Explosion Operation of Fireworks Algorithm

Yu, Jun
Graduate School of Design, Kyushu University

髙木, 英行
Faculty of Design, Kyushu University

https://hdl.handle.net/2324/2924868

出版情報：2020-05-01. IGI Global
バージョン：

Explosion Operation of Fireworks Algorithm

Jun Yu
Kyushu University, Japan

Hideyuki Takagi
Kyushu University, Japan

ABSTRACT

This chapter briefly reviews the basic explosion mechanism used in the fireworks algorithm (FWA) and comprehensively investigates relevant research on explosion operations. Since the explosion mechanism is one of the core operations directly affecting the performance of FWA, the authors focus on analyzing the FWA explosion operation and highlighting two novel explosion strategies: a multi-layer explosion strategy and a scouting explosion strategy. The multi-layer explosion strategy allows a firework individual to perform multiple explosions instead of the single explosion used in the original FWA, where each round of explosion can be regarded as a layer; the scouting explosion strategy controls a firework individual to generate spark individuals one by one instead of generating all spark individuals within the explosion amplitude at once. The authors then introduce several other strategies to efficiently use the information generated by the explosion operation. Finally, the authors list some open topics for discussion.

Keywords: fireworks algorithm, multi-layer explosion, scouting explosion, explosion analysis

INTRODUCTION

The fireworks algorithm (FWA) (Tan & Zhu, 2014) is a population-based meta-heuristic optimization algorithm that simulates the explosion process of real fireworks repeatedly in order to find the global optimum. Although it is a young member of the family of algorithms in the evolutionary computation (EC) community, it attracts a lot of attention from practitioners thanks to its huge potential due to its e.g. ease of use, robustness, efficiency, parallelism, and other characteristics. With the rapid increase in its popularity and application, development is booming and it has become an important branch in EC algorithms.

Since the basic FWA was first proposed in 2010, researchers have frequently proposed many effective strategies to further improve its performance. For example, Zheng et al. modified five operations used in FWA to develop a more efficient version, enhanced FWA (EFWA) (Zheng, 2013). Yu et al. used the explosion information to calculate a convergence point that has a high possibility to locate in the optimal area and used it as an elite individual to accelerate the convergence of FWA (Yu, Tan & Takagi, 2018). Pei et al. adopted different sampling methods to approximate the fitness landscape to accelerate the FWA search (Pei, 2012). Some work focuses on developing powerful hybrid algorithms by introducing operations from other EC algorithms into FWA to inherit their strengths, such as differential mutation (Yu & Kelley, 2014), covariance mutation (Yu & Tan, 2015), the gravitational search operator (Zhu, 2016), chaotic systems (Gong, 2016), and the firefly algorithm (Wang, 2019). Additionally, FWA has also been applied to solve various types of optimization problems, such as multimodal optimization (Yu, 2019), multi-objective optimization (Zhan, 2018), constrained optimization (Bacanin, 2015), dynamic optimization (Pekdemir, 2016), and large-scale optimization (Pandey, 2018).

FWA has not only flourished in an academic setting, but has also has appeared frequently in the industry in recent years. For example, FWA successfully solved the network reconfiguration required to reduce power loss and improve voltage distribution (Mohamed, 2014); it was also used to design the coefficients of a digital filter (Gao, 2011). Actually, FWA has also perfectly solved many other complex real-world problems, such as retinal image registration (Tuba, 2017), wireless sensor network coverage (Tuba, 2016), distributed resource scheduling (Reddy, 2016), image segmentation (Misra, 2017), and others.

The main objective of this chapter is to comprehensively analyze the explosion operation, one of FWA's three core operations, to thoroughly understand FWA and to highlight two effective new explosion strategies in detail. The second one is to introduce several strategies for accelerating FWA search by fully using the information generated by the explosion operation. Finally, the authors point out several potential research topics for discussion.

EXPLOSION OPERATION

FWA was created by observing the explosion phenomenon of real fireworks and believing that an explosion can be thought of as corresponding to a local search centered on a particular point. Based on this inspiration, FWA simulates an explosion operation repeatedly and, through cooperation among firework individuals, gradually evolves to the global optimal area. Similar to other EC algorithms, FWA randomly generates multiple firework individuals to form an initial population, and then each firework individual is adaptively assigned an explosion amplitude and a number of generated spark individuals according to its fitness before the explosion operation is performed. Usually, a firework individual with better fitness generates many spark individuals within a small explosion amplitude for exploitation, while a poor firework individual generates a few spark individuals within a large explosion amplitude for exploration. FWA also employs mutation operations to increase individual diversity, and spark individuals generated in this way are referred to as mutation spark individuals. Next, a selection operation is used to select firework individuals in the next generation from all current individuals, including current firework individuals, explosion spark individuals, and mutation spark individuals. The above three operations - the explosion operation, the mutation operation, and the selection operation - are repeated until a termination condition is satisfied. Fig. 1 illustrates the general optimization framework of a basic FWA.

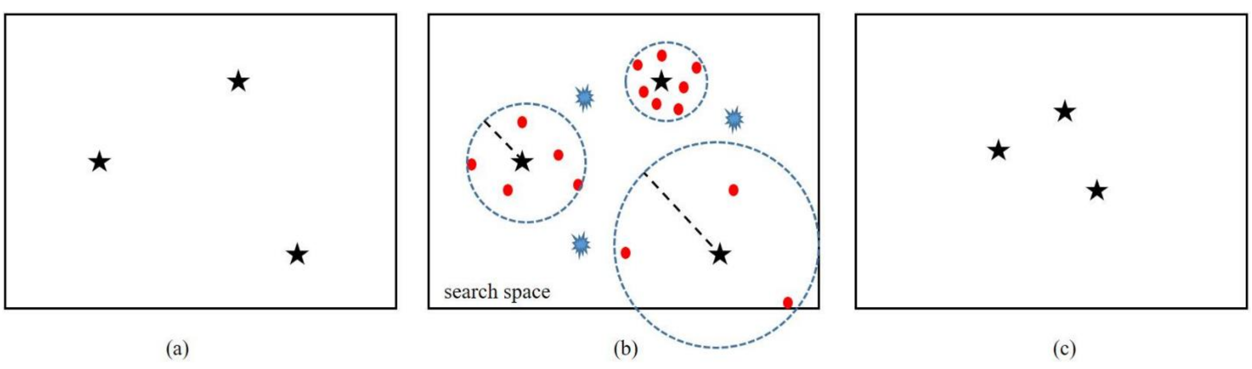

Fig. 1 The search process of the basic FWA. (a) initial firework individuals (black five-pointed stars) are randomly generated, (b), spark individuals (red solid circles) are generated around each firework individual by performing the explosion operation; mutation spark individuals (irregular blue points) are also generated by Gaussian mutation, (c) new firework individuals in the next generation are selected from all individuals in step (b). Steps (b) and (c) are iterated until a termination condition is satisfied.

Let us begin by defining symbols to describe the implementation of an explosion operation in detail. Suppose the population size is set to N, $\widehat{\mathrm{A}}$ and $\widehat{\mathrm{M}}$ represent the maximum explosion amplitude and the total number of spark individuals generated by explosion operations of all firework individuals. For a given i-th firework individual, x_i, Eq. (1.1) and (1.2) are used to determine its explosion amplitude, A_i, and the number of spark individuals, M_i, generated by the explosion operation for that individual. Note that we explain operations and algorithms using minimization problems in this Chapter, and they can be easily rewritten for maximization problem. The *spark individuals* and *a firework individual* are FWA's own terms and mean solution candidates of optimization.

$$\mathrm{A_i} = \widehat{\mathrm{A}} * \frac{f(x_i) - y_{min} + \xi}{\sum_{i=1}^{N}(f(x_i) - y_{min}) + \xi} \tag{1.1}$$

$$\mathrm{M_i} = \widehat{\mathrm{M}} * \frac{y_{max} - f(x_i) + \xi}{\sum_{i=1}^{N}(y_{max} - f(x_i)) + \xi} \tag{1.2}$$

where $f(x_i)$ returns the fitness of the i-th firework individual, and y_{min} and y_{max} are respectively the best and worst fitness found by the current population. ξ is the smallest constant in the computer to avoid a division-by-zero error.

To avoid the i-th individual generating too many or too few sparks, Eq. (1.3) is used to limit the number of spark individuals generated by the explosion operation.

$$\mathrm{M_i} = \begin{cases} round(a * \mathrm{M_i}), & if\ \mathrm{M_i} < a * \mathrm{M_i} \\ round(b * \mathrm{M_i}), & if\ b * \mathrm{M_i} < \mathrm{M_i} \\ \mathrm{M_i}, & others \end{cases} \tag{1.3}$$

where $round()$ is a rounded function, a and b are constant parameters ($0 < a < b < 1$).

The next problem is how can a firework individual generate multiple diverse spark individuals? The i-th firework individual randomly selects some dimensions, and a spark individual may undergo the effect of the explosion operation from these affected dimensions with the same random displacement. This process is repeated $\mathrm{M_i}$ times to generate $\mathrm{M_i}$ spark individuals. Thus, each firework individual performs its explosion operation according to the process described in Algorithm 1. When all firework individuals have performed the explosion operation, a selection strategy is then adopted to select firework individuals for the next generation.

Algorithm 1: The explosion process of an i-th firework individual. D: Dimension; s_j is the j-th spark individual generated by the i-th firework individual.

1. *count* = 0;
2. while *count* < $\mathrm{M_i}$ do
3. $\quad$ $\mathrm{z} = round(D * rand(0,1))$;
4. $\quad$ Randomly select z dimensions of the i-th firework individual;
5. $\quad$ Randomly generate a displacement, $h = \mathrm{A_i} * rand(-1,1)$);
6. $\quad$ for each affected k-th dimension do
7. $\quad\quad$ $s_j^k = x_i^k + h$
8. $\quad$ end for
9. $\quad$ if a generated spark individual, s_j, is outside the search area then
10. $\quad\quad$ use a mapping rule to bring back to the search area.
11. $\quad$ end if
12. $\quad$ *count ++;*

13. end while;
14. output the found global optimum.

Researchers have proposed many effective strategies to improve the performance of the explosion operation. For example, enhanced FWA (EFWA) (Zheng, 2013) proposed a new check strategy to limit the minimum explosion amplitude and a new operator to generate spark individuals through the explosion operation. Dynamic search (Zheng, 2014) is integrated into EFWA to further improve its performance by adaptively increasing or decreasing the explosion amplitude of a firework individual according to the search process. Yu et al. proposed a non-linear decreasing strategy to balance exploration and exploitation capabilities by controlling the explosion amplitude regardless of their fitness (Yu & Takagi, 2017). Li et al. proposed a new technique of guiding spark individuals using the information generated by the explosion operations to direct evolution more effectively (Li, 2017). Their explosion operation is effective and has been succeeded by many related studies. Authors extend this operator from a new perspective and propose two explosion strategies in this Chapter.

MULTI-LAYER EXPLOSION STRATEGY

A fireworks festival often attracts many people to enjoy the visual feast of various fascinating firework explosions. The explosive patterns formed by real fireworks are not only simple spheres, but also many customized explosive effects, e.g. several explosion shapes such as circle, heart, fan, star, or willow, single or multiple explosion stages, and their combinations. Through the observation and review of these explosion patterns, Yu et al. creatively proposed a multi-layer explosion strategy (Yu, Takagi & Tan, 2018) to fully exploit the local fitness landscape instead of the single-layer spherical explosion strategy used normally in FWA, where each round of explosions can be considered as a layer. Although the maximum number of explosion layers can theoretically be set to any positive integer, it should be set reasonably according to the characteristics of the optimization problem and calculation cost, e.g. the total number of fitness evaluations and CPU consumption time. Fig. 2 illustrates the overall optimization process of the multi-layer explosion strategy.

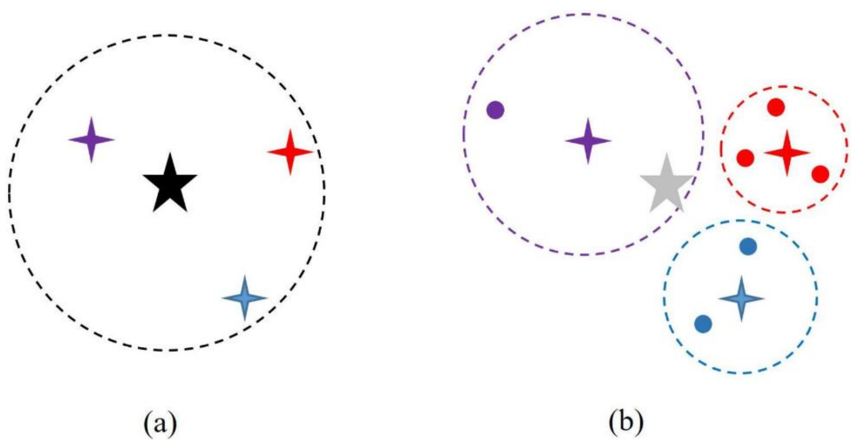

Fig. 2 The general framework of a multi-layer explosion strategy. (a) a few spark individuals (colored four-pointed stars) are generated by a firework individual (a black five-pointed star) in the first layer, (b) these generated spark individuals, rather than the firework, individually trigger the next round of explosions and generate new spark individuals in the subsequent layer, and the newly generated spark individuals continue to perform the next layer of explosion operations until a maximum predefined number of layers is reached.

First Layer Explosion

An explosion operation is determined by two parameters: the number of generated spark individuals and the explosion amplitude. Similar to basic FWA, Eq. (1.1) is used to adaptively determine the explosion amplitude of a firework individual according to its fitness.
However, the difference here is that all firework individuals generate the same number of few spark individuals in the first layer regardless of their fitness. Since this strategy does not focus on how to generate a spark individual, Algorithm 1 or any other method can be employed to generate the spark individuals in the first layer. Suppose the number of spark individuals generated by the i-th firework individual is set to $m_i^{(1)}$ all spark individuals generated in the first layer, $m_i^{(1)} * n$, start the next round of explosion operations instead of the firework individuals.

Subsequent Layer Explosion

Suppose the maximum number of explosion layers is set to L, and m_i is the total number of spark individuals in all layers under the i-th firework individual, i.e. $\widehat{\mathrm{M}} = \sum_{i=1}^{N} m_i$. $m_i^{(k)}$ represents the number of generated spark individuals in the k-th explosion layer ($\mathrm{k} < \mathrm{L}$) and $m_i = \sum_{i=1}^{L} m_i^{(k)}$.

To perform explosion operations in subsequent layers smoothly, the first problem to be solved is how to allocate the remaining assignable spark individuals, $\mathrm{M_i} - m_i^{(1)} * n$. Since the i-th firework individual and m_i spark individuals form a subgroup, the number of spark individuals in subsequent layers, $m_i - m_i^{(1)}$, is dynamically assigned to each subgroup according to the fitness of its initial firework individual using the Eq. (1.2). Next, each subsequent layer in the i-th subgroup is evenly allocated a number of spark individuals, i.e. the number of spark individuals in the k-th layer is set to $\frac{m_i - m_i^{(1)}}{L-1}$.

The other key problem is to decide on the setting of the two parameters, the number of spark individuals and the explosion amplitudes, for spark individuals in the $(\mathrm{k}-1)$ layer to trigger the k-th round of explosions. These two parameters are also dynamically determined by the fitness of the spark individuals in the $(\mathrm{k}-1)$ layer using Eq. (1.1) and (1.2), and then this layer explosion is repeated until the explosions have been repeated $(\mathrm{L}-1)$ times. Here, the Algorithm 2 summarizes the overall optimization process of a multi-layer explosion strategy.

Algorithm 2: The general framework of a multi-layer explosion strategy.

1. for $i = 0$; $i < \mathrm{N}$; $i++$ do
2. Decide the number of spark individuals, $m_i^{(1)}$, in the first layer for the i-th firework individual;
4. Decide an explosion amplitude for the i-th firework individual;
5. Perform the first layer explosion for i-th firework to generate spark individuals;
6. end for
7. while the number of explosions does not reach a predefined L layer do
8. Perform the next round of explosion operations for each spark individual in the previous layer described in the sub-section **Subsequent Layer Explosion**.
9. end while
10. end of the multi-layer explosion.

The multi-layer explosion strategy does not need to change the optimization framework of the basic FWA or add additional fitness calculations, and can be combined with any other FWA variants easily by replacing their explosion operation. Its core idea is to use the characteristics of the local fitness landscape to fully generate promising spark individuals by exploding layer by layer. Additionally, because spark individuals are derived from both firework individuals and spark individuals in previous layers, the diversity of the population is greatly increased. Thus, this strategy can explore local landscape information deeply and not easily fall into local minima, especially useful for some complex multi-modal problems.

Although the maximum number of explosive layers can be set to any positive number, the authors do not recommend setting it too large. When the number of layers is greater than 2, each spark individual can only generate one new individual because the total number of spark individuals in subsequent layers is equally distributed. To overcome this limitation, practitioners can use a non-equal method to assign the number of spark individuals in subsequent layers, or develop an adaptive version to tune the maximum number of explosion layers according to the optimization process. Additionally, some poor spark individuals can be forbidden from exploding again to save resources and transfer this part of the resources to promising individuals to explore. In short, there is still a lot of room and value to further improve with this strategy.

SCOUTING EXPLOSION STRATEGY BACKGROUND

There is a phenomenon that often occurs during a large fireworks festival where real fireworks can explode repeatedly in some specific directions rather than spreading everywhere. Inspired by these observations, the scouting explosion strategy (Yu, Tan & Takagi, 2018) is introduced into the FWA to increase local search capabilities and bias toward promising areas by generating spark individuals one by one instead of generating them all at once. Fig. 3 illustrates the general explosion effect of a firework individual when adopting a scouting strategy.

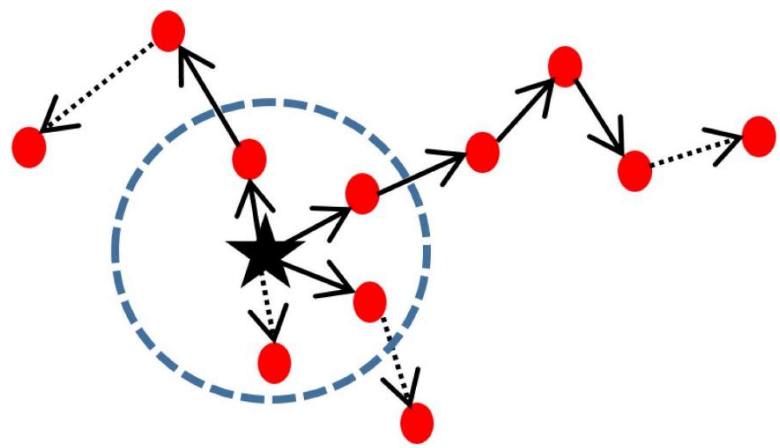

Fig. 3 An instance of a scouting explosion strategy for a firework individual. A black five-pointed star and red solid points represent respectively a firework individual and its spark individuals. Solid arrows indicate that the current direction is promising and continues to be tracked, whereas dotted arrows indicate that the continuous scouting stops here and new scouting starts from the initial point.

Since this strategy also focuses on generating spark individuals in a more efficient way without modifying the overall optimization framework of FWA, the only difference is that it gradually generates spark individuals to track promising areas, which gives the appearance of a scout looking for a safe direction forward in an unknown environment.

Take the i-th firework individual as an example; the following explosion operation replaces the original operation used in the FWA to develop a new FWA version. First, only one spark individual is generated at a time and compared with the i-th firework individual. If the spark individual is worse, the i-th firework individual generates a new spark individual randomly again until a better spark individual emerges. The better spark individual becomes the center of the next explosion operation to generate a new spark individual. This process is repeated until the $(\mathrm{j}+1)$-th spark individual is worse than the j-th spark individual, then the current continuous scouting stops and a new scouting starts from the initial firework individual. Since an explosion operation only generates one spark individual, the i-th firework individual repeats $\mathrm{M_i}$ times to generate multiple spark individuals. Finally, all firework individuals perform the scouting explosion strategy described in the Algorithm 3 in turn to generate spark individuals.

Algorithm 3: A scouting explosion strategy is used for the i-th firework individual to generate $\mathrm{M_i}$ spark individuals.

1. Determine the total number of spark individual, $\mathrm{M_i}$, generated by the i-th firework individual;
2. Determine the explosion amplitude, $\mathrm{A_i}$, of the i-th firework individual;
3. $i = 0$;
4. while i is less than S_i do
5. Randomly generate a spark individual within $\mathrm{A_i}$ and evaluate it.
6. $i = \mathrm{i} + 1$;
7. if the spark individual is worse than the firework individual then
8. Go to step 4;
9. end if
10. while the subsequent spark individual is better than previous one (firework or spark individual) do
11. Generate a new spark individual randomly around the previous individual rather than the firework within $\mathrm{A_i}$;
12. $i = \mathrm{i} + 1$;
13. end while
14. end while.

The scouting explosion strategy calculates the number of generated spark individuals, $\mathrm{M_i}$, and explosion amplitude, $\mathrm{A_i}$, for the i-th firework individual in the same way as in basic FWA; i.e. according to Eq. (1.1) and (1.2). However, it generates spark individuals one by one to dig deeper into local directions with high potential instead of randomly exploring within an explosion amplitude; this allows it to make better use of the local fitness landscape to speed up the FWA search because more resources (fitness evaluations) are allocated to areas of high potential. Since many promising spark individuals have an opportunity to generate their offspring individuals, population diversity is greatly enriched, avoiding the population becoming tracked in local areas. Obviously, this strategy can be easily combined with other FWA variants and only requires that they replace their explosion operations without any tedious modification. This strategy can thus be said to be a *low-cost, high-reward* strategy.

Although a single point tracking method is used to explore the local information, other tracking methods are also acceptable. For example, practitioners can try to use a multi-point tracking or tree tracking method to further reduce the risk of falling into a locally optimal area. Additionally, how to adaptively tune an explosion amplitude during the process of a tracking search is also a promising topic for future exploration.

IMPROVEMENTS ON EXPLOSION OPERATION

The explosion operation is one of the core components directly affecting FWA performance. The operation has two parameters that control the generation of the many spark individuals that are used to achieve a local search. However, this raises a problem in that many spark individuals are destroyed quickly after only participating in the selection operations; this is a great waste of limited resources (fitness evaluations) and the approach does not make reasonable use of the spark individuals. Here, the authors list two efficient strategies for controlling the parameters and making full use of existing spark individuals to better guide evolution than a random search.

Amplitude Reduction Strategy

The basic FWA dynamically determines the explosion amplitudes of firework individuals according to their fitness in order to balance well both exploration and exploitation; the better the firework individual, the smaller the explosion amplitudes. Since firework individuals become similar with the process of convergence while the maximum explosion amplitude, $\hat{A}$, is fixed in each generation, all firework individuals will share the same maximum amplitude,$\hat{A}$, equally. This results in the exploitation ability not being highlighted, which makes it difficult for FWA to converge. An amplitude reduction strategy (Yu & Takagi, 2017) is introduced to nonlinearly reduce the explosion amplitude of firework individuals regardless of their fitness. This means that each firework individual is treated equally and its explosion amplitude is determined by Eq. (1.4). Fig. 4 illustrates the process by which the explosion amplitudes are changed throughout the whole search period.

$$A_i = \begin{cases} \hat{A} * \left(1 - \frac{FE_{cur}}{FE_{max}}\right), & if \;\; FE_{cur} < c * FE_{max} \\ \hat{A} * (1 - c) & Others \end{cases} \tag{1.4}$$

where FE_{cur} and FE_{max} are the current and maximum number of fitness evaluations, respectively; and c is a constant to avoid an explosion amplitude becoming too small.

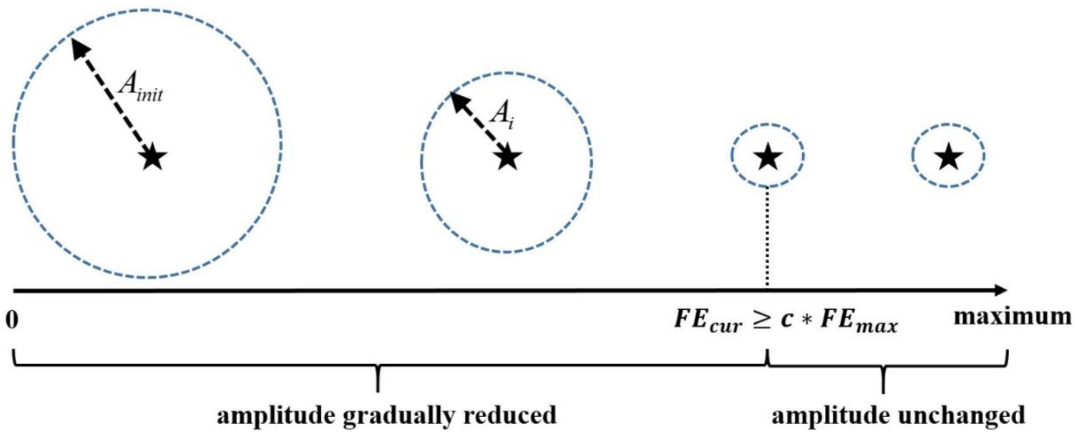

Fig. 4 Changes in explosion amplitudes of firework individuals throughout the whole search period.

This strategy emphasizes that all firework individuals possess the exploration ability in the early stages of the search process to quickly find areas of potential. As the convergence of the population progresses, this ability becomes gradually weaker while the exploitation ability becomes gradually stronger to improve convergence accuracy in the later stages of search process. Its core is to switch between two different abilities during different search stages. Obviously, applying appropriate methods for controlling explosion amplitudes is a key factor affecting performance, so developing other methods is a potential research direction.

Synthetic Spark Individuals

A firework individual usually generates multiple spark individuals within its explosion amplitude. Although these spark individuals can carefully search local areas, it is difficult for them all to survive to the next generation and most of them are abandoned after the selection operation is performed. To improve the efficiency of local fitness information utilization, a new type of individual, the synthetic spark individual (Yu, Tan & Takagi, 2018), is proposed.

A firework individual and its generated spark individuals are considered to belong to the same sub-area, and each sub-area can thus calculate a synthetic spark individual by fully using these spark individuals. Take any given i-th firework individual as an example. The first step is to construct $\mathrm{M_i}$ vectors from the firework individual to each spark individual. If an offspring individual (spark) is better than the parent individual (firework), this direction is considered to have potential. Otherwise, its opposite direction is considered to be potential and involved in the calculation of a synthetic spark individual. Next, the absolute value of the fitness difference between the endpoint and the start point of a vector is employed to evaluate the potential of these vectors even if the opposite direction is used, i.e. the larger the fitness difference, the higher the weight of the vector. Finally, a synthetic spark individual is calculated by weighting these vectors using Eq. (1.5); Fig. 5 illustrates the construction of a synthetic spark individual.

$$v_i = \sum_{j=1}^{\mathrm{M_i}} \frac{f(x_i)-f(s_j)}{\sum_{j=1}^{\mathrm{M_i}} ||f(x_i)-f(s_j)||} * (s_j - x_i) + x_i \quad (1.5)$$

where s_j is the j-th generated spark individual or antipodal individual in the i-th sub-group, and v_i is the i-th synthetic spark individual.

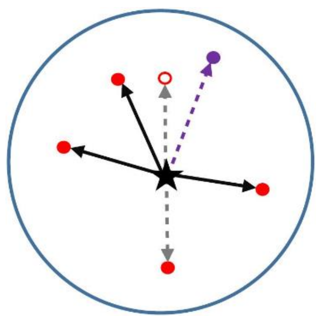

Fig. 5 A synthetic spark individual is calculated from a firework individual (black five-pointed star) and its generated spark individuals (the red solid points). The presence of a red hollow circle means that the antipode has been used. The purple solid point is the synthetic spark individual obtained by weighting these vectors.

Since a synthetic spark individual reuses multiple spark individuals, it has good anti-noise properties and a high possibility to find a better potential direction - although it does require an additional fitness evaluation. To avoid additional fitness costs, the reverse symmetry points of poor spark individuals are not evaluated but participate in the calculation, and the fitness difference of the original vector is roughly used to evaluate a used antipodal direction. However, these factors may cause the accuracy of synthetic spark individuals to decrease. One of the key tasks in the future is to further improve accuracy from the cost-performance point of view. For example, gradient information can replace the fitness difference of vectors to evaluate their weights more objectively. In addition, how to effectively use synthetic spark individuals is also a task that must be solved in the near future.

SUMMARY

This chapter focuses on analyzing the effects of the FWA's explosion operation. The basic explosion operation is first reviewed in detail, and two novel explosion strategies are demonstrated to accelerate FWA search. Attention is next focused on introducing some interesting strategies by which FWA performance could be further enhanced by controlling parameters or reusing existing information efficiently. Although all of these strategies have achieved satisfactory results, the authors finally introduce some other topics worthy of further study and discussion.

- How can diverse spark individuals be generated from a firework individual? Since the relationship between parent individuals and offspring individuals is one-to-many, i.e. lots of spark individuals are derived from a few firework individuals, it is difficult to generate diverse spark individuals when all firework individuals have become similar. It also means that once we have become trapped in a local minimum, it is difficult to jump out. How to increase the diversity of the population is thus one of the problems that need to be solved.

- How can the many generated spark individuals be used most efficiently? As described above, all spark individuals are involved in the selection operation, but only a few survive to the next generation. It is a great waste of limited resources, especially in expensive problems. Thus, it is an effective approach to accelerate FWA convergence by mining more hidden information from these spark individuals.

- How can resources be allocated appropriately to achieve a stronger performance? Due to the limitations of any real-world implementation, the maximum number of fitness evaluations often cannot be set too large. Furthermore, different optimization problems have different characteristics. Thus, balancing resource allocation between firework individuals and spark individuals as well as setting suitable parameters to solve various problems is also an important topic.

ACKNOWLEDGMENT

This work was supported in part by Grant-in-Aid for Scientific Research (17H06197, 18K11470, 19J11792).

REFERENCES

Bacanin, N., & Tuba, M. (2015). Fireworks algorithm applied to constrained portfolio optimization problem. In Proceedings of 2015 IEEE Congress on Evolutionary Computation (pp. 1242-1249). Sendai, Japan.

Gao, H., & Diao, M. (2011). Cultural firework algorithm and its application for digital filters design. International Journal of Modelling, Identification and Control, 14(4), 324-331.

Gong, C. (2016). Chaotic adaptive fireworks algorithm. In Proceedings of the Seventh International Conference on Swarm Intelligence (pp. 515-525). Bali, Indonesia.

Li, J.Z., Zheng, S.Q., & Tan, Y. (2017) The Effect of Information Utilization: Introducing a Novel Guiding Spark in the Fireworks Algorithm. IEEE Transactions on Evolutionary Computation, 21(1), 153-166.

Misra, P.R., & Si, T. (2017). Image segmentation using clustering with fireworks algorithm. In Proceedings of 11th International Conference on Intelligent Systems and Control (pp. 97-102). Coimbatore, India.

Mohamed Imran, A., & Kowsalya, M. (2014). A new power system reconfiguration scheme for power loss minimization and voltage profile enhancement using Fireworks Algorithm. International Journal of Electrical Power and Energy Systems, 62, 312-322.

Pandey, V.C., Jadoun, V.K., Gupta, N., Niazi, K.R., & Swarnkar, A. (2018). Improved Fireworks Algorithm with Chaotic Sequence Operator for Large-Scale Non-convex Economic Load Dispatch Problem. Arabian Journal for Science and Engineering, 43(6), 2919-2929.

Reddy, K.S., Panwar, L.K., Kumar, R., & Panigrahi, B.K. (2016). Distributed resource scheduling in smart grid with electric vehicle deployment using fireworks algorithm. Journal of Modern Power Systems and Clean Energy, 4(2), 188-199.

Pei, Y., Zheng, S.Q., Tan, Y., & Takagi, H. (2012). An empirical study on influence of approximation approaches on enhancing fireworks algorithm. In Proceedings of 2012 IEEE International Conference on Systems, Man, and Cybernetics (pp. 1322-1327). Seoul, Korea.

Pekdemir, H., & Topcuoglu, H.R. (2016). Enhancing fireworks algorithms for dynamic optimization problems. In Proceedings of 2016 IEEE Congress on Evolutionary Computation (pp. 4045-4052). Vancouver, Canada.

Tan, Y., & Zhu Y.C. (2010). Fireworks algorithm for optimization. In Proceedings of The First International Conference on Swarm Intelligence (pp. 355-364). Beijing, China.

Tuba, E., Tuba, M., & Dolicanin, E. (2017). Adjusted fireworks algorithm applied to retinal image registration. Studies in Informatics and Control, 26(1), 33-42.

Tuba, E., Tuba, M., & Simian, D. (2016). Wireless sensor network coverage problem using modified fireworks algorithm. In Proceedings of 2016 International Wireless Communications and Mobile Computing Conference (pp. 696-701). Cyprus.

Wang, X., Peng, H., Deng, C., Li, L., & Zheng, L. (2019). An improved firefly algorithm hybrid with fireworks. In Proceedings of 10th International Symposium on Intelligence Computation and Applications (pp. 27-37). Jiujiang, China.

Yu, C., Kelley, L., Zheng, S., & Tan, Y. (2014). Fireworks algorithm with differential mutation for solving the CEC 2014 competition problems. In Proceedings of 2014 IEEE Congress on Evolutionary Computation (pp. 3238-3245). Beijing, China.

Yu, C., & Tan, Y. (2015). Fireworks algorithm with covariance mutation. In Proceedings of 2015 IEEE Congress on Evolutionary Computation (pp. 1250-1256). Sendai, Japan.

Yu, J., Tan, Y., & Takagi, H. (2018). Accelerating Fireworks Algorithm with an Estimated Convergence Point. In Proceedings of the Ninth International Conference on Swarm Intelligence (pp. 263-272). Shanghai, China.

Yu, J., & Takagi, H. (2017) Acceleration for Fireworks Algorithm Based on Amplitude Reduction Strategy and Local Optima-based Selection Strategy. In Proceedings of the Eighth International Conference on Swarm Intelligence (pp. 477-484). Fukuoka, Japan.

Yu, J., Takagi, H., & Tan, Y. (2018) Multi-layer Explosion-Based Fireworks Algorithm. International Journal of Swarm Intelligence and Evolutionary Computation, 7(3), 1-9. DOI:10.4172/2090-4908.1000173.

Yu, J., Takagi, H., & Tan, Y. (2019). Fireworks Algorithm for Multimodal Optimization Using a Distance-based Exclusive Strategy. In Proceedings of 2019 IEEE Congress on Evolutionary Computation (pp. 2216-2221). Wellington, New Zealand.

Yu, J., Tan, Y., & Takagi, H. (2018) Scouting Strategy for Biasing Fireworks Algorithm Search to Promising Directions. In Proceedings of the genetic and evolutionary computation conference companion (pp.99-100). Kyoto, Japan.

Zhan, D., & Xie, C. (2018). An Improved Multi-Objective Fireworks Algorithm. In Proceedings of 9th International Symposium on Intelligence Computation and Applications (pp. 204-218). Guangzhou, China.

Zheng, S., Janecek, A., & Tan, Y. (2013). Enhanced fireworks algorithm. In Proceedings of 2013 IEEE Congress on Evolutionary Computation (pp. 2069-2077). Cancun, Mexico.

Zhu, Q.B., Wang, Z.Y., & Huang, M. (2016). Fireworks algorithm with gravitational search operator. Kongzhi yu Juece/Control and Decision, 31(10), 1853-1859. (in Chinese).

Zheng, S.Q., Janecek, A., & Tan, Y. (2013) Enhanced Fireworks Algorithm. In Proceedings of 2013 IEEE Congress on Evolutionary Computation (pp. 2069-2077). Cancun, Mexico.

Zheng, S.Q., Janecek, A., Li, J.Z., & Tan, Y. (2014) Dynamic search in fireworks algorithm. In Proceedings of 2014 IEEE Congress on Evolutionary Computation (pp. 3222-3229). Beijing, China.