Available online at www.sciencedirect.com
SCIENCE DIRECT®

Computers and Structures 82 (2004) 781–798

Computers & Structures

www.elsevier.com/locate/compstruc

A new structural optimization method based on the harmony search algorithm

Kang Seok Lee [a,*], Zong Woo Geem [b]

[a] Materials and Construction Research Division, National Institute of Standards and Technology, 100 Bureau Drive, Gaithersburg, MD 20899-8611, USA

[b] Department of Civil and Environmental Engineering, University of Maryland, College Park, MD 20742, USA

Received 14 April 2003; accepted 6 January 2004

Abstract

Most structural optimization methods are based on mathematical algorithms that require substantial gradient information. The selection of the starting values is also important to ensure that the algorithm converges to the global optimum. This paper describes a new structural optimization method based on the harmony search (HS) meta-heuristic algorithm, which was conceptualized using the musical process of searching for a perfect state of harmony. The HS algorithm does not require initial values and uses a random search instead of a gradient search, so derivative information is unnecessary. Various truss examples with fixed geometries are presented to demonstrate the effectiveness and robustness of the new method. The results indicate that the new technique is a powerful search and optimization method for solving structural engineering problems compared to conventional mathematical methods or genetic algorithm-based approaches.

Keywords: Structural optimization; Harmony search; Meta-heuristic algorithm; Size optimization; Continuous variables; Truss structures

1. Introduction

Structural design optimization is a critical and challenging activity that has received considerable attention in the last two decades. Designers are able to produce better designs while saving time and money through optimization. Traditionally, various mathematical methods such as linear, nonlinear, and dynamic programming have been developed to solve engineering optimization problems. However, these methods represent a limited approach, and no single method is completely efficient and robust for all types of optimization problems.

Some techniques, including the penalty-function, augmented Lagrangian, and conjugate gradient methods, search for a local optimum by moving in a direction related to the local gradient. Other methods apply the first- and second-order necessary conditions to seek a local minimum by solving a set of nonlinear equations. These methods become inefficient when searching for the optimum design of large structures due to the large amount of gradient calculations that are required. Usually, these techniques seek a solution in the neighborhood of the starting point, similar to local hill climbing. If there is more than one local optimum in the problem, the result will depend on the selection of the initial point, and the solution will not necessarily correspond to the global optimum. Furthermore, when the objective function and constraints have multiple or

* Corresponding author. Tel.: +1-301-975-2881; fax: +1-301-869-6275.

E-mail addresses: kslee@nist.gov (K.S. Lee), zwgeem@yahoo.com (Z.W. Geem).

0045-7949/$ - see front matter
doi:10.1016/j.compstruc.2004.01.002

sharp peaks, the gradient search becomes difficult and unstable [1].

The computational drawbacks of mathematical methods (i.e., complex derivatives, sensitivity to initial values, and the large amount of enumeration memory required) have forced researchers to rely on meta-heuristic algorithms based on simulations to solve optimization problems. The common factor in meta-heuristic algorithms is that they combine rules and randomness to imitate natural phenomena. These phenomena include the biological evolutionary process (e.g., the evolutionary algorithm proposed by Fogel et al. [2], De Jong [3], and Koza [4] and the genetic algorithm proposed by Holland [5] and Goldberg [6]), animal behavior (e.g., tabu search proposed by Glover [7] and the ant algorithm proposed by Dorigo et al. [8]), and the physical annealing process (e.g., simulated annealing proposed by Kirkpatrick et al. [9]). In the last decade, these meta-heuristic algorithms, especially the genetic algorithm have been broadly applied to solve various structural optimization problems, and have occasionally overcome several deficiencies of conventional mathematical methods. These include researches by Adeli and Cheng [1], Rajeev and Krishnamoorthy [10,11], Koumousis and Georgious [12], Hajela and Lee [13], Adeli and Kumar [14], Wu and Chow [15,16], Soh and Yang [17], Camp et al. [18], Shrestha and Ghaboussi [19], Erbatur et al. [20], and Sarma and Adeli [21]). To solve complicated optimization problems, however, new heuristic and more powerful algorithms based on analogies with natural or artificial phenomena remain to be explored.

Recently, Geem et al. [22] developed a new harmony search (HS) meta-heuristic algorithm that was conceptualized using the musical process of searching for a perfect state of harmony. Compared to mathematical optimization algorithms, the HS algorithm imposes fewer mathematical requirements and does not require initial values for the decision variables. Furthermore, the HS algorithm uses a random search, which is based on the harmony memory considering rate and the pitch adjusting rate (these are defined in the following section), instead of a gradient search, so derivative information is unnecessary. Although the HS algorithm is a comparatively simple method, it has been successfully applied to various optimization problems including the traveling salesperson problem, the layout of pipe networks, pipe capacity design in water supply networks, hydrologic model parameter calibrations, cofferdam drainage pipe design, and optimal school bus routings [22–26].

This paper proposes a new structural optimization method based on the HS algorithm. Various truss examples, including large-scale trusses under multiple loading conditions with continuous sizing variables (fixed geometry), are presented to demonstrate the effectiveness and robustness of the new method. Although the method proposed in this paper is applied to truss structures, it is a general optimization procedure that can be easily adapted to other types of structures, such as frame structures, plates, and shells.

2. Harmony search algorithm

The new HS meta-heuristic algorithm was derived by adopting the idea that existing meta-heuristic algorithms are found in the paradigm of natural phenomena. The algorithm was based on natural musical performance processes that occur when a musician searches for a better state of harmony, such as during jazz improvization [22]. Jazz improvization seeks to find musically pleasing harmony (a perfect state) as determined by an aesthetic standard, just as the optimization process seeks to find a global solution (a perfect state) as determined by an objective function. The pitch of each musical instrument determines the aesthetic quality, just as the objective function value is determined by the set of values assigned to each decision variable.

Fig. 1 shows the optimization procedure of the HS algorithm, which consists of Steps 1 through 5:

Step 1: Initialize the optimization problem and algorithm parameters. First, the optimization problem is specified as follows:

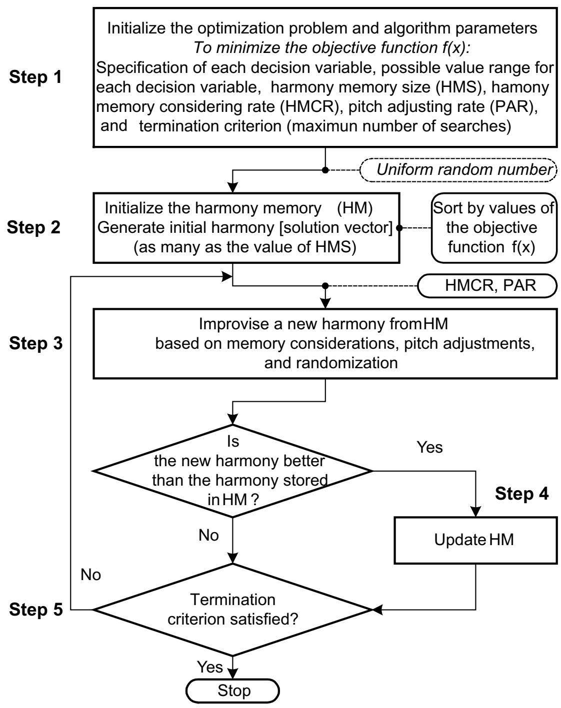


Fig. 1. Harmony search algorithm optimization procedure.

$$\begin{array}{ll} \text{Minimize} & f(\boldsymbol{x}) \\ \text{Subject to} & x_i \in \boldsymbol{X}_i, \quad i = 1, 2, \ldots, N \end{array} \tag{1}$$

where $f(\boldsymbol{x})$ = an objective function; $\boldsymbol{x}$ = the set of each decision variable x_i; $\boldsymbol{X}_i$ = the set of possible range of values for each decision variable, that is, $\boldsymbol{X}_i = \{x_i(1), x_i(2), \ldots, x_i(K)\}$ for discrete decision variables $(x_i(1) < x_i(2) < \cdots < x_i(K))$ or ${}_{L}x_i \leqslant \boldsymbol{X}_i \leqslant {}_{U}x_i$ for continuous decision variables; N = the number of decision variables; and K = the number of possible values for the discrete variables. The HS algorithm parameters that are required to solve the optimization problem (i.e., Eq. (1)) are also specified in this step: harmony memory size (number of solution vectors, HMS), harmony memory considering rate (HMCR), pitch adjusting rate (PAR), and termination criterion (maximum number of searches). Here, HMCR and PAR are parameters that are used to improve the solution vector. Both are defined in Step 3.

Step 2: Initialize the harmony memory (HM). In this step, the "harmony memory" (HM) matrix shown in Eq. (2) is filled with as many randomly generated solution vectors as the size of the HM (i.e., HMS) and sorted by the values of the objective function, $f(\boldsymbol{x})$.

$$\mathbf{HM} = \begin{bmatrix} \boldsymbol{x}^1 \\ \boldsymbol{x}^2 \\ \vdots \\ \boldsymbol{x}^{\mathrm{HMS}} \end{bmatrix} \tag{2}$$

Step 3: Improvise a new harmony from the HM. A new harmony vector, $\boldsymbol{x}' = (x'_1, x'_2, \ldots, x'_N)$ is generated from the HM based on memory considerations, pitch adjustments, and randomization. For instance, the value of the first decision variable (x'_1) for the new vector can be chosen from any value in the specified HM range $(x_1^1 \sim x_1^{\mathrm{HMS}})$. Values of the other decision variables (x'_i) can be chosen in the same manner. Here, there is a possibility that the new value can be chosen using the HMCR parameter, which varies between 0 and 1 as follows:

$$x'_i \leftarrow \begin{cases} x'_i \in \{x_i^1, x_i^2, \ldots, x_i^{\mathrm{HMS}}\} & \text{w.p. HMCR} \\ x'_i \in \boldsymbol{X}_i & \text{w.p. (1-HMCR)} \end{cases} \tag{3}$$

The HMCR sets the rate of choosing one value from the historic values stored in the HM, and (1-HMCR) sets the rate of randomly choosing one value from the possible range of values. For example, a HMCR of 0.95 indicates that the HS algorithm will choose the decision variable value from historically stored values in the HM with a 95% probability and from the entire possible range with a 5% probability. A HMCR value of 1.0 is not recommended, because there is a chance that the solution will be improved by values not stored in the HM. This is similar to the reason why genetic algorithms use a mutation rate in the selection process. On the other hand, every component of the new harmony vector, $\boldsymbol{x}' = (x'_1, x'_2, \ldots, x'_N)$, is examined to determine whether it should be pitch-adjusted. This procedure uses the PAR parameter that sets the rate of adjustment for the pitch chosen from the HM as follows:

$$\text{Pitch adjusting decision for } x'_i \leftarrow \begin{cases} \text{Yes} & \text{w.p. PAR} \\ \text{No} & \text{w.p. (1-PAR)} \end{cases} \tag{4}$$

The pitch adjusting process is performed only after a value is chosen from the HM. The value (1-PAR) sets the rate of doing nothing. A PAR of 0.1 indicates that the algorithm will choose a neighboring value with 10%×HMCR probability. If the pitch adjustment decision for x'_i is Yes, and x'_i is assumed to be $x_i(k)$, i.e., the kth element in $\mathbf{X}_i$, the pitch-adjusted value of $x_i(k)$ is

$$\begin{array}{ll} x'_i \leftarrow x_i(k+m) & \text{for discrete decision variables} \\ x'_i \leftarrow x'_i + \alpha & \text{for continuous decision variables} \end{array} \tag{5}$$

where m = the neighboring index, $m \in \{\ldots, -2, -1, 1, 2, \ldots\}$; α = the value of $bw \times u(-1, 1)$; bw = an arbitrary distance bandwidth for the continuous variable; and $u(-1, 1)$ = a uniform distribution between −1 and 1. The HMCR and PAR parameters introduced in the harmony search help the algorithm find globally and locally improved solutions, respectively.

Step 4: Update the HM. In Step 4, if the new harmony vector is better than the worst harmony in the HM, judged in terms of the objective function value, the new harmony is included in the HM and the existing worst harmony is excluded from the HM. The HM is then sorted by the objective function value.

Step 5: Repeat Steps 3 and 4 until the termination criterion is satisfied. The computations are terminated when the termination criterion is satisfied. If not, Steps 3 and 4 are repeated.

To further understand the HS heuristic algorithm, consider the following mathematical function problem:

$$f(\boldsymbol{x}) = 4x_1^2 - 2.1x_1^4 + \tfrac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4 \tag{6}$$

The objective function is the Six-Hump Camelback function, as shown in Fig. 2, which is one of the standard test functions in optimization problems [27]. Due to the six local optima that are present in the function, two of which are global, the result from gradient-based algorithms may depend on the selection of an initial point. Thus, the obtained optimal solution may not necessarily be the global optima that are located at either $\boldsymbol{x}^* = (-0.08983, 0.7126)$ or $\boldsymbol{x}^* = (0.08983, -0.7126)$, each with a corresponding function value equal to $f^*(\boldsymbol{x}) = -1.0316285$. The HS algorithm described in this study finds the solution vector using a different method. When applying the HS algorithm to the Six-Hump

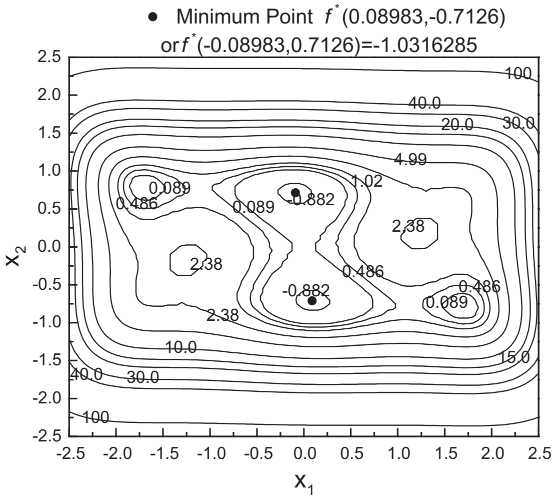


Fig. 2. Six-Hump Camelback function.

Camelback function, possible value bounds between −10.0 and 10.0 were used for the two design variables, x_1 and x_2, shown in Eq. (6). The total number of solution vectors, i.e., the HMS, was 10, and the HMCR and the PAR were 0.85 and 0.45, respectively (Step 1).

As shown in Table 1, the HM was initially structured with randomly generated solution vectors within the bounds prescribed for this example (i.e., −10.0–10.0); these were sorted according to the values of the objective function, Eq. (6) (Step 2). Next, a new harmony vector $x_i' = (3.183, 8.666)$ was improvised from possible values based on three rules: memory considerations with a 46.75% probability ($0.85\times0.55 = 0.4675$), pitch adjustments with a 38.25% probability ($0.85\times0.45 = 0.3825$), and randomization with a 15% probability ($1-0.85 = 0.15$), as described above in Step 3. As the objective function value of the new harmony (3.183, 8.666) was 22454.67, the new harmony was included in the HM and the worst harmony (−9.50, 3.333) was excluded from the HM, as shown in Table 1 (see Subsequent HM) (Step 4). The probability of finding the minimum vector, $\boldsymbol{x}^* = (0.08983, -0.7126)$, which is one of the global optima, increased with the number of searches (see HM after 50, 100, and 1,000 searches in Table 1). Finally, after 4870 searches, the HS algorithm improvised a near-optimal harmony vector, $\boldsymbol{x} = (0.08984, -0.71269)$, that had a function value of −1.0316285, as shown in Table 1. The HS algorithm also found the other optimal point (−0.08983, 0.7126) using a different random number seed.

The HS algorithm incorporates, by nature, the structure of the current meta-heuristic optimization algorithms. It preserves the history of past vectors (HM) similar to the tabu search, and is able to vary the

Table 1
Optimal results for the Six-Hump Camelback function using the HS algorithm

Rank	Initial HM			Subsequent HM			HM after 50 searches		
	x_1	x_2	$f(\boldsymbol{x})$	x_1	x_2	$f(\boldsymbol{x})$	x_1	x_2	$f(\boldsymbol{x})$
1	3.183	−0.400	169.95	3.183	−0.400	169.95	0.80558	−0.400	0.94270
2	−6.600	5.083	26274.83	**3.183**[a]	**8.666**[a]	**22454.67**[a]	0.80558	2.301	94.65
3	6.667	7.433	37334.24	−6.600	5.083	26274.83	0.80558	2.322	98.47
4	6.767	8.317	46694.70	6.667	7.433	37334.24	0.81676	2.419	117.35
5	−7.583	5.567	60352.77	6.767	8.317	46694.70	−0.88333	2.561	145.66
6	7.767	4.700	67662.40	−7.583	5.567	60352.77	−0.88333	2.589	152.54
7	8.250	2.750	95865.20	7.767	4.700	67662.40	3.074	−1.833	157.57
8	−8.300	8.533	120137.09	8.250	2.750	95865.20	3.183	−0.400	169.95
9	−9.017	−8.050	182180.00	−8.300	8.533	120137.09	3.183	−1.755	191.78
10	−9.500	3.333	228704.72	−9.017	−8.050	182180.00	3.183	2.308	271.38
	HM after 100 searches			HM after 1000 searches			HM after 4870 searches		
1	0.31672	0.40000	−0.2838402	0.09000	−0.71143	−1.0316159	**0.08984**[b]	**−0.71269**[b]	**−1.0316285**[b]
2	0.23333	0.32581	−0.2439473	0.09028	−0.71143	−1.0316149	0.09000	−0.71269	−1.0316284
3	0.26504	0.32581	−0.1951466	0.08863	−0.71143	−1.0316119	0.09000	−0.71277	−1.0316283
4	0.23333	0.28628	−0.1561579	0.09081	−0.71143	−1.0316114	0.09013	−0.71269	−1.0316281
5	0.35011	0.30594	0.0128968	0.09000	−0.71446	−1.0316020	0.08951	−0.71269	−1.0316280
6	0.26504	0.22232	0.0238899	0.09028	−0.71446	−1.0316019	0.08951	−0.71277	−1.0316279
7	0.35011	0.28628	0.0581726	0.09081	−0.71446	−1.0316000	0.08951	−0.71279	−1.0316278
8	0.31883	0.25029	0.0705802	0.09000	−0.71062	−1.0315942	0.09028	−0.71269	−1.0316277
9	0.35011	0.23078	0.1768801	0.08863	−0.71446	−1.0315939	0.08980	−0.71300	−1.0316275
10	0.54693	0.28628	0.5600001	0.09028	−0.71062	−1.0315928	0.09000	−0.71300	−1.0316274

[a] A new harmony improvised in first search and the corresponding function value.
[b] The best solution vector (0.08984, −0.71269) and the corresponding optimum point (−1.0316285) found using the HS algorithm.

adaptation rate (HMCR) from the beginning to the end of the computations, which resembles simulated annealing. It also considers several vectors simultaneously in a manner similar to genetic algorithms. However, the major difference between genetic algorithms and the HS algorithm is that the latter generates a new vector from all the existing vectors (all harmonies in the HM), while genetic algorithms generate a new vector from only two of the existing vectors (parents). In addition, the HS algorithm can independently consider each component variable in a vector when it generates a new vector; genetic algorithms cannot, because they have to maintain the gene structure.

3. Statement of the optimization design problem

Design objectives that can be used to measure design quality include minimum construction cost, minimum life cycle cost, minimum weight, and maximum stiffness, as well as many others. Typically, the design is limited by constraints such as the choice of material, feasible strength, displacements, eigen-frequencies, load cases, support conditions, and technical constraints (e.g., type and size of available structural members and cross-sections, etc.). Hence, one must decide which parameters can be modified during the optimization process; these parameters then become the optimization variables. Isotropic structures can usually be described by three different types of design variables: (1) sizing variables, (2) geometric variables, and (3) topological variables. Size optimization is concerned with determining the cross-sectional size using a given geometry. Configuration optimization searches for a set of geometric and sizing variables using a given topology. Topology optimization selects from various structural types. In general, topology optimization is a combinatorial optimization problem.

A method to optimize the continuous pure sizing variables was introduced in this study to investigate the applicability of the HS algorithm for structural optimization problems. Size optimization of structural systems involves arriving at optimum values for member cross-sectional areas $\boldsymbol{A}$ that minimize an objective function F, usually the structural weight W. This minimum design also has to satisfy q inequality constraints that limit design variable sizes and structural responses. Thus, the design problem may be expressed as

$$\text{Minimize} \quad F = W(\boldsymbol{A}) = \gamma \sum_{i=1}^{n} L_i A_i \tag{7}$$

$$\text{Subject to} \quad {}_{\mathrm{L}}G_j \leqslant G_j(\boldsymbol{A}) \leqslant {}_{\mathrm{U}}G_j, \quad j = 1, 2, \ldots, q \tag{8}$$

where L_i = the member length and γ = the material density. For the method presented in this study, the upper and lower bounds on the constraint function in Eq. (8) include the following: (a) member cross-sectional areas (${}_{\mathrm{L}}A_i \leqslant A_i \leqslant {}_{\mathrm{U}}A_i,\ i = 1, \ldots, n$); (b) member stresses (${}_{\mathrm{L}}\sigma_i \leqslant \sigma_i \leqslant {}_{\mathrm{U}}\sigma_i,\ i = 1, \ldots, n$); (c) elastic nodal displacements (${}_{\mathrm{L}}\delta_i \leqslant \delta_i \leqslant {}_{\mathrm{U}}\delta_i,\ i = 1, \ldots, m$); and (d) member buckling stresses (${}_{\mathrm{b}}\sigma_i \leqslant \sigma_i \leqslant 0,\ i = 1, \ldots, n$).

4. HS algorithm-based structural optimization and design procedure

Solutions to the size optimization problems described by Eqs. (7) and (8) can be obtained using the HS algorithm process by (a) defining value bounds for the design variables, (b) generating the initial harmony memory (HM), (c) improvising a new harmony, (d) evaluating the objective function under the constraint functions using structural analysis, and (e) updating the initialized HM. Here, we will focus on the HS algorithm mechanism to arrive at optimum values.

Fig. 3 shows the detailed procedure of the proposed HS algorithm-based method to determine optimal cross-sections in size optimization problems. The detailed procedure can be divided into the following two steps:

Step 1: Initialization. HS algorithm parameters such as HMS, HMCR, PAR, maximum number of searches, number of x_i, and design variable bounds (member cross-sections $\boldsymbol{A}$) are initialized. Harmonies (i.e., solution vectors) are then randomly generated from the possible variable bounds that are equal to the size of the HM. Here, the initial HM is generated based on a finite element method (FEM) structural analysis subjected to

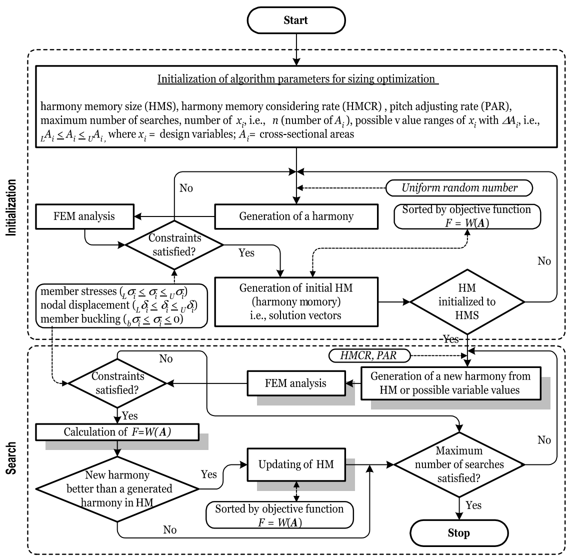


Fig. 3. HS algorithm-based structural optimization design procedure.

the constraint functions (Eq. (8)) and sorted by the objective function values (Eq. (7)).

Step 2: Search. A new harmony is improvised from the initially generated HM or possible variable values using the HMCR and PAR parameters. These parameters are introduced to allow the solution to escape from local optima and to improve the global optimum prediction in the HS algorithm. The new harmony is analyzed using the FEM method, and its fitness is evaluated using the constraint functions. If satisfied, the weight of the structure is calculated using the objective function. If the new harmony is better than the previous worst harmony, the new harmony is included in the HM and the previous worst harmony is excluded from the HM. The HM is then sorted by the objective function value. The computations terminate when the maximum number of the search criterion is satisfied. If not, this step is repeated.

5. Numerical examples

Standard test cases that have been used in previous truss size optimization papers were considered in this study using a FORTRAN computer program developed to demonstrate the efficiency and robustness of the HS algorithm. These cases include a 10-bar planar truss subjected to a single load condition, a 17-bar planar truss subjected to a single load condition, an 18-bar planar truss subjected to a single load condition, a 22-bar space truss subjected to three load conditions, a 25-bar space truss subjected to two load conditions, a 72-bar space truss subjected to two load conditions, a 200-bar planar truss subjected to three load conditions, and a 120-bar dome space truss subjected to a single load condition. These truss structures were analyzed using the FEM displacement method. For all examples presented in this paper, the HS algorithm parameters were set as follows: harmony memory size (HMS) = 20, harmony memory consideration rate (HMCR) = 0.8, pitch adjusting rate (PAR) = 0.3, and maximum number of searches = 50,000.

5.1. Ten-bar planar truss

The cantilever truss, shown in Fig. 4, was previously analyzed using various mathematical methods by Schmit and Farshi [28], Schmit and Miura [29], Venkayya [30], Gellatly and Berke [31], Dobbs and Nelson [32], Rizzi [33], Khan and Willmert [34], John et al. [35], Sunar and Belegundu [36], Stander et al. [37], Xu and Grandhi [38], and Lamberti and Pappalettere [39,40]. The material density was 0.1 lb/in.^{3} and the modulus of elasticity was 10,000 ksi. The members were subjected to stress limitations of ±25 ksi, and displacement limitations of ±2.0 in. were imposed on all nodes in both directions

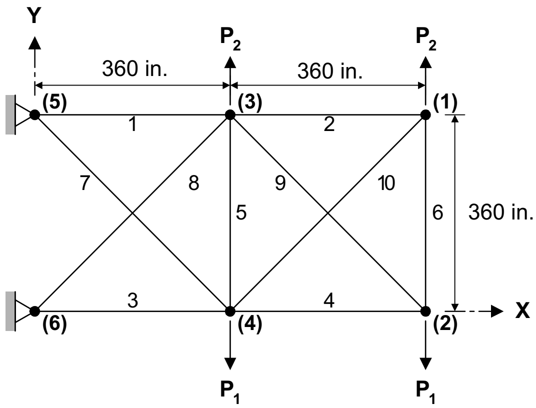


Fig. 4. Ten-bar planar truss.

(x and y). No design-variable linking was used; thus there are 10 independent design variables. In this example, two cases were considered: Case 1, in which the single loading condition of $P_1 = 100$ kips and $P_2 = 0$ was considered; and Case 2, in which the single loading condition of $P_1 = 150$ kips and $P_2 = 50$ kips was considered. The minimum cross-sectional area of the members was 0.1 in.^{2}.

The HS algorithm-based method was applied to each case. It found 20 different solution vectors (i.e., the values of the 10 design-independent variables) after 50,000 searches for both cases. Tables 2 and 3 give the best solution vector for Cases 1 and 2, respectively, and also provide a comparison between the optimal design results reported in the literature and the present work. For Case 1, the best HS solution vector was (30.15, 0.102, 22.71, 15.27, 0.102, 0.544, 7.541, 21.56, 21.45, 0.100) and the corresponding objective function value (minimum weight of the structure) was 5057.88 lb. For Case 2, the best HS solution vector was (23.25, 0.102, 25.73, 14.51, 0.100, 1.977, 12.21, 12.61, 20.36, 0.100) and the corresponding objective function value was 4668.81 lb. The best solutions for Cases 1 and 2 were obtained after approximately 20,000 and 15,000 searches, respectively, as shown in Fig. 5. These searches took three and two minutes on a Pentium 600 MHz computer. The HS algorithm results for each case were better optimized than by previous mathematical studies reported in the literature.

5.2. Seventeen-bar planar truss

The 17-bar planar truss, shown in Fig. 6, has been studied by Khot and Berke [41] and Adeli and Kumar [14]. The material density was 0.268 lb/in.^{3} and the modulus of elasticity was 30,000 ksi. The members were subjected to stress limitations of ±50 ksi, and displacement limitations of ±2.0 in. were imposed on all nodes in

Table 2
Optimal design comparison for the 10-bar planar truss (Case 1)

Variables		Optimal cross-sectional areas (in.^{2})													
		Schmit and Farshi [28]	Schmit and Miura [29]		Venkayya [30]	Gellatly and Berke [31]	Dobbs and Nelson [32]	Rizzi [33]	Khan and Willmert [34]	Sunar and Belegundu [36]	Stander et al. [37]	Xu and Grandhi [38]	Lamberti and Pappalettere		*This work*
			NEWSUMT	CONMIN									LEAML [39]	LESLP [40]	
1	A_1	33.43	30.67	30.57	30.42	31.35	30.50	30.73	30.98						*30.15*
2	A_2	0.100	0.100	0.369	0.128	0.100	0.100	0.100	0.100						*0.102*
3	A_3	24.26	23.76	23.97	23.41	20.03	23.29	23.93	24.17						*22.71*
4	A_4	14.26	14.59	14.73	14.91	15.60	15.43	14.73	14.81						*15.27*
5	A_5	0.100	0.100	0.100	0.101	0.140	0.100	0.100	0.100	*	*	*	*	*	*0.102*
6	A_6	0.100	0.100	0.364	0.101	0.240	0.210	0.100	0.406						*0.544*
7	A_7	8.388	8.578	8.547	8.696	8.350	7.649	8.542	7.547						*7.541*
8	A_8	20.74	21.07	21.11	21.08	22.21	20.98	20.95	21.05						*21.56*
9	A_9	19.69	20.96	20.77	21.08	22.06	21.82	21.84	20.94						*21.45*
10	A_{10}	0.100	0.100	0.320	0.186	0.100	0.100	0.100	0.100						*0.100*
Weight (lb)		5089.0	5076.85	5107.3	5084.9	5112.0	5080.0	5076.66	5066.98	5060.9	5060.85	5065.25	5060.96	5060.88	*5057.88*

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

Table 3
Optimal design comparison for the 10-bar planar truss (Case 2)

Variables		Optimal cross-sectional areas (in.^{2})								
		Schmit and Farshi [28]	Schmit and Miura [29]		Venkayya [30]	Dobbs and Nelson [32]	Rizzi [33]	Khan and Willmert [34]	John et al. [35]	*This work*
			NEWSUMT	CONMIN						
1	A_1	24.29	23.55	23.55	25.19	25.81	23.53	24.72	23.59	*23.25*
2	A_2	0.100	0.100	0.176	0.363	0.100	0.100	0.100	0.10	*0.102*
3	A_3	23.35	25.29	25.20	25.42	27.23	25.29	26.54	25.25	*25.73*
4	A_4	13.66	14.36	14.39	14.33	16.65	14.37	13.22	14.37	*14.51*
5	A_5	0.100	0.100	0.100	0.417	0.100	0.100	0.108	0.10	*0.100*
6	A_6	1.969	1.970	1.967	3.144	2.024	1.970	4.835	1.97	*1.977*
7	A_7	12.67	12.39	12.40	12.08	12.78	12.39	12.66	12.39	*12.21*
8	A_8	12.54	12.81	12.86	14.61	14.22	12.83	13.78	12.80	*12.61*
9	A_9	21.97	20.34	20.41	20.26	22.14	20.33	18.44	20.37	*20.36*
10	A_{10}	0.100	0.100	0.100	0.513	0.100	0.100	0.100	0.10	*0.100*
Weight (lb)		4691.84	4676.96	4684.11	4895.60	5059.7	4676.92	4792.52	4676.93	*4668.81*

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

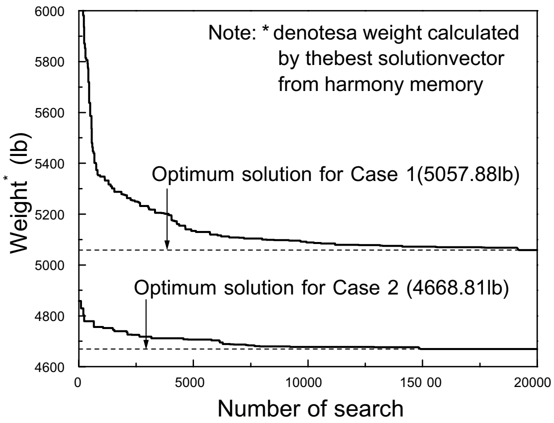


Fig. 5. Convergence history for the minimum weight of the 10-bar planar truss.

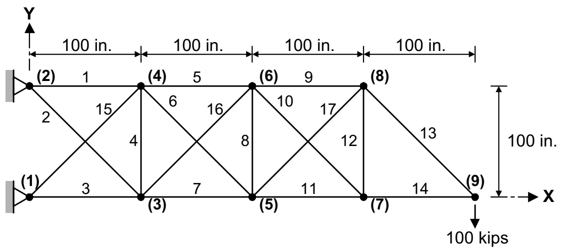


Fig. 6. Seventeen-bar planar truss.

both directions (x and y). The single vertical downward load of 100 kips at node 9 was considered. No design-variable linking was used; thus there are seventeen independent design variables. The minimum cross-sectional area of the members was 0.1 in.^{2}.

The HS algorithm was applied to the 17-bar planar truss with 17 independent design variables. The optimal results were compared to the earlier solutions reported by Khot and Berke [41] and Adeli and Kumar [14] in Table 4. Khot and Berke solved the problem using the optimality criterion method, and obtained a minimum weight of 2581.89 lb. However, Adeli and Kumar solved the problem using a variant of genetic algorithms, obtaining a minimum weight of 2594.42 lb. The HS heuristic algorithm-based technique found an optimum weight of 2580.81 lb after approximately 20,000 searches that took less than three minute. The optimal design obtained using the HS algorithm was slightly better than both of the previous design results.

5.3. Eighteen-bar planar truss

The 18-bar cantilever planar truss, shown in Fig. 7, was analyzed by Imai and Schmit [42] to obtain the optimal size variables. The material density was 0.1 lb/in.^{3} and the modulus of elasticity was 10,000 ksi. The members were subjected to stress limitations of ±20 ksi. Also, an Euler bucking compressive stress limitation was imposed for truss member i, according to

Table 4
Optimal design comparison for the 17-bar planar truss

Variables		Optimal cross-sectional areas (in.^{2})		
		Khot and Berke [41]	Adeli and Kumar [14]	*This work*
1	A_1	15.930	16.029	*15.821*
2	A_2	0.100	0.107	*0.108*
3	A_3	12.070	12.183	*11.996*
4	A_4	0.100	0.110	*0.100*
5	A_5	8.067	8.417	*8.150*
6	A_6	5.562	5.715	*5.507*
7	A_7	11.933	11.331	*11.829*
8	A_8	0.100	0.105	*0.100*
9	A_9	7.945	7.301	*7.934*
10	A_{10}	0.100	0.115	*0.100*
11	A_{11}	4.055	4.046	*4.093*
12	A_{12}	0.100	0.101	*0.100*
13	A_{13}	5.657	5.611	*5.660*
14	A_{14}	4.000	4.046	*4.061*
15	A_{15}	5.558	5.152	*5.656*
16	A_{16}	0.100	0.107	*0.100*
17	A_{17}	5.579	5.286	*5.582*
Weight (lb)		2581.89	2594.42	*2580.81*

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

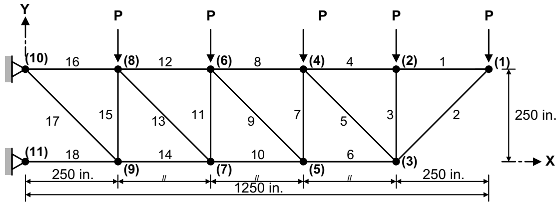


Fig. 7. Eighteen-bar planar truss.

$$_{\mathrm{b}}\sigma_i = \frac{-KEA_i}{L_i^2} \tag{9}$$

where K = a constant determined from the cross-sectional geometry; E = the modulus of elasticity; and L_i = the member length. In this study, the buckling constant was taken to be $K = 4$. The single loading condition was a set of vertical loads with $P = 20$ kips acting on the upper nodal points of the truss, as illustrated in Fig. 7. The cross-sectional areas of the members were linked into four groups, as follows: (1) $A_1 = A_4 = A_8 = A_{12} = A_{16}$, (2) $A_2 = A_6 = A_{10} = A_{14} = A_{18}$, (3) $A_3 = A_7 = A_{11} = A_{15}$, and (4) $A_5 = A_9 = A_{13} = A_{17}$. The minimum cross-sectional area was 0.1 in.^{2}.

The HS algorithm-based method was applied to the 18-bar truss with four independent design variables. The optimal results are compared to the earlier solutions reported by Imai and Schmit [42] in Table 5. Imai and Schmit solved the problem using the multiplier method, and obtained a minimum weight of 6430.0 lb. The HS algorithm-based method found an optimum weight of 6421.88 lb after approximately 2000 searches that took less than one minute. The optimal design obtained using the HS algorithm was slightly better than the previous design obtained by Imai and Schmit.

5.4. Twenty-two-bar space truss

In this structure, shown in Fig. 8, each node is connected to every other node by a member, except for members between the fixed support nodes 5, 6, 7, and 8. The structure was previously studied by Khan and Willmert [34] and Sheu and Schmit [43] to determine the global optimum of trusses with vanishing members. In the example considered in this study, however, only the case with all groups of members (non-vanishing) was considered.

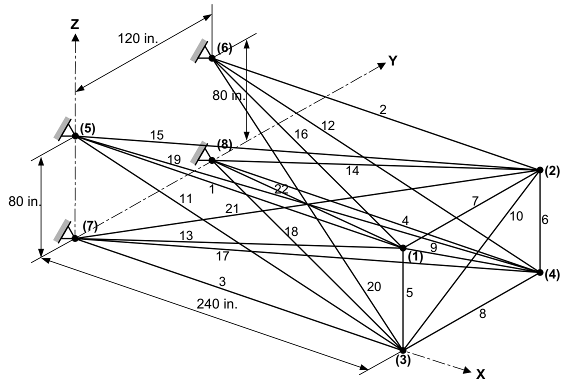


Fig. 8. Twenty-two-bar space truss.

The modulus of elasticity and the material density of all members were 10,000 ksi and 0.1 lb/in.^{3}, respectively. The 22 members were linked into seven groups, as follows: (1) $A_1 \sim A_4$, (2) $A_5 \sim A_6$, (3) $A_7 \sim A_8$, (4) $A_9 \sim A_{10}$, (5) $A_{11} \sim A_{14}$, (6) $A_{15} \sim A_{18}$, and (7) $A_{19} \sim A_{22}$. The truss members were subjected to the stress limitations shown in Table 6. Also, displacement constraints of ±2.0 in. were imposed on all nodes in all directions. Three loading conditions described in Table 7 were considered, and a minimum member cross-sectional area of 0.1 in.^{2} was enforced.

Table 8 lists the optimal values of the seven size variables obtained by the HS algorithm-based method, and compares them with earlier results reported by Khan and Willmert [34] and Sheu and Schmit [43]. The HS algorithm-based method achieves a design with a best solution vector of $(2.588, 1.083, 0.363, 0.422, 2.827, 2.055, 2.044)$ and a minimum weight of 1022.23 lb after approximately 10,000 searches. The optimal design obtained using the HS algorithm was slightly better than the results obtained by Sheu and Schmit [43], and had a

Table 5
Optimal design comparison for the 18-bar planar truss

Variables		Optimal cross-sectional areas (in.^{2})	
		Imai and Schmit [42]	*This work*
1	$A_1 = A_4 = A_8 = A_{12} = A_{16}$	9.998	*9.980*
2	$A_2 = A_6 = A_{10} = A_{14} = A_{18}$	21.65	*21.63*
3	$A_3 = A_7 = A_{11} = A_{15}$	12.50	*12.49*
4	$A_5 = A_9 = A_{13} = A_{17}$	7.072	*7.057*
Weight (lb)		6430.0	*6421.88*

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

Table 6
Member stress limitations for the 22-bar space truss

Variables		Compressive stress limitations (ksi)	Tensile stress limitations (ksi)
1	$A_1 \sim A_4$	24.0	36.0
2	$A_5 \sim A_6$	30.0	36.0
3	$A_7 \sim A_8$	28.0	36.0
4	$A_9 \sim A_{10}$	26.0	36.0
5	$A_{11} \sim A_{14}$	22.0	36.0
6	$A_{15} \sim A_{18}$	20.0	36.0
7	$A_{19} \sim A_{22}$	18.0	36.0

minimum weight that was 1.2% less than that obtained by Khan and Willmert [34].

5.5. Twenty-five-bar space truss

The 25-bar transmission tower space truss, shown in Fig. 9, has been size optimized by many researchers. These include Schmit and Farshi [28], Schmit and Miura [29], Venkayya [30], Gellatly and Berke [31], Rizzi [33], Khan and Willmert [34], Templeman and Winterbottom [44], Chao et al. [45], Adeli and Kamal [46], John et al. [35], Saka [47], Fadel and Clitalay [48], Stander et al. [37], Xu and Grandhi [38], and Lamberti and Pappalettere [39,40]. In these studies, the material density was 0.1 lb/in.^{3} and modulus of elasticity was 10,000 ksi. This space truss was subjected to the two loading conditions shown in Table 9. The structure was required to be doubly symmetric about the x- and y-axes; this condition grouped the truss members as follows: (1) A_1, (2) $A_2 \sim A_5$, (3) $A_6 \sim A_9$, (4) $A_{10} \sim A_{11}$, (5) $A_{12} \sim A_{13}$, (6) $A_{14} \sim A_{17}$, (7) $A_{18} \sim A_{21}$, and (8) $A_{22} \sim A_{25}$.

The truss members were subjected to the compressive and tensile stress limitations shown in Table 10. In addition, maximum displacement limitations of ±0.35 in. were imposed on every node in every direction. The minimum cross-sectional area of all members was 0.01 in.^{2}.

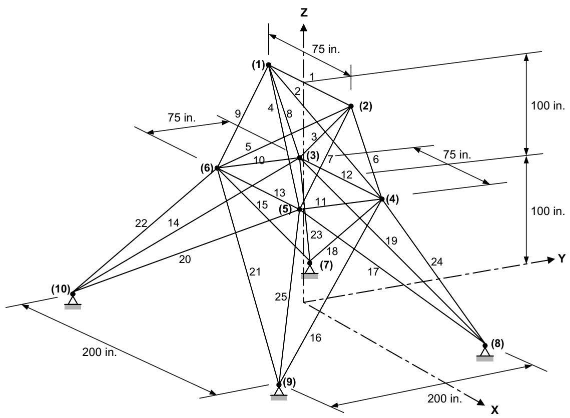


Fig. 9. Twenty-five-bar space truss.

Table 7
Loading conditions for the 22-bar space truss

Node	Condition 1			Condition 2			Condition 3		
	P_X	P_Y	P_Z	P_X	P_Y	P_Z	P_X	P_Y	P_Z
1	−20.0	0.0	−5.0	−20.0	−5.0	0.0	−20.0	0.0	35.0
2	−20.0	0.0	−5.0	−20.0	−50.0	0.0	−20.0	0.0	0.0
3	−20.0	0.0	−30.0	−20.0	−5.0	0.0	−20.0	0.0	0.0
4	−20.0	0.0	−30.0	−20.0	−50.0	0.0	−20.0	0.0	−35.0

Note: loads are in kips.

Table 8
Optimal design comparison for the 22-bar space truss

Variables		Optimal cross-sectional areas (in.^{2})		
		Sheu and Schmit [43]	Khan and Willmert [34]	*This work*
1	$A_1 \sim A_4$	2.629	2.563	*2.588*
2	$A_5 \sim A_6$	1.162	1.553	*1.083*
3	$A_7 \sim A_8$	0.343	0.281	*0.363*
4	$A_9 \sim A_{10}$	0.423	0.512	*0.422*
5	$A_{11} \sim A_{14}$	2.782	2.626	*2.827*
6	$A_{15} \sim A_{18}$	2.173	2.131	*2.055*
7	$A_{19} \sim A_{22}$	1.952	2.213	*2.044*
Weight (lb)		1024.80	1034.74	*1022.23*

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

Table 9
Loading conditions for the 25-bar space truss

Node	Condition 1			Condition 2		
	P_X	P_Y	P_Z	P_X	P_Y	P_Z
1	0.0	20.0	−5.0	1.0	10.0	−5.0
2	0.0	−20.0	−5.0	0.0	10.0	−5.0
3	0.0	0.0	0.0	0.5	0.0	0.0
6	0.0	0.0	0.0	0.5	0.0	0.0

Note: loads are in kips.

Table 10
Member stress limitations for the 25-bar space truss

Variables		Compressive stress limitations (ksi)	Tensilestress limitations (ksi)
1	A_1	35.092	40.0
2	$A_2 \sim A_5$	11.590	40.0
3	$A_6 \sim A_9$	17.305	40.0
4	$A_{10} \sim A_{11}$	35.092	40.0
5	$A_{12} \sim A_{13}$	35.092	40.0
6	$A_{14} \sim A_{17}$	6.759	40.0
7	$A_{18} \sim A_{21}$	6.959	40.0
8	$A_{22} \sim A_{25}$	11.082	40.0

The best HS algorithm solution vector for the eight design variables, obtained after approximately 15,000 searches, was $(0.047, 2.022, 2.95, 0.01, 0.014, 0.688, 1.657, 2.663)$. The corresponding optimum weight was 544.38 lb. Table 11 gives a comparison between the optimal solutions reported in the literature and the present work. The HS algorithm produced a slightly better solution than did any of the earlier mathematical studies.

5.6. Seventy-two-bar space truss

The 72-bar space truss, shown in Fig. 10, has also been size optimized by many researchers, including Schmit and Farshi [28], Schmit and Miura [29], Venkayya [30], Gellatly and Berke [31], Khan and Willmert [34], Chao et al. [45], Adeli and Kamal [46], Berke and Khot [49], Xicheng and Guixu [50], Erbatur et al. [20], Adeli and Park [51], and Sarma and Adeli [21]. In these studies, the material density and modulus of elasticity were 0.1 lb/in.^{3} and 10,000 ksi, respectively. This space truss was subjected to the following two loading conditions: Condition 1, in which $P_X = 5.0$ kips, $P_Y = 5.0$ kips, and $P_Z = -5.0$ kips on node 17; and Condition 2, in which $P_X = 0.0$ kips, $P_Y = 0.0$ kips, and $P_Z = -5.0$ kips on nodes 17, 18, 19, and 20. The structure was required to be doubly symmetric about the x- and y-axes. This condition divided the truss members into the following 16 groups: (1) $A_1 \sim A_4$, (2) $A_5 \sim A_{12}$, (3) $A_{13} \sim A_{16}$, (4) $A_{17} \sim A_{18}$, (5) $A_{19} \sim A_{22}$, (6) $A_{23} \sim A_{30}$, (7) $A_{31} \sim A_{34}$, (8) $A_{35} \sim A_{36}$, (9) $A_{37} \sim A_{40}$, (10) $A_{41} \sim A_{48}$, (11) $A_{49} \sim A_{52}$, (12) $A_{53} \sim A_{54}$, (13) $A_{55} \sim A_{58}$, (14) $A_{59} \sim A_{66}$, (15) $A_{67} \sim A_{70}$, and (16) $A_{71} \sim A_{72}$.

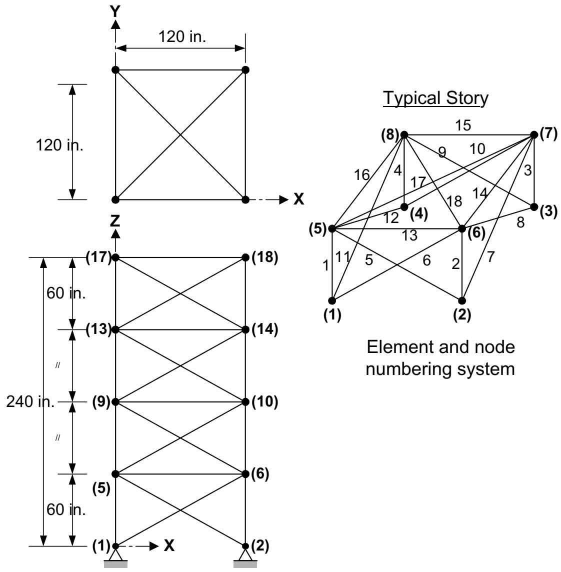


Fig. 10. Seventy-two-bar space truss.

The members were subjected to stress limitations of ±25 ksi, and the maximum displacement of uppermost nodes was not allowed to exceed ±0.25 in. in the x and y directions. In this example, two cases were considered: Case 1, in which the minimum cross-sectional area of all members was 0.1 in.^{2}; and Case 2, in which the minimum cross-sectional area of 0.01 in.^{2} was considered.

Tables 12 and 13 show the HS algorithm's optimal results for Cases 1 and 2 with the 16 size variables, and compares these results with those previously reported in the literature. For Case 1, the method proposed in this paper achieved a design with the best solution vector of $(1.790, 0.521, 0.100, 0.100, 1.229, 0.522, 0.100, 0.100, 0.517, 0.504, 0.100, 0.101, 0.156, 0.547, 0.442, 0.590)$ and a corresponding minimum weight of 379.27 lb after approximately 20,000 searches, which took 10 min on a

Table 11
Comparison of optimal design for 25-bar space truss

Variables		Optimal cross-sectional areas (in.^{2})								
		Schmit and Farshi [28]	Schmit and Miura [29] NEWSUMT	Schmit and Miura [29] CONMIN	Venkayya [30]	Gellatly and Berke [31]	Rizzi [33]	Khan and Willmert [34]	Templeman and Winter-bottom [44]	Chao et al. [45]
1	A_1	0.010	0.010	0.166	0.028	0.010	0.010	0.010	0.010	0.010
2	$A_2 \sim A_5$	1.964	1.985	2.017	1.964	2.007	1.988	1.755	2.022	2.042
3	$A_6 \sim A_9$	3.033	2.996	3.026	3.081	2.963	2.991	2.869	2.938	3.001
4	$A_{10} \sim A_{11}$	0.010	0.010	0.087	0.010	0.010	0.010	0.010	0.010	0.010
5	$A_{12} \sim A_{13}$	0.010	0.010	0.097	0.010	0.010	0.010	0.010	0.010	0.010
6	$A_{14} \sim A_{17}$	0.670	0.684	0.675	0.693	0.688	0.684	0.845	0.670	0.684
7	$A_{18} \sim A_{21}$	1.680	1.667	1.636	1.678	1.678	1.677	2.011	1.675	1.625
8	$A_{22} \sim A_{25}$	2.670	2.662	2.669	2.627	2.664	2.663	2.478	2.697	2.672
Weight (lb)		545.22	545.17	548.47	545.49	545.36	545.16	553.94	545.32	545.03

		Adeli and Kamal [46]	John et al. [35]	Saka [47]	Fadel and Clitalay [48]	Stander et al. [37]	Xu and Grandhi [38]	Lamberti and Pappalettere LEAML [39]	Lamberti and Pappalettere LESLP [40]	*This work*
1	A_1	0.010		0.010		0.010				*0.047*
2	$A_2 \sim A_5$	1.986		2.085		2.043				*2.022*
3	$A_6 \sim A_9$	2.961		2.988		3.003				*2.950*
4	$A_{10} \sim A_{11}$	0.010	*	0.010	*	0.010	*	*	*	*0.010*
5	$A_{12} \sim A_{13}$	0.010		0.010		0.010				*0.014*
6	$A_{14} \sim A_{17}$	0.806		0.696		0.683				*0.688*
7	$A_{18} \sim A_{21}$	1.680		1.670		1.623				*1.657*
8	$A_{22} \sim A_{25}$	2.530		2.592		2.672				*2.663*
Weight (lb)		545.66	555.58	545.23	545.49	545.03	545.60	545.17	545.17	*544.38*

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

Table 12
Optimal design comparison for the 72-bar space truss (Case 1)

Variables		Optimal cross-sectional areas (in.^{2})													
		Schmit and Farshi [28]	Schmit and Miura [29]		Venkayya [30]	Gellatly and Berke [31]	Khan and Willmert [34]		Chao et al. [45]	Adeli and Kamal [46]	Berke and Khot [49]	Xicheng and Guixu [50]	Erbatur et al. [20]		*This work*
			NEWSUMT	CONMIN			$\eta = 0.1$	$\eta = 0.15$					GAOS1	GAOS2	
1	$A_1 \sim A_4$	2.078	1.885	1.885	1.818	1.464	1.793	1.859	1.832	2.026	1.893	1.905	1.755	1.910	*1.790*
2	$A_5 \sim A_{12}$	0.503	0.513	0.512	0.524	0.521	0.522	0.526	0.512	0.533	0.517	0.518	0.505	0.525	*0.521*
3	$A_{13} \sim A_{16}$	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.105	0.122	*0.100*
4	$A_{17} \sim A_{18}$	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.155	0.103	*0.100*
5	$A_{19} \sim A_{22}$	1.107	1.267	1.268	1.246	1.024	1.208	1.253	1.252	1.157	1.279	1.286	1.155	1.310	*1.229*
6	$A_{23} \sim A_{30}$	0.579	0.512	0.511	0.524	0.542	0.521	0.524	0.524	0.569	0.515	0.516	0.585	0.498	*0.522*
7	$A_{31} \sim A_{34}$	0.100	0.100	0.100	0.100	0.10	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.110	*0.100*
8	$A_{35} \sim A_{36}$	0.100	0.100	0.100	0.100	0.10	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.103	*0.100*
9	$A_{37} \sim A_{40}$	0.264	0.523	0.523	0.611	0.552	0.623	0.581	0.513	0.514	0.508	0.509	0.460	0.535	*0.517*
10	$A_{41} \sim A_{48}$	0.548	0.517	0.5161	0.532	0.608	0.523	0.527	0.529	0.479	0.520	0.522	0.530	0.535	*0.504*
11	$A_{49} \sim A_{52}$	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.120	0.103	*0.100*
12	$A_{53} \sim A_{54}$	0.151	0.100	0.113	0.100	0.100	0.196	0.158	0.100	0.100	0.100	0.100	0.165	0.111	*0.101*
13	$A_{55} \sim A_{58}$	0.158	0.157	0.156	0.161	0.149	0.149	0.152	0.157	0.158	0.157	0.157	0.155	0.161	*0.156*
14	$A_{59} \sim A_{66}$	0.594	0.546	0.548	0.557	0.773	0.570	0.561	0.549	0.550	0.539	0.537	0.535	0.544	*0.547*
15	$A_{67} \sim A_{70}$	0.341	0.411	0.411	0.377	0.453	0.443	0.438	0.406	0.345	0.416	0.411	0.480	0.379	*0.442*
16	$A_{71} \sim A_{72}$	0.608	0.570	0.561	0.506	0.342	0.519	0.532	0.555	0.498	0.551	0.571	0.520	0.521	*0.590*
Weight (lb)		388.63	379.64	379.79	381.2	395.97	381.72	387.67	379.62	379.31	379.67	380.84	385.76	383.12	*379.27*

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

Pentium 600 MHz computer. For Case 2, the best HS solution vector was (1.963, 0.481, 0.010, 0.011, 1.233, 0.506, 0.011, 0.012, 0.538, 0.533, 0.010, 0.167, 0.161, 0.542, 0.478, 0.551) and a corresponding minimum weight was 364.33 lb, which were also obtained after approximately 20,000 searches. The optimal design results for both cases obtained using the HS approach were slightly better than all of the previous results that were obtained using mathematical and genetic algorithms.

On the other hand, Fig. 11 shows a comparison of convergence capability for Case 2 between the HS result and those obtained by Sarma and Adeli [21] using the simple and fuzzy genetic algorithm-based methods. A fuzzy genetic algorithm obtained a minimum weight of 364.40 lb after 1758 structural analyses (B in Fig. 11), while the HS algorithm obtained the same weight after 14,669 analyses (point b in Fig. 11). The fuzzy controlled genetic algorithm method shows a better convergence capability than the present approach proposed on the basis of the pure HS algorithm. However, it should be noted that the proposed HS approach outperforms a simple genetic algorithm-based method in terms of both the convergence capability and the optimal solution, as shown in Fig. 11. The simple genetic algorithm obtained a minimum weight of 372.40 lb after 2776 analyses (A in Fig. 11), but the HS approach required 1076 structural analyses for the same weight (point a in Fig. 11).

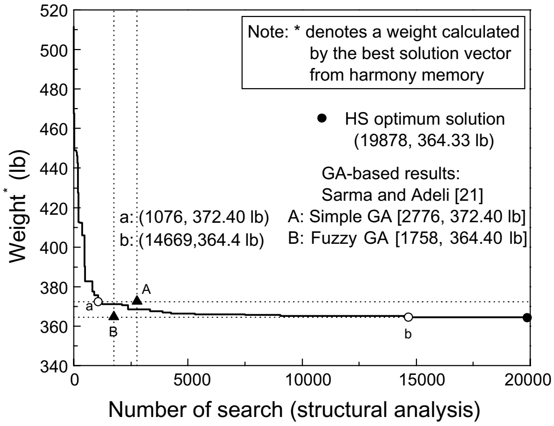


Fig. 11. Convergence history of the minimum weight for 72-bar space truss (Case 2).

Table 13
Optimal design comparison for the 72-bar space truss (Case 2)

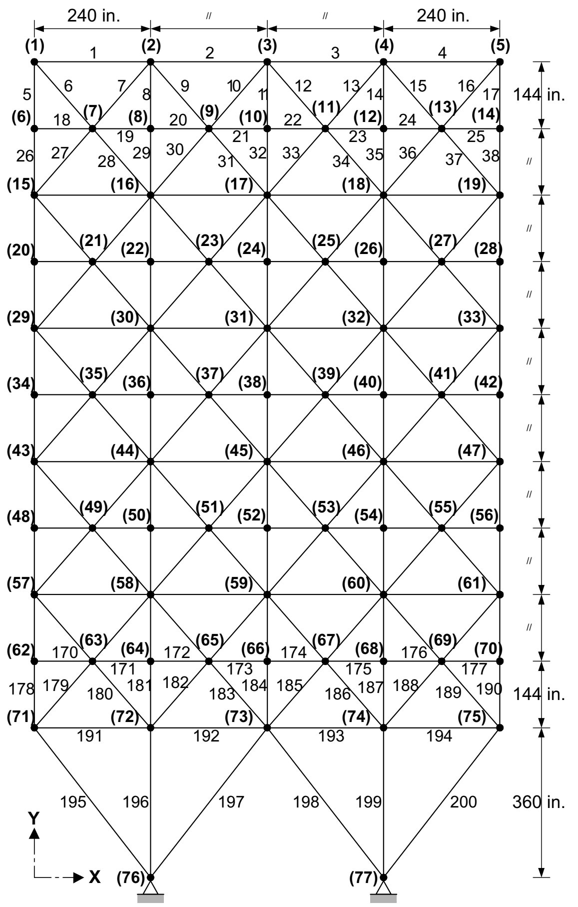
Variables		Optimal cross-sectional areas (in.^{2})			
		Adeli and Park [51]	Sarma and Adeli [21]		*This work*
			Simple GA	Fuzzy GA	
1	$A_1 \sim A_4$	2.755	2.141	1.732	*1.963*
2	$A_5 \sim A_{12}$	0.510	0.510	0.522	*0.481*
3	$A_{13} \sim A_{16}$	0.010	0.054	0.010	*0.010*
4	$A_{17} \sim A_{18}$	0.010	0.010	0.013	*0.011*
5	$A_{19} \sim A_{22}$	1.370	1.489	1.345	*1.233*
6	$A_{23} \sim A_{30}$	0.507	0.551	0.551	*0.506*
7	$A_{31} \sim A_{34}$	0.010	0.057	0.010	*0.011*
8	$A_{35} \sim A_{36}$	0.010	0.013	0.013	*0.012*
9	$A_{37} \sim A_{40}$	0.481	0.565	0.492	*0.538*
10	$A_{41} \sim A_{48}$	0.508	0.527	0.545	*0.533*
11	$A_{49} \sim A_{52}$	0.010	0.010	0.066	*0.010*
12	$A_{53} \sim A_{54}$	0.643	0.066	0.013	*0.167*
13	$A_{55} \sim A_{58}$	0.215	0.174	0.178	*0.161*
14	$A_{59} \sim A_{66}$	0.518	0.425	0.524	*0.542*
15	$A_{67} \sim A_{70}$	0.419	0.437	0.396	*0.478*
16	$A_{71} \sim A_{72}$	0.504	0.641	0.595	*0.551*
Weight (lb)		376.50	372.40	364.40	*364.33* [*364.40*][a] [*372.40*][b]
Number of structural analyses		–	2776	1758	*19878* [*14,669*][a] [*1076*][b]

Note: 1 in.^{2} = 6.452 cm^2, 1 lb = 4.45 N.
[a] HS obtained a weight of 364.40 lb after 14,669 analyses (the result of Fuzzy GA).
[b] HS obtained a weight of 372.40 lb after 1076 analyses (the result of Simple GA).

5.7. Two-hundred-bar planar truss

The 200-bar plane truss, shown in Fig. 12, has been size optimized using mathematical methods by Stander et al. [37] and Lamberti and Pappalettere [39,40]. All members are made of steel: the material density and modulus of elasticity were 0.283 lb/in.^{3} and 30,000 ksi, respectively. This truss was subjected to constraints only on stress limitations of ±10 ksi. There were three loading conditions: (1) 1.0 kip 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, …, 71, 72, 73, 74, and 75; and (3) conditions 1 and 2 acting together. The 200 members of this truss were linked into twenty-nine groups, as shown in Table 14. The minimum cross-sectional area of all members was 0.1 in.^{2}.

The HS algorithm-based method was applied to the 200-bar truss with 29 independent design variables. It found 20 different solution vectors after 50,000 searches.



Note: For the sake of clarity, not all members are numbered in this figure.

Fig. 12. Two-hundred-bar planar truss.

The best result is compared to the earlier solutions reported by Stander et al. [37] and Lamberti and Pappalettere [39,40] in Table 14. The HS algorithm-based method found an optimum weight of 25447.1 lb after approximately 48,000 searches. The optimal design obtained using the HS algorithm showed an excellent agreement with the previous mathematical designs reported in the literature.

5.8. One-hundred-twenty-bar dome truss

The design of 120-bar dome truss, shown in Fig. 13, was considered as a last example to demonstrate the practical capability of the HS heuristic algorithm-based method. This dome truss was first analyzed by Soh and Yang [17] to obtain the optimal sizing and configuration variables (i.e., the structural configuration optimization). In the example considered in this study, however, only sizing variables to minimize the structural weight were considered. In addition, the allowable tensile and compressive stresses were used according to the AISC ASD (1989) [52] code, as follows:

$$\begin{aligned} {}_{\mathrm{U}}\sigma_i &= 0.6F_y \quad \text{for } \sigma_i \geqslant 0 \text{ (tensile stress)} \\ {}_{\mathrm{L}}\sigma_i & \qquad\quad\;\; \text{for } \sigma_i < 0 \text{ (compressive stress)} \end{aligned} \tag{10}$$

$${}_{\mathrm{L}}\sigma_i = \begin{Bmatrix} \left[\left(1-\frac{\lambda_i^2}{2C_c^2}\right)F_y\right] \Big/ \left(\frac{5}{3}+\frac{3\lambda_i}{8C_c}-\frac{\lambda_i^3}{8C_c^3}\right) & \text{for } \lambda_i < C_c \\ \frac{12\pi^2 E}{23\lambda_i^2} & \text{for } \lambda_i \geqslant C_c \end{Bmatrix} \tag{11}$$

where E = the modulus of elasticity; F_y = the yield stress of steel; C_c = the slenderness ratio (λ_i) dividing the elastic and inelastic buckling regions $\left(C_c = \sqrt{2\pi^2 E/F_y}\right)$; λ_i = the slenderness ratio ($\lambda_i = kL_i/r_i$); k = the effective length factor; L_i = the member length; and r_i = the radius of gyration.

The modulus of elasticity (E) was 30,450 ksi and the material density was 0.288 lb/in.^{3}. The yield stress of steel (F_y) was taken as 58.0 ksi. On the other hand, the radius of gyration (r_i) can be expressed in terms of cross-sectional areas, i.e., $r_i = aA_i^b$ [47]. Here, a and b are the constants depending on the types of sections adopted for the members such as pipes, angles, and tees. In this example, pipe sections ($a = 0.4993$ and $b = 0.6777$) were adopted for bars. All members of the dome were linked into seven groups, as shown in Fig. 13. The dome was considered to be subjected to vertical loading at all the unsupported joints. These were taken as −13.49 kips at node 1, −6.744 kips at nodes 2 through 14, and −2.248 kips at the rest of the nodes. The minimum cross-sectional area of all members was 0.775 in.^{2}. In this example, two cases of displacement constraints were considered: no displacement constraints (Case 1) and displacement limitations of ±0.1969 in. imposed on all nodes in x- and y-directions.

Table 14
Optimal design comparison for the 200-bar planar truss

Group	Variables members (A_i, $i = 1, 200$)	Optimal cross-sectional areas (in.^{2})	Comparison	
			Optimizer	Weight (lb)
1	1,2,3,4	0.1253	*This work*	*25447.1*
2	5,8,11,14,17	1.0157		
3	19,20,21,22,23,24	0.1069	Stander et al. [37]	25446.70
4	18,25,56,63,94,101,132,139,170,177	0.1096		
5	26,29,32,35,38	1.9369	Lamberti and Pappalettere [39]	25446.2 (LEAML)
6	6,7,9,10,12,13,15,16,27,28,30,31,33 34,36,37	0.2686		25450.3 (CGML1)
				25446.9 (CGML2)
7	39,40,41,42	0.1042		25450.4 (CGML3)
8	43,46,49,52,55	2.9731		25450.2 (CGML4)
9	57,58,59,60,61,62	0.1309		25450.2 (CGML5)
10	64,67,70,73,76	4.1831		25446.7 (CGML6)
11	44,45,47,48,50,51,53,54,65,66,68,69 71,72,74,75	0.3967		
			Lamberti and Pappalettere [40]	25446.17 (LESLP)
12	77,78,79,80	0.4416		25446.17 (DOT)
13	81,84,87,90,93	5.1873		
14	95,96,97,98,99,100	0.1912		
15	102,105,108,111,114	6.2410		
16	82,83,85,86,88,89,91,92,103,104,106 107,109,110,112,113	0.6994		
17	115,116,117,118	0.1158		
18	119,122,125,128,131	7.7643		
19	133,134,135,136,137,138	0.1000		
20	140,143,146,149,152	8.8279		
21	120,121,123,124,126,127,129,130,141 142,144,145,147,148,150,151	0.6986		
22	153,154,155,156	1.5563		
23	157,160,163,166,169	10.9806		
24	171,172,173,174,175,176	0.1317		
25	178,181,184,187,190	12.1492		
26	158,159,161,162,164,165,167,168,179 180,182,183,185,186,188,189	1.6373		
27	191,192,193,194	5.0032		
28	195,197,198,200	9.3545		
29	196,199	15.0919		

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

Table 15 gives the best solution vectors and the corresponding weights for Cases 1 and 2, respectively. An optimal structural weight of 19707.77 lb was achieved for Case 1; an optimal weight of 19893.34 lb was also achieved for Case 2 considering both buckling and displacement constraints. Both design procedures obtained each optimum solution after approximately 35,000 searches.

6. Conclusions

The recently developed HS meta-heuristic algorithm was conceptualized using the musical process of searching for a perfect state of harmony. Compared to gradient-based mathematical optimization algorithms, the HS algorithm imposes fewer mathematical requirements to solve optimization problems and does not require initial starting values for the decision variables. The HS algorithm uses a stochastic random search based on the harmony memory considering rate (HMCR) and pitch adjusting rate (PAR), which effectively guide a global search rather than a gradient search, so that derivative information is unnecessary. Furthermore, the HS algorithm generates a new vector after considering all of the existing vectors based on the HMCR and the PAR, rather than considering only two (parents) as in genetic algorithms. These features increase the flexibility of the HS algorithm and produce better solutions.

This paper studied the HS algorithm-based optimization method for structures with continuous sizing variables (fixed geometry). Various truss examples including large-scale trusses under multiple loading

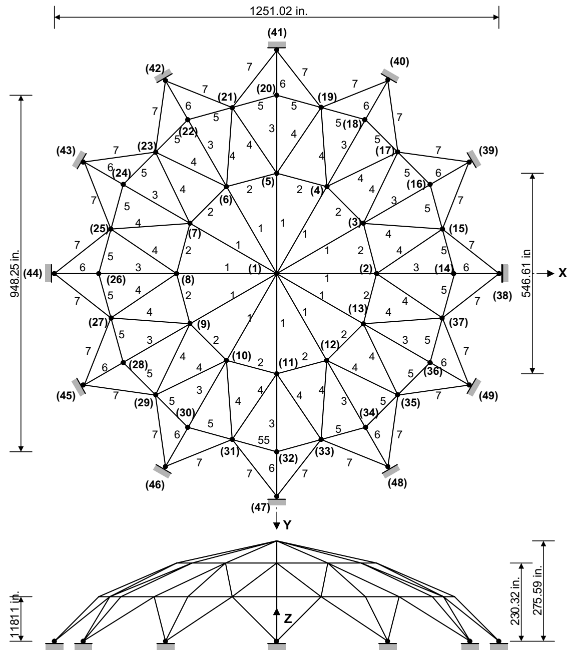


Fig. 13. One-hundred-twenty-bar dome truss.

Table 15
Optimal design for the 120-bar dome truss

Variables (groups)	Optimal cross-sectional areas (in.^{2})	
	Case 1	Case 2
1	3.295	3.296
2	2.396	2.789
3	3.874	3.872
4	2.571	2.570
5	1.150	1.149
6	3.331	3.331
7	2.784	2.781
Weight (lb)	19707.77	19893.34

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

conditions were presented to demonstrate the effectiveness and robustness of the new method. The numerical examples revealed that the proposed HS algorithm-based search strategy was capable of solving size optimization problems. Optimal weights of structures obtained using the proposed HS algorithm approach may yield better solutions than those obtained using conventional mathematical algorithm-based approaches or genetic algorithm-based approaches. The convergence capability results revealed that the proposed HS approach outperformed the simple genetic algorithm-based method, while the fuzzy controlled genetic algorithm methods were better than the HS approach. Note that the HS approach was proposed on the basis of the pure HS algorithm, and is a powerful search and optimization method for solving the continuous sizing variables of the structures compared to the simple genetic algorithm-based method in term of both the obtained optimal solution and the convergence capability.

The new method proposed in this study is not limited to truss structural optimization problems that treat continuous sizing variables. It can be easily used for problems with continuous and/or discrete sizing and configuration variables. Besides trusses, the HS algorithm can be applied to other types of structural optimization problems, including frame structures, plates, and shells.

References

[1] Adeli H, Cheng NT. Integrated genetic algorithm for optimization of space structures. J Aerospace Eng, ASCE 1993;6(4):315–28.

[2] Fogel LJ, Owens AJ, Walsh MJ. Artificial intelligence through simulated evolution. Chichester, UK: John Wiley; 1966.

[3] De Jong K. Analysis of the behavior of a class of genetic adaptive systems. Ph.D. Thesis, Ann Arbor, MI: University of Michigan; 1975.

[4] Koza JR. Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems. Report No. STAN-CS-90-1314, Stanford, CA: Stanford University; 1990.

[5] Holland JH. Adaptation in natural and artificial systems. Ann Arbor, MI: University of Michigan Press; 1975.

[6] Goldberg DE. Genetic algorithms in search optimization and machine learning. Boston, MA: Addision-Wesley; 1989.

[7] Glover F. Heuristic for integer programming using surrogate constraints. Decision Science 1977;8(1):156–66.

[8] Dorigo M, Maniezzo V, Colorni A. The ant system: Optimization by a colony of cooperating agents. IEEE Trans Systems Man Cybernet B 1996;26(1):29–41.

[9] Kirkpatrick S, Gelatt C, Vecchi M. Optimization by simulated annealing. Science 1983;220(4598):671–80.

[10] Rajeev S, Krishnamoorthy CS. Discrete optimization of structures using genetic algorithms. J Struct Eng, ASCE 1992;118(5):1233–50.

[11] Rajeev S, Krishnamoorthy CS. Genetic algorithm-based methodologies for design optimization of trusses. J Struct Eng, ASCE 1997;123(3):350–8.

[12] Koumousis VK, Georgious PG. Genetic algorithms in discrete optimization of steel truss roofs. J Comput Civil Eng, ASCE 1994;8(3):309–25.

[13] Hajela P, Lee E. Genetic algorithms in truss topological optimization. Int J Solids and Structures 1995;32(22):3341–57.

[14] Adeli H, Kumar S. Distributed genetic algorithm for structural optimization. J Aerospace Eng, ASCE 1995;8(3):156–63.

[15] Wu S-J, Chow P-T. Integrated discrete and configuration optimization of trusses using genetic algorithms. Comput & Structures 1995;55(4):695–702.
[16] Wu S-J, Chow P-T. Steady-state genetic algorithms for discrete optimization of trusses. Comput & Structures 1995;56(6):979–91.
[17] Soh CK, Yang J. Fuzzy controlled genetic algorithm search for shape optimization. J Comput Civil Eng, ASCE 1996;10(2):143–50.
[18] Camp C, Pezeshk S, Cao G. Optimized design of two-dimensional structures using a genetic algorithm. J Struct Eng, ASCE 1998;124(5):551–9.
[19] Shrestha SM, Ghaboussi J. Evolution of optimization structural shapes using genetic algorithm. J Struct Eng, ASCE 1998;124(11):1331–8.
[20] Erbatur F, Hasancebi O, Tutuncil I, Kihc H. Optimal design of planar and structures with genetic algorithms. Comput & Structures 2000;75:209–24.
[21] Sarma KC, Adeli H. Fuzzy genetic algorithm for optimization of steel structures. J Struct Eng, ASCE 2000;126(5): 596–604.
[22] Geem ZW, Kim J-H, Loganathan GV. A new heuristic optimization algorithm: harmony search. Simulation 2001;76(2):60–8.
[23] Geem ZW, Kim J-H, Loganathan GV. Harmony search optimization: Application to pipe network design. Int J Modell Simulat 2002;22(2):125–33.
[24] Geem ZW, Tseng C-L. Engineering applications of harmony search. Late-Breaking Papers of Genetic and Evolutionary Computation Conference, New York, NY; 2002. p. 169–73.
[25] Geem ZW, Tseng C-L. New methodology, harmony search and its robustness. Late-Breaking Papers of Genetic and Evolutionary Computation Conference, New York, NY; 2002. p. 174–8.
[26] Paik KR, Jeong JH, Kim JH. Use of a harmony search for optimal design of coffer dam drainage pipes. J Korean Soc Civil Eng 2001;21(2-B):119–28.
[27] Dixon LCW, Szego GP. Towards Global Optimization. Amsterdam: North-Holland; 1975.
[28] Schmit Jr LA, Farshi B. Some approximation concepts for structural synthesis. AIAA J 1974;12(5):692–9.
[29] Schmit Jr LA, Miura H. Approximation concepts for efficient structural synthesis. NASA CR-2552, Washington, DC: NASA; 1976.
[30] Venkayya VB. Design of optimum structures. Comput & Structures 1971;1(1–2):265–309.
[31] Gellatly RA, Berke L. Optimal structural design. AFFDL-TR-70-165, Air Force Flight Dynamics Lab., Wright-Patterson AFB, OH; 1971.
[32] Dobbs MW, Nelson RB. Application of optimality criteria to automated structural design. AIAA J 1976;14(10):1436–43.
[33] Rizzi P. Optimization of multiconstrained structures based on optimality criteria. AIAA/ASME/SAE 17th Structures, Structural Dynamics, and Materials Conference, King of Prussia, PA; 1976.
[34] Khan MR, Willmert KD, Thornton WA. An optimality criterion method for large-scale structures. AIAA J 1979;17(7):753–61.
[35] John KV, Ramakrishnan CV, Sharma KG. Minimum weight design of truss using improved move limit method of sequential linear programming. Comput & Structures 1987;27(5):583–91.
[36] Sunar M, Belegundu AD. Trust region methods for structural optimization using exact second order sensitivity. Int J Numer Methods Eng 1991;32:275–93.
[37] Stander N, Snyman JA, Coster JE. On the robustness and efficiency of the S.A.M. algorithm for structural optimization. Int J Numer Methods Eng 1995;38:119–35.
[38] Xu S, Grandhi RV. Effective two-point function approximation for design optimization. AIAA J 1998;36(12): 2269–75.
[39] Lamberti L, Pappalettere C. Comparison of the numerical efficiency of different sequential linear programming based on algorithms for structural optimization problems. Comput & Structures 2000;76:713–28.
[40] Lamberti L, Pappalettere C. Move limits definition in structural optimization with sequential linear programming—Part II: Numerical examples. Comput & Structures 2003;81:215–38.
[41] Khot NS, Berke L. Structural optimization using optimality criteria methods. In: Atrek E, Gallagher RH, Ragsdell KM, Zienkiewicz OC, editors. New directions in optimum structural design. New York: John Wiley; 1984.
[42] Imai K, Schmit Jr LA. Configuration optimization of trusses. J Structural Division, ASCE 1981;107(ST5):745–56.
[43] Sheu CY, Schmit Jr LA. Minimum weight design of elastic redundant trusses under multiple static load conditions. AIAA J 1972;10(2):155–62.
[44] Templeman AB, Winterbottom SK. Structural design by geometric programming. Second symposium on structural optimization, AGARD Conference, Preprint-123, Milan; 1973.
[45] Chao NH, Fenves SJ, Westerberg AW. Application of reduced quadratic programming technique to optimal structural design. In: Atrex E, Gallagher RH, Radgsdell KM, Zienkiewicz OC, editors. New Directions in Optimum Structural Design. New York: John Wiley; 1984.
[46] Adeli H, Kamal O. Efficient optimization of space trusses. Comput & Structures 1986;24(3):501–11.
[47] Saka MP. Optimum design of pin-jointed steel structures with practical applications. J Struct Eng, ASCE 1990; 116(10):2599–620.
[48] Fadel GM, Clitalay S. Automatic evaluation of move-limits in structural optimization. Struct Optimizat 1993;6:233–7.
[49] Berke L, Khot NS. Use of optimality criteria methods for large-scale systems. AGARD Lecture Series No. 70 on Structural Optimization; 1974; AGARD-LS-70: 1–29.
[50] Xicheng W, Guixu M. A parallel iterative algorithm for structural optimization. Comput Methods Appl Mech Eng 1992;96:25–32.
[51] Adeli H, Park H-S. Neurocomputing for design automation. Boca Raton, FL: CRC Press; 1998.
[52] American Institute of Steel Construction (AISC). Manual of steel construction—allowable stress design. 9th ed. Chicago, IL; 1989.