JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

An Improved Harmony Search Algorithm and Its Application in Function Optimization

Zhongda Tian* and Chao Zhang*

Abstract

Harmony search algorithm is an emerging meta-heuristic optimization algorithm, which is inspired by the music improvisation process and can solve different optimization problems. In order to further improve the performance of the algorithm, this paper proposes an improved harmony search algorithm. Key parameters including harmonic memory consideration (HMCR), pitch adjustment rate (PAR), and bandwidth (BW) are optimized as the number of iterations increases. Meanwhile, referring to the genetic algorithm, an improved method to generate a new crossover solutions rather than the traditional mechanism of improvisation. Four complex function optimization and pressure vessel optimization problems were simulated using the optimization algorithm of standard harmony search algorithm, improved harmony search algorithm and exploratory harmony search algorithm. The simulation results show that the algorithm improves the ability to find global search and evolutionary speed. Optimization effect simulation results are satisfactory.

Keywords

Complex Functions, Harmony Search, Intelligent Algorithm, Optimization

1. Introduction

Harmony Search (HS) algorithm is a swarm intelligence optimization algorithm, it comes from the creative process of music, and proposed in 2001 [1]. All individuals with harmony search algorithm to produce a new individual, does not depend on the initial conditions, and has a simple structure, easy and fast convergence characteristics achieved [2]. In [3] and [4], the authors show a search algorithm to optimize performance better harmony genetic algorithms and simulated annealing algorithms and other optimization algorithms ratio. Harmony search algorithms are widely used in power system optimization [5], economic cost optimization [6], routing optimization for wireless sensor networks [7], architectural design [8] and other practical issues [9-11].

Although harmony search algorithm has better global optimization capability, its disadvantages are randomness and instability. At the same time, search direction of the algorithm is uncertainty. So many scholars have made some improvements on the standard HS algorithm. Mahdavi et al. [12] propose an improved harmony search algorithm (IHS). At IHS, the bandwidth parameters are designed to exponentially decrease with an iterative increase. In [13], the authors proposed a global harmony search algorithm based on the improved search mechanism of the standard HS algorithm. Experimental

results show that the algorithm than the standard HS algorithm significantly better. Fesangharya et al. [14] proposed a hybrid and acoustic search algorithm, which combines the advantages of the sequential quadratic programming algorithm with the acoustic algorithm. It has fast seeking solution ability for the optimization problems. In [15], the authors analyzed and improved search ability of HS algorithm, which named as explorative harmony search algorithm (EHS). In EHS, the band width is adjusted by variance information of harmony variables. The explore ability of the algorithm is improved. The author proposes a novel global HS algorithm based on the standard global HS algorithm [16,17] and an improved global optimal HS algorithm [18]. Some researchers have developed a number of improved harmony search algorithm to determine the value of overcoming the bandwidth difficult limitations [19-21]. We propose an IHS algorithm [22] three main features. The numerical results confirm that the IHS algorithm has advantages in accuracy, convergence speed and robustness.

These IHS algorithms are only limited to the single parameter optimization. They can't achieve the overall performance enhancement. The HS algorithm has three key parameters. These parameters are harmony memory considerations, pitch adjustment rates, and bandwidth. In this paper, these parameters are optimized simultaneously. Meanwhile, the standard algorithm has randomness in the process of new solutions generation. This paper presents a new and improved method of generating solutions, it refers to a genetic algorithm. The results show that the IHS algorithm proposed in this paper has better performance, faster convergence can solve optimization problems.

The remainder of this paper is organized as follows. Section 2 describes the basic steps to achieve HS algorithm. Section 3 presents a number of improvements. In Section 4, the experimental results show, then the conclusion Section 5.

2. Harmony Search Algorithm

The following describes the process of standard HS optimization algorithm. In view of the optimization problem follows.

$$\min f(X),$$
$$s.t.\ x_i \in [L_i, U_i].$$

Wherein the dimension is a real-valued vector, it is a real continuous function.

Step 1: HS initialization parameter algorithm. Acoustic parameters include memory size (HMS), consider the sound memory rate (HMCR), tone adjustment ratio (PAR), the bandwidth (BW), the number of iterations (NI), a range between the decision variables.

Step 2: Randomly create harmony memory (HM) as follows.

$$HM = \begin{bmatrix} X_1 \\ X_2 \\ \cdot \\ \cdot \\ X_m \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}; m = HMS. \tag{1}$$

Step 3 : Produce a candidate solution. $X_{new} = (x_{new}(1), x_{new}(2), \cdots, x_{new}(j))$, $x_{new}(j)$ is generated by following steps.

$$x_{new}(j) = \begin{cases} x_{md(i),j} & r_1 < HMCR \\ x_{new}(j) \in [L_i, U_i] & \text{others} \end{cases}, \qquad (2)$$

where, $x_{md(i),j}$ is the jth row component of the harmony memory in randomly, $x_{new}(j)$ is a random value of the jth variable, r_1 is a random number uniformly distributed between [0, 1].

Then, if $x_{new}(j)$ is the solution of the components selected from HM, minor adjustments are as follows.

$$x_{new}(j) = \begin{cases} x_{new}(j) \pm r_3 \times BW & , r_2 < PAR \\ x_{new}(j) & \text{others} \end{cases}, \qquad (3)$$

where, r_2 and r_3 is random number which uniformly distributed between [0, 1].

Step 4 : Update the HM, if $f(X_{new}) < f(X_w)$, X is the worst solution in the harmony memory, then $X_w = X_{new}$.

Step 5 : Repeat steps 3 and 4 until the number of iterations reaches NI.

3. Improved Harmony Search Algorithm

3.1 Improvement of Algorithm

From standard harmony search algorithm, we can know HMCR and PAR algorithm can help find global and local solution. PAR and BW is the harmony pruning process two very important parameters. In the early of the optimization process, maintain a smaller PAR value and larger BW value are conducive to enhancing the diversity of the solution vector, which can quickly find local optimal value. On the other hand, in the latter of the optimization process, smaller BW and larger PAR value is conducive to quickly find the global optimal solution. Standard harmony search algorithm uses a fixed parameter value. It is not taking into account the requirements of the local optimal solution and the global optimal solution. At the same time, the search speed and convergence precision of the harmony search algorithm are related to the parameter selection of the algorithm. In order to improve the efficiency of the search algorithm, harmony search algorithm to overcome the lack of standards and improved as follows.

The HMCR is a probability of selecting one value from historical values stored in the HM, and (1-HMCR) is a probability of randomly selecting one feasible value, which is not limited to the value stored in the HM. Note that, considering the memory and sound based on random selection, the opposite vector from the new acoustic sound memory. Harmony part against the vector element selected from the opposite of the sound memory, and to rest against randomly selected from a given sound field

vector. If appropriate increase in the value of HMCR, it may help to local contraction algorithm. However, smaller HMCR value will increase the diversity of new solutions. With the increase HMCR value, the HS algorithm performance is improved. HMCR small value results in performance degradation HS algorithm. The main reason is HMCR is to select a value in the history value in the HM from the storage rate and 1-HMCR rate is a value selected from a range of possible values at random. Small value HMCR resulted only from HM select few elite harmony, and the algorithm performance closer to blind random search. As mentioned above, HMCR should be dynamic adjustment from largest to smallest. This makes complete harmony search algorithm first searches in harmony memory, and then transferred to harmony after iteration external memory, can increase the diversity of the population. Adjusting method is as follows.

$$HMCR(t) = \begin{cases} HMCR(t-1) \times \rho & HMCR_{max} > HMCR(t) > HMCR_{min} \\ HMCR_{min} & HMCR(t) \le HMCR_{min} \\ HMCR_{max} & t = 0 \end{cases} , \qquad (4)$$

where, ρ is a scale parameter. The reduction rate of HMCR can be controlled by this parameter ρ. $HMCR_{max}$ is maximum value of HMCR. $HMCR_{min}$ is minimum value of HMCR. t is iterations.

The changing curve of HMCR with different ρ can be shown in Fig. 1. Because HMCR in standard harmony search algorithm is from 0 to 1, $HMCR_{max}$ and $HMCR_{min}$ are also changes from 0 to 1. The scale parameter ρ controls the shape of the curve. It can be seen in Fig. 1, HMCR will dynamic adjustment from small to big when $\rho > 1$, dynamic adjustment from big to small to when $0 < \rho < 1$, constant value when $\rho = 1$. From the above analysis, HMCR in this paper should change from big to small. Meanwhile, too small parameters ρ will lead to the decline of HMCR too fast, resulting in the existence of blindness search. This paper suggests parameters satisfy $0.9 < \rho < 1$, $0 < HMCR_{max} < 1$, $0.1 < HMCR_{min} < 0.5$.
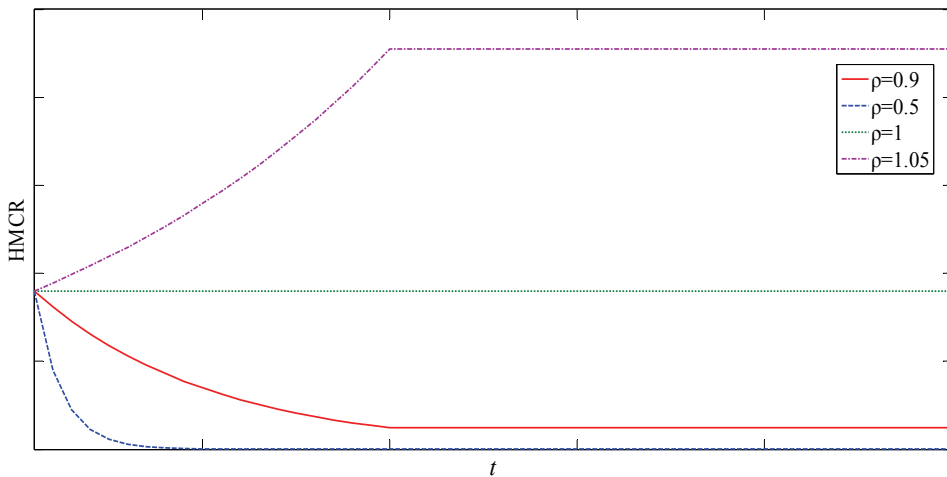

Fig. 1. The changing curve of HMCR.

HS algorithm for solving optimization PAR and BW is an important parameter vector fine-tuning, can be used to adjust the convergence speed of the algorithm to the optimal solution. The traditional PAR and BW HS algorithm uses a fixed value. In the HS algorithm, the adjustment in the initialization step PAR and BW values can not be changed during the next generation. The main drawback of this method appears to find the number of iterations required for optimal solutions in the algorithm. Small BW PAR value has a large value may result in poor performance of the algorithm as well as a significant increase needed to find the best solution for iteration. Despite the small value of the last generation of BW increase understanding trimming vector, but in the early BW, you must use a larger value to force the algorithm to increase the diversity of the solution vector. Moreover, the large PAR value having a value generally results in a small BW optimal solution to improve the final generation, and the algorithm converges to the optimal solution vector. In order to improve the performance of the algorithm and to eliminate the disadvantages HS HMCR fixed and PAR values, Mahdavi et al. [12] proposed an improved harmony search algorithm using the variable BW PAR and improvisation step. The improved HS algorithm proposed in this paper has exactly the same steps of classical HS with exception that Step 3, where the IHS dynamically updates PAR and BW with iteration numbers.

In the early HS algorithm, the smaller the value of PAR in favor of fast search algorithm better area. In the late HS algorithm, the larger PAR values conducive to escape from the local optimum value, so PAR value should be small to large. This paper follows the change of PAR.

$$PAR(t) = \frac{PAR_{\max} - PAR_{\min}}{\sqrt{NI}} \times \sqrt{t} + PAR_{\min}, \tag{5}$$

where, $PAR_{\max}$ is maximum value of PAR, $PAR_{\min}$ is minimum value of PAR. t is iterations.

PAR variation curve can be seen from Fig. 2, PAR value will change from small to large. From the above analysis, PAR should take a small value in the initial search phase and take a large value in the later search phase. If the difference between $PAR_{\max}$ and $PAR_{\min}$ is too smaller, the algorithm is difficult to jump out of local optimum value. This paper suggests parameters satisfy $0.8 < PAR_{\max} < 1$, $0.2 < PAR_{\min} < 0.4$, $0.4 < PAR_{\max} - PAR_{\min} < 0.6$.
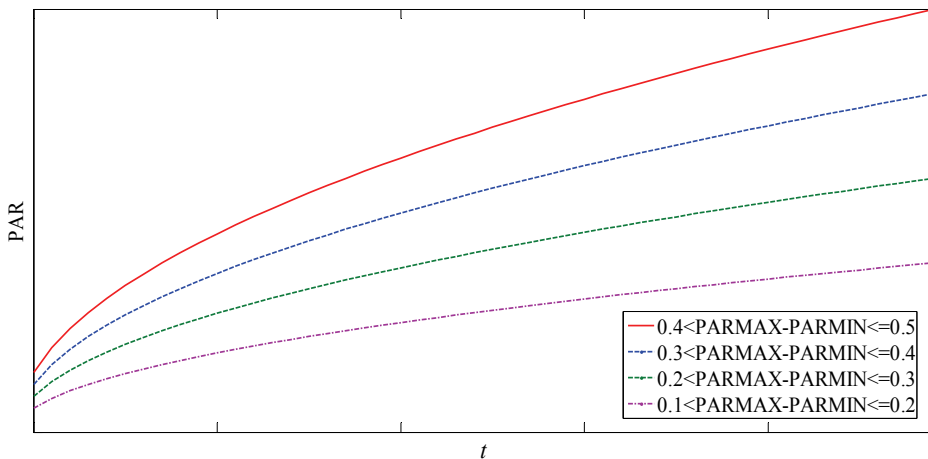

Fig. 2. The changing curve of PAR.

For BW, in the early stage of the algorithm, the larger BW conducive to harmony search algorithms can search over a wide range. In the latter time of the algorithm search process, the smaller BW will help HS algorithm has precise search ability within a small range, so the changing of BW value should be from small to large. The change formula of BW in this paper is as follows.

$$BW(t) = BW_{\min} + (BW_{\max} - BW_{\min}) \times e^{-t}, \tag{6}$$

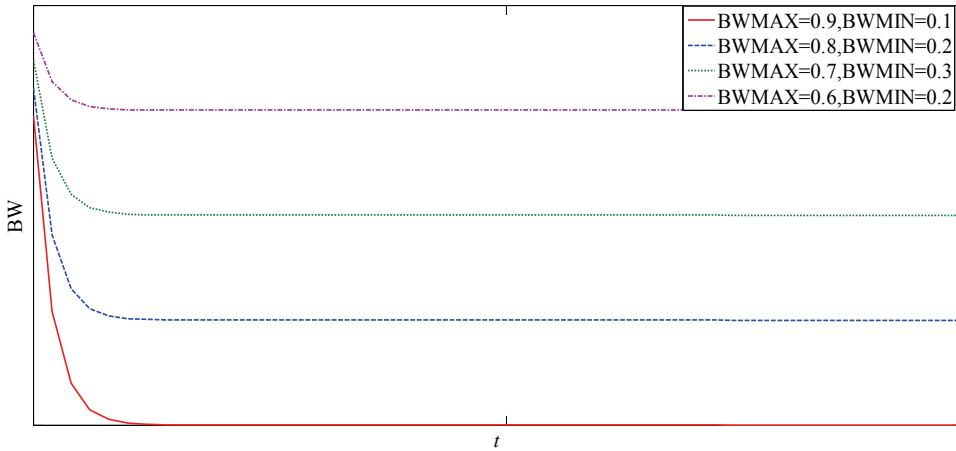where, $BW_{\max}$ is maximum value of BW. $BW_{\min}$ is minimum value of BW. t is iterations.

Fig. 3. The changing curve of BW.

In this paper, the changing curve of BW is showed in Fig. 3. It can be seen from Fig. 3, the different value of $BW_{\max}$ and $BW_{\min}$ will affect the shape of the curve. In the standard HS algorithm, BW always takes a small value close to 0, which can improve search ability in the small range. In order to balance the search ability in the small range and detect ability in the wide range. The BW should take a larger value in the initial search phase to obtain a global optimal value, and then quickly reduced to around 0. This paper suggests parameters satisfy $0.9 < BW_{\max} \leq 1$, $0 < BW_{\min} < 0.1$.

When standard HS algorithm updates memory, new solution x_{new} is generated through random selection of a component in the memory. This will cause much problem for algorithm search process, such as search direction uncertain, greater randomness. Crossover genetic algorithm (GA) is a method of exchanging the two individual genes. In this way, two new individuals created, and then use them instead of the original two individuals. CROSS mathematical operation is one of a real value in GA CROSS. If the parents of two individuals are x_{new1} and x_{new}, respectively. Then the offspring variable x_{new2} is the random linear interpolation result of x_{new1} and x_{new}. This reference to crossover idea GA algorithm to generate a new solution x_{new1} based on standard HS algorithm. The new solution process is as follows.

$$x_{new2} = u x_{new1} + (1-u) x_{new}, \tag{7}$$

where, u is a scale factor. If $f(x_{new1}) < f(x_{new2})$, x_{new1} is reserved. Otherwise, x_{new2} is reserved. It will keep the deterministic search direction.

According to the Eq. (7), the second new solution x_{new2} is generated through x_{new1} cross operation with another column component x_{new}. This action applies to each individual variable and then gets a new entity with its parent function. It can ensure that the proposed algorithm has strong global search capability during the initial optimization, has strong local search capability in the late optimization.

3.2 Proposed Performance Analysis of the Algorithm

The HMCR, PAR and BW coordinated search algorithms are very important parameters for obtaining the optimal solution vector and can be used to adjust the convergence speed of the algorithm to the optimal solution. Therefore, the fine-tuning of these parameters have a lot of interest. The traditional HS algorithms use a fixed value for HMCR, PAR, and BW. In the standard algorithm, the HMCR, PAR, and BW values are adjusted during the initialization step and cannot be changed during the new generation. The main drawback of this method appears to find the number of iterations required for optimal solutions in the algorithm. PAR value has a small and large BW HMCR values can lead to poor performance of the algorithm as well as a significant increase needed to find the best solution for iteration. Despite the small value of the last generation of BW increase understanding trimming vector, but must use a larger value in the early stages of BW and HMCR to force the algorithm to increase the diversity of solution vectors. Moreover, the large PAR value having a value generally results in a small BW optimal solution to improve the final generation, and the algorithm converges to the optimal solution vector. On the other hand, according to the mutation mechanism genetic algorithm to generate a new generation method for solving avoid the randomness of the original HS algorithm. The new solution improves the method for generating the initial optimized global search ability and local search capability in optimizing the later stages.

This paper will briefly discuss the space and time complexity of the proposed algorithm. The number of parameters to be set in the standard HS algorithm in [1] is 4. The number of parameters to be set in the IHS algorithm in [12] is 5. The number of parameters to be set in the EHS algorithm in [15] is 4. The number of parameters to be set in proposed algorithm of this paper is 9. In terms of the number of parameters that need to be set, the algorithm proposed in this paper need more space. However, the method of generating the harmony memory in these algorithms is the same, so in fact the space complexity of several algorithms is not different. With a little more space in exchange for optimization performance is very worthwhile. From the point of view of the time complexity of these HS algorithms, the computation of the initialization of the harmony memory is the same in the key steps of the algorithm. However, in the process of creating a new solution through improvisation, the improved algorithm in this paper has the mutation operation. In the EHS algorithm, there are many square, mean and variance calculations. So EHS algorithm needs more computing time. Because the mutation operation, the time complexity of the proposed algorithm is close to IHS algorithm and some more complex than standard HS algorithm. But on the whole, the time complexity of these algorithms is not much difference. The simulation results of this paper verify this conclusion.

Next, we will discuss the convergence of IHS algorithm is proposed. Compared with other types sound search algorithm, the improved algorithm proposed in this paper introduces a new method of

generating a new mutation operator via a solution. And by generating a relatively new process of adaptation solutions, generate new solutions through optimal mechanism. We can guarantee monotonically increasing manner to generate new solutions. Therefore, the essence of IHS algorithm is proposed based on greedy selection strategy, we can guarantee the convergence of the algorithm.

3.3 The Implementation Steps of the Algorithm

Step harmony to achieve improved search algorithm presented herein may be described as follows.

Step 1. Initialize the problem and algorithm parameters.

Step 2. Initialize the harmony memory.

Step 3. Improvise a new harmony. Generating a new harmony based on three rules: memory considerations, tone adjustment and random selection. Specific operation is as follows.

for (j=1 to n) do

   if (r_1 < HMCR) then

      $x_{new}(j) = x_{md(i),j}$ where $md \in (1, 2, \ldots, \text{HMS})$

     if (r_2 < PAR) then

       $x_{new}(j) = x_{new}(j) \pm r_3 \times BW$

     endif

   else

     $x_{new2} = ux_{new1} + (1-u)x_{new}$

  endif

endfor

Step 4. Update harmony memory.

Step 5. Adjust HMCR according to Eq. (4), PAR according to Eq. (5) and BW according to Eq. (6). Check stopping criterion.

4. Simulation

4.1 Complex Function Optimization Simulation

Optimize performance IHS algorithm in order to verify the proposed, we choose the next four complicated functions: Rastrigin, Griewank, Ackley, and Schwefel function studies [23]. With increasing functional size of an increasing number of these features local optima. Global difficult to find the optimal solution [24]. Expressions four functions as follows.

Rastrigin features:

$$f_1(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right]. \tag{8}$$

Griewank function:

$$f_2(x) = \frac{1}{4000} \sum_{i=1}^{n} [x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + 10]. \tag{9}$$

Ackley function:

$$f_3(x) = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^{n} \cos 2\pi x_i) + 20 + e. \tag{10}$$

Rosenbrock function:

$$f_4(x) = \sum_{i=1}^{n} 100(x_{i+1}^2 - x_i)^2 + (1 - x_i)^2. \tag{11}$$

In order to optimize the effect described herein the improved search algorithm, HS criteria document [1], the IHS algorithm [12], and the EHS algorithm [15] were compared. Each HS algorithm parameters and reference parameters in the respective same, which ensures fairness simulation. The standard HS algorithm parameters are: $HMS = 15$, $HMCR = 0.9$, $PAR = 0.3$, $BW = 0.01$. The parameters of the IHS algorithm [12] are: $HMS = 6$, $HMCR = 0.95$, $PAR_{max} = 0.99$, $BW_{max} = 0.5$, $BW_{min} = 0.0001$. The parameters of the EHS algorithm [15] are: $HMS = 15$, $HMCR = 0.99$, $PAR = 0.33$, k=1.17. The parameters of IHS algorithm in this paper are: $HMCR_{max} = 0.99$, $HMCR_{min} = 0.4$, $\rho = 0.97$, $PAR_{max} = 0.9$, $PAR_{min} = 0.4$, $BW_{min} = 0.0001$, $BW_{max} = 1$, $HMS = 6$, Cross-factor u is 0.8. Four algorithms iterations are set as $NI = 5000$. Four iterative algorithm to $NI = 5000$. Table 1 shows the dimensions of four test functions, and the global optimal initialization parameter values.

Table 1. Function parameters

Functions	Dimensions	Range of parameters	Optimal value	Permissible error
f_1	30	[-5.12, 5.12]	0	15
f_2	30	[-500, 500]	0	10
f_3	30	[-30, 30]	0	4
f_4	30	[-2.048, 2.048]	0	10

In order to eliminate the effects of randomness, all algorithms run 20 times. Selecting the average value as a result optimization, the fitness function testing processes of four convergence curves are shown in Figs. 4–7. 5000 is considered as the number of iterations, for ease of illustration, the horizontal ordinate adaptation values recorded every 50 iterations. Thus the level of the ordinate is a function of the range of 0 to 100. The value for the optimization function, the degree of adaptation is different values of x. Since the range of variation (fitness) function value is too large; longitudinal coordinate of the use forms 10. The optimization problem in this paper is to find the global minimum of the test function through the harmony search algorithm, so as the number of iterations increases, the curves in Figs. 3 and 4 are also the same. Figs 4–7 are monotonically decreasing. When the value of the function to find the global optimum adaptation curve remains unchanged. It should be noted that some random factors present optimization process, so there are some differences in the shape of the curve of each optimization. As can be seen from these figures, the IHS algorithm is better than the standard algorithm, the IHS algorithm and the EHS algorithm converges faster and have better fitness.
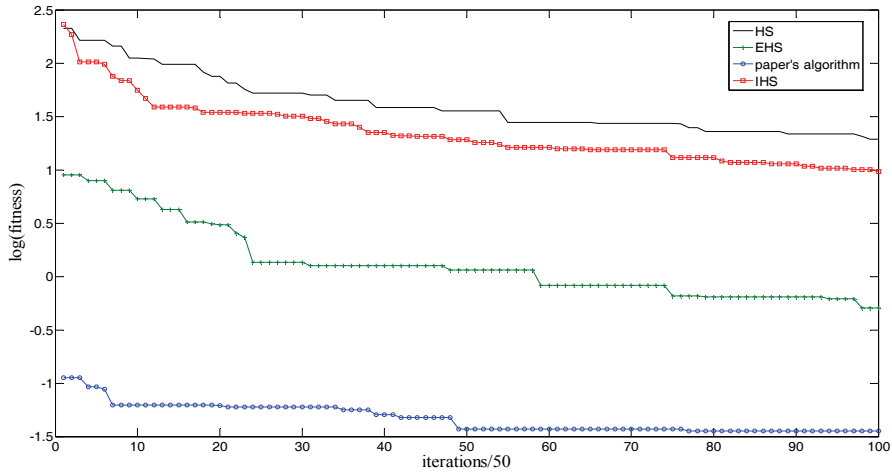
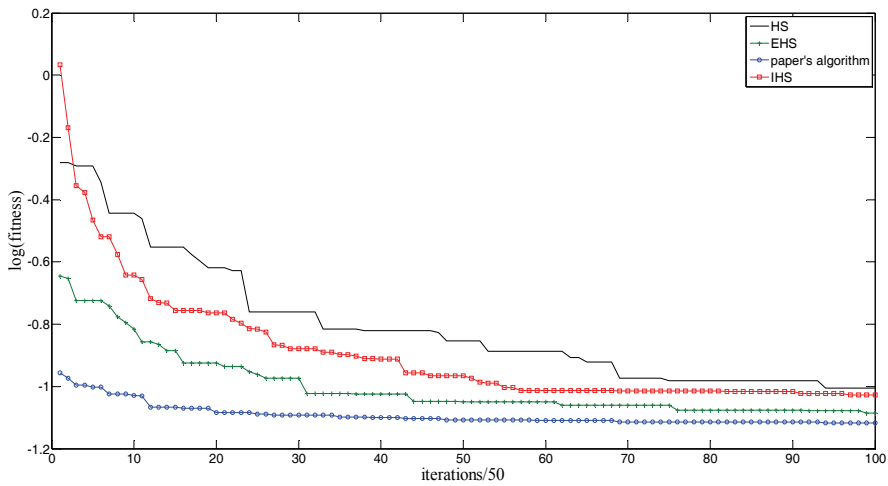**Fig. 4.** The fitness function comparison curve of Rastrigin function.

Fig. 5. The fitness function comparison curve of Griewank function.
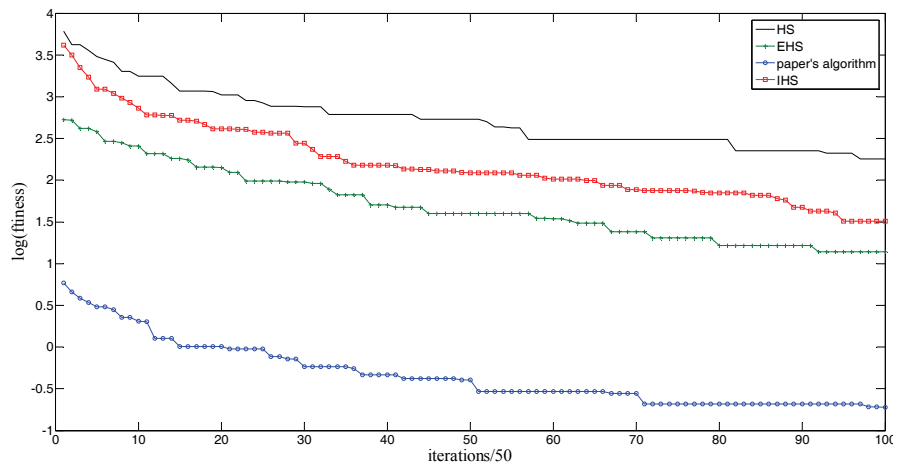

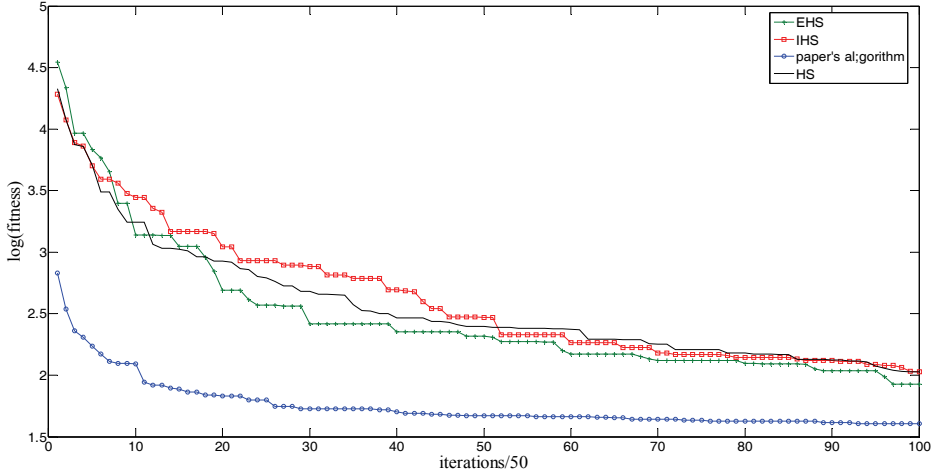**Fig. 6.** The fitness function comparison curve of Ackley function.

Fig. 7. Fitness function Rosenbrock function curve comparison.

Table 2. Simulation results of algorithms

Function	Algorithm	Average fitness	Best average	Standard deviation	Success rate (%)
f_1	HS	13.6480	11.3287	2.0176	55
	IHS	11.2023	10.5874	1.5621	60
	EHS	8.4558.	6.7412	1.2354	65
	Algorithm in this paper	0.2939	-0.2435	0.0623	100
f_2	HS	10.0691	7.5561	1.7824	50
	IHS	9.2632	7.1236	1.5202	65
	EHS	7.2102	5.3214	1.0232	70
	Algorithm in this paper	0.0748	0.0008	0.0695	100
f_3	HS	6.2356	3.1523	2.4232	45
	IHS	5.2541	2.321	1.9523	65
	EHS	2.2023	0.9523	1.1021	80
	Algorithm in this paper	0.6089	0.3598	0.0063	100
f_4	HS	15.1525	10.5421	2.5652	20
	IHS	13.2541	8.3652	1.8523	40
	EHS	6.5626	4.1202	1.2301	50
	Algorithm in this paper	0.1236	0.0952	0.0463	100

Table 2 shows the comparison result of three algorithms, including the best fitness value, the average value of adaptation and optimization of success. Best fitness and average fitness value reflects the convergence rate. The average fitness value reflects the robustness of the algorithm. Success rate reflects the global optimization algorithm. As can be seen from Table 2, the best fitness, success rate, and other performance is better than other algorithms.

In order to analyze the complexity of several algorithms include the standard HS algorithm, IHS algorithm, EHS algorithm, and the proposed algorithm in this paper, the simulation experiment is performed out. The number of iterations is set to 5000. Four algorithms are running 20 times independently. Table 3 shows the average running time (in seconds) of four functions in the same computer (CPU is Intel E6550 2.33 GHz, memory is 2 GB, Windows 7 operating system). The simulation software is MATLAB

2010b. The simulation result shows that the running time of the improved harmony search algorithm in this paper is slightly longer than the standard HS algorithm, lesser than EHS algorithm and close to IHS algorithm. The results show that the algorithm in this paper has better optimization effect without increasing the computational complexity.

Table 3. The average running time of the algorithms (unit: second)

Function	HS	IHS	EHS	Algorithm in this paper
f_1	0.3262	0.3346	0.4462	0.3348
f_2	0.4242	0.4312	0.5402	0.4332
f_3	0.9738	1.0254	1.0858	1.0254
f_4	0.3182	0.3313	0.4342	0.3321

4.2 Engineering Example Simulation

Pressure vessel design optimization problem is well known in the field of engineering benchmark problem [25]. It can be used to optimize the performance test algorithm. As showed in Fig. 8, the optimization problem can be described as to calculate optimized design variables $x_1(R)$, $x_2(L)$, $x_3(t_s)$ and $x_4(t_h)$, which make vessel material most provinces.
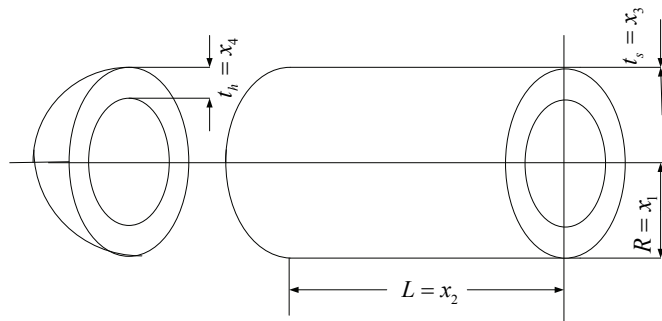

Fig. 8. The structure of the vessel.

Where, x_1 is the radius of the vessel, x_2 is the vessel tube length, x_3 is the cylinder wall thickness, x_4 is the hemispherical head wall thickness. x_1 and x_2 is the continuous variable. x_3 and x_4 is integer or discrete variables, and they are the integer value multiples of 0.0625. So vessel optimization design model is as follows.

Search variables $X(x_1, x_2, x_3, x_4)$, make $f(X)$ is minimized.

$$f(X) = 0.6224x_1x_2x_3 + 1.7781x_1^2x_4 + 3.1661x_2x_3^2 + 19.84x_1x_3^2, \tag{12}$$

Constraint conditions:

$$10 \le x_1 \le 200,$$

$$10 \le x_2 \le 200,$$

$$0.0625 \leq x_3 \leq 6.1875 \,,$$

$$0.0625 \leq x_4 \leq 6.1875 \,,$$

$$g_1(X) = \frac{0.0193 x_1}{x_3} - 1 \leq 0 \,,$$

$$g_2(X) = \frac{0.00954 x_1}{x_4} - 1 \leq 0 \,,$$

$$g_3(X) = \frac{x_2}{240} - 1 \leq 0 \,,$$

$$g_4(X) = \frac{1296000 - \dfrac{4}{3}\pi x_1^3}{\pi x_1^2 x_2} - 1 \leq 0 \,.$$

HS standard algorithms herein, IHS algorithm, EHS algorithm and the improved algorithm with the above parameter HS same 4.1. Fig. 9 shows the convergence curve four algorithms. As can be seen from the figure, the algorithm converges faster and with better fitness than other algorithms. Algorithm runs 20 times. Table 4 shows the results of several optimization algorithms, to adapt the average value of the pressure vessel 6356.17, which is better than the other algorithms results. This article about IHS algorithm is proven to be effective.
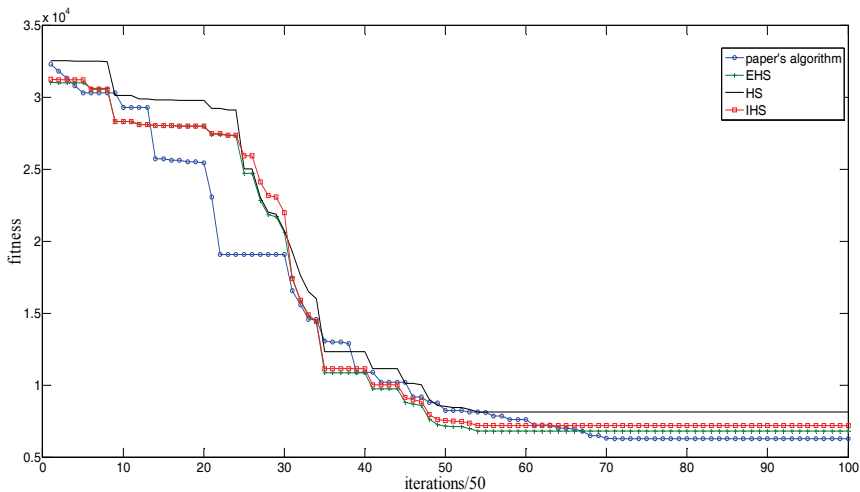

Fig. 9. The fitness curve of vessel optimization.

The fitness curve and optimization results can explain the simulation results correspond to the above deductions in theory. Firstly, the fitness curves in Figs. 4–7 and Fig. 9 show that the proposed IHS algorithm has faster convergence speed and better optimization results. Next, Table 2 shows the results of four kinds of optimization of complex functions. In Table 2, the average value and the best fitness value reflects the degree fitting algorithm convergence speed. The average fitness value reflects the robustness of the algorithm. Success rate reflects the global optimization algorithm. Table 4 shows that

the proposed algorithm can get better than several other harmony search algorithm results. Finally, Table 3 shows the calculation time of each algorithm. The simulation result shows that the running time of the improved harmony search algorithm in this paper is slightly longer than the standard HS algorithm, lesser than EHS algorithm and close to IHS algorithm.

Table 4. The contrast results of vessel optimization

The optimized value	HS	IHS	EHS	The algorithm in this paper
x_1	48.12	54.23	40.33	44.02
x_2	119.82	45.36	198.23	164.24
x_3	1.1250	1.1250	0.8386	0.8527
x_4	0.7852	0.7523	0.6052	0.4371
$g_1(X)$	-0.1744	-0.0697	-0.0728	-0.0037
$g_2(X)$	-0.4154	-0.3123	-0.3642	-0.0392
$g_3(X)$	-0.5008	-0.8110	-0.1741	-0.3157
$g_4(X)$	-0.0485	-0.4996	-0.0088	-0.0605
$f(X)$	8958.46	7199.81	6927.12	6356.17

5. Conclusions

In practice, many engineering optimization problems belong to the function optimization problem, usually with large-scale, high-dimensional and non-linear characteristics. When a precise optimization algorithms to solve the problem, there is a long computing time disadvantages. Using intelligent optimization algorithm function optimization is an effective method. Therefore, the intelligent optimization algorithm to function optimization problem has important theoretical and practical significance.

The HS algorithm is a new intelligent optimization algorithm. It has a conceptually simple, easy to implement and less adjustment parameters advantages. However, it also has the disadvantage of randomness, such as search directional uncertainty and so on. Although the introduction of different ideas and methods in HS algorithm, but the performance of the algorithm has been improved, and improves the convergence precision and convergence speed of the algorithm. However, HS algorithm and its improved algorithm still has a slower convergence speed, easy to fall into local optimum. At the same time, too many of these parameters IHS algorithm, need to be adjusted through a large number of simulation experiments or experience setting to reduce the applicability of the algorithm in practice. In order to improve the performance of HS algorithm, this paper presents an IHS algorithm. This algorithm optimizes the three important parameters. Based on numerical experiments to optimize complex functions and four pressure vessel problems show that the proposed IHS algorithm is simple, easy to implement, and to find a better solution than the other algorithms more efficiently. The main contribution of this paper includes:

1. These IHS algorithms are only limited to the single parameter optimization. They can't achieve the overall performance enhancement. Three important parameters are optimized simultaneously in proposed algorithm.

2. An improved new solution generating method refers to the genetic algorithm, which avoid randomness in the process of new solutions generation.

3. In addition, when the size increases, IHS algorithm proposed an overwhelming advantage compared with other HS algorithm.

In general, it can be concluded that the IHS algorithm, its simple, high-quality solutions to achieve, few set parameters and faster convergence. In dealing with other complex optimization algorithm is an ideal method. IHS algorithm can be used to optimize some difficulties and problems, multi-dimensional better choice in the real world.

Acknowledgement

References

[1] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulations*, vol. 76, no. 2, pp. 60-68, 2001.

[2] H. B. Ouyang, L. Q. Gao, D. X. Zou, and X. Y. Kong, "Exploration ability study of harmony search algorithm and its modification," *Control Theory and Applications*, vol. 31, no. 1, pp. 57-65, 2014.

[3] D. Manjarres, I. Landa-Torres, S. Gil-Lopez, J. D. Ser, M. N. Bilbao, S. Salcedo-Sanz, and Z. W. Geem, "A survey on applications of the harmony search algorithm," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 8, pp. 1818-1831, 2013.

[4] B. Alatas, "Chaotic harmony search algorithms," *Applied Mathematics and Computation*, vol. 216, no. 9, pp. 2687-2699, 2010.

[5] R. Arul, G. Ravi, and S. Velusami, "Solving optimal power flow problems using chaotic self-adaptive differential harmony search algorithm," *Electric Power Components and Systems*, vol. 41, no. 8, pp. 782-805, 2013.

[6] S. Sayah, A. Hamouda, and A. Bekra, "Efficient hybrid optimization approach for emission constrained economic dispatch with nonsmooth cost curves," *International Journal of Electrical Power and Energy Systems*, vol. 56, pp. 127-139, 2014.

[7] B. Zeng and Y. Dong, "An improved harmony search based energy-efficient routing algorithm for wireless sensor networks," *Applied Soft Computing*, vol. 41, pp. 135-147, 2016.

[8] G. F. de Medeiros and M. Kripka, "Optimization of reinforced concrete columns according to different environmental impact assessment parameters," *Engineering Structures*, vol. 59, pp. 185-194, 2014.

[9] X. Y. Li, K. Qin, B. Zeng, L. Gao, and J. Z. Su, "Assembly sequence planning based on an improved harmony search algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 84, no. 9, pp. 2367-2380, 2016.

[10] G. Naresh, M. R. Raju, and S. V. L. Narasimham, "Coordinated design of power system stabilizers and TCSC employing improved harmony search algorithm," *Swarm and Evolutionary Computation*, vol. 27, pp. 169-179, 2016.

[11] Z. D. Tian, S. J. Li, Y. H. Wang, and X. D. Wang, "LSSVM predictive control for calcination zone temperature in rotary kiln with IHS algorithm," *Journal of Harbin Institute of Technology (New Series)*, vol. 23, no. 4, pp. 67-74, 2016.

[12] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied Mathematics and Computation*, vol. 188, no. 2, pp. 1567-1579, 2007.

[13] M. G. H. Omran and M. Mahdavi, "Global-best harmony search," *Applied Mathematics and Computation*, vol. 198, no. 2, pp. 643-656, 2008.

[14] M. Fesangharya, M. Mahdavib, M. Minary-Jolandan, and Y. Alizadeha, "Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 33-40, pp. 3080-3091, 2008.

[15] 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," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 1, pp. 89-106, 2011.

[16] E. Valian, S. Tavakoli, and S. Mohanna, "An intelligent global harmony search approach to continuous optimization problems," *Applied Mathematics and Computation*, vol. 232, no. 3, pp. 670-684, 2014.

[17] D. X. Zou, L. Q. Gao, J. H. Wu, and S. Li, "Novel global harmony search algorithm for unconstrained problems," *Neurocomputing*, vol. 73, no. 16, pp. 3308-3318, 2010.

[18] W. L. Xiang, M. Q. An, Y. Z. Li, R. C. He, and J. F. Zhang, "An improved global-best harmony search algorithm for faster optimization," *Expert Systems with Applications*, vol. 41, no. 13, pp. 5788-5803, 2014.

[19] Z. W. Geem, "Effects of initial memory and identical harmony in global optimization using harmony search algorithm," *Applied Mathematics and Computation*, vol. 218, no. 22, pp. 11337-11343, 2012.

[20] Z. W. Geem and K. B. Sim, "Parameter-setting-free harmony search algorithm," *Applied Mathematics and Computation*, vol. 217, no. 8, pp. 3881-3889, 2010.

[21] M. Khalili, R. Kharrat, K. Salahshoor, and M. H. Sefat, "Global dynamic harmony search algorithm: GDHS," *Applied Mathematics and Computation*, vol. 228, pp. 195-219, 2014.

[22] H. B. Ouyang, L. Q. Gao, S. LI, X. Y. Kong, Q. Wang, and D. X. Zou, "Improved harmony search algorithm: LHS," *Applied Soft Computing Journal*, vol. 53, pp. 133-167, 2017.

[23] T. Hassanzadeh and H. R. Kanan, "Fuzzy FA: a modified firefly algorithm,: *Applied Artificial Intelligence*, vol. 28, no. 1, pp. 47-65, 2014.

[24] G. Z. Tan, K. Bao, and R. M. Rimiru, "A composite particle swarm algorithm for global optimization of multimodal functions," *Journal of Central South University*, vol. 21, no. 5, pp. 1871-1880, 2014.

[25] J. Kruzelecki and R. Proszowski, "Shape optimization of thin-walled pressure vessel end closures," *Structural and Multidisciplinary Optimization*, vol. 46, no. 5, pp. 739-754, 2012.

Zhongda Tian https://orcid.org/0000-0003-0379-4048

He received the Ph.D. degree in control theory and control engineering from Northeastern University, China in 2013. His research interests include predictive control, delay compensation and scheduling for networked control system and optimization algorithms. He is currently a lecturer at Shenyang University of Technology, Shenyang, China.

Chao Zhang https://orcid.org/0000-0001-6759-8914
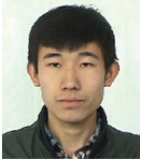
He received the B.E. degree in electrical and information engineering from Zaozhuang University, China, in 2014. He is currently pursuing his M.E. degree in control engineering at Shenyang University of Technology, Shenyang, China. His current research interests include networked control system and optimization algorithms.