

# 和声搜索算法的改进研究

杨树欣<sup>1</sup>, 李盼池<sup>2</sup>

(1. 解放军第 302 医院 医学工程保障管理中心, 北京 100039;  
2. 东北石油大学 计算机与信息技术学院, 黑龙江 大庆 163318)

**摘要:**由于传统和声搜索算法每次迭代只生成一个和声向量,因此搜索能力受到限制。针对这一问题,提出一种改进的和声搜索算法。在改进的方法中,每次迭代首先生成与种群规模相同的和声向量,然后放入和声记忆库中并按适应度从高到低排序,最后取前一半个体进入下一代种群。实验结果表明,改进算法不仅在相同迭代步数下明显优于原算法,而且在相同运行时间下也明显优于原算法,从而揭示出该算法能有效地提取种群中的优良个体特征,获得满意的优化结果。

**关键词:**智能计算;智能优化;和声搜索;算法设计

**中图分类号:**TP183

**文献标识码:**A

**文章编号:**1673-629X(2015)04-0093-05

doi:10.3969/j.issn.1673-629X.2015.04.022

## Research on Improvement of Harmony Search Algorithm

YANG Shu-xin<sup>1</sup>, LI Pan-chi<sup>2</sup>

(1. Department of Medical Engineering Support Management Center, the 302 Hospital of PLA,  
Beijing 100039, China;

2. School of Computer and Information Technology, Northeast Petroleum University, Daqing 163318, China)

**Abstract:** The only one harmony vector is obtained in each of iteration in classical harmony search algorithm, which affects its search ability. In view of this, propose an improved harmony search algorithm in this paper. In this approach, an equivalent number of harmony vectors with population size are obtained in each of iteration, and the newly generated harmony vectors are put into the harmony memory array. Then, all harmony vectors are sorted according to the fitness from high to low, and the first half individuals are removed into the next generation of populations. Experimental results show that the proposed approach is obviously superior to the classical one under both the same number of iterations and the same running time, which reveals this approach can effectively extract the characteristics of excellent individuals in the population and obtain the satisfactory optimization results.

**Key words:** intelligent computation; intelligent optimization; harmony search; algorithm design

## 0 引言

现有的很多启发式算法都是对自然现象的模拟。例如模拟退火算法是对物理退火原理的模拟<sup>[1]</sup>,禁忌搜索算法是对人类智力和记忆的模拟<sup>[2]</sup>,进化算法是对自然进化的模拟<sup>[3]</sup>,粒子群算法<sup>[4]</sup>和蚁群算法<sup>[5]</sup>则是对群体智能的模拟。作为新型启发式算法的和声搜索算法(Harmony Search, HS)也不例外,它是对音乐家通过反复调整不同乐器音调使之达到最优美和声这一过程的模拟。音乐家的音乐创作通常由以下三步构成:

(1)演奏一段自己能精确记忆的已经是优美和声的著名曲目;

(2)演奏与自己记忆和声相似的和声(即在自己记忆的和声基础上微调);

(3)即兴创作音符来组成新和声。

Geem等正是受到以上三种音乐演奏方式的启发,将以上三种方式转化为数学公式,于2001年提出了和声搜索算法<sup>[6]</sup>。和声搜索算法提出之后的第一个实际应用,是成功解决了旅行商问题,目前和声算法的工程应用主要有:运输方案设计<sup>[7]</sup>、聚类分析<sup>[8]</sup>、多目标优化<sup>[9]</sup>、结构有限元模型修正<sup>[10]</sup>、PID控制参数设计<sup>[11]</sup>、工程优化<sup>[12-14]</sup>、车辆路径<sup>[15]</sup>等。然而传统的和声搜索算法每次迭代只生成一个和声向量,尽管有较高的计算效率,但优化能力并不理想。特别是对于高

收稿日期:2014-06-11

修回日期:2014-09-16

网络出版时间:2015-02-23

基金项目:黑龙江省教育科学技术研究项目(12541059);东北石油大学校青年基金资助项目(2013NQ119)

作者简介:杨树欣(1970-),男,硕士,高级工程师,研究方向为智能优化算法。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20150223.1239.034.html>

维复杂的非线性优化,很难获得全局最优解。针对这一问题,文中提出一种改进的和声搜索算法。该算法每次迭代生成与和声记忆库同样规模的和声向量,然后择优进入下一代和声记忆库。典型函数极值优化实验结果表明,提出的算法明显优于原算法。

## 1 基本和声搜索算法

基本和声搜索算法的执行步骤如下:初始化算法参数、初始化记忆库、创作新的和声、更新和声记忆库和判定终止条件。下面分别予以介绍。

### 1.1 参数初始化

和声搜索算法的参数主要有:和声记忆库大小(HMS)、记忆库取值概率(HMCR)、音调微调概率(PAR)、微调步长(bw)。除此之外,在算法运行前还需要设置终止条件,以及优化问题的约束条件等。对于一个非约束最优化问题,一般可描述为

$$\begin{cases} \min f(\mathbf{X}), \mathbf{X} = (x_1, x_2, \dots, x_n) \\ x_i \in [x_{i\min}, x_{i\max}], i = 1, 2, \dots, n \end{cases} \quad (1)$$

其中,  $f(\mathbf{X})$  为目标函数;  $\mathbf{X}$  为决策变量。

### 1.2 和声记忆库初始化

和声记忆库的初始化方法为将和声向量的每一维初始化为该维变量取值区间内的随机数,如式(2)所示。

$$x_i^k = x_{i\min} + (x_{i\max} - x_{i\min}) \times \text{rand}(0,1) \quad (2)$$

其中,  $\text{rand}(0,1)$  为(0,1)内均匀分布的随机数。

和声记忆库的一般形式如式(3)所示。

$$\text{HM} = \begin{bmatrix} \mathbf{X}^1 & f(\mathbf{X}^1) \\ \mathbf{X}^2 & f(\mathbf{X}^2) \\ \vdots & \vdots \\ \mathbf{X}^{\text{HMS}} & f(\mathbf{X}^{\text{HMS}}) \end{bmatrix} = \begin{bmatrix} x_1^1 & \cdots & x_n^1 & f(\mathbf{X}^1) \\ x_1^2 & \cdots & x_n^2 & f(\mathbf{X}^2) \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{\text{HMS}} & \cdots & x_n^{\text{HMS}} & f(\mathbf{X}^{\text{HMS}}) \end{bmatrix} \quad (3)$$

### 1.3 生成新和声向量

这是和声搜索算法最核心部分。生成新和声就是产生新的解向量。新和声主要通过以下三种方式产生。

- (1) 从和声记忆库中学习;
- (2) 在学习的基础上微调音调;
- (3) 随机创作一个新音调。

从记忆库中学习的概率大小主要由和声记忆库取值概率决定。对于一个新的和声向量  $\mathbf{X}^{\text{new}} = [x_1^{\text{new}}, x_2^{\text{new}}, \dots, x_n^{\text{new}}]$ , 若某个分量  $x_i^{\text{new}}$  从记忆库中取值, 则该分量可以从记忆库第  $i$  列所有元素中随机取一个值。

在执行这一过程前,首先生成区间(0,1)内的一个随机数  $\text{rand}_1$ , 然后按式(4)进行操作。

$$\begin{cases} x_i^{\text{new}} = x_i \in \{x_i^1, x_i^2, \dots, x_i^{\text{HMS}}\}, \text{rand}_1 < \text{HMCR} \\ x_i^{\text{new}} = x_{i\min} + (x_{i\max} - x_{i\min}) \text{rand}, \text{rand}_1 \geq \text{HMCR} \end{cases} \quad (4)$$

由式(4)表明,新向量中的任意一个音调有 HMCR 的概率从记忆库中取值,有(1-HMCR)的概率从取值空间中随机取值。

当第  $i$  个音调从记忆库中取值以后,此音调又有 PAR 的概率按式(5)被微调。

$$\begin{cases} x_i^{\text{new}} = x_i^{\text{new}} \pm \text{rand}(0,1) \times \text{bw}, \text{rand}_2 < \text{PAR} \\ x_i^{\text{new}} = x_i^{\text{new}}, \text{rand}_2 \geq \text{PAR} \end{cases} \quad (5)$$

从式(4)、(5)可以看出,微调是在从记忆库中取出音调的基础上,再加上一个介于  $\pm \text{bw}$  之间的随机调整步长。

### 1.4 更新和声记忆库

新的和声向量生成以后,首先用目标函数对其进行评价。若新和声优于记忆库中最差和声,则用此新和声向量替换和声记忆库中的最差和声。对于如式(1)所示的最小值优化问题,若新和声的函数值小于记忆库中最差和声的函数值,则认为新和声优于记忆库中最差和声,此时就将最差和声从记忆库中删除,并将新和声放入记忆库中。

### 1.5 判定终止条件

检查是否满足终止条件,若不满足则返回步骤(3),否则结束。对于终止条件可以有多种形式,可以设置精度阈值,例如对于最小值优化,当优化结果小于该阈值时,算法终止;也可以设置最大迭代步数。

基本和声搜索算法的流程如图1所示。从图1可

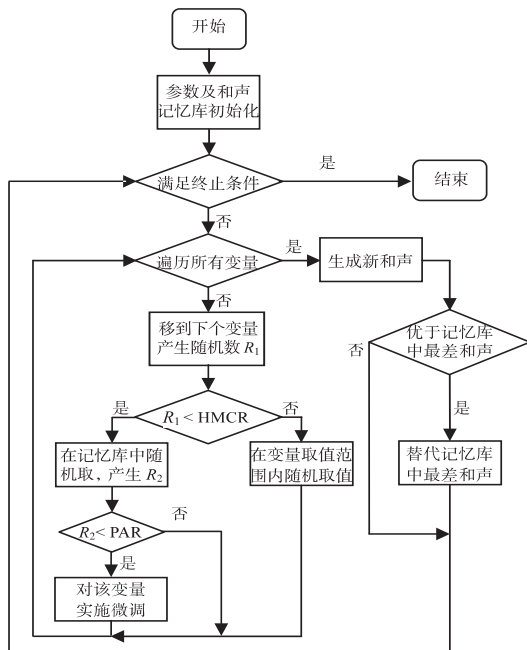


图1 基本和声算法流程图

以看出,和声搜索算法在形成新和声时有很多随机项,因此从总体上来说,HS 属于随机搜索算法,然而它是含有导向性的随机搜索算法,其导向性主要体现在对和声记忆库的学习和更新上。

## 2 改进的和声搜索算法

文中提出的改进措施,主要体现在对基本和声搜索算法实施步骤(3)和(4)的改进上。

从图1可以看出,在基本和声搜索算法的步骤(3)和步骤(4)中,每步迭代仅仅生成一个新的和声向量,并且用该向量替换和声记忆库中的最差向量。这种更新和声记忆库的方式,虽然有较高的计算效率,但搜索效率并不理想。针对这一问题,以极小值优化为例,提出的改进措施是:令和声记忆库中和声向量数为 $M$ ,每步迭代采用与基本和声算法相同的方式,循环产生 $M$ 个新的和声向量,然后将这些新产生的和声向量放入和声记忆库,将记忆库中所有 $2M$ 个和声向量按目标函数值从低到高排序,最后选出前面 $M$ 个具有低目标函数值的和声向量进入下一代和声记忆库。改进后的算法流程如图2所示。

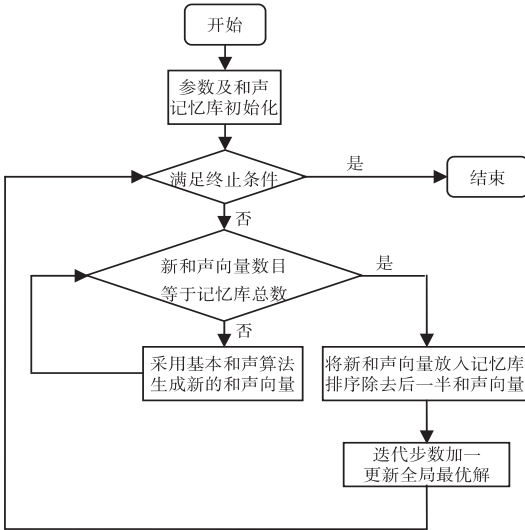


图2 改进和声算法流程图

## 3 对比实验

### 3.1 测试函数

为验证改进算法的优越性,本节采用如下9个典型的高维函数进行测试,并与基本和声搜索算法进行对比分析。

$$f_1(X) = \sum_{i=1}^D (\lfloor x_i + 0.5 \rfloor)^2$$

$$x_i \in [-100, 100], f_1(X^*) =$$

$$f_1(0, 0, \dots, 0) = 0$$

$$f_2(X) = 418.982\ 887 - \frac{1}{D} \sum_{i=1}^D x_i \sin(\sqrt{|x_i|})$$

$$x_i = [-500, 500], x_i^* = 420.968\ 746,$$

$$f_2(X^*) = 0$$

$$f_3(X) = \sum_{i=1}^D u(x_i, 10, 100, 4) + \frac{\pi}{D} \{10 \sin^2(\pi y_1) +$$

$$\sum_{i=1}^D (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_D - 1)^2\}$$

$$y_i = 1 + \frac{1}{4}(x_i + 1),$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & a \leq x_i \leq a \\ k(-x_i - a)^m, & x_i < -a \end{cases},$$

$$x_i = [-100, 100], X^* = [-1, -1, \dots, -1], f_3(X^*) = 0$$

$$f_4(X) = \frac{1}{10} \{ \sin^2(3\pi x_1) + \sum_{i=1}^{D-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_D - 1)^2 [1 + \sin^2(2\pi x_D)] \} + \sum_{i=1}^D u(x_i, 5, 100, 4)$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a \leq x_i \leq a \\ k(-x_i - a)^m, & x_i < -a \end{cases},$$

$$x_i = [-100, 100], X^* = [1, 1, \dots, 1],$$

$$f_4(X^*) = f_4(1, 1, \dots, 1) = 0$$

$$f_5(X) = -29.630\ 883\ 850\ 324\ 40 -$$

$$\sum_{i=1}^D \sin(x_i) \sin^{20}\left(\frac{ix_i^2}{\pi}\right)$$

$$x_i = [0, \pi], f_5(X^*) = 0$$

$$f_6(X) = \sum_{i=1}^D |x_i|^{(i+1)}$$

$$x_i = [-100, 100], f_6(X^*) =$$

$$f_6(0, 0, \dots, 0) = 0$$

$$f_7(X) = \sum_{i=1}^{D-1} (0.5 +$$

$$\frac{\sin^2 \sqrt{100x_i^2 + x_{i+1}^2} - 0.5}{1 + 0.001(x_i^2 - 2x_i x_{i+1} + x_{i+1}^2)^2})$$

$$x_i = [-100, 100], f_7(X^*) =$$

$$f_7(0, 0, \dots, 0) = 0$$

$$f_8(X) = 1 - \exp(-0.5 \sum_{i=1}^D x_i^2)$$

$$x_i = [-1, 1], f_8(X^*) = f_8(0, 0, \dots, 0) = 0$$

$$f_9(X) = -[A \prod_{i=1}^D \sin(x_i - z) +$$

$$\prod_{i=1}^D \sin(B(x_i - z))] + A + 1$$

$$x_i = [30, 30 + \pi], A = 2.5, B = 5, z = 30,$$

$$f_9(\mathbf{X}^*) = f_9(\pi/2 + z, \cdots, \pi/2 + z) = 0$$

以上 9 个函数均为极小值优化,其中  $D$  为优化空间维数,  $\mathbf{X}^*$  为精确全局最优解,  $f(\mathbf{X}^*)$  为  $\mathbf{X}^*$  的目标函数值。

3.2 相关定义

精度阈值  $\varepsilon$ :描述算法收敛与否的参数,在最大迭代步数 (Maximum Iterative Steps, MIS) 内,若  $|f(\mathbf{X}) - f(\mathbf{X}^*)| < \varepsilon$ ,认为算法收敛,否则认为不收敛。

误差  $E$ :一个优化解  $\mathbf{X}$  的误差定义为  $E = |f(\mathbf{X}) - f(\mathbf{X}^*)|$ 。

迭代步数 (Iterative Steps, IS):算法收敛时经过的迭代步数,若不收敛,则置 IS=MIS。

运行时间:算法独立完成一次优化所需平均时间。

终止条件:当达到最大迭代步数,算法终止。

3.3 参数设置

为体现对比的公平性,改进算法与基本算法取相同的算法参数,这些参数的设置如下:和声记忆库容量 HMS=30;记忆库取值概率 HMCR=0.9;音调微调概率 PAR=0.3;音调微调宽度 bw=0.001。目标函数的参数设置为:空间维数  $D=30$ ;精度阈值  $\varepsilon=0.1$ 。考虑到改进算法计算量较大,经过多次仿真发现,其运行时间大约为基本算法的 4 到 9 倍,为增强改进算法优良性能的可信度,有必要实施在相同时间下的对比。因此,对于改进算法,最大迭代步数 MIS=10<sup>4</sup>;对于基本算法,分别取 MIS=10<sup>4</sup>和 MIS=10<sup>5</sup>。为体现改进算法的优越性,并降低算法的随机性,每个函数分别用改进算法和基本算法独立优化 50 次,并以 50 次优化中,收敛次数,误差  $E$  的均值、方差,迭代步数 IS 的均值、方差,时间均值,作为对比指标。

3.4 对比结果

两种算法均采用 Matlab7.0 在主频 2.0 GHz,内存为 1.0 G 的微机上实现。50 次优化结果中,收敛次数、误差均值、误差方差的对比如表 1 所示,步数均值,步数方差,时间均值的对比如表 2 所示。

实验结果表明,就对比的各项指标而言,不仅当两种算法采用相同的迭代步数时,改进算法对所有 9 个标准测试函数的优化结果均明显优于基本算法,而且当对基本算法的迭代步数增加 9 倍,使其运行时间明显多于改进算法时,改进算法的各项指标也同样明显优于基本算法。

对于上述实验结果,可作如下分析。和声搜索算法的核心机制,是通过共享整个种群信息构造性能优良的新个体。逐代构造的新个体起着优化路标的作用,引导着优化过程逐渐向着最优解前进。显然新个体的数量决定着种群的更新程度,也决定着逼近全局最优界的速率。然而在基本算法中,每代只构造一个

新个体,即基本算法相邻两代至多只有一个不同个体,由于具有路标作用的指导个体太少,因而严重降低了优化效率。另一方面,由于相邻两代种群之间差异太小,也极易陷入早熟收敛,比如在函数  $f_7$  中,就出现了这种情形。

表 1 改进算法与基本算法的收敛次数、误差均值、误差方差的对比

函数	算法	收敛次数	误差均值	误差方差
$f_1$	改进算法 (10 <sup>4</sup> )	48	0.040 0	0.039 2
	基本算法 (10 <sup>4</sup> )	0	398.76	9.9e+003
	基本算法 (10 <sup>5</sup> )	0	3.940 0	2.139 2
$f_2$	改进算法 (10 <sup>4</sup> )	50	0.034 8	2.25e-004
	基本算法 (10 <sup>4</sup> )	0	15.197	13.528 0
	基本算法 (10 <sup>5</sup> )	0	0.375 8	0.019 4
$f_3$	改进算法 (10 <sup>4</sup> )	50	0.003 1	1.25e-005
	基本算法 (10 <sup>4</sup> )	0	1.0e+004	4.2e+008
	基本算法 (10 <sup>5</sup> )	29	0.103 4	0.005 0
$f_4$	改进算法 (10 <sup>4</sup> )	49	0.052 6	4.21e-004
	基本算法 (10 <sup>4</sup> )	0	3.8e+005	1.1e+011
	基本算法 (10 <sup>5</sup> )	0	0.860 4	0.115 2
$f_5$	改进算法 (10 <sup>4</sup> )	50	0.044 6	2.64e-004
	基本算法 (10 <sup>4</sup> )	0	2.159 1	0.147 6
	基本算法 (10 <sup>5</sup> )	4	0.162 2	0.002 6
$f_6$	改进算法 (10 <sup>4</sup> )	47	0.019 4	0.001 7
	基本算法 (10 <sup>4</sup> )	0	7.7e+018	2.6e+039
	基本算法 (10 <sup>5</sup> )	0	125.53	1.04e+005
$f_7$	改进算法 (10 <sup>4</sup> )	22	0.156 4	0.014 8
	基本算法 (10 <sup>4</sup> )	0	0.499 4	1.0e-007
	基本算法 (10 <sup>5</sup> )	0	0.362 4	0.010 7
$f_8$	改进算法 (10 <sup>4</sup> )	50	4.73e-007	4.75e-015
	基本算法 (10 <sup>4</sup> )	50	0.019 5	3.3e-005
	基本算法 (10 <sup>5</sup> )	50	2.06e-006	1.31e-012
$f_9$	改进算法 (10 <sup>4</sup> )	50	1.35e-005	4.09e-012
	基本算法 (10 <sup>4</sup> )	0	0.871 6	0.014 4
	基本算法 (10 <sup>5</sup> )	50	4.53e-004	3.53e-007

而对于改进算法,每代构造的和声数量与种群规模相等,通过与上代种群混合排序决定新个体的去留。这种方式较大程度地提高了当前种群中新个体的数量,也提高了相邻两代种群的更新程度。改进算法在较多路标个体的指引下,显著提高了优化效率,同时也有效避免了早熟收敛。值得指出,改进算法由于加大了计算量,因而运行时间有所延长,但改进算法正是以牺牲运行时间来换取优化能力提升的,这与无免费午餐定理的结论是一致的。另外,实验也表明,欲提高基本算法的优化能力,只能从改善搜索机制入手,而单靠



增加迭代步数的方法是不可行的。

表2 改进算法与基本算法的步数均值、步数方差、时间均值的对比

函数	算法	步数均值	步数方差	时间均值/s
$f_1$	改进算法( $10^4$ )	6 319	3.2e+006	2.010 3
	基本算法( $10^4$ )	10 000	0	0.501 9
	基本算法( $10^5$ )	100 000	0	24.542 5
$f_2$	改进算法( $10^4$ )	5 754	1.2e+006	2.715 3
	基本算法( $10^4$ )	10 000	0	0.550 0
	基本算法( $10^5$ )	100 000	0	25.222 8
$f_3$	改进算法( $10^4$ )	3 245	8.4e+005	7.610 6
	基本算法( $10^4$ )	10 000	0	0.873 8
	基本算法( $10^5$ )	90 174	1.48e+008	28.225 0
$f_4$	改进算法( $10^4$ )	7 886	9.6e+005	7.175 3
	基本算法( $10^4$ )	10 000	0	0.862 2
	基本算法( $10^5$ )	100 000	0	28.200 3
$f_5$	改进算法( $10^4$ )	5 067	1.9e+006	4.100 6
	基本算法( $10^4$ )	10 000	0	0.672 8
	基本算法( $10^5$ )	98 670	2.99e+007	26.522 8
$f_6$	改进算法( $10^4$ )	6 351	2.7e+006	3.887 2
	基本算法( $10^4$ )	10 000	0	0.676 9
	基本算法( $10^5$ )	100 000	0	26.440 3
$f_7$	改进算法( $10^4$ )	8 426	4.5e+006	6.858 4
	基本算法( $10^4$ )	10 000	0	0.743 1
	基本算法( $10^5$ )	100 000	0	27.388 1
$f_8$	改进算法( $10^4$ )	1 024	328.367 3	1.897 2
	基本算法( $10^4$ )	3 017	2.5e+005	0.529 1
	基本算法( $10^5$ )	30 652	3.23e+005	28.237 8
$f_9$	改进算法( $10^4$ )	1 224	1.69e+004	2.644 4
	基本算法( $10^4$ )	10 000	0	0.556 6
	基本算法( $10^5$ )	36 847	2.38e+007	36.028 8

4 结束语

文中提出了一种改进的和声搜索算法,该算法与基本和声搜索算法的区别在于,每次迭代都生成与种群规模相等的新个体。典型函数极值的优化结果表明,该方法切实能够大幅度提升基本和声搜索算法的优化能力。

参考文献:

[1] 杨艳霞.一种基于模拟退火操作的混合差分进化算法[J].智能系统学报,2014,9(1):109-114.

[2] 冶晓隆,兰巨龙,郭 通.基于 PCA 和禁忌搜索的网络流量特征选择算法[J].计算机科学,2014,41(1):187-191.

[3] 董 宁,王宇平.求解约束优化问题的偏好多目标进化算法[J].西安电子科技大学学报:自然科学版,2014,41(1):98-104.

[4] 李新鹏,张超勇,高 亮,等.基于元胞粒子群算法的数控切削参数优化[J].计算机工程与应用,2014,50(2):252-257.

[5] 赖智铭,郭躬德.基于自适应阈值蚁群算法的路径规划算法[J].计算机系统应用,2014,23(2):113-118.

[6] Geem Z W, Kim J H, Loganathan G V. A new heuristic optimization algorithm: harmony search [J]. Simulation, 2001, 76(2):60-68.

[7] 赖志柱.和声搜索算法优化多时间窗多式联运运输方案[J].计算机应用,2013,33(9):2640-2642.

[8] 依玉峰,高立群,郭 丽.和声搜索算法在聚类分析中的应用[J].东北大学学报:自然科学版,2012,33(1):47-51.

[9] 郝 冰,任献花,高岳林,等.和声搜索-分布估计混合算法求解多目标优化问题[J].计算机应用研究,2012,29(5):1659-1661.

[10] 杜永峰,李万润,李 慧,等.和声搜索算法在结构有限元模型修正中的应用[J].兰州理工大学学报,2013,39(5):106-110.

[11] 邹德旋,高立群,吴沛锋,等.一种全局和声搜索算法及在PID控制中的应用[J].东北大学学报:自然科学版,2010,31(11):1534-1537.


[12] Fesangharya M, Mahdavi M, Jolandan M M, et al. Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems[J]. Computer Methods in Applied Mechanics and Engineering, 2008, 197(33-40):3080-3091.

[13] Jaberipour M, Khorram E. Two improved harmony search algorithms for solving engineering optimization problems[J]. Communications in Nonlinear Science and Numerical Simulation, 2010, 15(11):3316-3331.

[14] Mahdavi M, Fesanghary M D E. An improved harmony search algorithm for solving optimization problems[J]. Applied Mathematics and Computation, 2007, 188(2):1567-1579.

[15] Geem Z W, Lee K S, Park Y. Application of harmony search to vehicle routing [J]. American Journal of Applied Sciences, 2005, 2(12):1552-1557.

和声搜索算法的改进研究

作者：[杨树欣](#)，[李盼池](#)，[YANG Shu-xin](#)，[LI Pan-chi](#)  
作者单位：[杨树欣, YANG Shu-xin\(解放军第302医院 医学工程保障管理中心, 北京, 100039\)](#)，[李盼池](#)  
[, LI Pan-chi\(东北石油大学 计算机与信息技术学院, 黑龙江 大庆, 163318\)](#)  
刊名：[计算机技术与发展](#)  
英文刊名：[Computer Technology and Development](#)  
年，卷(期)：2015(4)

引用本文格式：[杨树欣](#), [李盼池](#), [YANG Shu-xin](#), [LI Pan-chi](#) [和声搜索算法的改进研究](#)[期刊论文]-[计算机技术与发](#)  
[展](#) 2015(4)