하모니서치 알고리즘을 이용한 구조물의 새로운 이산최적 설계법

A New Discrete Size Optimization Method for Structures Using Harmony Search Heuristic Algorithm

이 강 석*

*전남대학교 건축학부 교수

1. Introduction

Structural design optimization is a critical and challenging activity that has received considerable attention in the last two decades. Typically, structural optimization problems involve searching for the minimum of a stated objective function, usually the structural weight. This minimum design is subjected to various constraints with respect to performance measures, such as stresses and displacements, and also restricted by practical minimum cross-sectional areas or dimensions of structural members or components. If the design variables can be varied continuously in the optimization, the problem is termed “continuous”; while if the design variables represent a selection from a set of parts, the problem is considered “discrete”.

Traditionally, many gradient-based mathematical programming methods have been developed, and they are frequently used to solve structural optimization problems. The majority of these methods assume that cross-sectional areas, or sizing variables, are continuous. In most practical structural engineering design problems, however, sizes have to be chosen from a list of discrete values due to the availability of components in standard sizes and constraints caused by construction and manufacturing practices. This leads to discrete optimization problems, which are somewhat difficult to solve. Although conventional mathematical methods can consider discreteness by employing round-off techniques based on continuous solutions, the rounded-off solutions may yield results that are far from optimum, or they may even become infeasible as the number of variables increases. Because most available optimization methods treat design variables as continuous, they are inadequate in the presence of discrete design variables. A few methods based on mathematical programming techniques have been developed to handle the discrete nature of design variables[1-4]. They provide useful strategies when solving limited problems, but every method has its drawbacks. These include low efficiency, limited reliability, and becoming trapped at local optimum. A more detailed literature survey of these methods was provided by Templeman[5].

Over the last decade, new optimization strategies based on heuristic algorithms, such as the simulated annealing algorithm and the genetic algorithm (GA), have been devised to obtain optimal designs for discrete structural systems and to overcome the computational drawbacks of conventional mathematical optimization methods. The GA-based discrete optimization methods, in particular, have been studied by many researchers[6-12]. The GA was originally proposed by Holland[13] and further developed by Goldberg[14] and by others. It is a global search algorithm that is based on concepts from natural genetics and the Darwinian survival-of-the-fittest code. Heuristic algorithm-based discrete optimization methods for structures, including GA-based methods, have occasionally overcome several deficiencies of conventional mathematical methods. However, structural engineers are still concerned with seeking a more powerful, effective, and robust method for discrete structural optimization problems.

The main purpose of this article is to introduce an efficient structural optimization method based on the harmony search (HS) heuristic algorithm that treats discrete sizing variables. In the previous research article[15], a new optimization method for structures with continuous pure sizing variables using the HS algorithm was introduced to investigate the applicability of the HS algorithm for structural optimization problems. The effectiveness and robustness of the HS method, compared to GA-based and mathematical optimization methods, were verified using various truss examples, including a 72-bar space truss, a 200-bar planar truss, and a 120-bar dome space truss.

The HS algorithm was originally developed by Geem *et al.*[16], and is based on natural musical performance processes that occur when a musician searches for a better state of harmony, such as during jazz improvisation. Jazz improvisation 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 design variable. Compared to conventional mathematical optimization algorithms, the HS algorithm imposes fewer mathematical requirements to solve optimization problems, and the probability of becoming entrapped in a local optimum is reduced because this algorithm is not a hill-climbing algorithm. Since the HS algorithm uses a stochastic random search, derivative information is unnecessary. This new algorithm also considers several solution vectors simultaneously, in a manner similar to the GA. However, the major difference between the GA and the HS algorithm is that the latter generates a new vector from all the existing vectors, while the former generates a new vector from only two of the existing vectors (parents). In addition, the HS algorithm can consider each component variable in a vector independently when it generates a new vector; the GA cannot, because it has to maintain the gene structure. Although the HS algorithm is a comparatively simple method, it has been successfully applied to various optimization problems including the traveling salesperson problem, the function minimization problems, the layout of pipe networks, pipe capacity design in water supply networks, and optimal pumping problem[16-18].

In this article, a new HS algorithm modified for the discrete size optimization of structural systems is introduced, and a standard benchmark truss example from the literature is also presented to demonstrate its effectiveness and robustness, as compared to current discrete optimization methods.

2. Formulation for Discrete Size Optimization Problems

The discrete size optimization of structural system involves arriving at optimum values for discrete member cross-sectional areas $\boldsymbol{A}$ that minimize an objective function, i.e., the structural weight W. This minimum design also has to satisfy q inequality constraint functions that limit the discrete variable sizes and the structural response. Thus, these size optimization problems can be stated mathematically as minimizing the structural weight as follows:

$$\begin{aligned} &\text{Minimize } W(A) = \gamma \sum_{i=1}^{n} A_i L_i \\ &s.t. \quad G_j^l \le G_j(A) \le G_j^u (j = 1,2,...,q) \end{aligned} \tag{1}$$

where $W(\boldsymbol{A})$ is the objective function, or the structural weight; $\boldsymbol{A}=(A_1, A_2,..., A_n)^T$ is the sizing variable vector, or the cross-sectional areas, which are chosen from a list of available discrete values; γ is the material density for each member; A_i and L_i are the cross-sectional area and length of the ith member, respectively; $G_j(\boldsymbol{A})$ represents the inequality

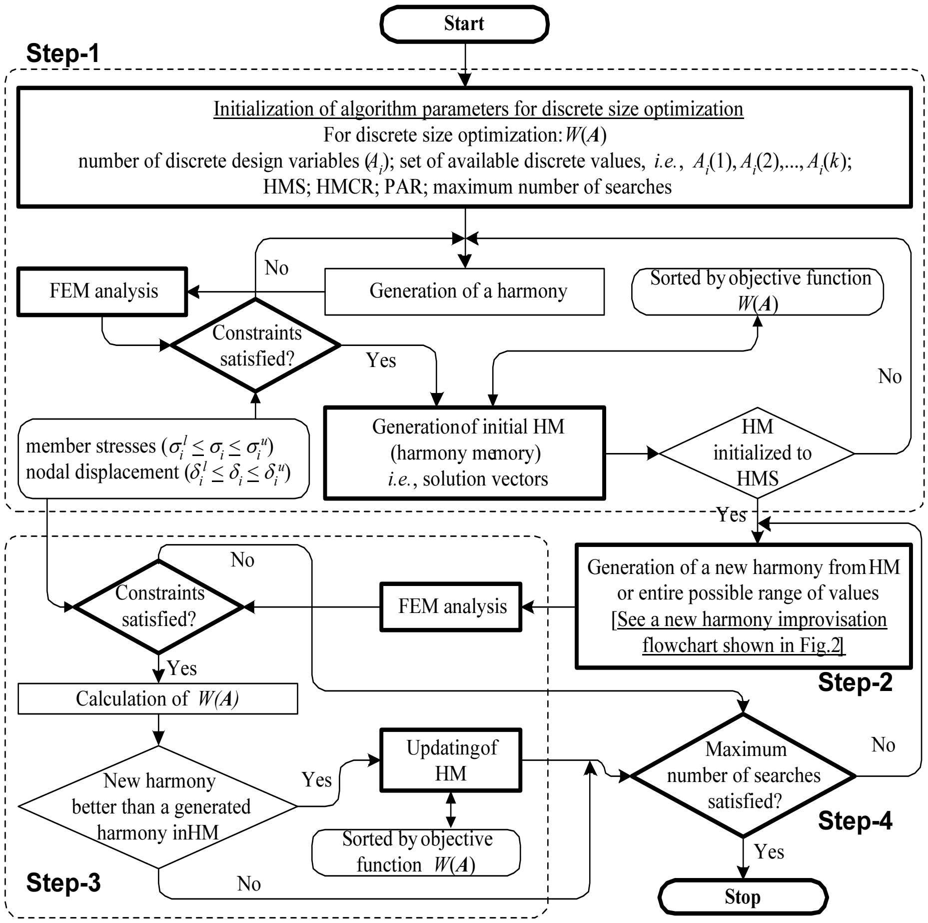


Fig. 1 Design procedure for discrete size optimization using the HS heuristic algorithm

constraints; and G_j^l and G_j^u are the lower and upper bounds on the constraints. For the truss example presented in this article , the lower and the upper bounds on the constraint function included the following: (1) member cross sections ($A_i(k), i=1,...,n$), (2) member stresses ($\sigma_i^l \le \sigma_i \le \sigma_i^u$, $i=1,...,n$), and (3) nodal displacements ($\delta_i^l \le \delta_i \le \delta_i^u$, $i=1,...,m$). Here, σ_i and δ_i are the member stresses and nodal displacements, respectively, calculated from the structural analysis; σ_i^l, σ_i^u, δ_i^l, and δ_i^u are the constraint limitation prescribed for optimization design purposes; and $A_i(k)$ is the available discrete cross-sectional areas, i.e., $A_i(1)$, $A_i(2),\cdots$, $A_i(k)$ ($A_i(1)<A_i(2)<\cdots<A_i(k)$).

3. HS Algorithem-Based Discrete Size Optimization Method

The penalty approach has frequently been employed to determine the fitness measures for the constrained optimization problems, described by Eq. (1), because the optimum solution typically occurs at the boundary between the feasible and infeasible regions[6-12]. However, to demonstrate the pure performance of the HS algorithm-based method proposed in this study, a rejecting strategy for the fitness measure was adopted, i.e., the optimum solution was approached only from the feasible region. Fig.1 shows the design procedure that was used to apply the HS heuristic algorithm to the discrete size optimization problems. The proposed method can be divided into four steps, as follows:

3.1. Step 1: Initialization

The discrete size optimization problem is first specified as $W(A)$ in Eq.(1). The number of discrete design variables (A_i) and the set of a vailable discrete values (D), i.e., $D \in \{A_i(1), A_i(2),\cdots, A_i(k)\}$ ($A_i(1)<A_i(2)<\cdots<A_i(k)$) are then initialized.

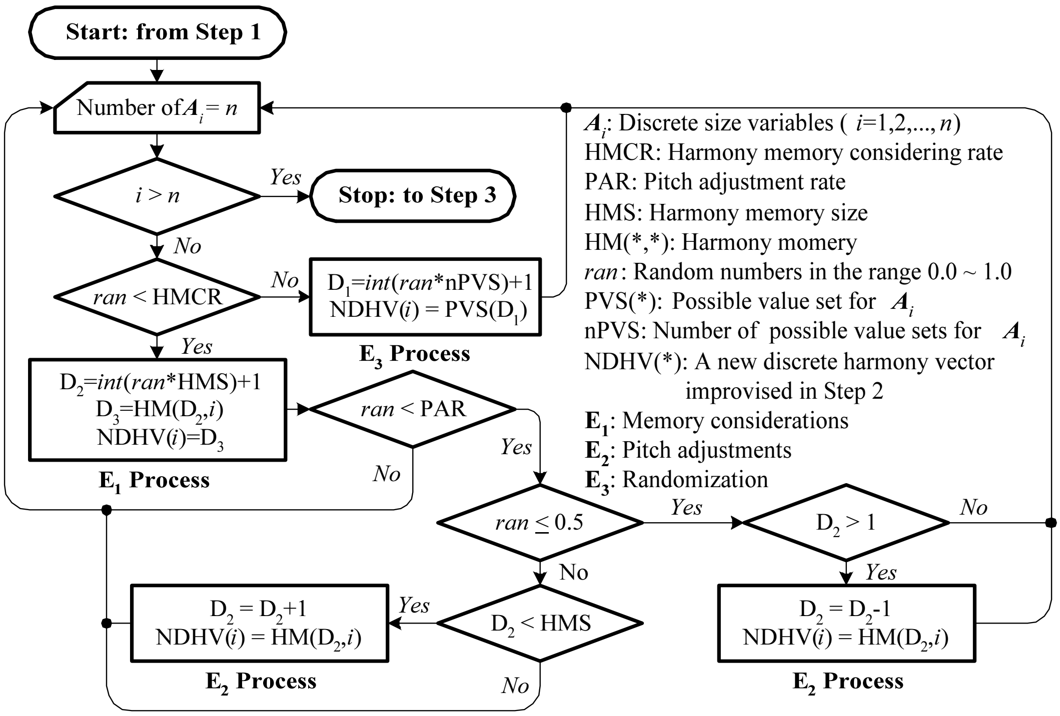


Fig. 2 A new harmony improvisation flowchart for discrete design variables

The HS algorithm parameters that are required to solve the optimization problem are also specified in this step. These include harmony memory size (number of solution vectors in the harmony search, HMS), harmony memory considering rate (HMCR), pitch adjusting rate (PAR), and termination criterion (maximum number of searches). The HMCR and the PAR are parameters that are used to improve the solution vector. Both are defined in Step 2. Subsequently, the “harmony memory” (HM) matrix, shown in Eq. (2), is filled with as many randomly generated solution vectors from the available discrete value set as can be stored in the size of the HM (i.e., HMS). Here, an initial HM is generated based on the FEM structural analysis results, subject to the constraint functions and sorted by the objective function values (Eq. [1]).

$$\boldsymbol{HM} = \begin{bmatrix} A_1^1 & A_2^1 & \cdots & A_n^1 \\ A_1^2 & A_2^2 & \cdots & A_n^2 \\ \vdots & \vdots & \cdots & \vdots \\ A_1^{HMS} & A_2^{HMS} & \cdots & A_n^{HMS} \end{bmatrix} \begin{matrix} \Rightarrow f(\boldsymbol{A}^1) \\ \Rightarrow f(\boldsymbol{A}^2) \\ \Rightarrow \vdots \\ \Rightarrow f(\boldsymbol{A}^{HMS}) \end{matrix} \qquad (2)$$

In Eq. (2), $A^1, A^2, \cdots, A^{HMS}$ and $f(A^1), f(A^2), \cdots, f(A^{HMS})$ are each solution vector and the corresponding objective function value (the structural weight), respectively.

3.2. Step 2: Generation of a new harmony

In the HS algorithm, a new harmony vector, $A' = (A_1', A_2', ..., A_n')$ is improvised from either the initially generated HM or the entire possible range of values. The new harmony improvisation process is based on memory considerations, pitch adjustments, and randomization. In the memory consideration process, the value of the first design variable (A_1') for the new vector is chosen from any value in the specified HM range, i.e., $\{A_1^1, A_1^2, ..., A_1^{HMS}\}$. Values of the other design variables(A_i') are chosen in the same manner. Here, the possibility that a new value will be chosen is indicated by the HMCR parameter, which varies between 0 and 1 as follows:

$$A_i' \leftarrow \begin{cases} A_i' \in \{A_i^1, A_i^2, \ldots, A_i^{HMS}\} & w.p. \;\; HMCR \\ A_i' \in \{A_i(1), A_i(2), ..., A_i(k)\} & w.p. \;\; (1\text{-}HMCR) \end{cases} \qquad (3)$$

The HMCR sets the rate of choosing a value from the historic values stored in the HM, and (1-HMCR) sets the rate of randomly choosing a value from the entire possible range of values (randomization process). For example, a HMCR of 0.85 indicates that the HS algorithm will choose the design variable value from historically stored values in the HM with a 85% probability, and from the entire possible range of values with a 15% 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. Every component of the new harmony vector, $A' = (A_1', A_2', ..., A_n')$, is examined to determine whether it should be pitch-adjusted (pitch adjustment process). This procedure uses the PAR parameter that sets the rate of adjustment for the pitch chosen from the HM asfollows:

Pitch adjusting design for

$$A_i' \leftarrow \begin{cases} Yes & w.p. \quad PAR \\ No & w.p. \quad (1-PAR) \end{cases} \tag{4}$$

The pitch adjusting process is performed only after a value has been chosen from the HM. The value (1-PAR) sets the rate of doing nothing. A PAR of 0.3 indicates that the algorithm will choose a neighboring value with 30%×HMCR probability. If the pitch adjustment decision for A_i' is Yes, and A_i' is assumed to be $A_i(l)$, i.e., the l-th element in $\{A_i(1),\ A_i(2), \cdots,\ A_i(l), \cdots,\ A_i(k-1),\ A_i(k)\}$, then the pitch-adjusted value of $A_i(l)$ is

$$A_i' \Leftarrow A_i(l+c) \tag{5}$$

where c=the neighboring index, c∈{-1,1}. A detailed flowchart for the new harmony discrete search strategy based on the HS heuristic algorithm is given in Fig.2. Note tha the HMCR and PAR parameters introduced in the harmony search help the algorithm find globally and locally improved solutions, respectively.

3.3. Step 3: Fitness measure and HM update

The new harmony improvised in Step 2 is analyzed using a FEM structural analysis method, and its fitness is determined using a rejection strategy based on the constrained function. If the new harmony vector is better than the worst harmony vector 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.

3.4. Step 4: Repeat Steps 3 and 4 until the termination criterion is satisfied

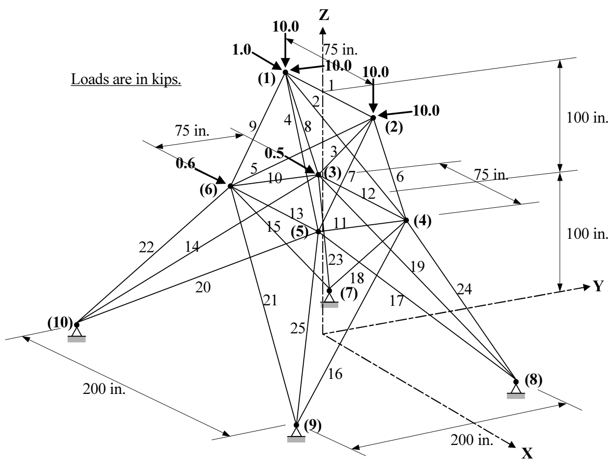
The computations terminate when the termination criterion is satisfied. If not, Steps 2 and 3 are repeated.

4. Example: 25-Bar Transmission Tower Space Truss

A 25-bar transmission tower space truss, which has been used as a benchmark to verify the efficiency of various discrete size optimization methods, was considered in this study using a FORTRAN computer program. The FEM displacement method was used to analyze the space truss. To demonstrate the discrete search efficiency of the HS algorithm-based method compared to current methods, the five cases shown in Table 1, each with a different set of HS algorithm parameters (i.e., HMS, HMCR, and PAR), were tested for the space truss example. These parameter values were arbitrarily selected, based on the empirical findings by Geem *et al.*[16-18], which determined that the HS algorithm performed well with values ranging between 10 and 50 for

Table 1 HS algorithm parameters used for the example

Cases	HMS	HMCR	PAR
Case-1	20	0.9	0.45
Case-2	40	0.9	0.45
Case-3	30	0.9	0.4
Case-4	30	0.8	0.3
Case-5	30	0.9	0.3



(Note: 1 in. = 2.54 cm.)

Fig. 3 25-bar space truss

Table 2 Optimal result for 25-bar space truss

Design variables A_i (in.^{2})	HS results					Rajeev et al.[6]	Wu & Chow[8]	Wu & Chow[9]	Eabatur et al.[12]	Adeli & Park[19]	Park & Sung[20]
	Case-1	Case-2	Case-3	Case-4	Case-5						
A1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.6	0.1
A2~A5	0.6	0.3	0.3	0.5	0.3	1.8	0.6	0.5	1.2	1.4	2.1
A6~A9	3.4	3.4	3.4	3.4	3.4	2.3	3.2	3.4	3.2	2.8	3.4
A10~A11	0.1	0.1	0.1	0.1	0.1	0.2	0.2	0.1	0.1	0.5	0.1
A12~A13	1.6	2.1	2.1	1.9	2.1	0.1	1.5	1.5	1.1	0.6	2.2
A14~A17	1.0	1.0	1.0	0.9	1.0	0.8	1.0	0.9	0.9	0.5	1.1
A18~A21	0.4	0.5	0.5	0.5	0.5	1.8	0.6	0.6	0.4	1.5	1.0
A22~A25	3.4	3.4	3.4	3.4	3.4	3.0	3.4	3.4	3.4	3.0	3.0
Weight (lb)	485.77 [521.04][a]	484.85 [504.72][a]	484.85 [514.20][a]	485.05 [514.21][a]	484.85 [504.28][a]	546.01	491.72	486.29	493.80	543.95	537.23
Num. of structural analyses	13,736 [13,445][b]	14,163 [4,414][b]	13,523 [2,160][b]	17,159 [5,226][b]	18,734 [6,850][b]	600	-	40,000	-	-	-

a denotes the HS optimal results obtained after 600 analyses (the result of Rajeev *et al.*[6)).
b denotes number of analyses for the HS required to obtain a weight of 486.29lb(the result of Wu and Chow[9)).
Note: 1in.^{2}=6.452cm^2, 1lb=4.448N.

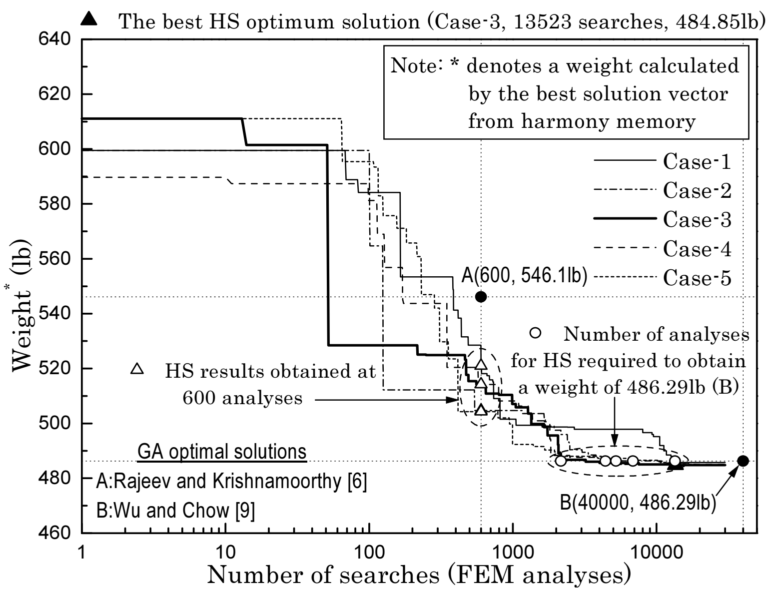


Fig. 4 Convergence history of the minimum weight for 25-bar space truss

the HMS, 0.7 and 0.95 for the HMCR, and 0.2 and 0.5 for the PAR. The maximum number of searches was set to 30,000. The 25-bar transmission tower space truss, shown in Fig. 3, has been optimized using discrete size algorithms by many researchers[6,8,9,12,19,20]. In these studies, the material density was 0.1lb/in.^{3} (27.14kN/m^3) and modulus of elasticity was 10,000ksi (68.95GPa). This space truss was subjected to the following loading condition: P_X=1.0kips (4.45kN) and P_Y=P_Z=-10.0kips (-44.48kN) acting on node1, P_X=0.0 and P_Y=P_Z=-10.0kips (-44.48kN) acting on node2, P_X=0.5kips (2.22kN) and P_Y=P_Z=0.0 acting on node3, and P_X=0.6kips (2.67kN) and P_Y=P_Z=0.0 acting on node6. 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~A_5, (3)A_6~A_9, (4)A_{10}~A_{11}, (5)A_{12}~A_{13}, (6)A_{14}~A_{17}, (7)A_{18}~A_{21}, and (8)A_{22}~A_{25}. All members were constrained to 40ksi (275.6 MPa) in both tension and compression. Inaddition, maximum displacement limitations of±0.35in. (±0.889cm) were imposed at each node in every direction. Discrete values for the cross-sectional areas were taken from the set D∈{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.8, 3.0, 3.2, 3.4}(in.^{2}), which has thirty discrete values.

The proposed discrete size optimization method based on the HS algorithm was applied to the space truss. Table 2 lists the HS result obtained with each set of parameters given in Table 1. The results reported by Rajeev and Krishnamoorthy[6], Wu and Chow[8,9], and Eabaturetal.[12], obtained with GA-based methods, by Adeli and Park[19], obtained with the neural dynamics model, and by Park and Sung[20], obtained with the simulated annealing-based method, are also included in the table. After 13,523 to 18,734 searches (FEM structural analyses), the best solution vector and the corresponding objective function value (the structural weight) were obtained for all five HS cases (see Table 2). All of the HS results were better than the values obtained by the previous investigations.

Fig. 4 shows a comparison of the convergence capability of each HS case and the GA-based approaches. While the pure GA proposed by Rajeev and Krishnamoorthy[6] obtained a minimum weight of 546.01lb (2428.65N) after 600 structural analyses, the HS cases obtained minimum weights of 504.28 lb (2243.15N) to 521.04lb (2317.7N) after the same number of analyses. The steady-state GA proposed by Wu and Chow[9] obtained a minimum weight of 486.29lb (2163.13N) after 40,000 analyses, while all HS cases except Case1 obtained the same weight after only 2,160 to 6,850 analyses. The HS approach therefore outperformed the pure and steady-state GA-based methods, in terms of both the obtained optimal value and the convergence capability.

5. Conclusions

A new discrete size optimization method for structures using the HS algorithm was proposed to calculate a minimum weight. A standard benchmark truss example was also presented to demonstrate the effectiveness and robustness of the proposed method. The results were compared to those obtained using current discrete optimization methods, especially GA-based techniques. The illustrative example revealed that the HS optimal results were better than those obtained from all previous investigations. Also, the convergence capability of the proposed HS method outperformed that of the GA-based methods. In conclusion, our study suggests that the new HS based method is a potential powerful search and optimization technique for solving structural optimization problems with discrete sizing variables.

The recently developed HS heuristic algorithm is simple and mathematically less complex than the GA. The HS algorithm generates a new vector based on the harmony memory considering rate and the pitch adjusting rate after considering all of the existing vectors, while the GA generates a new vector from only two of the exiting vectors (parents). These features increase the flexibility of the HS algorithm and allow it to find better solutions. Furthermore, the HS algorithm adopted a parameter-setting-free adaptive feature, enabling the algorithm users not to perform tedious parameter setting process[21,22]. The HS algorithm-based method proposed in this study is not limited to truss structural optimization problems. Besides trusses, the HS algorithm can also be applied to other types of structural optimization problems, including frame structures, plates, and shells.

ACKNOWLEDGMENTS

I wish to thank Dr. Zong Woo Geem for introducing me to the harmony search (HS) heuristic algorithm and for many insightful suggestions and stimulating discussions about the HS and its application.

REFERENCES

1. Liebman J. S., Khachaturian N., Chanaratna V. Discrete structural optimization. J. Struct. Div. (ASCE) 1981; 107 (ST11): 2177-2197.
2. Hua H. M. Optimization for structures of discrete-size elements. Comp. Struct. 1983;17(3):327-333.
3. Zhu D. E. An improved templeman's algorithm for optimum design of trusses with discrete member sizes. Engrg. Opt. 1986;9:303-312.
4. John K. V., Ramakrishnan C. V. Minimum weight design of trusses using improved move limit method of sequential linear programming. Comp. Struct. 1987; 27(5): 583-591.
5. Templeman A. B. Discrete optimum structural design. Comp. Struct. 1988;30(3):511-518.
6. Rajeev S., Krishnamoorthy C. S. Discrete optimization of

structures using genetic algorithms. J. Struct. Engrg. (ASCE) 1992; 118(5): 1233-1250.
7. Rajeev S., Krishnamoorthy C. S. Genetic algorithm-based methodologies for design optimization of trusses. J. Struct. Engrg. (ASCE)1997; 123(3): 350-358.
8. Wu S. -j., Chow P. -T. Integrated discrete and configuration optimization of trusses using genetic algorithms. Comp. Struct. 1995; 55(4): 695-702.
9. Wu S. -j., Chow P. -T. Steady-state genetic algorithms for discrete optimization of trusses. Comp. Struct. 1995; 56(6): 979-991.
10. Camp C., Pezeshk S., Cao G. Optimized design of two-dimensional structures using a genetic algorithm. J. Struct. Engrg. (ASCE)1998; 124(5): 551-559.
11. Pezeshk S., Camp C. V., Chen D. Design of nonlinear framed structures using genetic optimization. J. Struct. Engrg. (ASCE)2000; 126(3): 382-388.
12. Erbatur F., Hasancebi O., Tutuncil I., Kihc H. Optimal design of planar and space structures with genetic algorithms. Comp. Struct.2000;75:209-224.
13. Holland J. H. Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor, MI,1975.
14. Goldberg D. E. Genetic algorithm in search optimization and machine learning. Addision Wesley, Boston, MA, 1989.
15. Lee K. S., Geem, Z. W. A new structural optimization method based on the harmony search algorithm. Comp. Struct. 2004;82:781-798.
16. Geem Z. W., Kim J. -H., Loganathan G. V. A new heuristic optimization algorithm: harmony search. Simulation 2001; 76(2):60-68.
17. Geem Z. W., Kim J. -H., Loganathan G. V. Harmony search optimization: Application to pipe network design. Int. J. Model. Simulation 2002; 22(2): 125-133.
18. Geem Z. W., Tseng C. -L. New methodology, harmony search and its robustness. Late-Breaking Paper of Genetic and Evolutionary Computation Confrence 2002, New York, NY, 174-178.
19. Adeli H., Park H. S. Hybrid CPN-neural dynamics model for discrete optimization of steel structures. Microcomp. Civ. Engrg. 1996; 11(5): 355-366.
20. Park H. S., Sung C. W. Optimization of steel structures using distributed simulated annealing algorithm on a cluster of personal computers. Comp. Struct. 2002; 80: 1305-1316.
28. Xicheng, W., Guixu, M. A parallel iterative algorithm for structural optimization, Comp. Methods in Applied Mechanics and Eng. 1992; 96: 25-32.
29. Geem, Z. W. and Sim, K. -B. Parameter-Setting-Free Harmony Search Algorithm, Applied Mathematics and Computation, 2010; 217(8): 3881-3889.

[담당 : 황재승 편집위원]