An Improved Harmony Search Algorithm & Application

Zhang Ting, Jiao Bin, Liu Hai, Peng Long

Department of Electrical Engineering, Shanghai Dianji University, Shanghai, 200240, China

email: xunzhaoshuangdeyu@163.com

Keywords: Harmony Search; Improved Algorithm; Parameters; Memory; Optimization.

Abstract. Harmony search algorithm is an intelligent optimization algorithm. The function converges to its optimal solution with increased iterations and completes the optimization by adjusting the solution variables repeatedly in harmony memory. The original steps and processes of harmony search algorithm are given, and an improved harmony search algorithm is proposed, the search efficiency is improved.

Introduction

Harmony search (HS) algorithm is a novel intelligent optimization algorithm proposed by Korean scholars Geem Z W et al. in 2001 [1].Algorithm simulates a process that musicians adjustment band of each instrument in the tones, relying on his own memory in the music creation, and finally the tones reach a wonderful harmony state. At present, the method has been widely applied in clustering analysis [2]; Multi-objective optimization [3].structure finite element model updating [4]; the design of parameters of PID control [5] and other issues. Research shows that HS has a better optimization effect than genetic algorithm and simulated annealing algorithm .

Original harmony search algorithm

The steps of original HS algorithm can be divided into the following five parts: initialize algorithm parameters; initialize memory; create a new harmony; update the harmony memory; determine the termination conditions [6]. The original HS algorithm is carried out as follows.

1. Initialization of parameters

The parameters of HS algorithm mainly include: harmony memory size (HMS), harmony memory considering rate (HMCR), pitch adjusting rat (PAR), fine-tuning bandwidth (bw).At the same time the termination conditions and the constraint condition of optimization problem also needs to be set. A non-constraint optimization problems can generally be described as:

$$\begin{cases} \min f\left(\mathrm{X}\right), \mathrm{X}=\left(x_1, \mathrm{x}_2, \ldots, x_n\right) \\ x_i \in X_i=\left[x_{i\min}, x_{i\max}\right], i=1,2,\ldots n \end{cases} \tag{1}$$

where f(x)is the objective function, X is a vector consisting of decision variables. The original HS algorithm basically considers the objective function only. If the problem has equality and/or inequality constraints and a solution vector violates any of them, either the resulting infeasible solution will be abandoned or its objective function value will have a certain penalty score added.

2 . Initialization of harmony memory

The process of harmony memory (analogy to the genetic algorithm population) initialization is to initialize every dimensions of harmony vectors in the value range ,as shown in the following form.

$$\mathrm{x}_\mathrm{i}^\mathrm{k}=\mathrm{x}_\mathrm{imin}+(\mathrm{x}_\mathrm{imax}-\mathrm{x}_\mathrm{imin})\times\mathrm{rand}(0,1)$$

(2)

The harmony memory form is as follows:

$$\mathrm{HM}=\left[\begin{array}{c|c} \mathbf{X}^1 & f(\mathbf{X}^1) \\ \mathbf{X}^2 & f(\mathbf{X}^2) \\ \vdots & \vdots \\ \mathbf{X}^{\mathrm{HMS}} & f(\mathbf{X}^{\mathrm{HMS}}) \end{array}\right]=\left[\begin{array}{c|c} \mathrm{x}_1^1,\cdots,\mathrm{x}_\mathrm{n}^1 & f(\mathbf{X}^1) \\ \mathrm{x}_1^2,\cdots,\mathrm{x}_\mathrm{n}^2 & f(\mathbf{X}^2) \\ \vdots & \vdots \\ \mathrm{x}_1^{\mathrm{HMS}},\cdots,\mathrm{x}_\mathrm{n}^{\mathrm{HMS}} & f(\mathbf{X}^{\mathrm{HMS}}) \end{array}\right] \tag{3}$$

3. Generation of new harmony vectors

This is the most core part, generated harmony vectors is a new solution vectors. New harmony is produced mainly by the following three ways: (1) learning from harmony memory ;(2) fine-tuning a tone;(3) generating a new tone randomly .

In producing a new harmony $X^*=[x_1{}^*, x_2{}^*, x_3{}^*]$, a random number rand_1 is generated among the interval (0, 1), and then the generation of the variables is as following:

$$\mathrm{x}_\mathrm{i}^* = \begin{cases} \mathrm{x}_\mathrm{i} \in \left\{\mathrm{x}_\mathrm{i}^1, \mathrm{x}_\mathrm{i}^2, \cdots, \mathrm{x}_\mathrm{i}^{\mathrm{HMS}}\right\}, & \mathrm{rand}_1 < HMCR \\ \mathrm{x}_\mathrm{i} \in \mathbf{X}_\mathrm{i}, & \mathrm{rand}_1 \geq \mathrm{HMCR} \end{cases} \tag{4}$$

Second, if the new harmony $x_i{}^*$ is from HM, it should be fine-tuned, specific operation is as following:

$$\mathrm{x}_\mathrm{i}^* = \begin{cases} \mathrm{x}_\mathrm{i}^* + \mathrm{rand}\left(0,1\right) \times \mathrm{bw}, & \mathrm{rand}_1 < PAR \\ \mathrm{x}_\mathrm{i}^*, & \mathrm{rand}_1 \geq \mathrm{PAR} \end{cases} \tag{5}$$

4. Update of harmony memory

After new harmony vectors generated, it’s evaluated firstly with the objective function. If it is superior to the worst harmony in new harmony memory, then replaces the worst in harmony memory. The specific operation is as following:

$$\text{if } f\left(\mathbf{X}^*\right) < f\left(\mathbf{X}^{\mathrm{worst}}\right), \mathrm{then}\mathbf{X}^{\mathrm{worst}} = \mathbf{X}^* \tag{6}$$

5. Determination of termination conditions

Repeat the above two steps until the termination criterion (maximum iterative steps) is satisfied. Original HS algorithm process is shown in figure 1.

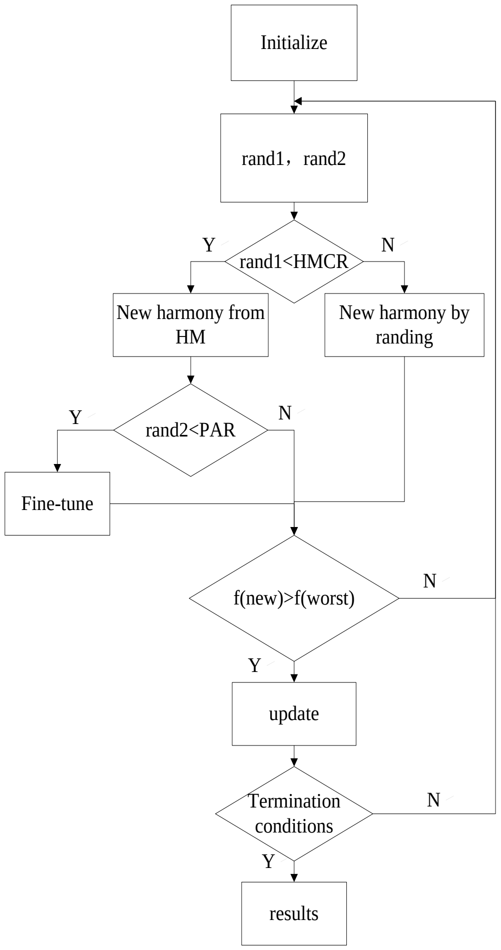


Fig.1. Original HS

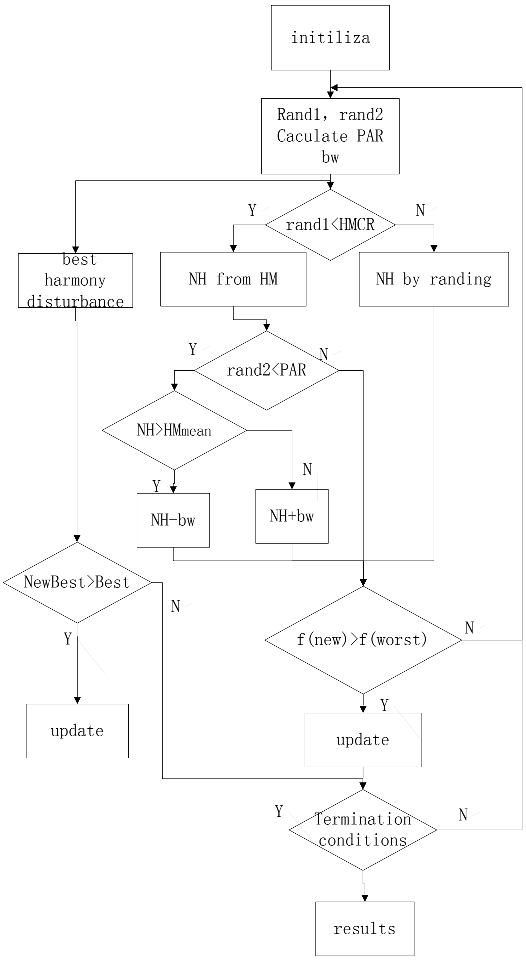


Fig.2. Improved HS

Improved harmony search algorithm

1.In the original harmony algorithm, the values of PAR and bw is fixed, cannot give consideration to both global optimal solution and local optimal solution search, the performance of search algorithm will be impacted. PAR can enhance the capacity of local search of algorithm when its value is small , on the contrary, it contribute to adjust the search area around the harmony memory and make the algorithm search space expand to the whole solution space gradually. bw help to fine search in local area when its value is small, otherwise, it can make the algorithm to jump out of local optimum. At the beginning of the search algorithm, smaller PAR contribute to quick search in better area, at the later period of the algorithm, the larger par is conducive to jump out of local optimal algorithm, therefore, PAR should be changed from small to large. For bw , larger bw is conducive to the search in a larger scope at the beginning, in the late algorithm search , smaller bw is conducive to the fine search on a small scale. Hence, bw should be changed from large to small.

Therefore, this paper puts forward a parameter dynamic update strategy: to set the value ranges of PAR and bw; at the beginning of the iteration, to choose greater parameters for updating in order to ensure the diversity of harmony memory; after a certain period of time, to choose the smaller for updating in order to improve the accuracy of the algorithm Adaptive Parameter Adjusting HS, APAHS). Specific update formula is as following:

$$\mathrm{PAR}\left(\mathrm{k}\right)=\mathrm{PAR}_{\min}+(\mathrm{PAR}_{\min}-\mathrm{PAR}_{\max})*k/K \qquad (7)$$

$$\mathrm{bw}\left(\mathrm{k}\right)=\mathrm{bw}_{\max}*\exp(\ln(\mathrm{bw}_{\min}/\mathrm{bw}_{\max})\mathrm{k}/\mathrm{K}) \qquad (8)$$

where K and k denote the maximum number of iterations and the kth-iteration, respectively. $\mathrm{PAR}_{\min}$ and $\mathrm{PAR}_{\max}$ indicate the minimum and maximum pitch adjusting rates, respectively. $\mathrm{bw}_{\min}$

and bw_{max} indicate the minimum and maximum bandwidths. The dynamic curve of PAR and bw is shown in figure 3 and 4(Adaptive Parameter Adjusting HS,APAHS).

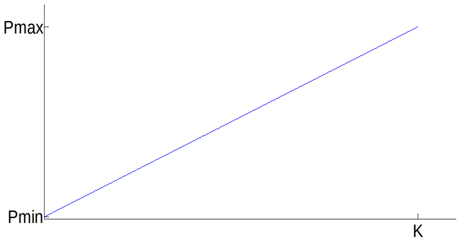


Fig.3. The dynamic curve of PAR

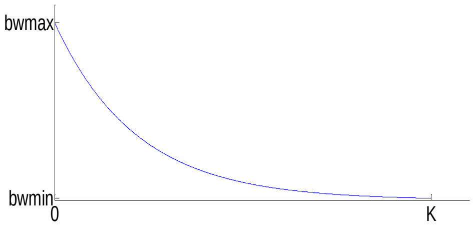


Fig.4. The dynamic curve of bw

2. In the original harmony algorithm, when $rand_1$<HMCR, a new harmony will be randomly selected from harmony memory. To a certain extent, the random choice is not conducive to update harmony memory in a better direction, thus to make sure it is towards a better direction, using the best harmony in harmony memory and fine-tuning at the same time, then comparing with the best, choosing the best one and updating(Best Harmony Evolution HS,BHEHS).

3.when $rand_2$<PAR, the new harmony need be fine-tuned, the value added is random in the original harmony algorithm. For fine-tuning to the better orientation, a approach to adjust the orientation is given below :first, to calculate the average value of HM, if the new harmony x_i^*>average(x_i),then bw is minus from x_i^*,otherwise, bw is plused to x_i^* .

$$X_i^* = \begin{cases} X_i^* - bw, & X_i^* \geq average(X_i); \\ X_i^* + bw, & others. \end{cases} \quad (9)$$

Improved algorithm (AB-HS) uses the dynamic update PAR and bw to generate new harmony and evolve the best harmony of every generation. So algorithm optimization precision and convergence speed are improved. Improved harmonic algorithm flow chart is shown in figure 2.

Typical function simulation experiment and result analysis

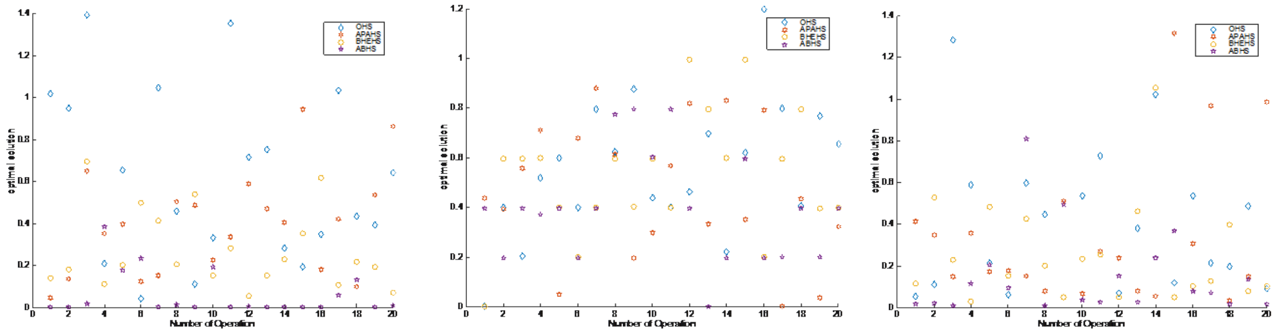
To test the search performance of the improved harmony search algorithm, we use the original HS algorithm [6] as a reference object, and optimize the minimum of the following three benchmark functions:

$$f_1(x) = \sum_{i=1}^{n} x_i^2, x_i \in [-100,100], \quad (10)$$

$$f_2(x) = \sum_{i=1}^{n-1} \left(100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2\right), \quad x_i \in [-100,100], \quad (11)$$

$$f_3(x) = \sum_{i=1}^{i=n} \left(x_i^2 - 10\cos\cos\left(2\pi x_i\right) + 10\right), \quad x_i \in [-10,10], \quad (12)$$

Among them: f_1 is the Spherical function, f_2 is Rosenbrock function,and f_3 is Rastigrin function.Global minimum of the test functions are 0(min f_i=0(i=1,2,3)), and parameters of each method are as follows: the variable dimension n=10, and the largest evolution iteration T=5000;Harmony memory size HMS=20[8];HMCR=0.90; PAR=0.30 and bw=0.01 in original HS;for the improved HS;PAR_{min}=0.01,bw_{min}=0.0001 and PAR_{max}=0.99,bwmax=(x_{max}-x_{min})/20.Use test functions as fitness functions and run 20 times independently, the optimization results of three benchmark functions is shown in figure 5 and the best ,worst and mean results and standard deviations is presented in table 1.



(a) Spherical function (b)Rosenbrock function (c)Rastigrin function

Fig.5. Results over 20 independent runs of three functions

TABLE 1

RESULTS OVER 20 INDEPENDENT RUNS OF 4 ALGORITHMS (N = 10, T = 50000)

Function	Result	OHS	APAHS	BHEHS	ABHS
F1	Best	1.2674	0.1757	0.3737	0.0000
	Av.	6.1852	2.3271	3.5077	0.0633
	Worst	12.940	16.6516	8.3891	1.0382
	Std	3.1315	3.6659	2.7023	0.2323
F2	Best	0.0597	0.0361	0.0284	0.0013
	Av.	0.4046	0.3509	0.3238	0.1492
	Worst	1.2580	1.5693	1.9262	0.5953
	Std	0.3684	0.3795	0.4453	0.1706
F3	Best	0.0069	0.0244	0.9987	0.0001
	Ave.	3.0027	2.8983	2.4058	1.7504
	Worst	5.9803	6.0568	6.9664	3.9886
	Std	1.5145	1.4425	1.6176	1.1539

The conclusions acquired from figure 5 and table 1:

1) From search results of the original HS algorithm, the optimization results are deviated from expected global optimal value for the defect of premature and local convergence; For high dimensional complex function optimization problems, the method is difficult to obtain ideal optimization results.

2) The comparison shows that the best,average and worst values precision and standard deviation of improved algorithm are better than the original algorithms; Improved algorithm improve the search ability of HS algorithm, and is more stable compared with the original algorithm.

Conclusion

This paper puts forward an improved harmony search algorithm, the difference between it and the original harmony search algorithm is that it adjusts the parameters and evolve the best harmony in every generation. The results of typical function extremum optimization show that the method is able to greatly enhance the optimization ability of the basic harmony search algorithm.

Acknowledgement

In this paper, the research was sponsored by Shanghai science and Technology Commission (Project No. 13DZ0511300)

References

[1] GEEM Z W, KIM J H, LOGANATHAN G V. A new heuristic optimization algorithm: harmony search [J]. Simulation, 2001, 76(2): 60-68.

[2] YI Yu-feng, GAO Li-qun, GUO Li. The Application of Harmony Search Algorithm in Clustering Analysis [J]. Journal of Northeastern University(Natural Science), 2012, 33(1): 47-51.

[3] Hao Bing，Ren Xian-hua，Gao Yue-lin, Jiang Qiao-yong. Hybrid harmony search and estimation of distribution algorithm for multi-objective optimization problems[J]. Application Research of Computers, 2012,29(5): 1659-1661.

[4] Du Yong-feng, Li Wan-run, Li Hui, Tang Shao-yu. Application of harmony search algorithm in structural finite element model updating[J]. Journal of Lanzhou University of Technology,2013, 39(5): 106-110.

[5] Cao Fang, Wang Wei, Wang Jie-sheng, Liu Yu-chao, Chao Sheng-xi. Self-Tuning of PID Controller Parameters Based on HS-PSO Algorithm [J]. J ,Control Engineering of China 2011:18(3):352-355.

[6] YONG Long-Quan, Advances in Harmony Search Algorithm[J]. Application of Computer System,2011: 20(7) ：244-248.

[7] Chen Yin-zhen，Gao Yue-lin，An Improved Adaptive Harmony Search[J]. Journal of Gansu Lianhe University(Natural Sciences) 2011：25（2）：63-66.

[8] Pan Yu-xia, Xie Shen-xian. The Research of Harmony Search Algorithm parameters [J].Software Guide，2013：12（6）：40-41.