

자가 적응형 메타휴리스틱 최적화 알고리즘 개발: Self-Adaptive Vision Correction Algorithm

이의훈¹, 이호민², 최영환³, 김종훈^{4*}

¹충북대학교 토목공학부, ^{2,3}고려대학교 방재과학기술연구, ⁴고려대학교 건축사회환경공학부

Development of Self-Adaptive Meta-Heuristic Optimization Algorithm: Self-Adaptive Vision Correction Algorithm

Eui Hoon Lee¹, Ho Min Lee², Young Hwan Choi³, Joong Hoon Kim^{4*}

¹School of Civil Engineering, Chungbuk National University

^{2,3}Research Center for Disaster Prevention Science and Technology, Korea University

⁴School of Civil, Environmental, and Architectural Engineering, Korea University

요약 본 연구에서 개발된 Self-Adaptive Vision Correction Algorithm (SAVCA)은 광학적 특성을 모방하여 개발된 Vision Correction Algorithm (VCA)의 총 6개의 매개변수 중 자가 적응형태로 구축된 Division Rate 1 (DR1) 및 Division Rate 2 (DR2)를 제외한 Modulation Transfer Function Rate (MR), Astigmatic Rate (AR), Astigmatic Factor (AF) 및 Compression Factor (CF) 등 4개의 매개변수를 변경하여 사용성을 증대시키기 위해 제시되었다. 개발된 SAVCA의 검증을 위해 기존 VCA를 적용하였던 2개 변수를 갖는 수학 문제 (Six hump camel back 및 Easton and fenton) 및 30개 변수를 갖는 수학 문제 (Schwefel 및 Hyper sphere)에 적용한 결과 SAVCA는 비교한 다른 알고리즘 (Harmony Search, Water Cycle Algorithm, VCA, Genetic Algorithms with Floating-point representation, Shuffled Complex Evolution algorithm 및 Modified Shuffled Complex Evolution)에 비해 우수한 성능을 보여주었다. 마지막으로 공학 문제인 Speed reducer design에서도 SAVCA는 가장 좋은 결과를 보여주었다. 복잡한 매개변수 조절과정을 거치지 않은 SAVCA는 여러 분야에서 적용이 가능할 것이다.

Abstract The Self-Adaptive Vision Correction Algorithm (SAVCA) developed in this study was suggested for improving usability by modifying four parameters (Modulation Transfer Function Rate, Astigmatic Rate, Astigmatic Factor and Compression Factor) except for Division Rate 1 and Division Rate 2 among six parameters in Vision Correction Algorithm (VCA). For verification, SAVCA was applied to two-dimensional mathematical benchmark functions (Six hump camel back / Easton and fenton) and 30-dimensional mathematical benchmark functions (Schwefel / Hyper sphere). It showed superior performance to other algorithms (Harmony Search, Water Cycle Algorithm, VCA, Genetic Algorithms with Floating-point representation, Shuffled Complex Evolution algorithm and Modified Shuffled Complex Evolution). Finally, SAVCA showed the best results in the engineering problem (speed reducer design). SAVCA, which has not been subjected to complicated parameter adjustment procedures, will be applicable in various fields.

Keywords : Self-Adaptive, Vision Correction, Meta-Heuristic, Optimization, Algorithm

이 논문은 2018학년도 충북대학교 학술연구지원사업의 연구비 지원에 의하여 연구되었음.

(This work was supported by the research grant of the Chungbuk National University in 2018).

*Corresponding Author : Joong Hoon Kim(Korea Univ.)

Tel: +82-2-3290-3316 email: jaykim@korea.ac.kr

Received March 19, 2019

Revised April 19, 2019

Accepted June 7, 2019

Published June 30, 2019

1. 서론

메타휴리스틱 최적화 알고리즘과 같은 새로운 최적화 알고리즘을 개발하는 것은 다양한 실제 문제를 해결하는데 필수적이다 [1]. Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) 및 Harmony Search (HS)과 같은 다양한 메타휴리스틱 최적화 알고리즘이 개발되었다 [2-5]. Fister는 모방형태에 따라 군집지능기반 알고리즘, 생체모방기반 알고리즘, 물리화학기반 알고리즘 및 기타 알고리즘 등으로 구분하였는데 Vision Correction Algorithm (VCA)은 물리화학기반 알고리즘으로 분류될 수 있다 [1, 6].

하지만 최적화 기법의 탐색성능 (전역탐색 및 지역탐색)은 최적화 알고리즘의 매개변수 설정에 큰 영향을 받는다. 따라서 많은 연구에서 민감도 분석을 통한 적절한 매개변수를 찾는 노력들이 선행되었다. 하지만 공학 문제에 최적화 기법을 적용할 때, 적절한 매개변수 설정은 연구자 및 엔지니어들에게 매우 번거로운 작업일 뿐만 아니라 적용문제에 따라 최적의 매개변수 조합을 찾는 과정이 필요하므로 실무적용성이 다소 떨어진다. 앞서 언급한 매개변수 설정 문제를 해결하기 위해 다양한 분야에서 적절한 매개변수를 자동으로 설정할 수 있는 기법 (Self-adaptive approach)이 제안되었다 [7-9].

많은 연구자들이 GA, ACO, PSO 및 HS 등과 같은 다양한 메타휴리스틱 최적화 알고리즘들을 개량하기 위해 여러 가지 방법들을 적용하였다. 대표적으로 최근 국내외에서 많은 연구들이 수행되고 있는 HS의 전역 및 지역탐색 성능에 대한 개선과 관련된 연구들을 살펴보면 다음과 같다.

Geem (2010)은 HS의 매개변수인 Harmony Memory Considering Rate (HMCR)과 Pitch Adjust Rate (PAR)를 매회 반복시산 시 결정변수의 개수별로 자동설정 할 수 있는 Parameter-Setting-Free Harmony Search (PSF-HS)를 개발하였으며, 이때 기존 Harmony Search의 메모리에 해탐색 방법을 보관할 수 있는 Operation type memory (OTM)를 제안하였다 [10]. 또한 Jiang et al. (2013)은 PSF-HS에서 고려하지 않았던 Bandwidth (Bw) 또한 자동으로 설정 할 수 있는 Almost-parameter-free Harmony Search (APF-HS)를 개발하여 기존의 PSF-HS를 발전시켰다 [11]. Choi et al. (2017)은 다목적 최적화를 위해 Self-adaptive multi-objective Harmony Search (SaMOHS)를 개발하여 상수도관망 최적설계에 사용하였으며, 개발된 알고리즘에 다양한 성

능평가인자를 고려하여 정량적인 성능비교를 수행하였다 [12]. 하지만 앞에 언급한 연구들은 2001년에 개발된 HS의 성능개량에만 집중하였으며 알고리즘의 모든 매개변수를 사용자의 수동적인 설정없이 자동적으로 설정해주는 방법을 개발한 연구들은 없었다.

2018년에 개발된 초기 VCA는 다양한 수학 문제 및 공학 문제에서 좋은 결과를 보여주었다 [1]. 하지만 VCA를 적용하기 위해서는 여러 매개변수의 설정에 많은 시간과 노력이 필요하다. 이러한 많은 매개변수는 VCA의 사용성을 떨어뜨리는 큰 요인이며 현재 알고리즘의 형태를 더 나은 방식으로 개선할 여지가 있다. 따라서 본 연구에서는 기존 VCA의 매개변수 설정을 위한 번거로움을 없애고, 자동으로 매개변수의 설정이 가능한 Self-Adaptive Vision Correction Algorithm (SAVCA)를 개발하였다.

2. VCA

VCA는 시력교정 과정에서 착안한 메타휴리스틱 최적화 알고리즘이며 다양한 문제들에서도 좋은 결과를 보였다 [1]. 하지만 알고리즘을 구성하는 해탐색방법들이 다양하고 이들의 작동을 결정하기 매개변수들이 많아 쉽게 적용하기 어려운 문제를 가지고 있다. VCA를 구성하는 매개변수들은 Division Rate 1 (DR1), Division Rate 2 (DR2), MTF Rate (MR), Astigmatic Rate (AR), Astigmatic Factor (AF) 및 Compression Factor (CF) 등이 있다.

DR1은 새로운 렌즈 (결정변수)를 만들 때, 기존에 있는 렌즈를 바탕으로 조정할 것인지 아니면 전체 영역에서 무작위로 만들어 낼 것인지를 결정하는 매개변수다. DR2는 렌즈를 만들 때, 볼록렌즈 (근시 탐색)를 쓸 것인지 오목렌즈 (원시 탐색)를 쓸 것인지를 결정하는 매개변수다. DR1 및 DR2는 다른 메타휴리스틱 최적화 알고리즘과 다른 VCA만의 특징을 보여준다.

그 외의 VCA의 특징을 보여주는 매개변수들의 성능은 다음과 같다. MR은 렌즈의 밝기를 증대하기 위해 기존의 가장 좋은 결과를 보이는 렌즈와의 거리를 고려함으로써 알고리즘의 성능을 향상시키는 매개변수이며 AR은 렌즈의 정확한 초점을 맞추기 위해 난시 교정을 실시할 것인지에 대한 매개변수이다. AF는 난시 교정을 위해 고려하는 난시축에 관한 매개변수이며 CF는 안경의 사용성을 증대하기 위해 렌즈의 압축을 결정하는 매개변수이다.

VCA는 총 6개의 매개변수를 조절하여 최적화 과정을 진행할 수 있다. VCA를 구성하고 있는 6개의 매개변수 중 DR1 및 DR2는 최적해 탐색 중 자동적으로 설정되기 때문에 원래의 형태대로 유지하였으며 SAVCA에서는 나머지 4개의 매개변수들을 대상으로 개량을 수행하였다. 매개변수별 민감도 분석을 통하여 값의 변화에 따른 해 탐색 성능의 영향이 큰 매개변수는 연산이 진행되면서 그 값이 자동으로 조절되도록 개량하였고, 변화에 따른 해 탐색 성능의 영향이 작은 매개변수는 고정값을 갖도록 수정하였다.

3. SAVCA

기존 VCA는 DR1 및 DR2를 제외한 4개 (MR, CF, AR 및 AF) 매개변수의 민감도 분석이 필요하였지만, SAVCA는 매개변수의 변경폭이 큰 CF에 대해 자가 적응형 기법을 적용하여 자동으로 설정되도록 하였다. SAVCA는 사용성을 높이기 위해 기존 VCA의 6개의 매개변수 중 4개의 매개변수를 자가 적응형 또는 고정형으로 수정하였다. Table 1은 본 연구에서 새롭게 개발된 SAVCA의 매개변수 설정방법을 기존 VCA와 비교하였다.

Table 1. Comparison for each operator in VCA and SAVCA

Operators	VCA	SAVCA
DR1	Self-adaptive	Self-adaptive
DR2	Self-adaptive	Self-adaptive
MR	0 ~ 1	Default (0.1)
CF	0 ~ 100	Self-adaptive
AR	0 ~ 1	Default (0.1)
AF	0 ~ 180	Default (45)

Table 1에서 MR, AR 및 AF 등의 값은 결과값에 상대적으로 큰 영향을 미치지 못하므로 고정값으로 수정하였다. VCA 및 SAVCA를 구성하고 있는 매개변수 중 CF는 결정변수별 변동계수를 적용하여 계산되도록 개량하였다. 각 결정변수별 변동계수를 적용한 이유는 각각의 결정변수는 다른 평균을 가지게 되므로 표준편차의 값만으로는 상대적인 차이를 고려할 수 없기 때문이다.

새롭게 수정된 CF는 반복시산의 초반부에는 지역해에서 빠져나오기 위해 탐색의 범위를 넓히고 후반부에는 지역탐색을 강화하기 위한 조절이 필요하므로 상대적인 결정변수별 차이를 고려하기 위하여 변동계수를 적용한

CF를 사용하였다. SAVCA에서 변경된 CF는 다음 식 (1)과 같은 방식으로 계산된다.

$$CF = 10 \times \left\{ \frac{std(x_i)}{avg(x_i)} \right\} \quad (1)$$

식 (1)에서 $std(x_i)$ 는 결정변수 x_i 의 표준편차이고 $avg(x_i)$ 는 결정변수 x_i 의 평균이다. 각 결정변수는 반복시산마다 새로운 CF값을 갖게 된다. 각 매개변수들을 새롭게 수정한 SAVCA의 의사코드는 아래의 Table 2와 같다.

Table 2. Pseudo code of SAVCA

```

Begin
Objective function f(x), x = (x1, x2, ..., xd)T
Generate initial group of candidate glasses
Calculate fitness based on rank of objective value

While (t < Max number of iterations)

If (DR1 < rand)
    Choose an existing glasses according to the fitness
Else generate new glasses via randomization
If (DR2 < rand)
    Generate new decision variables in positive direction
Else generate new decision variables in negative direction
Endif

If (0.1 > rand)
    Apply MTF with compression factor
Endif

If (0.1 > rand)
    Apply astigmatism
Endif

Evaluate the fitness of new solution by new decision variables
Replace new solution with current worst solution
Endif

Find the current best solution

Endwhile
End
    
```

3. 적용결과

SAVCA의 성능을 검토하기 위하여 기존 VCA에서 적용하였던 수학 문제와 공학 문제에 적용하였다. 알고리즘에 대한 성능은 일반적으로 알고리즘의 결과값 (Mean solution, Best solution, Worst solution 및 Standard deviation)과 연산속도 (Number of Function Evaluations)로 나타낼 수 있다. 평균값 (Mean

solution)의 경우 해당 알고리즘이 평균적으로 좋은 결과를 얻을 수 있는지를 나타내는 것이며, 가장 좋은 값 (Best solution)은 해당 알고리즘이 최적해를 찾아낼 수 있는지를 판단할 수 있는 기준이 된다. 가장 나쁜 값 (Worst solution)은 해당 알고리즘이 지역해에 빠져들지 않았는지를 판단하는 기준이 되며, 표준편차 (SD: Standard deviation, 이하 SD)는 각 시행 (Simulation)마다 해당 알고리즘의 결과가 꾸준히 산정되는지를 나타낸다. 본 연구에서는 각 문제당 100회의 시행에 대한 결과를 산출하여 비교하였다. 각 문제의 최적해와 변수의 개수는 다음의 Table 3과 같다.

Table 3. Specification of problems for comparison [1]

Problems	Optimal value	Number of decision variables
Six hump camel back	-1.0316	2
Easton and fenton	1.7441	2
Schwefel	0	30
Hyper sphere	0	30
Speed reducer design	2994.47	7

우선 2개 변수를 갖는 수학 문제에 적용하였다. 비교 결과에서는 평균값, 가장 좋은 값, 가장 나쁜 값, 결과의 표준편차 및 계산횟수 (NFEs: Number of function evaluations, 이하 NFEs)를 도시하였다. NFEs는 반복 시산 횟수 (Iteration)에 시산별 생성하는 함수의 개수를 곱하여 계산된다. Six Hump Camel Back 문제는 VCA를 적용하였던 문제이며 아래의 식 (2)와 같다.

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

$$\text{subject to} \quad \begin{aligned} -10 < x_1 < 10 \\ -10 < x_2 < 10 \end{aligned}$$

(2)

식 (2)에서 x_1 및 x_2 는 Six Hump Camel Back 문제의 결정변수이다. Table 4에서는 SAVCA의 적용결과를 HS, Water Cycle Algorithm (WCA) 및 VCA와 비교하였다 [1].

Table 4. Results for application of 2D mathematical benchmark function (Six Hump Camel Back) [1]

	HS	WCA	VCA	SAVCA
Mean solution	-	-1.0316	-1.0316	-1.0316
Best solution	-1.0316	-1.0316	-1.0316	-1.0316
Worst solution	-	-1.0316	-1.0316	-1.0316
SD	-	7.89E-10	0.00E+00	0.00E+00
NFEs	4,870	3,105	935	531

Six Hump Camel Back 문제의 적용결과에서 보면 HS, WCA, VCA 및 SAVCA 모두 최적해인 -1.0316을 찾는 것을 볼 수 있다. 하지만 최적해로 도달하는데 소요된 NFEs를 보면 다른 알고리즘에 비해 SAVCA가 가장 낮은 값을 보여주고 있다. 이 결과는 Six Hump Camel Back 문제에서 SAVCA가 가장 좋은 성능 나타내고 있다는 것을 의미한다. SAVCA는 빠르게 최적해로 수렴할 뿐만 아니라 매개변수의 설정 또한 필요없기 때문에 실무 적용성 측면에서 향상되었다고 볼 수 있다.

두 번째 2개 변수를 갖는 수학 문제로 선택된 Easton and Fenton 문제는 앞서 Six Hump Camel Back 문제와 마찬가지로 VCA를 적용하였던 문제이며 식 (3)과 같다.

$$f(x) = \left\{ 12 + x_1^2 + \frac{1 + x_2^2}{x_1^2} + \frac{x_1^2 x_2^2}{(x_1 x_2)^4} \right\} \left(\frac{1}{10} \right)$$

$$\text{subject to} \quad \begin{aligned} -10 < x_1 < 10 \\ -10 < x_2 < 10 \end{aligned}$$

(3)

식 (3)에서 x_1 및 x_2 는 Easton and Fenton 문제의 결정변수이다. Table 5에서는 SAVCA의 적용결과를 HS, WCA 및 VCA와 비교하였다 [1].

Eaton and Fenton 문제의 적용결과에서 보면 HS, WCA, VCA 및 SAVCA 모두 최적해인 1.7441을 찾는 것을 볼 수 있다. 하지만 최적해로 도달하는데 소요된 NFEs를 보면 다른 알고리즘에 비해 SAVCA가 가장 낮은 값을 보여주고 있다. 이 결과는 Eaton and Fenton 문제에서 SAVCA가 가장 좋은 성능을 보인다는 것을 의미한다.

Table 5. Results for application of 2D mathematical benchmark function (Easton and Fenton) [1]

	HS	WCA	VCA	SAVCA
Mean solution	-	1.7441	1.7441	1.7441
Best solution	1.7441	1.7441	1.7441	1.7441
Worst solution	-	1.7441	1.7441	1.7441
SD	-	1.96E-06	0.00E+00	0.00E+00
NFEs	800	650	328	189

종합적으로 2개 변수를 갖는 수학 문제에 적용하여 결과를 HS, WCA 및 VCA에 결과와 비교하였을 때 SAVCA가 가장 빠르게 최적해를 찾는 것을 확인할 수 있었다. 추가적인 수학 문제로 30개 변수를 갖는 수학 문제들에 적용하였다. 30개 변수를 갖는 수학 문제들은 앞서 적용하였던 2개 변수를 갖는 수학 문제들에서와 다르게 고정적인 연산종료 조건인 120,000의 NFEs를 기준으로 하여 최적해에 대한 결과를 산출하였다. Schwefel 문제는 VCA를 적용하였던 수학 문제이며 식 (4)와 같다.

$$f(x) = 418.9829 \times N - \sum_{i=1}^N x_i \sin(\sqrt{|x_i|})$$

$$\text{subject to} \quad -500 < x_i < 500,$$

$$i = 1, \dots, 30$$

(4)

식 (4)에서 x_i 는 Schwefel 문제의 결정변수이다. Table 6에서는 SAVCA의 적용결과를 Genetic Algorithms with Floating-point representation (GAF), Shuffled Complex Evolution algorithm (SCE-UA), Modified Shuffled Complex Evolution algorithm (MSCE) 및 VCA와 비교하였다 [1].

Schwefel 문제의 적용결과에서 보면 GAF, SCE-UA, MSCE, VCA 및 SAVCA 모두 최적해인 0에 도달하지 못하고 있다. 하지만 VCA 및 SAVCA에서 가장 근접한 값인 3.81E-04를 얻은 것을 볼 수 있다. 이 결과는 Schwefel 문제에서 VCA 및 SAVCA가 좋은 성능을 보인다는 것을 의미한다.

Table 6. Results for application of 30D mathematical benchmark function (Schwefel) [1]

	GAF	SCE-UA	MSCE	VCA	SAVCA
Mean solution	5434.8	8042.6031	1.5598	3.81E-04	3.81E-04
Best solution	3987.9	7394.4199	0.1072	3.81E-04	3.81E-04
Worst solution	6219.6	8594.3853	6.1420	3.81E-04	3.81E-04
SD	552.3	228.5129	1.4026	0.00E+00	0.00E+00
NFEs	120,000	120,000	120,000	120,000	120,000

두 번째 30개 변수를 갖는 수학 문제로 선택된 Hyper Sphere 문제는 앞서 Schwefel 문제와 마찬가지로 VCA를 적용하였던 문제이며 식 (5)와 같다.

$$f(x) = \sum_{i=1}^N x_i^2$$

$$\text{subject to} \quad -5.12 < x_i < 5.12,$$

$$i = 1, \dots, 30$$

(5)

식 (5)에서 x_i 는 Hyper Sphere 문제의 결정변수이다. Table 7에서는 SAVCA의 적용결과를 GAF, SCE-UA, MSCE 및 VCA와 비교하였다 [1].

Table 7. Results for application of 30D mathematical benchmark function (Hyper Sphere) [1]

	GAF	SCE-UA	MSCE	VCA	SAVCA
Mean solution	4.83E-05	5.92E-12	0.00E+00	0.00E+00	0.00E+00
Best solution	9.56E-11	3.48E-16	0.00E+00	0.00E+00	0.00E+00
Worst solution	2.29E-04	5.97E-11	0.00E+00	0.00E+00	0.00E+00
SD	4.29E-05	1.212E-11	0.00E+00	0.00E+00	0.00E+00
NFEs	120,000	120,000	120,000	120,000	120,000

Hyper Sphere 문제의 적용결과에서 보면 GAF 및 SCE-UA를 제외한 MSCE, VCA 및 SAVCA에서 최적값인 0을 보여주고 있다. 이 결과는 Hyper Sphere 문제에서 MSCE, VCA 및 SAVCA가 좋은 성능을 보인다는 것을 의미한다.

종합적으로 30개 변수를 갖는 수학 문제에 적용하여 결과를 GAF, SCE-UA, MSCE 및 VCA의 결과와 비교하였을 때 VCA 및 SAVCA가 정확하게 최적값을 찾는 것을 확인할 수 있었다. 마지막으로 실제 공학 문제에서의 검증에 위해 적용한 Speed reducer design 문제는 식 (6)과 같다.

$$\begin{aligned} \text{Min } f(x) = & 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 \\ & - 43.0934) - 1.508x_1(x_6^2 + x_7^2) \\ & + 7.4777(x_6^3 + x_7^3) \\ & + 0.7854(x_4x_6^2 + x_5x_7^2) \end{aligned}$$

subject to

$$g_1(x) = \frac{27}{x_1x_2^2x_3} - 1 \leq 0$$

$$g_2(x) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \leq 0$$

$$g_3(x) = \frac{1.93x_4^3}{x_2x_6^4x_3} - 1 \leq 0$$

$$g_4(x) = \frac{1.93x_5^3}{x_2x_7^4x_3} - 1 \leq 0$$

$$g_5(x) = \frac{[(745(x_4/x_2x_3))^2 + 16.9 \times 10^6]^{1/2}}{110x_6^3} - 1 \leq 0$$

$$g_6(x) = \frac{[(745(x_5/x_2x_3))^2 + 157.5 \times 10^6]^{1/2}}{85x_7^3} - 1 \leq 0$$

$$g_7(x) = \frac{x_2x_3}{40} - 1 \leq 0$$

$$g_8(x) = \frac{5x_2}{x_1} - 1 \leq 0$$

$$g_9(x) = \frac{x_1}{12x_2} - 1 \leq 0, g_{10}(x) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0$$

$$g_{11}(x) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0$$

where

$$\begin{aligned} 2.6 \leq x_1 \leq 3.6, \quad 0.7 \leq x_2 \leq 0.8, \quad 17 \leq x_3 \leq 28, \\ 7.3 \leq x_4 \leq 8.3, \quad 7.3 \leq x_5 \leq 8.3, \quad 2.9 \leq x_6 \leq 3.9, \\ 5.0 \leq x_7 \leq 5.5 \end{aligned}$$

(6)

식 (6)에서 $x_1, x_2, x_3, x_4, x_5, x_6$ 및 x_7 는 Speed reducer design 문제의 결정변수이다. Table 8에서는 SAVCA의 적용결과를 Particle Swarm Optimization with Differential Evolution (PSO-DE), Mine Blast Algorithm (MBA) 및 VCA와 비교하였다 [1].

Table 8. Results for application of engineering function (Speed reducer design) [1]

	PSO-DE	MBA	VCA	SAVCA
Mean solution	2996.482	2996.690	2994.673	2994.567
Best solution	2996.348	2994.482	2994.475	2994.472
Worst solution	2996.348	2996.769	2995.515	2995.339
SD	6.40E-06	1.56	2.65E-01	1.50E-01
NFEs	54.456	6,300	50,000	50,000

Speed reducer design 문제의 적용결과에서 보면 SAVCA에서 가장 좋은 값인 2994.472를 보여주고 있다. 평균값이나 가장 나쁜 값에서도 비교한 다른 알고리즘들에 비해 좋은 성능을 보여주고 있다. 이 결과는 수학 문제 뿐만 아니라 공학 문제인 Speed reducer design 문제에서도 SAVCA가 좋은 성능을 보인다는 것을 의미한다.

4. 결론

본 연구에서는 기존에 개발된 VCA의 단점인 많은 매개변수 설정의 문제를 개선하기 위하여 SAVCA를 개발하였다. SAVCA의 개발은 기존 연구에서 좋은 성능을 보여주었던 VCA를 사용하고자 하는 연구자들 및 엔지니어들이 실무에 쉽게 적용할 수 있게 하기 위해 진행되었다.

SAVCA는 기존 6개의 매개변수 중 자기 적응형으로 구성되어 있는 DR1 및 DR2를 제외한 4개 (MR, CF, AR 및 AF)를 자가 적응형 또는 고정형으로 수정하여 사용성을 높이고자 하였다. 기 개발된 VCA에 비해서 다양한 수학 문제 (2개 변수 및 30개 변수 수학 문제) 및 공학 문제에서 전반적으로 좋은 성능을 보여주는 것을 알 수 있었으며 비교한 다른 알고리즘들에 비해 좋은 성능을 보여주었다.

SAVCA의 개발은 VCA의 성능향상을 위한 초석이 될 것이며 다양한 후속 연구가 이어질 수 있다. 간단한 프로그램 코딩을 위해 6개의 매개변수를 압축하는 과정이 연구될 수 있으며 다목적 최적화를 적용한다면 다양한 공학 문제에 사용성과 성능이 뛰어난 메타휴리스틱 알고리즘의 적용이 가능할 것으로 예측된다.

References

- [1] E. H. Lee, D. G. Yoo, Y.H. Choi, J. H. Kim, "Development of the new meta-heuristic optimization algorithm inspired by a vision correction procedure: Vision Correction Algorithm", *Journal of the Korea Academia-Industrial cooperation Society*, Vol.17, No.3, pp.117-126, March 2016.
DOI: <https://dx.doi.org/10.5762/KAIS.2016.17.3.117>
- [2] D. E. Goldberg, J. H. Holland, "Genetic Algorithms and Machine Learning", *Machine Learning*, Vol.3, Issue2-3, pp.95-99, Oct. 1988.
DOI: <https://doi.org/10.1023/A:10226020>
- [3] M. Dorigo, *Optimization, learning and natural algorithms*, Ph. D. Thesis, Politecnico di Milano, Italy, 1992.
- [4] J. Kennedy, R. Eberhart, "Particle swarm optimization", *In Neural Networks, 1995. Proceedings, IEEE International Conference on*, IEEE, Perth, Australia Vol.4, pp.1942-1948, Nov. 1995.
DOI: https://doi.org/10.1007/978-0-387-30164-8_630
- [5] Z. W. Geem, J. H. Kim, G. V. Loganathan, "A new heuristic optimization algorithm: harmony search", *Simulation*, Vol.76, Issue2, pp.60-68, Feb. 2001.
DOI: <https://doi.org/10.1177/003754970107600201>
- [6] I. Fister Jr., X.S. Yang, I. Fister, J. Brest, D. Fister, "A Brief Review of Nature-Inspired Algorithms for Optimization", *Elektrotehniški vestnik*, Vol.80, No.3, pp.1-7, July 2013.
DOI: <https://arxiv.org/abs/1307.4186>
- [7] K. Deb, H. G. Beyer, "Self-adaptive genetic algorithms with simulated binary crossover", *Evolutionary computation*, Vol.9, Issue2, pp.197-221, March 2001.
DOI: <https://doi.org/10.1162/106365601750190406>
- [8] A. Ismail, A. P. Engelbrecht, "Self-adaptive particle swarm optimization" *In Asia-Pacific Conference on Simulated Evolution and Learning*. Springer, Berlin, Heidelberg, pp.228-237, Dec. 2012.
DOI: https://doi.org/10.1007/978-3-642-34859-4_23
- [9] M. G. Omran, A. Salman, A. P. Engelbrecht, "Self-adaptive differential evolution", *International Conference on Computational and Information Science*. Springer, Berlin, Heidelberg, pp.192-199, Dec. 2005.
DOI: https://dx.doi.org/10.1007/11596448_28
- [10] Z. W. Geem, "Parameter estimation of the nonlinear Muskingum model using parameter-setting-free harmony search", *Journal of Hydrologic Engineering*, Vol.16, Issue8, pp.684-688, Aug. 2011.
DOI: [https://doi.org/10.1061/\(ASCE\)HE.1943-5584.0000352](https://doi.org/10.1061/(ASCE)HE.1943-5584.0000352)
- [11] S. Jiang, Y. Zhang, P. Wang, M. Zheng, "An almost-parameter-free harmony search algorithm for groundwater pollution source identification", *Water Science and Technology*, Vol.68, Issue11, pp.2359-2366, Oct. 2013.
DOI: <https://doi.org/10.2166/wst.2013.499>
- [12] Y. H. Choi, H. M. Lee, D. G. Yoo, J. H. Kim, "Self-adaptive multi-objective harmony search for optimal design of water distribution networks" *Engineering Optimization*, Vol.49, Issue11, pp.1957-1977, 2017.
DOI: <https://doi.org/10.1080/0305215X.2016.1273910>

이 의 훈(Eui Hoon Lee)

[종신회원]



- 2003년 2월 : 고려대학교 토목환경공학과 (공학사)
- 2005년 2월 : 고려대학교 토목환경공학과 (공학석사)
- 2017년 2월 : 고려대학교 고려대학교 건축사회환경공학과 (공학박사)
- 2018년 9월 ~ 현재 : 충북대학교 토목공학부 교수

<관심분야>

도시수문학, 관망수리학, 수자원시스템공학

이 호 민(Ho Min Lee)

[정회원]



- 2012년 2월 : 고려대학교 사회환경시스템공학과 (공학사)
- 2017년 8월 : 고려대학교 건축사회환경공학과 (공학박사)
- 2017년 9월 ~ 현재 : 고려대학교 방재과학기술연구소 연구교수

<관심분야>

수리학, 수자원공학, 최적화 알고리즘

최 영 환(Young Hwan Choi)

[정회원]



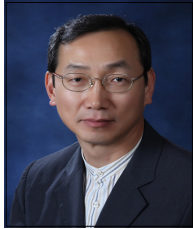
- 2010년 2월 : 한양대학교 건설환경시스템공학과 (공학사)
- 2018년 8월 : 고려대학교 건축사회환경공학과 (공학박사)
- 2018년 9월 ~ 현재 : 고려대학교 방재과학기술연구소 연구교수

<관심분야>

수리학, 수자원공학, 최적화 알고리즘

김 중 훈(Joong Hoon Kim)

[정회원]



- 1984년 2월 : 고려대학교 토목공학과 (공학사)
- 1986년 12월 : Virginia Polytechnic Institute & State Univ. (공학석사)
- 1992년 2월 : Univ. of Texas at Austin (공학박사)

• 1993년 3월 ~ 현재 : 고려대학교 건축사회환경공학부 교수

<관심분야>

도시수문학, 관망수리학, 수자원시스템공학