Cooperative Framework Fireworks Algorithm with Covariance Mutation

Chao Yu[1], Ling Chen Kelley[2,3], and Ying Tan[1,*]
[1] The Key Laboratory of Machine Perception and Intelligence (Ministry of Education), Department of Machine Intelligence, School of Electronics Engineering and Computer Science, Peking University, Beijing, China, 100871
[2] Department of History, School of Humanities, Tsinghua University. Beijing, China, 100084
[3] Department of World Languages and Cultures, College of Liberal Arts and Sciences, Auburn University at Montgomery, Montgomery, AL, United States 36117
Email: chaoyu@pku.edu.cn, lingchenkelley@gmail.com, ytan@pku.edu.cn

***Abstract*—Since fireworks algorithm (FWA) debuted in 2010, a dozen proposals of improvement for FWA had been published in an effort to enhance, refine and optimize accuracy while minimizing calculation speed and volume. In this paper, we introduce the covariance mutation operator into cooperative framework fireworks algorithm (CoFFWA) to create cooperative framework fireworks algorithm with covariance mutation (CoFFWA-CM) in order to solve the Congress on Evolutionary Computation (CEC 2016) competition functions on single objective optimization. The experimental results are calculated on all 10-, 30-, 50- and 100-dimensional functions.**

I. Introduction

Since its initial introduction by Tan and Zhu in 2010, FWA has witnessed several dozen designs that sought to improve, refine and optimize its output [1], [2]. FWA can be enhanced in three ways, as improvements in the explosion operator, the mutation operator and or the selection strategies.

The explosion operator was improved in 2013 by Zheng et al. that addressed to the weaknesses of conventional FWA and proposed the enhance FWA (EFWA) [3]. Liu et al. constructed a transfer function to generate sparks and proposed a constructed FWA (CFWA) [4]. In 2014, Zheng et al. improved the explosion amplitude of each firework and put forward dynamic search FWA (dynFWA) [5]. In 2015, Zheng et al. periodically reduced the number of explosion dimensions and put forward an exponentially decreased dimension FWA (eddynFWA) [6].

Enhancements of mutation operator include Yu et al, whom introduced the differential mutation operator into the EFWA and perpetuated a fireworks algorithm with differential mutation (FWA-DM) [7], [8], Li et al. calculated the difference values between the better and worse sparks and proposed orienting mutation based FWA (FWA-OM) [9]. The most effective improvement was the covariance mutation, which was proposed by Yu and Tan in 2015, called FWA with covariance mutation (FWACM) [10], [11].

The selection strategy was upgraded in 2012 by Pei et al, which utilized the approximation approaches to produce new sparks. If the fitness value of the new spark was better than the current selected spark, the new spark would be selected and passed down to the next generation. FWA was accelerated by this approximation method and the algorithm was named as accelerate FWA (AcFWA) [12]. In 2015, Zheng et al. invented an independent selection mechanism in the selection strategy. In addition, a crowd and bound back mechanism was introduced. When the selected sparks were too close, all but one would be generated randomly. The algorithm enhanced the cooperation between the sparks and called as the cooperative framework FWA (CoFFWA) [13].

Aside from the algorithmic improvements, FWA had been applied to many practical fields. Bacanin and Tuba applied FWA to solve the constrained portfolio optimization problems [14]. Bouarara et al. utilized FWA to solve the modern web information retrieval with visual results mining [15]. Rahmani et al. studied the privacy preservation problems in big data by FWA [16]. There were many other applications, as evidenced in article [17]–[19].

This study used CoFFWA with covariance mutation (CoFFWA-CM) to participate in the competition, as explained in section II [20]. By introducing the covariance mutation into the process of CoFFWA, the performance of the CoFFWA was further enhanced. The new mutation operator in CoFFWA-CM differed from the covariance mutation in FWACM while the new mutation operator is analyzed in section III. The experimental results are displayed in section IV, followed by discussion and conclusion.

II. Cooperative Framework Fireworks Algorithm with Covariance Mutation

CoFFWA-CM was proposed based on the previous work of CoFFWA and covariance mutation. Published in Oct, 2015, CoFFWA enhanced the diversity of the population by incorporating the independent selection strategy and bound back mechanism, thus outperforming both EFWA and dynFWA. In the same year, FWACM was put forward and defeated both EFWA and dynFWA. However, FWACM was based on the work of dynFWA, whereas dynFWA was not as good as CoFFWA. Therefore, covariance mutation was introduced to the most advanced FWA, which was CoFFWA, to proof its effectiveness.

CoFFWA-CM generated two kinds of sparks: explosion sparks from the explosion operator and mutation sparks from the covariance mutation. Similar to the FWA variants, the

* Prof. Y. Tan is the corresponding author.

sparks were then systematically evaluated and selected for the next generation. This iteration continued until the termination conditions were met, usually when the maximum number of function evaluations and or the accuracy requirements had been reached. CoFFWA-CM proposed to further improve accuracy with fewer generations. The covariance mutation was a mutation with Gaussian distribution and utilized the better sparks in each generation, instead of using the single best spark. The aim of covariance mutation was to use more effective information and thus increase the information utilization ratio.

CoFFWA-CM consisted of explosion operator, mutation operator, mapping rules and selection strategy. The details of them were as follows.

A. Explosion Operator

The explosion operator emulated a firework explosion, where sparks were generated encompassing a central firework. Before the explosion occurred, the number of sparks and the amplitude of the explosion were determined, and were generated in the same fashion as dynFWA, FWACM and CoFFWA. The explosion amplitude and the number of explosion of each firework all factored into the optimization results.

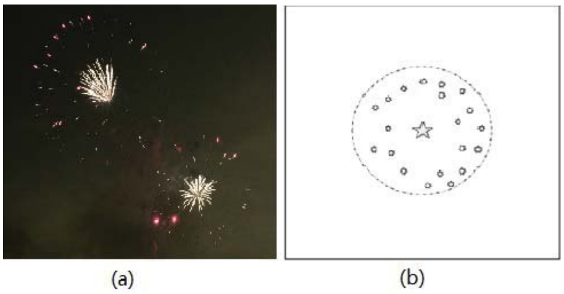

Fig. 1. Real fireworks explosion is illustrated in (a), and the explosion operation in FWA is shown in (b).

Before a firework exploded, the number of the sparks and the amplitude of the explosion were calculated in advance. In CoFFWA-CM, the number of sparks was inherited from the conventional FWA.

Let S_i denotes the number of sparks for the i^{th} firework.

$$S_i = \hat{S} * \frac{f(X_w) - f(X_i) + \varepsilon}{\sum_{i=1}^{N} (f(X_w) - f(X_i)) + \varepsilon}, \tag{1}$$

where the parameter $\hat{S}$ determines the sum of sparks in each generation. The Function $f(x)$ represents the fitness value of the input x. Since X_w is the individual with the worst fitness value, function $f(X_w)$ gives the fitness value of that individual X_w. The parameter N is the number of fireworks in a generation, whereas ε is used to prevent the denominator from becoming zero. There are lower and upper boundaries for S_i, which are set empirically.

The sign A_{CF} stands for the explosion amplitude of the core firework. When saying the 'core firework', it refers to the firework with the best fitness value in the current iteration. The calculation of the A_{CF} is as in Eq. 2.

$$A_{CF}(g) = \begin{cases} A_{CF}(1), if\ g = 1 \\ A_{CF}(g-1) * C_a, iff(X_{CF}(g)) < f(X_{CF}(g-1)) \\ A_{CF}(g-1) * C_r, otherwise \end{cases}, \tag{2}$$

where the parameter g is the number of generations, C_a and C_r stands for the amplification and reduction factors of the explosion amplitude, and $X_{CF}(g)$ is the core firework in the g^{th} generation.

Different from the core firework, the other fireworks have the traditional way to calculate the explosion amplitude, which is inherited from the conventional FWA. The amplitude of explosion of the i^{th} firework is denoted as A_i.

$$A_i = \hat{A} * \frac{f(X_i) - f(X_{CF}) + \varepsilon}{\sum_{i=1}^{N} (f(X_i) - f(X_{CF})) + \varepsilon}, \tag{3}$$

where the parameter $\hat{A}$ is a constant used to control the amplitude of explosions, X_i is the i^{th} individual. The parameter N and ε are the same as mentioned in Eq. 1.

Under the effect of the explosion operator, a firework produces a certain number of sparks within a preset amplitude, both of which are vital to the experimental results. However, compared to the number of sparks, the experimental results are much worse if the explosion amplitudes are set improperly. For example, if the number of sparks enlarges 10 times, the experimental results are The explosion amplitudes are always worth for studying.

B. Mutation Operator

In CoFFWA-CM, the covariance mutation utilizes information from the sparks generated by the core firework, instead of solely focusing on the single best spark. Unlike previous mutation operators, covariance mutation uses both the sparks produced by the fireworks with best fitness values from the current generation and from the single most recent generation to calculate an optimal solution. By using sparks from the previous generation, the covariant mutation selects better sparks with proper fitness values than previous algorithms.

The covariance mutation selects the sparks with better fitness values from the sparks produced by a firework, calculates the mean value of the selected sparks and the covariance matrix of all the sparks. With the mean value and covariance matrix, covariance mutation estimates the local distribution of a function and produces sparks according to normal distribution, aiming to find potential sparks with better fitness values. The covariance mutation contains three steps.

Firstly, the covariance mutation selects the sparks produced by a firework and calculates the mean value. Let the sign λ represents the sum of sparks and μ stands for the number of the selected sparks. The mean value of the selected sparks is represented as m.

$$m = \sum_{i=1}^{\mu} x_i, \tag{4}$$

where x_i is the i^{th} selected spark. Note that m is the mean value of the μ individuals, rather than the mean value of all individuals.

Secondly, the mutation operator calculates the covariance matrix C of all the λ sparks. The i^{th} row and j^{th} column of the matrix C is represented as C_{ij} in Eq. 5.

$$C_{ij} = \mathrm{cov}(d_i, d_j)(i, j = 1, ..., D), \tag{5}$$

where constant D is the dimension of the benchmark function and sample d_i is the sparks in their i^{th} dimension. The $cov(d_i, d_j)$ stands for the covariance of the sparks in i^{th} and j^{th} dimensions and it is calculated in Eq. 6.

$$\mathrm{cov}(d_i, d_j) = \frac{\sum_{k=1}^{\mu} (a_k - \bar{A})(b_k - \bar{B})}{\mu}. \tag{6}$$

where a_k and b_k are the k^{th} spark in its i^{th} and j^{th} dimensions, $\bar{A}$ and $\bar{B}$ are the mean value of all the λ sparks in dimension i and j. Different from calculating the covariance, the denominator here is not $\mu - 1$ as usual.

Thirdly, the mutation sparks are generated according to normal distribution, using the m and C obtained before. Figure 2 shows the contour of the mutation sparks in ellipse.

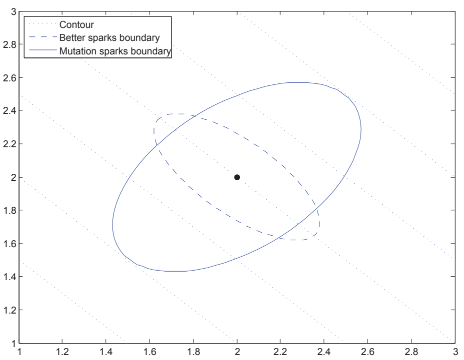

Fig. 2. The mutation sparks distribution with $N(m, C)$.

It can be seen from Fig. 2 that the possibility of finding useful sparks increased, as the mutation sparks produced by covariance mutation can now better navigate towards the optimal direction.

The algorithm of the covariance mutation is given in Alg. 1. The two groups of sparks are produced by the current and the most recent generation best sparks. The sign $N(m, C)$ denotes a normal distribution with mean m and covariance C.

Algorithm 1 Covariance Mutation

```
collect two groups of sparks
calculate the mean value m of the better sparks
obtain the covariance matrix C
generate mutation sparks with N(m, C)
```

C. Mapping Rules

Mapping rules are proposed to handle sparks that fell outside of the boundaries. When a firework is close to the boundary, the generated sparks can easily stray outside of the boundary when distributed by a relatively large explosion amplitude. In CoFFWA-CM, the mapping rules are the same as in conventional FWA, where the outlying sparks are generated randomly back into the feasible space.

$$X = X_l + rand(0, 1) * (X_u - X_l), \tag{7}$$

where X is the location of the spark X in the feasible space, X_u and X_l denote the upper and lower boundaries. Function $rand(0, 1)$ generates a random number within the area from 0 to 1 with uniform distribution.

D. Selection Strategies

The two strategies used in CoFFWA-CM are independent selection strategy and the crowd bounce off strategy. The independent selection strategy selects N individuals independently of the next generation, where N fireworks produce N groups of fireworks, followed by the best spark in each group being selected. The crowd and bounce off strategy filters the selected sparks. If two selected sparks are too close, then one of them is bounced off randomly to another location. The effectiveness of these two strategies are carefully researched in the reference [13].

To algorithm of CoFFWA-CM is shown in Alg. 2.

Algorithm 2 The algorithm of CoFFWA-CM

```
generate N fireworks with uniform distribution randomly.
evaluate the fitness values of the N fireworks
while terminate condition is not met do
    calculate the number of sparks S_i
    calculate the amplitude of explosion A_CF and A_i
    generate S_i explosion sparks within the amplitudes
    calculate m and covariance matrix C
    generate Gaussian sparks according with N(m, C)
    evaluate all the fitness values of explosion and Gaussian sparks
    keep N individuals for the next generation using selection strategies
end while
return the best individual and its fitness value
```

III. Analysis of the Mutation in CoFFWA-CM

Covariance mutation is an implementation of CoFFWA. In CoFFWA, there is no mutation operator, as such, it is easy to add the mutation operator to the CoFFWA without changing the original structure. The new algorithm firstly utilized the explosion operator, followed by the mutation operator and the selection strategies. If any sparks are produced out of the boundaries, they will be discarded and a new spark will be generated randomly within the feasible space.

A two-dimensional function $f(x) = x_1 + x_2$ is taken as an example to visualize the covariance mutation in CoFFWA.

In Fig. 3, the processes of generating mutation sparks by covariance mutation is shown one by one. The contour lines is drawn in Fig. 3(a), where the bottom left indicates better fitness values. Figure 3(b) illustrates the selected λ sparks. The better sparks are shown in Fig. 3(c) marked as black 'x'. The mutation sparks are represented in Fig. 3(d) marked with red '+'.

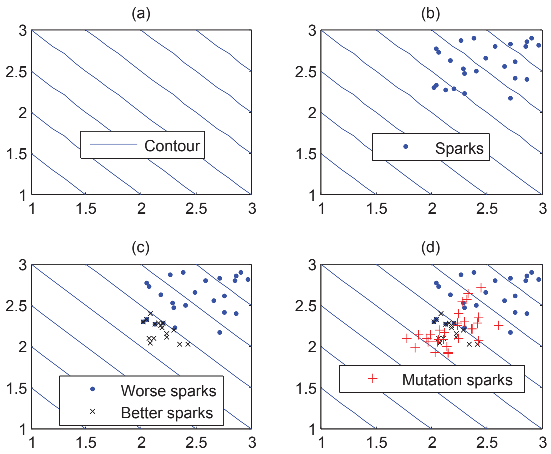

Fig. 3. The process of generating mutation sparks by covariance mutation.

It can be seen from Fig. 3 that the sparks marked with red '+' distributed in a direction from the bottom left to the upper right, which is close to the direction of the gradient. Besides, the relatively better sparks in Fig. 3d are all produced by covariance mutation, which are marked with red '+'. Therefore, the covariance mutation operator is effective at finding local optimal values with the information of the sparks. The characteristic of covariance mutation can be extended to higher dimensions and more complicated functions.

To validate the performance of covariance mutation, the rotated high conditioned elliptic function was utilized. This function was the number 1 function in CEC 2016 single objective optimization function, which was a unimodal function. The comparison experiments show the CoFFWA with and without covariance mutation. All of the four dimensions were considered, as 10, 30 and 50 dimensions. Each algorithm ran for 51 times and the average mean errors were recorded.

TABLE I. THE COMPARISON OF COFFWA WITH COFFWA-CM ON FUNCTION 1 IN 10, 30 AND 50 DIMENSIONS

Dimensions	CoFFWA	CoFFWA-CM
10	4.39e+04	**2.16e+04**
30	1.12e+06	**6.24e+05**
50	2.12e+06	**1.61e+06**

The experimental results in Table I indicated that when the dimension increased to 50, CoFFWA-CM still had the better performance than the original CoFFWA. Note that the parameters chosen for CoFFWA were its default parameters, whereas the parameters of CoFFWA-CM were the same as indicated in the experiments section. The better performance of CoFFWA-CM proof its effectiveness on 10, 30 and 50 dimensional of a unimodal function.

IV. EXPERIMENTS

This part first introduce the benchmark functions, followed by the parameter settings and experimental results. Note that a value of 100 was added to function 1, 200 to function 2, ..., and a value of 3000 to function 30, all of which were not subtracted. In this way, the experimental results were displayed as they were calculated from their corresponding benchmark functions.

A. Benchmark Functions

CoFFWA-CM was used to find the global optimum values of 30 benchmark functions from the CEC'16 competition. The details of the functions could be found in [21]. The names and numbers of the functions were given in Table II. The functions included unimodal, multi-modal, hybrid and composition functions, which were divided by horizontal lines in Table II.

TABLE II. BENCHMARK FUNCTIONS

No.	Name
1	Rotated High Conditioned Elliptic Function
2	Rotated Bent Cigar Function
3	Rotated Discus Function
4	Shifted and Rotated Rosenbrocks Function
5	Shifted and Rotated Ackley's Function
6	Shifted and Rotated Weierstrass Function
7	Shifted and Rotated Griewank's Function
8	Shifted Rastrigin's Function
9	Shifted and Rotated Rastrigin's Function
10	Shifted Schwefel's Function
11	Shifted and Rotated Schwefel's Function
12	Shifted and Rotated Katsuura Function
13	Shifted and Rotated HappyCat Function
14	Shifted and Rotated HGBat Function
15	Shifted and Rotated Expanded Griewank's plus Rosenbrock's Functions
16	Shifted and Rotated Expanded Scaffer's F6 Function
17	Hybrid Function 1 (N=3)
18	Hybrid Function 2 (N=3)
19	Hybrid Function 3 (N=4)
20	Hybrid Function 4 (N=4)
21	Hybrid Function 5 (N=5)
22	Hybrid Function 6 (N=5)
23	Composition Function 1 (N=5)
24	Composition Function 2 (N=3)
25	Composition Function 3 (N=3)
26	Composition Function 4 (N=5)
27	Composition Function 5 (N=5)
28	Composition Function 6 (N=5)
29	Composition Function 7 (N=3)
30	Composition Function 8 (N=3)

B. Parameters Settings

The parameters were set identically for all the dimensions. Some of the parameters' settings were shown in Table. III. It was clearly that the parameters were simple and easy to set. The other parameters are demonstrated in [13].

TABLE III. PART OF THE PARAMETERS' SETTING

Parameters	Values	Meanings
N	5	the number of fireworks
$\hat{S}$	150	the number of sparks
$\hat{A}$	40	the amplitude constant
μ/λ	0.25	the percentage of better sparks

C. Experimental Results

The experimental platform is Matlab 2015a and the program is run on a Windows 10 operating system. The experimental results on 10, 30, 50 and 100-dimension functions of CEC 2016 single objective optimization competition are given in Table IV, V, VI and VII, respectively.

The computational complexity of CoFFWA-CM is given in Table VIII.

TABLE IV. RESULTS FOR 10D

Function No.	Best	Worst	Median	Mean	Std.
1	1.75E+03	5.49E+04	1.97E+04	2.16E+04	1.37E+04
2	2.01E+02	9.73E+03	1.74E+03	2.57E+03	2.45E+03
3	3.00E+02	3.90E+02	3.01E+02	3.08E+02	1.55E+01
4	4.00E+02	4.35E+02	4.00E+02	4.04E+02	9.27E+00
5	5.20E+02	5.20E+02	5.20E+02	5.20E+02	1.40E-04
6	6.00E+02	6.05E+02	6.01E+02	6.01E+02	1.18E+00
7	7.00E+02	7.00E+02	7.00E+02	7.00E+02	7.52E-02
8	8.03E+02	8.16E+02	8.06E+02	8.07E+02	3.64E+00
9	9.06E+02	9.45E+02	9.17E+02	9.17E+02	7.73E+00
10	1.00E+03	1.36E+03	1.13E+03	1.11E+03	9.61E+01
11	1.13E+03	2.07E+03	1.62E+03	1.60E+03	2.26E+02
12	1.20E+03	1.20E+03	1.20E+03	1.20E+03	4.64E-02
13	1.30E+03	1.30E+03	1.30E+03	1.30E+03	5.65E-02
14	1.40E+03	1.40E+03	1.40E+03	1.40E+03	6.63E-02
15	1.50E+03	1.50E+03	1.50E+03	1.50E+03	3.00E-01
16	1.60E+03	1.60E+03	1.60E+03	1.60E+03	4.72E-01
17	1.75E+03	6.25E+03	2.41E+03	2.84E+03	1.03E+03
18	1.81E+03	6.68E+03	1.99E+03	2.25E+03	7.46E+02
19	1.90E+03	1.90E+03	1.90E+03	1.90E+03	4.12E-01
20	2.01E+03	3.04E+03	2.08E+03	2.17E+03	2.11E+02
21	2.10E+03	2.91E+03	2.28E+03	2.36E+03	2.04E+02
22	2.20E+03	2.36E+03	2.22E+03	2.23E+03	2.69E+01
23	2.30E+03	2.63E+03	2.63E+03	2.61E+03	6.35E+01
24	2.51E+03	2.54E+03	2.52E+03	2.53E+03	8.34E+00
25	2.62E+03	2.70E+03	2.65E+03	2.66E+03	2.98E+01
26	2.70E+03	2.70E+03	2.70E+03	2.70E+03	6.76E-02
27	2.70E+03	3.10E+03	2.71E+03	2.87E+03	1.87E+02
28	3.10E+03	3.38E+03	3.26E+03	3.25E+03	6.18E+01
29	3.16E+03	4.03E+03	3.35E+03	3.39E+03	1.59E+02
30	3.31E+03	4.44E+03	3.97E+03	3.97E+03	2.73E+02

TABLE V. RESULTS FOR 30D

Function No.	Best	Worst	Median	Mean	Std.
1	1.32E+05	1.98E+06	5.74E+05	6.24E+05	3.67E+05
2	2.04E+02	3.18E+04	9.41E+03	1.03E+04	8.47E+03
3	3.00E+02	3.00E+02	3.00E+02	3.00E+02	0.00E+00
4	4.00E+02	5.43E+02	4.74E+02	4.62E+02	4.52E+01
5	5.20E+02	5.20E+02	5.20E+02	5.20E+02	0.00E+00
6	6.00E+02	6.13E+02	6.06E+02	6.06E+02	3.33E+00
7	7.00E+02	7.00E+02	7.00E+02	7.00E+02	9.78E-03
8	8.42E+02	9.25E+02	8.80E+02	8.83E+02	2.09E+01
9	9.40E+02	1.03E+03	9.88E+02	9.85E+02	2.59E+01
10	2.36E+03	4.90E+03	3.60E+03	3.56E+03	5.05E+02
11	2.36E+03	5.40E+03	4.14E+03	4.06E+03	6.17E+02
12	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.04E-01
13	1.30E+03	1.30E+03	1.30E+03	1.30E+03	7.09E-02
14	1.40E+03	1.40E+03	1.40E+03	1.40E+03	3.54E-02
15	1.50E+03	1.51E+03	1.50E+03	1.50E+03	1.93E+00
16	1.61E+03	1.61E+03	1.61E+03	1.61E+03	4.83E-01
17	5.49E+03	1.23E+05	1.61E+04	3.14E+04	3.35E+04
18	2.12E+03	1.04E+04	3.50E+03	4.26E+03	2.08E+03
19	1.91E+03	1.92E+03	1.91E+03	1.91E+03	2.39E+00
20	2.09E+03	2.40E+03	2.22E+03	2.22E+03	7.22E+01
21	5.89E+03	3.62E+04	1.53E+04	1.57E+04	6.60E+03
22	2.24E+03	2.99E+03	2.60E+03	2.57E+03	1.64E+02
23	2.62E+03	2.62E+03	2.62E+03	2.62E+03	7.79E-03
24	2.62E+03	2.64E+03	2.63E+03	2.63E+03	4.04E+00
25	2.70E+03	2.71E+03	2.71E+03	2.71E+03	2.22E+00
26	2.70E+03	2.70E+03	2.70E+03	2.70E+03	1.04E-01
27	3.00E+03	3.26E+03	3.10E+03	3.12E+03	4.61E+01
28	3.67E+03	5.99E+03	3.86E+03	4.02E+03	4.18E+02
29	5.19E+03	1.63E+04	1.02E+04	1.03E+04	3.40E+03
30	5.22E+03	1.52E+04	7.63E+03	7.94E+03	1.85E+03

V. DISCUSSION

The covariance mutation is effective in estimating local distribution in all functions. In CoFFWA-CM, each group of sparks is produced around a firework, thus the local distribution can be estimated if the information of the sparks was utilized. The information includes the location of the sparks that are geologically closest and the fitness values of these sparks. Early in 2012, Pei et al. proposed an approximate method to estimate the local distribution of the functions, but now the covariance mutation is used. There are other ways to estimate the local distribution of the functions, such as sampling and

TABLE VI. RESULTS FOR 50D

Function No.	Best	Worst	Median	Mean	Std.
1	7.16E+05	3.10E+06	1.56E+06	1.61E+06	5.44E+05
2	2.00E+02	2.59E+04	5.25E+03	8.25E+03	8.45E+03
3	3.00E+02	3.00E+02	3.00E+02	3.00E+02	0.00E+00
4	4.23E+02	6.00E+02	4.99E+02	5.06E+02	2.83E+01
5	5.20E+02	5.20E+02	5.20E+02	5.20E+02	1.40E-04
6	6.04E+02	6.28E+02	6.14E+02	6.14E+02	5.27E+00
7	7.00E+02	7.00E+02	7.00E+02	7.00E+02	7.87E-03
8	8.97E+02	1.05E+03	9.66E+02	9.73E+02	4.07E+01
9	9.91E+02	1.18E+03	1.06E+03	1.06E+03	3.84E+01
10	4.94E+03	8.16E+03	6.23E+03	6.24E+03	8.06E+02
11	4.93E+03	8.88E+03	6.74E+03	6.79E+03	8.99E+02
12	1.20E+03	1.20E+03	1.20E+03	1.20E+03	1.10E-01
13	1.30E+03	1.30E+03	1.30E+03	1.30E+03	5.87E-02
14	1.40E+03	1.40E+03	1.40E+03	1.40E+03	3.00E-02
15	1.51E+03	1.52E+03	1.51E+03	1.51E+03	2.98E+00
16	1.62E+03	1.62E+03	1.62E+03	1.62E+03	7.69E-01
17	2.10E+04	1.81E+05	6.33E+04	7.34E+04	4.15E+04
18	2.30E+03	8.81E+03	2.74E+03	3.28E+03	1.23E+03
19	1.92E+03	1.98E+03	1.93E+03	1.93E+03	1.32E+01
20	2.26E+03	2.63E+03	2.43E+03	2.44E+03	8.81E+01
21	1.51E+04	2.30E+05	7.24E+04	8.14E+04	5.29E+04
22	2.53E+03	3.74E+03	3.15E+03	3.12E+03	2.27E+02
23	2.64E+03	2.65E+03	2.65E+03	2.65E+03	8.35E-01
24	2.66E+03	2.69E+03	2.68E+03	2.68E+03	5.24E+00
25	2.71E+03	2.73E+03	2.72E+03	2.72E+03	3.74E+00
26	2.70E+03	2.80E+03	2.80E+03	2.77E+03	4.81E+01
27	3.16E+03	3.81E+03	3.41E+03	3.44E+03	1.42E+02
28	4.62E+03	8.92E+03	5.97E+03	6.07E+03	9.73E+02
29	9.22E+03	7.90E+04	2.41E+04	2.64E+04	1.27E+04
30	3.09E+04	7.67E+04	4.15E+04	4.42E+04	9.36E+03

TABLE VII. RESULTS FOR 100D

Function No.	Best	Worst	Median	Mean	Std.
1	6.27E+06	2.58E+07	1.23E+07	1.32E+07	4.93E+06
2	2.01E+02	1.27E+05	1.33E+04	2.77E+04	3.02E+04
3	3.00E+02	3.00E+02	3.00E+02	3.00E+02	1.96E-04
4	5.60E+02	7.58E+02	6.59E+02	6.62E+02	4.24E+01
5	5.20E+02	5.20E+02	5.20E+02	5.20E+02	0.00E+00
6	6.30E+02	6.75E+02	6.47E+02	6.48E+02	8.67E+00
7	7.00E+02	7.00E+02	7.00E+02	7.00E+02	3.94E-03
8	1.09E+03	1.38E+03	1.25E+03	1.24E+03	6.48E+01
9	1.19E+03	1.60E+03	1.32E+03	1.33E+03	8.39E+01
10	1.05E+04	1.60E+04	1.31E+04	1.30E+04	1.23E+03
11	1.10E+04	1.68E+04	1.38E+04	1.39E+04	1.41E+03
12	1.20E+03	1.20E+03	1.20E+03	1.20E+03	8.57E-02
13	1.30E+03	1.30E+03	1.30E+03	1.30E+03	5.37E-02
14	1.40E+03	1.40E+03	1.40E+03	1.40E+03	2.61E-02
15	1.51E+03	1.54E+03	1.53E+03	1.53E+03	5.73E+00
16	1.64E+03	1.65E+03	1.64E+03	1.64E+03	1.13E+00
17	1.95E+05	1.81E+06	5.06E+05	6.33E+05	3.61E+05
18	2.83E+03	1.04E+04	3.86E+03	4.49E+03	1.81E+03
19	1.98E+03	2.05E+03	2.03E+03	2.02E+03	2.16E+01
20	2.50E+03	2.99E+03	2.76E+03	2.76E+03	1.19E+02
21	3.96E+04	1.04E+06	2.08E+05	2.64E+05	2.09E+05
22	2.87E+03	4.86E+03	4.04E+03	4.03E+03	3.83E+02
23	2.66E+03	2.67E+03	2.66E+03	2.66E+03	2.93E+00
24	2.79E+03	2.82E+03	2.80E+03	2.80E+03	6.73E+00
25	2.74E+03	2.78E+03	2.76E+03	2.76E+03	7.82E+00
26	2.80E+03	2.80E+03	2.80E+03	2.80E+03	1.05E-01
27	3.64E+03	4.98E+03	4.15E+03	4.15E+03	2.78E+02
28	8.53E+03	1.48E+04	1.22E+04	1.23E+04	1.43E+03
29	1.17E+04	5.96E+04	2.34E+04	2.50E+04	9.09E+03
30	4.68E+04	1.26E+05	7.97E+04	8.07E+04	1.65E+04

TABLE VIII. COMPUTATIONAL COMPLEXITY OF COFFWA-CM GIVEN FOR 10, 30, 50 AND 100 DIMENSIONAL FUNCTION 18

	T0	T1	T2	(T2-T1)/T0
D=10	0.129149	0.112624	0.916262	6.222565
D=30	0.129149	0.360556	1.381998	7.909020
D=50	0.129149	0.696860	1.666858	7.510689
D=100	0.129149	2.065286	3.545964	11.464880

based on the probabilistic distribution model, all of which are worth researching in the future.

As there is no comparison algorithm, the performance can be analyzed if the global best value has been found. From the experimental results of 10-dimensional functions, the optimal values were found on function 7, 12, 13, 14, 15, 16 and 19. However, when the values were subtracted and the optimal values became zero, the algorithm found no value that equals zero. The experimental results on the 3rd function in its 30-, 50- and 100-dimension were great, as the algorithm found all the values as zero. Therefore, even the global optimal values were not found on most functions, it didn't necessary mean the algorithm performed poorly. The only reasonable explanation was the functions that were complicated and the global optimal values of the functions were hard to pinpoint.

Another way to measure the experimental results was to judge the standard deviation. From the experimental results of 10-dimensional functions, 9 of the 30 standard deviation were lower than 1. This meant the results were steady with a relatively small variation after 51 cycles. The experimental results on the 30-dimensional functions were different. The algorithm found the global optimal value on function 3 and the local optimal value on function 5, both of the standard deviations were zero. It can be seen from to the experimental results on 50 dimensional functions that the algorithm found the global best value on function 3, whereas the algorithm found the local optimal on function 5 in 100-dimension functions. Both of the standard deviations were zero.

Although the overall experimental results were acceptable, there is still much space for improvement of the algorithms to ascertain better results. CoFFWA-CM uses sparks with better fitness values from previous generations with the current generation. However, it would be an interesting study to weigh how many previous generations could be used to refine the most recent spark before the refinement process becomes moot. Nevertheless, we expect to see progressive research to improve CoFFWA-CM in the near future.

VI. Conclusion

This paper utilized the most advanced FWA variant to participate the CEC 2016 single objective optimization problems. The CoFFWA-CM utilized the covariance mutation and increased the local search ability. By successfully integrating the covariant mutation into CoFFWA, CoFFWA-CM further enhanced the accuracy of CoFFWA. CoFFWA-CM outperformed its predecessors on all accounts and has achieved better accuracy while minimizing computation time.

Acknowledgment

This work was supported by the Natural Science Foundation of China (NSFC) under grant no. 61375119 and the Beijing Natural Science Foundation under grant no. 4162029, and partially supported by National Key Basic Research Development Plan (973 Plan) Project of China under grant no. 2015CB352302.

References

[1] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in *Advances in Swarm Intelligence*, pp. 355–364, Springer, 2010.

[2] Y. Tan, *Fireworks Algorithm*. Springer-Verlag Berlin Heidelberg, 2015.

[3] S. Zheng, A. Janecek, and Y. Tan, "Enhanced fireworks algorithm," in *Evolutionary Computation (CEC), 2013 IEEE Congress on*, pp. 2069–2077, IEEE, 2013.

[4] J. Liu, S. Zheng, and Y. Tan, "The improvement on controlling exploration and exploitation of firework algorithm," in *Advances in Swarm Intelligence* (Y. Tan, Y. Shi, and H. Mo, eds.), vol. 7928 of *Lecture Notes in Computer Science*, pp. 11–23, Springer Berlin Heidelberg, 2013.

[5] S. Zheng, A. Janecek, J. Li, and Y. Tan, "Dynamic search in fireworks algorithm," in *Evolutionary Computation (CEC), 2014 IEEE Congress on*, pp. 3222–3229, IEEE, 2014.

[6] S. Zheng, C. Yu, J. Li, and Y. Tan, "Exponentially decreased dimension number strategy based dynamic search fireworks algorithm for solving cec2015 competition problems," in *Evolutionary Computation (CEC), 2015 IEEE Congress on*, pp. 1083–1090, IEEE, May 2015.

[7] C. Yu, J. Li, and Y. Tan, "Improve enhanced fireworks algorithm with differential mutation," in *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pp. 264–269, IEEE, 2014.

[8] C. Yu, L. Kelley, S. Zheng, and Y. Tan, "Fireworks algorithm with differential mutation for solving the cec 2014 competition problems," in *Evolutionary Computation (CEC), 2014 IEEE Congress on*, pp. 3238–3245, IEEE, 2014.

[9] J. Li and Y. Tan, "Orienting mutation based fireworks algorithm," in *Evolutionary Computation (CEC), 2015 IEEE Congress on*, pp. 1265–1271, IEEE, May 2015.

[10] C. Yu and Y. Tan, "Fireworks algorithm with covariance mutation," in *Evolutionary Computation (CEC), 2015 IEEE Congress on*, IEEE, 2015. (Under review).

[11] C. Yu, L. C. Kelley, and Y. Tan, "Dynamic search fireworks algorithm with covariance mutation for solving the cec 2015 learning based competition problems," in *Evolutionary Computation (CEC), 2015 IEEE Congress on*, pp. 1106–1112, IEEE, May 2015.

[12] Y. Pei, S. Zheng, Y. Tan, and H. Takagi, "An empirical study on influence of approximation approaches on enhancing fireworks algorithm," in *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pp. 1322–1327, IEEE, October 2012.

[13] S. Zheng, J. Li, A. Janecek, and Y. Tan, "A cooperative framework for fireworks algorithm," *arXiv preprint arXiv:1505.00075*, May 2015.

[14] N. Bacanin and M. Tuba, "Fireworks algorithm applied to constrained portfolio optimization problem," in *Evolutionary Computation (CEC), 2015 IEEE Congress on*, pp. 1242–1249, May 2015.

[15] H. A. Bouarara, R. M. Hamou, A. Amine, and A. Rahmani, "A fireworks algorithm for modern web information retrieval with visual results mining," *International Journal of Swarm Intelligence Research (IJSIR)*, vol. 6, no. 3, pp. 1–23, 2015.

[16] A. Rahmani, A. Amine, R. M. Hamou, M. E. Rahmani, and H. A. Bouarara, "Privacy preserving through fireworks algorithm based model for image perturbation in big data," *International Journal of Swarm Intelligence Research (IJSIR)*, vol. 6, no. 3, pp. 41–58, 2015.

[17] M. Tuba, N. Bacanin, and M. Beko, "Fireworks algorithm for rfid network planning problem," in *Radioelektronika (RADIOELEKTRONIKA), 2015 25th International Conference*, pp. 440–444, IEEE, April 2015.

[18] M. Tuba, N. Bacanin, and A. Alihodzic, "Multilevel image thresholding by fireworks algorithm," in *Radioelektronika (RADIOELEKTRONIKA), 2015 25th International Conference*, pp. 326–330, IEEE, April 2015.

[19] S. Ye, H. Ma, S. Xu, W. Yang, and M. Fei, "An effective fireworks algorithm for warehouse-scheduling problem," *Transactions of the Institute of Measurement and Control*, pp. 1–11, August 2015.

[20] C. Yu and Y. Tan, "Cooperative fireworks algorithm with covariance mutation." Submitted.

[21] J. Liang, B. Qu, and P. Suganthan, "Problem definitions and evaluation criteria for the cec 2014 special session and competition on single objective real-parameter numerical optimization," *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore*, 2013.

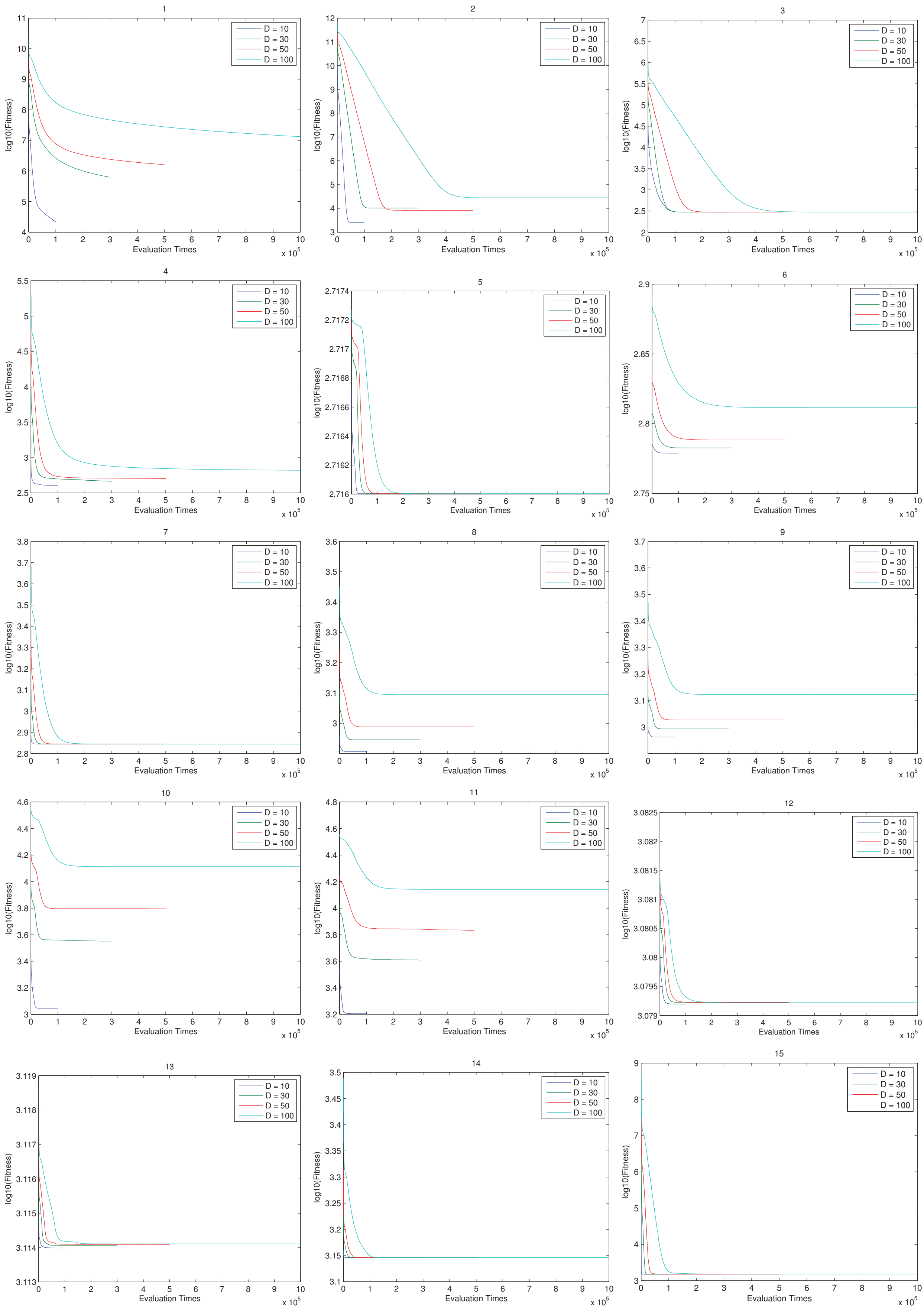

Fig. 4. Convergence curves for CoFFWA-CM on function 1 to 15 in Dimension 10, 30, 50 and 100.

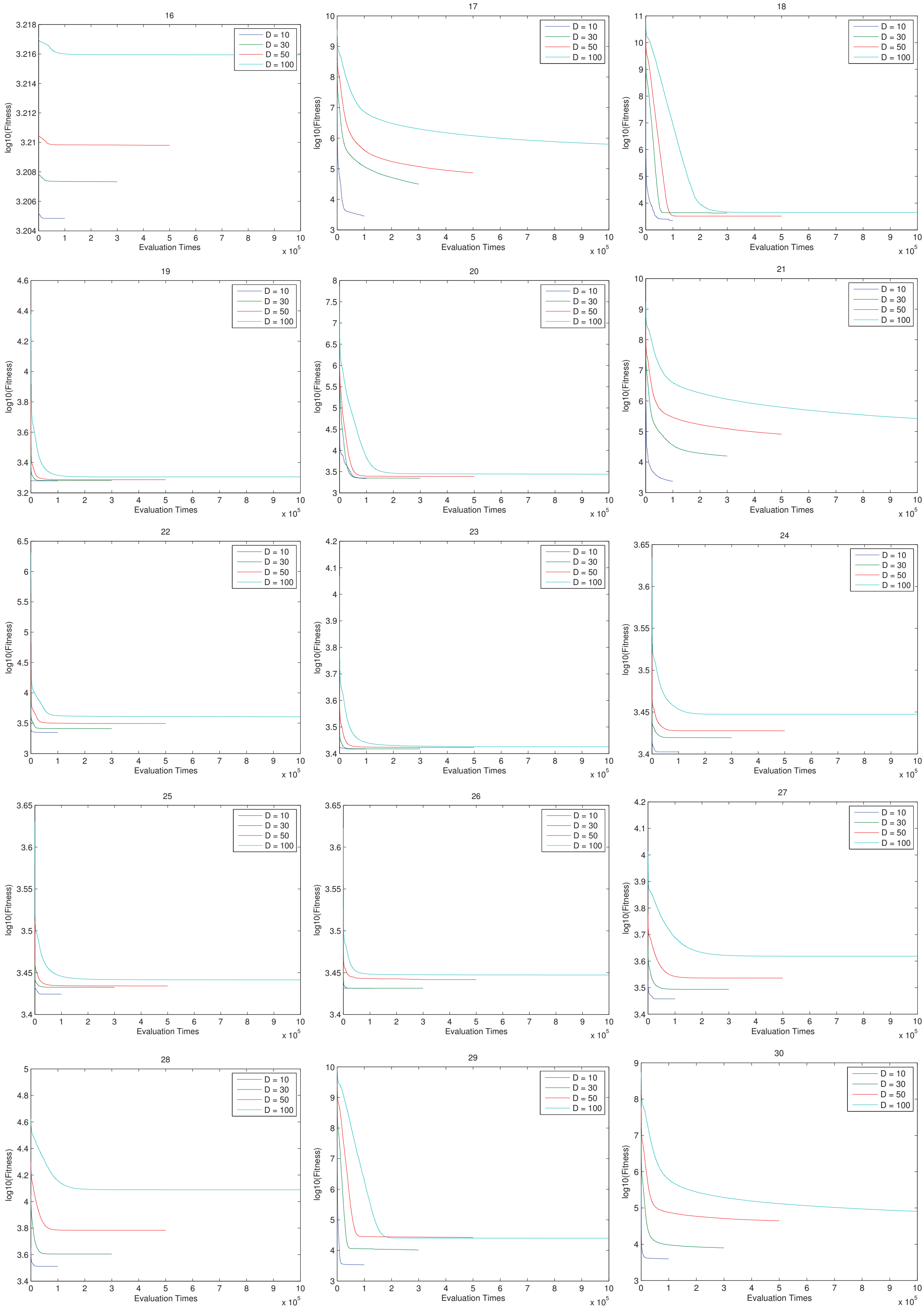

Fig. 5. Convergence curves for CoFFWA-CM on function 16 to 20 in Dimension 10, 30, 50 and 100.