https://doi.org/10.3311/PPci.13039
Creative Commons Attribution

Periodica Polytechnica Civil Engineering, 63(1), pp. 141–151, 2019

Application of an Improved Genetic Algorithm for Optimal Design of Planar Steel Frames

Mohammad Reza Baradaran[1], Morteza Madhkhan[2]

[1] Civil Engineering group, Pardis College,
Isfahan University of Technology,
Isfahan, 84156-83111, Iran
[2] Department of Civil Engineering,
Isfahan University of Technology,
Isfahan 84156-83111, Iran
* Corresponding author, e-mail: madhkhan@cc.iut.ac

Received: 26 August 2018, Accepted: 14 November 2018, Published online: 10 December 2018

Abstract

Genetic Algorithm (GA) is one of the most widely used optimization algorithms. This algorithm consists of five stages, namely population generation, crossover, mutation, evaluation, and selection. This study presents a modified version of GA called Improved Genetic Algorithm (IGA) for the optimization of steel frame designs. In the IGA, the rate of convergence to the optimal solution is increased by splitting the population generation process to two stages. In the first stage, the initial population is generated by random selection of members from among AISC W-shapes. The generated population is then evaluated in another stage, where the member that does not satisfy the design constraints are replaced with stronger members with larger cross sectional area. This process continues until all design constraints are satisfied. Through this process, the initial population will be improved intelligently so that the design constraints fall within the allowed range. For performance evaluation and comparison, the method was used to design and optimize 10-story and 24-story frames based on the LRFD method as per AISC regulations with the finite element method used for frame analysis. Structural analysis, design, and optimization were performed using a program written with MATLAB programming language. The results show that using the proposed method (IGA) for frame optimization reduces the volume of computations and increases the rate of convergence, thus allowing access to frame designs with near-optimal weights in only a few iterations. Using the IGA also limits the search space to the area of acceptable solutions.

Keywords

optimization, genetic algorithm, steel frame, optimum design

1 Introduction

Optimization is the process of finding the best solution of a problem subject to one or several constraints. In the optimization of structural designs, the choice of variables and specifically the decision to use continuous or discrete variables has a great impact on the volume of computations needed to reach the optimal solution. There are many different methods and algorithms for solving optimization problems, and recent decades have witnessed a steady increase in the number and quality of methods available for this purpose [1]. With the scientific progress in the field of optimization, new approaches have been introduced to reduce the volume of computations as well as the time needed to carry out computational operations. Regardless of the approach, one of the shared goals of optimization methods and algorithms is to reach the optimal solution in the shortest possible time by performing the least amount of computations possible. In the field of structural design, designers now have access to a wide variety of design optimization methods. Nevertheless, the final choice of optimization approach and algorithm heavily depends on the type of problem and the number of variables.

Search algorithms are a variety of optimization algorithms that greatly reduce the volume of computations and the size of design space. These algorithms can be divided into two broad categories of stochastic search algorithms and deterministic search algorithms, each with their own

Cite this article as: Baradaran, M. R., Madhkhan, M. "Application of an Improved Genetic Algorithm for Optimal Design of Planar Steel Frames", Periodica Polytechnica Civil Engineering, 63(1), pp. 141–151, 2019. https://doi.org/10.3311/PPci.13039

advantages and disadvantages. Stochastic search techniques do not need to obtain the derivatives of objective functions and constraints, as they converge to the optimum by checking random solutions [2]. Today, easy access to powerful computers and computer programs has made it easier to obtain an optimal solution in the shortest possible time through the generation of random solutions [3]. Hence, recent years have seen a growing interest in the use of stochastic search techniques in combination with the principles of probability theory in order to develop more effective optimization algorithms. The choice of algorithm to be used for a problem depends on the number of variables and constraints as well as the problem size. For large-scale problems, the use of stochastic search algorithms dramatically increases the rate of convergence to the optimal solution [4]. Major advantages of genetic algorithm over other optimization algorithms include the ability to use both discrete and continuous variables, flexibility in defining constraints, the ability to search large-scale spaces, reduced data processing time, and the ability to use parallel techniques to reach the optimal solution [5]. Genetic algorithm decreases the volume of computations by reducing the search domain at successive iterations; a feature that makes this algorithm very useful for large-scale problems with a large number of variables. Given its simplicity and outstanding features, genetic algorithm has been used as the core of many optimization methods developed for specific problems [6]. For example, Kaveh et al. used the genetic algorithm for the optimization of steel moment-resisting frames with simple and clamped supports [7].

It is reasonable to claim that the nature of the genetic algorithm makes it more suitable than other algorithms for solving large-scale problems [8]. However, as the search space grows, the rate of convergence of this algorithm to the optimal solution decreases. In this paper, the rate of convergence to the optimal solution of a structural design problem is increased by defining the formulations so that if the evaluation of the initial population reveals a failure to satisfy the design constraints, the population members will be replaced with stronger members and this process will continue until all design constraints for every member are met. As a result, the initial population will turn into an improved population where all design variables fall within their respective allowed ranges. Design and optimization of frame members are performed using the LRFD method and based on AISC regulations. The sections to be used in the design are selected from the group of AISC W-shapes.

2 Genetic algorithm

The genetic algorithm generally consists of five stages: population generation, crossover, mutation, evaluation, and selection. Since the introduction of this algorithm by Holland and its formulation by Goldberg, these five stages have evolved to address different needs in different venues of optimization [9]. The genetic algorithm is among the most popular algorithms for the optimization of structural designs, especially for steel structures.

The major methods developed for different stages of the genetic algorithm include: binary method [10], continuous method [11], hybrid method [12], variable length method [13], and multidimensional method [14] for the population generation stage; single-point crossover [15], multi-point crossover [16], heuristic method, and hybrid method [17] for the crossover operation; direct adaptive mutation [18] and power mutation [19] for the mutation operation; single-objective approach [20] and multi-objective approach [21] for the evaluation process; and tournament method [22], roulette wheel methods [23], and rank/merit based methods [24] for the selection operation. In fact, the choice of the method to be used at each stage of the genetic algorithm has a significant impact on its efficiency and effectiveness and how well it can converge to the optimal solution with the least amount of computation [25]. This choice also depends on the type of optimization to be carried out. To solve project scheduling problem with resource constraint, a new hybrid meta-heuristic ANGEL was first developed by Tseng and Chen [26]. Moreover, in addition of developing ANGLE method, Csebfalvi [27] introduced an effective hybrid metaheuristic to deal with discrete size optimization for stability constrained truss structures [28]. It is generally emphasized that the main mission of Angle method [29] is to combine practices of different methods such as gradient-based local search (L) strategy, ant colony optimization (ACO) and genetic algorithm (GA) [30]. According to the literature, for comparison of efficiency in population-based heuristic methods, Csebfalvi proposed another approach which is correct statistically. Such methodologies can lead to more proper solutions for huge number of optimization problems in a rational time range [31].

In the present study, the initial population is generated using the random binary method, and is then improved through an intelligent process until all the design criteria are satisfied. The crossover operation is performed by the simultaneous use of three different methods, namely single-point crossover, two-point crossover, and uniform crossover. The direct adaptive mutation method is used for the mutation operation, and single-objective approach is used for the evaluation. The selection operation is carried out by the simultaneous use of three methods, namely rank/merit based selection, tournament selection, and random selection.

3 Formulation of the optimization problem

In the proposed formulation, the problem variable is the cross-sectional area of the members and the objective function is the weight of the structure. Given the direct relationship between volume and weight, the objective function is defined as Eq. (1A). The problem constraints, including the limits on compressive force, tensile force, shear force, the simultaneous effect of axial force and bending moment, slenderness, cross sectional area, and story drift, are defined based on the LRFD design method as per AISC (2016) regulations. These constraints are presented as Eqs. (1B) to (1L) [32,33].

$$\text{Minimize}: \quad W = \sum_{i=1}^{N} \rho_i A_i L_i. \tag{1A}$$

Subject to:

$$g_1 = P_{ui} - \varnothing_c P_{ni} \quad i = 1.2 \ldots n \tag{1B}$$

$$g_2 = P_{ui} - \varnothing_t P_{ni} \tag{1C}$$

$$g_3 = P_{ui} - \varnothing_v P_{ni} \tag{1D}$$

$$g_4 = \begin{cases} \dfrac{P_{ui}}{P_c} + \dfrac{8}{9}\left(\dfrac{M_{uxi}}{M_{cx}} + \dfrac{M_{uyi}}{M_{cy}}\right) - 1. \ For \dfrac{P_{ui}}{P_c} \geq 0.2 & \text{(1E)} \\ \dfrac{P_{ui}}{2P_c} + \left(\dfrac{M_{uxi}}{M_{cx}} + \dfrac{M_{uyi}}{M_{cy}}\right) - 1. \ For \dfrac{P_{ui}}{P_c} < 0.2 & \text{(1F)} \end{cases}$$

$$g_5 = \begin{cases} \left(\dfrac{KL}{r}\right)_{i.max} - 200. \ For\ compresion\ member & \text{(1G)} \\ \left(\dfrac{L}{r}\right)_{i.max} - 300. \ For\ tension\ member & \text{(1H)} \end{cases}$$

$$g_6 = A_j - A_{j-1}. \quad j = 2.3 \ldots n_c \tag{1I}$$

$$g_7 = \frac{\delta_j - \delta_{j-1}}{h_j} - \delta_{ju} \quad j = 1.2 \ldots n_s \tag{1J}$$

$$g_8 = \Delta_j - \Delta_{max} \quad j = 1.2 \ldots n_s \tag{1K}$$

In Eq. (1A), w is the frame weight, ρ is the density, A is the cross-sectional area, and L is the length of the members. In Eqs. (1B) and (1C), which represent the axial force constraints of the members, P_u is the factored compressive/tensile force of the member, P_n is the nominal compressive/tensile strength of the member, ϕ_t is the tensile strength reduction factor, which is 0.90 for yield and 0.75 for fracture, and ϕ_c is the compressive strength reduction factor, which is equal to 0.9.

In Eq. (1D), which expresses the shear force constraint of the members, ϕ_v is the shear strength reduction factor, which is equal to 0.9, V_u is the factored shear force of the member, and V_n is the nominal shear strength of the member, which must be equal to the smallest value calculated based on shear yielding and shear buckling states.

Equations (1E) and (1F) are the constraints for the members that are subject to both axial force and bending moment. In these equations, P_u is the compressive force, P_c is the design compressive strength of the member, ϕ_c is the compressive strength reduction factor, which is 0.9, M_{ux} is the bending moment about the strong axis x, M_{uy} is the bending moment about the weak axis y, M_{cx} is the design bending strength about the strong axis x, and M_{cy} is the design bending strength about the weak axis y. The design compressive strength of the members is equal to $\phi_c P_n$, and the design bending strengths of the members about the axes x and y are $\phi_b M_{nx}$, and $\phi_b M_{ny}$, respectively. The strength reduction factor ϕ_b is equal to 0.9 and M_{nx} and M_{ny} are the nominal bending strengths of the members.

Equations (1G) and (1H) are the slenderness constraints for the members under compressive axial force. In these equations, K is the effective length factor given by Dumonteil's equation [34], L is the length without lateral anchorage, and r is the radius of gyration of the cross section. For tensile members, the slenderness ratio should not exceed 300. The extra constraints defined to prevent cross-sectional areas from becoming zero and to keep them in the normal range are expressed by Eq. (1I). In this equation, A_j is the cross-sectional area of the columns at upper stories, A_{j-1} is the cross-sectional area of the columns at lower stories, and n_c is the number of columns.

The constraints related to relative and total drifts are expressed in Eqs. (1J) and (1K). In these equations, δ_j is the displacement of the story j, δ_{j-1} is the displacement of the story below the story j, h_j is the height of the story j, Δ_j is the total drift of the story j relative to the base floor, δ_{ju} and $\Delta_{\max}$ are the maximum allowed relative and total story drifts as per AISC 360-16, and n_s is the number of stories. Equation (1J) limits the inter-story drift to δ_{ju} and Eq. (1K) limits the total drift of each story to $\Delta_{\max}$. According to AISC 360-16, the maximum relative drift should be between h/200 and h/600 and the maximum total drift should be between H/100 and H/600 (where *h* is the height of each story and *H* is the elevation of each story from the base floor). In the above equations, the only design variable is the cross sectional area of the members, and other problem parameters such as I_x, I_y, S_x, S_y, r_x, r_y are expressed as a function of cross-sectional area of the members. After quantifying the degree of violation of constraints in each population member using Eq. (1), the sum of g_1 to g_8 for each population member is obtained by the following equation:

$$g_9 = \sum_{i=1}^{8} g_i. \tag{2}$$

In Eq. (2), a positive g_9 is indicative of the violation of constraints. When g_9 is positive, it will be used to calculate a penalty function for the corresponding population member using the following equation:

$$v_z = \sum_{i=1}^{m} g_{9i}. \tag{3}$$

In Eq. (3), v_z is the penalty value that will be applied to the population members (designs) that violate the constraints. To act as a penalty, the normalized value of v_z will be added to the weight of the structure:

$$w_t = w + v_z \times w. \tag{4}$$

The weight value obtained from Eq. (4) will be considered as the final weight of the selected design. In Eqs. (3) and (4), *m* is the number of members, *w* is the frame weight, and w_t is the final frame weight after the application of the penalty function.

In the evaluation stage of the Improved Genetic Algorithm (IGA), any population member that results in the violation of constraints, that is, Eq. (1) becoming positive, will be immediately replaced with a stronger member. In other words, the initial population will be improved until all design constraints of the members are satisfied. Therefore, the improved population obtained after the evaluation stage is guaranteed to meet all of the constraints. This changes the domain of search for frame designs from the area of all possible solutions to the area of acceptable solutions. As a result, the value of v_z approaches zero.

4 Program algorithm

The algorithm of the analysis program is written based on the force-displacement approach. Using this approach, the internal forces of each member are determined with the help of the finite element method. Having the internal forces and the design constraints, the optimal cross sectional area of the members is then determined using the improved genetic algorithm. The program developed for analysis and optimization is written in MATLAB programming language. The general flowchart of the proposed program is illustrated in Fig. 1.

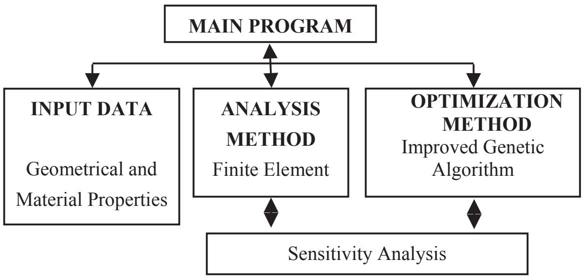


Fig. 1 Main flowchart of the program

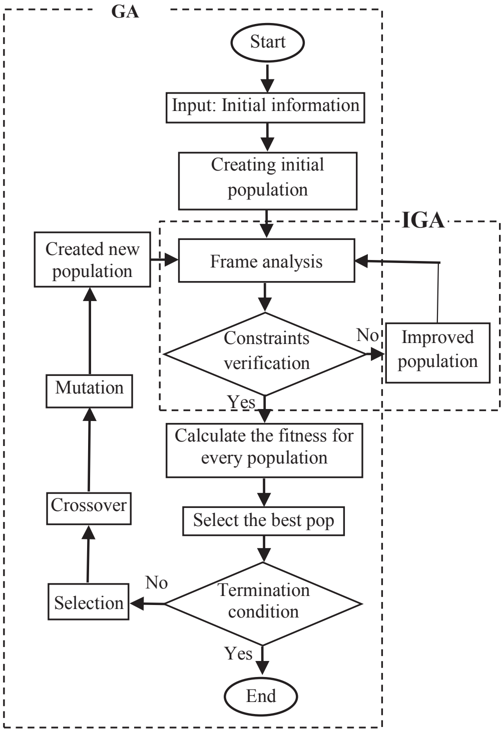


Fig. 2 Flowchart of IGA-based optimum design generation

The flowchart of the presented optimization algorithm is shown in Fig. 2. In this algorithm, the crossover probability is 0.85, mutation probability is 0.3, and mutation rate for each population is 0.05. The parent members to be used in the crossover operation are selected by three methods: random selection, rank/merit based selection, and tournament selection. This operation is performed by simultaneous use of three crossover methods, namely single-point crossover, two-point crossover, and uniform crossover, in order to benefit from the features of all of them in creating optimal solutions.

4.1 Population generation

The initial population is generated by random selection of members from 132 AISC W-shapes. After the generation of the initial population, the finite element method will be used to analyze the frame based on the sections allocated to its members. Once the internal forces are determined, the design constraints will be checked for each member. If any of the design variables are not within the allowed range (when a constraint is violated), the improved genetic algorithm (IGA) will replace the member with a stronger counterpart. This process will be repeated for every constraint and every member to ultimately transform the initial population into an improved population. The frame weight will be determined based on this population and the process will be repeated for other populations. This modification in the evaluation of the initial population significantly increases the rate of convergence to the optimal solution. In other words, the obtained values will approach the optimal solution after a few iterations.

4.2 Selection

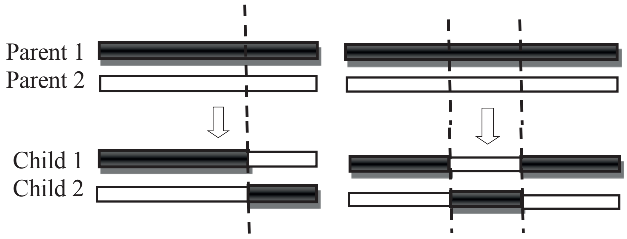
After determining the frame weight for each improved population, some of the populations must be selected for crossover. In this paper, this selection is performed by using random selection, rank/merit based selection, and tournament selection together. In the random selection, all populations have equal chance of being selected. In the rank/merit based selection, the chance of a population being selected depend on the objective function value.

$$P_i = Pr\{I = i\}. \tag{5}$$

In Eq. (5), P_i is the probability of population i being selected. In the rank/merit based selection, the probability of selection must satisfy the three following conditions:

$$\begin{cases} \sum_{i=1}^{npop} P_i = 1 \\ 0 \le P_i \le 1 \\ W_i \le W_j \rightarrow P_i \ge P_j \end{cases} \tag{6}$$

In Eq. (6), *npop* is the number of populations, *P* is the probability of each population, and *W* is the frame weight resulting from the population (with the penalty function taken into account). Equation (6) states that the probability of each population being selected is a value in the range [0,1], the sum of the probabilities for all populations is *1*, and if the final frame weight of population *i* (with the penalty function taken into account) is less than that of population *j*, there is a greater probability for the selection of population *i* than for the selection of population *j*. In Eq. (6), P_i is determined based on the roulette wheel. In the tournament selection, first three populations will be selected at random (with equal probability), then the best among them will be announced as the final selection. In the worst case scenario, the two populations that give the worst solutions will be discarded.



a) single-point crossover b) two-point crossover

Fig. 3 schematic diagram of the crossover operation

4.3 Crossover

In the presented algorithm, single-point crossover, two-point crossover, and uniform crossover are applied simultaneously. In the single-point crossover, the two population members that are selected as parents will be split at a random point and the resulting segments will be swapped with each other (Fig. 3a). In the two-point crossover, parents will be split at two random points and the middle segments will be swapped with each other (Fig. 3b).

In the uniform crossover, the two parents will be combined randomly. More specifically, for two parents such as X_1 and X_2, where:

$$\begin{cases} X_1 = (X_{11}.X_{12} \ \dots \ X_{1n}) \\ X_2 = (X_{21}.X_{22} \ \dots \ X_{2n}) \end{cases} \tag{7}$$

the children, Y_1 and Y_2, where:

$$\begin{cases} Y_1 = (Y_{11}.Y_{12} \ \dots \ Y_{1n}) \\ Y_2 = (Y_{21}.Y_{22} \ \dots \ Y_{2n}) \end{cases} \tag{8}$$

will be calculated as follows:

$$\begin{cases} Y_{1i} = \sum_{i=1}^{n} \alpha_i X_{1i} + (1 - \alpha_i) X_{2i} \\ Y_{2i} = \sum_{i=1}^{n} \alpha_i X_{2i} + (1 - \alpha_i) X_{1i} \end{cases} \tag{9}$$

$$\alpha = (\alpha_1.\alpha_1 \ \dots \ \alpha_n).\alpha \in \{0.1\}.$$

In Eq. (9), *α* is a random number between 0 and 1, which ensures that the parents are combined uniformly.

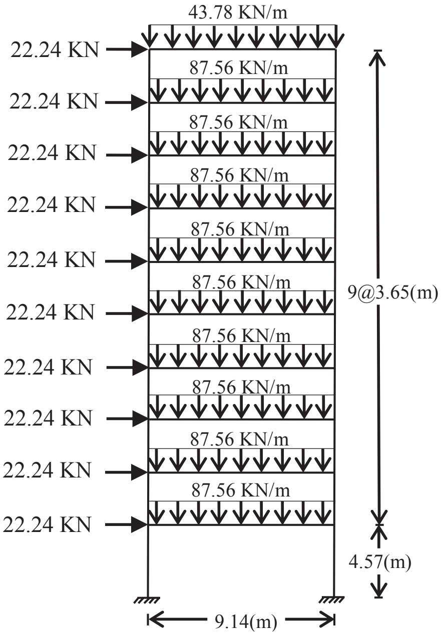


Fig. 4 Topology of the one-bay, 10-story frame

4.4 Mutation

Mutation plays a key role in the rate of convergence to the optimal solution. In the mutation operation, a percentage of the population will be selected to be replaced with other members of the population. In other words, this operation ensures a degree of randomness and diversity in the generated solutions. The mutation rate usually ranges from 5 to 20% depending on the problem. In the present study, the mutation rate is set to 5%.

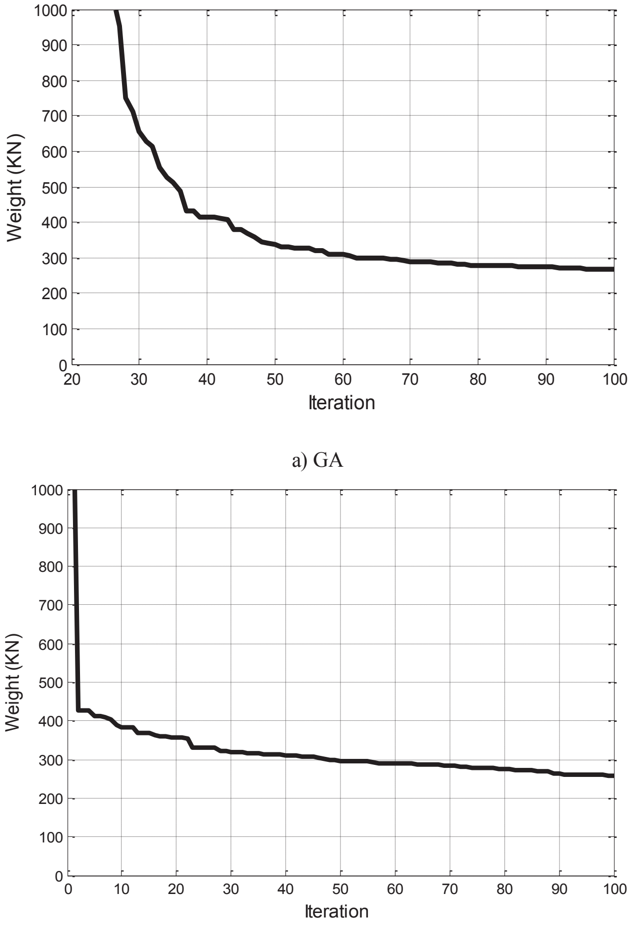
5 Design examples

To illustrate the performance of the proposed method, this section presents the results obtained by designing a one-bay 10-story frame and a three-bay 24-story frame with the Genetic Algorithm (GA) and the Improved Genetic Algorithm (IGA). The results are also compared with the results of other works. Through these two examples, we will show that the improved genetic algorithm is much more effective in optimizing the frames with a higher number of members. In the first example, the members were selected from a group of 132 AISC W-shapes, but in the second one, they were selected from a group of 283 AISC W-shapes. The design was performed using the LRFD method based on AISC 340-16 and AISC 360-16 regulations. Design, analysis, and optimization operations were carried out simultaneously using a program written with MATLAB programming language. In this program, frame analysis was performed using the finite element method and optimization was carried out based on either Genetic Algorithm (GA) or Improved Genetic Algorithm (IGA). At each stage, frame analysis was performed based on the sections generated by the Genetic Algorithm (GA) and improved by the Improved Genetic Algorithm (IGA).

5.1 One-bay 10-story frame

The loading in the one-bay 10-story frame is shown in Fig. 4. This frame consists of 30 members designed and optimized using the LRFD method in accordance with AISC (2016) regulations. Here, members were assumed to have a modulus of elasticity of 200 (GPa) and yield stress of 240 (MPa), and steel density was assumed to be 7680 (N/m^3).

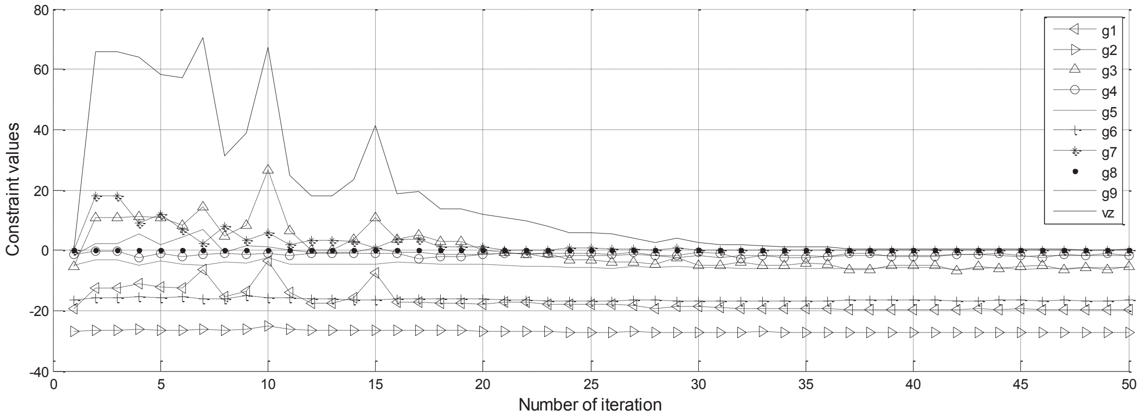
The initial population for the optimization under the loads of Fig. 4 was formed by random selection from a group of 132 AISC W-shapes. The optimum frame weight obtained after 100 iterations of Genetic Algorithm (GA) was 265.8 (KN) while the one obtained using the improved genetic algorithm (IGA) was 258.9 (KN). The diagram of convergence to the optimal solution in both algorithm is displayed in Fig. 5.



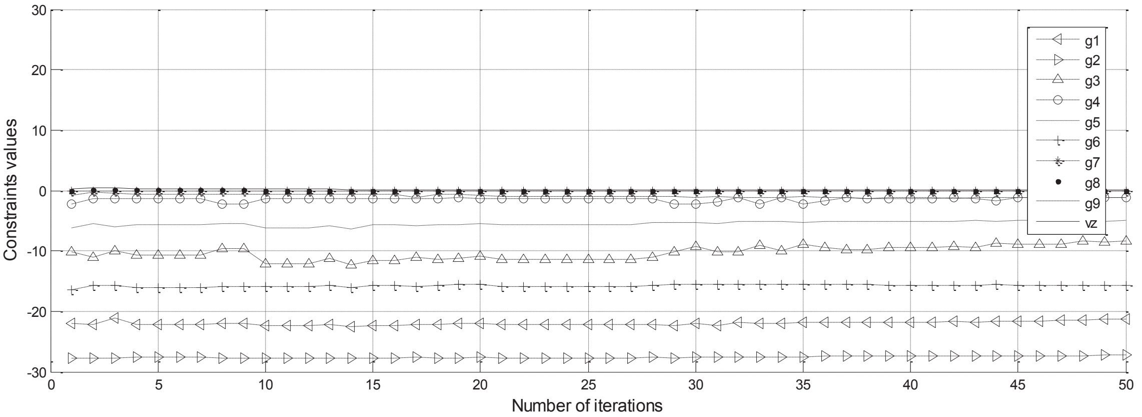
a) GA

b) IGA

Fig. 5 Optimum weight of one-bay, ten story frame



a) GA



b) IGA

Fig. 6 Constraint violation values obtained for one-bay, 10-story frame

As shown in Fig. 5, it takes the genetic algorithm 40 iterations to approach the frame weight of 400 (KN), but the improved genetic algorithm does that in just four iterations. Also, since the selection, crossover, and mutation operations of the IGA are performed on improved populations, the optimum weight is lower than what is achieved with the standard GA.

Fig. 6 shows the values of constraint violation measures g_1 to g_9. The results show that in the GA, these values approach zero after about 40 iterations, but in the IGA, the improvement of the initial population has led to these values becoming zero at the very first iteration. In other words, while it takes about 40 iterations for the genetic algorithm to improve the population, the improved genetic algorithm does this at the beginning of its operation. As shown in Fig. 6, the total constraint violation measure, V_z, is obtained by summing up the individual constraint violation values calculated for each frame member.

As instructed in AISC 340-16, story drift is limited to H/300. The story drift values obtained for this frame are shown in Fig. 7.

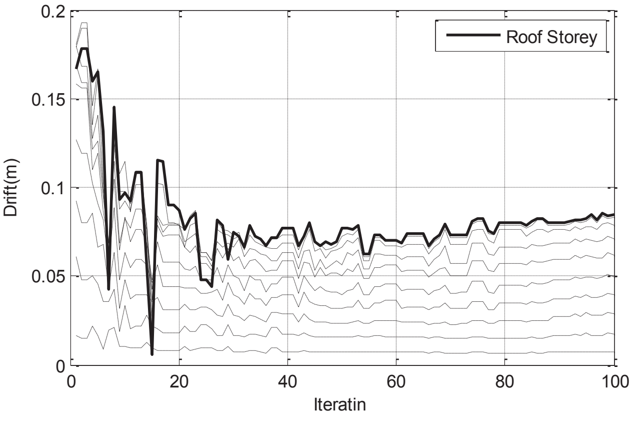


Fig. 7 Drift of stories in the one-bay, ten-story frame

In Table 1, the results of the proposed method are compared with the results of the methods of Pezeshk et al., Camp et al., Degertekin, Kaveh and Talatahari [35–38], which operate based on ant colony optimization, harmonic search, and improved ant colony optimization algorithms (for the same frame). This comparison shows that the optimal weight obtained by the IGA is 2.6% lower than the one obtained using the standard GA.

Table 1 Optimum weight of one-bay, ten story frame (AISC W-Shape)

Pezeshk [35]	Camp [36]	Degertekin [37]	Kaveh [38]	Present work			
				GA		IGA	
33×118^{3}*	30×108^{3}	33×118^{3}	30×90^{3}	27×178^{4}	21×101^{1}	27×178^{3}	18×71^{2}
30×90^{3}	30×90^{3}	30×99^{3}	27×84^{3}	27×146^{2}	21×57^{1}	24×162^{2}	24×68^{1}
27×84^{3}	27×54^{3}	24×76^{3}	27×84^{3}	27×102^{4}	18×119^{1}	27×146^{2}	24×55^{2}
24×55^{1}	21×44^{1}	18×46^{1}	21×44^{1}	24×131^{1}	18×76^{1}	24×131^{1}	18×50^{1}
14×233^{4}	14×233^{4}	14×211^{4}	40×215^{4}	24×84^{1}	18×60^{2}	27×114^{2}	10×12^{1}
14×176^{4}	14×176^{4}	14×176^{4}	30×173^{4}	24×76^{4}	18×55^{1}	24×104^{1}	
14×159^{4}	14×145^{4}	14×145^{4}	24×146^{4}	24×62^{1}	14×82^{1}	27×102^{2}	
14×99^{4}	14×99^{4}	14×90^{4}	21×111^{4}	24×55^{3}	12×72^{1}	18×97^{2}	
12×79^{4}	12×65^{4}	14×61^{4}	12×87^{4}	21×111^{1}		27×84^{8}	
Total weight of frame (KN)							
289.72	278.5	275.18	271.33	265.82		258.95	

* (Member count)

Table 2 Optimum weight of three-bay, 24 story frame (AISC W-Shape)

Saka [40]	Camp [36]	Degertekin [37]	Kaveh [38]	Present work	
				GA	IGA
958.75	980.63	955.74	884.88	884.98	877.98

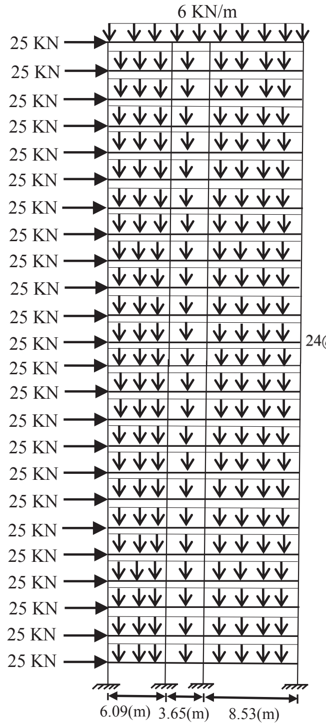


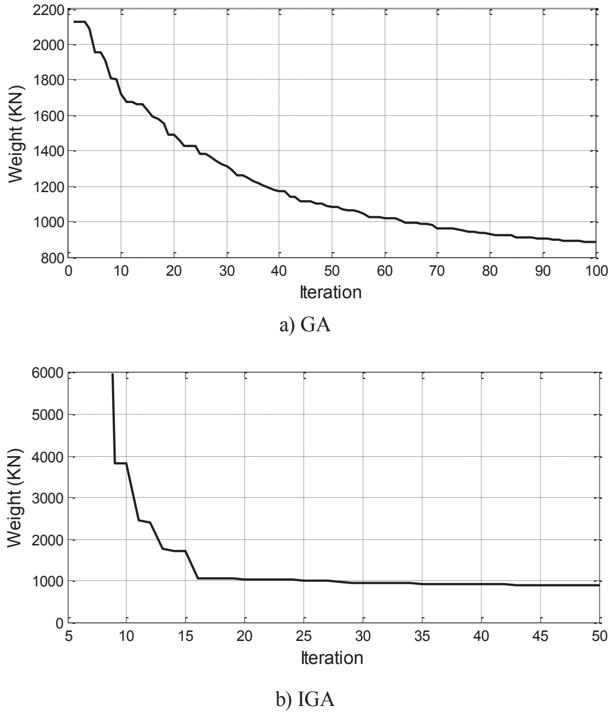
Fig. 8 Topology of the three-bay, 24 story frame

5.2 Three-bay 24-story frame

Fig. 8 shows the diagram of the three-bay 24-story frame under gravitational and lateral loading. This frame was first designed by Davison and Adams [39]. In a study by Saka and Kameshki, they used the genetic algorithm to calculate the minimum weight of this frame based on BS 5950 code [40]. In other studies, Camp et al. optimized this frame with the ant colony optimzation algorithm [36] and Degertekin optimized it with the harmonic search algorithm [37]. In 2012, this frame was optimized by Kaveh and Talatahari using a modified version of the ant colony optimization algorithm [38]. The frame shown in Fig.8 was optimized using the Genetic Algorithm (GA) and the Improved Genetic Algorithm (IGA) with 200 initial populations. The optimal frame weight obtained from these algorithm and other similar methods are presented in Table 2. This frame consists of 168 members. For this frame, modulus of elasticity was assumed to be 200 (GPa), yield stress was assumed to be 240 (MPa), and steel density was assumed to be 7680 (N/m^{3}). It should again be mentioned that the frame members were designed and optimized based on the LRFD method as per AISC 341-16 and AISC 360-16.

The results show that the frame obtained with the IGA is 8% lighter than the one obtained using the GA. The convergence diagram of this frame is plotted in Fig. 9. As shown in this diagram, the IGA approaches the optimal weight after about 16 iterations and ultimately reaches the best weight at 43th iteration. In the GA, however, the random selection of the initial population leads to a far more gradual movement toward the optimal weight, which is so slow that it takes 100 iterations for the algorithm to reach the solution. Fig. 9 shows the convergence rate of the IGA compared to that of the GA.

Fig. 10 shows the drift of the stories in this frame. In ASCE 341-16 code, the maximum allowed total story drift is between H/100 and H/600 and the maximum allowed relative drift is between h/200 and h/600 (where *H* is the elevation of each story from the base floor and *h* is the height



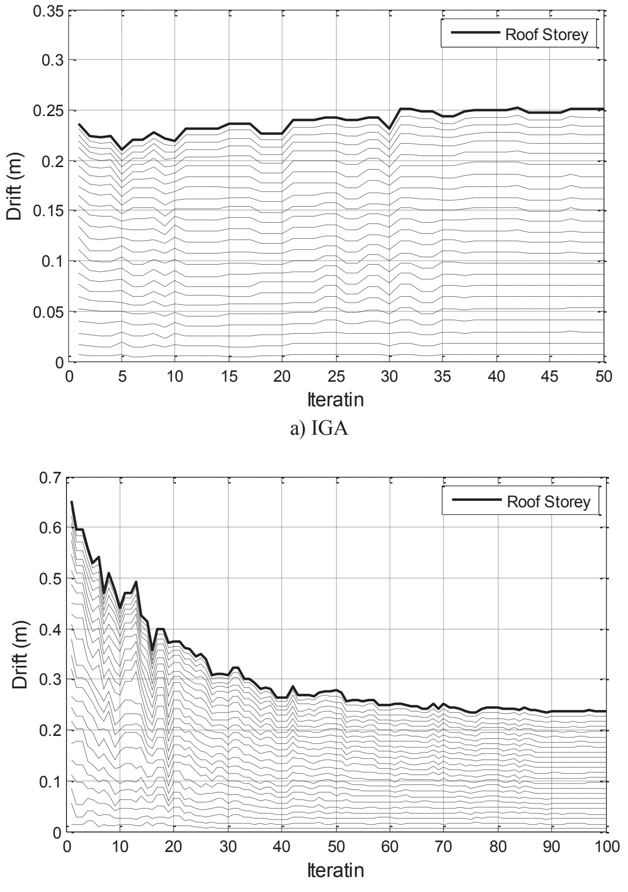
a) GA

b) IGA

Fig. 9 Optimum weight of three-bay, 24-story frame

of each story). In the present study, the maximum allowed drift was considered to be H/300. As shown in Figs. (10-A) and (10-B), with the IGA, the story drift falls in the allowed range right from the start, but it takes many iterations for the GA to reach an acceptable drift.

The values of constraint violation measures g_1 to g_9 in the design of three-bay 24-story frame are plotted in Fig. 11. As shown in Fig. 11, with the IGA, these values have approached zero from the beginning, and this has led to an increase in the rate of convergence. This is because of the improvement of the initial population at the evaluation stage by controlling the constraints.



a) IGA

b) GA

Fig. 10 Drift of stories in the one-bay, 24-story frame

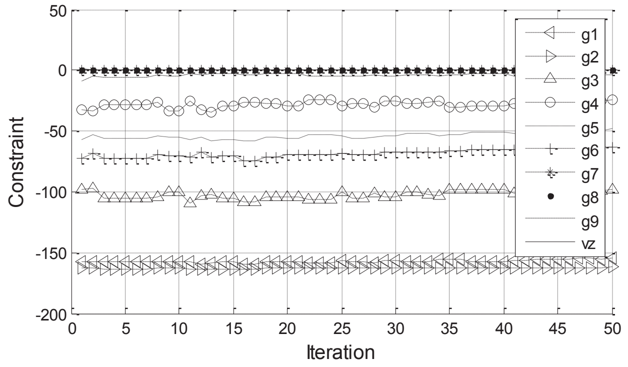


Fig. 11 Constraint violation values obtained for three-bay, 24-story frame

6 Conclusions

Genetic algorithm (GA) is among the most popular and most widely used metaheuristic algorithms used for optimization. This algorithm has shown to be particularly effective in solving design optimization problems with a high number of members. Because of the great size of the randomly generated initial population, this algorithm imposes significant computational burden when solving the problems with large number of variables. In this paper, the randomly generated initial population of the GA is improved through a modification in the evaluation stage in order to achieve better performance for the structural design optimization problem. As a result of this modification, the initial population will be improved at the first evaluation stage through the replacement of the undesirable members with the members that satisfy the design constraints. This technique reduces the volume of computations and increases the rate of convergence to the optimal solution. The results show that as the number of stories and bays in the frame increases, the improved genetic algorithm exhibits better convergence to the design with the optimal frame weight. This approach of the IGA also limits the search space to the area of acceptable solutions.

The use of three different types of crossover and population evaluation methods in the IGA leads to the generation of diverse solutions. In order to avoid uniformity in the improved populations, at each step of the algorithm, new random solutions are added to the initial population. Also, the simultaneous use of three different crossover operations, namely single-point crossover, two-point crossover, and uniform crossover, along with three different selection methods, namely random selection, rank/merit based selection, and tournament selection, has allowed the methods to obtain optimal designs with lower weights.

References

[1] Lógó, J. "New type of optimal topologies by iterative method", Mechanics Based Design of Structures and Machines, 33(2), pp. 149–171, 2005.
https://doi.org/10.1081/SME-200067035

[2] Lógó, J., Ghaemi, M., Vasarhelyi, A. "Stochastic compliance constrained topology optimization based on optimality critera method", Periodica Polytechnica Civil Engineering, 51(2), pp. 5–10, 2007.
https://doi.org/10.3311/pp.ci.2007-2.02

[3] Dao, S. D. "Modelling and intelligent optimisation of production scheduling in VCIM systems", 1st ed., Springer, Cham, Switzerland, 2018.
https://doi.org/10.1007/978-3-319-72113-2

[4] Zhang, C., Lin, Q., Gao, L., Li, X. "Backtracking Search Algorithm with three constraint handling methods for constrained optimization problems", Expert Systems with Applications, 42(21), pp. 7831–7845, 2015.
https://doi.org/10.1016/j.eswa.2015.05.050

[5] Paul, P. V., Moganarangan, N., Kumar, S. S., Raju, R., Vengattaraman, T., Dhavachelvan, P. "Performance analyses over population seeding techniques of the permutation-coded genetic algorithm: An empirical study based on traveling salesman problems", Applied Soft Computing, 32, pp. 383–402, 2015.
https://doi.org/10.1016/j.asoc.2015.03.038

[6] Alberdi, R., Khandelwal, K. "Comparison of robustness of metaheuristic algorithms for steel frame optimization", Engineering Structures, 102, pp. 40–60, 2015.
https://doi.org/10.1016/j.engstruct.2015.08.012

[7] Kaveh, A., Ghafari, M. H., Gholipour, Y. "Optimum seismic design of steel frames considering the connection types", Constructional Steel Research, 130, pp. 79–87, 2017.
https://doi.org/10.1016/j.jcsr.2016.12.002

[8] Kaveh, A. "Advances in metaheuristic algorithms for optimal design of structures", 2nd ed., Springer, Cham, Switzerland, 2017.
https://doi.org/10.1007/978-3-319-46173-1

[9] Goldberg, D. E. "Genetic Algorithms in Search, Optimization, and Machine Learning", 1st ed., Addison-Wesley Longman Publishing, Boston, Massachusetts, USA, 1989.

[10] He, Y., Hui, C.-W. "A binary coding genetic algorithm for multi-purpose process scheduling: A case study", Chemical Engineering Science, 65(16), pp. 4816–4828, 2010.
https://doi.org/10.1016/j.ces.2010.05.032

[11] Csébfalvi, A. "A hybrid meta-heuristic method for continuous engineering optimization", Periodica Polytechnica Civil Engineering, 53(2), pp. 93–100, 2009.
https://doi.org/10.3311/pp.ci.2009-2.05

[12] Chen, C., Xia, J., Liu, J., Feng, G. "Nonlinear inversion of potential-field data using a hybrid-encoding genetic algorithm", Computers and Geosciences, 32(2), pp. 230–239, 2006.
https://doi.org/10.1016/j.cageo.2005.06.008

[13] Dao, S. D., Marian, R. M. "Modeling and optimization of precedence-constrained production sequencing and scheduling for multiple production lines using genetic algorithm", Computer Technology and Application, 2(6), pp. 487–499, 2011.
https://doi.org/10.17265/1934-7332/2011.06.009

[14] Dao, S. D., Abhary, K., Marian, R. "Optimisation of partner selection and collaborative transportation scheduling in Virtual Enterprises using GA", Expert Systems with Applications, 41(15), pp. 6701-6717, 2014.
https://doi.org/10.1016/j.eswa.2014.04.030

[15] Dao, S. D., Marian, R. M. "Optimisation of precedence-constrained production sequencing and scheduling using genetic algorithms", In: International Multi Conference of Engineers and Computer Scientists, Hong Kong, China, 2011, pp. 59–64.

[16] Esen, İ., Koç, M. A. "Optimization of a passive vibration absorber for a barrel using the genetic algorithm", Expert Systems with Applications, 42(2), pp. 894–905, 2015.
https://doi.org/10.1016/j.eswa.2014.08.038

[17] Suresh, S., Huang, H., Kim, H. J. "Hybrid real-coded genetic algorithm for data partitioning in multi-round load distribution and scheduling in heterogeneous systems", Applied Soft Computing, 24, pp. 500–510, 2014.
https://doi.org/10.1016/j.asoc.2014.07.021

[18] Tang, P.-H., Tseng, M.-H. "Adaptive directed mutation for real-coded genetic algorithms", Applied Soft Computing, 13(1), 600–614, 2013.
https://doi.org/10.1016/j.asoc.2012.08.035

[19] Deep, K., Thakur, M. "A new mutation operator for real coded genetic algorithms", Applied Mathematics and Computation, 193(1), pp. 211–230, 2007.
https://doi.org/10.1016/j.amc.2007.03.046

[20] Faghihi, V., Reinschmidt, K. F., Kang, J. H. "Construction scheduling using genetic algorithm based on building information model", Expert Systems with Applications, 41(16), pp. 7565–7578, 2014.
https://doi.org/10.1016/j.eswa.2014.05.047

[21] Aiello, G., La Scalia, G., Enea, M. "A nondominated ranking multi-objective genetic algorithm and electre method for unequal area facility layout problems", Expert Systems with Applications, 40(12), pp. 4812–4819, 2013.
https://doi.org/10.1016/j.eswa.2013.02.026

[22] Castelli, M., Vanneschi, L. "Genetic algorithm with variable neighborhood search for the optimal allocation of goods in shop shelves", Operations Research Letters, 42(5), pp. 355–360, 2014.
https://doi.org/10.1016/j.orl.2014.06.002

[23] Lipowski, A., Lipowska, D. "Roulette-wheel selection via stochastic acceptance", Physica A: Statistical Mechanics and its Applications, 391(6), pp. 2193–2196, 2012.
https://doi.org/10.1016/j.physa.2011.12.004

[24] Boudissa, E., Bounekhla, M. "Genetic algorithm with dynamic selection based on quadratic ranking applied to induction machine parameters estimation", Electric Power Components and Systems, 40(10), pp. 1089–1104, 2012.
https://doi.org/10.1080/15325008.2012.682246

[25] Dao, S. D., Abhari, K., Marian, R. "An innovative framework for designing genetic algorithm structures", Expert Systems with Applications, 90, pp. 196–208, 2017.
https://doi.org/10.1016/j.eswa.2017.08.018

[26] Tseng, L.-Y., Chen, S.-C. "A hybrid metaheuristic for the resource constrained project scheduling problem", European Journal of Operation Research, 175(2), pp. 707–721, 2006.
https://doi.org/10.1016/j.ejor.2005.06.014

[27] Csébfalvi, A. "Hybrid metaheuristic methods in truss optimization: A review", Computational Technology Reviews, 8, pp. 63–92, 2013.
https://doi.org/10.4203/ctr.8.3

[28] Csébfalvi, A. "Angel method for discrete optimization problems", Periodica Polytechnica Civil Engineering, 51(2), pp. 37–46, 2007.
https://doi.org/10.3311/pp.ci.2007-2.06

[29] Csébfalvi, A. "Optimal design of frame structures with semi-rigid joints", Periodica Polytechnica Civil Engineering, 51(1), pp. 9–15, 2007.
https://doi.org/10.3311/pp.ci.2007-1.02

[30] Csébfalvi, A. "ANGEL: A Simplified Hybrid Metaheuristic for Structural Optimization", In: Ant Colony Optimization, 1st ed., IntechOpen Access Publisher, London, United Kingdom, 2013, pp. 107–128.
https://doi.org/10.5772/52188

[31] Csébfalvi, A., Csébfalvi, G. "Fair Comparison of Population-based Heuristic Approaches - The Evils of Competitive Testing", In: Proceedings of the 4th International Joint Conference on Computational Intelligence, Barcelona, Spain, 2012, pp. 306–309.
https://doi.org/10.5220/0004168403060309

[32] ANSI/AISC 360-16 "Specification for Structural Steel Buildings", American Institute of Steel Construction, Chicago, Ilinois, USA, 2016.

[33] ANSI/AISC 341-16 "Seismic Provisions for Structural Steel Buildings", American Institute of Steel Construction, Chicago, Ilinois, USA, 2016.

[34] Dumonteil, P. "Simple equations for effective length factors", Engineering Journal, AISE, 29(3), pp. 111–115, 1992.

[35] Pezeshk, S., Camp, C. V., Chen, D. "Design of nonlinear framed structures using genetic algorithms", Journal of Structural Engineering, 126(3), pp. 382–388, 2000.
https://doi.org/10.1061/(ASCE)0733-9445(2000)126:3(382)

[36] Camp, C. V., Bichon, J., Stovall, S. P. "Design of steel frames using ant colony optimization", Journal of Structural Engineering, ASCE, 131(3), pp. 369–79, 2005.
https://doi.org/10.1061/(ASCE)0733-9445(2005)131:3(369)

[37] Degertekin, S. O. "Optimum design of steel frames using harmony search algorithm", Structural and Multidisciplinary Optimization, 36(4), pp. 393–401, 2008.
https://doi.org/10.1007/s00158-007-0177-4

[38] Kaveh, A., Talatahari, S. "An improved ant colony optimization for the design of planar steel frames", Engineering Structures, 32(3), pp. 864–873, 2010.
https://doi.org/10.1016/j.engstruct.2009.12.012

[39] Davison, J. H., Adams, P. F. "Stability of braced and unbraced frames", Journal of Structural Division, 100(2), pp. 319–324, 1974.

[40] Saka, M. P., Kameshki, E. S. "Optimum design of multi-story sway steel frames to BS 5950 using a genetic algorithm", In: Advances in Engineering Computational Technology, Civil-Comp Press, Edinburgh, United Kingdom, pp 135–141, 1998.
https://doi.org/10.4203/ccp.53.3.6