A Parallel Optimization Algorithm Based on Communication Strategy of Pollens and Agents

Pei-Wei Tsai[1,2], Trong-The Nguyen[3], Jeng-Shyang Pan[1,2], Thi-Kien Dao[3,*], and Wei-Min Zheng[4]

[1] College of Information Science and Engineering,
[2] Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, China
[3] Department of Information Technology, Hai-Phong Private University, Vietnam
[4] Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093,China

vnthe@hpu.edu.vn

Abstract. Unwanted convergence to a local optimum, rather than global optimum, is possible to take place in practical multimodal optimization problems. Communication between artificial agents in the stochastic algorithms is one of the solutions to this issue. This paper proposes a novel parallel optimization algorithm, namely FDA, based on the communication of the pollen in Flower pollination algorithm (FPA) with the agents in Differential evolution algorithm (DEA) to solve the optimization problems. A communication strategy for Pollens and Agents is to take advantages of the strength points of each algorithm to explore and exploit the diversity solutions in avoiding of dropping to a local optimum. A set of benchmark functions is used to test the quality performance of the proposed algorithm. Simulation results show that the proposed algorithm increases the accuracy more than the existing algorithms.

Keywords: Parallel optimization algorithm; Differential evolution algorithm; Flower pollination algorithm.

1 Introduction

Parallel processing plays an important role in efficient and effective computations of function optimization, because of it is an essential requirement for optimum computations in modern equipment [1][2]. The parallelized strategies simply share the computation load over several processors. The sum of the computation time for all processors can be reduced compared with the single processor works on the same optimum problem. Moreover, due to physical constraints in real-world problems, there are different optimal solutions in the search space and the best results of the obtained optimum cannot always be realized [3]. Collaboration between two algorithms is to take the strength points of the two algorithms. The strength point of the algorithms is considered

J.-S. Pan et al. (eds.), *Advances in Intelligent Information Hiding and Multimedia Signal Processing*, Smart Innovation, Systems and Technologies 64,
DOI 10.1007/978-3-319-50212-0_38

motivation to merge them in parallel to overcome the issue of earlier dropping a local optimal.

The communications among agents while parallel processing would enhance the co-operating individuals, share the computation load, and increase the diversity optimizations. They can exchange information between the populations whenever a communication strategy is triggered. The parallelized structure can prove faster speed and more accuracy than the original structure, and it also extends the global search capacity than its original one [4].

Furthermore, advantages of the meta-heuristics that included the Flower pollination algorithm (FPA) [5], and Differential evolution algorithm (DEA)[6] have been considered for recently. The FPA was inspired from the pollination behaviors of flowering plants and the real-life processes of the flower pollination such as self-pollination or cross-pollination were mimicked for modeling mathematically for FPA. The cross-pollination can be considered as the global pollination because of the pollinators such as agents, bats, birds and flies can fly the long distances. However, the self-pollination is the fertilization of one flower, such as peach flowers, from the pollen of the same flower or different flowers of the same plant, thus they can be considered as the local search. Similarly, DEA is an optimization algorithm based on the stochastic direct search. This algorithm is to optimize a problem by iteratively trying to improve a candidate solution according to a fitness value. There are lots of advantages of these algorithms, and many applications have been solved successfully by them[7]. However, these algorithms also have the disadvantages such as a premature convergence in the later search period and the accuracy of the optimal value which cannot meet the requirements sometimes [8] [7].

In this paper, the communication strategy and the concepts of the parallel processing are applied to develop a diversity enhanced optimization. In the proposed method, the several weaker individuals in FPA will be replaced with the better artificial agents from DEA after fixed iterations. On the contrary, the poorer agents of DEA will be replaced with the better pollens of FPA. The benefit of this strategy is to avoid the locally converged optimal in complex constrained optimization problems.

2 Related work

Flower Pollination Algorithm.

Flower Pollination Algorithm (FPA) was emulated the characteristic of the biological flower pollination in flowering plant [5]. In this algorithm, the rules of the flowering plant were mimicked to formulate the equation of optimization in as follows.

1. The global pollination processes are biotic and cross-pollination through which the pollen transports pollinators in a way that obeys Lévy flights.
2. Local pollination is explored as abiotic and self-pollination. Reproduction probability is considered as flower constancy which is proportional to the resemblance of the two flowers in concerned.
3. The switching probability $p \in [0, 1]$ can be used to control between the local and global pollination.

4. Local pollination can have fraction p that is significant in the entire processes of the pollination because of physical proximity and the wind. To simplify the proposed algorithm development, it was assumed that each plant has a single flower and each flower emit only a single pollen gamete. This means that a flower or pollen gamete is viewed as a solution xi to a problem. FPA was designed with major stages as global and local pollination. To model the local pollination, both rule 2 and rule 3 can be represented ass.

$$x_i^{t+1} = x_i^t + u(x_j^t - x_k^t) \tag{1}$$

where x_j^t and x_k^t are pollen from different flowers of the same plant species. u is drawn from a uniform distribution in [0, 1] and it is considered as a local random walk, if x_j^t and x_k^t comes from the same species or selected from the same population. In the global pollination, the pollens of the flowers are moved by pollinators e.g. insects and pollens can move for a long distance since the insects typically fly for a long range of distances. This process guarantees pollination and reproduction of the fittest solution represented as g^*. The flower constancy can be represented mathematically as:

$$x_i^{t+1} = x_i^t + \gamma \times L(\lambda) \times (x_i^t - g^*) \tag{2}$$

where x_i is solution vector at iteration t, γ is a scaling factor to control the step size. Lévy flight can be used to mimic the characteristic transporting of insects over a long distance with various distance steps, thus, $L > 0$ from a Lévy distribution.

$$L = \frac{\lambda\Gamma(\lambda)\times\sin(\frac{\pi\lambda}{2})}{\pi\times s^{i+\lambda}}, \ (s \gg s_0) \tag{3}$$

where $\Gamma(\lambda)$ is the standard gamma function, and this distribution is valid for large steps $s > 0$.

The switching probability or the proximity probability p can be effectively used likely in the rule (4) to switch between common global pollination to intensive local pollination. The effectiveness of the PFA can be attributed to the following two reasons: In rule 1, insect pollinators can travel long distances which enable the FPA to avoid local landscape to search for a very large space (explorations). In rule 2, the FPA ensures that similar species of the flowers are consistently chosen which guarantee fast convergence to the optimal solution (exploitation). To begin with, a naive value of p = 0.55 can be set as an initial value. A preliminary parametric showed that p = 0.8 might work better for most applications.

Differential Evolution Algorithm

Differential evolution algorithm (DEA) [9] belongs to the class of genetic algorithms (GAs)[10] which use biology-inspired operations of crossover, mutation, and selection on a population in order to optimize an objective function over the course of successive generations. DEA have four main operations included initialization, mutation, crossover, and selection. Evolution is performed on a population of solutions and for a certain number of generations. The following steps show how DEA works:

Step1 Initialization: an initial population of N agents is generated randomly. Each agent is a candidate solution containing D dimension of unknown parameters. The population evolves through successive generations.

$$x_{j,i,0} = x_{j,min} + rand_j(0,1) \times \left(x_{j,max} - x_{j,min}\right);$$
$$j = 1,2,..D, i = 1,2,..N; rand_j(0,1) \sim U(0,1) \tag{4}$$

where $x_{j,i,0}$ is a vector indicates an agent in a population belonging to a current generation G. $x_{i,G} = \left[x_{1,i,G}, x_{2,i,G}, .. x_{D,i,D}\right], i = 1,..N; G = 1,..G_{max}$. All agents in a population are generated by enforcing the constraint of boundaries in which $x_{min} \leq x_{i,G} \leq x_{max}$, where $x_{min}$is set to[$x_{1,min}, \ldots, x_{D,min}$] and $x_{max}$is set to[$x_{1,max}, \ldots, x_{D,max}$].

Step2 Mutation: after the initialization, DEA runs a mutation to explore the search space. There are some mutation strategies that denoted as DE/x/y/z. It specifies the DEA mutation strategies by indicating the vector /x/ to be perturbed, the number /y/ of difference vectors used to perturb /x/, and the type /z/ of crossover. In this paper, the original DEA is considered. A vector $v_{i,G}$ is computed by each vector $x_{i,G}$.

$$v_{i,G} = x_{r_{i1}G} + F \times (x_{r_{i2}G} + x_{r_{i3}G}) \tag{5}$$

where $F \in (0, 2)$ is a factor of scaling variable to speed up convergence of the DEA; the indexes $r_{i,1}, r_{i,2}$, and $r_{i,3}$ are mutually exclusive integers randomly selected from the interval $[1, N]$.

Step 3 Crossover: a crossover operation recombines agents to a new solution. It can make up increasing the diversity in the population but including successful solutions from the previous generation. Usually, DEA adopts exponential or binomial crossover schemes. Here, the binomial crossover is used. It changes components that are chosen randomly from $\{1, 2, \ldots, D\}$ and makes the number of parameters inherited from the mutant obey a nearly binomial distribution. A new candidate solution is calculated as given.

$$u_{j,i,G} = \begin{cases} v_{j,i,G} & if\ rand_{j,i}(0,1) \leq CR\ or\ j = j_{rand} \\ x_{j,i,G} & otherwise \end{cases} \tag{6}$$

where $u_{j,i,G}$ is new a trial vector that $u_{i,G}$ assumpted as $\left[u_{1,i,G}, u_{2,i,G}, \ldots, u_{D,i,G}\right]$, with $u_{i,G} \neq x_{i,G}$; CR is crossover rate with $CR \in (0,1)$; $rand_{j,i}(0,1) \sim U(0,1)$; $j_{rand} \in \{1,2, \ldots, D\}$. The constant $0 < CR < 1$ obviously affects the amount of crossover operations. Usually, 0.6 $< CR <$ 1 is a good value for fast convergence.

Step 4 Selection: the population size constant is kept in consecutive generations. This operation determines if the vector $v_{i,G}$ or the vector $u_{i,G}$ survives in the next generation. The selection operation works by the following relations.

$$x_{i,G+1} = \begin{cases} u_{i,G}\ if\ f(u_{i,G}) \leq f(x_{i,G}) \\ x_{i,G}\ if\ f\left(u_{i,G}\right) > f(x_{i,G}) \end{cases} \tag{7}$$

where $f(.)$ is the objective function to be optimized. If the value given by $u_{i,G}$ is lower than the value of $x_{i,G}$, then $u_{i,G}$ replaces $x_{i,G}$in the next generation, otherwise $x_{i,G}$is kept. Therefore, the population can improve or be the same in optimization of the the $f(.)$, but it never becomes worst.

After selection, the algorithm goes back to iterate *Step 2*. Mutation, crossover, and selection are applied until a certain condition i.e. maximization of the number of generations G_{max} or minimization of the objective function stops iterations.

3 A Communication Strategy Agents and Pollens

The advantages of both FPA[5] and DEA[9] algorithms are strong robustness, fast convergence, and high flexibility. They have been applied to solve successfully many problems in engineering, financial, and management fields [11, 12]. However, the disadvantage of them also exists such as the premature convergence in the later search period. This could make the accuracy of their optimal values are not to meet the requirements in sometimes. It could be easy to converge to a local optimum if the swarm size is too small for searching solution based on its own best historical information. This issue could be overcome by applying the enhanced optimizations. Enhanced optimization can be implemented by constructing the communication between two algorithms. The exchange information among subpopulations can be figured out whenever the communication strategy triggers. The communication strategy for exchanging information between Agents and Pollen can be described as follows. The best agents in DEA could be copied to move to other subpopulations in FPA replace the poorer pollens of them, and update the positions of all subpopulations in every period of exchanging time. The flow information of communicating the agents and pollens is employed with the communication strategy. In contrast, the finest artificial pollens among all the flowers of FPA's population would migrate to the weaker agents in DEA, replace them and update all positions for each population during every period exchanging time.

A parallel structure is made up of several groups by dividing the population into subpopulations. The diversity agents for the optimal method are built based on constructing of the parallel processing. The subpopulations are evolved into regular iterations independently. The advantages of each side of algorithms are taken into account by replacing the poorer individuals of them with the finest ones, and the benefit of cooperation between them is archived. During all iterations of the proposed method FDA, the exchanging period time of communication between FPA and DEA is set to R. The population size of FDA is set to N. The numbers of the population sizes of DEA and FPA are N_1 and N_2 be set to $N/2$ respectively. The top fitness *k* agents of in group with N_1 will be copied to the place of worst agents in group with N_2 for replacing the same number of the agents, where t is the current iteration, during running with $\cap\ R \neq \emptyset$. The description of the proposed method can be summarized the basic steps as follows.

Step 1. Initialization: Population size of FDA is generated randomly by initializing the solutions of FPA and DEA. The number iteration of R is defined for executing the communication strategy. The N_1 and N_2 are the numbers of agents and pollens in solutions X_{ij}^t and S_{ij}^t for populations of DEA and FPA respectively, $i = 0, 1, .., N_{1,2} - 1, j = 0, 1, .. D$. where D is dimension of the solutions and t is current iteration with setting initializing to 1.

Step 2. Evaluation: The fitness function values of $f_2(X_{ij}^t)$, $f_1(S_{ij}^t)$ are evaluated by both DEA and FPA in each iteration according to the fitness function. The evolvement of the populations is executed by both FPA and DEA.

Step 3. Update: The global pollination and local pollination of FPA are updated by using Eqs. $(1-3)$ and the agents and food source positions of DEA are updated by using Eqs. (4) and (5). The best fitness value and their positions are memorized.

Step 4. Communication Strategy: The best pollens among all the flowers of FPA's population are copied with *k* the top fitness pollens in N_1, migrate to another place of group in DEA population then replace the weaker agents in N_2, and update for each population in every *R* iteration. In contract, do the same with agents among all the individuals of DEA's population.

Step 5. Termination checking: Go to Step 2 if the predefined value of the function is not achieved or the maximum number of iterations has not been reached, otherwise, ending with minimum of the best value of the functions: $Min(f(S^t), f(X^t))$, and the best bee position among all the agents S^tor the best pollen among all the agents X^t. are recorded.

4 Experimental results

The performance quality of the proposed algorithm of FDA is evaluated by using a set of multimodal benchmark functions [13][14] to test the accuracy and the speed of it. The outcome values of the test functions in the experiments are averaged over 30 runs with different random seeds. All the optimizations for the test functions are to minimize the outcome.

Table 1. The initial range and the total iteration of the benchmark functions

Test Functions	Ranges	Dimensions	Iterations
$F_1(x)=\sum_{i=1}^{n}[x_i^2-10\times\cos(2\pi x_i)+10]$	± 5.12	30	1000
$F_2(x)=\sum_{i=1}^{n}\sin(x_i)\times(\sin(\frac{ix_i^2}{\pi}))^{2m},\ m=10$	$0,\pi$	30	1000
$F_3(x)=\sum_{i=1}^{n}-x_i\times\sin(\sqrt{\lvert x_i\rvert})$	± 5.12	30	1000
$F_4(x)=[e^{-\sum_{i=1}^{n}(x_i/\beta)^{2m}}-2e^{-\sum_{i=1}^{n}x_i^2}]\prod_{i=1}^{n}\cos^2 x_i,\ m=5$	± 20	30	1000
$F_5(x)=-\sum_{i=1}^{4}c_i\times\exp(-\sum_{j=1}^{6}a_{ij}(x_j-p_{ij})^2)$	0,10	4	1000
$F_6(x)=-\sum_{i=1}^{5}[(X-a_i)\times(X-a_i)^T+c_i]^{-1}$	0,10	4	1000

The simulation results of the proposed method are compared with those obtained results of the previous algorithms as the DEA[6], FPA[5] and Genetic algorithm (GA) [15], in terms of their performance of the accuracy and running speed. Let $S = \{s_1, s_2, \dots, s_m\}$, $X = \{x_1, x_2, \dots, x_m\}$, and $G = \{g_1, g_2, \dots, g_m\}$ be the real value vectors of m-dimensional for DEA, FPA and GA respectively. The optimization goal is to minimize the outcome for all benchmarks. The outcome of the performed optimal for the test benchmark function is a minimizing problem. The population size for the methods of FDA, DEA, FPA and GA are set to 40 for all runs in the experiments. The setting parameters for DEA, FPA, and GA could be found in [3,4,10]. Table 1 lists the initial range, the dimension and total iterations for all test functions.

Table 2. The comparison quality performance evaluation of DEA, FPA, and FDA for solving the optimization problems

Test functions	**Function values**			**Comparison performance**	
	DEA[4]	FPA[3]	**FDA**	with DEA	with FPA
1	1.65E+02	1.73E+02	1.29E+02	27%	34%
2	1.45E+00	1.48E+00	1.05E+00	36%	41%
3	-4.87E+03	-4.10E+03	-6.09E+03	19%	33%
4	1.60E-03	1.60E-03	1.10E-03	42%	43%
5	-2.94E+00	-3.04E+00	-3.32E+00	8%	5%
6	-7.15E+00	-8.15E+00	-9.72E+00	25%	16%
Avge.	**-7.86E+02**	**-6.56E+02**	**-9.95E+02**	**26%**	**29%**

The parameters setting for FDA with DEA side is initial with setting Limit to 10. The percentage of the onlooker and employed agents are set to 50%. The total population size N_1 is set to $N/2$ as equal to 20 and the dimension of solution space d is set to 30, as in ref.[4]. Corresponding to the parameters setting with FPA side is the initial probability p is set to 0.55, λ is set to 1.5, the total population size N_2 set to 20 and the dimension d is set to 30, as in ref. [9][3]. Each benchmark function is tested with 1000 iterations per a run. The performance is evaluated in the average of the results from all runs. Comparing percentage is set to abs (FDA-original algorithm) *100/(FDA).

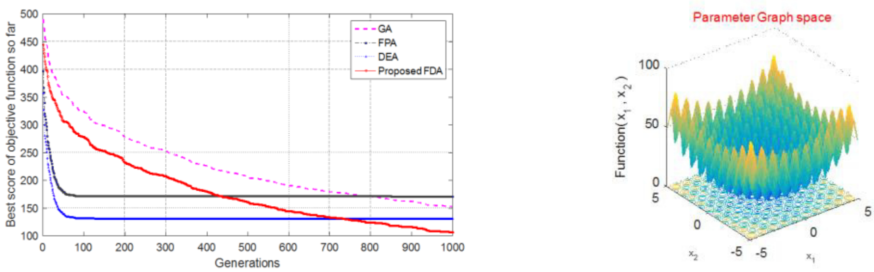

Fig. 1. The experimental results of function F1

Table 2 compares the performance quality for the multimodal optimization problems of three methods of DEA, FPA and the proposed FDA. Observed, the results of the proposed method on all of these cases of testing multimodal benchmark problems show that FDA method almost increases higher than those obtained from original methods of DEA and FPA. The maximum case obtained from FDA method increases higher than those obtained from the DEA and FPA methods are up to 42% and 44% respectively. However, the figure for the minimum cases is only the increase 07% and 06% for DEA and FPA respectively. Thus, in general of the proposed algorithm, FDA obtained the average cases of various tests multimodal optimization problems for the convergence, and accuracy increased more than those obtained from the DEA and FPA methods are 26% and 29% respectively.

Figures 1 and 2 show the experimental results for the first three multimodal benchmark functions over 30 runs output obtained from GA, DEA, FPA and proposed FDA methods with the same iteration of 1000.

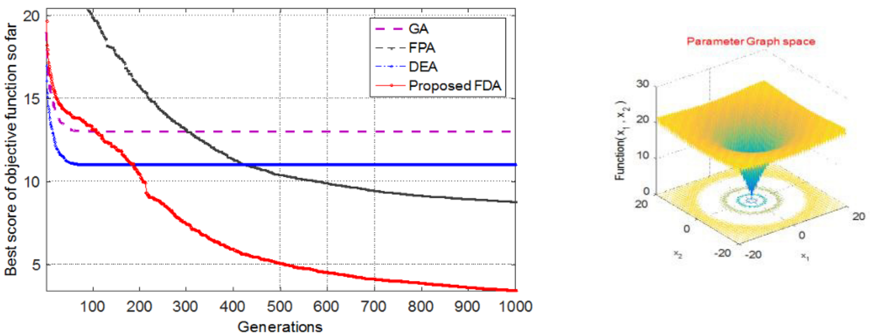

Fig. 2. The experimental results of function F2

The above figures show clearly that, all of the cases of testing functions for FDA have performance quality highest in terms of the accuracy and convergence.

Table 3 shows the performing quality and running time comparison of the proposed FDA with GA method for the multimodal optimization problems. The columns of comparison times and qualities are calculated as absolute of the obtained from FDA minus that obtained from GA then divided the obtained value of the FDA method. Clearly, the results of the proposed method on all of these cases of testing multimodal benchmark problems show that FDA method almost increases higher quality and shorter running time than those obtained from GA method. In general, the proposed algorithm obtained the average cases of various tests multimodal optimization problems for the convergence, and accuracy increased more than those obtained from the GA method is 35%, and for the speed is faster than that got from GA method is 3%.

Table 3. The comparison quality performance of GA and FDA for solving the optimization problems

Test functions	Consumption Time		Comp. times	Performances		Comp. qualities
	GA[10]	FDA		GA[10]	FDA	
1	2.4794	2.4370	2%	1.92E+00	1.26E+00	51%
2	0.896	0.8686	3%	-5.27E+03	-6.19E+03	14%
3	0.9874	0.9501	4%	1.89E+02	1.29E+02	45%
4	0.7775	0.7931	2%	1.710E-03	1.11E-03	39%
5	0.8917	0.9071	2%	-2.23E+00	-3.28E+00	47%
6	0.9892	0.9794	1%	-8.04E+00	-9.71E+00	21%
Avg.	**1.1703**	**1.1543**	3%	**-8.48E+02**	**-1.01E+03**	**35%**

5 Conclusion

This paper, a novel proposed method for the optimization problems was presented with the parallel optimization based on Agents and Pollens communication strategy, namely FDA. The enhanced diversity agents by the parallel process to optimization could take an important significance in the solutions for the issue of losing the global optimum in the optimal algorithms for the multimodal or complex constrained optimization problems. The proposed communication strategy is innovative because of the introduction of strength points of DEA and FPA in the cooperation of optimization algorithms. By this way, the poorer pollens in FPA could be replaced with new best agents from DEA after running the exchanging period. In contrast, the worst agents in DEA could be replaced with fresh finest pollens from FPA in every exchanging period. Compared with original DEA, FPA, and GA, the quality performance of the proposed FDA algorithm shows the better results of the testing set than those obtained from the DEA, FPA, and GA methods in terms of convergence and accuracy. For the maximum cases of testing set increase higher than those obtained from the DEA, FPA, and GA methods are up to 42%, 43%, and 45% respectively. However, these figures for the minimum cases are only the increase 08%, 05%, and 15%. Thus, in general of the proposed algorithm obtained the average cases of various test problems for the convergence, and accuracy increased more than those by 26%, 29%, and 35% from DEA, FPA, and GA respectively.

References

1. Chopard, B., Pictet, O., Tomassini, M.: Parallel and distributed evolutionary computation for financial applications. Parallel Algorithms Appl. 15, 15–36 (2000).
2. Tsai, C.-F., Dao, T.-K., Yang, W.-J., Trong-The, N., Pan, T.-S.: Parallelized Bat Algorithm with a Communication Strategy. In: Ali, M., Pan, J.-S., Chen, S.-M., and Horng, M.-F. (eds.) Modern Advances in Applied Intelligence, Iea/Aie 2014, Pt I. pp. 87 95. Springer International Publishing (2014).

3. Pan, T.-S., Dao, T.-K., Nguyen, T.-T., Chu, S.-C.: Optimal Base Station Locations in Heterogeneous Wireless Sensor Network Based on Hybrid Particle Swarm Optimization with Bat Algorithm. J. Computer, 254. 14 (2015).
4. Dao, T.-K., Pan, T.-S., Nguyen, T.-T., Pan, J.-S.: Parallel bat algorithm for optimizing makespan in job shop scheduling problems. J. Intell. Manuf. (2015).
5. Yang, X.-S.: Flower Pollination Algorithm for Global Optimization. In: Unconventional Computation and Natural Computation. pp. 240–249. Springer (2013).
6. Storn, R., Price, K.: Differential Evolution -- A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. J. Glob. Optim. 11, 341–359 (1997).
7. Gandomi, A.H., Yang, X.S., Talatahari, S., Alavi, A.H.: Metaheuristic Algorithms in Modeling and Optimization. In: Metaheuristic Applications in Structures and Infrastructures. pp. 1–24 (2013).
8. Baghel, M., Agrawal, S., Silakari, S.: Survey of Metaheuristic Algorithms for Combinatorial Optimization. Int. J. Comput. Appl. 58, 975–8887 (2012).
9. Storn, R., Price, K.: Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces. Science (80-.). 11, 1–15 (1995).
10. Holland, J.H.: Adaptation in Natural and Artificial Systems: An introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press (1975).
11. Das, S., Suganthan, P.N.: Differential evolution: A survey of the state-of-the-art. IEEE Trans. Evol. Comput. 15, 4–31 (2011).
12. Chiroma, H., Shuib, N.L.M., Muaz, S.A., Abubakar, A.I., Ila, L.B., Maitama, J.Z.: A Review of the Applications of Bio-inspired Flower Pollination Algorithm. Procedia Comput. Sci. 62, 435–441 (2015).
13. Jamil, M., Yang, X.-S.: A Literature Survey of Benchmark Functions For Global Optimization Problems Citation details: Momin Jamil and Xin-She Yang, A literature survey of benchmark functions for global optimization problems. Int. J. Math. Model. Numer. Optim. 4, 150–194 (2013).
14. Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., Chen, Y., Auger, A., Tiwari, S.: Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization. Nat. Comput. 1–50 (2005).
15. Whitley, D.: A genetic algorithm tutorial. Stat. Comput. 4, 65–85 (1994).