INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2015; 5(4):479-492

PLASTIC ANALYSIS OF PLANAR FRAMES USING CBO AND ECBO ALGORITHMS

A. Kaveh[1,2,*,†], M.H. Ghafari[1]
[1]*Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran, P.O. Box 16846-13114, Iran*
[2]*Guest Professor, Institute for Mechanics of Materials and Structures, Vienna University of Technology, Karlsplatz 13, A-1040 Wien, Austria*

ABSTRACT

In rigid plastic analysis one of the most widely applicable methods that is based on the minimum principle, is the combination of elementary mechanisms which uses the upper bound theorem. In this method a mechanism is searched which corresponds to the smallest load factor. Mathematical programming can be used to optimize this search process for simple frames, and meta-heuristic algorithms are the best choice for larger frame structures. In this paper, the Colliding Bodies Optimization (CBO) and its enhanced variant (ECBO) are employed to optimize the process of finding an upper bound for the collapse load factor of the planar frames. The efficiency of these algorithms is compared to that of the Particle Swarm Optimization (PSO) algorithm through four numerical examples form multi-bay multi-story frames and pitched roof frames.

Keywords: plastic analysis; colliding bodies optimization; enhanced colliding bodies optimization; collapse load factor; planar frames.

Received: 22 May 2015; Accepted: 27 July 2015

1. INTRODUCTION

The minimum and maximum principles are the basis of nearly all the analytical methods used for plastic analysis and design of frames [1]. One of the most widely applicable methods based on the minimum principle is the combination of elementary mechanisms, developed by Neal and Symonds [2–4]. However this method, aside from all of its

*Corresponding author: Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran, P.O. Box 16846-13114, Iran
†E-mail address: alikaveh@iust.ac.ir (A. Kaveh)

capabilities, has certain limitations that prevent its common application. For instance, as the structure becomes more complex, an extensive number of mechanisms have to be considered, not to mention the pains of combining them to find the actual collapse mechanism. This situation calls for the development of a methodology capable of finding solutions as fast and accurate as possible while accounting for the natural compromise that has to be made. This is where the heuristic and meta-heuristic algorithms come into the scene.

The problem of plastic analysis and design of frames having rigid joints was solved by Charnes and Greenberg [5] using the linear programming, as early as 1951. Further progress in this field can be traced to Baker and Heyman [1], Munro [6], Livesley [7], Watwood [8] and Kaveh and Mokhtarzadeh [9] among others. The progress during 1955–1977 is well documented in Ref. [10], and a survey of research results achieved in the subsequent 25 years on limit analysis and design in plasticity has been performed by Maier et al. [11]. Application of algorithmic heuristics dates back to Kohama et al. [12], Kaveh and Khanlari [13], Kaveh and Jahanshahi [14], Kaveh et al. [15].

Over 1990, many optimization algorithms have been developed. Genetic algorithms [16] (GA) and particles swarm optimization [17] (PSO) are the most common algorithms widely used by researchers. The PSO is based on sharing information between each particle in the swarm and update particle's position based on its memory and data gained by other particles.

This work focuses on two recently developed meta-heuristic algorithms, namely Colliding Bodies Optimization (CBO) and Enhanced Colliding Bodies Optimization (ECBO). CBO is one of the most simple and successful optimization techniques introduced by Kaveh and Mahdavai [18]. ECBO is the enhanced variant of the COB proposed by Kaveh and Ilchi Ghazaan [19]. A book is published recently covering many applications of these methods [20].

CBO is a population-based stochastic optimization algorithm based on the governing laws of one dimensional collision between two bodies from the physics. Each agent is modeled as a body with a specified mass and velocity. A collision occurs between pairs of objects to find the global or near-global solutions.

ECBO is the modified version of the CBO that uses a memory to save some best solutions and utilizes a technique to escape from local optima.

After this section, the paper is organized as follows. Section 2 briefly introduces the method of combination of elementary mechanisms. In Section 3 adaptations of the CBO and ECBO algorithms to the problem of finding the collapse load factor are presented. Section 4 studies various design examples to verify the efficiency of these algorithms. Finally, the concluding remarks are provided in Section 5.

2. COMBINATION OF ELEMENTARY MECHANISMS

In the method of combination of elementary mechanisms, the load factor is obtained using the virtual work theorem. Rotations and displacements are considered to be virtual, and internal work and external work are computed using these virtual quantities; then the load factor for a specific mechanism is described as ratio of the internal virtual work to the external virtual work:

$$\lambda = \frac{\text{internal virtual work}}{\text{external virtual work}} \tag{1}$$

The external virtual work is computed by summing all the joint forces P multiplied by corresponding joint displacements d in the direction of those forces:

$$\text{External virtual work} = \mathbf{P}^{t}\mathbf{d} \tag{2}$$

The internal virtual work is the sum of all rotations at active hinges multiplied by the plastic moments of members in which active hinges form. However, since the plastic moments $\mathbf{M}_P$ always resist the rotations at hinges r, the internal work is always positive and therefore the absolute values of rotations should always be used [1]:

$$\text{Internal virtual work} = \mathrm{M}_P^{t}\,|\mathrm{r}| \tag{3}$$

If a frame has *N* cardinal sections and its static degree of indeterminacy is equal to *R*, then there will be (*N*−*R*) elementary mechanisms for this frame, and all the other collapse mechanisms can be constructed by combining these elementary mechanisms. Elementary mechanisms cannot be acquired by combining any other mechanisms. From these (*N*-*R*) elementary mechanisms for a frame, some are joint mechanisms (equal to the number of joints) and the remaining mechanisms are called independent mechanisms. Independent mechanisms for rectangular and pitched roof frames are also categorized as sway and beam mechanisms [1].

After generating elementary mechanisms, a search is carried out to find the actual collapse mechanism by checking every possible combination of elementary mechanisms in order to reduce the load factor. The criterion for a mechanism to be the actual collapse mechanism is that there is no possibility of combining it with other elementary mechanisms without increasing the load factor and also its load factor should be lower than any possible collapse mechanism that can be constructed for the frame. For more information about terminology and details of combination of mechanisms, one may refer to Ref. [1].

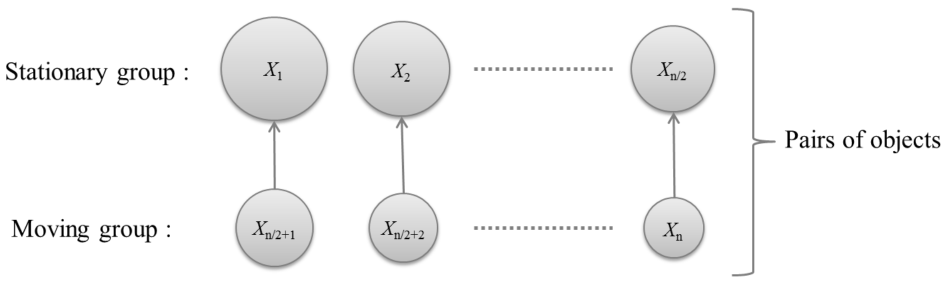

Figure 1. The pairs of CBs for collision [18]

3. CBO AND ECBO FOR FINDING THE COLLAPSE LOAD FACTOR OF FRAMES

3.1 Colliding bodies optimization (CBO)

Colliding bodies optimization (CBO) is a new meta-heuristic search algorithm that is developed by Kaveh and Mahdavi [18]. In this technique, one object collides with other object and they move towards a minimum energy level. The CBO is simple in concept and does not depend on any internal parameter. Each colliding body (CB), X_i, has a specified mass defined as:

$$m_k = \frac{\dfrac{1}{fit(k)}}{\dfrac{1}{\sum_{i=1}^{n} \dfrac{1}{fit(i)}}}, \qquad k = 1,2,\ldots,n \tag{4}$$

Where $fit(i)$ represents the objective function value of the ith CB and n is the number of colliding bodies.

In order to select pairs of objects for collision, CBs are sorted according to their mass in a decreasing order and they are divided into two equal groups: (i) stationary group, (ii) moving group (Fig. 1). Moving objects collide to stationary objects to improve their positions and push stationary objects towards better positions. The velocities of the stationary and moving bodies before collision (v_i) are calculated by

$$v_i = 0, \qquad i = 1,2,\ldots,\frac{n}{2} \tag{5}$$

$$v_i = x_{i-\frac{n}{2}} - x_i, \qquad i = \frac{n}{2}+1, \frac{n}{2}+2,\ldots,n \tag{6}$$

The velocity of stationary and moving CBs after the collision (v'_i) are evaluated by

$$v_i' = \frac{(m_{i+\frac{n}{2}} + \varepsilon m_{i+\frac{n}{2}})v_{i+\frac{n}{2}}}{m_i + m_{i+\frac{n}{2}}} \qquad i = 1,2,\ldots,\frac{n}{2} \tag{7}$$

$$v_i' = \frac{(m_i - \varepsilon m_{i-\frac{n}{2}})v_i}{m_i + m_{i-\frac{n}{2}}} \qquad i = \frac{n}{2}+1, \frac{n}{2}+2,\ldots,n \tag{8}$$

$$\varepsilon = 1 - \frac{iter}{iter_{\max}} \tag{9}$$

Where $iter$ and $iter_{max}$ are the current iteration number and the total number of iteration for optimization process, respectively. ε is the coefficient of restitution (COR).

New positions of each group are updated by

$$x_i^{new} = x_i + rand \circ v_i', \qquad i = 1,2,\ldots,\frac{n}{2} \tag{10}$$

$$x_i^{new} = x_{i-\frac{n}{2}} + rand \circ v_i', \qquad i = \frac{n}{2}+1, \frac{n}{2}+2,\ldots,n \tag{11}$$

where x_i^{new}, x_i and v'_i are the new position, previous position and the velocity after the collision of the ith CB, respectively. *rand* is a random vector uniformly distributed in the range of [-1,1] and the sign ‘‘∘’’ denotes an element-by-element multiplication.

The flowchart of CBO algorithm is depicted in Fig. 2.

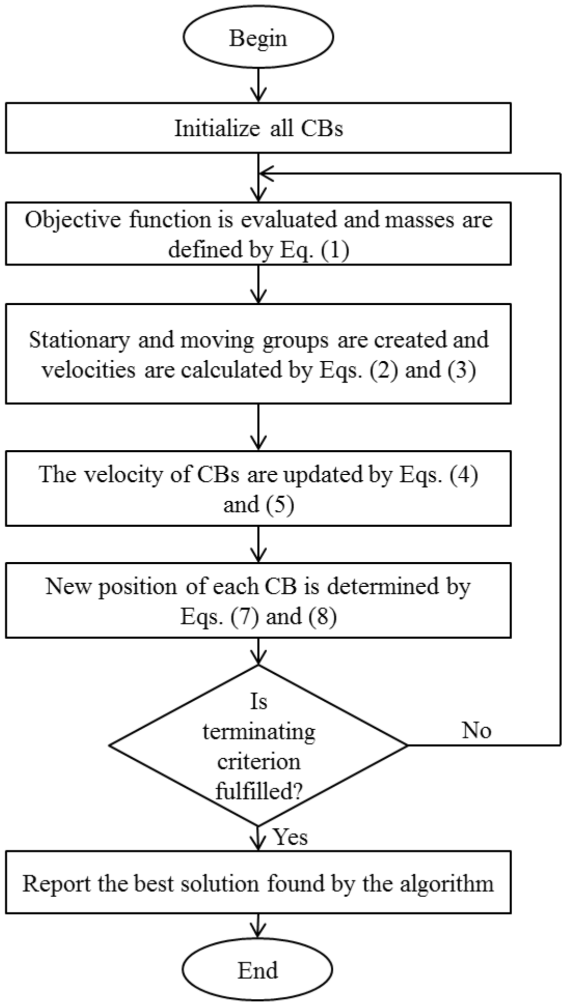

Figure 2. Flowchart of the CBO algorithm [18]

3.2 Enhanced colliding bodies optimization (ECBO)

In order to improve CBO to get faster and more reliable solutions, Enhanced Colliding Bodies Optimization (ECBO) was developed which uses memory to save a number of historically best CBs and also utilizes a mechanism to escape from local optima [19]. The flowchart of ECBO is shown in Fig. 3. The steps of this technique are given as follows:

Level 1: Initialization

The initial positions of all CBs are determined randomly in an m-dimensional search space.

$$x_i^0 = x_{\min} + random \circ (x_{\max} - x_{\min}), \qquad i = 1,2,\ldots,n \tag{12}$$

where x_i^0 is the initial solution vector of the ith CB. Here, x_{min} and x_{max} are the bounds of design variables; random is a random vector which each component is in the interval [0, 1].

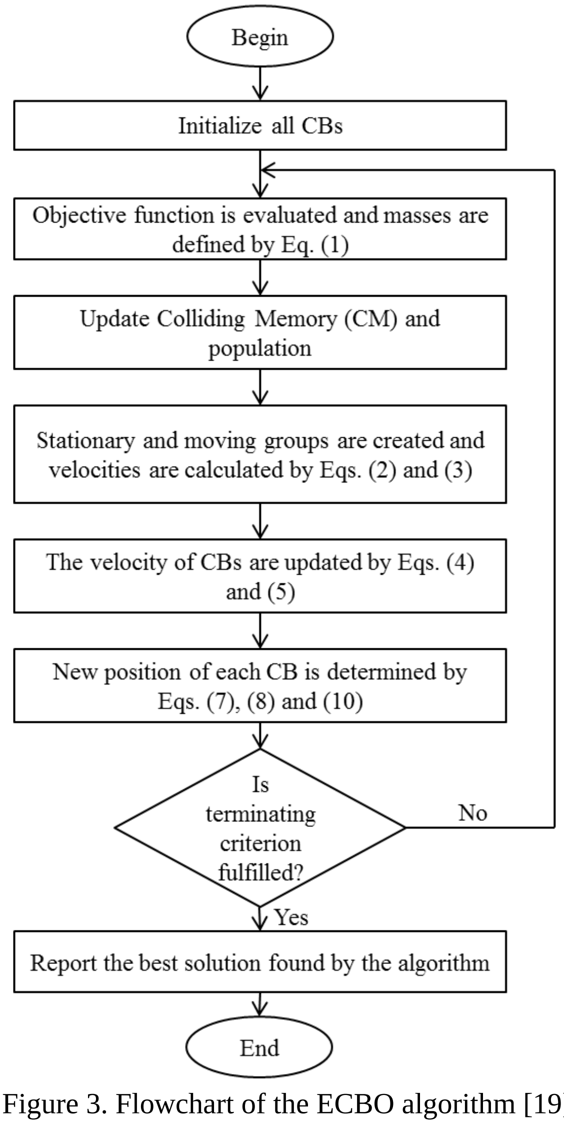

Figure 3. Flowchart of the ECBO algorithm [19]

Level 2: Search

Step 1: The value of mass for each CB is evaluated according to Eq. (4).

Step 2: Colliding memory (CM) is utilized to save a number of historically best CB vectors and their related mass and objective function values. Solution vectors which are saved in CM are added to the population and the same number of current worst CBs are deleted. Finally, CBs are sorted according to their masses in a decreasing order.

Step 3: CBs are divided into two equal groups: (i) stationary group, (ii) moving group (Fig. 1).

Step 4: The velocities of stationary and moving bodies before collision are evaluated by Eqs. (5) and (6), respectively.

Step 5: The velocities of stationary and moving bodies after the collision are evaluated using Eqs. (7) and (8), respectively.

Step 6: The new position of each CB is calculated by Eqs. (10) and (11).

Step 7: A parameter like Pro within (0, 1) is introduced and it is specified whether a component of each CB must be changed or not. For each colliding body Pro is compared with rn_i (i=1,2,…,n) which is a random number uniformly distributed within (0,1). If $rn_i <$ pro, one dimension of the ith CB is selected randomly and its value is regenerated as follows:

$$x_{ij} = x_{j,\min} + random.(x_{j,\max} - x_{j,\min}) \tag{13}$$

where x_{ij} is the jth variable of the ith CB. $x_{j,min}$ and $x_{j,max}$ respectively, are the lower and upper bounds of the jth variable. In order to protect the structures of CBs, only one dimension is changed.

Level 3: Terminal condition check

After the predefined maximum evaluation number, the optimization process is terminated.

4. NUMERICAL EXAMPLES

This section contains the implementations of the proposed algorithms in MATLAB® and their applications to four examples. In order to provide a measure of their comparative performance, each example is solved by all the algorithms. To contrast the efficiency of these algorithms, in each case, the convergence history of all the algorithms, demonstrating the load factor against the iteration number for each algorithm, are illustrated. In addition, comparisons are made with the exact results, further validating the merits of our obtained solutions. In the ECBO algorithm, the values of *cMs* (Colliding memory size) and *pro*, are set to 2 and 0.3, respectively.

4.1 Example 1

A three-bay and three-story frame is considered as shown in Fig. 4. Plastic moments are provided for each member. For all the algorithms, the number of populations is 20 and the number of iterations is considered as 20. The actual collapse mechanism is obtained by checking all possible combination of mechanisms is 1.8730 (see Table 1). The mechanism corresponding to the result of the best algorithm is shown in Fig. 5. The convergence histories are illustrated in Fig. 6.

Table 1: Load factor obtained for Example 1 using different algorithms

Exact load factor	PSO	CBO	ECBO
1.8730	1.9653	1.9012	1.8730

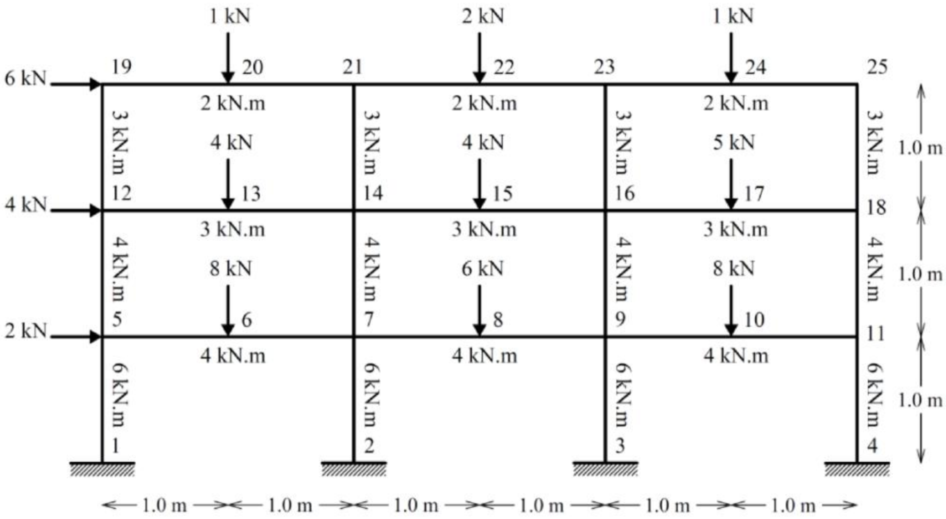

Figure 4. A three-bay and three-story frame: geometry, loading and plastic moments [21]

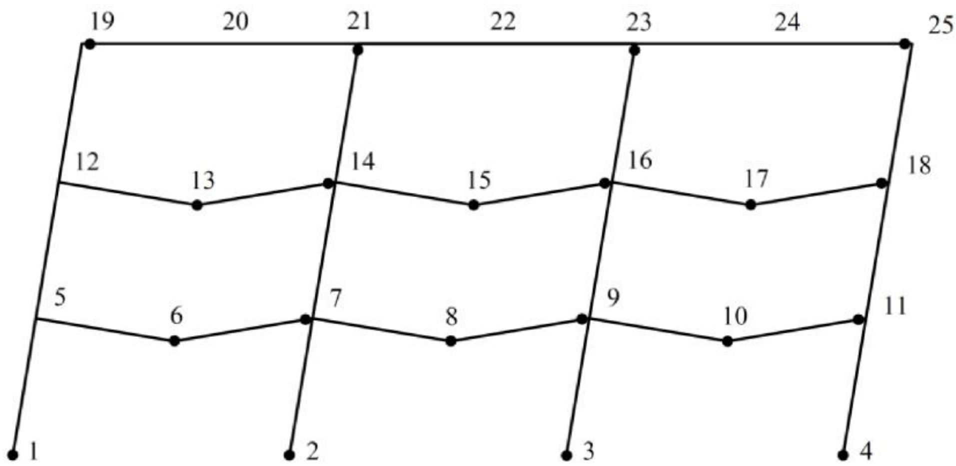

Figure 5. A three-bay and three-story frame: actual collapse mechanism [21]

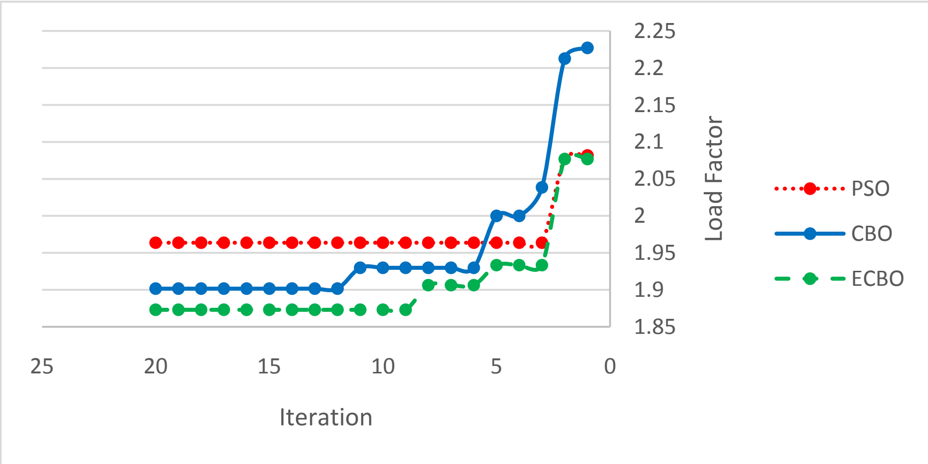

Figure 6. Convergence history of the three-bay and three-story frame

4.2 Example 2

A four-bay and four-story frame is considered as shown in Fig. 7. Plastic moments are provided for each member. For all the algorithms, the number of populations is set to 20 and the number of iterations is considered as 20. The actual collapse load factor of the mechanism is obtained as 1.8880 by checking all possible combination of mechanisms (see Table 2). The mechanism corresponding to the result of the best algorithm is shown in Fig. 8. The convergence histories are illustrated in Fig. 9.

Table 2: Load factor obtained for Example 2 using different algorithms

Exact load factor	PSO	CBO	ECBO
1.8880	1.9471	1.8880	1.8880

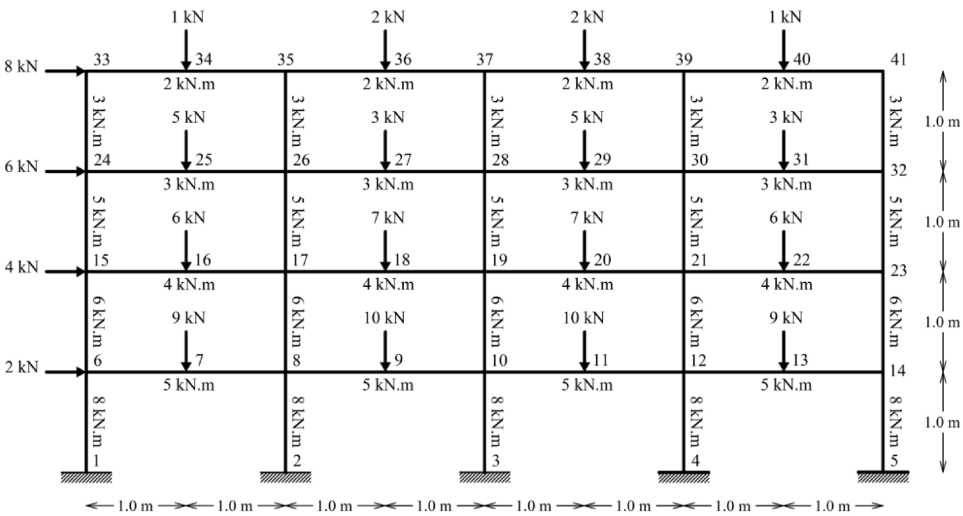

Figure 7. A four-bay and four-story frame: geometry, loading and plastic moments [21]

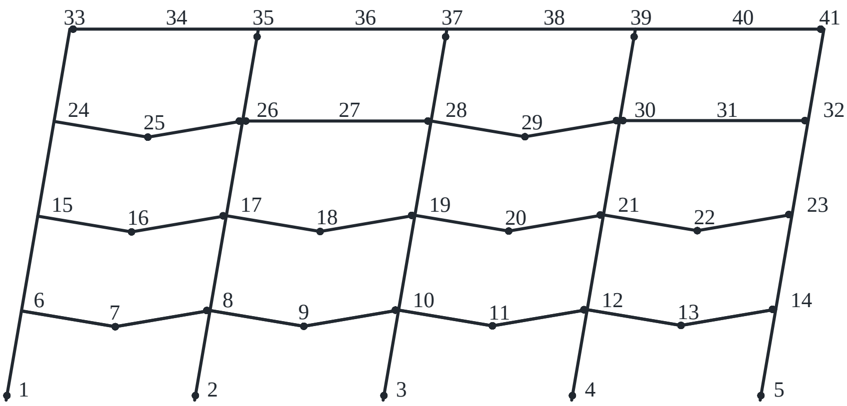

Figure 8. A four-bay- and four-story frame: The actual collapse mechanism

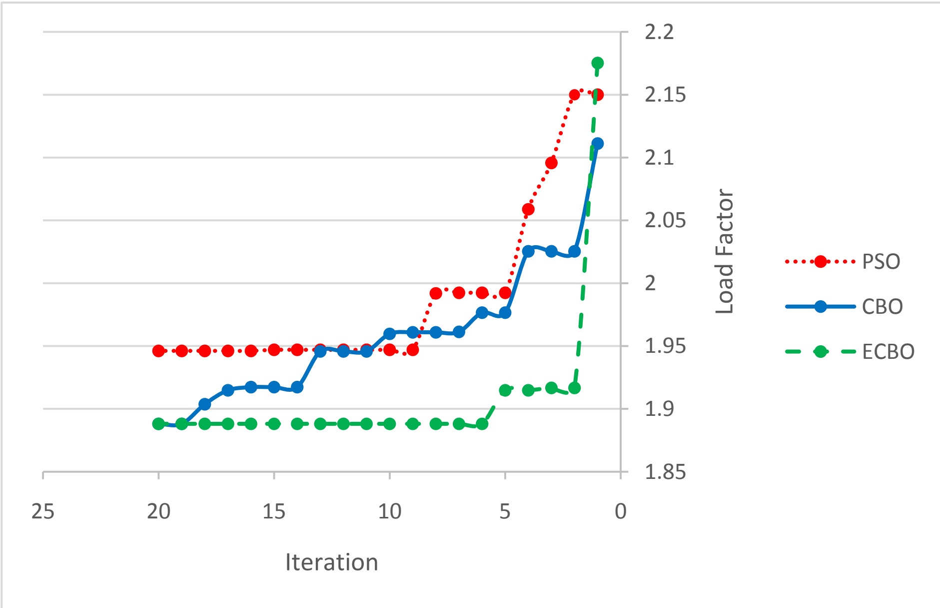

Figure 9. Convergence history of the four-bay and four-story frame

4.3 Example 3

Consider a pitched roof frame as shown in Fig. 10. All of the members of this frame have a full plastic moment equal to 5kN.m. For all algorithms, number of population is 10 and the number of iteration is 20. The actual collapse mechanism is obtained by checking all possible combination of mechanisms is 0.1626 (see Table 3). The mechanism corresponding to the result of the best algorithm is shown in Fig. 11. The convergence histories are illustrated in Fig. 12.

Table 3: Load factor obtained for Example 3 using different algorithms

Exact load factor	PSO	CBO	ECBO
0.1626	0.1626	0.1626	0.1626

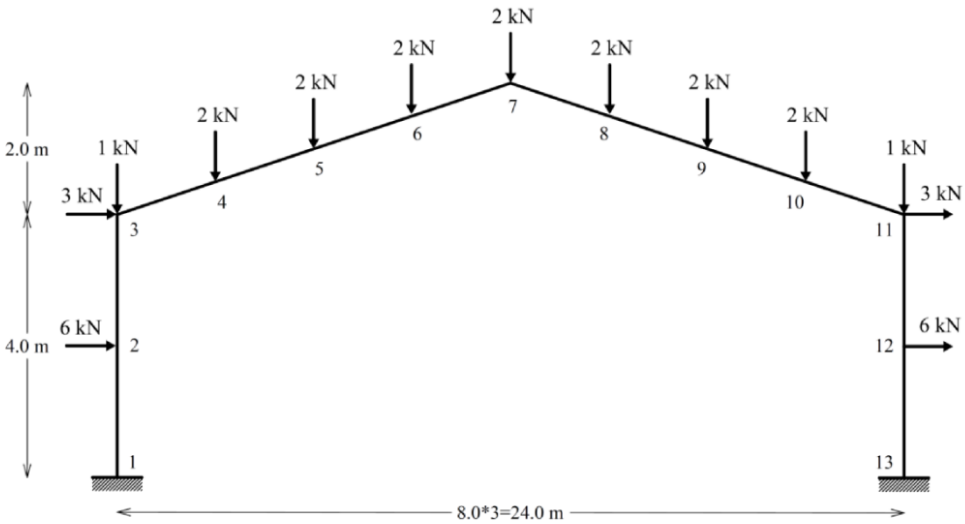

Figure 10. A pitched roof portal frame: geometry and loading [1]

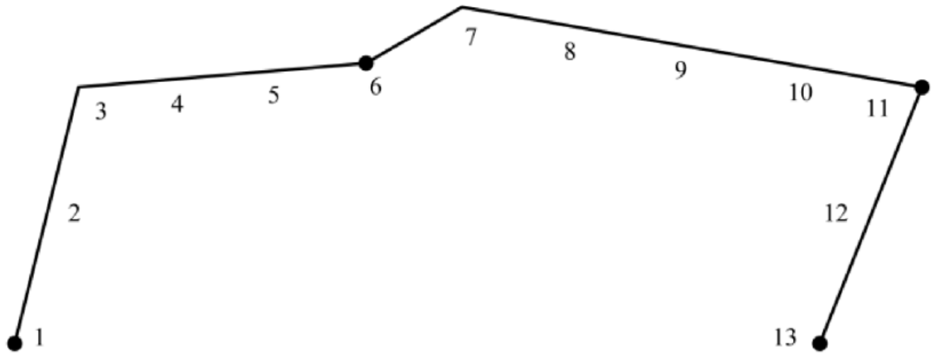

Figure 11. A pitched roof portal frame: Actual collapse mechanism [1]

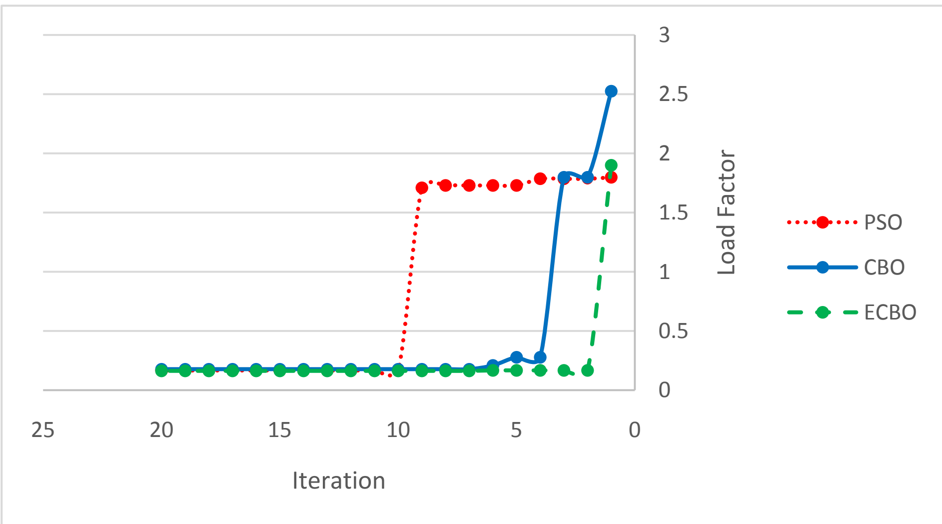

Figure 12. Convergence history of the pitched roof portal frame

4.4 Example 4

Consider a two-bay pitched roof frame as shown in Fig. 13. All of the members of this frame have a full plastic moment equal to 5kN.m. For all the algorithms, the number of population is taken as 20 and the number of iterations is set to 20. The actual collapse load factor is obtained by checking all possible combinations of the mechanisms as 0.2083 (see Table 4). The mechanism corresponding to the result of the best algorithm is shown in Fig. 14. The convergence histories are illustrated in Fig. 15.

Table 4: Load factor obtained for Example 4 using different algorithms

Exact load factor	PSO	CBO	ECBO
0.2083	0.3944	0.2187	0.2094

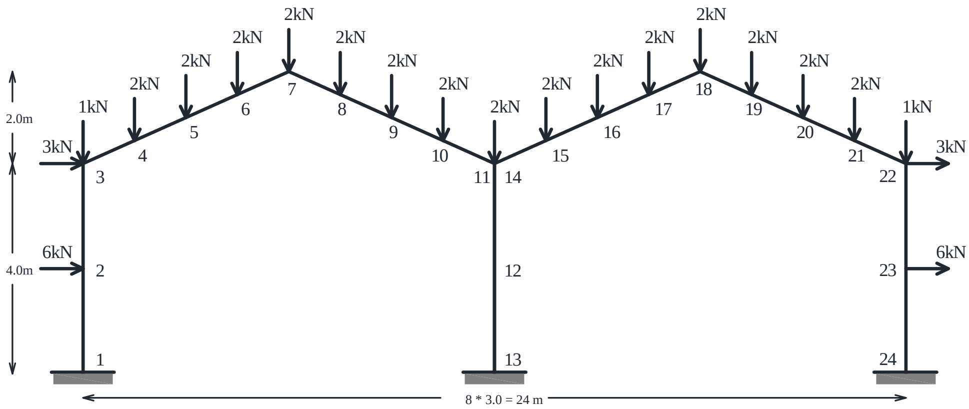

Figure 13. A two-bay pitched roof portal frame: geometry and loading [1]

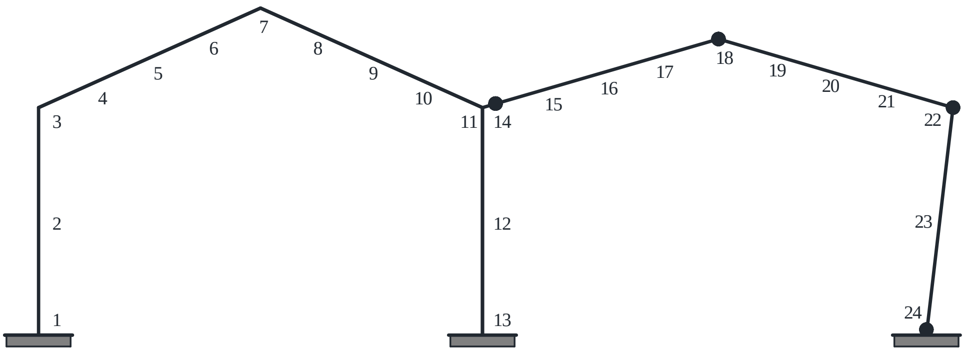

Figure 14. A two-bay pitched roof portal frame: Actual collapse mechanism

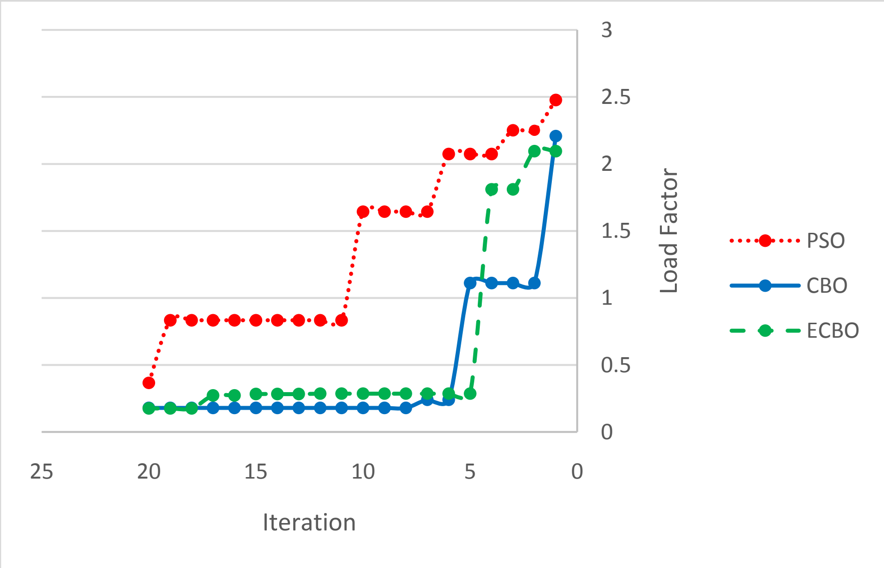

Figure 15. Convergence history of the two-bay pitched roof portal frame

5. CONCLUDING REMARKS

In this paper, the CBO and ECBO algorithms are utilized to optimize the process of finding the collapse load factor of planar frames. Comparison of the performance of these algorithms in Section 4 shows the efficiency of the proposed CBO and ECBO algorithms for the analysis of the frame structures. From the convergence curves it can be seen that the ECBO convergence to the solution faster than other algorithms and all algorithms often results in the exact load factor of the studied frame structures. In the complex problems such as the Example 4, the number of population and number of iterations are high for the solution. In such condition, collapse load factor is close to the load factor, but obtained mechanism can be non-correct.

REFERENCES

1. Baker J, Heyman J. *Plastic Design of Frames, Fundamentals*, Cambridge University Press, Cambridge, Vol. 1, 1969.
2. Neal BG, Symonds PS. The rapid calculation of plastic collapse loads for a framed structure, *Proc Institut Civil Eng* 1952, London; **1**(3): pp. 58-100.
3. Neal BG, Symonds PS. The calculation of plastic loads for plane frames, Preliminary Publication, *International Association for Bridge and Structural Engineering, 4th Congress*, Cambridge and London, 1952.
4. Neal BG, Symonds PS. The calculations of collapse loads for framed structures, *J Institut Civil Eng* 1951; **35**: 21-40.

5. Charnes A, Greenberg HJ. Plastic collapse and linear programming, *Summer Meeting of the American Mathematical Society*, 1959.
6. Munro J. Optimal plastic design of frames, In: *Proceedings of NATO, Waterloo: Advanced Study in Engineering Plasticity by Mathematical Programming* 1977; pp. 136-171.
7. Livesley RK. Linear programming in structural analysis and design. In: Gallagher RH et al., editors, Optimum Structural Design, Chapter 6, Wiley, New York, 1977.
8. Watwood VB. Mechanism generation for limit analysis of frames, *J Struct Eng, ASCE* 1979; **109**: 1-15.
9. Mokhtar-zadeh A, Kaveh A. Optimal plastic analysis and design of frames; graph-theoretical methods, *Comput Struct* 1999; **73**(2-5): 485-96.
10. Cohen MZ, Maier G. editors. Engineering plasticity by mathematical programming, *Proceedings of the NATO Advanced Study Institute*, University of Waterloo, Waterloo, Canada, 2–12 August 1977, Pergamon Press, 1979.
11. Maier G, Pastor J, Ponter ARS, Weichert D. Direct methods in limit and shakedown analysis. In: de Borst R, Mang HA, editors. Numerical and computational methods. Milne I, Ritchie RO, Karihaloo B, editors. Comprehensive Structural Integrity, Amsterdam: Elsevier-Pergamon, Chapter 12, 2003.
12. Kohama Y, Takada A, Miyamura A. Collapse analysis of rigid frame by genetic algorithm, *Proceedings of OPT197*, Wessex Institute of Technology, Rome, September 1997, pp. 456-461.
13. Kaveh A, Khanlari K. Collapse load factor of planar frames using a modified genetic algorithm, *Communications in Numerical Methods in Engineering* 2004; **20**: 911-25.
14. Kaveh A, Jahanshahi M. Plastic limit analysis of frames using ant colony systems, *Comput Struct* 2008; **86**: 1152-63.
15. Kaveh A, Jahanshahi M, Khanzadi M. Plastic analysis of frames using genetic algorithm and ant colony algorithm, *Asian J Civil Eng* 2008; **9**(3): 227-46.
16. Goldberg D. Genetic algorithms in search, optimization, and machine learning, New York, NY: Addison-Wesley, 1989.
17. Kennedy J, Eberhart R. Particle swarm optimization, In: *IEEE International Conference on Neural Networks* 1995, Piscataway, NJ, **IV**; pp. 1942-1948.
18. Kaveh A, Mahdavai VR. Colliding bodies optimization: A novel meta-heuristic method, *Comput Struct* 2014; **139**: 18-27.
19. Kaveh A, Ilchi Ghazaan M. Enhanced colliding bodies optimization for design problems with continuous and discrete variables, *Adv Eng Softw* 2014; **77**: 66-75.
20. Kaveh A, Mahdavi VR. *Colliding Bodies Optimization: Extensions and Applications*, Springer, Switzerland, 2015.
21. Kaveh A, Bakhshpoori T, Kalateh-Ahani T. Optimum plastic analysis of planar frames using ant colony system and charged system search algorithms, *Scientia Iranica A* 2013; **20**(3): 414-21.