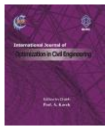
INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2015; 6(1):101-114



A FAST GA-BASED METHOD FOR SOLVING TRUSS OPTIMIZATION PROBLEMS

M. Salar[1], M.R. Ghasemi[1*, †] and B. Dizangian[2]
[1]*Department of Civil Engineering, University of Sistan and Baluchestan, Zahedan, Iran*
[2]*Department of Civil Engineering, Velayat University, Iranshahr, Iran*

ABSTRACT

Due to the complex structural issues and increasing number of design variables, a rather fast optimization algorithm to lead to a global swift convergence history without multiple attempts may be of major concern. Genetic Algorithm (GA) includes random numerical technique that is inspired by nature and is used to solve optimization problems. In this study, a novel GA method based on self-adaptive operators is presented. Results show that this proposed method is faster than many other defined GA-based conventional algorithms. To investigate the efficiency of the proposed method, several famous optimization truss problems with semi-discrete variables are studied. The results reflect the good performance of the algorithm where relatively a less number of analyses is required for the global optimum solution.

Kewords: genetic algorithm- optimization- structural optimization.

Received: 15 August 2015; Accepted: 2 October 2015

1. INTRODUCTION

In recent years, a great attention has been paid to structural optimization, since material consumption is one of the most important factors influencing building costs, while reducing the mass of structures normally is considered desirable for seismic behavior. Designers are able to produce better designs while saving time and money through optimization. Optimal design of truss-structures has always been an active research area in the field of search and optimization [1-10]. Generally, truss optimization can be classified into three main categories [11]: 1- Sizing optimization, 2- Configuration optimization and 3- Topology optimization.

[*]Corresponding author: Department of Civil Engineering, University of Sistan and Baluchestan, Zahedan, Iran
[†]E-mail address: mrghasemi@hamoon.usb.ac.ir (M.R. Ghasemi)

In the sizing optimization of the truss, cross-sectional areas of members are considered as design variables, while the coordinates of nodes and connectivity between various members are considered to be fixed [7, 12-14]. In the configuration optimization of the truss, the coordinates of the nodes are considered as design variables, while the connecting member is fixed nodes [15-16]. In the topology optimization of the truss, connecting members as design variables considered [6, 11, 17-18].

Many optimization problems in science and engineering involve constraints. The presence of constraints reduces the feasible region and complicates the search process. In addition, when solving constrained optimization problems, solutions that satisfy all the constraints are feasible individuals while individuals that fail to satisfy any of the constraints are infeasible individuals [19].

Various methods have been developed and used for optimization of truss structures. Optimization methods using mathematical programming techniques are used for discrete structural optimization and they are found to be useful in solving a few classes of problems [20-22]. Many evolutionary algorithms such as genetic algorithms [7, 12, 23], particle swarm optimization [13, 14], ant colony optimization [24], and artificial bee colony algorithms [25] are used to solve truss optimization problems. In this paper, semi-discrete optimization of truss and several standard test functions using proposed genetic algorithms will be considered.

Optimization by genetic algorithm is one of the useful methods for optimization, but slow process and the lack of recognition and understanding of the parameters that govern them, such as the number of generations, crossover rate, mutation rate and the type of selection, in some cases, cause the algorithm to get trapped in a local optimums or globally converges but in a slow approach. So, number of generations or the number of analyzes and then find the global optimum time will increase. In this work, this problem has been resolved with a new approach offered in the selection of individuals to combine together and produce a new generation, then the proposed algorithm will be able to find the global optimum more quickly.

2. THE GENETIC ALGORITHMS

The basic ideas of genetic algorithms proposed by John Holland and his colleagues at the University of Michigan in the United States in 1962 [26].

The performance of the genetic algorithm is based on a selected random individuals from a population. Everyone has the potential to answer the problem and it is generally expressed as a fixed-length string of binary numbers. This string is very similar to the natural chromosome. Then repeat the process with the enforcement of specific operators and population will modified to the optimal solution.

In each iteration, every string is decoded (real values are determined corresponding to each string) and the objective function will be obtained. Based on the obtained objective function values, each string is assigned a fitness value. This fitness value will be determined the probability of selection for each string. After that, a set of strings are selected, after of which are replaced with the initial population strings by employing the genetic operators new strings in order to keep the number of strings population in each computational iterations fixed.

Random mechanisms that act on the selection and remove the strings so that the strings are fitter and more likely to mix and produce new strings and in the replacement process are more resistant than other strings. In this way, population competed in a competition based on objective function in different generations and the average objective function value in the strings population is close to optimum value. For this reason, in each calculation, the three operators including operator selection and two genetic operators; crossover and mutation are affected on strings and cause new parts of the search space to be investigated. This is done in two steps. In the first step a mating pool was made using selection operator and in the next step crossover and mutation operators are applied on the mating pool to produce the next generation [27].

It is worth to note here that one of the most significant features of genetic algorithm is that its limitless ability of handling any size of optimization problems including linear or non-linear and continuous or discrete ones. This vast generality is however not possible with gradient-based mathematical algorithms, a reason for which a major tendency towards heuristic procedures is more sound.

3. THE PROPOSED GENETIC ALGORITHMS

3.1 Constraint-handling

An optimization problem using GAs can be generally expressed as

$$\text{Minimize/Maximize } f(\boldsymbol{X}), \quad x = (X_1, X_2, \ldots, X_n) \in R^n \tag{1}$$

Under constrain define as

$$\begin{aligned} g_i(\boldsymbol{x}) \le 0, \quad & \text{i=1,…,K,} \\ h_j(\boldsymbol{x}) = 0, \quad & \text{j=1,…,P} \end{aligned} \tag{2}$$

For structural design optimization, x is an N-dimensional vector called the design vector, representing design variables of N structural components to be optimized, and $f(\boldsymbol{x})$ is the objective function. Also, $g_i(\boldsymbol{x})$ and $h_i(\boldsymbol{x})$ are inequality and quality constrains, respectively. They represent constrains, such as stress and displacement limits to be satisfied by the optimum design.

In GAs, constrains are usually handled using the concept of penalty function as follows:

$$\text{Minimize} \quad \hat{F}_j = F_j.(1 + P_j) \tag{3}$$

$$\text{Maximize} \quad \hat{F}_j = F_j.\left(1 - P_j\right) \tag{4}$$

Where $\hat{F}_j$ represents an augmented fitness function after the penalization. Here, P_j is a penalty function whose value is greater than zero for infeasible search space and zero for feasible search space. In this paper the Squared Normalized Degree of Constraints Violation (SNDCV) are used.

$$P_j = \left(\sum_{j=1}^{K} G_i(x) + \sum_{j=1}^{P} H_i(x) \right)^2 \tag{5}$$

Here, $G_i(x)$ and $H_i(x)$ represent the degrees of inequality and equality constraint violations, respectively.

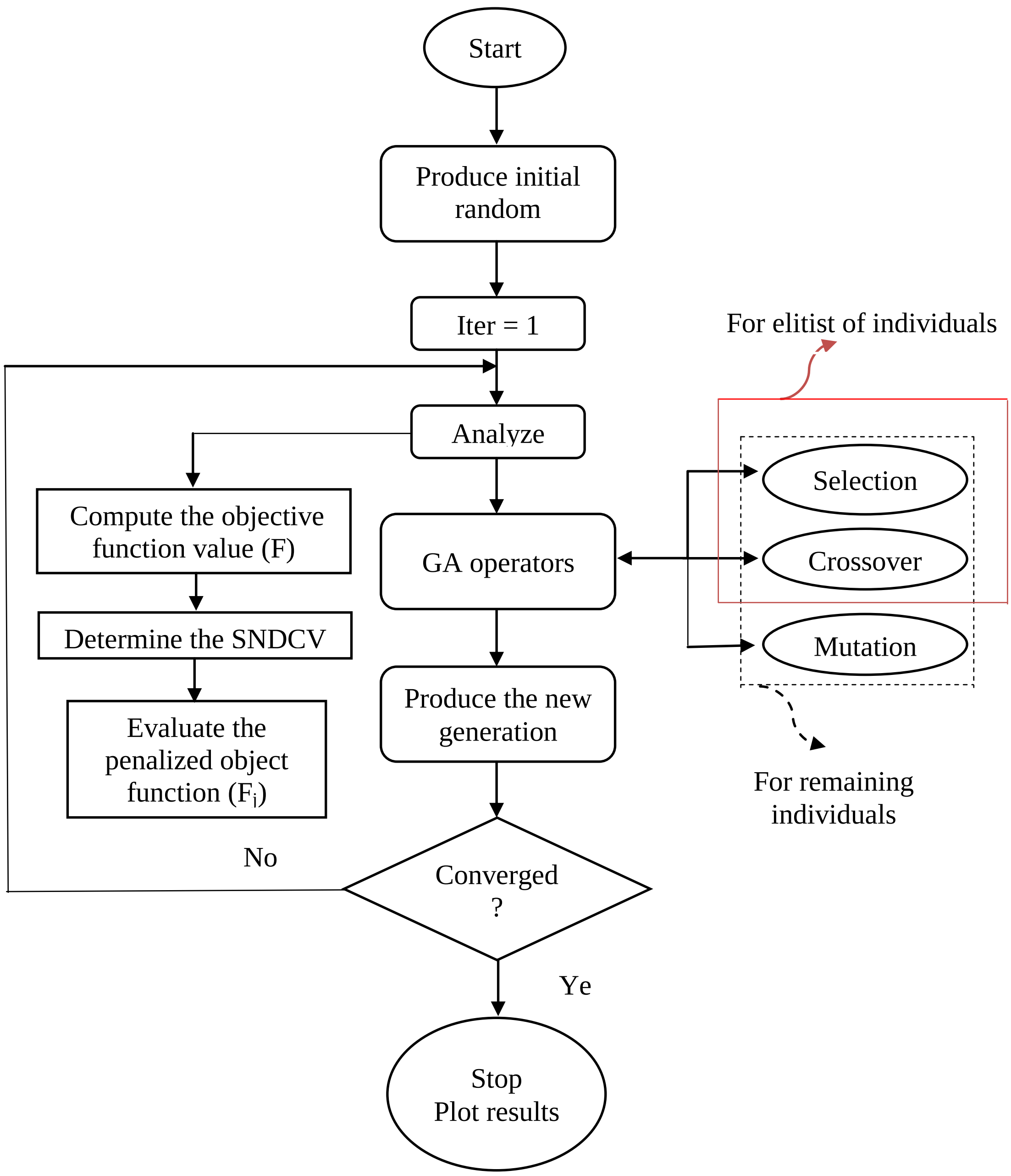


Figure 1. Flowchart of the proposed GA method

As stated earlier, the genetic algorithm benefits from various unique operators in its routine. Clearly in GA the role of selection operator is more highlighted than the crossover and mutation operators. In fact, if the best individual is defined based only on selection, it may assist fast convergence but it does not guarantee a global achievement for all cases and it may lead to premature convergence. In contrast, if a slight importance is given to the best individual in the selection process, time required to achieve the optimal solution will be very long.

Several well-known techniques including roulette wheel selection, ranking selection, tournament selection, probabilistic tournament selection and etc., are used for the selection in GA-based methods [27]. In this study, a new strategy is provided for the selection operator. For this purpose an individual who has more fitness than the other individuals of its current generation is combined with a certain percent of the elitist individuals of the same generation. These new individuals often will have superior features. In this study, 10 percent of the population of the next generation are produced with this method and the rest of the individuals (the remaining 90%) are combined by utilizing the tournament selection technique in order to prevent premature convergence. As we will see later, the proposed method produces excellent convergence in the benchmark problems considered. Also, the concept of "rebirthing" [7] is used to improve the results.

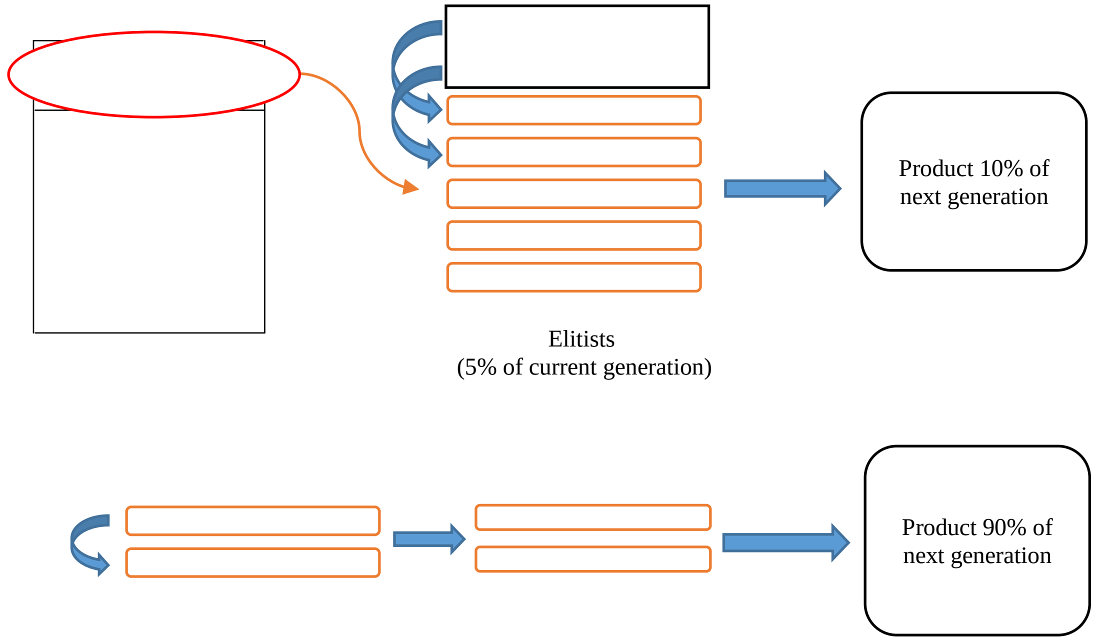


Figure 2. Representation of the proposed selection operator

4. NUMERICAL EXAMPLES

In this section, proposed algorithm is tested with various examples. Selected examples have been optimized by other researchers and the results obtained in this study were compared with those. The Table 1 contains the proposed GA properties for each example studied. It is to be noted that in all results tables, *NFEs* denotes the Number of Function Evaluations.

Table 1: The proposed GA properties

Example	Population size		Mutation probability	Crossover type	Maximum number of generations
Bumpy function	n=2	30	1	Single point	35
	n=20	250	1	Single point	400
	n=50	350	1	Single point	400
10-bar planar truss	Case 1	100	1	Single point	120
	Case 2	100	1	Single point	140
200-bar planar truss	1000		1	Single point	100

4.1 Bumpy function

This is an unconstrained optimization problem that introduced by keane [28]. The mathematical formulation of this problem is given below.

$$\text{Maximize:} \quad \frac{abs\left(\sum_{i=1}^{n}\cos^4(x_i) - 2\prod_{i=1}^{n}\cos^2(x_i)\right)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \tag{6}$$

$$\text{For}$$
$$0 < x_i < 10, \qquad i = 1, 2, 3, \ldots, n$$
$$\text{Subject to}$$
$$\prod_{i=1}^{n} x_i > 0.75 \quad and \quad \sum_{i=1}^{n} x_i < \frac{15\,n}{2} \tag{7}$$

Fig. 3 depicts the objective function and the boundaries of the constraint functions. Within the feasible region, there is one constrained global maximum on the boundary of the first constraint and many other local maxima.

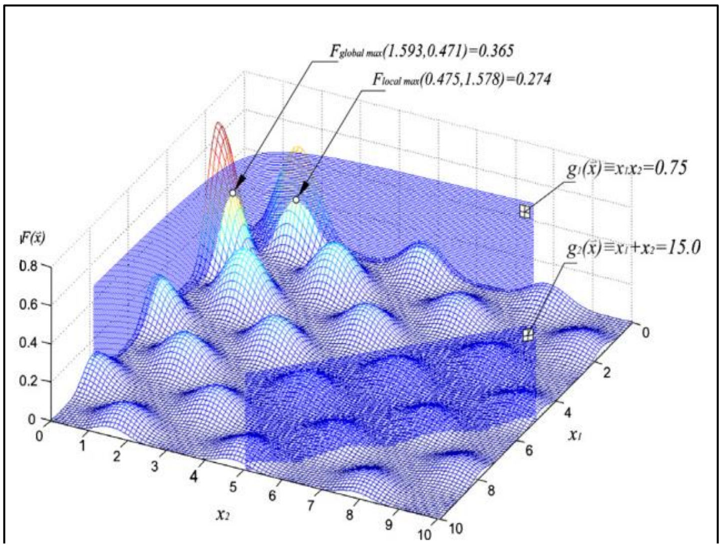


Figure 3. The constrained Bumpy problem (*n=2*) [29]

Table 2: Global solutions of Bumpy function for *n=2*

Variables	Exact Solution [28]	Chen and Cheng [30]	Lee [31]	DPF [31]	APF [31]	This study
X_1	1.593	1.601	1.639	1.650	1.563	1.6233
X_2	0.471	0.468	0.459	0.456	0.480	0.4623
Obj.	0.365	0.365	0.362	0.361	0.363	0.3642
NFEs	N.A.	1900	900	2500	2500	840

Table 2 gives the solutions obtained by various methods. The solution performed based on the proposed method is the second best one and the result is very close to exact solution. Besides number of analyses required was found the least among all.

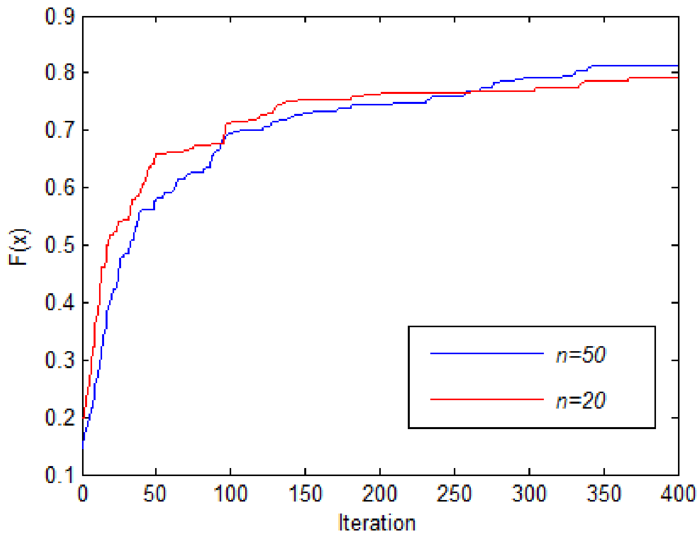


Figure 4. Convergence history of Bumpy function for *n=20 & n=50*

Fig. 4 illustrates the comparison of the convergence rates of two cases with n=20 and n=50. Also, in Table 3 the results obtained in this study were compared with the results of Keane for twenty- dimensional [32] and fifty- dimensional [28] bumpy function. Results show proposed GA can find global optimum and decrease the number of function evaluations rather than results presented by Keane.

Table 3: Global solutions of Bumpy function for *n=20 & n=50*

No. Variables	Keane [32,28]		This study	
	Obj.	*NFEs*	Obj.	*NFEs*
20	~ 0.75 *	100,000	0.7917	92,500
50	0.7905	140,000	0.8120	122,500

*Note: The number is derived from the figure

4.2 The 10-bar planar truss structure

The 10-bar planar truss structure, shown in Fig. 5. This famous optimization problem is described by Sunar and Belegundu [33]. In this problem the cross-sectional area for each of the 10 members in the structure are being optimized towards the minimization of total weight. The cross-sectional area varies between 0.1 in^2 and 35 in^2. Constraints are specified in terms of stress and displacement of the truss members. The allowable stress for each member is ±25 ksi for both tension and compression, and the allowable displacement on the nodes is ±2 in, in the *x* and *y* directions.

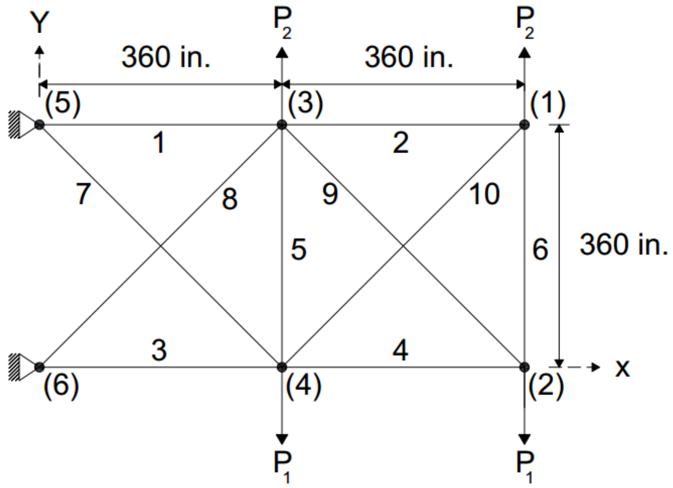


Figure 5. A 10-bar planar truss structure

Table 4: Optimization results for a ten-bar truss (Case 1).

Variables		Optimal cross section area (in^2)						
No	Design name	Perez and Behdinan [13]	Lee and Geem [34]	Li et al. [35]	Gellatly and Berke [36]	Dizangian and Ghasemi [37]	This study: Propose method (1)	This study: Proposed method (2)
1	A_1	33.500	30.150	30.704	31.350	31.1650	30.55	31.100
2	A_2	0.100	0.102	0.100	0.100	0.1000	0.100	0.100
3	A_3	22.766	22.710	23.167	20.030	23.1000	25.4125	21.100
4	A_4	14.417	15.270	15.183	15.600	14.7230	15.55	13.850
5	A_5	0.100	0.102	0.100	0.140	0.1000	0.100	0.100
6	A_6	0.100	0.544	0.551	0.240	0.4139	0.100	0.850
7	A_7	7.534	7.541	7.460	8.350	7.5712	8.3917	7.850
8	A_8	20.467	21.560	20.978	22.210	21.1630	20.0375	21.850
9	A_9	20.392	21.450	21.508	22.060	21.4230	20.6417	21.100
10	A_{10}	0.100	0.100	0.100	0.100	0.1000	0.100	0.100
Weight (lb)	5024.21		5057.88	5060.9	5112.00	5064.40	5088.62	5007.00
Constraint violation	23.95×10^{-3}		0.907×10^{-3}	0.907×10^{-3}	None	None	None	29.2×10^{-3}
NFEs	NA		400,000	75,000	NA	650	3,000	3,600

Note: 1 in^2 = 6.452 cm^2; 1lb = 4.45 N.

The material density is $0.1\ ^{\mathrm{lb}}/_{\mathrm{in}^3}$ and the modulus of elasticity is 10,000 ksi. Two cases are considered: Case 1, $P_1 = 100\ kips$ and $P_2 = 0$; and Case 2, $P_1 = 150\ kips$ and $P_2 = 50\ kips$.

The comparison of best solution with previous methods is given in Tables 4 and 5. These Tables also contain the optimum results from the two proposed optimization algorithm with the results reported by other researchers.

Table 5. Optimization results for a ten-bar truss (Case 2).

Variables		Optimal cross section area (in^2)					
No	Design name	Sonmez [25]	Lee and Geem [34]	Li et al. [35]	Dizangian and Ghasemi [37]	This study: Proposed method (1)	This study: Proposed method (2)
1	A_1	23.4692	23.25	23.353	25.0000	27.10	22.8453
2	A_2	0.1005	0.102	0.100	0.1000	0.10	0.2368
3	A_3	25.2393	25.73	25.502	25.0000	25.60	23.0378
4	A_4	14.3540	14.51	14.250	14.4320	12.10	16.2387
5	A_5	0.1001	0.100	0.100	0.1000	0.10	0.1061
6	A_6	1.9701	1.977	1.972	1.9876	2.10	2.1051
7	A_7	12.4128	12.21	12.363	12.4250	12.10	12.2040
8	A_8	12.8925	12.21	12.894	12.3590	11.10	11.8261
9	A_9	20.3343	20.36	20.356	19.9820	22.60	20.8582
10	A_{10}	0.1000	0.100	0.101	0.1000	0.10	0.1213
Weight (lb)		4677.077	4668.81	4677.29	4682.476	4752.44	4616.03
Constraint violation		None	3.561×10^{-3}	0.025×10^{-3}	None	None	25.7×10^{-3}
NFEs		500,000	400,000	75,000	1000	3,600	5,900

Note: 1 in^2 = 6.452 cm^2; 1lb = 4.45 N.

The Figs. 6 and 7 shown the convergence history plots of these two proposed optimization methods for the two loading cases. Based on these convergence plots one could claim that the proposed methods lead to fast convergence rather than the classical GA.

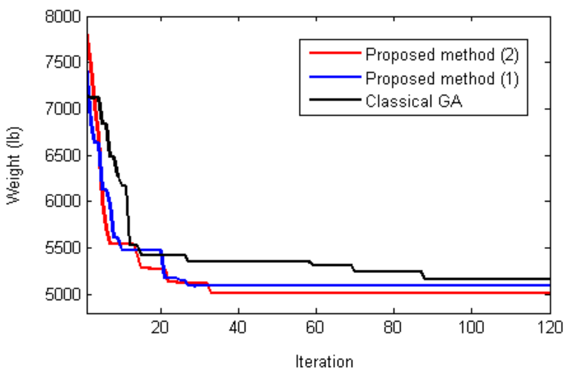


Figure 6. Comparison of the convergence rates between the two algorithms for a 10-bar planar truss structure (Case 1).

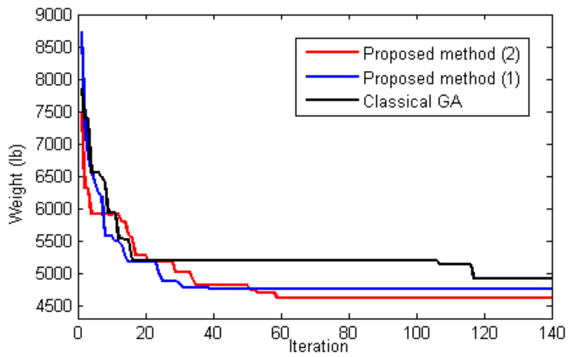


Figure 7. Comparison of the convergence rates between the two algorithms for a 10-bar planar truss structure (Case 2).

4.3 The 200-bar planar truss structure

A 200-bar plane truss, shown in Fig. 8. The members were linked together into twenty-nine groups. The material density and modulus of elasticity are 0.283 lb/in^3 and 30,000 ksi, respectively. The members are subjected to stress limitations of ±10 ksi. There are no displacement limit but the minimum cross-section area was not allowed to be less than 0.1 in^2. There are three independent loading conditions: (1) 1.0 kips acting in the positive x-direction at nodes 1, 6, 15, 20, 29, 34, 43, 48, 57, 62 and 71; (2) 10 kips acting in the negative y-direction at nodes 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 15, 16, 17, 18, 19, 20, 22, 24, 26, 28, 29, . . ., 72, 73, 74 and 75; (3) conditions 1 and 2 acting together.

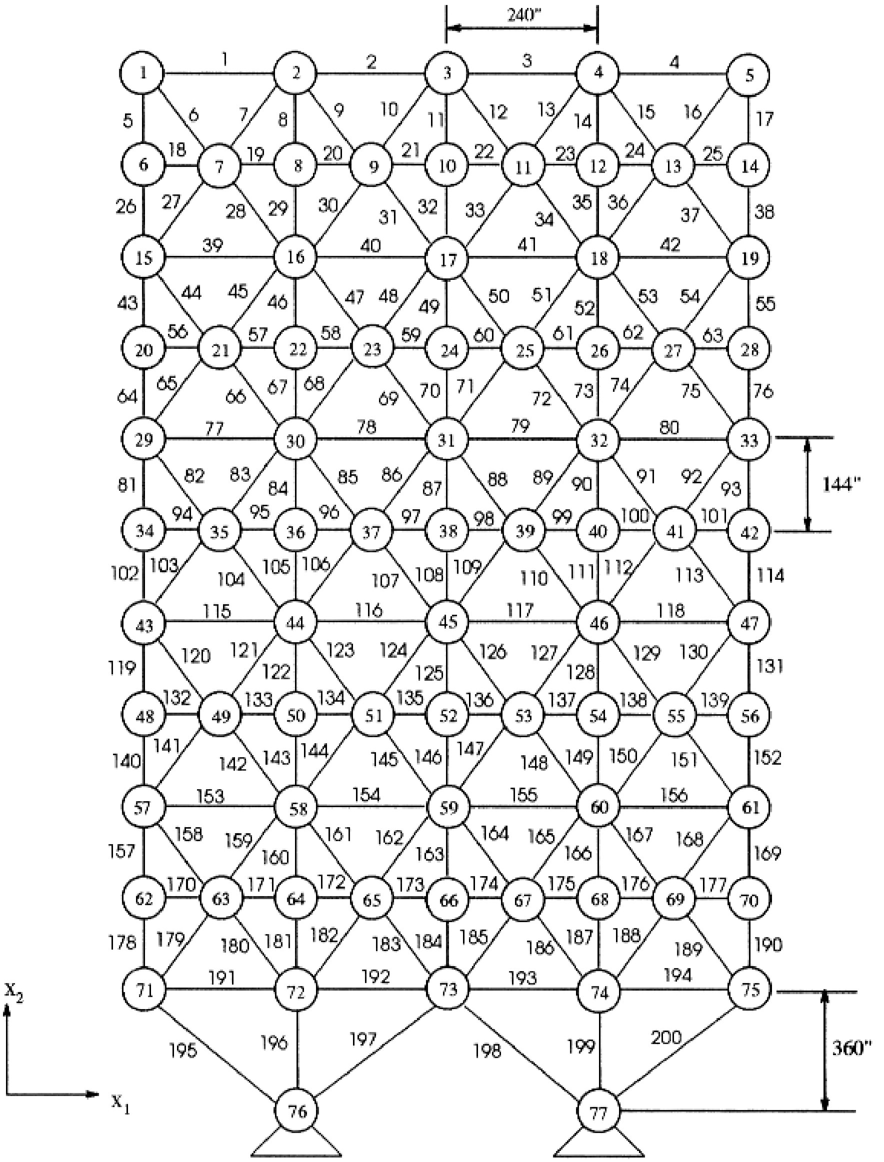


Figure 8. The 200- bar planar truss

It can be observed from the Table 6 that the computational performance of the proposed GA is reasonably appropriate with respect to that of the other researchers. Fig. 8 also demonstrates that the

Table 6: Optimization results for a two-hundred-bar truss structure

Variables		Optimal cross section area (in^2)					
Design group	Members	Sonmez [25]	Lee and Geem [34]	Lamberti [38]	Kaveh et al. [39]	Dizangian and Ghasemi [40]	This study
1	1, 2, 3, 4	0.1039	0.1253	0.1468	0.1480	0.1335	0.1222
2	5, 8, 11, 14, 17	0.9463	1.0157	0.9400	0.9460	1.0365	1.0374
3	19, 20, 21, 22, 23, 24	0.1037	0.1069	0.1000	0.1010	0.1000	0.1000
4	18, 25, 56, 63, 94, 101, 132, 139, 170, 177	0.1126	0.1096	0.1000	0.1010	0.1001	0.1036
5	26, 29, 32, 35, 38	1.9520	1.9369	1.9400	1.9461	1.9550	2.0077
6	6, 7, 9, 10, 12, 13, 15, 16, 27, 28, 30, 31, 33, 34, 36, 37	0.2930	0.2686	0.2962	0.2979	0.2830	0.2746
7	39, 40, 41, 42	0.1064	0.1042	0.1000	0.1010	0.1017	0.1000
8	43, 46, 49, 52, 55	3.1249	2.9731	3.1042	3.1072	3.1021	3.2418
9	57, 58, 59, 60, 61, 62	0.1077	0.1309	0.1000	0.1010	0.1087	0.1000
10	64, 67, 70, 73, 76	4.1286	4.1831	4.1042	4.1062	4.0886	4.1219
11	44, 45, 47, 48, 50, 51, 53, 54, 65, 66, 68, 69, 71, 72, 74, 75	0.4250	0.3967	0.4034	0.4049	0.4084	0.3899
12	77, 78, 79, 80	0.1046	0.4416	0.1912	0.1944	0.1782	0.1000
13	81, 84, 87, 90, 93	5.4803	5.1873	5.4284	5.4299	5.3992	5.3823
14	95, 96, 97, 98, 99, 100	0.1060	0.1912	0.1000	0.1010	0.1419	0.1724
15	102, 105, 108, 111, 114	6.4853	6.2410	6.4284	6.4299	6.4417	6.3619
16	82, 83, 85, 86, 88, 89, 91 92, 103, 104, 106, 107, 109, 110,112, 113	0.5600	0.6994	0.5734	0.5755	0.5838	0.5421
17	115, 116, 117,118	0.1825	0.1158	0.1327	0.1349	0.1171	0.1000
18	119, 122, 125, 128, 131	8.0445	7.7643	7.9717	7.9747	7.9493	7.9211
19	133, 134, 135, 136, 137, 138	0.1026	0.1000	0.1000	0.1010	0.1863	0.1000
20	140, 143, 146, 149, 152	9.0334	8.8279	8.9717	8.9747	8.9506	8.8643
21	120, 121, 123,124, 129, 127, 129, 130, 141, 142, 144, 145,147, 148, 150, 151	0.7844	0.6986	0.7049	0.70648	0.7322	0.6449
22	153, 154, 155,156	0.7506	1.5563	0.4196	0.4225	0.1327	0.1000
23	157, 160, 163, 166, 169	11.3057	10.9806	10.8636	10.8685	10.777	10.6280
24	171, 172, 173, 174, 175, 176	0.2208	0.1317	0.1000	0.1010	0.1084	0.1000
25	178, 181, 184, 187, 190	12.2730	12.1492	11.8606	11.8684	11.8057	11.5131
26	158, 159, 161, 162, 164, 165, 167, 168, 179, 180, 182,183, 185, 186, 188, 189	1.4055	1.6373	1.0339	1.03599	0.8506	0.7289
27	191, 192, 193, 194	5.1600	5.0032	6.6818	6.6859	7.2174	9.0623
28	195, 197, 198, 200	9.9930	9.3545	10.8113	10.8111	11.4243	11.6744
29	196, 199	14.70144	15.0919	13.8404	13.8464	13.5966	13.0687
	Weight (lb)	25533.79	25447.1	25447.528	25467.9	25530	25772.16
	Constraint violation	0.54234	0.40023	0.00310	None	None	None
	NFEs	1,450,000	48,000	NA	31,700	6,600	41,000

Note: 1 in^2 = 6.452 cm^2; 1lb = 4.45 N.

Performance of the proposed GA is better than the classical one.

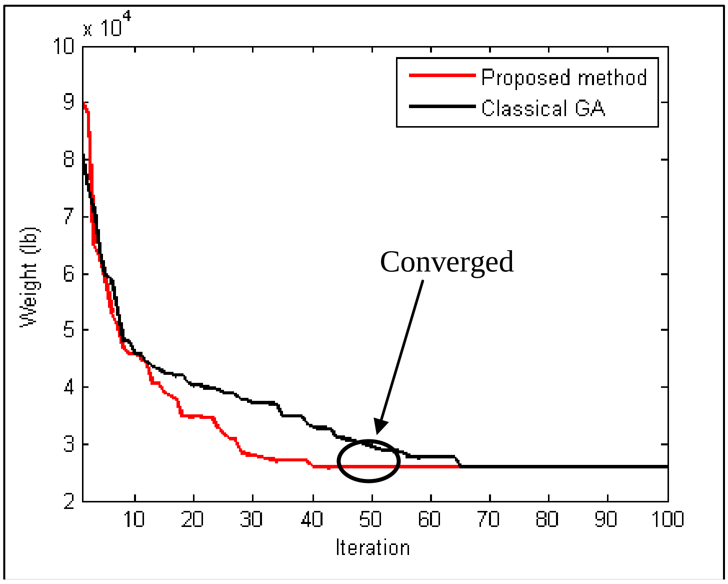


Figure 9. Comparison of the convergence rates between the two algorithms for the 200-bar planar truss structure

5. CONCLUSION

In this paper a new version of GA based on the modification of the selection operator was introduced. To do this, selection operator was defined to combine with a percent of the elitists individuals to product the next generation. In order to prove the adequacy and accuracy of the proposed GA, we first review a bumpy function. After that two challenging truss problems are studied. Results indicate that this version of GA could help for the fast convergence in all cases and a little improvement in the optimum solution for some problems.

REFERENCES

1. Dorn WS. Automatic design of optimal structures, *Journal de Mecanique* 1964; **3**: 25-52.
2. Vanderplaats GN, Moses F. Automated design of trusses for optimum geometry, *J Struct Div, ASCE* 1972; **98**(3): 671-90.
3. Topping B. Shape optimization of skeletal structures: a review, *J Struct Eng, ASCE* 1983; **109**(8): 1933-51.
4. Ringertz UT. On topology optimization of trusses, *Eng Optim* 1985; **9**(3): 209-218.
5. Haug E, Arora J. *Introduction to Optimal Design*, McGrawHill, NewYork, 1989.
6. Krish U. Optimal topologies of truss structures, *Comput Meth Appl Mech Eng* 1989; **72**: 15-28.

7. Ghasemi M, Hinton E,Wood R. Optimization of trusses using genetic algorithms for discrete and continuous variables, *Eng Comput* 1999; **16**(3): 272-303.
8. Hinton E, Bulman S, Sienz J, Ghasemi M. *Fully Integrated Design Optimization*, Oxbridge Press, Structural Optimisation (Australia), 1998; pp. 3-29.
9. Kaveh A. *Advances in Metaheuristic Algorithms for Optimal Design of Structures*, Springer, 2014.
10. Toğan V,Daloğlu AT. An improved genetic algorithm with initial population strategy and self-adaptive member grouping, *Comput Struct* 2008; **86**(11): 1204-18.
11. Deb K,Gulati S. Design of truss-structures for minimum weight using genetic algorithms, *Finite Elem Anal Des* 2001; **37**(5): 447-65.
12. Goldberg DE, Samtani MP. Engineering optimization via genetic algorithm, *Electronic Comput, ASCE* 1986; 471-482.
13. Perez R, Behdinan K. Particle swarm approach for structural design optimization, *Comput Struct* 2007; **85**(19): 1579-88.
14. Li L, Huang Z, Liu F, Wu Q. A heuristic particle swarm optimizer for optimization of pin connected structures, *Comput Struct* 2007; **85**(7): 340-9.
15. Imai K, Schmit LA. Configuration optimization of trusses, *J Struct Div, ASCE* 1981; **107**(5): 745-56.
16. Kaveh A, Talatahari S. An enhanced charged system search for configuration optimization using the concept of fields of forces, *Struct Multidiscip Optim* 2011; **43**(3): 339-51.
17. Kaveh A, Hassani B, Shojaee S,Tavakkoli S. Structural topology optimization using ant colony methodology, *Eng Struct* 2008; **30**(9): 2559-65.
18. Hinton E, Sienz J, Bulman S, Lee S,Ghasemi M. *Integrating Structural Topology, Shape and Sizing Optimization Methods*, Computational Mechanics, New Trends and Applications, Barcelona, Spain, 1998.
19. Mallipeddi R, Das S, Suganthan PN. *Ensemble of Constraint Handling Techniques for Single Objective Constrained Optimization, Evolutionary Constrained Optimization*, Springer, 2015, pp. 231-248.
20. Templeman AB, Yates DF. A linear programming approach to the discrete optimum design of trusses. In: Eschenauer, H.; Olhoff, N. (eds.) Optimization Methods in Structural Design, pp. 133–139. Mannheim: BI Wissenschaftsverlag, 1983.
21. Ming-Zhu D. An improved Templeman's algorithm for the optimum design of trusses with discrete member sizes, *Eng Optim* 1986; **9**(4): 303-12.
22. John K, Ramakrishnan C,Sharma K. Minimum weight design of trusses using improved move limit method of sequential linear programming, *Comput Struct* 1987; **27**(5): 583-91.
23. Rajeev S, Krishnamoorthy C. Discrete optimization of structures using genetic algorithms, *J Struct Eng* 1992; **118**(5);May 1992.
24. Camp CV, Bichon BJ. Design of space trusses using ant colony optimization, *J Struct Eng* 2004; **130**(5): 741-51.
25. Sonmez M. Artificial bee colony algorithm for optimization of truss structures, *Appl Soft Comput* 2011; **11**(2): 2406-18.
26. Holland JH. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, MIT press, 1992.

27. Sivanandam S, Deepa S. *Introduction to Genetic Algorithms*, Springer Science & Business Media, 2007.
28. Keane A. Experiences with optimizers in structural design, *Proceedings of the Conference on Adaptive Computing in Engineering Design and Control* 1994; pp. 14-27.
29. Chen T, Huang J. Application of data mining in a global optimization algorithm, *Adv Eng Softw* 2013; **66**: 24-33.
30. Chen TY, Cheng YL. Global optimization using hybrid approach, *WSEAS Trans Math* 2008; **7**(5): 254-62.
31. Lee C. Reproducing kernel approximation method for structural optimization using genetic algorithms, PhD dissertation, National Taiwan University, 2006.
32. Ong YS, Keane AJ. Meta-Lamarckian learning in memetic algorithms, *IEEE Trans Evol Comput* 2004; **8**(2): 99-110.
33. Sunar M, Belegundu A. Trust region methods for structural optimization using exact second order sensitivity, *Int J Numer Meth Eng* 1991; **32**(2): 275-93.
34. Lee KS, Geem ZW. A new structural optimization method based on the harmony search algorithm, *Comput Struct* 2004; **82**(9): 781-98.
35. Li L, Huang Z, Liu F, Wu Q. A heuristic particle swarm optimizer for optimization of pin connected structures, *Comput Struct* 2007; **85**(7): 340-9.
36. Gellatly RA, Berke L. *Optimal Structural Design*, DTIC Document, 1971.
37. Dizangian B, Ghasemi M. A fast marginal feasibility search method in size optimization of truss structures, *Asian J Civil Eng, BHRC* 2015; **16**(5): 567-85.
38. Lamberti L. An efficient simulated annealing algorithm for design optimization of truss structures, *Comput Struct* 2008; **86**(19): 1936-53.
39. Kaveh A, Sheikholeslami R, Talatahari S,Keshvari-Ilkhichi M. Chaotic swarming of particles: a new method for size optimization of truss structures, *Adv Eng Softw* 2014; **67**: 136-47.
40. Dizangian B, Ghasemi M. Ranked-based sensitivity analysis for size optimization of structures, *J Mech Des* 2015; Online, DOI: 10.1115/1.4031295.