International Journal of Civil Engineering

Structural Engineering

A comparative study of two meta-heuristic algorithms for optimum design of reinforced concrete frames

Ali Kaveh*, **Omid Sabzi**

Received: September 2010, Revised: January 2011, Accepted: July 2011

Abstract

This article presents the application of two algorithms: heuristic big bang-big crunch (HBB-BC) and a heuristic particle swarm ant colony optimization (HPSACO) to discrete optimization of reinforced concrete planar frames subject to combinations of gravity and lateral loads based on ACI 318-08 code. The objective function is the total cost of the frame which includes the cost of concrete, formwork and reinforcing steel for all members of the frame. The heuristic big bang-big crunch (HBB-BC) is based on BB-BC and a harmony search (HS) scheme to deal with the variable constraints. The HPSACO algorithm is a combination of particle swarm with passive congregation (PSOPC), ant colony optimization (ACO), and harmony search scheme (HS) algorithms. In this paper, by using the capacity of BB-BC in ACO stage of HPSACO, its performance is improved. Some design examples are tested using these methods and the results are compared.

Keywords: Structural optimization, Reinforced concrete plane frames, Big Bang-Big Crunch algorithm, Particle swarm, Ant colony optimization.

1. Introduction

In comparison to steel structures, the optimization of reinforced concrete structures is more complicated; because RC structures can be designed with a semi-infinite set of member sizes and various patterns of reinforcements. Also, in the optimization of steel structures, only one material is considered and the cost is directly proportional to the weight of the structure. However in the case of RC structures, because of having multi-material, the three different cost items consisting of concrete, steel and formwork to be considered, and each of these parameters influence the total cost of the structure. Therefore, the optimization problem converts into the selection of an appropriate combination of sections dimensions and the quantity of reinforcement so that the overall cost of structure is minimum.

In the last three decades, many researchers have studied the cost optimization of reinforced concrete structural elements and frames. Adamu *et al*. [1] used the continuum-type optimality criteria for minimizing the cost design of reinforced concrete beams. An optimality criteria procedure has been used to attain the optimal design of reinforced concrete frames based on the ACI code by Moharrami and Grierson [2]. Zielinski *et al*. [3] employed an internal penalty function algorithm to cost optimum design of reinforced concrete short-tied rectangular columns based on the Canadian Standard specifications. The cost optimum design of three-dimensional skeletal structures using the optimality criteria is performed by Fadaee and Grierson [4]. Balling and Yao [5] optimized three-dimensional RC frames using sequential quadratic programming and gradient based method. A review of research work associated with cost optimization of concrete structures was discussed by Sarma and Adeli [6]. Rajeev and Krishnamoorthy [7] applied a simple genetic algorithm (SGA) to obtain optimal design of planar RC frames. Optimization of T-shaped reinforce concrete sections under bending was performed by Ferreira et al. [8]. Optimum design of two dimensional frames using genetic algorithm was carried out by Lee and Ahn [9] and Camp et al. [10] via constructing a database of sections for beams and columns. Govindaraj and Ramasmy [11,12] have investigated the optimum design of continuous beams, and two and three dimensional RC frames using the genetic algorithm. Cost optimization of buildings with planar slabs is carried out by Sahab *et al*. [13,14] using a hybrid genetic algorithm. The cost of the rectangular beams and columns of RC buildings was minimized by Choi and Kwak [15] via direct search method.

* *Corresponding Author: alikaveh@iust.ac.ir*
Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran-16, Iran

Kwak and Kim [16] used a direct search method and Kwak and Kim [17] employed an integrated genetic algorithm complemented with a direct search for optimal design of planar RC frames.

In relation with BB-BC and HPSACO, the following developments can be mentioned:

Erol and Eksin [18] introduced the BB-BC as a new meta-heuristic approach. Camp [19], and Kaveh and Talatahari [20] employed BB-BC for optimal design of trusses, and Kaveh and Talatahari [21] performed the optimal design of Schwedler and ribbed domes via a hybrid BB-BC algorithm.

Shelokar *et al*. [22] proposed the PSACO (a hybrid particle swarm optimizer and ant colony approach) for the solution of continuous unconstrained problems. Recently, Kaveh and Talatahari [23-26] have presented an optimization algorithm, so-called heuristic particle swarm ant colony optimization (HPSACO), for steel truss and frames problems. In HPSACO, to reach to an efficient algorithm, the PSOPC algorithm (a hybrid PSO with passive congregation) is combined with the ant colony algorithm and harmony search approach. The HPSACO applies the PSOPC for global optimization and an ant colony approach is employed as a local search, wherein ants apply a pheromone-guided mechanism to update the positions found by the particles in the earlier stage. Harmony search (HS) works as a handling approach to deal with variable boundaries [26]. In this paper the equation of standard deviation in ACO stage is different with that of Ref. [24]. For extra study about PSO-based, ACO and HS algorithms, the interested reader can refer to Refs. [27-39].

This paper presents the application of two algorithms: heuristic big bang-big crunch (HBB-BC) and a heuristic particle swarm-ant colony optimization (HPS-ACO) to discrete optimization of reinforced concrete planar frames subject to combinations of gravity and lateral loads based on ACI 318-08 code. The remaining sections of this paper are organized as follows. The method to generate the databases for beam and column sections is described in section 2. Section 3 is related to frame analysis and slenderness calculations. Optimization formulation is given in section 4. Section 5 contains explanations of some meta-heuristic algorithms. Some RC frame design examples are studied in section 6. The paper is concluded in section 7.

2. Creating the sections database

In reinforced concrete frames a large number of sections and different patterns of reinforcements can be used for beams and columns. For reducing the complexity of the optimization of RC Frames, here two databases of sections for beams and columns are created. In construction of these databases, some practical limitation are imposed and some standard rules are followed. In practice, usually the sections are considered as rectangular ones with a depth to width ratio between 1.5 to 2.5 for beams and 1 to 2 for columns. The increment of the dimensions of the sections can be considered with steps of 5cm. The sizes of reinforcing bars which are usually used in RC structures are D19, D22, and D25. The ACI 318-08 code [40] considers some limitations on the sections. These limitations consist of the minimum and maximum of steel area in the cross sections, minimum thickness of the concrete cover equal to 40mm for RC members, minimum diameter of the ties and minimum distance between longitudinal reinforcement bars. Taking the above mentioned rules into account, many sections for beams and columns can be generated.

2.1. Beams

Considering the ACI 318-08 code, the following constraints should be imposed on the sections of the beams:

(i) At least 4 bars should be considered in four corners of the cross section, as shown in Fig. 1a.

(ii) The minimum distance between the longitudinal reinforcing bars is taken as S_b=40mm.

(iii) The minimum concrete cover is considered as t_c =40mm.

(iv) The diameter of the ties is assumed as D10.

(v) The layout of the bars is limited to at most two layers.

(vi) The reinforcing bars of the top layer should be positioned on the reinforcing bars of the bottom layer, and the minimum distance between two layers should be 25mm as shown in Fig. 1b.

(vii) In a beam section, if additional reinforcing bars are needed, all such bars will be positioned in a second layer in a symmetric form with respect to the vertical axis of the section, and placed directly above the reinforcing bars in the lower layer. When the aforementioned symmetry does not exist, then it is made symmetric by considering an additional bar as illustrated in Fig. 1c.

In chapter 10 of ACI 318-08 code [40] in relation with the minimum and maximum areas of flexural reinforcement bars, the following rules are imposed:

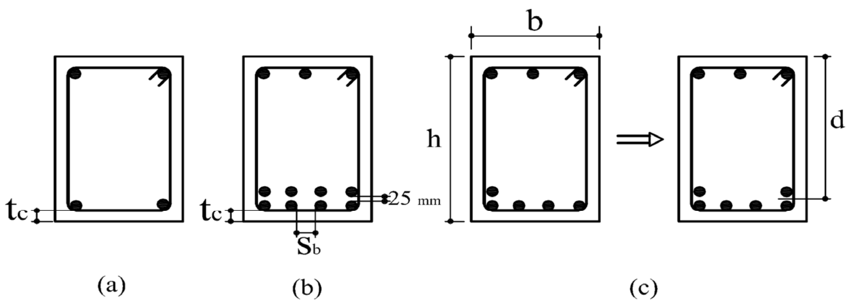

Fig. 1. Limitations on the layout of the reinforcement bars for beam members
(a) At least four bars in the corners (b) Minimum distance between the longitudinal bars in the two layers (c) Symmetric layout of the reinforcement bars with respect to the vertical axis of the section

$$A_{s,\min} = \frac{\sqrt{f_c'}}{4f_y}.b.d \geq \frac{1.4}{f_y}.b.d \quad [mm^2] \tag{1a}$$

$$A_{s,\max} = 0.75(0.85\beta_1)\frac{f_c'}{f_y}.\frac{600}{600+f_y}.b.d \quad [mm^2] \tag{1b}$$

Where b, f_c' and f_y are the width of the cross section, specified compressive strength of the concrete, and specified yield strength of the reinforcing bars, respectively. Here *d* is the effective depth of the section which is measured as the distance from extreme compression fiber to centroid of the longitudinal tensile reinforcing bars of the section. The coefficient β_1 is a factor relating the depth of the equivalent rectangular compressive stress block to the neutral axis depth: it is taken from section 10.2.7.3 of the ACI 318-08 code. The rebars of D19 and D22 are used for positive and negative moments in beams, respectively.

Considering the above rules, 18 types of sections are constructed as follows:

300×450, 300×500, 300×550, 300×600, 350×550, 350×600, 350×650, 350×700, 400×600, 400×650, 400×700, 400×750, 400×800, 450×700,450×750, 450×800, 450×850, 450×900 mm.

A total of 1014 sections with different layouts for the reinforcing bars are generated for beams, the details of which are provided in Table 1. Details of the formation of these sections can be found in Ref. [9].

For the beams, the factored moment capacities at the middle, and near the ends are calculated using

$$\phi M_n = \phi A_s.f_y.(d - \frac{a}{2}) \tag{2}$$

and stored in the database. In this relation, ϕ is the strength reduction factor (ϕ=0.9), A_s is the area of the tensile bars and a is the depth of the equivalent rectangular compressive stress block defined as

$$a = A_s.f_y/(0.85f_c' \, b) \tag{3}$$

Table 1 shows the database of the beam sections used for all the test problems of this paper. The table contains information on width, depth, area and moment of inertia, number of reinforcing bars for positive and negative moments, the corresponding factored bending moment capacities, and the cost per unit length of the beams. The calculation of the cost for the unit length will further be discussed.

2.2. Columns

Using the rules from ACI- 318-08 code, the following constraints should be applied on the sections of the columns:

(i) The minimum free distance between the parallel longitudinal bars is assumed to be s_c=40mm.

(ii) The minimum number of bars is 4 which should be positioned at the four corners of the cross section as shown in Fig. 2a.

(iii) The minimum thickness of the concrete cover is considered as t_c =40mm.

(iv) The diameter of the ties is assumed as D10.

(v) The pattern of the bars should be symmetric and in the two opposite sides of the section, as illustrated in Fig. 2b.

(vi) The minimum and maximum areas of the longitudinal bars are limited to 1 and 8 percent of the gross area of the cross section, respectively.

In column sections only D25 bar are used. In all the examples, for the columns a database consisting of 55 square cross sections with the dimensions 300mm to 900mm at steps of 50mm is used, as provided in Table 2.

The strength of a column under the applied loads (bending and axial force) is evaluated using the P-M interaction diagram. Here a simplified linear P-M interaction diagram is used, as illustrated in Fig. 3.

Table 2 demonstrates the database of the column sections used in the all test problems of this paper, including dimensions, number of reinforcing bars and the key points of the interaction diagram, and also the cost of the unit length of the columns. The procedure followed to compute cost will be clarified later in the paper.

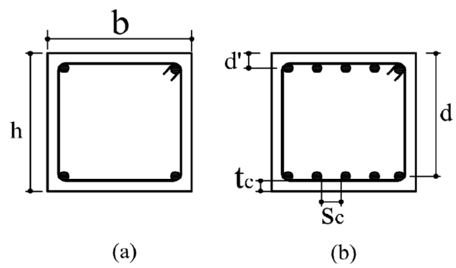

Fig. 2. Limitations of the reinforcement of the column sections (a) At least 4 longitudinal bars at four corners of the section (b) Symmetric pattern of the bars and the distance and cover of the reinforcing bars

Table 1. Database for the beams considered in all the examples

Beam number	Width (mm)	Depth (mm)	Area ($\times 10^2$mm^2)	Moment of inertia ($\times 10^6$mm^4)	Number of bars: Center (D19)	Number of bars: End (D22)	Factored moment resistance (kN.m): Center	Factored moment resistance (kN.m): End	Cost per unit length ($)
1	300	450	1350	2278.1	2	2	74.41	97.67	133.95
2	300	450	1350	2278.1	3	2	108.84	97.67	135.96
....	....	....	....	....	....	....	....	....	....
1013	450	900	4050	27338	10	12	761.3	1152.3	301.78
1014	450	900	4050	27338	12	12	893.43	1152.3	305.79

Table 2. Database of considered columns for all the examples

Column number	Width (mm)	Depth (mm)	Number of bars (D25)	P_0 (kN)	P_1 (kN)	P_3 (kN)	P_5 (kN)	M_2 (kN.m)	M_3 (kN.m)	M_4 (kN.m)	Cost per unit length ($)
1	300	300	4	1643.3	1314.7	429	692.7	22.19	81.98	70.52	133.72
2	300	300	6	1880.7	1504.6	405.7	1039.1	25.97	101.82	100.61	140.66
… …	…	…	… …	… …	… …	… …	… …	… … …	… … …	… … …	… … …
54	900	900	22	13128	10503	4961.3	3810	717.17	2230.7	1504.5	492.55
55	900	900	24	13366	10693	4954.9	4156.3	739.31	2326.2	1638.1	499.48

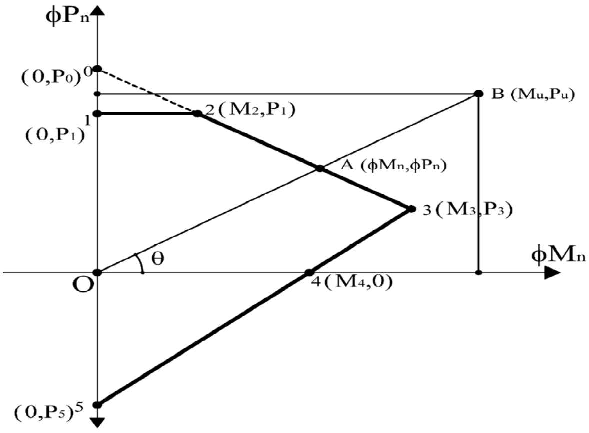

Fig. 3. Limitations of the reinforcement of the column sections
(a) At least 4 longitudinal bars at four corners of the section
(b) Symmetric pattern of the bars and the distance and cover of the reinforcing bars

3. Frame analysis

For optimal design of a frame the fitness of each design should be computed. For this purpose the internal forces including axial force, shear force and bending moments in each element of the frame are required. These structural response quantities are computed for each frame design via finite element analysis. In order to simplify the calculations, in the present study, only bending moments are considered for the beams while for columns both bending moments and axial forces are considered. The analysis of frame also consists of controlling the slenderness of the columns, and in case a section is recognized to be slender, then the moment magnification is made considered for that column.

3.1. Slenderness

When a column is regarded slender, the moment is magnified. ACI 318-08 code [40] states that for compression members, not braced against sidesway, the slenderness effects can be neglected when

$$\frac{kl_u}{r} < 22 \tag{4}$$

In this relation k is the effective length factor for compression members; l_u is the unbraced length of compression member; r is the radius of gyration of the cross section of a compression member.

The effective length factor of a column depends on the ratio of the stiffness of the columns to the stiffness of beams connected at the end of the compression member. This ratio at the end of a compression member can be expressed as

$$\psi = \frac{\sum (EI/l)_c}{\sum (EI/l)_b} \tag{5}$$

Where I is the moment of inertia considering the cracked section, E is the modulus of elasticity and l is the length of the beams or columns. Indices b and c respectively refer to beams and columns connected to the ends of a column.

After calculating ψ for two ends of each compressive member, the mean value of these values, ψ_m , is obtained and the coefficient of the effective length of the compression member, k, is calculated using the following relationships:

$$\psi_m < 2 : \quad k = (1 - 0.05\psi_m)\sqrt{1+\psi_m} \tag{6a}$$

$$\psi_m \geq 2 : \quad k = 0.9\sqrt{1+\psi_m} \tag{6b}$$

For a slender column, the magnified bending moment can be calculated as

$$M = M_{ns} + \delta_s . M_s \tag{7}$$

Where M_{ns} is the bending moment generated by the gravity loads and M_s is due to lateral load and δ_s is the moment magnification factor for frames not braced against sidesway. After determining the magnified moment separately for each end of a column, the biggest one is used to design the column. The calculation of the magnification factor δ_s is performed as indicated in Chapter 10 of the ACI 318-08 code.

4. Formulation of optimization

4.1. Objective function

The purpose of the optimal design of a RC frame is to minimize the total cost of the frame. Thus, the objective function is the total cost of all beams and columns due to individual cost components of concrete, steel and formwork. The cost of the material, fabrication and labor should be included in the cost of any component. Hence, the objective function of a RC frame can mathematically be stated as:

$$\text{Minimize:} \quad F = F_b + F_c \tag{8}$$

$$F_b = \sum_{beams} \{C_c.b.h.L + C_s.A_{st}.L.\gamma_s + C_f.L.(b+2h)\} \quad (9a)$$

$$F_c = \sum_{columns} \{C_c.b.h.L + C_s.A_{st}.L.\gamma_s + C_f.L.2(b+h)\} \quad (9b)$$

Subject to:

$$\text{For beams:} \quad \begin{cases} M_u^+ \le \phi M_n^+ \\ \left|M_{ul}^-\right| \le \left|\phi M_n^-\right| \\ \left|M_{ur}^-\right| \le \left|\phi M_n^-\right| \end{cases} \quad (10a)$$

$$\text{For columns:} \quad (M_u, P_u) \le (\phi M_n, \phi P_n) \quad (10b)$$

Where,

F = Total cost of all members of the frame ($);

F_b, F_c = Cost of all beams and columns, respectively ($);

C_c, C_f, C_s= Unit cost of concrete, formwork and steel, respectively;

b, h, L = Width, depth and length of the members (m);

A_{st} = Area of the reinforcing bars for each section (m²);

γ_s = Density of rebar (kg/m3);

M^+_u, M^-_{ul}, M^-_{ur} = Externally applied moment at mid-span, left and right joints of beams, respectively;

M^+_n, M^-_n =Nominal flexural strength at mid-span and joints of beams, respectively;

M_u, p_u = Externally applied moment and axial force of columns, respectively;

M_n, p_n = Nominal flexural and axial strength of columns, respectively;

In this study, the unit costs of the concrete, steel, formwork and the density of rebar are estimated as:

$$C_c=105\ \$/m^3,\ C_s=0.9\ \$/kg,\ C_f=92\ \$/m^2,\ \gamma_s=7850\ kg/m^3 \quad (11)$$

As mentioned before, the cost of the unit length of the beam and column sections, F_b and F_c, are calculated and stored in the last column of the corresponding database table.

4.2. Penalized objective function

In order to assess the fitness of a trial design and determine its distance from the global optimum, the eventual constraint violation should be computed by means of a penalty function. The penalty function consists of a series of geometric constraints corresponding to the dimensions and shape of the cross sections, and a series of constraint related to the deflection and internal forces of the members of the structure. Thus, penalty will be proportional to constraint violations, and the best design will have the minimum cost and no penalty. Geometric constraints are taken into account in the definition of the database of available beam and column profiles.

For the application of an optimization algorithm, the above constrained optimization problem should be transformed into an unconstrained one. For this purpose each internal force is normalized with respect to the corresponding member strength. For beams, three constraints are considered corresponding to the positive bending moments at the middle, and the negative bending moments at the two ends of the member:

$$g_1 = \frac{\left|M_u^+\right|}{\phi M_n^+} - 1 \ge 0 \quad (12a)$$

$$g_2 = \frac{\left|M_{ul}^-\right|}{\left|\phi M_n^-\right|} - 1 \ge 0 \quad (12b)$$

$$g_3 = \frac{\left|M_{ur}^-\right|}{\left|\phi M_n^-\right|} - 1 \ge 0 \quad (12c)$$

A column section is suitable and safe enough when the corresponding pair (M_u,P_u) under the applied loads does not fall outside the interaction diagram. In order to express this constraint in a mathematical form, the distance between the point representing the pair and the origin in the plane of interaction diagram is used, Fig. 3. Considering this figure, if the position of the pair is considered at B, and A is the crossing point of the line connecting B to the origin O and the interaction diagram, then the distance of the points A and B from O can easily be calculated. The ratio of these distances can be used as the constraint of the columns resistance. In order to specify the point A, the angle between line OB and the horizontal axis should be calculated as $\theta = \tan^{-1}(\frac{p_u}{M_u})$ and then considering the key points of the interaction diagram, it can be found out which line of the interaction diagram will be crossed by the line OB. In this way, if L_m and L_u are taken as the lengths of OA and OB, respectively, then we have

$$L_m = \sqrt{(\phi P_n)^2 + (\phi M_n)^2}\ ,\ L_u = \sqrt{(P_u)^2 + (M_u)^2} \quad (13)$$

Therefore the penalty function for the strength of the column can be expressed as:

$$g_4 = \frac{L_u}{L_m} - 1 \ge 0 \quad (14)$$

For columns, in addition to the strength requirements, another three constraints corresponding to the dimension and the number of reinforcing bars of the columns section which is located in the same line (the co-linear columns) are considered. This means that the dimensions of the top column should not be larger than those of the bottom one, and also the number of reinforcing bars in the top column should not be greater than that of the bottom column. If *T* and *B* represent the top column and bottom column, respectively, then these constraints can be expressed as follows:

$$\begin{cases} g_5 = \dfrac{b_T}{b_B} - 1 \ge 0 \\ g_6 = \dfrac{h_T}{h_B} - 1 \ge 0 \\ g_7 = \dfrac{n_T}{n_B} - 1 \ge 0 \end{cases} \quad (15)$$

Where n is the number of the reinforcing bars of the column sections, respectively. The total penalty of each design is determined by summing over different penalty terms for each element:

$$G = \sum_{beams}(g_1 + g_2 + g_3) + \sum_{columns}(g_4 + g_5 + g_6 + g_7) \tag{16}$$

The constrained optimization problem was transformed into an unconstrained optimization problem by collapsing the cost function and the penalty term according to literature:

$$\text{Minimize } F_p = F.(1+G)^{\varepsilon} \tag{17}$$

Where F_p is the penalized objective function, F is the cost function and ε is a parameter larger than 1 that depends on the structure type. The value ε =2 set in this study provided satisfactory results. After calculating the value of F_p for all the candidates, the optimization process is continued to obtain the optimal design using the following algorithms.

5. Meta-heuristic algorithms

5.1. Heuristic big bang-big crunch (HBB-BC) algorithm

The BB-BC algorithm as developed by Erol and Eksin [18] consists of two phases: a Big Bang phase, and a Big Crunch phase. In the Big Bang phase, candidate solutions are randomly distributed over the search space. Each candidate design is a possible design for the structure. The quality of each candidate design is evaluated by computing the penalty function. The first phase of the BB-BC algorithm ends when this evaluation is performed for all designs. In the Big Crunch phase, the centre of mass is defined for the population of candidate designs. In order to find the position of this centre, the mass of each candidate is considered to be proportional to the inverse of the corresponding penalized objective function. Therefore, the merit function is smaller for each candidate with small cost and low penalty, and such a candidate absorbs the mass centre towards itself. Thus, the centre of mass is located near the more qualified candidate designs. At this point, the BC stage is completed. In the new BB phase, a new population around the obtained centre of mass, produced in the previous BC stage, is formed. The BB and BC stages are sequentially repeated until the optimal design is obtained. In each iteration of the BB and BC, the search space shrinks until reaching the convergence to the optimum design. The BB-BC code utilized in this study is based on the classical formulation developed by Ref. [18]. The centre of mass is defined as:

$$\boldsymbol{X}_{cm} = \frac{\sum_{i=1}^{N} \frac{1}{F_{pi}} \boldsymbol{X}_i}{\sum_{i=1}^{N} \frac{1}{F_{pi}}} \tag{18}$$

Where X_{cm} is the position of the center of mass; X_i is the position of individual i; F_{pi} is the penalized objective function value of the individual i; and N is the population size.

The new position of the new population in the next iteration of Big Bang is obtained by a normal distribution around the centre of mass X_{cm} by the following relationship:

$$\boldsymbol{X}_i^{new} = \boldsymbol{X}_{cm} + \sigma \tag{19}$$

Where X_i^{new} is the position of the new individual i; σ is the standard deviation of a standard normal distribution and defined as:

$$\sigma = \frac{r\alpha(x_{\max} - x_{\min})}{s} \tag{20}$$

Where r is the random number from a standard normal distribution; α is the parameter not greater than 1 which limits the size of the search space around X_{cm}; x_{max} and x_{min} are the upper and lower limits on the values of the design variables; s is the number of explosions.

In order to improve the performance of the BB-BC, Camp [19] presented the following formula for producing the new candidate:

$$\boldsymbol{X}_i^{new} = \beta \boldsymbol{X}_{cm} + (1-\beta)\boldsymbol{X}_{best} + \frac{r\alpha(x_{\max} - x_{\min})}{s} \tag{21}$$

Where X_{best} is the best global solution of all the candidates obtained up to this stage of the iteration of BB-BC. The parameter β controls the effectiveness of the X_{best} in selecting the position of the new candidates.

Here, the variable X is considered as cost per unit length of each element and if the members of a structure are put in different groups, for each group, a centre of mass should be defined. The components of new vectors (candidate solutions), which fell outside the variables boundary, have to be regenerated in an alternative manner. Unlike the other papers which have used the BB-BC for optimization, here a new method is employed to handle these candidates. It is derived from harmony search (HS) algorithm [37]. The HS algorithm consists of some optimization operators, which will be discussed further. In the HBB-BC, only the harmony memory (HM) concept has been used. The best solutions obtained from each iteration of the BB-BC, is stored in a matrix as harmony memory. Thus, each component of the new candidates that violates its corresponding variables boundary, should be selected from HM in a random manner. Applying such a technique to the BB-BC can improve its performance and increase the convergence rate.

As it can be seen from Tables 1 and 2, the beam and column sections are ordered according to the cost of the unit length from smallest to the biggest. Thus for the group of beams we have x_{min}=133.95$ and x_{max}=305.79 $, and for the group of columns we have x_{min}=133.72 $ and x_{max}=499.48 $. The magnitudes of x_i^{new} for different groups of beams and columns can be attained. For each group of beams or columns, the member in the database of available structural elements with the cost per unit length closest to the computed one is selected as the new candidate design. The sorting of the available elements with respect to their cost per unit length, has facilitated the updating of design. The flowchart of the optimization by HBB-BC algorithm is shown schematically in Fig. 4.

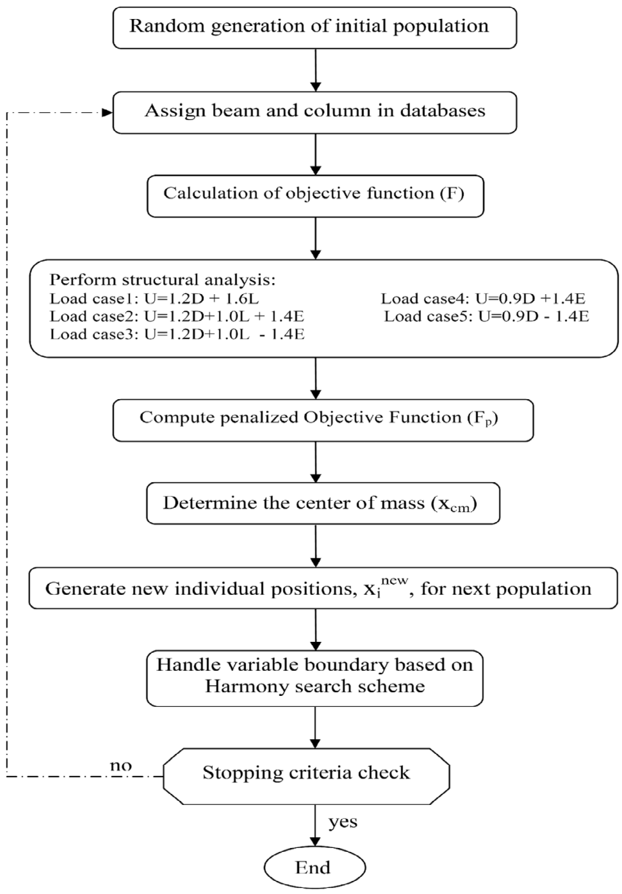

Fig. 4. Flowchart of the optimization by HBB-BC algorithm

5.2. Particle swarm optimization

The particle swarm optimization (PSO) was introduced by Eberhart and Kennedy [27], which was inspired by the social behavior of animals such as fish schooling and bird flocking. The standard PSO algorithm starts with a population (swarm) of random possible solutions of the optimization problem (particles). Each particle flies through the search space and its position is updated by the best position which is previously attained by the particle itself up to this iteration (local best) and by the best position among all adjacent particles (global best). This behavior mimics the cultural adaptation of a biological agent in a swarm: it evaluates its own position based on certain fitness criteria, compares to others, and imitates the best in the entire swarm [28].

In every iteration, the position of each particle is updated as follows:

$$\boldsymbol{V}_i^{k+1} = \omega \boldsymbol{V}_i^k + c_1 r_1 (\boldsymbol{P}_i^k - \boldsymbol{X}_i^k) + c_2 r_2 (\boldsymbol{P}_g^k - \boldsymbol{X}_i^k) \tag{22}$$

$$\boldsymbol{X}_i^{k+1} = \boldsymbol{X}_i^k + \boldsymbol{V}_i^{k+1} \tag{23}$$

Where, X_i^k and V_i^k are the current position and velocity of the ith particle, respectively; P_i^k and P_g^k represent the local best and global best, respectively; ω is an inertia weight to control the influence of the current velocity, r_1 and r_2 are two random numbers uniformly distributed in the range of (0,1) and c_1 and c_2 are two acceleration constants [29]. To improve the performance of standard PSO (SPSO), He *et al*. [30] introduced the passive congregation as an important biological force preserving swarm integrity. In PSOPC, the velocity is expressed as:

$$\boldsymbol{V}_i^{k+1} = \omega \boldsymbol{V}_i^k + c_1 r_1 (\boldsymbol{P}_i^k - \boldsymbol{X}_i^k) + c_2 r_2 (\boldsymbol{P}_g^k - \boldsymbol{X}_i^k) + c_3 r_3 (\boldsymbol{R}_i^k - \boldsymbol{X}_i^k) \tag{24}$$

Where R_i is a particle selected randomly from the swarm, c_3 is the passive congregation coefficient and r_3 is a uniform random sequence in the range (0, 1). Several benchmark functions have been tested in Ref. [30]. The results indicate that PSOPC has a better performance than SPSO.

5.3. Ant colony optimization

Ant colony optimization (ACO) was first suggested by Dorigo [32] as a multi-agent approach that simulates the ant foraging behavior to solve difficult combinatorial optimization problems, such as, the traveling salesman problem and the quadratic assignment problem. Ants are social insects whose behavior is directed more to the survival of the colony as a whole than to that of a single individual component of the colony. An important behavior of ant colonies is their foraging behavior, and in particular, how ants can find shortest paths between food sources and their nest. While walking from food sources to the nest and vice versa, ants deposit on the ground a substance called pheromone, forming in this way a pheromone trail. Ants can smell pheromone and when choosing their way, they tend to choose, in probability, paths marked by strong pheromone concentrations. The pheromone trail allows the ants to find their way back to the food source (or to the nest). Also, it can be used by other ants to find the location of the food sources found by their nest-mates. When more paths are available from the nest to a food source, a colony of ants will be able to exploit the pheromone trails left by the individual ants to discover the shortest path from the nest to the food source and back [33]. Further, the HPSACO uses the ACO capacities corresponding to local search and does not employ its formulation.

5.4. Harmony search algorithm

Harmony search (HS) algorithm is based on natural musical performance processes that occur when a musician searches for a better state of harmony, such as during jazz improvisation [37]. The engineers seek for a global solution as determined by an objective function, just like the musicians seek to find musically pleasing harmony as determined by an aesthetic [38]. The HS algorithm consists of some optimization operators, such as the harmony memory (HM), the harmony memory size (HMS), the harmony memory considering rate (HMCR), and the pitch adjusting rate (PAR). The components of new vectors (particles), which fell outside of the variables boundary, have to be regenerated in an alternative manner. For this purpose, HPSACO uses the harmony search algorithm. In this paper, only the harmony memory (HM) concept is used in the HPSACO and HBB-BC algorithms. The other operators have not been employed. Similar to the harmony memory in the HS algorithm, the PSOPC stores the feasible and "good" vectors (particles) in the local best matrix, i.e. P_i^k . Hence, each component of the new vector (particle) violating the variables' boundaries can be generated randomly again by such a technique-selecting for the components of different vectors in the local best swarm [31].

5.5. Heuristic particle swarm ant colony optimization (HPSACO)

Shelokar *et al*. [22] proposed the PSACO (particle swarm ant colony optimization) based on the common characteristics of both PSO and ACO algorithms. Like, survival as a swarm (colony) by coexistence and cooperation, individual contribution to food searching by a particle (an ant) by sharing information locally and globally in the swarm (colony) between particles (ants), etc. The implementation of the PSACO algorithm consists of two stages. In the first stage, it applies PSO, while ACO is implemented in the second stage. The ACO works as a local search, wherein, ants apply pheromone-guided mechanism to refine the positions found by particles in the PSO stage. In PSACO, a simple pheromone-guided mechanism of ACO is proposed to apply as local search. The proposed ACO algorithm handles *P* ants equal to the number of particles in the PSO [22].

Kaveh and Talatahari [23-26] have presented an optimization algorithm, so-called heuristic particle swarm ant colony optimization (HPSACO). They improved the performance of the PSACO, by adding harmony search scheme to PSACO as a handling approach to deal with variable boundaries. HPSACO has been used for discrete and continuous optimization of truss and steel frames problems.

In ACO stage, each ant generates a solution around P_g^k which can be expressed as:

$$\boldsymbol{Z}_i^k = N(\boldsymbol{P}_g^k, \sigma) \tag{25}$$

Based on the HPSACO, in Eq. (25), the N(P_g^k.σ) indicates a random number normally distributed with mean P_g^k and variance σ, where

$$\sigma = (x_{\max} - x_{\min}) \times \eta \tag{26}$$

Here, η is used to control the step size. The normal distribution with mean P_g^k can be considered as a continous pheromone which has the maximum in P_g^k and decreases with going away from it. In ACO algorithms, the probability of selecting a path with more pheromone is greater than other paths. Similarly, in the normal distribution, the probability of selecting a solution in the neighborhood of P_g^k is greater than the others. This principle is used in the HPSACO as a helping factor to guide the exploration and to increase the controlling in exploitation [24]. In this paper, by using the capacity of the BB-BC with regard to generation of new solution, we improved the performance of the HPSACO. Thus we used the standard deviation of a standard normal distribution, i.e. σ, under consideration in Eq. (20), instead of that in Eq. (26). Therefore, in ACO stage each ant generates a solution around P_g^k as follows:

$$\boldsymbol{Z}_i^k = \boldsymbol{P}_g^k + \frac{r\alpha(x_{\max} - x_{\min})}{s} \tag{27}$$

Where r is the random number from a standard normal distribution; α is the parameter limiting the size of the search space; x_{max} and x_{min} are the upper and lower limits on the values of the design variables; s is the number of ACO iterations.

In the present method, penalized objective function value $F_p(Z_i^k)$ is computed and the current position of ant i, Z_i^k , is replaced with the current position of particle i in the swarm, X_i^k; if $F_p(X_i^k)>F_p(Z_i^k)$. The pseudo-code for the HPSACO algorithm is listed in Table 3.

6. RC frame design examples

In order to demonstrate the efficiency of the algorithms described in this paper, three examples of RC plane frames are considered. These optimization examples consist of

- A three bay, four-story RC frame
- A three bay, eight-story RC frame
- A three bay, twelve-story RC frame

Loading cases acting on frames consist of joint loads and uniform distributed loads. Lateral equivalent static earthquake loads (E) are applied as joint loads, and uniform gravity loads are assumed for a dead load (D) and a live load (L). Five loading cases are considered as suggested in ACI 318-08 code [40] for strength design:

$$U=1.2D+1.6L \tag{28a}$$
$$U=1.2D+1.0L \pm 1.4E \tag{28b}$$
$$U=0.9D \pm 1.4E \tag{28c}$$

A uniform service dead load of D=22.3 kN/m, and a uniform service live load of L=10.7 kN/m were assumed in all the examples. The assumed specified compressive strength of concrete and yield strength of reinforcement bars in these examples were set to f_c' =23.5 and f_y =392MPa , respectively.

For the HBB-BC algorithm, our examinations showed that β=0.3 and α=0.7 are more suitable values. With these parameters, the optimal results and the speed of convergence were much higher. For the HPSACO algorithm, in all examples, the value of constants c_1 and c_2 are set 0.8 and the passive congregation coefficient c_3 is taken as 0.6. The value of inertia weight $\omega(k)$ decreases linearly from 0.75 in first iteration to 0.4. In the first example, the value of α existing in Eq. (27) is considered as 0.2 and 0.15 for beam groups and column groups, respectively, and in the other examples these are assumed as 0.3 and 0.2.

All optimization runs are carried out on a standard PC with a Pentium (R) Dual-Core CPU 2.60 GHz processor and 2.00 GB of RAM memory. The algorithms are coded in Matlab. Structures are analyzed via direct stiffness method. A typical stopping criterion of identical best solution for 20 last iterations is used for all examples in both algorithms.

Table 3. The pseudo-code for HPSACO

Initialize randomly all particles positions X_i^k and velocities V_i^k (from the range of [x_{min} ,x_{max}])
FOR(each particle i in the initial population)
WHILE(the constraints are violated)
Randomly re-generate the current particle Xi
END WHILE
Generate local best: assign $P_i^k = X_i^k$
Generate global best: Find minimum of $F_p(X_i^k)$, assign $P_g^k = X_{\min}^k$
END FOR
WHILE(stopping criterion not satisfied)
FOR(each particle (ant) i in the swarm(colony))
Generate the velocity and update the position of the current particle (vector) X_i^k
Variable boundary handling: Check whether each component of the current vector violates its corresponding boundary or not. If it does, select the corresponding component of the vector from P_i^k based on harmony search scheme.
Evaluate the merit value $F_p(X_i^k)$ of the current particle
Generate P solutions Z_i^k using Eq. (27)
Variable boundary handling: Check whether each component of the current vector violates its corresponding boundary or not. If it does, select the corresponding component of the vector from P_i^k based on harmony search scheme.
Evaluate the merit value $F_p(Z_i^k)$ of the current ant
Update current particle position: Compare the merit value of current ant with current particle. If $F_p(Z_i^k) < F_p(X_i^k)$, assign $F_p(X_i^k) = F_p(Z_i^k)$ and $X_i^k = Z_i^k$
Update local best: Compare the merit value of $F_p(P_i^k)$ with $F_p(X_i^k)$,
If $F_p(X_i^k) < F_p(P_i^k)$, assign $P_i^k = X_i^k$
END FOR
Update global best: Find the global best position in the swarm.
if $F_p(X_i^k) < F_p(P_g^k)$, assign $P_g^k = X_i^k$
END WHILE

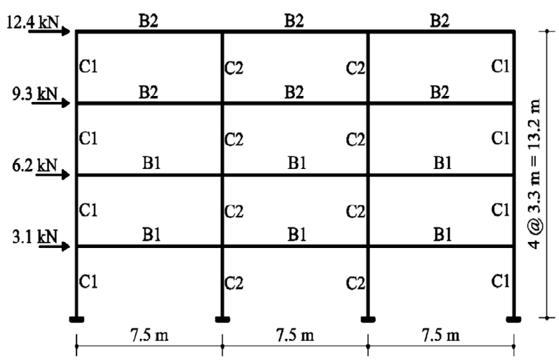

Fig. 5. Three-bay, four-story reinforced concrete frame

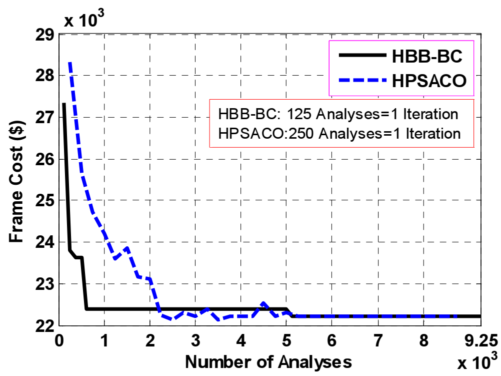

Fig. 6. Convergence rate comparison between the two algorithms for the three-bay, four-story reinforced concrete frame

6.1. A three bay, four-story reinforced concrete frame

The geometry, loading and grouping details are shown in Fig. 5. The frame has a total of 28 members, 12 beams and 16 columns. All elements are arranged into four groups; two groups for beams and two groups for columns. This frame is subjected to gravity and lateral loads. A population size of 125 is used for both algorithms. An optimum cost of 22207 $ is obtained by the HBB-BC and HPSACO after 74 and 35 iterations, respectively. Table 4 compares the results obtained

Table 4. Result of optimum design for three bay, four-story reinforced concrete frame

		OPTIMIZATION RESULTS							
		HPSACO				HBB-BC			
		Sectional dimensions		Reinforcements		Sectional dimensions		Reinforcements	
Member type	Element group	Width (mm)	Depth (mm)	Positive moment	Negative moment	Width (mm)	Depth (mm)	Positive moment	Negative moment
Beam	B1	300	500	3-D19	5-D22	300	500	3-D19	5-D22
	B2	300	500	4-D19	5-D22	300	500	4-D19	5-D22
Column	C1	350	350	8-D25		350	350	8-D25	
	C2	300	300	6-D25		300	300	6-D25	
Number of iterations		35				74			
Number of analyses		8500				9250			
Computing time (second)		32.67				35			
Frame cost		22207 $				22207 $			

by these two algorithms . Though HBB-BC and HPSACO obtained the same solution, but the convergence rate of the HBB-BC is better than HPSACO. Fig. 6 provides a comparison of convergence rates of the HBB-BC and HPSACO. The demand/capacity ratio (DCR), i.e. the maximum of ($M_u/\phi M_n$) for beams, and the maximum of (L_u/L_m) for columns, in the members of the optimum solution obtained by HBB-BC for the three bay, four-story RC frame is given in Fig. 7. Based on this figure, the using of section capacity in all beams is high, but in some columns is low. Also Table 5 demonstrates the maximum values of demand/capacity ratio under critical loading case for member groups in the optimum solutions obtained by both algorithms. For the HBB-BC algorithm, the ratio of the sampling space to the entire search space is

$(125\times74) / (1014\times1014\times55\times55)=2.97\times10^{-6}$.

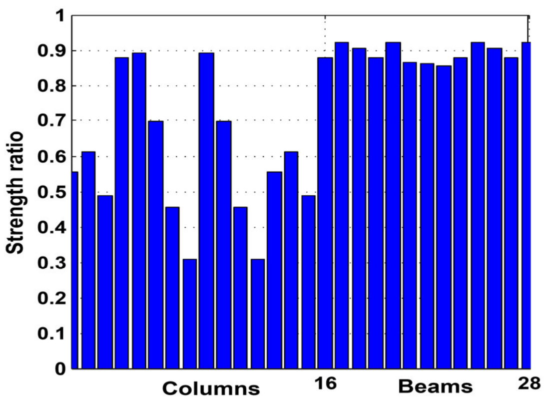

Fig. 7. Strength ratio in the members of the optimum solution obtained by HBB-BC for the three bay, four-story reinforced concrete frame

Table 5. Maximum strength ratio for member groups in the three bay, four-story RC frame

Member type	Element group	Strength ratio HPSACO	Strength ratio HBB-BC	Critical load case
Beam	B1	0.92136	0.92136	Load case 1
	B2	0.92232	0.92232	Load case 1
Column	C1	0.87963	0.87963	Load case 1
	C2	0.89030	0.89030	Load case 1

6.2. A three bay, eight-story reinforced concrete frame

Fig. 8 illustrates the geometry, loading and grouping details of a three bay, eight-story RC frame to be optimized. The frame is composed of 56 elements, 24 beams and 32 columns, which are divided into three beam groups and four column groups. The gravity and lateral loads are applied on this frame. A population size of 250 is considered for both algorithms. HBB-BC and HPSACO achieved to optimum costs of 48263 $ and 48514 $ after 158 and 106 iterations, respectively. Table 6 provides a comparison of the obtained results by these two algorithms. Unlike the first example, here the HPSACO could not find the optimum result as well as HBB-BC. A comparison of convergence rates of HBB-BC and HPSACO is given in Fig. 9. The demand/capacity ratio for beams and columns of the optimum solution attained by HBB-BC for this RC frame is demonstrated in Fig. 10. From

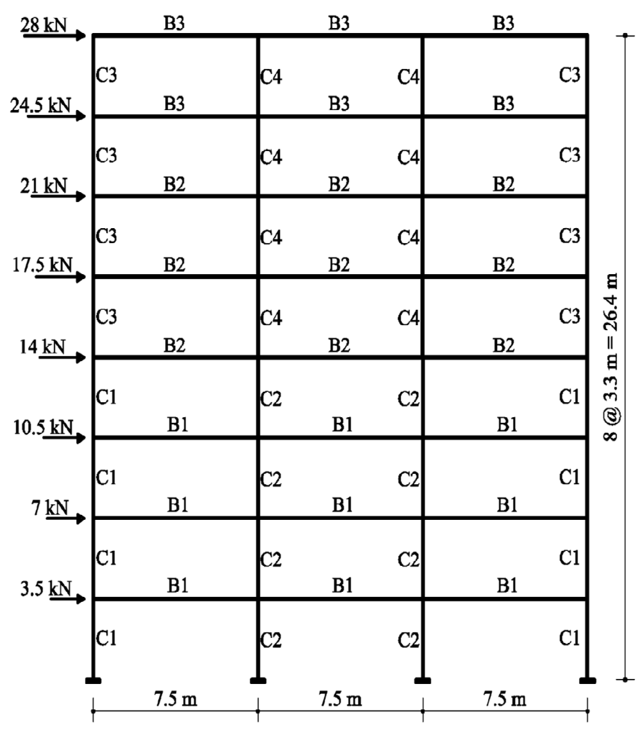

Fig. 8. Three-bay, eight-story reinforced concrete frame

Table 6. Result of optimum design for three bay, eight-story reinforced concrete frame

		OPTIMIZATION RESULTS							
		HPSACO				HBB-BC			
		Sectional dimensions		Reinforcements		Sectional dimensions		Reinforcements	
Member type	Element group	Width (mm)	Depth (mm)	Positive moment	Negative moment	Width (mm)	Depth (mm)	Positive moment	Negative moment
Beam	B1	300	500	3-D19	6-D22	300	500	3-D19	6-D22
	B2	300	500	3-D19	6-D22	300	500	3-D19	6-D22
	B3	300	500	3-D19	5-D22	300	500	3-D19	5-D22
Column	C1	400	400	8-D25		400	400	8-D25	
	C2	500	500	8-D25		450	450	12-D25	
	C3	350	350	8-D25		350	350	8-D25	
	C4	350	350	8-D25		350	350	8-D25	
Number of iterations		106				158			
Number of analyses		52500				39500			
Computing time (minute)		11.23				8.5			
Frame cost		48514 $				48263 $			

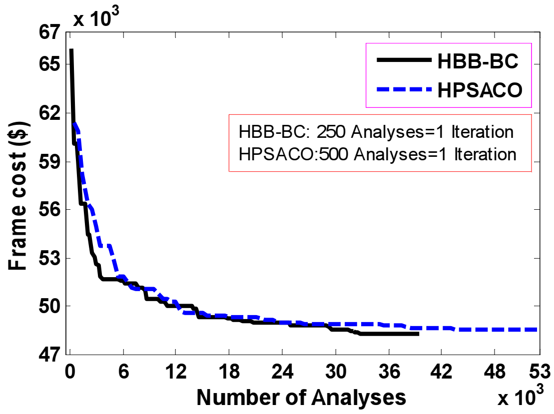

Fig. 9. Convergence rate comparison between the two algorithms for the three-bay, eight-story reinforced concrete frame

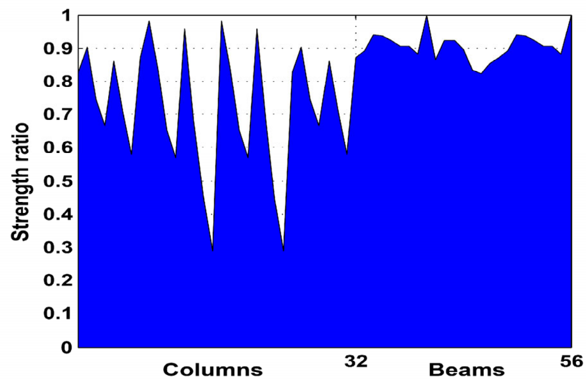

Fig. 10. Strength ratio in the members of the optimum solution obtained by HBB-BC for the three bay, eight-story reinforced concrete frame

this figure, it can be observed that with the exception of a few columns, the using of section capacity in the majority of elements is high. As well the maximum demand/capacity ratio under critical loading case for member groups in the optimum solutions achieved by both algorithms is shown in Table 7. In the HBB-BC method, the order of sampling space relative to domain space is $(250\times158)/[(1014)^3\times(55)^4] = 4.14\times10^{12}$.

Table 7. Maximum strength ratio for member groups in the three bay, eight-story RC frame

Member type	Element group	Strength ratio HPSACO	Strength ratio HBB-BC	Critical load case
Beam	B1	0.94039	0.94402	Load case 2
	B2	0.92315	0.91293	Load case 2
	B3	0.99783	0.99725	Load case 1
Column	C1	0.90198	0.92365	Load case 3
	C2	0.98237	0.96384	Load case 2
	C3	0.87317	0.88409	Load case 1
	C4	0.95716	0.92880	Load case 2

6.3. A three bay, twelve-story reinforced concrete frame

Fig. 11 represents the geometry, loading and grouping details of a three bay, twelve-story RC frame to be optimized. The frame is consisted of 84 elements, 36 beams and 48 columns, which are collected in three beam groups and six column groups. A population size of 300 is considered for both algorithms. The HBB-BC and HPSACO achieved to optimum costs of 81183 $ and 83250 $ after 182 and 108 iterations, respectively. Table 8 lists the optimal solutions achieved by two algorithms, and compares them with each other. Unlike the first example, here the HPSACO could not find the optimum result as well as HBB-BC. Fig. 12 provides a comparison of convergence rates of HBB-BC and HPSACO. The demand/capacity ratio for beams and columns of the optimum solution attained by HBB-BC for this RC frame is demonstrated in Fig. 13. From this figure, it can be noticed that with the exception of a few columns, the using of section capacity in the majority of elements is high. In addition, the maximum DCR under critical loading case for member groups in the optimum solutions attained by these two algorithms is shown in Table 9. In the HBB-BC method, the order of sampling space relative to domain space was $(300\ 182)/[(1014)^3\ (55)^6]= 1.89\ 10^{-15}$. In this example, for HPSACO method almost the half of computing time is consumed to find the feasible first population, it means if frame is large and a great loading applies on it, finding the feasible first population lasts.

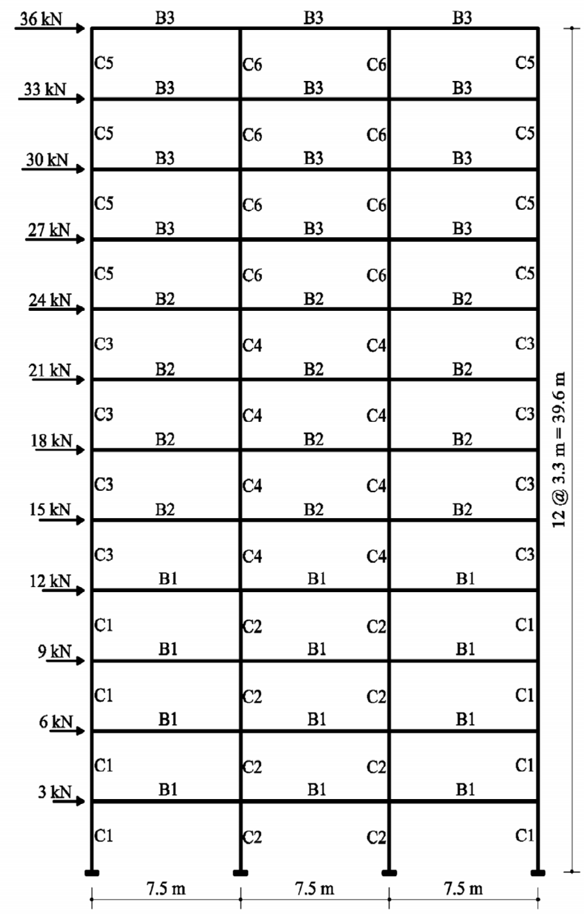

Fig. 11. three-bay, twelve-story reinforced concrete frame

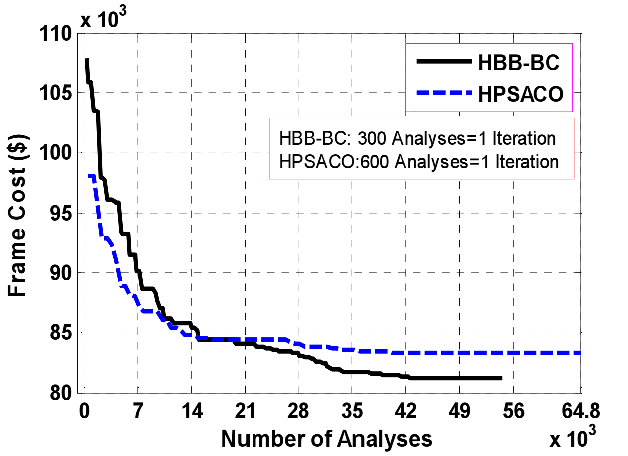

Fig. 12. Convergence rate comparison between the two algorithms for three-bay, twelve-story reinforced concrete frame

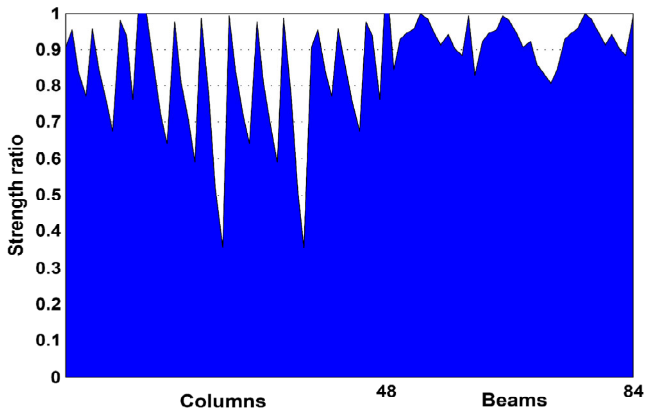

Fig. 13. Strength ratio in the members of the optimum solution obtained by HBB-BC for the three bay, twelve-story reinforced concrete frame

7. Conclusions

In this paper, the HBB-BC and HPSACO algorithms are employed for optimal design of reinforced concrete planar frames. The HBB-BC is a combination of the big bang-big crunch algorithm and harmony search scheme. In this method, each component of new vectors (candidate solutions) generated by the BB-BC, which violated the variables boundary, must be regenerated based on harmony search. The performance and the convergence rate of the HBB-BC are better than BB-BC. HPSACO is based on PSOPC, ACO and HS. In this algorithm, the ACO helps PSO

Table 8. Result of optimum design for three bay, twelve -story reinforced concrete frame

		OPTIMIZATION RESULTS							
		HPSACO				HBB-BC			
		Sectional dimensions		Reinforcements		Sectional dimensions		Reinforcements	
Member type	Element group	Width (mm)	Depth (mm)	Positive moment	Negative moment	Width (mm)	Depth (mm)	Positive moment	Negative moment
Beam	B1	350	550	3-D19	7-D22	350	600	3-D19	6-D22
	B2	350	550	3-D19	6-D22	300	550	5-D19	6-D22
	B3	350	550	3-D19	5-D22	300	500	3-D19	6-D22
Column	C1	500	500	8-D25		450	450	10-D25	
	C2	650	650	10-D25		600	600	12-D25	
	C3	450	450	8-D25		400	400	10-D25	
	C4	500	500	10-D25		500	500	10-D25	
	C5	350	350	6-D25		350	350	8-D25	
	C6	400	400	4-D25		400	400	4-D25	
Number of iterations		108				182			
Number of analyses		64500				54600			
Computing time (minute)		55				23			
Frame cost		83250				81138 $			

Table 9. Maximum strength ratio for member groups in the three bay, twelve-story RC frame

Member type	Element group	Strength ratio HPSACO	Strength ratio HBB-BC	Critical load case
Beam	B1	0.89430	0.95647	Load case 2
	B2	0.99323	0.99891	Load case 2
	B3	0.93151	0.99304	Load case 1
Column	C1	0.92511	0.95275	Load case 3
	C2	0.99069	0.99307	Load case 3
	C3	0.86719	0.95708	Load case 3
	C4	0.94699	0.97570	Load case 2
	C5	0.99411	0.86842	Load case 1
	C6	0.99629	0.98703	Load case 2

procedure in the global exploration phase and HS is employed for variables boundary handling. In this paper, by using the capacity of the BB-BC with regard to generation of new solution, the performance of the HPSACO is improved.

According to other research which used the HPSACO to the optimization of structures, the performance of the HPSACO is significantly better than other PSO-based algorithms, such as SPSO, PSOPC and HPSO. In this paper, it is observed that if the frame is large, HPSACO can not find a good result as well as the HBB-BC. Furthermore, if the frame is large and a great loading is applied on it, finding the feasible first population for the HPSACO increases the computational cost. In fact, in each iteration of the HPSACO, two populations were analyzed, one for the PSO stage and another for theACO stage. Though in HPSACO, the required numbers of iterations to convergence are less than that of the HBB-BC, however in large frames, certainly the required number of analyses is more than that of the HBB-BC, and this results in the increase of the computational cost, as well.

Based on the results of the four, eight and twelve story RC frames, specially the last two examples, although the order of sampling space relative to domain space was small, however the HBB-BC is obtained optimal or near optimal design. This shows the robustness of the HBB-BC. It should also be mentioned that working with HBB-BC is much easier than HPSACO, since programming is easier and also it has no excessive parameters for control. There are only two parameters α and β , and one can easily find suitable values for these parameters with a few runs of the program.

Acknowledgment: The first author is grateful to the Iran National Science Foundation for the support.

References

[1] Adamu A, Karihaloo BL, Rozvany GIN. Minimum cost design of reinforced concrete beams using continuum-type optimality criteria. Struct. Optim., 1994;7:91-102.

[2] Moharrami, H. , Grierson, D.E., Computer automated design of reinforced concrete frameworks. J. Struct. Eng.ASCE, 1993; 119 (7), 2036-2058.

[3] Zielinski ZA, Long W, Troitsky MS. Designing reinforced concrete short-tied columns using the optimization technique. ACI Struct. J., 1995;92(5):619-626.

[4] Fadaee MJ, Grierson DE. Design optimization of 3D reinforced concrete structures. Struct. Optim., 1996;12: 127-134.

[5] Balling RJ, Yao X. Optimization of reinforced concrete frames. J. Struct. Eng., 1997;123(2):193-202.

[6] Sarma, K.C. , Adeli, H., Cost optimization of concrete structures. ASCE J. Struct. Eng., 1998, 124 (5), 570-578.

[7] Rajeev S, Krishnamoorthy CS. Genetic algorithm-based methodology for design optimization of reinforced concrete frames. Computer-Aided Civil Infrastruct. Eng., 1998;13: 63-74.

[8] Ferreira CC, Barros MHFM, Barros AFM. Optimal design of reinforced concrete T-sections in bending. Eng. Struct., 2003;25(7):951-964.

[9] Lee C, Ahn J. Flexural design of reinforced concrete frames by genetic algorithm. J. Struct. Eng., 2003;129(6):762-774.

[10] Camp CV, Pezeshk S, Hansson H. Flexural design of reinforced concrete frames using a genetic algorithm J. Struct. Eng., 2003;129(1):105-115.

[11] Govindaraj V, Ramasmy JV. Optimum detailed design of reinforced concrete continuous beams using genetic algorithms. Comput. Struct., 2005;84(1):34-48.

[12] Govindaraj V, Ramasmy JV. Optimum detailed design of reinforced concrete frames using genetic algorithms. Eng. Optim., 2007;39(4):471-494.

[13] Sahab MG, Ashour AF, Toropov VV. Cost optimization of reinforced concrete flat slab buildings. Eng. Struct., 2004;27:313-322.

[14] Sahab MG, Ashour AF, Toropov VV. A hybrid genetic algorithm for reinforced concrete flat slab buildings. Comput. Struct., 2005;83:551-559.

[15] Choi, C.K. and Kwak, H.G., Optimum RC member design with predetermined discrete sections. ASCE J. Struct. Eng., 1990, 116 (10), 2634-2655.

[16] Kwak HG, Kim J. Optimum design of reinforced concrete plane frames based on predetermined section database. Comput. Aided Design, 2008;40(3):396-408.

[17] Kwak HG, Kim J. An integrated genetic algorithm complemented with direct search for optimum design of RC frames. Comput. Aided Design, 2009; 41:490-500

[18] Erol OK, Eksin I. A new optimization method: Big Bang-Big crunch Adv. Eng. Software, 2006;37:106-111.

[19] Camp CV. Design of space trusses using Big Bang-Big Crunch optimization. J Struct. Eng, ASCE, 2007;133(7):999-1008

[20] Kaveh A, Talatahari S. Size optimization of space trusses using Big Bang-Big Crunch algorithm. Comput. Struct., 2009;87:1129-1140

[21] Kaveh A, Talatahari S. Optimal design of Schwedler and ribbed domes via hybrid Big Bang_Big Crunch algorithm. Constr. Steel Res., 2010;66(3):412-419.

[22] Shelokar PS, Siarry P, Jayaraman VK, Kulkarni BD. Particle swarm and ant colony algorithms hybridized for improved continuous optimization. Appl. Math. Comput., 2007;188:129-142.

[23] Kaveh A, Talatahari S. A hybrid particle swarm and ant colony optimization for design of truss structures. Asian J. Civil Eng., 2008; 9(4):329-348.

[24] Kaveh A, Talatahari S. Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures. Comp. Struct. ,2009;87(5-6):267-283.

[25] Kaveh A, Talatahari S. A discrete particle swarm ant colony optimization for design of steel frames. Asian J. Civil Eng., 2008;9(6):563-575.

[26] Kaveh A, Talatahari S. A particle swarm ant colony optimization for truss structures with discrete variables. Constr. Steel Res. 2009;65(8-9):1558-1568.

[27] Eberhart RC, Kennedy J. A new optimizer using particle swarm theory. In: Proceedings of the sixth international symposium on micro machine and human science, Nagoya, Japan; 1995.

[28] Kennedy J, Eberhart RC, Shi Y. Swarm intelligence. San Francisco (CA): Morgan Kaufman Publishers; 2001.

[29] Shi Y, Eberhart RC. A modified particle swarm optimizer. In: Proceedings of IEEE international conference on evolutionary

computation; 1997.
[30] He S, Wu QH, Wen JY, Saunders JR, Paton RC. A particle swarm optimizer with passive congregation. Biosystem, 2004;78:135-147.
[31] Li LJ, Huang ZB, Liu F, Wu QH. A heuristic particle swarm optimizer for optimization of pin connected structures. Comp. Struct., 2007;85:340-349.
[32] Dorigo M. Optimization, learning and natural algorithms (in Italian), PhD Thesis. Dipartimento di Elettronica, Politecnico di Milano, IT; 1992.
[33] Dorigo M, Caro DG, Gambardella LM. Ant algorithms for discrete optimization. Artif Life, 1999;5(3):137-172.
[34] Kaveh A, Shojaee S. Optimal design of skeletal structures using ant colony optimization. Int. J. Numer. Meth. Eng., 2007;5(70):563-581.
[35] Kaveh A, Farahmand Azar B, Talatahari S. Ant colony
[35] optimization for design of space trusses. Int. J. Space Struct., 2008;23(3):167-181.
[36] Camp C, Bichon J. Design of space trusses using ant colony optimization. J Struct Eng ASCE, 2004;130(5):741-751.
[37] Geem ZW, Kim JH, Loganathan GV. A new heuristic optimization algorithm: harmony search. Simulation, 2001;76:60-68.
[38] Lee KS, Geem ZW. A new structural optimization method based on the harmony search algorithm. Comput. Struct. 2004;82:781-798.
[39] Saka MP. Optimum design of steel sway frames to BS5950 using harmony search algorithm. Constr. Steel Res., 2009;65(1):36-43.
[40] American Concrete Institute (ACI). Building code requirements for structural concrete and commentary. ACI 318-08, 2008.