Decision Analytics Journal 5 (2022) 100129

Contents lists available at ScienceDirect

Decision Analytics Journal

journal homepage: www.elsevier.com/locate/dajour

A swarm optimizer with modified feasible-based mechanism for optimum structure in steel industry

B. Nouhi [a], Y. Jahani [b], S. Talatahari [c,*], A.H. Gandomi [c]

[a] Department of Mathematical Sciences, University of Tabriz, Tabriz, Iran
[b] Analysis and Advanced Materials for Structural Design (AMADE), Polytechnic School, University of Girona, Girona, Spain
[c] Faculty of Engineering and IT, University of Technology Sydney, Australia

ARTICLE INFO

Keywords:
Swarm optimizer
Big bang–big crunch optimization
Optimum design
Modified feasible-based mechanism
Steel industry

ABSTRACT

This study proposes a swarm optimizer with a modified feasible-based mechanism approach for finding an optimum design for steel frames. The proposed optimization approach addresses the problem of stagnation possibility in the traditional particle swarm optimization in which none of the particles tries to explore a position better than the previous best position for multiple numbers of iterations. This method is based on accelerated particle swarm optimization and big bang–big crunch optimization algorithms. In addition, a modified feasible-based mechanism is used to correct the particle's position. The new method's performance is evaluated by solving two structural problems to minimize the weight of steel frames. The results show that the optimized designs obtained by the proposed algorithm are better than those found by the competing algorithms from the literature.

1. Introduction

The main aim of structural optimization is to reduce the weight of the structures and at the same time have a safe design. To this end, researchers present plenty of methods to optimize the structures. These methods are categorized into two groups: deterministic and probabilistic methods, which are based on mathematical programming and stochastic ideas, respectively. Many engineering design problems are too complex to be handled with mathematical programming methods. Therefore, for such cases, nature-inspired or meta-heuristic search methods can be useful. Nature-inspired methods are those in which "the computational algorithms model natural phenomena" [1]. Unlike mathematical optimization, meta-heuristic search methods do not require the data as in the conventional mathematical programming and they have better global search abilities than the classical optimization algorithms [2–4].

In the past few decades, many meta-heuristic methods have been developed [5–13] and applied for the optimum design of structures. Pezeshk et al. [14] performed the optimal design of plane steel frames using the genetic algorithms (GA) and later in the other studies, it has been utilized to design steel frame structures [15–17]. Kameshki and Saka [18] found optimum designs of plane steel frames with semi-rigid connections using a GA-based method and a geometrically nonlinear analysis. Moreover, Saka [19] used a harmony search (HS) algorithm in order to design the sway frames. Camp et al. [20] and Kaveh et al. [21] used the ant colony optimization (ACO) for the optimum design of steel frame structures. Kaveh and Talatahari presented different optimization methods to optimize the skeletal structures [22–25]. In these studies, an improved ACO (IACO) [22], imperialist competitive algorithm (ICA) [23], hybrid big bang–big crunch (HBB–BC) [24], and charge system search (CSS) algorithm [25] were presented and validated. In the other two studies, Aydoğdu et al. [26,27] found optimum designs of space steel frames with a firefly-based algorithm (FA) and artificial bee colony (ABC) algorithm. Degertekin [28] utilized the HS algorithm for the optimum design of steel frames. Furthermore, Toğan [29] utilized the teaching–learning-based optimization (TLBO) to design planner steel frames. In the other study, Kaveh and Talatahari [30] presented the hybrid harmony particle swarm ant colony (HPSACO) methodology to find an optimum design for different types of structures. In addition, Kaveh and Zakian [31] utilized CSS and HS algorithms for the design of steel frames. In the other study, Talatahari et al. [32] combined the eagle strategy algorithm with differential evolution (ES-DE) for optimum design of the frame structures. A more comprehensive review of meta-heuristic methods in frame design optimization can be found at [33,34].

Finding optimum design of structures, especially large-scale ones, is one of challenging problems in the field of engineering. The reason is due to large-number of variables which results a large-scale search space in on hand and difficulty of analyzing and controlling the high number of nonlinear constraints on the other hand. To fulfill handle this problem, one way is to introduce more efficient methods to reduce

* Corresponding author.
E-mail address: siamak.talat@gmail.com (S. Talatahari).

https://doi.org/10.1016/j.dajour.2022.100129
Received 21 April 2022; Received in revised form 11 September 2022; Accepted 11 September 2022
Available online 14 September 2022

the required computational cost. There are limited works that address this challenging problem, such as [35–37]. This aim is covered in this paper by presenting Developed Swarm Optimizer (DSO) [10] and Feasible-based mechanism as advanced methods. DSO is based on the accelerated particle swarm optimizer (APSO) and big bang–big crunch optimization (BB–BC) optimization algorithm. In this paper, the DSO method is adapted for solving two frame structures and compared with other algorithms. Furthermore, a modified feasible-based mechanism is utilized to correct the particle's position. The results show that the proposed method has a better result when compared to those from the literature.

2. Formulation of optimum design of steel frames according to AISC-LRFD

The purpose of size optimization of frame structures is to minimize the weight of the structure, *W*, through finding the optimal sections of members, in which all constraints exerted on the problem must be satisfied, simultaneously. Thus, the optimal design of frame structures can be formulated as:

Find:

$$X = \left[x_1, x_2, x_3, \dots, x_n\right] \tag{1}$$

To minimize:

$$W(X) = \sum_{i=1}^{nm} \Upsilon_i . x_i . L_i \tag{2}$$

where x_i, L_i and L_i are the area, material density and length of the steel section selected for member group *i*, respectively. Here, the objective of finding the minimum weight structure is subjected to several design constraints, including strength and serviceability requirements [38], as:

Displacement constraint:

$$\upsilon_i^d = \left|\frac{\delta_i}{\bar{\delta}_i}\right| - 1 \le 0 \quad i = 1, 2, \dots, nn \tag{3}$$

Shear constraint, for both major and minor axis:

$$\upsilon_i^s = \frac{V_u}{\phi_\upsilon V_n} - 1 \le 0 \quad i = 1, 2, \dots, nm \tag{4}$$

Constraints corresponding to the interaction of flexure and axial force are as follows:

$$\upsilon_i^I = \left\{ \begin{array}{l} \dfrac{P_u}{\phi_c P_n} + \dfrac{8}{9}\left(\dfrac{M_{ux}}{\phi_b M_{nx}} + \dfrac{M_{uy}}{\phi_b M_{ny}}\right) - 1 \le 0 \ for \ \dfrac{P_{uJ}}{\phi_c P_n} \ge 0.2 \\ \dfrac{P_u}{2\phi_c P_n} + \left(\dfrac{M_{ux}}{\phi_b M_{nx}} + \dfrac{M_{uy}}{\phi_b M_{ny}}\right) - 1 \le 0 \ for \ \dfrac{P_{uJ}}{\phi_c P_n} < 0.2 \end{array} \right\} \tag{5}$$

$i = 1, 2, \dots, nm$

where *nn* is the number of nodes; δ_i, $\bar{\delta}_i$ are the displacement of the joints and the allowable displacement, respectively; *nm* is the number of members; V_u is the required shear strength; V_n is the nominal shear strength which is defined by the equations in Chapter G of the LRFD Specification [38]; ϕ_υ is the shear resistance factor $\phi_\upsilon = 0.9$; P_u is the required strength (tension or compression); P_n is the nominal axial strength (tension or compression); ϕ_c is the resistance factor ($\phi_c = 0.9$ for tension, $\phi_c = 0.85$ for compression); M_u is the required flexural strength; i.e., the moment due to the total factored load (Subscript *x* or *y* denotes the axis about which bending occurs.); M_n is the nominal flexural strength determined in accordance with the appropriate equations in Chapter F of the LRFD Specification [38] and ϕ_b is the flexural resistance reduction factor ($\phi_b = 0.9$) according to AISC-LRFD [38].

2.1. Nominal strengths

Based on AISC-LRFD [38] specification, the nominal tensile strength of a member, based on yielding in the gross section, is equal to:

$$P_u = F_y A_g \tag{6}$$

where F_y is the member's specified yield stress and A_g is the gross section of the member. The nominal compressive strength of a member is the smallest value obtained from the limit states of flexural buckling, torsional buckling, and flexural–torsional buckling. For members with compact and/or non-compact elements, the nominal compressive strength of the member for the limit state of flexural buckling is as follows:

$$P_n = F_{cr} A_g \tag{7}$$

where F_{cr} is the critical stress based on flexural buckling of the member, calculated as:

$$for \ \lambda_c = \frac{Kl}{r\pi}\sqrt{\frac{F_y}{E}} \le 1.5 \quad F_{cr} = \left(0.658^{\lambda_c^2}\right) F_y \tag{8}$$

$$for \ \lambda_c = \frac{Kl}{r\pi}\sqrt{\frac{F_y}{E}} > 1.5 \quad F_{cr} = \left[\frac{0.877}{\lambda_c^2}\right] F_y \tag{9}$$

where l is the laterally unbraced length of the member, K is the effective length factor, r is the governing radius of gyration about the axis of buckling and E is the modulus of elasticity.

2.2. Effective length factor *k*

In order to calculate the nominal compressive strength, the effective length factor, *K*, should be determined for each member. This factor can be computed using the frame buckling monograph [38]. For sway frames, the effective length factor for columns is computed as follows:

$$\frac{\alpha^2 G_i G_j - 36}{6(G_i + G_j)} = \frac{\alpha}{\tan\alpha} \tag{10}$$

$$G_i = \frac{\sum I_{ci}/l_{ci}}{\sum I_{bi}/l_{bi}}, \qquad G_j = \frac{\sum I_{cj}/l_{cj}}{\sum I_{bj}/l_{bj}} \tag{11}$$

where $\alpha = \pi/K$, *i* and *j* subscripts correspond to end-*i* and end-*j* of the compression member, and subscripts *c* and *b*, in building structures, refer to columns and beams connecting to the joint under consideration, respectively. Parameters I and l in the above equations, represent the moment of inertia and unbraced length of the member, respectively.

3. A review of optimization algorithms

Since the utilized algorithm is based on the PSO and BB–BC algorithms, here a brief review of these algorithms is described in the following subsections and then in the next section, the DSO algorithm will be presented.

3.1. Particle swarm optimization

The PSO is based on a metaphor of social interaction, such as bird flocking and fish schooling, and is developed by Eberhart and Kennedy [8]. The PSO simulates a commonly observed social behavior, where members (particles) of a group (swarm) tend to follow the lead of the best of the group. In other words, the particles fly through the search space and their positions are updated based on the best positions of individual particles denoted by $\boldsymbol{P}_i^k$ and the best position among all particles in the search space represented by $\boldsymbol{P}_g^k$.

The procedure of the PSO is reviewed below:

- **Step 1**: *Initialization*. An array of particles and their associated velocities are initialized with random positions.
- **Step 2**: *Local and global best creation*. The initial particles are considered the first local best and the best of them corresponding to the minimum objective function will be the first global best.

- **Step 3**: *Solution construction*. The velocity and location of each particle are changed to the new position using the following equations:

$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{12}$$

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 \otimes \left(P_i^k - X_i^k\right) + c_2 r_2 \otimes \left(P_g^k - X_i^k\right) \tag{13}$$

where, X_i^k and V_i^k are the position and velocity for the ith particle at iteration k; ω is an inertia weight to control the influence of the previous velocity; r_1, and r_2 are two random numbers uniformly distributed in the range of (0, 1); c_1 and c_2 are two acceleration constants; P_i^k is the best position of the ith particle up to iteration k; P_g^k is the best position among all particles in the swarm up to iteration k and the sign "$\otimes$" denotes element-by-element multiplication.
- **Step 4**: *Local and global best updating*. The objective function of the particles is evaluated and thus P_i^k and P_g^k are updated if the new positions are better than the previous one.
- **Step 5**: Terminating criterion control. Steps 3 and 4 are repeated until a terminating criterion is satisfied.

The accelerated PSO (APSO) [39] is an improved variant of the standard PSO in which the velocity vector is updated as:

$$V_j^{k+1} = V_j^k + c_1 \times rn_j^k + c_2 \times \left(P_g^k - X_j^k\right) \tag{14}$$

where, rn_j^k is a random vector whose elements are normally distributed with zero mean and a unit standard deviation. Therefore, the new position vector in the APSO is written as:

$$X_j^{k+1} = (1 - c_2) \times X_j^k + c_1 \times rn_j^k + c_2 \times P_g^k \tag{15}$$

3.2. Big bang–big crunch algorithm

The BB–BC method developed by Erol and Eksin [9] consists of two phases: a big bang phase, and a big crunch phase. During the big bang phase, new solution candidates are randomly generated around a "center of mass", which is later calculated in the big crunch phase with respect to their fitness values. After the big bang phase, a contraction operation is applied during the big crunch. In this case, the contraction operator takes the current positions of each candidate solution in the population and its associated fitness function value and computes a center of mass.

The procedure of the BB–BC is reviewed below:

- **Step 1**: *Initialization*. Initial generation of candidates in a random manner in the search space (the first big bang).
- **Step 2**: *Individual best creation*. Calculate the fitness function values for all of the candidate solutions. The initial candidates are considered as the first individual best value to minimize the objective function.
- **Step 3**: *Finding the center of mass*. The center of mass is calculated by Eq. (16), (the big crunch phase):

$$X_c^k = \frac{\sum_{j=1}^{N} \frac{1}{f_j^k} X_j^k}{\sum_{j=1}^{N} \frac{1}{f_j^k}} \tag{16}$$

where X_j is the position of jth solution, f_j^k is a fitness function value of this point at the kth iteration, and N is the population size.
- **Step 4**: *Solution construction*. Calculate the new candidate fitness values around the center of mass and update the center of mass using Eq. (17), (second big bang):

$$X_j^{new} = X_c^k + rn_j^k \otimes \frac{\alpha \left(X^{\max} - X^{\min}\right)}{k+1} \tag{17}$$

where X_j^{new} is the new position of the jth candidate solution, $X^{\min}$ and $X^{\max}$ are the lower and upper bounds of the design variables, respectively; rn_j^k is a random vector from a standard normal distribution, and α is a parameter for limiting the size of the search space.
- **Step 5**: *Terminating criterion control*. Steps 2–4 are repeated until a terminating criterion is satisfied.

4. Developed swarm optimizer

The developed swarm optimizer (DSO) was recently developed by Sheikholeslami and Talatahari [10] to solve water network systems. Based on the fact that one of the important disadvantages of the PSO is its higher speed of convergence with a higher possibility of diversity loss which leads to an undesirable premature convergence, the DSO was proposed [9] in which the process of escaping from a local optimum is dealt with. In this algorithm, the modification in the PSO is conducted in which the previously defined center of mass in the BB–BC method is inserted in the position updating process of the PSO. The procedure of the DSO is summarized in the following steps:

- **Step 1**: *Initialization*. Initialize an array of particles with random positions.
- **Step 2**: Local best, *global best and center of mass creation*. Calculate the fitness function values for all of the candidate solutions. Local best, global best and center of mass are determined.
- **Step 3**: *Solution construction*. This step contains two phases:
Step 3.1: *Global searching*. Global searching of the DSO method is performed by adding the big crunch phase of the BB–BC algorithm into the APSO according to Eq. (18):

$$X_j^{k+1} = (1-c_2)\times X_j^k + c_1 \times rn_j^k + c_2 \times \left\{ r_{1j}^k \otimes P_g^k + (1 - r_{1j}^k) \otimes X_c^k \right\} \tag{18}$$

where, r_{1j}^k is a random vector uniformly distributed in the range of [0, 1]. Eq. (18) contains three parts: (i) part one represents the influence of the previous position towards the current position, (ii) part two makes the algorithm explore the whole search space effectively, and (iii) part three represents the cooperation among the particles in finding the global optimal solution.
Step 3.2: *Local searching*. In the local searching step, each particle generates a solution (Z_j^k) around the global best-center of mass points which can be calculated using a normal distribution:

$$Z_j^k = N\left(\left(r_{1j}^k \otimes P_g^k + (1 - r_{1j}^k) \otimes X_c^k\right), \sigma\right) \tag{19}$$

In order to account for the information received over time that reduces uncertainty about the global best position, σ in the kth iteration is modeled using a non-increasing function as:

$$\sigma = rn_j^k \otimes \frac{\alpha \left(X^{\max} - X^{\min}\right)}{k+1} \tag{20}$$

where rn_j^k is a random vector from a standard normal distribution, and α is a parameter for limiting the size of the search space.

- **Step 4**: *Constraint handling methods and fitness finding:* This step is performed in two phases, as:
Step 4.1: *Position correction*. For both solutions generated in global and local steps, if they move out of the search space, their positions are corrected using the harmony memory (HM) concept of the HS method.
Step 4.2: *Problem-specified constraint handling*. The modified feasible-based mechanism is performed as described in the next subsection.
- **Step 5**: *Update global best and center of mass positions*. The new best global and center of mass are updated and stored.
- **Step 6**: *Terminating criterion control*. Steps 2–5 are repeated until a terminating criterion is satisfied.

The flowchart of the DSO is shown in Fig. 1.

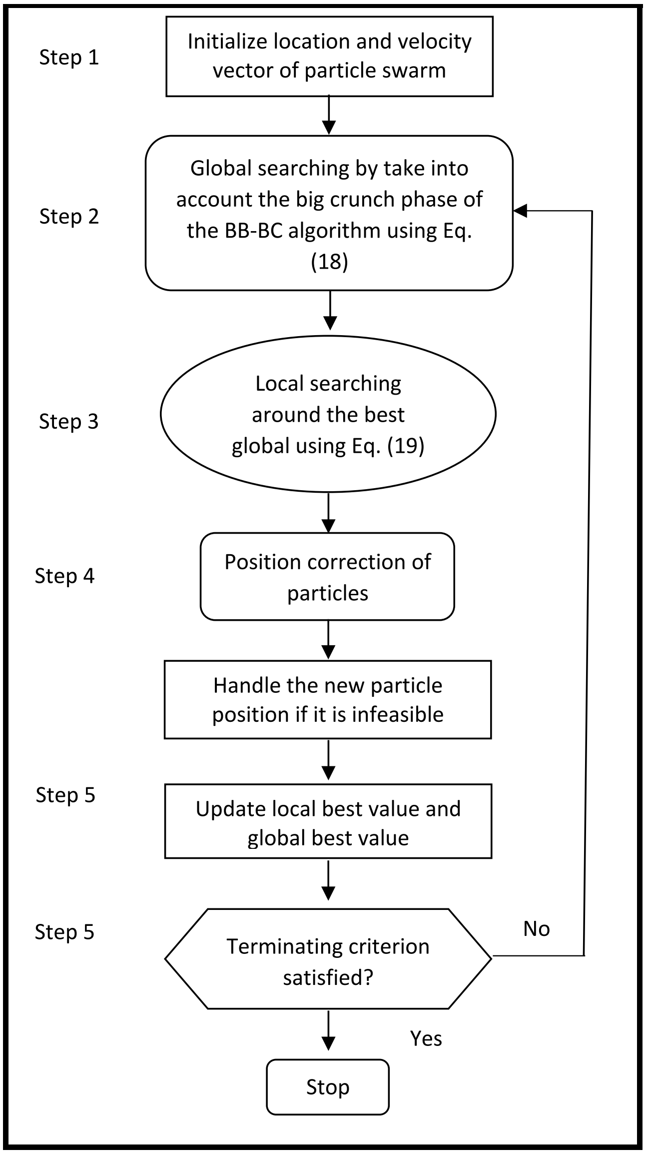

Fig. 1. Flow-chart of DSO algorithm.

4.1. A modified feasible-based mechanism added to the DSO

In the proposed DSO algorithm, a modified feasible-based mechanism (FBM) is also used to handle the problem-specific constraints. In the original FBM, also known as constraint tournament selection, pair-wise solutions are compared using the following rules:

- **Rule 1**: Any feasible solution is preferred to any infeasible solution.
- **Rule 2:** Between two feasible solutions, the one having a better objective function value is preferred.
- **Rule 3**: Between two infeasible solutions, the one having a smaller sum of constraint violation is preferred. This sum is calculated by:

$$Viol = \sum_{j=1}^{n_g} \max\left(0, g_i(\boldsymbol{X})\right) \quad (21)$$

where g_j is the jth inequality constraint, $\mathbf{X}$ is the set of decision variables, and n_g is the total number of inequality constraints.

By using the first and third rules, the search tends to the feasible region rather than the infeasible region, and the second rule persuades the search to remain in the feasible region with good solutions. In order to overcome to maintain diversity population problem, in the proposed DSO, an additional rule is added and defined as follows [10]:

- **Rule 4**: Infeasible solutions containing slight violations of the constraints (from 0.01 in the first iteration to 0.001 in the last iteration) are considered as feasible solutions.

By applying Rule 4, the particles can approach the boundaries and can move towards the global minimum with a high probability. Fig. 2 shows the flowchart of the modified feasible-based mechanism.

5. Numerical examples

In this section, the performance of the DSO algorithm is investigated by solving two real-size frame structures, containing:

- 135-member, 3-story, 3D frame
- 1026-member, 10-story, 3D frame

For these examples, the simple DSO [40], UBB–BC [41], UMBB–BC [41], UEBB–BC [41], UPSO [42], CSS [43] and ISA [44] were utilized before. In the DSO method, the BB–BC algorithm was combined with an accelerated PSO algorithm to improve the searching ability of the agents in the search space, therefore, the new method can find the minimum structural weight. Optimal results were compared with the literature to demonstrate the validity of the proposed approach. The optimization algorithms were coded in MATLAB while structural analysis was performed with the SAP2000 software. In this study, the total number of parameters is the same as its original variant of DSO. As a result, since the parameters of the original DSO was evaluated in Ref. [10], we here utilized the same values. It is worth to note that one may reach better performance for the presented method by tunning the parameters for these problems, however we aim to evaluate the abilities of the method without such time-consuming process. The details of the numerical examples and optimum results are summarized in the following subsections.

5.1. Example 1: design of a 135-member 3-story steel frame

This example contains 135 elements including 66 beams, 45 columns and 24 bracing members as indicated in Fig. 3. The geometry, load combination and other details of the example are taken from [43]. The material properties for this example are modulus of elasticity, E = 200 GPa, yield stress, F_y = 248.2 MPa, and unit weight of the steel, $\rho = 7.85$ ton/m^3. The stability of the structure is provided through moment-resisting connections as well as bracing systems (inverse V - type) along the x directions. The 135-member frame is placed into 10 member groups. The columns are grouped into four sizing variables in a plan level as corner, inner, side x–z and side y–z columns, and they are assumed to have the same cross-section over the three stories of the frame. The columns grouping in the plan level is illustrated in Fig. 4. All of the beams in each story are grouped into one sizing variable, resulting in three beam-sizing design variables for the frame. Similarly, all the bracings in each story are grouped into one sizing variable, resulting in three bracing-sizing design variables for the frame. The beam elements are continuously braced along their lengths by the floor system, and columns and bracings are assumed to be unbraced along their lengths. The effective length factor, K, is taken as 1 for all beams and bracings. The K factor is conservatively taken as 1.0 for buckling of columns about their minor (weak) direction, and for buckling of columns about their major direction, the K factor has been calculated from Section 2.2.

Optimization results obtained by the new method and the BB–BC-based ones as well as UPSO reported in the literature are summarized

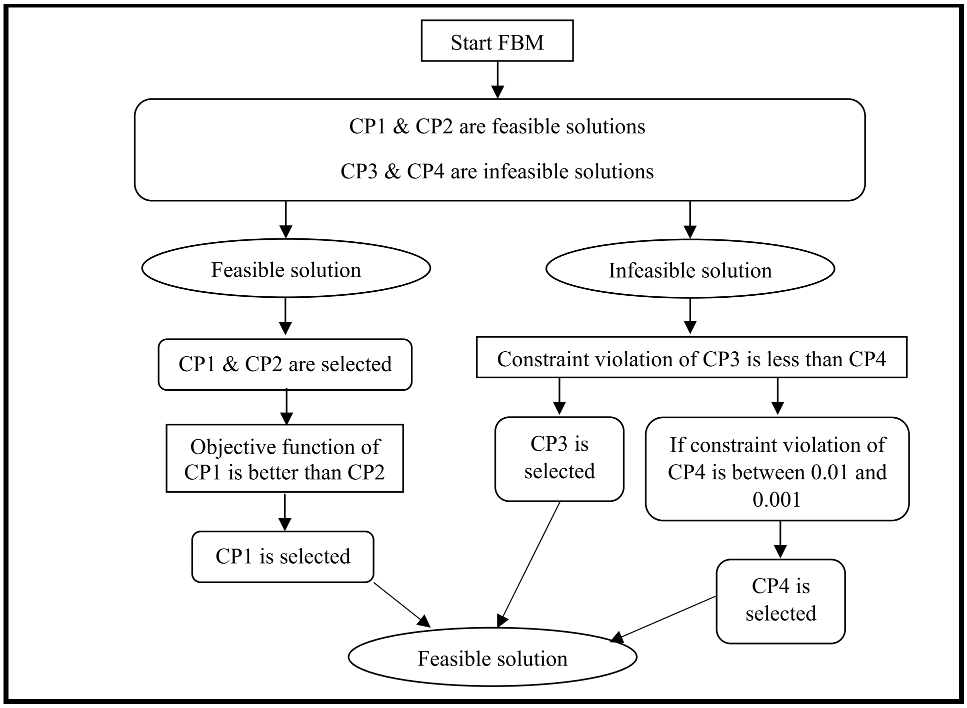

Fig. 2. Flow-chart of FBM.

Table 1
Optimum designs obtained for 135-member 3-story steel frame.

Element group	Optimal W-shape sections				
	UPSO [42]	UBB–BC [41]	UMBB–BC [1]	UEBB–BC [41]	Current work
CG1[a]	W8X28	W10X39	W30X90	W21X62	W16X40
CG2	W33X118	W27X84	W14X48	W14X48	W27X84
CG3	W40X167	W40X149	W40X215	W36X150	W24X76
CG4	W14X53	W18X65	W27X84	W21X68	W21X62
B1[a]	W14X30	W21X44	W14X34	W18X40	W16X36
B2	W24X55	W16X40	W12X35	W18X35	W21X44
B3	W16X26	W10X22	W18X35	W16X26	W14X22
BR1[a]	W14X30	W27X84	W21X44	W8X24	W6X25
BR2	W14X149	W16X26	W10X22	W16X26	W6X20
BR3	W27X84	W21X44	W6X15	W6X15	W6X15
Weight (ton)	55.66	47.3	45.67	38.91	38.18

[a]CG denotes column group with respect to Fig. 4, B: beams, BR: bracings.

in Table 1. This work found the best design overall corresponding to a structural weight of 38.18 tons. Optimized weights reported in the literature were heavier than the present study and equal to 55.66, 47.3, 45.67 and 38.91 tons for UPSO, UBB–BC, UMBB–BC and UEBB–BC, respectively. The DSO algorithm needs 1000 analyses to complete the optimization process which is almost equal to those of the UBB–BC, UMBB–BC and UEBB–BC i.e., 880, 1794 and 1235, respectively. It is clear the proposed DSO algorithm has a good performance compared to those other improved BB–BC-based algorithms.

5.2. Example 2: design of a 1026-member 10-story steel frame

The 10-story steel frame indicated in Fig. 5 is selected as the second example. The geometry, load combination and other details of the example are taken from [41]. The frame consists of 1026 structural members, including 580 beams, 350 columns and 96 bracing elements. The stability of the structure is provided through moment-resisting connections as well as bracing systems (X - type) along the x directions. For optimizing purposes, the 1026 members of the frame are placed under 32 member groups. The member grouping is considered in both plan and elevation levels. At elevation level, the structural members are grouped in every three stories except the first story. At the plan level, columns are considered in 5 different column groups as depicted in Fig. 6; beams are divided into outer and inner beams, and bracings are assumed to be in one group. Therefore, based on both elevation and plan level groupings, there are a total of 20 column groups, 8 beam groups, 4 bracing groups, and a total of 32 design variables. The unbraced lengths of all beam elements are set to one-fifth of their lengths and columns and bracings are assumed to be unbraced along their lengths. The effective length factor, K, for buckling of columns about their minor direction as well as beams and bracings is taken as 1, and for buckling of columns about their major direction, the K factor has been calculated from Section 2.2. The cross-sections of the elements are selected from 267 W-shape sections in the optimization processes.

The optimization results of the proposed method are compared with the ones reported in the literature in Table 2. The DSO found the optimum structural weight of 544.14 tons. Optimized weights reported in literature equal to 557.95, 634.12, 612.05, 584.93, 559.32 and 549.17 tons for the simple DSO [40], UBB–BC [41], UMBB–BC [41], UEBB–BC [41], CSS [43] and ISA [44], respectively. The DSO algorithm needs 21,000 analyses to complete the optimization process which equals the simple DSO [40]. It is clear the proposed DSO algorithm can find better results than those other algorithms in the literature. The

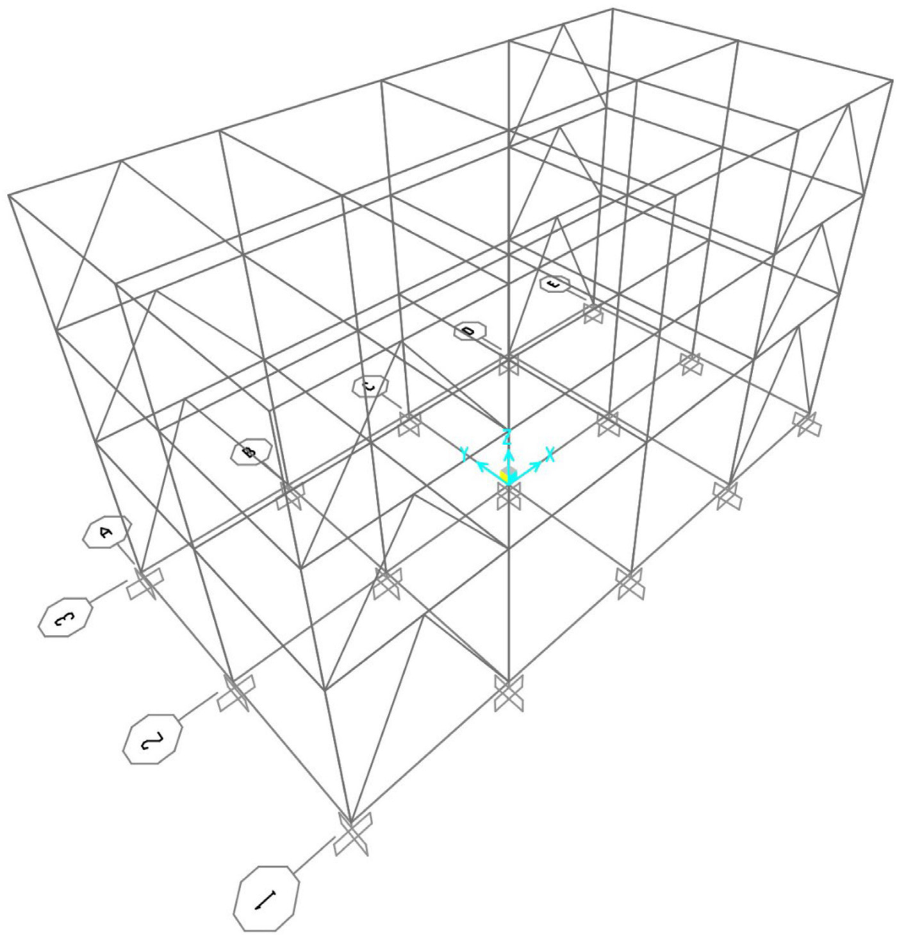

Fig. 3. 3D view of the 135-Member 3-story steel frame.

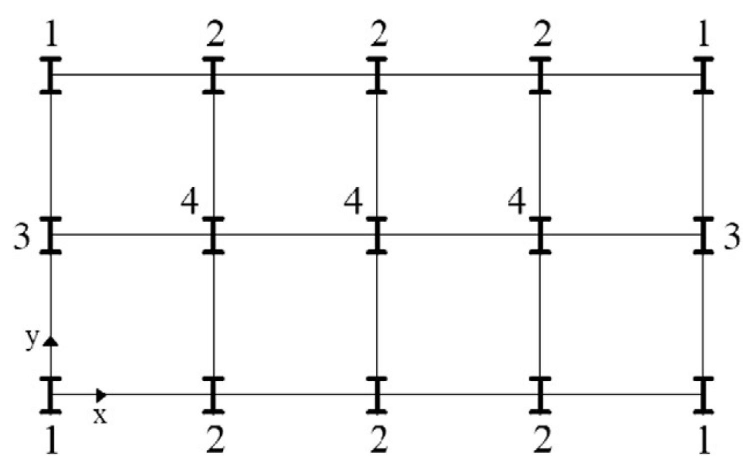

Fig. 4. Columns grouping of 135-member 3-story steel frame in plan level, [41].

convergence history of the proposed method for this example is shown in Fig. 7.

5.3. Statistical analysis

The statistical results of the optimum design procedure for the DSO and the proposed method based on 30 independent optimization runs are presented in Table 3. It is concluded that the proposed method is able to provide better results than the standard DSO method by considering the mean and standard deviation results.

6. Conclusions

This paper presents a developed swarm-based algorithm (DSO) with a modified feasible-based mechanism for the optimum design of the frame structures. The proposed DSO method is based on accelerated PSO and BB–BC optimization algorithms. For evaluating the robustness

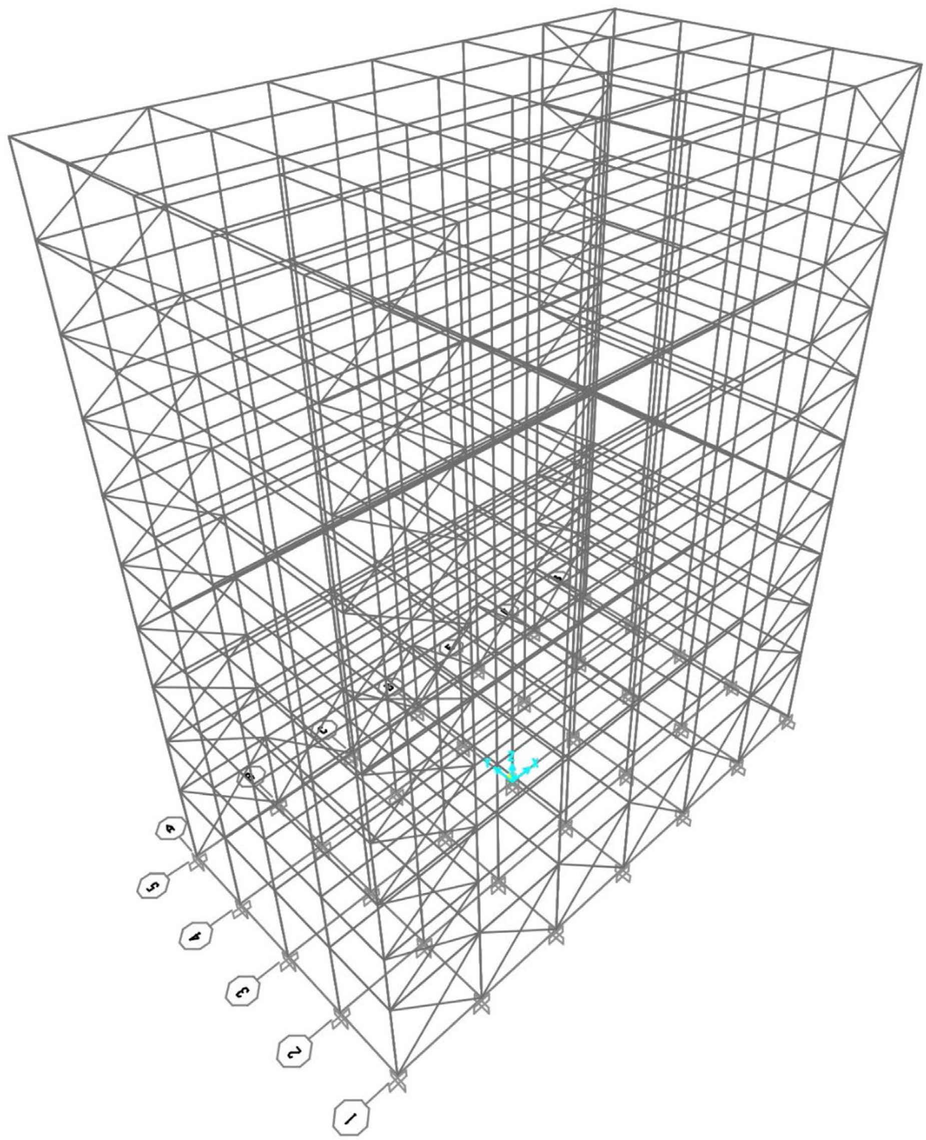

Fig. 5. 3-D view of the 1026-Member 10-story steel frame.

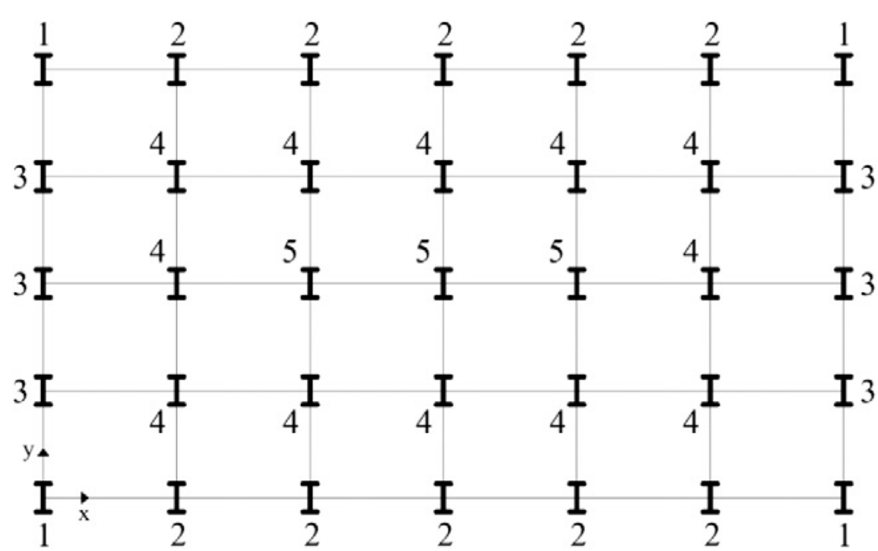

Fig. 6. Columns grouping of 1026-member 10-story steel frame in plan level, [41].

Table 2
Optimum designs obtained for 1026-member 10-story steel frame.

Stories	Groups	Optimal W-shape sections						
		UBB–BC [41]	UMBB–BC [41]	UEBB–BC [41]	CSS [43]	DSO(40)	ISA [44]	Current work
	CG1[a]	W27X258	W24X492	W33X201	W27X368	W40X211	W40X277	W36X194
	CG2	W27X161	W27X146	W24X146	W40X183	W12X96	W21X182	W33X141
	CG3	W27X102	W21X101	W24X104	W27X146	W33X201	W27X161	W21X147
1	CG4	W27X146	W27X161	W40X174	W40X149	W21X122	W33X201	W27X194
	CG5	W27X146	W27X258	W40X321	W12X152	W21X182	W12X120	W36X160
	IB[a]	W27X84	W21X44	W27X84	W10X33	W18X46	W16X26	W18X46
	OB[a]	W27X84	W27X84	W27X84	W16X40	W21X62	W24X76	W6X25
	BR[a]	W27X94	W30X90	W18X76	W12X30	W14X99	W8X21	W10X54
	CG1	W27X258	W21X201	W36X328	W40X297	W33X263	W44X230	W36X170
	CG2	W27X146	W24X162	W36X245	W30X148	W14X176	W24X131	W21X166
	CG3	W27X84	W24X131	W36X135	W40X149	W33X241	W33X118	W40X174
2–4	CG4	W27X102	W40X174	W33X118	W24X146	W36X135	W33X118	W24X104
	CG5	W27X114	W27X102	W44X262	W10X100	W21X111	W21X111	W30X132
	IB	W27X84	W27X84	W16X26	W27X102	W14X34	W24X76	W16X26
	OB	W27X84	W30X90	W36X135	W24X68	W33X141	W24X62	W40X167
	BR	W27X84	W40X149	W21X62	W10X60	W10X54	W12X72	W16X67
	CG1	W27X161	W40X235	W27X258	W27X129	W36X135	W30X173	W24X192
	CG2	W27X114	W24X131	W18X106	W14X159	W24X117	W36X170	W14X120
	CG3	W27X84	W30X90	W33X130	W30X108	W21X93	W14X109	W24X104
5–7	CG4	W27X84	W18X86	W27X94	W14X120	W27X94	W33X221	W24X146
	CG5	W30X99	W14X90	W24X192	W21X93	W14X82	W14X145	W16X67
	IB	W27X84	W21X44	W21X44	W21X73	W21X57	W30X99	W24X55
	OB	W27X84	W30X108	W21X73	W24X68	W24X84	W24X55	W21X83
	BR	W27X94	W33X118	W30X90	W10X49	W12X65	W16X31	W12X53
	CG1	W27X84	W36X194	W18X86	W21X44	W10X22	W12X26	W18X55
	CG2	W27X146	W27X146	W21X50	W14X109	W14X132	W14X132	W33X130
	CG3	W27X84	W40X174	W36X135	W10X68	W16X100	W33X141	W18X65
8–10	CG4	W27X84	W21X62	W33X201	W27X146	W30X191	W12X79	W14X109
	CG5	W27X84	W24X76	W30X108	W40X215	W27X146	W16X50	W14X311
	IB	W27X84	W14X30	W21X57	W16X45	W16X31	W14X26	W18X40
	OB	W27X84	W16X31	W16X26	W16X36	W16X67	W24X55	W21X62
	BR	W27X84	W33X118	W18X76	W8X31	W8X40	W14X43	W10X49
Weight (ton)		634.12	612.05	584.93	559.32	557.95	549.17	544.14

[a]CG denotes column group with respect to Fig. 6, IB: inner beams, OB: outer beams, BR: bracings.

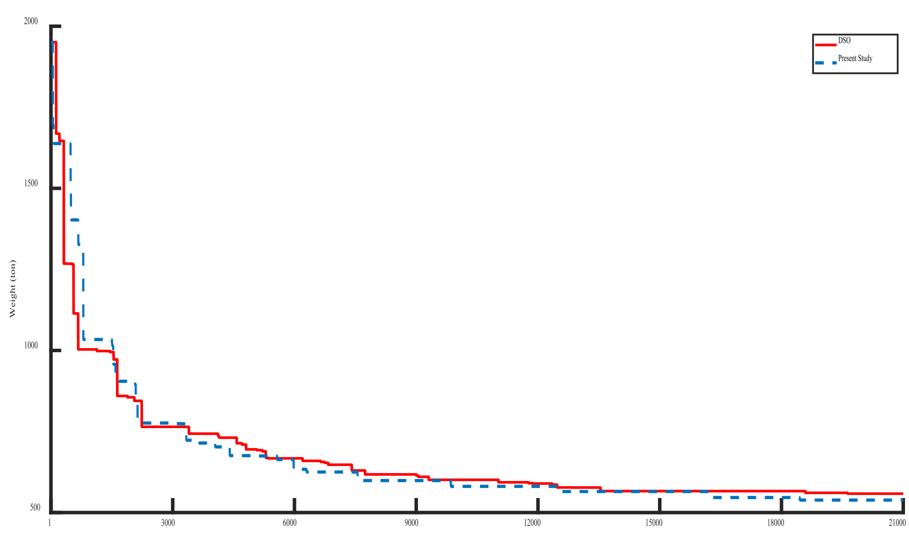

Fig. 7. Convergence history of the proposed method for 1026-member 10-story steel frame.

of the proposed method, two real-size structures were optimized and compared with other metaheuristic algorithms in the literature. The optimization algorithm was implemented by interfacing MATLAB with the SAP2000 structural analysis code. The results indicated that the proposed method had better result when compared to those algorithms in the literature and led to a lighter structure. As future works, more complicated structures can be considered as the optimization problem. In this way, the number of constraint and complexity of search space will increase and the requirement of advanced algorithms become clearer. Also, improving the present method to handle the structural problems with less computational cost is always interesting research.

Table 3
Statistical results for the DSO and the proposed methods based on 30 independent runs.

Algorithm	*Example*	*Best*	*Mean*	*Std.*
DSO	***3-Story***	*42.35*	*50.65*	*5.29*
	10-Story	*557.95*	*621.21*	*58.17*
Current work	***3-Story***	*38.18*	*43.95*	*3.26*
	10-Story	*544.14*	*582.35*	*35.36*

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] A.H. Gandomi, Interview by Marjan Eggermont, NASAVINE biomimicry summit for aerospace, Zygote Q. ZQ 18 (2017) 76–83.
[2] A.H. Gandomi, X.S. Yang, S. Talatahari, A.H. Alavi, Metaheuristic algorithms in modeling and optimization, in: Metaheuristic Applications in Structures and Infrastructures, Elsevier, 2013, pp. 1–24.
[3] H. Bayzidi, S. Talatahari, M. Saraee, C.P. Lamarche, Social network search for solving engineering optimization problems, Comput. Intell. Neurosci. (2021).
[4] M. Soltanifar, An investigation of the most common multi-objective optimization methods with propositions for improvement, Decis. Anal. J. 1 (2021) 100005.
[5] S. Talatahari, M. Azizi, M. Tolouei, B. Talatahari, P. Sareh, Crystal structure algorithm (CryStAl): A metaheuristic optimization method, IEEE Access 9 (2021) 71244–71261.
[6] S. Talatahari, M. Azizi, A.H. Gandomi, Material generation algorithm: A novel metaheuristic algorithm for optimization of engineering problems, Processes 9 (5) (2021) 859.
[7] A. Kaveh, S. Talatahari, A novel heuristic optimization method: Charged system search, Acta. Mech. 213 (2010) 267–289.
[8] R.C. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 1995.
[9] O.K. Erol, I. Eksin, A new optimization method: Big bang–big crunch, Adv. Eng. Softw. 37 (2006) 106–111.
[10] R. Sheikholeslami, S. Talatahari, Developed swarm optimizer: A new method for sizing optimization of water distribution systems, J. Comput. Civil. Eng. ASCE. 30 (5) (2016) 04016005.
[11] V. Goodarzimehr, S. Shojaee, S. Hamzehei-Javaran, S. Talatahari, Special relativity search: A novel metaheuristic method based on special relativity physics, Knowl.-Based Syst. (2022) 109484.
[12] B. Nouhi, N. Darabi, P. Sareh, H. Bayazidi, F. Darabi, S. Talatahari, The fusion–fission optimization (FuFiO) algorithm, Sci. Rep. 12 (1) (2022) 1–44.
[13] M. Azizi, S. Talatahari, A.H. Gandomi, Fire hawk optimizer: A novel metaheuristic algorithm, Artif. Intell. Rev. (2022) http://dx.doi.org/10.1007/s10462-022-10173-w.
[14] S. Pezeshk, C.V. Camp, D. Chen, Design of nonlinear framed structures using genetic optimization, J. Struct. Eng. 126 (2000) 382–388.
[15] M.S. Hayalioglu, S.O. Degertekin, Minimum cost design of steel frames with semi-rigid connections and column bases via genetic optimization, Comput. Struct. 83 (2005) 1849–1863.
[16] S. Rajeev, C.S. Krishnamoorthy, Discrete optimization of structures using genetic algorithms, J. Struct. Eng. 118 (1992) 1233–1250.
[17] M.S. Hayalioglu, Optimum design of geometrically non-linear elastic–plastic steel frames via genetic algorithm, Comput. Struct. 77 (2000) 527–538.
[18] E.S. Kameshki, M.P. Saka, Optimum design of nonlinear steel frames with semi-rigid connections using a genetic algorithm, Comput. Struct. 79 (2001) 1593–1604.
[19] M.P. Saka, Optimum design of steel sway frames to BS5950 using harmony search algorithm, J. Constr. Steel Res. 65 (2009) 36–43.
[20] C.V. Camp, J.B. Barron, P.S. Scott, Design of steel frames using ant colony optimization, J. Struct. Eng. 131 (2005) 369–379.
[21] A. Kaveh, B.F. Azar, A. Hadidi, F.R. Sorochi, S. Talatahari, Performance-based seismic design of steel frames using ant colony optimization, J. Constr. Steel Res. 66 (2010) 566–574.
[22] A. Kaveh, S. Talatahari, An improved ant colony optimization for the design of planar steel frames, Eng. Struct. 32 (2010) 864–873.
[23] A. Kaveh, S. Talatahari, Optimum design of skeletal structures using imperialist competitive algorithm, Comput. Struct. 88 (2010) 1220–1229.
[24] A. Kaveh, S. Talatahari, A discrete big bang-big crunch algorithm for optimal design of skeletal structures, Asian J. Civil. Eng. 11 (2010) 103–122.
[25] A. Kaveh, S. Talatahari, Charged system search for optimal design of frame structures, Appl. Soft Comput. 12 (2012) 382–393.
[26] İ. Aydoğdu, A. Akın, M.P. Saka, Discrete design optimization of space steel frames using the adaptive firefly algorithm, in: Proceedings of the Eleventh International Conference, 2012.
[27] İ. Aydoğdu, A. Akın, M.P. Saka, Design optimization of real world steel space frames using artificial bee colony algorithm with Levy flight distribution, Adv. Eng. Softw. 92 (2016) 1–14.
[28] S.O. Degertekin, Optimum design of steel frames using harmony search algorithm, Struct. Multidisc. Optim. 2008 36 (2008) 393–401.
[29] V. Toğan, Design of planar steel frames using teaching–learning based optimization, Eng. Struct. 34 (2012) 225–232.
[30] A. Kaveh, S. Talatahari, Hybrid algorithm of harmony search, particle swarm and ant colony for structural design optimization, in: Harmony Search Algorithms for Structural Design Optimization, Vol. 239, Springer, 2009, pp. 159–198.
[31] A. Kaveh, P. Zakian, Optimal design of steel frames under seismic loading using two meta-heuristic algorithms, J. Constr. Steel Res. 82 (2013) 111–130.
[32] S. Talatahari, A.H. Gandomi, X.S. Yang, S. Deb, Optimum design of frame structures using the eagle strategy with differential evolution, Eng. Struct. 91 (2015) 16–25.
[33] A.H. Gandomi, D. Roke, A multi-objective evolutionary framework for formulation of nonlinear structural systems, IEEE Trans. Indust. Inform. (2021).
[34] A.R. Kashani, CV. Camp, M. Rostamian, K. Azizi, A.H. Gandomi, Population-based optimization in structural engineering: A review, Artif. Intell. Rev. 55 (1) (2022) 345–452.
[35] M. Azizi, S. Talatahari, M. Basiri, M.B. Shishehgarkhaneh, Optimal design of low-and high-rise building structures by tribe-harmony search algorithm, Decis. Anal. J. (2022) 100067.
[36] N. Sadrekarimi, S. Talatahari, B.F. Azar, A.H. Gandomi, A surrogate merit function developed for structural weight optimization problems, Soft Comput. (2022) http://dx.doi.org/10.1007/s00500-022-07453-6.
[37] P. Motamedi, M. Banazadeh, S. Talatahari, Seismic loss optimum design of steel structures using learning-based charged system search, Struct. Design Tall Special Build. (2022) e1945.
[38] American Institute of Steel Construction (AISC), Steel Construction Manual, thirteenth ed., Chicago, Illinois, USA, 2005.
[39] S. Talatahari, E. Khalili, S.M. Alavizadeh, Accelerated particle swarm for optimum design of frame structures, Math. Probl. Eng. 2013 (2013).
[40] S. Talatahari, M. Azizi, Optimal design of real-size building structures using quantum-behaved developed swarm optimizer, Struct. Des. Tall Special Build. 29 (11) (2020) e1747.
[41] S.K. Azad, O. Hasancebi, S.K. Azad, Upper bound strategy for metaheuristic based design optimization of steel frames, Adv. Eng. Softw. 57 (2013) 19–32.
[42] S.K. Azad, O. Hasancebi, Improving computational efficiency of particle swarm optimization for optimal structural design, Int. J. Optim. Civil Eng. 3 (4) (2013) 563–574.
[43] S. Talatahari, M. Azizi, M. Toloo, M.B. Shishehgarkhaneh, Optimization of large-scale frame structures using fuzzy adaptive quantum inspired charged system search, Int. J. Steel Struct. (2022) 1–22.
[44] S. Talatahari, M. Azizi, Optimum design of building structures using tribe-interior search algorithm, Structures 28 (2020) 1616–1633.