Exponentially Decaying Explosion in Fireworks Algorithm

Maiyue Chen
Key laboratory of Machine Perception (MOE)
Department of Machine Intelligence
School of Electronics Engineering and Computer Science
Peking University
Bejing, China
mychen@pku.edu.cn

Ying Tan
Key laboratory of Machine Perception (MOE)
Department of Machine Intelligence
School of Electronics Engineering and Computer Science
Peking University
Bejing, China
ytan@pku.edu.cn (Corresponding author)

2021 IEEE Congress on Evolutionary Computation (CEC) | 978-1-7281-8393-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/CEC45853.2021.9504974

***Abstract*—Fireworks algorithm (FWA) as an efficient and robust swarm intelligence algorithm can successfully deal with complex multi-modal problems. In this paper, a novel new explosion operator called exponentially decaying explosion is proposed to enhance the local search ability of fireworks algorithm based on the principle of utilizing more information. The proposed method takes the idea of guided mutation a step further and dismantled the explosion process into an exponentially decaying series of guided explosion. The FWA variant with this explosion operator is called exponentially decaying fireworks algorithm (EDFWA). Theoretical analysis proved the superiority of EDFWA in terms of information utilization ratio compared with GFWA. Experimental results showed that EDFWA not only surpassed LoTFWA in low dimensional situations, but also exhibited powerful searching capability on 1000 dimensional high dimensional problems compared with multiple representative optimizers specially designed for large-scale problems.**

***Index Terms*—Fireworks Algorithms, Large-scale Optimization, Multi-modal Optimization, Swarm Intelligence, Evolutionary Algorithm, Information Utilization, Exponential Decay**

I. INTRODUCTION

Fireworks algorithm (FWA) is a swarm intelligence algorithm proposed by Tan et al. [1] in 2010. The general framework of FWA consists of two layers, one is the global coordination between fireworks, the other is the local search of fireworks. This hierarchical architecture gives FWA the ability to adapt to a wide range of complex and difficult multi-modal problems. As its name implies, FWA features a local search procedure in which each firework generates a group of sparks as candidates with a uniform distribution. This design choice on the one hand makes FWA very robust and stable, but on the other hand can restrict the local search ability of FWA to some extent, since the uniform distribution incorporates no bias of the local landscape of the function.

Some efforts have been made to enhance the local search ability of FWA, and the most notable one is guided FWA (GFWA) [2]. The main principle proposed by GFWA is to increase the information utilization ratio in each generation, and this principle leads to the mechanism called guiding spark. The introduction of guiding spark greatly increases the local search ability of FWA as it can capture the local feature of the function to certain extent. However, according to the idea of utilizing more information, guiding spark still hasn't utilized enough information acquired by sparks, and thus the local search ability still has room for further improvement.

This work extends the idea of the guiding spark in GFWA a step further. Instead of guiding the uniform search with one single guiding spark, a mechanism called exponentially decaying explosion is proposed and the resulting algorithm is called exponentially decaying fireworks algorithm (EDFWA). During each generation of EDFWA, a series of guided uniform distributed sparks are sampled and the magnitude as well as the spark number decays in exponential manner. This mechanism can enhance the local search ability of fireworks algorithm, especially in high dimension large-scale problems.

In the following section, we provide essential background information and related works in section II, where the main procedure of FWA and some notable FWA variants including GFWA and LoTFWA [2] [17] are introduced. Section III contains detailed description and analysis of our proposed method. Then in section IV we'll present the experimental results to showcase the performance of EDFWA on two benchmark suites: CEC2017 real value bounded benchmark [14] and CEC2013 large-scale benchmark [18]. Finally we conclude our work in section V.

II. RELATED WORKS

A. Fireworks Algorithm Framework

Inspired by the real-world fireworks explosion, FWA initializes multiple fireworks which iteratively conducts explosion and selection operation to find the optimal solution. In the explosion process, each fireworks generate its explosion sparks around it according to an explosion operator. Then those generated sparks together with their parents (fireworks) go through a selection phase where fireworks for the next generation will be selected. The most conventional selection strategy resembles Evolution Strategy, where every individual is included in a single selection pool. However, later research [17] indicated that letting each firework form their own selection pool is a better choice in the case of multi-modal optimization. Apart from selection, fireworks also cooperate with each other according to global cooperation strategies. These strategies can be very versatile, for example, the number of sparks can be globally coordinated [17] [7], the restart schedule can be designed [17], etc. As a matter of fact, the overall framework of fireworks algorithm can be abstracted into two main procedures: 1. Conduct local search by explosion. 2. Conduct global coordination of local explosions [21]. These two principle together enable fireworks algorithm to solve a large variety of optimization problems efficiently. Figure 1 provides a intuition of the framework of FWA. Algorithm 1 describes the main procedure of FWA.

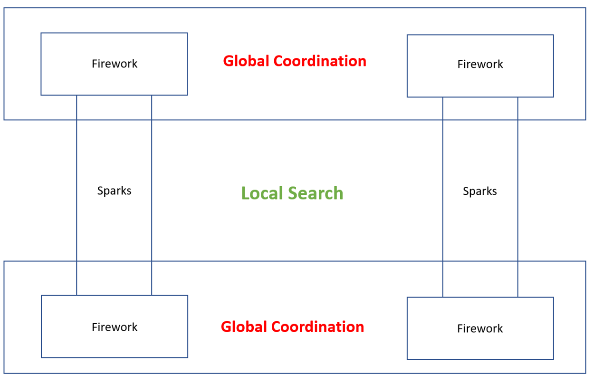

Fig. 1. An illustration of the fundamental framework of fireworks algorithm. Two main components of fireworks algorithm is local search via explosion and global coordination.

Algorithm 1: Fireworks Algorithm

Input : μ, λ
Output: Optimal solution
Initialize μ fireworks randomly within the search space
while *termination condition not met* **do**
  for $Fireworks_i$ *in* $Fireworks$ **do**
    Generate λ_i sparks U_i uniformly around $Firework_i$ within amplitude A_i
    Select the best candidate among $\{U_i \cup Firework_i\}$ as the next $Firework_i$
    Adapt A_i according to selection result
  end
end

Over the course of development, FWA has attracted much attention from the research community. On the one hand, FWA has been applied to solve real world problems such as image processing [13], matrix factorization [12], spam detection [11], vehicle routing problem [5], large-scale travel salesman problem [9]. On the other hand, there has been numerous theoretical attempts made to improve the performance of FWA itself. Some important variants are enhanced FWA (EFWA) [8], dynFWA [7], bare-bones FWA (BBFWA) [6], guided FWA (GFWA) [2], loser-out tournament FWA (LoTFWA) [17]. Among them, GFWA and LoTFWA marks two of the most milestones. GFWA greatly improves the local search ability of FWA while LoTFWA introduced an effective global coordination mechanism between fireworks based on GFWA.

B. Information Utilization Ratio

$$IUR(g) = \frac{\sum_{i=1}^{g} H\left(Z_i \mid \bar{X}_{i-1}, \bar{Z}_{i-1}\right)}{\sum_{i=1}^{g} H\left(Y_i \mid \bar{X}_i, \bar{Y}_{i-1}\right)} \tag{1}$$

Information utilization ratio (IUR) of an algorithm can act as an effective metric to compare heuristic algorithms within the same algorithm family. For example, [20] showed that in the the PSO family, $IUR_{PSO} \leq IUR_{SPSO}$, in the ES family, $IUR_{CMA-ES} \leq IUR_{(\mu,\lambda)-ES}$, in the DE family, $IUR_{JADE} \leq IUR_{DE}$, which correspond to the fact that those algorithms with larger IUR value tend to perform better.

For typical $(\mu, \lambda) - ES$, [20] has derived its IUR to be equation (2), where g represents the generation, μ is the number of parents, λ is the number of offspring.

$$IUR_{(\mu,\lambda)-ES}(g) = \frac{(g-1)\ln(C_{\lambda}^{\mu})}{g\lambda H(f(x))} \tag{2}$$

In the simple case of Luus-Jaakola algorithm where a single sample is generated with a uniform distribution, it has an IUR of eqaution (3)

$$\pi(i) = -\frac{g}{g+1}\ln\frac{g}{g+1} - \frac{1}{g+1}\ln\frac{1}{g+1} \tag{3}$$

$$IUR_{LJ}(g) = \frac{\sum_{i=1}^{g-1}\pi(i)}{gH(f(x))} \tag{4}$$

In the case of FWA family, the simpler BBFWA has an IUR bounded according to equation (5), This equation is derived from the fact that valina BBFWA enjoys similar properties with $(\mu, \lambda) - ES$ and Luus-Jaakola algorithm, and its IUR can be bounded by their respectivie IUR.

$$\frac{(g-1)\ln(\lambda+1)}{g\lambda H(f(x))} \leq IUR_{BBFWA}(g) \tag{5}$$

$$\leq \frac{\sum_{i=1}^{g-1}\pi(i)}{gH(f(x))} \tag{6}$$

C. Guided Fireworks Algorithm and Loser-out Tournament Fireworks Algorithm

The Guided Fireworks Algorithm takes advantage of Information Utilization Ratio to enhance the local search ability of Fireworks Algorithm. In each generation, each firework first conducts a uniform explosion to generate sparks as usual, and then those sparks are evaluated and ranked according to their fitness. With a predefined guiding mutation ratio σ, a guiding vector is calculated by calculating the difference vector between the the best set of sparks and worst set of sparks. Then this guiding vector is added to the firework position to form a guiding spark. The procedure of GFWA is described the Algorithm 5.

For GFWA, it has an IUR larger than BBFWA due to the fact that it utilizes the information from the ranking of sparks

Algorithm 2: Guided Fireworks Algorithm

Input : μ, λ, σ
Output: Optimal solution
Initialize μ fireworks randomly within the search space
while *termination condition not met* **do**
  for μ_i *in* μ **do**
    Generate λ_i sparks U_i uniformly around $Firework_i$ within amplitude A_i
    Evaluate λ_i and rank sparks according to fitness in ascending order
    $G_i = \bar{U}_i[:\sigma\lambda_i] - \bar{U}_i[-\sigma\lambda_i:] + Firework_i$
    Evaluate G_i
    Select the best candidate among $\{U_i \cup G_i \cup Firework_i\}$ as the next $Firework_i$
    Adapt A_i according to selection result
  end
end

during the generation of the guiding spark. Thus GFWA has a larger IUR which is bounded by equation (7)

$$\frac{(g-1)(\ln C_{\lambda}^{\sigma\lambda} + \ln C_{\lambda-\sigma\lambda}^{\sigma\lambda})}{g(\lambda+1)H(f(x))} \leq IUR_{GFWA}(g) \tag{7}$$

$$\leq \frac{(g-1)\left(\ln C_{\lambda}^{\sigma\lambda} + \ln C_{\lambda-\sigma\lambda}^{\sigma\lambda} + \ln(\sigma\lambda+1)\right)}{g(\lambda+1)H(f(x))} \tag{8}$$

Loser-out tournament fireworks algorithm introduced competition between fireworks on top of GFWA. To be specific, in each generation, every firework will calculate its expected final fitness at the end of optimization process with a linear estimation. When the estimated fitness of a firework falls below the current best real fitness value in the population, it is considered a loser and will be randomly restarted in the next generation. Algorithm 3 describes the workflow of LoTFWA in detail. Compared to GFWA, LoTFWA greatly enhances its ability to deal with more complex multi-modal problems. But on some uni-modal problems and large-scale problems, GFWA is still a better option.

III. Proposed Method

A. Exponentially Decaying Fireworks Algorithm

The guided fireworks algorithm improves the local search ability of fireworks algorithm, but this idea only exploits the information from sampled candidates with one guiding mutation. In order to further utilize the information of the population and better adapt to the local landscape of the function, a new explosion operator to improve the explosion operator with a series of guided explosion in an exponentially decaying manner is proposed.

For a specific firework **f** in generation **g**, its local search samples is denoted with S. In the case of GFWA, these samples consist of a group of uniform isomorphic samples and an additional guided mutation sample. We denote samples

Algorithm 3: Loser-out Tournament Step

for $Fireworks_i$ *in* $Fireworks$ **do**
  Conducts local search according to algorithm 2
  $p = f(Fireworks_i^{current}) - f(Fireworks_i^{previous})$
  if $p < 0$ **then**
    $g_{left} = floor(\frac{evaluation\_left}{\mu})$
    $\hat{f}(Fireworks_i^{final}) = f(Fireworks_i^{current}) + p * g_{left}$
    if $\hat{f}(Fireworks_i^{final}) > min(Fireworks^{current})$ **then**
      Restart $Fireworks_i$ and reset its parameters
    end
  end
end

generated by the uniform search distribution with U, and the guiding spark as G. The explosion center of U is denoted as P, the amplitude of U is denoted as R, and the number of samples (sparks) of U is determined by M. The tuple $\{P, R, M\}$ is a complete description of U.

$$S(\mathbf{GFWA}) = U + G \tag{9}$$

In EDFWA's case, S now consists of a series of uniform samples whose amplitudes decay in an exponential manner.

$$S(\mathbf{EDFWA}) = U_0 + U_1 + ... + U_n \tag{10}$$

In the above equation, it can be seen that the G term is omitted.This design choice is made out of the consideration that our samples will search in the neighborhood region of the guided spark, and thus there is no need to sample another point.

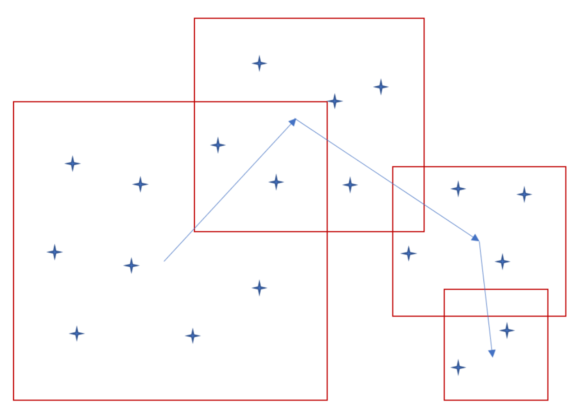

Fig. 2. Illustration of exponentially decaying explosion sparks generated for one firework in one generation. The red lines indicate explosion amplitudes, blue stars represent sampled candidates, blue arrows represent guided mutation vectors.

Each $U_i (0 < 1 \leq n)$ is completely determined by U_{i-1}. To be specific, after the uniform explosion of U_{i-1}, we first evaluate the fitness of the sparks in U_{i-1}, and calculate the guiding mutation vector and add it to P_{i-1} to form P_i. In terms of A_i and M_i, they are obtained by multiplying an exponentially decaying ratio parameter γ with A_{i-1} and M_{i-1}. This procedure is repeated until M_i becomes too small (when $M_i < 2$) to generate future samples. The full procedure of the exponentially decaying algorithm is described in Algorithm 4

$$P_{i+1} = P_i + G_{i-1} \tag{11}$$

$$A_{i+1} = A_i * \gamma \tag{12}$$

$$M_{i+1} = M_i * \gamma \tag{13}$$

The global coordination between fireworks is the same as LoTFWA, which means that fireworks whose expected final fitness is worse than the current best fitness will be restarted.

Based on the idea of utilizing more information, and making it possible for explosion sparks to better adapt to local structure, a new explosion operator called exponentially decaying explosion operator is proposed. The overall procedure of EDFWA is as described in algorithm 5, and the explosion process of EDFWA is depicted in Figure 2. Figure 3 illustrates the sampling distribution of EDFWA vs GFWA. It can be clearly seen from the 2-dimensional illustration that EDFWA stretched its sampling distribution according to the global structure of the function landscape while GFWA could only sampling inside a cube area.

Algorithm 4: Exponentially decaying explosion

```
Input  : f, U_0, P_0, A_0, M_0, γ, σ
Output: S
Let S = U_0 , i = 0
while M_i ≥ 2 do
    Sort U_i according to f(U_i)
    N_gm = M_i * σ
    G_i = U_i[: N_gm] − U_i[−N_gm :]
    Calculate P_{i+1}, A_{i+1}, M_{i+1} according to equation
      (11)
    Sample U_{i+1} according to P_{i+1}, A_{i+1}, M_{i+1}
    S = S + U_{i+1}
    i = i + 1
end
```

B. Analysis of the IUR of Exponentially Decaying Explosion

Now consider the case in EDFWA, its IUR is analysed and compared to GFWA under the condition of identical spark numbers. Since EDFWA differs from GFWA only in terms of the explosion operator, we only need to focus on the sum $\ln C_{\lambda}^{\sigma\lambda} + \ln C_{\lambda-\sigma\lambda}^{\sigma\lambda}$ to do the comparison. This sum comes from the fact that GFWA utilizes the information of the best $\sigma\lambda$ and the worst $\sigma\lambda$ sparks. Suppose in generation g, EDFWA generates a total of $\lambda = \lambda_0 + \lambda_1 + ... + \lambda_n$, where $\lambda_i = \lambda_0\gamma^i$ sparks, and these sparks sum up to λ. Let's first construct a function $I(x)$ according to equation (14), we claim that $I_{EDFWA} = I(\lambda_0)+I(\lambda_1)+...+I(\lambda_n) \geq I_{GFWA}(\lambda)$. Though this inequality does not directly imply that EDFWA has larger IUR than GFWA since it only lifts the lower bound and the upper bound of the original IUR_{GFWA}, our proof do convey the idea that EDFWA do utilize more information than GFWA.

$$I(\lambda) = \ln C_{\lambda}^{\sigma\lambda} + \ln C_{\lambda-\sigma\lambda}^{\sigma\lambda} \tag{14}$$

Proposition 1: suppose $\lambda = \sum_{i=0}^{n} \lambda_0\gamma^i$, we have the following inequality:

$$\sum_{i=0}^{n} I(\lambda_0\gamma^i) \geq I(\lambda) \tag{15}$$

Proof 1: The above inequality holds if $I(x)$ is a convex function according to Jensen's inequality. We now show that I is indeed a convex function due to the fact that its second derivative is positive.
Let

$$f(\lambda) = \ln C_{\lambda}^{\sigma\lambda} \tag{16}$$

since

$$\ln C_{\lambda}^{\sigma\lambda} = \frac{\lambda!}{(\sigma\lambda)!(\lambda-\sigma\lambda)!} \tag{17}$$

and

$$n! = \Gamma(n+1) \tag{18}$$

where Γ is the Gamma function, we then have:

$$f(\lambda) = \ln\Gamma(\lambda+1) - \ln\Gamma(\sigma\lambda+1) - \ln\Gamma(\lambda-\sigma\lambda+1) \tag{19}$$

consider the fact that:

$$\frac{\ln\Gamma(x)}{dx} = \frac{\Gamma'(x)}{\Gamma(x)} \tag{20}$$

$$= -\gamma' + \sum_{k=1}^{\infty}\left(\frac{1}{k} - \frac{1}{k+x+r-1}\right) \tag{21}$$

In our case, x can only take positive integer values, so we have:

$$\frac{\ln\Gamma(x)}{dx} = \gamma' + H_{x-1} \tag{22}$$

$$H_{x-1} = \sum_{k=1}^{x-1}\frac{1}{k} \tag{23}$$

where γ' is the Euler-Mascheroni Constant, and H_{x-1} is the harmonic number. Then the first derivative would be:

$$\frac{df}{d\lambda} = H_{\lambda} - \sigma H(\sigma\lambda) - (1-\sigma)H(\lambda-\sigma\lambda) \tag{24}$$

consider the relation of harmonic number and Riemann zeta function ζ, there is a such relation that can help us get the second derivative:

$$\frac{d^n H_x}{dx^n} = (-1)^{n+1} n! \left[\zeta(n+1) - H_{x,n+1}\right] \tag{25}$$

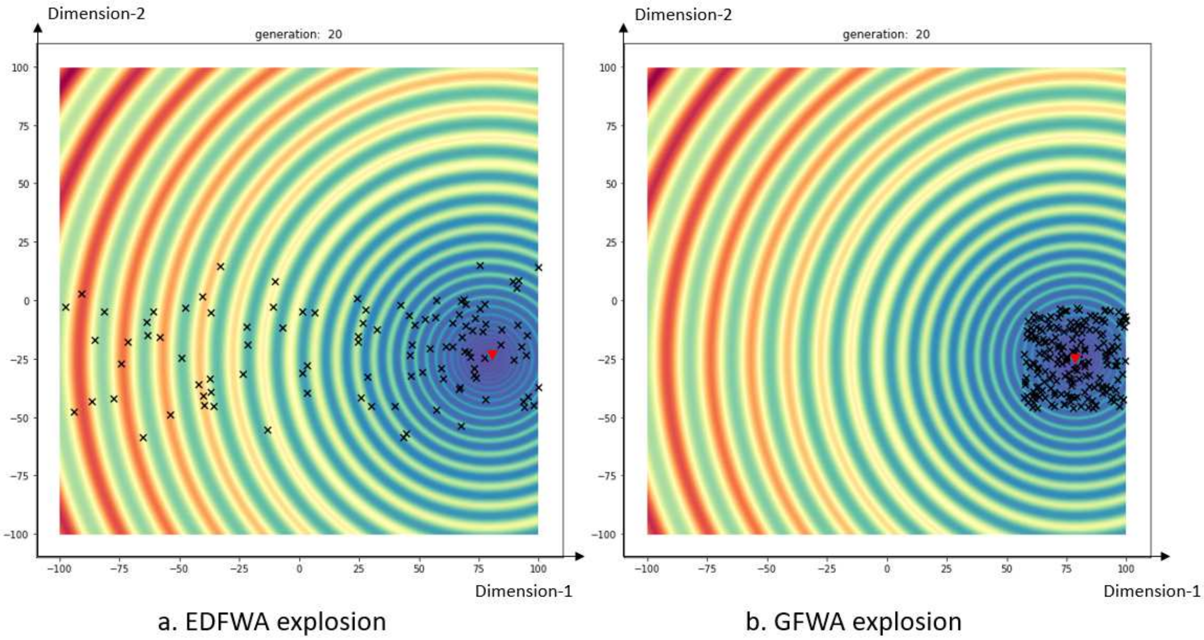

Fig. 3. A direct comparison of a single explosion between EDFWA and GFWA on 2 dimensional Shifted and Rotated Schaffer's F7 Function, which is multi-modal and has a huge number of local optima. In the above figure, the function value is represented by color differentiated contour plot. The blue color corresponds to a small function value (better). The global optimum lies on the right side of the figure with large number of blue circles surrounding it. Each black cross is a firework. Figure a corresponds to EDFWA, and figure b corresponds to GFWA. It can be seen that EDFWA can capture the global structure to some extent while GFWA can only rely on uniform sampling.

this leads to the second derivative of f:

$$\frac{d^2 f}{d\lambda^2} = 2\sigma\zeta(2) + \sigma^2 H_{\sigma\lambda,2} + (1-\sigma)^2 H_{(1-\sigma)\lambda,2} - H_{\lambda,2} \geq 0 \tag{26}$$

Considering the relationship of generalized harmonic numbers with polygamma functions and Gamma functions in the following form:

$$H_{n,r} = n^{-r} + \frac{\psi_{r-1}(n)}{\Gamma(r)} + \zeta(r) \tag{27}$$

This relationship leads us to the following equation:

$$\begin{aligned}\frac{d^2 f}{d\lambda^2} &= \Gamma(2)(\sigma^2\psi(\sigma\lambda) \\ &\quad + (1-\sigma)^2\psi((1-\sigma)\lambda) - \psi(\lambda) + 2\sigma^2\zeta(2))\end{aligned} \tag{28}$$

Here, $\Gamma(2) > 0, 2\sigma^2\zeta(2) > 0$, and $0 < \sigma < 1$. It is sufficient to prove the sum of the first three terms inside the parenthesis is larger or equal to 0. To prove this fact, consider the definiton of polygamma function:

$$\psi_n(z) = (-1)^{n+1} n! \sum_{k=0}^{\infty} \frac{1}{(z+k)^{n+1}} \tag{29}$$

Since in our case, the order of ψ is 1, we then have:

$$\begin{aligned}s(\sigma) &= \sigma^2 \sum \frac{1}{(k+\sigma\lambda)^2} \\ &\quad + (1-\sigma)^2 \sum \frac{1}{(k+(1-\sigma)\lambda)^2} - \sum \frac{1}{(k+\lambda)^2}\end{aligned} \tag{30}$$

To prove that $s(\sigma) \geq 0$, it is sufficient to prove that every term inside the summation is greater or equal to 0, that is:

$$\frac{\sigma^2}{(k+\sigma\lambda)^2} + \frac{(1-\sigma)^2}{(k+(1-\sigma)\lambda)^2} - \frac{1}{(k+\lambda)^2} \tag{31}$$

The above equation is equivalent to the following form:

$$\begin{aligned}&\frac{\sigma^2(k+(1-\sigma)\lambda)^2(k+\lambda)^2 + (1-\sigma)^2(k+\sigma\lambda)^2(k+\lambda)^2}{(k+\sigma\lambda)^2(k+(1-\sigma)\lambda)^2(k+\lambda)^2} \\ &- \frac{(k+\sigma\lambda)^2(k+(1-\sigma)\lambda)^2}{(k+\sigma\lambda)^2(k+(1-\sigma)\lambda)^2(k+\lambda)^2}\end{aligned} \tag{32}$$

It is then sufficient to prove that the sum of the denominators is greater or equal to 0. The following substitution is made:

$$\begin{aligned} a &= k + \lambda \\ b &= k + \sigma\lambda \\ c &= a - \sigma\lambda \end{aligned} \tag{33}$$

Then the denominator becomes:

$$\begin{aligned} sd(\sigma) &= \sigma^2 a^2 c^2 + (1-\sigma)^2 a^2 b^2 - b^2 c^2 \\ &= \sigma^2 c^2 (a^2 - b^2) + ((1-\sigma)^2 a^2 - (1-\sigma^2) c^2) b^2 \end{aligned} \tag{34}$$

Notice that:

$$(1-\sigma)^2 - (1-\sigma^2) \geq 0 \tag{35}$$

this leads to:

$$sd(\sigma) \geq \sigma^2 c^2 (a^2 - b^2) + (1-\sigma^2) b^2 (a^2 - c^2) \tag{36}$$

TABLE I
COMPARING EDFWA WITH LOTFWA ON CEC2017.

	LoTFWA				EDFWA				
FuncId	Mean	Min	Max	Std	Mean	Min	Max	Std	Won
1	**1.65e+02**	2.30e-03	2.17e+03	2.65e+02	2.48e+02	7.63e-06	2.19e+03	3.81e+02	=
2	6.53e-03	3.57e-03	1.67e-02	1.77e-03	**1.17e-03**	8.09e-04	1.59e-03	1.51e-04	+
3	4.83e-04	0.00e+00	3.72e-03	5.44e-04	**5.39e-06**	0.00e+00	3.05e-05	1.16e-05	+
4	6.72e+01	5.89e-03	1.18e+02	3.02e+01	**5.28e+01**	2.75e-04	1.16e+02	3.96e+01	+
5	6.85e+01	3.59e+01	1.04e+02	1.19e+01	**4.05e+01**	2.29e+01	6.67e+01	8.86e+00	+
6	4.08e-01	0.00e+00	9.90e+00	1.01e+00	**4.81e-03**	0.00e+00	1.41e-01	2.18e-02	+
7	6.08e+01	3.23e+01	8.58e+01	7.74e+00	**6.07e+01**	4.63e+01	7.68e+01	5.79e+00	=
8	6.25e+01	3.28e+01	8.66e+01	1.03e+01	**3.49e+01**	2.19e+01	4.97e+01	5.41e+00	+
9	4.60e+01	0.00e+00	9.11e+02	1.60e+02	**0.00e+00**	0.00e+00	0.00e+00	0.00e+00	+
10	2.47e+03	1.25e+03	3.56e+03	3.34e+02	**1.98e+03**	1.03e+03	2.58e+03	3.04e+02	+
11	1.15e+02	1.99e+01	2.97e+02	4.17e+01	**9.77e+01**	4.88e+01	1.76e+02	2.39e+01	+
12	1.16e+06	2.51e+04	6.12e+07	4.37e+06	**1.92e+05**	1.25e+04	1.02e+06	1.38e+05	+
13	**2.14e+04**	6.41e+03	4.83e+04	8.53e+03	2.38e+04	7.96e+03	5.16e+04	9.73e+03	-
14	**8.05e+02**	2.57e+02	3.50e+03	5.98e+02	8.20e+02	2.15e+02	6.67e+03	7.76e+02	=
15	**1.17e+04**	1.49e+03	3.17e+04	5.37e+03	1.32e+04	1.69e+03	3.52e+04	6.79e+03	=
16	6.08e+02	1.71e+02	1.11e+03	1.53e+02	**5.48e+02**	1.85e+02	8.74e+02	1.34e+02	=
17	1.19e+02	4.63e+01	3.36e+02	4.15e+01	**1.14e+02**	4.30e+01	2.56e+02	4.25e+01	=
18	5.63e+04	4.76e+03	1.35e+05	2.50e+04	**4.92e+04**	1.53e+04	1.29e+05	2.04e+04	=
19	9.07e+04	1.66e+03	3.96e+05	7.22e+04	**3.94e+04**	3.71e+03	9.05e+04	1.91e+04	+
20	2.36e+02	8.28e+01	4.49e+02	6.49e+01	**1.99e+02**	7.83e+01	3.49e+02	5.84e+01	+
21	2.66e+02	1.02e+02	3.07e+02	2.35e+01	**2.44e+02**	1.00e+02	2.65e+02	2.20e+01	+
22	**1.00e+02**	1.00e+02	1.00e+02	0.00e+00	**1.00e+02**	1.00e+02	1.00e+02	0.00e+00	=
23	4.66e+02	1.00e+02	6.01e+02	3.89e+01	**4.45e+02**	3.92e+02	4.95e+02	2.15e+01	+
24	6.46e+02	4.84e+02	9.70e+02	1.09e+02	**5.12e+02**	4.48e+02	5.52e+02	1.89e+01	+
25	3.99e+02	3.81e+02	4.65e+02	1.28e+01	**3.87e+02**	3.84e+02	3.90e+02	1.08e+00	+
26	1.19e+03	2.00e+02	2.42e+03	8.78e+02	**1.14e+03**	2.00e+02	2.31e+03	7.70e+02	=
27	5.78e+02	4.97e+02	8.65e+02	9.63e+01	**5.51e+02**	5.24e+02	5.96e+02	1.47e+01	+
28	3.54e+02	3.00e+02	4.58e+02	5.00e+01	**3.11e+02**	3.00e+02	4.07e+02	3.04e+01	+
29	7.21e+02	3.86e+02	1.27e+03	1.44e+02	**6.70e+02**	4.60e+02	9.25e+02	7.83e+01	=
30	1.83e+05	4.48e+02	1.18e+06	1.97e+05	**1.48e+05**	2.88e+04	4.99e+05	8.31e+04	+
	won:1 lost:19 draw:10				won:19 lost:1 draw:10				

It is not hard to observe the following relationship:

$$\begin{aligned} a, b, c &> 0 \\ a &> b \\ a &> c \end{aligned} \tag{37}$$

Then it is pretty obvious that $sd(\sigma) \geq 0$. This proves the fact that:

$$\frac{d^2 f}{d\lambda^2} \geq 0 \tag{38}$$

This second order derivative becomes 0 only when $\sigma = 0$ and is always positive when $\sigma > 0$. This directly proves the convexity of f, and the other part of I can follow a similar proof and lead to the same conclusion. Since I is a positive combination of two convex terms, it is also convex, which then leads to the conclusion of proposition 1. This proof indicates that EDFWA can take advantage more information than GFWA, and thus can lead to better performance.

Algorithm 5: Exponentially Decaying Fireworks Algorithm

Input : $\mu, f, U_0, P_0, A_0, M_0, \gamma, \sigma$
Output: Optimal solution
Initialize μ fireworks randomly within the search space
while *termination condition not met* **do**
    for $Fireworks_i$ *in* $Fireworks$ **do**
        Conduct local search according to algorithm 4
        Conduct global coordination according to algorithm 3
    end
end

IV. EXPERIMENTS

A. Benchmark Functions

To test the performance of EDFWA, we adopted two benchmark suites to evaluate different aspects of EDFWA. The first benchmark suite is CEC2017 benchmark for bounded real parameter optimization, where the problems dimension

TABLE II
COMPARING EDFWA WITH THREE STRONG BASELINES SPECIALLY DESIGNED FOR LARGE-SCALE OPTIMIZATIONS AND GFWA ON CEC2013 LARGE-SCALE

FuncId	MPS	DECC-G	CC-CMA-ES	GFWA	EDFWA
1	6.68e+08	**0.00e+00**	**0.00e+00**	5.95e+07	2.35e+07
2	4.20e+03	**1.31e+03**	1.37e+03	1.59e+04	1.26e+04
3	1.94e+00	1.09e+00	**0.00e+00**	2.03e+01	2.08e+01
4	1.07e+11	2.16e+11	**2.82e+09**	8.31e+11	2.52e+10
5	**1.20e+06**	8.30e+06	7.28e+14	8.22e+06	6.46e+06
6	**6.01e+03**	1.74e+05	4.56e+05	1.00e+06	1.00e+06
7	7.19e+07	1.02e+09	**2.26e+06**	3.21e+13	4.10e+06
8	2.04e+14	6.94e+15	3.32e+14	1.08e+14	**6.52e+13**
9	**1.66e+08**	5.47e+08	3.82e+08	8.53e+08	7.36e+08
10	**3.53e+06**	2.43e+07	4.51e+06	9.08e+07	9.08e+07
11	2.20e+09	1.21e+11	**1.24e+08**	2.04e+15	6.37e+08
12	1.75e+04	4.53e+03	1.33e+03	1.47e+12	**1.31e+03**
13	9.87e+08	9.40e+09	1.80e+09	2.99e+15	**5.83e+07**
14	1.03e+09	1.36e+11	3.58e+08	2.32e+15	**3.17e+08**
15	2.76e+07	1.17e+07	3.13e+07	2.19e+12	**5.93e+06**
AR	2.60	3.07	2.27	4.53	2.47

is 30. This benchmark is used to show that our proposed method outperforms the state-of-the-art FWA variant LoTFWA (which is GFWA with loser-out tournament mechanism). The second benchmark suite is CEC2013 large-scale optimization benchmark, which is used to show that our proposed algorithm is very competitive on high-dimensional large scale problems even without additional mechanisms like cooperative co-evolution. In this benchmark, the problems are very large-scale and has 1000 dimensions to be optimized.

B. Parameter Settings

For FWA variants, is has been shown that two coefficient of dynamic amplitude namely C_r, C_a are set to $C_r = 0.9, C_a = 1.2$ respectively. We also choose to follow this convention. Other parameters that arise in FWA variants are the firework number μ, the spark number λ and the guided mutation ratio σ. For LoTFWA, these values are set to be $\mu = 5, \lambda = 60, \sigma = 0.2, A = (UpperBound - LowerBound)/2$ as described in [17]. Note that the setting of λ can be different between fireworks, we initialized all λs to be the same since we do note consider more complex spark assignment strategies in this paper.

In the case of EDFWA, we discarded parameter λ in favor of two new parameters λ_0 and γ because the number of sparks for each firework can't be set in advance, as our exponentially decaying explosion will generate a series of sparks and the spark number is the sum of a geometric progression which is determined by the initial number of explosion sparks λ_0 and the decaying factor γ. Several different combinations of λ_0 and γ are compared, and we find that setting $\mu = 3, \lambda_0 = 30, \gamma = 0.75$ yields the best results in the case of CEC2017.

C. Results

The experiments platform is Ubuntu 18.04 with Intel(R) Xeon(R) CPU E5-2675 v3. For the CEC2017 benchmark, we ran 51 times repeatedly for each function of dimension 30 with a maximum evaluation number of 300000. For CEC2013 Large-Scale benchmark, we ran 25 times for each function of dimension 1000 with a maximum evaluation numebr of 3000000. These settings are standard experimental settings for those benchmarks.

Wilcoxon rank-sum tests are conducted on CEC2017 results to verify the performance gain of EDFWA compared to LoTFWA on CEC2017 benchmark suite. We consider a confidence level of 95% to be of significance. Results from I indicates that EDFWA outperforms LoTFWA and only lost on 1 problem.

In large-scale benchmark, the story is more complicated. We chose 4 algorithms as competitors to test how EDFWA performs on larger-scale problems. MPS is Minimum Population Search algorithm which is a recently developed metaheuristic specifically designed to optimize high dimensional multimodal functions. DECC-G is a variant from the differential evolution family with JADE and cooperative coevolution mechanism that can serve as a strong baseline from the DE family. CC-CMA-ES is a CMA-ES variant coupled with cooperative coevolution which aims to scale CMA-ES to large-scale problems. The last algorithm for comparison is GFWA which is used to test how much improvement is made due to our proposed method. It can be seen from Table II that the average ranking of EDFWA is 2.47, which is just slightly worse than CC-CMA-ES, making EDFWA the second best algorithm.

EDFWA as an algorithm without any special mechanism for large-scale optimization exihibited very strong performance compared to those specially designed ones. Except the first 3 fully separable functions where dimension variable grouping can yield huge performance gain, EDFWA performs better or close to other competitors. On the last 4 functions where the relationship between dimensions is hard to decode or the problem is fully non-separable, EDFWA showed dominating performance. In comparison with GFWA, it is surprising for us to observe a large performance jump. On several functions, EDFWA outperforms GFWA by orders of magnitude, which indicates that our proposed method is very effective under large-scale high dimensional situations.

Overall, our theoretical and experimental results suggest that exponentially decaying explosion can greatly enhance the local search ability of fireworks algorithm, especially in large-scale high dimensional situations. There are still many future works to be done to further push the boundary of fireworks algorithm, such as integrating EDFWA with cooperative co-evolution mechanisms, and designing more efficient global coordination strategies.

V. CONCLUSION

In this paper, an exponentially decaying explosion fireworks algorithm is proposed. Its information utilization ratio was theoretically analysed and the result showed that EDFWA can make better use of information than GFWA. Its performance was tested against LoTFWA on the CEC2017 benchmark, where EDFWA showed significant advantage. Then EDFWA is compared on the CEC2013 large-scale benchmark suite against three representative algorithms specially designed for high dimensional optimization and GFWA. Experimental results indicated that even without any mechanism specially designed for high dimensional scenarios, EDFWA is very competitive against those specialized optimizers from different families. It can even outperform GFWA with orders of magnitude improvement. We expect future variants of FWA to utilize this new mechanism and be able to solve a larger range of problems.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (Grant No. 62076010), and partially supported by Science and Technology Innovation 2030 - "New Generation Artificial Intelligence" Major Project (Grant Nos.: 2018AAA0102301 and 2018AAA0100302).

REFERENCES

[1] Y. Tan and Y. Zhu, "Fireworks Algorithm for Optimization," in Advances in Swarm Intelligence, vol. 6145, Y. Tan, Y. Shi, and K. C. Tan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 355–364.

[2] J. Li, S. Zheng, and Y. Tan, "The Effect of Information Utilization: Introducing a Novel Guiding Spark in the Fireworks Algorithm," IEEE Trans. Evol. Computat., vol. 21, no. 1, pp. 153–166, Feb. 2017, doi: 10.1109/TEVC.2016.2589821.

[3] A. Bolufé-Röhler, S. Chen, and D. Tamayo-Vera, "An Analysis of Minimum Population Search on Large Scale Global Optimization," in 2019 IEEE Congress on Evolutionary Computation (CEC), Jun. 2019, pp. 1228–1235, doi: 10.1109/CEC.2019.8789978.

[4] R. E. Odeh and D. E. Knuth, "The Art of Computer Programming. Volume 1: Fundamental Algorithms.," Journal of the American Statistical Association, vol. 64, no. 325, p. 401, Mar. 1969, doi: 10.2307/2283757.

[5] W. Yang and L. Ke, "An improved fireworks algorithm for the capacitated vehicle routing problem," Front. Comput. Sci., vol. 13, no. 3, pp. 552–564, Jun. 2019, doi: 10.1007/s11704-017-6418-9.

[6] J. Li and Y. Tan, "The bare bones fireworks algorithm: A minimalist global optimizer," Applied Soft Computing, vol. 62, pp. 454–462, Jan. 2018, doi: 10.1016/j.asoc.2017.10.046.

[7] S. Zheng, A. Janecek, J. Li, and Y. Tan, "Dynamic search in fireworks algorithm," in 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, Jul. 2014, pp. 3222–3229, doi: 10.1109/CEC.2014.6900485.

[8] S. Zheng, A. Janecek and Y. Tan, "Enhanced Fireworks Algorithm," 2013 IEEE Congress on Evolutionary Computation, 2013, pp. 2069-2077, doi: 10.1109/CEC.2013.6557813.

[9] H. Luo, W. Xu, and Y. Tan, "A Discrete Fireworks Algorithm for Solving Large-Scale Travel Salesman Problem," in 2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, Jul. 2018, pp. 1–8, doi: 10.1109/CEC.2018.8477992.

[10] J. Liu and K. Tang, "Scaling Up Covariance Matrix Adaptation Evolution Strategy Using Cooperative Coevolution," in Intelligent Data Engineering and Automated Learning – IDEAL 2013, Berlin, Heidelberg, 2013, pp. 350–357.

[11] W. He, G. Mi, and Y. Tan, "Parameter Optimization of Local-Concentration Model for Spam Detection by Using Fireworks Algorithm," in Advances in Swarm Intelligence, vol. 7928, Y. Tan, Y. Shi, and H. Mo, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 439–450.

[12] A. Janecek and Y. Tan, "Swarm Intelligence for Non-Negative Matrix Factorization:," International Journal of Swarm Intelligence Research, vol. 2, no. 4, pp. 12–34, Oct. 2011, doi: 10.4018/jsir.2011100102.

[13] S. Zheng and Y. Tan, "A unified distance measure scheme for orientation coding in identification," in 2013 IEEE Third International Conference on Information Science and Technology (ICIST), Yangzhou, Jiangsu, China, Mar. 2013, pp. 979–985, doi: 10.1109/ICIST.2013.6747701.

[14] N. H. Awad, M. Z. Ali, P. N. Suganthan, J. J. Liang, and B. Y. Qu, "Problem Definitions and Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization," p. 34.

[15] J. Li and Y. Tan, "A Comprehensive Review of the Fireworks Algorithm," ACM Comput. Surv., vol. 52, no. 6, pp. 1–28, Jan. 2020, doi: 10.1145/3362788.

[16] G. He, D. Conte, R. G. Melton, and D. B. Spencer, "Fireworks Algorithm Applied to Trajectory Design for Earth to Lunar Halo Orbits," Journal of Spacecraft and Rockets, vol. 57, no. 2, pp. 235–246, Mar. 2020, doi: 10.2514/1.A34469.

[17] J. Li and Y. Tan, "Loser-Out Tournament-Based Fireworks Algorithm for Multimodal Function Optimization," IEEE Transactions on Evolutionary Computation, vol. 22, no. 5, pp. 679–691, Oct. 2018, doi: 10.1109/TEVC.2017.2787042.

[18] X. Li, K. Tang, M. N. Omidvar, Z. Yang, and K. Qin, "Benchmark Functions for the CEC'2013 Special Session and Competition on Large-Scale Global Optimization," p. 23.

[19] Z. Yang, J. Zhang, K. Tang, X. Yao, and A. C. Sanderson, "An adaptive coevolutionary Differential Evolution algorithm for large-scale optimization," in 2009 IEEE Congress on Evolutionary Computation, May 2009, pp. 102–109, doi: 10.1109/CEC.2009.4982936.

[20] J. Li and Y. Tan, "Information Utilization Ratio in Heuristic Optimization Algorithms," arXiv:1604.01643 [cs], Sep. 2016, Accessed: Jan. 23, 2021. [Online]. Available: http://arxiv.org/abs/1604.01643.

[21] Y. Li and Y. Tan, "Multi-Scale Collaborative Fireworks Algorithm," 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, UK, 2020, pp. 1-8, doi: 10.1109/CEC48606.2020.9185563.

[22] Y. Pei, S.Q. Zheng, Y. Tan and Hideyuki Takagi, An Empirical Study on Influence of Approximation Approaches on Enhancing Fireworks Algorithm IEEE International Conference on System, Man and Cybernetics (SMC 2012) , Seoul, Korea. October 14-17, 2012.

[23] Abramowitz, Milton, Irene A. Stegun, and Robert H. Romer. "Handbook of mathematical functions with formulas, graphs, and mathematical tables." (1988): 958-958.

[24] X. Ye, J. Li, B. Xu and Y. Tan, "Which Mapping Rule in the Fireworks Algorithm is Better for Large Scale Optimization," 2018 IEEE Congress on Evolutionary Computation (CEC), 2018, pp. 1-8, doi: 10.1109/CEC.2018.8477645.

[25] J. Li and Y. Tan, "Enhancing interaction in the fireworks algorithm by dynamic resource allocation and fitness-based crowdedness-avoiding strategy," 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, pp. 4015-4021, doi: 10.1109/CEC.2016.7744299.

[26] C. Yu, L. C. Kelley and Y. Tan, "Cooperative framework fireworks algorithm with covariance mutation," 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, pp. 1196-1203, doi: 10.1109/CEC.2016.7743923.