

# Enhanced quadratic approximation integrated with butterfly optimization: a new search algorithm tested on structural and mathematical problems

**Ali Mortazavi** (Main and Corresponding Author)

Civil Engineering Department, Izmir Democracy University, Izmir, Turkey  
35100 (Turkey)

Email: ali.mortazavi.phd@gmail.com

ORCID: <https://orcid.org/0000-0002-6089-7046>

**Soner Şeker**

Civil Engineering Department, Uşak University, Uşak, Turkey  
64000 (Turkey)

Email: soner.seker@usak.edu.tr

ORCID: <https://orcid.org/0000-0002-7632-9713>

**Manuscript Code:** 14814

**Date of Acceptance/Reception:** 02.03.2021/12.05.2020

**DOI:** 10.7764/RDLC.20.1.15

## Abstract

The Butterfly Optimization Algorithm (BOA) is a swarm-based technique, inspired by mating and food searching process of butterflies, developed last year. Experiments indicate that the BOA provides substantial exploration capability on conventional unconstrained benchmark problems, however for the cases with more complex and noisy domains the algorithm can easily be trapped into local minima due to its restricted exploitation behavior. To tackle this issue, the current study deals with introducing an alternative search strategy to explore the region of the search domain with high certainty. Such that, firstly a weighted agent is defined and then a quadratic search is performed in the vicinity of this pre-defined agent. This alternative search strategy is named Enhanced Quadratic Approximation (EQA) and it is combined with the BOA method to improve its exploitation behavior and provide an efficient search algorithm. Thus, obtained new method is named as Enhanced Quadratic Approximation Integrated with Butterfly Optimization (EQB) algorithm. Different properties of the proposed EQB are tested on mathematical and structural benchmark problems. Acquired results show that the introduced algorithm, in comparison with its parent method and some other well-established reported algorithms in the literature, provides a competitive performance in terms of stability, accuracy and convergence rate.

**Keywords:** quadratic approximation, butterfly optimization algorithm, hybrid methods.

## Introduction

The optimization process is the scheme to select the most proper alternative among the available options. Although there are different optimization techniques available, generally, they can be categorized into two main groups as deterministic and non-deterministic approaches (Degertekin, 2012; Nobile et al., 2018; Pence, Cesmeli, Senel, & Cetisli, 2016). Deterministic approaches are gradient-based methods which apply the gradient information of the optimization objective function to determine their search pattern. These methods are fast and accurate, but they have two main drawbacks. First, they require a continuous (or partially continuous) objective function and its gradient(s) information to determine the step size and search direction of the optimization process (Mortazavi, 2021a). Defining such an objective function for many engineering problems is very difficult or even impossible. Second, these methods are highly sensitive to the starting point of the process and for certain step sizes, they can easily be trapped into the closest local optima (Finotto, da Silva, Valášek, & Štemberk, 2013; Hasançebi, Çarbaş, Doğan, Erdal, & Saka, 2010). Consequently, to solve the complex optimization problems an alternative method seems to be required. In this regard, non-deterministic approaches can be utilized to search the complicated domains since they do not demand any continuous objective function and/or its gradient(s) information.

Metaheuristic algorithms constitute the important category of the non-deterministic methods. These algorithms mostly provide simple but efficient mathematical searching strategies which are inspired by natural phenomena, physical rules or social behaviors (Shi & Eberhart, 1998). One can chronologically sort some of these methods as Particle Swarm optimization (PSO), Ant Colony Optimization (ACO) (Dorigo & Blum, 2005), Gravitational Search Algorithm (GSA) Symbiotic Organisms Search (SOS) (Cheng & Prayogo, 2014), Search Group Algorithm (SGA) (Gonçalves, Lopez, & Miguel, 2015), Heat Transfer Search (HTS) (Patel & Savsani, 2015), Water Wave Optimization (WWO) (Zheng, 2015), Virus Optimization Algorithm (VOA) (Liang & Cuevas Juarez, 2016), Quantum inspired Social Evolution (QSE) (Pavithr & Gursaran, 2016), Interactive Fuzzy Search

Algorithm (IFSA) (Mortazavi, 2019b). In the last two decades along with developments in the computer technologies the application of these methods on solving different type of optimization problems are highly risen (Dede & Ayvaz, 2015; Ding, Huang, & Lu, 2016; Moloodpoor, Mortazavi, & Ozbalta, 2019; Montes, Mortazavi, 2020, 2021b; Mortazavi & Togan, 2017b, 2017a; Mortazavi, Toğan, Daloğlu, & Nuhoglu, 2018; Mortazavi, Togan, & Moloodpoor, 2019; Mortazavi, Toğan, & Nuhoglu, 2018; Mortazavi, Toğan, & Nuhoğlu, 2017; Quagliaroli, Malerba, Albertin, & Pollini, 2015; Souza, Fadel Miguel, Lopez, Miguel, & Torii, 2016; Tang, Li, Luo, & Liu, 2015).

The butterfly optimization algorithm (BOA) is a recently developed metaheuristic approach (Arora & Singh, 2019). This method is mathematically inspired by the butterflies mating and food-searching behaviors. The BOA uses two different paradigms for navigating the agents (i.e. butterflies): tracing of the best agent of the colony for global search and pairwise interacting of agents for local search. Current authors test the performance of the BOA on various constrained and unconstrained problems (Mortazavi, 2019a, 2019c) and the results indicate that the BOA works considerable well on solving the unconstrained cases with smooth domains. However, for the problems with complex search domain (i.e. shifted and rotated) or for those in which domains are bounded with different constraints (e.g. structural optimization problems) performance of the BOA is considerably declined. Indeed, the BOA shows considerable good performance when problem has a distinct global optimum, but when the number of local optima is raised the search capability of this method significantly drops. Considering that several engineering problems have constrained and complex search spaces, in order to use the BOA as a general optimizer tool, the exploitation behavior of this method required to be fundamentally fortified.

To address this problem, in the current study, initially, the quadratic approximation search strategy is adjusted employing the concept of weighted agent. Weighted agent is defined as the weighted average of all population. Indeed, it applies each agent's information based on its degree of quality (determined by its objective function value). So, making a quadratic search in the vicinity of this agent can provide an efficient exploitation search behavior for the algorithm. Presented strategy is named as enhanced quadratic approximation (EQA) and it is combined with affirmative exploration ability of the BOA to provide an effective search method.

For the sake on enhanced quadratic approximation (EQA) strategy, the BOA rather than using fairly fully random search, applies more reasonable neighborhood search around promising area of the domain. Since the new hybrid method combines the affirmative search behaviors of BOA and EQA methods, it is named as Enhanced Quadratic Approximation Integrated with butterfly optimization (EQB) technique. The search ability of the proposed EQB method is tested on several unconstrained and constrained problems. The acquired results are presented through illustrative tables as diagrams. Also, to determine the best balance between exploration and exploitation behaviors a comprehensive sensitivity analyses are performed over regulator parameter.

The rest of this study is arranged as follows: in the next section definitions of the BOA, EQA and proposed EQB methods are given. Section 3, is devoted to providing the sensitivity analysis over balancing parameter of EQB approach and comparatively assessing the accuracy, diversity, convergence rate and complexity of proposed algorithm. In section 4, the search capability of EQB is tested on four real world structural problems and results are compare with some other well-established techniques. Consequently, in last section the corresponding conclusion is given.

## Methodology

### Formulation of optimization methods

In the current section for more clarity initially butterfly optimization algorithm (BOA) and Enhanced Quadratic Approximation (EQA) are described. Then the proposed Enhanced Quadratic Approximation Integrated with Butterfly Optimization (EQB) method and its related terms are explained.

#### Butterfly optimization algorithm (BOA)

In BOA method each agent (i.e. butterfly) applies two different patterns to search the problem's domain. For global search, agents track the best agent of the colony and for local search, they trace the path determined by pairwise interacting between random agents. This formulation is defined as below:

$$\begin{aligned}
& \text{if } r_1 \leq p \\
{}^{t+1}X_i &= {}^tX_i + (r_2^2 \cdot X^G - {}^tX_i) \cdot f_i \\
& \text{if } r_1 > p \\
{}^{t+1}X_i &= {}^tX_i + (r_2^2 \cdot {}^tX_j - {}^tX_k) \cdot f_i \\
& \text{where} \\
f_i &= C \cdot f^a
\end{aligned} \tag{1}$$

In this formulation  $r_1$  and  $r_2$  are two random numbers uniformly selected from [0,1] interval (Arora & Singh, 2019) (Arora & Singh, 2019). Superscripts  $t$  and  $t+1$  show the current and updated values of the corresponding variable. Also,  $X^G$  is the best agent of the colony and  ${}^tX_j$  and  ${}^tX_k$  are two random agents of the colony and  $j$  and  $k$  are two random integers selected from [1, M] interval and  $j \neq k$ . The number of population in the colony is shown by  $M$ , and  $p$  is the adjusting parameter which is set as 0.8 (Arora & Singh, 2019). The fragrance factor ( $f$ ) indicates the objective function value and  $C$  and  $a$  are two tuning parameters which are taken as 0.001 and 0.1, respectively (Arora & Singh, 2019). As illustrated in the given formulation the local search pattern of BOA algorithm is highly stochastic, since  $k$ th agent moves toward  $j$ th agent no matter which of them is in the better position (i.e. have better objective value). So, most of the time such an approach for local search does not provide a promising movement (Mortazavi, 2019a, 2019c). To tackle this problem, the current phase is replaced with the quadratic neighborhood search around the weighted particle scheme. This approach is explained in the next section.

### Enhanced quadratic approximation (EQA)

In the quadratic approximation (QA) approach three parents ( $R_1$ ,  $R_2$  and  $R_3$ ) produce a child agent (Deep & Das, 2009). This process is formulated as follows:

$$R^{new} = 0.5 \frac{(R_2^2 - R_3^2)f(R_1) + (R_3^2 - R_1^2)f(R_2) + (R_1^2 - R_2^2)f(R_3)}{(R_2 - R_3)f(R_1) + (R_3 - R_1)f(R_2) + (R_1 - R_2)f(R_3)} \tag{2}$$

where,  $R_1$ ,  $R_2$  and  $R_3$  are three randomly selected different agents from colony while  $R_1 \neq R_2 \neq R_3$ . Also,  $f(\cdot)$  returns the objective value of any corresponding agent. In the current study, to provide more efficient local search, this scheme is enhanced by setting the one parent (e.g.  $R_1$ ) as the weighted agent ( $R_1 = \mathbf{X}^W$ ). The weighted agent is the weighted average of all available agents, and since it contributes the data of all agents, it is able to spot more promising area in the search domain (Mortazavi & Toğan, 2016). Weighted agent is defined as follows:

$$\begin{aligned}
\mathbf{X}^W &= \sum_{i=1}^M \bar{c}_i^W \mathbf{X}_i^P \\
\bar{c}_i^W &= \left( \hat{c}_i^W / \sum_{i=1}^M \hat{c}_i^W \right) \\
\text{in which } \hat{c}_i^W &= \frac{\max_{1 \leq k \leq M} (f(\mathbf{X}_k^P)) - f(\mathbf{X}_i^P)}{\max_{1 \leq k \leq M} (f(\mathbf{X}_k^P)) - \min_{1 \leq k \leq M} (f(\mathbf{X}_k^P)) + \varepsilon}, \quad i = 1, 2, \dots, M
\end{aligned} \tag{3}$$

in which  $M$  is the number of all agents,  $\mathbf{X}^W$  designates the weighted agent,  $\hat{c}_i^W$  is the coefficient shows the effect of each agent based on its objective function value,  $f(\cdot)$  returns objective function,  $\max_{1 \leq kw \leq M} (f(\mathbf{X}_{kw}^P))$  and  $\min_{1 \leq kw \leq M} (f(\mathbf{X}_{kw}^P))$ , are the worst and best objective values of all agents, respectively. The  $\varepsilon$  is a small positive number (e.g. set as 0.00001) to avoid division by zero condition, if any. So, by substitution the  $R_1$  for  $\mathbf{X}^W$  proposed EQA formulation is defined as follows:

$$R^{new} = 0.5 \frac{(R_2^2 - R_3^2)f(\mathbf{X}^W) + (R_3^2 - (\mathbf{X}^W)^2)f(R_2) + ((\mathbf{X}^W)^2 - R_2^2)f(R_3)}{(R_2 - R_3)f(\mathbf{X}^W) + (R_3 - \mathbf{X}^W)f(R_2) + (\mathbf{X}^W - R_2)f(R_3)} \tag{4}$$

## Enhanced quadratic approximation integrated with butterfly optimization algorithm (EQB)

Despite of effective global search strategy of BOA, the local search of this method is not enough to handle the complex search domains with high number of local optima or engineering problems with different (convex or non-convex) boundaries (Mortazavi, 2019a, 2019c). To mitigate this shortcoming, the global search strategy of BOA is combined with EQA search scheme. The new technique is named as enhanced quadratic approximation integrated with butterfly optimization (EQB) method and it is mathematically formulated as follows:

$$\begin{aligned}
 & \text{if } r_1 \leq p \\
 & \quad t^{+1}X_i = {}^tX_i + (r_2^2 \cdot X^G - {}^tX_i) \cdot f_i \\
 & \text{if } r_1 > p \\
 & \quad t^{+1}X_i = 0.5 \frac{({}^tX_j^2 - {}^tX_k^2)f(X^W) + ({}^tX_k^2 - (X^W)^2)f({}^tX_j) + ((X^W)^2 - {}^tX_j^2)f({}^tX_k)}{({}^tX_j - {}^tX_k)f(X^W) + ({}^tX_k - X^W)f({}^tX_j) + (X^W - {}^tX_j)f({}^tX_k)}
 \end{aligned} \tag{5}$$

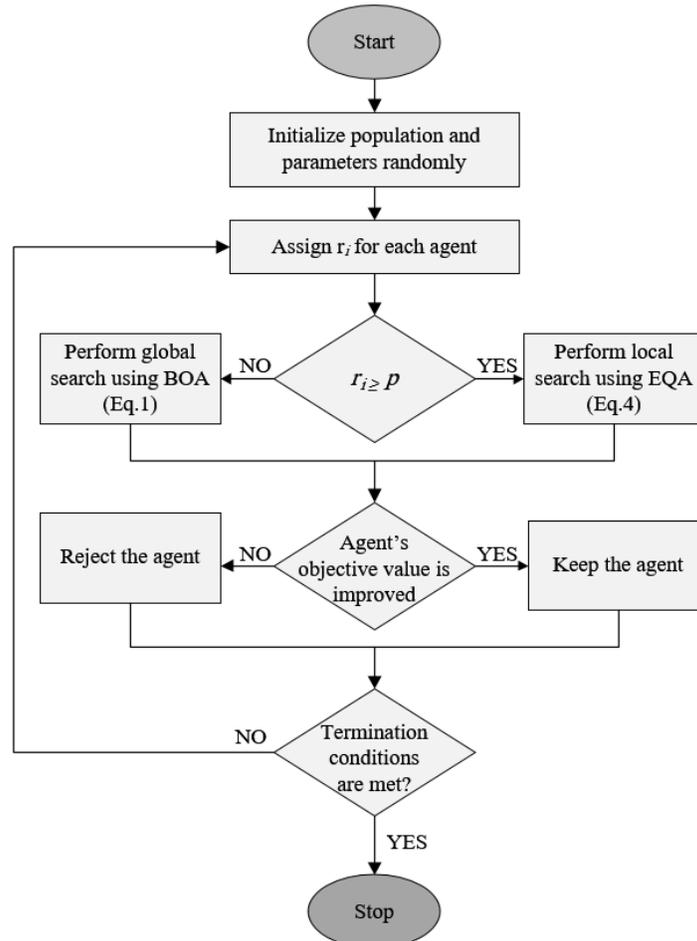
where

$$f_i = C \cdot f^a$$

$$i \neq j \neq k$$

where,  $f(\cdot)$  returns the objective value of any corresponding agent and  $X^G$  and  $X^W$  are the best and weighted agents, respectively.  $C$  and  $a$  are defined same as standard BOA. The probability factor ( $p$ ) is determined via sensitivity analyses and its value is set as  $p=0.5$ . The number of population in the colony is demonstrated by  $M$ . Also,  $i$  index indicates the current agent while  $j$  and  $k$  indexes are two random integer values selected from  $[1, M]$  interval. For more clarity the flowchart for EQB is given in Figure 1. The efficiency and search capability of BOA is tested in the next section.

Figure 1. Flowchart for EQB method. (Self-Elaboration).



### Assessing EQB on unconstrained problems

In this section, the search ability of proposed EQB is verified on some unconstrained problems. Initially, to determine the most proper value for balance parameter ( $p$ ) a sensitivity analysis is performed over this parameter. The convergence rate, accuracy level, diversity and complexity indexes of EQB is comparatively evaluated. To prevent any premature convergence each example is solved for 30 times. For all runs the machine equipped with intel CORE-i7™ processor with 6 MB RAM installed. The parameter setting for applied algorithms are presented in Table 1 as follows:

**Table 1.** Parameter setting for the applied algorithms. (Self-Elaboration).

Algorithm	Year	Pop. size	Parameter
FA (Yang XS, 2009)	2009	50	$\alpha = 0.5, \beta_{min} = 0.2, \gamma = 1$
MBO (Sadollah, Bahreininejad, Eskandar, & Hamdi, 2012)	2012	50	$Ns=12, d_0^f = 300, \mu = 0$
DSO (Kedar Nath Das & Singh, 2014)	2014	50	-
IMO (Javidy, Hatamlou, & Mirjalili, 2015)	2015	50	-
iPSO (Mortazavi & Toğan, 2016)	2017	50	$\alpha = 0.4, C_3 = C_4 = 1, C_2 = 2$
BOA (Arora & Singh, 2019)	2019	50	$C = 0.01, a = 0.1, p = 0.8$
EQB	Current work	50	$C = 0.01, a = 0.1, p = 0.5$

### Sensitivity analysis over balancing parameter ( $p$ )

To select the most proper value for balancing parameter ( $p$ ) a sensitivity analysis is performed over this parameter considering 10 different mathematical functions. The functions and their specifications are addressed in Table 2 and 3, respectively. The results of the sensitivity analysis is tabulated in Table 4. Based on given values in this table  $p=0.5$  is the most proper value for balancing parameter ( $p$ ).

**Table 2.** Benchmark functions for accuracy and sensitivity assessment of EQB. (Self-Elaboration).

No	Function	Formulation
F1	Sphere	$f(\mathbf{X}) = \sum_{i=1}^D x_i^2$
F2	Schwefel 2.22	$f(\mathbf{X}) = \sum_{i=1}^D  x_i  + \prod_{i=1}^D  x_i $
F3	Schwefel 1.2	$f(\mathbf{X}) = \sum_{i=1}^D \left( \sum_{j=1}^i x_j \right)^2$
F4	Schwefel 2.21	$f(\mathbf{X}) = \max_{i=1, \dots, D}  x_i $
F5	Rosenbrock	$f(\mathbf{X}) = \sum_{i=1}^{D-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$
F6	Rastrigin	$f(\mathbf{X}) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10)$
F7	Alpine	$f(\mathbf{X}) = \sum_{i=1}^D  x_i \sin(x_i) + 0.1x_i $
F8	Griewank	$f(\mathbf{X}) = \sum_{i=1}^D \frac{x_i^2}{4000} - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
F9	Ackley	$f(\mathbf{X}) = -20 \exp\left(-0.2 \sqrt{\frac{\sum_{i=1}^D x_i^2}{D}}\right) - \exp\left(\sum_{i=1}^D \cos\frac{(2\pi x_i)}{D}\right) + 20 + e$
F10	Powell	$f(\mathbf{X}) = \sum_{i=1}^{\frac{n}{4}} [(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4]$

**Table 3.** Properties of the benchmark functions. (Self-Elaboration).

No	Features	Range	$f_{\min}$	$D$	Accuracy level
F1	US	[-100, 100]	0	50	1.0E-06
F2	UN	[-10, 10]	0	50	1.0E-06
F3	UN	[-100, 100]	0	50	1.0E-06
F4	UN	[-100, 100]	0	50	1.0E-06
F5	MN	[-30, 30]	0	50	1.0E-02
F6	MS	[-5.12, 5.12]	0	50	1.0E-02
F7	MS	[-10, 10]	0	50	1.0E-02
F8	MN	[-600, 600]	0	50	1.0E-02
F9	MN	[-32, 32]	0	50	1.0E-02
F10	UN	[-4, 5]	0	50	1.0E-06

U: unimodal; M: multimodal; S: separable; N: non-separable; D: dimension

Table 4. Sensitivity result for balancing parameter ( $p$ ). (Self-Elaboration).

$p$	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
0.1	4.36E-085	7.52E-72	8.43E-09	2.89E-04	6.32E+02	9.41E-45	8.87E-05	9.98E-10	3.65E-12	9.87E-15
0.2	6.82E-132	5.01E-73	2.85E-10	1.23E-05	6.98E+01	9.65E-82	5.38E-12	2.35E-12	2.68E-18	6.58E-18
0.3	9.12E-132	5.55E-90	1.08E-11	8.76E-15	4.24E+01	5.58E-85	2.55E-14	1.58E-14	4.57E-20	5.83E-18
0.4	6.99E-140	6.11E-91	5.65E-14	8.11E-15	4.24E+01	<b>2.29E-89</b>	1.45E-14	<b>8.3E-15</b>	2.33E-20	2.40E-19
0.5	<b>4.98E-140</b>	<b>4.12E-91</b>	<b>4.99E-14</b>	<b>7.09E-15</b>	<b>4.24E+01</b>	2.30E-89	<b>1.42E-14</b>	8.33E-15	<b>5.80E-21</b>	<b>2.40E-19</b>
0.6	5.25E-140	8.02E-91	4.99E-14	7.09E-15	4.24E+01	8.02E-89	1.42E-14	8.73E-15	2.29E-20	2.40E-19
0.7	5.06E-139	2.54E-85	5.54E-13	5.78E-14	4.24E+01	9.28E-85	3.98E-14	8.91E-15	2.38E-20	8.36E-18
0.8	5.22E-138	1.09E-39	9.85E-13	1.66E-13	5.08E+01	3.65E-84	4.19E-14	8.92E-15	5.24E-18	7.14E-17
0.9	8.16E-137	2.94E-38	9.98E-13	3.90E-11	5.97E+01	3.33E-80	3.42E-11	9.04E-15	9.65E-17	1.36E-17

### Accuracy analysis

In this section the global search capacity of EQB method is evaluated by solving 10 benchmark functions announced in Table 2. The properties of them are reported in Table 3. These functions are applied for testing the ability of the EQB algorithm to spot the global optimum point of the domain, so all of them are selected to be converged into the origin. Indeed, in comparison with functions with shifted, rotated and/or composite domains they have fairly simpler search spaces. To perform a comprehensive evaluation of the proposed method the dimension of the problem ( $D$ ) is set as  $D=50$  for all functions.

To prevent any premature convergence each test is run for 30 times. The success rate (SR) is defined as the percentage of runs which algorithm can achieve the predefined accuracy level (Suganthan et al., 2005). The accuracy level for each function is given in Table 3. To check the stability of the algorithm, the standard deviation (Std.) of each run is given. The stop criterion for process is considered as  $10000 * D$  (Suganthan et al., 2005). All obtained values for explained properties are reported in Table 5 for all tested methods. Based on ranking values given in this EQB improves the exploitation behavior of BOA (e.g. in Ackley function). The statistical data as Success Rate (SR) and standard deviation (Std.) values demonstrate that EQB also enhances the stability of the BOA algorithm. The outcomes indicate that the proposed EQB comparatively shows superior performance on handling the selected benchmark problems.

Table 5. Comparative results for EQB and other techniques. (Self-Elaboration).

Func.	Value	FA	MBO	DSO	IMO	iPSO	BOA	EQB
F1	Mean	8.2258E-04	1.0428E-08	5.9987E-25	8.1242E-01	3.4178E-78	2.0258E-140	5.7643E-138
	Std.	5.6698E-18	2.6587E-09	9.0998E-38	7.1960E+01	1.1360E-80	2.0982E-145	2.102E-153
	SR	16	92	<b>100</b>	0	<b>100</b>	<b>100</b>	<b>100</b>
	Rank	6	5	4	7	3	<b>1</b>	2
F2	Mean	9.5589E-04	5.1934E-07	3.1123E-22	2.1851E+00	3.8675E-54	5.2587E-86	4.5128E-88
	Std.	8.7431E-09	9.6398E-06	9.4865E-25	1.8235E+00	6.8712E-55	9.9465E-88	1.1221E-98
	SR	16	83	<b>100</b>	0	<b>100</b>	<b>100</b>	<b>100</b>
	Rank	6	5	4	7	3	2	<b>1</b>
F3	Mean	8.2201E-02	2.3001E+00	9.2587E-01	6.3548E+00	8.0052E-03	1.0025E-04	2.1245E-11
	Std.	7.3641E-02	1.3687E+00	5.5458E-02	4.2247E+00	0.0032E-02	2.0012E-06	9.9984E-14
	SR	6	0	0	0	84	84	<b>100</b>
	Rank	4	6	5	7	3	2	<b>1</b>
F4	Mean	2.3658E-04	3.5600E+00	2.3365E-04	1.3658E+01	1.0025E-12	1.3407E-11	4.3627E-14
	Std.	8.3247E-05	2.9176E-01	1.8396E-04	8.2714E+01	6.2554E-20	7.0031E-19	6.3167E-18
	SR	10	0	40	0	96	93	<b>100</b>
	Rank	5	6	4	7	2	3	<b>1</b>
F5	Mean	5.3625E+01	4.6124E+01	6.3265E+01	9.0811E+01	3.0987E+01	4.9819E+01	4.2509E+01
	Std.	1.3168E+00	6.3678E+00	5.3014E+00	1.0987E+00	4.6548E+00	2.8824E-01	1.9812E-2
	SR	0	0	0	0	0	0	0
	Rank	3	4	6	7	<b>1</b>	5	2
F6	Mean	9.3654E+01	6.2547E+01	1.8589E-10	1.5487E+02	2.0236E-10	1.6598E-88	2.3114E-89
	Std.	7.3405E+01	2.2523E+00	3.3698E-12	5.0058E+00	8.9162E-12	2.1245E-99	8.8467E-101
	SR	0	0	73	0	84	<b>100</b>	<b>100</b>
	Rank	5	4	2	6	3	<b>1.5</b>	<b>1.5</b>
F7	Mean	2.7569E-02	2.8658E+00	8.3652E-08	1.3654E+01	8.3698E-11	2.3365E-14	1.4378E-14
	Std.	1.3136E-02	2.0269E+00	9.3587E-09	8.3654E+00	9.3654E-12	4.3265E-16	5.7234E-16
	SR	3	0	13	0	54	96	<b>100</b>
	Rank	5	6	4	7	3	2	<b>1</b>
F8	Mean	5.6058E-07	5.0987E+01	2.0587E-12	5.7391E+02	8.4708E-09	7.9087E-14	8.8356E-15
	Std.	8.6987E-08	2.8778E-01	1.8259E-14	7.9456E+01	5.2978E-10	6.9236E-16	2.3678E-16
	SR	26	0	66	0	20	<b>100</b>	<b>100</b>
	Rank	4	5	2	6	3	<b>1.5</b>	<b>1.5</b>
F9	Mean	2.5498E-01	1.2896E+00	5.3698E-08	3.3715E+00	1.8856E+00	1.8186E+00	1.4578E-22
	Std.	7.9632E-02	5.5569E-01	9.2468E-10	1.8026E-01	9.0921E-01	6.4987E-01	7.2634E-12
	SR	0	0	23	0	0	0	<b>100</b>
	Rank	3	4	2	7	6	5	<b>1</b>
F10	Mean	3.5874E-03	4.3658E+03	5.3264E-03	6.3254E+03	6.3254E-04	9.3654E-17	5.5290E-18
	Std.	6.7632E-04	5.3164E+02	7.9865E-03	2.3654E+03	2.3698E-08	2.3658E-18	3.5432E-22
	SR	16	0	13	0	26	93	96
	Rank	4	6	5	7	3	2	<b>1</b>

## Diversity analysis

The diversity is one of the important properties highly affect the search efficiency of the algorithm. High level of the diversity can delay convergence and low level of diversity can lead to premature convergence. So, a proper level of diversity is required for any efficient search strategy. In the current section the diversity level of EQB method is compared and discussed in comparison with BOA method. To evaluate the population diversity level a 2D sphere function is considered as below:

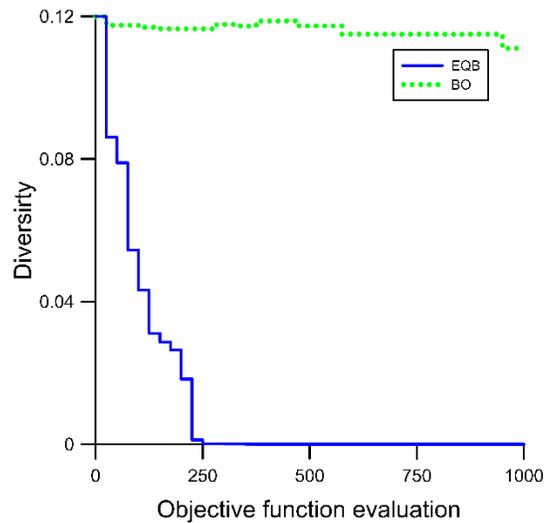
$$f(x) = x_1^2 + x_2^2, \quad x_i \in [-10, +10] \quad (6)$$

The diversity index is defined as follows (Tang et al., 2015):

$$Diversity(t) = \frac{1}{N|L|} \sum_{i=1}^N \sqrt{\sum_{j=1}^D (x_i^j - \bar{x}^j)^2} \quad (7)$$

where, the current step is shown by  $t$ , the population number is demonstrated by  $N$ , the longest diagonal length of search domain is given by  $L$ , problem dimension is shown by  $D$ ,  $\bar{x}^j$  is the  $i$ th agent's  $j$ th component, and  $\bar{x}^j$  is the mean value of all agent's  $j$ th components of the colony. The diversity diagrams for EQB and BOA are plotted in Figure 2. Based on the given diagrams BOA has very high diversity level through whole optimization process. However, EQB can achieve required convergence as expected for simple and smooth domain of 2D sphere function. This shows that BOA can perform high level of exploration but its exploitation behavior is limited, this issue also is shown in the next section via testing the convergence property of the method on different types of search domains.

Figure 2. Diversity diagrams for EQB and BOA. (Self-Elaboration).



## Convergence analysis

In the current section the performance of the proposed EQB is assessed in comparison with its parent methods and some other well-established techniques. To meet this aim, the different functions with different properties are selected. F5-F10 functions are picked from CEC2005 database (Suganthan et al., 2005). The formulations and specifications of selected functions are given in Table 6-7. To give more insight about the search domain of these functions their 2D schemes are given in Figure 3. The convergence history for these functions are given in Figure 3.

For domains of F1 and F3 functions the exploration is the most required search behavior since they have very distinct global optimum points (see Figure 3). BOA shows remarkable performance on solving these two examples, this indicates that the exploration search behavior of this method works proper. Search domains of functions F4 and F5 contain several number

of local minima which challenges the local search ability of the search algorithm. According to the given diagrams for these examples (see Figure 3) BOA performance is highly limited on solving these examples. It indicates that exploitation behavior of BOA is not in the admissible level. Other examples include shifted and/or rotated complex search spaces which challenge both global and local search abilities of any optimization algorithm. Based on the optimization history diagrams given for F6-F10 functions BOA is not able to provide a proper efficiency and search performance.

Considering all optimization history diagrams, in EQB method when the global search is required, BOA provides exploration behavior, and when the local search is dominant EQA search the high certainty region of the search domain which explored so far. Such that, for F13 function, although BOA show very well performance, EQB yields faster convergence. It is notable that for this function DSO cannot catch the minimum required accuracy. For F18 function DSO provides a good convergence rate in comparison with BOA however still EQB provides better rate. Based on these results EQB can successively combine the affirmative exploration and exploitation behaviors of BOA and EQA and put forward an efficient search strategy.

Figure 3. 2D scheme of F11-F20 benchmark functions. (Self-Elaboration).

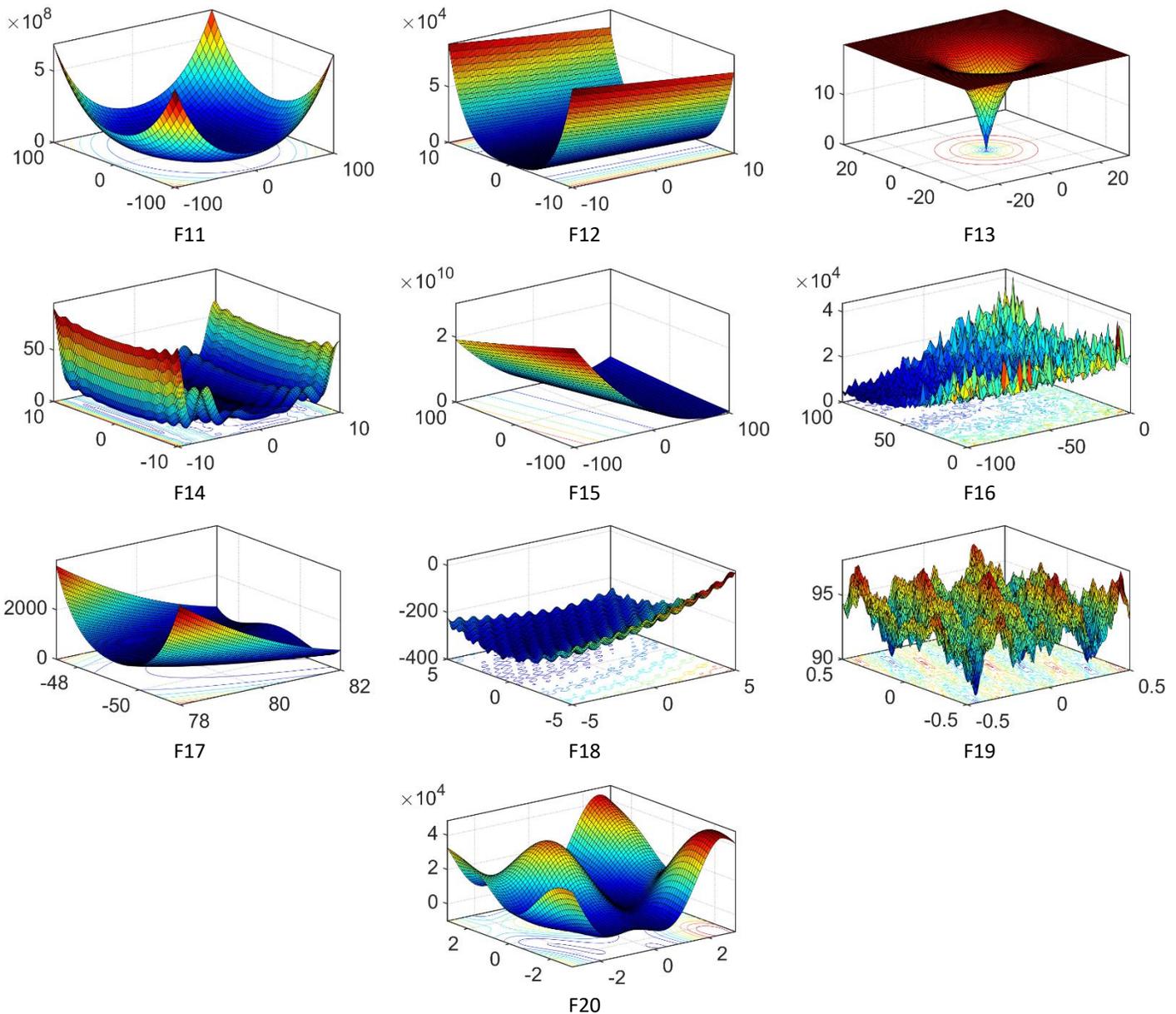


Table 6. Benchmark functions for convergence test. (Self-Elaboration).

No	Function	Formulation
F11	Perm	$f(\mathbf{X}) = \sum_{i=1}^D \left( \sum_{j=1}^D (j^i + 0.1) \left( \left( \frac{x_i}{j} \right)^i - 1 \right) \right)^2$
F12	Dixon-Price	$f(\mathbf{X}) = (x_1 - 1)^2 + \sum_{i=2}^D i(2x_i^2 - x_{i-1})^2$
F13	Ackley	$f(\mathbf{X}) = -20 \exp \left( -0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2} \right) - \exp \left( \sum_{i=1}^D \cos \frac{(2\pi x_i)}{D} \right) + 20 + e$
F14	Levy	$f(\mathbf{X}) = \sin^2(\pi w_1) + \sum_{i=1}^{D-1} (w_i - 1)^2 [1 + 10 \sin^2(\pi w_i + 1)] + (w_D - 1)^2 [1 + \sin^2(2\pi w_D)]$ where, $w_i = 1 + \frac{x_{i-1}}{4}$ , for all $i = 1, \dots, D$
F15	Shifted Rotated Conditioned Elliptic	High $f(\mathbf{X}) = \sum_{i=1}^D (10^6)^{\frac{i-1}{D-1}} z_i^2 - 450$
F16	Shifted Schwefel's Problem 1.2 with Noise in Fitness	$f(\mathbf{X}) = \left( \sum_{i=1}^D \left( \sum_{j=1}^i z_j \right)^2 \right) * (1 + 0.4 N(0,1) ) - 450$
F17	Shifted Rosenbrock's	$f(\mathbf{X}) = \sum_{i=1}^{D-1} (100(z_i^2 - z_{i+1})^2 + (z_i - 1)^2) + 390$
F18	Shifted Rotated Rastrigin's	$f(\mathbf{X}) = \sum_{i=1}^D (z_i^2 - 10 \cos(2\pi z_i) + 10) - 330$
F1	Shifted Rotated Weierstrass	$f(\mathbf{X}) = \sum_{i=1}^D \left( \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k (z_i + 0.5))] \right) - D \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k * 0.5)] + 90$
F20*	Schwefel's Problem 2.13	$f(\mathbf{X}) = \sum_{i=1}^D (\mathbf{A}_i - \mathbf{B}_i(x))^2 - 460$ $\mathbf{A}_i = \sum_{j=1}^D (a_{ij} \sin \alpha_j + b_{ij} \cos \alpha_j), \mathbf{B}_i(x) = \sum_{j=1}^D (a_{ij} \sin x_j + b_{ij} \cos x_j), \text{ for } i = 1, \dots, D$

*D*: dimensions  
**A**, **B** are two  $D \times D$  matrix,  $a_{ij}$ ,  $b_{ij}$  are integer random numbers in the range [-100,100]

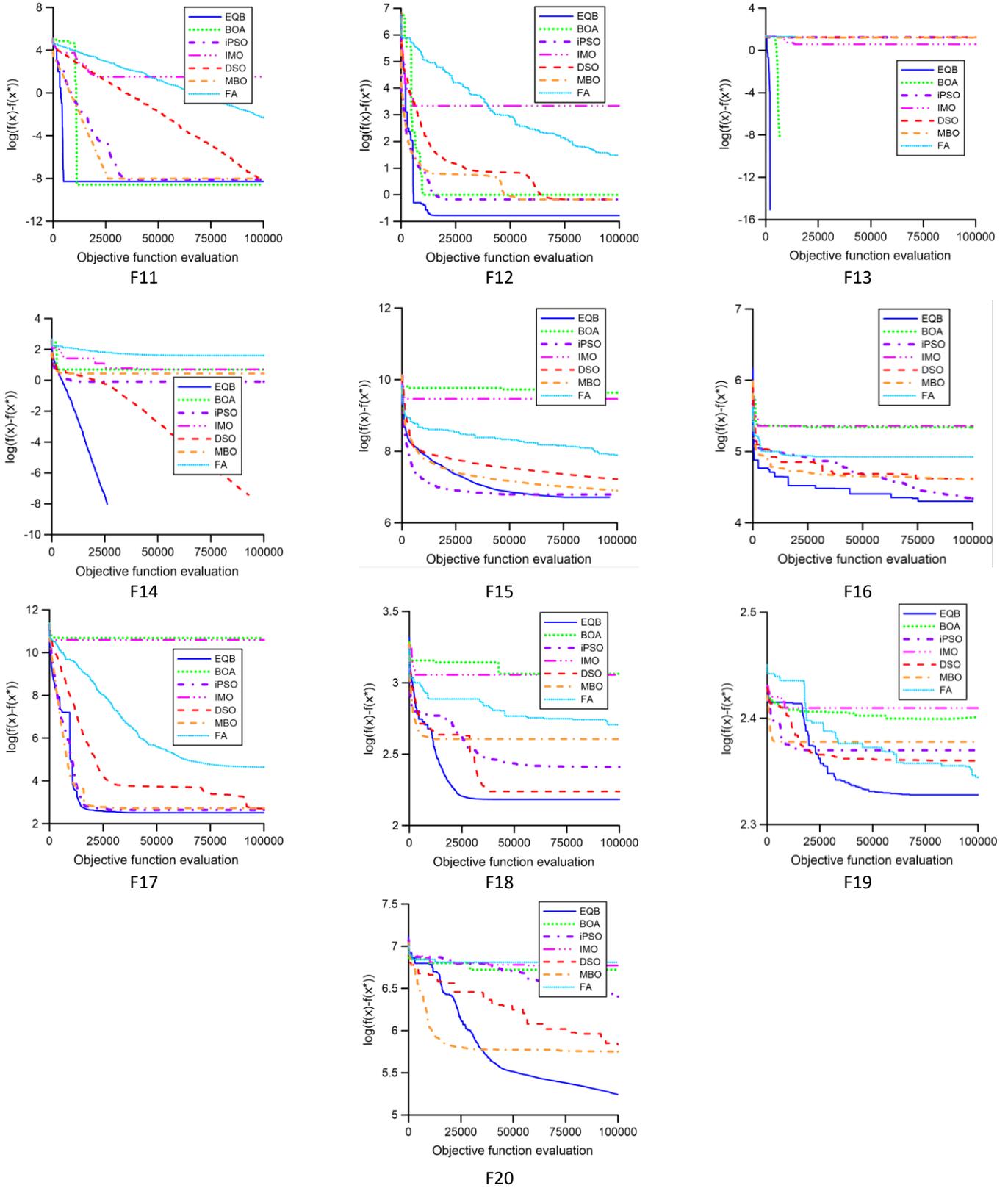
Table 7. Properties of the benchmark functions for convergence rate. (Self-Elaboration).

No	Features	Range	$f_{\min}$	$D$	Accuracy level
F11	US	[-100, 100]	0	50	1.0E-06
F12	UN	[-10, 10]	0	50	1.0E-06
F13	UN	[-100, 100]	0	50	1.0E-06
F14	UN	[-100, 100]	0	50	1.0E-06
F15	UHNRC	[-30, 30]	-450	50	1.0E-06
F16	UHNOC	[-5.12, 5.12]	-450	50	1.0E-06
F17	MHNVC	[-10, 10]	390	50	1.0E-06
F18	MHNLC	[-600, 600]	-330	50	1.0E-06
F19	MHRNDC	[-32, 32]	90	50	1.0E-06
F20	MHNC	[-4, 5]	-460	50	1.0E-06

U: unimodal; M: multimodal; S: separable; N: non-separable; D: dimension; C: scalable; R: rotated; H: shifted; O: noises in fitness; V: very narrow valley from local optimum; L: local optima's number is huge

D:continuous but differentiable only on a set of points

Figure 4. Convergence diagrams for F11-F20 functions. (Self-Elaboration).



## Complexity analysis

In this section the complexity level of EQB in comparison with other selected techniques is assessed. The complexity is evaluated based on the method revealed in (Suganthan et al., 2005). Based on the given formulation a three different durations should be calculated as  $T_0$ ,  $T_1$  and  $\hat{T}_2$ . Such that,  $T_0$  is the required time for running the predefined simple loop (Suganthan et al., 2005).  $T_1$  is defined as required time for 200000 times evaluation of a specific function with certain dimension. In the current study it is taken as Sphere function is selected for  $D=30$  and  $D=50$  dimensions. Eventually,  $T_2$  is defined as the time necessary for complete run of the proposed algorithm (optimization process) for the same function, and  $\hat{T}_2$  is the mean time required for five calculated  $T_2$ . The results for complexity values are given in Tables 8 and 9. Based on the given results although EQB have very close complexity with BOA however it is slightly more complex due to calculating the weighted agent in some iterations.

**Table 8.** Complexity computation result for selected algorithms for  $D=30$ . (Self-Elaboration).

Algorithm	$T_0$	$T_1$	$\hat{T}_2$	$(\hat{T}_2-T_1)/T_0$	Rank
FA	1.40E-01	1.9E-01	5.00E+02	3.58E+03	5
MBO	1.40E-01	1.9E-01	6.00E+02	4.32E+03	6
DSO	1.40E-01	1.9E-01	6.67E+02	4.71E+03	7
IMO	1.40E-01	1.9E-01	3.95E+02	2.83E+03	1
iPSO	1.40E-01	1.9E-01	4.85E+02	3.46E+03	4
BOA	1.40E-01	1.9E-01	4.60E+02	3.28E+03	2
EQB	1.40E-01	1.9E-01	4.61E+02	3.29E+03	3

**Table 9.** Complexity computation result for selected algorithms for  $D=50$ . (Self-Elaboration).

Algorithm	$T_0$	$T_1$	$\hat{T}_2$	$(\hat{T}_2-T_1)/T_0$	Rank
FA	1.40E-01	3.05E-01	5.70E+02	4.07E+03	5
MBO	1.40E-01	3.05E-01	6.85E+02	4.89E+03	6
DSO	1.40E-01	3.05E-01	7.10E+02	5.07E+03	7
IMO	1.40E-01	3.05E-01	4.05E+02	2.89E+03	1
iPSO	1.40E-01	3.05E-01	5.20E+02	3.71E+03	4
BOA	1.40E-01	3.05E-01	4.98E+02	3.55E+03	2
EQB	1.40E-01	3.05E-01	5.01E+02	3.58E+03	3

## Testing EQB on real world problems

In this section the search performance of proposed EQB is evaluated on handling the constrained engineering problems. To handle the constraints of problems the penalty approach is applied. The formulation of penalty function is as below:

$$f_{penalized}(\mathbf{X}) = (1 + \varepsilon_1 v)^{\varepsilon_2} \times f(\mathbf{X})$$

$$v = \sum_{i=1}^q \max\{0, g_i(\mathbf{X})\} \quad (8)$$

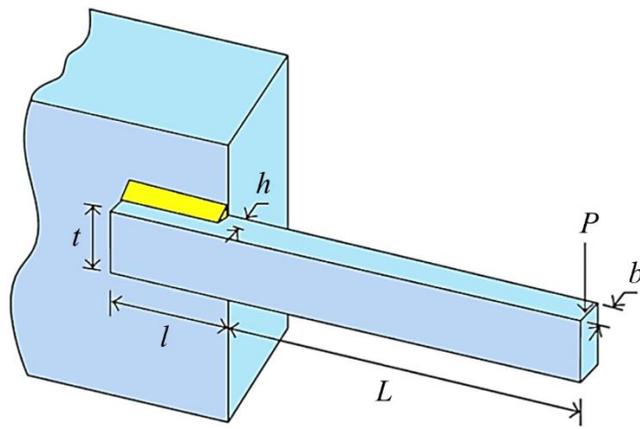
in which,  $f_{penalized}$  indicates the problem's penalized objective function and  $f(\mathbf{X})$  is the value of non-penalized objective function and  $g_i(\mathbf{X})$  returns violation of  $i$ th constraint. To provide more adaptive scheme,  $\varepsilon_1$  and  $\varepsilon_2$  as the tuning terms are respectively taken as 1 and 1.5 at the start of the process while linearly increased up to 6 (Mortazavi, 2019c). Initially Two mechanical cases and finally a structural problem are investigated in this section.

## Design of welded beam problem

As shown in Figure 5 the welded beam problem is studied as the constrained engineering optimization problem. The total cost of the welding mechanism is considered as the objective function of this problem. This function should be minimized concerning seven different linear and non-linear constraints. Required terms on these limitations are defined as the deflection ( $\delta$ ), shear stress ( $\tau$ ), normal stress ( $\sigma$ ). Design variables of this problem are taken into account as the welding part's (shown with yellow color) length ( $l$ ) and height ( $h$ ) and cross-section specifications as width ( $b$ ) and thickness ( $t$ ). The corresponding objective function and its constraints and required terms are given in Eq. (9). The archived results for tested

method are tabulated in Table 10. Based on the given results EQB outperforms other selected methods and also its parent methods (i.e. BOA) in both accuracy and required objective function evaluations (OFEs). This shows that the proposed EQB can efficiently combine the affirmative search abilities of its parental methods. According to statistical outcomes, after iPSO method EQB shows the highest stability in optimization process.

Figure 5. The welded beam problem. (Self-Elaboration).



To minimize

$$\text{cost}(\mathbf{X}) = 1.10471 x_1^2 x_2 + 0.04811 x_3 x_4 (14 + x_2) \quad (9)$$

$$\mathbf{X} = \{x_1, x_2, x_3, x_4\}$$

Subjected to

$$g_1(x) = \tau(x) - \tau_{max} \leq 0$$

$$g_2(x) = \sigma(x) - \sigma_{max} \leq 0$$

$$g_3(x) = x_1 - x_4 \leq 0$$

$$g_4(x) = 0.10471 x_1^2 + 0.04811 x_3 x_4 (14 + x_2) - 5 \leq 0 \quad (9.1)$$

$$g_5(x) = 0.125 - x_1 \leq 0$$

$$g_6(x) = \delta(x) - \delta_{max} \leq 0$$

$$g_7(x) = p - p_c(x) \leq 0$$

$$0.1 \leq x_1 \leq 2 \quad 0.1 \leq x_2 \leq 10 \quad 0.1 \leq x_3 \leq 10 \quad 0.1 \leq x_4 \leq 2$$

where

$$\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau'' \frac{x_2}{2R} + (\tau'')^2}$$

$$\tau' = \frac{P}{\sqrt{2}x_1x_2} \quad \tau'' = \frac{MR}{J} \quad M = P \left( L + \frac{x_2}{2} \right)$$

$$R = \sqrt{\frac{x_2^2}{4} + \left( \frac{x_1 + x_3}{2} \right)^2} \quad J = 2 \left\{ \sqrt{2}x_1x_2 \left[ \frac{x_2^2}{12} + \left( \frac{x_1 + x_3}{2} \right)^2 \right] \right\} \quad (9.2)$$

$$\sigma(x) = \frac{6PL}{x_4x_3^2} \quad \delta(x) = \frac{4PL^3}{Ex_3^3x_4} \quad P_c(x) = \frac{4.013\sqrt{E(x_2^3x_4^6/36)}}{L^2} \left( 1 - \frac{x_3}{2L} \sqrt{\frac{E}{4G}} \right)$$

Table 10. The results for the welded beam problem. (Self-Elaboration).

Variables	FA	MBO	DSO	IMO	iPSO	BOA	EQB
$x_1(h)$	0.202369	0.206428	0.199742	0.234974	0.205729	0.248729	0.205730
$x_2(l)$	3.544214	3.461341	3.612060	2.94398	3.470493	2.95378	3.470489
$x_3(t)$	9.048210	9.021457	9.037500	9.011408	9.036626	8.362973	9.036624
$x_4(b)$	0.205723	0.206422	0.206082	0.206889	0.205729	0.249008	0.205730
$f(\mathbf{X})$							
Best	1.728024	1.727353	1.737297	2.174305	1.724857	1.900415	1.724856
Worst	1.782143	1.800025	1.994651	2.975961	1.724853	3.854357	1.951018
Mean	1.748831	1.729874	1.813290	2.573985	1.724853	2.559287	1.824997
Std.	0.013	0.005898	0.092100	0.589632	6.9E-019	1.983998	0.024100
OFEs	47850	12850	11680	1435	1240	8865	1165

### Design of 582-bar truss tower

The spatial 582-bar spatial tower shown in Figure 6 is studied as the last test case. By considering the symmetry of the tower the elements are categorized into 32 distinct groups. The modulus of elasticity and yielding stress are 29000 ksi and 36 ksi, respectively. The structure is designed for its minimum weight under the loading condition of 1.12, 1.12 and -6.75 kips in x, y and z directions on all nodes, respectively. The size variables are selected from a discrete set of 140 W-shape profiles shown in Table 11. These sections are selected from pre-defined profiles given in AISC-ASD9 code, so the upper and lower boundaries for cross-sectional areas are limited to 6.16 in<sup>2</sup> (39.74 cm<sup>2</sup>) and 215.0 in<sup>2</sup> (1387.09 cm<sup>2</sup>), respectively. Displacements for all of the nodes must be less than 8 cm (3.15 in). Allowable stress limitation of the elements can be calculated according to the AISC-ASD89 standard as below:

$$\begin{cases} \sigma_i^+ = 0.6F_y & \sigma_i \geq 0 \\ \sigma_i^- & \sigma_i < 0 \end{cases} \quad (10)$$

where  $\sigma_i^+$  indicates the allowable tensile stress and  $\sigma_i^-$  indicates the allowable compressive stress. Also,  $\sigma_i^-$  as a function of the slenderness ratio is defined as below:

$$\sigma_i^- = \begin{cases} \left[ \left( 1 - \frac{\lambda_i^2}{2C_c^2} \right) F_y / \left( \frac{5}{3} + \frac{3\lambda_i}{8C_c} - \frac{\lambda_i^3}{8C_c^3} \right) \right] & \text{for } \lambda_i < C_c \\ \frac{12\pi^2 E}{23\lambda_i^2} & \text{for } \lambda_i \geq C_c \end{cases} \quad (11)$$

in which,  $C_c$  is the slenderness ratio, and it is computed as below:

$$C_c = \sqrt{\frac{2\pi^2 E}{F_y}} \quad (12)$$

According to the AISC-ASD code' provisions the maximum allowable slenderness ratio are taken as 200 and 300 for compressive and tensile elements, respectively. Based on this code, the mentioned constraint can be stated as below:

$$\lambda_i = \frac{k_i l_i}{r_i} \leq \begin{cases} 300 & \text{for tension members} \\ 200 & \text{for compression members} \end{cases} \quad (13)$$

where  $\lambda_i$  is slenderness ratio of the  $i$ th member;  $l_i$  and  $r_i$  are respectively the element length and the radius of gyration. In the condition of not satisfying the constraint on the slenderness ratio of compression elements, the allowable stress must not exceed the value achieved by  $\left(\frac{12\pi^2 E}{23\lambda_i^2}\right)$  (AISC-ASD, 1989).

The attained optimal results are comparatively tabulated in Table 12. According to the addressed outcomes EQB can find the lightest structure. Based on reported results the quadratic phase by improving the exploitation search behavior of the proposed algorithm considerably reduces the inefficient iteration and improves the convergence rate and accuracy of the optimal solution. Considering the complexity of the discrete search space of the current example added proposed auxiliary enhanced quadratic approach significantly increase the efficiency of algorithm specially in comparison with its parent method (i.e. BOA) in solving problems with discrete variables and constrained search domains.

**Table 11.** W-shape profile taken from AISC-ASD9. (Self-Elaboration).

W27×178	W21×122	W18×50	W14×455	W14×74	W12×136	W10×77
W27×161	W21×111	W18×46	W14×426	W14×68	W12×120	W10×68
W27×146	W21×101	W18×40	W14×398	W14×61	W12×106	W10×60
W27×114	W21×93	W18×35	W14×370	W14×53	W12×96	W10×54
W27×102	W21×83	W16×100	W14×342	W14×48	W12×87	W10×49
W27×94	W21×73	W16×89	W14×311	W14×43	W12×79	W10×45
W27×84	W21×68	W16×77	W14×283	W14×38	W12×72	W10×39
W24×162	W21×62	W16×67	W14×257	W14×34	W12×65	W10×33
W24×146	W21×57	W16×57	W14×233	W14×30	W12×58	W10×30
W24×131	W21×50	W16×50	W14×211	W14×26	W12×53	W10×26
W24×117	W21×44	W16×45	W14×193	W14×22	W12×50	W10×22
W24×104	W18×119	W16×40	W14×176	W12×336	W12×45	W8×67
W24×94	W18×106	W16×36	W14×159	W12×305	W12×40	W8×58
W24×84	W18×97	W16×31	W14×145	W12×279	W12×35	W8×48
W24×76	W18×86	W16×26	W14×132	W12×252	W12×30	W8×40
W24×68	W18×76	W14×730	W14×120	W12×230	W12×26	W8×35
W24×62	W18×71	W14×665	W14×109	W12×210	W12×22	W8×31
W24×55	W18×65	W14×605	W14×99	W12×190	W10×112	W8×28
W21×147	W18×60	W14×550	W14×90	W12×170	W10×100	W8×24
W21×132	W18×55	W14×500	W14×82	W12×152	W10×88	W8×21

Figure 6. The 582-bar truss tower structure. (Self-Elaboration).

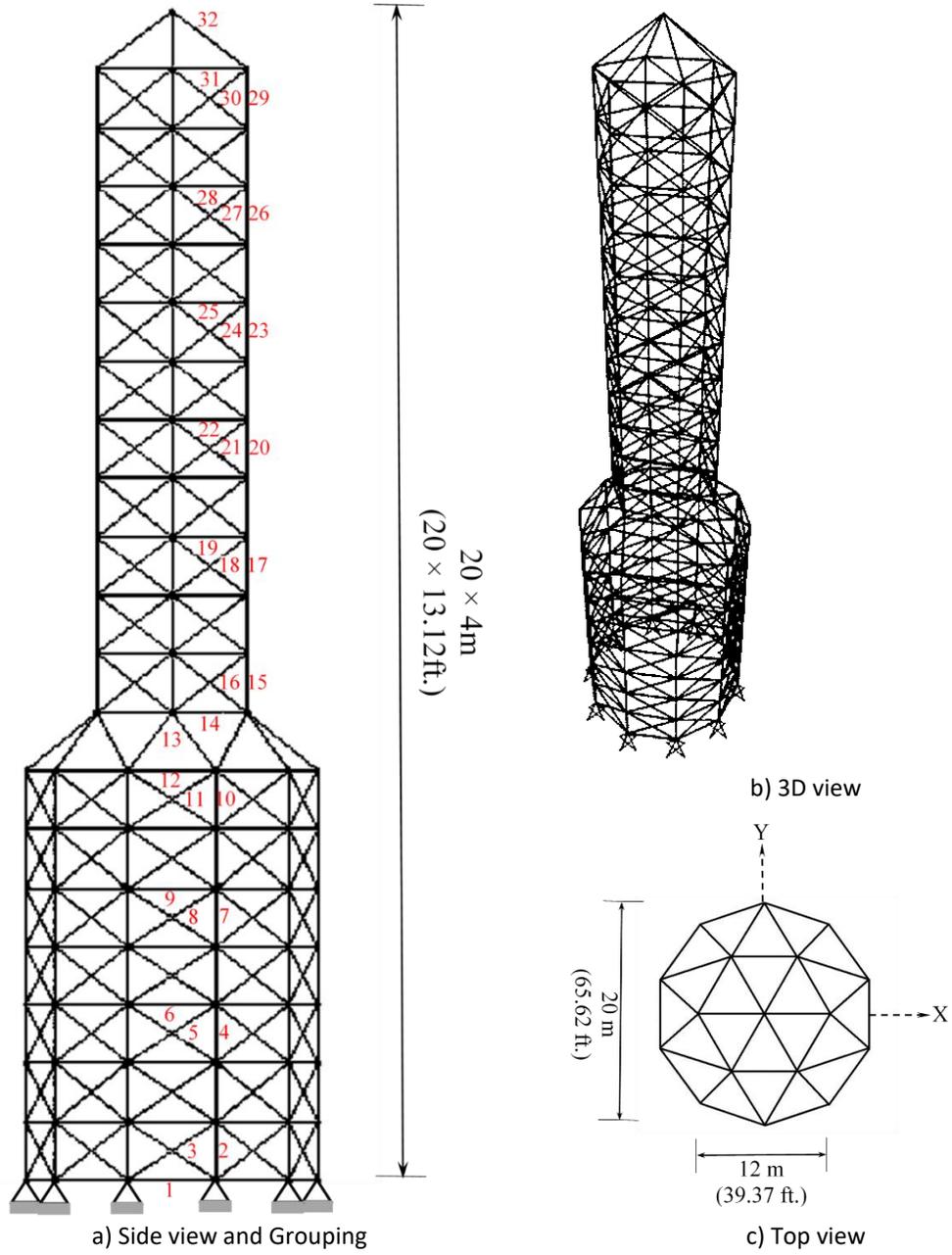


Table 12. Comparison of optimal designs for the 582-bar truss tower. (Self-Elaboration).

Design variables	Optimal cross-sectional areas						
	FA	MBO	DSO	TLBO	iPSO	BOA	EQB
A <sub>1</sub>	W8×21	W8 × 21	W8×24	W8 × 24	W8 × 21	W8×21	W8 × 21
A <sub>2</sub>	W12×79	W24 × 76	W12×72	W24 × 84	W8 × 21	W12×79	W24 × 84
A <sub>3</sub>	W8×28	W8×28	W8×28	W8 × 21	W8 × 21	W8×24	W8 × 21
A <sub>4</sub>	W10×60	W12 × 65	W12×58	W24 × 62	W21 × 73	W10×60	W24 × 62
A <sub>5</sub>	W8×24	W8 × 21	W8×24	W8 × 21	W12 × 53	W8×24	W8 × 21
A <sub>6</sub>	W8×21	W8 × 21	W8×24	W8 × 21	W8 × 21	W8×21	W8 × 21
A <sub>7</sub>	W10×68	W10 × 54	W10×49	W16 × 57	W8 × 21	W8×48	W16 × 57
A <sub>8</sub>	W8×24	W8 × 21	W8×24	W8 × 21	W8 × 21	W8×24	W8 × 21
A <sub>9</sub>	W8×21	W8 × 21	W8×24	W8 × 21	W8 × 21	W8×21	W8 × 21
A <sub>10</sub>	W14×48	W12 × 50	W12×40	W12 × 53	W8 × 21	W10×45	W12 × 53
A <sub>11</sub>	W12×26	W8 × 21	W12×30	W8 × 21	W18 × 76	W8×24	W8 × 21
A <sub>12</sub>	W21×62	W10 × 68	W12×72	W10 × 77	W24 × 62	W10×68	W10 × 77
A <sub>13</sub>	W18×76	W24 × 76	W18×76	W21 × 83	W10 × 49	W14×74	W21 × 83
A <sub>14</sub>	W12×53	W14 × 53	W10×49	W21 × 57	W10 × 49	W8×48	W21 × 57
A <sub>15</sub>	W14×61	W12 × 79	W14×82	W18 × 76	W12 × 79	W18×76	W18 × 76
A <sub>16</sub>	W8×40	W8 × 21	W8×31	W8 × 21	W21 × 62	W8×31	W8 × 21
A <sub>17</sub>	W10×54	W12 × 65	W14×61	W10 × 22	W14 × 43	W8×21	W10 × 22
A <sub>18</sub>	W12×26	W8 × 21	W8×24	W18 × 55	W16 × 26	W16×67	W18 × 55
A <sub>19</sub>	W8×21	W8 × 21	W8×21	W8 × 21	W8 × 21	W8×24	W8 × 21
A <sub>20</sub>	W14×43	W12 × 45	W12×40	W8 × 21	W8 × 21	W8 × 21	W8 × 21
A <sub>21</sub>	W8×24	W8 × 21	W8×24	W14 × 30	W8 × 21	W8×40	W14 × 30
A <sub>22</sub>	W8×21	W8 × 21	W14×22	W8 × 21	W8 × 24	W8×24	W8 × 21
A <sub>23</sub>	W10×22	W16 × 26	W8×31	W8 × 21	W8 × 24	W8 × 21	W8 × 21
A <sub>24</sub>	W8×24	W8 × 21	W8×28	W8 × 21	W8 × 21	W10×22	W8 × 21
A <sub>25</sub>	W8 × 21	W8 × 21	W8 × 21	W8 × 21	W16 × 67	W8×24	W8 × 21
A <sub>26</sub>	W8 × 21	W8 × 21	W8 × 21	W8 × 21	W8 × 31	W8 × 21	W8 × 21
A <sub>27</sub>	W8×24	W8 × 21	W8×24	W10 × 22	W8 × 24	W8 × 21	W10 × 22
A <sub>28</sub>	W8 × 21	W8 × 21	W8×28	W8 × 21	W8 × 21	W8×24	W8 × 21
A <sub>29</sub>	W8 × 21	W8 × 21	W16×36	W8 × 21	W8 × 21	W8 × 21	W8 × 21
A <sub>30</sub>	W6×25	W8 × 21	W8×24	W8 × 31	W8 × 21	W8 × 21	W8 × 31
A <sub>31</sub>	W10×33	W8 × 21	W8 × 21	W8 × 21	W8 × 21	W8×24	W8 × 21
A <sub>32</sub>	W8×28	W8 × 21	W8×24	W12 × 22	W8 × 28	W8×24	W12 × 22
Best vol. (m <sup>3</sup> )	21.0376	20.5688	22.0607	20.3410	20.9464	22.3958	20.3042
Mean vol. (m <sup>3</sup> )	21.9876	21.6831	22.6542	20.6542	21.3241	22.9834	20.6018
Worst vol. (m <sup>3</sup> )	22.0981	21.9101	22.7901	20.7761	21.4782	23.1472	20.6955
Std. (m <sup>3</sup> )	1.07	0.82	0.51	0.37	0.99	0.72	0.36
NSAs	6,450	12,030	15,420	14,940	2,370	17,500	6,300

## Conclusions

The current study deals with introducing an efficient exploitation search pattern by defining a weighted agent and making a quadratic approximation search around this agent. Since the weighted agent is the gravity center of the whole population, it permanently includes the all population information in the search process. The proposed pattern is named enhanced quadratic approximation (EQA). Then, the EQB is combined with the effective exploration search behavior of the BOA approach for introducing a new hybrid search algorithm named enhanced quadratic approximation integrated with butterfly optimization (EQB). The search performance of the EQB, in comparison with some other well-established techniques, is evaluated on handling both constrained and unconstrained mathematical and structural optimization problems. Also, the

accuracy level, diversity index, convergence rate and complexity level of the EQB algorithm are assessed and discussed in details. Based on the achieved results there are three significant acquisitions can be mentioned for the EQB method.

First, the proposed hybrid algorithm by incorporating the weighted agent in the search process dynamically adjusted the diversity level of the population and decreases the probability of premature convergences. Second, using the EQA strategy improves the local search of the algorithm and makes it able to perform a more efficient search in the vicinity of the promising region of the search domain and subsequently raises the accuracy level of the given solution. Third, for the sake of the applied quadratic local search paradigm, the number of ineffective intentions (i.e., iterations without improving the solution) is considerably reduced, and the convergence rate of the algorithm is considerably speeded up. Consequently, acquired numeric outcomes reveal that the EQB provides competitive performance in handling mathematical and structural problems with discrete and continuous decision variables.

## References

- Arora, S., & Singh, S. (2019). Butterfly optimization algorithm: a novel approach for global optimization. *Soft Computing*, 23(3), 715–734. <https://doi.org/10.1007/s00500-018-3102-4>
- Cheng, M.-Y., & Prayogo, D. (2014). Symbiotic Organisms Search: A new metaheuristic optimization algorithm. *Computers & Structures*, 139, 98–112. <https://doi.org/https://doi.org/10.1016/j.compstruc.2014.03.007>
- Das, Kedar Nath, & Singh, T. K. (2014). Drosophila Food-Search Optimization. *Applied Mathematics and Computation*, 231, 566–580. <https://doi.org/https://doi.org/10.1016/j.amc.2014.01.040>
- Dede, T., & Ayvaz, Y. (2015). Combined size and shape optimization of structures with a new meta-heuristic algorithm. *Applied Soft Computing*, 28, 250–258. <https://doi.org/https://doi.org/10.1016/j.asoc.2014.12.007>
- Deep, K., & Das, K. (2009). Performance improvement of real coded genetic algorithm with Quadratic Approximation based hybridisation. *International Journal of Intelligent Defence Support Systems*, 2.
- Degertekin, S. O. (2012). Improved harmony search algorithms for sizing optimization of truss structures. *Computers & Structures*, 92–93, 229–241. <https://doi.org/https://doi.org/10.1016/j.compstruc.2011.10.022>
- Ding, Z. H., Huang, M., & Lu, Z. R. (2016). Structural damage detection using artificial bee colony algorithm with hybrid search strategy. *Swarm and Evolutionary Computation*, 28, 1–13. <https://doi.org/https://doi.org/10.1016/j.swevo.2015.10.010>
- Dorigo, M., & Blum, C. (2005). Ant colony optimization theory: A survey. *Theoretical Computer Science*, 344(2), 243–278. <https://doi.org/https://doi.org/10.1016/j.tcs.2005.05.020>
- Finotto, V. C., da Silva, W. R. L., Valášek, M., & Štemberk, P. (2013). Hybrid fuzzy-genetic system for optimising cabled-truss structures. *Advances in Engineering Software*, 62–63, 85–96. <https://doi.org/https://doi.org/10.1016/j.advengsoft.2013.04.012>
- Gonçalves, M. S., Lopez, R. H., & Miguel, L. F. F. (2015). Search group algorithm: A new metaheuristic method for the optimization of truss structures. *Computers & Structures*, 153, 165–184. <https://doi.org/https://doi.org/10.1016/j.compstruc.2015.03.003>
- Hasançebi, O., Çarbaş, S., Doğan, E., Erdal, F., & Saka, M. P. (2010). Comparison of non-deterministic search techniques in the optimum design of real size steel frames. *Computers & Structures*, 88(17), 1033–1048. <https://doi.org/https://doi.org/10.1016/j.compstruc.2010.06.006>
- Javidy, B., Hatamlou, A., & Mirjalili, S. (2015). Lons motion algorithm for solving optimization problems. *Applied Soft Computing*, 32, 72–79. <https://doi.org/https://doi.org/10.1016/j.asoc.2015.03.035>
- Liang, Y.-C., & Cuevas Juarez, J. R. (2016). A novel metaheuristic for continuous optimization problems: Virus optimization algorithm. *Engineering Optimization*, 48(1), 73–93. <https://doi.org/10.1080/0305215X.2014.994868>
- Moloodpoor, M., Mortazavi, A., & Ozbalta, N. (2019). Thermal analysis of parabolic trough collectors via a swarm intelligence optimizer. *Solar Energy*, 181, 264–275. <https://doi.org/https://doi.org/10.1016/j.solener.2019.02.008>
- Mortazavi, A. (2019a). Comparative assessment of five metaheuristic methods on distinct problems. *Dicle University Journal of Engineering*, 10(3), 879–898.
- Mortazavi, A. (2019b). Interactive fuzzy search algorithm: A new self-adaptive hybrid optimization algorithm. *Engineering Applications of Artificial Intelligence*, 81, 270–282. <https://doi.org/https://doi.org/10.1016/j.engappai.2019.03.005>
- Mortazavi, A. (2019c). The Performance Comparison of Three Metaheuristic Algorithms On the Size, Layout and Topology Optimization of Truss Structures. *Mugla Journal of Science and Technology*, 5(2), 28–41.
- Mortazavi, A. (2020). A new fuzzy strategy for size and topology optimization of truss structures. *Applied Soft Computing*, 93, 106412. <https://doi.org/https://doi.org/10.1016/j.asoc.2020.106412>
- Mortazavi, A. (2021a). Bayesian Interactive Search Algorithm: A New Probabilistic Swarm Intelligence Tested on Mathematical and Structural Optimization

- Problems. *Advances in Engineering Software*, 155, 102994. <https://doi.org/https://doi.org/10.1016/j.advengsoft.2021.102994>
- Mortazavi, A. (2021b). Size and layout optimization of truss structures with dynamic constraints using the interactive fuzzy search algorithm. *Engineering Optimization*, 53(3), 369–391. <https://doi.org/10.1080/0305215X.2020.1726341>
- Mortazavi, A., & Togan, V. (2017a). Sizing and layout design of truss structures under dynamic and static constraints with an integrated particle swarm optimization algorithm, 51, 239–252. <https://doi.org/10.1016/j.asoc.2016.11.032>
- Mortazavi, A., & Togan, V. (2017b). Triangular units based method for simultaneous optimizations of planar trusses. *Advances in Computational Design*, 2(3), 195–210.
- Mortazavi, A., & Toğan, V. (2016). Simultaneous size, shape, and topology optimization of truss structures using integrated particle swarm optimizer. *Structural and Multidisciplinary Optimization*, 54(4). <https://doi.org/10.1007/s00158-016-1449-7>
- Mortazavi, A., Toğan, V., Daloğlu, A., & Nuhoglu, A. (2018). A. Comparison of Two Metaheuristic Algorithms on Sizing and Topology Optimization of Trusses and Mathematical Functions. *Gazi University Journal of Science*, 31(2), 416–435.
- Mortazavi, A., Togan, V., & Moloodpoor, M. (2019). Solution of structural and mathematical optimization problems using a new hybrid swarm intelligence optimization algorithm. *Advances in Engineering Software*, 127. <https://doi.org/10.1016/j.advengsoft.2018.11.004>
- Mortazavi, A., Toğan, V., & Nuhoglu, A. (2018). Comparison of Two Metaheuristic Algorithms on Sizing and Topology Optimization of Trusses and Mathematical Functions. *Gazi University Journal of Science*, 31(2), 416–435.
- Mortazavi, A., Toğan, V., & Nuhuğlu, A. (2017). An integrated particle swarm optimizer for optimization of truss structures with discrete variables. *Structural Engineering and Mechanics*, 61(3). <https://doi.org/10.12989/sem.2017.61.3.359>
- Nobile, M. S., Cazzaniga, P., Besozzi, D., Colombo, R., Mauri, G., & Pasi, G. (2018). Fuzzy Self-Tuning PSO: A settings-free algorithm for global optimization. *Swarm and Evolutionary Computation*, 39, 70–85. <https://doi.org/https://doi.org/10.1016/j.swevo.2017.09.001>
- Patel, V. K., & Savsani, V. J. (2015). Heat transfer search (HTS): a novel optimization algorithm. *Information Sciences*, 324, 217–246. <https://doi.org/https://doi.org/10.1016/j.ins.2015.06.044>
- Pavithr, R. S., & Gursaran. (2016). Quantum Inspired Social Evolution (QSE) algorithm for 0-1 knapsack problem. *Swarm and Evolutionary Computation*, 29, 33–46. <https://doi.org/https://doi.org/10.1016/j.swevo.2016.02.006>
- Pence, I., Cesmeli, M. S., Senel, F. A., & Cetisli, B. (2016). A new unconstrained global optimization method based on clustering and parabolic approximation. *Expert Systems with Applications*, 55, 493–507. <https://doi.org/https://doi.org/10.1016/j.eswa.2016.02.036>
- Quagliaroli, M., Malerba, P. G., Albertin, A., & Pollini, N. (2015). The role of prestress and its optimization in cable domes design. *Computers & Structures*, 161, 17–30. <https://doi.org/https://doi.org/10.1016/j.compstruc.2015.08.017>
- Sadollah, A., Bahreininejad, A., Eskandar, H., & Hamdi, M. (2012). Mine blast algorithm for optimization of truss structures with discrete variables. *Computers & Structures*, 102–103, 49–63. <https://doi.org/https://doi.org/10.1016/j.compstruc.2012.03.013>
- Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. In *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)* (pp. 69–73). <https://doi.org/10.1109/ICEC.1998.699146>
- Souza, R. R. de, Fadel Miguel, L. F., Lopez, R. H., Miguel, L. F. F., & Torii, A. J. (2016). A procedure for the size, shape and topology optimization of transmission line tower structures. *Engineering Structures*, 111, 162–184. <https://doi.org/https://doi.org/10.1016/j.engstruct.2015.12.005>
- Suganthan, P., Hansen, N., Liang, J., Deb, K., Chen, Y., Auger, A., & Tiwari, S. (2005). *Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization, Technical Report. KanGAL #2005005, IIT Kanpur*. Singapore.
- Tang, K., Li, Z., Luo, L., & Liu, B. (2015). Multi-strategy adaptive particle swarm optimization for numerical optimization. *Engineering applications of artificial Intelligence*, 37, 9–19. <https://doi.org/https://doi.org/10.1016/j.engappai.2014.08.002>
- Yang XS. (2009). Firefly Algorithms for Multimodal Optimization. In O. Watanabe & T. Zeugmann (Eds.), *Stochastic algorithms: foundations and applications*. Berlin, Heidelberg: Springer. [https://doi.org/https://doi.org/10.1007/978-3-642-04944-6\\_14](https://doi.org/https://doi.org/10.1007/978-3-642-04944-6_14)
- Zheng, Y.-J. (2015). Water wave optimization: A new nature-inspired metaheuristic. *Computers & Operations Research*, 55, 1–11. <https://doi.org/https://doi.org/10.1016/j.cor.2014.10.008>