Optimal Preventive Maintenance Scheduling of Multi State System. A Comparative Study of Different Meta-heuristic Algorithms

Kamel BELKACEM(1)*, Noureddine BALI(1), Hilal LABDELAOUI(1)
(1) Electrical and Industrial Systems Laboratory (LSEI), Faculty of Electrical Engineering University of Sciences and Technology HouariBoumediene (USTHB) Algiers, Algeria
*kbelkacem1@usthb.dz, Kamelbelkacem2019@gmail.com

Abstract: This work presents a comparative study of different meta-heuristic algorithms which are commonly used in the literature, such as Genetic Algorithms, Particle Swarm Optimization, Artificial Bee Colony and Grey Wolf Optimizer. These algorithms are applied to solve a multi-state system maintenance optimization problem considering time and system availability constraints. The objective is to find the optimal inspection and maintenance intervals for each component of the system in order to minimize the preventive maintenance cost of overall the system. The performances of the algorithms are compared using different metrics, regarding the quality and stability of the results and the speed of convergence, in order to determine the most efficient algorithms.

Keywords: preventive maintenance, multi-state system, meta-heuristic, optimization.

1. INTRODUCTION

Industrial systems often degrade through use and exposure to environmental factors. This degradation can eventually lead to system failure, resulting in safety problems and equipment damage [1], which makes the improvement and optimization of maintenance actions a necessity. In real-life situations, the system and its components can present a wide range of performances, from perfect operation to total failure [2].

A significant effort has been made by researchers to solve optimization problems related to the maintenance of multi-state systems. In the literature, we can distinguish two types of methods, the exact methods which guarantee optimality in a finite time and the indirect methods which can provide approximate solutions in less time. The complexity of the systems and the information available in the field of maintenance make exact methods inappropriate for this type of problems and can be very time-consuming. However, indirect methods, such as meta-heuristics, remain powerful for large combinatorial problems that require efficient exploration of the search space. The most recent developments tend to use meta-heuristic algorithms, which are increasingly applied for solving difficult optimization problems [3].

In this paper, a comparative study of standard meta-heuristic algorithms, commonly used in the field of maintenance optimization, is presented in order to study their efficiency and robustness using different criteria. These algorithms are applied to a series-parallel multi-state system with binary components to solve a preventive maintenance problem. The objective is to determine for each component of the system the optimal maintenance period that minimizes a cost function under availability constraint in a specified time horizon.

2. DESCRIPTION OF THE SYSTEM

In this work, we consider a series-parallel multi-state system with non-identical binary components. Fig. 1 shows the structure of a series-parallel multi-state system, consisting of k subsystems in series and each subsystem is composed of multiple components used in parallel. The general assumptions of the model are given as follows [4]:

1) All components are immediately and perfectly repaired after a failure.
2) The failure time of each component for a fixed load occurs according to an exponential distribution (constant rate).
3) The cost of inspection and maintenance remains the same for each component throughout the mission.
4) The maintenance cost is a function of the inspection cost for each component.
5) The inspection cost of the component is constant for the duration of the mission.

6) It is possible to evaluate the distribution of the component performance and its maintenance cost once the maintenance period has been set.
7) The time during which a component is unavailable due to preventive maintenance activities is negligible.
8) Each component is characterised by its failure rate λ(t) and a preventive maintenance cost of one inspection.

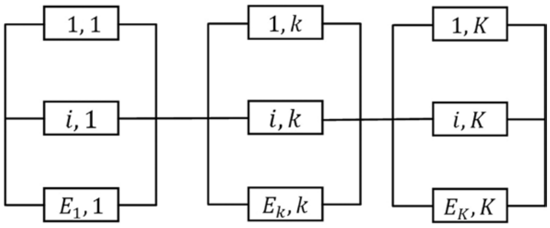

Fig. 1. Structure of a series-parallel system

Assuming that the preventive maintenance cost of the system is a function of the component inspection costs, the cost can be calculated using the following equation:

$$C_{pm} = \sum_{k=1}^{k}\sum_{i=1}^{E_k}\sum_{l=1}^{\eta_{e(i,k)}} c_l(e(i,k)) \quad (1)$$

The total number of inspections is given by:

$$N = \sum_{K=1}^{K}\sum_{i=1}^{E_K} e(i,k)$$

The cost of inspecting a component is constant over the duration of the mission, so we can write:

$$C_{PM}(e(i,k)) = \sum_{l=1}^{\eta_{e(i,k)}} c_l = e(i,k)) = \eta_{e(i,K)} * c(e(i,k))$$

with

$$\eta_{e(i,k)} = 1 + \left[\frac{T_M(e(i,k)) - T_0(e(i,k))}{T_p(e(i,k))}\right]$$

Where, K is the number of subsystems in series, E_k is the number of components in parallel in the *k*th subsystem in series, $\eta_{e(i,k)}$ is the total number of inspections of the *i*th component in the *k*th subsystem in series (parallel block) during the mission, $c_l(e(i,k))$ is the cost of the *l*th inspection of the ith component in the *k*th subsystem in series. C(e (i,k)) is the cost of inspecting the *i*th component in the *k*th subsystem in series, T_p is the inspection period, T_0 is the inspection start time and T_m is the mission duration.

3. PROBLEM FORMULATION

The problem considered in this work is to find the optimal inspection and maintenance periods T_p that minimize the maintenance cost, i.e. the objective function, for each component of the system, taking into account the time and availability constraints. At the beginning of the mission, the components are often reliable, hence it is unnecessary to plan costly maintenance actions. To achieve this task, we need to determine a vector of the first T_0 inspections when the components are less reliable. Therefore, we have to find the availability of each component as well as the availability of the entire system using the Universal Generating Function (UGF) method. Then, using the Birnbaum importance factor (IFB), we can calculate the ideal times to start inspections. The optimization problem is formulated as follows:

$$\begin{cases} C_{PM} \rightarrow Min \\ A(t) \geq A_0 \\ t \leq T_M \end{cases}$$

where A(t) is the system availability.

Consequently, the objective function can be formulated as follows:

$$F = \begin{cases} C_{PM}, & \text{if } A \geq A_0 \\ C_{PM} + W(A_0 - A) & \text{else} \end{cases} \quad (2)$$

System availability

In this work, the method used to find the performance distribution of the multi-state system is the UGF method. The UGF method is based on simple algebraic procedures to find the system performance distribution. This method is shown to be better than other methods such as the Monte-Carlo simulation and the Markov process which are very time-consuming when dealing with a complicated multi-state system [5]. In a multi-state system, each element j can have Kj different states which correspond to performances represented by gj={gj0, gj1....gk-1}, where gji is the performance of element j in state i, i {0,1, ...,kj-1}. The probability associated with the different states of element j can be represented by Pj= {pj0, pj1, ..., pjk-1} [6].

The performance distribution can be calculated using the following equation:

$$U_j(z) = \sum_{l=1}^{n_{gj}} P_{jl}^{G_{jl}} \quad (3)$$

Where G_j is the performance of element j in state i, P_j is the probability associated with the different states of element j with $G_j \in g_j$.

For binary components, the performance distribution can be calculated using the following equation:

$$(1 - A_j)z^0 + A_j z^{G_j} \quad (4)$$

In our case, we assume that the failure distribution of element j follows an exponential distribution, and the availability of a component will be calculated using:

$$A_j(t) = \exp\left(\frac{-T_p(j)}{MTTF(j)}\right) \quad (5)$$

where MTTF is the mean time to failure.

Optimal inspection start time T_0

The inspection start time vector T_0 must be calculated efficiently, taking into account the cost and reliability of the components, where the availability criterion must be minimized. To do this, the IFB_j index must be minimized. The details of Birnbaum importance factor can be found in [7]. In this work, we consider a vector of 20 maintenance times, and for each time we evaluate the importance factor for each component j using the following equation:

$$R_j(t) = \frac{c(j)}{IFB_j(t)}, j = 1,2...., N \quad (6)$$

where R_j is the componentreliability.More details can be found in [5].

4. RESULTS AND DISCUSSION

System under study

Several metaheuristic algorithms are applied to solve the optimization problem described by equation (2), for a power plant multistage coal feeding system, consisting of nine conveyors, as depicted in fig. 2. The data of this system, i.e. the cost C(e(i,k)) and the failure rate λ(t) are taken from [4]. The mission time is Tm= 50 years and the vector of the first inspections is T_0=[12 13 13 9 9 15 15 14 14].

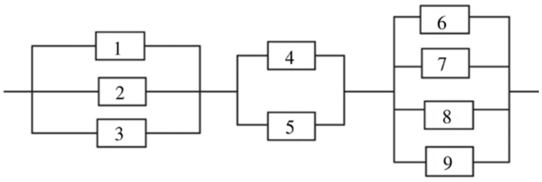

Fig. 2. Block diagram of the power plant feeding system

TABLE 1 characteristics of the system components

	λ(y^{-1})	C(e(i,k))
1	0.0692	6.92
2	0.1005	8.04
3	0.1229	9.83
4	0.0383	7.66
5	0.0383	7.66
6	0.1203	9.63
7	0.1203	9.63
8	0.0929	11.15
9	0.0929	11.15

The algorithms used to solve the aforementioned optimization problem are: Genetic Algorithms (GA) introduced by Holland in 1975 [8], Simulated Annealing (SA) introduced by Kirkpatric in 1983 [9], Particle Swarm Optimization (PSO) introduced by Kennedy in 1995 [10], Differential Evolution (DE) introduced by Storn in 1997 [11], Harmony Search (HS) introduced by Geem in 2001 [12], Artificial Bee Colony (ABC) introduced by Karaboga in 2005 [13], Cuckoo Search (CS) introduced by Yang in 2009 [14], Teaching Learning-Based Optimization (TLBO) introduced by Rao in 2011 [15], Grey Wolf Optimizer (GWO) introduced by Mirjalili in 2014 [16], Stochastic Fractal Search (SFS) introduced by Salimi in 2015[17], Jaya Algorithm (JA) introduced by Rao in 2016 [18], Coyote Optimization Algorithm (COA) introduced by Pierezan in 2018 [19], Political Optimizer (PO) introduced by Askari in 2020 [20], Social Network Search (SNS) introduced by Bayzidi in 2021 [21].

The chosen algorithms are well-known in the field of optimization, and are widely applied to solve difficult and real-world problems

Results

In this work, we examine the performances of each algorithm, in particular the quality and stability of the final and intermediate results, as well as the convergence speed and sensitivity to variations in the parameters A_0 and W. The selected algorithms are implemented with their standard version as proposed in the original papers. Furthermore, the parameters of each algorithm are tuned in order to provide the best results at the end of the optimization process. Additionally, for a fair comparison, each algorithm is tested 30 times and the number of evaluations of the objective function are fixed to 10^5 at each test.

Some of the applied algorithms, such as JA and GWO, do not involve any control parameters other than the population size, which makes them easier to implement. However, their performances are limited and depend strongly on this parameter.

The applied algorithms are compared with regard to the quality of the results, represented by the following criteria:

Mean(F): the average of the final objective function values considering only the feasible solutions,

Sdv(F): the standard deviation of the final objective function values considering only the feasible solutions,

Mean(A): the average of the intermediate objective function values during each step of the optimization process. In this work, we consider a step of 10^3 evaluations.

Feas(%): the feasibility rate of the final solutions with regard to the availability constraint A_0.

Five cases are considered for study, regarding the selected values of A_0 and W. The results are presented in the following tables and figures.

Case 1: A_0=0.9, W=10^4

TABLE 2optimization results obtained for A_0=0.9

Method	Mean(F)	Sdv(F)	Mean(A)	Feas(%)
GA	613.97	1.61	636.17	100.0
SA	618.21	4.15	632.44	93.3
PSO	623.31	5.63	671.92	96.7
DE	611.72	1.16	617.60	96.7
HS	613.23	1.21	623.33	86.7
ABC	615.50	2.13	629.66	93.3
CS	618.72	2.39	639.68	96.7
TLBO	623.13	6.94	671.78	80.0
GWO	624.04	3.81	645.89	90.0
SFS	612.85	1.16	621.67	100.0
JA	634.74	7.86	692.03	76.7
COA	616.50	2.67	655.56	100.0
PO	615.47	1.59	626.06	100.0
SNS	616.87	1.80	643.88	100.0

The best solutions are obtained using DE, SFS and HS:

DE:

T_p=[2.926 4.117 7.407 4.560 5.141 5.014 5.013 6.010 6.042], C_{PM}= 610

SFS:

T_p=[3.177 3.713 7.427 4.574 5.137 5.033 5.034 6.037 6.007], C_{PM}= 611

HS:

T_p=[3.169 3.717 7.460 5.132 4.575 5.001 5.009 6.061 6.027], C_{PM}= 611

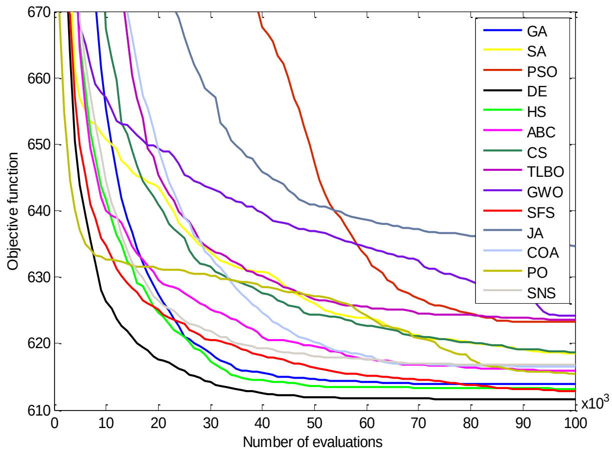

Fig 3 Variations of the objective function forA_0=0.9

Case 2: A_0=0.8, W=10^4

TABLE 3Optimization results obtained for A_0=0.8

Method	Mean(F)	Sdv(F)	Mean(A)	Feas(%)
GA	450.83	1.95	462.76	100.0
SA	456.29	4.97	466.59	93.3
PSO	460.33	5.89	493.16	100.0
DE	449.00	2.45	455.18	100.0
HS	449.28	1.07	456.91	96.7
ABC	452.79	2.76	463.00	96.7

CS	456.57	3.23	472.06	100.0
TLBO	459.00	3.14	483.23	93.3
GWO	461.61	4.12	475.25	93.3
SFS	462.40	3.30	467.31	66.7
JA	464.67	4.03	493.63	90.0
COA	455.47	4.38	475.13	100.0
PO	454.27	2.39	462.86	100.0
SNS	454.93	2.16	467.63	100.0

The best solutions are obtained using DE, HS and GA:

DE :

T_p=[4.236 5.298 9.289 6.874 6.849 7.011 7.010 7.205 9.019], C_{PM}= 446

HS :

T_p=[3.473 4.687 37.143 6.876 7.031 5.880 7.096 7.221 9.011], C_{PM}= 448

GA:

T_p=[3.457 4.626 49.123 6.859 7.074 5.851 7.056 9.017 7.225], C_{PM}= 448

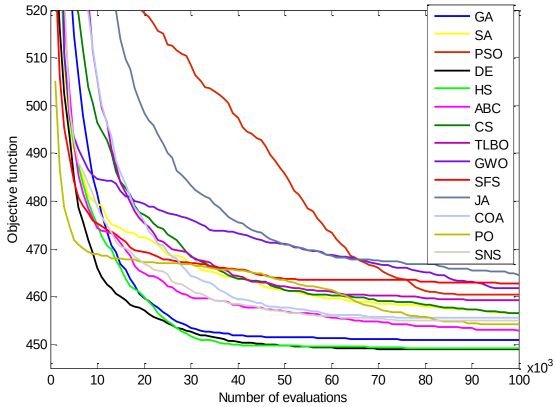

Fig 4 Variations of the objective function for A_0=0.8

Case 3: A_0=0.7, W=10^4

TABLE 4Optimization results obtained for A_0=0.7

Method	Mean(F)	Sdv(F)	Mean(A)	Feas(%)
GA	365.13	1.36	372.92	100.0
SA	367.53	2.40	376.86	100.0
PSO	370.93	3.58	395.50	100.0
DE	364.13	1.72	369.22	100.0
HS	365.00	1.31	370.32	96.7
ABC	365.60	1.73	373.69	100.0
CS	368.83	3.06	380.09	100.0
TLBO	372.46	3.24	384.89	93.3
GWO	373.83	3.76	383.43	100.0
SFS	375.57	3.12	380.55	70.0
JA	376.46	4.00	395.36	93.3
COA	367.37	2.48	379.28	100.0
PO	365.73	2.00	375.00	100.0
SNS	367.31	2.85	377.67	96.7

The best solutions are obtained using DE, HS and GA:

DE :

T_p=[4.296 6.226 45.394 8.206 8.267 7.018 8.777 12.048 9.055], C_{PM}= 362

HS :

T_p=[4.233 6.180 38.185 8.237 8.302 8.836 7.013 9.024 12.023], C_{PM}= 362

GA :

T_p=[3.808 6.172 44.677 10.262 8.334 8.865 8.820 9.113 9.105], C_{PM}= 362

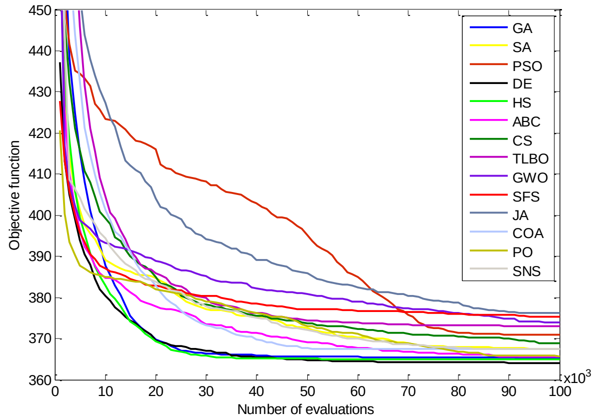

Fig. 5. Variations of the objective function for A_0=0.7

Case 4:A_0=0.9, W=7500

TABLE 5Optimization results obtained forA_0=0.9 and w=7500

Method	Mean(F)	Sdv(F)	Mean(A)	Feas(%)
GA	612.68	1.16	632.93	93.3
SA	618.44	4.73	631.24	83.3
PSO	623.38	5.77	666.03	96.7
DE	611.52	1.60	616.64	96.7
HS	612.88	1.84	622.87	86.7
ABC	615.39	3.47	626.74	93.3

CS	617.36	3.86	635.51	83.3
TLBO	624.82	9.56	667.11	73.3
GWO	623.10	2.62	642.56	70.0
SFS	612.22	1.31	620.56	90.0
JA	634.06	7.31	685.44	53.3
COA	616.07	3.27	650.43	100.0
PO	614.97	1.97	625.41	96.7
SNS	616.33	2.25	638.09	100.0

Case 5: A_0=0.9, W=5000

TABLE 6Optimization results obtained forA_0=0.9 and w=5000

Method	Mean(F)	Sdv(F)	Mean(A)	Feas(%)
GA	612.00	1.62	628.59	90.0
SA	618.42	6.38	629.98	80.0
PSO	625.08	11.40	663.63	86.7
DE	610.24	1.26	614.53	70.0
HS	612.53	1.07	620.32	63.3
ABC	614.19	2.62	623.49	86.7
CS	615.28	2.49	630.97	83.3
TLBO	619.04	5.04	653.42	76.7
GWO	622.94	3.47	641.47	56.7
SFS	611.67	1.40	618.92	80.0
JA	630.33	6.18	665.78	50.0
COA	614.28	2.43	643.95	96.7
PO	614.24	1.62	624.50	96.7
SNS	615.40	1.33	632.52	100.0

Discussions

For case 1, according to table II, the algorithms that achieve the best results according to Mean(F) and Sdv(F) are DE,SFS, HS and GA. Also, it is clear that DE has the best average performance with regard to Mean(A) followed by SFS and HS. This result is verified in fig. 3, where it can be seen that these algorithms converge rapidly during the optimization process, and DE remains the most stable and fastest algorithm. The feasibility rates indicate that most of the obtained solutions are feasible and the value of $W=10^4$ seems to be a good choice.

The algorithms SA, PO, ABC, SNS, COA, CS and TLBO are shown to be less efficient than the first algorithms. Yet, the algorithm ABC is the most stable and the fastest amongst these algorithms whereas TLBO is the slowest. The algorithms that give the worst results are JA, PSO and GWO, although these ones are widely used and very popular in the literature.

In case 2, the search space is wider which means that it is easier to find feasible solutions and the convergence speed can be higher than before. This fact is verified with the values of Feas(%) in table III and the curves in fig. 4. Table III shows also that algorithms DE, HS and GA achieve the best results, especially DE which seems to be is the most efficient and the fastest algorithm between the three. The other algorithms are shown to be less efficient with low convergence rate. The algorithms that give the worst results are again JA, PSO and GWO.

In case 3, DE provide the best solutions followed by HS, GA and ABC, while the HS algorithm is the most stable regarding the standard deviation value of the objective function. The worst results are obtained by algorithms JA, PSO, GWO and SFS.

Regarding the sensitivity to variations in W, the results given in tables V and VI show that, for all algorithms, the objective function values are improved whereas the feasibility rate decrease. This result means that these two criteria cannot be improved simultaneously and a compromise value of W should be chosen. On the other hand, it is shown that the rate of feasible solutions is decreased for GA, whereas it is stable for DE and HS. This result indicates that DE and HS are more robust than GA considering variations in W. Furthermore, the solutions provided by COA, PO and SNS algorithms are the most feasible although their provided objective values are not the best. It is also shown that JA is the worst algorithm along with PSO.

5. CONCLUSION

In this work, we carry out a comparative study of several well-known meta-heuristic optimization algorithms in order to study their performances. These algorithms are used to solve a single-objective maintenance optimization problem for a multi-state series-parallel system. The objective is to find the optimal inspection intervals that minimise a cost function constrained by mission duration and system availability. The results show that the most efficient algorithm for solving this problem is DE, followed by HS and GA, whereas PSO which is a well-known and widely used metaheuristic give the worst results. Therefore, one can conclude that the performance of an optimization method depends strongly on the nature and complexity of the problem under consideration.

References

[1] B. de Jonge and P. A. Scarf, "A review on maintenance optimization," Eur. J. Oper. Res., vol. 285, no. 316, pp. 805–824, 2020.

[2] B. Natvig, Multistate Systems Reliability Theory with Applications. Hoboken, NJ, USA: Wiley, 2011.

[3] X.-S. Yang, Engineering Optimization: An Introduction With Metaheuristic Applications. Hoboken, NJ, USA: Wiley, 2010.

[4] I. Maatouk, I. Jarkass, E. Chätelet, and N. Chebbo, "Preventive maintenance optimization and comparison of genetic algorithm models in a Series–Parallel multi-state system," J. Intell. Syst., vol. 28, no. 2, pp. 219–230, Apr. 2019.

[5] I. Maatouk, N. Chebbo and E. Châtelet, Preventive maintenance optimization for multi states series-parallel systems, *PCO Global Conference*, Prague, Czech Republic, Vol. 2008 (2013), 220–226.

[6] Levitin G. The universal generating function in reliability analysis and optimization. Berlin, Germany: Springer-Verlag; 2005.

[7] A. Lisnianski, I. Frenkel, Y. Ding, A. Lisnianski, I. Frenkel, and Y. Ding, Multi-State System Reliability Analysis and Optimization for Engineers and Industrial Managers. London, U.K.: Springer, 2010.

[8] J. H. Holland, ''Genetic algorithms,'' Sci. Amer., vol. 267, no. 1, pp. 66–73, 1992.

[9] Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680 .

[10] J. Kennedy and R. Eberhart, " Particle swarm optimization," in Proc. ICNN'95–Int. Conf. Neural Networks, Perth, Australia, 1995, pp. 1942−1948.

[11] R. Storn and K. Price, "Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces," ICSI, Berkeley, CA, USA, Rep. TR-95-012, Mar. 1995.

[12] Geem ZW, Kim JH, Loganathan GV (2001) A new heuristic optimization algorithm: harmony search. Simulation 76(2):60–68.

[13] Karaboga D, An idea based on honey bee swarm for numerical optimization. Report, technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[14] X.-S. Yang and S. Deb, "Cuckoo search via levy flights," in Proc. World Congr. Nature Biologically Inspired Comput., 2009, pp. 210–214.

[15] R. V. Rao, V. J. Savsani, and D. P. Vakharia, ''Teaching–learning-based optimization: An optimization method for continuous nonlinear large scale problems,'' Inf. Sci., vol. 183, no. 1, pp. 1–15, Jan. 2012.

[16] S. Mirjalili, S. M. Mirjalili, and A. Lewis, ''Grey wolf optimizer,'' Adv. Eng. Softw., vol. 69, pp. 46–61, Mar. 2014.

[17] H. Salimi, ''Stochastic fractal search: A powerful metaheuristic algorithm,'' Knowl.-Based Syst., vol. 75, pp. 1–18, Oct. 2015.

[18] R. V. Rao, ''Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems,'' Int. J. Ind. Eng. Computations, vol. 7, no. 1, pp. 19–34, 2016.

[19] J. Pierezan and L. Dos Santos Coelho, ''Coyote optimization algorithm: A new metaheuristic for global optimization problems,'' in Proc. IEEE Congr. Evol. Comput. (CEC), Jul. 2018, pp. 1–8.

[20] Q. Askari, I. Younas, and M. Saeed, ''Political optimizer: A novel socioinspired meta-heuristic for global optimization,'' Knowl.-Based Syst., vol. 195, May 2020, Art. no. 105709.

[21] H. Bayzidi, S. Talatahari, M. Saraee, and C.-P. Lamarche, ''Social network search for solving engineering optimization problems,'' Comput. Intell. Neurosci., vol. 2021, pp. 1–32, Sep. 2021.