Available online at www.sciencedirect.com
ScienceDirect
Computers and Structures 86 (2008) 1204–1218
Computers & Structures
www.elsevier.com/locate/compstruc

An improved genetic algorithm with initial population strategy and self-adaptive member grouping

Vedat Toğan *, Ayşe T. Daloğlu

Karadeniz Technical University, Department of Civil Engineering, 61080 Trabzon, Turkey

Received 20 March 2007; accepted 27 November 2007
Available online 25 January 2008

Abstract

The performance of genetic algorithms (GA) is affected by various factors such as coefficients and constants, genetic operators, parameters and some strategies. Member grouping and initial population strategies are also examples of factors. While the member grouping strategy is adopted to reduce the size of the problem, the initial population strategy is applied to reduce the number of search to reach the optimum design in the solution space. In this study, two new self-adaptive member grouping strategies, and a new strategy to set the initial population are discussed. Previously proposed self-adaptive approaches for both the penalty function and the mutation and crossover operators are also adopted in the design. The effect of the proposed strategies on the performance of the GA for capturing the global optimum is tested on the optimization of 2d and 3d truss structures. It is worthy to say that the proposed strategies reduce the number of searches within the solution space and enhance the convergence capability and the performance of the GA.

Keywords: Genetic algorithms; Initial population; Optimization; Self-adaptive member grouping

1. Introduction

Genetic algorithms (GAs) are stochastic optimization techniques based on the mechanics of natural evolution and survival of the fittest strategy found in biological organisms. GAs have been successfully applied to solve many combinatorial optimization problems in business, engineering, and science. Goldberg [1] proposed the concept of competent GAs to solve complex problems quickly, reliably, and accurately without the need for problem-specific coding, operators, or other forms of human intervention. Ahn and Ramakrishna [2] presented a genetic algorithmic approach to the shortest path (SP) routing problem. Ahn et al. [3] presented a memory-efficient elitist genetic algorithm (me2GA) for solving complex optimization problems quickly and effectively. Leite and Topping [4] discussed different types crossover operation in detail and used them in the optimization of structural engineering problems. Topping et al. [5] described the parallel implementations of genetic algorithms. Saka et al. [6] developed a genetic algorithm based method for the optimum design of grid systems. Kaveh and Khanlari [7] used the genetic algorithm to identify the mechanism corresponding to the least possible load factor. Wang and Tai [8] applied the graph representation of the GA to structural topology optimization problems and compared its performance with other methods. Kaveh and Kalatjari [9] performed size and topology optimization of trusses using a genetic algorithm, the force method, and some concepts of graph theory. And today the GA is a well-known method for global optimization of complex systems [10]. One characteristic of the GA that distinguishes it from other optimization algorithms such as gradient-based mathematical programming or simulated annealing is that the GA iterates on population of designs rather than on a single design. This graces a gift to the GA with the potential for converging to a variety of good designs in the final generation [11].

* Corresponding author. Tel.: +90 462 377 2671; fax: +90 462 377 2606.
E-mail address: togan@ktu.edu.tr (V. Toğan).

0045-7949/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compstruc.2007.11.006

The GA starts with a set of randomly selected potential solutions to the problem at hand, and makes them evolve by iteratively applying a set of stochastic operators, known as selection, crossover and mutation. The technique relies on objective function (fitness) evaluation [12]. The better solution will have the higher fitness value. No gradient information is required; only evaluation of the objective function and the constraints are necessary to determine the fitness value. Such a derivative freeness technique makes the GA versatile and gives it the ability to deal with problems with a complicated objective function where derivative is difficult to obtain or unattainable (nondifferentiable function). The stochastic and randomness nature of the GA avoids the gradient-based optimization methods drawback of getting trapped in local optima [10]. More details about the concepts and methods for the structural optimization may be found in [13–15].

The first step of any genetic algorithm is to generate a set of possible solutions randomly as an initial generation or population to the problem. Each member (individual) of the population is usually known as set of chromosomes (phenotypes) and represents a solution for the problem to be investigated. A sampling of this initial population creates an intermediate population. Thus the operators like reproduction, crossover and mutation can be applied to the new intermediate population in order to obtain a new one. The process, which starts from the present population and leads to the new population [16], continues until the desired number of generations is completed or a convergence adopted in the design is reached.

As summarized above, an initial population must be generated to start the evolution process in the GA. There is no condition like selecting all individual constituting the initial population from the solution space of the problem. Therefore, this step is random. Although it seems simple, the convergence, the performance and the ability of the GA are critically affected by the initial generation. If the size of design space is small, these properties of the GA may not be influenced. However, for most practical application of the GA in the area of structural engineering, the solution space is very complicated and large. And member grouping strategies can be adopted to reduce the size of the problem.

In this study, two new self-adaptive member grouping strategies and a new strategy for initial population are proposed. The self-adaptive approaches for both the penalty function, and mutation and crossover given by Toğan and Daloğlu [17] are also adopted in the design. The intention is to catch the global optimum instead of being stuck in a local optimum, and to reduce the number of search within the solution space.

2. A new strategy for the initial population

In the current work, a strategy for the initial population is adopted in the GA process for the intentions mentioned above. The structural members are collected together into different groups to reduce the size of the problem and to make it more practical [17–20]. Number of groups is kept as few as possible, and then needed cross-section areas for the groups are specified. The list of the cross-section values is picked from the list given as input data in the GA process. And the list is assigned to the corresponding design variables as initial values to set the initial population automatically.

2.1. Member grouping

For a given problem, all of the cross-sectional areas of the structural members can be taken as design variables. In this case, however, the computation time gets very high and the result obtained from optimization process might be the local optima due to the expanding design space. Therefore in the GA applications, member grouping is generally applied for the members of the structural system in order to reduce the size of the problem. On the other hand, the member grouping adopted a-priori might not lead an accurate grouping and if the number of members of the structural systems becomes very large, i.e. 3d roof trusses, transmission towers, this leads to very large string lengths, which delays convergence and precludes useful information exchange [19,20].

In the present study, two new member grouping strategies are proposed to reduce the size of the search space of the design problem as much as possible, to increase the probability of catching the global solution and enhance the performance of the GA. A convenient member grouping ends up with the smallest number of cross-section possible in the final set, and reduces the size of the design space as much as possible.

2.1.1. Strategy one

Strategy one is based on the one proposed by previous researchers [17–20], but it is modified. To implement this strategy, the same cross-section areas are assigned for all the structural members first. Following a static analysis with the initial areas for each load cases; the entire range of internal forces is divided into several ranges both for tension and compression members. And structural members are grouped according to the magnitude of their internal forces [17–20].

Toğan and Daloğlu [17] designed the truss structures using three or four groups, one or two for the tension and two for the compression members. An additional group is added to the system for zero force members or members with very low internal forces. It is known that although the zero force members are not necessary to ensure the over all stability of the structure, they should be sized with minimum area specified in the design limitations for the constructive point of view. Hence, a complex solution space is avoided.

2.1.2. Strategy two

For strategy two, while the magnitude of the axial force is considered as the factor for grouping the tension members [17–20], slenderness ratio is considered as the main factor for the compression members to set the groups. Hence, as the tension members of the truss structure are grouped depending on the axial forces, the compression members are grouped according to their slenderness ratio, which is a function of the radius of gyration of the cross-section and the effective length of the member. An extra member group for the zero force members or members with very low internal forces is also arranged.

The groups for the compression members are set following the steps below:

An initial area of cross-section is assigned for all the structural members first. When the axial forces of the members are calculated following the static analysis, the area of cross-sections needed for the compression members are specified according to their slenderness ratio to satisfy following expression

$$\frac{P_i}{A_i} \leqslant \sigma_{ac} \tag{1}$$

where P_i = axial force in compression member i; A_i = cross-sectional area of member i; σ_{ac} = allowable compression stress. The allowable compression stress can be taken to be equal to allowable tension stress at the beginning of the process to calculate an initial cross-section A_i using Eq. (1). Then a section is selected from the list staying on the safe side. After specifying the cross-section, the slenderness ratio of the member is determined depending on the radius of gyration of the section and effective length of the member. The allowable compression stress is determined with the new slenderness ratio, and Eq. (1) is checked. If Eq. (1) is not satisfied, a larger cross-section is picked and the process is repeated until the condition is satisfied [21,22].

Each compression member of the truss is placed into a suitable group according to the value of the radius of gyration of the member. Of course, it is not possible to specify the correct value of radius of gyration satisfying the compression stress at the beginning. However, as known, the GA finds the result satisfying all the constraint(s) and member grouping is independent of the GA process. This operation is meaningful indeed to collect the compression members into the convenient groups.

2.2. Determination of the initial points for the member groups

Since each member group is represented with a design variable, the list number of the maximum A_i within each member group can be a good point to start searching the solution space for the related design variable instead of assigning a randomly generated number. However, since the most structural systems are statically indeterminate, the value of A_i may not be satisfactory, but it will be better than a randomly generated one for the start.

The following steps are implemented to find the initial areas for each design variables (groups) and to create the initial population automatically.

The list number of the member that has the maximum area of cross section, A_i, in the group is taken as the initial point for each group of tension members, and it is stored to create the initial population.

For the groups of the compression member, two or three surplus of the list number of the member that has maximum A_i in the group is taken from the list of sections. Both the value of cross-sectional area, A_i, and radius of gyration of that section must be bigger than the values found previously. And the corresponding section list number gives the initial points and is stored to create the initial population. The rationale of this is that since A_i is calculated as if it is a tension member at the beginning, it is not possible to specify the correct value of radius of gyration that will comply with the required value of compression stress obeying the standard codes, such as AISC-ASD [21], TS 648 [22].

3. Self-adaptive approaches in GAs

In contrast to simple genetic algorithms (SGA), the researchers have recently been proposing new adaptive approaches in the GA for both the penalty function and the mutation and crossover operators to increase the probability of capturing the global optimum, to enhance the performance of the GA and to relieve the user from the burden of having to determine sensitive parameter(s) existed in the GA [17,23–27]. In this study, in addition to a new initial population strategy, adaptive approaches for both the penalty function, and mutation and crossover operators given by Toğan and Daloğlu [17,18] are included in the design. Adaptive approach for the penalty function proposed by Toğan and Daloğlu [17] is slightly modified as follows, Eq. (2) and (3).

$$\Phi(X) = F(X)(1 + \text{penalty}) \tag{2}$$

$$\text{penalty} = g_{ave}\left(\frac{(g_{max} + g(i))}{(g_{max} - g_{ave})}\right) \quad g(i) \geqslant g_{ave} \tag{3a}$$

$$\text{penalty} = g_{ave}\left(\frac{(g_{ave} + g(i))}{(g_{ave} - g_{min})}\right) \quad g(i) < g_{ave} \tag{3b}$$

$$\text{penalty} = 0 \quad g(i) = 0 \quad i = 1, \ldots, m \tag{3c}$$

In Eqs. (2) and (3), X is the vector for the design variables, $F(X)$ is the objective function for minimum volume, n is the number of the total constraints, g_{max}, g_{min}, and g_{ave} represent maximum, minimum and average violation value of the generation respectively. $\Phi(X)$ is modified objective function. $g(i)$ is the total violation value of normalized displacement, $g_{dj}(x)$, and stress, $g_{sr}(x)$, constraints of the ith individual and $g_{dj}(x)$ and $g_{sr}(x)$ are taken as

$$\begin{aligned} g_{dj} &= d_j/d_{uj} - 1 \quad j = 1 \ldots m \\ g_{sr} &= g_r/g_{ar} - 1 \quad r = 1 \ldots l \end{aligned} \tag{4}$$

where d_{uj} is the upper bound of the displacement at jth node and g_{ar} is the allowable stress value in the rth element. m is the number of the displacement constraints and l is the number of stress constraints. The formulation of the unconstrained optimization problem is based on the violations of normalized constraints. In the previous version of these formulation presented by Toğan and Daloğlu [17], g_{ave} is absent.

The rationale of adding g_{ave} as a penalty parameter is that the probability of survival is not eliminated for the designs with minor violations and smaller objective value, and the elimination of the designs with severe violation and bigger objective value from the generation are forced in the algorithm. The adaptive approaches proposed by Toğan and Daloğlu [17] for mutation and crossover operator of the GA are as follows:

$$p_m = 0.5(f_{max} - f)/(f_{max} - f_{ave}) \quad f \geqslant f_{ave} \tag{5a}$$

$$p_m = (f_{ave} - f)/(f_{ave} - f_{min}) \quad f < f_{ave} \tag{5b}$$

$$p_c = (f_{max} - f')/(f_{max} - f_{ave}) \quad f' \geqslant f_{ave} \tag{5c}$$

$$p_c = 1.0 f' < f_{ave} \tag{5d}$$

Here, f is the fitness of individual, f_{ave} average fitness value of the population, f_{max} and f_{min} maximum and minimum fitness value of the population respectively. f' is the lower of the fitness value of the solutions to be crossed.

For adaptive mutation, design variables in an individual are arranged according to the level of violation of the constraints. And then the design variables are renewed by the mutation rate starting with the most violated one. Crossover points can be varied from 1 to string length of the solution by adaptive crossover, and the probabilities of mutation and crossover are calculated by multiplying Eqs. (5) with the string length of each individual in the generation. Probabilities stating the number of design variables to be disrupted for an individual vary according to the fitness value of the individual. Consequently, the adaptive approaches are able to adjust themselves automatically during the evolutionary process. Therefore, the algorithm does not need any pre-defined parameters. Moreover, the crossover operator is capable of performing the flexible point crossover due to its adaptive properties.

4. Numerical examples

The algorithm is tested with various examples. A transmission tower is studied first with the implemented improvements in the study. The minimization of the truss weight and the minimization of the number of cross-section in the final set are obtained without initial population strategy for this example. Later several more numerical examples are solved to investigate and demonstrate the efficiency, accuracy and reliability of the proposed initial population strategy. The test problems include a 10-bar truss subjected to a single load case, a 200-bar planar truss subjected to three different load cases, a 25-bar space truss under two loading conditions, a 120-bar dome space truss subjected to a single load condition, a 240-bar roof truss is also optimized. The population size is taken as 40 for all the examples, except example 1. Real value coding is employed in the genetic algorithm.

At the beginning of the genetic process 40% of the initial population is created by proposed initial population strategy automatically. Therefore the diversity of the population is preserved and the algorithm is protected from being stuck in local minima and also from premature convergence. If all the individuals of the initial population would have been created automatically, the search in the solution space would start in a certain region. Such a process violates all the principles of evolutionary computation search, so it is not meaningful. On the other hand, as the adaptive schemes for both the penalty functions and the mutation and crossover operators are able to adjust themselves automatically during the genetic process [17] they completely disrupt the initial population containing the same individuals.

4.1. Example 1: 244-bar truss tower

The 244-bar transmission tower is selected as the first example to make a comparison between the coded program and reference studies, and to show the efficiency, accuracy and reliability of the implemented improvement in the GA in terms of adaptive approaches and member grouping strategies. Parameters of the optimization process are taken as they are given in the reference studies. The 244-bar transmission tower shown in Fig. 1 was examined by Saka [28] using optimality criteria method under the three load cases. The members of the tower are collected into 26 different groups and the design criteria obeying AISC-ASD [21] is adopted. In addition the stress constraints, Saka [28] also considered the bounds imposed on the displacements for some nodes in the design. The tower was also designed by the Ülker and Hayalioğlu [29] using spreadsheets under the design constraints presented by Saka [28]. Toğan and Daloğlu [17] optimized the same transmission tower adopting the groups of members reported by Saka [28] first. And then the structure was designed again with the groups of members formed after a preliminary analysis and with the adaptive approach in the GA.

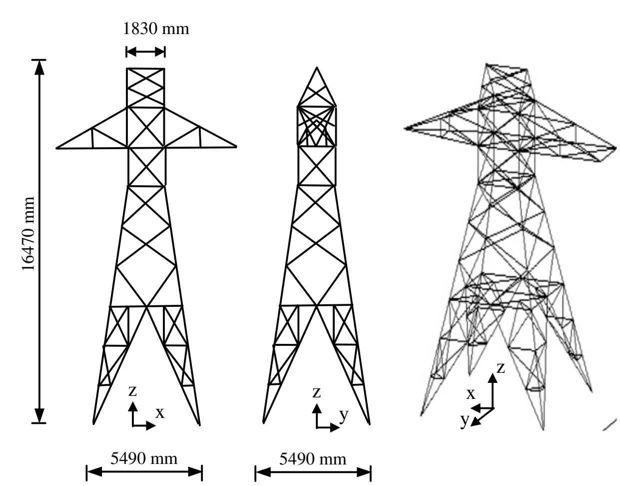


Fig. 1. 244-bar trussed steel transmission tower.

In the current work, the structure is designed using five groups at two for tension, one for zero force members, and two for compression members with the strategy one. Another design is performed setting five groups at two for tension, one for zero force members, and two for compression members forcing strategy two by arranging the compression members depending on their radius of gyration. Population size for this example is 80. Table 1 summarizes the optimized weight for the best solution. The variation of the volume of the tower with the number of generation is plotted in Fig. 2 for both the strategies, while Fig. 3 illustrates the number of member groups according to member grouping strategies. As seen in Fig. 3, some members of the tower moves from one group to another within strategy two. These changes lead more appropriate member grouping and reduce the design space. Also, the implemented improvements proposed in the current work end up with a better design than the one obtained by Toğan and Daloğlu [17].

The final solution consists of an optimized volume and also the least number of sections. Moreover, the designs obtained with the proposed algorithm confirm the intension of "both the weight of structure and the number of groups should be minimized to obtain an economical structure" as indicated in [17–20,30].

4.2. Example 2: 10-bar truss

The 10-bar truss as shown in Fig. 4 is usually used as a standard test problem, a benchmark problem, by researchers to test the results obtained by various optimization methods such as mathematical programming methods and evolutionary algorithms [24,30–38]. Since cross section of each member is taken as a design variable in this problem, there are 10 independent design variables selected from continuous or discrete values. For the sake of comparison of the results given in the references the units are kept in kips and inches. The material density is 0.1 lb/in.^{3}, the modulus of elasticity is 10^4 ksi, and the members are subjected to stress limitation of ± 25 ksi and displacement limitations of 2.0 in. are imposed at each node in both the directions. In this example, two cases are considered: the cross-sectional areas of members are selected from the list given in [35] for Case 1; and for Case 2, list of cross-sections as given in [24] is adopted, and they are presented in Table 2.

This example is stated clearly to clarify the proposed initial population strategy. There is no need to set groups for this example since each cross-sectional area is taken as a group. The axial forces of the members are divided by the allowable stress and the necessary areas of cross-sections are found. For the tension members, next closest area of cross-section staying on the safe side is picked from the list, and the section list number is taken as the individual in the population. For the compression members, three surplus of the calculated area of cross-section is picked and corresponding section list number is included as the individual into the population. Hence, the initial value for each design variable or group is determined and 40% of the individuals of the initial population are created automatically as summarized in Table 3.

The coded value of design variables for the compression member groups of initial population are determined as shown in Table 4 when the buckling is considered as design constraints. The section list number for the compression member groups is increased, i.e. from 21 to 24. Here, both the value of the area of cross-section and the radius of gyration of 24 should be bigger than those of 21.

The truss is optimized using the algorithm improved with the proposed initial population strategy, member grouping strategy, and the adaptive approaches in the GA. After approximately 30,000 searches for both the cases, the best solution vector for Case 1 and 2 are obtained as represented in Tables 5 and 6, respectively. The tables also provide a comparison between the optimal design results reported in the literature and the current work. Figs. 5–8 show how the proposed initial population strategy affects the search within the solution space of the design problem depending on the automatically created individual. Figs. 5 and 6 illustrate a comparison between the results obtained by the random initial population and the proposed initial population strategy while Figs. 7 and 8 display the adaptation of the random initial population and the proposed initial population strategy for both cases. Studying Figs. 5–8, it can be said that proposed initial population strategy offers more coherent individuals than the random ones and increases the adaptation of initial population in the search space. Besides, the search number needed to get an optimum result decreases. Ten runs are performed to plot Figs. 5–8 without worrying if the results are feasible or not.

For Case 1, the individuals that form 40% of the initial population are coded as 21 10 21 12 9 10 18 18 15 12. The value of the objective function is 1816.33 lb and the violation of all constrains is 2.849. For Case 2, the coded value of design variables are 18 1 19 1 1 1 17 16 11 8, the objective function is 2023.594 lb and violation of the constraints is 2.278. The convergence histories of both cases for the minimum weight are plotted as shown in Fig. 9. The algorithm proposed in the current study present less weight for each case than the weights reported in Tables 5 and 6. Moreover, the result for case 1 is lighter than the result obtained with continuous design variables [32,34].

4.3. Example 3: 200-bar truss

A 200-bar plane truss, shown in Fig. 10, is optimized as the last 2d standard problem. The truss was optimized using mathematical methods or evolutionary algorithms by many researchers [33,34,40,41]. In the designs, the truss members were collected in 96 groups, [33], and in 29 groups, [34,40,41]. While only the stress limitation was considered by some of the researchers [34,40,41]. Both the stress and the displacement limitation were considered by some others as the design constraints. In contrast to the others, Coello and Christiansen [41] performed the multi-objective optimization of this truss. In the current work, the parameters used for the design of 200-bar truss are as follows. For the sake of comparison of the results given in the references the units are kept in kips and inches. The material density and modulus of elasticity are 0.283 lb/in.^{3} and 30,000 ksi, respectively. Stress limitations of $\pm$10 ksi is adopted for the truss members and the truss is subjected to three loading conditions: Case 1: one kip acting in positive x direction at nodes 1, 6, 15, 20, 29, 34, 43, 48, 57, 62, and 71; Case 2: 10 kips acting in negative y direction at nodes $1, 2, \ldots, 6, 8, 10, 12, 14, 15, \ldots, 20, 22, 24, 25, \ldots, 73, 74,$ and 75; Case 3: Cases 1 and 2 are combined. The members of the truss are linked into 29 groups as given in Lee and Geem [34] and the truss optimized with the cross-sectional areas provided in [35]. The result related to optimization cases are presented in Table 7. The table also includes some results given in the literature. The results obtained in this study show a remarkable agreement with the previous studies including continuous design variables, [34,40] as listed in Table 7, and a lighter design than the ones obtained using the discrete design variables [33,41]. The algorithm reached an optimum weight of 28544.014 lb, for 29 linked member groups after approximately 51,360 searches. As it is seen in Table 7, the

Table 1
Comparison of results for 244-bar transmission tower

Design variables (mm)	Toğan and Daloğlu [17] at 26 groups	Toğan and Daloğlu [17] at three groups	Toğan and Daloğlu [17] at four groups	This study	
				With the strategy one at five groups	With the strategy two at five groups
A_1		2812	3780	5148.4	3509.7
A_2		3064	2135	1251.6	1548.38
A_3		7096	2696	3064.5	2329.03
A_4			7096	5445.1	5148.38
A_5				461.29	312.26
Volume (cm^3)	920,050	1,561,445	1,377,851	1,025,384.6	921,315.68

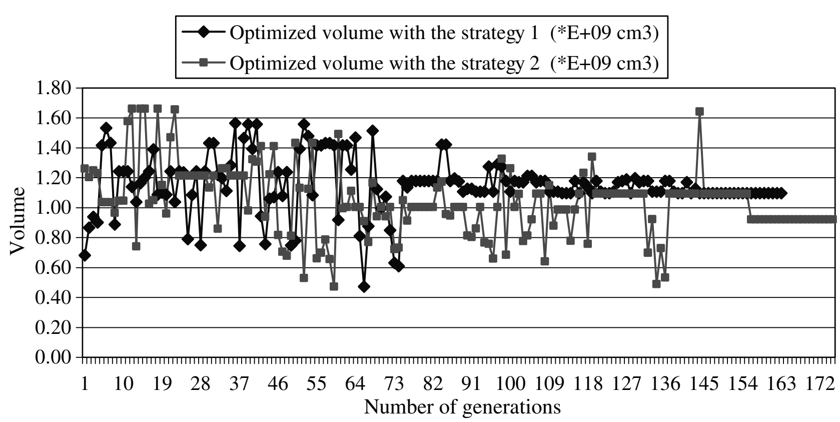


Fig. 2. Variation of volume with number of generations for both members grouping strategies.

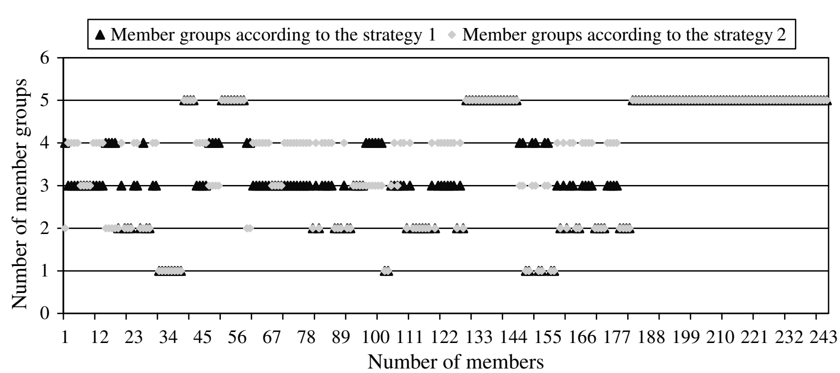


Fig. 3. Member groups for 244-bar tower.

automatically created individual for the initial population is not violated any of the constraints, and the value of the objective function is heavier than the optimal solution. It is observed that if the displacement constraints are not dominant; the adaptation of the automatically created individual for the initial population into the solution space gets very good almost in all cases.

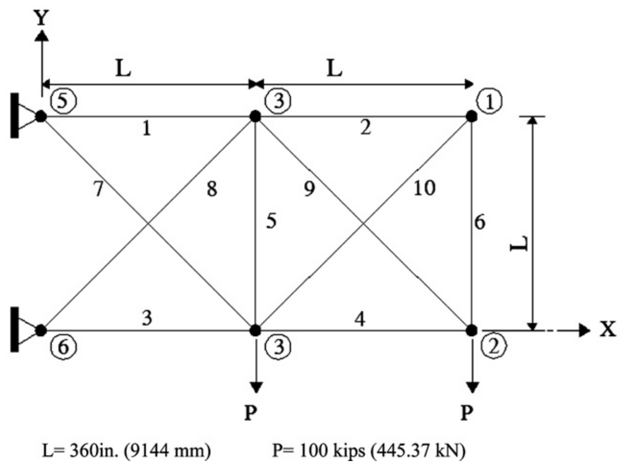


Fig. 4. Ten-bar truss.

Table 2
Discrete section lists given in Refs. [27] and [38]

Reference	S {in.^{2}}
Nanakorn et al. [24	{1.62, 1.80, 1.99, 2.13, 2.38, 2.62, 2.63, 2.88, 2.93, 3.09, 3.13, 3.38, 3.47, 3.55, 3.63, 3.84, 3 3.88, 4.18, 4.22, 4.49, 4.59, 4.80, 4.97, 5.12, 5.74, 7.22, 7.97, 11.5, 13.5, 13.9, 14.2, 15.5, 16 16.9, 18.80, 19.9, 22.0, 22.9, 26.5, 30.0, 33.5}
Jenkins [35]	{0.10, 0.347, 0.440, 0.539, 0.954, 1.081, 1.174, 1.333, 1.488, 1.764, 2.142, 2.697, 2.80, 3.13 3.565, 3.813, 4.805, 5.952, 6.572, 7.192, 8.525, 9.30, 10.850, 13.330, 14.29, 17.17, 19.18, 23 28.08, 33.70}

4.4. Example 4: 25-bar space truss

Next example is a 25-bar space truss as shown in Fig. 11. This problem is the first 3d standard problem to verify the correctness and efficiency of the proposed algorithm in this study. The truss was studied by many researchers for optimum weight using different optimization methods [28,31,32,34,37–42]. Size and shape optimization of the truss was performed by some other researchers [23,43,44]. And the optimum design considering the nonlinear behavior was also done in another research [45]. The first group of researchers kept the truss to be doubly symmetric about X and Y axes while studying for the optimum weight only. And the members of the 25-bar space truss were linked in eight groups to implement it. Some of the researchers optimized the truss under two independent loading cases [28,32,34,40,42] while some others were taken one loading case only [31,38,39]. In this work, two load cases are considered: Case 1, the single loading case given in [31,38,39] is adopted; and Case 2, two independent load cases given in [28,30,32,40] are considered. While the members are subjected to a stress limitations of ±40 ksi in the Case 1, a pre-assigned allowable compressive stress as specified in [28,34] are adopted for Case 2. The units are kept in kips and inches to compare the results given in the references. The material density and modulus of elasticity are 0.1 lb/in.^{3} and 10,000 ksi, respectively. The discrete values for the design variables are taken from [39] and the upper limit for the displacements of joints 1 and 2 are ±0.35 in. in x and y directions.

The best solution vector for the eight design variables are presented in Table 8 after approximately 17,500 searches. Corresponding optimum weights are 487.87 lb and 551.026 lb for Case1 and Case 2, respectively. Table 8 also shows a comparison between the current work and the optimal solutions reported in the literature. Some of these solutions include continuous design variables, while some include discrete values. It can be said that the proposed algorithm reaches a better solution between the discrete ones for the Case 1 and an excellent agreement with

Table 3
Details of the initial population strategy

Truss member	Variables	Axial force (kips)	Needed area (in.^{2})	Selected area from the adopted section list (in.^{2})[a]	Section list number of area	Coded value of design variables for initial population
1	A_1	195.36	7.814	8.525	21	21
2	A_2	40.125	1.605	1.764	10	10
3	A_3	−204.64	8.186	8.525	21	21
4	A_4	−59.875	2.395	2.697	12	12
5	A_5	35.49	1.419	1.488	9	9
6	A_6	40.125	1.605	1.764	10	10
7	A_7	147.98	5.919	5.952	18	18
8	A_8	−134.87	5.395	5.952	18	18
9	A_9	84.677	3.387	3.565	15	15
10	A_{10}	−56.745	2.27	2.697	12	12

Note: 1 in.^{2} = 6.452 cm^2, 1 lb = 4.45 N.
[a] Adopted section list is given in [28]. The initial area for preliminary static analysis = 1.0 in.^{2}.

Table 4
Details of the initial population strategy including buckling constraint

Truss member	Variables	Axial force (kips)	Needed area (in.^{2})	Selected area from the adopted section list (in.^{2})[a]	Section list number of area	Section list number of radius of gyration	Coded value of design variables for initial population
1	A_1	195.36	7.814	8.525	21		21
2	A_2	40.125	1.605	1.764	10		10
3	A_3	−204.64	8.186	8.525	21	21	24
4	A_4	−59.875	2.395	2.697	12	12	15
5	A_5	35.49	1.419	1.488	9		9
6	A_6	40.125	1.605	1.764	10		10
7	A_7	147.98	5.919	5.952	18		18
8	A_8	−134.87	5.395	5.952	18	18	21
9	A_9	84.677	3.387	3.565	15		15
10	A_{10}	−56.745	2.27	2.697	12	12	15

[a] Adopted section list is given in [25]. The initial area for preliminary static analysis=1.0 in.^{2}.

Table 5
Comparison for the 10-bar truss (Case 1)

Variables	Optimal cross-section areas (in.^{2})							
	Venkayya [32]	Thierauf and Cai [33]	Lee and Geem [34]	Jenkins [35]	Rajeev and and KrishnaMoorthy [31]	Kaveh and Kalatjari [37]	Coello et al. [38]	This study
A_1	30.42		30.15	28.08	33.5	29.5	30	28.08
A_2	0.128		0.102	0.1	1.62	0.1	1.62	0.1
A_3	23.41		22.71	23.68	22	23.5	22.9	23.68
A_4	14.91		15.27	17.17	15.5	15.5	13.5	17.17
A_5	0.101		0.102	0.347	1.62	0.1	1.62	0.1
A_6	0.101		0.544	0.1	1.62	0.5	1.62	0.1
A_7	8.696		7.541	7.192	14.2	7.5	13.9	7.192
A_8	21.08		21.56	19.18	19.9	21.5	22	19.18
A_9	21.08		21.45	23.68	19.9	21.5	22	23.68
A_{10}	0.186		0.1	0.1	2.62	0.1	1.62	0.1
Weight (lb)	5084.9	5100	5057.88	5054	5613.84	5067.3	5586.59	5045.6
Max. def. (in.)				2.02				2.0046

Table 6
Comparison for the 10-bar truss (Case 2)

Variables	Optimal cross-section areas (in.^{2})				
	Nanakorn and Meesomklin [24]	Camp et al. [36]	Galante [30]	This study	This study
A_1	33.5	30	33.5	33.5	33.5
A_2	1.62	1.62	1.62	1.62	1.62
A_3	22.9	26.5	22	22.9	22.9
A_4	15.5	13.5	14.2	13.9	16
A_5	1.62	1.62	1.62	1.62	1.62
A_6	1.62	1.62	1.62	1.62	1.62
A_7	7.22	7.22	7.97	7.97	7.97
A_8	22.9	22.9	22.9	22.9	22.9
A_9	22	22	22	22	19.9
A_{10}	1.62	1.62	1.62	1.62	1.62
Weight (lb)	5499.3	5556.9	5458.3	5479.94	5448.62
Max. def. (in.)				2.004	2.0173

Note. 1 in.^{2} = 6.452 cm^2, 1 lb = 4.45 N.

the ones obtained using the continuous design variables for the Case 2 as listed in Table 8. It is observed that since automatically created individual for the initial population is initialized depending on a preliminary analysis without the consideration of deflection, the violation of the displacement constraints individual is high. But the final designs are very satisfactory.

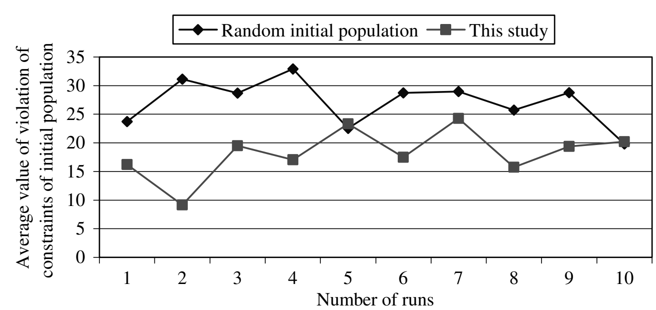


Fig. 5. Comparison of adaptation for the random initial population and the proposed initial population strategy.

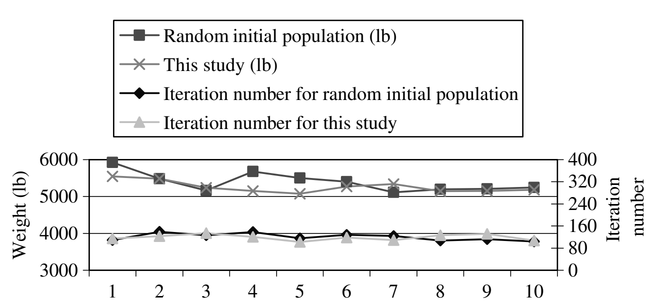


Fig. 6. Comparison of the optimum weight and number of iteration.

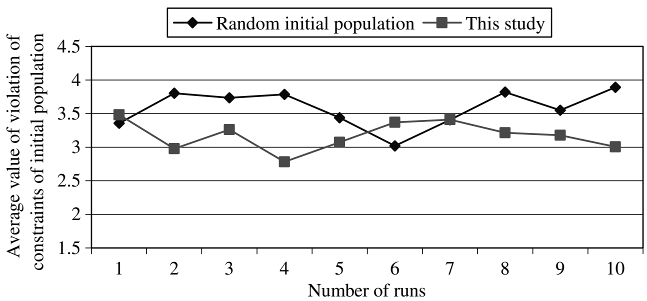


Fig. 7. Comparison of adaptation for the random initial population and the proposed initial population strategy (Case 2).

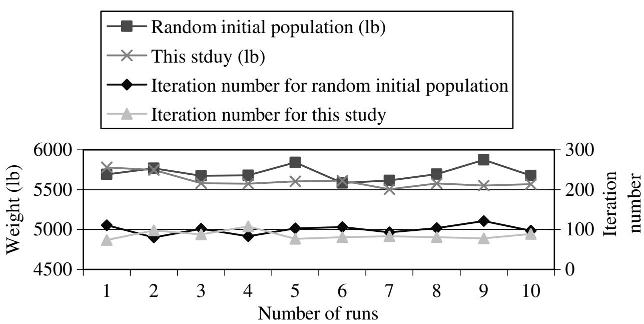


Fig. 8. Comparison of the optimum weight and the number of iteration (Case 2).

4.5. Example 5: 120-bar space truss

The design of 120-bar dome truss, shown in Fig. 12, is investigated as the other standard test problem to demonstrate the efficiency and correctness of the algorithm proposed in this study. The dome truss has 49 joints and 120 members which are collected into seven different groups, Fig. 12. Pipe sections given in AISC-ASD [21] are adopted. The truss is subjected to vertical loading at all the unsupported joints. These are taken as −13.49 kips at node 1, −6.744 kips at nodes 2 through 14, and −2.248 kips at rest of the nodes [34]. In addition to allowable tensile and compressive stresses, an upper limit for the displacement is taken as ±0.1969 in. at each node in x and y directions. The allowable compressive stress is calculated according to AISC-ASD [21]. The other design data needed to start the optimization process are taken from the Lee and Geem [34]. The units are kept in kips and inches for the sake of comparison of the results given in the reference. The minimum cross-sectional area to be assigned for all members is taken as 0.775 in.^{2} after Lee and Geem [34].

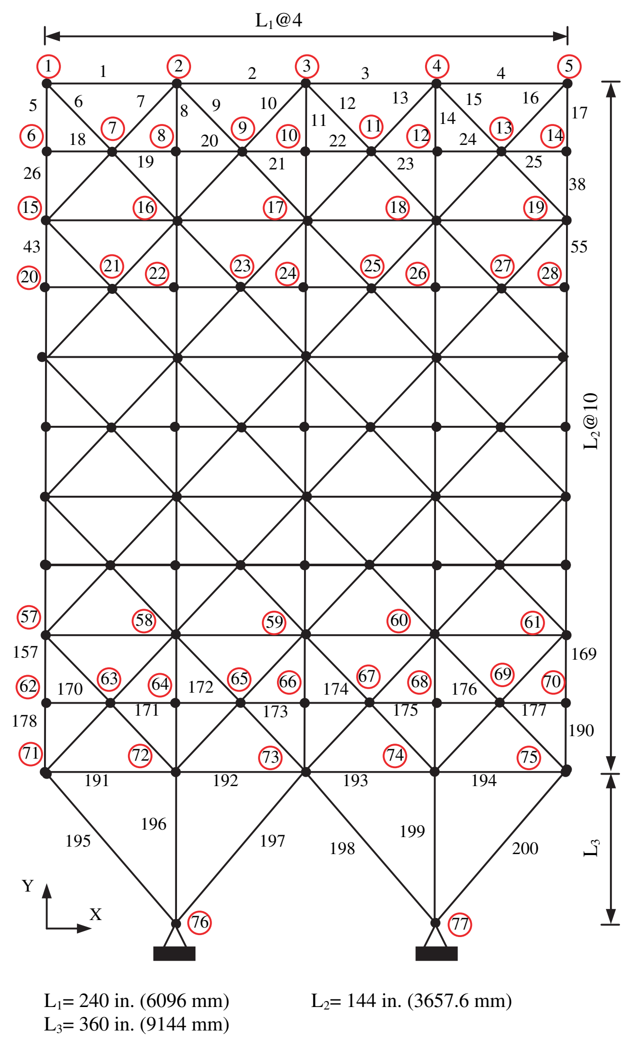


Fig. 10. 200-bar truss.

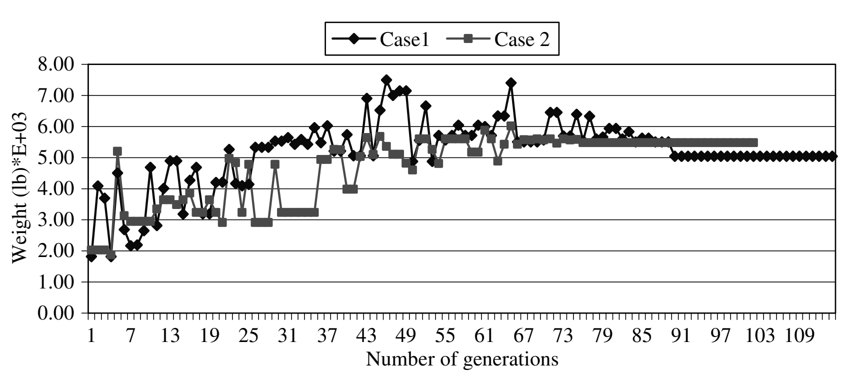


Fig. 9. Convergence history for the minimum weight of 10-bar truss; for Case 1 and 2.

For this example, two design cases are implemented. The dome is designed adopting the group reported by Lee and Geem [34] and Saka and Ülker [45] as Case 1. And then it is designed again with the groups of members formed using the member grouping strategy in this study (strategy two). The fixed member groups include three groups at one in tension and two for compression is described as Case 2. The distribution of elements within the three member groups is illustrated in Fig. 13. Table 9 shows the best solution vectors and the related weights for Case 1 and 2, respectively. Minimum structural weight of 18,293.72 lb is obtained for Case 1; an optimal weight of 17,970.86 lb is reached for Case 2 considering both buckling and displacement constraints. The number of searches needed for both the designs are 32,160 and 29,280, respectively.

The result by Lee and Geem [34] is obtained using continuous design variables, so they are not discrete and not available in practice. Although the discrete set is adopted for the design variables, the final designs of the dome truss obtained in the current study are lighter than those obtained by Lee and Geem [34]. Moreover, when the members are grouped following a preliminary analysis, the weight of the truss gets smaller as listed in Table 9. Thus, the final solution consists of an optimized weight and also the least number of sections.

4.6. Example 6: 240-bar roof truss

The design of 240-bar roof truss, Fig. 14, was first analyzed by Toğan and Daloğlu [18] to show the affect of the member grouping strategy proposed in Toğan and Daloğlu [17] on the optimum volume. They optimized the space truss by using three groups imposed a priori for the purpose. Then the optimal volume of the truss was obtained with three groups of members, one

Table 7
Optimal design comparison for the 200-bar truss

Group	Optimal cross-section areas (in.^{2})					
	Members	Lee and Geem[34]	Lamberti and Pappaletere [40]	Thierauf and Cai [33]	Coello and Christiansen [41]	This work
1	1, 2, 3, 4	0.1253				0.347
2	5, 8, 11, 14, 17	1.0157				1.081
3	19, 20, 21, 22, 23, 24	0.1069				0.1
4	18, 25, 56, 63, 94, 101, 132, 139, 170, 177	0.1096				0.1
5	26, 29, 32, 35, 38	1.9369				2.142
6	6, 7, 9, 10, 12, 13, 15, 16, 27, 28, 30, 31, 33, 34, 36, 37	0.2686				0.347
7	39, 40, 41, 42	0.1042				0.1
8	43, 46, 49, 52, 55	2.9731				3.565
9	57, 58, 59, 60, 61, 62	0.1309				0.347
10	64, 67, 70, 73, 76	4.1831				4.805
11	44, 45, 47, 48, 50, 51, 53, 54, 65, 66, 68, 69, 71, 72, 74, 75	0.3967				0.44
12	77, 78, 79, 80	0.4416				0.44
13	81, 84, 87, 90, 93	5.1873				5.952
14	95, 96, 97, 98, 99, 100	0.1912				0.347
15	102, 105, 108, 111, 114	6.241				6.572
16	82, 83, 85, 86, 88, 89, 91 92, 103, 104, 106, 107, 109, 110,112, 113	0.6994				0.954
17	115, 116, 117,118	0.1158				0.347
18	119, 122, 125, 128, 131	7.7643				8.525
19	133, 134, 135, 136, 137, 138	0.1				0.1
20	140, 143, 146, 149, 152	8.8279				9.3
21	120, 121, 123,124, 129, 127, 129, 130, 141, 142, 144, 145, 147, 148, 150, 151	0.6986				0.954
22	153, 154, 155,156	1.5563				1.764
23	157, 160, 163, 166, 169	10.9806				13.3
24	171, 172, 173, 174, 175, 176	0.1317				0.347
25	178, 181, 184, 187, 190	12.1492				13.3
26	158, 159, 161, 162, 164, 165, 167, 168, 179, 180, 182, 183, 185, 186, 188, 189	1.6373				2.142
27	191, 192, 193, 194	5.0032				4.805
28	195, 197, 198, 200	9.3545				9.3
29	196, 199	15.0919				17.17
	Weight (lb)	25447.1	25446.17	29,737	36167.73	28544.014

Note. The coded values of design variables for the automatically created individuals	Weight (lb)	Violation
3 5 2 18 10 5 5 11 5 14 10 8 15 6 17 12 11 17 11 18 18 18 20 13 21 21 21 26 26	46289.11 0	0.0

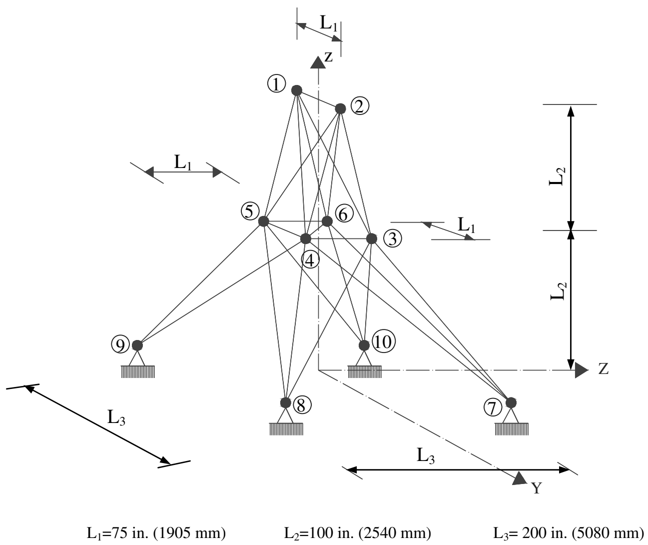


Fig. 11. 25-bar space truss.

for tension and two for compression members, after a preliminary analysis. For these cases, the optimal solutions obtained by Toğan and Daloğlu [18] were 96.97 kN and 48.13 kN, respectively.

The roof truss is analyzed again to investigate the performance and to show the efficiency of the algorithm proposed in the current study. The truss is supported at four corners. The nodal spacing in x and y directions are 1500 mm, and 2500 mm in z direction. Circular hollow sections given in Turkish specification are adopted for the members of the space truss in the study. The edge joints of top chord are subjected to vertical loading of 13,500 N whereas the other joints of top chord are subjected to 15,000 N. The allowable displacements in y direction for the top edge chord joints are restricted to 40 mm while 50 mm is used for the other joints. The allowable tensile stress is taken as 150 N/mm^2 and the modulus of elasticity is 210 kN/mm^2. And the truss members are subjected to compressive stress limitation given in TS 648 [22].

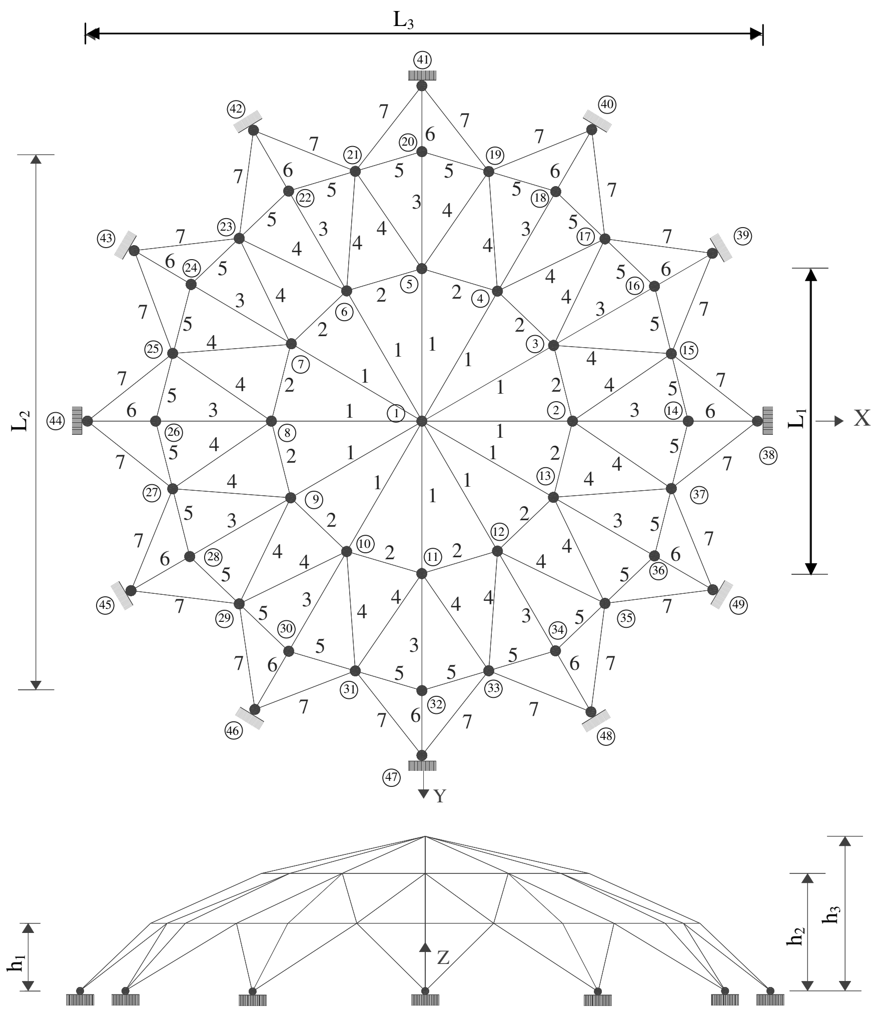
In this study, four groups of members are assumed as one for tension, two for compression, and one for zero force members after a preliminary analysis applying strategy two. The results are presented in Table 10. The best solution obtained in this study is smaller than the results reported by Toğan and Daloğlu [18]. Approximately 25,080 searches are required to reach the final design. This shows that the proposed algorithm is very efficient and effective. Fig. 15 shows the effect of percentage of automatically created individual for the initial population on the performance of the GA. For this investigation, 40%, 50%, 60%, and 70% of population size is created by proposed initial population strategy automatically. It can be concluded that the performance of the GA and

Table 8
Comparison of the optimal designs for the 25-bar space truss

Variables	Optimal cross-section areas (in.^{2}) for Case 1								
	Rajeev and Krishnamoorthy [31]	Erbatur et al. [39]	Coello et al. [38]	Venkayya [32]	Lamberti and Pappalettere [40]	Lee and Geem [34]	Saka [28]	Adeli and Kamal [42]	This work
A_1	0.10	0.10	0.10						0.10
A_2	1.80	1.20	0.70						0.30
A_3	2.30	3.20	3.20						3.40
A_4	0.20	0.10	0.10						0.10
A_5	0.10	1.10	1.40						2.00
A_6	0.80	0.90	1.10						1.00
A_7	1.80	0.40	0.50						0.50
A_8	3.00	3.40	3.40						3.40
Weight (lb)	546.01	493.8	493.94						483.354
Max. def. (in.)									$y_2 = 0.3505$
	Optimal cross-section areas (in.^{2}) for Case 2								
A_1				0.028		0.047	0.010	0.010	0.100
A_2				1.964		2.022	2.058	1.986	2.100
A_3				3.081		2.950	2.988	2.961	2.800
A_4				0.010		0.100	0.010	0.010	0.100
A_5				0.010		0.014	0.010	0.010	0.100
A_6				0.693		0.688	0.696	0.806	0.700
A_7				1.678		1.657	1.670	1.680	1.700
A_8				2.627		2.663	2.592	2.530	2.700
Weight (lb)				545.49	545.17	544.38	545.23	545.66	551.026
Max. Def. (in.)									$y_2 = 0.3489$

Note. The coded values of design variables for the automatically created individuals	Weight (lb)	Violation (stresses + displacements)
For Case 1: 1 2 4 1 1 2 2 4	81.615	11.793
For Case 2: 1 14 11 1 1 7 16 13	359.87	3.048

the best solution vector are not affected critically by increasing percentage of automatically created individuals. However, the number of iteration needed to reach the best solution tends to fluctuate for each percentage. Besides, as the percentage increases, the number of the loops required for the adaptation of the initial population



L_1=546.61 in. (13883.89 mm) L_2=948.25 in. (24085.55 mm) L_3=1251.02 in (31775.91 mm)

h_1=118.11 in. (3000 mm) h_2= 230.32 in (5850.13 mm) h_3= 275.59 in. (7000 mm)

Fig. 12. 120-bar space truss.

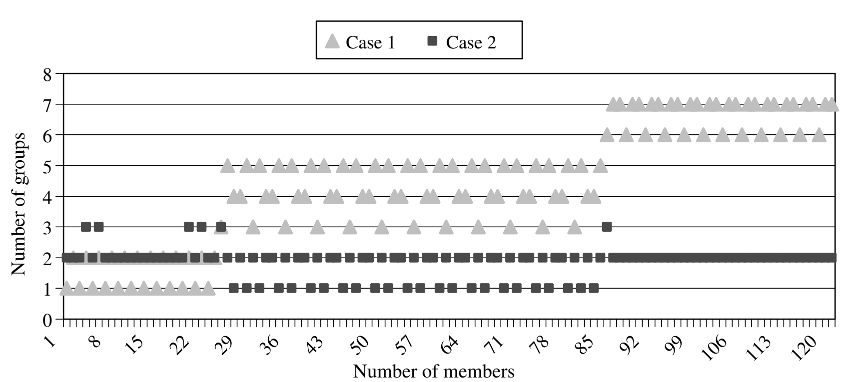


Fig. 13. Member group distributions for 120-bar space truss.

Table 9
Comparison of the optimal designs for the 120-bar space truss

Variables	Optimal cross-section areas (in.^{2})		
	Lee and Geem [34]	This work	
		Case 1	Case 2
A_1	3.296	2.6798	0.7989
A_2	2.789	2.2299	2.6798
A_3	3.872	4.2997	4.2997
A_4	2.570	2.2299	
A_5	1.149	0.7989	
A_6	3.331	3.1698	
A_7	2.781	2.6798	
Weight (lb)	19893.34	18293.72	17970.86

Note. The coded values of design variables for the automatically created individuals

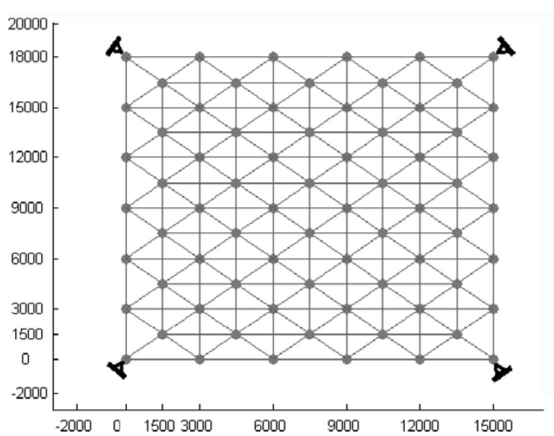
For Case 1	weight (lb)	Violation (stresses + displacements)
4 6 6 4 1 7 4	9033.14	271.146
For Case 2		
1 6 7	11539.94	46.772

into the solution space increases relatively. This indicates that the assumption of creating 40% of population size automatically is reasonable.

5. Conclusions

A new strategy to create an initial population is proposed. The aim is to be protected from being stuck in a local optimum, to get close to the global optimum, and to reduce the number of searches within the solution space. To enhance the performance and ability of the GA, the algorithm also covers the adaptive approach both for penalty function and mutation and crossover operators and a member grouping strategy to form the member groups automatically. Various 2d and 3d trusses are analyzed to demonstrate the effectiveness and accuracy of the initial population strategy. In addition to standard test problems, two large-scale truss examples, example 1 and 6, are presented for the same purposes.

The numerical examples reveal that the proposed algorithm is capable of setting the individual of the initial population and arranging the member groups automatically, and relieving the user from the burden of determining the parameters to start the GA process. Optimal weights or volumes of structures obtained using the proposed algorithm yields better solutions than those obtained by conventional mathematical algorithms, genetic algorithms and HS meta-heuristic search algorithm. So the proposed



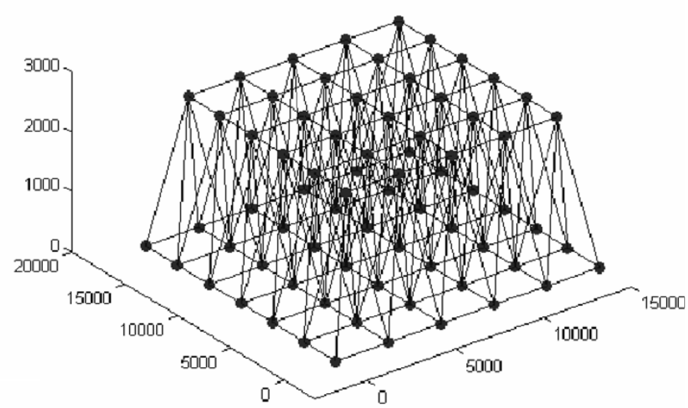


Fig. 14. 240-bar space roof truss.

Table 10
Results for the 240-bar space roof truss

	Weight (kN)	Design variables (mm^2)
Optimum weight at three groups	96.97	$A_1 = 819.96$; $A_2 = 2115.87$; $A_3 = 1552.26$
Optimum weight at three groups after the preliminary analysis	48.13	$A_1 = 394.08$; $A_2 = 1066.89$; $A_3 = 2115.87$
This work	40.866	$A_1 = 394.08$; $A_2 = 819.96$; $A_3 = 2115.87$; $A_4 = 198.49$

Note. The coded values design variables for the automatically created individuals

3 6 11 1	Weight = 39.402 kN	Violation (stresses + displacements) = 16.166

Adopted discrete set (area (mm^2); radius of gyration (mm))

1 – 152.74; 6.7	4 – 394.08; 13.9	7 – 819.96; 25.7	10 – 2115.87; 47.7
2 – 198.49; 8.6	5 – 453.40; 16.0	8 – 1066.89; 30.0	11– 2514.85; 56.6
3 – 306.62; 10.8	6 – 641.26; 20.1	9 – 1552.26; 38.9	

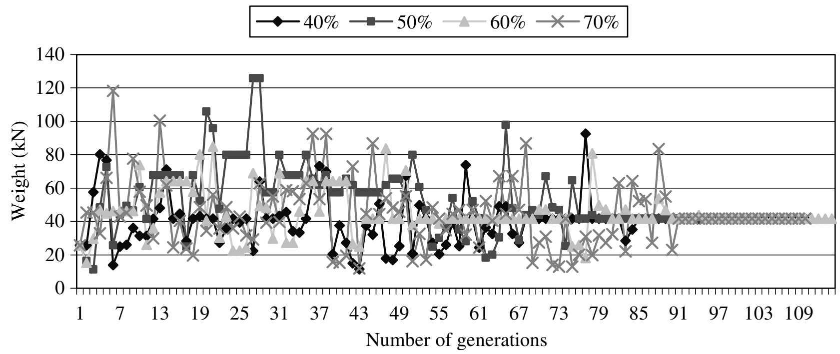


Fig. 15. The effect of the percentage of automatically created individual on the performance.

algorithm is a powerful search and a tool for solving the discrete sizing variables of the structures.

It is concluded that beginning a search with specific individuals instead of generating the initial population randomly enables to reach optimum design with less number of iteration or/and to get closer to the global optimum. It also enables the user to search the solution space better without skipping any of regions. In addition to the initial population strategy, adaptive approaches in the penalty function, mutation and crossover along with the member grouping strategies increase the chance to reach the global optimum and enhance the performance of the GA.

Finally, it must be emphasized that the algorithm proposed is capable of finding the optimum weight or volume with the least number of groups possible to make the design practical. Hence, the solution is feasible and the construction of the structure is easy.

References

[1] Goldberg DE. The design of innovation: lessons from and for competent genetic algorithms. Kluwer Academic Publishers; 2002.
[2] Ahn CW, Ramakrishna RS. A genetic algorithm for shortest path routing problem and the sizing of populations. IEEE Trans Evolut Comput 2002;6(6).
[3] Ahn CW, Kim KP, Ramakrishna RS. A memory-efficient elitist genetic algorithm. In: Wyrzykowski R et al., editors. Proc PPAM 2003, LNCS 3019; 2004. p. 552–9.
[4] Leite JPB, Topping BHV. Improved genetic operators for structural engineering optimization. Adv Eng Softw 1998;29(7–9):529–62.
[5] Topping BHV, Sziveri J, Bahreinejad A, Leite JPB, Cheng B. Parallel processing, neural networks and genetic algorithms. Adv Eng Softw 1998;29(10):763–86.
[6] Saka MP, Daloğlu A, Malhas F. Optimum spacing design of grillage systems using genetic algorithm. In: Topping BHV, Leeming MB, editors. Innovation in computer methods for civil and structural engineering. Edinburg: Civil-Comp Press; 1997.
[7] Kaveh A, Khanlari K. Collapse load factor of planar frames using modified genetic algorithm. Commun Numer Meth Eng 2004;20:911–25.
[8] Wang SY, Tai K. Graph representation for structural topology optimization using genetic algorithms. Comput Struct 2004;82(20–21):1609–22.
[9] Kaveh A, Kalatjari V. Topology optimization of trusses using genetic algorithm, force method and graph theory. Int J Numer Meth Eng 2003;58(5):771–91.
[10] Alkhatib R, Jazar GN, Golnaraghi MF. Optimal design of passive linear suspension using genetic algorithm. J Sound Vib 2004;275:665–91.
[11] Balling RJ, Briggs RR, Gillman K. Multiple optimum size/shape/topology designs for skeletal structures using a genetic algorithm. J Struct Eng, ASCE 2006;132(7):1158–65.
[12] Michalewicz Z. Genetic algorithms + data structures = evolution programs. New York: Springer; 1996.
[13] Haftka RT, Gürdal Z. Element of structural optimization. Academic Publisher: Kluwer; 1992.
[14] Xie Ym, Steven GP. Evolutionary structural optimization. Springer; 1997.
[15] Gutkowski W. Discrete structural optimization. Springer; 1997.
[16] Ramirez-Rosado IJ, Bernal-Agustin JL. Genetic algorithms applied to the design of large power distribution systems. IEEE Trans Power Syst 1998;13(2):696–703.
[17] Toğan V, Daloğlu AT. Optimization of 3d trusses with adaptive approach in genetic algorithms. Eng Struct 2006;28:1019–27.
[18] Toğan V, Daloğlu AT. Adaptive approaches in genetic algorithms to catch the global optimum. In: Proc 7th international congress advances in civil engineering, Istanbul; 2006. p. 244.
[19] Krishnamoorthy CS, Venkatesh PP, Sudarshan R. Object-oriented framework for genetic algorithms with application to space truss optimization. J Comput Civil Eng, ASCE 2002;16:66–75.
[20] Sudarshan R. Genetic algorithms and application to the optimization of space trusses, A Project Report, Madras (India), Indian Institute of Technology; 2000.
[21] American Institute of Steel Construction (AISC). Manual of steel construction-allowable stress design. 9th ed., Chicago; 1989.
[22] Turkish Standard Institute (TSE). Building code for steel structures TS 648, Ankara, Turkey; 1982.
[23] Kaveh A, Kalatjari V. Size/geometry optimization of trusses by the force method and genetic algorithm. Z Angew Math Mech 2004;84(5):347–57.
[24] Nanakorn P, Meesomklin K. An adaptive penalty function in genetic algorithms for structural design optimization. Comput Struct 2001;79:2527–39.
[25] Barbosa HJC, Lemonge ACC. A new adaptive penalty scheme for genetic algorithms. Inform Sci 2003;156:215–51.
[26] Coello CAC. Use of a self-adaptive penalty approach for engineering optimization problems. Comput Ind 2000;41:113–27.
[27] Coit DW, Smith AE, Tate DM. Adaptive penalty methods for genetic optimization of constrained combinatorial problems. INFORMS J Comput 1996;8(2):173–82.
[28] Saka MP. Optimum design of pin-jointed steel structures with practical applications. J Struct Eng, ASCE 1990;116(10):2599–620.
[29] Ülker M, Hayalioğlu MS. Optimum design of space trusses with buckling constraints by means of spreadsheets. Turkish J Eng Environ Sci 2001;25:355–67.
[30] Galante M. Genetic algorithms as an approach to optimize real-world trusses. Int J Numer Meth Eng 1996;39:361–82.

[31] Rajeev S, Krishnamoorthy CS. Discrete optimization of structures using genetic algorithms. J Struct Eng, ASCE 1992;118(5):1233–50.
[32] Venkayya VB. Design of optimum structures. Comput Struct 1971;1(1–2):265–309.
[33] Thierauf G, Cai J. Parallelization of evolution strategy for discrete structural optimization problems. Comput-Aided Civ Inf Eng 1998;13:23–30.
[34] Lee KS, Geem ZW. A new structural optimization method based on the harmony search algorithm. Comput Struct 2004;82:781–98.
[35] Jenkins WM. A decimal-coded evolutionary algorithm for constrained optimization. Comput Struct 2002;80:471–80.
[36] Camp C, Pezeshk S, Cao G. Optimized design of two-dimensional structure using genetic algorithm. J Struct Eng, ASCE 1998;124:551–9.
[37] Kaveh A, Kalatjari V. Genetic algorithm for discrete-sizing optimal design of trusses using the force method. Int J Numer Meth Eng 2002;55:55–72.
[38] Coello CA, Rudnick R, Christiansen AD. Using genetic algorithms for optimal design of trusses. In: Proc of the sixth international conference on tools with artificial intelligence. New Orleans, LA, Silver Spring MD: IEEE Computer Soc. Press; 1994. p. 88–94.
[39] Erbatur F, Hasançebi O, Tütüncü I, Kılıç H. Optimal design of planar and space structures with genetic algorithms. Comput Struct 2000;75:209–24.
[40] Lamberti L, Pappalettere C. Move limits definition in structural optimization with sequential linear programming-Part II: numerical examples. Comput Struct 2003;81:215–38.
[41] Coello CA, Christiansen AD. Multiobjective optimization of trusses using genetic algorithms. Comput Struct 2000;75:647–60.
[42] Adeli H, Kamal O. Efficient optimization of space trusses. Comput Struct 1986;24(3):501–11.
[43] Toğan V, Daloğlu A. Shape and size optimization of 3d trusses with genetic algorithm. Technical J Turkish Chamber Civil Eng 2006;17:3809–26.
[44] Wu SJ, Chow PT. Integrated discrete and configuration optimization of trusses using genetic algorithms. Comput Struct 1995;55(4):695–702.
[45] Saka MP, Ülker M. Optimum design of geometrically nonlinear space trusses. Comput Struct 1991;41(6):1387–96.