

음계를 기반으로 한 HS 구현

이태봉*

HS Implementation Based on Music Scale

Tae-Bong Lee*

요약 Harmony Search(HS)는 비교적 최근에 개발된 메타 휴리스틱 최적화 알고리즘으로 최근 이에 관한 연구가 다양하게 진행되고 있다. HS는 음악인의 즉흥 연주를 기반으로 하고 있으며 목적변수는 악기의 역할을 한다. 그러나 각 악기는 음대역만 주어질 뿐 음악의 기본이라 할 수 있는 음계의 개념이 없다. 본 연구에서는 기존 HS에 음계를 도입하고 대역폭을 양자화하여 알고리즘의 성능을 향상시키고자 한다. 도입한 음계는 음대역 범위에서 무작위로 초기화되던 기존 방식을 대신하여 HM 초기화에 적용하였다. 양자화 단계는 임의로 정할 수 있도록 하였으며 이를 통해 알고리즘 초반에는 상대적으로 큰 대역폭을 사용하여 알고리즘의 탐색성을 향상시키고 후반에는 작은 대역폭을 통해 탐지성을 향상시키고자 하였다. 음계 도입과 대역폭 양자화를 통하여 기존 HS보다 초기값에 따른 알고리즘 성능 편차를 줄이고 알고리즘 수렴속도 및 성공률을 향상시킬 수 있었다. 본 연구의 성과는 여러 함수에 대한 최적화 수치 예를 종래의 방식과 비교하여 확인하였다. 구체적인 비교 수치는 모의실험에 서술하였다.

Abstract Harmony Search (HS) is a relatively recently developed meta-heuristic optimization algorithm, and various studies have been conducted on it. HS is based on the musician's improvisational performance, and the objective variables play the role of the instrument. However, each instrument is given only a sound range, and there is no concept of a scale that can be said to be the basis of music. In this study, the performance of the algorithm is improved by introducing a scale to the existing HS and quantizing the bandwidth. The introduced scale was applied to HM initialization instead of the existing method that was randomly initialized in the sound band. The quantization step can be set arbitrarily, and through this, a relatively large bandwidth is used at the beginning of the algorithm to improve the exploration of the algorithm, and a small bandwidth is used to improve the exploitation in the second half. Through the introduction of scale and bandwidth quantization, it was possible to reduce the algorithm performance deviation due to the initial value and improve the algorithm convergence speed and success rate compared to the existing HS. The results of this study were confirmed by comparing examples of optimization values for various functions with the conventional method. Specific comparative values were described in the simulation.

Key Words : bandwidth, HS, HM(harmony memory), meta-heuristic, quantization, music scale

1. 서론

harmony search(HS)는 Geem 등이 음악 연주자들의 즉흥 연주에서 영감을 받아 개발한 새로운 메타 휴리스틱(meta-heuristic) 알고리즘이다[1,2]. 메타 휴리스틱 기법은 수학적 알고리즘이 갖는 복잡함과 초

깃값에 대한 민감성 등과 같은 단점을 극복하고자 1970년대 이후 많이 연구되었다[3-5]. 대표적인 것으로는 유전자 알고리즘(GA: Genetic Algorithm), 개미군집최적화(ACO: Ant Colony Optimization), 그리고 입자군집최적화(PSO: Particle Swarm Optimization), 등이 있다[6-8]. 고전적 기법과

*Department of Electronic Engineering, Gachon University
 Received August 02, 2022 Revised September 20, 2022

Accepted October 17, 2022

다르게 HS는 비수학적 알고리즘으로 설계변수에 대한 초깃값을 요구하지 않으며 경사법(gradient search) 대신 확률변수 기법(stochastic random search)을 기반으로 하고 있다. 지난 몇 년 동안 HS는 전통적인 최적화 기법이 효과적이지 못한 복잡한 최적화 문제나 어려운 공학적 최적화 문제를 해결하는데 활발하게 적용되어 왔다[9-13]. 이와 더불어 알고리즘 매개변수 조율이나 particle swarm optimization (PSO)과 같은 다른 최적화 기법을 도입하여 HS의 성능 향상을 위한 연구 또한 활발하게 진행되어 왔다[14-17].

구체적으로 Mahdavi 등은 PAR(pitch adjusting ratio)과 bw(band width)를 알고리즘 반복에 따라 변화도록 동적으로 설계하여 improved harmony search (IHS)라 명명하였으며 HS 알고리즘 성능을 크게 개선 할 수 있었다[15]. Omran과 Mahdavi는 bw을 대신하기 위해 PSO의 아이디어를 도입한 global harmony search(GHS) 알고리즘을 소개하였으며[16] Al-Omouh등[14]은 HS의 성능 개선을 위한 다양한 방법에 대한 연구와 응용에 대해 발표하기도 하였다.

HS와 같은 진화적 알고리즘의 성능은 활용성(exploitation)과 탐색성(exploration)이 매우 중요하다[11]. 활용성은 최적해를 찾기 위한 기존 정보 이용능력을 말하고 탐색성은 새로운 영역을 탐색하는 알고리즘의 특성이다. Das[18] 등은 선택과정이 없다는 것을 전제로 HS의 정보 탐색성에 대한 분석을 하였고 Lee[19]는 HMCR(harmony memory considering rate)과 PAR 값에 따른 HS의 새로운 하모니 조성의 확률적 특성과 정보 활용성 및 새로운 정보 탐색성과 연관성을 분석하였다.

HS는 알고리즘 매개변수 HMCR과 PAR를 통해 활용과 탐색이라는 두 요소의 조화를 이룰 수 있다. 특히 HMCR은 기존 정보활용과 새로운 값 탐색을 확률적으로 결정하는 매개변수이다. 기존 정보를 활용할 때 HS는 지금까지 가장 좋은 값을 중심으로 해벡터를 개선해 가는 PSO와 다르게 HM에 저장된 벡터의 모든 요소를 균일분포 확률함수를 이용하여 선택한다. 이것은 음악인들이 즉흥 연주를 하며 하모니를 개선할 때 기억에 남아 있는 음을 선택해 더 좋은 하

모니를 만들기 위해 임의로 피치 조절을 하거나 기존 음을 그대로 사용하며 음을 개선해 나가는 방식을 모방한 것이다.

본 연구에서는 기존 HS에 음악의 기본인 음계(scale)의 개념을 도입해 HS의 성능을 개선하고자 한다. 음계는 서로 다른 음의 조합으로 구성되며 현악기나 관악기의 경우 길이의 차이로 구분된다. 그리고 연주하는 각 악기에 따라 정해진 음계를 통해 이루어진다. 음계 구성의 일차적 목적은 HM을 초기화할 때 이용하는 것이다. 고전적인 HS에서 HM의 초기화는 무작위 방식이다. 이에 따른 단점은 초기화 값에 따라 그 결과에 차이가 크다는 것이다. 이러한 방식은 음악을 연주하기 전에 음의 튜닝을 먼저 하는 기본적인 절차와도 다르다 할 수 있다. 따라서 본 연구에서는 음계를 정의하고 정의된 음계를 이용하여 HM을 초기화하는 방식을 제안하였다. 음계를 이용한 HM의 초기화는 연주에 앞선 음의 튜닝과 같은 개념이라 할 수 있다.

본 논문은 1장 서론에 이어 2장에서 HS 이론을 소개하였고 3장에서는 본 논문의 연구내용을 서술하였다. 마지막으로 4장에서는 여러 시험 평가함수를 대상으로 제시된 알고리즘을 다른 방식과 비교하여 그 우수성을 입증하였다.

2. HS

Geem[1]등은 연주자들이 즉흥 연주시 좋은 하모니를 이루어 가는 과정이 다음과 같은 세 가지 방법을 통해 이루어진다고 정의하였다.

- 1) 경험으로 남은 기억 속 음 선택
- 2) 기억 속 음을 기준으로 그에 이웃한 음을 선택
- 3) 기억에 의존하지 않고 악기 음역에서 임의로 선택

위 세 가지 방식을 Geem은 각각 ‘기억회상’, ‘피치조정’ 및 ‘무작위 선택’이라 구분하고 최적화 문제의 결정변수 값을 3가지 방식 중 하나를 확률적으로 선택 결정하여 해벡터를 구성하였다. HS는 이렇게 결정된 해벡터를 평가함수로 평가하여 더 좋은 값을 선택해 가는 방식이다. HS 수행에 필요한 알고리즘 매개변수는 좋은 음을 기억하기 위한 기억 장소 메모리

HM(harmony memory)과 크기 HMS(harmony memory size). 결정변수 결정 방법 선택 확률 정의에 필요한 HMCR(harmony memory considering rate) 및 피치조정 여부를 결정할 확률 정의에 필요한 PAR(pitch adjusting rate) 및 피치조정시 대역폭 bw 이다.

알고리즘 구현은 단계별로 다음과 같이 이루어진다.

- 단계 1. 최적화 문제와 알고리즘 매개변수 초기화.
- 단계 2. HM 초기화.
- 단계 3. 새로운 하모니 즉 해 벡터 생성.
- 단계 4. HM 최신화.
- 단계 5. 중단 요건 검사.

2.1 문제 정의 및 알고리즘 초기화

제한 조건이 없는 최적화 문제는 다음과 같은 식으로 표현 할 수 있다.

$$\text{Min(or Max)}f(x), x = [x_1 \ x_2 \ \dots \ x_n], x_i \in B_i, \\ (i = 1, \dots, n) \quad (1)$$

(1)식에서 $f(x)$ 는 목적함수, x_i 는 결정변수이며 연속적이거나 이산적이다. n 은 결정변수의 수이며 B_i 는 각 결정변수 범위를 나타내는 집합으로 다음 식과 같이 정의된다.

$$B_i = \{x_i | L_i \leq x_i \leq U_i\}, (i = 1, \dots, n) \quad (2)$$

HS 알고리즘 매개변수 초기화에는 HM의 크기인 HMS, 결정변수 선택 방식을 결정할 확률 매개변수 HMCR 및 PAR. 그리고 피치조정시 적용되는 대역폭 bw와 알고리즘 반복시수 또는 종료 조건에 대한 정의가 필요하다.

2.2 HM 초기화

HM은 알고리즘 수행 과정에서 최근의 좋은 화음 즉, 현재까지 선택된 해를 저장하는 장소로 그 크기 HMS를 정의한 후 균일분포 확률변수를 이용하여 다

음과 같이 초기화한다.

$$x_i^j = L_i + U(0,1) \cdot (U_i - L_i), i = 1, \dots, n \quad j = 1, 2, \dots, \text{HMS} \quad (3)$$

$U(a,b)$ 은 구간 (a,b) 에서 정의되는 균일분포 확률수에 의해 결정되는 무작위 값이다. HM에 저장된 초기 해벡터는 다음과 같이 나타낼 수 있다.

$$\text{HM} = \begin{bmatrix} x^1 \\ \vdots \\ x^{\text{HMS}} \end{bmatrix}, x^j = [x_1^j \ x_2^j \ \dots \ x_n^j], j = 1, \dots, \text{HMS} \quad (4)$$

2.3 새로운 하모니 생성

새로운 하모니 $x^{\text{New}} = \{x_1^{\text{New}}, x_2^{\text{New}}, \dots, x_n^{\text{New}}\}$ 의 각 원소 x_i^{New} 는 '기억회상', '피치조정' 및 '무작위 선택' 세 가지 방식 중 하나가 확률적으로 결정되어 정해진다. 기억회상과 피치조정은 x_i^{New} 을 HM에 저장된 값 중 하나를 선택하여 결정하는 HM 참조방식이고 무작위 선택은 기존 값이 아닌 정의된 변수 범위 내에서 임의의 값을 선택하는 방식이다. HM의 참조 여부를 결정하는 확률은 HMCR 값에 의해 다음 식과 같이 결정된다.

$$x_i^{\text{New}} = \begin{cases} x_i^{\text{New}} \in \{x_i^1, x_i^2, \dots, x_i^{\text{HMS}}\}, & \text{w.p. HMCR} \\ x_i^{\text{New}} = L_i + U(0,1) \cdot (U_i - L_i) & \text{w.p. (1 - HMCR)} \end{cases} \quad (5)$$

(5)식에서 HM의 값이 선택된 경우 그 값을 그대로 사용할 것인지 아니면 피치조정을 할 것인지 또한 한 번의 확률적 결정을 하게 된다. 이때 사용되는 알고리즘 확률 매개변수가 PAR로 그 확률적 관계는 다음 식과 같다.

$$x_i^{\text{New}} = \begin{cases} x_i^{\text{New}} + U(-1,1) \cdot \text{bw}_i, & \text{w.p. HMCR} \cdot \text{PAR} \\ x_i^{\text{New}}, & \text{w.p. HMCR} \cdot (1 - \text{PAR}) \end{cases} \quad (6)$$

(6)식에서 bw_i 는 변수 x_i 의 피치 조정 대역폭이다.

결과적으로 새로운 하모니 조성을 위해 각 변수를 선택하는 방식은 ‘기억회상’, ‘피치조정’ 및 ‘무작위 선택’ 세 가지 중 하나이고 각 방식의 확률적 관계는 다음 식 (7)과 같다.

$$x_i^{New} = \begin{cases} x_i^{New} \in \{x_i^1, x_i^2, \dots, x_i^{HMS}\}, & \text{w.p. HMCR} \cdot (1 - PAR) \\ x_i^{New} = x_i^{New} + U(-1, 1) \cdot bw_i, & \text{w.p. HMCR} \cdot PAR \\ x_i^{New} = L_i + U(0, 1) \cdot (U_i - L_i), & \text{w.p. (1 - HMCR)} \end{cases} \quad (7)$$

2.4 HM 최신화

단계 3에서 결정된 새로운 하모니는 목적함수에 적용되고 그 결과와 HM내 가장 나쁜 결과와 비교해 더 좋은 경우 새로운 하모니는 해당 하모니를 대체한다. 이러한 해 벡터 개선과정은 그림 1과 같다[19].

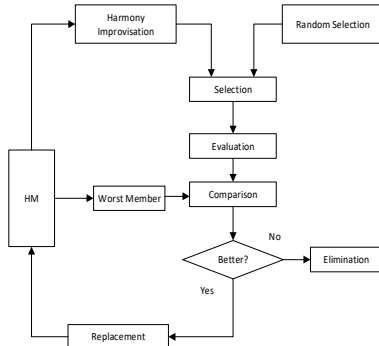


그림 1 HS의 하모니 개선과정
Fig. 1 HS process of harmony improvisation

단계 3과 4의 새로운 하모니 생성과 최신화 과정은 미리 설정된 종료 기준을 만족할 때까지 반복된다. 지금까지 설명한 HS의 단계별 알고리즘 전개 과정을 그림으로 나타내면 그림 2와 같다[19].

3. 음계 HS(SHS)

음악의 기본은 음계(scale)이다. 연주는 음계를 구성하는 개별 음의 조합 즉 화음을 통하여 이루어진다. 본 연구에서는 기존 HS에 음계 개념을 도입하여 좀더 세밀하게 음악적 특성을 모방하여 기존 HS 성능을 개선하고자 하며 제안한 방식을 SHS(scale harmony search)라 명명하기로 한다.

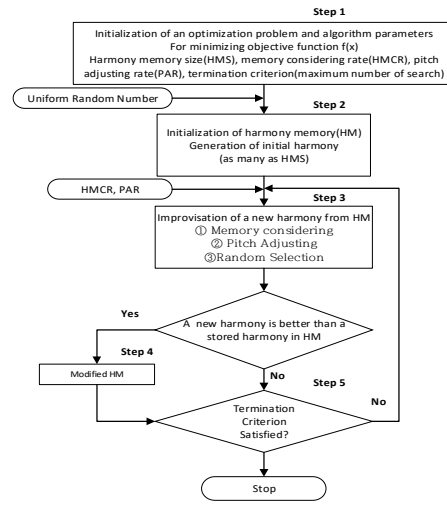


그림 2. HS 알고리즘의 단계별 최적화 절차
Fig. 2. Optimization step procedure of the harmony search algorithm

3.1 음계 구성

현악기나 관악기 등 모든 악기는 현이나 관의 길이를 특정한 간격으로 조정함으로써 음을 구분하고 음계는 이러한 서로 다른 음의 모임으로 구성된다. HS에서 목적변수는 악기에 비유되며 목적변수의 주어진 범위는 해당 목적변수 악기가 표현할 수 있는 음대역이라 할 수 있다. 따라서 음대역 즉, 변수의 범위를 적절한 간격으로 구분함으로써 해당 목적변수의 음계를 구성할 수 있다. 음대역을 N개의 구간으로 나누고자 하는 경우 그 간격 SIT는 다음과 같이 정의된다.

$$SIT = (x_{max} - x_{min}) / N \quad (8)$$

그러면 음계를 구성하는 개별 음 s_j 는 다음과 같이 나타낼 수 있다.

$$s_j = x_{min} + j * SIT, \quad j = 0, \dots, N \quad (9)$$

그러면 음계의 수는 $SN = N + 1$ 의 음으로 구성되고 첫 음은 x_{min} 이고 마지막 음은 x_{max} 가 된다. 음계는 목적변수별로 존재해야 하므로 목적변수 수가 n인 경

우 (8) 식은 다음과 같이 표현된다.

$$S_j^i = x_{\min}^i + j^*SIT_i, \quad j=0,1,\dots,N, \quad i=1,2,\dots,n \quad (10)$$

(10) 식의 음계를 저장하기 위해서는 $(SN \times n)$ 크기의 공간이 필요하며 이를 SM(Scale Memory)이라 정의하기로 한다.

3.2 HM 초기화(화음 조성)

기존 HS에서 HM은 각 변수 음대역 즉 변수 범위에서 균등분포 확률함수를 이용하여 무작위 값을 선택하여 HMS만큼 초기화하였다. 본 연구에서는 앞에서 정의한 SM을 이용하여 HM을 초기화하고자 한다. SM을 기반으로 초기화하는 여러 가지 방법이 있을 수 있다. 본 연구에서는 SM의 악기별 음계를 무작위로 조합하여 화음을 조성하여 평가한 후 그중 가장 좋은 화음 HMS개를 선택하였다. 무작위로 화음을 조성하는 의사코드는 다음과 같다.

```
for i=1 to n
  dx=randomperm(1,...,SN)
  SM(1,...,SN,i)=SM(idx,i)
end
```

음계를 정의하여 HM을 초기화 할 경우 그림 2의 절차는 다음 그림 3과 같이 바뀐다.

3.3 bw 양자화

HS에서 bw은 알고리즘의 탐색능력과 수렴속도 및 해의 정밀성과 밀접한 관계가 있다. 초기 HS는 고정된 값의 bw를 사용하였으나 이후 가변적인 bw를 사용하여 성능을 개선하였다. 본 연구에서도 가변적인 bw를 사용되 종래와 같이 연속적인 값이 아닌 양자화로 구현하고자 한다.

즉 본 연구에서는 최적해를 찾는 알고리즘 최대 반복 횟수를 일정 간격으로 N개의 구간으로 구분하고 각 구간에 대한 bw를 다음과 같이 양자화하여 결정하였다.

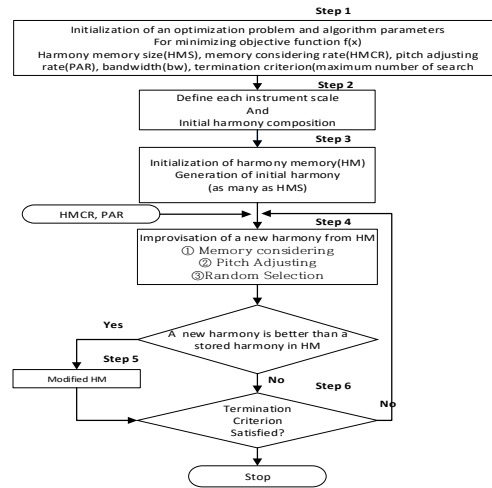


그림 3. 음계를 적용한 HS 알고리즘의 단계별 최적화 절차
Fig. 3. Optimization step procedure of the harmony search algorithm with scale

$$bw(n) = \Delta_{bw} 10^{-n}, \quad n=0,\dots,N-1 \quad (11)$$

다음 그림은 수식 (11)에서 $\Delta_{bw} = 1$, 알고리즘 최대 수행 Maxit=10000이고 양자화 단계를 N=5로 설정한 경우 구간별 대역폭, $bw(n)$ 을 나타낸다.

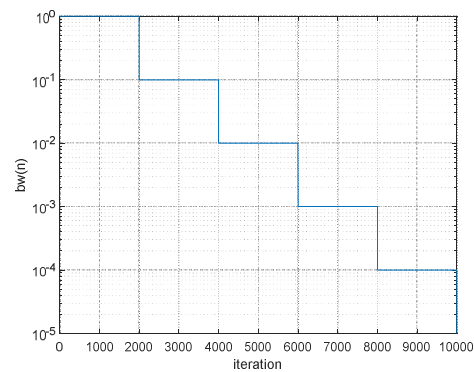


그림 4. 대역폭 양자화의 예($\Delta_{bw} = 1, N=5$)
Fig. 4. Example of bw quantization($\Delta_{bw} = 1, N=5$)

4. 모의 실험

제시된 알고리즘 성능 확인을 위해 고전적인 HS와 더불어 IHS에 적용하여 비교하였으며 비교 대상으로 선정된 함수는 다음과 같다[1,19].

표 1. 예시에 사용한 함수
Table 1. Functions for Numerical example

Function Name	Expression	Optimal Value	Search Range
1. Six-Hump Camelback	$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	$f^* (\mp 0.08984, \pm 0.71266) = -1.0316285$	$-10 \leq x_i \leq 10$
2. Rosenbrock	$f(x) = 100(x_2 - x_1^2) + (1 - x_1)^2$	$f^*(1, 1) = 0$	$-10 \leq x_i \leq 10$
3. Goldstein and Price 1	$f(x) = \left\{ 1 + (x_1 + x_2 + 1)^2 \left(19 - 14x_1 + 3x_1^2 - 14x_2 \right) + 3x_2^2 \right\} \times \left\{ 30 + (2x_1 - 3x_2)^2 \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2 \right) \right\}$	$f^*(0, -1) = 3.0$	Multimodal $-10 \leq x_i \leq 10$
4. Goldstein and Price 2	$f(x) = \exp\left\{ \frac{1}{2}(x_1^2 + x_2^2 - 25)^2 \right\} + \sin^4(4x_1 - 3x_2) + \frac{1}{2}(2x_1 + x_2 - 10)^2$	$f^*(3, 4) = 1.0$	$-10 \leq x_i \leq 10$
5. Eason & Fenston	$f(x) = \frac{1}{10} \left\{ 12 + x_1^2 + \frac{1 + x_2^2}{x_1^2} + \frac{x_1^2x_2^2 + 100}{(x_1x_2)^4} \right\}$	$f^*(1.7435, 2.0297) = 1.74415$	$-10 \leq x_i \leq 10$
6. Wood	$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2]$	$f^*(1, 1, 1, 1) = 0$	$-10 \leq x_i \leq 10$
7. Powell quadratic	$f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + (10(x_1 - x_4))^4$	$f^*(0, 0, 0, 0) = 0$	$-10 \leq x_i \leq 10$

다음 표 2는 비교 알고리즘과 제안된 방식에 사용된 알고리즘 매개변수를 나타낸다.

음의 간격을 정의하기 위한 값 : $N = HMS \times 10$
음계를 저장할 메모리 SM의 크기 : $SN \times V_n$

표 2. 알고리즘에 사용한 매개변수
Table 2. Parameters of algorithm

Algorithm	HS	IHS	SHS
HMS	10	10	10
HMCR	0.9	0.95	0.9
PAR (PAR _{min})	0.9	(0.35)	0.9
PAR _{max}	/	0.99	/
bw (bw _{min})	0.01	(0.0001)	0.01
bw _{max}	/	$(x_{max} - x_{min})/20$	/

본 연구에서 제시한 알고리즘을 구현을 위해 필요한 추가적인 알고리즘 매개변수는 다음과 같다.

먼저 양자화 대역폭을 사용하지 않고 고정 대역폭을 사용한 경우 전통적인 HS와 음계를 적용한 SHS의 성능을 비교하였다. 함수별로 알고리즘 iteration을 다르게 적용하였으며 성능 비교는 함수별로 100회 반복 실행하여 허용오차에 대한 성공률과 결과의 표준편차 및 평균값으로 하였다.

그 결과는 표 3과 같다. 표 3을 보면 고정 대역폭을 사용한 경우 성공률과 표준편차에 있어 전통적인 방식과 음계를 적용한 방식에 있어 차이가 매우 크다는 것을 확인 할 수가 있다. 구체적으로 절대적 차이에서는 적게는 8%(함수 6) 최대 30%(함수 1)의 차이가 났으며 상대적 차이는 최소 56%(함수 3)에서 최대 91%(함수 6)이다. 표준편차의 경우 그 값이 작을수록 초기화의 영향이 적다는 것을 의미한다.

표 3. 고정 대역폭 수치 예 수행 결과(허용오차 10^{-5})
Table 3. The results of numerical example with fixed bandwidth

Func. No	Max iteration	Success rate(%)		Standard deviation		Average	
		HS	SHS	HS	SHS	HS	SHS
		1	1000	55	85	0.6768	0.1821
2	30000	60	80	0.7069	0.5570	0.3433	0.1891
3	5000	30	54	26161	87.4293	3907.4	39.0547
4	40000	9	15	0.9316	0.4701	1.7266	1.3566
5	1000	72	96	0.0386	0.0016	1.7577	1.7444
6	70000	78	86	4.2289e-06	2.8302e-06	7.9390e-06	7.4394e-06
7	30000	74	85	1.8804e-05	2.5624e-06	1.2656e-05	8.7959e-06

표준편차 크기를 비교해보면 함수에 따라 편차가 크기는 하나 고전적 방식이 음계를 적용한 경우보다 평균 37배 크게 나타난다. 다만 함수 7의 경우 고전

적 방식이 SHS보다 작게 나타났다.

다음은 양자화된 대역폭을 사용하여 전통적인 HS에 적용하여 고정 대역폭과 성능 차이를 확인하고 이와 더불어 가변 대역폭을 갖는 IHS와 성능을 비교하였다. 대역폭은 $10^0 \sim 10^{-5}$, 6단계로 양자화하였다. IHS의 경우 고유의 전통적인 초기화 방식과 음계를 이용한 초기화 방식을 같이 수행하였으며 전통적인 방식에 대해서는 성공률만 ()로 구별하여 표시하였으며 표 4는 그 결과이다.

먼저 고정 대역폭 결과표 3과 양자화 대역폭 결과표 4를 비교해보면 전통적인 방식 HS와 SHS 모두 양자화 대역폭을 사용한 결과가 상대적으로 매우 우수함을 알 수 있다. 구체적으로 성공률에 있어 고전적 HS는 평균 2.21배의 증가를 나타내고 SHS의 경우 평균 1.65배의 증가를 나타내고 있다. 참고로 IHS의 경우 성공률은 양자화 대역폭을 사용한 경우 IHS 고유의 가변대역폭을 사용하였을 때 보다 평균 1.46배 높게 나타났다.

표 4. 양자화 대역폭 수치 예 수행 결과 (허용오차 10^{-6})
Table 4. The result of Numerical example with quantization bandwidth

Func. No	Max ltr.	Success rate(%)			Standard deviation			Average		
		HS	IHS	SHS	HS	IHS	SHS	HS	IHS	SHS
1	500	76	91(53)	97	0.1202	0.0558	1.2558e-04	-1.0044	-1.0223	-1.0316
2	10000	71	55(36)	96	0.3562	0.4850	2.0538e-04	0.0939	0.1737	3.1892e-05
3	1000	68	53(32)	96	3.7316e+04	40.450	14.3094	4.2604e+03	31.1944	5.7368
4	10000	66	43(38)	64	0.0322	0.1491	0.0241	1.0171	1.0895	1.0153
5	500	87	100(59)	100	0.0021	2.3025e-06	2.0890e-06	1.7445	1.7442	1.7442
6	30000	81	54(40)	93	1.2197e-04	0.0139	2.9940e-05	1.8817e-05	0.0027	4.1351e-06
7	50000	77	63(55)	88	4.2416e-07	6.4152e-07	6.0661e-07	3.5150e-07	1.1735e-06	8.6917e-07

초기화의 영향을 알 수 있는 표준편차의 경우 결과표 4는 음계를 적용하여 알고리즘을 초기화하는 경우 성공한 값들의 표준편차의 차이가 함수에 따라 편차

가 크기는 하나 SHS보다 크다는 것을 보여주고 있다. 평균적으로는 SHS 기준 고전적 HS는 900배 IHS 경우 468배가 큼을 보여주고 있다. 다만 함수

7의 경우 HS가 표준편차가 제일 작게 나타났다. IHS에 음계를 통해 HM을 초기화할 경우 성공률이 매우 커진다는 것은 음계의 효과를 입증하는 것이라 할 수 있다.

5. 결론

본 연구에서는 음악의 기본이 되는 음계를 HS에 도입하여 HM의 초기화에 적용하였다. 그와 더불어 피치조정시 적용되는 대역폭을 양자화하여 알고리즘 초반에는 상대적으로 큰 대역폭을 사용함으로써 탐색성을 크게하고 알고리즘 후반에는 대역폭을 작게 함으로써 기존해를 이용하여 좀 더 세밀하게 해를 찾을 수 있는 활용성을 크게하고자 하였다.

먼저 음계도입의 효과를 살펴보기 위해 고정 대역폭을 갖는 고전적 HS에 음계를 적용한 결과 표 3에서 보듯이 표준편차 값이 크게 줄고 성공률 또한 향상됨을 확인 할 수 있었다. 양자화 대역폭과 음계 도입의 효과를 입증하기 위해서는 고전적 HS와 더불어 가변 대역폭을 사용하는 IHS를 선정하여 비교하였다. 이 결과 역시 표 4에서 보듯이 성공률과 표준편차에 있어 SHS 방식이 우수함을 알 수 있다. 다만 함수 7의 경우 표준편차 값이 고전적 방식에서 제일 작게 나타났다.

향후 과제로는 정의된 음계를 이용하여 보다 효과적으로 HM을 초기화하는 방법과 HM 초기화 뿐만 아니라 알고리즘 전반에 이용하여 HS 알고리즘의 성능을 향상시키는 것이다.

REFERENCES

- [1] Geem ZW, Kim JH, Loganat han GV, "A new heuristic optimization algorithm: harmony search", *Simulation*, vol. 76(2), pp. 60-68. 2001.
- [2] K. S. Lee and Z. W. Geem, "A new metaheuristic algorithm for continuous engineering optimization: Harmony search theory and practice", *Comput. Methods Appl. Mech. Eng.*, vol. 194 pp. 36-38, 2004.
- [3] A. P. Engelbrecht, "Fundamentals of Computational Swarm Intelligence", Hoboken, NJ: Wiley, 2006.
- [4] J. Kennedy, R. C. Eberhart, and Y. Shi, "Swarm Intelligence", *San Francisco, CA: Morgan Kaufmann*, 2001.
- [5] Slowik, A., Kwasnicka, H. "Evolutionary algorithms and their applications to engineering problems", *Neural Comput. & Applic.* 32, 12363-12379, 2020.
- [6] DORIGO, Marco; DI CARO, Gianni. "Ant colony optimization: a new meta-heuristic", *In: Proceedings of the 1999 congress on evolutionary computation-CEC99, IEEE*, pp. 1470-1477, 1999.
- [7] EBERHART, Russell; KENNEDY, James. "Particle swarm optimization", *In: Proceedings of the IEEE international conference on neural networks*. pp.1942-1948. 1995.
- [8] GOLDBERG, David E.; HOLLAND, John H. "Genetic algorithms and machine learning", *Machine learning*, 3.2: pp. 95-99. 1988.
- [9] Kim JH, Geem ZW, Kim ES., "Parameter estimation of the nonlinear Muskingum model using harmony search", *Water Resour Assoc.* vol. 37(5), pp. 1131-1138. 2001.
- [10] Geem ZW, Kim JH, Loganat han GV, "Harmony search optimization: application to pipe network design", *Int. J. Model. Simul.* vol. 22(2), pp. 125-133, 2002.
- [11] SL Kang, Geem ZW., "A new structural optimization method based on the harmony search algorithm", *Comput. Struct.* vol. 82(9-10), pp. 781-798. 2004.
- [12] Geem ZW, C Tseng, Y Park, Harmony search for generalized orienteering problem: best touring in China, *in: Springer Lecture Notes in Computer Science*, vol. 3 412, pp. 741-750, 2005.
- [13] Fesanghary M, Damangir E, Soleimani I., "Design optimization of shell and tube heat exchangers using global analysis and harmony search algorithm", *Appl. Therm. Eng.* vol. 29, pp. 1026-1031. 2009.
- [14] A. A. Al-Omoush, A. A. Alsewari, H. S. Alamri and K. Z. Zamli, "Comprehensive Review of the Development of the Harmony Search Algorithm and its Applications," *in IEEE Access*, vol. 7, pp. 14233-14245, 2019.

- [15] MAHDAVI, Mehrdad; FESANGHARY, Mohammad; DAMANGIR, E. "An improved harmony search algorithm for solving optimization problems", *Applied mathematics and computation*, vol.188, No.2: pp. 1567-1579, 2007.
- [16] OMRAN, Mahamed GH; MAHDAVI, Mehrdad. "Global-best harmony search", *Applied mathematics and computation*, vol. 198, No. 2: pp.643-656, 2008,
- [17] WANG, Chia-Ming; HUANG, Yin-Fu. "Self-adaptive harmony search algorithm for optimization", *Expert Systems with Applications*, vol.37, No.4, pp. 2826-2837, 2010.
- [18] S. Das, A. Mukhopadhyay, A. Roy, A. Abraham and B. K. Panigrahi, "Exploratory Power of the Harmony Search Algorithm: Analysis and Improvements for Global Numerical Optimization," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 1, pp. 89-106, Feb. 2011.
- [19] T. B. Lee, "Analytic Approach to Determining Algorithm Parameters of HS Optimization", *Trans. of KIEE* vol. 68, no. 4, pp. 200-206, 2019.

저자약력

이 태 봉(Tae-Bong Lee)

[정회원]



- 1986년 홍익대학교 전자공학과 졸업.
- 1989년 동 대학원 전자공학과 석사
- 1994년 동 대학원 전자공학과 박사
- 1995년 ~현재 : 가천대학교 전자공학부 교수

<관심분야>

최적이론, 견실제어, 휴리스틱 알고리즘