

# 基于复数编码的多策略人工蜂群算法

单 娴<sup>1</sup>, 杜学东<sup>2</sup>

(1. 中国石油大学(华东)理学院, 山东 青岛 266580;

2. 山东科技大学信息科学与工程学院, 山东 青岛 266510)

**摘要:** 人工蜂群算法是解决不同类型优化问题的优秀算法之一, 但该算法在处理复杂优化问题时仍存在收敛速度慢、易早熟和局部搜索能力弱等缺陷. 为克服这些问题, 从基本人工蜂群算法出发, 提出基于复数编码的多策略人工蜂群算法. 该算法针对人工蜂群算法的特点, 设计搜索策略知识库, 由种群个体在搜索过程中自适应选择最佳搜索方式, 并引入复数编码方法构造双倍体种群个体, 改善种群个体的多样性, 进一步提高解的质量. 应用15个测试函数对算法性能进行仿真实验, 并将实验结果与其他算法进行比较, 结果表明本文提出的算法在收敛速度和计算精度上明显优于对比方法, 能够有效地解决全局优化问题.

**关键词:** 人工蜂群算法; 复数编码; 多种搜索策略; 函数优化

中图分类号: TP273 文献标识码: A 文章编号: 1000-5781(2018)05-0597-09

doi: 10.13383/j.cnki.jse.2018.05.003

## Multi-strategy artificial bee colony algorithm based on complex coding

Shan Xian<sup>1</sup>, Du Xuedong<sup>2</sup>

(1. School of Science, China University of Petroleum (East China), Qingdao 266580, China;

2. College of Information Science and Engineering, Shandong University of Science and Technology, Qingdao 266510, China)

**Abstract:** Artificial bee colony algorithm is one of the good optimization algorithms for various optimization problems. However, this algorithm some disadvantages in solving complicated problems, such as low convergence rate, premature problem and bad local search capability. In order to overcome these disadvantages, based on the characteristics of basic artificial bee colony algorithm, a self-adaptive selection method is designed to select the best way to search automatically. Besides, a complex coding method that effectively improves the diversity of the population and the quality of the solution is employed to construct diploid individuals. The algorithm is tested on 15 test functions. Results compared with other methods show that the proposed algorithm is obviously superior to other optimization algorithms in terms of convergence speed and accuracy. It is competitive and effective in solving global optimization problems.

**Key words:** artificial bee colony algorithm; complex coding; multi-strategy; function optimization

## 1 引言

随着社会经济和科学技术的不断发展, 在工程技术、国防科技、经济管理和制造业等多个领域中涌现出大量的复杂优化问题. 由于这些实际问题往往具有数据规模大、信息不确定等特点, 在有限的时间内运用传

收稿日期:2016-11-20; 修订日期:2017-02-09.

基金项目: 国家自然科学基金资助项目(61402534); 中央高校基本科研业务费专项资金资助项目(16CX02010A).

统的精确优化方法已经难以解决. 因此, 设计高效、实用的优化算法成为科研工作人员研究的热点. 通过模拟自然界中生物行为特征而衍生出的启发式优化算法应运而生.

目前常用的启发式优化智能算法包括以遗传算法<sup>[1]</sup>、差分进化算法<sup>[2]</sup>和分布估计算法<sup>[3]</sup>为代表的进化算法, 和模拟动物行为特征的粒子群算法<sup>[4]</sup>、蚁群算法<sup>[5]</sup>、鱼群算法<sup>[6]</sup>、猴群算法<sup>[7]</sup>和人工蜂群算法<sup>[8]</sup>等群体智能算法. 近年来, 涌现出一批新的模拟人类行为特征的群体智能算法, 如 SBA 算法<sup>[9]</sup>、TLBA 算法<sup>[10]</sup>和 JOA 算法<sup>[11]</sup>等. 由于这些智能算法不需要梯度信息, 而且不要求优化函数必须满足凸条件, 因而在复杂优化问题的求解中显示出强大的优势. 如何通过对各种智能算法进行改进改善其解决优化问题的能力, 成为研究者们广泛关注的问题.

人工蜂群算法(artificial bee colony algorithm, ABC)是 2005 年由 Karaboga<sup>[8,12]</sup>基于蜜蜂采蜜过程中的分工机制和群体智能行为提出的一种启发式群体智能算法. 该算法是一种并行直接搜索方法, 计算过程中不需要任何梯度信息, 算法结构简单, 易于在计算机上实现, 控制参数较少, 具有较好的收敛性和稳定性. 因此, 自提出以来便被广泛应用至参数优化<sup>[13]</sup>、聚类分析<sup>[14]</sup>与生产调度<sup>[15]</sup>等多个领域, 解决了国内外现代工程领域的多种复杂优化问题<sup>[16]</sup>.

国内外学者针对人工蜂群算法展开多个角度的研究, 研究的内容主要分为两个方面.

#### 1) 通过改进搜索策略提高算法搜索效率.

人工蜂群算法主要由雇佣蜂搜索、观察蜂搜索和侦察蜂搜索三个阶段完成. 算法性能的优劣依赖于各个阶段的搜索方程. Zhu 等<sup>[17]</sup>将全局最优解的信息融入搜索方程, 提出受 gbest 引导的 ABC 算法(GABC). Banharnsakun 等<sup>[18]</sup>考虑将目前最优解(best-so-far)设置为引导项, 提出改进的 ABC 算法. Gao 等<sup>[19]</sup>受 DE/best/1 和 DE/best/2 两种变异策略的启发, 运用最优解的信息, 提出 ABC/best/1 和 ABC/best/2 两种改进的搜索策略, 有效地提高了算法的开发能力, 但与 ABC/rand/1 相比, 则显得探索能力不足. 为此, Gao 等<sup>[20]</sup>考虑将 ABC/rand/1 和 ABC/best/1 结合, 并引入参数  $M$ , 将两种搜索策略优势互补, 更好地平衡了探索能力和开发能力. Li 等<sup>[21]</sup>受 PSO 算法的启示, 在搜索方程中加入由适应度值确定的惯性权重和加速系数. Gao 等<sup>[22]</sup>对搜索公式进行改进, 将正交学习策略引入搜索过程, 提出 CABC 算法. Gao 等<sup>[23]</sup>将种群划分为多个子种群, 运用各个子种群内部及子种群之间的信息交换进行搜索, 提出基于信息学习的改进 ABC 算法.

从标准 ABC 算法和上述改进算法中可以看出, 不同的引导向量和搜索方程能够对搜索方向产生不同的影响, 进而影响算法的收敛速度和计算精度. 各种搜索策略在不同类型优化问题中的表现各异, 即使对应同一类型优化问题, 对于不同规模和不同求解阶段求解效率也略有不同.

#### 2) 将人工蜂群算法与其他优化算法相结合改善算法性能.

Kang 等<sup>[24-26]</sup>将 Nelder-Mead 单形搜索机制、Rosenbrock 方法和 Hooke-Jeeves 方法引入 ABC 算法, 提出混合的单形 ABC 算法、Rosenbrock-ABC 算法和 HABC 算法. Gao 等<sup>[27]</sup>运用传统的 Powell 方法作为局部搜索算子对 ABC 算法进行改进, 提高了 ABC 算法的整体性能. Chen 等<sup>[28]</sup>将模拟退火算法嵌入搜索过程, 改善了算法的搜索能力. Mustafa 等<sup>[29]</sup>提出基于 PSO 和 ABC 的混合算法. 上述算法的改进过程中, 运用其他优化算法的特点, 将局部搜索与全局搜索有机结合, 实现开发能力与勘探能力的均衡.

尽管上述改进方法在一定程度上改善了 ABC 算法的性能, 但这些算法在提高精度, 改进效率和避免过早收敛等方面仍具有较大的改进空间. 目前对蜂群算法的研究中, 普遍考虑采用一种搜索策略展开寻优, 而不同的搜索策略在不同问题的各个求解阶段寻优效果存在明显差异, 现有研究尚不能为不同优化问题如何选择最佳策略提供有价值的参考.

为此, 本文考虑种群个体编码方式和搜索策略对算法效率的影响, 提出基于复数编码的多策略人工蜂群算法. 利用复数编码方法维持解集的多样性, 增强种群个体信息容量. 并建立搜索策略知识库, 在搜索过程中由种群个体依据自身获取信息自适应选择搜索策略, 进而提高算法的搜索能力. 最后, 采用测试函数对算法性能进行比较和分析.

## 2 基本人工蜂群算法

人工蜂群由雇佣蜂、观察蜂和侦察蜂三部分构成. 其中, 雇佣蜂负责搜索食物源, 并将获取的食物源信息以摇摆舞的形式传递给观察蜂; 观察蜂以一定的概率选择食物源, 或者在食物源附近继续搜索新的食物源; 若某个雇佣蜂获得的食物源经过若干次搜索后仍然没有得到改善, 则该雇佣蜂会放弃该食物源, 并转化为侦察蜂重新进行搜索.

食物源的数目与雇佣蜂的数目相同, 即等于种群规模  $SN$ . 一个食物源的位置代表优化问题的一个候选解, 称为种群个体, 用  $\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$  表示, 其中  $D$  为优化问题的维数,  $i = 1, 2, \dots, SN$ . 食物源的花蜜量对应解的适应度值.

根据蜂群的构成, 将整个搜索过程分为四个阶段:

### 1) 种群个体初始化阶段

初始食物源在各分量边界范围内由式(1)随机产生

$$x_{ij} = x_{ij}^{\min} + \phi (x_{ij}^{\max} - x_{ij}^{\min}), \quad (1)$$

其中  $j = 1, 2, \dots, D$ ,  $x_{ij}$  为第  $i$  个食物源的第  $j$  个分量,  $x_{ij}^{\min}$  和  $x_{ij}^{\max}$  表示第  $j$  个分量的下界和上界,  $\phi$  为  $[0, 1]$  之间的随机数.

### 2) 雇佣蜂搜索阶段

雇佣蜂在食物源位置附近随机产生新的食物源  $\mathbf{V}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ , 搜索方程为

$$v_{ij} = x_{ij} + \varphi_{ij} (x_{ij} - x_{kj}), \quad (2)$$

其中  $j$  和  $k$  是随机选择的下标, 满足  $j \in \{1, 2, \dots, D\}$ ,  $k \in \{1, 2, \dots, SN\}$ , 且  $k \neq i$ .  $\varphi_{ij}$  为  $[-1, 1]$  之间的随机数.

### 3) 观察蜂选择阶段

观察蜂从雇佣蜂处获取食物源信息, 运用适应度函数  $f$  对第  $i$  个食物源进行评估, 并根据轮盘赌和贪婪策略选择食物源. 当选择概率  $p_i > s$  时, 观察蜂选择该食物源; 否则, 根据式(2)重新搜索新的食物源, 并对新食物源进行评估. 其中  $s$  为  $[0, 1]$  之间的随机数, 选择概率

$$p_i = \text{fitness}_i \left( \sum_{i=1}^{SN} \text{fitness}_i \right)^{-1}, \quad (3)$$

其中  $\text{fitness}_i$  为第  $i$  个食物源的适应度值.

### 4) 侦察蜂搜索阶段

若某个食物源经过  $\text{limit}$  次循环后仍然未被更新, 则该食物源将被放弃, 发现该食物源的雇佣蜂转化为侦察蜂, 并且由式(1)随机产生一个新的食物源.

## 3 改进的人工蜂群算法

### 3.1 基于复数编码的人工蜂群算法

传统智能优化算法中, 种群个体的编码采用二进制编码或者实数编码, 这使得每个种群个体包含的信息在一定程度上受到限制. 受神经网络权值表达的复数编码方法启示<sup>[30]</sup>, 考虑用双倍体表示每个食物源个体, 从而将个体信息量翻倍, 增加了种群的多样性, 更加充分地利用到搜索空间的信息.

对于  $D$  维的优化问题, 食物源个体  $\mathbf{Y}$  的第  $j$  个分量对应一个复数,

$$Y_j = R_j + iI_j, \quad (4)$$

其中  $i$  是虚数单位,  $j = 1, 2, \dots, D$ ,  $R_j$  表示第  $j$  个分量的实部,  $I_j$  表示第  $j$  个分量的虚部.

### 3.1.1 种群个体的初始化

设优化问题第  $j$  个分量的取值范围为  $[A_j, B_j]$ , 随机产生  $D$  个模和  $D$  个幅角

$$\rho_j \in [0, (B_j - A_j)/2], \quad (5)$$

$$\theta_j \in [-2\pi, 2\pi], \quad (6)$$

第  $i$  个个体的第  $j$  个分量对应的双倍体为  $(R_j, I_j)$ . 由于

$$R_j + iI_j = \rho_j (\cos \theta_j + i \sin \theta_j), \quad (7)$$

可得

$$R_j = \rho_j \cos \theta_j, \quad (8)$$

$$I_j = \rho_j \sin \theta_j. \quad (9)$$

由此产生  $SN \times D$  个双倍体, 即  $SN$  个食物源个体, 每个个体包含  $D$  个双倍体.

### 3.1.2 种群个体的更新

与传统编码方式不同的是, 复数编码个体的更新需要对实部和虚部并行进行, 从而在增加种群多样性的同时, 还可以提高算法的运行效率.

#### 1) 复数编码的实部更新

$$R_{ij} \leftarrow R_{ij} + \varphi_{ij} (R_{ij} - R_{kj}), \quad (10)$$

其中  $j$  和  $k$  是随机选择的下标, 满足  $j \in \{1, 2, \dots, D\}$ ,  $k \in \{1, 2, \dots, SN\}$ , 且  $k \neq i$ .  $R_{ij}$  为第  $i$  个个体的第  $j$  个分量的实部,  $\varphi_{ij}$  为  $[-1, 1]$  之间的随机数.

#### 2) 复数编码的虚部更新

$$I_{ij} \leftarrow I_{ij} + \varphi_{ij} (I_{ij} - I_{kj}), \quad (11)$$

其中  $j$  和  $k$  是随机选择的下标, 满足  $j \in \{1, 2, \dots, D\}$ ,  $k \in \{1, 2, \dots, SN\}$ , 且  $k \neq i$ .  $I_{ij}$  为第  $i$  个个体的第  $j$  个分量的虚部,  $\varphi_{ij}$  为  $[-1, 1]$  之间的随机数.

### 3.1.3 适应度函数的计算方法

由于每个个体的各个分量都是由实部和虚部构成, 需要先将各个分量的编码由二维空间确定的复数转化到一维空间确定的实数, 然后再计算相应的适应度. 具体的操作过程可以描述为

#### 1) 由复数的模确定实数值

$$\rho_{ij} \leftarrow \sqrt{R_{ij}^2 + I_{ij}^2}. \quad (12)$$

#### 2) 由幅角确定实数变量

$$x_{ij} \leftarrow \rho_{ij} \operatorname{sgn}(\sin(I_{ij}/\rho_{ij})) + B_j + A_j/2, \quad (13)$$

其中  $x_{ij}$  表示转换后的实数自变量. 然后得到新个体的适应度值, 并对其进行评价, 若优于当前个体的适应度值, 则进行替换, 否则进行下一次迭代.

## 3.2 多策略人工蜂群算法

已有的人工蜂群算法中, 对雇佣蜂和观察蜂均采用某一种搜索策略获取食物源. 仅采用一种搜索策略,

每次只产生一个新的个体,无法兼顾搜索的多样性.而且,某一种搜索策略可能适用于一类或几类优化问题,而对其他的优化问题则无法得到理想的求解结果.不同的搜索策略对于同一优化问题,获得的结果也可能存在较大的差异.为解决单一搜索策略可能造成的不足,考虑采用多种搜索策略自适应选择的方式对食物源进行搜索.

论文设计一种自适应搜索策略调整方法.首先建立产生食物源个体的搜索策略知识库,然后对应每一次搜索,从知识库中选取三种搜索策略产生三个新的个体,并从三个候选个体中选取适应度最优者进入下一代种群.

**策略 1** 个体  $\mathbf{X}_i$  从自身邻域范围内选取个体  $\mathbf{X}_k$ , 根据自身信息和个体  $\mathbf{X}_k$  的信息进行更新, 搜索方程为

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}), \quad (14)$$

其中  $j$  和  $k$  是随机选择的下标,  $j \in \{1, 2, \dots, D\}$ ,  $k \in \{1, 2, \dots, SN\}$ , 且  $k \neq i$ .

**策略 2** 个体  $\mathbf{X}_i$  从自身邻域范围内选取两个个体  $\mathbf{X}_{k_1}$  和  $\mathbf{X}_{k_2}$ , 根据自身信息和个体  $\mathbf{X}_{k_1}$ 、 $\mathbf{X}_{k_2}$  的信息进行更新, 搜索方程为

$$v_{ij} = x_{ij} + \varphi_{1ij}(x_{ij} - x_{k_1j}) + \varphi_{2ij}(x_{ij} - x_{k_2j}), \quad (15)$$

其中  $j, k_1$  和  $k_2$  是随机选择的下标,  $j \in \{1, 2, \dots, D\}$ ,  $k_1 \in \{1, 2, \dots, SN\}$ ,  $k_2 \in \{1, 2, \dots, SN\}$ , 且  $k_1 \neq i$ ,  $k_2 \neq i, k_1 \neq k_2$ .

**策略 3** 个体  $\mathbf{X}_i$  从自身邻域范围内选取两个个体  $\mathbf{X}_{k_1}$  和  $\mathbf{X}_{k_2}$ , 根据当前种群中最优个体  $\mathbf{X}_{\text{best}}$  的信息和个体  $\mathbf{X}_{k_1}$ 、 $\mathbf{X}_{k_2}$  的信息进行更新, 搜索方程为

$$v_{ij} = x_{\text{best}j} + \varphi_{ij}(x_{k_1j} - x_{k_2j}), \quad (16)$$

其中  $j, k_1$  和  $k_2$  是随机选择的下标,  $j \in \{1, 2, \dots, D\}$ ,  $k_1 \in \{1, 2, \dots, SN\}$ ,  $k_2 \in \{1, 2, \dots, SN\}$ ,  $k_1 \neq i$ ,  $k_2 \neq i, k_1 \neq k_2$ .

其中策略 1 适用于搜索范围较小, 食物源分布较为密集的情形; 策略 2 在策略 1 的基础上, 局部搜索能力有所改善; 策略 3 适用于搜索范围较大, 食物源分布较为分散的情形.

### 3.3 改进人工蜂群算法流程

基于复数编码方式和多种搜索策略, 提出改进的人工蜂群算法(简称 CCABC 算法). 算法流程如下:

**步骤 1** 参数设定. 设置种群个体的数量 SN、个体更新限制次数 limit 和最大迭代次数 MaxCycle;

**步骤 2** 初始化种群. 采用复数编码方法产生初始种群, 并计算种群个体的适应度值, 记录当前最优个体  $\mathbf{X}_{\text{best}}$  的相关信息;

**步骤 3** 雇佣蜂运用式(10)、式(11)和式(14)~式(16)对个体  $\mathbf{X}_i$  进行更新, 产生新个体的实部和虚部, 并根据式(12)和式(13)转换为相应实数, 得到备选个体  $\mathbf{V}_1$ 、 $\mathbf{V}_2$  和  $\mathbf{V}_3$ . 计算三个个体的适应度值, 选择最优个体  $\mathbf{V}_i$ . 若  $f(\mathbf{V}_i) < f(\mathbf{X}_i)$ , 则  $\mathbf{X}_i \leftarrow \mathbf{V}_i$ ,  $\text{trial}_i \leftarrow 0$ ; 否则,  $\text{trial}_i \leftarrow \text{trial}_i + 1$ ;

**步骤 4** 按照式(3)计算个体的选择概率. 观察蜂依据轮盘赌方法选择个体  $\mathbf{X}_i$ , 并利用式(10)~式(16)进行更新, 得到备选个体  $\mathbf{V}_1$ 、 $\mathbf{V}_2$  和  $\mathbf{V}_3$ . 计算三个个体的适应度值, 选择最优个体  $\mathbf{V}_i$ . 将  $\mathbf{V}_i$  与  $\mathbf{X}_i$  进行比较, 保留较优个体;

**步骤 5** 若  $\text{trial}_i > \text{limit}$ , 雇佣蜂转化为侦察蜂, 并依据复数编码方法产生新个体 Sol 取代原个体  $\mathbf{X}_i$ .

**步骤 6** 计算当前种群的适应度值, 并记录最优个体  $\mathbf{X}_{\text{best}}$  的相关信息. 迭代次数  $\text{iter} \leftarrow \text{iter} + 1$ ;

**步骤 7** 判断是否达到终止条件或最大迭代次数, 若满足, 则返回最优个体和最优解, 停止迭代; 否则, 继续执行步骤 3.

## 4 实验仿真与分析

### 4.1 算法测试环境

为验证算法的有效性,对 15 个基准测试函数<sup>[19,20]</sup>进行仿真实验.其中  $f_1 \sim f_6$  是单峰函数,  $f_7$  是不连续的阶梯函数,  $f_8 \sim f_{12}$  是多峰多极值函数,  $f_{13} \sim f_{15}$  是转移的多峰多极值函数.表 1 给出了 15 个测试函数的名称、维数、搜索空间范围和最优值.

表 1 测试函数说明  
Table 1 Benchmark functions used in experiments

函数	名称	维数	搜索空间	最优值
$f_1$	Sphere	30	$[-100, 100]^D$	0
$f_2$	Elliptic	30	$[-100, 100]^D$	0
$f_3$	Sumsquare	30	$[-10, 10]^D$	0
$f_4$	Sumpower	30	$[-10, 10]^D$	0
$f_5$	Schwefel222	30	$[-10, 10]^D$	0
$f_6$	Rastrigin	30	$[-5.12, 5.12]^D$	0
$f_7$	Step	30	$[-100, 100]^D$	0
$f_8$	Griewank	30	$[-600, 600]^D$	0
$f_9$	Ackley	30	$[-32, 32]^D$	0
$f_{10}$	Alpine	30	$[-10, 10]^D$	0
$f_{11}$	Levy	30	$[-10, 10]^D$	0
$f_{12}$	Weierstrass	30	$[-0.5, 0.5]^D$	0
$f_{13}$	Shifted Sphere	30	$[-100, 100]^D$	0
$f_{14}$	Shifted Rastrigin	30	$[-5.12, 5.12]^D$	0
$f_{15}$	Shifted Griewank	30	$[-600, 600]^D$	0

对所有的测试函数,维数  $D$  和种群规模  $SN$  分别设置为 30 和 50,最大迭代次数  $MaxCycle = 1\ 000$ , 阈值  $limit = 1\ 000$ .为了说明算法的有效性和优越性,运用 MATLAB 编程并进行仿真.将算法的计算结果与单一搜索策略 ABC/rand/1 和 ABC/best/1 进行比较,每个测试函数在不同算法上均独立运行 30 次,统计其最优值运算结果的均值 Mean 和标准方差 SD.均值能够反映算法所能达到的精度,标准方差则可以反映算法的稳定性.结果如表 2 所示.

表 2 CCABC 算法, ABC/rand/1 算法和 ABC/best/1 算法性能比较  
Table 2 Result comparisons of CCABC, ABC/rand/1 and ABC/best/1

函数	ABC/rand/1		ABC/best/1		CCABC	
	Mean	SD	Mean	SD	Mean	SD
$f_1$	5.21E-10	2.46E-10	1.57E-27	1.14E-27	3.80E-52	1.65E-52
$f_2$	4.10E-06	3.85E-06	3.66E-28	5.96E-28	3.16E-48	1.59E-48
$f_3$	2.22E-11	1.14E-11	1.67E-24	1.58E-24	2.26E-53	1.50E-54
$f_4$	1.45E-16	1.55E-16	2.70E-69	5.38E-69	3.28E-88	3.88E-91
$f_5$	1.83E-06	4.80E-07	3.45E-15	8.79E-16	1.49E-27	8.91E-28
$f_6$	4.81E-03	2.57E-02	0	0	0	0
$f_7$	0	0	0	0	0	0
$f_8$	1.61E-08	3.99E-08	4.23E-11	2.16E-11	0	0
$f_9$	4.83E-06	2.12E-06	1.26E-13	3.48E-14	7.11E-15	0
$f_{10}$	7.66E-05	2.76E-05	1.32E-14	5.04E-15	2.02E-26	4.76E-27
$f_{11}$	7.34E-10	3.26E-10	6.05E-28	3.30E-28	1.35E-31	0
$f_{12}$	1.46E-01	1.09E-02	0	0	0	0
$f_{13}$	1.55E-09	5.54E-10	0	0	0	0
$f_{14}$	1.49E-01	3.55E-01	0	0	0	0
$f_{15}$	4.93E-04	2.25E-03	0	0	0	0

从表 2 中可以看出,CCABC 算法在 15 个函数上的性能都要明显优于实数编码单一策略的 ABC 算

法. 具体而言, 在  $f_6, f_7$  和  $f_{12} \sim f_{15}$  上, CCABC 算法和 ABC/best/1 均取得全局最优值; 而在函数  $f_8$  上, 仅有 CCABC 算法取得了全局最优值. 对于函数  $f_1 \sim f_5$  和  $f_9 \sim f_{11}$ , CCABC 算法的计算精度与 ABC/rand/1 算法、ABC/best/1 算法相比均有提高. 因此, 与实数编码单一策略的 ABC/rand/1 算法和 ABC/best/1 算法相比, CCABC 在性能上具有明显的改善, 能够获取更高的数值精度.

为了进一步直观展示出 CCABC 算法的寻优效果, 将 ABC/best/1 算法和 CCABC 算法对两种测试函数的收敛曲线进行比较, 如图 1 和图 2 所示.

在函数  $f_3$  的进化过程中, 算法均以较快的速度收敛. 由函数  $f_9$  进化过程曲线可以看出, 在 CCABC 算法的运行前期, 算法的搜索速度较快, 每一代的最优值随迭代次数增加下降较为明显; 随着进化的进行, 适应度函数曲线变化逐渐平缓, 种群个体函数最优值向测试函数的最优解靠近. 与 ABC/best/1 算法相比, CCABC 算法在算法的计算精度和收敛速度上均有大幅度的提高.

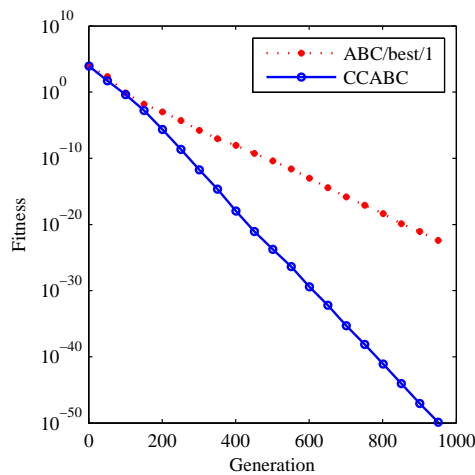


图 1 函数  $f_3$  收敛性能比较

Fig. 1 Convergence performance of different ABCs on  $f_3$

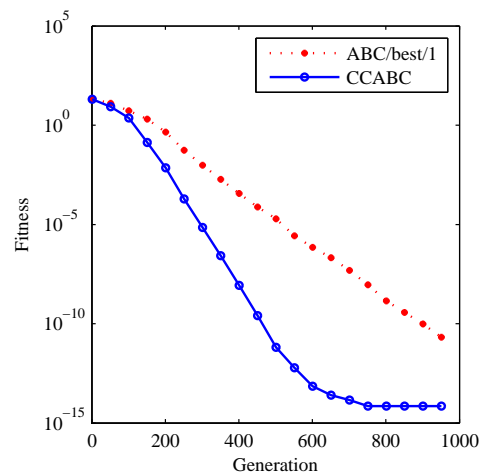


图 2 函数  $f_9$  收敛性能比较

Fig. 2 Convergence performance of different ABCs on  $f_9$

表 3 CCABC 算法与 MPEDE 算法、HCLPSO 算法和 JOA 算法的性能比较  
Table 3 Result comparisons of CCABC, MPEDE, HCLPSO and JOA

函数	MPEDE		HCLPSO		JOA		CCABC	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
$f_1$	4.49E-43	1.00E-42	6.60E-43	8.94E-43	6.32E-37	1.39E-36	3.80E-52	1.65E-52
$f_3$	1.33E-46	1.64E-46	9.91E-46	1.31E-45	4.45E-38	8.82E-38	2.26E-53	1.50E-54
$f_4$	1.13E-98	2.35E-98	9.00E-140	1.31E-139	1.48E-88	3.07E-88	3.28E-88	3.88E-91
$f_5$	1.26E-20	1.65E-20	4.06E-23	5.43E-23	4.05E-23	4.63E-23	1.49E-27	8.91E-28
$f_7$	0	0	2.96E-03	4.05E-03	0	0	0	0
$f_9$	3.55E-15	0	2.34E-14	4.77E-15	1.15E-14	3.55E-15	7.11E-15	0
$f_{10}$	3.33E-06	4.76E-06	5.87E-15	7.65E-15	2.67E-15	114E-15	2.02E-26	4.76E-27
$f_{11}$	1.35E-31	0	1.35E-31	0	1.37E-31	6.21E-33	1.35E-31	0
$f_{13}$	0	0	0	0	0	0	0	0
$f_{14}$	0	0	0	0	0	0	0	0

为评估 CCABC 算法在求解全局优化问题上的性能, 选取 10 个测试函数, 将 CCABC 算法与近期相关文献提出的三种优化算法(MPEDE 算法<sup>[31]</sup>, HCLPSO 算法<sup>[32]</sup>和 JOA 算法<sup>[11]</sup>)的计算结果进行对比. 表 3 给出每种算法在每个测试函数上的最优解的均值和标准方差. 从表 3 中可以看出, CCABC 算法在 8 个测试函数上取得了最佳性能. 在函数  $f_9$  上得到的最优值略差于 MPEDE 算法, 但具有相同的数量级, 且优于 HCLPSO 算法和 JOA 算法. 仅在函数  $f_4$  上, CCABC 算法的表现逊于其他三种算法. 通过与其他算法的对比可以看出, CCABC 算法在全局优化问题上能够取得相对满意的优化效果.

## 5 结束语

优化问题广泛存在于社会经济、科学技术与工业制造等多个领域. 智能算法为优化问题的求解提供了新的思路和方法. 人工蜂群算法是继遗传算法、粒子群算法等方法之后提出的一种群体智能优化算法, 在解决优化问题方面表现出良好的性能. 基本人工蜂群算法计算精度较低, 要提高求解精度, 可以从解的编码方式、搜索策略和参数设置等多个方面进行改进. 本文提出基于复数编码的多策略人工蜂群算法. 该算法运用复数编码方法产生多样性好的解集, 并建立策略知识库, 在搜索过程中自适应选择搜索策略对解集进行更新. 选取 15 个测试函数进行仿真实验, 并将实验结果与其他人工蜂群算法、MPEDE算法、HCLPSO 算法和 JOA 算法进行比较. 数值计算结果证明了 CCABC 算法的有效性. 该算法在解决全局优化问题上与其他算法相比具有一定的优越性. 对于如何将其应用到其他类型优化问题及实际工程领域中, 将是下一步研究的方向.

### 参考文献:

- [1] Holland J. *Adaptation in Natural and Artificial Systems*. Michigan, America: University of Michigan Press, 1975.
- [2] Storn R, Price K V. Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 1997, 11(4): 341–359.
- [3] Larra-naga P, Lozano J. *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*. Boston: Kluwer Academic Publishers, 2002.
- [4] Kennedy J, Eberhart R C. Particle swarm optimization // *Proceedings of the 1995 IEEE International Conference on Neural Networks*. Perth: IEEE Press, 1995, 4: 1942–1948.
- [5] Dorigo M, Maniezzo V. The ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, 1996, 26(1): 30–41.
- [6] Bastos Filho C J, Lima Neto F B. A novel search algorithm based on fish school behavior // *IEEE International Conference on Systems, Man and Cybernetics*. Suntec Singapore: IEEE Press, 2008: 2646–2651.
- [7] Zhao R Q, Tang W S. Monkey algorithm for global numerical optimization. *Journal of Uncertain Systems*, 2008, 2(3): 164–175.
- [8] Karaboga D. *An Idea Based on Honey Bee Swarm for Numerical Optimization*. Kayseri: Erciyes University, 2005.
- [9] Ramezani F, Lotfi S. Social-based algorithm (SBA). *Applied Soft Computing*, 2013, 13(5): 2837–2856.
- [10] Rao R V, Savsani V J. Teaching-learning-based optimization: An optimization method for continuous non-linear large scale problems. *Information Sciences*, 2012, 183(1): 1–15.
- [11] Sun G J, Zhao R Q, Lan Y F. Joint operations algorithm for large-scale global optimization. *Applied Soft Computing*, 2016, 38: 1025–1039.
- [12] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 2007, 39(3): 459–471.
- [13] 贾宗圣, 司锡才, 王 桐. 基于人工蜂群技术的海杂波参数优化方法. *中南大学学报(自然科学版)*, 2012, 43(9): 3485–3489.  
Jia Z S, Si X C, Wang T. Optimum method for sea clutter parameter based on artificial bee colony. *Journal of Central South University (Natural Science Edition)*, 2012, 43(9): 3485–3489. (in Chinese)
- [14] Karaboga D, Ozturk C. A novel clustering approach: Artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 2011, 11(1): 652–657.
- [15] Pan Q, Tasgetiren M, Suganthan P. A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Information Sciences*, 2011, 181(12): 2455–2468.
- [16] Karaboga D, Gorkemli B, Ozturk C, et al. A comprehensive survey: Artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, 2014, 42(1): 21–57.
- [17] Zhu G, Kwong S. Gbest-Guided artificial bee colony algorithm for numerical function optimization. *Applied Mathematics and Computation*, 2010, 217(7): 3166–3173.
- [18] Banharnsakun A, Achalakul T, Sirinaovakul B. The best-so-far selection in artificial bee colony algorithm. *Applied Soft Computing*, 2011, 11(2): 2888–2901.



- [19] Gao W, Liu S, Huang L. A global best artificial bee colony algorithm for global optimization. *Journal of Computational and Applied Mathematics*, 2012, 236(11): 2741–2753.
- [20] Gao W, Liu S. Improved artificial bee colony algorithm for global optimization. *Information Processing Letters*, 2011, 111(17): 871–882.
- [21] Li G Q, Niu P F, Xiao X J. Development and investigation of efficient artificial bee colony algorithm for numerical optimization. *Applied Soft Computing*, 2012, 12(1): 320–332.
- [22] Gao W F, Liu S Y, Huang L L. A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. *IEEE Transactions on Cybernetics*, 2013, 43(3): 1011–1024.
- [23] Gao W F, Huang L L, Liu S Y. Artificial bee colony algorithm based on information learning. *IEEE Transactions on Cybernetics*, 2015, 45(12): 2827–2839.
- [24] Kang F, Li J, Xu Q. Structural inverse analysis by hybrid simplex artificial bee colony algorithms. *Computers & Structures*, 2009, 87(13): 861–870.
- [25] Kang F, Li J J, Li Z Y. Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions. *Information Sciences*, 2011, 181(15): 3508–3531.
- [26] Kang F, Li J J, Li H J. Artificial bee colony algorithm and pattern search hybridized for global optimization. *Applied Soft Computing*, 2013, 13(4): 1781–1791.
- [27] Gao W, Liu S, Huang L. A novel artificial bee colony algorithm with Powell's method. *Applied Soft Computing*, 2013, 13(8): 3763–3775.
- [28] Chen S M, Sarosh A, Dong Y F. Simulated annealing based artificial bee colony algorithm for global numerical optimization. *Applied Mathematics and Computations*, 2012, 219(8): 3575–3589.
- [29] Mustafa S K, Mesut G. A recombination-based hybridization of particle swarm optimization and artificial bee colony algorithm for continuous optimization problems. *Applied Soft Computing*, 2013, 13(4): 2188–2203.
- [30] Casasent D, Natarajan S. A classifier neural network with complex-valued weights and square-law nonlinearities. *Neural Network*, 1995, 8(6): 989–998.
- [31] Wu G H, Mallipeddi R. Differential evolution with multi-population based ensemble of mutation strategies. *Information Sciences*, 2016, 329: 329–345.
- [32] Lynn N, Suganthan P N. Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. *Swarm and Evolutionary Computation*, 2015, 24: 11–24.

#### 作者简介:

单 娴 (1983—), 女, 河南邓州人, 博士, 讲师, 研究方向: 智能优化算法, Email: shanxian@163.com;

杜学东 (1963—), 男, 山东青岛人, 博士, 教授, 博士生导师, 研究方向: 计算机应用算法与模型, 优化模型与算法, Email: dxdlwzy@163.com.