Chapter 11
Imperialist Competitive Algorithm

11.1 Introduction

In this chapter an optimization method is presented based on a socio-politically motivated strategy, called Imperialist Competitive Algorithm (ICA). ICA is a multi-agent algorithm with each agent being a country, which is either a colony or an imperialist. These countries form some empires in the search space. Movement of the colonies toward their related imperialist, and imperialistic competition among the empires, form the basis of the ICA. During these movements, the powerful Imperialists are reinforced and the weak ones are weakened and gradually collapsed, directing the algorithm towards optimum points. Here, ICA is utilized to optimize the skeletal structures which is based on [1, 2].

This algorithm is proposed by Atashpaz et al. [3, 4] and is a socio-politically motivated optimization algorithm which similar to many other evolutionary algorithms starts with a random initial population. Each individual agent of an empire is called a *country*, and the countries are categorized into *colony* and *imperialist* states that collectively form *empires*. Imperialistic competitions among these empires form the basis of the ICA. During this competition, weak empires collapse and powerful ones take possession of their colonies. Imperialistic competitions direct the search process toward the powerful imperialist or the optimum points.

On the other hand, finding the optimum design of the skeletal structures is known as benchmark examples in the field of difficult optimization problems due to the presence of many design variables, large size of the search space, and many constraints. Thus, this chapter presents an ICA-based algorithm to solve optimization skeletal structures problems which can be considered as a suitable field to investigate the efficiency of the new algorithm. The chapter covers both the discrete and continuous structural design problems. Comparison of the results of the ICA with some well-known metaheuristics demonstrates the efficiency of the present algorithm.

A. Kaveh, *Advances in Metaheuristic Algorithms for Optimal Design of Structures*,
DOI 10.1007/978-3-319-05549-7_11,

11.2 Optimum Design of Skeletal Structures

The aim of optimizing a structure is to find a set of design variables that has the minimum weight satisfying certain constraints. This can be expressed as

$$\begin{aligned}&\text{Find} && \{x\} = \left[x_1, x_2, \ldots, x_{ng}\right], \\ & && x_i \in D_i \\ &\text{to minimize} && W\left(\{x\}\right) = \sum_{i=1}^{nm} \rho_i \cdot x_i \cdot L_i \\ &\text{subject to :} && g_j\left(\{x\}\right) \leq 0 \qquad j = 1, 2, \ldots, n\end{aligned} \tag{11.1}$$

where $\{x\}$ is the set of design variables; ng is the number of member groups in structure (number of design variables); D_i is the allowable set of values for the design variable x_i; $W(\{x\})$ presents weight of the structure; nm is the number of members of the structure; ρ_i denotes the material density of member i; L_i and x_i are the length and the cross-sectional of member i, respectively; $g_j(\{x\})$ denotes design constraints; and n is the number of the constraints.

D_i can be considered either as a continuous set or as a discrete one [5]. In the continuous problems, the design variables can vary continuously in the optimization process

$$D_i = \left\{x_i \middle| x_i \in [x_{i,\min}, x_{i,\max}]\right\} \tag{11.2}$$

where $\mathrm{x}_{\mathrm{i,min}}$ and $x_{i,max}$ are minimum and maximum allowable values for the design variable i, respectively. If the design variables represent a selection from a set of parts as

$$D_i = \left\{d_{i,1}, d_{i,2}, \ldots, d_{i,r(i)}\right\} \tag{11.3}$$

Then the problem is considered as a discrete one, where $r(i)$ is the number of available discrete values for the ith design variable.

In order to handle the constraints, a penalty approach is utilized. In this method, the aim of the optimization is redefined by introducing the cost function as

$$f_{\text{cost}}\left(\{x\}\right) = \left(1 + \varepsilon_1 \cdot \upsilon\right)^{\varepsilon_2} \times W\left(\{x\}\right), \qquad \upsilon = \sum_{i=1}^{n} \max[0, \upsilon_i] \tag{11.4}$$

where n represents the number of evaluated constraints for each individual design. The constant ε_1 and ε_2 are selected considering the exploration and the exploitation rate of the search space. Here, ε_1 is set to unity, ε_2 is selected in a way that it decreases the penalties and reduces the cross-sectional areas. Thus, in the first steps of the search process, ε_2 is set to 1.5 and ultimately increased to 3.

This chapter investigates two types of skeletal structures consisting of trusses and frames. The constraint conditions for these structures are briefly explained in the following sections.

11.2.1 Constraint Conditions for Truss Structures

For truss structures, the stress limitations of the members are imposed according to the provisions of ASD-AISC [6] as follows:

$$\begin{cases} \sigma_i^+ = 0.6F_y & for \;\; \sigma_i \geq 0 \\ \sigma_i^- & for \;\; \sigma_i < 0 \end{cases} \tag{11.5}$$

where σ_i^- is calculated according to the slenderness ratio:

$$\sigma_i^- = \begin{cases} \left[\left(1 - \frac{\lambda_i^2}{2C_C^2}\right)F_y\right] \Big/ \left(\frac{5}{3} + \frac{3\lambda_i}{8C_C} - \frac{\lambda_i^3}{8C_C^3}\right) & for \;\; \lambda_i < C_C \\ \frac{12\pi^2 E}{23\lambda_i^2} & for \;\; \lambda_i \geq C_C \end{cases} \tag{11.6}$$

where E is the modulus of elasticity; F_y is the yield stress of steel; C_c denotes the slenderness ratio (λ_i) dividing the elastic and inelastic buckling regions; λ_i presents the slenderness ratio.

The other constraint is the limitation of the nodal displacements:

$$\delta_\mathrm{i} \leq \delta_i^u \qquad i = 1, 2, \ldots., nn \tag{11.7}$$

where δ_i is the nodal deflection; δ_i^u is the allowable deflection of node i; and nn is the number of nodes.

11.2.2 Constraint Conditions for Steel Frames

Optimal design of frame structures is subjected to the following constrains according to LRFD-AISC provisions [7]:

Maximum lateral displacement

$$\frac{\Delta_T}{H} \leq R \tag{11.8}$$

Inter-story displacements constraints

$$\frac{d_i}{h_i} \leq R_I, \quad i = 1, 2, \ldots, ns \tag{11.9}$$

The strength constraints

$$\begin{aligned} &\frac{P_u}{2\phi_c P_n} + \left(\frac{M_{ux}}{\phi_b M_{nx}} + \frac{M_{uy}}{\phi_b M_{ny}}\right) \leq 1, \quad \text{For} \quad \frac{P_u}{\phi_c P_n} < 0.2 \\ &\frac{P_u}{\phi_c P_n} + \frac{8}{9}\left(\frac{M_{ux}}{\phi_b M_{nx}} + \frac{M_{uy}}{\phi_b M_{ny}}\right) \leq 1, \quad \text{For} \quad \frac{P_u}{\phi_c P_n} \geq 0.2 \end{aligned} \tag{11.10}$$

where Δ_T is the maximum lateral displacement; H is the height of the frame structure; R is the maximum drift index (1/300); d_i is the inter-story drift; h_i is the story height of the ith floor, ns is the total number of stories; R_I presents the inter-story drift index permitted by the code of the practice (1/300); 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_{ux} and M_{uy} are the required flexural strengths in the x and y directions, respectively; M_{nx} and M_{ny} are the nominal flexural strengths in the x and y directions (for two-dimensional structures, $M_{ny} = 0$); and ϕ_b denotes the flexural resistance reduction factor ($\phi_b = 0.90$). The nominal tensile strength for yielding in the gross section is computed as

$$P_n = A_g \cdot F_y \tag{11.11}$$

and the nominal compressive strength of a member is computed as

$$P_n = A_g \cdot F_{cr} \tag{11.12}$$

$$F_{cr} = \left(0.658^{\lambda_c^2}\right) F_y, \quad \text{For} \quad \lambda_c \leq 1.5 \tag{11.13}$$

$$F_{cr} = \left(\frac{0.877}{\lambda_c^2}\right) F_y, \quad \text{For} \quad \lambda_c > 1.5$$

$$\lambda_c = \frac{kl}{r\pi}\sqrt{\frac{F_y}{E}} \tag{11.14}$$

where A_g is the cross-sectional area of a member.

11.3 Imperialist Competitive Algorithm

ICA simulates the social-political process of imperialism and imperialistic competition. This algorithm contains a population of agents or countries. The pseudo-code of the algorithm is as follows:

Step 1: Initialization The primary locations of the agents or countries are determined by the set of values assigned to each decision variable randomly as

$$x_{i,j}^{(\mathrm{o})} = x_{i,\min} + rand \cdot (x_{i,\max} - x_{i,\min}) \tag{11.15}$$

where $x_{i,j}^{(\mathrm{o})}$ determines the initial value of the ith variable for the jth country; $x_{i,\min}$ and $x_{i,\max}$ are the minimum and the maximum allowable values for the ith variable; *rand* is a random number in the interval [0,1]. If the allowable search space is a discrete one, using a rounding function will also be necessary.

For each country, the cost identifies its usefulness. In the optimization process, the cost is proportional to the penalty function. When the values of cost for initial countries are calculated [as defined by (11.4)], some of the best countries (in optimization terminology, countries with the least costs) will be selected to be the imperialist states and the remaining countries will form the colonies of these imperialists. The total number of initial countries is set to $N_{country}$ and the number of the most powerful countries to form the empires is equal to N_{imp}. The remaining N_{col} of the initial countries will be the colonies each of which belongs to an empire. In this chapter, a population of 30 countries consisting of 3 empires and 27 colonies are used. All the colonies of initial countries are divided among the imperialists based on their power. The power of each country, the counterpart of fitness value, is inversely proportional to its cost value. That is, the number of colonies of an empire should be directly proportionate to its power. In order to proportionally divide the colonies among the imperialists, a normalized cost for an imperialist is defined as

$$C_j = f_{\mathrm{cost}}^{(imp,j)} - \max_{\mathrm{i}} \left(f_{\mathrm{cost}}^{(imp,i)} \right) \tag{11.16}$$

where $f_{cost}^{(imp,j)}$ is the cost of the jth imperialist and C_j is its normalized cost. The colonies are divided among empires based on their power or normalized cost and for the jth empire it will be as follows:

$$NC_j = Round\left(\left| \frac{C_j}{\sum_{i=1}^{N_{imp}} C_i} \right| \cdot N_{col} \right) \tag{11.17}$$

where NC_j is the initial number of colonies associated to the jth empire which are

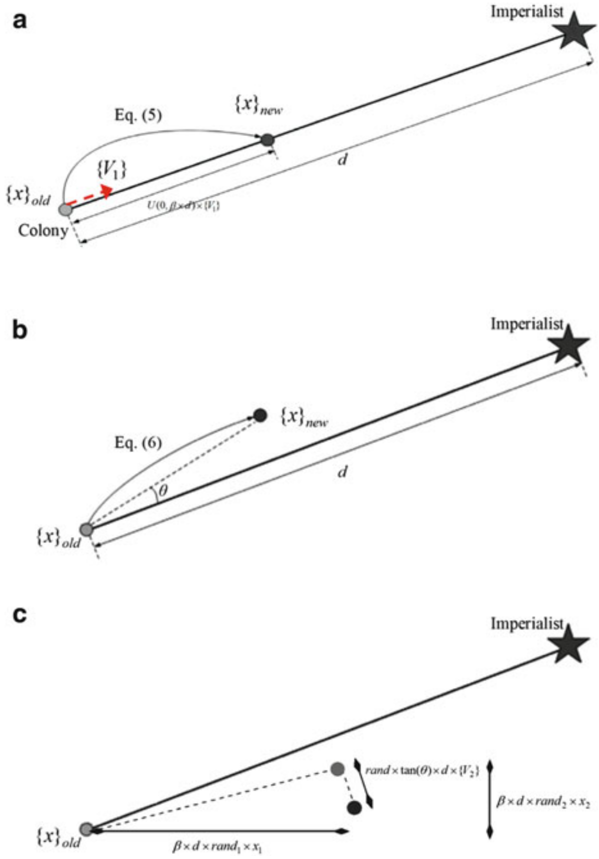

Fig. 11.1 Movement of colonies to its new location in the ICA [2] (**a**) toward their relevant imperialist, (**b**) in a deviated direction (**c**) using various random values

selected randomly among the colonies. These colonies together with the *j*th imperialist, form the empire number *j*.

Step 2: Colonies Movement In the ICA, the assimilation policy pursued by some of former imperialist states, is modeled by moving all the colonies toward the imperialist. This movement is shown in Fig. 11.1a in which a colony moves toward the imperialist by a random value that is uniformly distributed between 0 and $\beta \times d$ [3]:

$$\{x\}_{new} = \{x\}_{old} + U(0, \beta \times d) \times \{V_1\} \tag{11.18}$$

where β is a parameter with a value greater than one, and d is the distance between colony and imperialist. $\beta > 1$ peruseds the colonies to get closer to the imperialist state from both sides. $\beta \gg 1$ gradually results in a divergence of colonies from the imperialist state, while a very close value to 1 for β reduces the search ability of the algorithm. $\{V_1\}$ is a vector which its start point is the previous location of the colony and its direction is toward the imperialist locations. The length of this vector is set to unity.

In order to increase the searching around the imperialist, a random amount of deviation is added to the direction of movement. Figure 11.1b shows the new direction which is obtained by deviating the previous location of the country as big as θ. In this figure θ is a random number with uniform distribution as

$$\theta = U(-\gamma, +\gamma) \tag{11.19}$$

where γ is a parameter that adjusts the deviation from the original direction. In most of the implementations, a value of about 2 for β [3] and about 0.1 (Rad) for γ, result in a good convergence of the countries to the global minimum.

In order to improve the performance of the ICA, we change the movement step as follow:

First: different random values are utilized for different components of the solution vector inplace of only one value (11.18) as

$$\{x\}_{new} = \{x\}_{old} + \beta \times d \times \{rand\} \otimes \{V_1\} \tag{11.20}$$

where $\{V_1\}$ is the base vector starting the previous location of colony and directing to the imperialistic; $\{rand\}$ is a random vector and the sign "$\otimes$" denotes an element-by-element multiplication. Since these random values are not necessarily the same, the colony is deviated automatically without using the definition of θ. However, for having a suitable exploration ability, the utilization of θ is modified by defining a new vector.

Second: From the above equation, it is possible to obtain the orthogonal colony-imperialistic contacting line (denoted by $\{V_2\}$). Then, deviation process is performed by using this vector in place of using θ as

$$\begin{aligned}\{x\}_{new} = \{x\}_{old} + \beta \times d \times \{rand\} \otimes \{V_1\} + U(-1, +1) \times \tan(\theta) \times d \\ \times \{V_2\}, \{V_1\} \cdot \{V_2\} = 0, \left|\left|\{V_2\}\right|\right| = 1\end{aligned} \tag{11.21}$$

Figure 11.1c describes the performance of this movement. In order to access the discrete results after performing the movement process, a rounding function is utilized which changes the magnitude of the results by the value of the nearest discrete value. Although this may reduce the exploration of the algorithm [8], as explained in the above, however we increase this ability by considering different random values and by defining a new deviation step.

Step 3: Imperialist Updating If the new position of the colony is better than that of its relevant imperialist (considering the cost function), the imperialist and the colony change their positions and the new location with a lower cost becomes the imperialist. Then the other colonies move toward this new position.

Step 4: Imperialistic Competition Imperialistic competition is another strategy utilized in the ICA methodology. All empires try to take the possession of colonies of other empires and control them. The imperialistic competition gradually reduces the power of weaker empires and increases the power of more powerful ones. The imperialistic competition is modeled by just picking some (usually one) of the weakest colonies of the weakest empires and making a competition among all empires to possess these (this) colonies. In this competition based on their total power, each of empires will have a likelihood of taking possession of the mentioned colonies.

Total power of an empire is mainly affected by the power of imperialist country. But the power of the colonies of an empire has an effect, though negligible, on the total power of that empire. This fact is modeled by defining the total cost as

$$TC_j = f_{\cos t}^{(imp,j)} + \xi \cdot \frac{\sum_{i=1}^{NC_j} f_{\cos t}^{(col,i)}}{NC_j} \tag{11.22}$$

where TC_n is the total cost of the jth empire and ξ is a positive number which is considered to be less than 1. A small value for ξ causes the total power of the empire to be determined by just the imperialist and increasing it will add to the role of the colonies in determining the total power of the corresponding empire. The value of 0.1 for ξ is found to be a suitable value in most of the implementations [3]. Similar to (11.16), the normalized total cost is defined as

$$NTC_j = TC_j - \max_{\mathrm{i}} \ (TC_i) \tag{11.23}$$

where NTC_j is the normalized total cost of the jth empire. Having the normalized total cost, the possession probability of each empire is evaluated by:

$$P_j = \left| \frac{NTC_j}{\sum_{i=1}^{N_{imp}} NTC_i} \right| \tag{11.24}$$

Step 5: Implementation When an empire loses all of its colonies, it is assumed to be collapsed. In this model implementation, where the powerless empires collapse in the imperialistic competition, the corresponding colonies will be divided among the other empires.

Step 6: Terminating Criterion Control Moving colonies toward imperialists are continued and imperialistic competition and implementations are performed during the search process. When the number of iterations reaches to a pre-defined value or the amount of improvement in the best result reduces to a pre-defined value, the searching process is stopped.

The movement of colonies towards their relevant imperialist states along with competition among empires and also the collapse mechanism will hopefully cause all the countries to converge to a state in which there exist just one empire in the world and all the other countries are colonies of that empire. In this ideal new world, colonies will have the same position and power as the imperialist.

11.4 Design Examples

In this section, the optimal design of four steel structures is performed by the present algorithm. The final results are compared to the solutions of other methods to demonstrate the efficiency of the present approach. The examples contain a dome shaped truss example with continuous search space and a 72-bar spatial truss with the discrete variables. In addition, two benchmark frames are optimized by the ICA to find the optimum designs.

11.4.1 Design of a 120-Bar Dome Shaped Truss

The topology and elements group numbers of 120-bar dome truss are shown in Fig. 11.2. The modulus of elasticity is 30,450 ksi (210,000 MPa), and the material density is 0.288 lb/in^3 (7,971.810 kg/m^3). The yield stress of steel is taken as 58.0 ksi (400 MPa). The dome is considered to be subjected to vertical loading at all the unsupported joints. These loads are taken as −13.49 kips (−60 kN) at node 1, −6.744 kips (−30 kN) at nodes 2 through 14, and −2.248 kips (−10 kN) at the rest of the nodes. The minimum cross-sectional area of all members is 0.775 in^2 (2 cm^2) and the maximum cross-sectional area is taken as 20.0 in^2 (129.03 cm^2). The constraints are stress constraints [as defined by (11.5) and (11.6)] and displacement limitations of ±0.1969 in (±5 mm) imposed on all nodes in x, y and z directions.

Table 11.1 shows the best solution vectors, the corresponding weights and the required number of analyses for convergence of the present algorithm and some other metaheuristic algorithms. ICA-based algorithm needs 6,000 analyses to find the best solution while this number is equal to 150,000, 32,600, 10,000, 10,000 and 7,000 analyses for a PSO-based algorithm [11], a PSO and ACO hybrid algorithm [11], a combination algorithm based on PSO, ACO and HS [11], an improved BB–BC method using PSO properties [12] and the CSS algorithm [13], respectively. As a result, the ICA optimization algorithm has best convergence rates among the considered metaheuristics. Figure 11.3 shows the convergence history for the best results of the ICA. Comparing the final results of the ICA and those of the other metaheuristics, ICA finds the third best result while the difference between the

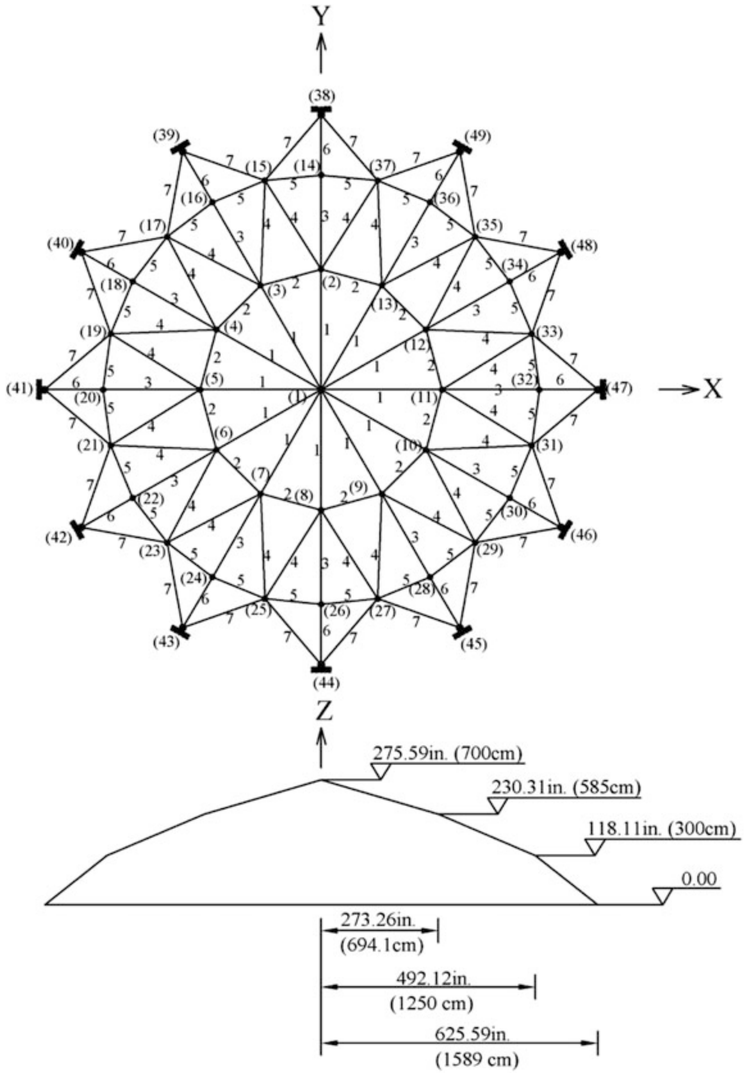

Fig. 11.2 Schematic of a 120-bar dome shaped truss

result of the ICA and those obtained by the HPSACO and the CSS methods, as the first and second best results, are very small. The maximum value for displacement is equal to 0.1969 in (5 mm) and the maximum stress ratio is equal to 99.999 %.

Table 11.1 Performance comparison for the 120-bar dome truss

Element group		Optimal cross-sectional areas (in^2) PSOPC [10]	PSACO [10]	HPSACO [10]	HBB–BC [9]	CSS [6]	Present work [2] in^2	Present work [2] cm^2
1	A_1	3.040	3.026	3.095	3.037	3.027	3.0275	19.532
2	A_2	13.149	15.222	14.405	14.431	14.606	14.4596	93.288
3	A_3	5.646	4.904	5.020	5.130	5.044	5.2446	33.836
4	A_4	3.143	3.123	3.352	3.134	3.139	3.1413	20.266
5	A_5	8.759	8.341	8.631	8.591	8.543	8.4541	54.543
6	A_6	3.758	3.418	3.432	3.377	3.367	3.3567	21.656
7	A_7	2.502	2.498	2.499	2.500	2.497	2.4947	16.095
Best weight (lb)		33,481.2	33,263.9	33,248.9	33,287.9	33,251.9	33,256.2	147,931 N
No. of required analyses		150,000	32,600	10,000	10,000	7,000	6,000	

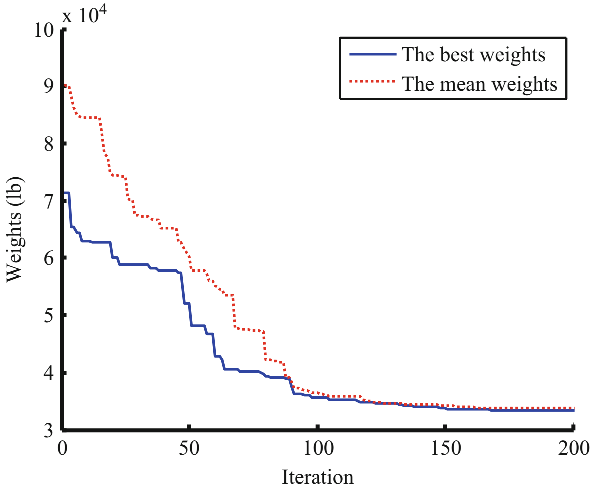

Fig. 11.3 The convergence for the dome shaped truss obtained by the ICA [2]

11.4.2 Design of a 72-Bar Spatial Truss

For the 72-bar spatial truss structure shown in Fig. 11.4, the material density is 0.1 lb/in^3 (2,767.990 kg/m^3) and the modulus of elasticity is 10,000 ksi (68,950 MPa). The members are subjected to the stress limits of ±25 ksi (±172.375 MPa). The nodes are subjected to the displacement limits of ±0.25 in (±0.635 cm). The 72 structural members of this spatial truss are categorized as 16 groups using symmetry: (1) A_1–A_4, (2) A_5–A_{12}, (3) A_{13}–A_{16}, (4) A_{17}–A_{18},

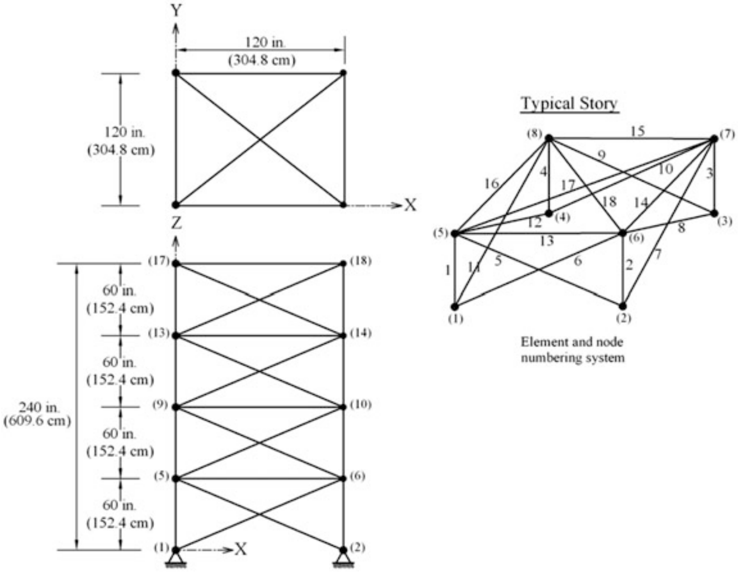

Fig. 11.4 Schematic of a 72-bar spatial truss

(5) A_{19}–A_{22}, (6) A_{23}–A_{30}, (7) A_{31}–A_{34}, (8) A_{35}–A_{36}, (9) A_{37}–A_{40}, (10) A_{41}–A_{48}, (11) A_{49}–A_{52}, (12) A_{53}–A_{54}, (13) A_{55}–A_{58}, (14) A_{59}–A_{66} (15), A_{67}–A_{70}, and (16) A_{71}–A_{72}. The discrete variables are selected from Table 11.2. The values and directions of the two load cases applied to the 72-bar spatial truss are listed in Table 11.3.

The ICA algorithm can find the best design among the other existing studies. The best weight of the ICA algorithm is 392.84 lb (178.19 kg), while it is 393.38 lb (178.43 kg), for the HPSACO [8]. The weight of the GA-based algorithm is equal to 427.203 lb (193.77 kg) [14]. The PSOPC and the standard PSO algorithms do not find optimal results when the maximum number of iterations is reached [10]. The HPSO and HPSACO algorithms get the optimal solution after 50,000 [10] and 5,330 [11] analyses while it takes only 4,500 analyses for the ICA. Table 11.4 compares the results of the CSS algorithm to those of the previously reported methods in the literature. In this example, stress constraints are not dominant while the maximum nodal displacement (0.2499 in or 0.635 cm) is close to its allowable value.

11.4.3 Design of a 3-Bay, 15-Story Frame

The configuration and applied loads of a three-bay fifty-story frame structure [5] is shown in Fig. 11.5. The displacement and AISC combined strength constraints are the performance constraint of this frame. The sway of the top story is limited to

Table 11.2 The available cross-section areas of the AISC code

No.	in.^{2}	mm^2	No.	in.^{2}	mm^2
1	0.111	(71.613)	33	3.840	(2,477.414)
2	0.141	(90.968)	34	3.870	(2,496.769)
3	0.196	(126.451)	35	3.880	(2,503.221)
4	0.250	(161.290)	36	4.180	(2,696.769)
5	0.307	(198.064)	37	4.220	(2,722.575)
6	0.391	(252.258)	38	4.490	(2,896.768)
7	0.442	(285.161)	39	4.590	(2,961.284)
8	0.563	(363.225)	40	4.800	(3,096.768)
9	0.602	(388.386)	41	4.970	(3,206.445)
10	0.766	(494.193)	42	5.120	(3,303.219)
11	0.785	(506.451)	43	5.740	(3,703.218)
12	0.994	(641.289)	44	7.220	(4,658.055)
13	1.000	(645.160)	45	7.970	(5,141.925)
14	1.228	(792.256)	46	8.530	(5,503.215)
15	1.266	(816.773)	47	9.300	(5,999.988)
16	1.457	(939.998)	48	10.850	(6,999.986)
17	1.563	(1,008.385)	49	11.500	(7,419.430)
18	1.620	(1,045.159)	50	13.500	(8,709.660)
19	1.800	(1,161.288)	51	13.900	(8,967.724)
20	1.990	(1,283.868)	52	14.200	(9,161.272)
21	2.130	(1,374.191)	53	15.500	(9,999.980)
22	2.380	(1,535.481)	54	16.000	(10,322.560)
23	2.620	(1,690.319)	55	16.900	(10,903.204)
24	2.630	(1,696.771)	56	18.800	(12,129.008)
25	2.880	(1,858.061)	57	19.900	(12,838.684)
26	2.930	(1,890.319)	58	22.000	(14,193.520)
27	3.090	(1,993.544)	59	22.900	(14,774.164)
28	1.130	(729.031)	60	24.500	(15,806.420)
29	3.380	(2,180.641)	61	26.500	(17,096.740)
30	3.470	(2,238.705)	62	28.000	(18,064.480)
31	3.550	(2,290.318)	63	30.000	(19,354.800)
32	3.630	(2,341.931)	64	33.500	(21,612.860)

Table 11.3 Loading conditions for the 72-bar spatial truss

	Case 1			Case 2		
Node	P_X *kips (kN)*	P_Y *kips (kN)*	P_Z *kips (kN)*	P_X	P_Y	P_Z *kips (kN)*
17	5.0 (22.25)	5.0 (22.25)	−5.0 (22.25)	0.0	0.0	−5.0 (22.25)
18	0.0	0.0	0.0	0.0	0.0	−5.0 (22.25)
19	0.0	0.0	0.0	0.0	0.0	−5.0 (22.25)
20	0.0	0.0	0.0	0.0	0.0	−5.0 (22.25)

23.5 cm (9.25 in.). The material has a modulus of elasticity equal to $E = 200$ GPa (29,000 ksi) and a yield stress of $F_y = 248.2$ MPa (36 ksi). The effective length factors of the members are calculated as $K_x \geq 0$ for a sway-permitted frame and the out-of-plane effective length factor is specified as $K_y = 1.0$. Each column is

Table 11.4 Optimal design comparison for the 72-bar spatial truss

Element group		Optimal cross-sectional areas (in^2)				
		GA [14]	PSOPC [10]	HPSO [10]	HPSACO [11]	Present work [2]
1	$A_1 \sim A_4$	0.196	4.490	4.970	1.800	1.99
2	$A_5 \sim A_{12}$	0.602	1.457	1.228	0.442	0.442
3	$A_{13} \sim A_{16}$	0.307	0.111	0.111	0.141	0.111
4	$A_{17} \sim A_{18}$	0.766	0.111	0.111	0.111	0.141
5	$A_{19} \sim A_{22}$	0.391	2.620	2.880	1.228	1.228
6	$A_{23} \sim A_{30}$	0.391	1.130	1.457	0.563	0.602
7	$A_{31} \sim A_{34}$	0.141	0.196	0.141	0.111	0.111
8	$A_{35} \sim A_{36}$	0.111	0.111	0.111	0.111	0.141
9	$A_{37} \sim A_{40}$	1.800	1.266	1.563	0.563	0.563
10	$A_{41} \sim A_{48}$	0.602	1.457	1.228	0.563	0.563
11	$A_{49} \sim A_{52}$	0.141	0.111	0.111	0.111	0.111
12	$A_{53} \sim A_{54}$	0.307	0.111	0.196	0.250	0.111
13	$A_{55} \sim A_{58}$	1.563	0.442	0.391	0.196	0.196
14	$A_{59} \sim A_{66}$	0.766	1.457	1.457	0.563	0.563
15	$A_{67} \sim A_{70}$	0.141	1.228	0.766	0.442	0.307
16	$A_{71} \sim A_{72}$	0.111	1.457	1.563	0.563	0.602
Weight (lb)		427.203	941.82	933.09	393.380	392.84
No. of required analyses		–	150,000	50,000	5,330	4,500

considered as non-braced along its length, and the non-braced length for each beam member is specified as one-fifth of the span length.

The optimum design of the frame is obtained after 6,000 analyses by using the ICA, having the minimum weight of 417.46 kN (93.85 kips). The optimum designs for HBB–BC [9], HPSACO, PSOPC and PSO [5] has the weights of 434.54 (97.65kN), 426.36 (95.85), 452.34 kN (101.69 kips) and 496.68 kN (111.66 kips), respectively. Table 11.5 summarizes the optimal designs for these algorithms. The HBB–BC approach could find the result after 9,900 analyses [9] and the HSPACO needs 6,800 analyses to reach a solution [5].

Figure 11.6 shows the convergence history for the result of the ICA method. The global sway at the top story is 11.52 cm, which is less than the maximum sway. The maximum value for the stress ratio is equal to 98.45 %. Also, the maximum drift story is equal to 1.04 cm.

11.4.4 Design of a 3-Bay 24-Story Frame

Figure 11.7 shows the topology and the service loading conditions of a three-bay twenty four-story frame consisting of 168 members originally designed by Davison and Adams [15]. Camp et al. utilized ant colony optimization [16], Degertekin developed least-weight frame designs for this structure using a harmony search [17]

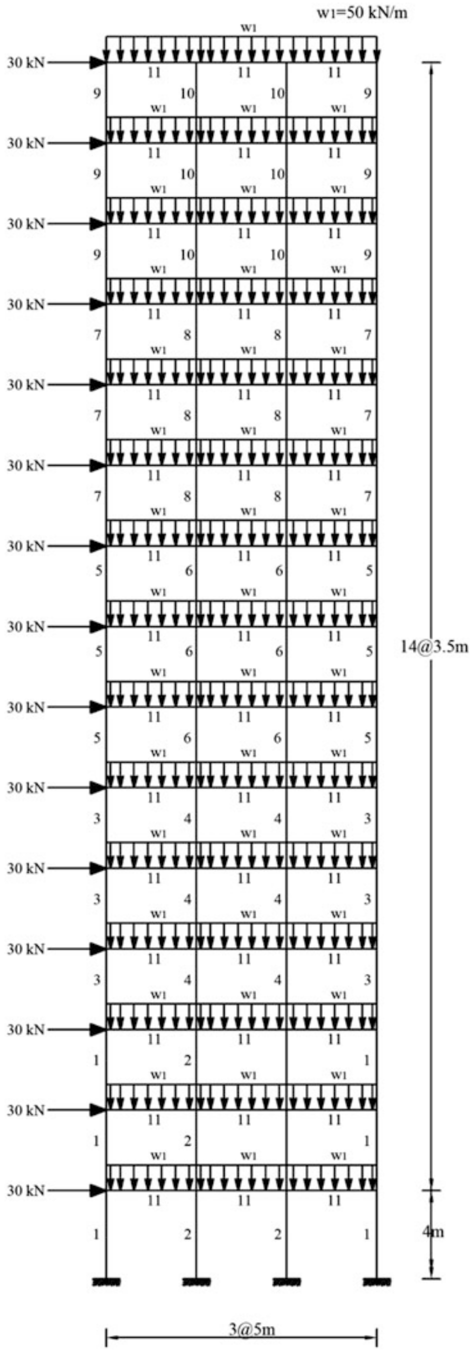

Fig. 11.5 Schematic of a three-bay fifteen-story frame

Table 11.5 Optimal design comparison for the 3-bay 15-story frame

	Optimal W-shaped sections				
Element group	PSO [5]	PSOPC [5]	HPSACO [5]	HBB–BC [9]	Present work [2]
1	W33X118	W26X129	W21X111	W24X117	W24X117
2	W33X263	W24X131	W18X158	W21X132	W21X147
3	W24X76	W24X103	W10X88	W12X95	W27X84
4	W36X256	W33X141	W30X116	W18X119	W27X114
5	W21X73	W24X104	W21X83	W21X93	W14X74
6	W18X86	W10X88	W24X103	W18X97	W18X86
7	W18X65	W14X74	W21X55	W18X76	W12X96
8	W21X68	W26X94	W26X114	W18X65	W24X68
9	W18X60	W21X57	W10X33	W18X60	W10X39
10	W18X65	W18X71	W18X46	W10X39	W12X40
11	W21X44	W21X44	W21X44	W21X48	W21X44
Weight (kN)	496.68	452.34	426.36	434.54	417.466
No. of required analyses	50,000	50,000	6,800	9,900	6,000

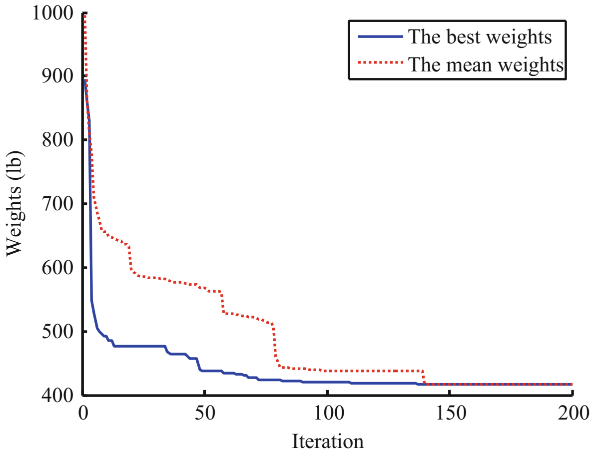

Fig. 11.6 The convergence for the three-bay fifteen-story frame obtained by the ICA [2]

and the authors utilized a hybrid PSO and BB–BC algorithm to solve this example [9].

The frame is designed following the LRFD specification and uses an inter-story drift displacement constraint. The material properties are: the modulus of elasticity $E = 205$ GPa (29,732 ksi) and a yield stress of $F_y = 230.3$ MPa (33.4 ksi). The detailed information is available in [9].

Table 11.6 lists the designs developed by: the ICA, the HBB–BC algorithm [9], the ant colony algorithm [16] and harmony search [17]. The ICA algorithm required 7,500 frame analyses to converge to a solution, while the 10,500 analyses were required by HBB–BC [9], 15,500 analyses by ACO [16] and 13,924 analyses by HS [17]. In this example, ICA can find the best results with 946.25 kN which is 3.67 %, 1.01 % and 1.60 % lighter than the results of the ACO [16], HS [17], and HBB–BC

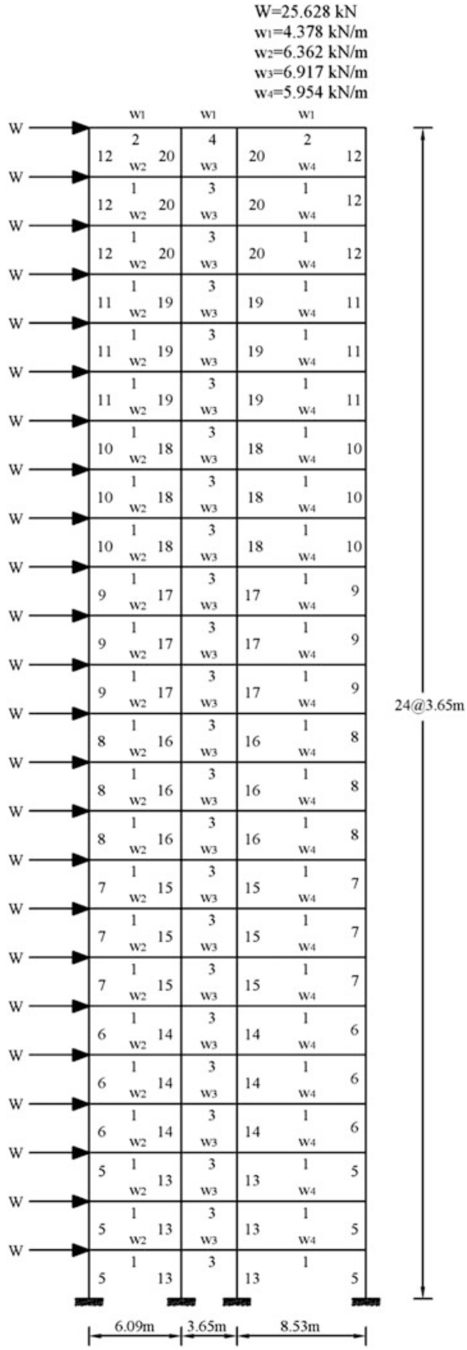

Fig. 11.7 Schematic of a three-bay twenty four-story frame

Table 11.6 Optimal design comparison for the 3-bay 24-story frame

	Optimal W-shaped sections			
	Camp et al. [16]	Degertekin [17]		
Element group	ACO	HS	HBB–BC [9]	Present work [2]
1	W30X90	W30X90	W30X90	W30X90
2	W8X18	W10X22	W21X48	W21X50
3	W24X55	W18X40	W18X46	W24X55
4	W8X21	W12X16	W8X21	W8X28
5	W14X145	W14X176	W14X176	W14X109
6	W14X132	W14X176	W14X159	W14X159
7	W14X132	W14X132	W14X109	W14X120
8	W14X132	W14X109	W14X90	W14X90
9	W14X68	W14X82	W14X82	W14X74
10	W14X53	W14X74	W14X74	W14X68
11	W14X43	W14X34	W14X38	W14X30
12	W14X43	W14X22	W14X30	W14X38
13	W14X145	W14X145	W14X159	W14X159
14	W14X145	W14X132	W14X132	W14X132
15	W14X120	W14X109	W14X109	W14X99
16	W14X90	W14X82	W14X82	W14X82
17	W14X90	W14X61	W14X68	W14X68
18	W14X61	W14X48	W14X48	W14X48
19	W14X30	W14X30	W14X34	W14X34
20	W14X26	W14X22	W14X26	W14X22
Weight (kN)	980.63	956.13	960.90	946.25
No. of required analyses	15,500	13,924	10,500	7,500

[9], respectively. The global sway at the top story is 25.52 cm (10.05 in.) which is less than the maximum sway. The maximum value for the stress ratio is 99.37 % and the maximum inter-story drift is equal to 1.215 cm (0.4784 in.). Figure 11.8 shows the values of the stress ratios for all elements of the optimum design obtained by the ICA algorithm.

11.5 Discussions

Many of metaheuristic algorithms are proposed based on the simulation of the natural processes. The genetic algorithms, particle swarm optimization, ant colony optimization, harmony search and charged system search are the most well-known metaheuristic algorithms. As an alternative to these metaheuristic approaches, this chapter investigates the performance of a new metaheuristic algorithm to optimize the design of skeletal structures. This method is called ICA which is a socio-politically motivated optimization algorithm.

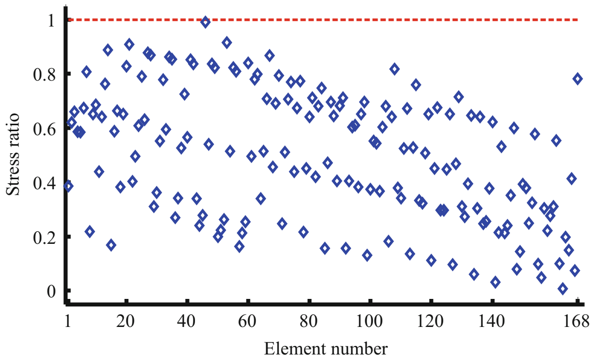

Fig. 11.8 The values of the stress ratios of elements for the ICA result [2]

In the ICA, an agent or a country can be treated as a colony or imperialist and the agents collectively form a number of empires. This algorithm starts with some random initial countries. Some of the best countries are selected to be the imperialist states and all the other countries form the colonies of these imperialists. Imperialistic competitions among the empires direct the search process towards the powerful imperialist and thus to the optimum spaces. During the competition, when weak empires collapse, the powerful ones take possession of their colonies. In addition, colonies of an empire move toward their related imperialist. In order to improve the ICA performance, here two movement steps are defined by using: (1) different random values for the components of the solution vector instead of only one value; (2) deviation by using orthogonal colony-imperialistic contacting line instead of using θ.

Four design examples consisting of two trusses and two frames are considered to illustrate the efficiency of the present algorithm. The comparisons of the numerical results of these structures utilizing the ICA and those obtained by other advanced optimization methods are performed to demonstrate the robustness of the present algorithm in finding good results in a less number of iterations. In order to highlight the positive characters of the ICA, a comparison of the ICA and the PSO algorithm is provided in the following:

– In the ICA algorithm, there is no need to save the pervious location of agents (velocity), while the PSO requires two positions saving memory (the current position and the pervious position).
– In the ICA algorithm, $\{V_1\}$ determines the movement direction of agents, while in the PSO, this is performed by the global and local best vectors. The vector $\{V_1\}$ is the best of the empire, i.e., it is the best agent among a predefined number of agents, while in the PSO the global best, denoted by $\{P_g\}$, is the position of the best agent of all agents. Therefore, $\{V_1\}$ will change for different agents during an iteration (depending on the empire which they belong to) and this helps the algorithm to increase the exploration ability, while $\{P_g\}$ is constant for all the agents in an iteration.

– In the ICA algorithm, saving the local best position of agents is not necessary, and instead the vector $\{V_2\}$ is utilized.

References

1. Kaveh A, Talatahari S (2010) Imperialist competitive algorithm for engineering design problems. Asian J Civil Eng 11(6):675–697
2. Kaveh A, Talatahari S (2010) Optimum design of skeletal structures using imperialist competitive algorithm. Comput Struct 88:1220–1229
3. Atashpaz-Gargari E, Lucas C (2007) Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: IEEE congress on evolutionary computation, Singapore, pp 4661–4667
4. Atashpaz-Gargari E, Hashemzadeh F, Rajabioun R, Lucas C (2008) Colonial competitive algorithm: a novel approach for PID controller design in MIMO distillation column process. Int J Intell Comput Cybern 1(3):337–355
5. Kaveh A, Talatahari S (2009) Hybrid algorithm of harmony search, particle swarm and ant colony for structural design optimization. In: Geem ZW (ed) Harmony search algorithms for structural design. Springer, Berlin, Chapter 5
6. American Institute of Steel Construction (AISC) (1989) Manual of steel construction–allowable stress design, 9th edn. AISC, Chicago
7. American Institute of Steel Construction (AISC) (2001) Manual of steel construction–load resistance factor design, 3rd edn. AISC, Chicago
8. Kaveh A, Talatahari S (2009) A particle swarm ant colony optimization for truss structures with discrete variable. J Constr Steel Res 65(8–9):1558–1568
9. Kaveh A, Talatahari S (2010) A discrete big bang–big crunch algorithm for optimal design of skeletal structures. Asian J Civil Eng 11(1):103–122
10. Li LJ, Huang ZB, Liu F (2009) A heuristic particle swarm optimization method for truss structures with discrete variables. Comput Struct 87(7–8):435–443
11. Kaveh A, Talatahari S (2009) Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures. Comput Struct 87(5–6):267–283
12. Kaveh A, Talatahari S (2009) Size optimization of space trusses using Big Bang–Big Crunch algorithm. Comput Struct 87(17–18):1129–1140
13. Kaveh A, Talatahari S (2010) A novel heuristic optimization method: charged system search. Acta Mech 213(3–4):267–286
14. Wu SJ, Chow PT (1995) Steady-state genetic algorithms for discrete optimization of trusses. Comput Struct 56(6):979–991
15. Davison JH, Adams PF (1974) Stability of braced and unbraced frames. J Struct Div ASCE 100(2):319–334
16. Camp CV, Bichon J (2005) Design of steel frames using ant colony optimization. J Struct Eng ASCE 131:369–379
17. Degertekin SO (2008) Optimum design of steel frames using harmony search algorithm. Struct Multidiscip Optim 36:393–401