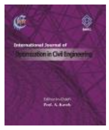
INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2023 13(2): 143-154



A METAHEURISTIC-BASED ARTIFICIAL NEURAL NETWORK FOR PLASTIC LIMIT ANALYSIS OF FRAMES

A. Kaveh[1*, †] M.R. Seddighian[2] and N. Farsi[1]
[1]*School of Civil Engineering, Iran University of Science and Technology, Narmak, Tehran, Iran*
[2]*Ph. D. Candidate at Center of Excellence in Structures and Earthquake Engineering, Department of Civil Engineering, Sharif University of Technology, Tehran, Iran*

ABSTRACT

Despite the advantages of the plastic limit analysis of structures, this robust method suffers from some drawbacks such as intense computational cost. Through two recent decades, metaheuristic algorithms have improved the performance of plastic limit analysis, especially in structural problems. Additionally, graph theoretical algorithms have decreased the computational time of the process impressively. However, the iterative procedure and its relative computational memory and time have remained a challenge, up to now. In this paper, a metaheuristic-based artificial neural network (ANN), which is categorized as a supervised machine learning technique, has been employed to determine the collapse load factors of two-dimensional frames in an absolutely fast manner. The numerical examples indicate that the proposed method's performance and accuracy are satisfactory.

Keywords: Optimization, Metaheuristic Algorithms, Plastic Limit Analysis, Artificial Neural Networks, Machine Learning, Finite Element Method, Nonlinear Structural Analysis.

Received: 25 August 2022; Accepted: 10 December 2022

1. INTRODUCTION

One of the powerful structural analysis methods, especially for ductile materials, is Plastic Analysis (PA). The basis of almost all the analytical methods employed for PA is the maximum and minimum principles [1]. Among minimum principal methods, the most frequent one is the combination of elementary mechanisms. This method has developed by Neal and Symonds, first [2]–[4].

In 1951, due to Charnes and Greenberg's effort [5], the PA problem of frames with rigid

[*]Corresponding author: School of Civil Engineering, Iran University of Science and Technology, Tehran-16, Iran
[†]E-mail address: alikaveh@iust.ac.ir (A. Kaveh)

joints started to solve. Linear Programming (LP) was the main tool employed by them. Heyman [6] utilized simple plastic theory to investigate the minimum weight of rectangular two-dimensional frames under different loading conditions. The principal theory in his study was the Foulkes Theory. Further progress in this field is attributed to Watwood [7], Baker and Heyman [8], Jennings [9], Thierauf [10], Horne [11], and Gorman [12].

Despite the advantages of the combination of elementary mechanisms, this method suffers from some drawbacks. These drawbacks prevent it to be considered a frequent analysis tool. Among these drawbacks, the extensive number of mechanisms that have to be combined for evaluating the collapse load factor, impose an intense computational cost, either in time or memory, to the problem solver. Therefore, some novel approaches, such as graph theory, and modern algorithms, such as metaheuristics, have been used in this manner. For the first time, Kaveh [13] has employed graph theoretical concepts for flexibility analysis of structures. In this manner, he has improved cycle bases in flexibility analysis for an efficient and accurate analysis of structures. Using this novel methodology, Mokhtarzadeh and Kaveh [14] presented an efficient graph theoretical method for optimal plastic analysis and design of frames. Kaveh and Khanlari [15] employed the genetic algorithm to estimate the collapse load factor of planar frames. Kaveh and Rahami [16] have utilized Genetic Algorithm for the structural analysis using force method. Palizi and Saedi Daryan [17], presented a comparative study between different metaheuristic algorithms for plastic analysis of braced frames. an Automatic method for evaluation of plastic collapse conditions of planar frames has been presented by Greco et al. [18]. Smail and Laid [19], proposed a second-order analysis of plane steel structures using the Rankine-Merchant-Wood approach. Kaveh et al. [20] utilized the ant colony system and Charge System Search (CSS) algorithms for optimal PA. Colliding Bodies Optimization (CBO) and its enhanced version (ECBO) have been employed by Kaveh and Ghafari [21] for these structures. The rectangular grid’s collapse load factor has been studied by Kaveh et al. [22]. Also, Kaveh and Jahanshahi [23], developed a metaheuristic-based framework for Plastic Limit Analysis (PLA) of frames.

Although all the above-mentioned efforts have been considered crucial studies in this field and they improved some computational drawbacks of PA, the intense computational cost of the plastic analysis, due to their iterative approaches has remained up to now. Therefore, it seems that a new computational method is required to be added to this field. In this paper, a metaheuristic-based artificial neural network (ANN), which is categorized as a supervised machine learning technique, has been employed to determine the collapse load factors of two-dimensional frames in an absolutely fast manner. The numerical examples indicate that the proposed method's performance and accuracy are satisfactory.

The paper is organized as follows. Section 2 introduces the formulation of the PA, including the generation of elementary mechanisms, determination of the collapse load factor, and combination of elementary mechanisms. An introduction to the metaheuristic algorithms and artificial neural networks is presented in Section 3. Section 4 is dedicated to the proposed new algorithm and its numerical validations. Finally, Section 5 concludes the study.

2. PLASTIC ANALYSIS

One of the computational methods to find a set of independent mechanisms has been developed by Watwood [7]. However, this method also calculates joint mechanisms which causes some extra computational overhead. Also, the axial deformation can be neglected due to the effects of rotational degrees of freedom on the creation of the plastic hinges. Hence, the modified method proposed by Pellegrino and Calladine [24] and Deeks [25], may be employed.

By indicating the elongation of each member in terms of its displacements in global coordinates, i.e., two displacement components for each joint or node, Eq. 1 can be obtained.

$$e = (d_{x_i} - d_{x_j})\cos\alpha + (d_{y_i} - d_{y_j})\sin\alpha \tag{1}$$

Using matrix notation, Eq. 1 leads to Eq. 2.

$$\mathbf{e} = \mathbf{Cd} \tag{2}$$

where d is the nodal displacement vector, C is the compatibility matrix, and e is the elongation vector.

In an acceptable mechanism, elements do not elongate. Therefore, Eq. 3, should be solved as the principal problem system of equations.

$$\mathbf{e} = \mathbf{Cd} \tag{3}$$

In Eq. 3, the number of columns of matrix C exceeds the number of rows. The difference indicates the number of independent mechanisms. Therefore, Eq. 3 can be decomposed into Eq. 4.

$$[\mathbf{I}, \mathbf{C_d}]\begin{Bmatrix}\mathbf{d^i}\\ \mathbf{d^d}\end{Bmatrix} = \begin{Bmatrix}\mathbf{0}\\ \mathbf{0}\end{Bmatrix} \tag{4}$$

Rearrangement of the Eq. 4, leads to Eq. 5.

$$\mathbf{d^i} = -\mathbf{C^d d^d} \tag{5}$$

The selection of the dependent vectors for dd is the selection of the independent mechanisms. Therefore, the calculation of the di in Eq. 5, leads to a solution to Eq. 3.

The virtual work theorem should be employed to calculate the collapse load factor. The obtained rotations and displacements can be used for the calculation of the internal and external works. Therefore, the collapse load factor can be obtained using Eq. 6.

$$\lambda_c = \frac{W_{\text{int}enral}}{W_{external}} \tag{6}$$

Finally, for obtaining a logical collapse mechanism, the elementary mechanisms should be combined. This procedure can be considered an optimization problem. The objective is to combine elementary mechanisms in a manner that the collapse load factor will be minimized. For this aim, the elementary mechanisms and their corresponding coefficient are considered the decision variables. At least, the problem can be solved using optimization methods, such as meta-heuristic algorithms.

3. **SOFT COMPUTING METHODS**

This section is dedicated to soft computing method details. Firstly, metaheuristic algorithms have been introduced. Due to the application of the Enriched Firefly Algorithm (EFA) in this paper, the details of this method are presented in the following. Finally, the artificial neural networks (ANNs) formulation follows them.

3.1. Metaheuristic Algorithms

In these times, those types of computational algorithms, which are called meta-heuristics have gained wide usage in engineering, applied mathematics, economics, medicine, and other fields. The first category of meta-heuristic algorithms is the nature-inspired one. Some behavior of Animals such as migrating, hunting, flocking, and foraging procedures are very suitable for computational simulation. Therefore, these behaviors can be studied and implemented as swarm intelligent rules to develop an appropriate meta-heuristic algorithm. For instance, one of the most powerful meta-heuristic algorithms that are known as Particle Swarm Optimization or PSO is inspired by the social behavior of birds flocking or fish schooling [26]. As another example, Water Strider Algorithm (WSA) mimics the life cycle of water strider bugs and their intelligent ripple communication [27]. other types of meta-heuristic algorithms are developed according to the physical laws such as Black Holes Mechanics Optimization (BHMO) algorithm [28] and Charged System Search (CSS) algorithm [29]. Also, there are some meta-heuristics based on mathematical models. The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm is one of them [30]. Finally, some meta-heuristic algorithms such as Teaching-Learning Based Optimization (TLBO) algorithm [31] and Tug of War (TOW) algorithm [32] are developed based on human behaviors.

In this paper, one of the recently developed meta-heuristic algorithms, known as the Enriched Firefly Algorithm (EFA), is utilized to improve the performance of artificial neural networks (ANNs). Therefore, FA and its enriched version are explained in the following.

3.2. Firefly Algorithm and its Enriched Version

The basic version of the Firefly algorithm (FA) was presented by Yang [33] and has been applied successfully in either continuous or discrete optimization problems. Although it is proven that FA is a better algorithm than many other optimization meta-heuristic algorithms, there are some drawbacks to its computational processes. For instance, Khadwilard et al. [34] indicated that the FA could not find the optimum solution to some problems and was trapped in the local optima. Therefore, many adaptive, hybrid, chaotic, improved, and enhanced versions of basic FA have been developed so far.

For the implementation of the FA, there are two critical considerations. First, the variation of light intensity, and second, the formulation of attractiveness. The appropriate assumption, for simplicity, is that the attractiveness of a firefly is indicated by its brightness which is, in turn, mapped to the encoded cost function. in minimization cases, the brightness of a firefly at location $\boldsymbol{x}$ can be selected approximately as Eq. 7.

$$I(\boldsymbol{x}) \cong \frac{1}{f(\boldsymbol{x})} \tag{7}$$

where $I(\boldsymbol{x})$ is the brightness, $f(\boldsymbol{x})$ is the objective function, and $\boldsymbol{x}$ is the position vector of the firefly. Therefore, if the cost function has a higher value in this type of problem, the corresponding firefly will have less brightness.

The variations of light intensity and attractiveness are monotonically reducing functions being as the light intensity and the attractiveness reduces, the distance from the source increases, and vice-versa. This procedure is formulated as Eq. 8.

$$I(r) = \frac{I_0}{1 + \gamma r^2} \tag{8}$$

where $I(r)$ is the light intensity, r is the firefly distance, and I_0 is the light intensity at the source. Since the air absorbs part of the light and makes it weaker, the air absorption is modeled mathematically by the light absorption coefficient γ.

In many problems and applications, the combined effect of both can be approximated using the Gaussian form as formulated in Eq. 9.

$$I(r) = I_0 e^{-\gamma r^2} \tag{9}$$

The attractiveness of a firefly (β) is proportional to its brightness (or light intensity). β can be defined as Eq. 10.

$$\beta(r) = \beta_0 e^{-\gamma r^2} \tag{10}$$

where β_0 is the attractiveness at r = 0.

Finally, the ith firefly movement (exploration and exploitation) towards the brighter and more attractive one j is modeled using Eq. 11.

$$x_i(t) = x_i(t) + \beta_0 r^{-\gamma r_{i,j}^2}(x_j - x_i) + \alpha(rand - 0.5) \tag{11}$$

where α is the randomization parameter, *rand* is a random number uniformly distributed between *0* and *1*, and $r_{i,j}$ is the Euclidean distance between fireflies *i* and *j*.

The Enriched version of the basic Firefly Algorithm or EFA is proposed by Kaveh and Seddighian [35]. As they mentioned in their paper, some computational steps in the firefly algorithm increase its computational complexity. The first is to calculate the Euclidean distance between each pair of fireflies. The corresponding computational complexity to this process is equal to $O(N^2)$, where *N* is the number of fireflies. Therefore, the first enrichment is devoted to linearizing this step. There are many programming tricks to handle this drawback. The authors of the EFA paper [35] propose to define a radius ξ as Eq. 12.

$$\xi = \lambda r_{m,n} \tag{12}$$

where λ is the region coefficient corresponding to the problem type, *m* and *n* are the best and worst fireflies in each iteration, respectively. Also, λ can be decreased linearly in each iteration.

Herein, it is possible to select those fireflies that are in a feasible circular region of radius ξ in which the best firefly in each iteration is in the center. Using this trick, the algorithm's computational complexity will be converted to $T \times O(N)$, where *N* is the total number of fireflies, and *T* is the number of fireflies within the feasible region.

The second change in the firefly algorithm that seems to make it more effective is to modify the position vector dimension. Each firefly position vector contains *N* variables, where *N* is the number of decision variables utilized to obtain the Euclidean distance between two fireflies. It is proposed that the corresponding value of the objective function with each firefly be added to the vector position. Therefore, each position vector includes the *N+1* variable. This minor enrichment on the vector position improves the convergence rate of the algorithm impressively.

Finally, the last enrichment applied to the basic version of FA is to change from the Euclidean to the Mahalanobis distance. Since the deviation from the best cost is important, it is essential to consider how each variable indicates similar behavior to the best objective. This tendency can be formulated mathematically using the covariance matrix.

In statistics and probability theory, covariance measures how much two random variables change together. A positive covariance means that the two variables tend to move along, while two variables move inversely if and only if their relative covariance is negative. In the case of two jointly distributed real-valued random variables, the covariance of *X* and *Y* are defined by Eq. 13.

$$cov(X,Y) = E[\big(X - E(X)\big), \big(Y - E(Y)\big)] \tag{13}$$

where *E(X)*, and *E(Y)* is the expected values of *X* and *Y*, respectively.

The covariance matrix is a square and symmetric matrix given by Eq. 14.

$$C_{ij} = cov(X_i, X_j), C \in R^d \tag{14}$$

where d is the number of problem dimensions.

For example, for two-dimensional problems, it can be written as Eq. 15.

$$C = \begin{bmatrix} cov(X,X) & cov(X,Y) \\ cov(Y,X) & cov(Y,Y) \end{bmatrix} \tag{15}$$

where C is the covariance matrix.

The Mahalanobis distance is a measure of the distance between points or vectors. It can be stated that the Mahalanobis distance indicates dissimilarity measured between two vectors. The Mahalanobis distance can be formulated as Eq. 16.

$$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(\boldsymbol{x} - \boldsymbol{y})C^{-1}(\boldsymbol{x} - \boldsymbol{y})} \tag{16}$$

where C is the covariance matrix between position vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. If the covariance matrix is identity, then the Mahalanobis distance will be equal to the Euclidean distance.

The results obtained from the mathematical benchmark functions and other structural ones indicate that the last enrichment increases the convergence rate of the algorithm efficiently.

3.3. Artificial Neural Networks

Nowadays, Artificial Neural Networks (ANNs) are utilized widely to simulate complex systems such as mechanical, Medical, industrial, and many others. In this computational method, the considered variables, have been imported into the system as inputs. The labeled results corresponding to each input have been considered as the network targets. Therefore, this method is categorized as the supervised machine learning method. An activation function tries to simulate a network to convert inputs to targets. The results of the activation function and its weights initially are not compatible with the targets. Therefore, there is an error between outputs (results of the activation function and its weights) and targets should be neglected. Hence, an optimization problem is utilized to minimize this simulation error. Different activation functions, number of layers, optimization algorithms, and other features make different architectures of an ANN.

4. PROPOSED METHOD AND NUMERICAL VALIDATIONS

As introduced before, the computational cost, either in memory or in time, of the conventional methods for plastic analysis is absolutely considerable. Therefore, it is vital to propose a new efficient method for this type of problem.

For this aim, it is proposed in this paper to employ an ANN to detect the collapse mechanism of frames and their corresponding load factor. The procedure is presented in the following.

Firstly, a dataset of the collapse mechanism and its corresponding load factors are generated. For this aim, an implemented python code, based on the Hinge-by-Hinge method [36], is used to create and analyze about 3000 portal frames including different support conditions, geometry properties, loading conditions, and element connectivity. These features are considered as the ANN inputs and the computed collapse mechanism and load factors are assumed as its targets.

Due to the enormous dataset, the optimization procedure of the ANN remains a time-consuming problem. For the treatment of this issue, the EFA method has been employed for optimization. Finally, the trained ANN can be used to detect the collapse mechanism of new frames, in less than 1 second. This efficiency is validated via the following numerical examples. There are no specific units for each example. Any consistent unit system can be employed for simulation.

4.1. Example 1: Two-Bay, Three-Story Frame

The first example, as shown in Fig. 1, is a two-bay, three-story frame that is considered a famous benchmark problem. The geometry and load conditions are passed into the trained ANN and the collapse mechanism and load factor are obtained. Fig. 2 comprises the real mechanism [23] and the obtained one together.

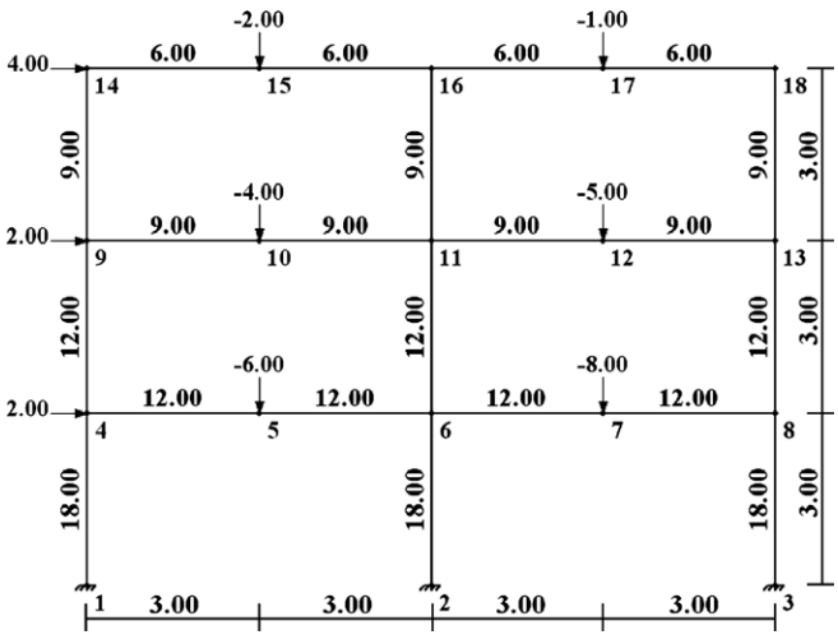
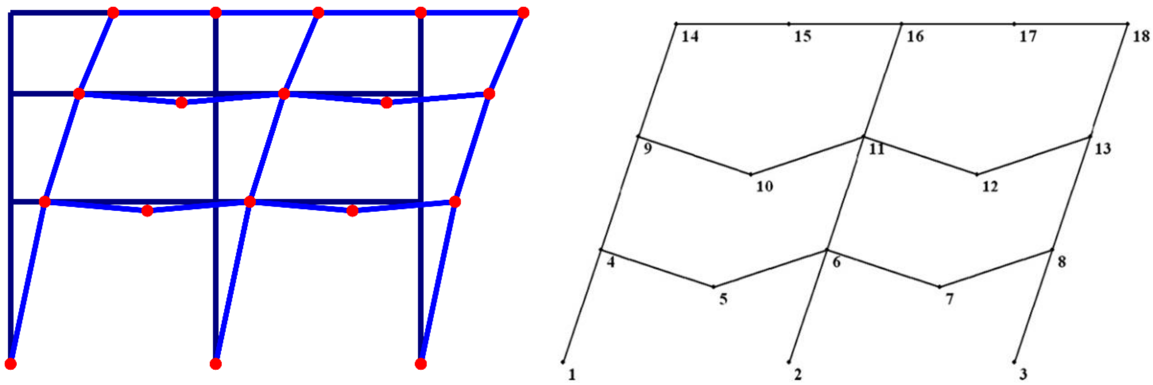


Figure 1. The geometry and loading conditions of example 1.



(a) (b)

Figure 2. The comparison of the real collapse mechanism and obtained one via ANN of Example 1.

The real and computed collapse load factor is equal to 1.97. the most important difference is in the computational time. In a similar computer, the conventional methods require 1463.23 seconds and the proposed method needs 0.005 seconds, for analysis. Therefore, the efficiency of the proposed method as well as its accuracy can be deducted.

4.2. Example 2: Three-Bay, Three-Story Frame

The second example, as illustrated in Fig. 3, is more complicated than example 1. In this example, the collapse mechanism and load factor of a three-story, three-bay frame are investigated.

Similar to example 1, the comparison of the real mechanism and the obtained one is presented (Fig. 4). The collapse load factors for both methods are equal to 1.6. however, the efficiency of the proposed method is about 0.007 seconds in comparison to 1501.44 seconds of the evolutionary methods.

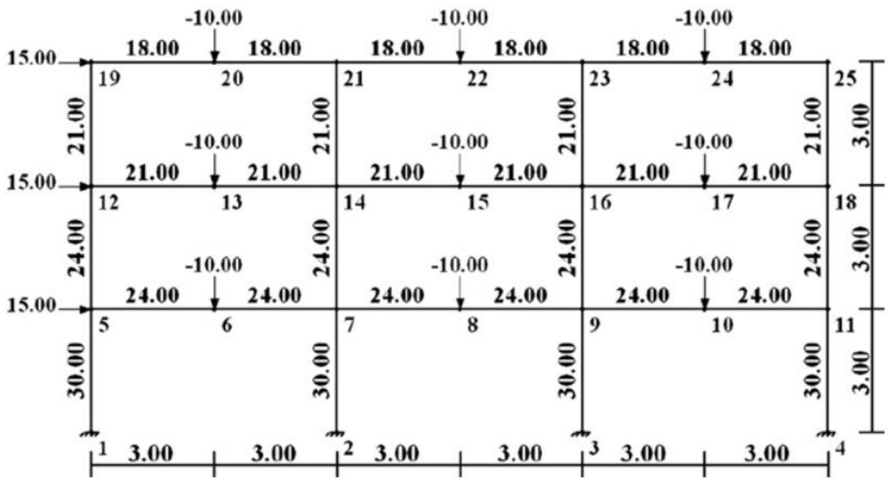
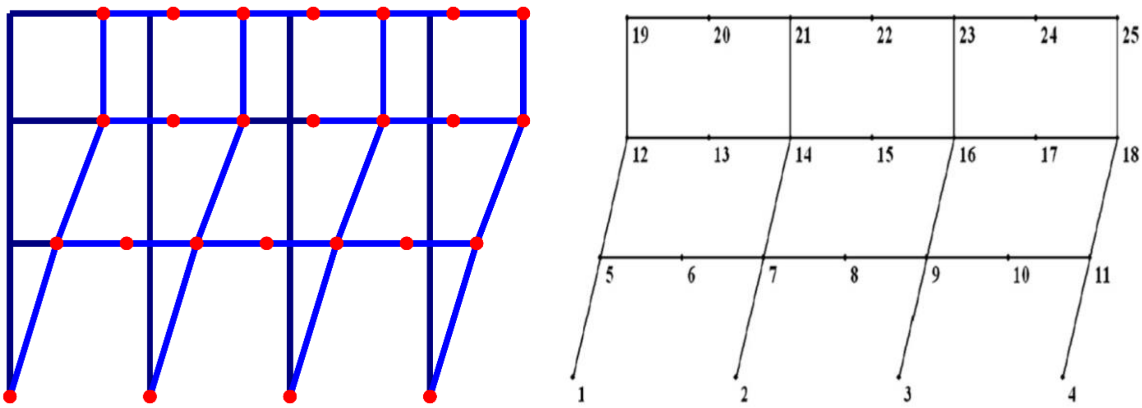


Figure 3. The geometry and loading conditions of example 2.



(a) (b)

Figure 4. The comparison of the real collapse mechanism and obtained one via ANN of Example 2.

Due to the similarity of the collapse mechanisms and load factors of the evolutionary and the proposed method and the obvious differences between their computational costs, it is possible to claim that the new proposed algorithm is considerable, either in accuracy or efficiency.

4.3. Example 4: Two-Bay Gable Frame

The last example is more complicated than the two previous problems. This complexity is due to the slope of the rafters of the gable frame that influences overall on the collapse mechanism. Fig. 5 and Fig. 6 illustrate the problem condition and its comparison with the evolutionary method result, respectively.

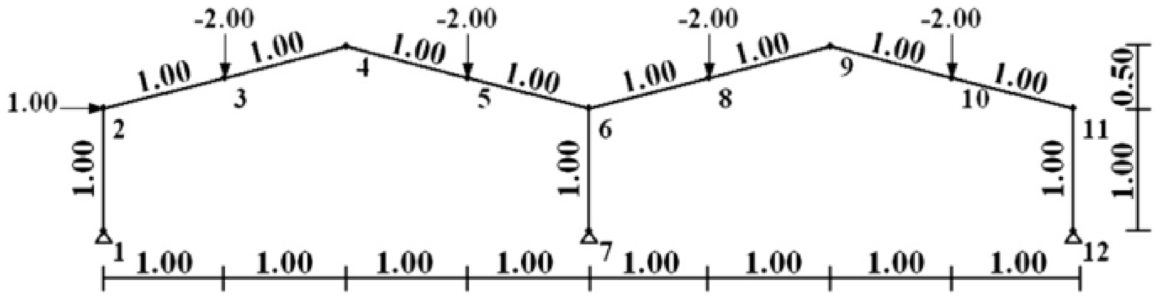
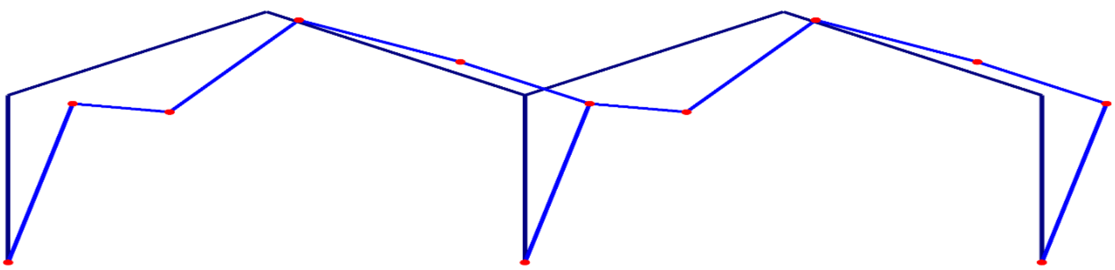
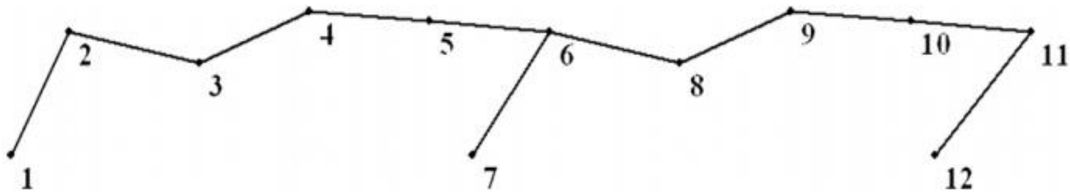


Figure 5. The geometry and loading conditions of Example 3.



(a)



(b)

Figure 6. The comparison of the real collapse mechanism and obtained one via ANN of Example 3.

The actual collapse load factor of example 3 is equal to 1 and both evolutionary and proposed methods have been obtained from it. However, the computational time of the evolutionary methods is nearly 3600 seconds. This time for the newly proposed method is equal to 0.012 seconds.

Similar to previous examples, this example also confirms the efficiency and accuracy of the new proposed algorithm.

5. CONCLUSION

In this paper, a new computational method according to a metaheuristic-based Artificial Neural Network (ANN) has been presented. In this new framework, firstly a computational dataset, including 3000 structures and their corresponding collapse mechanisms and load factors has been generated using the hinge-by-hinge method. After that, a hybrid metaheuristic-ANN framework has been employed to simulate the procedure of the calculation of the collapse mechanism and its corresponding load factor. The trained framework is now capable to obtain actual collapse mechanisms and load factors for new frames. This claim has been proved via three different, challenging, and benchmark problems. All numerical examples confirm the efficiency and accuracy of the new proposed algorithm for plastic limit analysis.

Declarations

Competing interests

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

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

REFERENCES

1. Baker SJF, Horne MR, Heyman J. *The Steel Skeleton*. English Language Book Society, 1961.
2. Neal BG, Symonds PS. The calculation of plastic loads for plane frames, in *International Association for Bridge and Structural Engineering, Fourth Congress*,

1952.
3. Neal BG, Symonds PS. The rapid calculation of the plastic collapse load for a framed structure., *Proceedings of the Institution of Civil Engineers*, 1952;1(2): 58–71.
4. Neal BG, Symonds PS. The calculation of collapse loads for framed structures.(includes appendix)., *J Institut Civil Eng*, 1950; 35(1):21–40.
5. Charnes A, Greenberg HJ. Plastic collapse and linear programming-preliminary report, in *Bulletin of the American Mathematical Society*, 1951; 57(6):480.
6. Heyman J. On the minimum-weight design of a simple portal frame, *Int J Mech Sci*, 1960; 1(1): 121–34.
7. Watwood VB. Mechanism generation for limit analysis of frames, *J Struct Div ASCE*, 1979; 105(1): 1–15.
8. Baker J, Heyman J. *Plastic Design of Frames 1 Fundamentals*. Cambridge University Press, 1969.
9. Jennings PA. *Adapting the Simplex Method to Plastic Design*, 1983.
10. Thierauf G. A method for optimal limit design of structures with alternative loads, *Comput Methods Appl Mech Eng*, 1978; 16(2): 135–49.
11. Horne MR, Determination of the shape of fixed ended beams for maximum economy according to the plastic theory, in *International Association of Bridge and Structural Engineering, Fourth Congress*, 1953.
12. Gorman MR. Automated generation for limit analysis of frames, 1981.
13. Kaveh A. Improved cycle bases for the flexibility analysis of structures, *Comput Methods Appl Mech Eng*, 1976; 9(3): 267–72.
14. Mokhtar-zadeh A, Kaveh A. Optimal plastic analysis and design of frames: graph theoretical methods, *Comput Struct*, 1999, 73(1–5), 485–96.
15. Kaveh A, Khanlari K. Collapse load factor of planar frames using a modified genetic algorithm, *Commun Numer Methods Eng*, 2004; 20(12): 911–25.
16. A. Kaveh A, Rahami H. Analysis, design and optimization of structures using force method and genetic algorithm, *Int J Numer Methods Eng*, 2006; 65(10):1570–84.
17. Palizi S, Saedi Daryan A. Plastic analysis of braced frames by application of metaheuristic optimization algorithms, *Int **J Steel Struct***, 2020; 20(4): 1135–50.
18. Greco A, Cannizzaro F. Pluchino A. Automatic evaluation of plastic collapse conditions for planar frames with vertical irregularities, *Eng Comput*, 2019; 35(1): 57–73.
19. Smail B, Laid SM. Second-order analysis of plane steel structures using Rankine-Merchant-Wood approach, *Asian J Civil Eng*, 2021; 22(4): 701–11.
20. Kaveh A, Bakhshpoori T, Kalateh-Ahani M. Optimum plastic analysis of planar frames using ant colony system and charged system search algorithms, *Scientia Iranica*, 2013; 20 (3): 414–21.
21. Kaveh A, Ghafari MH. Plastic analysis of planar frames using CBO and ECBO algorithms, *Int J Optim Civil Eng*, 2015; 5(4): 479–92.
22.] Kaveh A, Seddighian MR, Ghanadpour E. Upper and lower bounds for the collapse load factor of rectangular grids using FEM, *Int J Optim Civil Eng*, 2019; 9(3): 543–54.
23. Kaveh A, Jahanshahi M. Plastic limit analysis of frames using ant colony systems,

Comput Struct, 2008; 86(11–12): 1152–63.

24. Pellegrino S, Calladine CR. Structural computation of an assembly of rigid links, frictionless joints, and elastic springs, 1991.
25. Deeks AJ. Automatic computation of plastic collapse loads for frames, *Comput Struct*, 1996; 60(3): 391–402.
26. Kennedy J, Eberhart R. Particle swarm optimization, in *Proceedings of ICNN'95-international conference on neural networks*, 1995; 4: 1942–8.
27. Kaveh A, Dadras Eslamlou A, Water strider algorithm: A new metaheuristic and applications, in *Structures*, 2020; 25: 520–41.
28. Kaveh A, Seddighian MR, Ghanadpour E. Black Hole Mechanics Optimization: a novel meta-heuristic algorithm, *Asian J Civil Eng.* 2020; 21(7): 1129–49.
29. Kaveh A, Zolghadr A Topology optimization of trusses considering static and dynamic constraints using the CSS, *Appl Soft Comput*, 2013; 13(5): 2727-34.
30. Iruthayarajan MW, Baskar S. Covariance matrix adaptation evolution strategy based design of centralized PID controller, *Expert Syst Appl*, 2010; 37(8): 5775–81.
31. Rao RV, Savsani VJ, Vakharia DP. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, *Comput-Aided Des*, 2011; 43(3): 303–15.
32. Kaveh A, Zolghadr A. A novel meta-heuristic algorithm: tug of war optimization, *Int J Optim Civil Eng*, 2016; 6(4): 469–92.
33. Yang .-S, Slowik A. Firefly algorithm, in *Swarm Intelligence Algorithms*, CRC Press, 2020; 163–74.
34. Khadwilard A, Chansombat S, Thepphakorn T, Chainate W, Pongcharoen P. Application of firefly algorithm and its parameter setting for job shop scheduling, *J Indust Technol*, 2012; 8(1): 49–58.
35. Kaveh A, Seddighian MR, Simultaneously multi-material layout, and connectivity optimization of truss structures via an Enriched Firefly Algorithm, in *Structures*, 2020; 27: 2217–31.
36. Chen W-F, Zhang H, Structural Plasticity: Theory, Problems, and CAE Software, 2. Springer, 1991.