biogeology and Environmental Geology, China University of Geosciences, under Grant GBL21801, in part by the Hubei Province Department of Science and Technology Soft Science Project under Grant 2018ADC068, in part by the National Nature Science Foundation of China under Grant 61972136 and Grant 61471161, and in part by the Hubei Provincial Department of Education Outstanding Youth Scientific Innovation Team Support Foundation under Grant T201410.

ABSTRACT The environmental/economic dispatch (EED) problem, as one of the most important optimization problems in power systems operations, is a highly constrained, nonlinear, multiobjective optimization problem. Multiobjective evolutionary algorithms have become effective tools for solving the EED problem. To obtain higher quality Pareto solutions for EED as well as further improve the uniformity and diversity of the Pareto set, this paper proposes a novel multiobjective evolutionary algorithm, namely multiobjective grey prediction evolution algorithm (MOGPEA). The MOGPEA first develops a novel grey prediction evolution algorithm (GPEA) based on the even grey model (EGM(1,1)). Unlike other evolutionary algorithms, the GPEA considers the population series of evolutionary algorithms as a time series and uses the EGM(1,1) model to construct an exponential function as a reproduction operator for obtaining offspring. In addition, the MOGPEA adopts two learning strategies to improve the uniformity and diversity of the Pareto optimal solutions of the EED. One is a leader-updating strategy based on the maximum distance of each solution in an external archive, and the other is a leader-guiding strategy based on one solution of each external archive. To validate the effectiveness of the MOGPEA, a standard IEEE 30-bus 6-generator test system (with/without considering losses) is studied with fuel cost and emission as two conflicting objectives to be simultaneously optimized. The experimental results are compared with those obtained using a number of algorithms reported in the literature. The results reveal that the MOGPEA generates superior Pareto optimal solutions of the multiobjective EED problem. Matlab_Codes of this article can be found in https://github.com/Zhongbo-Hu/Prediction-Evolutionary-Algorithm-HOMEPAGE.

INDEX TERMS Environmental/economic dispatch, evolutionary algorithm, EGM(1,1) model, grey prediction.

I. INTRODUCTION

The economic dispatch (ED) problem is a single objective optimization problem in power system operations [1], [2]. The purpose of the traditional ED is to meet the load demand in the most economical way. However, with increasing public awareness of environmental pollution, the clean air act has forced utilities to reduce the emission of SO_2 and NO_x [3]. In these circumstances, environmental/economic dispatch (EED) [4], [5] becomes pertinent and can not only bring great economic benefit but also reduce the pollutant emission.

The EED problem can be modeled as a multiobjective optimization problem with highly constrained and nonlinear. The fuel cost and the emission, as two conflicting objectives of the EED problem, are minimized in the conditions of satisfying the equality and inequality constraints. There have been many studies pertaining to the EED problem since it was proposed. Initially, the EED problem is generally converted into a single objective problem by considering the emission as a constraint or as a weighted function. The linear programming technique [6] is one of the representative of

The associate editor coordinating the review of this manuscript and approving it for publication was Qiu Ye Sun.

early approaches. The weighted sum method [7] is another usually used method, it transforms the objectives into a single objective problem by using a linear combination of different objectives. Although these conventional methods are easy to operate, there is a common problem that they all require multiple runs to achieve a trade-off between the two objectives.

In recent years, more and more multiobjective evolutionary algorithms (MOEAs) have been successfully used to solve the EED problem [8]. This kind of algorithms can find multiple trade-off solutions in a single run. They have thereby gradually become the main technique for solving the EED problem. According to the difference of the basic algorithms for constructing MOEAs, these multiobjective technologies can be divided into the following five categories.

The first type of MOEAs for the EED problem is based on Genetic Algorithms (GAs), e.g., the Vector Evaluated Genetic Algorithm (VEGA) [9], the Nondominated Sorting Genetic Algorithm (NSGA) [10], the NSGA-II [11], the Modified NSGA-II (MNSGA-II) [12], [13], the Niched Pareto Genetic Algorithm (NPGA) [14], the Improved Genetic Algorithm (IGA) [15] and Learner NonDominated Sorting Genetic Algorithm (NSGA-RL) [16].

The second type of MOEAs for the EED problem is based on Particle Swarm Optimization (PSO), e.g., the Multiobjective Particle Swarm Optimization (MOPSO) [17], [18], the Fuzzified Multiobjective Particle Swarm Optimization (FMOPSO) [19], the Fuzzy Clustering-based Particle Swarm Optimization (FCPSO) [20], the Multiobjective Chaotic Particle Swarm Optimization (MOCPSO) [21], the Parameter-free Bare-bones Multiobjective Particle Swarm Optimization Algorithm (BB-MOPSO) [22] and the Cultural Quantum-behaved Particle Swarm Optimization Algorithm (CMO-QPSO) [23].

The third type of MOEAs for the EED problem is based on Differential Evolution (DE), e.g., the Multiobjective Differential Evolution (MODE) [24], the Modified MODE (MMODE) [25], the Enhanced Multiobjective Differential Evolution Algorithm (EMODE) [26] and the Summation Based Multiobjective Differential Evolution Algorithm (SMODE) [27].

The fourth type of MOEAs for the EED problem is based on hybrid approaches, e.g., the New Multiobjective Stochastic Search Technique (MOSST) [28], the Hybrid MOEA based on the techniques of PSO and DE (POS-DE) [29], the Modified NSGA-II, which integrated a Convergence Accelerator Operator (CAO) into the original NSGA-II (NSGA II-CAO) [30], the combination of DE and biogeography-based optimization (BBO) algorithm (DE-BBO) [31] and the Hybrid MultiObjective Differential Evolution/Tabu Search (MODE/TS) [32].

The other types of MOEAs for the EED problem are based on other evolutionary algorithms, e.g., the Multi-objective Evolutionary Algorithm based on Decomposition (MOEA/D) [33], the Strength Pareto Evolutionary Algorithm (SPEA) [34], the Multiobjective Evolutionary Programming (MOEP) [35], the Fast Multiobjective Evolutionary Programming (FMOEP) [36], the Modified Bacterial Foraging Algorithm (MBFA) [37], the Multiobjective Bacteria Foraging Algorithm (MOBF) [38], the Multi-objective Artificial Bee Colony Algorithm (MOABC) [39], the New Multiobjective Global Best Artificial Bee Colony algorithm (MOGABC) [40], the Multiobjective Directed Bee Colony Optimization Algorithm (MODBC) [41], the Modified Shuffle Frog Leaping Algorithm (MSFLA) [42], the Multiobjective Harmony Search Algorithm (MOHS) [43], the Multiobjective Adaptive Clonal Selection Algorithm (MOACSA) [44], the Enhanced Multiobjective Cultural Algorithm (EMOCA) [45], the Multiobjective Chaotic Ant Swarm Optimization (MOCASO) [46], the Multiobjective Backtracking Search Algorithm (MOBSA) [47], the Multiobjective Scatter Search Approach (MOISS) [48], the Quasi-Oppositional Teaching Learning Based Optimization (QOTLBO) [49], the Multiobjective Collective Decision Optimization Algorithm (MOCDOA) [50], the Permutation-based Multiobjective Environmental Adaptation Method (pMOEAM) [51] and so on.

In order to obtain higher quality Paretp EED solutions as well as further improve the uniformity and diversity of the Pareto optimal set, this paper attempts to develop a novel multiobjective grey prediction evolutionary algorithm (MOGPEA) inspired by grey prediction theory.

Grey prediction theory, as an important embranchment of the grey system proposed by Deng in 1982 [52], is applicable to the problem of "incomplete information and small sample size". The grey prediction model (GM(1,1)) [53]–[56] is the core model of grey prediction theory. The accumulated generating operation (1-AGO) of the GM(1,1) can transform nonnegative discrete data sets into sequences with an approximate exponential law under specific conditions. This reduces the randomness of the original data. Then, the GM(1,1) uses the converted sequence to construct a prediction expression. If the time response sequence of the GM(1,1) is conducted by the solution of its whitenization differential equation, then the GM(1,1) is referred to as an even grey model (EGM(1,1)).

The comparison experiments on the standard IEEE 30-bus 6-generators systems show the effectiveness and superiority of the proposed MOGPEA, which can obtain higher quality, uniform and diverse EED solutions than many state-of-the-art algorithms. The major contributions of this work are as follows:

- Introduce a novel and competitive MOGPEA for the EED field. It first uses the EGM(1,1) to develop a grey prediction evolution algorithm (GPEA), and then designs two learning strategies to improve the uniformity and diversity of the Pareto optimal solutions of the EED.
- Introduce a novel evolution notion in the GPEA. Unlike other metaheuristics, the GPEA treats population sequences as a time series and then uses the EGM(1,1) model to forecast offspring (without any mutation and crossover operators).

The rest of the paper is organized as follows. Section 2 describes related preliminaries including the mathematical model for the EED problem and some background material for the EGM(1,1). Section 3 provides the description of the GPEA based on the EGM(1,1). In addition, Section 4 introduces the MOGPEA extended for the multi-objective problem. A comprehensive experimental study for the MOGPEA is carried out in Section 5. The related experimental results are provided and discussed in Section 6. Finally, Section 7 draws some conclusions and future expectations.

II. PRELIMINARY

A. FORMULATION OF THE EED PROBLEM

A classical EED problem is to simultaneously minimize competing fuel cost and emission objective functions while fulfilling certain system constraints. The components of the problem , including its objective functions and constraints , are as follows.

1) OBJECTIVE FUNCTIONS

Objective 1: Minimization of Fuel Cost: The total fuel cost $F(P)$ can be represented as follows:

$$F(P) = \sum_{i=1}^{N} (a_i + b_i P_i + c_i P_i^2) \tag{1}$$

where $F(p)$ is the total fuel cost, i.e., the sum of the power generation costs of each generator in a thermal power plant , P_i is the active output of the ith generator , N is the number of generators , and a_i, b_i, and c_i are the coal consumption characteristic coefficients of the ith generator.

Objective 2: Minimization of Emission: Let the total emission be $E(P)$ as follows:

$$E(P) = \sum_{i=1}^{N} (10^{-2}(\alpha_i + \beta_i P_i + \gamma_i P_i^2) + \eta_i \exp(\delta_i P_i) \tag{2}$$

where α_i,β_i, γ_i, η_i, and δ_i are coefficients of the ith generator emission characteristics. All the parameters are presented in Tab. 1.

2) CONSTRAINTS

Constraint 1: Power Balance Constraint: The total power generation must cover the total load and the system network loss in the transmission lines as follows.

$$\sum_{i=1}^{N} P_i - P_D - P_L = 0 \tag{3}$$

where $\sum_{i=1}^{N} P_i$ is the total power generation, P_D is the system load, P_L is the system network loss , which can be calculated as follows:

$$P_L = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B_{0i} P_i + B_{00} \tag{4}$$

where B_{ij}, B_{0i} and B_{00} are the transmission network power loss coefficients. The correlation parameters are shown in subsection 4.3.

Constraint 2: Generation Capacity Constraint: The generating capacity of the unit itself is also constrained by the upper and lower limits of the output, which can be written as follows:

$$P_i^{min} \le P_i \le P_i^{max} \tag{5}$$

where P_i^{min} and P_i^{max} are the minimum and maximum generation limit of ith generator, respectively.

3) PROBLEM FORMULATION

The EED problem is formulated as a constrained, multiobjective optimization problem and is given by the following:

$$Minimaize\ [F(P), E(P)] \tag{6}$$

$$Subjected\ to: \sum_{i=1}^{N} P_i - P_D - P_L = 0$$
$$P_i^{min} \le P_i \le P_i^{max} \tag{7}$$

B. EGM(1,1) MODEL

The even grey model (EGM(1,1)) is first proposed by professor Deng Julong, and it is the most influential and widely used grey prediction model at present [52]. The main steps of the EGM(1,1) are as follows. First, a data transformation process from a nonnegative discrete disordered data sequence into a approximate ordered sequence is implemented by the first-order accumulated generating operation (1-AGO). Second, an exponential function based on the transformed data sequence is constructed to predict the next value of the ordered sequence. Last, the prediction values of the original data are obtained by the inverse operation of the 1-AGO operator (1-IAGO).

Assume that an original data sequence $X^{(0)} = (x^{(0)}(1), x^{(0)}(2), \cdots, x^{(0)}(n))$, where $X^{(0)}(k) \geq 0, k = 1, 2, \cdots, n$.

Definition 1 (1-AGO) [52]: $X^{(1)}$ is the sequence of the 1-AGO of $X^{(0)}$:

$$X^{(1)} = (x^{(1)}(1), x^{(1)}(2), \cdots, x^{(1)}(n)),$$

where

$$x^{(1)}(k) = \sum_{i=1}^{k} x^{(0)}(i),\ k = 1, 2, \cdots, n. \tag{8}$$

Definition 2 (EGM(1,1) Model) [52]: the sequence $Z^{(1)}$ is the mean sequence of the $X^{(1)}$:

$$Z^{(1)} = (z^{(1)}(2), z^{(1)}(3), \cdots, z^{(1)}(n)),$$

where $z^{(1)}(k)$ satisfies

$$z^{(1)}(k) = \frac{1}{2}(x^{(1)}(k) + x^{(1)}(k-1)), \quad k = 1, 2, \cdots, n. \tag{9}$$

the following equation is called even grey model EGM(1,1),

$$x^{(0)}(k) + az^{(1)}(k) = b, \quad k = 2, 3, \cdots, n. \tag{10}$$

here, parameter a is called as a grey developmental coefficient, and b is called as a gray control parameter. Then the time response function of Eq.10 is solved:

$$\hat{x}^{(1)}(k+1) = (x^{(0)}(1) - \frac{b}{a})e^{-ak} + \frac{b}{a}, \quad k = 1, 2, \cdots, n. \tag{11}$$

Definition 3 (1-IAGO) [52]: The prediction value for the raw data sequence $X^{(0)}$ can be estimated by the first-order inverse accumulated generating operation the (1-IAGO).

$$\hat{x}^{(0)}(k+1) = \hat{x}^{(1)}(k+1) - \hat{x}^{(1)}(k) = (1 - e^{a}) \times (x^{(0)}(1) - \frac{b}{a})e^{-ak} \quad k = 1, 2, \cdots, n. \tag{12}$$

Here, specifically, $\hat{x}^{(0)}(1) = \hat{x}^{(1)}(1) = x^{(0)}(1)$.

III. GREY PREDICTION EVOLUTION ALGORITHM (GPEA) BASED ON EGM(1, 1)

This section introduces a grey prediction evolution algorithm based on the EGM(1,1)(GPEA) [57], [58]. The algorithm includes an initialization operator, a reproduction operator and a selection operator. The GPEA is unique in that its reproduction operator which replaces the common mutation and crossover operator with the EGM(1,1) prediction. The process of the GPEA can be described as follows.

A. INITIALIZATION OPERATOR

In the initialization phase of the GPEA, $3N$ D-dimensional individuals are initialized in the search space and each individual is expressed through $\boldsymbol{y}_i^g = (y_{i,1}^g, y_{i,2}^g, ..., y_{i,D}^g)$, $i = 1, 2, \cdots, N$ and $g = 1, 2, ..., g_{max}$, where g and g_{max} are the current generation and the maximum number of generations respectively. The j^{th} dimension of the i^{th} individual is randomly produced according to the following:

$$y_{i,j}^{(0)} = lb_j + rand(0, 1) \cdot (ub_j - lb_j), \quad j = 1, 2, \cdots, D. \tag{13}$$

Here, $rand(0, 1)$ is a random number with a uniform distribution between 0 and 1, lb_j and ub_j are the lower bound and upper bound of the j^{th} dimension, respectively. Then, we distribute $3N$ individuals into three populations on average according to the fitness value of the individuals (from small to large). In detail, the top N individuals are divided into the first generation $Y^0(g = 0)$. Simultaneously, the middle N individuals are divided into the $Y^1(g = 1)$, and the bottom N individuals are divided into the $Y^2(g = 2)$. These three populations constitute an initial population series as a time series to predict the next generation population.

B. REPRODUCTION OPERATOR

In this section, a novel reproduction operator based on the EGM(1,1) model, called the *egm*11 reproduction operator, is proposed. The *egm*11 reproduction operator fits a exponential function by using successive three generations of a population sequence to forecast offspring. In addition, considering the calculating characteristics of the EGM(1,1) model, the *egm*11 reproduction operator is supplemented by a random perturbation and a linear fitting. The *egm*11 reproduction operator is shown as follows. Y^{g-2}, Y^{g-1}, and Y^g, $(g \geq 2)$ denote three consecutive population series and three individuals $\boldsymbol{y}_{r1}$, $\boldsymbol{y}_{r2}$, and $\boldsymbol{y}_{r3}$ are randomly chosen from Y^{g-2}, Y^{g-1}, and Y^g, respectively. Set $Max_y = max\{|y_{r1,j} - y_{r2,j}|, |y_{r1,j} - y_{r3,j}|, |y_{r2,j} - y_{r3,j}|\}$, and $Min_y = min\{|y_{r1,j} - y_{r2,j}|, |y_{r1,j} - y_{r3,j}|, |y_{r2,j} - y_{r3,j}|\}$. Then the j^{th} dimension of the i^{th} individual of the trial population U^g is produced:

$$u_{i,j}^g = \begin{cases} (1 - e^{a})(y_{r1,j} - d\frac{b}{a})e^{-3a}, & \text{if } Max_y \geq th, \\ \dfrac{4y_{r3,j} + y_{r2,j} - 2y_{r1,j}}{3}, & \text{elseif } \; Min_y < th \\ y_{r3,j} + w \cdot Max_y, & \text{otherwise.} \end{cases} \tag{14}$$

here

$$\begin{cases} a = \dfrac{2(y_{r2,j} - y_{r3,j})}{y_{r2,j} + y_{r3,j}} \\ b = \dfrac{2((y_{r2,j})^2 + y_{r1,j} \cdot y_{r2,j} - y_{r1,j} \cdot y_{r3,j})}{y_{r2,j} + y_{r3,j}} \\ w = rand(-1, 1)(0.01 - \dfrac{3.99(I - M)}{M}) \end{cases} \tag{15}$$

w is able to control the disturbance range, $th \in [0.001, 0.1]$ is a preset value and used to control forecast, M is the maximum number of iterations and I is the current iteration number. Alg. 1 presents the pseudo code of the *egm*11 reproduction operator.

Algorithm 1 *egm*11 Reproduction Operator

```
Input: Y^{g-2}, Y^{g-1}, Y^g, (g ≥ 2)
Output: U^g: a trial population of Y^g
for i = 1 to N do
    Select parents;
    Parents are composed of three individuals from
    random select in Y^{g-2}, Y^{g-1}, and Y^g, respectively.
    The three individuals are assigned to series
    y_r = {y_{r1}, y_{r2}, y_{r3}}.
    for j = 1 to D do
        if Max_y ≥ th then
            u^g_{i,j} = (1 − e^a)(y_{r1,j} − b/a)e^{−3a} ; //EGM(1,1)
            prediction
        else if Min_y < th then
            u^g_{i,j} = (4y_{r3,j}+y_{r2,j}−2y_{r1,j})/3; // linear prediction
        else
            u^g_{i,j} = y_{r3,j} + w · Max_y; //random disturbance
    end
end
return U^g;
```

Algorithm 2 The Pseudo Code of MOGPEA

Input: N: size of the population, D: dimension of the population, Na: maximum capacity of the archive, T_{max}: maximum number of iterations, ux: upper limit of problem, lx: lower limit of problem
Output: Op
Initialization
Initialize X^2, X^1, X^0 according to the formula(13);
origin$X$={X^0, X^1, X^2};
$F(X^0, X^1, X^2) \leftarrow ObjFun(X^0, X^1, X^2)$;
$Ar \leftarrow Non\_dominated(X^0, X^1, X^2)$;
if $|Ar| > Na$ **then**
    Circular_crowded_sorting(Ar);
end
$t = 4$;
while $t \leq T_{max}$ **do**
    Reproduction
    $Ar \leftarrow sort(Ar)$;
    $x_r = \{\boldsymbol{x}_{r1}, \boldsymbol{x}_{r2}, \boldsymbol{x}_{r3}\}$ is randomly selected from X^{g-2}, X^{g-1} and X^g $(g \geq 2)$, respectively;
    for $i = 1$ *to* N **do**
        $X_{Li} \leftarrow$ ***Leader_updating***(Ar) ;
        for $j = 1$ *to* D **do**
            $d_{12} = \left|x_{r1,j} - x_{r2,j}\right|$, $d_{13} = \left|x_{r1,j} - x_{r3,j}\right|$, $d_{23} = \left|x_{r2,j} - x_{r3,j}\right|$, and $Md_r = max\{d_{12}, d_{23}, d_{13}\}$;
            if $Md_r \geq d$ **then**
                $T(i,j) = (1 - e^a) \cdot (x_{r1,j} - \frac{b}{a}) \cdot e^{-3a}$;
            else
                $T(i,j) \leftarrow$ ***Leader_guiding***$(X_L(i,j))$;
            end
        end
    end
    Selection
    $F(T) = ObjFun(T)$;
    $X \leftarrow$ ***Dominance_relation_selection***$(originX\{1, 3\}, T)$;
    $Ar \leftarrow Non\_dominated(X \bigcup Ar)$;
    if $|Ar| > Na$ **then**
        Circular_crowded_sorting(Ar);
    end
    $originX(1, 4) = \{X\}$;
    $originX = origin(2:4)$;
    $t = t + 1$;
end
$Op \leftarrow Ar$ and stop the algorithm ;
return Op;

C. SELECTION OPERATOR

In order to select the better individual into the next generation, the GPEA carries out selection operation between the trial individual $\boldsymbol{u}_i^g$ and target individuals $\boldsymbol{y}_i^g$. The individuals with a better fitness is selected to survive. This operation is described by the following expression.

$$\boldsymbol{y}_i^{g+1} = \begin{cases} \boldsymbol{u}_i^g, & if\ f(\boldsymbol{u}_i^g) < f(\boldsymbol{y}_i^g) \\ \boldsymbol{y}_i^g, & otherwise \end{cases} \tag{16}$$

IV. MULTIOBJECTIVE GREY PREDICTION EVOLUTION ALGORITHM (MOGPEA)

Based on the GPEA, this section develops a multiobjective grey prediction evolution algorithm (MOGPEA) for solving the EED problem. First, in order to improve the uniformity and diversity of the Pareto optimal solutions of the EED problem, a leader-updating strategy [50] and a leader-guiding strategy with learning characteristics are introduced to the MOGPEA. Second, the MOGPEA embeds an external archive strategy and a selection strategy based on dominance relation. This algorithm works as follows:

Step 1 Initialization
Step 1.1: Randomly initialize three generation population to form the initial populations chain.
Step 1.2: Evaluate the fitness of each generation population.
Step 1.3: Store all the nondominated solutions to external archive set.
Step 1.4: Maintain external archive by removing redundant solutions.

Step 2 Reproduction
Step 2.1:Find the global best of each individual by leader updating strategy.
Step 2.2: Generate new individual by using the EGM(1,1) prediction or leader guiding strategy.
Step 2.3: Evaluate the fitness of new population.

Step 3 Selection
Step 3.1: Select next generation using dominance relation.
Step 3.2: Store new nondominated solutions to external archive set.
Step 3.3: Maintain external archive by removing redundant solutions.
Step 3.4: The new population and the previous two generations form a new population chain.

Step 4 Stoping criteria: If the stop criteria is met, then stop. Otherwise, go back to Step 2.

Here are the four strategies described above, and Alg . 2 represents the pseudo code of the MOGPEA.

1) LEADER-UPDATING STRATEGY

To improve the uniformity and diversity of the nondominated solutions in the Pareto front, the MOGPEA first uses the maximum distance to measure the sparsity of the nondominated solutions, and then designs a leader-updating strategy based on the maximum distance. As shown in Fig. 1, the maximum distance of the ith point (nondominated solution) is calculated:

$$d_i = \max(ld_i, ud_i)\ i = 1, 2, \cdots, Ne. \tag{17}$$

$$ld_i = \sum_{k=1}^{N} \left|\frac{f_k(X_i) - f_k(X_{i+1})}{f_k^{\max} - f_k^{\min}}\right| \tag{18}$$

$$ud_i = \sum_{k=1}^{N} \left|\frac{f_k(X_i) - f_k(X_{i-1})}{f_k^{\max} - f_k^{\min}}\right| \tag{19}$$

here ud_i and ld_i are the distances from the ith point to the previous point and the next point. Ne is the maximum capacity of the archive; N is the number of objective functions; $f_j^{\max}$ and $f_j^{\min}$ are the maximum and minimum values of the kth objective function, respectively.

After calculating the maximum distance of all non-dominant solutions, MOGPEA adopts the roulette wheel selection mechanism to select the leader of the archive.

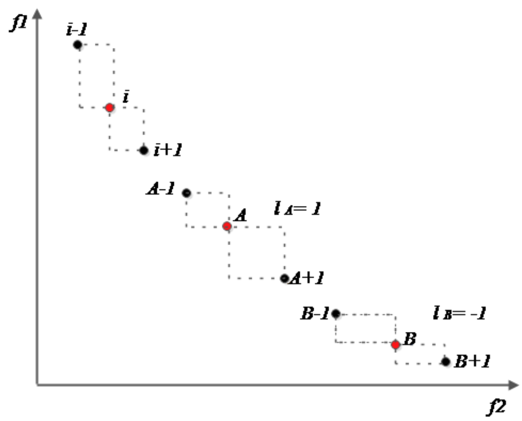

FIGURE 1. Calculation of maximum distance and sparse direction.

In other words, the i solution can be the leader as long as its maximum distance is the greatest. Alg. 3. describes the leader-updating strategy.

2) LEADER-GUIDING STRATEGY

This section firstly introduces the sparse direction in the leader-guiding strategy. The sparse direction of the i solution is defined as follows:

$$l_i = \begin{cases} 1, & ld_i > ud_i \\ -1, & ld_i \le ud_i \end{cases} \tag{20}$$

such as in Fig. 1, since ld_A is greater than ud_A, so $l_A = 1$. While $l_B = -1$ for the Bth solution. Next, the leader-guiding strategy based on the sparse direction is used to update the individual. The update formula is as follows

$$T_i = X_{Li} + rand \cdot (Ar_{index_i + l_i} - X_{Li}) \tag{21}$$

The above formula indicates that X_L moves towards the sparse direction to generate a new individual. This can increase the uniformity of the Pareto front.

Algorithm 3 Leader-Updating

```
Input: Ar(t)
Output: X_L
Nt = |Ar(t)| ;
for i = 1 to Nt do
    calculate d_i ;
end
for i = 1 to N do
    index_i = Roulette_wheel_selection(d_i);
    X_Li(t) = Ar_index_i(t);
end
return X_L;
```

3) SELECTION STRATEGY BASED ON DOMINANCE RELATION

As we all know, greedy selection is a common strategy for single objective optimization problems. However, in multiobjective problem, the selection strategy based on dominance relation is used to select the promising solutions into the next generation. This kind of selection strategy can improve the global search capability further. In the strategy, the trial individual T_i can enter the next generation when T_i dominates the target individual P_i ($T_i \prec P_i$). When two individuals do not dominate each other, there is a half chance that each individual will go on to the next generation. The selection strategy based on dominance relation can be described as follows.

$$P_i = \begin{cases} T_i, & T_i \prec P_i \\ P_i, & P_i \prec T_i \\ T_i \ or \ P_i, & T_i \nprec P_i \wedge P_i \nprec T_i \end{cases} \quad (22)$$

4) EXTERNAL ARCHIVE MAINTENANCE STRATEGY

At present, there are many external archive maintenance strategies. The most famous one is the fast non-dominated sort in NSGA-II. However, this method may delete several connected solutions with smaller crowding distances through calculating the crowding distance. This can lead to the remaining solutions too sparse. To avoid the above problems, in this paper, a cyclic crowded sorting algorithm [59] is used to pick out individuals. The pseudo code of the cyclic crowded sorting is shown in Alg. 4.

Algorithm 4 Cycled Crowding Sorting

```
Input: Ar(t), Na
Output: Ar(t + 1)
Nt = |Ar(t)|;
while Nt > Na do
    for i = 1 to Nt do
        Ar_i(t).distance = 0;
    end
    for m = 1 to M do
        Ar(t) = sort(Ar(t), m);
        Ar_1(t).distance = Inf;
        Ar_Nt(t).distance = Inf;
        for i=2 to Nt-1 do
            Ar_i(t).distance = Ar_i(t).distance +
              (Ar_{i+1}(t).distance - Ar_{i-1}(t).distance) / (Ar_Nt(t).distance - Ar_1(t).distance);
        end
    end
    k = min_Ar(t).distance;
    Ar_k(t) = [ ];
    Nt = Nt - 1;
end
return Ar(t + 1) ← Ar(t);
```

5) COMPLEXITY ANALYSIS OF THE MOGPEA

This section analyzes the complexity of MOGPEA. The introduction of two learning strategy and cyclic crowded sorting method of the MOGPEA will consume storage space and increase the time complexity. The complexity of the leader updating strategy is $O(N)$. The complexity of the leader guiding strategy is $O(N)$. The computational complexity of cyclic crowded sorting is $O(Na)$, where Na is the current capacity of the archive and $Na > N$. Therefore, the final time complexity of the MOGPEA in one generation is $O(Na)$. In addition, the space complexity of cyclic crowded sorting mechanism is $O(Na)$. Overall, the consumption of the time and space is very small.

V. THE MOGPEA IMPLEMENTATION

In this section, the MOGPEA first introduces a constraint-handling strategy to deal with the equality and inequality constraints of the EED problem. Second, a fuzzy set mechanism is used to extract the best compromise solution from the final external archive. Last, the design of experiments and the setting of parameters are described.

A. CONSTRAINT HANDLING

In addition to high-dimensional and multiple objectives, high constraints are another difficult problem to deal with for the EED problem. For the inequality constraint of the EED problem, it is very easy to deal with the over-limited values by simply setting it to the corresponding boundary value. On the contrary, for the equality constraints, it becomes very complicated since the strong coupling between variables. In order to better solve the equality constraints and avoid consuming too much time, this paper employs a special constraint handling method to deal with the power balance constraint of the EED problem. A constraint violation threshold σ is set in advance and $\sigma = 1e - 12$. The constraint process is as follows:

Step 1: For each infeasible solution $\mathbf{x}$, set k is a random integer from 1 to D.
Step 2: Calculate the violation $V(x)$:

$$V(x) = P_L + P_D - sum(x_i) \quad (23)$$

If $V(x) > \sigma$, then go to Step 3; otherwise, go to Step 4.
Step 3: Adjust $\mathbf{x}$ to make it satisfy the constraint:

$$x_{i,k} = x_{i,k} * (P_L + P_D)/sum(x_i), \quad (i = 1, 2, \cdots, N) \quad (24)$$

If the new $x_{i,k}$ violates the inequality constraint, and then it will be addressed by the inequality constraint method. Let $k = mod(k, D) + 1$, and go to Step 2.
Step 4: End the constraint handling process.

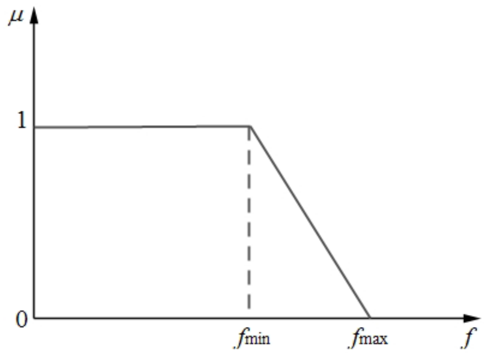

FIGURE 2. The fuzzy-based membership function.

B. FUZZY-BASED THEORY FOR THE COMPROMISE SOLUTION

In practical applications, the decision maker only usually needs one solution in the final external archive. This solution is called the best compromise solution, and to some extent it satisfies all objectives. This paper extracts the best compromise solution by using a fuzzy-based mechanism [34] and Fig. 2 illustrates a fuzzy-based membership function. The membership function μ_{ik} of the kth objective of the ith solution is calculated in the following way:

$$\mu_{ik} = \begin{cases} 1, & if\ f_{i,k} \le f_k^{\min} \\ \dfrac{f_k^{\max} - f_{i,k}}{f_k^{\max} - f_k^{\min}}, & if\ f_k^{\min} \le f_{i,k} \le f_k^{\max} \\ 0, & if\ f_{i,k} \ge f_k^{\max} \end{cases} \tag{25}$$

$f_k^{\max}$ and $f_k^{\min}$ are the maximum and minimum values of the kth objective function among all nondominated solutions of archive, respectively. The normalized membership function μ_i is calculated:

$$\mu_i = \frac{\sum_{k=1}^{M} \mu_{i,k}}{\sum_{j=1}^{Nt} \sum_{k=1}^{M} \mu_{jk}} \tag{26}$$

Here, M denotes the number of objective functions ($M = 2$ in this paper), and Nt is the number of nondominated solutions. The best compromise solution is the solution for which μ_k is the largest.

C. EXPERIMENTAL DESIGN AND PARAMETER SELECTION

The MOGPEA is tested on the standard IEEE 30-bus 6-generator system (as shown in Fig.3). The fuel cost coefficient, emission coefficient, and generation limit are referenced in [22] and given in Tab. 2. The transmission loss coefficients are given in Tab. 1. The load demand is 2.834 MW. The simulation program is written in MATLAB and run at 1.6GHz Intel Pentium core i7 processor with 4GB-RAM. The source codes of this algorithm can be found in https://github.com/Zhongbo-Hu/Prediction-Evolutionary-Algorithm-HOMEPAGE.

In order to investigate the effectiveness of the MOGPEA for solving the EED problems, two different cases are studied as follows:

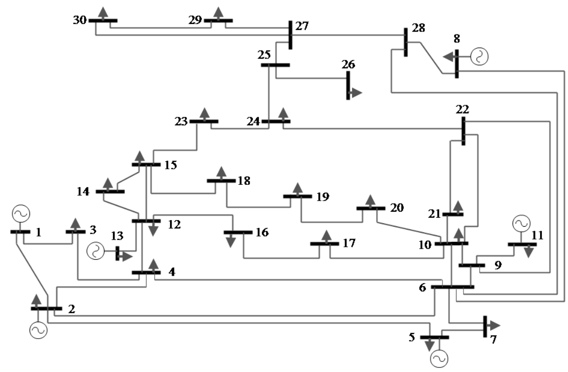

FIGURE 3. IEEE 30-bus 6 generator test system.

TABLE 1. Transmission loss coefficients.

	0.1382	-0.0299	0.0044	-0.0022	-0.0010	-0.0008
	-0.0299	0.0487	-0.0025	0.0004	0.0016	0.0041
$B =$	0.0044	-0.0025	0.0182	-0.0070	-0.0066	-0.0066
	-0.0022	0.0004	-0.0070	0.0137	0.0050	0.0033
	-0.0010	0.0016	-0.0066	0.0050	0.0109	0.0005
	-0.0008	0.0041	-0.0066	0.0033	0.0005	0.0244
$B_0 =$	-0.0107	0.0060	-0.0017	0.0009	0.0002	0.0030
$B_{00} =$	0.00098573					

TABLE 2. Generator cost and emission coefficients.

		G_1	G_2	G_3	G_4	G_5	G_6
	a	10	10	20	10	20	10
Cost	b	200	150	180	100	180	150
	c	100	120	40	60	40	100
	α	4.091	2.543	4.258	5.326	4.258	6.131
	β	-5.554	-6.047	-5.094	-3.550	-5.094	-5.555
Emission	γ	6.490	5.638	4.586	3.380	4.586	5.151
	ζ	2.0E-4	5.0E-4	1.0E-6	2.0E-3	1.0E-6	1.0E-5
	λ	2.857	3.333	8.000	2.000	8.000	6.667
P_G^{min}		0.05	0.05	0.05	0.05	0.05	0.05
P_G^{max}		0.5	0.6	1	1.2	1	0.6

- **Case 1:** the transmission loss of power balance constraint is not considered.
- **Case 2:** the transmission loss of power balance constraint is considered.

Here, the number of population $N = 50$ and the capacity of the archive $Na = 50$. Stopping criterion for two case are taken as 100 and 200 maximum number of iterations, respectively. Thirty independent runs of the MOGPEA are carried out to collect the statistical results. In addition, the parameters of several compared algorithms are given below.

MOPSO [17]: the inertia weight is 0.7, the personal learning coefficient is 1.4, the global learning coefficient is 1.4, the number of grids per dimension is 7.

NSGA-II [11]: the crossover percentage is 0.7 and the mutation percentage is 0.7.

PESA-II [60] : the crossover percentage is 0.7 and the mutation percentage is 0.7.

VI. RESULTS AND DISCUSSIONS

First, a simple multi-objective unconstrained test function is used to verify the effectiveness of the two learning strategies.

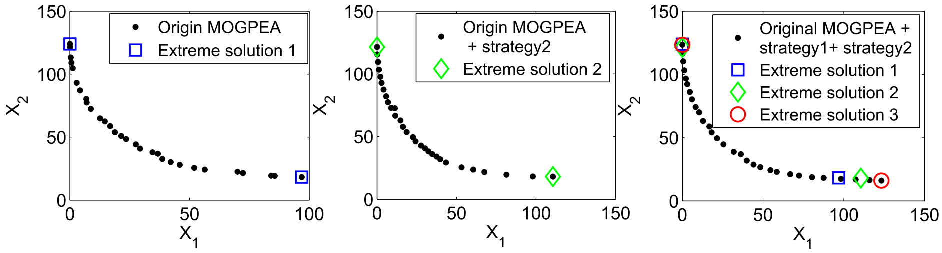

FIGURE 4. Comparisons of original MOGPEA and adding strategies MOGPEA.

Second, experimental result is obtained on the IEEE 30-bus 6-generator test system.

A. EFFECTIVENESS ANALYSIS OF TWO LEARNING STRATEGIES

A unconstrained multiobjective function is firstly used to verify the effectiveness of the two learning strategies. The function is described as follows:

$$\begin{cases} \min f_1 = 4x_1^2 + 4x_2^2 \\ \min f_2 = (x_1 - 5)^2 + 4(x_2 - 5)^2 \\ st.\ 0 < x_1 < 5,\ 0 < x_2 < 3 \end{cases} \tag{27}$$

Next, the original MOGPEA (MOGPEA1), the original MOGPEA + strategy 2 (MOGPEA2) and the original MOGPEA + strategy 1 + strategy 2 (MOGPEA3) are verified on this test function. Here, $N = 40$, $D = 2$, $Na = 35$, and $T_{max} = 30$. Fig. 4 shows the Pareto fronts obtained by the above three algorithms. From the figure, we can infer two conclusions:

- From Fig. 4, the uniformity of the Pareto front of MOGPEA3 is better than that of MOGPEA2, which is better than that of MOGPEA1. Therefore, we can obtain conclusion 1: the strategy 1 and strategy 2 can improve the uniformity of the Pareto front.
- From the third graph in Fig. 4, the coverage of the extreme solutions of the Pareto front for the MOGPEA3 marked by two red circles is more widespread than that of the MOGPEA2 marked by two green diamonds. It is more widespread than that of the MOGPEA1 marked by two blue squares. From the above, we can obtain conclusion 2: the strategy 1 and strategy 2 can increase the diversity of the Pareto front.

B. IEEE 30-BUS 6-GENERATOR TEST SYSTEM

This section conducts two experiments in the IEEE 30-bus 6-generator test system to evaluate the performance of the MOGPEA. One experiment is to compare the extreme solutions and compromise solutions of the MOGPEA with other famous algorithms. Another is to use some evaluation indicator to test solution quality, such as SP, HV and CM.

1) COMPARISON OF EXTREME SOLUTIONS AND COMPROMISE SOLUTIONS

First, the original GPEA is carried out to search for the extreme solutions of the two objective of the EED problem in two cases respectively. Tab. 3 shows the obtained best extreme solution for two cases, and Fig. 5 and 6 give the convergence of two objectives for two cases. As can be observed from Tab. 3, the optimal values of fuel cost objective for Case 1 and Case 2 are 600.111417 \$/h and 605.998378 \$/h and the optimal values of the emission objective are 0.194203 ton/h and 0.194179 ton/h for Case 1 and Case 2.

TABLE 3. Best cost and emission optimized individually.

	Case1		Case2	
	Best cost	Best emission	Best cost	Best emission
G_1	0.1095	0.4060	0.1211	0.4109
G_2	0.2996	0.4589	0.2862	0.4636
G_3	0.5242	0.5379	0.5833	0.5443
G_4	1.0162	0.3830	0.9928	0.3903
G_5	0.5244	0.5378	0.5241	0.5444
G_6	0.3598	0.5101	0.3517	0.5155
Fuel cost (\$/h)	**600.11**	638.26	**605.99**	646.20
Emission (ton/h)	0.2221	**0.1942**	0.2207	**0.1941**

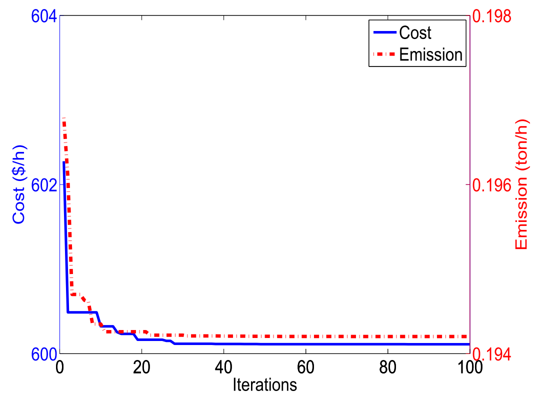

FIGURE 5. Convergence of cost and emission objective on Case1.

Next, the proposed MOGPEA is implemented to simultaneously optimize both objectives of the EED problem, and the results of extreme solutions and compromise solutions for two cases are discussed below.

Case 1: Applying the MOGPEA to Case 1, the Pareto front is displayed in Fig. 7. The figure clearly indicates that the solutions are well-distributed and almost cover the entire Pareto front of the problem. Tab. 4 and 5 compare the best extreme solutions of the MOGPEA for fuel cost and emission with results reported in the literatures that were obtained by

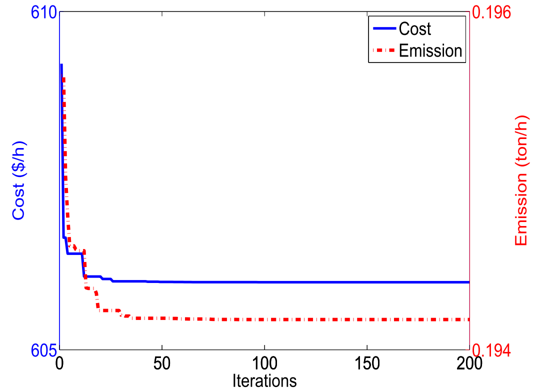

FIGURE 6. Convergence of cost and emission objective on Case2.

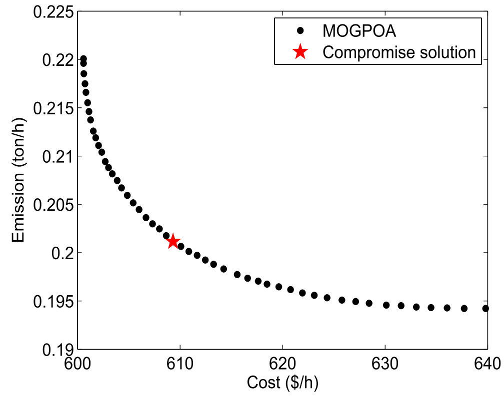

FIGURE 7. Pareto front using MOGPEA on Case1.

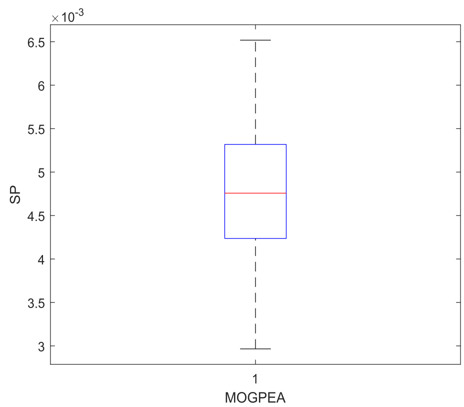

FIGURE 8. The box plot of the SP value on Case2.

using NSGA [10], NPGA [14], SPEA [34], MOPSO [17], BB-MOPSO [22], FMOEP [36], MBFA [37], NSGA-II [11], MOACSA [44] and SMODE [27]. In addition, to compare the compromise solutions of different algorithms, the average satisfactory degree (ASD) [22] is calculated. Tab. 6 compares the results obtained by the MOGPEA for the compromise solution and the ASD with results reported in the literatures that were obtained by using NSGA [10], NPGA [14], SPEA [34], FCPSO [20], BB-MOPSO [22] and MOCDOA [50]. The bold values in Tab. 4-6 are the best obtained results.

As seen from Tab. 4 and 5, 600.11 \$/h and 0.1942 ton/h are the two best extreme values for the fuel cost and emission, respectively. In Tab. 4, although the MOGPEA generates the same best cost value as BB-MOPSO, NSGA-II, MOACSA and SMODE, the obtained corresponding emission by the MOGPEA is 0.2219 ton/h, which is the best among the four compared algorithms. In addition, the MOGPEA is better than the other six comparison algorithms in terms of the best cost (including NSGA, NPGA, SPEA, MOPSO, FMOEP and MBFA). In Tab. 5, it is clear that the proposed MOGPEA gives minimum emission of 0.1942 ton/h, which is equal to the emission obtained from the SPEA, MOPSO, BB-MOPSO, FMOEP, MBFA, NSGA-II, MOACSA and SMODE algorithms, and it performs better than NSGA and NPGA for the problem. In Tab. 6, according to the ASD, the MOGPEA attains the best ASD value among seven algorithms (equal to 0.7677).In other words, the proposed MOGPEA gives the best compromise solution of 609.54 \$/h and 0.2009 ton/h, which are significantly better than the solutions given by the other algorithms. All of these results demonstrate the potential and effectiveness of the MOGEPA to solve the EED problem.

Case 2: The best cost and best emission of Case 2 obtained by the MOGPEA are given in Tab. 7 and 8, respectively. The results are compared to those reported in the literature including NSGA, NPGA, SPEA, MOPSO, MODE, FMOEP, MBFA, NSGA-II, MOACSA and SMODE. From Tab. 7 and 8, it is easy to observe that the MOGPEA obtains the best fuel cost (equal to 606.004453\$/h) and the best emission (equal to 0.194181 ton/h) compared to all the other algorithms. The above results obtained by the MOGPEA are very competitive among all comparison algorithms, which demonstrates the effectiveness of the MOGPEA in solving EED problems.

2) COMPARISON OF SOLUTION QUALITY

Three commonly used multiobjective performance metrics are used to evaluate the quality of the solution obtained by MOGEPA. In addition, the solution quality of the MOGPEA is compared with that of MOPSO [17], NSGA-II [11] and PESA-II [60]. This section only considers the solutions for Case 2.

(1) Spacing (SP): The spacing (SP) [61], proposed by Schott, is adopted to evaluate the uniformity of the Pareto optimal set found. The calculation of the SP is as follows:

$$SP = \sqrt{\frac{1}{|Ar|-1}\sum_{i=1}^{|Ar|}(\bar{d}-d_i)^2},$$
$$d_i = \min_{q_j \in Ar \wedge q_j \neq q_i} \sum_{k=1}^{m} |f_k(q_i) - f_k(q_j)| \tag{28}$$

where d_i is the Euclidean distance between two consecutive solutions in the nondominated solution set and $\bar{d}$ is the mean values of all d_i. The SP values are closely related to the uniformity of the solution set, that is, the smaller the value, the more

TABLE 4. Best cost of ten algorithms on Case 1.

	G_1	G_2	G_3	G_4	G_5	G_6	Fuel cost	Emission
MOGPEA	0.1110	0.3025	0.5233	1.0155	0.5194	0.3621	**600.11**	0.2219
NSGA	0.1038	0.3228	0.5123	1.0387	0.5324	0.3241	600.34	0.2241
NPGA	0.1116	0.3143	0.5419	1.0415	0.4726	0.3512	600.31	0.2238
SPEA	0.1009	0.3186	0.5400	0.9903	0.5336	0.3507	600.22	0.2206
MOPSO	0.1183	0.3019	0.5224	1.0116	0.5254	0.3544	600.12	0.2216
BB-MOPSO	0.1090	0.3005	0.5234	1.0170	0.5238	0.3603	**600.11**	0.2222
FMOEP	0.0872	0.2868	0.5488	1.0114	0.5477	0.3521	600.24	0.2232
MBFA	0.1133	0.3005	0.5202	0.9882	0.5409	0.3709	600.17	0.2200
NSGA-II	0.1094	0.2994	0.5236	1.0167	0.5244	0.3605	**600.11**	0.2222
MOACSA	0.1090	0.2989	0.5262	1.0183	0.5227	0.3589	**600.11**	0.2223
SMODE	0.1077	0.2990	0.5269	1.0128	0.5269	1.0128	**600.11**	0.2221

TABLE 5. Best emission of ten algorithms on Case 1.

	G_1	G_2	G_3	G_4	G_5	G_6	Fuel cost	Emission
MOGPEA	0.4069	0.4613	0.5353	0.3813	0.5381	0.5108	638.55	**0.1942**
NSGA	0.4072	0.4536	0.4888	0.4302	0.5836	0.4707	633.83	0.1946
NPGA	0.4146	0.4419	0.5411	0.4067	0.5318	0.4979	636.04	0.1943
SPEA	0.4240	0.4577	0.5301	0.3721	0.5311	0.5190	640.42	**0.1942**
MOPSO	0.4015	0.4590	0.5332	0.3891	0.5456	0.5057	637.42	**0.1942**
BB-MOPSO	0.4071	0.4591	0.5374	0.3838	0.5369	0.5098	638.26	**0.1942**
FMOEP	0.3926	0.4570	0.5549	0.3799	0.5434	0.5061	638.97	**0.1942**
MBFA	0.3943	0.4627	0.5423	0.3946	0.5346	0.5056	636.73	**0.1942**
NSGA-II	0.4059	0.4586	0.5382	0.3832	0.5385	0.5097	638.22	**0.1942**
MOACSA	0.4062	0.4577	0.5373	0.3821	0.5404	0.5105	638.30	**0.1942**
SMODE	0.4002	0.4531	0.5430	0.4019	0.5361	0.4997	635.99	**0.1942**

TABLE 6. Best compromise solution of seven algorithms on Case 1.

	G_1	G_2	G_3	G_4	G_5	G_6	Fuel cost	Emission	ASD
MOGPEA	0.2540	0.3676	0.5444	0.6948	0.5367	0.4362	609.54	0.2009	**0.7677**
BB-MOPSO	0.2595	0.3698	0.5351	0.6919	0.5500	0.4277	609.75	0.2008	0.7555
NSGA	0.2571	0.3774	0.5381	0.6872	0.5404	0.4337	610.07	0.2006	0.7551
NPGA	0.2696	0.3673	0.5594	0.6496	0.5396	0.4486	612.13	0.1994	0.7491
SPEA	0.2785	0.3764	0.5300	0.6931	0.5406	0.4153	610.25	0.2005	0.7527
FCPSO	0.3193	0.3934	0.5359	0.5921	0.5457	0.4470	620.00	0.1971	0.7267
MOCDOA	0.2699	0.3721	0.5291	0.6997	0.5468	0.4162	609.66	0.2009	0.7594

uniform it is. When SP value is equal to 0, the obtained nondominated solution is equidistant. Fig. 8 shows the SP value of the MOGPEA for 30 runs on Case 2. The obtained The SP values of four algorithms are compared in Tab. 9. It can be observed that, compared with the MOPSO, NSGA-II and PESA-II, the average performance of the MOGPEA is the best. To more clearly and intuitively demonstrate the uniformity of the obtained nondominated solution, Fig. 9 shows the Pareto front of the MOGPEA and the other three algorithms. From Fig. 9, the uniformity of the MOGPEA is clearly superior to that of MOPSO, NSGA-II and PESA-II. All in all, the uniformity of the obtained solutions by the MOGPEA is very competitive compared with other three algorithms.

(2) Hypervolume (HV): The hypervolume (HV) [25], proposed by Zitler and Thiele, is a comprehensive performance index that can be used to evaluate both convergence and diversity. The larger the HV value, the better the comprehensive performance of the algorithm. The definition of HV is as follows:

$$HV = \bigcup_{i=1}^{|Ar|} v_i \tag{29}$$

Here, v_i is the hypervolume of the reference point and the ith solution in the solution set. The reference point is same for the four algorithms. Fig. 10 shows the HV value of the MOGPEA for 30 runs on Case 2. Tab. 10 shows the comparison results among four different algorithms in terms of HV. From this table, we can see that the HV value of MOGPEA is the largest. In other words, the comprehensive performance of the MOGPEA is superior to those of the MOPSO, NSGA-II and PESA-II.

(3) C-metric (CM): When the true Pareto front of multi-objective problem is not known, CM [62] is often used to evaluate the quality of the obtained solutions. Let $S_1, S_2 \subseteq S$ be the two solution sets obtained by two different algorithms. The CM is defined by the following formula.

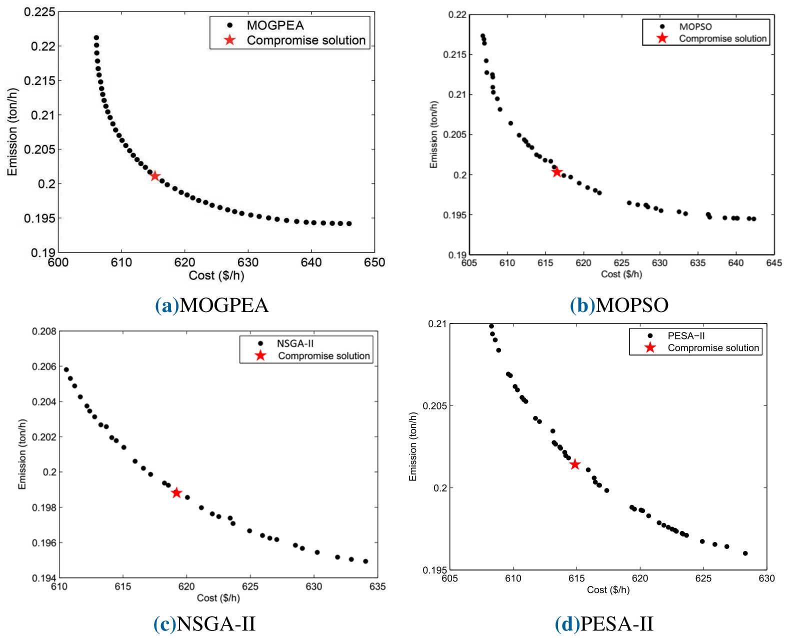

FIGURE 9. Pareto fronts and compromise solution for the four algorithms on Case 2.

TABLE 7. Best cost of ten algorithms on Case 2.

	G_1	G_2	G_3	G_4	G_5	G_6	Fuel cost	Emission
MOGPEA	0.1176	0.2833	0.5877	0.9902	0.5307	0.3496	**606.00**	0.2208
NSGA	0.1358	0.3151	0.8418	1.0431	0.0631	0.4664	620.87	0.2368
NPGA	0.1127	0.3747	0.8057	0.9031	0.1347	0.5331	620.46	0.2243
SPEA	0.1319	0.3654	0.7791	0.9282	0.1308	0.5292	619.60	0.2244
MOPSO	0.1524	0.3427	0.7857	1.0180	1.0995	0.4669	618.54	0.2308
MODE	0.1361	0.3455	0.7573	0.6016	0.5998	0.4162	618.46	0.2051
FMOEP	0.1866	0.3531	0.7587	0.5982	0.5400	0.4214	619.44	0.2031
MBFA	0.1175	0.3617	0.7899	0.9591	0.1457	0.4916	618.06	0.2264
NSGA-II	0.1619	0.3629	0.6068	0.6059	0.7155	0.4055	618.35	0.2034
MOACSA	0.1619	0.3491	0.6047	0.6059	0.7144	0.4149	618.34	0.2032
SMODE	0.1730	0.3564	0.7404	0.5946	0.5914	0.4023	619.07	0.2034

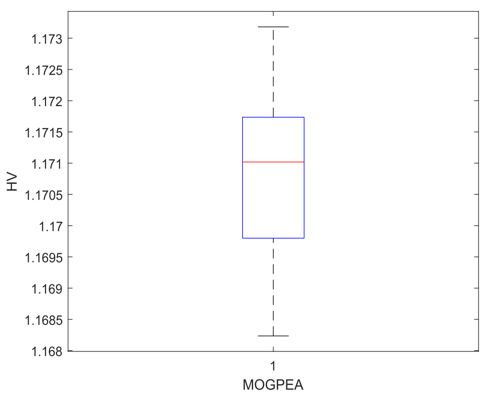

FIGURE 10. The box plot of the HV value on Case2.

$$CM(S_1, S_2) = \frac{|\{a_2 \in S_2, \exists a_1 \in S_1 : a_1 \prec a_2\}|}{|S_2|} \tag{30}$$

$CM(S_1, S_2)$=0 means that none of the solutions in S_2 are dominated by the solutions in S_1. $CM(S_1, S_2)$=1 means that all of the solutions of S_1 dominate or are equal to some solutions of S_2. Tab. 11 shows the comparison results among the four algorithms in terms of CM. From Tab. 11, nearly 12% and 10% of the solutions obtained by the NSGA-II and PESA-II respectively, are dominated by those of the MOGPEA. However, 96%, 22% and 6% of the solutions obtained by the MOGPEA are dominated by those of the MOPSO, NSGA-II and PESA-II, respectively. Thus, the coverage of the MOGPEA is better than that of the PESA-II. However, it not as good as that of the MOPSO and NSGA-II.

According to the above analysis, it can be concluded that the MOGPEA has a better performance for Case 2 in terms of the uniformity and diversity of the solutions. However, it may not be very competitive for the convergence of solutions. This result can be attributed to the collective efforts of the two learning strategies proposed. On the one hand, the update strategy of the leader based on the maximum distance is able to assign the leader to the location with sparse solutions, while the leader-guiding strategy proposed is able to make the leader move in the direction of its sparse neighborhoods. Therefore, their collective efforts ensure the uniformity and

TABLE 8. Best emission of ten algorithms on Case 2.

	G_1	G_2	G_3	G_4	G_5	G_6	Fuel cost	Emission
MOGPEA	0.4114	0.4660	0.5425	0.3955	0.5401	0.5137	645.89	**0.1941**
NSGA	0.4403	0.4940	0.7509	0.5060	0.1375	0.5364	649.24	0.2048
NPGA	0.4753	0.5162	0.6513	0.4363	0.1896	0.5988	657.59	0.2017
SPEA	0.4419	0.4598	0.6944	0.4616	0.1952	0.6131	651.71	0.2019
MOPSO	0.4589	0.5121	0.6524	0.4331	0.1981	0.6129	656.87	0.2014
MODE	0.4184	0.4622	0.5441	0.3793	0.5520	0.5068	645.74	0.1942
FMOEP	0.3980	0.4778	0.5628	0.3795	0.5403	0.5049	645.24	0.1942
MBFA	0.4716	0.5127	0.6189	0.5032	0.1788	0.5822	651.93	0.2019
NSGA-II	0.4103	0.4637	0.5459	0.3881	0.5425	0.5146	645.39	0.1942
MOACSA	0.4090	0.4624	0.5412	0.3933	0.5445	0.5146	644.84	0.1942
SMODE	0.3983	0.4601	0.5423	0.4045	0.5448	0.5139	643.01	0.1942

TABLE 9. System statistical results of the SP for Case2.

	Best	Worst	Median	Average	Std
MOGPEA	**0.003058**	**0.006271**	**0.004520**	**0.004496**	**0.000794**
MOPSO	0.015180	0.024172	0.018031	0.018250	0.001839
NSGA-II	0.016372	0.024364	0.020987	0.020640	0.002010
PESA-II	0.016314	0.079100	0.022854	0.025968	0.012981

TABLE 10. Statistical results of the HV for Case2.

	Best	Worst	Median	Average	Std
MOGPEA	**1.166006**	**1.173780**	**1.171789**	**1.171715**	**0.001655**
MOPSO	1.152740	1.165447	1.161064	1.160788	0.002821
NSGA-II	1.101082	1.163402	1.150145	1.146295	0.013986
PESA-II	1.117663	1.163896	1.149626	1.147648	0.010905

TABLE 11. Statistical results of the CM for Case2.

	MOGPEA	MOPSO	NSGA-II	PESA-II
CM(MOGPEA,*)	-	**0.00**	**0.12**	**0.10**
CM(MOPSO,*)	**0.96**	-	1.00	1.00
CM(NSGA-II,*)	**0.22**	0.00	-	0.20
CM(PESA-II,*)	**0.06**	0.00	0.40	-

diversity of solutions in the external archive set. On the other hand, the high exploration capability of the proposed reproduction operator makes the convergence rate of our algorithm slow. All in all, the MOGPEA has good uniformity and diversity . However, there is still room to improve its convergence.

VII. CONCLUSION

This paper first proposes a grey prediction evolution algorithm (GPEA) by resorting to the even grey model (EGM(1,1)). The GPEA differs from most of metaheuristics in that its reproduction operator does not make use of any mutation and crossover operators but rather considers the consecutive three population series of evolutionary algorithms as time series and uses the EGM(1,1) model to construct an exponential function for obtaining offspring. To solve the environmental/economic dispatch (EED) problem , which is a constrained multiobjective optimization problem with conflicting fuel cost and emission objectives, a multiobjective grey prediction evolution algorithm (MOGPEA) is developed in which two learning strategies are introduced for improving the uniformity and diversity of the obtained Pareto optimal solutions. One is a leader-updating strategy based on the maximum distance to measure the degree of sparseness of the solutions, and the other is a leader-guiding strategy based on the sparse mark to search the area around a leader. Furthermore, the constraints of the EED problem are solved using a special function processing strategy, a selection strategy based on the dominance relation replaces greedy selection of the GPEA, and a cyclic crowded sorting method maintains the external archive.

A standard IEEE 30-bus 6-generator test system is used to verify the effectiveness of the MOGPEA. Two cases in this system have been considered. The extreme solutions and compromise solutions of the MOGPEA are compared with that of state-of-the-art algorithms. The compared results exhibit that the MOGPEA has a good compromise solution and highly diverse Pareto optimal solutions. Three metrics (SP, HV, CM) all show that MOGPEA yields solutions exhibiting better diversity and uniformity compared with the MOPSO, NSGA-II and PESA-II. The experimental results show that the MOGPEA is efficient and competitive for solving the multiobjective EED problem. This paper demonstrates that the novel GPEA can obtain competitive solutions for the EED problem, and the two learning strategies (i.e., Leader-updating strategy and Leader-guiding strategy) have good effects on improving the uniformity and diversity of the Pareto front. In the future, we will use the algorithm and the two learning strategies to investigate more realistic dynamic EED (DEED) problems.

REFERENCES

[1] J.-J. Cai, X.-Q. Ma, L.-X. Li, Y.-X. Yang, and H. P. Peng, "Economic dispatch in power plant based on chaotic ant swarm optimization," in *Proc. IET Int. Conf. Adv. Power Syst. Control*, 2006, p. 227.

[2] J. Cai, X. Ma, L. Li, and P. Haipeng, "Chaotic particle swarm optimization for economic dispatch considering the generator constraints," *Energy Convers. Manage.*, vol. 48, no. 2, pp. 645–653, Feb. 2007.

[3] K. D. Le, J. L. Golden, C. J. Stansberry, R. L. Vice, J. T. Wood, J. Ballance, G. Brown, J. Y. Kamya, E. K. Nielsen, H. Nakajima, M. Ookubo, I. Iyoda, and G. W. Cauley, ''Potential impacts of clean air regulations on system operations,'' *IEEE Trans. Power Syst.*, vol. 10, no. 2, pp. 647–656, May 1995.

[4] L. Wang and C. Singh, ''Stochastic economic emission load dispatch through a modified particle swarm optimization algorithm,'' *Electric Power Syst. Res.*, vol. 78, no. 8, pp. 1466–1476, Aug. 2008.

[5] J. H. Talaq, F. El-Hawary, and M. E. El-Hawary, ''A summary of environmental/economic dispatch algorithms,'' *IEEE Trans. Power Syst.*, vol. 9, no. 3, pp. 1508–1516, 1994.

[6] A. Farag, S. Al-Baiyat, and T. C. Cheng, ''Economic load dispatch multi-objective optimization procedures using linear programming techniques,'' *IEEE Trans. Power Syst.*, vol. 10, no. 2, pp. 731–738, May 1995.

[7] M. Basu, ''Particle swarm optimization based goal-attainment method for dynamic economic emission dispatch,'' *Electric Power Compon. Syst.*, vol. 34, no. 9, pp. 1015–1025, Sep. 2006.

[8] B. Y. Qu, Y. S. Zhu, Y. C. Jiao, M. Y. Wu, P. N. Suganthan, and J. J. Liang, ''A survey on multi-objective evolutionary algorithms for the solution of the environmental/economic dispatch problems,'' *Swarm Evol. Comput.*, vol. 38, pp. 1–11, Feb. 2018.

[9] J. D. Schaffer, ''Multiple objective optimization with vector evaluated genetic algorithms,'' in *Proc. 1st Int. Conf. Genetic Algorithms Appl.* Mahwah, NJ, USA: Lawrence Erlbaum Associates. Inc., 1985, pp. 93–100.

[10] M. A. Abido, ''A novel multiobjective evolutionary algorithm for environmental/economic power dispatch,'' *Electr. Power Syst. Res.*, vol. 65, no. 1, pp. 71–81, Apr. 2003.

[11] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, ''A fast and elitist multiobjective genetic algorithm: NSGA-II,'' *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[12] S. Dhanalakshmi, S. Kannan, K. Mahadevan, and S. Baskar, ''Application of modified NSGA-II algorithm to combined economic and emission dispatch problem,'' *Int. J. Electr. Power Energy Syst.*, vol. 33, no. 4, pp. 992–1002, May 2011.

[13] R. Muthuswamy, M. Krishnan, K. Subramanian, and B. Subramanian, ''Environmental and economic power dispatch of thermal generators using modified NSGA-II algorithm,'' *Int. Trans. Electr. Energy Syst.*, vol. 25, no. 8, pp. 1552–1569, Aug. 2015.

[14] M. A. Abido, ''A niched Pareto genetic algorithm for multiobjective environmental/economic dispatch,'' *Int. J. Elect. Power Energy Syst.*, vol. 25, no. 2, pp. 97–105, 2003.

[15] C.-L. Chiang, ''Optimal economic emission dispatch of hydrothermal power systems,'' *Int. J. Electr. Power Energy Syst.*, vol. 29, no. 6, pp. 462–469, Jul. 2007.

[16] T. C. Bora, V. C. Mariani, and L. D. S. Coelho, ''Multi-objective optimization of the environmental-economic dispatch with reinforcement learning based on non-dominated sorting genetic algorithm,'' *Appl. Thermal Eng.*, vol. 146, pp. 688–700, Jan. 2019.

[17] Z. Bo and C. Yi-jia, ''Multiple objective particle swarm optimization technique for economic load dispatch,'' *J. Zhejiang Univ.-Sci. A*, vol. 6, no. 5, pp. 420–427, May 2005.

[18] M. A. Abido, ''Multiobjective particle swarm optimization for environmental/economic dispatch problem,'' *Electr. Power Syst. Res.*, vol. 79, no. 7, pp. 1105–1113, Jul. 2009.

[19] L. Wang and C. Singh, ''Environmental/economic power dispatch using a fuzzified multi-objective particle swarm optimization algorithm,'' *Electr. Power Syst. Res.*, vol. 77, no. 12, pp. 1654–1664, Oct. 2007.

[20] S. Agrawal, B. K. Panigrahi, and M. K. Tiwari, ''Multiobjective particle swarm algorithm with fuzzy clustering for electrical power dispatch,'' *IEEE Trans. Evol. Comput.*, vol. 12, no. 5, pp. 529–541, Oct. 2008.

[21] J. Cai, X. Ma, Q. Li, L. Li, and H. Peng, ''A multi-objective chaotic particle swarm optimization for environmental/economic dispatch,'' *Energy Convers. Manage.*, vol. 50, no. 5, pp. 1318–1325, May 2009.

[22] Y. Zhang, D.-W. Gong, and Z. Ding, ''A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch,'' *Inf. Sci.*, vol. 192, pp. 213–227, Jun. 2012.

[23] T. Liu, L. Jiao, W. Ma, J. Ma, and R. Shang, ''Cultural quantum-behaved particle swarm optimization for environmental/economic dispatch,'' *Appl. Soft Comput.*, vol. 48, pp. 597–611, Nov. 2016.

[24] M. Varadarajan and K. S. Swarup, ''Solving multi-objective optimal power flow using differential evolution,'' *IET Gener., Transmiss. Distrib.*, vol. 2, no. 5, p. 720, 2008.

[25] L. H. Wu, Y. N. Wang, X. F. Yuan, and S. W. Zhou, ''Environmental/economic power dispatch problem using multi-objective differential evolution algorithm,'' *Electr. Power Syst. Res.*, vol. 80, no. 9, pp. 1171–1181, Sep. 2010.

[26] Y. Lu, J. Zhou, H. Qin, Y. Wang, and Y. Zhang, ''Environmental/economic dispatch problem of power system by using an enhanced multi-objective differential evolution algorithm,'' *Energy Convers. Manage.*, vol. 52, no. 2, pp. 1175–1183, Feb. 2011.

[27] B. Y. Qu, J. J. Liang, Y. S. Zhu, Z. Y. Wang, and P. N. Suganthan, ''Economic emission dispatch problems with stochastic wind power using summation based multi-objective evolutionary algorithm,'' *Inf. Sci.*, vol. 351, pp. 48–66, Jul. 2016.

[28] D. B. Das and C. Patvardhan, ''New multi-objective stochastic search technique for economic load dispatch,'' *IEE Proc.-Gener., Transmiss. Distrib.*, vol. 145, no. 6, p. 747, 1998.

[29] D.-W. Gong, Y. Zhang, and C.-L. Qi, ''Environmental/economic power dispatch using a hybrid multi-objective optimization algorithm,'' *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 6, pp. 607–614, Jul. 2010.

[30] E. A. Amorim, F. G. M. Lima, R. Romero, and J. R. S. Mantovani, ''Multiarea optimal power flow using multiobjective evolutionary algorithm,'' in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2009, pp. 506–511.

[31] A. Bhattacharya and P. K. Chattopadhyay, ''Hybrid differential evolution with biogeography-based optimization algorithm for solution of economic emission load dispatch problems,'' *Expert Syst. Appl.*, vol. 38, no. 11, pp. 14001–14010, 2011.

[32] S. Chansareewittaya, ''Hybrid MODE/TS for environmental dispatch and economic dispatch,'' *ECTI Trans. Electr. Eng., Electron., Commun.*, vol. 17, no. 1, pp. 78–86, 2019.

[33] Q. Zhang and H. Li, ''MOEA/D: A multiobjective evolutionary algorithm based on decomposition,'' *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[34] M. A. Abido, ''Multiobjective evolutionary algorithms for electric power dispatch problem,'' *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 315–329, Jun. 2006.

[35] P. Venkatesh and K. Y. Lee, ''Multi-objective evolutionary programming for economic emission dispatch problem,'' in *Proc. IEEE Power Energy Soc. Gen. Meeting-Convers. Del. Electr. Energy 21st Century*, Jul. 2008, pp. 1–8.

[36] B.-Y. Qu, P. N. Suganthan, V. R. Pandi, and B. K. Panigrahi, ''Multi objective evolutionary programming to solve environmental economic dispatch problem,'' in *Proc. 11th Int. Conf. Control Autom. Robot. Vis.*, Dec. 2010, pp. 673–1679.

[37] P. K. Hota, A. K. Barisal, and R. Chakrabarti, ''Economic emission load dispatch through fuzzy based bacterial foraging algorithm,'' *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 7, pp. 794–803, Sep. 2010.

[38] B. K. Panigrahi, V. R. Pandi, R. Sharma, S. Das, and S. Das, ''Multiobjective bacteria foraging algorithm for electrical load dispatch problem,'' *Energy Convers. Manage.*, vol. 52, no. 2, pp. 1334–1342, Feb. 2011.

[39] S. Hemamalini and S. P. Simon, ''Economic/emission load dispatch using artificial bee colony algorithm,'' *iPi*, vol. 1, p. 2, 2009.

[40] S. Ma, Y. Wang, and Y. Lv, ''Multiobjective Environment/Economic power dispatch using evolutionary multiobjective optimization,'' *IEEE Access*, vol. 6, pp. 13066–13074, 2018.

[41] R. Kumar, A. Sadu, R. Kumar, and S. K. Panda, ''A novel multi-objective directed bee colony optimization algorithm for multi-objective emission constrained economic power dispatch,'' *Int. J. Electr. Power Energy Syst.*, vol. 43, no. 1, pp. 1241–1250, Dec. 2012.

[42] T. Niknam, M. R. Narimani, M. Jabbari, and A. R. Malekpour, ''A modified shuffle frog leaping algorithm for multi-objective optimal power flow,'' *Energy*, vol. 36, no. 11, pp. 6420–6432, Nov. 2011.

[43] S. Sivasubramani and K. S. Swarup, ''Environmental/economic dispatch using multi-objective harmony search algorithm,'' *Electric Power Syst. Res.*, vol. 81, no. 9, pp. 1778–1785, Sep. 2011.

[44] B. Srinivasa Rao and K. Vaisakh, ''Multi-objective adaptive clonal selection algorithm for solving environmental/economic dispatch and OPF problems with load uncertainty,'' *Int. J. Electr. Power Energy Syst.*, vol. 53, pp. 390–408, Dec. 2013.

[45] R. Zhang, J. Zhou, L. Mo, S. Ouyang, and X. Liao, ''Economic environmental dispatch using an enhanced multi-objective cultural algorithm,'' *Electric Power Syst. Res.*, vol. 99, pp. 18–29, Jun. 2013.

[46] J. Cai, X. Ma, Q. Li, L. Li, and H. Peng, ''A multi-objective chaotic ant swarm optimization for environmental/economic dispatch,'' *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 5, pp. 337–344, Jun. 2010.

[47] M. Modiri-Delshad and N. A. Rahim, "Multi-objective backtracking search algorithm for economic emission dispatch problem," *Appl. Soft Comput.*, vol. 40, pp. 479–494, Mar. 2016.

[48] M. de Athayde Costa e Silva, C. E. Klein, V. C. Mariani, and L. dos Santos Coelho, "Multiobjective scatter search approach with new combination scheme applied to solve environmental/economic dispatch problem," *Energy*, vol. 53, pp. 14–21, May 2013.

[49] P. K. Roy and S. Bhui, "Multi-objective quasi-oppositional teaching learning based optimization for economic emission load dispatch problem," *Int. J. Electr. Power Energy Syst.*, vol. 53, pp. 937–948, Dec. 2013.

[50] X. Xu, Z. Hu, Q. Su, and Z. Xiong, "Multiobjective collective decision optimization algorithm for economic emission dispatch problem," *Complexity*, vol. 2018, pp. 1–20, Nov. 2018.

[51] T. Singh, K. K. Mishra, and Ranvijay, "Multiobjective environmental adaptation method for solving environmental/economic dispatch problem," *Evol. Intell.*, vol. 12, no. 2, pp. 305–319, Jun. 2019.

[52] D. Ju-Long, "Control problems of grey systems," *Syst. Control Lett.*, vol. 1, no. 5, pp. 288–294, Mar. 1982.

[53] J. Cui, S.-F. Liu, B. Zeng, and N.-M. Xie, "A novel grey forecasting model and its optimization," *Appl. Math. Model.*, vol. 37, no. 6, pp. 4399–4406, 2013.

[54] T. Yao, S. Liu, and N. Xie, "On the properties of small sample of GM(1,1) model," *Appl. Math. Model.*, vol. 33, no. 4, pp. 1894–1903, Apr. 2009.

[55] S. A. Javed and S. Liu, "Predicting the research output/growth of selected countries: Application of even GM (1, 1) and NDGM models," *Scientometrics*, vol. 115, no. 1, pp. 395–413, Apr. 2018.

[56] G. Li, "Image contrast enhancement algorithm based on GM(1,1) and power exponential dynamic decision," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 2, Feb. 2018, Art. no. 1854002.

[57] Z. Hu, X. Xu, Q. Su, H. Zhu, and J. Guo, "Grey prediction evolution algorithm for global optimization," *Appl. Math. Model.*, vol. 79, pp. 145–160, Mar. 2020.

[58] X. Xu, Z. Hu, Q. Su, Y. Li, and J. Dai, "Multivariable grey prediction evolution algorithm: A new Metaheuristic," *Appl. Soft Comput.*, vol. 89, Apr. 2020, Art. no. 106086.

[59] C.-Y. Luo, M.-Y. Chen, and C.-Y. Zhang, "Improved NSGA-II algorithm with circular crowded sorting," *Control Decis.*, vol. 25, no. 2, pp. 227–231, 2010.

[60] D. W. Corne, N. R. Jerram, J. D. Knowles, and M. J. Oates, "PESA-II: Region-based selection in evolutionary multiobjective optimization," in *Proc. Conf. Genetic Evol. Comput.*, 2001, pp. 1–8.

[61] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithm optimization," Air Force Inst. Technol., Wringt-Patterson Afb, OH, USA, Tech. Rep. AFIT/CI/CIA-95-039, 1995.

[62] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, Jun. 2000.

ZHONGBO HU received the Ph.D. degree from the Wuhan University of Technology, China, in 2014. He is currently a Professor with Yangtze University. His research interests include intelligent computing, data mining, and statistical calculation.

ZHENG LI received the B.Sc. degree from Henan Polytechnic University, China, in 2017. He is currently a Postgraduate Student with the School of Information and Mathematics, Yangtze University, China.

CANYUN DAI received the B.Sc. degree from the Xinlian College, Henan Normal University, China, in 2018. She is currently a Postgraduate Student with the School of Information and Mathematics, Yangtze University, China.

XINLIN XU received the B.Sc. degree from Huanggang Normal University, China, in 2016. She is currently a Postgraduate Student with the School of Information and Mathematics, Yangtze University, China.

ZENGGANG XIONG received the Ph.D. degree from the University of Science and Technology Beijing, Beijing, China, in 2009. He is currently a Professor with the School of Computer and Information Science and a Master Tutor with Hubei Engineering University, Xiaogan, China. His main research interests include cloud computing, big data, pattern recognition, and computer vision.

QINGHUA SU received the Ph.D. degree from the Wuhan University of Technology, China, in 2013. She is currently an Associate Professor with Yangtze University. Her research interests include intelligent computing and image processing.