

•智能交叉科学与工程•

DOI:10.12454/j.jsuese.202301064



本刊网刊

## 基于探采分工的人工蜂群算法及其在食管癌预测中的应用

王英聪,严 军,孙军伟,王延峰\*

(郑州轻工业大学 电气信息工程学院,河南 郑州 450002)

**摘要:**人工蜂群(ABC)算法在搜索过程中侧重探索但不擅长开采,造成收敛速度慢、求解精度低等问题。为此,本文提出一种基于探采分工的ABC算法,包含雇佣蜂探索、跟随蜂开采和侦察蜂补充3个阶段。在探索阶段,考虑到过多地利用随机解信息会偏向随机搜索,为雇佣蜂设计基于多样性精英引导的解搜索方程,并引入广度优先搜索策略加强探索。在开采阶段,考虑到过多地利用最优解信息会导致早熟收敛,为跟随蜂设计基于目标值精英引导的解搜索方程,并利用深度优先搜索策略强化开采。在侦察蜂补充阶段,考虑到侦察蜂的随机初始化方式会丢失前期搜索经验,且雇佣蜂和跟随蜂搜索方程形式单一,为侦察蜂设计兼顾基于目标值的最优解、基于多样性的最优解和前期搜索经验的邻域搜索方程。在CEC2021测试集和食管癌预测问题上检验算法性能,并与6种ABC算法进行对比。对于数值优化问题,本文算法在大部分函数上取得了最优解,Friedman检验时排名第1,Wilcoxon检验时显著优于另外5种ABC算法。同时,本文算法在收敛速度和时间效率方面也占优。对于预测模型优化问题,与基本算法相比,本文算法准确率提高了15.72%,敏感性提高了20.14%,特异性提高了15.03%,F1分数提高了16.23%。

**关键词:**人工蜂群算法;探索与开采;精英引导;搜索方程

**中图分类号:**TP391

**文献标志码:**A

**文章编号:**2096-3246(2025)04-0112-11

优化问题在科学研究和工程应用中常见且关键,该问题的解决一直不断推动着科技和工程领域的创新。对于具有非凸、不连续、不可微分等特点的复杂优化问题,基于梯度的传统优化方法已不再适用。为此,学者们受自然界中生物集群行为的启发提出了群智能优化算法,具有与初始值无关、不依赖梯度信息、对函数要求低、求解性能良好等特点。目前,常见的群智能优化算法有人工蜂群算法<sup>[1]</sup>、萤火虫算法<sup>[2]</sup>、鸡群算法<sup>[3]</sup>、人工鱼群算法<sup>[4]</sup>、磷虾觅食算法<sup>[5]</sup>等。

人工蜂群(ABC)算法受蜜蜂的觅食行为启发,模拟了蜂群采蜜过程中个体的分工和信息共享机制。与遗传算法、差分进化算法、蚁群算法和粒子群算法相比,ABC算法的寻优能力和算法精度都具有明显优势<sup>[1]</sup>。此外,ABC算法还具有结构简单、易于实现、鲁棒性强、控制参数少等优点。ABC算法一经提出便受到众多学者的关注,已被广泛应用于路径规划<sup>[6]</sup>和布局优化<sup>[7]</sup>等领域。

然而,与其他群智能优化算法相似,ABC算法也

存在收敛速度慢、求解精度低等不足。究其原因,主要是ABC算法在搜索过程中侧重探索但不擅长开采<sup>[8]</sup>。学者们针对这一问题提出了很多改进方法,大致可以分为3类。第1类指出搜索方程是ABC算法生成候选解的唯一方式,通过在搜索方程中加入优良解信息来提高算法的开采能力,常用的优良解形式有全局最优解<sup>[9]</sup>、精英解<sup>[10-11]</sup>和邻域最优解<sup>[12-13]</sup>等。第2类认为ABC算法的单一搜索策略难以平衡探索和开采,可通过构造策略池并设计策略选择方法来提高搜索效率,如包含4种<sup>[14]</sup>、5种<sup>[15]</sup>和6种<sup>[16]</sup>搜索策略的策略池。第3类根据“没有免费午餐定理”,将ABC算法与其他算法混合,通过优势互补提高算法性能,用于混合的算法有差分进化算法<sup>[17]</sup>、粒子群算法<sup>[18]</sup>等。

上述改进方法均提高了ABC算法的性能,但学者们仍在不懈地进行改进研究。针对人工蜂群算法探索强、开采弱的问题,本文提出一种基于探采分工的人工蜂群算法(artificial bee colony algorithm based on the divi-

收稿日期:2023-12-25 修回日期:2024-04-09 网络出版日期:2024-06-03

基金项目:国家自然科学基金项目(62473342);中原高端人才计划项目(204200510003);河南省科技攻关项目(222102210277;242102210004)

作者简介:王英聪(1987—),男,副教授,博士。研究方向:智能计算。E-mail: ying\_cong\_wang@163.com

\*通信作者:王延峰, E-mail: yanfengwang@yeah.net

sion between exploration and exploitation, ABC\_DEE)。本文算法令雇佣蜂执行探索,跟随蜂实施开采,侦察蜂进行补充。在探索和开采阶段,分别利用多样性和目标值评价解质量,进而设计基于相应精英解引导的搜索方程,同时配合广度优先搜索策略和深度优先搜索策略强化探索和开采。令侦察蜂采用邻域搜索算子代替随机搜索,以弥补雇佣蜂和跟随蜂的搜索方程形式单一的不足。为验证ABC\_DEE算法的性能,将其用于函数优化和食管癌预测模型优化,并与其他改进ABC算法进行对比。

## 1 人工蜂群算法

ABC算法是一种模拟自然界蜂群分工协作觅食行为的群智能优化算法<sup>[1]</sup>。将食物源看作优化问题的解,食物源的蜜量对应解的质量。ABC算法按照式(1)随机生成 $N$ 个初始解(即食物源):

$$x_{ij} = x_j^{\min} + \text{rand}(0, 1)(x_j^{\max} - x_j^{\min}) \quad (1)$$

式中: $x_{ij}$ 为第 $i$ 个食物源的第 $j$ 维元素, $i=1,2,\dots,N,N$ 为食物源(雇佣蜂或跟随蜂)个数, $j=1,2,\dots,D,D$ 为搜索空间维数; $x_j^{\min}$ 和 $x_j^{\max}$ 分别为第 $j$ 维分量的下界和上界; $\text{rand}(0,1)$ 为生成 $[0,1]$ 上均匀分布的随机数的函数。初始化种群后,ABC算法进入雇佣蜂、跟随蜂和侦察蜂循环搜索阶段,直至满足终止条件(如达到最大函数评价次数 $E_{\max}$ )。

### 1.1 雇佣蜂阶段

每只雇佣蜂都对应一个食物源,雇佣蜂在食物源 $X_i=(x_{i1}, x_{i2}, \dots, x_{iD})$ 附近搜索新的食物源 $V_i=(v_{i1}, v_{i2}, \dots, v_{iD})$ ,搜索方程如下:

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

式中: $k$ 是随机选取的下标, $k \in \{1, 2, \dots, N\}$ ,且 $k \neq i$ ; $\varphi_{ij}$ 为 $[-1, 1]$ 上均匀分布的随机数。注意,雇佣蜂在食物源附近进行的是1维搜索。同时,雇佣蜂在 $X_i$ 和 $V_i$ 之间进行贪婪选择,若 $V_i$ 优于 $X_i$ ,则令 $X_i = V_i$ 。

### 1.2 跟随蜂阶段

雇佣蜂搜索结束后,将食物源信息分享给跟随蜂。跟随蜂按照概率 $P_i$ 选择食物源 $X_i$ :

$$P_i = F_i / \sum_{m=1}^N F_m \quad (3)$$

式中: $F_i$ 为食物源 $X_i$ 的适应度; $m$ 为求和索引,用于遍历所有食物源。由式(3)可知,食物源的适应度越大,被选中的概率越大。以最小化问题为例, $F_i$ 的计算公式为:

$$F_i = \begin{cases} 1/(1+f_i), & f_i \geq 0; \\ 1+|f_i|, & f_i < 0 \end{cases} \quad (4)$$

式中, $f_i$ 为食物源 $X_i$ 的目标函数值。一旦 $X_i$ 被选中,跟随蜂将按照式(2)搜索一个新的食物源 $V_i$ ,并在二者

之间进行贪婪选择。

### 1.3 侦察蜂阶段

如果雇佣蜂对应的食物源连续 $l$ 次未得到改进(用 $\delta_i$ 记录食物源 $X_i$ 未得到改进的次数),说明该食物源已耗尽。此时,雇佣蜂将放弃该食物源,转变为侦察蜂在搜索空间内随机生成一个新的食物源。

## 2 基于探采分工的人工蜂群算法

ABC算法侧重探索但不擅长开采,主要原因是雇佣蜂和跟随蜂的搜索方式(式(2))具有较强的探索能力却在开采方面表现不足<sup>[9]</sup>。在搜索食物源时,雇佣蜂在整个搜索空间内搜索,跟随蜂在优质食物源附近搜索。从探索与开采的角度来看,雇佣蜂主要负责探索,跟随蜂主要执行开采。本文旨在根据种群搜索过程中探索与开采的特点设计相应策略,分别强化雇佣蜂的探索和跟随蜂的开采,提出一种基于探采分工的人工蜂群算法(ABC\_DEE)。

### 2.1 雇佣蜂探索阶段

杜振鑫等<sup>[19]</sup>指出,探索是算法搜索未知区域的能力。在搜索方程中加入随机解信息有利于探索,如式(2)的探索能力强,就是源于其中食物源随机解 $X_k$ 的使用<sup>[9]</sup>。又如,将式(2)中的 $X_i$ 替换为另一个随机解后,其探索能力得到增强<sup>[20]</sup>。然而,过多地利用随机解信息会偏向随机搜索,降低算法收敛速度。Singh等<sup>[21]</sup>进一步指出,探索指搜索未知区域以发现有前景解的能力。探索一般与种群多样性正相关,多样性好说明算法探索能力强<sup>[22]</sup>。因此,可以利用多样性好且有前景的解来引导搜索,提高探索效率。

多样性反映了种群的分布情况,可以通过个体之间的差异来度量。在种群搜索过程中,个体有位置和目标值两个属性,相应的有两种差异度量方式。基于位置差异的度量方式直观但计算成本高,Cheng等<sup>[23]</sup>设计了一种基于目标值差异的方法,将种群中个体 $s$ 的多样性定义为:

$$d_s = |O_s - O_{\text{mid}}| \quad (5)$$

式中: $O_s$ 为个体 $s$ 的目标值, $O_{\text{mid}}$ 是按目标值排序后中间个体的目标值。

Cheng等<sup>[23]</sup>研究表明,式(5)定义的多样性有助于发现较好解,因而可以用来引导种群探索。同时,式(5)还可以作为一种解质量评价方式,将种群中最好的 $p \times N$ ( $p$ 为精英解比例)个解定义为基于多样性的精英解。受精英引导ABC算法启发<sup>[24]</sup>,为雇佣蜂设计如下探索方程:

$$v_{ij} = x_{ej}^{\text{div}} + \varphi_{ij}(x_{ej}^{\text{div}} - x_{ij}) \quad (6)$$

式中, $x_{ej}^{\text{div}}$ 是随机选择的基于多样性的精英解 $X_e^{\text{div}} =$

$(x_{e1}^{div}, x_{e2}^{div}, \dots, x_{eD}^{div})$  的第  $j$  维元素,  $x_{rj}$  是从  $\{X_1, X_2, \dots, X_N\}$  中选择的随机解  $X_r$  的第  $j$  维元素,  $i \neq e \neq r, e, r$  分别为精英解和随机解的下标。

最后, 考虑到广度优先搜索侧重探索<sup>[24]</sup>, 基于随机选择设计广度优先策略进一步加强雇佣蜂的探索能力。具体的, 令雇佣蜂每次从  $\{X_1, X_2, \dots, X_N\}$  中随机选择一个食物源进行搜索, 其不同于 ABC 算法中雇佣蜂对  $N$  个食物源的遍历搜索。以  $N=5$  为例, 图 1 简要对比了两种搜索策略(实线圆与虚线圆分别代表搜索前后的食物源)。图 1(a) 中, 5 个食物源均得到 1 次搜索, 图 1(b) 中, 随机选择食物源 1、3、4、2、4 进行搜索, 即食物源 5 未得到搜索, 而食物源 4 得到 2 次搜索。由图 1 可知, 广度优先搜索策略比基本搜索策略多了一定的随机性, 因而探索能力得到加强。

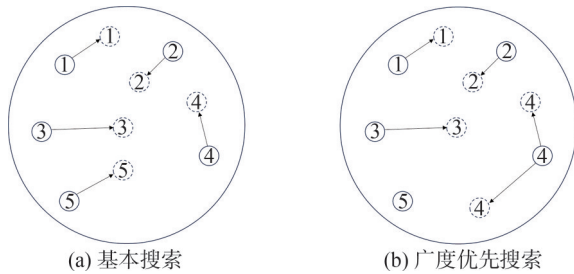


图 1 雇佣蜂搜索策略对比

Fig. 1 Comparison of search strategies for employed bees

## 2.2 跟随蜂开采阶段

开采指在已经搜索过的区域附近详细搜索的能力<sup>[9]</sup>, 尤其是利用经验知识在有前景区域进行精细搜索的能力<sup>[21]</sup>。在 ABC 算法中, 跟随蜂的开采取决于两部分: 优质食物源的选择和搜索方程的使用。下面从这两个方面对开采进行强化。

ABC 算法中跟随蜂根据式(3)选择食物源, 该方法存在两点不足: 一是食物源差异较大时会出现选择概率很大的超级食物源, 容易使算法陷入局部最优; 二是食物源差异较小时会退化随机选择, 不利于算法收敛。为此, 孔德鹏等<sup>[10]</sup>设计了一种基于排序的反比例选择方法。即将食物源按照目标值由好到差排序, 其中, 最好食物源的排名为 1, 最差食物源的排名为  $N$ 。令  $R_i$  表示食物源  $X_i$  的排名, 其被跟随蜂选择的概率  $Q_i$  为:

$$Q_i = (1/R_i) / \sum_{m=1}^N (1/R_m) \quad (7)$$

式(7)使用的是基于排序位置的固定概率, 其不受目标值变化的影响, 一定程度上克服了式(3)的两个不足。然而, 对于食物源差异较大或较小以外的情况, 固定概率难以反映食物源之间的真实差异, 而这种差异能够在式(3)得以体现。为此, 本文兼顾式(3)和(7), 为食物源  $X_i$  设计了选择概率  $\Gamma_i$  如下:

$$\Gamma_i = (P_i + Q_i) / 2 \quad (8)$$

对于搜索方程, 采用与式(6)相同的精英引导模式, 为跟随蜂设计如下开采方程:

$$v_{ij} = x_{ej}^{obj} + \varphi_{ij}(x_{ej}^{obj} - x_{rj}) \quad (9)$$

式中,  $x_{ej}^{obj}$  是随机选择的基于目标值的精英解  $X_e^{obj} = (x_{e1}^{obj}, x_{e2}^{obj}, \dots, x_{eD}^{obj})$  的第  $j$  维元素。

最后, 考虑到深度优先搜索侧重开采<sup>[24]</sup>, 基于奖励机制设计深度优先策略进一步加强跟随蜂的开采能力。具体地, 对于所选择的优质食物源, 令跟随蜂搜索成功后继续搜索, 直至食物源不再改进为止, 其不同于 ABC 算法中跟随蜂对所选优质食物源的单次搜索。同样, 以  $N=5$  为例, 假设跟随蜂分别选择优质食物源 5、2、3、5、3 进行搜索, 图 2 简要对比了两种搜索策略。图 2(a) 中共搜索了 5 次, 图 2(b) 中共搜索了 10 次, 多出的搜索可以看作是对成功搜索的奖励。由图 2 可知, 与基本搜索策略相比, 有前景的优质食物源在深度优先搜索策略下得到更多的计算资源, 因而开采能力得到加强。

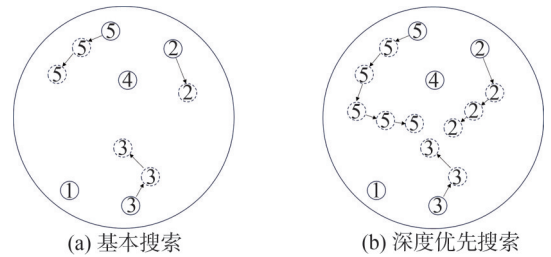


图 2 跟随蜂搜索策略对比

Fig. 2 Comparison of search strategies for onlooker bees

## 2.3 侦察蜂补充阶段

ABC 算法中侦察蜂负责抛弃枯竭的食物源, 并随机生成新的食物源继续搜索。随机的方式比较简单, 但是容易丢失搜索经验, 而且随机生成有前景的概率很小。学者们提出许多方法提高侦察蜂的搜索效率, 代表性的工作有: Wang 等<sup>[12]</sup>令侦察蜂采用 3 种搜索方程, 进而选择其中的最优解作为新的食物源; Peng 等<sup>[25]</sup>设计了一种不同于传统“基向量+扰动”的新颖搜索方程:

$$X_i^N = r_1 X_i^A + r_2 g_{best} + r_3 (X_a - X_b) \quad (10)$$

式中:  $X_i^N$  为侦察蜂生成的新解;  $r_1, r_2, r_3$  为  $[-1, 1]$  上均匀分布的随机数, 且满足  $r_1 + r_2 + r_3 = 1$ ;  $X_i^A$  为被侦察蜂抛弃的解,  $g_{best}$  为最优解;  $X_a$  和  $X_b$  为不同于  $X_i^A$  和  $g_{best}$  的下标分别为  $a, b$  的随机解。式(10)兼顾了搜索经验、最优解信息和随机扰动, 取得了不错的效果。

雇佣蜂和跟随蜂采用的都是“基向量+扰动”的搜索方程, 搜索方式相对单一。同时, 本文从多样性和目标值两个方面评价食物源质量, 可以引申出基于多样性的最优解和基于目标值的最优解。采用这两个最优解代替式(10)中  $g_{best}$ , 则可为侦察蜂设计两个搜索方程如下:

$$X_i^N = r_1 X_i^A + r_2 g_{\text{best}}^{\text{obj}} + r_3 (X_a - X_b) \quad (11)$$

$$X_i^N = r_1 X_i^A + r_2 g_{\text{best}}^{\text{div}} + r_3 (X_a - X_b) \quad (12)$$

式(11)~(12)中,  $g_{\text{best}}^{\text{obj}}$  为基于目标值的最优解,  $g_{\text{best}}^{\text{div}}$  为基于多样性的最优解。最终,令侦察蜂在式(11)和(12)之间进行贪婪选择。

## 2.4 算法伪代码

基于上述描述,ABC\_DEE算法的伪代码如下所示。

1. 初始化参数  $N, p, l, E_{\text{max}}$ ;
2. //  $p$  为控制精英解的比例系数
3. 按照式(1)生成初始种群,计算目标函数值;
4. 令函数评价次数  $E=N$ ;
5. while  $E \leq E_{\text{max}}$
6. for  $t=1$  to  $N$  //雇佣蜂探索阶段,  $t$  为循环计数变量
7. 利用广度优先策略选择食物源  $X_i$ ;
8. 按照式(6)搜索新的食物源  $V_i$ ;
9. if  $f(V_i) \leq f(X_i)$  //  $f(\cdot)$  为食物源目标函数值
10.  $X_i = V_i; \delta_i = 0$ ;
11. else
12.  $\delta_i = \delta_i + 1$ ;
13. end if
14. end for
15.  $E = E + N$ ;
16. 根据式(8)计算选择概率  $\Gamma_i$ ;
17. for  $t=1$  to  $N$  //跟随蜂开采阶段,
18. 根据选择概率按轮盘赌选择食物源  $X_i$ ;
19. 按照式(9)搜索新的食物源  $V_i$ ;
20. 初始化食物源更新状态  $U=0; E=E+1$ ;
21. while  $f(V_i) \leq f(X_i)$
22.  $X_i = V_i; \delta_i = 0; U=1$ ;
23. 按照式(9)搜索新的食物源  $V_i$ ;
24.  $E = E + 1$ ;
25. end while
26. if  $U = 0$
27.  $\delta_i = \delta_i + 1$ ;
28. end if
29. end for
30. if  $\max(\delta_i) > l$  //侦察蜂补充阶段
31. 按照式(11)和(12)择优搜索新的食物源;
32.  $\delta_i = 0; E = E + 2$ ;
33. end if
34. end while
35. 输出全局最优解

## 2.5 时间复杂度分析

按照文献[26]的计算方法,令  $M$  为迭代次数,则

基本ABC算法中种群初始化的复杂度为  $O(ND)$ ,雇佣蜂单维搜索、食物源选择概率计算和跟随蜂单维搜索的复杂度均为  $O(MN)$ ,侦察蜂全维搜索的复杂度为  $O(MD)$ 。因此,基本ABC算法的时间复杂度大约为  $O(ND+3MN+MD)$ 。与基本ABC算法相比,ABC\_DEE算法主要增加了多样性排序、目标值排序、多样性最优解和目标值最优解计算等步骤。排序的复杂度为  $O(M \log N)$ ,计算最优解复杂度为  $O(N)$ 。因此,ABC\_DEE算法的时间复杂度大约为  $O(ND+3MN+MD+2MN \log N)+2MN=O(ND+5MN+MD+2MN \log N)$ 。

尽管ABC\_DEE的复杂度大于ABC,但复杂度的增加一般是可以接受的。一方面,当函数评价成本较高时,算法的计算负担主要取决于函数评价<sup>[11]</sup>,算法改进带来的额外开销相对较小。另一方面,大部分情况下ABC\_DEE的收敛速度比ABC快,这意味着ABC\_DEE需要较少的函数评价就能获得一定精度的解。

## 3 数值实验

数值实验在优化算法的参数分析与性能评估等方面具有重要作用,围绕测试函数对ABC\_DEE开展数值实验,包括参数敏感性分析、策略有效性分析、与相关算法的对比分析等3部分。

### 3.1 测试函数

为了检验ABC\_DEE的性能,在CEC2021测试集上进行数值实验。CEC2021是一套求解难度较高的测试函数集,包含10个基本函数和70个变换函数。基本函数涵盖单峰、多峰、混合和复合4种类型,对基本函数进行偏差(bias)、移位(shift)和旋转(rotation)等操作后得到变换函数,详细函数信息见文献[27]。按照CEC2021的建议,测试函数的维度  $D=10$  和  $20$ ,最大函数评价次数  $E_{\text{max}}=200\,000$  和  $1\,000\,000$ 。实验环境为Windows 10操作系统,编程语言环境为MATLAB R2022b,硬件环境为英特尔酷睿i7-1165G7处理器,主频为2.8 GHz,内存为16 GB。

### 3.2 参数敏感性分析

与ABC相比,ABC\_DEE为雇佣蜂和跟随蜂设计了基于精英引导的搜索方程。参数  $p$  控制精英解的比例,对算法性能会有一定影响。本节选取  $p=0.01, 0.10, 0.40, 0.70, 0.90, 1.00$  这6个典型值在CEC2021的10个基本函数上进行敏感性测试,函数维度  $D=10$ ,种群规模  $N=50$ ,函数最大评价次数  $E_{\text{max}}=200\,000$ 。ABC\_DEE算法在不同  $p$  取值下独立运行30次的平均结果如表1所示,表1中,带\*数据表示最好结果,下表同。

表1 ABC\_DEE 采取不同 $p$ 取值的平均误差对比  
Tab. 1 Comparison of mean results for ABC\_DEE with different  $p$ -values

函数	$p=0.01$	$p=0.10$	$p=0.40$	$p=0.70$	$p=1.00$
$f_1$	$6.87 \times 10^{-254}$	$1.48 \times 10^{-257}$	$1.23 \times 10^{-220}$	$3.35 \times 10^{-216}$	$2.89 \times 10^{-190}$
$f_2$	*0	*0	$4.20 \times 10^{-3}$	$6.26 \times 10^{-2}$	$6.97 \times 10^{-2}$
$f_3$	*0	*0	*0	*0	*0
$f_4$	*0	*0	*0	*0	*0
$f_5$	$5.52 \times 10^{-101}$	$2.80 \times 10^{-193}$	$6.36 \times 10^{-144}$	$1.56 \times 10^{-120}$	$9.43 \times 10^{-98}$
$f_6$	$2.00 \times 10^{-3}$	$3.77 \times 10^{-4}$	$6.27 \times 10^{-3}$	$1.80 \times 10^{-2}$	$1.43 \times 10^{-2}$
$f_7$	$3.36 \times 10^{-5}$	$3.27 \times 10^{-7}$	$3.31 \times 10^{-7}$	$1.03 \times 10^{-6}$	$1.45 \times 10^{-6}$
$f_8$	*0	*0	*0	*0	*0
$f_9$	$6.27 \times 10^{-204}$	$9.42 \times 10^{-226}$	$1.64 \times 10^{-197}$	$9.56 \times 10^{-192}$	$4.57 \times 10^{-174}$
$f_{10}$	$6.76 \times 10^0$	$3.76 \times 10^{-4}$	$2.35 \times 10^{-4}$	$4.67 \times 10^{-4}$	$1.63 \times 10^0$

从表1可以看出, $p$ 取0.01和1.00这两个极端值时,算法整体表现较差。可能的原因是:一般而言,精英解介于随机解和最优解之间; $p=0.01$ 时,精英解进化为最优解,使得式(6)和(9)趋向贪婪; $p=1.00$ 时,精英解退化为随机解,使得式(6)和(9)失去精英引导价值。除此之外, $p$ 取较小值(即0.10)时,精英群体中的个体都是解质量评价相对较好的个体,有较好的引导作用,使得算法整体表现更好,这与文献[10,24]中对 $p$ 的选择一致。综合考虑下,本文设置 $p=0.10$ 。

与ABC算法相比,ABC\_DEE算法为侦察蜂设计了兼顾搜索经验的搜索方程。参数 $l$ 控制侦察蜂的执行频率,对算法性能会有一定影响。本节选取 $l=10$ 、50、100、200、ND这5个典型值进行敏感性测试,实验设置与对参数 $p$ 的敏感性实验保持一致。实验结果如表2所示。

表2 ABC\_DEE 采取不同 $l$ 取值的平均误差对比  
Tab. 2 Comparison of mean results for ABC\_DEE with different  $l$ -values

函数	$l=10$	$l=50$	$l=100$	$l=200$	$l=ND$
$f_1$	$2.45 \times 10^{-247}$	$1.48 \times 10^{-257}$	$8.85 \times 10^{-256}$	$4.46 \times 10^{-256}$	$2.93 \times 10^{-255}$
$f_2$	*0	*0	*0	*0	$2.08 \times 10^{-2}$
$f_3$	*0	*0	*0	*0	*0
$f_4$	*0	*0	$3.69 \times 10^{-18}$	$7.74 \times 10^{-13}$	$9.16 \times 10^{-4}$
$f_5$	$5.93 \times 10^{-87}$	$2.80 \times 10^{-193}$	$2.31 \times 10^{-179}$	$4.54 \times 10^{-182}$	$6.24 \times 10^{-71}$
$f_6$	$8.21 \times 10^{-4}$	$3.77 \times 10^{-4}$	$4.61 \times 10^{-3}$	$1.54 \times 10^{-2}$	$1.03 \times 10^{-1}$
$f_7$	$1.04 \times 10^{-2}$	$3.27 \times 10^{-7}$	$3.20 \times 10^{-5}$	$5.96 \times 10^{-4}$	$2.28 \times 10^{-2}$
$f_8$	*0	*0	*0	*0	*0
$f_9$	$9.39 \times 10^{-228}$	$9.42 \times 10^{-226}$	$2.62 \times 10^{-205}$	$4.83 \times 10^{-154}$	$8.06 \times 10^{-37}$
$f_{10}$	$3.82 \times 10^{-4}$	$3.76 \times 10^{-4}$	$5.30 \times 10^{-3}$	$2.39 \times 10^{-3}$	$3.78 \times 10^{-3}$

从表2可以看出, $l=10$ 时,算法在5个函数上取得最优值; $l=50$ 时,算法在9个函数上表现最好;此后,随着 $l$ 取值增大,算法性能逐渐变差。可能的原因是, $l$ 取值较大时,调用侦察蜂的频率较小,算法不能及时放弃枯竭的食物源,导致其占用较多的计算资源,降低搜索效率;当 $l$ 取值过小时,会过早地放弃一些有前景的食物源。虽然式(11)和(12)会兼顾部分搜索经验,但仍会导致搜索经验丢失。同时,也说明这种搜索模式只适合作为补充搜索,这与文献[28-29]中以0.1的概率选择同类型的搜索模式一致。综合考虑,本文设置 $l=50$ 。

### 3.3 策略有效性分析

ABC\_DEE算法采用了两种折中兼顾策略(即跟随蜂阶段兼顾适应度和排序的选择策略、侦察蜂阶段综合目标值和多样性最优解形式的搜索策略),以及广度优先策略和深度优先策略。为了验证各策略对ABC\_DEE的有效性,设计如下对比算法:

ABC\_DEE-1,跟随蜂阶段未采用折中策略的ABC\_DEE;

ABC\_DEE-2,侦察蜂阶段未采用折中策略的ABC\_DEE;

ABC\_DEE-3,未采用广度优先策略的ABC\_DEE;

ABC\_DEE-4,未采用深度优先策略的ABC\_DEE;

实验设置与第3.2节保持一致,对比结果如表3所示。

表3 不同策略的平均误差对比

Tab. 3 Comparison of mean results for different strategies

函数	ABC_DEE-1	ABC_DEE-2	ABC_DEE-3	ABC_DEE-4	ABC_DEE
$f_1$	$1.17 \times 10^{-193}$	$1.20 \times 10^{-256}$	$7.59 \times 10^{-257}$	$1.58 \times 10^{-216}$	$1.48 \times 10^{-257}$
$f_2$	*0	$1.25 \times 10^{-1}$	*0	*0	*0
$f_3$	*0	$1.04 \times 10^1$	*0	*0	*0
$f_4$	*0	$1.56 \times 10^{-5}$	*0	*0	*0
$f_5$	$1.66 \times 10^{-120}$	$5.86 \times 10^{-2}$	$3.14 \times 10^{-187}$	$2.90 \times 10^{-160}$	$2.80 \times 10^{-193}$
$f_6$	$1.70 \times 10^{-3}$	$1.37 \times 10^{-1}$	$2.97 \times 10^{-3}$	$2.52 \times 10^{-3}$	$3.77 \times 10^{-4}$
$f_7$	$1.36 \times 10^{-5}$	$1.20 \times 10^{-1}$	$4.33 \times 10^{-7}$	$3.65 \times 10^{-7}$	$3.27 \times 10^{-7}$
$f_8$	*0	*0	*0	*0	*0
$f_9$	$5.44 \times 10^{-216}$	$1.21 \times 10^{-14}$	$2.89 \times 10^{-224}$	$2.36 \times 10^{-192}$	$9.42 \times 10^{-226}$
$f_{10}$	$6.41 \times 10^0$	$1.64 \times 10^1$	$4.82 \times 10^0$	$3.22 \times 10^0$	$3.76 \times 10^{-4}$

从表3可以看出,所采用的4种策略均能提高算法的优化性能。具体的,ABC\_DEE在函数 $f_1$ 、 $f_5$ 、 $f_6$ 、 $f_7$ 、 $f_9$ 、 $f_{10}$ 上的均值优于ABC\_DEE-1,在剩余4个函数上二者均找到了理论最优值。可能的原因是:对于

食物源差异较大或较小的情况,排序选择能够稳定地选出优质食物源;对于其他情况,适应度选择能够较好地反映出食物源之间的真实差异;兼顾二者之后,ABC\_DEE较ABC\_DEE-1能更好地应对多样化的搜索环境。ABC\_DEE在9个函数上优于ABC\_DEE-2,侦察蜂阶段折中策略的改进效果最明显,同时,也说明基本ABC中侦察蜂的搜索方式有较大的改进空间。基本ABC中的侦察蜂采用随机初始化的方式跳出局部最优解,导致前期搜索经验丢失。ABC\_DEE通过式(11)和(12)择优跳出局部最优,该方法同时考虑了基于目标值的最优解、基于多样性的最优解和前期搜索经验。与ABC\_DEE-3和ABC\_DEE-4相比,ABC\_DEE在6个函数上找到了更好的解,说明广度优先策略和深度优先策略在一定程度上分别提高了雇佣蜂的探索能力和跟随蜂的开采能力。

### 3.4 与其他ABC算法对比

为了进一步测试ABC\_DEE的性能,选择近年来提出的国内外有代表性的ABC<sup>[1]</sup>、GABC<sup>[9]</sup>、基于排序选择和精英引导的人工蜂群算法(REABC)<sup>[10]</sup>、基于精英解和随机个体邻域信息的人工蜂群算法(ENABC)<sup>[11]</sup>、NSABC<sup>[12]</sup>和RNSABC<sup>[13]</sup>等6种ABC类算法作为对比算法。为确保比较公平,按照惯例,对比算法的参数设置与原文献保持一致,种群规模设置相同(本文设置 $N=50$ )。按照CEC2021的建议,测试函数的维度 $D=10$ 和 $20$ ,最大函数评价次数 $E_{\max}=200\ 000$ 和 $1\ 000\ 000$ ,所有算法独立运行30次。对比实验从均值与标准差、非参数检验、收敛速度和时间效率4个方面展开。

表4为 $D=10$ 时7种算法的对比结果,包括基于平均值和标准差的解质量对比与基于非参数检验的综合对比(详细结果见附录A表A1)。在进行解质量对比时, $49^+/5^-/26^-$ 表示ABC\_DEE算法分别在49、5、26个函数上的寻优结果优于、相当于、劣于对比算法ABC,其他同。由表4可知,ABC\_DEE分别在49、46、51、61、52、48个函数上优于ABC、GABC、REABC、ENABC、NSABC和RNSABC,在26、26、24、12、22、23个函数上劣于这些算法。为确保对比具有统计意义,采用Friedman和Wilcoxon两种非参数检验方法。Friedman检验以平均排名衡量算法的整体性能,排名值越小算法性能越好。表4的平均排名表明,ABC\_DEE在所有算法中排名第1。Wilcoxon检验ABC\_DEE与其他算法是否存在显著差异,显著性水平设为0.05。检验结果以 $R^+$ 、 $R^-$ 和 $P$ 值的形式给出,其中, $R^+$ 代表Wilcoxon检验中ABC\_DEE在优势函数上

的秩和, $R^-$ 代表在劣势函数上的秩和, $P$ 值小于0.05表明算法结果有显著性差异。表4的Wilcoxon检验结果表明,ABC\_DEE在所有对比情况下获得的 $R^+$ 均大于 $R^-$ ,且除GABC以外的 $P$ 值均小于0.05,说明本文所提算法显著优于除GABC以外的其他对比算法,与GABC无显著差异。

表4  $D=10$ 时各算法的统计检验结果对比

Tab. Comparison of statistical test results among different algorithms ( $D=10$ )

算法	解质量对比	Friedman 检验 平均排名	Wilcoxon 检验		
			$R^+$	$R^-$	$P$ 值
ABC	$49^+/5^-/26^-$	4.61	2 000.0	1 240.0	$4.48 \times 10^{-2}$
GABC	$46^+/8^-/26^-$	3.85	1 945.0	1 295.0	$6.82 \times 10^{-2}$
REABC	$51^+/5^-/24^-$	4.84	2 206.0	1 034.0	$1.97 \times 10^{-3}$
ENABC	$61^+/7^-/12^-$	4.84	2 593.5	646.5	$8.70 \times 10^{-8}$
NSABC	$52^+/6^-/22^-$	3.18	2 152.5	1 087.5	$4.12 \times 10^{-3}$
RNSABC	$48^+/9^-/23^-$	3.66	2 110.0	1 130.0	$4.99 \times 10^{-3}$
ABC_DEE		3.01			

表5为 $D=20$ 时7种算法的对比结果(详细结果见附录A表A2)。可以发现,函数维度增加以后,对ABC\_DEE与其他算法的性能对比影响不大。在解质量对比方面,ABC\_DEE至少在47个函数优于对比算法,至多在26个函数上差于对比算法。在综合对比方面,Friedman检验时所有算法的平均排名情况为ABC\_DEE<NSABC<GABC<RNSABC<ENABC<ABC<REABC,即ABC\_DEE的整体性能排名第一。Wilcoxon检验结果表明,ABC\_DEE对比6个算法所得 $R^+$ 均大于 $R^-$ 。根据 $P$ 值,ABC\_DEE与GABC无显著差异,但显著优于其他算法。附录A图A1进一步描绘了所有算法在 $D=10$ 和 $D=20$ 下3个函数( $f_7$ 、 $f_{21}$ 、 $f_{63}$ )上的收敛曲线。

表5  $D=20$ 时各算法的统计检验结果对比

Tab. 5 Comparison of statistical test results among different algorithms ( $D=20$ )

算法	解质量对比	Friedman 检验 平均排名	Wilcoxon 检验		
			$R^+$	$R^-$	$P$ 值
ABC	53/3/24	4.70	2 046.5	1 119.5	$3.03 \times 10^{-2}$
GABC	47/7/26	3.52	1 915.5	1 324.5	$1.04 \times 10^{-1}$
REABC	55/4/21	4.77	2 289.0	951.0	$5.33 \times 10^{-4}$
ENABC	55/7/18	4.60	2 355.0	885.0	$3.71 \times 10^{-5}$
NSABC	54/4/22	3.44	2 163.0	1 077.0	$4.14 \times 10^{-3}$
RNSABC	50/9/21	3.85	2 180.0	1 060.0	$1.33 \times 10^{-3}$
ABC_DEE		3.12			

最后,对比各算法的时间效率。考虑到同一算法在不同函数上的求解精度不同,不同算法在同一函数上的求解精度也不尽相同,在比较各算法的时间效率时,对不同的函数设置不同的求解精度。具体地,将所有算法在某个函数上取得的最差解设置为对应的求解精度目标,这样可以保证所有算法都能完成任务。表6为7种算法的平均计算时间,可以看出,本文算法所消耗的计算时间最短(详细结果见附录表A3、A4)。

表6 各算法的平均计算时间对比

Tab. 6 Comparison of average runtime consumed by different algorithms on all functions

算法	平均时间	
	$D=10$	$D=20$
ABC	$4.01 \times 10^{-1}$	2.83
GABC	$3.98 \times 10^{-1}$	2.80
REABC	$4.76 \times 10^{-1}$	3.10
ENABC	1.11	6.50
NSABC	$6.01 \times 10^{-1}$	3.76
RNSABC	$9.62 \times 10^{-1}$	5.73
ABC_DEE	$3.92 \times 10^{-1}$	2.61

## 4 实际应用

为进一步评估ABC\_DEE算法在解决实际优化问题中的表现,将其用于食管癌预测模型优化,并与其他算法优化的模型做比较。

### 4.1 数据来源

实验样本来自郑州大学省部共建食管癌防治国家重点实验室,其中,疾病组为60例未接受放疗和化疗治疗的食管癌患者,对照组为60例无任何肿瘤相关证据的健康者。每个样本包含15 102个有效蛋白质数据,通过T检验、富集分析、专家评价等方法筛选出81个关键蛋白作为食管癌标志物。

### 4.2 ABC\_DEE-KELM预测模型

核极限学习机(KELM)是一种单隐层前馈神经网络算法,具有训练速度快、泛化能力强等优点,已被成功用于回归和分类问题<sup>[30]</sup>。KELM的性能主要受正则化系数 $C$ 和核函数参数 $g$ 影响,前者控制模型的复杂度,后者决定核函数的作用范围。因此,利用智能算法优化参数 $C$ 和 $g$ ,是提高KELM性能的一种常用方法。本节选择通用性强的高斯核函数,建立基于ABC\_DEE优化KELM(ABC\_DEE-KELM)的食管癌预测模型,模型的输入为81个食管癌标志物,输出为患病或正常,目标函数 $f(C, g)$ 为预测准确率:

$$f(C, g) = \frac{S_{TP} + S_{TN}}{S_{TP} + S_{TN} + S_{FP} + S_{FN}} \quad (13)$$

式中, $S_{TP}$ 是真阳性的数量, $S_{TN}$ 是真阴性的数量, $S_{FP}$ 是假阳性的数量, $S_{FN}$ 是假阴性的数量。

ABC\_DEE-KELM预测模型流程如图3所示。

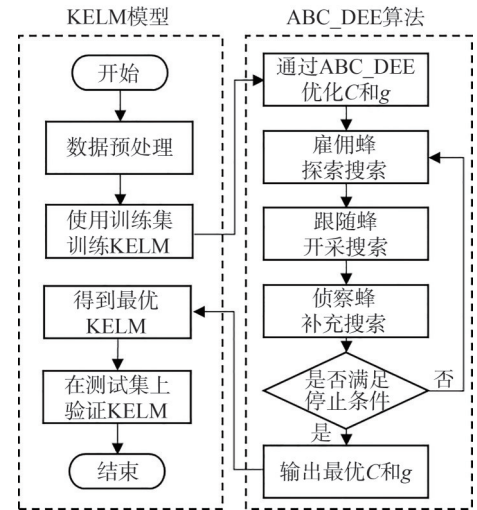


图3 ABC\_DEE-KELM模型流程

Fig. 3 Flowchart of ABC\_DEE-KELM model

ABC\_DEE-KELM的主要步骤如下。

步骤1:对蛋白质数据进行归一化处理;

步骤2:将数据集划分为训练集和测试集;

步骤3:初始化ABC\_DEE算法参数,包括 $N$ 、 $p$ 、 $l$ 和 $E_{\max}$ ;

步骤4:将参数 $C$ 和 $g$ 作为蜜蜂食物源的两个维度,设置搜索空间范围;

步骤5:以预测准确率为适应度函数,令ABC\_DEE在训练集上对KELM中的参数 $C$ 和 $g$ 进行寻优;

步骤6:利用最优参数 $C$ 和 $g$ 构建预测模型,在测试集上验证训练好的KELM,输出预测结果。

### 4.3 实验结果与分析

为了评估ABC\_DEE-KELM模型的预测性能,将其与ABC<sup>[1]</sup>、GABC<sup>[9]</sup>、REABC<sup>[10]</sup>、ENABC<sup>[11]</sup>、NSABC<sup>[12]</sup>和RNSABC<sup>[13]</sup>等6种算法优化的KELM模型作比较。所有算法参数与第3节保持一致,函数最大评价次数 $E_{\max}$ 设置为100 000。采用10折交叉验证进行实验。除了准确率,医学诊断领域还经常使用敏感性、特异性、F1分数等来评价预测效果。敏感性(计算式为 $S_{TP}/(S_{TP} + S_{FN})$ )代表模型从患者中准确分类出患者的能力。特异性(计算式为 $S_{TN}/(S_{TN} + S_{FP})$ )代表模型从健康者中准确分类出健康者的能力。F1分数(计算式为 $2S_{TP}/(2S_{TP} + S_{FP} + S_{FN})$ )是精确率和敏感性的调和平均值,可以平衡模型的准确性和识别能力。

图4为7种模型进行10折交叉验证后的平均预测

结果。由图4可以看出,ABC\_DEE-KELM的预测结果在所有模型中排名第1。在准确率方面,ABC\_DEE-KELM达到85.83%,比排名第2的RNSABC-KELM和ABC-KELM分别提高了3.00%和15.72%。在敏感性方面,ABC\_DEE-KELM达到91.98%,比排名第2的ENABC-KELM和ABC-KELM分别提高了0.61%和20.14%。

在特异性方面,ABC\_DEE-KELM达到79.31%,比排名第2的RNSABC-KELM和ABC-KELM分别提高了0.70%和15.03%。在F1分数方面,ABC\_DEE-KELM达到85.49%,比排名第2的RNSABC-KELM和ABC-KELM分别提高了2.54%和16.23%。附录A图A2进一步描绘了7种模型预测结果的混淆矩阵。

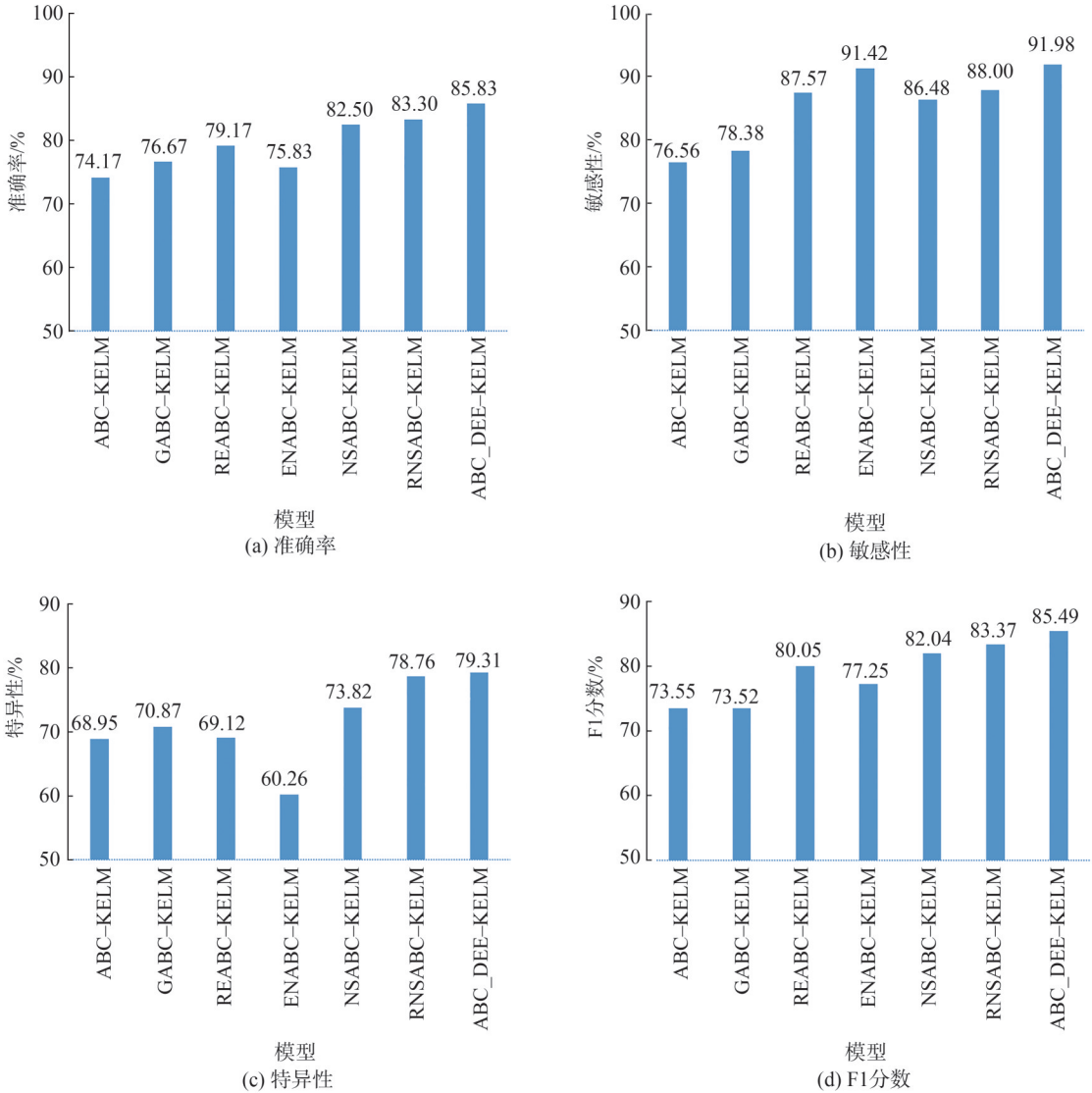


图 4 7种结合ABC和KELM模型的平均预测结果  
 Fig. 4 Average predictive results of seven models combining ABC and KELM

## 5 结论

经典ABC算法存在探索与开采不平衡的问题,调控二者之间的关系是提高ABC算法性能的一种有效方法。本文在不改变ABC算法框架的前提下,提出一种基于探采分工的ABC算法。在“基向量+扰动”的搜索模式下,以多样性和目标值双精英引导的方式强化雇佣蜂和跟随蜂之间关于探索与开采的分工。同时,为侦察蜂设计了兼顾搜索经验与多元化搜索的解搜索方程。在80个基准函数上与6种ABC及改进算

法进行对比,从均值与标准差、非参数检验、收敛速度和时间效率4个方面验证了所提算法的优越性,并以食管癌预测为例检验了其在实际优化问题上的有效性。下一步工作可从两个方面展开:一是利用本文算法求解路径规划、图像处理等更多实际问题,二是采用邻域拓扑、惯性权重等策略调节ABC算法中探索与开采的关系。

附录见本刊网络版,扫描标题旁的二维码可阅读网络全文。

## 参考文献:

- [1] Akay B, Karaboga D, Gorkemli B, et al. A survey on the Artificial Bee Colony algorithm variants for binary, integer and mixed integer programming problems[J]. *Applied Soft Computing*, 2021, 106: 107351.
- [2] Shu Jian, Li Ruirui, Xiong Tao, et al. Link prediction based on learning automaton and firefly algorithm[J]. *Advanced Engineering Sciences*, 2021, 53(2): 133–140. [舒坚, 李睿瑞, 熊涛, 等. 基于学习自动机与萤火虫算法的链路预测[J]. *工程科学与技术*, 2021, 53(2): 133–140.]
- [3] Wang Yingcong, Sui Chengcheng, Liu Chi, et al. Chicken swarm optimization with an enhanced exploration–exploitation tradeoff and its application[J]. *Soft Computing*, 2023, 27(12): 8013–8028.
- [4] Yu Xiuwu, Qin Xiaokun, Liu Yong, et al. DV–Hop localization algorithm optimized based on global artificial fish swarm algorithm[J]. *Advanced Engineering Sciences*, 2022, 54(4): 228–234. [余修武, 秦晓坤, 刘永, 等. 基于全局人工鱼群算法优化的 DV–Hop 定位算法[J]. *工程科学与技术*, 2022, 54(4): 228–234.]
- [5] Liu Zhen, Lu Huajie, Ren Jiancun. Co-evolutionary gravitational krill herd algorithm[J]. *Advanced Engineering Sciences*, 2018, 50(6): 217–224. [刘振, 鲁华杰, 任建存. 协同进化引力磷虾觅食算法[J]. *工程科学与技术*, 2018, 50(6): 217–224.]
- [6] Cui Yibing, Hu Wei, Rahmani A. A reinforcement learning based artificial bee colony algorithm with application in robot path planning[J]. *Expert Systems with Applications*, 2022, 203: 117389.
- [7] Jawad F K J, Ozturk C, Wang Dansheng, et al. Sizing and layout optimization of truss structures with artificial bee colony algorithm[J]. *Structures*, 2021, 30: 546–559.
- [8] Bajer D, Zorić B. An effective refined artificial bee colony algorithm for numerical optimisation[J]. *Information Sciences*, 2019, 504: 221–275.
- [9] Zhu Guopu, Kwong S. Gbest-guided artificial bee colony algorithm for numerical function optimization[J]. *Applied Mathematics and Computation*, 2010, 217(7): 3166–3173.
- [10] Kong Depeng, Chang Tianqing, Dai Wenjun, et al. An improved artificial bee colony algorithm based on the ranking selection and the elite guidance[J]. *Control and Decision*, 2019, 34(4): 781–786. [孔德鹏, 常天庆, 戴文君, 等. 基于排序选择和精英引导的改进人工蜂群算法[J]. *控制与决策*, 2019, 34(4): 781–786.]
- [11] Meng Hongyun, Wei Bingke. An improved artificial bee colony algorithm based on elite solution and random individual neighborhood information[J]. *Control and Decision*, 2020, 35(9): 2169–2174. [孟红云, 位冰可. 基于精英解和随
- 机个体邻域信息的改进人工蜂群算法[J]. *控制与决策*, 2020, 35(9): 2169–2174.]
- [12] Wang Hui, Wang Wenjun, Xiao Songyi, et al. Improving artificial bee colony algorithm using a new neighborhood selection mechanism[J]. *Information Sciences*, 2020, 527: 227–240.
- [13] Ye Tingyu, Wang Wenjun, Wang Hui, et al. Artificial bee colony algorithm with efficient search strategy based on random neighborhood structure[J]. *Knowledge-Based Systems*, 2022, 241: 108306.
- [14] Wang Zhigang, Wang Minggang. Multi-search strategy of artificial bee colony algorithm based on symbolic function[J]. *Control and Decision*, 2016, 31(11): 2037–2044. [王志刚, 王明刚. 基于符号函数的多搜索策略人工蜂群算法[J]. *控制与决策*, 2016, 31(11): 2037–2044.]
- [15] Wang Zhigang, Shang Xudong, Xia Huiming, et al. Artificial bee colony algorithm with multi-search strategy cooperative evolutionary[J]. *Control and Decision*, 2018, 33(2): 235–241. [王志刚, 尚旭东, 夏慧明, 等. 多搜索策略协同进化的人工蜂群算法[J]. *控制与决策*, 2018, 33(2): 235–241.]
- [16] Hakli H, Kiran M S. An improved artificial bee colony algorithm for balancing local and global search behaviors in continuous optimization[J]. *International Journal of Machine Learning and Cybernetics*, 2020, 11(9): 2051–2076.
- [17] Jadon S S, Tiwari R, Sharma H, et al. Hybrid artificial bee colony algorithm with differential evolution[J]. *Applied Soft Computing*, 2017, 58: 11–24.
- [18] Kuang Fangjun, Jin Zhong, Xu Weihong, et al. Hybridization algorithm of Tent chaos artificial bee colony and particle swarm optimization[J]. *Control and Decision*, 2015, 30(5): 839–847. [匡芳君, 金忠, 徐蔚鸿, 等. Tent 混沌人工蜂群与粒子群混合算法[J]. *控制与决策*, 2015, 30(5): 839–847.]
- [19] Du Zhenxin, Liu Guangzhong, Han Dezhi, et al. Artificial bee colony algorithm with global and unbiased search strategy[J]. *Acta Electronica Sinica*, 2018, 46(2): 308–314. [杜振鑫, 刘广钟, 韩德志, 等. 基于全局无偏搜索策略的精英人工蜂群算法[J]. *电子学报*, 2018, 46(2): 308–314.]
- [20] Song Xiaoyu, Zhao Ming, Yan Qifeng, et al. A high-efficiency adaptive artificial bee colony algorithm using two strategies for continuous optimization[J]. *Swarm and Evolutionary Computation*, 2019, 50: 100549.
- [21] Singh A, Deep K. Exploration–exploitation balance in Artificial Bee Colony algorithm: A critical analysis[J]. *Soft Computing*, 2019, 23(19): 9525–9536.
- [22] Črepinšek M, Liu S H, Mernik M. Exploration and exploitation in evolutionary algorithms[J]. *ACM Computing Surveys*, 2013, 45(3): 1–33.
- [23] Cheng Jianchao, Pan Zhibin, Liang Hao, et al. Differential

- evolution algorithm with fitness and diversity ranking-based mutation operator[J].Swarm and Evolutionary Computation,2021,61:100816.
- [24] Cui Laizhong,Li Genghui,Lin Qiuzhen,et al.A novel artificial bee colony algorithm with depth-first search framework and elite-guided search equation[J].Information Sciences,2016,367:1012–1044.
- [25] Peng Hu,Deng Changshou,Wu Zhijian.Best neighbor-guided artificial bee colony algorithm for continuous optimization problems[J].Soft Computing,2019,23(18):8723–8740.
- [26] Imanian N,Shiri M E,Moradi P.Velocity based artificial bee colony algorithm for high dimensional continuous optimization problems[J].Engineering Applications of Artificial Intelligence,2014,36:148–163.
- [27] Mohamed A,Hadi A,Mohamed A,et al.Problem definitions and evaluation criteria for the CEC 2021 special session and competition on single objective bound constrained numerical optimization[R].Singapore:Nanyang Technological University,2020.
- [28] Zhou Xinyu,Wang Hui,Wang Mingwen,et al.Enhancing the modified artificial bee colony algorithm with neighborhood search[J].Soft Computing,2017,21(10):2733–2743.
- [29] Zhou Xinyu,Lu Jiabin,Huang Junhong,et al.Enhancing artificial bee colony algorithm with multi-elite guidance[J].Information Sciences,2021,543:242–258.
- [30] Huang Guangbin,Wang Dian hui,Lan Yuan.Extreme learning machines: A survey[J].International Journal of Machine Learning and Cybernetics,2011,2(2):107–122.

## Artificial Bee Colony Algorithm Based on the Division Between Exploration and Exploitation and Its Application in Esophageal Cancer Prediction

WANG Yingcong, YAN Jun, SUN Junwei, WANG Yanfeng\*

(School of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China)

### Abstract:

**Objective** In the artificial bee colony (ABC) algorithm, employed bees search the entire search space while onlooker bees concentrate their efforts near high-quality food sources. From the perspective of exploration and exploitation, employed bees primarily handle exploration, whereas onlooker bees are responsible for exploitation. However, the basic ABC algorithm prioritizes exploration during the search process and performs poorly in exploitation, leading to slow convergence speed and low solution accuracy. Therefore, this study proposes an artificial bee colony algorithm based on the division between exploration and exploitation (called ABC\_DEE), which consists of three stages: employed bees for exploration, onlooker bees for exploitation, and scout bees for supplementation.

**Methods** In the exploration stage, excessive reliance on random solution information biased the process toward random search. Therefore, a solution-search equation guided by diverse elites was designed for employed bees, along with the introduction of a breadth-first search strategy to enhance exploration. In the exploitation stage, overutilization of optimal solution information led to premature convergence. Hence, a solution-search equation guided by objective-oriented elites was designed for onlooker bees, using a depth-first search strategy to strengthen exploitation. Considering that the random initialization method of scout bees discarded the previous search experience, and the search equation of employed and onlooker bees was singular, a neighborhood search equation was designed for scout bees. This equation considered the optimal solution based on objective value, the optimal solution based on diversity, and the previous search experience.

**Results and Discussions** The performance of ABC\_DEE was assessed on both the CEC2021 test set and the esophageal cancer prediction problem, with comparisons made against six ABC variants. In numerical optimization problems with  $D=10$ , ABC\_DEE outperformed ABC, GABC, REABC, ENABC, NSABC, and RNSABC on 49, 46, 51, 61, 52, and 48 functions, respectively. In contrast, ABC\_DEE performed worse than these algorithms on 26, 26, 24, 12, 22, and 23 functions, respectively. The Friedman test results indicated that ABC\_DEE ranked first. The Wilcoxon test results indicated that there was no significant difference between ABC\_DEE and GABC, but ABC\_DEE significantly outperformed the other algorithms. When  $D=20$ , ABC\_DEE was superior to the comparison algorithms on at least 47 functions and inferior on at most 26 functions. The results of both the Friedman test and the Wilcoxon test were consistent with those when  $D=10$ . In addition, ABC\_DEE exhibited superiority in terms of convergence speed and time efficiency. Regarding the optimization problem of the esophageal cancer prediction model based on Kernel Extreme Learning Machine (KELM), in terms of accuracy, ABC\_DEE-KELM achieved 85.83%, which was 3.00% higher than the second-ranked RNSABC-KELM and 15.72% higher than ABC-KELM. Regarding sensitivity, ABC\_DEE-KELM reached 91.98%, outperforming the second-ranked ENABC-KELM by 0.61% and ABC-KELM by 20.14%. In terms of specificity, ABC\_DEE-KELM attained 79.31%, exceeding the second-ranked RNSABC-KELM by 0.70% and ABC-KELM by 15.03%. Regarding the F1 score, ABC\_DEE-KELM achieved 85.49%, showing a lead of 2.54% over the second-ranked RNSABC-KELM and 16.23% over ABC-KELM.

**Conclusions** The classical ABC suffers from an imbalance between exploration and exploitation. Regulating the relationship between exploration

and exploitation is an effective method for improving the performance of ABC. This study proposes a novel ABC based on the separation of exploration and exploitation without altering the original ABC framework. Under the “base vector + perturbation” search mode, the proposed approach strengthens the division of labor between employed bees and onlooker bees in terms of exploration and exploitation through a dual-elite-guided strategy based on diversity and objective values. Simultaneously, a search equation is formulated for scout bees that incorporates both search experience and diversified optimal solution information. The proposed algorithm is compared to six other ABC algorithms across 80 benchmark functions, with its superiority evaluated in terms of solution quality, non-parametric tests, convergence speed, and time efficiency. In addition, its effectiveness in practical optimization problems is validated using esophageal cancer prediction as an example. Future research can be approached from two perspectives. First, the proposed algorithm will be applied to address more practical problems such as path planning and image processing. Second, by exploring strategies such as neighborhood topology and inertia weights to fine-tune the balance between exploration and exploitation in ABC.

**Key words:** artificial bee colony algorithm; exploration and exploitation; elite guidance; search equation

(编辑 吴芝明)

引用格式: Wang Yingcong, Yan Jun, Sun Junwei, et al. Artificial bee colony algorithm based on the division between exploration and exploitation and its application in esophageal cancer prediction[J]. *Advanced Engineering Sciences*, 2025, 57(4): 112–122. [王英聪, 严军, 孙军伟, 等. 基于探采分工的人工蜂群算法及其在食管癌预测中的应用[J]. *工程科学与技术*, 2025, 57(4): 112–122.]