Applied Soft Computing Journal 89 (2020) 106086

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Multivariable grey prediction evolution algorithm: A new metaheuristic

Xinlin Xu [a], Zhongbo Hu [a,*], Qinghua Su [a], Yuanxiang Li [b], Jianhua Dai [c]

[a] School of Information and Mathematics, Yangtze University, Jingzhou, Hubei, 434023, PR China
[b] School of Computer Science, Wuhan University, Wuhan, Hubei, 430000, PR China
[c] College of Information Science and Engineering, Hunan Normal University, Changsha, Hunan, 410000, PR China

ARTICLE INFO

Article history:
Received 2 May 2019
Received in revised form 10 November 2019
Accepted 10 January 2020
Available online 20 January 2020

Keywords:
Engineering design problems
Evolutionary algorithms
Grey prediction
MGM(1,n)

ABSTRACT

The theoretical foundation of the grey prediction system, proposed by Deng J. in 1982, is built on the fact that appropriate conversion can transform unordered data to series data with an approximate exponential law under certain conditions. Inspired by the grey prediction theory, this paper introduces a novel evolutionary algorithm based on the multivariable grey prediction model MGM(1,n), called MGPEA. The proposed MGPEA considers the population series of an evolutionary algorithm as a time series. It first transforms the population data to series data with an approximate exponential law and then forecasts its next population using MGM(1,n). Philosophically, MGPEA implements the optimizing process by forecasting the development trend of the genetic information chain of a population sequence. The performance of MGPEA is validated on CEC2005 benchmark functions, CEC2014 benchmark functions and a test suite composed of five engineering constrained design problems. The comparative experiments show the effectiveness and superiority of MGPEA.

The proposed MGPEA could be regard as a case of constructing metaheuristics by using the grey prediction model. It is hoped that this design idea leads to more metaheuristics inspired by other prediction models.

© 2020 Elsevier B.V. All rights reserved.

1. Introduction

In real life some practical science and engineering problems can be transformed into complex optimization problems, such as nonlinear, nondifferentiable and nonconvex objective function problems. To solve these optimization problems, researchers need to explore and design suitable optimization algorithms. Inspired by a variety of natural phenomena and/or biological social behaviour, researchers have proposed many successful and effective metaheuristic algorithms over the last few decades. Metaheuristic algorithms have shown remarkable performance in various problems due to their advantages of simplicity, flexibility, and derivation-free mechanisms. In general, these algorithms can be divided into three main categories: evolution-based, swarm-based and physics-based algorithms.

- *Evolution-based algorithms.* Evolution-based algorithms, which are called evolutionary algorithms (EAs), are inspired by the laws of biological evolution. Traditional EAs include genetic algorithms (GAs), evolution strategies (ES), evolutionary programming (EP), and genetic programming (GP), while recently development EAs include estimation of distribution algorithms (EDA), differential evolution (DE) algorithms, coevolutionary algorithms (CEAs), cultural algorithms (CAs) and so on.
- *Swarm-based algorithms.* Swarm-based algorithms simulate the intelligent behaviour of animals or human swarms, such as particle swarm optimization (PSO); based on the foraging behaviour of bird flocking, ant colony optimization (ACO), based on the foraging behaviour of a real ant colony; cuckoo search (CS), based on the brood parasitism of cuckoo species; artificial bee colony algorithm (ABC), based on the foraging behaviour of honey bee swarm; and teaching-learning-based optimization (TLBO), based on the influence of a teacher on learners.
- *Physics-based algorithms.* Physics-based algorithms imitate the laws of physics, such as simulated annealing (SA), based on modelling the steel annealing process; Black hole (BH), inspired by the black hole phenomenon; gravitational search algorithm (GSA), based on the law of gravity and mass interactions; and intelligent water drops (IWDs) algorithm on the law of natural water drops.

This paper proposes a new evolutionary algorithm, called the multivariable grey prediction evolution algorithm (MGPEA). The proposed algorithm is inspired by grey prediction theory, which is

* Corresponding author.
E-mail addresses: xuxlyangtze@163.com (X. Xu), huzbdd@126.com (Z. Hu), suqhdd@126.com (Q. Su), yxli@whu.edu.cn (Y. Li), jhdai@hunnu.edu.cn (J. Dai).

https://doi.org/10.1016/j.asoc.2020.106086
1568-4946/© 2020 Elsevier B.V. All rights reserved.

an important branch of grey system theory pioneered by Deng J. in 1982 [1]. Grey prediction theory processes raw data to generate a series of data with strong regularity and then establishes the corresponding differential equation models to predict the future development trends of the data. The multivariable grey model (MGM(1,n)) [2,3] plays an important role in the grey prediction theory. Unlike any metaheuristic algorithm, MGPEA uses MGM(1,n) to evolve the data series consisting of more than three populations. The performance of MGPEA is validated on CEC2005 and CEC2014 benchmark functions and a test suite composed of five engineering constrained design problems. Comparative experiments demonstrate the effectiveness and superiority of the proposed MGPEA.

The major contributions of this paper are as follows.

- *The algorithmic design.* This paper insights a fact that the population chain of an evolutionary algorithm can be considered the time series of a prediction model. This enables the proposed MGPEA to use MGM(1,n) to predict the offspring of current individuals (without employing any mutation and crossover operators).
- *The evolutionary philosophy.* If traditional EAs (i.e., GAs) simulate the chiasmata process of chromosomes by designing mutation and crossover operators, then MGPEA simulates the evolutionary information of population sequences by using MGM(1,n). The former simulates the microscopic properties of evolutionary processes and the latter simulates the macroscopic properties.

The rest of this paper is organized as follows. Section 2 introduces the basic theories of MGM(1,n) involved in MGPEA. Section 3 introduces the proposed MGPEA in detail. Section 4 discusses the searching mechanism and behaviour of the proposed algorithm. Several computational experiments are performed and discussed in Section 5. Finally, conclusions and some future work are introduced in Section 6.

2. Multivariable grey prediction model (MGM(1,n))

The multivariable grey model (MGM(1,n)) proposed by Zhai J. et al. in 1997 [2] is an extension and supplement of the grey model (GM(1,1)) [1,4–9], which is the most commonly used grey prediction model. MGM(1,n) [10,11] is appropriate for forecasting on multidimensional situations, while GM(1,1) is appropriate for univariate problems. It uniformly describes each variable from a systematic angle and is not a simple combination of GM(1,1).

Similar to GM(1,1), the core idea of MGM(1,n) is based on a transformation from unordered data to ordered data. It obtains the predicted value of the original ordered data by modelling based on the ordered data. MGM(1,n) involves the following three key concepts: 1-AGO operator, mean sequence and its basic model.

Assume that $\boldsymbol{X}^{(0)} = (X_1^{(0)}, X_2^{(0)}, \ldots, X_j^{(0)}, \ldots, X_n^{(0)})$ is an original nonnegative sequence, where $X_j^{(0)} = (x_j^{(0)}(1), x_j^{(0)}(2), \ldots, x_j^{(0)}(i), \ldots, x_j^{(0)}(m))^T$ is the observation sequence of the jth variable at times $1, 2, \ldots, m$.

Definition 1 (*1-AGO [1]*)**.** The matrix $\boldsymbol{X}^{(1)} = (X_1^{(1)}, X_2^{(1)}, \ldots, X_j^{(1)}, \ldots, X_n^{(1)})$ is the first-order accumulating generation matrix of $\boldsymbol{X}^{(0)}$. $X_j^{(1)} = (x_j^{(1)}(1), x_j^{(1)}(2), \ldots, x_j^{(1)}(i), \ldots, x_j^{(1)}(m))^T$ is the first-order accumulating generation sequence of $X_j^{(0)}$, where

$$x_j^{(1)}(i) = \sum_{k=1}^{i} x_j^{(0)}(k), \quad j = 1, 2, \ldots, n, \quad i = 1, 2, \ldots, m \tag{1}$$

Definition 2 (*Mean Sequence [1]*)**.** The matrix $\boldsymbol{Z}^{(1)} = (Z_1^{(1)}, Z_2^{(1)}, \ldots, Z_j^{(1)}, \ldots, Z_n^{(1)})$ is the mean generation matrix of $\boldsymbol{X}^{(1)}$, and the sequence $Z_j^{(1)} = (z_j^{(1)}(2), \ldots, z_j^{(1)}(i), \ldots, z_j^{(1)}(m))^T$ is the generated mean sequence of $X_j^{(1)}$, where

$$z_j^{(1)}(i) = \frac{1}{2}(x_j^{(1)}(i) + x_j^{(1)}(i-1)), \quad j = 1, 2, \ldots, n, \quad i = 2, \ldots, m \tag{2}$$

Definition 3 (*Basic MGM(1,n) Model [2]*)**.** The equation

$$X^{(0)}(i) + AZ^{(1)}(i) = B \tag{3}$$

is called the basic multivariable grey prediction model MGM(1,n).

Here $X^{(0)}(i) = (x_1^{(0)}(i), x_2^{(0)}(i), \ldots, x_n^{(0)}(i))$, $Z^{(1)}(i) = (z_1^{(1)}(i), z_2^{(1)}(i), \ldots, z_n^{(1)}(i))$, A is a developing grey matrix, and B is an endogenous control grey matrix.

By the least square method, the parameter matrix A and B can be obtained as

$$(A, B)^T = (L^T L)^{-1} L^T M \tag{4}$$

where

$$L = \begin{bmatrix} -z_1^{(1)}(2) & -z_2^{(1)}(2) & \cdots & -z_n^{(1)}(2) & 1 \\ -z_1^{(1)}(3) & -z_2^{(1)}(3) & \cdots & -z_n^{(1)}(3) & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -z_1^{(1)}(m) & -z_2^{(1)}(m) & \cdots & -z_n^{(1)}(m) & 1 \end{bmatrix},$$
$$M = \begin{bmatrix} x_1^{(0)}(2) & x_2^{(0)}(2) & \cdots & x_n^{(0)}(2) \\ x_1^{(0)}(3) & x_2^{(0)}(3) & \cdots & x_n^{(0)}(3) \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{(0)}(m) & x_2^{(0)}(m) & \cdots & x_n^{(0)}(m) \end{bmatrix} \tag{5}$$

The time response function of Eq. (3) is acquired and described as follows.

$$\hat{X}^{(1)}(i) = e^{-A\cdot(i-1)}(X^{(1)}(1) - A^{-1}B) + A^{-1}B, \quad i = 1, 2, \ldots, m \tag{6}$$

where $\hat{X}^{(1)}(i)$ denotes the prediction vector of $X^{(1)}(i)$.

Finally, by using the following inverse accumulated generation, we can obtain the prediction value of matrix $X^{(0)}$

$$\hat{X}^{(0)}(i) = \hat{X}^{(1)}(i) - \hat{X}^{(1)}(i-1), i = 2, 3, \ldots, m \tag{7}$$

where $\hat{X}^{(0)}(i)$ denotes the prediction vector of $X^{(0)}(i)$.

Combining Eqs. (6) and (7), the final prediction formula is as follows:

$$\hat{X}^{(0)}(i) = e^{-(i-1)\cdot A}(X^{(0)}(1) - A^{-1}B) - e^{-(i-2)\cdot A}(X^{(0)}(1) - A^{-1}B) \tag{8}$$

In particular, given an integer $C (C \geq 3)$, when $i = C + 1$,

$$\hat{X}^{(0)}(C+1) = e^{-C\cdot A}(X^{(0)}(1) - A^{-1}B) - e^{-(C-1)\cdot A}(X^{(0)}(1) - A^{-1}B) \tag{9}$$

Eq. (9) will be used to construct the reproduction operator of the following proposed algorithm in the next section.

3. Multivariable grey prediction evolution algorithm (MGPEA)

As with other EAs (i.e., GA and DE), after a population *initialization operator*, the proposed MGPEA carries out the optimization of functions by looping a *reproduction operator* and a *selection operator* to update each population. The main characteristic of MGPEA is its landmark reproduction operator, which is inspired by MGM(1,n). Moreover, the reproduction operator replaces the common mutation and crossover operators with MGM(1,n) prediction.

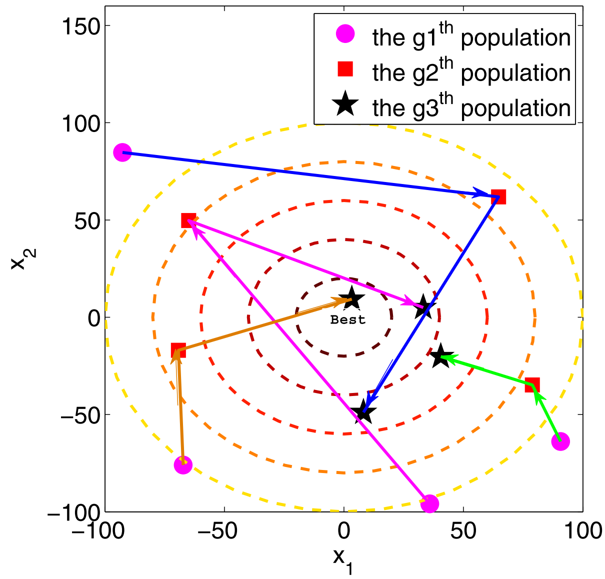


Fig. 1. Four IECs ($C = 3$).

A definition, information evolution chain, will be given before introducing MGPEA in this section.

Let $\boldsymbol{Y}^{(g)} = (Y_1^{(g)}, Y_2^{(g)}, \ldots, Y_i^{(g)}, \ldots, Y_N^{(g)})^T$ denote the gth generation population of N individuals, where $Y_i^{(g)} = (y_{i1}^{(g)}, y_{i2}^{(g)}, \ldots, y_{ij}^{(g)}, \ldots, y_{iD}^{(g)})$ denotes the ith individual with D dimensions.

Definition 4 (*Information Evolution Chain*)**.** Given a sequence $(Y_1, Y_2, \ldots, Y_k, \ldots, Y_C)^T$ consisting of $C(C \geq 3)$ individuals, the individual Y_k in the sequence come from different C populations $\boldsymbol{Y}^{(gk)}$ in turn, $k = 1, 2, \ldots, C$. Here, Y_k denotes the kth individual of the sequence, while $\boldsymbol{Y}^{(gk)}$ denotes the gkth generation population of an evolutionary process. If $gk_1 < gk_2$ when $k_1 < k_2$, then the sequence is said to be an information evolution chain (IEC). The parameter C is the length of the IEC.

An IEC is an individual series that simulates the genetic information of an evolutionary process. Thus, the C individuals of the IEC should be arranged by their genetic order of the evolutionary process, but the C populations for the IEC can come from successive generations or can be interval generations. As shown in Fig. 1, three generations, $\boldsymbol{Y}^{(g1)}, \boldsymbol{Y}^{(g2)}, \boldsymbol{Y}^{(g3)}$, with four individuals in each generation are given. The order of the three populations is $g1 < g2 < g3$. Then, the four IECs ($C = 3$), which are displayed in the direction of the flow arrows in Fig. 1, are constructed by randomly selecting four groups with three individuals from the three populations.

Next, the proposed MGPEA is described in detail according to its implementation steps, i.e., the initialization operator, the reproduction operator and the selection operator.

3.1. Initialization operator

To construct IECs, MGPEA must initialize at least three populations. We initialize $C(C \geq 3)$ populations, i.e., $\boldsymbol{Y}^{(0)}, \boldsymbol{Y}^{(1)}, \ldots, \boldsymbol{Y}^{(C-1)}$, containing N D-dimensional potential solutions (individuals) over the search space. The initialization operator is implemented by using a random uniform number to generate the potential individuals in the search space. We can initialize the jth dimension of the ith individual $Y_i^{(g)}$ according to

$$y_{i,j}^{(g)} = Lb_j + rand(0,1) \cdot (ub_j - lb_j), \quad g = 0, 1, \ldots, C-1;$$
$$i = 1, 2, \ldots, N; \quad j = 1, 2, \ldots, D. \tag{10}$$

Here, $rand(0,1)$ denotes a random number with uniform distribution between (0,1), and ub_j and lb_j denote the upper and lower bounds of the jth dimension in the search space, respectively. The $C * N$ individuals are sorted according to their fitness values from low to high. The sorted $C * N$ individuals are evenly divided into C parts and then assigned to $\boldsymbol{Y}^{(0)}$ to $\boldsymbol{Y}^{(C-1)}$ in turn.

3.2. Reproduction operator

Give an IEC of D-dimensional individuals. MGMEA's reproduction operator, named *mgm1n reproduction*, regards the IEC as a time series of MGM(1,n) to construct an exponential function for forecasting the next individual. In addition, considering the fact that the matrix $(L^T L)$ shown in Eq. (4) could be irreversible, *mgm1n* reproduction uses linear fitting to supplement. We can express *mgm1n* using the following formula:

$$U = \begin{cases} \text{MGM(1,D)(IEC)}, & \text{if} \quad (L^T L) \text{ is reversible.} \\ \text{Linear\_fitting(IEC)}, & \text{otherwise} \end{cases} \tag{11}$$

The pseudocode of *mgm1n* reproduction is given as the following Algorithm 1.

Algorithm 1: The simple pseudocode of *mgm1n* reproduction Operator

Input: An IEC $= (Y_1, Y_2, \cdots, Y_i, \cdots, Y_C)^T$, where $Y_i = (y_{i1}, y_{i2}, \cdots, y_{iD})$, $i = 1, 2, \cdots, C$
Output: A trail individual $U = (u_1, u_2, \cdots, u_D)$
//*Step1. Calculate 1-AGO matrix according to Eq.* (1)
for $j = 1$ *to* D **do**
  for $i = 1$ *to* C **do**
    $x_j^{(0)}(i) = y_{ij}$;
  end
end
Set the initial values: $x_j^{(1)}(1) = x_j^{(0)}(1)$, for $j = 1, 2, \cdots, D$
for $j = 1$ *to* D **do**
  for $i = 2$ *to* C **do**
    for $k = 2$ *to* i **do**
      $x_j^{(1)}(i) = x_j^{(1)}(k-1) + x_j^{(1)}(k)$;
      break; %%% "break" means to jump out of the loop.
    end
  end
end
//*Step2. Calculate the trail individual U according to Eqs.* (4) (5) (9)
Calculate L matrix by using the Eq. (5);
if $L^T L$ *is reversible* **then**
  %%% Use MGM(1,D) model to forecast.
  Calculate M matrix by using the Eq. (5);
  Use the least square method to calculate $\hat{A}$ and $\hat{B}$ by using the Eq. (4);
  Calculate the predicted values $\hat{X}^{(0)}(C+1)$ of $X^{(0)}(C+1)$ by using the Eq. (9);
else
  %%% Use linear fitting model to forecast.
  for $j = 1$ *to* D **do**
    $\hat{x}_j^{(0)}(C+1) = Linear\_fitting(x_j^{(0)}(1), x_j^{(0)}(2), \cdots, x_j^{(0)}(C))$;
  end
end
$U = \hat{X}^{(0)}(C+1)$;

Table 1
The statistical results of the MGPEA of C = 3 and the MGPEA of C = 4 on CEC2005.

F	The MGPEA of C = 3				The MGPEA of C = 4			
	Time	Best	Mean	Std.	Time	Best	Mean	Std.
F1	521	5.09E−10	2.24E−09	1.82E−09	532	4.10E−28	6.12E−28	1.54E−28
F6	534	0.028952	10.489449	30.271653	532	3.37E−21	1.78E−15	5.60E−15
F15	539	203.467486	348.801357	112.543628	539	50.364911	158.564109	96.610756

3.3. Selection operator

Finally, a selection operator is used to maintain the most promising trial individuals in the next-generation population. MGMEA adopts a greedy selection scheme. For minimization problems, if the objective function value of $Y_i^{(g)}$ is greater than the trial individual U, then $Y_i^{(g)}$ is replaced by U; otherwise, the old individual $Y_i^{(g)}$ is retained. The selection operator can be expressed as follows:

$$Y_i^{(g+1)} = \begin{cases} Y_i^{(g)}, & \text{if} \quad ObjFun(Y_i^{(g)}) < ObjFun(U) \\ U, & \text{otherwise} \end{cases} \tag{12}$$

where *ObjFun* denotes the function value of an individual.

3.4. Algorithm flow of MGPEA

After the above initialization operator, MGPEA solves optimization problems by looping *mgm1n* reproduction and the above selection operator. Taking minimization problems as an example, this subsection gives the algorithmic pseudocode (as shown in Alg. 2) and flow chat (as shown in Fig. 2).

Algorithm 2: Pseudo code of MGPEA

Input: $ObjFun, C, N, D, iter_{max}, lb_j, ub_j (j = 1, 2, \cdots, D)$
Output: global_minimum, global_minimizer
Initialize C populations
$(\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_C) = (\boldsymbol{Y}^{(0)}, \boldsymbol{Y}^{(1)}, \cdots, \boldsymbol{Y}^{(C-1)})$, and $iter = C$.
for $iter = 0$ *to* $C - 1$ **do**
  $BestInd_{iter}$ = the best individual of$\boldsymbol{Y}^{(iter)}$;
end
while $iter \le iter_{max}$ **do**
  for $i = 1$ *to* N **do**
    if *ObjFun*($Y_i^{(iter)}$) *is the worst of* $\mathbf{Y}^{(iter)}$ **then**
      Constuct IEC by randomly choosing C differential individuals in *BestInd*.
    else
      Constuct IEC by randomly choosing an individual in $\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_C$, respectively.
    end
    Run the *mgm1n* reproduction operator (Algorithm 1) and **Obtain** a trail individual U_i.
  end
  Select survivors from $\boldsymbol{U}$ and $\boldsymbol{Y}^{(iter)}$ (Eq. (12)) and **Assign** the survivors to $\boldsymbol{Y}^{(iter+1)}$. % Trial population $\boldsymbol{U} = (U_1, U_2, \cdots, U_N)$
  Update populations for IEC:
  $\boldsymbol{P}_1 = \boldsymbol{Y}^{(iter+2-C)}$, $\boldsymbol{P}_2 = \boldsymbol{Y}^{(iter+3-C)}, \cdots, \boldsymbol{P}_C = \boldsymbol{Y}^{(iter+1)}$,
  $BestInd_{iter+1}$ = the best individual of $Y^{(iter+1)}$.
end
Output the global_minimum and global_minimizer.

Note. $iter_{max}$ denotes the maximal number of iterations. The *BestInd* reports the best individual of every generation population.

4. Discussions about MGPEA

This section discusses the key parameter C (the length of the IEC) and the searching mechanism of MGPEA.

4.1. Discussion about the parameter C

To investigate the effect of parameter C on the performance of MGPEA, three types of unconstrained benchmark functions are used. Because of the stepwise optimizing nature of an iterative algorithm, MGPEA does not need to obtain accurately predicted values at each step. Therefore, this article only sets C to 3 and 4. These three functions are unimodal function F1, multimodal function F6 and hybrid composition function F15 from CEC2005 benchmark functions. Their details are provided in [12]. The parameters are set as follows: $N = 50$, $D = 10$ $iter_{max} = 2000$. MGPEA runs independently for 10 times on each function. The experimental results record the mean of operation times (*Time*), the best function value (*Best*), the mean of best function values (*Mean*), and the standard deviation of best function values (*Std.*). Table 1 presents the statistical results on three functions when $C = 3$ and $C = 4$.

From the results in Table 1, it can be observed that the operation times of the MGPEA of $C = 3$ and the MGPEA of $C = 4$ are basically the same, but the *Best*, *Mean* and *Std.* of the MGPEA of $C = 4$ are better than those of the MGPEA of $C = 3$. Therefore, the length of IEC for *mgm1n* reproduction is set to 4 (see Table 1).

4.2. Searching mechanism of MGPEA

Obviously, it is very difficult to directly forecast the development trend of unordered data. However, it is a relatively simple problem for ordered data. MGM(1,n) uses the 1-AGO operator to implement a data transformation from a nonnegative unordered data sequence to a sequence holding an approximate exponential law and then constructs an exponential function to forecast the trend of the transformed data sequence. Finally, the development trend of the original unordered data sequence is captured by the inverse operation of the 1-AGO operator.

The design philosophy of MGPEA is derived from the insight that IECs of an evolutionary process can be considered a time series to forecast the next population. IEC is an unordered data sequence for which it is difficult to forecast the development trend. First, MGPEA transforms the IEC to a nonnegative sequence, and second, it transforms the non-negative sequence to a sequence obeying an approximate exponential law by the 1-AGO operator. Finally, MGPEA uses the MGM(1,n) to forecast the offspring of the IEC.

The *mgm*1*n* reproduction, which undertakes the algorithmic essential exploration and exploitation tasks, is the key operator of the algorithm. From searching behaviour of view, the *mgm*1*n* reproduction has the property of adaptivity. That is to say, the size of the region, in which the offspring generated by the *mgm*1*n* operator are located adapts to the natural scaling of the populations for IECs. Taking the following *Camel* function (Fig. 3) as an example, we can observe the adaptivity of the *mgm*1*n*:

$$\min f = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4, x_1 \in (-2, 2),$$
$$x_2 \in (-1, 1) \tag{13}$$

In this test, 30 individuals are used to search for the minimum value. Fig. 4(a) shows the three populations of the first IEC and their offspring, furthermore, Fig. 4(b–f) shows the distributions of

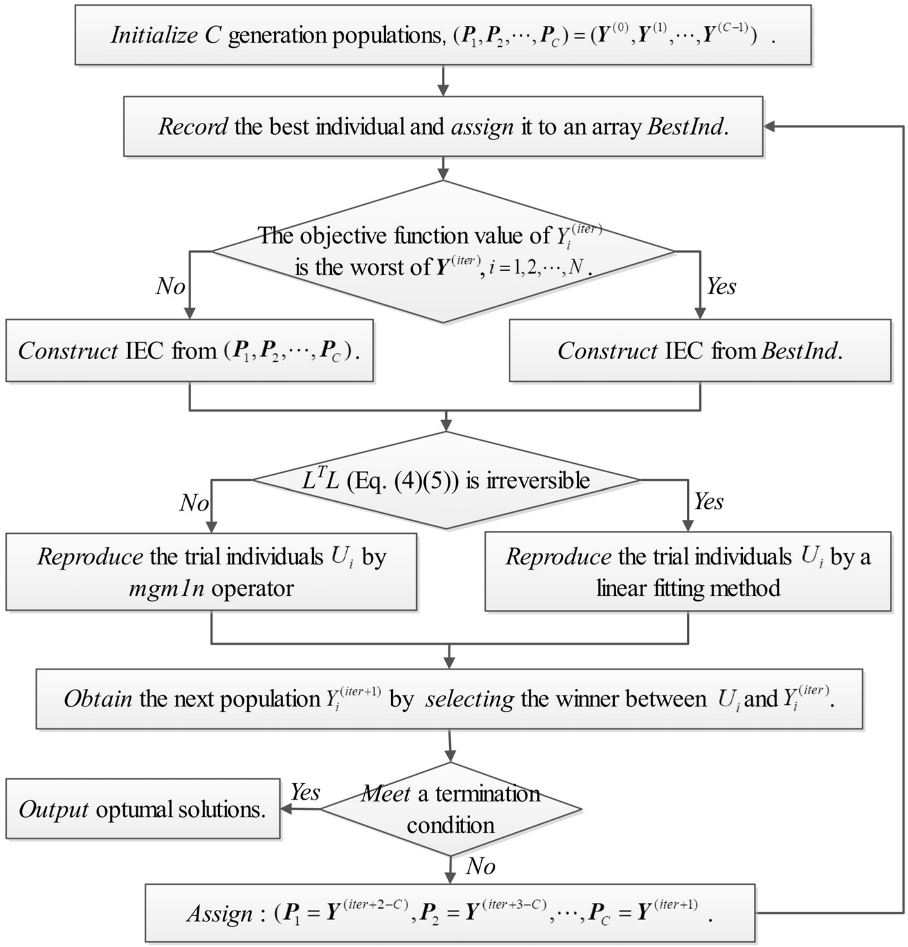


Fig. 2. Flow chat of MGPEA.

the tenth, twentieth, thirtieth, sixtieth, and eightieth generation, respectively. We can obtain that the offspring tend to locate in the region around the individuals of IECs.

5. Experimental studies

In order to verify the effectiveness of MGPEA versus other optimization techniques, different experiments have been performed. Different examples are investigated based on CEC2005 and CEC2014 benchmark test functions [12,13] and five engineering design problems. All experiments are conducted on the same platform with an Intel(R) Core(TM) i5-4590 CPU @ 3.30 GHz with 4 GB RAM in MatlabR2012a.

5.1. Experiments for benchmark functions

To evaluate the performance of the basic MGPEA, it is compared with four well-known algorithms (i.e., PSO, DE, covariance matrix adaptation-evolutionary strategies (CMA-ES) and backtracking search algorithm (BSA)) on CEC2005 and CEC2014 benchmark functions. In the experimental studies, *Best*, *Mean* and *std.*, which is the difference between the results generated by each algorithm and the known optimal results, are employed as performance metrics to evaluate the search capability of each algorithm.

5.1.1. Parameter settings

To ensure comparison as fair a comparison as possible, each experiment is independently run 30 times for all the compared algorithms, and the results of every run are recorded. The maximum number of function evaluations (*Max_FES*) is regarded as the termination condition, which is set to D∗10,000 (D is the dimension of the feasible solution space; we test 10-dimensions) for each method. The following provides the parameter settings of MGPEA and the compared methods.

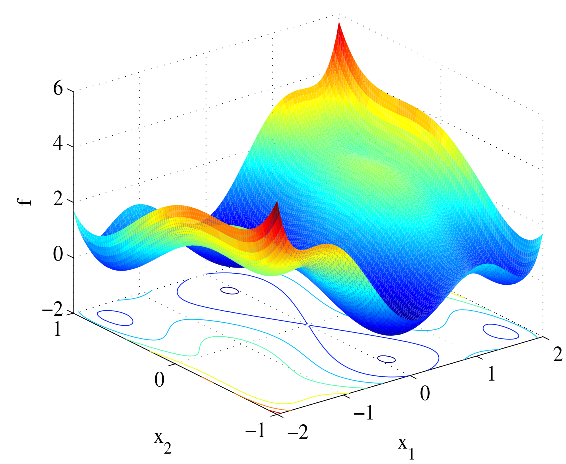


Fig. 3. 3-dimension contours of *Camel* function.

- MGPEA Settings : population size $N = 50$, the length of IEC $C = 4$.

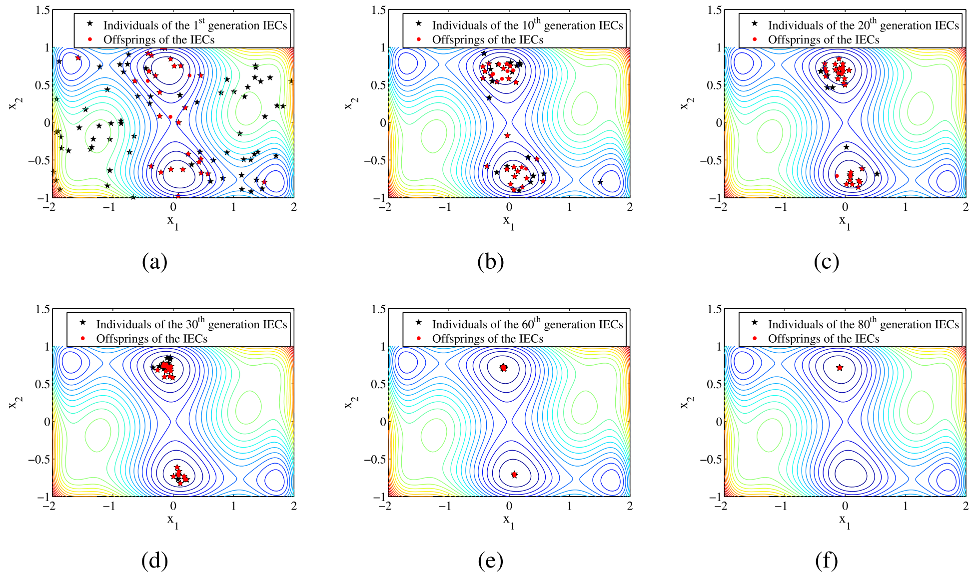


Fig. 4. Convergence plots of MGPEA in different iterations.

- PSO Settings : population size $N = 50$, positive constant $c1 = c2 = 1.4$, inertia weight $w = 0.7$.
- DE/rand/1 Settings : population size $N = 50$, mutation percentage $F = 0.7$, crossover percentage $CR = 0.7$.
- CMA-ES Settings : population size $N = 50$, number of parents $\mu = N/2$.
- BSA Settings : population size $N = 50$, crossover percentage $mixrate = 1$.

5.1.2. Comparison results in CEC2005 on 10-dimensional (10D) data

The CEC2005 test functions include many properties, such as unimodality, multimodality, separability, nonseparability, scalability rotation and shift. Since unimodal functions have only one global optimum and no local optimum, they are effective in evaluating the exploitation capability of algorithm, while multimodal functions contain numerous local optimal solutions, which exponentially increase in number with increasing problem dimension. This type of test function is helpful to verify the exploration capability of an algorithm and reflects that the algorithm has the capability to escape from the local optima and locates a good near-global optimum. Among the CEC2005 test functions, F1–F5 are unimodal functions, and F6–F14 are multimodal functions. F15–F25 are hybrid composition functions.

In this part, Table 2 summarizes the statistical results, including the *Best*/*Mean*/*Std* of the error values (F(x)-F(x*)) obtained using the MGPEA, PSO, DE, CMA-ES and BSA algorithms. The last three rows of Table 2 report the ranked results of the three algorithms. The best results are highlighted in boldface. Comparing the *Best* values of the five algorithms reveals that MGPEA ranks first for functions F6, F8, F12, F14, F19, F20, F21, F22, and F24 and ranks last only on function F25. Therefore, it should be noted that MGPEA performs significantly better than its competitors on hybrid composition functions (F15–F25). In terms of the *Mean* index, MGPEA outperforms all or some competitors on functions F3, F6, F8, F10, F14, and F24. The standard deviation (std.) of MGPEA is the smallest among all algorithms on functions F3, F6, F8 and F24. The average rank of MGPEA in terms of *Best* equal to 2.08 is better than those of other competitors. The average rank of MGPEA in terms of *Mean* is only worse than that of BSA, and it is better than those of other four algorithms. The average rank of *Std*. obtained by MGPEA is worse than those of DE and BSA, and better than those of PSO and CMA-ES. That is to say, MGPEA has excellent algorithmic accuracy but not excellent algorithmic robustness.

Moreover, to compare the convergence rate of the proposed algorithm with those of its competitors, the convergence curves of the test functions are given in Fig. 5. It can be observed from the figure that MGPEA not only has faster convergence speed but also has higher accuracy than its competitors. Based on the above results, it is concluded that MGPEA has certain competitiveness with the other four competitors evaluated.

5.1.3. Comparison results in CEC2014 on 10-dimensional (10D) data

The above experiment mainly discuss the experimental results of CEC2005. This part of the experiment is based on CEC2014. The details of the 30 functions in CEC2014 are reported in [13]. The functions of CEC2014 can be divided into four types, among which F1–F3 are unimodal functions, F4–F16 are multimodal functions. F17–F22 are hybrid functions, and F23–F30 are composition functions.

In the experiment, our statistical method is the same as that of the above experiment in terms of *Best*/*Mean*/*Std*. of the (F(x)-F(x*)) values and their average ranking (*Meanrank*). The results of these statistics are shown in Table 3. It can be observed from the table that MGPEA outperforms the other two comparative algorithms on functions F1–F5, F13–F15, F17–F18, F23, and F25–F29 when only the *Best* values are considered. For functions F1, F2, F3, F13, F14, F17, F18, F23–F26 and F29, MGPEA is superior to the other four algorithms in terms of *Mean*. The results for *Std*. obtained by MGPEA on functions F1, F2, F3, F18, F23 and F30 are better than those of other four competitors. In addition, the table also lists the average ranking of the three statistics fr the 30 functions. From the *Meanrank*, we can observe that MGPEA is superior to the other four algorithms considering the average ranking of *Best* and *Mean*. MGPEA ranks better than PSO and CMA-ES and worse than DE and BSA when considering *Std*. From these results, we can see that MGPEA is competitive with the other four algorithms in terms of algorithmic accuracy.

In addition, the convergence curves of the six test functions are shown in Fig. 6 to more intuitively observe the convergence rate of MGPEA. It can be observed from the figure that compared with other algorithms, MGPEA has faster convergence speed and higher algorithmic accuracy for these six functions.

Table 2
Comparison of statistical results given by five algorithms on 10D functions of CEC2005.

F^{rank}	runs = 30, D = 10, N = 50, T = 2000					
	AL	MGPEA	PSO	DE	CMA-ES	BSA
F1	Best	4.10E−28$^{(3)}$	7.00E−06$^{(4)}$	**0**$^{(1)}$	**0**$^{(1)}$	3.16E−3$^{(2)}$
	Mean	7.63E−28$^{(2)}$	4.80E−05$^{(4)}$	**0**$^{(1)}$	**0**$^{(1)}$	2.17E−25$^{(3)}$
	Std.	2.16E−28$^{(2)}$	3.50E−05$^{(4)}$	**0**$^{(1)}$	**0**$^{(1)}$	6.04E−25$^{(3)}$
F2	Best	2.22E−27$^{(2)}$	0.000134$^{(4)}$	5.91E−15$^{(3)}$	**0**$^{(1)}$	0.000457$^{(5)}$
	Mean	4.46E−27$^{(2)}$	4.867308$^{(5)}$	4.91E−13$^{(3)}$	**0**$^{(1)}$	0.005705$^{(4)}$
	Std.	2.11E−27$^{(2)}$	26.654746$^{(5)}$	6.51E−13$^{(3)}$	**0**$^{(1)}$	0.005992$^{(4)}$
F3	Best	1.76E−23$^{(2)}$	1278.159676$^{(3)}$	93487.183802$^{(5)}$	**0**$^{(1)}$	1369.335687$^{(4)}$
	Mean	**3.81E−23**$^{(1)}$	2,49E+04$^{(2)}$	2.70E+05$^{(4)}$	6,98E+06$^{(5)}$	2,8E+04$^{(3)}$
	Std.	**1.04E−23**$^{(1)}$	2,30E+04$^{(3)}$	1,11E+05$^{(4)}$	5,80E+06$^{(5)}$	1,7e+04$^{(2)}$
F4	Best	2.10E−22$^{(2)}$	0.000992$^{(4)}$	1.10E−11$^{(3)}$	**0**$^{(1)}$	0.032318$^{(5)}$
	Mean	2.92E−21$^{(2)}$	1.894521$^{(5)}$	1.53E−1$^{(3)}$	**0**$^{(1)}$	0.871043$^{(4)}$
	Std.	3.20E−21$^{(2)}$	10.3451$^{(5)}$	2.43E−1$^{(3)}$	**0**$^{(1)}$	1.585957$^{(4)}$
F5	Best	0.005747$^{(3)}$	0.003729$^{(2)}$	**0**$^{(1)}$	106.924266$^{(5)}$	0.078134$^{(4)}$
	Mean	0.009478$^{(2)}$	0.021553$^{(3)}$	**9.09E−13**$^{(1)}$	257.661807$^{(5)}$	1.702596$^{(4)}$
	Std.	0.002919$^{(2)}$	0.014683$^{(3)}$	**1.041E−12**$^{(1)}$	107.543012$^{(5)}$	2.057548$^{(4)}$
F6	Best	**1.24E−19**$^{(1)}$	0.1591$^{(5)}$	0.00423$^{(3)}$	0.009964$^{(4)}$	0.000712$^{(2)}$
	Mean	**3.10E−17**$^{(1)}$	79.418015$^{(4)}$	2.965722$^{(3)}$	157.703134$^{(5)}$	1.193565$^{(2)}$
	Std.	**7.92E−17**$^{(1)}$	163.853053$^{(4)}$	0.928894$^{(2)}$	863.474424$^{(5)}$	1.870164$^{(3)}$
F7	Best	1267.052211$^{(3)}$	1267.045949$^{(2)}$	1267.045949$^{(2)}$	**0**$^{(1)}$	1267.045949$^{(2)}$
	Mean	1267.066885$^{(3)}$	1267.519764$^{(4)}$	1267.045949$^{(2)}$	**0.006879**$^{(1)}$	1267.045949$^{(2)}$
	Std.	0.008819$^{(3)}$	0.256501$^{(5)}$	**5.97E−14**$^{(1)}$	0.02522$^{(4)}$	1.47E−07$^{(2)}$
F8	Best	**2**$^{(1)}$	20.14564$^{(2)}$	20.221739$^{(4)}$	20.242655$^{(5)}$	20.177217$^{(3)}$
	Mean	**20.000006**$^{(1)}$	20.345107$^{(2)}$	20.374981$^{(4)}$	20.422004$^{(5)}$	20.363322$^{(3)}$
	Std.	**3.00E−05**$^{(1)}$	0.066748$^{(2)}$	0.069088$^{(3)}$	0.099049$^{(5)}$	0.070579$^{(4)}$
F9	Best	1.989918$^{(4)}$	1.397875$^{(3)}$	**0**$^{(1)}$	0.994959$^{(2)}$	**0**$^{(1)}$
	Mean	9.491877$^{(4)}$	4.005882$^{(3)}$	**0**$^{(1)}$	12.305361$^{(5)}$	1.70E−12$^{(2)}$
	Std.	4.944462$^{(4)}$	1.521912$^{(3)}$	**0**$^{(1)}$	10.131159$^{(5)}$	7.02E−12$^{(2)}$
F10	Best	2.984877$^{(3)}$	6.984477$^{(4)}$	9.64102$^{(5)}$	**0.994959**$^{(1)}$	2.939488$^{(2)}$
	Mean	**9.096275**$^{(1)}$	21.008057$^{(5)}$	20.52063$^{(4)}$	17.555087$^{(3)}$	10.661739$^{(2)}$
	Std.	4.032144$^{(2)}$	8.943042$^{(5)}$	4.310969$^{(3)}$	8.832385$^{(4)}$	**3.849207**$^{(1)}$
F11	Best	1.159779$^{(2)}$	4.992165$^{(4)}$	6.44578$^{(5)}$	**0**$^{(1)}$	3.076094$^{(3)}$
	Mean	6.4625$^{(2)}$	8.175509$^{(4)}$	7.750858$^{(3)}$	9.070352$^{(5)}$	**4.951474**$^{(1)}$
	Std.	2.905294$^{(5)}$	1.502088$^{(3)}$	**0.546855**$^{(1)}$	2.536369$^{(4)}$	0.739364$^{(2)}$
F12	Best	**1.67E−18**$^{(1)}$	112.960411$^{(2)}$	1092.124737$^{(4)}$	1,52E+04$^{(5)}$	206.765309$^{(3)}$
	Mean	2328.062857$^{(3)}$	1178.388846$^{(2)}$	4476.675353$^{(4)}$	3,89E+04$^{(5)}$	**501.314403**$^{(1)}$
	Std.	9134.591958$^{(4)}$	1090.604634$^{(2)}$	1533.995902$^{(3)}$	1,09E+04$^{(5)}$	**264.430900**$^{(1)}$
F13	Best	0.415402$^{(3)}$	0.606206$^{(4)}$	0.746313$^{(5)}$	**0**$^{(1)}$	0.157831$^{(2)}$
	Mean	0.954505$^{(2)}$	1.724088$^{(5)}$	1.022662$^{(3)}$	1.149194$^{(4)}$	**0.353929**$^{(1)}$
	Std.	0.356688$^{(3)}$	0.551093$^{(4)}$	0.140134$^{(2)}$	0.740202$^{(5)}$	**0.089875**$^{(1)}$
F14	Best	**1.462861**$^{(1)}$	2.586677$^{(3)}$	2.966729$^{(4)}$	3.286044$^{(5)}$	2.338932$^{(2)}$
	Mean	**3.049309**$^{(1)}$	3.422733$^{(3)}$	3.457324$^{(4)}$	3.875801$^{(5)}$	3.144243$^{(2)}$
	Std.	0.611003$^{(5)}$	0.300011$^{(4)}$	**0.175871**$^{(1)}$	0.259778$^{(3)}$	0.21989$^{(2)}$
F15	Best	77.121423$^{(3)}$	0.978955$^{(2)}$	**0**$^{(1)}$	112.869861$^{(4)}$	0$^{(1)}$
	Mean	161.459894$^{(3)}$	288.134541$^{(4)}$	33.26681$^{(2)}$	324.716464$^{(5)}$	**0.003847**$^{(1)}$
	Std.	78.138286$^{(3)}$	205.433512$^{(5)}$	36.752005$^{(2)}$	128.983933$^{(4)}$	**0.019757**$^{(1)}$
F16	Best	90.774575$^{(2)}$	112.103826$^{(4)}$	114.217446$^{(5)}$	**61.497896**$^{(1)}$	96.418064$^{(3)}$
	Mean	111.679414$^{(2)}$	152.18704$^{(5)}$	137.563839$^{(4)}$	114.933954$^{(3)}$	**110.883738**$^{(1)}$
	Std.	15.362644$^{(3)}$	26.887605$^{(4)}$	11.894849$^{(2)}$	28.423999$^{(5)}$	**8.132409**$^{(1)}$
F17	Best	90.957974$^{(2)}$	106.421698$^{(3)}$	134.656653$^{(5)}$	**82.784635**$^{(1)}$	113.119299$^{(4)}$
	Mean	132.937652$^{(2)}$	156.664312$^{(5)}$	153.450594$^{(4)}$	140.218508$^{(3)}$	**129.801698**$^{(1)}$
	Std.	23.960439$^{(3)}$	30.580239$^{(4)}$	11.069079$^{(2)}$	33.742036$^{(5)}$	**10.648811**$^{(1)}$
F18	Best	3$^{(2)}$	501.019211$^{(3)}$	3$^{(2)}$	**133.116939**$^{(1)}$	3$^{(2)}$
	Mean	730.388365$^{(3)}$	946.417748$^{(5)}$	704.526327$^{(2)}$	852.947992$^{(4)}$	**408.08899**$^{(1)}$
	Std.	172.195623$^{(3)}$	**111.495366**$^{(1)}$	207.195897$^{(4)}$	310.168488$^{(5)}$	155.200249$^{(2)}$
F19	Best	**3**$^{(1)}$	696.373888$^{(2)}$	**3**$^{(1)}$	**3**$^{(1)}$	**3**$^{(1)}$
	Mean	583.333333$^{(2)}$	925.252151$^{(5)}$	656.310058$^{(3)}$	866.687739$^{(4)}$	**421.568674**$^{(1)}$
	Std.	252.003466$^{(5)}$	**103.553134**$^{(1)}$	222.412085$^{(4)}$	116.745431$^{(2)}$	153.042662$^{(3)}$

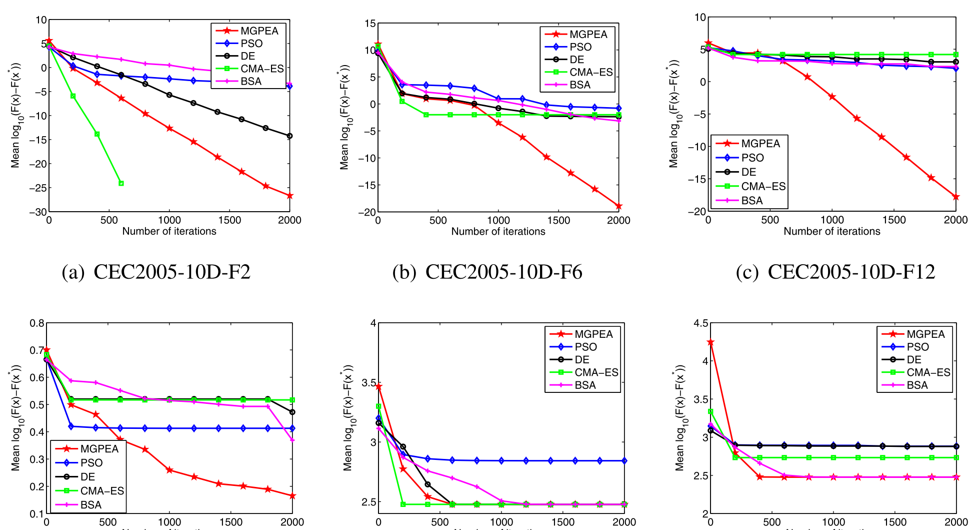
(*continued on next page*)

5.1.4. Comparisons using sign test

The sign test [14], which is used to determine whether there is significant difference between the two algorithms, is one of the most commonly used test methods. In this paper, the two-tailed sign test with a significance level of 0.05 is adopted to test the significance difference between the results of MGPEA and the results of other different algorithms. The test results of 55 functions, the combination of the 25 functions of CEC2005 and the 30 functions of CEC2014, are reported in Table 4. The value of *Best* is one of the most important criteria for algorithms, so it is chosen as the target of the sign test. The signs " + ", " ≈ " and " − " respectively indicate that the performance of MGPEA is significantly better, almost the same or significantly lower than that of the algorithm compared. The null hypothesis herein is that

Table 2 (*continued*).

F^{rank}	runs = 30, D = 10, N = 50, T = 2000					
	AL	MGPEA	PSO	DE	CMA-ES	BSA
F20	Best	**3**$^{(1)}$	548.004137$^{(3)}$	**3**$^{(1)}$	**3**$^{(1)}$	300.001583$^{(2)}$
	Mean	727.252785$^{(3)}$	935.267466$^{(5)}$	711.569484$^{(2)}$	795.131433$^{(4)}$	**451.009235**$^{(1)}$
	Std.	169.136697$^{(2)}$	**112.481374**$^{(1)}$	185.824324$^{(4)}$	229.806815$^{(5)}$	178.954284$^{(3)}$
F21	Best	**2**$^{(1)}$	300.041571$^{(4)}$	3$^{(3)}$	1051.268739$^{(5)}$	200.000608$^{(2)}$
	Mean	504.480305$^{(2)}$	741.991803$^{(4)}$	538.568503$^{(3)}$	1070.919729$^{(5)}$	**373.133835**$^{(1)}$
	Std.	133.663091$^{(3)}$	391.506869$^{(5)}$	192.907547$^{(4)}$	**8.837798**$^{(1)}$	106.384238$^{(2)}$
F22	Best	**3**$^{(1)}$	763.382568$^{(4)}$	756.927998$^{(3)}$	541.207127$^{(2)}$	**3**$^{(1)}$
	Mean	639.236751$^{(2)}$	811.6695$^{(5)}$	764.892851$^{(3)}$	784.104702$^{(4)}$	**625.720368**$^{(1)}$
	Std.	208.212382$^{(4)}$	58.269029$^{(2)}$	**3.855355**$^{(1)}$	158.889109$^{(3)}$	217.05774$^{(5)}$
F23	Best	559.468311$^{(2)}$	559.468993$^{(3)}$	559.468311$^{(2)}$	1092.182417$^{(4)}$	**425.17251**$^{(1)}$
	Mean	583.952655$^{(2)}$	943.512325$^{(4)}$	741.725127$^{(3)}$	1097.644286$^{(5)}$	**540.669874**$^{(1)}$
	Std.	83.725699$^{(3)}$	246.006319$^{(5)}$	176.184821$^{(4)}$	**6.378194**$^{(1)}$	60.542507$^{(2)}$
F24	Best	**2**$^{(1)}$	200.255368$^{(2)}$	**2**$^{(1)}$	376.123079$^{(3)}$	**2**$^{(1)}$
	Mean	**2**$^{(1)}$	825.689$^{(3)}$	**2**$^{(1)}$	405.066398$^{(2)}$	**2**$^{(1)}$
	Std.	**0**$^{(1)}$	252.731288$^{(3)}$	**0**$^{(1)}$	43.390754$^{(2)}$	**0**$^{(1)}$
F25	Best	823.803952$^{(5)}$	**200.003745**$^{(1)}$	819.099154$^{(4)}$	378.718752$^{(2)}$	618.88655$^{(3)}$
	Mean	826.720845$^{(5)}$	420.698628$^{(2)}$	821.880098$^{(4)}$	**393.034654**$^{(1)}$	790.614698$^{(3)}$
	Std.	1.67941$^{(2)}$	168.351707$^{(5)}$	**1.332850**$^{(1)}$	29.289577$^{(3)}$	73.556931$^{(4)}$
Mean rank	Best	**2.08**$^{(1)}$	3.08$^{(5)}$	2.96$^{(4)}$	2.36$^{(2)}$	2.44$^{(3)}$
	Mean	2.16$^{(2)}$	3.92$^{(5)}$	2.84$^{(3)}$	3.64$^{(4)}$	**1.88**$^{(1)}$
	Std.	2.76$^{(3)}$	3.52$^{(4)}$	**2.32**$^{(1)}$	3.56$^{(5)}$	2.4$^{(2)}$



(a) CEC2005-10D-F2 (b) CEC2005-10D-F6 (c) CEC2005-10D-F12

(d) CEC2005-10D-F14 (e) CEC2005-10D-F19 (f) CEC2005-10D-F22

Fig. 5. Convergence curves of five algorithms for 10D functions in CEC2005.

there is no significant difference in performance between MGPEA and one of the other algorithms.

As reported in Table 4, the p values supporting the null hypothesis of the sign test for 4 pairs of algorithms (MGPEA vs. PSO, MGPEA vs. DE, MGPEA vs. CMA-ES, MGPEA vs. BSA) are 0.000, 0.001, 0.004, and 0.033, respectively, and therefore we can reject the null hypothesis. This result illustrates that the optimization performance of the proposed MGPEA is significantly better than those of the four other algorithms. In addition, the statistics in the table also prove that MGPEA is superior to other known algorithms for the 55 functions.

5.2. Experiments for constrained engineering design problems

Five constrained engineering design problems (linear and nonlinear) are studied experimentally to evaluate the performance of the MGPEA. These engineering design problems are given in Appendix. These problems have been used by many researchers to test the performance of different algorithms, and some problems have mixed discrete–continuous design variables, such as the pressure vessel design problem. For constraint handling, many researchers have provided numerous types of constraint handling strategies, such as penalty functions, feasibility and dominance rules, stochastic ranking, and multi-objectives concepts. Deb's heuristic constraint handling method [15] is applied to the proposed MGPEA.

- If one solution is feasible and the other is not, then the feasible solution is preferred.
- If both solutions are feasible, then the solution with better objective function values is preferred.

Table 3
Comparison of statistical results given by five algorithms on 10D functions of CEC2014.

F^{rank}	runs = 30, D = 10, N = 50, T = 2000					
	AL	MGPEA	PSO	DE	CMA-ES	BSA
F1	Best	**0**$^{(1)}$	581.731001$^{(3)}$	1.13E+03$^{(4)}$	1.65E+06$^{(5)}$	137.1981780$^{(2)}$
	Mean	**0**$^{(1)}$	6.66E+04$^{(4)}$	2.67E+03$^{(3)}$	5.25E+06$^{(5)}$	1.01E+03$^{(2)}$
	Std.	**0**$^{(1)}$	9.14E+03$^{(4)}$	1.34E+03$^{(3)}$	3.22E+06$^{(5)}$	1.30E+03$^{(2)}$
F2	Best	**0**$^{(1)}$	50.571435$^{(3)}$	**0**$^{(1)}$	2.55E+04$^{(4)}$	1.17E−08$^{(2)}$
	Mean	**0**$^{(1)}$	1.34E+03$^{(3)}$	**0**$^{(1)}$	2.24E+05$^{(4)}$	8.00E−07$^{(2)}$
	Std.	**0**$^{(1)}$	2.24E+03$^{(3)}$	**0**$^{(1)}$	3.18E+05$^{(4)}$	9.94E−07$^{(2)}$
F3	Best	**0**$^{(1)}$	0.046996$^{(3)}$	**0**$^{(1)}$	3.04E+04$^{(4)}$	3.48E−11$^{(2)}$
	Mean	**0**$^{(1)}$	0.766644$^{(3)}$	**0**$^{(1)}$	8.47E+04$^{(4)}$	1.80E−09$^{(2)}$
	Std.	**0**$^{(1)}$	0.777878$^{(3)}$	**0**$^{(1)}$	2.43E+04$^{(4)}$	3.04E−09$^{(2)}$
F4	Best	**0**$^{(1)}$	0.017931$^{(3)}$	0.081191$^{(4)}$	0.473072$^{(5)}$	0.000601$^{(2)}$
	Mean	0.144514$^{(2)}$	28.189270$^{(5)}$	17.772544$^{(4)}$	1.00763$^{(3)}$	**0.061265**$^{(1)}$
	Std.	0.791533$^{(3)}$	13.437396$^{(4)}$	17.328727$^{(5)}$	0.374976$^{(2)}$	**0.130497**$^{(1)}$
F5	Best	**1.53E−07**$^{(1)}$	20.001489$^{(4)}$	16.493519$^{(3)}$	20.202780$^{(5)}$	0.068029$^{(2)}$
	Mean	19.107634$^{(2)}$	20.105590$^{(4)}$	19.784919$^{(3)}$	20.419239$^{(5)}$	**17.531811**$^{(1)}$
	Std.	3.814715$^{(4)}$	**0.085076**$^{(1)}$	0.942212$^{(3)}$	0.110406$^{(2)}$	6.083569$^{(5)}$
F6	Best	7.900e−05$^{(2)}$	3.7183110$^{(4)}$	**0**$^{(1)}$	**0**$^{(1)}$	0.001332$^{(3)}$
	Mean	0.202642$^{(3)}$	6.027768$^{(4)}$	**0**$^{(1)}$	**0**$^{(1)}$	0.201407$^{(2)}$
	Std.	0.825588$^{(3)}$	1.320187$^{(4)}$	**0**$^{(1)}$	**0**$^{(1)}$	0.209785$^{(2)}$
F7	Best	3.00E−06$^{(2)}$	0.061955$^{(5)}$	0.007738$^{(4)}$	**0**$^{(1)}$	1.30E−05$^{(3)}$
	Mean	0.098930$^{(3)}$	0.261897$^{(5)}$	0.118939$^{(4)}$	**0.001315**$^{(1)}$	0.019720$^{(2)}$
	Std.	0.088671$^{(4)}$	0.180342$^{(5)}$	0.062197$^{(3)}$	**0.003018**$^{(1)}$	0.015193$^{(2)}$
F8	Best	2.985244$^{(3)}$	11.939513$^{(4)}$	**0**$^{(1)}$	0.994959$^{(2)}$	**0**$^{(1)}$
	Mean	10.692523$^{(3)}$	28.190636$^{(5)}$	**0**$^{(1)}$	11.845319$^{(4)}$	3.02E−11$^{(2)}$
	Std.	4.575806$^{(3)}$	10.934809$^{(5)}$	**0**$^{(1)}$	8.912550$^{(4)}$	1.56E−10$^{(2)}$
F9	Best	0.994984$^{(2)}$	4.975000$^{(4)}$	8.866140$^{(5)}$	**0.994959**$^{(1)}$	1.315515$^{(3)}$
	Mean	7.555979$^{(2)}$	25.769531$^{(5)}$	12.849932$^{(3)}$	17.355848$^{(4)}$	**6.260748**$^{(1)}$
	Std.	3.645126$^{(3)}$	8.021032$^{(5)}$	2.367372$^{(2)}$	6.644096$^{(4)}$	**1.920047**$^{(1)}$
F10	Best	15.119641$^{(3)}$	128.685681$^{(5)}$	**0**$^{(1)}$	78.338277$^{(4)}$	0.062536$^{(2)}$
	Mean	574.444179$^{(3)}$	650.642297$^{(4)}$	0.349701$^{(2)}$	1.31E+03$^{(5)}$	**0.147367**$^{(1)}$
	Std.	307.560412$^{(5)}$	279.452526$^{(4)}$	0.998395$^{(2)}$	277.498234$^{(3)}$	**0.061517**$^{(1)}$
F11	Best	143.494721$^{(2)}$	498.362127$^{(4)}$	393.240810$^{(3)}$	1.03E+03$^{(5)}$	61.746525$^{(1)}$
	Mean	661.369636$^{(2)}$	894.028000$^{(4)}$	699.063959$^{(3)}$	1.39E+03$^{(5)}$	**183.804807**$^{(1)}$
	Std.	284.938184$^{(5)}$	238.000311$^{(4)}$	115.409035$^{(2)}$	147.024351$^{(3)}$	**63.278083**$^{(1)}$
F12	Best	2.39E−07$^{(2)}$	0.074869$^{(3)}$	0.319631$^{(5)}$	**0**$^{(1)}$	0.117879$^{(4)}$
	Mean	0.161912$^{(2)}$	0.306400$^{(4)}$	0.533199$^{(5)}$	**0**$^{(1)}$	0.240101$^{(3)}$
	Std.	0.197842$^{(5)}$	0.165806$^{(4)}$	0.098427$^{(3)}$	**0**$^{(1)}$	0.049080$^{(2)}$
F13	Best	**0.032804**$^{(1)}$	0.157704$^{(5)}$	0.045578$^{(2)}$	0.066537$^{(3)}$	0.100398$^{(4)}$
	Mean	**0.110483**$^{(1)}$	0.525151$^{(5)}$	0.150729$^{(3)}$	0.114949$^{(2)}$	0.172142$^{(4)}$
	Std.	0.052308$^{(4)}$	0.187105$^{(5)}$	0.031234$^{(2)}$	**0.022101**$^{(1)}$	0.043648$^{(3)}$
F14	Best	**0.056981**$^{(1)}$	0.094688$^{(4)}$	0.076754$^{(3)}$	0.288685$^{(5)}$	0.064903$^{(2)}$
	Mean	**0.140239**$^{(1)}$	0.364737$^{(4)}$	0.152424$^{(2)}$	0.428673$^{(5)}$	0.152803$^{(3)}$
	Std.	0.049983$^{(3)}$	0.129004$^{(5)}$	**0.034209**$^{(1)}$	0.053326$^{(4)}$	0.047913$^{(2)}$
F15	Best	**0.475407**$^{(1)}$	1.018306$^{(3)}$	1.047010$^{(4)}$	1.369677$^{(5)}$	0.598721$^{(2)}$
	Mean	1.103165$^{(2)}$	4.402100$^{(5)}$	1.586311$^{(3)}$	1.757943$^{(4)}$	**0.856596**$^{(1)}$
	Std.	0.471138$^{(4)}$	1.942185$^{(5)}$	0.305370$^{(3)}$	0.278583$^{(2)}$	**0.153334**$^{(1)}$
F16	Best	2.079341$^{(3)}$	2.166731$^{(4)}$	1.796276$^{(2)}$	2.660388$^{(5)}$	**0.615192**$^{(1)}$
	Mean	2.708952$^{(3)}$	3.040754$^{(4)}$	2.339601$^{(2)}$	3.418459$^{(5)}$	**2.000371**$^{(1)}$
	Std.	0.491386$^{(5)}$	0.424072$^{(4)}$	**0.262478**$^{(1)}$	0.362427$^{(3)}$	0.331446$^{(2)}$
F17	Best	**0.025453**$^{(1)}$	375.39911$^{(5)}$	22.829545$^{(3)}$	96.6542859$^{(4)}$	21.036851$^{(2)}$
	Mean	**4.229335**$^{(1)}$	2.02E+03$^{(5)}$	51.070337$^{(2)}$	96.654285$^{(4)}$	64.103619$^{(3)}$
	Std.	8.259012$^{(2)}$	2.41E+03$^{(5)}$	20.020706$^{(3)}$	**6.94E−13**$^{(1)}$	35.441680$^{(4)}$
F18	Best	**0.049337**$^{(1)}$	194.621153$^{(4)}$	1.040414$^{(3)}$	196.654285$^{(5)}$	0.519819$^{(2)}$
	Mean	**0.140805**$^{(1)}$	7.23E+03$^{(4)}$	2.1904$^{(3)}$	1.84E+04$^{(5)}$	2.167981$^{(2)}$
	Std.	**0.085455**$^{(1)}$	8.52E+03$^{(4)}$	0.599365$^{(2)}$	1.95E+04$^{(5)}$	0.936579$^{(3)}$
F19	Best	0.245817$^{(3)}$	1.636027$^{(4)}$	**0.075671**$^{(1)}$	2.638843$^{(5)}$	0.140979$^{(2)}$
	Mean	0.41088$^{(3)}$	3.589396$^{(4)}$	**0.194478**$^{(1)}$	834.041128$^{(5)}$	0.339448$^{(2)}$
	Std.	0.140002$^{(3)}$	1.266396$^{(4)}$	**0.089389**$^{(1)}$	3161.246427$^{(5)}$	0.105261$^{(2)}$
F20	Best	0.020899$^{(2)}$	23.051503$^{(4)}$	**0.000973**$^{(1)}$	688.94705$^{(5)}$	0.130248$^{(3)}$
	Mean	0.398164$^{(2)}$	705.800518$^{(4)}$	**0.130891**$^{(1)}$	2.64E+04$^{(5)}$	0.606303$^{(3)}$
	Std.	0.201971$^{(2)}$	1.31E+03$^{(4)}$	**0.110488**$^{(1)}$	1.91E+04$^{(5)}$	0.310825$^{(3)}$
F21	Best	0.079557$^{(2)}$	61.798004$^{(4)}$	**0.016977**$^{(1)}$	1.82E+03$^{(5)}$	0.302717$^{(3)}$
	Mean	0.425523$^{(2)}$	2.37E+03$^{(4)}$	**0.192754**$^{(1)}$	6.89E+03$^{(5)}$	1.670477$^{(3)}$
	Std.	0.217129$^{(2)}$	2.51E+03$^{(4)}$	**0.193280**$^{(1)}$	4.86E+03$^{(5)}$	1.088112$^{(3)}$
F22	Best	0.435906$^{(3)}$	22.103942$^{(4)}$	**1.100E−05**$^{(1)}$	46.772829$^{(5)}$	0.295666$^{(2)}$
	Mean	2.569984$^{(3)}$	128.396609$^{(4)}$	**0.106849**$^{(1)}$	2.67E+03$^{(5)}$	0.678564$^{(2)}$
	Std.	1.766780$^{(3)}$	57.976562$^{(4)}$	**0.149856**$^{(1)}$	5.28E+03$^{(5)}$	0.253879$^{(2)}$

(*continued on next page*)

Table 3 (*continued*).

F^{rank}	runs = 30, D = 10, N = 50, T = 2000					
	AL	MGPEA	PSO	DE	CMA-ES	BSA
F23	Best	**329.457475**$^{(1)}$	329.45748$^{(2)}$	**329.457475**$^{(1)}$	**329.457475**$^{(1)}$	**329.457475**$^{(1)}$
	Mean	**329.457475**$^{(1)}$	329.457527$^{(2)}$	**329.457475**$^{(1)}$	**329.457475**$^{(1)}$	**329.457475**$^{(1)}$
	Std.	**9.25E−13**$^{(1)}$	4.40E−05$^{(3)}$	**9.25E−13**$^{(1)}$	**9.25E−13**$^{(1)}$	9.44E−13$^{(2)}$
F24	Best	107.139811$^{(2)}$	127.094714$^{(5)}$	113.204029$^{(4)}$	**100**$^{(1)}$	107.556598$^{(3)}$
	Mean	**114.298504**$^{(1)}$	160.190750$^{(5)}$	119.776823$^{(4)}$	117.737456$^{(3)}$	115.004815$^{(2)}$
	Std.	5.510879$^{(3)}$	24.196373$^{(5)}$	3.171923$^{(2)}$	11.193678$^{(4)}$	**2.762322**$^{(1)}$
F25	Best	**109.724006**$^{(1)}$	166.472932$^{(5)}$	126.819966$^{(4)}$	114.976438$^{(2)}$	118.647349$^{(3)}$
	Mean	**125.130537**$^{(1)}$	194.260534$^{(4)}$	156.101634$^{(3)}$	197.711182$^{(5)}$	126.278098$^{(2)}$
	Std.	10.705604$^{(2)}$	11.145495$^{(3)}$	25.183327$^{(5)}$	16.106071$^{(4)}$	**4.763749**$^{(1)}$
F26	Best	**100.01958**$^{(1)}$	100.121346$^{(5)}$	100.113144$^{(4)}$	100.076647$^{(2)}$	100.112126$^{(3)}$
	Mean	**100.08959**$^{(1)}$	100.291803$^{(5)}$	100.169543$^{(2)}$	100.279298$^{(4)}$	100.171784$^{(3)}$
	Std.	0.043453$^{(3)}$	0.138403$^{(5)}$	**0.027981**$^{(1)}$	0.086645$^{(4)}$	0.039649$^{(2)}$
F27	Best	**1.194875**$^{(1)}$	2.692290$^{(4)}$	1.887996$^{(2)}$	3$^{(5)}$	2.370468$^{(3)}$
	Mean	42.272637$^{(2)}$	376.871531$^{(5)}$	151.198551$^{(3)}$	366.613573$^{(4)}$	**3.571699**$^{(1)}$
	Std.	121.327066$^{(3)}$	198.311211$^{(5)}$	151.345609$^{(4)}$	98.560558$^{(2)}$	**0.787119**$^{(1)}$
F28	Best	**3**$^{(1)}$	520.581613$^{(5)}$	356.826879$^{(3)}$	306.297956$^{(2)}$	356.913072$^{(4)}$
	Mean	393.06156$^{(4)}$	997.627382$^{(5)}$	361.630604$^{(3)}$	**321.424150**$^{(1)}$	361.569986$^{(2)}$
	Std.	48.718971$^{(4)}$	234.752426$^{(5)}$	18.088497$^{(2)}$	34.61323$^{(3)}$	**5.149379**$^{(1)}$
F29	Best	**100.000004**$^{(1)}$	381.054582$^{(5)}$	228.028774$^{(4)}$	209.943029$^{(3)}$	166.000593$^{(2)}$
	Mean	**112.831115**$^{(1)}$	6.85E+05$^{(5)}$	236.977416$^{(4)}$	228.749951$^{(3)}$	212.86503$^{(2)}$
	Std.	13.812833$^{(2)}$	1.61E+06$^{(5)}$	**8.575512**$^{(1)}$	27.879317$^{(4)}$	20.050295$^{(3)}$
F30	Best	458.649363$^{(2)}$	860.629279$^{(5)}$	506.450017$^{(4)}$	**287.990206**$^{(1)}$	499.2171$^{(3)}$
	Mean	470.643422$^{(2)}$	2.00E+03$^{(5)}$	547.571559$^{(3)}$	**425.32239**$^{(1)}$	564.706396$^{(4)}$
	Std.	**13.786606**$^{(1)}$	470.374944$^{(5)}$	27.360347$^{(2)}$	101.144553$^{(4)}$	50.717076$^{(3)}$
Mean rank	Best	**1.63**$^{(1)}$	4.07$^{(5)}$	2.67$^{(3)}$	3.4$^{(4)}$	2.4$^{(2)}$
	Mean	**1.9**$^{(1)}$	4.3$^{(5)}$	2.43$^{(3)}$	3.63$^{(4)}$	2.03$^{(2)}$
	Std.	2.87$^{(3)}$	4.2$^{(5)}$	**2.03**$^{(1)}$	3.2$^{(4)}$	2.07$^{(2)}$

Table 4
Comparisons between MGPEA and other algorithms in sign tests.

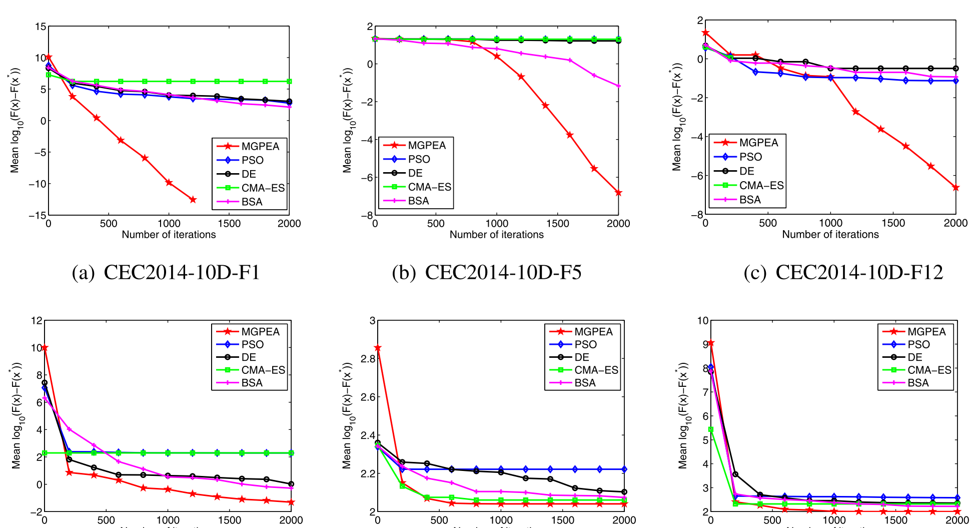
MGPEA-methods	+	≈	−	*p* value
PSO	50	0	5	0.000
DE	33	8	14	0.001
CMA-ES	32	3	20	0.004
BSA	36	5	14	0.033

Table 5
Parameter setting of MGPEA for engineering problems.

Problem	N	T	D
Three-bar truss	20	500	2
Pressure vessel	20	1500	4
Tension/Compression spring	20	1000	3
Welded beam	20	2000	4
Gear train	20	500	4

- If both solutions are infeasible, then solution with the minimal constraint violation is preferred.

Different parameter settings are applied for different engineering design problems. The parameters are given in Table 5, including the population size (N), the maximum number of iterations (T) and the dimension of the optimization problems (D).



(a) CEC2014-10D-F1 (b) CEC2014-10D-F5 (c) CEC2014-10D-F12

(d) CEC2014-10D-F18 (e) CEC2014-10D-F25 (f) CEC2014-10D-F29

Fig. 6. Convergence curves of five algorithms for 10D functions in CEC2014.

Table 6
Comparison of best solutions of the three-bar truss design problem.

Method	DEDS [16]	HEAA [17]	PSO-DE [18]	DELC [19]	MBA [20]	CSA [21]	MGPEA
X_1	0.788675	0.788680	0.788675	0.788675	0.788565	0.788675	0.788675
X_2	0.408248	0.408234	0.408248	0.408248	0.408560	0.408248	0.408248
$g_1(X)$	1.77E−08	N.A	−5.29E−11	N.A	−5.29E−11	−1.69E−14	0
$g_2(X)$	−1.464102	N.A	−1.463748	N.A	−1.463748	−1.464102	−1.464102
$g_3(X)$	−0.535898	N.A	−0.536252	N.A	−0.536252	−0.535898	−0.535898
$f(X)$	**263.895843**	**263.895843**	**263.895843**	**263.895843**	263.895852	**263.895843**	**263.895843**

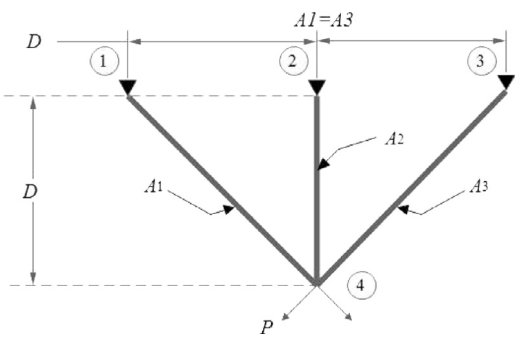


Fig. 7. Three-bar truss design problem.

Table 7
Comparison of statistical results of the three-bar truss design problem.

Method	Worst	Mean	Best	Std.	FEs
DEDS [16]	263.895849	**263.895843**	**263.895843**	9.7E−07	15,000
HEAA [17]	263.896099	263.895865	**263.895843**	4.9E−05	15,000
PSO-DE [18]	**263.895843**	**263.895843**	**263.895843**	4.5E−10	17,600
DELC [19]	**263.895843**	**263.895843**	**263.895843**	**4.3E−14**	10,000
MBA [20]	263.915983	263.897996	263.895852	3.93E−03	13,280
CSA [21]	**263.895843**	**263.895843**	**263.895843**	1.01E−10	25,000
SC [22]	263.969756	263.903356	263.895846	1.3E−02	17,610
MGPEA	**263.895843**	**263.895843**	**263.895843**	1.20E−12	**9740**

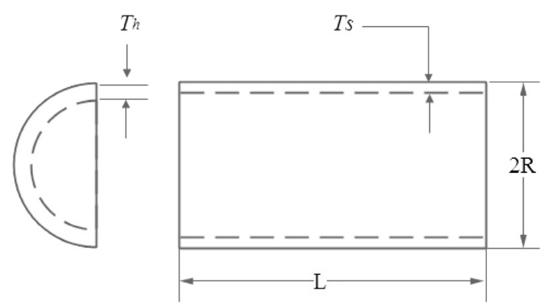


Fig. 8. Pressure vessel design problem.

5.2.1. Three-bar truss design problem

The three-bar truss design problem is the volume minimization problem of the structural optimization problem in civil engineering. The cross-sectional areas of two parameters should be adjusted to achieve the minimum weight constrained by stress, deflection and buckling. This problem has two design variables and three nonlinear inequality constraints. The different components of this problem are shown in Fig. 7. The formulation for this problem is shown in Appendix A.1.

This problem was solved using the different optimization algorithms: dynamic stochastic selection (DEDS) [16], hybrid evolutionary algorithm (HEAA) [17], hybrid particle swarm optimization with differential evolution (PSO-DE) [18], differential evolution with level comparison(DELC) [19], mine blast algorithm (MBA) [20], and crow search algorithm (CSA) [21]. The best solution obtained by MGPEA is compared with those obtained by other algorithms, and the results are reported in Table 6. It can be observed from the table that the function value of the best solution obtained by MGPEA, equal to 263.895843, reaches the lowest value in the current literature.

Table 7 reports the worst value (*Worst*), average value (*Mean*), best value (*Best*), standard deviation (Std) and the minimum number of function evaluations (*FEs*) obtained by several algorithms on this problem. These algorithms include MGPEA, DEDS, HEAA, PSO-DE, DELC, MBA, CSA and society and civilization (SC) [22]. It can be observed from the table that the *Worst*, *Best*, *Mean* and *FEs* obtained by MGPEA are better than those of the comparison algorithms and that MGPEA is worse than DELC only in terms of the *Std* is. These comparisons show that MGPEA has a certain competitiveness on this problem.

5.2.2. Pressure vessel design problem

The pressure vessel design problem, first proposed by Kannan and Kramer [23], is another popular engineering test problem in the metaheuristic literature; the problem involves minimization of the total material cost of a cylindrical vessel. As shown in Fig. 8, one side of the vessel is flat, and the other side is hemispherical. The structural parameters are the thickness of the shell $T_s(x_1)$, the thickness of the head $T_h(x_2)$, the inner radius $R(x_3)$ and the length of the cylindrical section $L(x_4)$, where T_s and T_h have to be integer multiples of 0.0625. The problem has three linear inequality constraints and one nonlinear inequality constraints. The formulation of this problem is shown in Appendix A.2.

MGPEA is used to find the best design for this problem, and compared with other well-known methods, including GA-based coevolution model(CGA) [24], GA through the use of dominance-based tour tournament selection (DGA) [25] , coevolutionary PSO(CPSO) [26], hybrid PSO(HPSO) [27], coevolutionary differential evolution (CDE) [28], CSA, PSO-DE and artificial bee colony (ABC)algorithm [29], in Tables 8 and 9.

Except for HPSO, the best value of which is equal to 6059.7143 that only keeps 4 digits after the decimal point and cannot be compared with those of the other algorithms, MGPEA is better than other six algorithms in terms of the best solution. Table 9 reports the statistical results obtained by MGPEA and the other eight algorithms. Form Table 9, MGPEA is far superior to the other algorithms in terms of *Worst*, *Best* and *FEs*. The *Std.* of MGPEA is only worse than that of CGA. The results in this two tables show that MGPEA has a faster convergence rate and provides a more accurate solution than the other algorithms.

5.2.3. Tension compression spring design problem

Another engineering test problem is tension compression spring design. It is mainly constrained by the minimum deflection, shear stress, surge frequency, outer diameter limit and design variable limit. That is, the problem has one linear inequality constraint, three nonlinear inequality constraints and multiple boundary constraints. The objective is to minimize the fabrication cost of a spring with three structural parameters: the wire diameter (d), mean coil diameter (D), and number of active coils (N), which are expressed by x1, x2 and x3, respectively. Fig. 9

Table 8
Comparison of best solutions of the pressure vessel design problem.

Method	CGA [24]	DGA [25]	CPSO [26]	HPSO [27]	CDE [28]	CSA [21]	MGPEA
X_1	0.8125	0.8125	0.8125	0.8125	0.8125	0.812500	0.8125
X_2	0.4375	0.4375	0.4375	0.4375	0.4375	0.437500	0.4375
X_3	40.3239	42.0974	42.0913	42.0984	42.098411	42.098445	42.098446
X_4	200.0000	176.6540	176.7465	176.6366	176.637690	176.636599	176.636596
$g_1(X)$	−3.42E−02	−2.01E−03	−1.37E−06	−8.80E−07	−6.67E−07	−4.02E−09	0
$g_2(X)$	−5.28E−02	−3.58E−02	−3.59E−04	−3.58E−02	−3.58E−02	−0.035880	−0.035881
$g_3(X)$	−304.4020	−24.7593	−118.7687	3.1226	−3.705123	−7.12E−04	0
$g_4(X)$	−400.0000	−63.3460	−63.2535	−63.3634	−63.362310	−63.363401	−63.363404
$f(X)$	6288.7445	6059.9463	6061.0777	**6059.7143**	6059.7340	6059.714363	**6059.714335**

Table 9
Comparison of statistical results of the pressure vessel design problem.

Method	Worst	Mean	Best	Std	FEs
CGA [24]	6308.4970	6293.8432	6288.7445	**7.4133**	900,000
DGA [25]	6469.3220	6177.2533	6059.9463	130.9297	80,000
CPSO [26]	6363.8041	6147.1332	6061.0777	86.45	240,000
HPSO [27]	6288.6770	6099.9323	**6059.7143**	86.20	81,000
CDE [28]	6371.0455	6085.2303	6059.7340	43.0130	204,800
CSA [21]	7332.841621	6342.499106	6059.714363	384.945416	250,000
PSO-DE [18]	N.A	**6059.714**	**6059.714**	N.A	42,100
ABC [29]	N.A	6245.308	**6059.714**	205	30,000
MGPEA	**6090.526202**	6062.795527	**6059.714335**	9.401579	**29,340**

Table 10
Comparison of best solutions of the tension compression spring design problem.

Method	X_1	X_2	X_3	$g_1(X)$	$g_2(X)$	$g_3(X)$	$g_4(X)$	$f(X)$
MDE [30]	0.051688	0.356692	11.290483	−0.000000	−0.000000	−4.053734	−0.727747	**0.012665**
CPSO [26]	0.051728	0.357644	11.244543	−8.25E−04	−2.52E−05	−4.051306	−0.727085	0.012675
HPSO [27]	0.051706	0.357126	11.265083	−3.06E−06	1.39E−06	−4.054583	−0.727445	**0.012665**
DEDS [16]	0.051689	0.356717	11.288965	1.45E−09	−1.19E−09	−4.053785	−0.727728	**0.012665**
HEAA [17]	0.051689	0.356729	11.288293	3.96E−10	−3.59E−10	−4.053808	−0.72772	**0.012665**
DELC [19]	0.051689	0.356717	11.288965	−3.40E−09	2.44E−09	−4.053785	−0.727728	**0.012665**
MBA [20]	0.051656	0.35594	11.344665	0	0	−4.052248	−0.728268	**0.012665**
CSA [21]	0.051689	0.356717	11.289012	−4.44E−16	−4.11E−15	−4.053784	−0.727729	**0.012665**
MGPEA	0.051688	0.356701	11.289950	−2.02E−09	−4.28E−10	−4.053752	−0.727740	**0.012665**

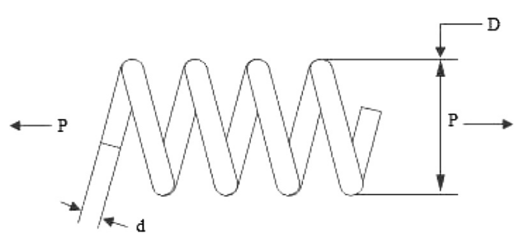


Fig. 9. Tension compression spring design problem.

shows the spring and its parameters. The formulation for this problem is shown in Appendix A.3.

There are several solutions to this problem in the literature, such as modified differential evolution (MDE) [30], CPSO, HPSO, DEDS, HEAA, DELC, MBA, CSA, CGA, DGA, PSO-DE, SC, CDE and ABC. The best solutions and statistical results, as shown in Tables 10 and 11, are calculated. From Table 10, the optimal function value of CPSO is slightly worse than those of the other algorithms, while the optimal function values of eight other algorithms including MGPEA are equal to 0.012665. From Table 11, the *Std*. and *FEs* are the fifth and second among the 15 algorithms, respectively. Therefore, in summary, MGPEA has good convergence and short computational times.

5.2.4. Welded beam design problem

The pressure vessel design problem is a well-known engineering problem in the field of structural optimization, which was first proposed by Coello [24]. The objective of this problem is to minimize the manufacturing cost of welded beam. As shown in Fig. 10 and Appendix A.4, this problem has four design variables h(x_1), l(x_2), t(x_3) and b(x_4) and seven constraints, two of which are linear inequality constraints and five of which are non-linear inequality constraints.

Table 11
Comparison of statistical results of the tension compression spring design problem.

Method	Worst	Mean	Best	Std.	FEs
MDE [30]	0.012674	0.012666	**0.012665**	2.00E−06	24,000
CPSO [26]	0.012924	0.012730	0.012675	5.20E−04	240,000
HPSO [27]	0.012719	0.012707	**0.012665**	1.58E−05	81,000
DEDS [16]	0.012738	0.012669	**0.012665**	1.30E−05	24,000
HEAA [17]	**0.012665**	**0.012665**	**0.012665**	**1.40E−09**	24,000
DELC [19]	0.012666	**0.012665**	**0.012665**	1.30E−07	20,000
MBA [20]	0.012900	0.012713	**0.012665**	6.30E−05	**7650**
CSA [21]	0.012670	0.012666	**0.012665**	1.36E−06	50,000
CGA [24]	0.012822	0.012769	0.012705	3.94E−05	900,000
DGA [25]	0.012973	0.012742	0.012681	5.90E−05	80,000
PSO-DE [18]	0.012665	**0.012665**	**0.012665**	1.2E−08	24,950
SC [22]	0.016717	0.012923	0.012669	5.9E−04	25,167
CDE [28]	N.A	0.012703	0.012670	N.A	240,000
ABC [29]	N.A	0.012709	**0.012665**	0.012813	30,000
MGPEA	0.012674	0.012666	**0.012665**	1.86E−06	19,740

MGPEA is used to solve this problem and is compared with CGA, MDE, CPSO, HPSO, MBA, CSA, PSO-DE, SC, CDE and ABC. Table 12 presents the best results. The results from the table show that the optimal function values of MDE, HPSO, CSA and MGPEA are equal to 1.724852, which is the smallest value obtained at present. Table 13 shows the statistical results obtained using the 11 algorithms. MGPEA's *Std*. and *FEs* are ranked third and second,

Table 12
Comparison of best solutions of the welded beam design problem.

Method	CGA [24]	MDE [30]	CPSO [26]	HPSO [27]	MBA [20]	CSA [21]	MGPEA
$X_1(h)$	0.205986	0.205730	0.202369	0.20573	0.205729	0.205730	0.205730
$X_2(l)$	3.471328	3.470489	3.544214	3.470489	3.470493	3.470489	3.470489
$X_3(t)$	9.020224	9.036624	9.04821	9.036624	9.036626	9.036624	9.036624
$X_4(b)$	0.206480	0.205730	0.205723	0.20573	0.205729	0.205730	0.205730
$g_1(X)$	−0.103049	−0.000335	−12.839796	−0.025399	−0.001614	0	0
$g_2(X)$	−0.231747	−0.000753	−1.247467	−0.053122	−0.016911	0	0
$g_3(X)$	−5E−04	−0.000000	−1.49E−03	0	−2.40E−07	0	2.78E−17
$g_4(X)$	−3.430044	−3.432984	−3.429347	−3.432981	−3.432982	−3.432984	−3.432984
$g_5(X)$	−0.080986	−0.080730	−0.079381	−0.08073	−0.080729	−0.080730	−0.080730
$g_6(X)$	−0.235514	−0.235540	−0.235536	−0.235540	−0.235540	−0.235540	−0.235540
$g_7(X)$	−58.646888	−0.000882	−11.681355	−0.031555	−0.001464	−3.637979	−5.46E−12
f(X)	1.728226	**1.724852**	1.728024	**1.724852**	1.724853	**1.724852**	**1.724852**

Table 13
Comparison of statistical results of the welded beam design problem.

Method	Worst	Mean	Best	Std.	FEs
CGA [24]	1.993408	1.792654	1.728226	7.47E−02	80,000
MDE [30]	1.724854	1.724853	**1.724852**	N.A	24,000
CPSO [26]	1.782143	1.748831	1.728024	1.29E−02	240,000
HPSO [27]	1.814295	1.749040	**1.724852**	4.01E−02	81,000
MBA [20]	1.724853	1.724853	1.724853	**6.94E−19**	47,340
CSA [21]	**1.724852**	**1.724852**	**1.724852**	1.19E−15	100,000
PSO-DE [18]	**1.724852**	**1.724852**	**1.724852**	6.7E−16	66,600
SC [22]	6.399679	3.002588	2.385435	9.60E−01	33,095
CDE [28]	N.A	1.76815	1.73346	N.A	240,000
ABC [29]	N.A	1.741913	**1.724852**	3.1E−02	30,000
MGPEA	**1.724852**	**1.724852**	**1.724852**	1.13E−15	32,140

Table 14
Comparison of best solutions of the gear train design problem.

Method	ABC [29]	MBA [20]	CSA [21]	MGPEA
X_1	49	43	49	49
X_2	16	16	19	19
X_3	19	19	16	16
X_4	43	49	43	43
f(X)	**2.7E−12**	**2.E−12**	**2.7E−12**	**2.70E−12**

respectively in the table. The results show that, compared with other algorithms, MGPEA can obtain a more accurate solution with a lower calculation cost.

5.2.5. Gear train design problem

The gear train design problem is a mechanical engineering problem that aims to minimize the gear ratio, expressed as $(n_B n_D)/(n_F n_A)$, for a given set of four gears of a train. The overall schematic diagram of the system is shown in Fig. 11. From the figure, the parameter in this problem is the number of teeth of the gear, so there are four variables, which are integers between 12 and 60. Therefore, the gear train design problem is a discrete problem. The problem has no constraints, but the ranges of variables are treated as constraints. The formulation of this problem is shown in Appendix A.5.

The problem is solved using MGPEA, and the best results are compared with those of ABC, MBA, and CSA in Table 14. MGPEA finds the same optimal gear ratio as ABC, MBA and CSA. Table 15 reports the statistical results of four algorithms. From the table, MGPEA is slightly worse than ABC in terms of *std.* and *FEs*. This result proves that MGPEA is also effective in solving discrete problems.

5.3. Results analysis of engineering design problems

The performance of MGPEA in five engineering constraints has been discussed in the five parts above, but the comparison algorithms of each part are different. This will cause us to be

Table 15
Comparison of statistical results of the gear train design problem.

Method	Worst	Mean	Best	Std.	NFEs
ABC [29]	N.A	3.6E−10	**2.7E−12**	**5.52E−10**	**60**
MBA [20]	**2.1E−08**	2.5E−09	**2.7E−12**	3.94E−09	1120
CSA [21]	3.2E−08	2.06E−09	**2.7E−12**	5.1E−09	100,000
MGPEA	2.72E−08	1.23E−09	**2.70E−12**	4.96E−09	840

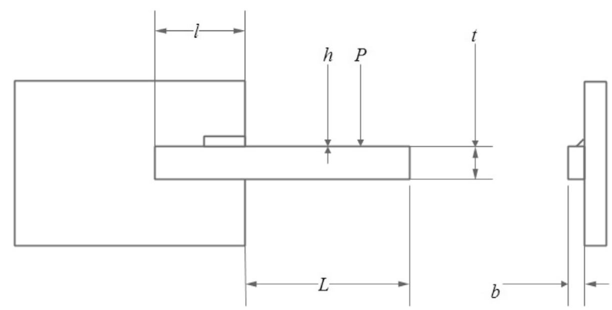


Fig. 10. Welded beam design problem.

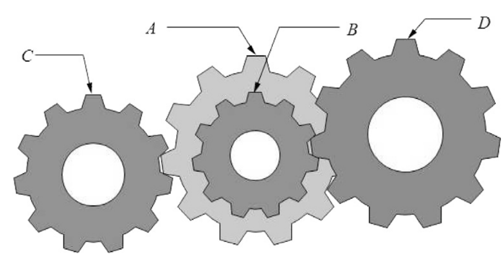


Fig. 11. Gear train design problem.

unable to accurately evaluate the performance of MGPEA. In order to further analyse the comprehensive performance of MGPEA, this section lists the smallest *Best*, *Std*. and *FEs* of the five engineering problems in the above tables, and compares them with MGPEA. The comparison results are reported in Table 16.

The following conclusions can be drawn from Table 16.

- *The algorithmic accuracy of MGPEA.* The value of *Best* reflects the accuracy of each algorithm in solving engineering optimization problems. The smaller the *Best* value is, the higher the solution accuracy. It can be observed from Table 16 that the average value of MGPEA's *Best* has reached the best value known in the literature for five engineering problems. Therefore, MGPEA has obvious advantages in terms of the accuracy.
- *The convergence speed of MGPEA.* The value of *FEs* reflects the calculation cost of the current optimal function values. The smaller the *FEs* value is, the faster the convergence rate. The

Table 16
Comparison for Best, Std and FEs obtained by MGPEA and best known results.

F	Smallest Best	Best of MGPEA	Smallest FEs	FEs of MGPEA	Smallest Std.	Std. of MGPEA
Three-bar truss	**263.895843**	**263.895843**	10,000	**9740**	**4.3E−14**	1.20E−12
Pressure vessel	**6059.7143**	**6059.714335**	30,000	**29,340**	7.4133	9.401579
Tension/Compression spring	**0.012665**	**0.012665**	**7650**	19,740	**1.40E−09**	1.86E−06
Welded beam	**1.724852**	**1.724852**	24,000	32,140	**6.94E−19**	1.13E−15
Gear train	**2.7E−12**	**2.7E−12**	**60**	840	**5.52E−10**	4.96E−09

FEs of MGPEA is the smallest for the three-bar truss design problem and pressure vessel design problem. In the other three problems, although the *FEs* of MGPEA is not the smallest, its ranking among all comparison algorithms is competitive. In general, regarding these optimization problems, MGPEA has better performance than the latest algorithms.

- *The robustness of MGPEA.* The value of *Std*. reflects the robustness of the algorithm. The smaller the value is, the higher the robustness. From the above results, it can be observed that the *Std*. of MGPEA is not as small as the minimum value known in the literature. According to "No free lunch", this does not mean that MGPEA is not competitive with other algorithms.

6. Concluding remarks and future work

A novel metaheuristic, the multivariable grey prediction evolution algorithm (MGPEA), is proposed in this paper. The algorithm considers the population series of an evolutionary algorithm as a time series of genetic information and defines the corresponding individual series as an information evolution chain (IEC). MGPEA obtains offspring by forecasting the development trend of the IEC. Unlike other evolutionary algorithms, its reproduction tool is the *mgm*1*n* operator, inspired by multivariable grey model MGM(1,n). In MGPEA, the IEC is the carrier of macroscopic genetic information. Thus, the proposed MGPEA simulates evolutional process of macroscopic genetic information rather than the macroscopic chromosome chiasmata.

The numerical results on CEC2005 and CEC2014 benchmark functions show that the performance of MGPEA, as a basic algorithm, is better than those of PSO, DE, CMA-ES and BSA. In particular, the comparison with some start-of-the-art algorithms on a test suite composed of five engineering constrained design problems demonstrates the superiority of the proposed algorithm.

MGPEA has two advantages. One is that it has only two parameters, i.e., the population size N and the length C of the IEC. And the parameter C is generally set to 4. This makes it easy to use. The other is that grey prediction theory can be the algorithmic theoretical basis, which could be used to design improved algorithms.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2020.106086.

CRediT authorship contribution statement

Xinlin Xu: Waiting - original draft. **Zhongbo Hu:** Methodology. **Qinghua Su:** Writing - review & editing. **Yuanxiang Li:** Supervision. **Jianhua Dai:** Investigation.

Acknowledgements

This work was supported in part by the National Nature Science Foundation of China (No. 61663009) and Heibei Province Department of Science and Technology Soft Science Project, China (No. 2018ADC068).

Appendix. Engineering design problems

A.1. Three-bar truss design problem

$$\min f(x) = (2\sqrt{2}x_1 + x_2) \times l$$

subject to:

$$g_1(x) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}\mathrm{P} - \sigma \le 0$$

$$g_2(x) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2}\mathrm{P} - \sigma \le 0$$

$$g_3(x) = \frac{1}{\sqrt{2}x_2 + x_1}\mathrm{P} - \sigma \le 0$$

$$0 \le x_i \le 1, \quad i = 1, 2$$

$$l = 100\,\mathrm{cm}, \quad \mathrm{P} = 2\,\mathrm{kN/cm^2}, \quad \sigma = 2\,\mathrm{kN/cm^2}$$

A.2. Pressure vessel design problem

$$\min f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

subject to:

$$g_1(x) = -x_1 + 0.0193x_3 \le 0$$

$$g_2(x) = -x_2 + 0.00954x_3 \le 0$$

$$g_3(x) = -\pi x_3^2x_4 - (4/3)\pi x_3^3 + 1296000 \le 0$$

$$g_4(x) = x_4 - 240 \le 0$$

$$0 \le x_i \le 100, \quad i = 1, 2$$

$$10 \le x_i \le 200, \quad i = 3, 4$$

A.3. Tension/compression spring design problem

$$\min f(x) = (x_3 + 2)x_2x_1^2$$

subject to:

$$g_1(x) = -x_2^3x_3/(71785x_1^4) + 1 \le 0$$

$$g_2(x) = (4x_2^2 - x_1x_2)/(12566(x_2x_1^3 - x_1^4)) + 1/(5108x_1^2) - 1 \le 0$$

$$g_3(x) = -140.45x_1/(x_2^2x_3) + 1 \le 0$$

$$g_4(x) = (x_1 + x_2)/1.5 - 1 \le 0$$

$$0.05 \le x_1 \le 2.00$$

$$0.25 \le x_2 \le 1.30$$

$$2.00 \le x_3 \le 15.00$$

A.4. Welded beam design problem

$$\min f(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14 + x_2)$$

subject to:

$$g_1(x) = \tau(x) - \tau_{max} \le 0$$
$$g_2(x) = \sigma(x) - \sigma_{max} \le 0$$
$$g_3(x) = x_1 - x_4 \le 0$$
$$g_4(x) = 0.10471x_1^2 + 0.04811x_3x_4(14 + x_2) - 5 \le 0$$
$$g_5(x) = 0.125 - x_1 \le 0$$
$$g_6(x) = \delta(x) - \delta_{max} \le 0$$
$$g_7(x) = P - P_c(x) \le 0$$
$$0.1 \le x_i \le 2, \quad i = 1, 4$$
$$0.1 \le x_i \le 10, \quad i = 2, 3$$

where

$$\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}, \quad \tau' = \frac{P}{\sqrt{2}x_1x_2}, \quad \tau'' = \frac{MR}{J}$$
$$M = P(L + \frac{x_2}{2}), \quad R = \sqrt{\frac{x_2^2}{4} + (\frac{x_1 + x_3}{2})^2},$$
$$J = 2\{\sqrt{2}x_1x_2[\frac{x_2^2}{12} + (\frac{x_1 + x_3}{2})^2]\}$$
$$\sigma(x) = \frac{6PL}{x_4x_3^2}, \quad \delta(x) = \frac{4PL^3}{Ex_3^3x_4},$$
$$P_c(x) = \frac{4.013E\sqrt{(x_3^2x_4^6/36)}}{L^2} \times (1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}})$$
$$P = 6000\,\text{lb}, \quad L = 14\,\text{in.}, \quad E = 30 \times 10^6\,\text{psi}, \quad G = 12 \times 10^6\,\text{psi}$$
$$\tau_{max} = 13\,600\,\text{psi}, \quad \sigma_{max} = 30\,000\,\text{psi}, \quad \delta_{max} = 0.25\,\text{in.}$$

A.5. Gear train design problem

$$\min f(x) = (\frac{1}{6.931} - \frac{x_3x_2}{x_1x_4})^2$$

subject to:

$$12 \le x_1, x_2, x_3, x_4 \le 60$$

References

[1] J.L. Deng, Control problems of grey systems, Systems Control Lett. 1 (1982) 288–294.

[2] J. Zhai, J.M. Sheng, Y.J. Feng, The grey model MGM (1, n) and its application, Syst. Eng.-Theory Pract. 17 (1997) 109–113, (in Chinese).

[3] D.Q. Truong, K.K. Ahn, Wave prediction based on a modified grey model MGM (1, 1) for real-time control of wave energy converters in irregular waves, Renew. Energy 43 (2012) 242–255.

[4] S. Mao, M. Guo, X. Xiao, M. Zhu, A novel fractional grey system model and its application, Appl. Math. Model. 40 (2016) 5063–5076.

[5] L. Wu, S. Liu, Y. Yang, Grey double exponential smoothing model and its application on pig price forecasting in China, Appl. Soft. Comput. 39 (2016) 117–123.

[6] Z.X. Wang, Q. Li, L.L. Pei, Grey forecasting method of quarterly hydropower production in china based on a data grouping approach, Appl. Math. Model. 51 (2017) 302–316.

[7] X. Ma, Z. Liu, Application of a novel time-delayed polynomial grey model to precict the natural gas consumption in China, J. Comput. Appl. Math. 324 (2017) 17–24.

[8] N. Xie, N. Chen, Flexible job shop scheduling problem with intervel grey processing time, Appl. Soft. Comput. 70 (2018) 513–524.

[9] B. Zeng, H. Duan, Y. Bai, W. Meng, Forecasting the output of shale gas in China using an unbiased grey model and weakening buffer operator, Energy 151 (2018) 238–249.

[10] X. Guo, S. S. Liu, L. Wu, et al., A multi-variable grey model with a self-memory component and its application on engineering prediction, Eng. Appl. Artif. Intell. 42 (2015) 82–93.

[11] B. Zeng, C. Lou, S. Liu, C. Li, A novel multi-variable grey forecasting model and its application in forecasting the amount of motor vehicles in Beijing, Comput. Ind. Eng. 101 (2015) 479–489.

[12] P.N. Suganthan, N. Hansen, J.J. Liang, et al., Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization, KanGAL report, 2005, p. 2005005.

[13] J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization, in: Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013, p. 635.

[14] J. Derrac, S. Garcła, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (2011) 3–18.

[15] C. Zhang, Q. Lin, L. Gao, X. Li, Backtracking search algorithm with three constraint handling methods for constrained optimization problems, Expert Syst. Appl. 42 (2015) 7831–7845.

[16] M. Zhang, W. Luo, X. Wang, Differential evolution with dynamic stochastic selection for constrained optimization, Inform. Sci. 178 (2008) 3043–3074.

[17] Y. Wang, Z. Cai, Y. Zhou, et al., Constrained optimization based on hybrid evolutionary algorithm and adaptive constraint-handling technique, Struct. Multidiscip. Optim. 37 (2009) 395–413.

[18] H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, Appl. Soft. Comput. 10 (2010) 629–640.

[19] L. Wang, L. Li, An effective differential evolution with level comparison for constrained engineering design, Struct. Multidiscip. Optim. 41 (2010) 947–963.

[20] A. Sadollah, A. Bahreininejad, H. Eskandar, M. Hamdi, Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems, Appl. Soft. Comput. 13 (2013) 2592–2612.

[21] A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm, Comput. Struct. 169 (2016) 1–12.

[22] T. Ray, K.M. Liew, Society and civilization: an optimization algorithm based on the simulation of social behavior, IEEE Trans. Evol. Comput. 7 (2003) 386–396.

[23] B.K. Kannan, S.N. Kramer, An augmented lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design, J. Mech. Des. 116 (1994) 405–411.

[24] C.A.C. Coello, Use of a self-adaptive penalty approach for engineering optimization problems, Comput. Ind. 41 (2000) 113–127.

[25] C.A.C. Coello, E.M. Montes, Constraint-handling in genetic algorithms through the use of dominance-based tournament selection, Adv. Eng. Inf. 16 (2002) 193–203.

[26] Q. He, L. Wang, An effective co-evolutionary particle swarm optimization for constrained engineering design problems, Eng. Appl. Artif. Intell. 20 (2007) 89–99.

[27] Q. He, L. Wang, A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization, Appl. Math. Comput. 186 (2007) 1407–1422.

[28] F.Z. Huang, L. Wang, Q. He, An effective co-evolutionary differential evolution for constrained optimization, Appl. Math. Comput. 186 (2007) 340–356.

[29] B. Akay, D. Karaboga, Artificial bee colony algorithm for large-scale problems and engineering design optimization, J. Intell. Manuf. 23 (2012) 1001–1014.

[30] E. Mezura-Montes, C.A.C. Coello, J. Velzquez-Reyes, Increasing successful offspring and diversity in differential evolution for engineering design, in: Proceedings of the Seventh International Conference on Adaptive Computing in Design and Manufacture, 2006, 131–139.