

# 求解连续空间优化问题的改进蜂群算法\*

王永琦<sup>1</sup>, 吴飞<sup>1</sup>, 孙建华<sup>2</sup>

(1. 上海工程技术大学 电子电气工程学院, 上海 201620; 2. 湖南大学 信息科学与工程学院, 长沙 410082)

**摘要:** 为了有效地解决人工蜂群算法容易陷入局部最优的缺陷, 提出了一种改进蜂群算法。利用反向学习方法构建初始种群, 以提高初始化解的质量。同时, 利用分布估计算法构造优秀个体解空间的概率模型来进行邻域搜索, 以改善算法的搜索性能并防止陷入局部最优。对连续空间优化问题进行了仿真实验, 结果表明改进算法具有较快的收敛速度, 全局寻优能力显著提高。

**关键词:** 人工蜂群算法; 连续空间优化; 反向学习; 分布估计算法

**中图分类号:** TP183; TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2018)03-0658-03

doi:10.3969/j.issn.1001-3695.2018.03.004

## Modified artificial bee colony algorithm for solving continuous space optimization problems

Wang Yongqi<sup>1</sup>, Wu Fei<sup>1</sup>, Sun Jianhua<sup>2</sup>

(1. School of Electronic & Electrical Engineering Science, Shanghai University of Engineering, Shanghai 201620, China; 2. College of Information Science & Engineering, Hunan University, Changsha 410082, China)

**Abstract:** This paper proposed a modified artificial bee colony algorithm to tackle the dilemma of easily trapping in local optimum in original artificial bee colony. Firstly, it applied an opposition-based learning method for the initial population generation, which aimed to improve the quality of initial solutions. Meanwhile, it employed the estimation of distribution metaheuristic to establish the probability model of solution domain about good individuals for neighbor search. This operation was capable of improving the searching performance and avoiding local optimum. Finally, it conducted the simulations to tackle the continuous space optimization problems. The experimental results demonstrate that the modified algorithm has fast convergence speed and the global optimization capability is enhanced.

**Key words:** artificial bee colony(ABC) algorithm; continuous space optimization; opposition-based learning; estimation of distribution algorithm(EDA)

连续空间优化问题普遍存在于科学研究、经济管理等多个领域,其目标函数具有连续性、多峰值、噪声、旋转等特征,这使得传统的优化算法在求解质量、收敛性能以及计算效率等方面远不能满足工程运用需求<sup>[1,2]</sup>。与此同时,仿生学学科的快速衍生了一系列具有自适应、自组织、自学习且高度并行的仿生优化算法,包括遗传算法<sup>[3]</sup>、差分进化算法<sup>[4]</sup>、粒子群算法等<sup>[5]</sup>,这类智能算法已经成功运用于连续空间优化问题的求解中,并取得了良好的效果。人工蜂群(artificial bee colony, ABC)算法是 Karaboga 等人<sup>[6]</sup>受蜜蜂智能群体采蜜过程启发而提出的一种新型启发式优化算法。该算法通过具有控制参数少、求解精度高、搜索性能强等优点,已经成功地运用于连续空间优化<sup>[7]</sup>、数据挖掘<sup>[8]</sup>和神经网络训练<sup>[9]</sup>等多个方面。尽管如此,ABC 算法仍然存在着易陷入局部最优、过早收敛等缺点。为此,国内外许多学者提出了多种改进方法来提升人工蜂群算法的性能<sup>[7-9]</sup>。然而到目前为止,如何在提高算法收敛性的同时避免陷入局部最优仍然是一个难题。

分布估计算法(estimation of distribution algorithm, EDA)是一种基于统计学原理的智能优化算法。该算法通过对搜索空

间进行采样和统计学习来预测搜索的最佳区域,进而产生优秀的新个体,有着较强的全局搜索能力和较快的收敛速度<sup>[10]</sup>。

本文受分布估计算法思想的启发,提出了一种改进蜂群算法(modified artificial bee colony, MABC)。首先,借助反向学习技术构建初始种群;然后,构建优势个体解空间的概率分布模型,并利用该模型更新当前种群。将改进蜂群算法运用于连续空间优化问题,并与其他几种改进算法进行比较。实验结果表明,本文构建的改进蜂群算法具有较强的全局寻优能力。

### 1 标准蜂群算法

标准 ABC 算法通过模拟蜂群寻找蜜源的过程来实现问题的优化。该算法包含雇佣蜂(employed bees)、观察蜂(onlookers)和侦察蜂(scouts)三种角色的蜜蜂。其中,雇佣蜂、观察蜂的数量相同,每一个雇佣蜂对应着一个蜜源。算法采用随机初始化操作生成多个蜜源,后续每次迭代搜索过程都包含雇佣蜂、观察蜂和侦察蜂阶段三部分。在雇佣蜂阶段,每个雇佣蜂在对应的食物源附近搜索新的食物源并对其进行评价,采用贪婪选择操作进行一次食物源更新;在观察蜂阶段,各个观察

**收稿日期:** 2016-11-08; **修回日期:** 2016-12-19 **基金项目:** 国家自然科学基金资助项目(F020207);上海市科委资助项目(13510501400);上海市工程技术大学《信号与系统》平台课程建设项目(k201602004)

**作者简介:** 王永琦(1977-),女,河南焦作人,副教授,博士研究生,主要研究方向为智能计算、信号处理(wangyongqi17008@163.com);吴飞(1968-),男,山东人,教授,博士,主要研究方向为计算机并行处理、节能控制;孙建华(1978-),女,河南焦作人,副教授,博士,主要研究方向为网络安全和信息安全、入侵检测。

蜂结合当前所有食物源的适应度值采用轮盘赌算子选择某一食物源进行邻域搜索及更新;在侦察蜂阶段,当某个食物源在迭代一定次数后仍没有改进,则采用随机初始化操作重置该食物源位置。假设问题的维度为  $D$ ,  $x_d^{\min}$ 、 $x_d^{\max}$  分别表示第  $d$  自变量取值的最小值和最大值。令雇佣蜂的种群规模为  $S$ ,以最小化问题为例,上述算法的实现过程可归纳如下:

初始化阶段,  $\forall i \in \{1, 2, \dots, S\}$ ,  $d \in \{1, 2, \dots, D\}$ , 利用式(1)构建初始种群  $x = (x_1, x_2, \dots, x_S)$ 。

$$x_{id} = x_d^{\min} + \text{rand}(0, 1) \times (x_d^{\max} - x_d^{\min}) \quad (1)$$

雇佣蜂阶段和观察蜂阶段,采用式(2)进行邻域搜索以生成新蜜源。

$$v_{id} = x_{id} + \mu_{id} \times (x_{id} - x_{kd}) \quad (2)$$

其中:  $\mu_{id}$  表示  $[-1, 1]$  的随机数;  $i \in \{1, 2, \dots, S\}$ ,  $k \in \{1, 2, \dots, S\}$ , 且  $i \neq k$ 。

$\forall i \in \{1, 2, \dots, S\}$ , 各个蜜源  $x_i$  的适应度  $F_i$  计算公式为

$$F_i = \begin{cases} \frac{1}{1 + \text{fit}_i} & \text{若 } \text{fit}_i \geq 0 \\ 1 + \text{abs}(\text{fit}_i) & \text{否则} \end{cases} \quad (3)$$

其中:  $\text{fit}_i$  表示蜜源  $x_i$  目标函数值。观察蜂采用轮盘赌算子更新蜜源,  $\forall i \in \{1, 2, \dots, S\}$ , 各蜜源  $x_i$  被选中的概率值为

$$P_i = \frac{F_i}{\sum_{i=1}^S F_i} \quad (4)$$

重复上述雇佣蜂搜索、基于轮盘赌概率的观察蜂搜索和侦查蜂操作,直至满足算法收敛条件。

## 2 改进蜂群算法

### 2.1 反向学习初始化操作

传统蜂群算法采用随机初始化的方法,这在一定程度上限制了算法的优化效率。为了提升算法初始蜜源的质量,本文在改进蜂群算法中借助反向学习初始化操作。构建随机初始种群及其反向种群,并选择其中的优秀个体构建初始种群  $x = (x_1, x_2, \dots, x_S)$ 。给定雇佣蜂种群数量为  $S$ 、问题维度为  $D$ ,以及各维度自变量取值  $x_d^{\min}$ 、 $x_d^{\max}$  ( $d \in \{1, 2, \dots, D\}$ ),反向学习初始化操作的具体步骤如下:

a) 令  $i \leftarrow 1, d \leftarrow 1$  转步骤 b); b) 若  $i \leq S$ , 转步骤 c), 否则转步骤 g); c) 若  $d \leq D$ , 转步骤 d), 否则步骤 e); d) 令  $x_{id}^p = x_d^{\min} + \text{rand}(0, 1) \times (x_d^{\max} - x_d^{\min})$ ,  $x_{id}^o = x_d^{\min} + x_d^{\max} - x_{id}^p$ , 并转步骤 e); e) 令  $d \leftarrow d + 1$ , 若  $d > D$ , 转步骤 f), 否则转步骤 c); f) 令  $i \leftarrow i + 1$ , 转步骤 b); g) 选择集合  $\{x_i^p\}$  和  $\{x_i^o\}$  中前  $S$  个优秀个体构建初始种群  $x = (x_1, x_2, \dots, x_S)$ 。

### 2.2 基于分布估计算法的邻域更新机制

在标准的蜂群算法中,邻域变换采用式(2)搜索当前蜜源附近新的蜜源,并利用贪婪选择操作进行邻域更新。然而随着迭代次数的增加,各个蜜源(个体)之间的相似度增加,当存在多个相同的优质蜜源时,式(2)则无法产生新的蜜源。

为了说明这一点,不妨假设 ABC 算法经过多次迭代后获得两蜜源  $x_i$  和  $x_k$ 。其中,两个解中第  $d$  维变量  $x_{id}$ 、 $x_{kd}$  均为 3.2, 则采用式(2)的差分计算公式进行邻域变换时,等式  $v_{id} = 3.2 + \mu_{id} \times (3.2 - 3.2)$  成立,其中参数  $\mu_{id}$  表示  $[-1, 1]$  的随机数。据此可知产生的新个体中第  $d$  维变量  $v_{id}$  仍为 3.2, 这将导致算法陷入局部最优,并一定程度上限制了 ABC 算法的寻优能力。随着迭代次数的增加,上述问题使得标准蜂群算法的搜索过程陷入停滞。与此同时,由于侦查蜂操作的存在,基于式(2)的邻域搜索操作使得 ABC 算法不得不放弃当前较优质的解,转

而接受新的随机初始蜜源,这在一定程度上也限制了算法的搜索性能。为此,本文结合分布估计算法的设计思想,通过构建优秀个体解空间的概率模型来进行个体更新,从而改善算法的搜索性能并防止陷入局部最优。

给定参数  $\lambda \in (0, 1)$ , 令  $L = \lfloor \lambda \times S \rfloor$ , 选择当前种群中  $L$  个优秀个体构建各维度分布的概率模型。以分布向量  $V = (v_1, v_2, \dots, v_D)$  表示当前优秀种群的分布情况,假设各维度分量服从标准正态分布<sup>[11]</sup>, 则分布向量  $V$  可分解为两个子分布向量  $V_d = \{v_d^m, v_d^o\}$ ,  $\tilde{V}_d = \{\tilde{v}_d^m, \tilde{v}_d^o\}$ , 以分别表示各维度分量的均值及方差,第  $t$  次迭代完成后,第  $t+1$  次迭代的分布向量中的两类参数均值  $v_d^m(t+1)$  及方差  $v_d^o(t+1)$  可以由式(5)~(8)获得。

$$v_d^m(t+1) = (1 - \alpha_m) \times v_d^m(t) + \alpha_m \times \tilde{v}_d^m(t) \quad (5)$$

$$v_d^o(t+1) = (1 - \alpha_o) \times v_d^o(t) + \alpha_o \times \tilde{v}_d^o(t) \quad (6)$$

$$\tilde{v}_d^m(t+1) = \frac{\sum_{i \in L} x_{id}}{L} \quad (7)$$

$$\tilde{v}_d^o(t+1) = \sqrt{\frac{\sum_{i \in L} (x_{id} - \tilde{v}_d^m(t+1))^2}{L}} \quad (8)$$

其中:参数  $\alpha_m$ 、 $\alpha_o$  表示平滑系数。优秀个体解空间的概率模型更新公式包含两部分:a)对当前优势个体的统计学习过程,由式(7)(8)表示;b)对历史优势个体的统计学习过程,由式(5)(6)表示。借助该概率模型,利用正态分布模型生成优秀的新个体,以进行雇佣蜂、观察蜂阶段的邻域搜索,能够有效地避免式(2)在出现邻域更新失效的情形,一定程度上避免陷入局部最优,提高了算法的收敛性能。

### 2.3 MABC 算法流程及复杂度分析

基于以上讨论,通过在标准 ABC 优化算法中嵌入分布估计方法构建改进蜂群算法。MABC 算法流程归纳如下:

a) 初始化 MABC 算法各参数,包括种群规模  $S$ 、优秀个体参数  $L$ 、侦查蜂参数  $\xi$  等,转步骤 b);

b) (初始化阶段) 利用反向学习初始化操作构建初始种群,并选择其中  $L$  个优秀个体生成初始概率模型,转步骤 c);

c) (雇佣蜂操作) 针对当前种群中的每个蜜源,利用基于分布估计算法的邻域更新机制进行个体更新操作,转步骤 d);

d) (观察蜂操作) 基于当前种群的各蜜源的适应度值,利用轮盘赌操作选择某一蜜源,并采用基于分布估计算法的邻域更新机制进行个体更新操作,重复  $S$  次,转步骤 e);

e) (侦查蜂操作) 针对当前种群中完成  $\xi$  次迭代仍没有改进的蜜源进行重新初始化操作,转步骤 f);

f) (概率模型更新) 选择当前种群中  $L$  个优秀个体更新概率模型,转步骤 g);

g) (终止条件判断) 若算法尚未满足终止条件,则转步骤 c); 否则,终止 MABC 算法,并输出到当前所得的最优解。

由上述算法流程可知,MABC 算法的各次迭代过程包含以下关键操作:当前种群的适应度值排序、种群更新、轮盘赌选择算子和概率模型更新。给定连续空间优化问题的维度  $D$ ,则各项操作的时间复杂度分析如下:

a) 利用快速排序法对当前种群中的  $S$  个个体进行排序,以获取  $L$  个优秀个体的时间复杂度为  $O(S \times \log_2 S)$ ;

b) 采用基于分布估计算法的邻域更新机制对当前种群中的  $S$  个个体进行更新操作,所需的算法复杂度为  $O(S \times D)$ ;

c) 利用轮盘赌算子从当前种群中的  $S$  个个体选择某一个体的时间复杂度为  $O(S)$ ;

d) 利用当前种群中  $L$  个优秀个体进行概率模型更新的时间复杂度为  $O(L \times D)$ 。

综合 NABC 算法迭代搜索过程的各项关键操作可知,该算法的时间复杂度并不大,能够有效地应对复杂的连续空间优化问题的求解。

### 3 仿真实验及分析

为了验证 MABC 算法寻优性能,将 MABC 算法运用于连续空间优化问题的求解,选取文献[7]中的五个测试函数进行仿真实验。表 1 给出了这五个测试函数的名称、表达式及自变量的取值范围, $n$  表示函数的维度。

表 1 测试函数

函数名称	函数定义	取值区间
Tablet	$f_1(x) = 10^6 x_1^2 + \sum_{i=2}^n x_i^2$	$(-100, 100)^n$
Sphere	$f_2(x) = \sum_{i=1}^n x_i^2$	$(-100, 100)^n$
Rosenbrock	$f_3(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$(-50, 50)^n$
Rastrigin	$f_4(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$(-5.12, 5.12)^n$
Griewank	$f_5(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	$(-100, 100)^n$

表 1 中,  $f_1$ 、 $f_2$ 、 $f_3$  为典型的单峰值测试函数,  $f_4$ 、 $f_5$  为多峰值函数。其中,  $f_3$  (Rosenbrock 函数) 和  $f_5$  (Griewank 函数) 的 2D 空间分布如图 1、2 所示。这类测试函数极其复杂,难以获得全局最优,因此常用于智能优化算法的性能评价。

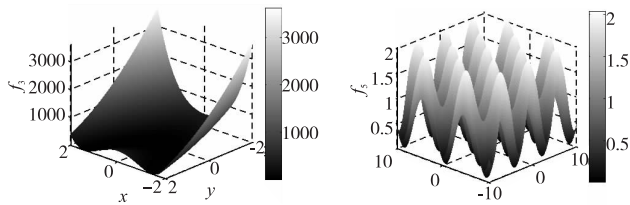


图1 Rosenbrock函数2D空间分布 图2 Griewank函数2D空间分布

为了评价本文构建的改进蜂群算法的寻优性能,基于 MATLAB 2015a 仿真平台,在主频为 2.4 GHz、内存 4 GB、Intel Core i5-2430M CPU 的便携式计算机上进行编程及仿真。MABC 算法参数设置如下:种群规模  $S$  为 80,最大迭代次数为 2 000,调节参数  $L$  为 300。此外,结合测试实验结果,优秀个体规模参数  $\lambda$ 、学习系数  $\alpha_m$ 、 $\alpha_\sigma$  分别设置为 0.4、0.4、0.5。实验结果分析如下。

#### 3.1 MABC 与标准蜂群算法寻优结果比较

将 MABC 与标准蜂群算法(ABC)进行寻优性能对比,ABC 算法的相关参数设置与 MABC 算法相同。分别考虑维度为 20、40 两种情形,针对每个算例 ABC 和 MABC 算法分别运行 20 次,并对优化结果的均值和标准差进行统计,如表 2 所示。

由表 2 可知,与标准人工蜂群算法相比,MABC 算法优化结果的均值和标准差相对较小,说明 MABC 算法具有较好的寻优能力,能够在一定程度上避免陷入局部最优。此外,给出了部分测试函数的 ABC 和 MABC 算法进化曲线,如图 3、4 所示,收敛曲线也在一定程度上验证了改进算法的有效性。

由表 3 可知,融合分布估计算法思想的改进蜂群算法 MABC(R) 的寻优结果由于 GABC、ABC/Best/1 算法,表明借助优质种群的解空间概率模型进行个体更新有利于防止算法陷入局部最优;同时,MABC 的实验结果优于 MABC(R),表明融合反向学习的初始化操作能有效地提高算法的收敛性能。

#### 3.2 MABC 与几种改进蜂群算法寻优结果比较

为了进一步验证 MABC 算法的收敛性能,将本文算法与

其他几种改进的蜂群算法进行对比,包括文献[12]中的 GABC 算法和文献[13]中的 ABC/Best/1 算法。此外,将不包含反向学习初始化的改进蜂群算法 MABC(R) 也加入算法优化结果对比,三种比较算法的参数设置与 MABC 算法相同。针对每个算例,各算法运行 20 次,并统计寻优结果的平均值与标准差,如表 3 所示。

表 2 MABC 与 ABC 算法优化结果比较

函数	维度	算法	平均值	标准差
$f_1$	20	ABC	$8.18e-14$	$2.88e-15$
		MABC	$2.79e-20$	$6.50e-20$
	40	ABC	$2.43e-08$	$4.80e-09$
		MABC	$5.52e-16$	$1.97e-16$
$f_2$	20	ABC	$5.68e-16$	$7.36e-17$
		MABC	$4.65e-22$	$8.37e-22$
	40	ABC	$1.76e-12$	$7.91e-12$
		MABC	$5.81e-20$	$4.39e-20$
$f_3$	20	ABC	$8.99e-02$	$7.97e-02$
		MABC	$1.85e-04$	$5.70e-04$
	40	ABC	$3.58e-01$	$3.82e-01$
		MABC	$2.09e-03$	$5.03e-03$
$f_4$	20	ABC	$8.99e-16$	$7.978e-16$
		MABC	$1.85e-19$	$5.70e-20$
	40	ABC	$3.58e-13$	$3.82e-13$
		MABC	$2.09e-17$	$5.03e-17$
$f_5$	20	ABC	$5.24e-15$	$3.47e-15$
		MABC	$1.48e-17$	$2.72e-18$
	40	ABC	$8.89e-12$	$6.72e-12$
		MABC	$2.29e-15$	$5.60e-16$

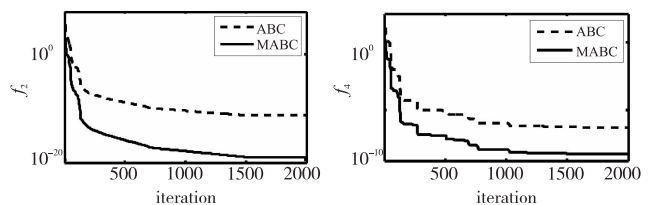


图3 Sphere函数寻优进化曲线 图4 Rastrigin函数寻优进化曲线

表 3 四种改进蜂群算法对比

函数	维度	算法	平均值	标准差
$f_1$	40	GABC	$5.13e-13$	$7.43e-13$
		ABC/Best/1	$1.25e-14$	$3.67e-14$
		MABC(R)	$6.30e-15$	$8.05e-15$
		MABC	$5.05e-16$	$3.60e-16$
$f_2$	40	GABC	$7.42e-17$	$6.48e-17$
		ABC/Best/1	$4.99e-16$	$2.66e-16$
		MABC(R)	$6.16e-18$	$2.95e-19$
		MABC	$5.27e-20$	$6.01e-21$
$f_3$	40	GABC	$1.61e-02$	$6.72e-02$
		ABC/Best/1	$2.51e-03$	$8.04e-02$
		MABC(R)	$5.93e-03$	$3.24e-03$
		MABC	$3.14e-03$	$6.91e-04$
$f_4$	40	GABC	$8.94e-14$	$2.90e-14$
		ABC/Best/1	$5.47e-17$	$2.56e-16$
		MABC(R)	$7.87e-16$	$8.96e-17$
		MABC	$8.10e-17$	$6.16e-17$
$f_5$	40	GABC	$1.59e-13$	$6.78e-13$
		ABC/Best/1	$3.85e-12$	$7.61e-13$
		MABC(R)	$6.03e-14$	$5.70e-14$
		MABC	$8.78e-15$	$7.33e-15$

### 4 结束语

本文在标准人工蜂群算法的基础上,构建了一种改进蜂群算法。MABC 算法采用融合反向学习的初始化操作以有效地提升初始解的质量;同时,结合分布估计算法的思想,通过构造优质个体解空间的概率模型进行邻域搜索 (下转第 704 页)

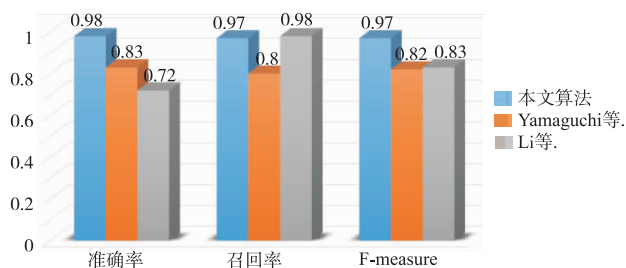


图9 公共语料库同类算法对比示意图

通过对公共语料库 medstract gold standard evaluation corpus 上测试的九对不正确结果进行分析,笔者发现错误结果都只是提取了缩略语全称中的一部分。例如,算法提取的〈Pol II, polymerase II〉的正确结果是〈Pol II, RNA polymerase II〉,〈Pol II, polymerase I〉的正确结果是〈Pol II, RNA polymerase I〉。

另外,本算法在公共语料库 Medstract Gold Standard Evaluation Corpus 上的测试中共丢失了 13 对。其中,8 对是缩略语 (short-form) 和全称 (long-form) 之间没有任何的匹配规则,如〈5-HT, serotonin〉;2 对是缩略语与全称字母顺序不匹配,如〈ATN, anteriore thalamus〉;3 对是缩略语中忽略了全称中某个内部单词,如〈CNSI, cyclophilin seven suppressor〉。

#### 4 结束语

提出了一种基于逆序扫描和共现分析相结合的缩略语快速提取算法,该算法在缩略语中字母无须全部正确匹配全称的情况下,能够快速提取出缩略语及其全称。从测试的结果来看,该算法的准确率和召回率同传统算法相比均有明显提升。但同时本文发现,尽管该算法对于不规则缩略语采用了共现分析的统计方法,取得了一定的效果,但对于单篇科技资源中不规则缩略语的提取准确率还需要进一步提高。

#### 参考文献:

- [1] Schwartz A S, Hearst M A. A simple algorithm for identifying abbreviation definitions in biomedical text[C]//Proc of Pacific Symposium on Biocomputing. 2013:451-462.

(上接第 660 页)以避免陷入局部最优。最后,将 MABC 算法运用于连续空间优化问题的求解,并与多种改进的蜂群算法的测试结果进行比较。实验结果表明,MABC 算法的收敛效果显著,具有较强的寻优能力。

#### 参考文献:

- [1] Munoz M, Kirley M, Halgamuge S K. Exploratory landscape analysis of continuous space optimization problems using information content [J]. IEEE Trans on Evolutionary Computation, 2015, 19(1): 74-87.
- [2] Stein O, Oldenburg J, Marquardt W. Continuous reformulations of discrete-continuous optimization problems[J]. Computers & Chemical Engineering, 2004, 28(10): 1951-1966.
- [3] Hadj-Alouane A B, Bean J C. A genetic algorithm for the multiple-choice integer program[J]. Operations Research, 2010, 45(1): 92-101.
- [4] Das S, Suganthan P N. Differential evolution: a survey of the state-of-the-art[J]. IEEE Trans on Evolutionary Computation, 2011, 15(1): 4-31.
- [5] Park J B, Jeong Y W, Shin J R, et al. An improved particle swarm optimization for nonconvex economic dispatch problems [J]. IEEE Trans on Power Systems, 2010, 25(1): 156-166.
- [6] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm

- [2] Yamaguchi A, Yamamoto Y, Kim J D, et al. Discriminative application of string similarity methods to chemical and non-chemical names for biomedical abbreviation clustering[C]//Proc of IEEE International Conference on Bioinformatics and Biomedicine. [S. l.]: IEEE Press, 2011: 12-15.
- [3] Sun Xu, Li Wenjie, Meng Fang, et al. Generalized abbreviation prediction with negative full forms and its application on improving Chinese Web search[C]//Proc of the 6th International Joint Conference on Natural Language Processing. 2013: 641-647.
- [4] Aravalli R N, Talbot N C, Steer C J. Gene expression profiling of MYC-driven tumor signatures in porcine liver stem cells by transcriptome sequencing[J]. World Journal of Gastroenterology, 2015, 21(7): 2011-2029.
- [5] Tsuchiya S, Imono M, Yoshimura E, et al. Meaning judgment method for alphabet abbreviation using the association mechanism[C]//Proc of the 16th International Conference on Knowledge Based and Intelligent. 2012: 209-218.
- [6] Park Y, Kang S, Seo J. named entity recognition using wikipedia and abbreviation generation[C]//Proc of International Conference on Big Data and Smart Computing. [S. l.]: IEEE Press, 2014: 169-172.
- [7] Lestari D P, Furi S. Automatic abbreviation detection using syllable composition rules for Indonesian spoken query-based information retrieval[C]//日本音响学会 2011 年春季讲演论文集. 2011: 105-108.
- [8] Yamaguchi A, Yamamoto Y, Kim J D, et al. Discriminative optimization of string similarity and its application to biomedical abbreviation clustering[C]//Proc of the 10th International Conference on Machine Learning and Applications and Workshops. [S. l.]: IEEE Press, 2011: 18-21.
- [9] Xie Lixing, Zheng Yabin, Liu Zhiyuan, et al. Extracting Chinese abbreviation-definition pairs from anchor texts[C]//Proc of International Conference on Machine Learning and Cybernetics. [S. l.]: IEEE Press, 2011: 1485-1491.
- [10] Li Feng, Ma Wei-tong. The acquisition of the semantic meaning of abbreviations in Japanese and the usage strategies: a study on katakana abbreviation on alphabet abbreviation form[J]. Journal of Foshan University: Social Science, 2012(3): 53-57.
- [11] Zerkina N, Kostina N, Pitina S A. Abbreviation semantics[J]. Procedia-Social and Behavioral Sciences, 2015, 199(2): 137-142.
- [J]. Journal of Global Optimization, 2007, 39(3): 459-471.
- [7] Karaboga D, Basturk B. On the performance of artificial bee colony (ABC) algorithm[J]. Applied Soft Computing, 2008, 8(1): 687-697.
- [8] Karaboga D, Ozturk C. A novel clustering approach: artificial bee colony (ABC) algorithm[J]. Applied Soft Computing, 2011, 11(1): 652-657.
- [9] Hsieh T J, Hsiao H F, Yeh W C. Forecasting stock markets using wavelet transforms and recurrent neural networks: an integrated system based on artificial bee colony algorithm[J]. Applied Soft Computing, 2011, 11(2): 2510-2525.
- [10] Larraaga P, Lozano J A. Estimation of distribution algorithms: a new tool for evolutionary computation[J]. Studies in Fuzziness & Soft Computing, 2002, 192(1): 1982-1987.
- [11] Gao Shang, Qiu Ling, Cao Cungen. Improved estimation of distribution algorithms based on Gaussian distribution[J]. Research Journal of Applied Sciences Engineering & Technology, 2013, 6(10): 1841-1845.
- [12] Zhu Guopu, Kwong S. Gbest-guided artificial bee colony algorithm for numerical function optimization[J]. Applied Mathematics & Computation, 2010, 217(7): 3166-3173.
- [13] Gao Weifeng, Liu Sanyang, Huang Lingling. A global best artificial bee colony algorithm for global optimization[J]. Journal of Computational & Applied Mathematics, 2012, 236(11): 2741-2753.