ICIC Express Letters
Volume **11**, Number **3**, March **2017**

ICIC International ©2017 ISSN 1881-803X
pp. 643–648

HARMONY SEARCH WITH IMPROVED SEARCH STRATEGY

Zhaolu Guo[1,*], Huogen Yang[1], Caiying Zhou[1], Xiaosheng Liu[2]
and Zhijian Wu[3]

[1]Institute of Medical Informatics and Engineering
School of Science
[2]School of Architectural and Surveying and Mapping Engineering
Jiangxi University of Science and Technology
No. 86, Hongqi Ave., Ganzhou 341000, P. R. China
*Corresponding author: gzl@whu.edu.cn

[3]State Key Laboratory of Software Engineering
Computer School
Wuhan University
No. 299, Bayi Road, Wuhan 430072, P. R. China

Received August 2016; accepted November 2016

Abstract. *Harmony search (HS) is a popular evolutionary algorithm, which has been successfully applied in many fields. However, the basic HS often exhibits poor exploitation ability when handling some complex optimization problems. To enhance the exploitation ability of HS, this paper presents a harmony search with improved search strategy (ISHS). In the proposed ISHS, the best harmony from the current harmony memory is utilized in the memory consideration step to guide the search direction. Furthermore, the search experience of the best harmony is employed in the pitch adjustment phase to accelerate the convergence speed. Experiments on a set of well-known test functions indicate that the improved search strategy is able to enhance the exploitation ability of HS.*

Keywords: Global optimization, Harmony search, Search strategy, Best harmony

1. **Introduction.** Evolutionary algorithm (EA) is a kind of effective optimization technique, which has been widely employed in many applications [1]. So far, EA has become a research hot spot in the field of optimization technique. Therefore, various kinds of EAs have been proposed in the evolutionary computation community. Harmony search (HS) is an emerging EA, which is inspired by the process of music improvisation [2]. Like other EAs, HS has many advantages [3, 4], such as simple structure, easy implementation, and efficient performance [5]. Due to its advantages, HS has attracted many researchers from various fields since its development [6].

However, the basic HS often suffers from poor exploitation ability when facing some complicated optimization problems [7]. Aiming at this disadvantage of HS, researchers have presented many HS variants to enhance the exploitation ability [8]. Mahdavi et al. [9] introduced an improved HS (IHS), which employs the linear and exponential models to take values for the parameters. Omran and Mahdavi [10] proposed an HS variant, called GHS, which is inspired by the mechanism of particle swarm optimization. In GHS, it improves the pitch adjustment scheme, which directly inherits a random parameter of the best harmony in the current harmony memory. The experiments show that the improved pitch adjustment scheme in GHS is able to enhance the performance. Later, Pan et al. [11] presented a self-adaptive global best HS (SGHS), which exploits the information of the best harmony to accelerate the convergence rate. In addition, SGHS adaptively tunes the harmony memory consideration rate (HMCR), pitch adjustment rate (PAR) and bandwidth (BW). The experimental results indicate that SGHS can outperform HS, IHS and GHS on the majority of the test functions. Zou et al. [12] presented a novel global HS (NGHS), which introduces a novel search strategy by combining the global best harmony. The experiments demonstrate that NGHS can surpass HS, IHS, and SGHS on the most of the test functions.

Although many HS variants have been proposed, HS still tends to demonstrate poor exploitation capability when solving some highly non-linear and multimodal optimization problems [3]. Therefore, the HS performance needs to be further enhanced. To promote the exploitation capability of HS, this paper presents a harmony search with improved search strategy (ISHS). In the search process, ISHS employs the best harmony from the current harmony memory to guide the search direction. Furthermore, ISHS exploits the search experience of the best harmony to accelerate the convergence speed.

The rest of the paper is organized as follows. The basic operations of HS are introduced in Section 2. In Section 3, we present the proposed HS algorithm. Experimental results are reported in Section 4 for the comparison and analysis. Finally, Section 5 concludes this paper.

2. **Harmony Search.** HS has a simple computation procedure. In the initial stage, it randomly initializes a harmony memory that consists of HMS solutions $HM(t) = \{X_i^t\}$, where $X_i^t = \left[x_{i,1}^t, x_{i,2}^t, \ldots, x_{i,D}^t\right]$, $i = 1, 2, \ldots, HMS$; D is the dimension of the optimization problem; HMS represents the size of the harmony memory and t is the generation. After initialization, HS performs a loop of evolutionary operations to create new candidate harmonies, and then selects the better harmonies into the next generation to guide the harmony memory towards the optimum [2]. In HS, the main evolutionary operations include memory consideration, pitch adjustment, and random sampling operations, which are introduced as follows.

In the memory consideration step, HS randomly selects a harmony from the harmony memory [7], and then a component of the selected harmony is employed to constitute the new candidate harmony.

The pitch adjustment operation focuses on exploiting the neighborhood information of the current harmony memory, which conducts a random perturbation on a component of the new candidate harmony. In addition, the step-size of the random perturbation is determined by the bandwidth BW [3].

The random sampling operation is used to maintain the diversity of the harmony memory, which randomly generates a component from the search space for the new candidate harmony [10].

3. **The Proposed ISHS.** In this section, we introduce a harmony search with improved search strategy (ISHS). We first describe the improved search strategy in which the best harmony from the current harmony memory is utilized to generate new candidate harmony. Then, the algorithmic framework of the proposed ISHS is presented.

3.1. **Improved search strategy.** In the search strategy of HS, it generates a new candidate harmony by inheriting the information from a random harmony in the current harmony memory [10]. In this way, the search direction is guided by a random harmony. Therefore, the search strategy has no bias to any special direction and selects search direction using a random mechanism. Such search strategy is useful for maintaining the diversity of the harmony memory and is beneficial to avoid trapping in local minima [12], which indicates the search strategy of HS does well in exploration search. However, due to the selection of random search direction, the search strategy of HS often exhibits poor exploitation ability to converge to a near optimum. Thus, HS often suffers from slow convergence speed [3]. In order to enhance the exploitation ability of HS, we propose an improved search strategy.

In the improved search strategy, it conducts the same random sampling operation as the basic HS, which is formulated as follows [2]:

$$v_j = LB_j + \text{rand}(0,1) \times (UB_j - LB_j) \tag{1}$$

where LB_j and UB_j denote the lower and upper boundaries of the search space, respectively; $\text{rand}(0,1)$ is a random real number between 0 and 1, which is generated anew for each j.

Algorithm 1 ISHS Algorithm

```
t = 0;
FEs = 0;
Generate a random initial harmony memory;

while FEs < MaxFEs do
    PAR = 0.01 + 0.98 × t/MaxT;

    /* Create a new candidate harmony */
    for j= 1 to D do
        if rand(0, 1) < HMCR then
            r1 = randInt(1, HMS);
            v_j = x^t_{r1,j} + rand(−1, 1) × (x^t_{Best,j} − x^t_{r1,j});
            if rand(0, 1) < PAR then
                k = randInt(1, D);
                v_j = x^t_{Best,k};
            end if
        else
            v_j = LB_j+ rand(0, 1) × (UB_j − LB_j);
        end if
    end for
    Evaluate the fitness value of harmony V;
    FEs = FEs + 1;

    /*Conduct the selection operation */
    if f(V) < f(X^t_{Worst}) then
        X^t_{worst} = V;
        if f(V) < f(X^t_{Best}) then
            X^t_{Best} = V;
        end if
    end if
    t = t + 1;
end while
```

Besides the random sampling operation, the improved search strategy simultaneously performs a new memory consideration operation with a probability of $HMCR$. Compared with the original memory consideration operation in the basic HS, the new memory consideration operation employs the best harmony from the current harmony memory to guide the search direction. Therefore, the exploitation ability to converge to a near optimum can be enhanced and the convergence speed of the proposed ISHS can be accelerated. The new memory consideration operation is defined as follows:

$$v_j = x_{r1,j} + \text{rand}(-1,1) \times (x_{Best,j} - x_{r1,j}) \tag{2}$$

where rand$(-1, 1)$ is a real number randomly generated in the range $[-1, 1]$; $x_{r1,j}$ is the jth component of a harmony randomly selected from the current harmony memory, and $x_{Best,j}$ is the jth component of the global best harmony in the current harmony memory.

After the memory consideration operation, the improved search strategy executes the pitch adjustment operation with a probability of PAR. In the pitch adjustment operation, it employs the pitch adjustment strategy of GHS [10], which directly inherits the search experience of the best harmony to further enhance the exploitation ability. Therefore, the pitch adjustment operation is able to make the search process quickly converge to the near optimum, while the random sampling operation aims at maintaining the diversity of the harmony memory. Accordingly, the proposed ISHS can keep a good balance between exploration and exploitation, which is helpful for enhancing the total search efficiency of the proposed ISHS. Based on these considerations, the pitch adjustment operation of GHS [10] is incorporated into ISHS to enhance the exploitation ability, which is defined by:

$$\begin{aligned} &k = \text{randInt}(1, D) \\ &v_j = x_{Best,k} \end{aligned} \tag{3}$$

where randInt$(1, D)$ is an integer randomly generated in the range $[1, D]$.

In the improved search strategy, it has an important control parameter PAR, which is used to control the probability of executing the pitch adjustment operation. As seen, the ISHS performance is influenced by the parameter PAR. To enhance the robustness of the improved search strategy, the parameter PAR is adaptively tuned by using the scheme of IHS [9], which is defined as follows:

$$PAR = 0.01 + 0.98 \times \frac{t}{MaxT} \tag{4}$$

where t is the current generation and $MaxT$ is the maximum generation. From the mechanism of tuning the parameter PAR, it can be concluded that the value of PAR is small at the beginning of the search process, which indicates the probability of executing the pitch adjustment operation is small. In other words, at the beginning of the search process, the probability of inheriting the search experience of the best harmony is small, which is coincident with the expected property of an efficient EA. As the search proceeds, the probability of executing the pitch adjustment operation is gradually increased. Therefore, ISHS gradually switches its search behavior from exploration to exploitation with increasing generations.

3.2. **Algorithmic description of ISHS.** In the search process, ISHS first generates a random initial harmony memory, and then executes a loop of the improved search strategy. In the evolutionary loop, ISHS first takes a value for the control parameter PAR. After that, a new candidate harmony is created by the improved search strategy, and then the selection operation is conducted to select the better harmony for the next generation. The above steps are repeated generation after generation until the stopping criterion is satisfied. The framework of ISHS is described in Algorithm 1, where FEs denotes the number of fitness evaluations, $MaxFEs$ represents the maximum number of evaluations, and X^t_{worst} indicates the worst harmony in the current harmony memory.

4. **Experiments.**

4.1. **Experimental setup.** In this section, the proposed ISHS is applied to case studies. In the case studies, ISHS is employed to handle 13 well-known test functions with $D = 30$ and is compared with the basic HS, GHS, and NGHS. The details of the test functions can be seen in [13]. The parameter settings of ISHS are set to $HMS = 50$ and $HMCR = 0.96$, following the suggestions in [14]. To achieve fair comparisons, $MaxFEs$ is set to 200,000 for all HS, as recommended in [5]. In addition, the other parameters of HS, GHS, and NGHS are the same as their original papers.

4.2. **Comparisons and discussions.** The experimental results are described in Table 1, where "Mean" and "SD" indicate the mean and standard deviation of the optimization error values achieved by 30 independent runs, respectively. The symbols "+", "–", and "≈" represent that the proposed ISHS obtains better, worse, and similar results than the corresponding algorithms according to the two-tailed t-test, respectively. The results of HS, GHS, and NGHS are based on the reports in [5].

TABLE 1. Experimental results of HS, GHS, NGHS, and ISHS over 30 independent runs for the 13 classical test functions

Function	Mean ± SD			
	HS	GHS	NGHS	ISHS
$f1$	7.65E-05±7.74E-06+	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00
$f2$	2.08E-02±1.54E-03+	0.00E+00±0.00E+00≈	4.23E+01±5.99E+01+	0.00E+00±0.00E+00
$f3$	1.58E+03±6.91E+02≈	1.17E+02±1.15E+02-	9.54E-04±1.35E-03-	2.29E+03±3.24E+03
$f4$	9.87E-01±5.75E-02+	4.22E+00±3.26E+00+	3.97E-19±5.06E-19-	1.62E-12±2.23E-12
$f5$	5.15E+01±2.70E+01≈	4.76E+00±5.11E+00-	1.14E+01±8.36E+00-	5.11E+01±3.28E+01
$f6$	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00≈	0.00E+00±0.00E+00
$f7$	8.97E-03±2.71E-03+	5.52E-04±1.31E-04+	4.38E-03±2.27E-03+	1.04E-04±2.44E-04
$f8$	1.31E-04±1.50E-04+	3.26E-04±0.00E+00+	1.82E-11±4.46E-12+	3.64E-12±0.00E+00
$f9$	1.14E-02±2.11E-03+	0.00E+00±0.00E+00≈	1.42E-09±1.99E-09+	0.00E+00±0.00E+00
$f10$	5.89E-03±2.43E-04+	4.44E-16±0.00E+00-	8.71E-06±9.95E-06+	5.18E-15±1.67E-15
$f11$	1.35E-02±9.59E-03+	8.95E-04±1.27E-03+	1.72E-02±1.22E-02+	0.00E+00±0.00E+00
$f12$	4.78E-07±1.54E-07+	8.03E-07±0.00E+00+	7.98E-32±2.56E-32+	1.57E-32±0.00E+00
$f13$	1.54E-05±1.03E-05+	1.13E-05±0.00E+00+	6.81E-30±8.96E-30+	1.35E-32±0.00E+00
−	0	3	3	
+	10	6	8	
≈	3	4	2	

From the experimental results, it can be known that ISHS demonstrates superior performance on the majority of the test functions. Specifically, ISHS obtains better results than HS, GHS, and NGHS on 10, 6, and 8 out of 13 test functions, respectively, whereas GHS and NGHS are better than ISHS only on 3 and 3 test functions, respectively. In addition, HS cannot perform better than ISHS on any test function. Moreover, ISHS is similar to HS, GHS, and NGHS on 3, 4, and 2 test functions, respectively. We also conduct the average ranking of Friedman test on the experimental results [15, 16]. The average rankings of HS, GHS, NGHS, and ISHS are shown in Table 2. As known, ISHS obtains the highest average ranking. According to the above analyses, we can conclude that the improved search strategy can enhance the search ability of ISHS.

TABLE 2. Average rankings of HS, GHS, NGHS, and ISHS for the 13 test functions obtained by the Friedman test

Algorithm	Ranking
ISHS	**1.81**
GHS	2.35
NGHS	2.42
HS	3.42

5. **Conclusions.** The basic HS tends to suffer from poor exploitation ability when tackling some optimization problems with highly non-linear and multimodal characteristics. Aiming at this weakness of the basic HS, we propose a new harmony search with improved search strategy, called ISHS. In the proposed ISHS, the best harmony from the current harmony memory is used to guide the search direction towards the promising region. Moreover, the search experience of the best harmony is utilized to make the search process quickly converge to the near optimum. The proposed ISHS is evaluated on a set of 13 well-known test functions and is compared with three HS algorithms, namely, the basic HS, GHS, and NGHS. The experimental results demonstrate that the proposed ISHS can achieve better results than HS, GHS, and NGHS on the majority of the test functions. In the future, we will employ the proposed ISHS to tackle machine learning problems, such as classification and clustering problems.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China (Nos. 61662029, 61462036, and 41561091), the Natural Science Foundation of Jiangxi Province, China (Nos. 20151BAB217010 and 20151BAB201015), and the Education Department Scientific Research Foundation of Jiangxi Province, China (No. GJJ14433).

REFERENCES

[1] Z. Guo, X. Yue, K. Zhang, S. Wang and Z. Wu, A thermodynamical selection-based discrete differential evolution for the 0-1 knapsack problem, *Entropy*, vol.16, no.12, pp.6263-6285, 2014.
[2] Z. W. Geem, J. H. Kim and G. V. Loganathan, A new heuristic optimization algorithm: Harmony search, *Simulation*, vol.76, no.2, pp.60-68, 2001.
[3] Q. K. Pan, P. N. Suganthan, J. J. Liang and M. F. Tasgetiren, A local-best harmony search algorithm with dynamic sub-harmony memories for lot-streaming flow shop scheduling problem, *Expert Systems with Applications*, vol.38, no.4, pp.3252-3259, 2011.
[4] R. D. Zarro, U. K. Yusof and M. Sabudin, Maintenance cost optimization with replacement policy using island model and harmony search, *ICIC Express Letters*, vol.9, no.3, pp.713-720, 2015.
[5] Z. Guo, X. Yue, S. Wang, H. Yang and Z. Wu, Global-best harmony search algorithm with adaptive bandwidth, *ICIC Express Letters*, vol.9, no.11, pp.2887-2892, 2015.
[6] L. Yong and S. Liu, An improved harmony search algorithm with differential operator for absolute value equation, *ICIC Express Letters*, vol.8, no.4, pp.1151-1157, 2014.
[7] D. Manjarres, I. Landa-Torres, S. Gil-Lopez, J. Del 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.
[8] M. P. Saka, O. Hasançebi and Z. W. Geem, Metaheuristics in structural optimization and discussions on harmony search algorithm, *Swarm and Evolutionary Computation*, vol.28, pp.88-97, 2016.
[9] 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.
[10] M. G. Omran and M. Mahdavi, Global-best harmony search, *Applied Mathematics and Computation*, vol.198, no.2, pp.643-656, 2008.
[11] Q. K. Pan, P. N. Suganthan, M. F. Tasgetiren and J. J. Liang, A self-adaptive global best harmony search algorithm for continuous optimization problems, *Applied Mathematics and Computation*, vol.216, no.3, pp.830-848, 2010.
[12] D. Zou, L. Gao, S. Li and J. Wu, Solving 0-1 knapsack problem by a novel global harmony search algorithm, *Applied Soft Computing*, vol.11, no.2, pp.1556-1564, 2011.
[13] X. Yao, Y. Liu and G. Lin, Evolutionary programming made faster, *IEEE Trans. Evolutionary Computation*, vol.3, no.2, pp.82-102, 1999.
[14] M. El-Abd, An improved global-best harmony search algorithm, *Applied Mathematics and Computation*, vol.222, pp.94-106, 2013.
[15] H. Wang, Z. Wu, S. Rahnamayan, Y. Liu and M. Ventresca, Enhancing particle swarm optimization using generalized opposition-based learning, *Information Sciences*, vol.181, no.20, pp.4699-4714, 2011.
[16] Z. Guo, H. Huang, C. Deng, X. Yue and Z. Wu, An enhanced differential evolution with elite chaotic local search, *Computational Intelligence and Neuroscience*, vol.2015, Article ID 583759, 2015.