

## 基于动态参数的人工搜索群算法

陈堂功<sup>1,2</sup>, 刘超<sup>1,2†</sup>, 王梦莹<sup>1</sup>, 周小婷<sup>1</sup>, 黄涛<sup>1</sup>

(1. 河北工业大学 省部共建电工装备可靠性与智能化国家重点实验室, 天津 300130;

2. 河北工业大学 河北省电磁场与电器可靠性重点实验室, 天津 300130)

**摘要:** 人工搜索群算法(Artificial search swarm algorithm, ASSA)是受人类士兵通过信息交流完成目标搜索行为及过程启发而设计的一种仿生智能算法. 针对基本人工搜索群算法中固定参数可能导致算法过早陷入局部最优解的缺陷, 提出一种动态参数改进人工搜索群算法(Improved artificial search swarm algorithm, IASSA). 通过引入全局权重系数改善侦查行为中搜索群的历史经验位置, 从而加强全局最优个体对整个种群的引导作用; 利用动态协同参数提高士兵协同行为的搜索概率, 以加强种群之间局部信息交流; 采用动态步长策略提高算法的收敛速度和收敛精度; 为了检验改进算法的优化性能, 采用15个测试函数进行仿真实验. 实验结果表明, 所提出的改进算法可有效避免早熟现象, 在收敛速度和收敛精度上较基本人工搜索群算法和若干同类优化算法有显著提高.

**关键词:** 仿生智能算法; 人工搜索群算法; 全局权重系数; 动态协同参数; 动态步长; 函数优化

中图分类号: TP18

文献标志码: A

## Artificial search swarm algorithm based on dynamic parameters

CHEN Tang-gong<sup>1,2</sup>, LIU Chao<sup>1,2†</sup>, WANG Meng-ying<sup>1</sup>, ZHOU Xiao-ting<sup>1</sup>, HUANG Tao<sup>1</sup>

(1. State Key Laboratory of Reliability and Intelligence of Electrical Equipment, Hebei University of Technology, Tianjin 300130, China; 2. Key Laboratory of Electromagnetic Field and Electrical Apparatus Reliability of Hebei Province, Hebei University of Technology, Tianjin 300130, China)

**Abstract:** The artificial search swarm algorithm (ASSA) is a kind of bionic intelligent algorithm, which is inspired by the human communicating with each other in search. The fixed collaborative parameter and global weight in the standard artificial search swarm algorithm easily lead to local convergence and influence the convergence speed. Therefore, this paper proposes an improved artificial search swarm algorithm (IASSA) by introducing global weight, linearly dependent synergistic coefficient and dynamic step. The global weight improves the historical experience of the search group in reconnaissance behavior, so as to strengthen the guiding role of global optimal individual for all population. The dynamic collaborative parameters are used to improve the search probability of the soldier's cooperative behavior, so as to enhance the local information exchange between the populations. The dynamic step strategy is used to improve the convergence speed and convergence accuracy. The performance of the algorithm is verified on 15 benchmark functions. The results show that the improved algorithm can effectively avoid the local convergence and significantly improve the solution accuracy and convergence speed compared with the standard ASSA algorithm and the other three kinds of optimization algorithms.

**Keywords:** bionic intelligence algorithm; artificial search swarm algorithm; global weight; synergistic coefficient; dynamic step; function optimization

## 0 引言

优化设计是工程设计领域永恒的主题. 由于实际工程会面临很多难以处理的问题, 比如待处理问题的未知变量很多, 各子问题又存在耦合关系, 传统优化算法处理这类问题存在一定的局限性. 遗传算法(Genetic algorithm, GA)、差分进化算法(Differential evolution, DE)、和声搜索算法(Harmony search, HS)

以及粒子群算法(Particle swarm optimization, PSO)等仿生智能算法的提出为解决这类问题提供了新的思路. 目前, 国内外学者对智能优化算法领域进行了大量的研究和开发: 文献[1]模仿鱼群的觅食行为提出了人工鱼群算法(Artificial fish swarm algorithm, AFSA); 文献[2]模拟蜜蜂采蜜行为提出了人工蜂群算法(Artificial bee colony, ABC); 文献[3]提出了人工

收稿日期: 2018-01-18; 修回日期: 2018-06-07.

责任编辑: 张化光.

†通讯作者. E-mail: liuchao\_hebut@126.com.

藻类算法(Artificial algae algorithm, AAA). 由于智能算法具有控制参数少、易于实现和高鲁棒性等优点,大量的智能算法已被应用到农业、电力、航天等工程领域<sup>[4-7]</sup>.

人工搜索群算法(ASSA)是于2009年提出的一种新型仿生智能算法<sup>[8]</sup>,模拟人类士兵执行特定搜寻任务的特点和过程,通过执行相应的行为规则寻找搜索目标. 与其他智能算法相比,ASSA算法也存在着收敛速度慢和易陷入局部收敛的缺点. 为了解决这一问题,学者们提出了各种改进的算法理论. 智能算法的控制参数与其搜索能力密切相关,如何有效、合理地设置算法参数就显得尤为重要<sup>[9]</sup>. 文献[10]对PSO算法的 $w$ 参数进行了改进,提出了多种动态权重系数改进方案,加强了PSO全局搜索能力;文献[11]利用动态学习理论将父代种群分成多个子种群,不同的子种群进行搜寻的同时又相互交流,提高了算法的种群多样性. HS算法即兴创作过程中搜索步长直接影响了算法的探索能力,研究学者提出采用指数递减<sup>[12]</sup>、多项式<sup>[13]</sup>或者各种非线性<sup>[14]</sup>的动态步长调整策略,增加和声记忆库的多样性. 文献[15]为了改善进化算法对于多目标约束问题的搜索性能,提出了两种定向权重使优化信息直指Pareto前沿.

受以上改进算法的启发,本文提出一种基于动态参数的人工搜索群算法. 将全局惯性权重系数引入侦察阶段,增加搜索个体之间的局部信息交流. 同时,ASSA算法在计算初期协同行为的搜索频率较高,中后期几乎没有协同行为的参与. 这样的搜索机制限制了种群的多样性,针对这一缺点引入动态协同参数,从而提高协同行为的搜索率和种群多样性. 选取常用的测试函数检验新算法的优化性能,将优化结果与其他算法进行比较,以验证动态参数人工搜索群算法的优越性.

## 1 基本人工搜索群算法

人工搜索群算法以由搜索个体(模拟人类士兵)组成的搜索群作为算法的执行主体,把协同行进、侦查行进和随机行进作为搜索个体的行为规则,将优化设计问题的最优解作为搜索群的搜索目标,以目标函数适应值的大小作为衡量标准,引进迭代式运行机理,每个搜索个体在一次迭代中根据具体情况选择相应的行为规则. 随着迭代的进行,搜索群逐步向搜索目标行进,并最终接近或获得最优目标,相应的优化问题也得以解决.

人工搜索群搜索个体的3种行为规则的数学模型描述如下.

协同行进: 搜索过程中每个搜索个体之间都保

持着基本的通讯联系. 若有个体接到呼叫且满足

$$r_1 < P_c, \quad (1)$$

则该个体按下式向呼叫者的位置行进一步:

$$X_{i+1} = X_i + r_2 \times \text{Step} \times \frac{X_{\text{call}} - X_i}{\|X_{\text{call}} - X_i\|}; \quad (2)$$

若接到多次呼叫,则通过比较择优选取. 其中: $r_1, r_2$ 为 $[0, 1]$ 之间的均匀分布的随机数, $P_c$ 为搜索个体之间的协同率, $X_i$ 为搜索个体第 $i$ 次迭代的搜索位置,Step为每个个体的搜索步长, $X_{\text{call}}$ 为呼叫者的具体位置. 在第 $i$ 次迭代中,一旦搜索个体发现更优的适应值,则向整个搜索群体发出联络信号.

侦查行进: 当搜索个体没有接收到呼叫时,依据自身和搜索群体的经验按下式实施侦查:

$$X_{\text{reco}} = X_i + r_1 \times (X_s - X_i) + r_2 \times (X_g - X_i), \quad (3)$$

$$X_{i+1} = X_i + r_3 \times \text{Step} \times \frac{X_{\text{reco}} - X_i}{\|X_{\text{reco}} - X_i\|}. \quad (4)$$

其中: $X_{\text{reco}}$ 为侦查目标, $X_s$ 为搜索个体历史最优解, $X_g$ 为搜索群最优解, $r_3$ 为 $[0, 1]$ 之间的均匀分布的随机数. 在第 $i$ 次迭代中,一旦搜索个体发现更佳目标,则向整个群体发出呼叫并向该目标行进.

随机行进: 当搜索个体既没有发现新的更佳目标,又没有收到其他搜索个体的呼叫时,按下式向前随机行进一步:

$$X_{i+1} = X_i + r_1 \times \text{Step}. \quad (5)$$

其中: 在第 $i$ 次迭代中,一旦搜索个体发现更佳的目标,则向整个搜索群体发出呼叫.

ASSA算法具体流程如下.

Step 1: 根据具体优化对象设置各项参数,对于ASSA算法的初始种群进行随机初始化.

Step 2: 计算每个搜索个体的适应度值 $Y_{\text{object}} = f(x)$ .

Step 3: 根据每个搜索个体的具体搜索情况决定其下一步相应的搜索行为:

1) 当搜索个体收到其他个体的呼叫信号,并且满足式(1)时,按照协同行为进行搜索;

2) 若没有收到其他搜索个体的信号,则根据侦查行为向优化目标行进;

3) 若搜索个体既不执行协同行为也不执行侦查行为,则搜索个体按照随机行为在搜索空间中行进.

Step 4: 计算子代群体的适应度值,更新搜索群和搜索个体的历史记录.

Step 5: 判断是否满足终止条件. 若满足,则结束迭代,否则继续搜寻直至达到最大迭代数.

## 2 ASSA算法的改进

### 2.1 全局惯性权重

人工搜索群算法搜索个体搜寻的行为规则对算法的性能具有重要作用. 侦察行为可以帮助人们根据他和种群的历史经验找到更好的全局最优目标. 由于包含全局信息, 搜索群体经验与搜索个体经验相比, 搜索群体经验能找到更好的目标, 增强搜索能力. 适当增加搜索群体经验的权重可以获得比预期更好的结果.

IASSA通过引入全局惯性权重增强搜索能力并加快收敛速度. 协同行为和随机行为与ASSA相同, 侦察行为修改如下:

$$X_{i+1} = X_i + r_1 \times (X_s - X_i) + w \times r_2 \times (X_g - X_i), \tag{6}$$

$$w = w_{\max} - \frac{\text{Current.iteration}}{\text{Max.iteration}} \times (w_{\max} - w_{\min}). \tag{7}$$

其中:  $w$  为全局惯性权重,  $w_{\max}$  为惯性权重的最大值,  $w_{\min}$  为惯性权重的最小值,  $\text{Current.iteration}$  为当前迭代次数,  $\text{Max.iteration}$  为最大迭代次数.

### 2.2 动态协同参数

协同参数  $P_c$  为搜索个体呼叫其他个体的初始概率, 是群搜索过程中的协同行为中重要的参数.  $P_c$  值如果过大, 则会使得搜索个体呼叫联系的概率增大, 容易过早使搜索个体聚集在一起引起早熟现象;  $P_c$  值过小, 则容易使协同行为失去作用, 减少搜索个体之间的信息交流, 导致最后算法的搜寻精度和收敛速度降低. 本文引入系数  $C$  并且修改协同规则如下:

$$C = C_{\min} + \frac{\text{Current.iteration}}{\text{Max.iteration}} \times (C_{\max} - C_{\min}), \tag{8}$$

$$r_1 < C \times P_c. \tag{9}$$

其中:  $C$  为系数,  $C_{\max}$  为系数的最大值,  $C_{\min}$  为系数的最小值.

### 2.3 动态步长

人工搜索群算法的步长参数影响算法的优化性能. 步长决定搜索个体下一步搜索位置, 步长过大影响算法的收敛时效性, 步长过小导致搜索个体的实际搜索范围缩小. IASSA算法引入动态步长改善算法优化性能, 算法运行前期步长较大有助于加快收敛速度; 运行后期步长较小有助于提高收敛精度. 动态步长原理如下:

$$\begin{cases} \alpha = \exp \frac{\text{Max.iteration} - \text{Current.iteration}}{\text{Max.iteration}}, \\ \text{Step} = \alpha \times \text{Step}_{\min}. \end{cases} \tag{10}$$

其中  $\text{Step}_{\min}$  为最小步长.

## 3 IASSA算法性能测试

### 3.1 测试函数

为了检验 IASSA 算法的收敛性能和寻优效果, 选取 21 个标准测试函数进行实验仿真, 测试函数参考文献 [16] 及 CEC2014 标准库函数<sup>[17]</sup>. 测试函数分为单峰函数、多峰函数以及复合函数 3 类. 单峰函数可以测试算法的寻优精度, 考察算法的执行能力; 多峰函数由于局部最优点的数目会伴随着维数的增加呈指数增长, 可用来检验算法跳出局部陷阱的能力.

本文在 3 类函数中摘取 15 个表现算法性能差异性较大的测试函数进行仿真数据分析, 表 1 给出了函数的搜索范围, 所有测试函数的理论最优值为 0.

表 1 测试函数

函数类型	序号	函数名称	搜索范围
单峰函数	$f_1$	Bent cigar function	$[-1, 1]$
	$f_2$	Sphere function	$[-100, 100]$
	$f_3$	Step function	$[-100, 100]$
	$f_4$	Noisy quartic function	$[-1.28, 1.28]$
	$f_5$	Discus function	$[-50, 50]$
多峰函数	$f_6$	Rastrigin's function	$[-5.12, 5.12]$
	$f_7$	Griewank's function	$[-600, 600]$
	$f_8$	Ackley's function	$[-32, 32]$
	$f_9$	HGBat function	$[-10, 10]$
	$f_{10}$	Rosenbrock's function	$[-2, 2]$
	$f_{11}$	Expended griewank's plus rosenbrock's function	$[-10, 10]$
	$f_{12}$	Expended scaffer's F6 function	$[-10, 10]$
	$f_{13}$	Modified schwefers function	$[-50, 50]$
复合函数	$f_{14}$	Penalized function 1	$[-50, 50]$
	$f_{15}$	Penalized function 2	$[-50, 50]$

### 3.2 实验结果与分析

本文进行了两组实验,第1组是通过比较IASSA、ASSA和CASSA算法,验证本文提出的改进策略对ASSA算法性能的提高.第2组是将IASSA算法的实验结果与粒子群算法(PSO)、差分算法(DE)和人工蜂群算法(ABC)的计算结果进行比较,从而验证该算法的优越性.

#### 3.2.1 ASSA、IASSA和CASSA的实验结果比较

3种算法参数设置如下:ASSA算法的参数设置参考文献[8],CASSA算法的参数设置参考文献[18],IASSA算法参数设置如表2所示.

表2 设置参数

序号	种群规模	空间维度	$P_c$	最小步长
$f_5$	50	30/200/500	0.002	10
$f_6$	50	20/200/500	0.002	1
$f_7$	70	50/200/500	0.003	150
$f_8$	70	50/200/500	0.003	4

表3 IASSA、ASSA与CASSA优化结果

序号	$D$	ASSA		CASSA		IASSA	
		mean	best	mean	best	mean	best
$f_5$	30	5.74e-1	2.26e-1	4.56e-1	1.75e-1	<b>2.55e-1</b>	<b>1.09e-1</b>
	200	1.99e+0	1.02e+0	1.87e+0	1.37e+0	<b>1.82e+0</b>	<b>9.41e-1</b>
	500	2.36e+0	1.36e+0	<b>2.11e+0</b>	1.60e+0	2.28e+0	<b>1.28e+0</b>
$f_6$	20	1.20e+2	5.11e+1	<b>9.89e+1</b>	<b>2.17e+1</b>	1.17e+2	7.36e+1
	200	3.81e+3	1.58e+3	<b>3.19e+3</b>	<b>1.34e+3</b>	3.31e+3	3.07e+3
	500	4.12e+3	<b>5.35e+3</b>	<b>3.69e+3</b>	5.82e+3	8.69e+3	8.34e+3
$f_7$	50	2.31e+0	<b>1.54e+0</b>	<b>2.13e+0</b>	1.63e+0	2.48e+0	1.85e+0
	200	4.18e+2	2.98e+2	1.05e+3	3.84e+2	<b>2.20e+2</b>	<b>1.41e+2</b>
	500	5.46e+3	2.87e+3	6.43e+3	2.68e+3	<b>1.19e+3</b>	<b>1.01e+3</b>
$f_8$	50	9.38e+0	8.78e+0	<b>8.70e+0</b>	7.27e+0	1.23e+1	<b>5.24e+0</b>
	200	4.33e+1	1.58e+1	3.24e+1	1.68e+1	<b>1.93e+1</b>	<b>1.14e+1</b>
	500	3.88e+1	1.83e+1	<b>1.94e+1</b>	<b>1.67e+1</b>	1.99e+1	1.99e+1

#### 3.2.2 IASSA与其他算法实验结果比较

本组实验选取PSO、DE和ABC三种算法分别与IASSA算法仿真结果相比较,PSO算法的参数按文献[10]设置,DE算法参数按文献[19]设置,ABC算法参数按文献[2]设置.对于15个测试函数,IASSA算法参数设置如下:空间维度( $D$ )为50,最大迭代次数为100,种群规模( $N$ )为50,最小步长设置为1,独立运行20次.3种算法优化结果如表4所示.

由表4中可知,在空间维度为50时,IASSA算法对 $f_3 \sim f_8$ 函数不论在平均值和最优值上都明显优于PSO和DE算法.从数值上看,IASSA算法的优化结果均优于其对比算法一两个数量级,其中 $f_2$ 、 $f_3$ 和 $f_{15}$ 函数的优化结果比较显著.全局惯性权重策略可以提升每一代的全局最优位置的历史经验,所以采用

在表2中:系数 $C_{max}$ 为100, $C_{min}$ 为1; $w_{max}$ 参数设置为1.5, $w_{min}$ 为0.2.选取 $f_5 \sim f_8$ 四个函数对每种算法分别独立运行20次,最大迭代次数为100.

实验结果如表3所示,选取3种算法运行的平均值和最优值作为评估算法性能的标准.由表3可知,引入混沌理论和动态参数的改进策略都提高了ASSA算法的优化性能,不论是处理单峰问题还是多峰问题,改进策略都是有效的.对于 $f_5$ 函数的优化结果,不论在低维还是高维,IASSA算法都优于CASSA算法和ASSA算法. $f_6$ 函数是一种典型的非线性多模态的多峰函数,动态参数的策略比混沌策略结果要差.这是由于两种策略改进的方向不同,混沌理论使初始种群分布更加均匀,对多局部最优函数的寻优能力较强. $f_7$ 函数在50维时,IASSA算法的优化结果劣于其他两种算法,但3种算法的平均值和最优值并没有显著的差异.对于 $f_8$ 函数,IASSA算法的优化结果在50维和200维时优于CASSA算法和ASSA算法.

表4 IASSA、DE与PSO在 $D = 50$ 时的优化结果

序号	IASSA		DE		PSO	
	mean	best	mean	best	mean	best
$f_2$	<b>7.55e+1</b>	<b>4.81e+1</b>	1.28e+4	8.21e+3	1.17e+4	6.10e+3
$f_3$	<b>1.34e+2</b>	<b>1.01e+2</b>	1.19e+4	7.68e+3	1.03e+4	4.72e+3
$f_4$	<b>5.92e-1</b>	<b>3.33e-1</b>	4.56e+1	2.48e+1	1.16e+3	4.99e+2
$f_5$	<b>1.41e+3</b>	<b>7.12e+1</b>	4.36e+3	3.18e+3	1.47e+4	9.30e+3
$f_6$	<b>3.83e+2</b>	<b>1.97e+2</b>	8.31e+2	7.89e+2	5.05e+2	4.55e+2
$f_7$	<b>2.48e+0</b>	<b>1.85e+0</b>	1.14e+2	6.39e+1	7.25e+2	5.78e+2
$f_8$	<b>2.48e+0</b>	<b>1.85e+0</b>	1.40e+1	1.21e+1	1.37e+1	1.23e+1
$f_{14}$	<b>3.02e+1</b>	<b>4.53e+0</b>	1.05e+6	3.92e+3	1.26e+2	8.27e+1
$f_{15}$	<b>1.02e+2</b>	<b>3.84e+1</b>	9.44e+6	2.88e+6	6.07e+2	4.18e+2

该算法处理非线性多模态问题能很好地避免局部收敛,且收敛速度和收敛精度都有很大的提升.

测试函数的收敛曲线如图1~图5所示.由图1~图5可知,在 $f_2$ 、 $f_3$ 、 $f_5$ 、 $f_7$ 、 $f_8$ 函数的收敛曲线

中, IASSA算法的收敛速度和收敛精度显著优于DE、PSO算法.

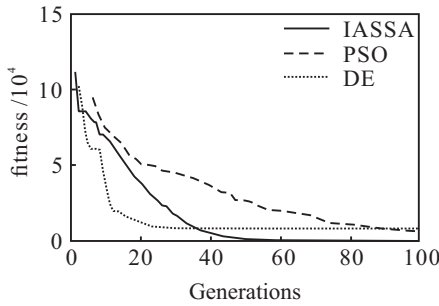


图1 函数  $f_2$  的收敛曲线

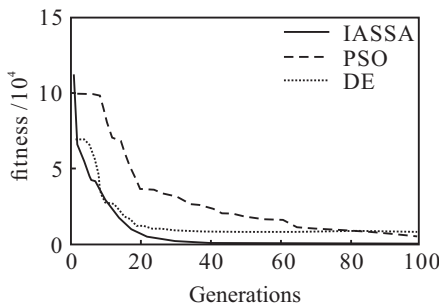


图2 函数  $f_3$  的收敛曲线

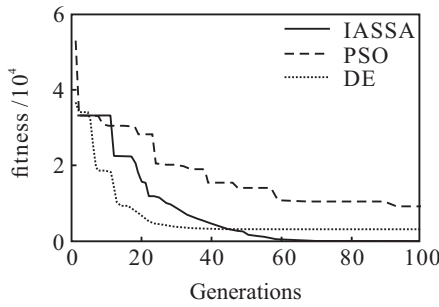


图3 函数  $f_5$  的收敛曲线

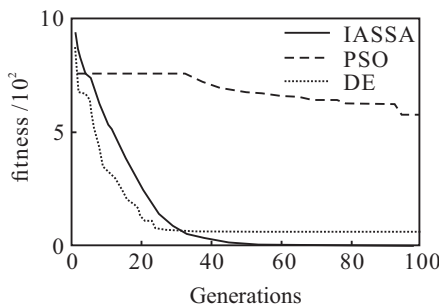


图4 函数  $f_7$  的收敛曲线

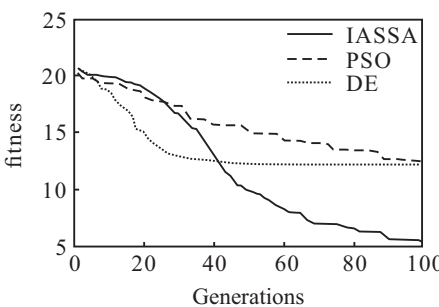


图5 函数  $f_8$  的收敛曲线

对表5中的实验结果进行分析, 当空间维度提高到200时, 所有算法的求解精度都受到了影响, 明显劣于低维的收敛结果. 本文提出的IASSA算法在全部15个测试函数中7个函数优于其对比算法. 在  $f_1$ 、 $f_3$ 、 $f_4$  函数中, IASSA算法对单峰函数的寻优能力优于DE算法. 但是从  $f_5 \sim f_{11}$  函数的优化结果可以看出, IASSA算法在处理多峰函数上性能劣于DE算法. 对复合函数的实验结果分析可知, IASSA算法的优化性能远超DE、ABC算法.

表5 IASSA、ABC与DE在  $D = 200$  时的优化结果

序号	IASSA		ABC		DE	
	mean	best	mean	best	mean	best
$f_1$	<b>3.07e+6</b>	<b>2.26e+6</b>	3.96e+7	3.78e+7	4.87e+6	3.52e+6
$f_2$	3.08e+4	2.30e+4	<b>2.08e+4</b>	<b>7.17e+3</b>	4.70e+5	3.50e+5
$f_3$	<b>1.16e+4</b>	<b>7.96e+3</b>	3.81e+5	3.68e+5	2.00e+4	1.39e+4
$f_4$	<b>7.81e+1</b>	<b>1.51e+1</b>	6.00e+3	5.65e+3	1.61e+3	1.29e+3
$f_5$	6.50e+4	3.67e+4	7.35e+4	6.97e+4	<b>1.86e+4</b>	<b>1.01e+4</b>
$f_6$	3.31e+3	3.07e+3	3.30e+3	2.42e+3	<b>7.23e+2</b>	<b>2.60e+2</b>
$f_7$	2.20e+2	1.40e+2	3.71e+3	3.18e+3	<b>1.46e+2</b>	<b>9.98e+1</b>
$f_8$	1.93e+1	1.14e+1	2.63e+1	2.04e+1	<b>1.08e+1</b>	<b>8.71e+0</b>
$f_9$	1.51e+2	1.02e+2	4.97e+3	3.74e+3	<b>1.00e+2</b>	<b>8.71e+1</b>
$f_{10}$	4.49e+3	2.57e+3	5.95e+4	4.02e+4	<b>1.95e+3</b>	<b>1.43e+3</b>
$f_{11}$	9.58e+8	3.22e+8	1.84e+9	1.76e+9	<b>3.98e+4</b>	<b>1.73e+4</b>
$f_{12}$	<b>1.18e+1</b>	<b>7.32e+0</b>	1.73e+2	1.43e+2	2.58e+2	1.20e+2
$f_{13}$	<b>2.83e+2</b>	<b>2.72e+2</b>	1.58e+3	1.58e+3	1.20e+4	1.02e+4
$f_{14}$	<b>1.07e+2</b>	<b>7.72e+1</b>	3.37e+9	3.29e+9	7.43e+4	1.62e+4
$f_{15}$	<b>1.91e+3</b>	<b>1.52e+3</b>	6.29e+9	5.30e+9	5.26e+6	1.19e+6

### 4 结论

固定的协同参数、权重系数和步长易使算法陷入局部收敛, 本文针对人工搜索群算法协同参数、权重系数和步长3个控制参数对算法的影响, 提出了一种动态参数人工搜索群算法. 引入全局权重系数改善侦查行为中搜索群的历史经验位置, 加强每代全局最优搜索个体对下一代搜索群体的引导能力; 利用动态协同参数提高士兵协同行为的搜索概率, 以加强种群之间局部信息交流, 避免搜索陷入局部收敛; 采用动态步长策略提高搜索个体在前期的搜索效率和后期的搜索精度, 以此提高算法的收敛速度和收敛精度. 最后通过4个标准测试函数的仿真实验, 验证了动态参数策略的有效性, 其中第2组实验利用15个标准测试函数检测IASSA算法的优化性能, 并与DE、PSO以及ABC算法的仿真结果进行了比较. 实验结果表明, IASSA算法优化性能优于其他算法. 相较于其他算法, IASSA算法依赖参数的设定, 如何在减少算法参数的同时不影响算法的优化性能将是下一步研究的重点. 此外, 可以将IASSA算法应用于智能电网的经济优化调度、智能电器设计等方面.

## 参考文献(References)

- [1] 李晓磊. 一种新型的智能优化算法——人工鱼群算法[D]. 杭州: 浙江大学信息科学与工程学院, 2003.  
(Li X L. A new intelligent optimization method—Artificial fish school algorithm[D]. Hangzhou: College of Information Science & Electronic Engineering, Zhejiang University, 2003.)
- [2] Karaboga D, Basturk B. On the performance of artificial bee colony (ABC) algorithm[J]. *Applied Soft Computing*, 2008, 8(1): 687-697.
- [3] Sait A U, Gulay T, Esra Y. Artificial algae algorithm (AAA) for nonlinear global optimization[J]. *Applied Soft Computing*, 2015, 31: 153-171.
- [4] Omidvar M N, Li X, Mei Y, et al. Cooperative co-evolution with differential grouping for larger scale optimization[J]. *IEEE Trans on Evolutionary Computation*, 2014, 18(3): 378-393.
- [5] Yeh W, Hsieh T. Artificial bee colony algorithm-neural for S-system models of biochemical networks approximation[J]. *Neural Computing and Applications*, 2012, 21(2): 365-375.
- [6] Roy P K. Solution of unit commitment problem using gravitational search algorithm[J]. *Int J of Electrical Power & Energy System*, 2013, 53(1): 85-94.
- [7] Deng W, Zhao H M, Yang X H, et al. Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment[J]. *Applied Soft Computing*, 2017, 59: 288-302.
- [8] Chen T G. A simulative bionic intelligent optimization algorithm: Artificial searching swarm algorithm and its performance analysis[C]. *The 2nd Int Joint Conf on Computational Sciences and Optimization*. Sanya, 2009: 864-866.
- [9] 欧阳海滨, 高立群, 邹德旋, 等. 和声搜索算法探索能力研究及其修正[J]. *控制理论与应用*, 2014, 31(1): 57-65.  
(Ouyang H B, Gao L Q, Zou D X, et al. Exploration ability study of harmony search algorithm and its modification[J]. *Control Theory & Applications*, 2014, 31(1): 57-65.)
- [10] Hao C, Wang Y Q, Tuo J Y. A novel particle swarm optimization algorithm with intelligent weighting mechanism[C]. *The 2nd Int Conf on Information Science and Control Engineering*. New York: IEEE, 2015: 45-49.
- [11] Zhang L, Tang Y, Hua X, et al. A new particle swarm optimization algorithm with adaptive inertia weight based on bayesian techniques[J]. *Applied Soft Computing*, 2015, 28: 138-149.
- [12] Mahdavi M, Fesanghary M, Damangir E. An improved harmony search algorithm for solving optimization problem[J]. *Applied Mathematics and Computing*, 2007, 188(2): 1567-1579.
- [13] Chen J, Pan Q, Li J. Harmony search algorithm with dynamic control parameters[J]. *Applied Mathematics and Computing*, 2012, 219(2): 592-604.
- [14] Kattan A, Abdullah R. A dynamic self-adaptive harmony search algorithm for continuous optimization problems[J]. *Applied Mathematics and Computing*, 2013, 219(16): 8542-8567.
- [15] Peng C D, Liu H L, Gu F Q. An evolutionary algorithm with directed weights for constrained multi-objective optimization[J]. *Applied Soft Computing*, 2017, 60: 613-622.
- [16] Zhang J Sanderson. JADE: Adaptive differential evolution with optional external archive[J]. *IEEE Trans on Evolutionary Computation*, 2009, 13(5): 945-958.
- [17] Liang J J, Qu B Y, Suganthan P N. Problems definitions and evaluation criteria for the CEC2014 special session competition on single objective real-parameter numerical optimization[R]. Zhengzhou: Zhengzhou University, 2013.
- [18] 刘亚贤. 基于改进人工搜索群算法的电磁装置化设计[D]. 天津: 河北工业大学电气工程学院, 2013.  
(Liu Y X. Optimization design of electromagnetic device based on improved artificial searching swarm algorithm[D]. Tianjin: School of Electrical Engineering, Hebei University of Technology, 2013.)
- [19] 张贵军, 王柳静, 周晓根, 等. 基于共轭增强策略的差分进化算法[J]. *控制与决策*, 2017, 32(7): 1313-1318.  
(Zhang G J, Wang L J, Zhou X G, et al. Differential evolution algorithm with conjugate enhancement strategy[J]. *Control and Decision*, 2017, 32(7): 1313-1318.)

## 作者简介

陈堂功(1963—), 男, 副教授, 博士, 从事智能优化算法及其应用等研究, E-mail: tgchen@hebut.edu.cn;

刘超(1993—), 男, 硕士生, 从事智能算法及全局优化的研究, E-mail: liuchao\_hebut@126.com;

王梦莹(1996—), 女, 硕士生, 从事智能算法及其应用的研究, E-mail: wmyingwyx@163.com;

周小婷(1993—), 女, 硕士生, 从事智能算法及全局优化算法的研究, E-mail: 2710740716@qq.com;

黄涛(1991—), 男, 硕士, 从事智能算法及电磁计算的研究, E-mail: 2360224096@qq.com.

(责任编辑: 闫妍)