

Harmony Search 알고리즘을 이용한 입체트러스의 단면최적화 Size Optimization of Space Trusses Based on the Harmony Search Heuristic Algorithm

이 강 석*
Lee, Kang Seok

김 정 희**
Kim, Jeong-Hee

최창식***
Choi, Chang-Sik

이 리 형****
Lee, Li-Hyung

ABSTRACT

Most engineering optimization are based on numerical linear and nonlinear programming methods that require substantial gradient information and usually seek to improve the solution in the neighborhood of a starting point. These algorithm, however, reveal a limited approach to complicated real-world optimization problems. If there is more than one local optimum in the problem, the result may depend on the selection of an initial point, and the obtained optimal solution may not necessarily be the global optimum. This paper describes a new harmony search(HS) meta-heuristic algorithm-based approach for structural size optimization problems with continuous design variables. This recently developed HS algorithm is conceptualized using the musical process of searching for a perfect state of harmony. It uses a stochastic random search instead of a gradient search so that derivative information is unnecessary. Two classical space truss optimization problems are presented to demonstrate the effectiveness and robustness of the HS algorithm. The results indicate that the proposed approach is a powerful search and optimization technique that may yield better solutions to structural engineering problems than those obtained using current algorithms.

1. 서 론

지난 40년 동안, 대부분의 최적설계용 알고리즘은 다양한 공학문제의 해결에 있어 큰 기여를 하였다. 이들 알고리즘의 대부분은 복잡한 도함수와 초기조건을 요구하는 수학적 선형(Linear Programing), 비선형 계획법(Nonlinear Programing)을 기반으로 하고 있다. 수학적 최적화 알고리즘은 단순 최적화문제에서는 전역적 최적값(Global Optimum)을 얻는데 유용한 방법을 제공하지만, 사실상 공학적 최적화 문제들은 매우 복잡 또는 복잡적이어서 수학적 알고리즘을 사용하여 해결하는데에는 한계가 있다.

현존 수학적 방법의 계산상의 결점을 극복하기 위하여 최근 시뮬레이션에 기반을 둔 자기학습형 알고리즘(Heuristic Algorithms), 즉 GA, SA, Tabu Search 등의 연구가 활발히 수행되고 있다. 일반적으로 자기학습형 알고리즘은 일정한 규칙과 생물학적 진화과정, 동물적 습성, 물리적 가열 냉각 과정 등의 자연현상을 모방하고 있으며, 지난 10년 동안 자기학습형 알고리즘, 특히 유전자 알고리즘은 많은 연구자들에 의해 다양한 구조물 최

* 정회원 · 한양대학교 초대형구조시스템연구센터 연구조교수, 공학박사

** 학생회원 · 한양대학교 대학원 건축공학과 석사과정

*** 정회원 · 한양대학교 건축공학과 부교수, 공학박사

**** 정회원 · 한양대학교 건축공학과 교수, 공학박사

적화 문제의 해결에 이용되었다. 유전자 알고리즘을 포함한 자기학습형 알고리즘을 이용한 구조물의 최적화 방법은 기존의 수학적 방법의 결함을 점차 극복하고 있지만, 고도의 복잡한 구조물의 최적화문제에서는 여전히 보다 효과적인 자기학습형 알고리즘이 요구되고 있다.

한편, 완벽한 하모니를 창출한다는 음악적 개념을 도입시켜 개념화한 Harmony Search Heuristic Algorithm (이하, HSHA)[1]이 최근 개발되었다. HSHA는 음악에서의 하모니는 최적의 해벡터와 같고, 음악가들의 연주는 최적화기법에서 국부적 또는 전역적 탐색방법과 같다는 개념에서 개발되었으며, 기존의 자기학습형 알고리즘과 비교해보면 HSHA는 단순한 수학적 조건이 요구되며, 다양한 유형의 구조물의 최적화 문제에 보다 쉽게 이용할 수 있다. 그리고 HSHA는 변수의 초기조건을 요구하는 경사도 탐색법과 달리 HSHA는 하모니 메모리 기여율(Harmony Memory Considering Rate)과 피치 조정률(Pitch Adjusting Rate)을 연산자로 한 확률론적 임의의 탐색방법을 사용한다.

본 연구에서는 연속설계변수의 구조물 최적화 문제를 위해 HSHA를 이용한 방법을 제안한다. 함수의 최적화 문제를 통해서 HSHA의 최적화과정 및 개념을 소개함과 동시에, 2개의 표준 입체트러스 최적화 문제를 통해 HSHA의 유용성을 검증한다.

2. Harmony search heuristic algorithm

HSHA는 완벽한 하모니를 창출한다는 음악적 개념을 도입시켜 개발된 알고리즘으로서, 하모니의 보다 좋은 상태를 찾으려고 하는 재즈연주자의 연주과정을 바탕으로 개발되었다[1]. 음악공인이 심미적 표준에 의해서 보다 좋은 하모니 상태를 찾으려고 시도하는 것과 같이 최적화 과정은 목적함수로 결정된 전역 최적값을 찾으려고 시도한다. 또한 각 악기의 높낮이는 심미적 음질을 결정하는 것처럼 각 변수의 열은 목적함수의 값을 결정한다.

그림 1에 HSHA를 이용한 최적화 과정(1-5단계)을 나타낸다.

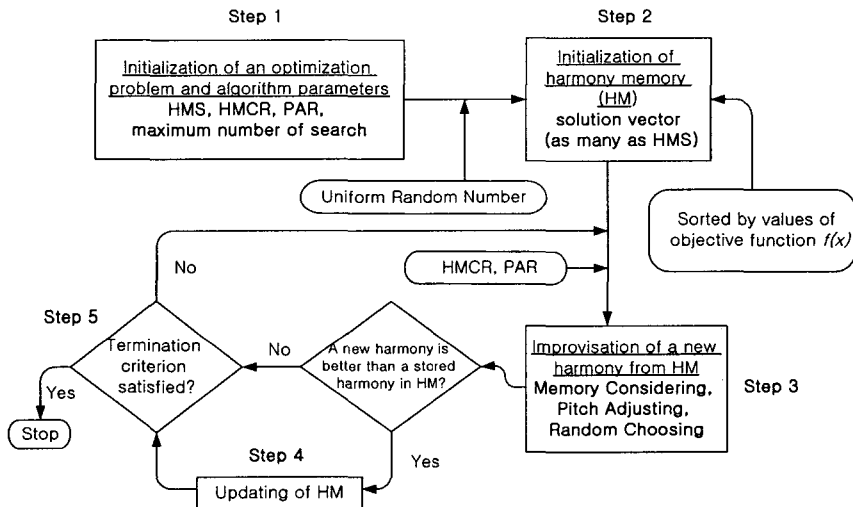


그림 1. Harmony Search Heuristic Algorithm의 최적화과정(step 1-5)

2.1. HSHA를 이용한 최적화 과정

1 단계: 최적화 문제와 알고리즘 매개변수의 초기화
일반적으로 최적화 문제는 식(1)에 의하여 정의된다.

$$\text{Minimize } f(x) \text{ s.t. } x_i \in X_i, \quad i=1,2,L,N \quad (1)$$

여기서, $f(x)$: 목적함수

x : 설계변수

X : 각 설계변수에 대한 가능한 변수의 집합($l x_i \leq X_i \leq u x_i$)

N : 설계변수의 개수

식(1)의 최적화문제의 해결에 요구되는 HSHA의 매개변수값도 1단계에서 설정되며, 그 매개변수는 아래와 같이 정의된다.

- HMS(Harmony Memory Size): HM에서 해벡터의 수
- HMCR(Harmony Memory Considering Rate): 하모니메모리 기여율
- PAR(Pitch Adjusting Rate): 피치 조정률
- 종료기준: 최대 탐색수

HMCR과 PAR은 HSHA의 핵심 연산자로서 해벡터(Solution Vector)가 국부최적에 빠지지 않고 전역 최적값에 도달하도록 유도하는 매개변수이다. HMCR 및 PAR의 정의는 3단계(Step 3)에서 규정된다.

2단계: HM의 초기화

식(2)의 HM 행렬은 어떤 최적화문제에서 임의로 선출된 해벡터를 채우고 이 값을 목적함수의 값에 의해 정렬한다.

$$HM = \begin{bmatrix} x^1 \\ x^2 \\ \vdots \\ M \\ x^{HMS} \end{bmatrix} \quad (2)$$

3단계: HM으로부터 새로운 하모니를 산출

이 단계에서 새로운 하모니 벡터, $x' = (x'_1, x'_2, \dots, x'_N)$ 는 식(3)과 같이 하모니메모리 기여율(HMCR), 피치 조정률(PAR) 및 랜덤 선출에 의하여 HM으로부터 선출된다.

$$x'_i \leftarrow \begin{cases} x'_i \in \{x_i^1, x_i^2, \dots, x_i^{HMS}\} & \text{with probability HMCR} \\ x'_i \in X_i & \text{with probability } (1 - HMCR) \end{cases} \quad (3)$$

예를 들면, 첫 번째 새로운 설계변수의 값(x'_1)은 식(3)의 조건으로 지정된 HM열 $x' = (x'_1, x'_2, \dots, x'_N)$ 로부터 어떤 임의값을 선택한다. 다른 설계변수의 값(x'_i) 또한 같은 방법으로 구할 수 있다. 여기서, HMCR변수값의 범위는 0~1 사이이며, 선정된 HMCR값은 HM에 저장된 값으로부터 한 값을 선택할 확률이다. (1-HMCR)은 HM에 저장된 것에 제한되지 않고 가능한 값을 임의로 선택할 확률이다. 예를 들면, 0.95의 HMCR은 95%의 확률로 HM에 저장된 값으로부터 설계변수 값을 선택하며, 나머지 5%의 확률로 전체의 가능한 열로부터 설계변수 값을 선출한다. 여기서 HM에 저장되지 않은 값으로부터 해가 선출될 확률이 있기 때문에 1.0의 HMCR은 일반적으로 사용하지 않는다[1]. 이것은 유전자 알고리즘의 돌연변이 선택과정과 유사하다.

새로운 하모니벡터, $x' = (x'_1, x'_2, \dots, x'_N)$ 의 구성요소는 다음 식(4)의 피치 조정률(PAR)에 의해 결정된다.

$$Pitch \text{ adjusting decision for } x_i \leftarrow \begin{cases} Yes & \text{with probability } PAR \\ No & \text{with probability } (1 - PAR) \end{cases} \quad (4)$$

피치조정과정은 HM으로부터 선택된 한 개의 값에 의하여 실행되며, (1-PAR)은 아무런 역할을 수행하지 않는다. 예를 들면, 0.3의 PAR은 $30\% \times HMCR$ 의 확률에 의해 선택되며, 피치조정과정은 다음의 식(5)로 정식화할 수 있다.

$$x'_i \leftarrow x'_i + \alpha \quad (5)$$

여기서, α 는 $bw \times u(-1, 1)$, bw 는 연속 설계변수의 임의의 밴드폭, $u(-1, 1)$ 는 -1과 1사이의 균일 랜덤분포

4단계: HM의 개선

목적함수의 값으로 정렬된 HM에서 새로운 하모니벡터가 기존의 하모니 값보다 더 좋다면, 이 값은 HM에 포함시키고 기존의 가장 안 좋은 값을 배제시킨다. 또한 위의 값을 목적함수의 값에 의해 HM에 정렬시킨다.

5단계: 3단계와 4단계를 종료기준을 만족할 때 까지 반복

2.2. HSHA를 이용한 함수 최적화 예

HSHA의 최적화과정 및 개념이해를 위해서 함수 최적화 문제에 있어 표준적인 함수인 Six-Hump Camelback 함수를 이용하여 최적화 예를 나타낸다.

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

식(6)에 나타난 Six-Hump Camelback 함수는 6개의 지역최적치와 2개의 전역최적치, 즉 $x^* = (-0.08984, 0.71266)$, $x^* = (0.08984, -0.71266)$ 과 이에 대응하는 함수값 $f^*(x) = -1.0316285$ 로 이루어져 있으며 현행의 수학적 최적알고리즘을 사용하면 그 최적값은 전술한 바와 같이 초기조건의 선택에 크게 의존한다. 반면 확률론적 랜덤 탐색방법인 HSHA는 수학적 알고리즘과 다른 방법을 사용하여 최적해 벡터를 찾아냈다. HSHA방법에서 식(6)의 두개의 변수 x_1 과 x_2 의 설계변수의 범위는 $-10.0 \sim 10.0$ 으로 설정하였으며, 전체 해 벡터의 수 HMS= 10, 하모니메모리 기여율 HMCR= 0.85, 피치 조정률 PAR= 0.45로 각각 설정하였다.(1단계). 표 1에 나타내는 바와 같이, HM은 지정된 범위 내의 해벡터를 임의로 선출한 것을 초기에 구성하고, 이것을 목적함수의 값에 의해 정렬한다(2단계). 새로운 하모니벡터, $x'_i = (3.183, 8.666)$ 은 전술한 바와같이 3가지 규칙에 의해 확률적으로 선출된다(2.1. 정의 3단계 참조). 즉, 46.75% ($0.85 \times 0.55 = 0.4675$)의 하모니메모리 기여율(HMCR) 및 38.25% ($0.85 \times 0.45 = 0.3825$)의 피치 조정률(PAR)에 의해 선출하고, 나머지 15% ($1 - 0.85 = 0.15$)의 확률로 랜덤 선출한다(3단계). 표 1의 중간부분에 나타난 바와 같이 위의 규칙으로 얻어진 새로운 하모니(3.183, 8.666)의 목적함수값은 22454.67이므로 이 값은 HM에 포함되며, 기존의 (-9.5, 3.333)은 HM으로부터 배제된다(4단계). 탐색수(50, 100, 1000번) 증가에 따라 하모니벡터값은 전역최적값에 근접하게 되며, 최종적으로 4870번에 전역최적값과 아주 유사한 값 -1.0316285를 유도했으며, 이때의 변수값 즉, $x_1 = 0.08984$, $x_2 = -0.71269$ 이다

표 1. HS 알고리즘을 이용한 Six-Hump Camelback function의 최적결과 값

Rank	x_1	x_2	$f(x)$	x_1	x_2	$f(x)$	x_1	x_2	$f(x)$
	Initial HM			Subsequent HM			HM after 50 searches		
1	3.183	-0.400	169.95	3.183	-0.400	169.95	0.80558	-0.400	0.9427
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	46694.7	6.6667	7.433	37334.24	0.81676	2.419	117.35
5	-7.583	5.567	60352.77	6.767	8.317	46694.7	-0.88333	2.561	145.66
6	7.767	4.700	67662.4	-7.583	5.567	60352.77	-0.88333	2.589	152.54
7	8.250	2.750	95865.2	7.767	4.700	67662.4	3.074	-1.833	157.57
8	-8.300	8.533	120137.09	8.250	2.750	95865.2	3.183	-0.400	169.95
9	-9.017	-8.050	182180	-8.300	8.533	120137.09	3.183	-1.755	191.78
10	-9.500	3.333	228704.72	-9.017	-8.050	182180	3.183	3.183	271.38
Rank	HM after 100 searches			HM after 1000 searches			HM after 4870 searches		
1	0.31672	0.40000	-0.2838402	0.09	-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	0.09	-0.71269	-1.0316284
3	0.26504	0.32581	-0.1951466	0.08863	-0.71143	-1.0316119	0.09	-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	0.09	-0.71446	-1.031602	0.08951	-0.71269	-1.031628
6	0.26504	0.22232	0.238899	0.09028	-0.71446	-1.0316019	0.08951	-0.71277	-1.0316279
7	0.35011	0.28268	0.581726	0.09081	-0.71446	-1.0316	0.08951	-0.71279	-1.0316278
8	0.31883	0.25029	0.0705802	0.09	-0.71062	-1.0315942	0.09028	-0.71269	-1.0316277
9	0.35011	0.23078	0.1768801	0.08863	-0.71446	-1.3015939	0.0898	-0.713	-1.0316275
10	0.54693	0.28268	0.5600001	0.09028	-0.71062	-1.0315928	0.09	-0.713	-1.0316274

^a 처음 탐색으로 산출된 새로운 하모니와 그에 상응하는 함수 값

^b HSHA를 이용하여 얻어진 최적의 해벡터

3. 입체트러스의 최적화

3.1. 25부재 입체 트러스

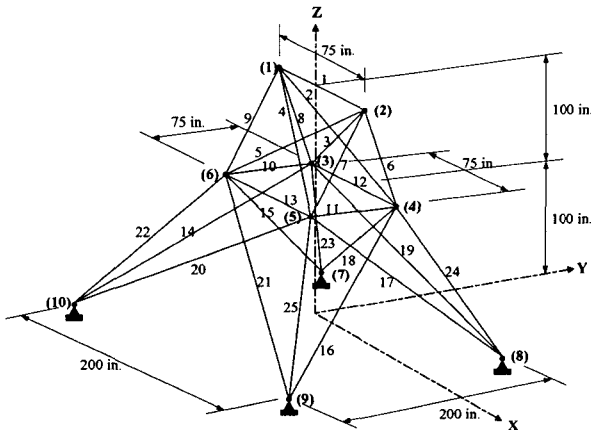


그림 2. 25부재 입체 트러스

표 2. 25부재 입체 트러스의 하중조건

절점 번호	하중조건 1(kips)			하중조건 2(kips)		
	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

그림2의 25부재 입체 트러스는 Schmit와 Farshi[2], Schmit와 Miura[3], Venkayya[4], Gellatly와 Berke[5], Rizzi[6], Templeman과 Winterbottom[7], Adeli and Kamal[8] 등 많은 연구자들에 의해 단면 최적화되었다. 25

부재 입체트러스의 재료비중은 0.1 lb/in.³, 탄성계수는 10,000 ksi이며, 또한 표2에 나타낸 바와 같이 2개의 하중을 조건으로 하며, x와 y축에 대하여 대칭이다. 전체부재는 8그룹 즉, (1)A₁, (2)A₂~A₅, (3)A₆~A₉, (4)A₁₀~A₁₁, (5)A₁₂~A₁₃, (6)A₁₄~A₁₇ (7)A₁₈~A₂₁ 및 (8)A₂₂~A₂₅으로 나누어지며, 따라서 단면변수는 8개이다.

이 트러스 부재는 표3에서 나타내는 것처럼 압축응력과 인장응력을 제약조건으로하며, ±0.35의 최대 변위제약조건을 설정하였으며, 이 변위제약은 모든 방향의 절점에 부과된다. 모든 부재의 최소 단면적은 0.01in.²이다.

한편, HSHA의 변수 즉, HMS=20, HMCR=0.8 및 PAR=0.3으로 설정하였으며, 탐색수는 15,000번으로 설정하였다. 25부재 입체트러스에 부과된 각 제약조건을 바탕으로 15,000번 탐색후 얻어진 8개의 설계변수에 대한 HS 알고리즘의 최적해 벡터는 (0.047, 2.022, 2.95, 0.01, 0.014, 0.688, 1.657, 2.663)이며, 이에 대응하는 최적 중량은 544.38lb이다. 표4에는 HSHA의 최적값을 포함한 기존연구의 최적값을 비교하여 나타내었다. HSHA는 구조물 단면 최적화에 유효하게 적용 가능하였으며, 기존의 수학적 알고리즘을 이용한 최적설계결과 및 GA등의 연구결과 보다 좋은 값을 얻었다.

표 3. 부재의 허용응력

변수	허용압축응력(ksi)	허용인장응력(ksi)	
1	A ₁	35.092	40.0
2	A ₂ ~A ₅	11.590	40.0
3	A ₆ ~A ₉	17.305	40.0
4	A ₁₀ ~A ₁₁	35.092	40.0
5	A ₁₂ ~A ₁₃	35.092	40.0
6	A ₁₄ ~A ₁₇	6.759	40.0
7	A ₁₈ ~A ₂₁	6.959	40.0
8	A ₂₂ ~A ₂₅	11.082	40.0

표 4. 25부재 입체트러스의 최적 결과 값 비교

변수	최적 단면(in. ²)									
	Schmit and Farshi	Schmit and Miura		Venkayya	Gellatly and Berke	Rizzi	Templeman and Winterbottom	Adeli and Kamal	This work	
		NEWSUMT	CONMIN							
1	A ₁	0.010	0.010	0.166	0.010	0.010	0.010	0.010	1.986	0.047
2	A ₂ ~A ₅	1.964	1.985	2.017	1.988	2.007	1.988	2.022	2.961	2.022
3	A ₆ ~A ₉	3.033	2.996	3.026	2.991	2.963	2.991	2.938	0.010	2.950
4	A ₁₀ ~A ₁₁	0.010	0.010	0.087	0.010	0.010	0.010	0.010	0.010	0.010
5	A ₁₂ ~A ₁₃	0.010	0.010	0.097	0.010	0.010	0.010	0.010	0.806	0.014
6	A ₁₄ ~A ₁₇	0.670	0.684	0.675	0.688	0.684	0.684	0.670	1.680	0.688
7	A ₁₈ ~A ₂₁	1.680	1.667	1.636	1.678	1.677	1.677	1.675	2.530	1.657
8	A ₂₂ ~A ₂₅	2.670	2.662	2.669	2.627	2.663	2.663	2.697	545.66	2.663
중량(lb)		545.22	545.17	548.47	545.49	545.36	545.16	545.32	545.03	544.38

3.2. 72부재 입체 트러스

그림3의 72부재 입체트러스의 재료중량과 탄성계수가 각각 0.1 lb/in.³, 10,000 ksi이며, 표5에 나타낸바와 같이 2개의 하중을 조건으로 하며, x축과 y축에 대하여 양방향 대칭이다. 전체부재는 16그룹으로 나누어지며, 따라서 단면변수는 16개이다. 72부재 입체트러스의 x와 y방향에 대한 압축 및 인장응력 제약조건은 ±25 ksi.이며, 최상위 절점의 최대 변위는 ±0.25 in.을 초과해서는 안 된다. 또한 모든 부재의 최소단면적은 0.01 in.²로 제약된다.

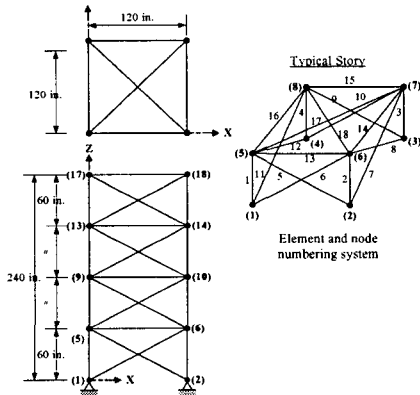


그림 3. 72부재 입체 트러스

표 5. 부재의 허용응력

절점 번호	하중조건 1(kips)			하중조건 2(kips)		
	P _x	P _y	P _z	P _x	P _y	P _z
17	5.0	5.0	-5.0	0.0	0.0	-5.0
18	0.0	0.0	0.0	0.0	0.0	-5.0
19	0.0	0.0	0.0	0.0	0.0	-5.0
20	0.0	0.0	0.0	0.0	0.0	-5.0

한편, HSHA의 변수 즉, HMS=20, HMCR=0.8 및 PAR=0.3으로 설정하였으며, 탐색수는 20,000번으로 설정하였다. 72부재 입체트러스에 부과된 각 제약조건을 바탕으로 20,000번 탐색후 얻어진 16개의 설계변수에 대한 HSHA의 최적해 벡터는 (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, 0.487, 0.551)이며, 이에 대응하는 최적 중량은 364.33 lb이다. 표6에는 HSHA의 최적값을 포함한 기존연구의 최적값을 비교하여 나타내었다. HSHA를 이용하여 얻어진 최적중량은 기존 수학적 알고리즘을 이용한 방법과 유전자알고리즘을 이용한 방법보다 더 좋은 값을 나타내었다. 한편, Simple Genetic Algorithm을 이용한 방법은 2,776번의 구조해석 후 372.40 lb의 최소중량값을 얻었으며, HSHA를 이용한 방법은 Simple Genetic Algorithm에서 획득한 372.40 lb의 중량을 1,076번의 구조해석 후에 획득하였다. 이와 같은 사실은 본 연구에서 도입한 HSHA방법이 현재의 대표적인 자기학습형 알고리즘인 GA보다 효과적인 알고리즘이라는 사실을 나타내고 있다.

표 6. 72부재 입체 트러스의 최적 결과값 비교

변수		최적 단면(in ²)		
		Adeli and Park	Sarma and Adeli Simple GA	This work
1	A ₁ ~ A ₄	2.755	2.141	1.963
2	A ₅ ~ A ₁₂	0.510	0.510	0.481
3	A ₁₃ ~ A ₁₆	0.010	0.054	0.010
4	A ₁₇ ~ A ₁₈	0.010	0.010	0.011
5	A ₁₉ ~ A ₂₂	1.370	1.489	1.233
6	A ₂₃ ~ A ₃₀	0.507	0.551	0.506
7	A ₃₁ ~ A ₃₄	0.010	0.057	0.011
8	A ₃₅ ~ A ₃₆	0.010	0.013	0.012
9	A ₃₇ ~ A ₄₀	0.481	0.565	0.538
10	A ₄₁ ~ A ₄₈	0.508	0.527	0.533
11	A ₄₉ ~ A ₅₂	0.010	0.010	0.010
12	A ₅₃ ~ A ₅₄	0.643	0.066	0.167
13	A ₅₅ ~ A ₅₈	0.215	0.174	0.161
14	A ₅₉ ~ A ₆₆	0.518	0.425	0.542
15	A ₆₇ ~ A ₇₀	0.419	0.437	0.478
16	A ₇₁ ~ A ₇₂	0.504	0.641	0.551
중량(lb)		376.50	372.40	364.33
구조해석 수(탐색 수)		-	2776	19878

4. 결 론

최근에 개발된 HSHA는 완벽한 하모니를 창출한다는 음악적 개념을 도입시켜 개념화했다. 기존의 자기학습형 알고리즘과 비교해보면 HSHA는 단순한 수리적 조건이 요구되며, 다양한 유형의 구조물의 최적화 문제에 보다 쉽게 이용할 수 있다. 또한 HSHA는 변수의 초기조건을 요구하는 경사도 탐색법과 달리 하모니메모리 기여율(HMCR), 피치 조정률(PAR)을 연산자로 한 확률론적 랜덤 탐색방법을 사용한다. 한편, 대표적인 자기학습형 알고리즘인 유전자 알고리즘의 경우는 단지 2개의 부모인자를 고려하여 새로운 해백터를 선출하지만, 본 연구의 HSHA는 HMCR, PAR을 바탕으로 존재하는 모든 영역에서 새로운 해백터를 선출한다.

본 연구에서는 상기의 HSHA를 토대로 한 입체트러스의 단면 최적화방법을 제안하였다. HSHA를 토대로 한 탐색방법은 입체트러스의 최적화 문제에 적용 가능하였으며, 그 최적결과는 현행의 수학적 알고리즘을 이용한 구조물 최적설계법 및 유전자 알고리즘(Simple Genetic Algorithm)을 이용한 최적설계법을 통해 획득된 결과보다 우수하였다. 이 같은 사실은 HSHA가 기존의 알고리즘을 이용한 구조물 최적화방법보다 효과적임을 나타낸다.

본 연구에서 제안된 HSHA를 이용한 최적화 방법은 연속변수를 다루는 트러스 구조물의 최적화 문제에만 한정되지 않으며, 골조 등의 트러스이외의 구조물, 이산변수의 최적화 등에도 용이하게 적용가능하다고 판단된다.

참고문헌

1. Z.W. Geem, J.-H. Kim and G.V. Loganathan, A new heuristic optimization algorithm: harmony search. *Simulation* 76 2 (2001), pp. 60 - 68.
2. L.A. Schmit, Jr and B. Farshi, Some approximation concepts for structural synthesis. *AIAA J.* 12 5 (1974), pp. 692 - 699.
3. Schmit Jr LA, Miura H. Approximation concepts for efficient structural synthesis. NASA CR-2552, Washington, DC: NASA; 1976
4. V.B. Venkayya, Design of optimum structures. *Comput. & Structures* 1 1 - 2 (1971), pp. 265 - 309.
5. Gellatly RA, Berke L. Optimal structural design. AFFDL-TR-70-165, Air Force Flight Dynamics Lab., Wright-Patterson AFB, OH; 1971
6. Rizzi P. Optimization of multiconstrained structures based on optimality criteria, PA:1976
7. Templeman AB, Winterbottom SK. Structural design by geometric programming. Second symposium on structural optimization, AGARD Conference, Preprint-123, Milan; 1973
8. Adeli H, Kamal O, Efficient optimization of space trusses. *Computer & Structures* 1986;24(3):501-11
9. Z.W. Geem, J.-H. Kim and G.V. Loganathan, Harmony search optimization: Application to pipe network design. *Int. J. Modell. Simulat.* 22 2 (2002), pp. 125 - 133.
10. 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
11. Adeli H, Cheng N.T. Integrated genetic algorithm for optimization of space structures. *J. Aerospace Eng., ASCE* 4 (1993), pp. 315 - 328.
12. D.E. Goldberg, Genetic algorithms in search optimization and machine learning. , Addison-Wesley, Boston, MA (1989).