Paper 71

©Civil-Comp Press, 2012
Proceedings of the Eleventh International Conference
on Computational Structures Technology,
B.H.V. Topping, (Editor),
Civil-Comp Press, Stirlingshire, Scotland

Design Optimization of Moment Resisting Steel Frames using a Cuckoo Search Algorithm

M.P. Saka[1] and E. Doğan[2]
[1] Department of Civil Engineering
University of Bahrain, Isa Town, Bahrain
[2] Civil Engineering Department
Celal Bayar University, Manisa, Turkey

Abstract

The Cuckoo search algorithm is a recent addition to metaheuristic techniques. It simulates the breeding behaviour of certain cuckoo species into a numerical optimization technique. Cuckoo birds lay their eggs in the nests of other host birds so that their chicks when hatched can be nurtured by the host birds. The optimum design algorithm presented for moment resisting steel frames is based on the cuckoo search algorithm. The design algorithm selects the appropriate W sections for the beams and column of a steel frame out of 272 W sections listed in the LRFD-AISC (Load and Resistance Factor Design, American Institute of Steel Construction) [52] such that the code requirements are satisfied and the weight of steel frame is the minimum. Code specifications necessitate the consideration of a combined strength constraint with lateral torsional buckling for beam-column members. Furthermore displacement constraints as well as inter-storey drift restrictions of multi-storey frames are also included in the design formulation. Further constraints related with the constructability of a steel frame are also considered. The number of steel frames are designed by the algorithm presented to demonstrate its efficiency. The same steel frames are also designed by using the big bang-big crunch algorithm as well as the particle swarm optimizer for comparison.

Keywords: structural optimization, discrete optimization, metaheuristic search techniques, steel frames, swarm intelligence, cuckoo search algorithm, particle swarm optimizer.

1 Introduction

The improvements took place in the design optimization techniques based on metaheuristic algorithms in recent years have helped engineers tremendously in finding the solutions of computationally intractable design problems [1-5]. By means of these techniques it became possible to determine the solution of discrete structural optimization problems more efficiently than with those based on mathematical programming methods. The fundamental properties of metaheuristic algorithms are that they imitate certain strategies taken from nature, social culture, biology or laws of physics that guide the search process. Their goal is to efficiently explore the search space using these guiding mechanisms in order to find near optimal solutions. They also use some strategies to avoid getting trapped in confined areas of search space. Furthermore they do not even require an explicit relationship between the objective function and the constraints. The mechanisms used in search of the optimum solution are not deterministic but stochastic. They are not problem specific and proven to be very efficient and robust in obtaining the solution of practical engineering design optimization problems with both continuous and discrete design variables. Several algorithms have been developed that are based on these techniques. A detailed review of these applications are carried out in [6,7]. After the successful applications of early metaheuristic techniques such as genetic algorithms, evolutionary strategies and simulated annealing in structural optimization, number of new metaheuristic algorithms have been emerged in recent years which are even more efficient and robust than the earlier techniques.

Particle swarm optimizer is one of the recent algorithms that is based on swarm intelligence [8,9]. In nature fish school, birds flock and bugs swarm not only for reproduction but for other reasons such as finding food and escaping predators. There are implicit rules that each member of bird flock and fish school has to abide by so that they can move in a synchronized way without colliding. Each individual in a flock maintains optimum distance from the neighboring individuals so that the flock can move smoothly from one place to another. Particle swarm optimizer is a simulator of social behavior that is used to realize the movement of a birds' flock. It is population based optimization algorithm. Its population is called a swarm and each individual in the swarm is called a particle. Each particle flies through the problem space to search for optimum. Particle swarm optimization has been successfully applied to in developing structural optimization algorithms [10-20].

Harmony search method is another recent addition to metaheuristic techniques [21-25]. This algorithm was inspired by the observation of music improvisation. Trying to find a pleasing harmony in a musical performance is analogous to finding the optimum solution in an optimization problem. The aim of the musician is to procedure a piece of music with harmony. Similarly a designer intends to determine the best solution in an optimization problem under the given objective and limiting constraints. Both have the same target; to determine the best. Harmony search method is widely applied in structural design optimization since its emergence. These applications have shown that harmony search algorithm is robust, effective and reliable optimization method [26-39].

Big Bang-Big Crunch algorithm is developed by Erol and Eksin [40] which is population based algorithm similar to genetic algorithm and particle swarm optimizer. It consists of two phases as its name implies. First phase is the big bang in which the randomly generated candidate solutions are randomly distributed in the search space. In the second phase these points are contracted to a single representative point that is the weighted average of randomly distributed candidate solutions. First application of the algorithm in the optimum design of steel frames is carried out by Camp [41] which is followed by other applications [42-46].

Cuckoo search algorithm is also one of the very recent (2010) metaheuristic technique which simulates the breeding behavior of certain cuckoo species into a numerical optimization technique [47]. Cuckoo birds lay their eggs in the nests of other host birds sometimes removing the host's eggs to provide more possibility of hatching their own eggs. Cuckoo search algorithm is population based metaheuristic technique and its basic steps are as follows. Initially population of nests is randomly generated and each nest contains number of eggs representing potential solution to the design problem. Each nest is evaluated by computing its objective function value which is its fitness. Among all nests the best one is selected. Depending on the quality of the new eggs host eggs are replaced. Some cuckoo eggs are discovered by host birds and thrown out of nests. These are replaced by new eggs. These steps are continued until termination criteria is satisfied. Comparison of cuckoo search algorithm with some of other metaheuristic techniques is carried out in [48] and it is stated that the performance of cuckoo search algorithm in obtaining the optimum solution of selected benchmark optimization problems is the best. Application of the cuckoo search algorithm in structural design optimization has shown that it yield lighter optimum designs with less number structural analysis compare to other algorithms [49,50,51].

In this study the optimum design problem of moment resisting steel frames is formulated according to LRFD-AISC (Load and Resistance Factor Design, American Institute of Steel Construction) [52]. Design constraints include the displacement limitations, inter-storey drift restrictions of multi-storey frames, strength requirements for beams and beam-columns. Furthermore, additional constraints are considered to satisfy practical requirements. These include three types of inequalities. The first type ensures that the flange width of the beam section at each beam-column connection of each storey is less than or equal to the flange width of column section. The second and third type of constraints are required to be included to make sure that the depth and the mass per meter of column section at each storey at each beam-column connection are less than or equal to width and mass of the column section at the lower storey. The design problem turns out to be discrete nonlinear programming problem. Cuckoo search optimization technique is employed to determine the optimum solution. The same design problems are also solved by big bang-big crunch algorithm as well as particle swarm optimizer. The optimum results obtained by the three different metaheuristic techniques are compared to evaluate their performance.

2 Discrete Optimum Design of Steel Frames to LRFD-AISC

The design process of moment resisting steel frames necessitates selection of steel profile sections for its columns and beams from a standard steel section tables. This selection should be carried out in such a way that the frame with the selected steel sections satisfy the serviceability and strength requirements specified by the code of practice while the economy is observed in the overall or material cost of the frame.

When the constraints are implemented from LRFD–AISC in the formulation of the design problem the following discrete programming problem is obtained.

Find a vector of integer values $\mathbf{I}$ (Eqn. 1) representing the sequence numbers of steel sections assigned to ng member groups

$$\mathbf{I}^T = \left[I_1, I_2, ..., I_{ng}\right] \tag{1}$$

to minimize the weight (W) of the frame

Minimize $$W = \sum_{k=1}^{ng} m_k \sum_{i=1}^{nk} L_i \tag{2a}$$

Subject to

$$(\delta_j - \delta_{j-1})/h_j \le \delta_{ju} \qquad j = 1,...., ns \tag{2b}$$

$$\delta_i \le \delta_{iu} \qquad i = 1,...., nd \tag{2c}$$

$$V_u \le \phi V_n \tag{2d}$$

$$\left(\frac{P_u}{\phi_c P_n}\right)_{il} + \left(\frac{8}{9}\frac{M_{ux}}{\phi_b M_{nx}}\right)_{il} \le 1.0 \quad for \quad \frac{P_u}{\phi_c P_n} \ge 0.2$$

$$\left(\frac{P_u}{2\phi_c P_n}\right)_{il} + \left(\frac{M_{ux}}{\phi_b M_{nx}}\right)_{il} \le 1.0 \quad for \quad \frac{P_u}{\phi_c P_n} \le 0.2 \tag{2e}$$

$$B_{sc} \le B_{sb} \qquad s = 1,...., nu \tag{2f}$$

$$D_s \le D_{s-1} \tag{2g}$$

$$m_s \le m_{s-1} \tag{2h}$$

where Eq. (2a) defines the weight of the frame, ng is total numbers of groups in the structural system, m_k is the unit weight of the steel section selected from the standard steel sections table for group k, L_i is the length of member i which belongs to group k, nk is total number of members in group k.

Eq. (2b) represents the inter-storey drift of the multi-storey frame. δ_j and δ_{j-1} are lateral deflections of two adjacent storey levels and h_j is the storey height. ns is the total number of storeys in the frame.

Eq. (2c) defines the displacement restrictions that may be required to include other than drift constraints such as mid-span deflections of beams. nd is the total number of restricted displacements in the frame. δ_{ju} is the allowable lateral displacement. The horizontal deflection of columns is limited due to unfactored imposed load and wind loads to height of column / 300 in each storey of a building with more than one storey. δ_{iu} is the upper bound on the deflection of beams which is given as (span / 300) if they carry plaster or other brittle finish.

Eq. (2d) represents the shear capacity check for beam-columns. ϕ is resistance factor in shear, V_u required shear strength, V_n is nominal shear strength.

Eq. (2e) defines the local capacity check for beam-columns. nm is number of members, nl is number of load cases, M_{nx} is nominal flexural strength, M_{ux} is applied moment, P_n is nominal axial strength, P_u is applied axial load, $Ø_c$ is resistance factor for columns if the axial force is in compression, $Ø_b$ is resistance factor in bending. It is apparent that computation of compressive strength $\phi_c P_n$ of a compression member requires its effective length. The computation of the effective length of a compression member in a frame can be automated by using Jackson and Moreland monograph [53]. The relationship for the effective length of a column in a swaying frame is given as:

$$\frac{(\gamma_1\gamma_2)(\pi/k)^2-36}{6(\gamma_1+\gamma_2)}=\frac{\pi/k}{\tan(\pi/k)} \tag{3}$$

where k is the effective length factor and γ_1 and γ_2 are the relative stiffness ratio for the compression member which are given as:

$$\gamma_1=\frac{\sum I_{c1}/\ell_{c1}}{\sum I_{b1}/\ell_{b1}} \quad \text{and} \quad \gamma_2=\frac{\sum I_{c2}/\ell_{c2}}{\sum I_{b2}/\ell_{b2}} \tag{4}$$

The subscripts c and b refer to the compressed and restraining members respectively and the subscripts 1 and 2 refer to two ends of the compression member under investigation. The solution of the nonlinear equation (3) for k results in the effective length factor for the member under consideration. The Eqn. (3) has the following form for non-swaying frames.

$$\frac{\gamma_1\gamma_2}{4}\left(\frac{\pi}{k}\right)^2+\left(\frac{\gamma_1+\gamma_2}{2}\right)\left(1-\frac{\pi/k}{\tan(\pi/k)}\right)+\frac{2\tan(\pi/2k)}{\pi/k}=1 \tag{5}$$

Eq. (2f) is included in the design problem to ensure that the flange width of the beam section at each beam-column connection of storey s should be less than or equal to the flange width of column section.

Eqs. (2g) and (2h) are required to be included to make sure that the depth and the mass per meter of column section at storey s at each beam-column connection are less than or equal to width and mass of the column section at the lower storey $s-1$. nu is the total number of these constraints.

3 Cuckoo Search Algorithm

Cuckoo search algorithm is originated by Yang and Deb [47] which simulates reproduction strategy of cuckoo birds. Some species of cuckoo birds lay their eggs in the nests of other birds so that when the eggs are hatched their chicks are fed by the other birds. Sometimes they even remove existing eggs of host nest in order to give more probability of hatching of their own eggs. Some species of cuckoo birds are even specialized to mimic the pattern and color of the eggs of host birds so that host bird could not recognize their eggs which gives more possibility of hatching. In spite of all these efforts to conceal their eggs from the attention of host birds, there is a possibility that host bird may discover that the eggs are not its own. In such cases the host bird either throws these alien eggs away or simply abandons its nest and builds a new nest somewhere else. In cuckoo search algorithm cuckoo egg represents a potential solution to the design problem which has a fitness value. The algorithm uses three idealized rules as given in [47]. These are: a) each cuckoo lays one egg at a time and dumps it in a randomly selected nest. b) the best nest with high quality eggs will be carried over to the next generation. c) the number of available host nests is fixed and a host bird can discover an alien egg with a probability of $p_a \in [0,1]$. In this case the host bird can either throw the egg away or abandon the nest to build a completely new nest in a new location. The pseudo code of the cuckoo search algorithm is given in Fig. 1.

Cuckoo search algorithm initially requires selection of a population of n eggs each of which represents a potential solution to the design problem under consideration. This means that it is necessary to generate n solution vector of $\boldsymbol{x} = \{\boldsymbol{x_1}, \dots\dots\dots, \boldsymbol{x_{ng}}\}^{\boldsymbol{T}}$ in a design problem with ng variables. For each potential solution vector the value of objective function $f(\boldsymbol{x})$ is also calculated. The algorithm then generates a new solution $\boldsymbol{x}_i^{\nu+1} = \boldsymbol{x}_i^{\nu} + \beta\lambda$ for cuckoo i where $\boldsymbol{x}_i^{\nu+1}$ and $\boldsymbol{x}_i^{\nu}$ are the previous and new solution vectors. $\beta > 1$ is the step size which is selected according to the design problem under consideration. $\boldsymbol{\lambda}$ is the length of step size which is determined according to random walk with Levy flights. A random walk is a stochastic process in which particles or waves travel along random trajectories consists of taking successive random steps. The search path of a foraging animal can be modeled as random walk. A Levy flight is a random walk in which the steps are defined in terms of the step-lengths which have a certain probability distribution, with the directions of the steps being isotropic and random. Hence Levy flights necessitates selection of a random direction and generation of steps under chosen Levy distribution.

Cuckoo Search Algorithm

begin

Initialize a population of n host nests $x_i,\ i = 1, 2, ..., n$;

while (until the termination criterion is satisfied);

Get a cuckoo randomly, (let it be x_i)

and generate a new solution by Levy flights;

Evaluate its fitness (let it be F_i);

Choose a nest among n nests randomly, (let it be x_j);

if $(F_i > F_j)$

replace x_j by the new solution x_i;

end

Abandon a fraction (P_a) of worse nests and

built new ones at new locations via levy flights;

Keep the best nests (or solutions);

Rank the solutions and find the current best;

end while

Post process results;

end

Figure 1. Pseudo code for Cuckoo search algorithm

Mantegna [54] algorithm is one of the fast and accurate algorithm which generates a stochastic variable whose probability density is close to Levy stable distribution characterized by arbitrary chosen control parameter α $(0.3 \le \alpha \le 1.99)$. Using the Mantegna algorithm, the step size **λ** is calculated as

$$\lambda = \frac{x}{|y|^{1/\alpha}} \tag{6}$$

where x and y are two normal stochastic variables with standard deviation σ_x and σ_y which are given as

$$\sigma_x(\alpha) = \left[\frac{\Gamma(1+\alpha)\sin(\pi\alpha/2)}{\Gamma((1+\alpha)/2)\alpha 2^{(\alpha-1)/2}}\right]^{1/\alpha} \quad \text{and} \quad \sigma_y(\alpha) = 1 \text{ for } \alpha = 1.5 \tag{7}$$

in which the capital Greek letter Γ represents the gamma function that is the extension of the factorial function with its argument shifted down by 1 to real and complex numbers. That is if k is a positive integer $\Gamma(k) = (k-1)!$.

4 Cuckoo Search Based Optimum Design Algorithm with Discrete Variables

The solution of the discrete design problem given in Eqns. 2(a-h) is obtained using the cuckoo search algorithm. The optimum design technique based on cuckoo search method treats the sequence number of the steel sections in the standard W-section list as a design variable as mentioned in section 2. For this purpose complete set of 272 W-sections starting from W100x19.3 to W1100x499mm as given in LRFD-AISC is considered as a design pool from which the optimum design algorithm selects W-sections for frame members. Once a sequence number is selected, then the sectional designation and properties of that section becomes available from the section table for the algorithm. Consequently the design vector consists of integer numbers between 1 to 272 which corresponds to the sequence numbers of W-sections in the discrete set. The cuckoo search based optimum design algorithm consists of the following steps.

1. Select the values of cuckoo search parameters. These are the total number of host bird's nests (n), step size parameter β, p_a probability of cuckoo egg being discovered by host bird and maximum number of iteration to terminate the design process.
2. Generate host nests. Select randomly sequence number of steel sections from the discrete list for each group in the frame.

$$\mathrm{I}_0^{\mathrm{i}} = \mathrm{I}_{\min} + \mathrm{r}\left(\mathrm{I}_{\max} - \mathrm{I}_{\min}\right) \qquad I = \left[I_1, I_2, .., I_{ng}\right] \qquad i = 1, ..., n \tag{8}$$

   where the term r represents a random number between 0 and 1, I_{min} is equal to 1 and I_{max} is the total number of values in the discrete set respectively. ng is the total number of design variables.

3. Carry out the analysis of the steel frame with these sections and check whether the design limitations are satisfied or not. If this nest violates the design constraints severely discard it and repeat the selection of a new nest. If it is slightly infeasible consider it for better solutions.

4. Check whether the newly selected nest is acceptable. If not, go to step 3.

5. After the selection of acceptable nests, calculate the objective function value for each nest. Determine the one which has the best objective function value.

6. Get new solution for each cuckoo randomly by levy flights using Eq. (9)

$$I_{\ell+1}^{i} = I_{\ell}^{i} + \beta \times \mathrm{Levy} \tag{9}$$

$$\mathrm{Levy} = \mathrm{r} \times \lambda \times \left(\mathrm{I}_{\ell}^{\mathrm{i}} - \mathrm{I}_{\mathrm{b}}^{\mathrm{i}}\right) \tag{10}$$

Where β symbolizes the step size which is assumed to be 0.1 for this study. λ refers to step length, r is random number from standard normal distribution, I_b^i is the position of best nest so far.

7. Analyze the steel frame with these sections. If it is acceptable then compare it with the one randomly selected from the population. If it produces better design then replace it with this new solution.

8. Depending on *pa* probability parameter, find out whether each nest keep its current position Eq.(11). ***R*** matrix stores 0 and 1 values such that anyone of them is assigned to each component of *i-th* nest, in which 0 means that current position is kept and 1 implies that the current position is to be updated.

$$\boldsymbol{R}^i \leftarrow \begin{cases} 1 & if \ rand < pa \\ 0 & if \ rand \geq pa \end{cases} \tag{11}$$

This is conducted by means of Eq. (12).

$$\mathrm{I}_{\ell+1}^{\mathrm{i}} = \mathrm{I}_{\ell}^{\mathrm{i}} + r \times \boldsymbol{R}^{\mathrm{i}} \times \left(\mathrm{P}erm1^{\mathrm{i}} - \mathrm{Perm2}^{\mathrm{i}} \right) \tag{12}$$

Where, r is random number between 0 and 1. *Perm1* and *Perm2* are two row permutations of the corresponding nest. R defines the probability matrix.

9. Perform analysis of steel frame with these sections. If all the constraints are satisfied, compare this design with the older one. If a better design is produced then replace this nest with older one. Otherwise, continue with the current design.

10. Repeat step 6 - 9 until a predefined number of iterations is reached. This number is selected large enough such that within this number of design cycles no further improvement is observed in the objective function.

4.1 Constraint handling

Most structural optimization problems include problem-specific constraints, which are difficult to solve using the traditional mathematical optimization algorithms. Penalty functions have been commonly used to deal with constraints. However, the major disadvantage of using the penalty function is that some tuning parameters are added in the algorithm and the penalty coefficients have to be tuned in order to balance the objective and penalty functions. If appropriate penalty coefficients cannot be provided, difficulties will be encountered in the solution of the optimization problems. To avoid such difficulties, a new method, called 'fly-back mechanism', was developed in [11] for particle swarm optimizer. For most of the optimization problems containing constraints, the global minimum locates on or close to the boundary of a feasible design space. The particles are initialized in the feasible region. When the optimization process starts, the particles fly in the feasible space to search the solution. If any one of the particles flies into infeasible region, it will be forced to fly back to the previous position to guarantee a feasible solution. The particle which flies back to the previous position may be closer to the boundary at the next iteration. This makes the particles to fly to the global minimum in a great probability. Therefore, such a 'fly-back mechanism' technique is suitable for handling the optimization problem containing the constraints. Compared with the other constraint handling techniques, this method is relatively simple and easy to implement. Some experimental results have shown that it can find a better solution with a fewer iterations than the other techniques [11].

In this study fly-back mechanism is used for handling the design constraints. Once all nests (I_i) are generated, the objective functions are evaluated for each of these and the constraints in the problem are then computed with these values to find out whether they violate the design constraints. If one or a number of the nests give infeasible solutions, these are discarded and new ones are re generated. If a nest is slightly infeasible then such nests are kept in the solution. These particles having one or more constraints slightly infeasible are utilized in the design process that might provide a new nest that may be feasible. This is achieved by using larger error values initially for the acceptability of the new design vectors and then reduce this value gradually during the design cycles and finally to use an error value of 0.001 or whatever necessary value that is required to be selected for the permissible error term towards the end of iterations. This adaptive error strategy is found quite effective in handling the design constraints in large design problems.

5 Design Examples

Three moment resisting steel frames are designed using cuckoo search method based optimum design algorithm presented in the previous section. The modulus of elasticity is taken as 200kN/mm^2 in all the examples. Cuckoo search algorithm parameters total number of nests, step size and the fraction of the worst nests (*pa)* are selected as 40, 0.1, 0.35 respectively in the design examples considered.The maximum number of iterations is taken as 50000 as to give equal opportunity to each technique to reach the optimum designs. However the optimum designs are attained in much less number of iterations than the maximum number of iterations as shown in the tables of the design examples.

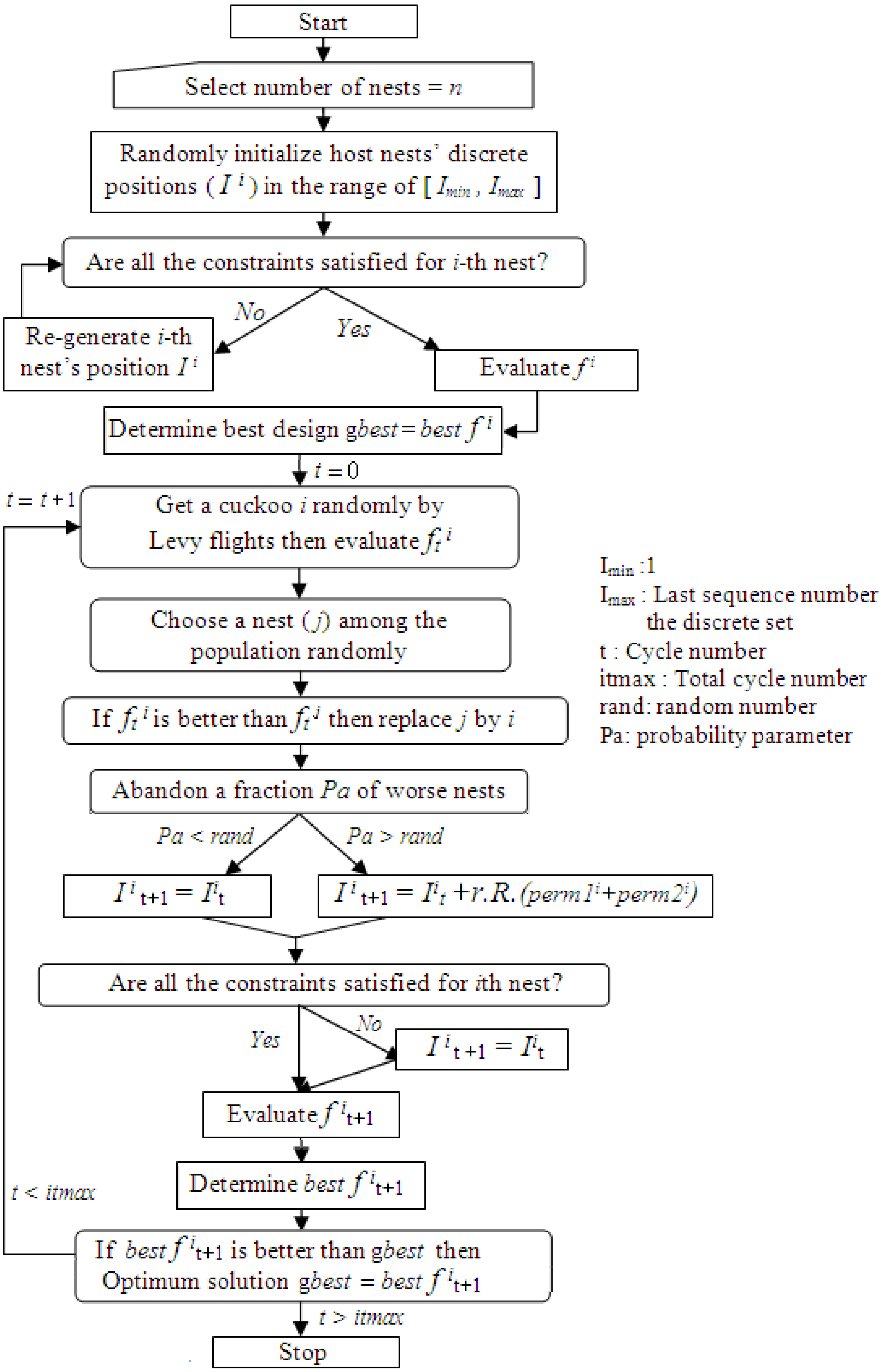

Figure 2. Flow chart of cuckoo based optimum design algorithm

5.1 Six-Storey, Two-bay Steel Frame

The six storey two-bay steel frame shown in Fig. 3 is considered as first design example. The frame consists of thirty members that are collected in eight groups as shown in the figure. The allowable inter-storey drift is 1.17cm while the lateral displacement of the top storey is limited to 7.17cm. The frame is designed by three different optimum design algorithms that are based on three different metaheuristic algorithms. These are cuckoo search algorithm, particle swarm optimizer and big bang-big crunch algorithm

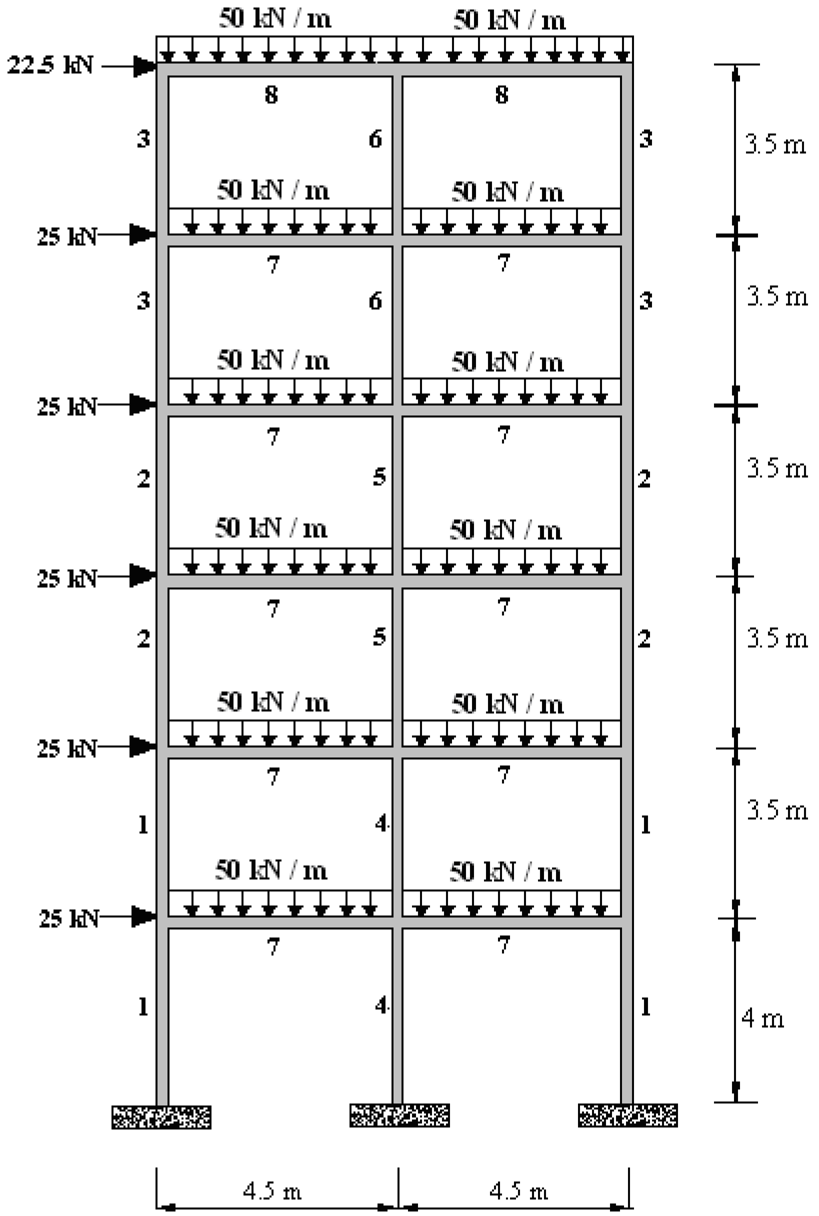

Figure 3. Six-storey, two-bay moment resisting steel frame

Group No.	Member Type	CSO	PSO	BBBC
1	Column	W460X82	W530X74	W460X106
2	Column	W410X53	W310X52	W410X60
3	Column	W310X38.7	W200X41.7	W250X49.1
4	Column	W610X82	W460X89	W360X64
5	Column	W530X66	W460X89	W360X64
6	Column	W150X29.8	W360X72	W200X41.7
7	Beam	W460X60	W460X60	W460X60
8	Beam	W460X52	W460X68	W460X60
Max. Int. St. Drift Ratio		0.77	0. 78	0.78
Maximum Strength Ratio		0.94	0.99	0.98
Top storey drift (cm)		4.421	4.5325	4.654
Minimum Weight. kg (kN)		6970.60 (68.358)	7532.11 (73.865)	7583.56 (74.369)
Number of Str. Analysis		18000	6890	9250

Table 1. Optimum designs for six-storey, two-bay steel frame

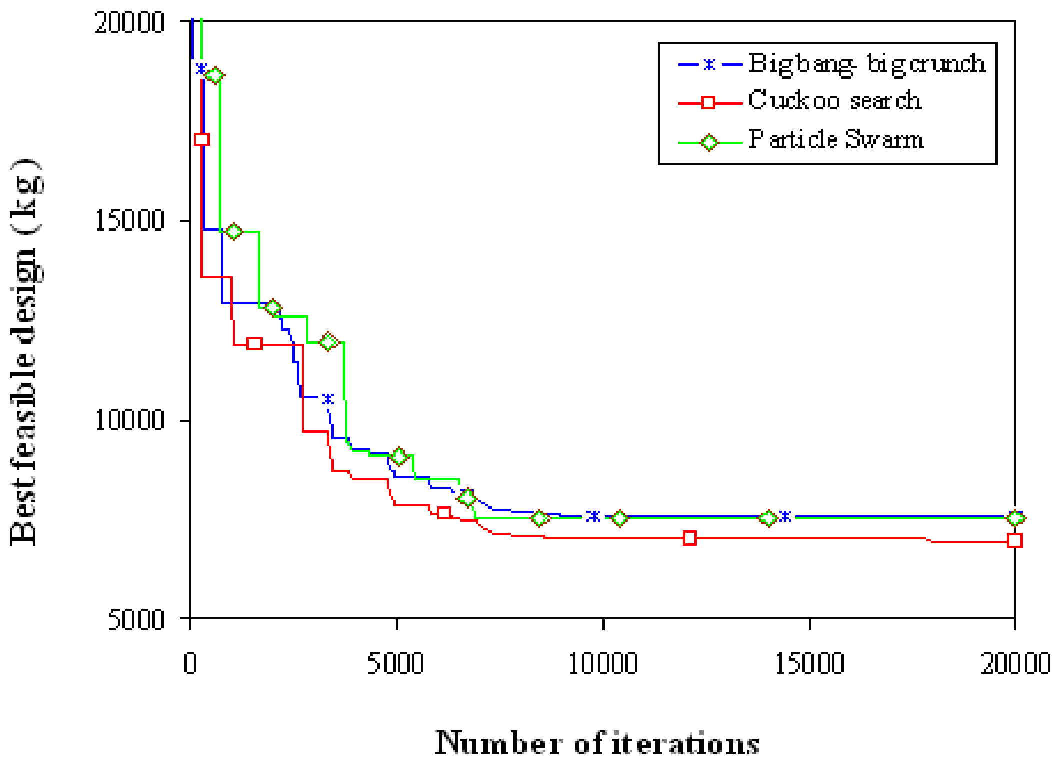

Figure 4. Design histories of three different metaheuristic methods for six-storey, two-bay steel frame

The optimum W-section designations for each group obtained by three metaheuristic algorithms are given in Table 1. The optimum design attained by the cuckoo search algorithm is the lightest among three optimum designs having 6970.60kg of minimum weight. This weight is 8% and 8.8% less than the ones obtained by particle swarm and big bang-big crunch algorithms. It is noticed that in all three designs inter-storey as well as top storey drift values are much less than their upper bounds while the ultimate strength constraints are around 0.99 which is very close to their upper bound 1. This clearly indicates that strength constraints dominate the design. It is apparent from Table 1 that cuckoo search algorithm required more structural analysis than particle swam and big bang-big crunch algorithms to reach the optimum design. It should be worthwhile to mention that the cuckoo search algorithm used in this study is the standard one not the improved version. The design history of each metaheuristic algorithm is illustrated in Fig. 4. It is apparent from the figure that cuckoo search algorithm has better convergence rate compare to other two algorithms. Therefore, it may be concluded that cuckoo search algorithm has demonstrated better performance in the optimum design of six storey, two bay steel moment resisting frame.

5.2 Ten-Storey, Three-Bay Frame

Ten-storey, three-bay steel frame shown in Fig. 5 is considered as second design example which consists of seventy members that are collected in nine groups. The dimensions and loading of the frame is shown in the figure. The allowable inter-storey drift is 1.17cm while the lateral displacement of the top storey is limited to 11.83cm. The ultimate strength and geometric constraints are implemented as they are described in section 2. This frame is also designed by using particle swarm optimizer and big bang-big crunch algorithm in addition to cuckoo search technique for comparison.

The optimum W-section designations for each group obtained by three metaheuristic algorithms are given in Table 2. The optimum design attained by the cuckoo search algorithm is the lightest among three optimum designs having 22477.59kg of minimum weight. This weight is 1.8% and 2.7% less than the ones obtained by particle swarm and big bang-big crunch algorithms. It is noticed that in all three designs inter-storey as well as top storey drift values are much less than their upper bounds while the ultimate strength constraints are equal to 1 which is their upper limits. This clearly indicates that strength constraints dominate the design. The design histories of three different metaheuristic algorithms are given in Fig. 4. It is apparent from the figure that cuckoo search algorithm has better convergence rate compare to other two algorithms. Once again cuckoo search approach required more structural analysis than particle swarm and big bang-big crunch algorithms. This may be due to the fact that in this study the standard cuckoo search technique is used in the optimum design algorithm. It is quite probable that the improved version of the same technique may find the optimum design requiring less number of structural analysis than the standard cuckoo search method. Therefore, it may be concluded that cuckoo search algorithm demonstrates better performance compare to particle swarm and big bang-big crunch algorithms in finding the optimum design of six storey, two bay steel moment resisting frame.

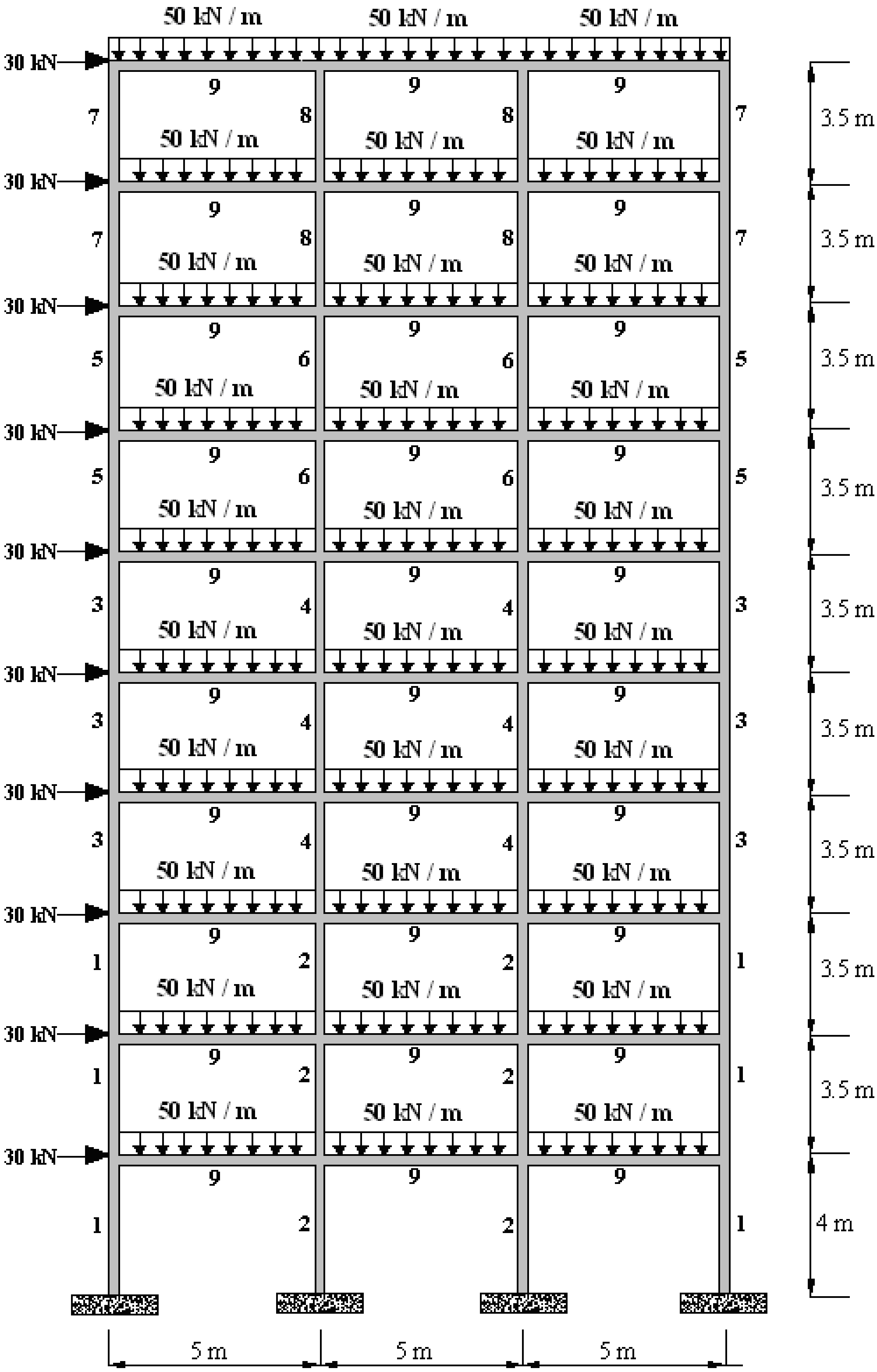

Figure 5. Ten-storey, three-bay moment resisting steel frame

Group No.	Member Type	CSO	PSO	BBBC
1	Column	W530X150	W610X153	W840X176
2	Column	W610X113	W610X113	W460X97
3	Column	W530X92	W530X92	W610X92
4	Column	W460X82	W460X82	W460X89
5	Column	W200X59	W310X60	W310X60
6	Column	W410X53	W410X53	W310X60
7	Column	W200X59	W310X60	W310X60
8	Column	W410X53	W410X53	W310X60
9	Beam	W530X66	W460X68	W530X66
Max. Int. St. Drift Ratio		0. 82	0. 87	0.79
Maximum Strength Ratio		1.00	1.00	1.00
Top storey drift (cm)		7.45	7.86	7.14
Minimum Weight. kg (kN)		22477.59 (220.429)	22879.35 (224.369)	23079.73 (226.334)
Number of Str. Analysis		18000	13600	13100

Table 2. Optimum designs for ten-storey, three-bay steel frame

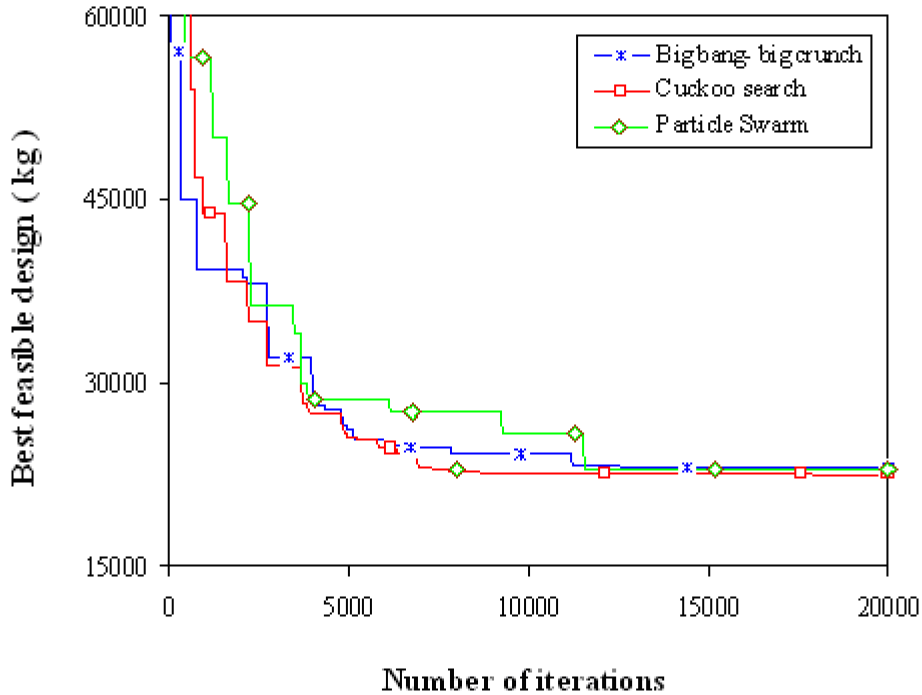

Figure 6. Design histories of three different metaheuristic methods for ten-storey, three-bay steel frame

5.3 Fifteen-Storey, Three-bay Steel Frame

The three-bay, fifteen-storey frame shown in Figure 7 is considered as third design example. The dimensions and the external loading of the frame are shown in the figure. The frame is subjected to gravity loading of 12.4kN/m on the beams of roof level and 20kN/m on the beams of each floor. The lateral loading is the wind loading. Frame consists of 105 members that are collected in 12 groups. Inner columns and outer columns in every three storey considered to be different groups. The beams of roof and intermediate floors are considered to be two different groups as shown in the figure. The allowable inter-storey drift is 1.17cm while the lateral displacement of the top storey is limited to 17.67cm. The strength capacities of steel members are computed according to LRFD-AISC as explained in section 2.

This frame also designed by particle swarm as well as big bang-big crunch algorithms in addition to cuckoo search approach. The W-sections designations obtained by the three techniques in the optimum designs are shown in Table 3. The minimum weight of the frame is determined as 22902.10kg by the cuckoo search method, 29092.81kg by the particle swarm algorithm and 29003.48kg by the big bang-big crunch approach. The optimum design attained by the cuckoo search algorithm is 27% and 26.6% less than the ones obtained by the particle swarm and big bang-big crunch techniques. This clearly indicates the fact that the cuckoo search method performs better than the other two in larger size optimum design problem. It is noticed that in the optimum frame while the drift constraints are not at their bounds the ultimate strength constraint are almost equal to 1. It is expected for inter-storey drift constraints to reach to their upper limits in the design of such a tall frame. However, it should be remembered that geometric constraints which forces the design to have larger W-sections at the lower storey than the one at the upper storey produces heavier design. This in turn makes the ultimate strength constraints active in the design process not displacement constraints.

Group No.	**Member Type**	**CSO**	**PSO**	**BBBC**
1	Beam	W250X38.5	W410X46.1	W310X32.7
2	Beam	W410X46.1	W410X46.1	W410X46.1
3	Column	W310X38.7	W410X38.8	W250X67
4	Column	W310X38.7	W410X38.8	W250X67
5	Column	W410X53	W460X52	W460X89
6	Column	W410X67	W460X193	W530X138
7	Column	W460X113	W530X196	W610X140
8	Column	W200X41.7	W250X32.7	W250X49.1
9	Column	W410X46.1	W410X60	W360X51
10	Column	W410X46.1	W410X60	W360X51
11	Column	W460X60	W460X60	W530X109
12	Column	W530X92	W690X170	W690X125
Max. Int. St. Drift Ratio		0.76	0.64	0.63
Maximum Strength Ratio		0.98	0.99	0.98
Top storey drift (cm)		10.35	8.59	8,76
Minimum Weight. kg (kN)		22902.10 (224.59)	29092.81 (285.301)	29003.48 (284.425)
Number of Strc. Analysis		16800	12800	12600

Table 3. Optimum designs for fifteen-storey, three-bay steel frame

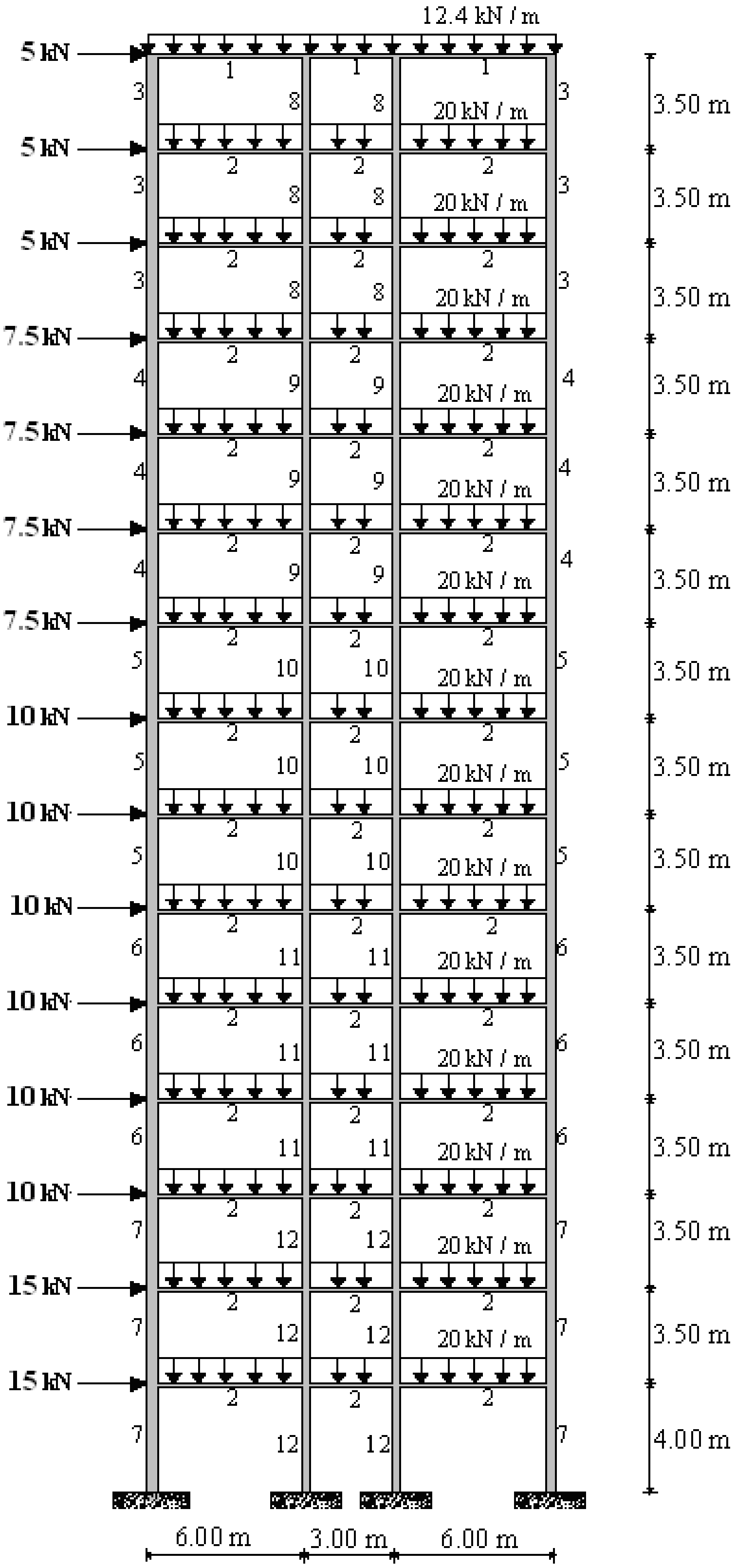

Figure 7. Fifteen–storey, three-bay moment resisting steel frame

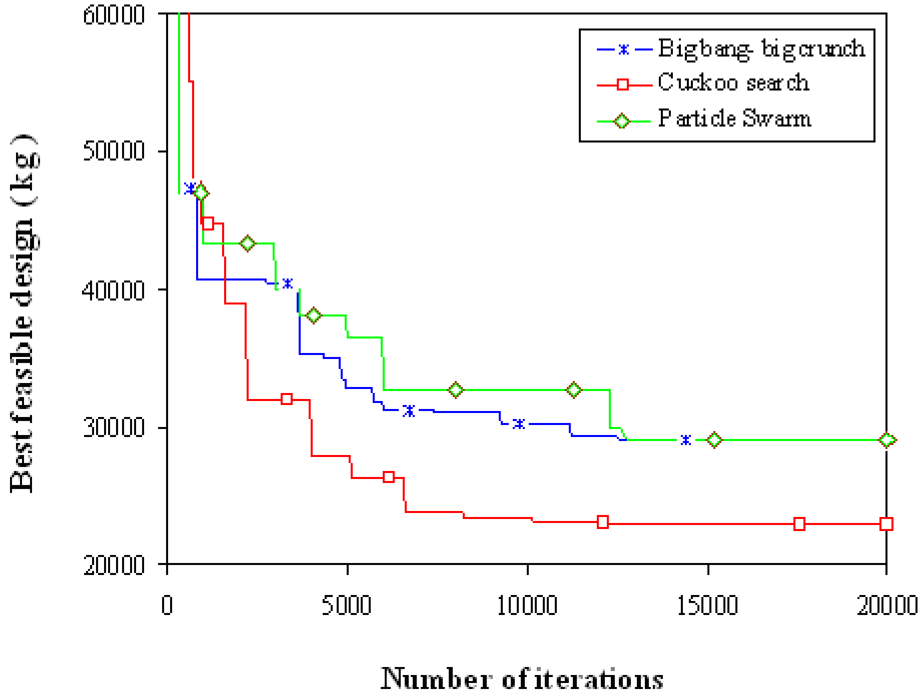

Figure 8. Design histories of three different metaheuristic methods for fifteen-storey, three-bay steel frame

Design histories of three algorithms are shown in Fig. 8. It is very clear from the figure that cuckoo search method has a better convergence rate and reached lighter optimum design than the other ones attained by particle swarm and big bang-big crunch algorithm. Hence it can be concluded that cuckoo search algorithm is reliable, robust and efficient algorithm that can be effectively used to obtain the optimum design of steel moment resisting frames.

6 Conclusions

The optimum design algorithm developed is based on cuckoo search method which selects the optimum W-section designations from W-sections table for the beams and columns of a moment resisting steel frame such that design constraints described in LRFD-AISC are satisfied and the frame has the minimum weight. In view of the results obtained it is concluded that the cuckoo search method is an efficient and robust technique that can successfully be used in optimum design of steel frames and determines lighter optimum solutions compare to particle swarm and big bang-big crunch methods. In the optimum design of fifteen storey, three bay frame, the optimum frame obtained by the cuckoo search approach is 27% lighter than the one attained by the other two metaheuristic techniques. Furthermore, the cuckoo search method basically has only one parameter to be specified by a user which is the total number of nests. This provides a robustness to the algorithm compared to many other metaheuristic techniques that require pre-determination of more parameters. Selection of these parameters most of the time are problemdependent. It is further noticed that the adaptive error strategy combined with fly-back mechanism for constraint handling increases the efficiency of the cuckoo search method and removes the necessity of selecting the value of penalty coefficient in the penalty function method.

Acknowledgement

Authors would like to thank Dr. Xie-She Yang for providing the computer code for the cuckoo search method.

References

[1] G. A. Kochenberger and F. Glover, “Handbook of Meta-Heuristics”, Kluwer Academic Publishers, 2003

[2] J. Dreo, A. Petrowski, P. Siarry and E. Taillard, “Meta-Heuristics for Hard Optimization”, Springer-Verlag, Berlin, Heidelberg, 2006

[3] X-S. Yang, “Nature-Inspired metaheuristic Algorithms”, Luniver Press, 2008.

[4] X-S. Yang, “Engineering Optimization: An Introduction with Metaheuristic Applications”, John Wiley, 2010.

[5] S. Luke, “Essentials of Metaheuristics”, http://cs.gmu.edu/~sean/book/metaheuristics/, 2010.

[6] M.P. Saka, "Optimum Design of Skeletal Structures: A Review", in B.H.V. Topping, (Editor), "Progress in Civil and Structural Engineering Computing", Saxe-Coburg Publications, Stirlingshire, UK, Chapter 10, pp 237-284, 2003. doi:10.4203/csets.10.10

[7] M.P. Saka, "Optimum Design of Steel Frames using Stochastic Search Techniques Based on Natural Phenomena: A Review", in B.H.V. Topping, (Editor), "Civil Engineering Computations: Tools and Techniques", Saxe-Coburg Publications, Stirlingshire, UK, Chapter 6, pp 105-147, 2007. doi:10.4203/csets.16.6

[8] J. Kennedy, R. Eberhart and Y. Shi, “Swarm Intelligence”, Morgan Kaufmann Publishers, 2001.

[9] S. He, Q. H. Wu, J. Y. Wen, J. R. Saunders and R. C. Paton, “A Particle Swarm Optimizer with Passive Congregation”, BioSystems, 78, 135-147, 2004

[10] P. Fourie and A. Goenwold, “The particle swarm Optimization Algorithm in Size and Shape Optimization”, Structural and Multidisciplinary Optimization, 23, 4, 259-267, 2002.

[11] S. He, E. Prempain and Q. H. Wu “An Improved Particle Swarm Optimizer for Mechanical Design Optimization Problems”, Engineering Optimization, 36, 5, 585-605, 2004.

[12] L. J. Li, Z. B. Huang, F. Liu and Q. H. Wu, “A heuristic Particle Swarm Optimizer for Optimization of Pin Connected structures”, Computers and structures, 85, 340-349, 2007.

[13] R. E. Perez and K. Behdinan, “Particle Swarm Approach for Structural Design Optimization”, Computers and Structures, 85,1579-1588, 2007

[14] C-J Liao, C-T Tseng, P. Luarn, “A discrete version of particle swarm optimization for flowshop scheduling problems”, Computers & Operations Research 34 , 3099 – 3111, 2007.
[15] P. W.Jansen, R. E. Perez, “Constrained Structural design Optimization via a parallel augmented Lagrangian Particle Swarm Optimization Aproach”, Computers and Structures, 89, 13-14, 1352-1366, 2011
[16] G. Poitras, G. Lefrancois, G. Cormier, “Optimization of Steel Floor Systems Using Particle Swarm optimization”, Journal of Constructional Steel research, 67, 8, 1225-1231, 2011.
[17] O. Duran, L. Perez, A. Batoccia, “Optimization of Modular Structures using Particle swarm optimization”, Expert Systems with Applications, 39, 3, 3507-3515, 2012.
[18] G-C. Luh, C-Y, Lin, “Optimal design of truss-structures using particle Swarm Optimization”, Computers and Structures, 89, 23-23, 2221-2232, 2011.
[19] V. Plevris, A. Batavanis, M. Papadrakakis, “Optimum design of Steel Structures with Particle Swarm Optimization Method based on EC3”, Proc. of 3rd Int. Conf. on Computational Methods in Structural Dynamics and earthquake Eng’g, COMPDYN 2011, Corfu, Greece.
[20] E. Doğan, M. P. Saka. “Optimum design of unbraced steel frames to LRFD–AISC using particle swarm optimization”, Advances in Engineering Software. DOI: 10.1016/j.advengsoft.2011.05.00, 2011
[21] Z. W. Geem, J. H. Kim, G.V. Loganathan, “A New Heuristic Optimization Algorithm; Harmony Search”. Simulation 76:60-68, 2001.
[22] K. S. Lee and Z. W. Geem, “A New Structural Optimization Method Based on the Harmony Search Algorithm”, Computers and Structures, 82, 781-798, 2004.
[23] K. S. Lee and Z. W. Geem, “A New Meta-Heuristic Algorithm for Continuous Engineering Optimization: Harmony Search Theory and Practice”, Computer Methods in Applied Mechanics and Engineering, 194, 3902-3933, 2005.
[24] Z. W. Geem, “Novel derivative of harmony search algorithm for discrete design variables”. Applied Mathematics and Computation 199, 1, 223–230, 2008.
[25] Z. W. Geem (Ed.), “Music-Inspired Harmony Search Algorithm”, Springer, 2009.
[26] S. Carbas and M. P. Saka, “Optimum Design of Single Layer Network Domes Using Harmony Search Method”, Asian Journal of Civil Engineering, 10, 1, 97-112, 2009.
[27] M. P. Saka and F. Erdal, “Harmony search based algorithm for the optimum design of grillage systems to LRFD-AISC”. Structural and Multidisciplinary Optimization, 38, 1,25-41, 2009.
[28] M. P. Saka, “Optimum Design of Steel Sway Frames to BS5950 Using Harmony Search Algorithm”, Journal of Constructional Steel Research, Vol. 65, No. 1, 36-43, 2009.
[29] S. O. Degertekin, “Optimum Design of Steel Frames via harmony Search Algorithm”, Studies in Computational Intelligence, 239, 51-78, 2009.
[30] S. O. Degertekin, M. S. Hayalioglu and H. Gorgun, “Optimum Design of Geometrically Nonlinear Steel frames with Semi-Rigid Connections Using a

Harmony Search Algorithm", Steel and Composite Structures, 9, 6, 535-55, 2009.
[31] M. P. Saka and O. Hasancebi, "Adaptive Harmony Search Algorithm for Design Code Optimization of Steel structures", Chapter 3, Harmony Search Algorithms for Structural Design Optimization, Dr. Z. W. Geem (Ed.), SCI 239, pp:79-120, Springer, 2009.
[32] Z. W. Geem (Ed.), "Recent Advances in Harmony Search Algorithm", Springer, 2010.
[33] Z. W. Geem (Ed.), "Harmony Search Algorithms for Structural Design Optimization", Springer, 2010.
[34] O. Hasançebi, F. Erdal and M. P. Saka, "An Adaptive Harmony Search Method for Structural Optimization", Journal of Structural Engineering, ASCE, 136,4, April, 419-431, 2010.
[35] Z. W. Geem, "Harmony Search Algorithm". www.harmonysearch.info. Accessed 04 May 2011.
[36] M. P. Saka, "Optimum Geometry Design of Geodesic Domes Using Harmony search Algorithm". Advances in Structural Engineering, 10, 6, 595-606, 2007.
[37] S. Carbas and M. P. Saka, "Optimum topology design of various geometrically nonlinear latticed domes using improved harmony search method", Structural and Multidisciplinary Optimization, 45, 3, 377-399, 2012.
[38] F. Erdal, E. Doğan and M. P. Saka "Optimum design of cellular beams using harmony search and particle swarm optimizers", Journal of Constructional Steel Research, 67, 2, 237-247, 2011
[39] M. P. Saka, I. Aydogdu, O. Hasancebi and Z. W. Geem, "Harmony Search Algorithms in Structural Engineering", Chapter 6 in the book titled "Computational Optimization and Applications in Engineering and Industry", X-S Yang and S. Koziel, (Editors), Springer, 145-182, 2011.
[40] O. Erol and I. Eksin, "A New Optimization Method: Big Bang-Big Crunch", Advances in Engineering Software, 37, 106-111, 2006.
[41] C. V. Camp, "Design of Space Trusses Using Big Bang-Big Crunch Optimization", Journal of Structural Engineering, ASCE, 133, 7,999-1008, 2007.
[42] A. Kaveh and S. Talatahari, "Size Optimization of Space Trusses Using Big Bang-Big Crunch Algorithm", Computers and Structures, 87, 1129-1140, 2009.
[43] Kaveh and S. Talatahari, "Optimal design of single layer domes using metaheuristic algorithms: a comparative study", International Journal of Space Structures, 25, 4, 217-227, 2010.
[44] Kaveh and S. Talatahari, "Optimal Design of Schwedler and Ribbed Domes via Hybrid Big Bang Big Crunch Algorithm", Journal of Constructional Steel research, 66, 3, 412-419, 2010.
[45] Kaveh and S. Talatahari, "A discrete big bang-big crunch algorithm for optimal design of skeletal structures", Asian Journal of Civil Engineering, 11,1, 103-122, 2010.
[46] Kaveh and H. Abbasgholiha, "Optimum Design of Steel Sway Frames Using Big Bang Big Crunch Algorithm", Asian Journal of Civil Engineering, 12, 3, 293-317, 2011.

[47] X-S Yang, S. Deb, “Engineering Optimization by Cuckoo Search”, Int. J. Mathematical Modeling and Numerical Optimization”, Vol. 1, No. 4, 330–343, 2010.
[48] P. Civicioglu, E. Besdok, “A conceptual comparison of the cuckoo search, particle swarm optimization, differential evolution and artificial bee colony algorithms”, Artificial Intelligence Review, DOI: 10.1007/s10462-011-9276-0, 2011.
[49] A.H. Gandomi, X-S Yang, A. H. Alavi, “ Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems”, Engineering with Computers, DOI: 10.1007/s00366-011-0241-y, 2011
[50] Kaveh, T. Bakhspoori, “Optimum design of steel frames using cuckoo search algorithm with Levy flights”, The Structural design of Tall and Special Buildings, DOI: 10.1002/tal.754, 2011.
[51] X-S Yang, S. Deb, “Multi-objective cuckoo search for design optimization”, Computers and Operations Research, DOI: 10.1016/j.cor.2011.09.026, 2011.
[52] LRFD-AISC, Manual of Steel Construction, “Load and Resistance Factor Design”, Metric Conversion of the Second Edition, AISC, Vol. I&II, 1999.
[53] W. McGuire, “Steel Structures”, Prentice-Hall, 1968.
[54] R. N. Mantegna, “Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes”, Physical Review E, 49, 5, 4677-4683, 1994.