Scientia Iranica A (2021) 28(4), 1973–1993

SCIENTIA IRANICA

Sharif University of Technology
Scientia Iranica
Transactions A: Civil Engineering
http://scientiairanica.sharif.edu

Enhanced imperialist competitive algorithm for optimal structural design

M. Shahrouzi* **and A. Salehi**

Department of Civil Engineering, Faculty of Engineering, Kharazmi University, Tehran, Iran.

Received 25 June 2019; received in revised form 21 July 2020; accepted 16 November 2020

KEYWORDS
Enhanced imperialist competitive algorithm;
Hybrid optimization method;
Diversity index;
Constrained problem;
Structural sizing design.

Abstract. Solving complex engineering problems using meta-heuristics requires powerful operators to maintain sufficient diversification as well as proper intensification during the search. Standard *Imperialist Competitive Algorithm*, ICA, delays search intensification by propagating it via a number of artificial empires that compete each other until one concurs with the others. An *Enhanced Imperialist Competitive Algorithm* (EICA) is developed here by adding an evolutionary operator to the standard ICA followed by greedy replacement in order to improve its effectiveness. The new operator introduces a walking step directed from the less significant fit with a fitter individual in each pair of the search agents together with a random scaling and pick-up scheme. EICA performance is then compared with ICA as well as genetic algorithm, particle swarm optimization, differential evolution, colliding bodies optimization, teaching-learning-based optimization, symbiotic organisms search in a set of fifteen test functions. Second, a variety of continuous and discrete engineering benchmarks and structural sizing problems are solved to evaluate EICA in constrained optimization. In this regard, a diversity index and other convergence metrics are traced. The results exhibit a considerable improvement on the algorithm using the proposed features of EICA and its competitive performance, compared to other treated methods.

1. Introduction

Optimal design is a rewarding task in several engineering fields. These problems are categorized into continuous and discrete types due to the nature of design variables. In addition, real-world requirements usually bring about a number of constraints to be satisfied. They increase the complexity of engineering problems and raises the need to develop more efficient algorithms to solve them. Nevertheless, no-free-lunch theorem states that no single algorithm is best suited for and capable to solving all problems [1].

Meta-heuristic algorithms constitute a class of optimization methods that can reveal a near-optimal solution in a practical time period. Up to date, several meta-heuristic methods have been developed due to some natural, biological, cultural, or human-based behaviors to explore the search space [2,3]. Among them, the present study concerns *Imperialist Competitive Algorithm* (ICA) [4] that has already been applied in several fields of engineering problems [5]. ICA is distinguished from similar evolutionary algorithms by applying the searching method in a number of interacting subpopulations (*empires*) which will finally collapse into one empire. The best individual within

*. *Corresponding author. Tel.: +98 21 88830891;*
Fax: +98 26 34511001
E-mail addresses: shahruzi@khu.ac.ir (M. Shahrouzi); std_alireza.salehi@khu.ac.ir (A. Salehi)

doi: 10.24200/sci.2020.53827.3441

each empire is called *imperialist*, while others are known as *colonies*. In its basic form, ICA employs two crucial operators called *assimilation* and *competition*. Assimilation deals with the colonies within each empire, while competition operator forces one colony from the weakest empire to move to the best. Such a process is slow to converge and vulnerable to being trapped in local optima [6].

Consequently, several investigators have presented basic ICA variants to improve it. Talatahari et al. [7] modified the assimilation operator in two folds: application of orthogonal vectors and use of four chaotic maps instead of common random generators. As a result, their developed OICA and CICA led to greater robustness than ICA. Chaotic maps were employed by Bahrami et al. [8] to improve ICA. Some other investigators have utilized *revolution* operator to increase diversity of each empire during assimilation [9–11]. Atashpaz-Gargari [4,9] also combined empires with too close imperialists and lowered the frequency of competition to prevent premature convergence. Jain and Nigam [11] hybridized *Genetic Algorithm*, GA, through assimilation, while Khorani et al. [12] suggested recursive implementation of ICA and GA to make use of both during optimization. Karimi et al. suggested an electromagnetism-inspired algorithm to replace assimilation in ICA [13]. Lin et al. [6] enhanced the interaction between different empires through their IEICA method rather than improving only assimilation operator. They utilized information sharing between all empires once by offering an Artificial Imperialist (ICAAI) via weighted-sum over all imperialists and applying crossover on each pair of imperialists (ICACI) [6].

Meta-heuristics enjoy diversified stochastic operators to overpass local optima and use some intensification strategies to refine the search for global optimum. Proper performance of a meta-heuristic method relies on balancing these features during the search. A common approach to achieving such a goal is to hybridize powerful features of distinct algorithms in a new framework [14]. In this regard, the present work investigates the improvement of ICA by applying two distinct solutions; first, a new walking phase is embedded to ICA to increase diversification in the assimilation part; then, a greedy selection is utilized to accelerate the search refinement capability. The proposed algorithm is here-in-after called Enhanced Imperialist Competitive Algorithm (EICA).

The rest of this article paper is organized as follows. Section 2 describes ICA in brief. Theoretical basis and algorithm of EICA are presented in Section 3. Performance of EICA is then evaluated by treating some continuous test functions in Section 4 and well-studied constrained problems in Section 5. Discussion of the numerical results via comparison with a set of other well-known meta-heuristics will conclude the present study.

2. Imperialist Competitive Algorithm (ICA)

Since 2007 when Imperialist Competition or Colonial Competitive Algorithm was first introduced [4], it has been successfully applied to a variety of engineering problems [15–23].

ICA simulates the interaction between a weaker country known as *colony* and its possessing *imperialist* within several subpopulations called *empires*. In other words, the most powerful colony in each empire is its relevant imperialist. The power of an empire depends mostly on the imperialist and partially on their other colonies.

An imperialist tends to absorb not only the colonies in its empire but also the ones in the other empires. The former constitutes a local search, while the latter may result in global search. Such imperialists compete with each other to take control of colonies in other empires. Consequently, some weak empires will collapse during optimization, while more powerful ones will become larger.

Such a process is repeated for the remaining empires. The movement of colonies toward their relevant imperialists as well as the competition among empires and the collapsing mechanism will hopefully cause all countries to converge into one empire. This final empire will possess all the other countries, among which the best colony reveals the optimal solution. Procedure of the aforementioned ICA in a *fitness* maximization problem is briefed as follows where the fitness is taken as opposed to the cost:

Step 1. Initialize N_{imp} number of empires over randomly positioned N_{pop} countries. Evaluate fitness of all countries and identify the fittest empire as its imperialist.

Step 2. For every empire do.

Step 3. For each colony in the current empire, perform Steps 4 and 5.

Step 4. Walking Phase I: Take the current colony in the direction between the colony and its relevant imperialist by:

$$X_{c,new} = X_c + X_{new}, \tag{1}$$

$$V_{new} = (\beta \times rand - 1) + (X_{imp} - X_c), \tag{2}$$

where X_c and $X_{c,new}$ are the current and new positions of the colony, respectively. β is a control parameter greater than 1 and X_{imp} stands for the empire position. The function *rand* generates random numbers between 0 and 1 with a uniform distribution. The term -1 is added to the basic formula in ICA to allow some portion of colonies to get away from the imperialist [24]. The other portion, however, should be greater to guide most of the colonies toward their best. It is utilized by taking a larger value for β, say 4, as recommended in the literature [6].

- Evaluate $X_{c,new}$. Replace X_c with $X_{c,new}$ if $X_{c,new}$ is better.

Step 5. Update the imperialist in each empire by identifying its fittest colony.

Step 6. Compute the total fitness and power of each empire by the following relations:

$$T_{imp} = fitness(X_{imp}) + k \times \frac{\sum_{i=1}^{N_{imp}} fitness(X_{imp})}{N_{imp}}, \tag{3}$$

$$NT_{imp} = T_{imp} - \max(T_{imp}), \tag{4}$$

$$P_{imp} = \left| \frac{NT_{imp}}{\sum_{i=1}^{N_{imp}} NT_{imp}} \right|, \tag{5}$$

where k is a positive number less than 1 and N_{imp} denotes the number of empires. For every empire, P_{imp} is its power, whereas T_{imp} and NT_{imp} denote the total fitness and its normalized value, respectively.

Step 7. Transfer the worst colony from the weakest empire to the empire that has the most likelihood based on P_{imp}.

Step 8. Eliminate the powerless empires.

Step 9. Loop from Step 2 until termination criterion is satisfied, that is, completing the prescribed Number of Function Evaluations (NFE).

According to ICA algorithm, its control parameters are NFE, N_{pop}, N_{imp}, k, and β.

3. Enhanced Imperialist Competitive Algorithm (EICA)

ICA can be distinguished from several other metaheuristics due to its special approach to employing dynamic-size subpopulations and collapsing all of them into one empire. In other words, it provides delayed transfer of the distributed search among subpopulations into the intensifying search within the final empire. Having several control parameters facilitates fine tuning of the algorithm at the expense of extra computational effort as a practical challenge.

An EICA is developed here that embeds a new walking phase to the standard ICA. Through this walk, a colony moves toward (or backward from) another randomly picked-up colony (the neighbor) in the same empire, provided that the neighbor is fitter (or less fit) than that. In doing so, a candidate colony is generated for possible replacement. Such an operator acts as a real-type crossover between these agents upon applying randomized weighting factors. Therefore, it is expected that the effectiveness of the algorithm be improved. Furthermore, the newly generated colony takes the place of the current one if it is fitter. Such a greedy replacement helps intensify the search for better positions and improves the algorithm efficiency. Figure 1 reveals the flowchart of the proposed EICA which is detailed in the following steps:

Step 1. *Initialization:* Generate N_{pop} randomly positioned countries and distribute them via $N_{imp} = N_{pop}/5$ empires. Evaluate the fitness of all countries and identify the fittest of each empire as the imperialist of it.

Step 2. For every empire, perform Step 3.

Step 3. For each colony in the current empire, repeat Steps 4 to 6.

Step 4. Walking Phase I:

- Move any colony except the imperialist by the following relation:

$$X_{c,new} = X_c + (4 \times rand - 1) + (X_{imp} - X_c). \tag{6}$$

- Evaluate $X_{c,new}$ and replace X_c with $X_{c,new}$ if $X_{c,new}$ is better.

Step 5. Walking Phase II:

- Randomly select a neighbor colony, X_{Nb};
- If X_{Nb} is fitter than X_c, then $s = +1$; otherwise, $s = -1$. Generate candidate $X_{c,new}$ by:

$$X_{c,new} = X_c + rand \times s + (X_{Nb} - X_c). \tag{7}$$

- Evaluate $X_{c,new}$. Then, replace X_c with $X_{c,new}$ if $X_{c,new}$ is better.

Step 6. If there is a colony with lower cost (fitter) than the imperialist, exchange their position within the corresponding empire.

Step 7. Compute the total power T_{imp}, normalized total power NT_{imp}, and likelihood probability P_{imp} for all empires by Eqs. (8) to (10):

$$T_{imp} = fitness(X_{imp}) + k \times \frac{\sum_{i=1}^{N_{imp}} fitness(X_{imp})}{2N_{imp}}, \tag{8}$$

$$NT_{imp} = T_{imp} - \max(T_{imp}), \tag{9}$$

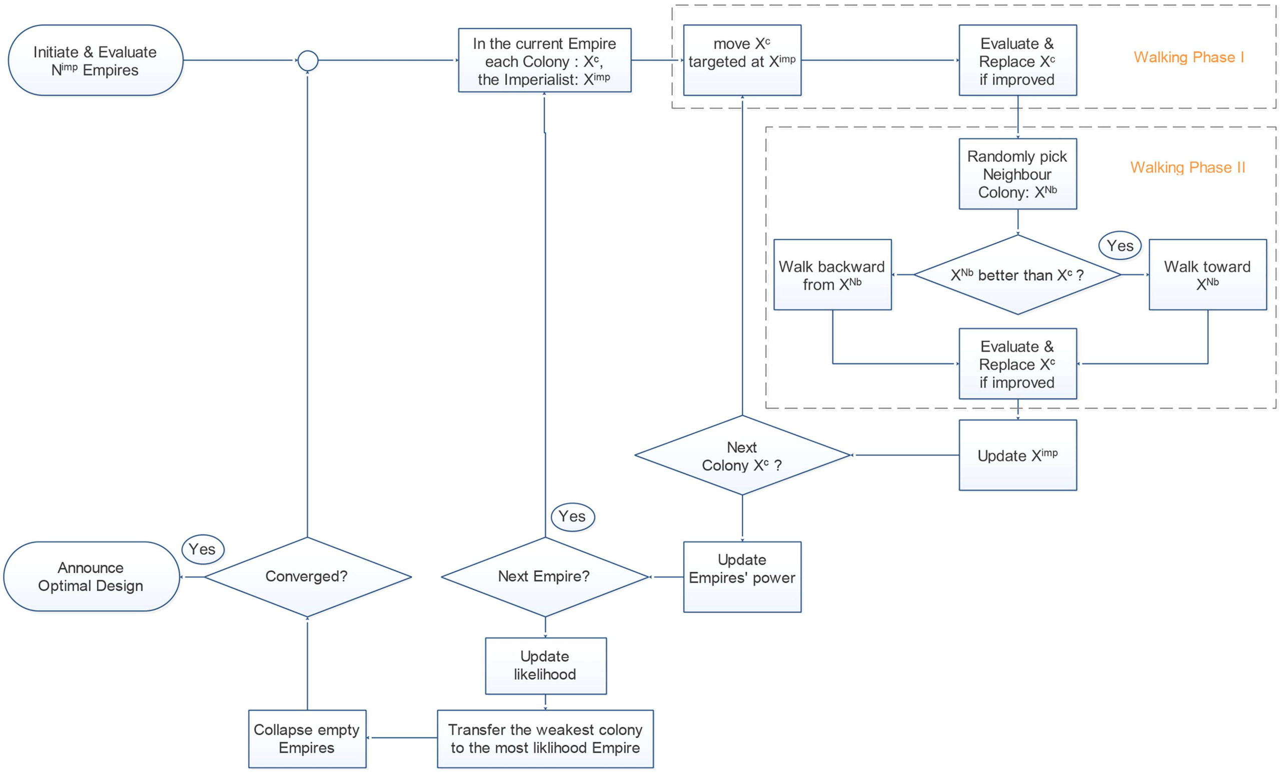

Figure 1. Flowchart of the proposed Enhanced Imperialist Competitive Algorithm (EICA).

Table 1. Unimodal benchmark functions.

Function	Dim	Range
$f_1(x) = (x_1 - 1)^2 + \sum_{i=2}^{d} i(2x_1^2 - x_{i-1})^2$	5	[–10,10]
$f_2(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	5	[–5,10]
$f_3(x) = (\sum_{i=1}^{d} {x_i}^2)^2$	5	[–100,100]
$f_4(x) = \sum_{i=1}^{d} \lvert {x_i}^5 - 3{x_i}^4 + 4{x_i}^3 + 2{x_i}^2 - 10x_i - 4\rvert$	5	[–10,10]
$f_5(x) = \sum_{i=1}^{d} {x_i}^4$	5	[–10,10]

$$P_{imp} = \left| \frac{NT_{imp}}{\sum_{i=1}^{N_{imp}} NT_{imp}} \right| . \tag{10}$$

Step 8. Transfer the worst colony from the weakest empire to the empire with the highest likelihood based on P_{imp}.

Step 9. Eliminate the powerless empires.

Step 10. Repeat Step 2 until termination criterion is satisfied, that is, reaching a prescribed number of iterations, $MaxIter$, or a prescribed number of objective function evaluations, NFE.

4. Unconstrained numerical examples

In order to test the performance of the proposed method for optimization of continuous problems, a set of 15 benchmark functions was selected. They include unimodal, multimodal, and fixed-dimension multimodal functions [25], as described in Tables 1, 2, and 3, respectively.

Table 2. Multimodal benchmark functions.

Function	Dim	Range
$f_6(x) = -20\exp(-0.2\sqrt{(\frac{1}{d}\sum_{i=1}^{d} {x_i}^2)}) - exp(\sqrt{(\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)})) - 20 + exp(1)$	30	[–32.7,32.7]
$f_7(x) = \sum_{i=1}^{d}\frac{{x_i}^2}{4000} - \prod_{i=1}^{d}\cos(\frac{x_i}{\sqrt{i}}) + 1$	30	[–600,600]
$f_8(x) = -\sum_{i=1}^{d}\sin(x_i)\sin\left((i{x_i}^2)/\pi\right)^{20}$	30	[0,π]
$f_9(x) = \sum_{i=1}^{d}{x_i}^6(2+\sin(\frac{1}{x_i}))$	30	[–1,1]
$f_{10}(x) = 1 - \cos(2\pi\sum_{i=1}^{d}{x_i}^2) + 0.1\sqrt{\sum_{i=1}^{d}{x_i}^2}$	30	[–100,100]

Table 3. Fixed-dimension multimodal benchmark functions.

Function	Dim	Range
$f_{11}(x) = -\cos(x_1)\cos(x_2)exp(-(x_1-\pi)^2-(x_2-\pi)^2)$	2	[–100,100]
$f_{12}(x) = (\sum_{i=1}^{5} i\cos((i+1)x_1+i))(\sum_{i=1}^{5} i\cos((i+1)x_2+i))$	2	[–5.12,5.12]
$f_{13}(x) = \sin(x_1)\exp\left(1-\cos(x_2)\right)^2 +$ $\cos(x_2)\exp\left(1-\sin(x_1)\right)^2 + (x_1+x_2)^2$	2	$[-2\pi,\ 2\pi]$
$f_{14}(x) = \sin(x_1+x_2) + (x_1-x_2)^2 - (3/2)x_1 + (5/2)x_2 + 1$	2	$-1.5 \leq x_1 \leq 4$ $-3 \leq x_2 \leq 3$
$f_{15}(x) = (exp(-x_1)-x_2)^4 + 100(x_2-x_3)^6 + (\tan(x_3-x_4))^4 + {x_1}^8$	4	[–1,1]

Table 4. Specific control parameters of the treated algorithms rather than N_{pop} and Number of Function Evaluation (NFE).

Algorithm	Control parameters	Algorithm	Control parameters
PSO	$c_1 = 1,\ c_2 = 2,\ c_3 = 2$	CBO	–
GA	$P_m = 0.1,\ b = 2$	TLBO	–
ICA	$\beta = 4,\ k = 0.5$	SOS	–
DE	–	EICA	–

In this section, N_{pop} and NFE are set to 50 and 2000, respectively. Performance comparison for Particle Swarm Optimization (PSO), GA, ICA, Differential Evolution (DE), Colliding Bodies Optimization (CBO), TLBO, SOS, and the proposed EICA is made, where the last five are considered parameter-less methods. To achieve more reliable results, 30 independent runs are performed for each benchmark problem.

For such mathematical test problems, the fitness is defined as opposed to the corresponding cost function, f, so that the problem formulation is given below:

$$Maximize\ fitness(X) = -\ f(X). \tag{11}$$

During optimization, any design vector X is forced to fall within the range $L \leq X \leq U$ denoting the lower and upper bounds on design variables, respectively.

Intrinsic parameters of PSO include c_1, c_2, and c_3 as the inertial, cognitive, and social factors, respectively. They arise in the following velocity-updating relation according to standard PSO:

$$V_{new} = c_1.V + rand \times c_2(Xpb - X) + rand \times c_3(Xgb - X), \tag{12}$$

where Xpb introduces the previous best position of each particle with the current position, X, and velocity, V. Xgb stands for the global best over all Xpb's. The new position for every particle is thus calculated as follows:

$$X_{new} = X + V_{new}. \tag{13}$$

The employed control parameters are given for each algorithm in Table 4. ICA parameters are used as suggested in [6].

Table 5. Comparison of the best results of the treated test functions over different algorithms.

Test function	PSO	GA	ICA	DE	CBO	TLBO	SOS	EICA
F1	25.79	3.46	1.82	1.50	604.47	7.56	4.52	1.00
F2	62.31	1.98	15.35	1.04	1.28	29.40	1.23	1.00
F3	1005513.00	119.42	20.32	14.74	218488.18	4995.08	54.20	1.00
F4	25.53	5.14	5.92	1.37	844.66	17.18	1.46	1.00
F5	19303.56	30.13	102.45	6.38	238718.13	12120.76	78.69	1.00
F6	1.73	1.00	1.15	2.11	2.55	1.07	2.12	1.00
F7	3.51	1.00	2.18	2.80	3.99	1.91	2.77	1.00
F8	1.02	1.15	1.02	1.19	1.48	1.00	1.06	1.15
F9	445.07	148.42	1.40	19272.51	1.00	2.35	28444.00	1.00
F10	2.14	1.34	1.43	2.66	1.00	1.15	1.33	1.00
F11	1.00	2.62	1.00	1.02	2.62	1.00	1.01	1.00
F12	1.02	1.00	1.01	1.00	1.13	1.00	1.00	1.02
F13	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
F14	1.00	1.00	1.00	1.00	1.01	1.00	1.00	1.00
F15	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 6. Comparison of mean optimization results in the treated test functions.

Test function	PSO	GA	ICA	DE	CBO	TLBO	SOS	EICA
F1	6.59	1.01	1.00	8.58	222.18	4.16	14.06	2.15
F2	4.41	1.01	1.53	1.24	9.65	2.38	1.14	1.00
F3	65.08	1.00	1.00	23.01	1903.05	13.29	26.25	2.06
F4	7.55	1.58	2.11	1.26	418.47	3.16	1.40	1.00
F5	135.81	1.00	6.07	23.73	28243.09	61.27	31.52	1.00
F6	1.64	1.00	1.12	2.47	3.90	1.18	2.42	1.00
F7	1.54	1.00	1.15	3.30	4.07	1.20	3.44	1.00
F8	1.00	1.12	1.10	1.16	1.34	1.01	1.01	1.15
F9	121.11	1.00	1.19	68.80	1362.82	5.46	89.84	1.00
F10	1.77	1.00	1.23	1.67	2.33	1.23	1.37	1.00
F11	1.00	2.00	1.00	1.14	2.00	1.00	1.25	1.00
F12	1.01	1.16	1.01	1.04	1.61	1.00	1.01	1.02
F13	1.00	1.01	1.02	1.01	1.14	1.01	1.00	1.02
F14	1.00	1.00	1.00	1.00	1.09	1.00	1.00	1.00
F15	1.00	1.00	1.00	177.98	73.60	1.00	188.12	1.00

For a clear comparison between the best and mean results in Tables 5 and 6, the values are normalized to the optimal result of each function and the most desired one (the least cost) is highlighted in each case.

Functions F1∼F5 are unimodal and have only one global optimum. These functions allow evaluating the exploitation capability of the investigated meta-heuristic algorithms. According to Table 5, EICA has a superior exploitation capability with respect to others in achieving the best results, exhibiting the efficiency of the proposed algorithm in dealing with a majority of the test cases.

On the other hand, multimodal functions include several local optima whose number can increase with the problem size (the number of design variables). Therefore, this kind of test problems returns useful information when the purpose is to evaluate the exploration capability of an optimization algorithm. The results of Table 5 for the functions F6–F15 indicate that EICA has been very competitive against other treated

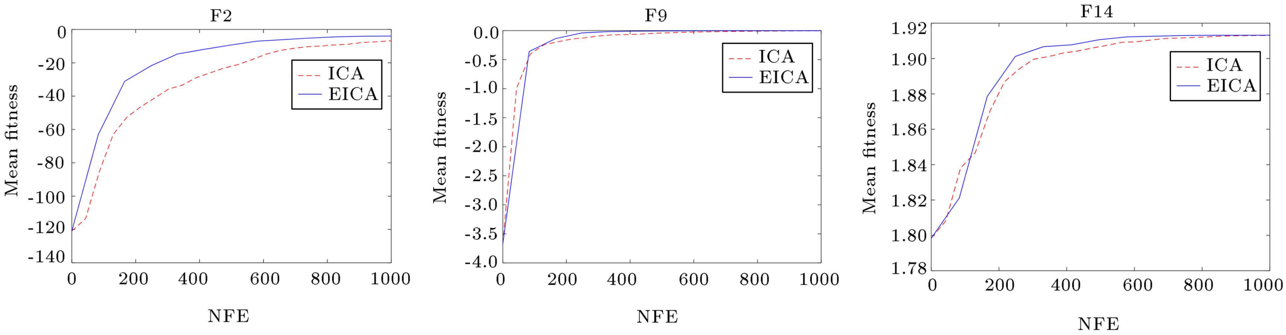

Figure 2. Convergence comparison of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for sample test functions F2, F9, and F14.

methods. Comparison of the mean results in Table 6 shows the overall superior rank of EICA with respect to others in this experiment. Figure 2 demonstrates the superior convergence of EICA over ICA in the sample test functions F2, F9, and F14.

5. Constraint engineering problems

In this section, the performance of EICA is validated by solving several constrained engineering problems (as given in Appendix A) which are widely used in the literature. The maximum NFE has been reported in each case, while other control parameters are taken the same as those in Section 4. For each example, the results have been compared with other well-known optimizers through 30 independent trials, over which the corresponding best solution, mean, and Standard Deviation (SD) are reported. In every run, a randomly initiated population is identically used by the treated algorithms in terms of fair comparison. In addition to the total NFE, the one corresponding to the last improvement in the best run has been reported as NFEb when available.

5.1. Design of tension/compression spring

Arora [26] described this problem for which the aim is to minimize the weight $f(x)$ of a tension/compression spring (as depicted in Figure 3) subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter, and variables' bounds. The design variables include wire diameter $d(x1)$, mean coil diameter $D(x2)$, and number of active coils $P(x3)$.

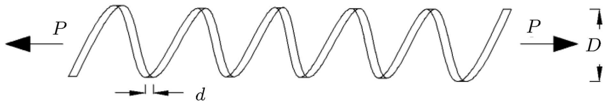

Figure 3. Tension/compression spring design problem.

In the present work, the problem is first solved by PSO, GA, ICA, DE, CBO, TLBO, SOS, and EICA using the same initial population in every trial run by 5000 function calls. Table 7 reports the corresponding results in which EICA outperformed other algorithms by achieving the best result of 0.01266 and the mean of 0.01290 for the cost function. Furthermore, EICA runs have continued up to 10000 function calls. Consequently, the best result (global optimum) has not changed, but the mean and SD have improved, as declared in Table 8.

Table 8 compares the results of EICA in optimal spring design with those reported in the related literature. It can be noted that the proposed hybridization has considerably improved the result of ICA. In addition, EICA has captured global optimum of 0.01266 via just NFEb of 5000 while Mine Blast Algorithm (MBA) has elapsed NFEb of 7650 to find the same result. The second quality rank belongs to Heuristic Particle Swarm Optimization (HPSO) [31] and Whale Optimization Algorithm (WOA) [32] in determining

Table 7. Result of the present work for tension/compression spring design problems.

Design variable	PSO	GA	ICA	DE	CBO	TLBO	SOS	EICA
X_1	0.0500	0.07227	0.05000	0.05000	0.05000	0.0510	0.06312	0.0521
X_2	0.3161	0.67730	0.31147	0.31652	0.31732	0.3396	0.67524	0.3673
X_3	14.4327	10.55366	15.00000	14.75751	14.10949	12.5560	3.85898	10.6963
Best	0.01298	0.04441	0.01324	0.01326	0.01278	0.01284	0.01576	0.01266
Mean	0.01320	0.04442	0.03321	0.13562	0.06228	0.01324	0.09362	0.01290
SD	0.00100	0.00001	0.02100	0.0562	0.0065	0.0008	0.0004	0.0002

Table 8. Comparison results obtained from various algorithms for tension/compression spring design.

Design variable	RO [27]	ES [28]	TWO [29]	MBA [30]	HPSO [31]	WOA [32]	GSA [32,33]	EICA	EICA
X_1	0.0513	0.0519	0.0515	0.0516	0.0517	0.0512	0.0502	0.0521	0.0521
X_2	0.3490	0.3639	0.3543	0.3559	0.3571	0.3452	0.3236	0.3673	0.3673
X_3	11.7627	10.8905	11.4287	11.3446	11.2650	12.0040	13.52 54	10.6963	10.6963
Best	0.01267	0.01268	0.01266	0.01266	0.01267	0.01267	0.01270	0.01266	0.01266
Mean	N/A	N/A	0.01297	0.01271	0.01270	0.01270	0.0136	0.01290	0.01282
SD	N/A	N/A	$2.6E-4$	$6.3E-05$	$1.5E-05$	0.0003	0.0026	0.00020	0.00007
NFE	N/A	N/A	N/A	50000		5000		5000	10000
(NFEb)				(7650)	(81000)	(4410)	(4980)	(5000)	(5000)

Table 9. Results of the present work for pressure vessel design problem.

Design variable	PSO	GA	ICA	DE	CBO	TLBO	SOS	EICA
X_1	1.3426	1.3663	0.9612	1.2772	0.9675	1.2807	1.2672	1.2588
X_2	0.6250	0.6250	0.6250	0.6250	0.6250	0.6250	0.6250	0.6250
X_3	65.4917	65.2428	49.7836	65.4348	50.1309	65.3235	65.4119	65.2252
X_4	10.0000	10.0000	112.2087	10.0000	99.7065	10.0000	10.0000	10.000
Best	5728.54	5714.73	8481.01	5653.77	7950.1	5640.23	5640.35	5603.51
Mean	7059.63	11385.75	8536.40	6318.8	7960.43	6643.83	5883.62	5604.60
SD	893.9	2820.3	170.5	529	24.9	588.1	166.4	1.41

the objective value of 0.01267 by taking 81000 and 4980 function calls, respectively. In this example, EICA has superior quality of final design compared to ICA.

5.2. Pressure vessel design problem

Pressure vessel design problem was introduced by Kannan and Kramer [34] for which the costs of material, forming, and welding were unified into total cost to be minimized. The cylindrical vessel was capped at both ends by hemispherical heads, as shown in Figure 4. There are four design variables in this problem: Ts (X_1, thickness of the shell), Th (X_2, thickness of the head), R (X_3, inner radius), and L (X_4, length of the cylindrical section of the vessel). Here, this example is solved by continuous variables.

According to Table 9, EICA has outperformed the other treated methods in capturing 5603.51 as the best cost. It is well superior to 8481.01 by ICA and 5640.23 by TLBO even within 30000 function evaluations.

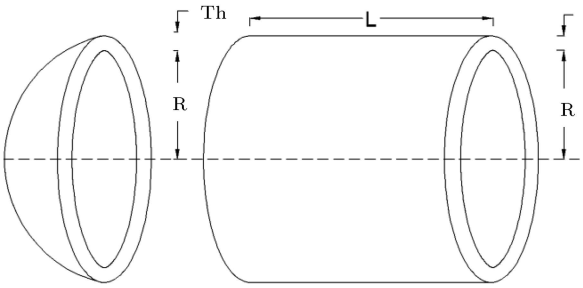

Figure 4. Pressure vessel design problem.

It is noticed that the best result of EICA, i.e., 5603.51, has not changed after NFEb of 16000 where the mean and SD are 5610.27 and 7.54, respectively. According to Table 10, increasing *NFE* to 30000 has improved SD and mean results of EICA. A comparison of our findings with those in the literature in this example points to the superior performance of EICA in terms of effectiveness and robustness.

5.3. I-beam design

This case is modified and derived from the original problem reported in [37] with the aim of minimizing the vertical deflection of an I-beam as the objective (see Figure 5). The beam is simply supported at both ends and should satisfy stress constraints, while 4 continuous design variables are confined within their bounds.

According to Table 11, in this example, EICA

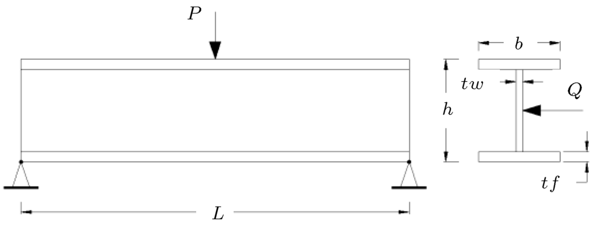

Figure 5. Minimizing vertical deflection of a simply-supported I-beam.

Table 10. Comparison of Enhanced Imperialist Competitive Algorithm (EICA) with literature results for pressure vessel design problem.

Design variable	MBA [30]	GSA [32,33]	VCS [35]	GWO [36]	EICA	EICA
X_1	0.7802	1.1250	0.7782	0.7791	1.2588	1.2588
X_2	0.3856	0.6250	0.3846	0.3846	0.6250	0.6250
X_3	40.4292	58.2901	40.3196	40.3277	65.2252	65.2252
X_4	198.4964	84.4542	199.9999	199.6503	10.000	10.000
Best	5889.32	8538.83	5885.33	5889.37	5603.51	5603.51
Mean	5889.32	8932.95	5911.62	5891.52	5610.27	5604.60
SD	160.34	683.5	57.36	13.91	7.54	1.41
NFE	100000	N/A	N/A	N/A	16000	30000
(NFEb)	(70650)	(7110)	(36020)		(16000)	(16000)

Table 11. Result of the present work for I-beam design problem.

Design variable	PSO	GA	ICA	DE	CBO	TLBO	SOS	EICA
X_1	80.0000	80.0000	80.0000	80.0000	72.5342	80.0000	80.0000	80.0000
X_2	50.0000	16.7337	50.0000	50.0000	42.5342	50.0000	42.5614	50.0000
X_3	1.0890	2.5401	1.4852	0.9747	1.8504	0.9875	1.2053	0.9594
X_4	2.1741	3.3336	0.9000	2.2633	1.8504	2.2440	2.2836	2.2761
Best	0.013566	0.020200	0.025004	0.013265	0.020247	0.013344	0.014849	0.013221
Mean	0.015064	0.025689	0.027439	0.014246	0.027668	0.014118	0.016034	0.014010
SD	0.0008	0.0015	0.0028	0.0014	0.0029	0.0006	0.0007	0.0008

Table 12. Comparison of Enhanced Imperialist Competitive Algorithm (EICA) with literature results for I-beam design example.

Design variable	CS [38]	ARSM [39]	EICA	EICA
X_1	80.0000	80.0000	80.0000	80.0000
X_2	50.0000	37.05	50.0000	50.0000
X_3	0.9000	1.71	0.9594	0.9594
X_4	2.3216	2.31	2.2761	2.2761
Best	0.013074	0.0157	0.013221	0.013221
Mean	0.013216	N/A	0.014010	0.013400
SD	0.00013	N/A	0.00080	0.00027
NFE	25000	N/A	5000	10000
(NFEb)	(5000)		(5000)	(5000)

has captured the best result of 0.013221 and mean of 0.014010 which are both better than the results of other treated algorithms via 5000 function calls.

The best result of the literature, i.e., 0.013074, belongs to Cuckoo Search (CS) [38] with 1% difference from the best result of EICA. Note that the literature works in Table 12 have been conducted with different initial populations and their outcome is consistent with the results of the present works in Table 11. Extending *NFE* from 5000 to 10000 points to the robustness of EICA upon improving SD and mean results.

6. Structural sizing problems

In this section, the performance of EICA is examined in a process that involves solving several structural problems. The results are compared with literature results of other well-known optimizers. Structural weight minimization is of concern, provided that member properties can only be chosen from a discrete set of available structural profiles. In addition to variable bounds, there are several behavioral constraints that should be evaluated via structural analysis.

A *Diversity Index*, DI, is also utilized via Eq. (14) [40]. It is further used to compare the convergence behavior of the algorithms:

$$DI = \underset{j}{mean}\left(\frac{SD_j}{U_j - L_j}\right), \tag{14}$$

where L_j and U_j are the lower and upper bounds on every jth design variable, respectively; meanwhile, SD_j stands for the SD of the corresponding design variable among colonies of the population. DI is evaluated for the entire population at each iteration. The DI should be high at early iterations and it would decrease throughout the search progress as the population agents converge to the optimum. The DI trace versus iterations varies for various algorithms and it can be employed to ensure a better understanding of the behavioral differences. For the structural examples in the present work, a population of size 50 has been identically used in ICA and EICA.

6.1. 10-bar truss design

The 10-bar truss, shown in Figure 6, has been widely addressed by many researchers. The material density is 0.1 lb/in^3 (0.0272 N/cm^3) and the modulus of elasticity equals $E = 10^4$ ksi (68947.57 MPa). Stress limitation in compression and tension for each member is taken ±25 ksi (±172.37 MPa). Maximum nodal displacement in each direction is limited to ±2 in (±0.0508 m). A vertical load of 105 lb is exerted at Nodes 2 and 4.

Optimal results of GA, PSO, PSOPC, HPSO, MBA and ICA, and EICA are compared in the present work for the 10-bar truss, as shown in Table 13. Of note, EICA has successfully captured the best design among the others. The least NFE is 5850 for EICA, while it is more than 10000 for PSO, PSOPC, HPSO, and ICA. The SD of EICA about its mean result is 93.9, while it is 664.1, 12.8, 3.8, and 257.0 for PSO, PSOPC, HPSO, and ICA, respectively.

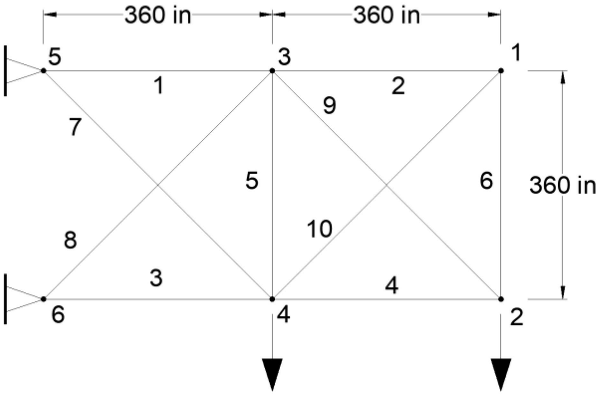

Figure 6. 10-bar truss example.

Figure 7 exhibits the superior convergence of EICA over ICA in this example. The reason for the above finding can be determined by tracing the DI of the methods. According to Figure 8, EICA value has reduced DI more rapidly, while ICA has kept oscillations of the constant DI after an initial decrease. This issue points to quite a different behavior of these algorithms, although EICA is the extension of ICA with effective hybridization.

6.2. 15-bar truss design

The 15-bar truss given in Figure 9 was studied by Li et al. [42] and Sadollah et al. [43]. Material density and elasticity modulus are 7800 kg/m^3 and E = 200 MPa, respectively. The allowable stress for each member of

Table 13. Comparison of results obtained by different algorithms for the 10-bar truss example.

Variables (in^2)	GA [41]	PSO [42]	PSOPC [42]	HPSO [42]	MBA [30]	ICA	EICA
A1	33.5	30	30	30	30	30	33.5
A2	1.62	1.62	1.8	1.62	1.62	1.62	1.62
A3	22	30	26.5	22.9	22.9	30	22.9
A4	15.5	13.5	15.5	13.5	16.9	18.8	14.2
A5	1.62	1.62	1.62	1.62	1.62	1.62	1.62
A6	1.62	1.8	1.62	1.62	1.62	1.62	1.62
A7	14.2	11.5	11.5	7.97	7.97	13.9	7.97
A8	19.9	18.8	18.8	26.5	22.9	16	22.9
A9	19.9	22	22	22	22.9	19.9	22
A10	2.62	1.8	3.09	1.8	1.62	3.13	1.62
Best weight (lb)	5613.8	5581.8	5593.4	5531.9	5507.7	5706.52	5490.73
Mean (lb)	–	–	–	–	–	5920.40	5611.20
SD	–	664.1	12.8	3.8	–	257	93.9
NFE	–	50000	50000	50000	20000	15000	15000
(NFEb)	–	(15000)	(15000)	(12500)	(3600)	(6250)	(5850)

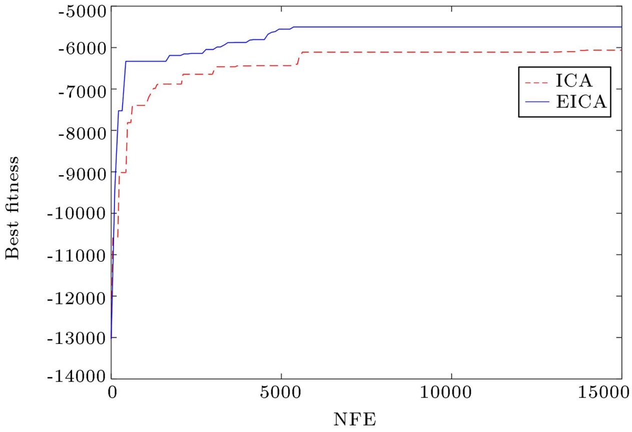

Figure 7. Convergence comparison of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for the 10-bar truss example.

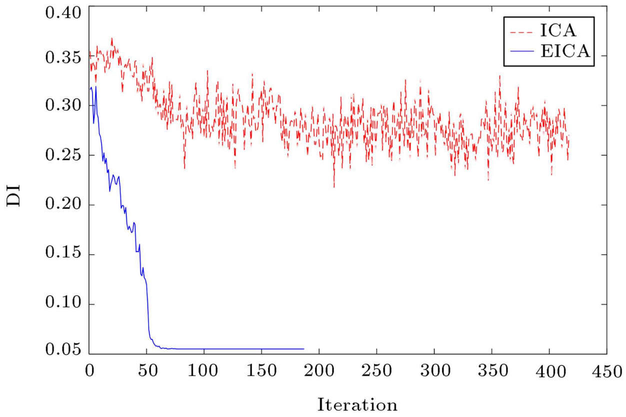

Figure 8. Diversity Index (DI) traces of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for the 10-bar truss.

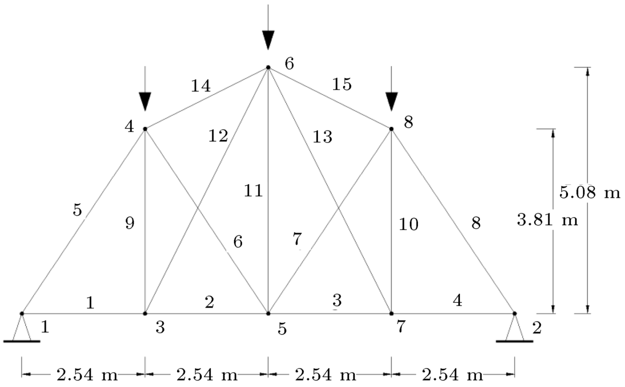

Figure 9. 15-bar truss example.

this structure equals ±120 MPa. Nodal displacements are confined at ±10 mm in any direction.

Discrete design variables are selected from the set {113.2, 143.2, 145.9, 174.9, 185.9, 235.9, 265.9, 297.1, 308.6, 334.3, 338.2, 497.8, 507.6, 736.7, 791.2, 1063.7} (mm^2). Concentrated loads of 35kN are applied at Nodes 4, 6, and 8.

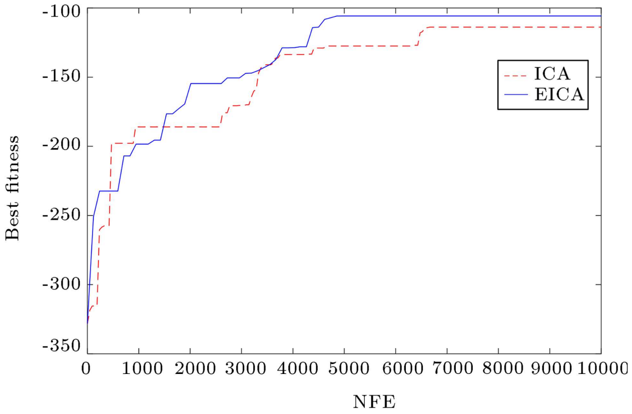

Figure 10. Convergence comparison of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for the 15-bar truss.

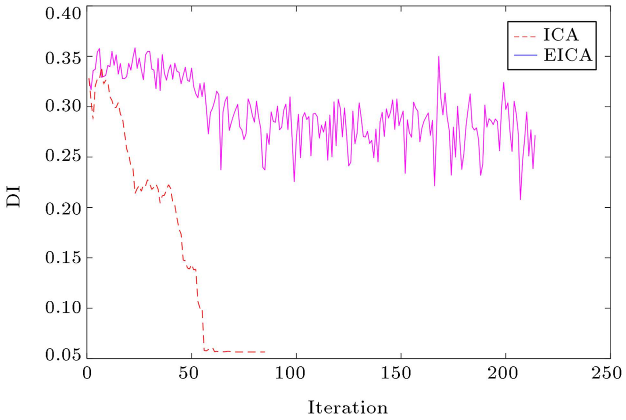

Figure 11. Diversity Index (DI) traces of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for the 15-bar truss.

According to Table 14, the global best design of this example with a weight of 105.735 is captured by HPSO [42], MBA [30], and EICA. As shown in Figure 9, EICA has captured this global optimum through 4850 function evaluations with a better convergence rate than ICA.

According to Figure 10, EICA has rapidly converged to zero velocity, while ICA has a more gradual trend of diversity variation in this example. According to Figure 11, DI trace in EICA rapidly decreases in several runs and the population moves towards an optimal solution; however, it has not been caught in local minimum and has achieved the best fitness. On the contrary, in the ICA, the diversity of the population in each run is almost constant and the convergence process is slow.

6.3. 582-bar tower truss design

As an example of a large-scale problem, a 582-bar

Table 14. Comparison of results obtained from various algorithms for 15-bar truss example.

Variables (mm^2)	PSO [42]	PSOPC [42]	HPSO [42]	MBA [43]	ICA	EICA
A1	185.9	113.2	113.2	113.2	113.2	113.2
A2	113.2	113.2	113.2	113.2	113.2	113.2
A3	143.2	113.2	113.2	113.2	338.2	113.2
A4	113.2	113.2	113.2	113.2	297.1	113.2
A5	736.7	736.7	736.7	736.7	736.7	736.7
A6	143.2	113.2	113.2	113.2	113.2	113.2
A7	113.2	113.2	113.2	113.2	113.2	113.2
A8	736.7	736.7	736.7	736.7	736.7	736.7
A9	113.2	113.2	113.2	113.2	113.2	113.2
A10	113.2	113.2	113.2	113.2	113.2	113.2
A11	113.2	113.2	113.2	113.2	113.2	113.2
A12	113.2	113.2	113.2	113.2	113.2	113.2
A13	113.2	185.9	113.2	113.2	113.2	113.2
A14	334.3	334.3	334.3	334.3	334.3	334.3
A15	334.3	334.3	334.3	334.3	334.3	334.3
Best weight (kg)	108.84	108.96	105.735	105.735	113.836	105.735
Mean	–	–	–	–	126.84	112.02
SD	–	–	–	–	11.36	10.79
NFE	25000	25000	25000	25000	10000	10000
(NFEb)	(18700)	(16000)	(7500)	(2000)	(6500)	(4850)

truss in Figure 12 (80 m tower) was considered. This optimization problem was already solved with discrete variables by Hasançebi et al. [44], Kaveh and Talatahari [45], Kaveh and Mahdavi [46], and Shahrouzi et al. [47]. To keep symmetry of the tower around x- and y-axes, the members are considered for sizing in 32 groups. A single load case consisting of 5 kN force in both x and y directions and a vertical force of 30 kN in the downward z-direction is applied at every node of the tower. The tower is optimized by a minimum volume, while member cross-sections are selected from a list of AISC W-sections based on the area and radii of gyration. The corresponding lower and upper bounds of the section area are 39.74 cm^2 and 1387.09 cm^2, respectively. Nodal displacements are limited to 8.0 cm in each direction. The allowable tensile and compressive stresses are calculated due to the AISC_ASD provisions [48] as follows:

$$\sigma_i^+ = 0.6\ F_y \qquad \text{for} \qquad \sigma_i \geq 0, \tag{15}$$

$$\sigma_i^- = \begin{cases} [(1-\frac{\lambda_i^2}{2C_c^2})\,F_y\,]/(\frac{5}{3}+\frac{3\lambda_i}{8C_c}-\frac{\lambda_i^3}{8C_c^3}) & \text{for} \ \ \lambda_i < C_c \\ \frac{12\pi^2 E}{23\lambda_i^2} & \text{for} \ \ \lambda_i \geq C_c \end{cases} \tag{16}$$

where E (the modulus of elasticity) is 203893.6 MPa and F_y (the yield stress of steel) is taken 253.1 MPa. λ_i is the slenderness ratio ($\lambda_i = kL_i/r_i$) where L_i stands for the length of the ith member and r_i is the corresponding minimal radius of gyration. $C_c = \sqrt{2\pi^2 E/F_y}$ denotes the slenderness measure by which the elastic and inelastic buckling regions are distinguished from

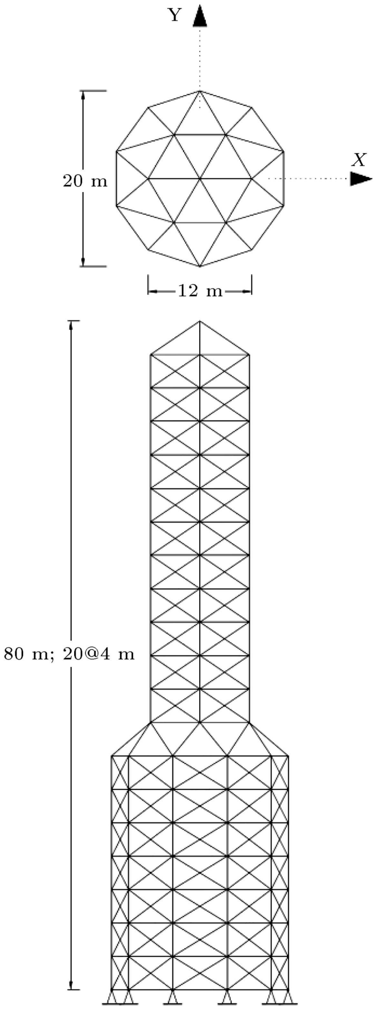

Figure 12. 582-bar tower truss.

Table 15. Comparison of results obtained by different algorithms for the 582-bar truss example.

Variables (cm^2)	**PSO [44]**	**DHPSACO [45]**	**CBO [46]**	**OTLBO [47]**	**ICA**	**EICA**
A1	39.74	45.68	39.74	39.74	39.74	39.74
A2	149.68	136.13	149.68	136.13	123.23	165.16
A3	45.68	53.16	53.23	53.23	39.74	39.74
A4	113.55	109.68	90.96	118.06	104.52	123.23
A5	45.68	45.68	45.68	45.68	39.74	39.74
A6	39.74	45.68	39.74	39.74	39.74	39.74
A7	90.97	92.90	128.38	92.90	155.48	98.84
A8	45.68	45.68	45.68	45.68	39.74	39.74
A9	39.74	92.90	39.74	39.74	39.74	39.74
A10	85.81	45.68	90.96	90.97	66.45	136.13
A11	45.68	45.68	49.35	39.74	39.74	39.74
A12	129.03	75.48	118.06	136.13	155.48	83.87
A13	140.65	56.71	143.87	144.52	156.77	167.10
A14	90.97	136.13	100.64	92.90	101.94	101.94
A15	143.87	143.87	115.48	149.68	101.94	155.48
A16	55.90	92.90	75.48	58.90	57.03	39.74
A17	39.74	155.48	101.93	118.06	231.61	127.10
A18	127.10	45.68	49.35	45.68	39.74	39.74
A19	45.68	39.74	39.74	39.74	87.10	39.74
A20	39.74	75.48	81.29	87.10	72.26	107.74
A21	75.48	45.68	45.68	39.74	39.74	39.74
A22	45.68	41.87	39.74	39.74	160.00	39.74
A23	39.74	58.84	41.89	47.35	39.74	39.74
A24	41.87	53.23	45.68	39.74	39.74	39.74
A25	45.68	39.74	39.74	39.74	39.74	39.74
A26	39.74	39.74	39.74	39.74	39.74	68.39
A27	39.74	45.68	45.68	39.74	39.74	39.74
A28	45.68	53.23	39.74	39.74	39.74	39.74
A29	39.74	68.39	39.74	39.74	39.74	39.74
A30	39.74	45.68	47.35	39.74	39.74	39.74
A31	45.68	39.74	62.64	39.74	41.87	39.74
A32	45.68	45.68	53.22	39.74	100.64	39.74
Best volume (m^3)	22.3958	22.0607	21.8376	20.9835	22.3458	20.5618
Mean	–	–	–	21.2646	22.9541	22.4675
SD	–	–	–	–	2.602	2.550
NFE	50000	17500	20000	132000	25000	25000
(NFEb)	(17500)	(8500)	(17700)	(80400)	(17500)	(12500)

each other. Furthermore, the maximum slenderness ratio λ_m is limited to 300 and 200 for tension and compression members, respectively.

Table 15 reports the optimal design of this problem by PSO [44], DHPSACO [45], CBO [46], OTLBO [47], and ICA and EICA. According to the reported results, EICA has the best quality of final solution among others at 20.56 m^3 and the second rank belongs to OTLBO at 20.89 m^3.

As evident in Figure 13, EICA found the best solutions through 12500 function evaluations; however, ICA has not been able to capture such optimum within 25000 structural analyses. Figure 13 also shows a higher convergence rate of EICA than that of ICA.

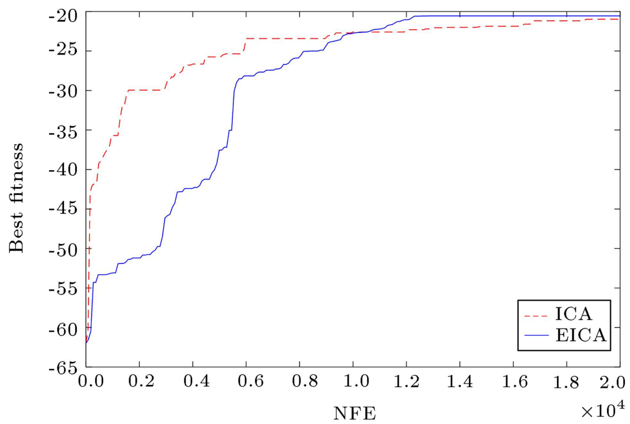

Figure 13. Convergence comparison of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for the 582-bar truss.

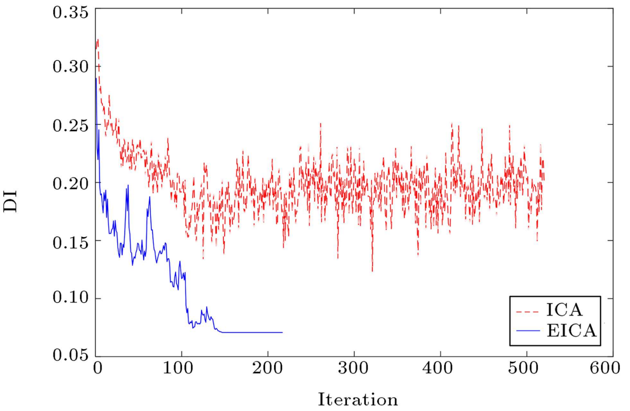

Figure 14. Diversity Index (DI) traces of Enhanced Imperialist Competitive Algorithm (EICA) vs. Imperials Competitive Algorithm (ICA) for the 582-bar truss.

This finding is confirmed by observing DI traces in Figure 14, while ICA has a more gradual trend of diversity variation. The results show that EICA has outperformed ICA in terms of convergence accuracy. EICA controls DI slowly in every iteration. It facilitates EICA to achieve search refinement near the last function calls, while a greater value of DI prevents the real improvement of ICA result.

6.4. 384-bar double-layer barrel vault

Another optimization problem investigated in this article paper is a 384-bar double-layer barrel vault. The span of the barrel vault is 24.82 m with its rise and length being 5.12 m and 26.67 m, respectively. The depth of the structure, i.e., the distance between the top and bottom layers, is equal to 1.35 m. This structure consists of 111 pinned joints and 384 bar elements, which are grouped into 31 independent sizing variables, as identified in Figure 15.

The structural material properties are assumed as follows: the modulus of elasticity is considered to be 30,450 ksi (210,000 MPa), the yield stress of steel

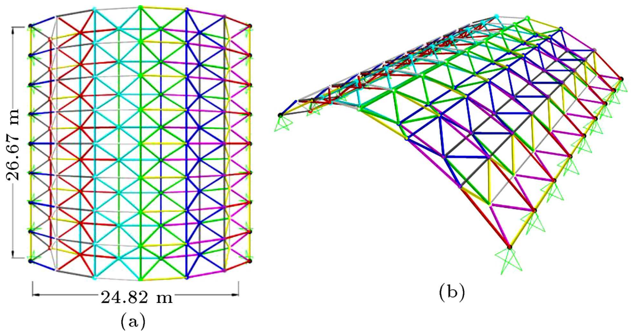

Figure 15. 384-bar double-layer barrel vault: (a) Top view and (b) perspective view.

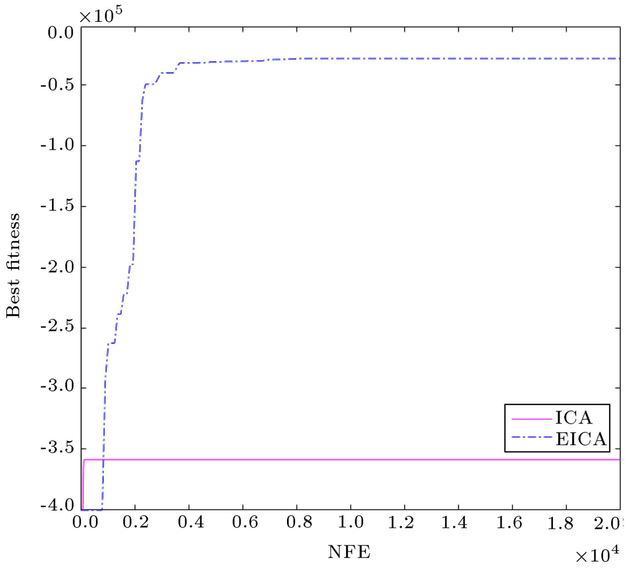

Figure 16. Comparison of convergence rates for the 384-bar double-layer barrel vault.

58 ksi (400 MPa), and the density of steel 0.288 lb/in^3 (7833.413 kg/m^3). All connections are assumed as ball jointed and the supports are considered at two external edges of the top layer of the barrel vault. Vertical concentrated loads of -20 kips (-88.964 kN) are applied to all free joints (nonsupport joints) of the top layer. Strength and slenderness limitations are set according to AISC-ASD provision. Displacement constraints of ±0.1969 in (5 mm) are imposed on all nodes in x, y, and z directions [3].

The design variables are the cross-sectional areas of the bar elements which are selected from a list of steel pipe sections from AISC-ASD. These pipe sections are shown in Table 16.

384-bar barrel vault is a complex spatial example and the total NFE is set to 20000; however, the last improvement of EICA is identified in NFEb of 7800, leading to the best result in Table 17. EICA yields the least weight of 28041.6 kg, confirming its effectiveness, while ICA does not experience any improvement in a considerably heavier design, weighing 35867.2 kg. Figures 16 and 17 compare the convergence rates

Table 16. Structural steel pipe sections.

No.	Type	Nominal diameter (in.)	Area (cm^2)	Gyration radius (cm)
1	ST	$\frac{1}{2}$	1.6129	0.662432
2	EST	$\frac{1}{2}$	2.064512	0.635
3	ST	$\frac{3}{4}$	2.129028	0.846582
4	EST	$\frac{3}{4}$	2.774188	0.818896
5	ST	1	3.161284	1.066038
6	EST	1	4.129024	1.034542
7	ST	$1\frac{1}{4}$	4.322572	1.371346
8	ST	$1\frac{1}{2}$	5.16128	1.582166
9	EST	$1\frac{1}{4}$	5.677408	1.331214
10	EST	$1\frac{1}{2}$	6.903212	2.003806
11	ST	2	6.903212	1.53543
12	EST	2	9.548368	1.945132
13	ST	$2\frac{1}{2}$	10.96772	2.41681
14	ST	3	14.387068	2.955798
15	EST	$2\frac{1}{2}$	14.5161	2.346452
16	DEST	2	17.161256	1.782572
17	ST	$3\frac{1}{2}$	17.290288	3.395726
18	EST	3	19.483832	2.882646
19	ST	4	20.451572	3.835908
20	EST	$3\frac{1}{2}$	23.741888	3.318002
21	DEST	$2\frac{1}{2}$	25.999948	2.143506
22	ST	5	27.74188	4.775454
23	EST	4	28.451556	3.749548
24	DEST	3	35.290252	2.65811
25	ST	6	35.999928	5.700014
26	EST	5	39.419276	4.675124
27	DEST	4	52.25796	3.490976
28	ST	8	54.19344	7.462012
29	EST	6	54.19344	5.577332
30	DEST	5	72.90308	4.379976
31	ST	10	76.77404	9.342628
32	EST	8	82.58048	7.309358
33	ST	12	94.19336	11.10361
34	DEST	6	100.64496	5.236464
35	EST	10	103.87076	9.216898
36	EST	12	123.87072	11.028934
37	DEST	8	137.41908	7.004812

and DI history of ICA and EICA in this example, respectively. The fluctuating trend of DI prevents ICA from true global search such that it stops at a local minimum, thus justifying the lack of proper search refinement by ICA. On the other hand, EICA has not only escaped from local optimum but also obtained the highest quality solution with proper diversity reduction during iterations of the search.

6.5. 1104-bar helipad truss design

In order to evaluate the performance of the proposed algorithms in dealing with a real-world problem, optimal design of a helipad truss in Figure 18 with 1104 bar elements has been introduced in the discrete form for the first time. In the main double-layer grid, the distance between the top and bottom layers is 1 m, while the bottom layer has a bay length of 18 m. The

Table 17. Comparison of results obtained from various algorithms for 384-bar double-layer barrel vault.

Section name	**CBO [3]**	**ECBO [3]**	**VPS [3]**	**MDVC-UVPS [3]**	**ICA**	**EICA**
A1	ST$\frac{1}{2}$	ST$\frac{1}{2}$	ST$\frac{3}{4}$	ST$\frac{1}{2}$	EST 6	ST$\frac{1}{2}$
A2	EST 2	ST 2$\frac{1}{2}$	EST 2$\frac{1}{2}$	EST 2	EST 6	EST 3
A3	EST 2	EST 2	EST 2$\frac{1}{2}$	EST 2	EST$^{3/4}$	EST 3
A4	ST 3	ST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	ST 1$\frac{1}{2}$	ST 4	EST 1$\frac{1}{2}$
A5	DEST 2$\frac{1}{2}$	EST 4	DEST 3	DEST 3	ST3$\frac{1}{2}$	EST 3$\frac{1}{2}$
A6	ST 2$\frac{1}{2}$	ST 1$\frac{1}{2}$	ST 1$\frac{1}{2}$	ST 1$\frac{1}{2}$	EST 12	EST 2$\frac{1}{2}$
A7	ST 12	ST 12	ST 12	ST 12	ST 8	EST 10
A8	DEST 4	ST 10	EST 8	DEST 5	ST 8	ST 10
A9	DEST 5	ST 12	EST 10	EST 10	EST 6	EST 10
A10	ST 12	DEST 8	EST 10	EST 10	ST 10	EST 12
A11	DEST 5	DEST 5	DEST 5	DEST 5	DEST 5	EST 8
A12	DEST 6	EST 8	DEST 5	ST 12	EST 4	DEST 5
A13	DEST 3	ST 6	ST 6	ST 6	DEST 2	DEST 2$\frac{1}{2}$
A14	EST 3$\frac{1}{2}$	EST 3$\frac{1}{2}$	DEST 3	ST 4	EST 4	DEST 2
A15	ST 2$\frac{1}{2}$	ST 2$\frac{1}{2}$	ST 2$\frac{1}{2}$	EST 2$\frac{1}{2}$	ST 2$\frac{1}{2}$	ST$\frac{1}{2}$
A16	EST 6	ST 5	ST 5	ST 4	EST 1$\frac{1}{2}$	ST 4
A17	EST 6	EST 4	DEST 3	ST 6	ST 12	DEST 2
A18	EST 2	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 3	EST 4
A19	EST 2	ST 1$\frac{1}{4}$	ST 1$\frac{1}{4}$	ST 1$\frac{1}{4}$	EST1	ST 4
A20	EST 2$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 2$\frac{1}{2}$	ST$\frac{3}{4}$
A21	EST 4	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST$\frac{1}{2}$	ST$\frac{1}{2}$
A22	ST 3$\frac{1}{2}$	ST 1$\frac{1}{4}$	EST 1$\frac{1}{2}$	ST 1$\frac{1}{4}$	EST$\frac{3}{4}$	ST 1$\frac{1}{2}$
A23	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 6	ST 2
A24	ST 3$\frac{1}{2}$	EST 2$\frac{1}{2}$	EST 2$\frac{1}{2}$	ST 3$\frac{1}{2}$	EST$\frac{1}{2}$	DEST 2
A25	ST 2$\frac{1}{2}$	ST 2$\frac{1}{2}$	EST 2$\frac{1}{2}$	EST 2	ST 6	ST 2$\frac{1}{2}$
A26	DEST 4	ST 2$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 2	EST 1$\frac{1}{2}$	ST 2$\frac{1}{2}$
A27	EST 3	DEST 2	ST 3	ST 3$\frac{1}{2}$	DEST 3	ST 2$\frac{1}{2}$
A28	EST 2	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 2	EST 2	EST 1$\frac{1}{2}$
A29	ST 2$\frac{1}{2}$	ST 2$\frac{1}{2}$	EST 2	EST 2	ST 1$\frac{1}{4}$	ST 2
A30	ST 3	EST 1$\frac{1}{2}$	EST 2	EST 2	EST1	DEST 2
A31	ST 2$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 1$\frac{1}{2}$	EST 2	DEST 5	ST 1$\frac{1}{2}$
Best weight (kg)	31501.3	28343.1	28329.2	28456.3	35867.2	28041.6
Mean	55971.9	29839.5	30798.9	29818.2	38147.1	29645.3
SD	47099.6	1535.8	1321.3	1307.2	1532	945.7
NFE	20000	20000	20000	20000	20000	20000
(NFEb)	(4320)	(15980)	(12780)	(3460)	(300)	(7800)

structural members are divided into 9 groups for sizing design. At the top layer, the concentrated vertical load of 350 kgf is applied at each of 4 central nodes, in addition to the distributed load of 300 kgf/m^2. Nodal displacements are limited to 0.5 cm in each direction. The allowable tensile and compressive stresses are calculated due to the AISC_ASD provisions [48]. In this example, the material density and modulus of elasticity are 7850 kg/m^3 and 2.1×10^8 kN/m^2, respectively. The yield stress of steel material is taken 253.1 MPa. Cross-sectional areas of the bar elements can be selected from a list of standard AISC pipe sections, given in Table 16.

Upon applying N_{pop} of 50 and NFE of 10000, this example is solved by PSO, TLBO, ICA, and EICA. Additional parameters of PSO and ICA are set according to Table 4.

Figure 19 demonstrates a sample convergence history of PSO, TLBO, and ICA. As can be realized,

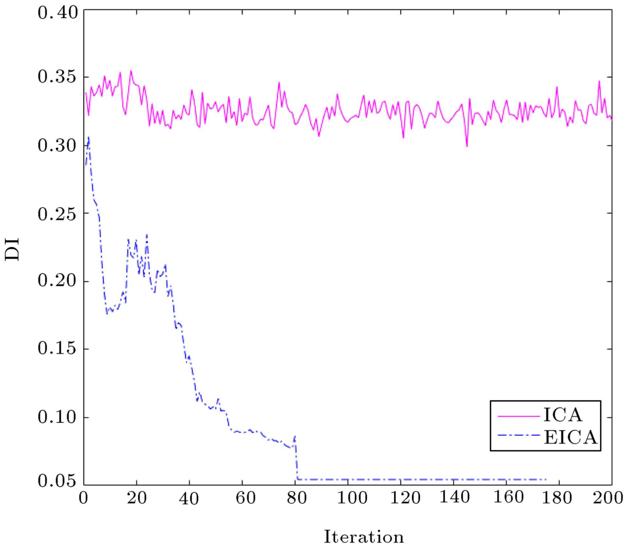

Figure 17. Comparison of diversity history for the 384-bar double-layer barrel vault.

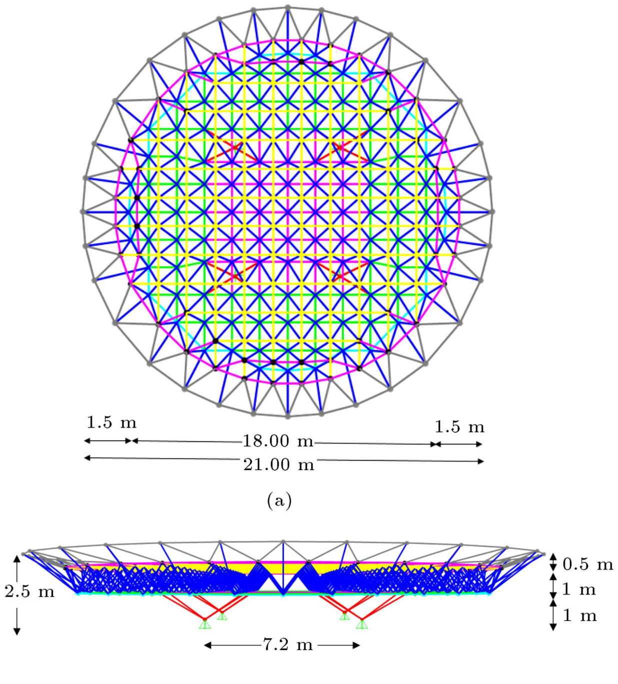

Figure 18. Helipad (1104-bar) truss: (a) Top view and (b) side view.

PSO is trapped in local optima. On the other hand, EICA has captured the highest quality result among the others, reported as 24459.8 kg in Table 18.

The diversity variation for these algorithms is compared in Figure 20. Of note, PSO fluctuates at about a relatively high DI after an initial increase, while TLBO and EICA have a decreasing trend of diversity up to the end of their search. ICA begins with a decreasing trend, but keeps oscillating about a constant DI that is lower than PSO and greater than

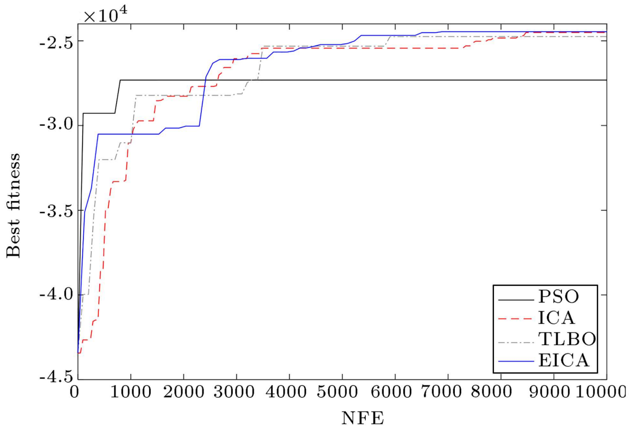

Figure 19. Convergence comparison for the 1104-bar truss.

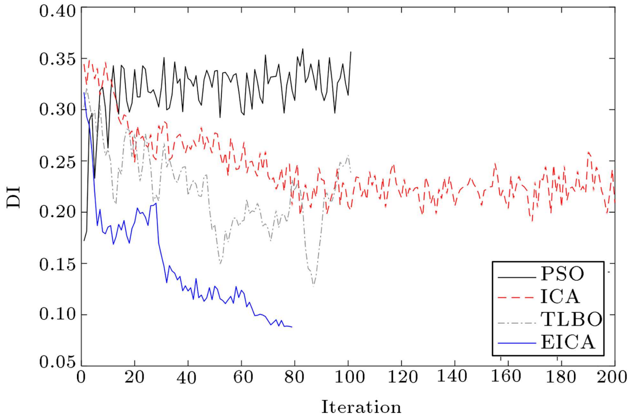

Figure 20. Comparison of Diversity Index (DI) history for the 1104-bar truss.

the others. The highest convergence rate belongs to EICA which causes a more rapid DI decrease among the others. Note that not all methods spend the same number of iterations to reach a prescribed NFE.

According to DI trace comparison in Figure 20, it may be observed that the best trend of diversity variation belongs to EICA which allows for significant exploration upon the start of optimization, but properly decreases DI to allow for search refinement in final iterations. On the other hand, keeping a high DI (such as PSO) can prevent the method from proper search refinement and affect its efficiency and robustness. This issue is numerically confirmed by the quality of the results in Table 18. Figure 20 shows that the proposed modifications of ICA via EICA have significantly changed (improved) its behavior by making a balance between exploration and exploitation as a key factor in the performance of a meta-heuristic algorithm.

7. Conclusion

The present work proposed a hybrid method that was

Table 18. Optimal section indices and structural weights obtained for the helipad design problem.

Design variable	PSO	TLBO	ICA	EICA
X1	20	19	14	16
X2	25	21	25	22
X3	16	29	28	27
X4	15	13	12	13
X5	23	20	22	21
X6	6	4	1	1
X7	23	23	18	19
X8	10	14	10	13
X9	30	33	32	35
Best weight (kg)	27314.7	24748.5	24505.6	24459.8
Mean	28960.5	25513.7	24901.4	24897.2
SD	1155.4	564.3	321.5	490.2
NFE	10000	10000	10000	10000
(NFEB)	(950)	(6000)	(8700)	(6700)

formed by adding a walking phase to the standard Imperialist Competitive Algorithm (ICA) together with greedy replacement strategy. Exploitation capability of Enhanced Imperialist Competitive Algorithm (EICA) was then evaluated by treating unimodal test functions. EICA could capture true optimum with competitive efficiency using well-known optimization methods including Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), Colliding Bodies Optimization (CBO), Teaching-Learning-Based Optimization (TLBO), Symbiotic Organisms Search (SOS), and standard ICA. The exploration capability was, however, tested by multimodal functions with several local optima. In a majority of test functions, the proposed EICA outperformed such optimization methods in capturing global optima, even with a lower computational effort. Particularly, the performance of EICA was quite better than ICA, which confirmed the effectiveness of the proposed hybridization technique.

In order to validate the function of the proposed method in constrained optimization, it was tested on a number of engineering benchmarks and discrete structural sizing problems. By starting from an identical initial population in each of the independent runs, EICA achieved competitive or superior final results compared to other well-known optimizers.

A comparison between the final results of the present work and those already reported in the literature confirmed the capability of EICA in capturing high quality optima in discrete problems. In some cases, EICA required higher NFE than literature works to achieve better standard deviation; however; comparing the quality of results shows EICA superiority.

Furthermore, EICA had a desired trend of Diversity Index (DI) variation, i.e., high diversification, in early iterations followed by great intensification as the search progresses to the end. This is while ICA keeps fluctuating about a higher value. This issue provides reasoning for better search refinement of EICA than that of ICA. Lower standard deviation of EICA than ICA showed its better stability over several independent runs. The effectiveness of EICA in global search was also evaluated in a large-scale example and a real-word constrained problem. In other words, the proposed enhanced algorithm could better overpass local optima traps to capture global optimum.

Numerical tests confirmed the theoretical expectation of performance improvement of ICA via the proposed algorithm. In the light of our experiments, it is concluded that the proposed hybrid strategy in EICA has considerably enhanced the performance of ICA and changed its behavior in making a balance between exploitation and exploration. EICA was utilized with the least number of parameters, which is of practical interest in structural sizing and engineering optimization problems. The application of the proposed method to other types of problems, such as building frames, under dynamic loading will be a future research direction.

References

1. Wolpert, D.H. and Macready, W.G. "No free lunch theorems for optimization", *IEEE Trans. Evol. Comput.*, **1**, pp. 67–82 (1997). DOI: 10.1109/4235.585893

2. Binitha, S. and Sathya, S.S. "A survey of bio inspired

optimization algorithms", *Int. J. Soft Comput. Eng.*, **2**, pp. 137–151 (2012).

3. Kaveh, A. and Ilchi Ghazaan, M., *Meta-Heuristic Algorithms for Optimal Design of Real-Size Structures*, Springer International Publishing (2018).

4. Atashpaz-Gargari, E. and Lucas, C. "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition", *2007 IEEE Congr. Evol. Comput. CEC 2007.*, pp. 4661–4667 (2007). DOI: 10.1109/CEC.2007.4425083

5. Hosseini, S. and Al Khaled, A. "A survey on the imperialist competitive algorithm metaheuristic: Implementation in engineering domain and directions for future research", *Appl. Soft Comput. J.*, **24**, pp. 1078–1094 (2014). DOI: 10.1016/j.asoc.2014.08.024

6. Lin, J.-L., Tsai, Y.-H., Yu, C.-Y., and Li, M.-S. "Interaction enhanced imperialist competitive algorithms", *Algorithms*, **5**, pp. 433–448 (2012).

7. Talatahari, S., Kaveh, A., and Sheikholeslami, R. "Chaotic imperialist competitive algorithm for optimum design of truss structures", *Struct. Multidiscip. Optim.*, **46**, pp. 355–367 (2012). DOI: 10.1007/s00158-011-0754-4

8. Bahrami, H., Faez, K., and Abdechiri, M. "Imperialist competitive algorithm using chaos theory for optimization: (CICA). UKSim2010 - UKSim 12th", *Int. Conf. Comput. Model. Simul.*, pp. 98–103 (2010). DOI: 10.1109/UKSIM.2010.26

9. Atashpaz-Gargari, E. "Imperialist competitive algorithm (ICA)", http://www.mathworks.com/matlabcentral/ fileexchange/22046-imperialist- competitive-algorithm-ica (accessed on 10 January 2012).

10. Duan, H.B., Xu, C.F., Liu, S.Q., and Shao, S. "Template matching using chaotic imperialist competitive algorithm", *Pattern Recognit. Lett.*, **31**, pp. 1868–1875 (2010).

11. Jain, T. and Nigam, M.J. "Synergy of evolutionary algorithm and socio-political process for global optimization", *Expert Syst. Appl.*, **37**, pp. 3706–3713 (2010). DOI: 10.1016/j.eswa.2009.11.055

12. Khorani, V., Forouzideh, N., and Nasrabadi, A.M. "Artificial neural network weights optimization using ICA, GA, ICA-GA and R-ICA-GA: Comparing performances", *IEEE SSCI 2011 - Symp. Ser. Comput. Intell. - HIMA 2011 2011 IEEE Work. HybridIntell. Model. Appl.*, pp. 61–67 (2011). DOI: 10.1109/HIMA.2011.5953956

13. Karimi, N., Zandieh, M., and Najafi, A.A. "Group scheduling in flexible flow shops: A hybridised approach of imperialist competitive algorithm and electromagnetic-like mechanism", *Int. J. Prod. Res.*, **49**, pp. 4965–4977 (2011). DOI: 10.1080/00207543.2010.481644

14. Blum, C., Puchinger, J., Raidl, G.R., and Roli, A. "A brief survey on hybrid metaheuristics", *Bioinspired Optim. Methods Their Appl. - Proc. 4th Int. Conf. Bioinspired Optim. Methods Their Appl. BIOMA 2010.*, pp. 3–16 (2010). DOI: 10.1007/978-3-540-78295-7_1

15. Ghaffarzadeh, H. and Raeisi, F. "Damage identification in truss structures using finite element model updating and imperialist competitive algorithm", *Jordan Journal of Civil Engineering*, **10**, pp. 266–277 (2016).

16. Eskandar, H., Salehi, P., and Sabour, M.H. "Imperialist competitive ant colony algorithm for truss structures", *Appl. Sci.*, **12**, pp. 94–105 (2011).

17. Talatahari, S., Kaveh, A., and Sheikholeslami, R. "Chaotic imperialist competitive algorithm for optimum design of truss structures", *Struct. Multidiscip. Optim.*, **46**, pp. 355–367 (2012). DOI: 10.1007/s00158-011-0754-4

18. Mohammadi-ivatloo, B., Rabiee, A., Soroudi, A., and Ehsan, M. "Imperialist competitive algorithm for solving non-convex dynamic economic power dispatch", *Energy.*, **44**, pp. 228–240 (2012). DOI: 10.1016/j.energy.2012.06.034

19. Shaban, H., Vahidi, B., and Ebrahimpour, M. "A robust PID controller based on imperialist competitive algorithm for load-frequency control of power systems", *ISA Trans.*, **52**, pp. 88–95 (2012).

20. Roozbeh Nia, A., Hemmati Far, M., and Akhavan Niaki, S.T. "A hybrid genetic and imperialist competitive algorithm for green vendor managed inventory of multi-item multi-constraint EOQ model under shortage", *Appl. Soft Comput.*, **30**, pp. 353–634 (2015).

21. Hajihassani, M., Armaghani, D.J., Marto, A., and Tonnizam, E.M. "Ground vibration prediction in quarry blasting through an artificial neural network optimized by imperialist competitive algorithm", *Bull Eng Geol Env.*, **74**, pp. 873–886 (2014). DOI: 10.1007/s10064-014-0657-x

22. Kashani, A.R., Gandomi, A.H., and Mousavi, M. "Imperialistic competitive algorithm: A metaheuristic algorithm for locating the critical slip surface in 2-dimensional soil slopes", *Geosci. Front.*, **7**, pp. 83–89 (2016).

23. Ali, E.S. "Speed control of induction motor supplied by wind turbine via imperialist competitive algorithm", *Energy*, **89**, pp. 593–600 (2015).

24. Lin, J.L., Cho, C.W., and Chuan, H.C. "Imperialist competitive algorithms with perturbed moves for global optimization", *Appl. Mech. Mater.*, **284–287**, pp. 3135–3139 (2013). DOI: 10.4028/www.scientific.net/AMM.284-287.3135

25. Jamil, M. and Yang, X.-S. "A literature survey of benchmark functions for global optimization", *Int. J. Math. Model. Numer. Optim.*, **4**, pp. 150–194 (2013). DOI: 10.1504/IJMMNO.2013.055204

26. Arora, J.S., *Introduction to Optimum Design*, Elsevier Academic Press, London, UK (2004)

27. Kaveh, A. and Khayatazad, M. "A new meta-heuristic method: Ray optimization", *Comput. Struct.*, **112–113**, pp. 283–294 (2012). DOI: 10.1016/j.compstruc.2012.09.003

28. Mezura-Montes, E. and Coello, C.A.C. "An empirical study about the usefulness of evolution strategies to solve constrained optimization problems", *Int. J. Gen. Syst.*, **37**, pp. 443–473 (2008). DOI: 10.1080/03081070701303470

29. Kaveh, A. and Zolghadr, A. "A novel meta-heuristic algorithm: Tug of war optimization", *Int J. Optim. Civil Eng.*, **6**, pp. 469–492 (2016).

30. Sadollah, A., Bahreininejad, A., Eskandar, H., and Hamdi, M. "Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems", *Appl. Soft Comput.*, **13**, pp. 2592–2612 (2013). DOI: 10.1016/j.asoc.2012.11.026

31. He, Q. and Wang, L. "A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization", *Appl. Math. Comput.*, **186**, pp. 1407–1422 (2007). DOI: 10.1016/j.amc.2006.07.134

32. Mirjalili, S. and Lewis, A. "The whale optimization algorithm", *Adv. Eng. Softw.*, **95**, pp. 51–67 (2016). DOI: 10.1016/j.advengsoft.2016.01.008

33. Rashedi, E., Nezamabadi-pour, H., and Saryazdi, S. "GSA: A gravitational search algorithm", *Inf. Sci. (Ny)*, **179**, pp. 2232–2248 (2009). DOI: 10.1016/j.ins.2009.03.004

34. Kannan, B.K. and Kramer, S.N. "An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design" *ASME J. Mech. Des.*, **1**, pp. 405–411 (1994).

35. Li, M.D., Zhao, H., Weng, X.W., and Han, T. "A novel nature-inspired algorithm for optimization: Virus colony search", *Adv. Eng. Softw.*, **92**, pp. 65–88 (2016). DOI: 10.1016/j.advengsoft.2015.11.004

36. Mirjalili, S., Mirjalili, S.M., and Lewis, A. "Grey wolf optimizer", *Adv. Eng. Softw.*, **69**, pp. 46–61 (2014). DOI: 10.1016/j.advengsoft.2013.12.007

37. Gold, S. and Krishnamurty, S. "Tradeoffs in robust engineering design", In: *1997 ASME Design Engineering Technical Conferences* (1997).

38. Gandomi, A.H., Yang, X.S., and Alavi, A.H. "Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems", *Eng. Comput.*, **29**, pp. 17–35 (2013). DOI: 10.1007/s00366-011-0241-y

39. Wang, G.G. "Adaptive response surface method using inherited latin hypercube design points", *J. Mech. Des.*, **125**, pp. 210–220 (2003). DOI: 10.1115/1.1561044

40. Shahrouzi, M. and Kaveh, A. "Dynamic fuzzy-membership Optimization: an enhanced meta-heuristic search", *Asian J. Civ. Eng.*, **16**, pp. 249–268 (2015).

41. Rajeeve, S. and Krishnamurty, S. "Discrete optimization of structures using genetic algorithm", *ASCE J Struct Eng.*, **118**, pp. 1123–1250 (1992).

42. Li, L.J., Huang, Z.B., and Liu, F. "A heuristic particle swarm optimization method for truss structures with discrete variables", *Comput. Struct.*, **87**, pp. 435–443 (2009). DOI: 10.1016/j.compstruc.2009.01.004

43. Sadollah, A., Bahreininejad, A., Eskandar, H., and Hamdi, M. "Mine blast algorithm for optimization of truss structures with discrete variables", *Comput. Struct.*, **102–103**, pp. 49–63 (2012). DOI: 10.1016/j.compstruc.2012.03.013

44. Hasançebi, O., Çarbaş, S., Dogan, E., Erdal, F., and Saka, M.P. "Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures", *Comput. Struct.*, **87**, pp. 284–302 (2009). DOI: 10.1016/j.compstruc.2009.01.002

45. Kaveh, A. and Talatahari, S. "A particle swarm ant colony optimization for truss structures with discrete variables", *J. Constr. Steel Res.*, **65**, pp. 1558–1568 (2009). DOI: 10.1016/j.jcsr.2009.04.021

46. Kaveh, A. and Mahdavi, V.R. "Colliding bodies optimization: A novel meta-heuristic method", *Comput. Struct.*, **139**, pp. 18–27 (2014). DOI: 10.1016/j.compstruc.2014.04.005

47. Shahrouzi, M., Aghabagloua, M., and Rafiee, F. "Observer-teacher-learner-based optimization: An enhanced meta-heuristic for structural sizing design", *Struct. Eng. Mech.*, **62**, pp. 537–550 (2017). DOI: 10.12989/sem.2017.62.5.537

48. AISC. American Construction Institutes of Steel, *Manual of steel construction. Allowable stress design*, Chicago, Illinois (1989)

Appendix

Tension/compression spring design problem

$$\min \; f(x) = (x_3 + 2)x_2x_1^2,$$

subject to:

$$g_1(x) = 1 - \left(\frac{x_2^3x_3}{71.785x_1^4}\right) \leq 0,$$

$$g_2(x) = 4x_2^2 - \frac{x_1x_2}{12.566(x?_1^3x_2 - x_1^4)} + \left(\frac{1}{5108x_1^2}\right)$$

$$-1 \leq 0,$$

$$g_3(x) = 1 - \left(\frac{140.45x_1}{x_2^2x_3}\right) \leq 0,$$

$$g_4(x) = 1 - \left(\frac{x_1 + x_2}{1.5}\right) \leq 0,$$

$$0.05 \le x_1 \le 2.00,$$

$$0.25 \le x_2 \le 1.30,$$

$$2.00 \le x_3 \le 15.00.$$

Pressure vessel design problem

$$\min\ f(x) = 0.6224x_1x_3x_4 + 1.7781x_2{x_3}^2 + 3.1661x_4{x_1}^2 + 19.84x_3{x_1}^2,$$

subject to:

$$g_1(x) = -x_1 + 0.0193x_3 \le 0,$$

$$g_2(x) = -x_2 + 0.00954x_3 \le 0,$$

$$g_3(x) = -\pi {x_3}^2 x_4 - \frac{4}{3}\pi {x_3}^3 + 1296000 \le 0,$$

$$g_4(x) = x_4 - 240 \le 0,$$

$$0 \le x_i \le 100 \qquad i = 1, 2,$$

$$10 \le x_i \le 200 \qquad i = 3, 4.$$

I-beam design problem

$$\min\ f(b, h, t_w, t_f) = \frac{5000}{\frac{t_w(h-2t_f)^3}{12} + \frac{b{t_f}^3}{6} + 2bt_f\left(\frac{h-t_f}{2}\right)^2}.$$

subject to:

$$g_1 = b.t_w + t_w(h - 2t_f) \le 300,$$

$$g_2 = \frac{18h \times 10^4}{t_w(h-2t_f)^3 + 2bt_w(4{t_f}^2 + 3h(h-2t_f))} + \frac{15b \times 10^3}{(h-2t_f){t_w}^3 + 2t_wb^3} \le 6,$$

$$10 \le b \le 50, \qquad 10 \le h \le 80,$$

$$0.9 \le t_w \le 5, \qquad 0.9 \le t_f \le 5.$$

Biographies

Mohsen Shahrouzi received BS in Civil Engineering and MS in Earthquake Engineering from Sharif University of Technology in 1997 and 2000, respectively. He continued his studies at International Institute of Earthquake Engineering and Seismology until he graduated with a PhD degree in 2006. Dr. Shahrouzi is a faculty member at the Civil Engineering Department of Kharazmi University and has already been the author of over 100 research articles including 45 journal papers. His current research interests include stochastic search and applied graph theory in optimization of civil engineering problems and earthquake-resistant design of structures.

Alireza Salehi has been a PhD candidate since 2013 at Kharazmi University in Iran at the Faculty of Engineering. He received his MS degree in School of Civil Engineering from Tehran University in 2013. His current research interests include optimization, metaheuristic algorithms, and application of soft computing in structural problems.