Accelerating Fireworks Algorithm with Dynamic Population Size Strategy

YU, Jun
Institute of Science and Technology, Niigata University

高木, 英行
Faculty of Design, Kyushu University

https://hdl.handle.net/2324/4488110

出版情報：2020 Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems (SCIS-ISIS), pp.276-281, 2020-12-05. IEEE
バージョン：

Accelerating Fireworks Algorithm with Dynamic Population Size Strategy

Jun YU
Institute of Science and Technology
Niigata University
Niigata, Japan
URL: www.eng.niigata-u.ac.jp/~yujun

Hideyuki TAKAGI
Faculty of Design
Kyushu University
Fukuoka, Japan
URL: www.design.kyushu-u.ac.jp/~takagi

***Abstract*—A dynamic population size strategy is proposed for the fireworks algorithm (FWA) to adjust the population size based to the search results of the current generation. When the currently found optimal individual is updated, a linear decreasing method is activated to maintain an efficient exploitation speed. The population size is reduced by 1 until the minimum preset population size is reached, then the population size remains unchanged. Otherwise, we randomly generate a larger population size than the initial population and expand the explosion amplitudes of all firework individuals artificially, which the expectation that we can escape current local minima. To analyze the effectiveness of the proposed strategy, we combined it with the enhanced FWA (EFWA) together, and run the EFWA and (the EFWA + our proposed strategy) on 28 CEC 2013 benchmark functions in three different dimensions. Each function is run 30 trial times independently, and the Wilcoxon signed-rank test is applied to check significant differences. The statistical results showed that the proposed dynamic population size strategy can not only achieve a faster convergence speed for the FWA but also can jump out of trapped local minima more easily to maintain a higher performance, especially for high-dimensional problems.**

***Index Terms*—Evolutionary Computation, Fireworks Algorithm, Dynamic Population Size, Optimization**

I. Introduction

Optimization has always been a hot topic and has attracted lots of attention from both industry and academia. As real-world applications become more and more complicated, many classical mathematical methods are unable to deal with these problems effectively; they usually have a variety of characteristics such as strong constraints, large scale, and non-differentiability. Thus, population based evolutionary computation (EC) algorithms are widely spread and have been applied to many optimization tasks that are hard to be described mathematically or gotten their differential information. After decades of development, EC has become an important branch of optimization algorithms, and many influential algorithms have been proposed one after another, such as genetic algorithm [1], ant colony optimization [2], differential evolution [3], and vegetation evolution [4], exhibit outstanding performance on several kinds of optimization problems. Researchers have also introduced various novel strategies to further improve their performance [5]–[10].

This work was supported in part by Grant-in-Aid for Scientific Research (JP18K11470).

The fireworks algorithm (FWA) [11], as a potential member of the EC community of algorithms, quickly caught most people's eye, and many practitioners have become involved in the improvement and application of FWA. For example, Zheng et al. proposed a more powerful version called enhanced FWA (EFWA) [12] by introducing five new modifications to overcome the limitations of standard FWA. Since then, many effective strategies have been introduced to solve various problems, such as multi-modal problems [13], multi-objective problems [14], and large-scale problems [15]. Meanwhile, FWA has successfully solved many industrial problems, e.g. regional seismic waveform inversion [16], web information retrieval [17], and multilevel image thresholding [18]. Generally speaking, the FWA and its variants have satisfactory performance - but there is room for further improvement.

The first purpose is to propose a dynamic population size strategy for the FWA to maintain the high performance at all times in the face of optimization problems with different characteristics. The proposal reduces the population size to improve exploitation speed when the currently found optimal individual is updated. Otherwise, a random population size larger than the initial population is generated to help the FWA jump out of current trapped local areas. The second purpose is to analyze the contribution of the proposed dynamic population size strategy and its applicable scenarios. Finally, we also give several valuable research directions for discussion.

In addition to this introductory section, we briefly introduce the working mechanism of the FWA in Section II, and the proposed strategy is described comprehensively in Section III. Then, a controlled experiment uses 28 benchmark functions to analyze the performance of the proposed strategy in Section IV. Finally, we discuss the effectiveness of the proposed dynamic population size strategy and give some potential topics for further research in Section V, and then summarize our work in Section VI.

II. Fireworks Algorithm

Inspired by the observed explosion scenes of real fireworks, the FWA repeatedly simulates the cooperation of multiple firework individuals to find the global optimum, where each firework individual generates an unequal number of spark individuals within a specified explosion amplitude (explosion radius) according to its fitness. If a firework individual has a better fitness, it can generate the greater the number of spark individuals in a smaller explosion amplitude. In short, the fitness is proportional to the number of spark individuals, but inversely proportional to the explosion amplitude. Thus, the FWA can control the explosion parameters of firework individuals to balance exploration and exploitation well.

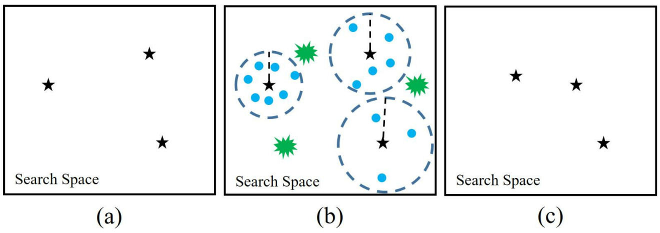


Fig. 1. The general optimization framework of the FWA. (a) Initial firework individuals are generated randomly, (b) spark individuals from explosion operations (blue solid points) and spark individuals from mutation operations (green irregular points) are generated, (c) new firework individuals that survive to the next generation are selected from all individuals in the (b). The (b) and (c) steps are executed repeatedly until a stopping condition is reached.

Similar to the optimization process of most EC algorithms, the FWA also randomly generates several initial firework individuals in the search space, and the explosion parameters of all firework individuals are automatically determined based on their fitness. Then, explosion operations are performed to find potential local areas, while Gaussian mutation operations are introduced to increase diversity. Finally, the best individual of all individuals survives directly to the next generation, and the remaining firework individuals in the next generation are randomly selected according to distance-based selection probability. The above-described explosion and selection operations are executed repeatedly until a stopping condition is reached. Fig. 1 describes the general optimization process of the FWA consisting of four major operations: initialization, explosion, mutation, and selection.

III. Dynamic Population Size Strategy

Since the FWA models the explosion of real fireworks to gradually improve the quality of the candidate solutions (firework individuals), it means that a large number of resources (fitness evaluations) are allocated to spark individuals generated by a small number of firework individuals (see step (b) in Fig. 1). Generally speaking, the number of spark individuals is several times or even ten times that of firework individuals. Suppose that the total number of spark individuals in every generation remains the same, the smaller the population size, the more resources each firework individual allocates, that is, the more spark individuals are generated. Conversely, each firework individual is allocated fewer resources and the explosion amplitude is relatively reduced. This gives us a hint that population size may be an important parameter affecting the FWA performance.

There are examples in the literature of efforts to focus on the parameter tuning of the FWA and improve the FWA performance significantly. For example, Yu et al. proposed a strategy to gradually weaken exploration and simultaneously enhance exploitation by decreasing the explosion amplitude of firework individuals [19]. Dynamic FWA increases or decreases the explosion amplitude of the current best firework individual dynamically to maintain high performance in dealing with different convergence periods [20]. However, as far as we know, little attention has been paid to the population size of the FWA, which is the motivation for this paper: to enhance the FWA performance by changing the population size.

The proposed strategy is divided into two cases which increase or decrease the population size. In the first, we use whether the current optimal individual has been updated as an indicator to tune the population size. When the indicator is updated, this means that the current population has a high probability of finding more potential areas; reducing the population size can thus further enhance the exploitation ability because firework individuals can search their vicinities more precisely. Unfortunately, when the indicator is not updated, this means that the current population may be trapped in locally optimal areas from which it is difficult to escape. Increasing the population size helps to escape from the current trapped local areas by shifting the focus to exploration. Fig. 2 shows the effect of the two cases, and Algorithm 1 demonstrates a general optimization framework in which the proposed strategy is integrated into the FWA.

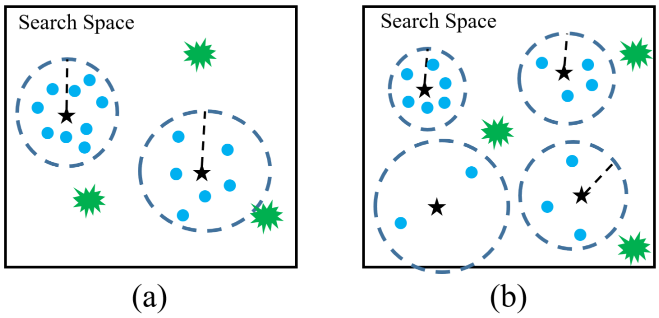


Fig. 2. The effect of changing the population size. Icons used have the same meaning as in Fig. 1. (a) Reduce the population size in exchange for a fast convergence speed, (b) Increase the population size in exchange for diversity and avoidance of being trapped in local areas.

Although there are many dynamic methods that have been proposed in other EC algorithms, we use a linear decreasing method and a random method to reduce and increase the population size respectively, and we refer them to them as *rule 1* and *2*. Note that to ensure the same fitness evaluations in each generation, we increase or decrease the number of spark individuals corresponding to the change in the number of firework individuals, i.e. if the number of firework individuals increases (decreases) by k, then the number of spark individuals decreases (increases) by k.

rule 1: The current population size is reduced by 1 to become the population size of the next generation. To avoid the population size being too small, we set half of the initial population size as the lower limit.

When the population size is reduced to the lower limit, it remains unchanged until rule 2 is executed.

rule 2: The population size of the the next generation is randomly generated between the initial population size and the upper limit which is set to 1.5 times the initial population size. Additionally, a modification to the explosion amplitude is introduced; i.e. the explosion amplitude of all firework individuals is doubled, to leave the current areas in time.

Algorithm 1 The optimization framework of the proposed dynamic population size strategy combined with the FWA. Steps 10-14 comprise our proposed strategy.

1: Initialize PS firework individuals randomly.
2: Evaluate the fitness of all firework individuals.
3: **while** a stopping condition is not reached **do**
4: Generate explosion spark individuals.
5: Generate mutant spark individuals. (optional).
6: **if** spark individuals are located outside the space **then**
7: Bringing them back to the space.
8: **end if**
9: Evaluate the fitness of each generated spark individuals.
10: **if** a better individual than the current best is found **then**
11: The population size in the next generation, PS_{next}, is reduced based on *rule 1*;
12: **else**
13: The population size in the next generation, PS_{next}, is increased based on *rule 2*;
14: **end if**
15: Select PS_{next} new individuals to the next generation.
16: **end while**
17: end of program.

IV. Experimental Evaluations

Since many FWA variants have been proposed, we select the powerful EFWA as our baseline algorithm and combine it with our proposed strategy. To analyze the performance of our proposal, we use 28 benchmark functions with three different dimensions, i.e., 2-D, 10-D, and 30-D, from the CEC2013 test suite [21] in our evaluation experiments. Table II summarizes the search range, optimal value, and the characteristics of these functions, including multimodality, shifted, and rotated. The parameter settings of EFWA used in the experiments are described in Table I, where the definition of the symbols can be found in the original literature [11].

We use the number of fitness evaluations instead of generations as the termination condition for a fair comparison, and run EFWA and (EFWA + the proposed strategy) on each function in three different dimensions with 30 independent trial runs. Finally, we apply the Wilcoxon signed-rank test to detect significant differences between EFWA and (EFWA + the proposed strategy) at the stop condition, e.g. the maximum number of fitness evaluations. The results of the statistical tests are summarized in Table III, and Fig. 4 shows the average convergence curve of all functions in 30-D space.

TABLE I
Parameter settings of the EFWA.

Parameters	Values
# of fireworks for any dimension search	10
# of sparks m	50
# of Gauss mutation sparks	5
constant parameters	$a = 0.04$ $b = 0.2$
Maximum amplitude A_{max}	40
Dimensions D	2, 10, and 30
stop criterion; max. # of fitness evaluations for 2-D, 10-D, and 30-D search	1,000, 10,000, 40,000
Lower limit of population size	5
Upper limit of population size	15

TABLE II
Benchmark Functions: Uni=unimodal, Multi=multimodal, Comp.=Composition

No.	Types	Characteristics	Optimum fitness
F_1		Sphere function	-1400
F_2		Rotated high conditioned elliptic function	-1300
F_3	Uni	rotated Bent Cigar function	-1200
F_4		Rotated discus function	-1100
F_5		different powers function	-1000
F_6		Rotated Rosenbrock's function	-900
F_7		Rotated Schaffers function	-800
F_8		Rotated Ackley's function	-700
F_9		Rotated Weierstrass function	-600
F_{10}		Rotated Griewank's function	-500
F_{11}		Rastrigin's function	-400
F_{12}	Multi	Rotated Rastrigin's function	-300
F_{13}		Non-continuous rotated Rastrigin's function	-200
F_{14}		Schwefel's function	-100
F_{15}		Rotated Schwefel's function	100
F_{16}		Rotated Katsuura function	200
F_{17}		Lunacek BiRastrigin function	300
F_{18}		Rotated Lunacek BiRastrigin function	400
F_{19}		Expanded Griewank's plus Rosenbrock's function	500
F_{20}		Expanded Scaffer's F_6 function	600
F_{21}		Composition Function 1 (n=5,Rotated)	700
F_{22}		Composition Function 2 (n=3,Unrotated)	800
F_{23}		Composition Function 3 (n=3,Rotated)	900
F_{24}	Comp.	Composition Function 4 (n=3,Rotated)	1000
F_{25}		Composition Function 5 (n=3,Rotated)	1100
F_{26}		Composition Function 6 (n=5,Rotated)	1200
F_{27}		Composition Function 7 (n=5,Rotated)	1300
F_{28}		Composition Function 8 (n=5,Rotated)	1400

V. Discussions

We begin the discussion with an explanation of the superiority of our proposed strategy. Since FWA uses a one-to-many generational relationship - i.e. a single firework individual generates multiple spark individuals - fluctuations in population size can cause the redistribution of spark individuals and even affect the explosion amplitude of firework individuals. In other words, changing the population size can indirectly affect the explosion operations that are a core factor affecting FWA performance. As the total number of individuals remain the same, reducing the population size allows firework individuals to explore local areas more precisely, while increasing the population size allows firework individuals to explore wider areas. Thus, the proposed strategy can further balance exploration and exploitation well by dynamically tuning the population size according to different optimization problems, or even dif-

TABLE III
STATISTICAL TEST RESULTS OF THE WILCOXON SIGNED-RANK TEST FOR AVERAGE NUMBER OF FOUND OPTIMUM AT THE STOP CONDITION. $A \gg B$ AND $A > B$ MEAN THAT A IS SIGNIFICANTLY BETTER THAN B WITH SIGNIFICANT LEVELS OF 1% AND 5%, RESPECTIVELY. $A \approx B$ MEANS THAT THERE IS NO SIGNIFICANT DIFFERENCE BETWEEN THEM ALTHOUGH A IS BETTER THAN B. EFWADYPS: EFWA + OUR PROPOSED STRATEGY.

	2D	10D	30D
F_1	EFWADyPS $\gg$ EFWA	EFWA $\gg$ EFWADyPS	EFWA $\gg$ EFWADyPS
F_2	EFWADyPS $\approx$ EFWA	EFWADyPS $>$ EFWA	EFWADyPS $>$ EFWA
F_3	EFWADyPS $>$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA
F_4	EFWA $\approx$ EFWADyPS	EFWA $\approx$ EFWADyPS	EFWADyPS $\gg$ EFWA
F_5	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA
F_6	EFWA $\approx$ EFWADyPS	EFWADyPS $>$ EFWA	EFWADyPS $>$ EFWA
F_7	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $>$ EFWA
F_8	EFWADyPS $>$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA
F_9	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA
F_{10}	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA
F_{11}	EFWADyPS $\gg$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $>$ EFWA
F_{12}	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA
F_{13}	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA
F_{14}	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA
F_{15}	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA
F_{16}	EFWA $\gg$ EFWADyPS	EFWA $\approx$ EFWADyPS	EFWA $\gg$ EFWADyPS
F_{17}	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA
F_{18}	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA
F_{19}	EFWA $\approx$ EFWADyPS	EFWADyPS $\gg$ EFWA	EFWADyPS $\gg$ EFWA
F_{20}	EFWADyPS $\gg$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA
F_{21}	EFWA $\approx$ EFWADyPS	EFWA $\approx$ EWFADyPS	EFWA $\gg$ EFWADyPS
F_{22}	EFWADyPS $>$ EFWA	EFWADyPS $\approx$ EFWA	EFWADyPS $>$ EFWA
F_{23}	EFWADyPS $\approx$ EFWA	EFWA $\approx$ EFWADyPS	EFWA $\approx$ EFWADyPS
F_{24}	EFWA $\approx$ EFWADyPS	EFWADyPS $>$ EFWA	EFWADyPS $\gg$ EFWA
F_{25}	EFWADyPS $\approx$ EFWA	EFWADyPS $\gg$ EFWA	EFWADyPS $\approx$ EFWA
F_{26}	EFWA $\approx$ EFWADyPS	EFWADyPS $\approx$ EFWA	EFWADyPS $\approx$ EFWA
F_{27}	EFWADyPS $\approx$ EFWA	EFWADyPS $>$ EFWA	EFWADyPS $\gg$ EFWA
F_{28}	EFWA $\approx$ EFWADyPS	EFWADyPS $\gg$ EFWA	EFWADyPS $>$ EFWA

ferent periods of the same problem. Furthermore, our proposal does not need to add any additional fitness calculations, and the increased CPU computation time is negligible. We can thus say that it is a low-cost, high return strategy.

Secondly, we would like to discuss the applicability of our proposed strategy. Not limited to the EFWA selected as the baseline algorithm for this work, the strategy can be easily combined with other versions of FWA, e.g. dynamic FWA, and adaptive FWA [22], without changing the originality of those optimization frameworks. Furthermore, other methods for tuning population size can also be introduced into our strategy. For example, a nonlinear decreasing (increasing) method can also replace methods used in our proposed strategy, i.e. the linear decreasing method and random method, to obtain better performance. Thus, our proposed strategy still has a lot of room for improvement.

Next, we outline some potential topics which would address the further improvement of the proposed strategy's performance. As unsuitable population size can even hinder convergence, one of our priorities in the near future is to determine how best to maintain a suitable population size throughout the convergence process. The proposed strategy observes whether the optimal individual in the previous generation is updated to determine the population size in the next generation. However, such a frequent operation is not conducive for accurately grasping the characteristics of and making the correct decision for problems with noise. An alternative approach is to tune the population size by observing the indicator update for a plurality of successive generations, which not only provides us with a better understanding of the information update in local areas but also reduces CPU computational costs. Furthermore, we also intend to add more indicators to more reasonably determine the population size. For example, the distribution and diversity can help us to select appropriate individuals into the next generation according to different needs, i.e. increase or decrease, to avoid inefficient searches. Thus, how to tune the population size intelligently to face different situations will comprise one of our future works.

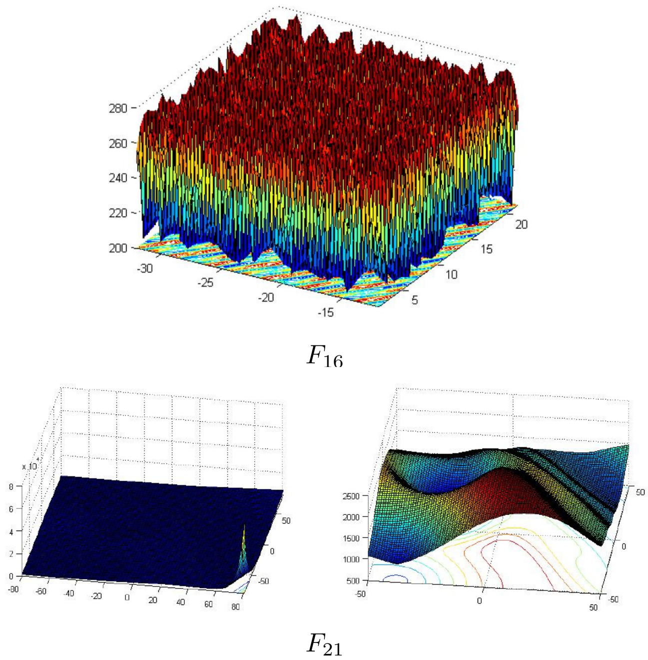


Fig. 3. The 3-D fitness landscape for 2-D function of F_{16} and F_{21}.

Finally, we apply the Wilcoxon signed-rank test to the average found optimal fitness of 30 trial runs at the termination condition and check for significant differences between the EFWA and (the EFWA + our proposed strategy). The results of the statistical tests show that our proposed strategy is effective and the acceleration effect is more obvious as the dimension increases. However, the proposal has a deleterious effect with F_{16} and F_{21}, for which the 2D fitness landscape is shown in Fig. 3. These two functions have many local areas, and different properties around their different local minima; their average convergence curves in Fig. 4 shows that our strategy is effective at the beginning of the search on F_{21}, but become worse than canonical EFWA at the end; however our strategy for F_{16} is worse throughout. This also supports our view that a reasonable allocation of population size is conducive to maintaining high performance of the FWA. We believe that F_{21} can be solved by using other methods to tune the population size in the later stage of the search. For F_{16}, however, we still need to analyze the reasons in more depth to help improve our strategy.

VI. Conclusion

We propose a strategy for FWA which dynamically tunes the population size in real-time to meet different optimization needs. Reducing the population size can further enhance the algorithm's exploitation ability to maintain a fast convergence speed, while increasing the population size can further enhance its exploration ability to increase diversity and escape from local minima. The controlled experiments confirmed that the proposal can improve FWA performance significantly, especially for high-dimensional problems.

In future work, we will continue to study the impact of the relationship between the number of firework individuals and the number of spark individuals on FWA performance. We also intend to propose an intelligent method for allocating resources based on the information gathered during the evolution.

References

[1] J. H. Holland, "Outline for a logical theory of adaptive systems," Journal of the ACM, vol. 9, pp. 297–314, 1962.

[2] M. Dorigo, V. Maniezzo and A. Colorni, "Ant system: optimization by a colony of cooperating agents," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 26, no. 1, pp. 29–41, 1996.

[3] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," Journal of global optimization, vol. 11, no. 4, pp. 341–359, 1997.

[4] J. Yu, "Study on Acceleration for Evolutionary Computation," Ph.D. dissertation, Kyushu University, Fukuoka, Japan, 2019.

[5] J. Yu, Y. Tan, and H. Takagi, "Scouting Strategy for Biasing Fireworks Algorithm Search to Promising Directions," The Genetic and Evolutionary Computation Conference Companion, 2018, pp. 99–100.

[6] J. Yu, Y. Pei, and H. Takagi, "Competitive Strategies for Differential Evolution," IEEE International Conference on Systems, Man, and Cybernetics, 2018, pp. 268–273.

[7] Y.H. Li, J. Yu, and H. Takagi, "Accelerating Fireworks Algorithm with Weight-based Guiding Sparks," 10th International Conference on Swarm Intelligence, 2019, pp. 257–266.

[8] J. Yu and H. Takagi, "Multi-species Generation Strategy-Based Vegetation Evolution," 2020 IEEE Congress on Evolutionary Computation, 2020, pp. 1–6.

[9] J. Yu and B. Niu, "Simplified Bacterial Foraging Optimization Based on Reverse Chemotaxis Strategy," 2020 IEEE Congress on Evolutionary Computation, 2020, pp. 1–6.

[10] Q.G. Xiao, C.B. Li, Y. Tang, J. Pan, J. Yu, X.Z Chen, "Multi-component energy modeling and optimization for sustainable dry gear hobbing," Energy, vol. 187, pp. 1-16, 2019.

[11] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," The First International Conference on Swarm Intelligence, 2010, pp. 355–364.

[12] S. Zheng, A. Janecek, and Y. Tan, "Enhanced fireworks algorithm," IEEE Congress on Evolutionary Computation, 2013, pp. 2069–2077.

[13] J. Yu, H. Takagi, and Y. Tan, "Fireworks Algorithm for Multimodal Optimization Using a Distance-based Exclusive Strategy," IEEE Congress on Evolutionary Computation, 2019, pp. 2215–2220.

[14] L. Liu, S. Zheng, and Y. Tan, "S-metric based multi-objective fireworks algorithm," IEEE Congress on Evolutionary Computation, 2015, pp. 1257–1264.

[15] H. Luo, W. Xu, and Y. Tan, "A Discrete Fireworks Algorithm for Solving Large-Scale Travel Salesman Problem," IEEE Congress on Evolutionary Computation, 2018, pp. 1–8.

[16] K. Ding, Y. Chen, Y. Wang, and Y. Tan, "Regional seismic waveform inversion using swarm intelligence algorithms," IEEE Congress on Evolutionary Computation, 2015, pp. 1235–124.

[17] 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, vol. 6, no. 3, pp. 1–23, 2015.

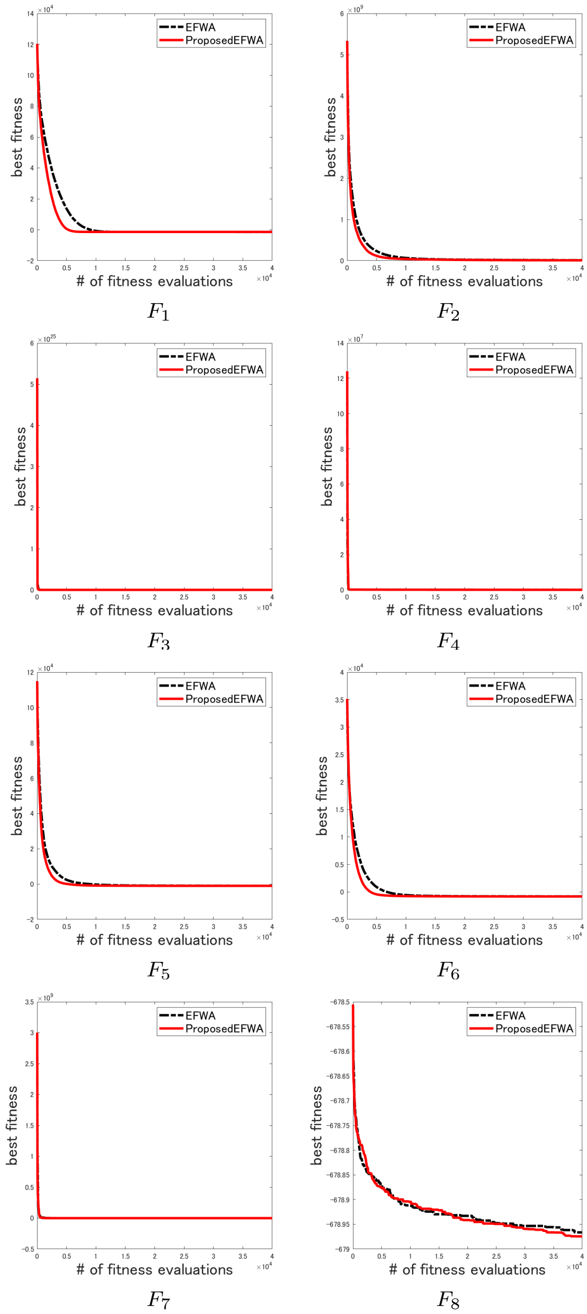
[18] M. Tuba, N. Bacanin, and A. Alihodzic, "Multilevel image thresholding by fireworks algorithm," 25th International Conference Radioelektronika, 2015, pp. 326–330.

[19] J. Yu and H. Takagi, "Acceleration for fireworks algorithm based on amplitude reduction strategy and local optima-based selection strategy," The 8th International Conference on Swarm Intelligence, 2017, pp. 477–484.

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

[21] J. Liang, B. Qu, P. Suganthan, and A. G. Hernández-Díaz, "Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization," 2013.

[22] J. Li, S. Zheng, and Y. Tan, "Adaptive fireworks algorithm," IEEE Congress on Evolutionary Computation, 2014, pp. 3214–3221.



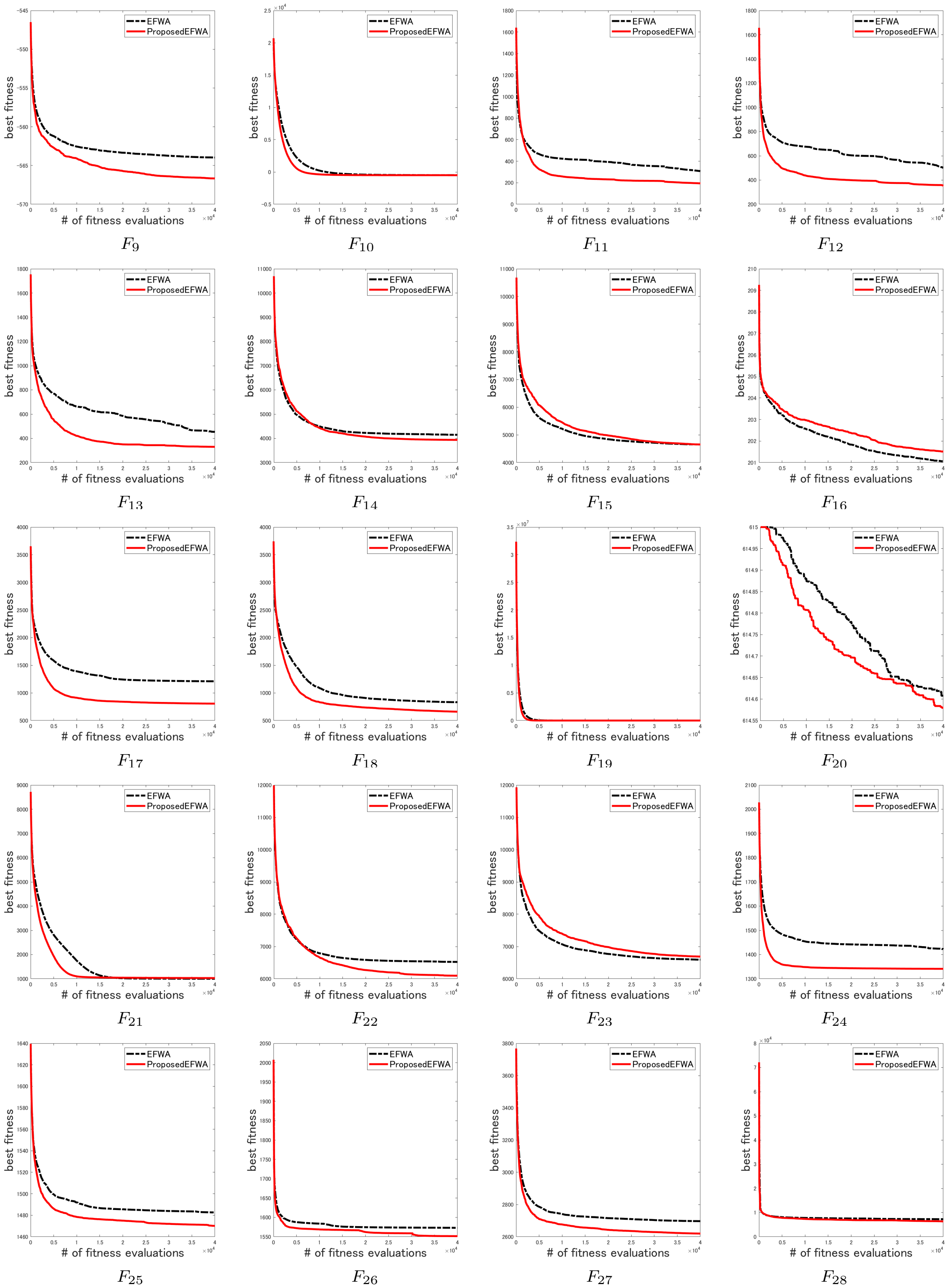


Fig. 4. Average convergence curve of 30 runs of 28 functions in 30-D.