

反向学习全局和声搜索算法

翟军昌^{1†}, 秦玉平²

(1. 渤海大学 信息科学与技术学院, 辽宁 锦州 121013; 2. 渤海大学 工学院, 辽宁 锦州 121013)

摘要: 提出一种反向学习全局和声搜索(OLGHS)算法. 基于反向学习技术初始化和声记忆库, 提高初始和声向量的质量; 通过当前最差和声向当前最优和声学习进化, 提高算法的全局搜索性能; 通过其他和声向量之间不断回溯交互的随机学习策略, 提高算法局部搜索性能; 用由两种不同学习策略随机交叉动态产生的新和声与反向和声二者较优的个体更新和声记忆库, 提高算法的搜索性能. 将 OLGHS 算法与其他启发式优化算法以及目前文献中较优的改进 HS 算法进行性能测试, 测试结果表明 OLGHS 算法具有较高的寻优精度和较快的收敛速度.

关键词: 和声搜索; 反向学习; 随机学习; 变异; 回溯; 局部最优

中图分类号: TP301.6

文献标志码: A

Opposition-based learning in global harmony search algorithm

ZHAI Jun-chang^{1†}, QIN Yu-ping²

(1. College of Information Science and Technology, Bohai University, Jinzhou 121013, China; 2. College of Engineering, Bohai University, Jinzhou 121013, China)

Abstract: This paper proposes an opposition-based learning global harmony search (OLGHS) algorithm. An opposition-based learning initialization technique is employed for initialize the harmony memory to enhance the quality of the initial harmony vector. The worst harmony learns from the best harmony, which can improve the global search performance of the algorithm. The local search performance of the algorithm is enhanced by means of random learning strategy of backtracking interaction among other harmony vectors. The new harmony is dynamically generated by means of random global crossover with two different learning strategies, and the harmony memory is updated by the optimal individual of the improvising harmony and its opposition harmony. Finally, a comparison test with other heuristic optimization algorithms and HS variants is carried out to test the optimization performance of the proposed algorithm. The simulation results demonstrate the OLGHS algorithm has higher convergence precision and convergence rate.

Keywords: harmony search; opposition based learning; random learning; mutation; back tracking; local optimum

0 引 言

近年来, 基于进化、群体智能、物理等模拟自然现象的随机优化方法受到越来越多的关注^[1-7]. Geem 等^[7]受音乐创作这一现象的启发, 提出了一种新的启发式优化算法和声搜索(Harmony search, HS). HS 算法通过和声记忆库考虑、随机选取和基因调整的策略, 分别对优化问题的每个决策变量在一定的范围内不断地搜索和调整, 从而获得最优解.

针对 HS 算法存在的不足, 文献[8-10]提出了动态参数调整策略, 虽然提高了算法的优化性能, 但又引入了一些新的静态参数, 降低了算法的适用性. 文献[11-13]将 HS 算法与其他启发式优化算法进行融合, 以提高算法的优化性能, 同时也增加了算法处理

的复杂度和过多的主观参数设置. 文献[14-17]修改了 HS 算法即兴创作产生新和声的机理, 通过当前最好和声引导产生新和声. 这种方法虽然可以使和声记忆库中的和声快速向最好和声聚集, 使算法具有较好的全局搜索性能, 同时也很容易忽略其他一些有价值的和声附近的信息, 降低了算法局部搜索的性能, 使算法容易陷入局部最优. 最近, 文献[18-19]将反向学习技术引入 HS 算法中, 提高了算法对解空间信息的开发能力和算法的优化性能. 上述研究成果虽然使 HS 算法的性能得到了不断的改进, 但仍然存在收敛速度慢、易陷入局部最优等问题.

受反向学习和 NGHS 算法^[15]最优和声引导即兴创作思想的启发, 本文提出反向学习全局和声搜索

收稿日期: 2017-12-21; 修回日期: 2018-07-18.

基金项目: 国家自然科学基金项目(61603055); 辽宁省自然科学基金重点项目(20170540011); 辽宁省教育厅项目(LQ2017003); 辽宁省博士启动项目(201601349); 教育部人文社会科学研究青年基金项目(15YJC870021).

责任编辑: 刘德荣.

[†]通讯作者. E-mail: zhajunchang@163.com.

(Opposition-based learning in global harmony search, OLGHS)算法,通过反向学习与和声之间随机交叉学习的方式动态产生新和声,提高算法的优化性能.仿真实验验证了本文算法的有效性.

1 预备知识

1.1 HS算法

HS算法的主要思想是从随机产生的和声记忆库出发,基于对和声记忆库的考虑、随机选择和基音调整的操作策略产生候选解向量,然后将候选解向量与和声记忆库中的最差解向量进行适应度函数值对比,决定是否更新和声记忆库. HS算法的主要参数包括和声记忆库大小(HMS)、和声记忆库考虑概率(HMCR)、基音调整步长(bw)、基音调整概率(PAR)和迭代次数(J). HS算法的操作步骤包括:初始化优化问题和算法的参数、初始化和声记忆库、即兴创作、更新和声记忆库和判断终止准则5部分.关于HS算法的详细介绍请参见文献[7].

1.2 NGHS算法

文献[15]引入了位置更新和变异操作,以取代HS算法即兴创作过程中的和声记忆考虑和基音调整操作,并舍去了贪婪准则,直接用新和声取代和声记忆库中的最差和声,提出了一种新颖的全局和声搜索(NGHS)算法.在NGHS算法中,利用最差和声分量关于最优和声分量的对称区间进行位置更新,即

$$x_R = 2x_j^{\text{best}} - x_j^{\text{worst}}, \quad (1)$$

$$x_j^{\text{new}} = x_j^{\text{worst}} + \text{rand} \times (x_R - x_j^{\text{worst}}). \quad (2)$$

其中: x_R 表示 x_j^{worst} 关于 x_j^{best} 的对称点, x_j^{best} 和 x_j^{worst} 分别表示最优和声和最差和声向量的第 j 维分量,rand表示0~1之间的随机数, x_j^{new} 表示新生成和声向量 x^{new} 的第 j 维分量.

NGHS算法位置更新操作容易陷入局部最优,因此在小于变异概率 p_m 下引入变异操作,即

$$x_j^{\text{new}} = x_{jL} + \text{rand} \times (x_{jU} - x_{jL}), \quad (3)$$

x_{jL} 和 x_{jU} 分别代表第 j 维分量的下界和上界.

1.3 反向学习

反向学习技术是由Tizhoosh^[20]提出的一种通过寻找一个问题可行解的反向解,并对原始可行解和其反向解进行评估,从中选出较优的解作为下一代个体的优化学习策略.在反向学习中,反向点和反向学习优化定义如下.

定义1 假设 $x = (x_1, x_2, \dots, x_D)$ 为 D 维空间中任意一点,且 $x_1, x_2, \dots, x_D \in R, x_i \in [a_i, b_i]$,则与 x 对应的全局反向点定义为 $\text{ox} = (\text{ox}_1, \text{ox}_2, \dots, \text{ox}_D)$,其中 $\text{ox}_i = a_i + b_i - x_i$.

定义2 假设 $x = (x_1, x_2, \dots, x_D)$ 是 D 维空间中的任意一点,其全局反向点定义为 $\text{ox} = (\text{ox}_1, \text{ox}_2, \dots, \text{ox}_D)$,对于考虑最小化的问题,如果 $f(\text{ox}) < f(x)$,则有 $x = \text{ox}$,称为反向学习优化.

2 OLGHS算法

2.1 反向学习初始化和声记忆库操作

为了提高初始和声记忆库中和声的质量,本文通过反向学习优化的方法初始化和声记忆库,通过与虚拟和声记忆库的和声对比,选择其中优秀的和声建立初始和声记忆库.

假设和声记忆库中的第 i ($i = 1, 2, \dots, \text{HMS}$)个和声 x^i 的第 j ($= 1, 2, \dots, N$)维分量为 x_j^i ,则 x_j^i 随机初始化为

$$x_j^i = x_{jL}^i + (x_{jU}^i - x_{jL}^i) \times \text{rand}, \quad (4)$$

其中 x_{jL}^i 和 x_{jU}^i 分别为 x_j^i 的下限和上限.于是, x^i 的反向和声 ox^i 的第 j ($j = 1, 2, \dots, N$)维分量可始化为

$$\text{ox}_j^i = x_{jL}^i + x_{jU}^i - x_j^i. \quad (5)$$

如果 $f(\text{ox}^i) < f(x^i)$,则有 $x^i = \text{ox}^i$.

2.2 即兴创作

1) 最差和声向最优和声学习即兴创作.和声记忆库中最差和声向当前最优和声学习,动态产生新和声向量 x^{new} 的第 j 维分量 x_j^{new} ,即

$$x_j^{\text{new}} = x_j^{\text{worst}} + \text{rand} \times 2(x_j^{\text{best}} - x_j^{\text{worst}}). \quad (6)$$

其中: x_j^{best} 和 x_j^{worst} 分别表示最优和声和最差和声向量的第 j 维分量,rand表示0~1之间的随机数.最差和声向最优和声学习即兴创作过程如图1所示.

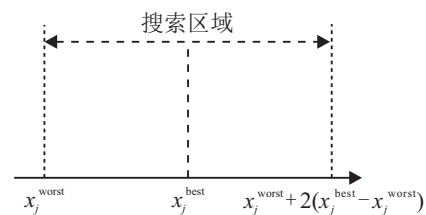


图1 最差和声向最优和声学习

由图1可知,通过最差和声在最优和声附近的对称区间内随机学习,可以有效开发最优和声解空间附近信息,使当前最差和声可以快速向最优和声聚集,提高算法的全局搜索性能.

这种快速的收敛过程使得算法在经过几代更新之后,大部分和声彼此接近.尤其在解决包含多个局部最优值的问题时,经过一定的迭代次数之后,和声向量集中于局部最优值附近,导致整个和声记忆库中的和声陷入一个狭小的空间内,从而使算法局部最优.为此,引入其他和声之间的相互学习策略.

2) 其他和声之间随机交互学习即兴创作. 算法在进化过程中,虽然有些和声不是当前最优和声,但其某些维可能对新和声创作提供更多有价值的信息. 如果当前最差和声向最好和声学习,没有对其他和声分量所在的解空间进行有效开发,则很容易忽略一些有价值声附近的信息. 因此,其他和声之间的随机学习可以增加其他和声之间彼此交互的机会,实现对该类和声附近的解空间信息进行有效的开发,为和声记忆库即兴创作提供更有价值的信息.

为了避免和声记忆库即兴创作过程中,最差和声向最优和声学习的单一性和对解空间信息的过渡开发,在OLGHS算法中,从和声记忆库中随机选择两个不同的和声向量 x^r 和 x^p , $p, r \in \{1, 2, \dots, HMS\}$,通过和声向量 x^r 和 x^p 的对应维进行随机交互学习产生新和声,即

$$x_j^{new} = x_j^r + rand \times (x_j^p - x_j^r), \quad (7)$$

其中 x_j^r 和 x_j^p 分别表示 x^r 和 x^p 的第 j 维分量. 其他和声之间随机交互学习过程如图2所示.

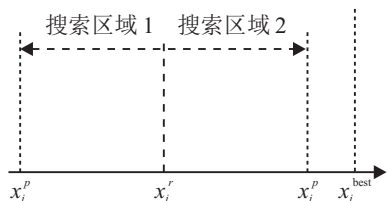


图2 随机交互学习

由图2可知:当 x_j^r 位于 x_j^p 左侧时,新和声向量分量 x_j^{new} 将在 x_j^r 与 x_j^p 之间的区域(搜索区域2)内随机搜索动态产生,并且在搜索的过程中不断向当前最优靠近;当 x_j^r 位于 x_j^p 右侧时,新和声向量分量 x_j^{new} 依然在 x_j^r 与 x_j^p 之间的区域(搜索区域1)内随机搜索动态产生,但此时的新和声向量分量 x_j^{new} 不再盲目地向当前最优 x_j^{best} 靠近,而是采用一种背离当前最优回溯的方式在 x_j^r 的左邻域内搜索. 这种策略通过一种不断回溯的搜索方式对 x_j^r 的左邻域信息进行更加精细的搜索,可以提高算法的局部搜索能力.

新的学习方式使新和声以一种不断回溯的搜索方法向当前最优和声靠近,实现了其他和声向量之间的交互,从而对其他和声附近的解空间信息有效开发,创作出更有潜力的和声,提高和声记忆库的多样性,为后续即兴创作产生新和声提供参考.

此外,新的学习方式避免了算法在进化后期和声向量一味地向当前最优和声向量学习的单一性,尤其当算法进化到后期时,整个和声记忆库的所有和声向量都接近于最优和声,并且聚集在一个狭小的空间内. 如果此时的和声还是盲目地向最优和声学习,则

容易导致算法对最优和声向量附近信息的过渡开采,而陷入局部最优难以跳出. 相对于过渡向最优和声学习的方式,通过其他和声之间不断回溯交互的学习策略开发解空间的信息,则可以避免这一问题.

3) 随机交叉学习策略. 为了使最差和声向最优和声学习与其他和声向量之间随机学习两种策略有效结合实现即兴创作,OLGHS算法将二者通过随机交叉的方式结合,提高了算法的全局搜索和局部搜索性能. 在和声记忆库即兴创作产生新和声向量 x^{new} 时,其对应的第 j 维分量 x_j^{new} 按照下面的规则执行:

```

if rand < rand
    执行式(6)
else
    执行式(7)
end
    
```

两种不同学习策略的结合,可以使新和声向量在解空间的不同区域内随机更新,通过一种动态的方式产生,实现全局搜索和局部搜索之间的平衡,加快了算法的收敛速度,提高了算法的优化性能.

4) 变异操作. 在即兴创作产生新和声向量的过程中,通过两种不同的学习策略实现和声向量的位置更新,增强了算法的收敛性和稳定性,而通过小概率的基因变异操作引入式(3)可以使算法提高摆脱局部最优点的能力,从而促使算法更快地找到全局最优点.

5) 产生反向和声向量. 根据新产生的和声向量 x^{new} ,即兴产生其对应的反向和声向量 ox^{new} . 反向和声向量 ox^{new} 对应的第 j ($j = 1, 2, \dots, N$)维分量按照下面的规则执行,即

$$ox_j^{new} = x_{jL} + x_{jU} - x_j^{new}. \quad (8)$$

2.3 更新和声记忆库

和声记忆库即兴创作产生的和声向量 x^{new} 与其反向和声向量 ox^{new} 中适应度较优的个体直接替换和声记忆库中的最差和声向量.

OLGHS算法的主要流程如图3所示.

2.4 收敛性分析

OLGHS算法在和声记忆库初始化阶段引入反向学习技术,保证了和声记忆库在初始化的同时搜索当前和声与反向和声,并选择较好的解作为初始和声,提高了和声记忆库初始和声的质量,为后续即兴创作产生高质量的和声提供了保障.

在即兴创作初期,通过两种不同学习策略随机交叉动态产生新和声. 第1种学习策略采用当前最差和声向当前最优和声学习进化,可以使当前最差和声快

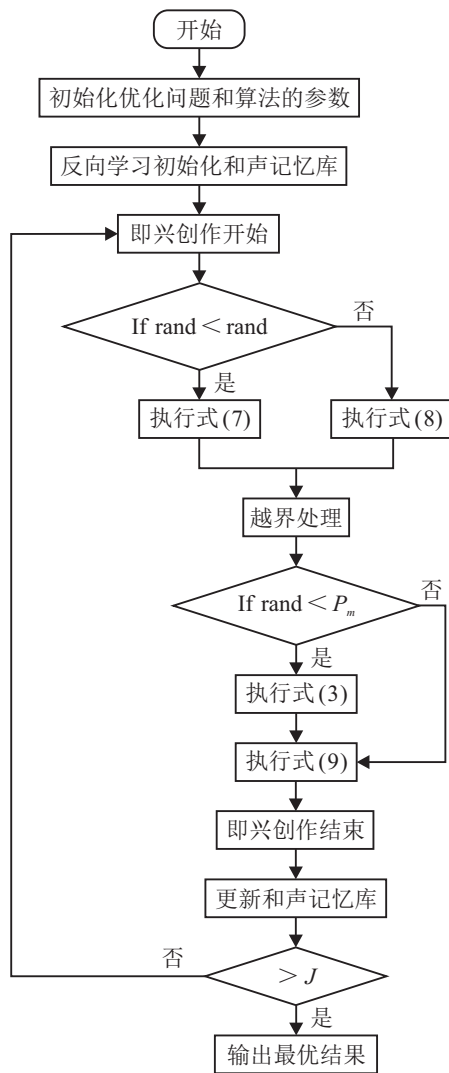


图3 OLGHS算法主要流程

速向最优和声聚集,使新产生的和声成为最优和声的可能性增加,提高了算法的全局搜索性能.在第2种学习策略中,通过其他和声向量之间不断回溯交互的随机学习策略,可以有效地开发其他和声附近解空间局部信息,提高算法局部搜索的性能.将两种学习策略随机交叉,使新和声在解空间不同区域内随机更新,可实现算法全局搜索和局部搜索之间的平衡.

此外,在即兴创作初期通过随机交叉学习策略使新和声动态产生的过程中,既保持了最优和声引导的特性,同时又继承了其他和声之间交互学习的经验,逐渐向最优和声聚集,避免了算法单一地依赖于最优和声引导很难摆脱局部最优的困扰,提高了算法跳出局部最优向全局最优收敛的能力.

在即兴创作后期,基于反向学习技术搜索反向解空间,将二者适应度值较优的和声存储于和声记忆库中,其本质仍然是一种贪婪的选择策略.因此,通过两种学习策略即兴创作动态产生的新和声,再结合反向学习策略创作新和声可以保证算法的收敛性.

3 仿真实验

为验证OLGHS算法的性能,首先,将其与其他启发式优化算法(差分进化(DE/rand/1)算法^[1]、粒子群优化(PSO)算法^[2]、人工蜂群(ABC)算法^[3]、正弦余弦(SCA)算法^[4]和灰狼(GWO)算法^[5])进行算法性能测试;然后,将OLGHS算法与HS算法^[7]和几种优秀的改进和声搜索算法(GHS算法^[14]、NGHS算法^[15]、IGHS算法^[16]、GDHS算法^[17]和GOGHS算法^[18])进行优化性能测试.在Windows XP下,以Matlab 7.0为仿真实验环境.实验中,选取优化算法10个经典标准测试函数,具体表达如下:

$$\min f_1(x) = \sum_{i=1}^D x_i^2,$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_2(x) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2],$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_3(x) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10),$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_4(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1,$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_5(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) -$$

$$\exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right) + 20 + e,$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_6(x) = \sum_{i=1}^D |x_i| + \prod_{i=1}^D |x_i|,$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_7(x) = 418.9829 - \sum_{i=1}^D x_i \sin(\sqrt{|x_i|}),$$

$-500 \leq x_i \leq 500$,全局最优为0;

$$\min f_8(x) = \sum_{i=1}^D \left(\sum_{j=1}^N x_j\right)^2,$$

$-100 \leq x_i \leq 100$,全局最优为0;

$$\min f_9(z) = \sum_{i=1}^D z_i^2 - 450,$$

$z = x - o$, $o = [o_1, o_2, \dots, o_n]$ 是偏移量, $-100 \leq x_i \leq 100$,全局最优为-450;

$$\min f_{10}(z) = \sum_{i=1}^D (z_i^2 - 10 \cos(2\pi z_i) + 10) - 330,$$

$z = x - o, o = [o_1, o_2, \dots, o_n]$ 是偏移量, $-5 \leq x_i \leq 5$, 全局最优为 -330.

仿真1 将 OLGHS 算法与其他启发式优化算法进行对比实验. 在仿真实验中: 本文 OLGHS 算法取 $HMS = 5, P_m = 0.005$; DE/rand/1、PSO、ABC、

SCA 和 GWO 五种算法的参数设置均选择参考文献中最优参数设置. 为了保证算法对比的公平性, 实验将每种算法对目标函数计算 60 000 次, 独立运行 30 次, 分别用 M 代表平均值, S 代表方差, 向量空间维数 $D = 50$. 对 10 个函数的测试结果如表 1 所示.

表 1 几种启发式优化算法对标准函数测试结果

函数	指标	DE/rand/1	PSO	ABC	SCA	GWO	OLGHS
f_1	M	1.10e-04	3.40e+00	3.70e+03	3.74e-01	1.38e-91	0
	S	5.37e-04	9.14e-01	2.03e+04	7.94e-01	3.92e-91	0
f_2	M	5.30e+01	1.41e+03	2.95e+02	5.05e+06	4.73e+01	4.67e+01
	S	6.32e+01	7.61e+02	8.75e+01	1.78e+07	6.42e-01	1.60e+00
f_3	M	1.37e+00	9.98e+01	1.19e+01	8.25e+01	0	0
	S	2.07e+00	2.35e+01	1.90e+00	5.91e+01	0	0
f_4	M	4.02e-02	1.01e+00	1.61e-03	5.31e-01	1.43e-03	0
	S	6.33e-02	2.70e-02	2.08e-03	5.11e-01	4.57e-03	0
f_5	M	5.34e-02	2.97e+00	8.00e-02	1.78e+01	1.47e-14	3.43e-15
	S	1.84e-01	7.34e-01	3.43e-02	6.27e+00	2.23e-15	6.49e-16
f_6	M	5.13e-01	3.74e+21	3.04e-02	1.10e-04	1.68e-52	0
	S	6.87e-01	2.03e+22	1.54e-02	2.89e-04	2.54e-52	0
f_7	M	6.65e-04	1.09e+04	1.54e+03	1.57e+04	1.18e+04	7.52e-04
	S	8.70e-05	1.14e+03	2.29e+02	3.69e+02	6.61e+02	2.75e-04
f_8	M	2.58e+02	5.24e+02	5.21e+04	2.20e+04	3.68e-17	0
	S	4.32e+02	1.04e+02	1.54e+04	1.08e+04	1.29e-16	0
f_9	M	-4.34e+02	1.48e+03	-4.46e+02	5.12e+04	5.29e+03	-4.50e+02
	S	6.78e+01	1.80e+03	1.93e+01	5.87e+03	2.82e+03	1.36e-04
f_{10}	M	6.41e+00	1.56e+01	-2.84e+02	2.50e+02	-7.53e+01	-3.30e+02
	S	6.58e+01	2.77e+01	1.80e+02	3.16e+01	3.48e+01	2.03e-03

由表 1 的实验结果可知, OLGHS 算法除了对函数 f_2 和 f_7 的优化结果稍差于 DE 和 GWO 算法外, 对其他 8 个函数的优化结果 (无论是平均值还是方差) 都优于其他几种启发式优化算法, 且 OLGHS 算法对函数 f_1 、 f_3 、 f_4 、 f_6 、 f_8 、 f_9 和 f_{10} 优化获得的平均值与最优解相同.

仿真2 将 OLGHS 算法与其他优秀改进 HS 算法在高维空间进行对比实验. 为了保证算法对比的公平性, 在仿真实验中, 所有算法取 $HMS = 5$, 每种算法用到的其他参数均选择参考文献中的最优设置, 具体为: OLGHS 算法, $P_m = 0.005$; HS 算法, $HMCr = 0.95, PAR = 0.33, bw = 0.01$; GHS 算法, $HMCr = 0.9, PAR_{min} = 0.1, PAR_{max} = 0.99$; NGHS 算法, $P_m = 0.005$; IGHS 算法, $HMS = 5, HMCr = 0.995, PAR = 0.4$; GOGHS 算法, $HMS = 5, P_m = 0.005$. 每种算法迭代 60 000 次, 独立运行 30 次, 分别用 B 代表最优值, W 代表最差值, M 代表平均值, S 代表方差, 向量空间维数 $D = 100$. 对 10 个函数的测试结果如表 2 所示.

由表 2 中对函数 f_1 、 f_3 、 f_4 、 f_6 和 f_8 的优化结果可以看出: HS 算法和 GHS 算法几乎陷入了局部最优; NGHS 算法和 GOGHS 算法的优化精度有所提

升, 但是其所得结果与最优解仍相差较大; IGHS 算法和 GDHS 算法的优化结果相对有较大的提升; 相比之下, 本文 OLGHS 算法的优化结果明显优于其他几种和声搜索算法, 除了函数 f_8 外, 本文 OLGHS 算法的最优值、最差值、平均值和方差均为 0.

由表 2 中对函数 f_2 、 f_5 和 f_7 的优化结果可以看出, 几种算法都陷入了局部最优. 其中: IGHS 算法对函数 f_2 的优化效果相对较好; OLGHS 算法对函数 f_5 优化所得平均值和最差值与最好值相同, 且方差为 0; IGHS 算法和 GDHS 算法对函数 f_7 优化所得最好值相对较好, 但本文 OLGHS 算法对函数 f_7 优化所得最好值的数量级与 IGHS 算法和 GDHS 算法的优化结果相同, 且 OLGHS 算法对函数 f_7 优化所得平均值、最差值、方差均优于其他几种算法所得优化结果.

实验中, 函数 f_9 和 f_{10} 分别是两个带有偏移量的单峰和多峰高维优化函数. 由表 2 中对函数 f_9 和 f_{10} 优化的结果可以看出: 本文 OLGHS 算法对函数 f_9 优化可以得到最优解, 且优化所得的平均值与 IGHS 算法相同, 但是 OLGHS 算法优化所得最差值则稍差于 IGHS 算法的优化结果; OLGHS 算法对函数 f_{10} 的优化结果均明显优于其他几种算法的优化结果.

表2 7种和声搜索算法对标准函数的测试结果

函数	指标	HS	GHS	NGHS	IGHS	GDHS	GOGHS	OLGHS
f_1	B	1.25e+03	2.43e+03	1.22e-02	2.87e-102	1.42e-57	4.12e-09	0
	W	2.91e+03	8.68e+03	4.31e-02	4.43e-93	3.07e-54	1.34e-07	0
	M	1.91e+03	4.64e+03	2.53e-02	2.50e-94	2.50e-55	3.42e-08	0
	S	3.15e+02	1.44e+03	7.84e-03	8.77e-94	5.77e-55	3.25e-08	0
f_2	B	8.40e+06	8.52e+07	2.21e+02	1.23e-28	8.35e+01	9.62e+01	9.59e+01
	W	3.51e+07	6.00e+08	9.32e+03	9.71e+01	5.77e+02	9.79e+01	9.86e+01
	M	1.70e+07	2.82e+08	1.83e+03	6.45e+01	1.93e+02	9.69e+01	9.80e+01
	S	5.80e+06	1.27e+08	2.65e+03	4.64e+01	1.01e+02	3.89e-01	7.83e-01
f_3	B	1.79e+02	6.87e+01	1.04e+01	0	0	3.13e-08	0
	W	2.53e+02	3.30e+02	2.17e+01	4.76e+02	1.22e+01	6.13e+00	0
	M	2.14e+02	1.63e+02	1.65e+01	1.54e+02	3.87e+00	4.78e-01	0
	S	1.92e+01	5.92e+01	3.14e+00	1.67e+02	3.47e+00	1.49e+00	0
f_4	B	1.32e+01	1.83e+01	7.23e-03	0	0	3.29e-09	0
	W	2.35e+01	6.39e+01	1.75e-01	1.11e-16	7.40e-03	1.02e-07	0
	M	1.85e+01	4.07e+01	3.05e-02	3.70e-18	7.40e-04	2.95e-08	0
	S	2.47e+00	1.08e+01	3.07e-02	2.03e-17	2.26e-03	2.45e-08	0
f_5	B	5.61e+00	5.30e+00	1.55e-02	3.55e-15	3.20e-14	9.26e-06	3.55e-15
	W	7.17e+00	1.08e+01	3.42e-02	7.11e-15	5.68e-14	4.69e-05	3.55e-15
	M	6.39e+00	8.08e+00	2.46e-02	5.21e-15	4.22e-14	2.07e-05	3.55e-15
	S	3.57e-01	1.09e+00	4.22e-03	1.80e-15	6.58e-15	8.95e-06	0
f_6	B	1.43e+02	2.37e+01	6.89e-01	1.07e-78	4.25e-15	1.74e-04	0
	W	2.57e+02	1.57e+02	1.32e+00	4.46e-70	8.32e-11	1.03e-03	0
	M	1.96e+02	9.38e+01	9.08e-01	1.53e-71	7.60e-12	5.17e-04	0
	S	3.01e+01	3.33e+01	1.61e-01	8.14e-71	1.88e-11	2.20e-04	0
f_7	B	2.01e+03	2.98e+02	2.82e-01	1.27e-03	1.27e-03	2.02e-01	6.28e-03
	W	3.49e+03	2.23e+03	4.78e+02	1.17e+04	2.44e+02	3.58e+02	1.18e+02
	M	2.58e+03	1.08e+03	9.36e+01	3.88e+02	4.66e+01	1.01e+02	4.47e+00
	S	3.57e+02	4.53e+02	1.11e+02	2.13e+03	7.60e+01	1.07e+02	2.17e+01
f_8	B	9.23e+04	2.52e+05	1.67e+04	3.50e+04	1.23e+04	7.73e-08	0
	W	1.51e+05	4.01e+05	3.23e+04	3.11e+05	2.02e+04	6.32e-03	1.96e-265
	M	1.16e+05	3.21e+05	2.58e+04	1.61e+05	1.67e+04	2.43e-04	6.52e-267
	S	1.64e+04	3.95e+04	3.95e+03	6.56e+04	2.05e+03	1.15e-03	0
f_9	B	1.07e+03	7.16e+04	-4.50e+02	1.66e+05	-4.50e+02	-4.50e+02	-4.50e+02
	W	2.18e+03	9.88e+04	-4.50e+02	2.03e+05	-4.50e+02	-4.50e+02	-4.50e+02
	M	1.51e+03	8.49e+04	-4.50e+02	1.87e+05	-4.50e+02	-4.50e+02	-4.50e+02
	S	2.57e+02	6.58e+03	7.68e-03	7.76e+03	1.50e-03	6.07e-03	4.06e-03
f_{10}	B	-2.38e+02	3.97e+02	-3.26e+02	9.70e+02	-3.06e+02	2.20e+04	-3.30e+02
	W	-2.02e+02	6.13e+02	-3.14e+02	1.12e+03	-2.86e+02	5.69e+04	-3.26e+02
	M	-2.20e+02	4.91e+02	-3.22e+02	1.06e+03	-2.95e+02	3.60e+04	-3.28e+02
	S	1.03e+01	4.56e+01	2.86e+00	4.26e+01	4.97e+00	7.91e+03	1.28e+00

对于表2中的实验结果,总体而言,OLGHS算法优化得到的结果均得到了有效的提升,且精度变化相对较小.由此可知,OLGHS算法比其他算法更稳定.

为了对比算法收敛速度和收敛精度,图4~图7

给出了函数 f_1 、 f_3 、 f_4 和 f_5 的最优值进化曲线.

由图4~图7可知,不论是从收敛速度还是收敛精度来看,引入新策略后的OLGHS算法的优化效果得到了明显的提升,均优于其他几种和声搜索算法.

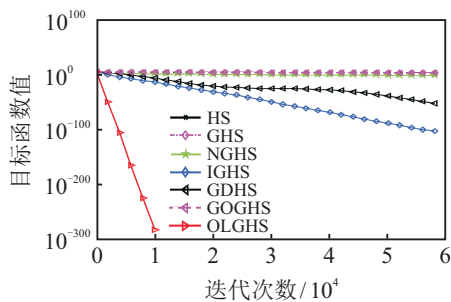


图4 f_1 最优值进化曲线

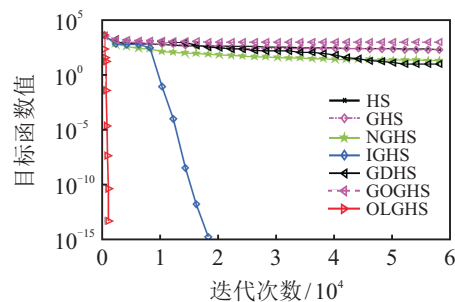


图5 f_3 最优值进化曲线

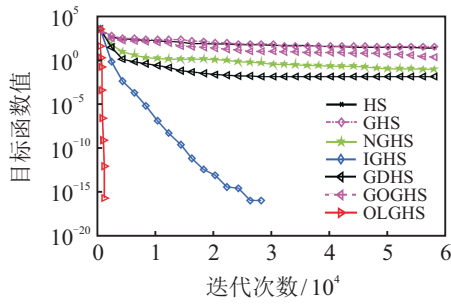


图6 f_4 最优值进化曲线

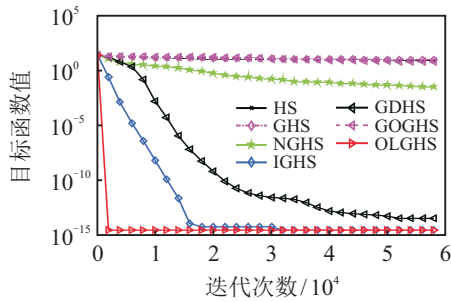


图7 f_5 最优值进化曲线

4 结论

本文针对HS算法容易过早收敛、容易陷入局部最优的问题,提出了OLGHS算法。OLGHS算法具有以下特点:一是反向学习策略初始化和声记忆库;二是通过不同的学习策略对解空间信息开发,从而即兴创作产生新和声,同时结合反向学习策略与其反向和声向量进行对比,选择其较优的个体更新和声记忆库。最后对OLGHS算法进行寻优性能测试,测试结果表明,OLGHS算法的寻优性能得到了有效的提升,验证了本文算法的有效性和稳定性。

参考文献(References)

[1] Storn R, Price K. Differential evolution — A simple and efficient heuristic for global optimization over continuous spaces[J]. J of Global Optimization, 1997, 11(4): 341-359.

[2] Eberhart R, Kennedy J. A new optimizer using particle swarm theory[C]. Proc of the 6th Int Symposium on Micro Machine and Human Science. Nagoya: IEEE, 1995: 39-43.

[3] Karaboga D. An idea based on honey bee swarm for numerical optimization[R]. Kayseri: Erciyes University, 2005: 1-10.

[4] Mirjalili S. SCA: A sine cosine algorithm for solving optimization problems[J]. Knowledge-Based Systems, 2016, 96: 120-133.

[5] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer[J]. Advances in Engineering Software, 2014, 69: 46-61.

[6] Ouyang H, Gao L, Li S, et al. Improved global-best-guided particle swarm optimization with learning operation for global optimization problems[J]. Applied Soft Computing, 2017, 52(C): 987-1008.

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

[8] Chen J, Pan Q, Li J. Harmony search algorithm with dynamic control parameters[J]. Applied Mathematics and Computation, 2012, 219(2): 592-604.

[9] Li X, Qin K, Zeng B, et al. A dynamic parameter controlled harmony search algorithm for assembly sequence planning[J]. The Int J of Advanced Manufacturing Technology, 2017, 92(9/12): 3399-3411.

[10] Kumar V, Chhabra J K, Kumar D. Parameter adaptive harmony search algorithm for unimodal and multimodal optimization problems[J]. J of Computational Science, 2014, 5(2): 144-155.

[11] Ouyang H, Gao L, Kong X, et al. Hybrid harmony search particle swarm optimization with global dimension selection[J]. Information Sciences, 2016, 346: 318-337.

[12] Zhao S Z, Suganthan P N, Pan Q K, et al. Dynamic multi-swarm particle swarm optimizer with harmony search[J]. Expert Systems with Applications, 2011, 38(4): 3735-3742.

[13] Bansal P, Sabharwal S, Mittal N. A hybrid artificial bee colony and harmony search algorithm to generate covering arrays for pair-wise testing[J]. Int J of Intelligent Systems and Applications, 2017, 9(8): 59-70.

[14] Omran M G H, Mahdavi M. Global-best harmony search[J]. Applied Mathematics and Computation, 2008, 198(2): 643-656.

[15] Zou D, Gao L, Wu J, et al. Novel global harmony search algorithm for unconstrained problems[J]. Neurocomputing, 2010, 73(16): 3308-3318.

[16] Valian E, Tavakoli S, Mohanna S. An intelligent global harmony search approach to continuous optimization problems[J]. Applied Mathematics and Computation, 2014, 232: 670-684.

[17] Khalili M, Kharrat R, Salahshoor K, et al. Global dynamic harmony search algorithm: GDHS[J]. Applied Mathematics and Computation, 2014, 228: 195-219.

[18] Guo Z, Wang S, Yue X, et al. Global harmony search with generalized opposition-based learning[J]. Soft Computing, 2017, 21(8): 2129-2137.

[19] Ouyang H, Gao L, Li S, et al. Improved harmony search algorithm: LHS[J]. Applied Soft Computing, 2017, 53: 133-167.

[20] Tizhoosh H R. Opposition-based learning: A new scheme for machine intelligence[C]. Int Conf on Computational Intelligence for Modelling, Control and Automation and Int Conf on Intelligent Agents, Web Technologies and Internet Commerce. Vienna: IEEE, 2005: 695-701.

作者简介

翟军昌(1978—),男,讲师,博士,从事智能优化算法及其应用等研究, E-mail: zhajunchang@163.com;
秦玉平(1965—),男,教授,博士,从事模式识别和机器学习等研究, E-mail: qinyujz@163.com.

(责任编辑: 齐 粟)