Modified Multi Verse Optimizer for Solving Optimization Problems Using Benchmark Functions

Julakha Jahan Jui
Faculty of Electrical and Electronics Engineering Technology (FTKEE)
Universiti Malaysia Pahang (UMP)
Pekan, Pahang, Malaysia
julakha.ump@gmail.com

Mohd Ashraf Ahmad
Faculty of Electrical and Electronics Engineering Technology (FTKEE)
Universiti Malaysia Pahang (UMP)
Pekan, Pahang, Malaysia
mashraf@ump.edu.my

Muhammad Ikram Mohd Rashid
Faculty of Electrical and Electronics Engineering Technology (FTKEE)
Universiti Malaysia Pahang (UMP)
Pekan, Pahang, Malaysia
mikram@ump.edu.my

Abstract— **The hybrid version of multi-verse optimizer (MVO) namely the modified multi-verse optimizer (mMVO) is developed in this paper by modifying the position updating equation of MVO. Here two modification is proposed in the standard MVO. Firstly, an average position selection mechanism is proposed for solving the local optima problem and secondly, the MVO algorithm is hybrid with another metaheuristics algorithm namely the Sine Cosine Algorithm (SCA) for better balancing the exploration and exploitation of standard MVO algorithm so that it can improve its searching capability. The proposed version of MVO has been evaluated on 23 well known benchmark functions namely unimodal, multimodal and fixed-dimension multimodal benchmark functions and the results are then verified with the standard MVO algorithm. Experimental results demonstrate that the proposed mMVO algorithm gives much better improvement than the standard MVO in the optimization problems in the sense of preventing local optima and increasing the search capability.**

Keywords—optimization, multi-verse optimizer (MVO), modified multi-verse optimizer (mMVO), sine cosine algorithm (SCA).

I. Introduction

Now a days, optimization problems are one of the common terms [1]. These optimization problems are sometimes nonlinear, multimodal or discontinuous with multiple local minima and they are very complicated to solve with traditional approaches of optimization. Therefore, conventional approaches such as gradient-based approach tend to struggle to solve these issues and so scientists have tried various ways to try to solve these challenging problems with varying performance levels. A variety of meta-heuristic algorithms has been developed successfully in the last few years to create better solutions for optimization problems [2,3]. Genetic Algorithm (GA) [4], Differential Evolution (DE) [5], Cuckoo Search (CS) [6], Ant Colony Optimization (ACO) [7], Firefly Algorithm (FA) [8], Particle Swarm Optimization (PSO) [9], Artificial Bee Colony (ABC) [10], and Moth-flame Optimization (MFO) [11] are the most popular metaheuristic algorithms and the advantages of this algorithms are it has a high convergence capability under certain condition and they have a very low probability to trapped into local optima.

A newly proposed metaheuristic algorithm called Multi-verse optimizer (MVO) introduced by Seyedali Mirjalili [12] which is used to evaluate the optimization problems in this paper. As the name implies, it is a physics inspired algorithm. The MVO algorithm inspired by three major concepts of the cosmology called white hole, black hole, and wormhole. MVO utilized the black hole and white hole concepts to explore the search spaces. For exploiting the search spaces the wormholes help MVO algorithm. The MVO algorithm used in 19 difficult test metrics for the first time and also tested its performance by implementing in the practical engineering problems. The experimental findings show that the implemented algorithm can generate very successful results and outperform other metaheuristics algorithms outlined in the literature [12]. Nonetheless, this algorithm still poses some problems possibly get trapped in the local modes.

For improving the capability to escape from local optima, in the MVO algorithm a new position selection technique called average position selection is applied. Also, for improving the search capability, the MVO algorithm is hybridized with Sine Cosine Algorithm (SCA) in this paper. SCA [13] is a wrapper-based metaheuristics which use the sine and cosine functions for mathematical formulations. It cycles from exploration to exploitation by using the sine and cosine functions of the algorithm. It has strong exploration ability than the exploitation ability, when SCA compared with any other population based nature inspired algorithms. By considering this, our proposed method is hybrid with SCA.

This paper is organized as follows: an overview of the standard MVO is illustrated in the next section and then an overview of the proposed mMVO algorithm is discussed in detailed in section 3. The analysis of the experimental results is provided in the fourth section. The conclusions are finally drawn in the last section.

II. Overview of Multi-verse Optimizer

The Big Bang theory and Quantum mechanics inspire MVO. Each universe in the search space act as a feasible solution vector in standard MVO algorithm and the elements in the universe act as a parameter in consequent solution vector. Like the universe principle, every possible solution of the universes must have their own inflation rate that relates to the solution's fitness function of that universe.

There are five rules during the entire optimization:

1. If the inflation rate is higher, there is a high chance of having white holes.
2. If the inflation rate is lower, there is a high chance of having black holes.
3. Higher-inflation universes usually send objects through white holes.

4. Lower-inflation universes tend to receive more objects through black holes.

5. Via wormholes, objects in all universes may face random movement toward the best universe regardless of inflation rate. There is always a high chance of transferring objects from a high-inflation universe to a low-inflation universe.

A sketch of the proposed algorithm is shown in Fig. 1. In the figure each circle with the same color shows the values for the variables of a particular solution in a given population.

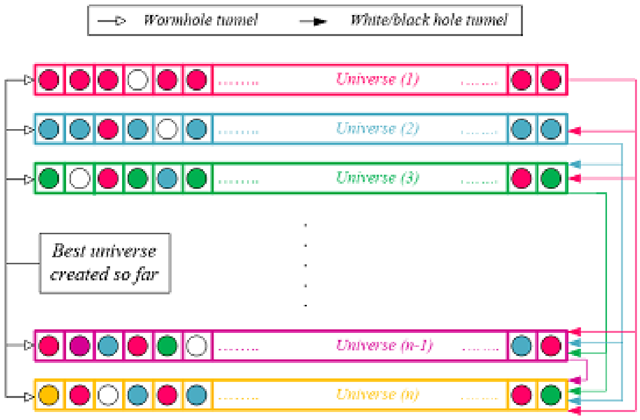

Fig.1: A sketch of MVO [12].

The above mentioned rules can guarantee an increase over variations of the average levels of inflation of the total universes. At each iteration, the universes were sorted according to their inflation rates and choose one of them by the roulette wheel selection mechanism to have a white hole. To do so, the following steps are taken [12]. Suppose that

$$u = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^d \\ x_2^1 & x_2^2 & \cdots & x_1^d \\ \vdots & \vdots & & \vdots \\ x_n^1 & x_n^2 & \cdots & x_n^d \end{bmatrix} \quad (1)$$

where *n* is the number of universes (candidate solutions) and *d* is the number of parameters (variables):

$$x_i^j = \begin{cases} x_k^j & r_1 < NI(U_i) \\ x_i^j & r_1 \geq NI(U_i) \end{cases} \quad (2)$$

where r_1 is a random number in [0, 1], x_i^j indicates the *j*th parameter of *i*th universe, x_k^j indicates the *j*th parameter of *k*th universe selected by a roulette wheel selection mechanism. U_i shows the *i*th universe and $NI(U_i)$ is normalized inflation rate of the *i*th universe, and the position updating mechanism is as follows:

$$x_i^j = \begin{cases} \begin{cases} x_j + \mathrm{TDR} * (ub_j - lb_j) * r_4 + lb_j, & r_3 < 0.5 \\ x_j - \mathrm{TDR} * (ub_j - lb_j) * r_4 + lb_j, & r_3 \geq 0.5 \end{cases} & r_2 < \mathrm{WEP} \\ x_i^j & r_2 \geq \mathrm{WEP} \end{cases} \quad (3)$$

where r_2, r_3, r_4 are random numbers in [0, 1], x_i^j indicates the *j*th parameter of *i*th universe, x_j indicates the *j*th parameter of best universe formed so far, TDR and WEP are the coefficients, lb_j shows the lower bound of *j*th variable and ub_j is the upper bound of *j*th variable. From the above equation it is implied that MVO has two primary coefficients here: wormhole existence probability (WEP) and traveling distance rate (TDR). For both coefficients the adaptive formulation is as follows:

$$\mathrm{WEP} = \min + l * \left(\frac{\max - \min}{L}\right) \quad (4)$$

where min is the minimum value which is 0.2 in the original MVO paper, max is the maximum value which is 1 the original MVO paper, *l* is the current iteration, and *L* is the maximum iterations.

$$\mathrm{TDR} = 1 - \frac{l^{1/p}}{L^{1/p}} \quad (5)$$

where p is the exploitation accuracy over the iterations which is chosen 0.6 in the original MVO paper. When the p value goes higher it is possible to get the earlier and more precise exploitation. The MVO algorithm relies on number of iterations, number of universes, roulette wheel mechanism and mechanism for sorting the universe. In the original paper the Quicksort algorithm is being used to sort the universe after each iteration, and the roulette wheels selection is executed over iterations for each variable in each universe. Details of MVO algorithm can be found in [12].

III. Proposed Modified Multi-verse Optimizer

MVO and SCA, both algorithms have their own advantages and both functions well for a diverse scale of optimization problems. The newly proposed version of MVO called modified multi-verse optimizer (mMVO) based on MVO and SCA by combining the advantages of both algorithms is discussed more details in this section.

In this paper the proposed mMVO algorithm conducts global search with random walks more effectively. When the universes are unable to develop better solutions, they are re-formed with Sine Cosine function of the SCA algorithm so that the best universe searching mechanism is influenced and also it prevented from being lost in local optima.

In short, the work in this research area aims to improve the effectiveness of the previous version of MVO algorithm by enhancing its process for updating the Eq. (3). Basically, Eq. (3) is the primary equation for improving the inflation rate of the standard MVO algorithm and this equation also help to balance the TDR and WEP because this are the main coefficient for balancing the exploitation and exploration.

Also r_4 is an important parameter that effects in updating position of the universe in the exploration phase. The equation is modified in two section. Firstly, in the standard MVO, for ensuring local changes across the universes and possibility of enhancing the inflation rate, it was necessary to assume that wormhole tunnels will be always formed between the current universe and best universe. But for finding the next position of the universe and getting the better inflation rate the MVO algorithm will be improved by considering that the wormhole tunnels will be established in the current universe where the current universe will be formed by taking the average of the previous universe and the best universe which can help the mMVO algorithm to get the better inflation rate of the universe. Secondly, in the modified multi-verse optimizer (mMVO) the universe x_i^j is modified by the position updating equation of SCA algorithm.

The mathematical formula of the proposed mMVO algorithm for updating the position of the universe is as follow:

$$BU = (x_j + x_i^j)/2$$

$$x_i^j = \begin{cases} \begin{cases} BU + \text{TDR} * (\sin(2*\pi*r_5)) * \left|2*r_6*x_j - x_i^j\right| & r_3 < 0.5 \\ BU + \text{TDR} * (\cos(2*\pi*r_5)) * \left|2*r_6*x_j - x_i^j\right| & r_3 \geq 0.5 \end{cases} & r_2 < \text{WEP} \\ x_i^j & r_2 \geq \text{WEP} \end{cases} \quad (6)$$

where r_5 and r_6 are the random numbers in [0,1] and the parameter r_3 is used to randomly switch between sine and cosine functions in equation (6). The flowchart and the Pseudocode of the mMVO algorithm are illustrated in Fig. 2.

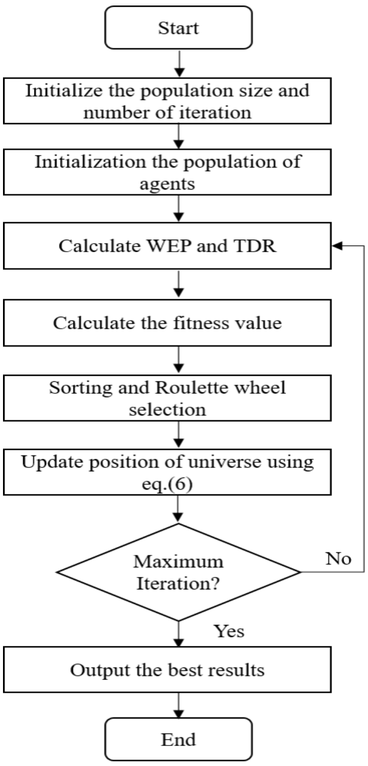

Fig. 2: Flow chart of mMVO algorithm.

IV. Experimental Results and Discussion

This section will compare mMVO using 23 benchmark functions. The effectiveness of the proposed method will be evaluated by using two ranking method, one is tide rank method and another one is Wilcoxon's test method. The performance of the proposed method will be compared with standard MVO.

Both algorithms were runs 30 times with a total 30 universes and 1000 iterations.

A. Testing Functions

In this experiment, a set of 23 different benchmark functions or test functions were used to investigate the effectiveness of our proposed mMVO algorithm. Basically, two group of benchmark functions has been used one is unimodal and other is multimodal and fixed-dimension multimodal function.

1) Unimodal benchmark function (F1~F7), which has only one global optimum and suitable for measuring the local exploitation capability of an algorithm. This type of function allows to focus more on the convergence rates except the final output.

2) Multimodal benchmark function (F8-F13), which has many local minima and number of local optima typically rises exponentially with the dimension of the problem which making them ideal for benchmarking the algorithms global exploration capability and help the algorithm to escape from the local optima.

3) Fixed-dimension multimodal benchmark function (F14~F23), it is also a type of multimodal function but here the functions dimension is fixed.

All the benchmark functions are summarized in Tables I-III where F represents the functions, D represents the dimensions of the functions, R represents the range of variables and F_{min} represents the global optimum value in the variable shale.

TABLE I. Unimodal Benchmark Functions

Function Name	**F**	**D**	**R**	**F_{min}**
Sphere	F1	30	[-100,100]	0
Schwefel 2.22	F2	30	[-10,10]	0
Schwefel 1.2	F3	30	[-100,100]	0
Schwefel 2.21	F4	30	[-100,100]	0
Resenbrock	F5	30	[-30,30]	0
Step	F6	30	[-100,100]	0
Quartic	F7	30	[-1.28,1.28]	0

TABLE II. Multimodal Benchmark Functions

Function Name	**F**	**D**	**R**	**F_{min}**
Schwefel	F8	30	[-500,500]	18.9829×D
Rastrigin	F9	30	[-5.12,5.12]	0
Ackley	F10	30	[-32,32]	0
Griewank	F11	30	[-600,600]	0
Penalized	F12	30	[-50,50]	0
Penalized 2	F13	30	[-50,50]	0

TABLE III. FIXED-DIMENSION MULTIMODAL BENCHMARK FUNCTIONS

Function Name	F	D	R	F_{min}
Foxholes	F14	2	[-65.536,65.536]	0.998004
Kowalik	F15	4	[-5,5]	0.00030
Six-hump Camel Back	F16	2	[-5,5]	-1.0316
Branin	F17	2	[-5,5]	0.398
Goldstein-Price	F18	2	[-2,2]	3
Hartman 3	F19	3	[1,3]	-3.86
Hartman 6	F20	6	[0,1]	-3.32
Shekel 5	F21	4	[0,10]	-10.1532
Shekel 7	F22	4	[0,10]	-10.4028
Shekel 10	F23	4	[0,10]	-10.5363

B. Performance enaluation

Two different measurement methods (mean and standard deviation) are chosen in this section to evaluate the mMVO efficiency over MVO. The mathematical descriptions of Mean and variance are defined as follows,

$$std = \sqrt{\frac{\sum_{i=1}^{P}(X_i - Mean)^2}{P}} \quad (8)$$

$$Mean = \frac{1}{P}\sum_{i=1}^{P} X_i \quad (9)$$

where *P* represents the number of iterations and in this paper *P*=1000. *X* is the function fitness value in each iterations and *i* represents the *i*th iteration.

This paper ranks the optimization performance of mMVO and MVO algorithms by "tied rank". Particularly, the algorithms are ranked according to the mean values. However, this paper also shown the average rank and overall rank of mMVO and MVO algorithms for 23 benchmark functions. In addition, Wilcoxon rank sum test also evaluated with the p-value of 0.05 significant level. Mainly, statistical tests evaluation is necessary to show that mMVO provides significant improvements in optimization over MVO.

C. Comparisn and anlysis of test results

For evaluating the optimization impact of proposed mMVO algorithm over MVO and also evaluating the effectiveness of mMVO for making an appropriate balance between exploration and exploitation, two statistical analysis tests has been performed. Therefore, the optimization performance of the algorithms also evaluated by comparing the statistical results (mean and standard deviation). The statistical analysis results of experiments are tabulated in tables IV-VII. Wilcoxon's rank sum test shown in tables VIII-X which is employed to estimate the statistically significant difference among two algorithms.

a) Comparisn test Results: The test results of 23 benchmark functions of mMVO and MVO are tebulated in tables IV-VII. In the tables the mean value and the standard deviation value of both algorithms is tebulated. Here the lowest mean fitness and the lowest std value is outlined in bold. tables IV-VII ranked the optimization performance of both algorithms by "tied rank". Mainly, the algorithms are ranked as first and second according to the lowest mean values, and the lowest fitness variance of the algorithms. Moreover, Table VII offer the average rank and overall rank of mMVO and MVO for all the 23 benchmark functions. From the findings it is noticeable that mMVO is quite improved than the conventional MVO. In the table column is represented with R define rank of the algorithms.

TABLE IV. RESULTS AND COMPARISON OF MMVO AND MVO FOR UNIMODEL BENCHMARK FUNCTIONS

F	mMVO			MVO		
	Mean	*Std.*	*R*	*Mean*	*Std.*	*R*
F1	**2.63E-25**	**6.49E-25**	1	4.14E-03	1.99E-03	2
F2	**7.06E-15**	**4.99E-15**	1	1.83E-02	6.19E-03	2
F3	**2.06E-16**	**4.04E-16**	1	2.86E-02	1.91E-02	2
F4	**2.48E-07**	**1.90E-07**	1	0.044487	0.014966	2
F5	**10.31243**	**1.06E+01**	1	90.13400	377.1273	2
F6	0.801694	4.28E-01	2	**0.003281**	**0.001876**	1
F7	**0.001033**	**6.69E-04**	1	0.001727	0.000935	2

TABLE V. RESULTS AND COMPARISON OF MMVO AND MVO FOR MULTIMODEL BENCHMARK FUNCTIONS

F	mMVO			MVO		
	Mean	*Std.*	*R*	*Mean*	*Std.*	*R*
F8	-2542.50	**2.79E+02**	2	**-3019.8**	330.718	1
F9	**7.222035**	**2.01E+00**	1	11.1454	5.36360	2
F10	**0.067140**	**3.68E-01**	1	0.25678	0.59509	2
F11	**0.130855**	**7.07E-02**	1	0.30867	0.13373	2
F12	**0.023660**	4.42E-01	2	0.03140	**0.09534**	1
F13	0.041350	2.11E-01	2	**0.00201**	**0.003689**	1

TABLE VI. RESULTS AND COMPARISON OF MMVO AND MVO FOR FIXED-DIMENSION MULTIMODEL BENCHMARK FUNCTIONS

F	mMVO			MVO		
	Mean	*Std.*	*R*	*Mean*	*Std.*	*R*
F14	5.56388	4.46E+00	2	**0.99800**	**6.68E-12**	1
F15	**0.00389**	**7.61E-03**	1	0.00457	0.01149	2
F16	**-1.0316**	**9.56E-10**	1	-1.0316	1.03E-07	2
F17	**0.39788**	5.71E-07	1	0.39789	**1.37E-07**	2
F18	**8.4000**	**1.65E+01**	1	8.40000	20.55036	2
F19	**-3.8637**	2.43E-03	1	-3.8627	**4.09E-07**	2
F20	**-3.2651**	**6.78E-02**	1	-3.2463	0.058520	2
F21	**-7.9699**	**2.01E+00**	1	-7.8792	2.9120853	2
F22	**-9.7224**	**2.00E+00**	1	-9.6195	2.066118	2
F23	**-9.1789**	**2.84E+00**	1	-8.8412	2.935246	2

TABLE VII. AVERAGE AND OVERALL RANK OF MMVO AND MVO FOR ALL BENCHMARK FUNCTIONS

Algorithms	mMVO	MVO
Average Rank	1.0123	1.7826
Overall Rank	1	2

Table V visualize the comparison results for the unimodal functions F1~F7. From the table, it seen that for only one function (F6) the mMVO fails to converge in optimal solution and mMVO results much better than the conventional MVO. This proves that the mMVO has a great impact on the exploitation fitness than the MVO algorithm.

From the table VI it can be seen that for the multimodal functions F8~F13, mMVO provide average results compare to the MVO. The results show that the mMVO can converge to the optimum results of each multimodal benchmark functions (except F8 and F13). This proves that the mMVO has a greater impact on the optimization exploration ability than MVO. It can prevent mMVO algorithm from trapped into local optima.

The table VII shows that, for the fixed-dimension multimodal functions F14~F23, mMVO outperforms the MVO for all functions except the function F14. The experimental results are compatible with several other benchmark functions, whereby the proposed mMVO algorithm produces very competitive results compared with MVO.

b) Wilcoxon's rank sum test results: Wilcoxon's rank test is an helpful test method for estimating the statistical difference between any two or more algorithms. It is mainly employed at the 0.05 significance level. This testing method takes the test results between two different algorithms and perform the Wilcoxon rank test for obtaining the p and h values, so that the obtained values can work as significance level indicator of the corresponding algorithms. Mainly, if the p value is below 0.05 or the h value is equal to 1, then the experimental results of both algorithms are considered to be significantly unsimilar. It may also said that the performance between the two algorithms is significant. Alternatively, the performance of both algorithms is consider to be same if the value of p is greater than 0.05 or the value of h is equal to 0. The Wilcoxon rank sum test results of our proposed mMVO algorithm and the standard MVO algorithm on the benchmark functions F1~F23 has been shown in Table VIII-X.

TABLE VIII. WILCOXONS RANK TEST OF MMVO AND MVO FOR UNIMODEL BENCHMARK FUNCTIONS

Functions	**mMVO Vs MVO**	
	p-value	*h-value*
F1	7.0661e-18	1
F2	7.0661e-18	1
F3	7.0661e-18	1
F4	7.0661e-18	1
F5	0.0091	1
F6	7.0661e-18	1
F7	0.0109	1

TABLE IX. WILCOXONS RANK TEST OF MMVO AND MVO FOR MULTIMODEL BENCHMARK FUNCTIONS

Functions	**mMVO Vs MVO**	
	p-value	*h-value*
F8	0.0011	1
F9	2.9154e-04	1
F10	5.3192e-04	1
F11	5.2721e-05	1
F12	2.0710e-17	1
F13	7.0661e-18	1

TABLE X. WILCOXONS RANK TEST OF MMVO AND MVO FOR MULTIMODEL BENCHMARK FUNCTIONS

Functions	**mMVO Vs MVO**	
	p-value	*h-value*
F14	1.7382e-06	1
F15	2.2165e-06	1
F16	7.0661e-18	1
F17	4.5870e-07	1
F18	1.0129e-17	1
F19	9.3723e-11	1
F20	0.0041	1
F21	0.0020	1
F22	0.0098	1
F23	**0.1217**	**0**

Form the Table VIII-X it can clearly see that, the proposed mMVO algorithm has significant difference compared with the conventional MVO except for function F23.So it can be conclude that the optimization performance of the mMVO is significantly improved over the MVO.

V. CONCLUSION

In this research a modified form of the MVO algorithm has been proposed which is denoted as mMVO to tackle continuous large scale optimization problems. MVO is one of the well-known metaheuristics algorithm for solving many real world problems also but main problem is that, such metaheuristic algorithm can trap into local optima. For this reason, a proper balancing of exploration and exploitation is necessary to overcome this problem. For balancing the two main criteria of any optimization algorithm hybridization can be the best solution. In this study, the main concept for proper balancing the exploration and exploitation precision is hybridizing MVO with another optimization algorithm called SCA. The proposed algorithm is applied in 23 popular benchmark functions and the performance also verified over the conventional MVO. Results from the investigation show that the proposed mMVO algorithm could produce competitive outcome compared with the traditional MVO algorithm. This preliminary report indicate that the proposed algorithm is capable of becoming an efficient tool for finding solutions of any real-world problems optimization.

ACKNOWLEDGMENT

This research was partially funded under research grant RDU1903117 by the Department of Research and Innovation, University of Malaysia Pahang and the Ministry of Higher Education.

REFERENCES

[1] M. Shehab,"A hybrid method based on cuckoo search algorithm for global optimization problems," J. Info. Commu. Tech., vol. 17, pp. 469-491, January 2020.

[2] W. C. Hong, M. W. Li, J. Geng, and Y. Zhang, "Novel chaotic bat algorithm for forecasting complex motion of floating platforms," App. Math. Model., vol. 72, pp. 425-443, August 2019.

[3] S. Gupta, and K. Deep, "A novel random walk grey wolf optimizer," Swa. evolu. compu., vol. 44, pp. 101-112, February 2019.

[4] R. Tanese, "Distributed genetic algorithms for function optimization," 1989.

[5] K. V. Price, "Differential evolution," In Handbook of Optimization, pp. 187-214. Springer, Berlin, Heidelberg, 2013.

[6] A. H. Gandomi, X. S. Yang and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," Engineering with computers, vol. 29, pp. 17-35, January 2013.

[7] M. Dorigo, M. Birattari and T. Stutzle, "Ant colony optimizatio," IEEE comput. intel. maga., vol. 1, pp. 28-39, November 2016.

[8] X. S. Yang, "Firefly algorithm," Natu.-insp. meta. algo., vol. 20, pp. 79-90, 2008.

[9] J. Kennedy and R. Eberhart, "Particle swarm optimization," In Proceedings of ICNN'95-International Conference on Neural Networks, Vol. 4, pp. 1942-1948, November 1995.

[10] D. Karaboga, "Artificial bee colony algorithm," scholarpedia, vol. 5, pp.6915, March 2010.

[11] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," Know.-base. sys., vol. 89, pp. 228-249, November 2015.

[12] S. Mirjalili, S. M. Mirjalili and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," Neur. Compu. Appli., vol. 27, pp. 495-513, February 2016.

[13] S.Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," Know.-bas. sys., vol. 96, pp. 120-133, March 2016.