

文章编号:1004-9037(2013)06-0000-09

增强寻优能力的改进蜂群算法

易正俊 韩晓晶

(重庆大学数学与统计学院,重庆,401331)

摘要:针对人工蜂群算法易陷入局部最优问题,首先将蜂群算法中的跟随蜂数量翻倍,用其一半采用轮盘赌选择机制更新,保持蜂群沿着蜜源浓度大的方向进化;增加的一半跟随蜂采用反向的轮盘赌选择机制,用以维持种群的多样性。其次,将所有未更新计数器次数大于阈值的蜜源对应的引领蜂变成侦察蜂,并对相应蜜源进行更新搜索。最后,求出每一轮迭代后所得蜜源的中心位置,通过中心位置与每个蜜源所在的邻域内产生一新解,再比较适应度值的大小,选择优者。仿真实验证明,该改进算法具有更高的收敛精度和很好的鲁棒性,且增加了算法跳出局部最优的机会,增强了蜂群算法的寻优能力,具有更好的优化性能。

关键词:人工蜂群算法;轮盘赌;适应度值;中心位置

中图分类号:TP18

文献标志码:A

Modified Artificial Bee Colony Algorithm to Enhance Ability of Search

Yi Zhengjun, Han Xiaojing

(College of Mathematics and Statistics, Chongqing University, Chongqing, 401331, China)

Abstract: Considering the problem which the artificial bee colony is inclined to fall into local optimum. Firstly, the onlooker bees' number is doubled, the roulette wheel selection mechanism is adopted to update with its half to keep evolution along the high concentration of nectar sources; and the roulette wheel selection mechanism of the reverse with others is used to maintain population diversity. Secondly, employed bees is transformed corresponding to all the nectar sources which the times of counter without updating is greater than the threshold value into scout bees, then are these nectar sources updated. Finally, the center position of nectar sources is calculated which are gotten after each iteration, a new solution is produced in the neighborhood that is formed by the center position and every nectar source, then compared the values of fitness to choose optimum. Experiments show that the improved colony algorithm not only has better convergence precision and good robustness, but also increases the chance of the algorithm to jump out of local optimum, enhances the ability of search, and has a better optimization performance.

Key words: artificial bee colony algorithm; roulette wheel; fitness values; center position

引 言

Karaboga 于 2005 年提出的人工蜂群(Artificial bee colony, ABC)算法^[1-3],是一种模拟蜜蜂采蜜行为的群体智能优化算法。该算法通过蜂群各自的分工不同并不断交换蜜源信息寻找优化问题

的最优解,在机器人路径规划^[4]、人工神经网络训练^[5-6]、滤波器设计^[7]、网络优化^[8]、生产调度^[9]等领域得到了成功的应用。与差分进化算法^[10]、遗传算法^[11-15]、粒子群算法^[16-18]等相比,蜂群算法具有控制参数少、计算简单、易于实现等优点^[2]。但传统的蜂群算法在搜索更新时因采用轮盘赌的选择策略,导致群体多样性下降,算法过早收敛,使得

算法存在易陷入局部最优的缺陷。针对该算法存在的不足,有许多学者对其进行了不同的改进,如胡珂等人^[19]采用外推技巧设置一个搜索更新公式,将微调机制引入其中,构造出一种具有引导趋势的蜂群算法;丁海军等人^[20]引用 Boltzmann 选择策略代替蜂群算法中的轮盘赌选择策略来防止算法过早收敛;向万里等人^[21]在跟随蜂选择蜜源阶段,采用轮盘赌的反向选择机制,使其进化不会快速向适应度值高的蜜源处集中以保持种群的多样性,这些改进在一定程度上提高了获得最优解的概率。

本文为了进一步提高获得最优解的概率,使算法具有更好的收敛精度,在原始蜂群算法的选择机制上进行三方面的改进:首先,在传统蜂群算法 ABC 算法的选择机制上增加一个反向轮盘赌的选择机制,将蜂群算法中的跟随蜂数量翻倍。用其一半采用原始的轮盘赌选择机制,保持传统的蜂群算法的进化方向,不断地向最优解逼近;另一半跟随蜂则采用反向的轮盘赌选择机制,来维持种群的多样性,避免算法早熟收敛。其次,考虑到侦察蜂具有跳出局部最优的作用,在迭代过程中将所有未更新计数器次数大于阈值的蜜源对应的引领蜂变成侦察蜂,使侦察蜂的个数进行自适应动态调整,来增加蜂群算法有效跳出局部最优解的机会,避免陷入局部最优。最后,为提高算法的进化能力,在每一轮迭代结束后,对得到的所有解求中心位置,在中心位置与每个现有解之间产生一新解,把新解与现有解比较,选取较优者。通过以上三方面的改进,增加了算法跳出局部最优的机会,增强了蜂群算法的寻优能力,使算法能够获得极高的收敛精度,并很好地避免了算法出现早熟停滞的现象。

1 传统的人工蜂群算法原理

蜂群算法中的采蜜蜂包括引领蜂、跟随蜂和侦察蜂 3 种。引领蜂引导算法方向;跟随蜂加快算法的收敛速度;侦察蜂使算法能有效跳出局部最优解。蜂群算法用于求解多元函数的优化问题时,可行解的位置对应着蜂群算法的蜜源位置,函数值就对应蜂群算法的花蜜数量(适应度值),寻找最优解就是寻求花蜜数量最多的位置。

考虑全局优化问题

$$\min f(X), \text{ s. t. } X \in D \subset \mathbf{R}^n \quad (1)$$

设 $X_i = (x_{i1}, x_{i2}, \dots, x_{in}) \in D$ 是优化问题的一个

可行解,相当于蜂群算法的一个蜜源位置,其适应度值为 f_{X_i} 。设定初始解、引领蜂和跟随蜂的个数均为 N ,则蜂群算法首先生成含有 N 个蜜源(解) $X_i (1 \leq i \leq N)$ 的初始种群,每个蜜源与一个引领蜂对应。将引领蜂按照式(2)对蜜源位置进行更新

$$v_{ij} = x_{ij} + r_{ij}(x_{ij} - x_{kj}) \quad k \neq i \quad (2)$$

式中: $x_{ij} (1 \leq i \leq N, 1 \leq j \leq n)$ 为蜜源 X_i 的第 j 个分量位置, $v_{ij} (1 \leq i \leq N, 1 \leq j \leq n)$ 为第 i 个引领蜂在 x_{ij} 基础上产生新蜜源 $V_i = (v_{i1}, v_{i2}, \dots, v_{in}) \in D$ 的第 j 个分量位置, r_{ij} 是 $[-1, 1]$ 上的一个随机数。

设 Bas_i 为第 i 个解 X_i 未更新计数器,初始值设置为 0,若新解 V_i 的适应度值 f_{V_i} 大于 X_i 的适应度值 f_{X_i} ,则用 V_i 代替 X_i ,相应引领蜂更新到该蜜源处;否则,解 X_i 仍保持不变, Bas_i 加 1。当所有引领蜂都搜索完后,回到舞蹈区把适应度值信息通过跳摇摆舞传达给跟随蜂。

跟随蜂是采用轮盘赌的选择更新策略,在选择更新时需要计算当前 N 个引领蜂所在的蜜源 $X_i (1 \leq i \leq N)$ 各自的信息概率值 p_i ,即

$$p_i = \frac{f_i}{\sum_{i=1}^N f_i} \quad (3)$$

每个跟随蜂根据所得信息依概率按轮盘赌的选择方式选择蜜源,并对其按照式(2)进行更新。

随后,判断是否出现侦察蜂:设置阈值 limit ,选取大于 limit 的未更新计数器 Bas_j 中最大者 $\text{Bas}_i = \max(\text{Bas}_j)$,将其对应的引领蜂变成侦察蜂,随机搜索新蜜源 $V_i = (v_{i1}, v_{i2}, \dots, v_{in})$,设 x_{\max}^j 和 x_{\min}^j 分别表示第 i 个解 X_i 的第 j 个分量所在的上、下界,则 v_{ij} 采用式(4)确定

$$v_{ij} = x_{\min}^j + \text{rand}(0, 1)(x_{\max}^j - x_{\min}^j) \quad (4)$$

将 V_i 和 X_i 的适应度值进行比较,从中选择较优者。按照上述方式反复循环迭代,设置循环迭代的次数为 $\alpha (\alpha = 1, 2, \dots, \max_{\text{cycle}})$,直到达到最大循环迭代次数 \max_{cycle} 为止。

2 人工蜂群算法的改进

传统的蜂群算法有两个原因导致算法容易陷入局部最优。(1) 跟随蜂选择蜜源时采用轮盘赌选择机制。由式(3)可以看出:轮盘赌是依据适应度值占总适应度值的比例 P_i 来作为选择更新的机制,该比例值 P_i 越大,相对应的蜜源被选中更新的机会就越大,导致跟随蜂沿着适应度值高的蜜源搜索,是一种基于贪婪的选择方式。但实际上经过若干次迭代循环后,当前较差的解(即适应度值低的

蜜源)同样包含着有用的信息,在其邻域范围内可能存在全局最优解。采用轮盘赌策略可能会将其忽略,降低了种群的多样性,减少了获得全局最优解的可能性,容易陷入局部最优。(2)每一轮迭代结束后,将大于 limit 的未更新计数器 Bas_j 的最大者所对应的引领蜂变成侦察蜂,侦察蜂只有一只,按照式(4)搜索新蜜源,再比较适应度值的大小,选择较优者。此时,其他大于 limit 的未更新计数器 Bas_j 对应的蜜源有可能陷入局部最优,延缓蜂群算法的寻优进度,减少了跳出局部最优的机会,这同样使得算法容易陷入局部最优。为增加算法获得全局最优解的可能性,本文对蜂群算法作出以下三方面的改进。

2.1 双向轮盘赌的选择更新机制

在不影响蜂群算法原来进化方向的同时,为增加算法获得全局最优解的可能性,将跟随蜂的数量扩增为初始解个数(N)的两倍 $2N$ 。前 N 个跟随蜂仍依概率公式(3)采用原始的轮盘赌选择机制来选择要更新的解,使其沿着适应度值高的蜜源搜索。后 N 个跟随蜂采用轮盘赌反向选择机制按照式(5)计算出的概率 P_i 来选择要更新的解。

$$P_i = \frac{1/f_i}{\sum_{i=1}^N (1/f_i)} \quad (5)$$

从式(5)和选择更新的机制可以看出:适应度值 f_i 越小,适应度值倒数 $1/f_i$ 的值就越大,进而 P_i 越大,相应的蜜源被选中更新的可能性就越大,即适应度值越低的蜜源被选择更新的概率就越大。这样避免了较差蜜源存在的有用信息被丢失,保持了种群的多样性,增加了算法获得全局最优解的可能性,避免算法过早收敛。

通过将蜂群算法沿着两个方向进化,提高了算法的寻优进度,降低了陷入局部最优的可能性,进一步增强了算法的寻优能力。

2.2 侦察蜂在迭代过程中的确定

在传统的蜂群算法中,将大于预设阈值(limit)的未更新计数器 Bas_j 中最大者所对应的引领蜂变成侦察蜂,侦察蜂只有一只。但此时,其他大于 limit 的未更新计数器 Bas_j 对应的蜜源有可能陷入局部最优,延缓蜂群算法的寻优进度,减少了跳出局部最优的机会。因此本文的改进方法是将所有 $\text{Bas}_i > \text{limit}$ 的相应引领蜂全部转为侦察蜂,由式(4)随机产生新蜜源,然后选择较优者。这样在每一轮的迭代过程中自适应动态地调整侦察蜂数

量,增加了算法跳出局部最优的机会,避免算法早熟收敛,同样增强了获得全局最优解的可能性。

2.3 新增迭代中心解邻域的更新搜索

算法在每次迭代结束后,获得当前的 N 个新解 X_1, X_2, \dots, X_N , 其中心解 X_0 比当前 N 个解 X_1, X_2, \dots, X_N 的任何一个解包含更多有关最优解的信息,在这个中心解 X_0 与当前每个解 $X_i (1 \leq i \leq N)$ 所在的邻域内进行一次搜索得到新解 Y_i 。其中 $X_i = (x_{i1}, x_{i2}, \dots, x_{im})$, $X_0 = (x_{01}, x_{02}, \dots, x_{0n})$, $Y_i = (y_{i1}, y_{i2}, \dots, y_{im})$ 则中心解的求解如下

$$x_{0j} = \frac{1}{N} \sum_{i=1}^N x_{ij}, \quad 1 \leq j \leq n \quad (6)$$

再利用式(7)求得 $X_i (1 \leq i \leq N)$ 所对应的新解 Y_i

$$y_{ij} = x_{0j} + \text{rand}(0,1)(x_{ij} - x_{0j}) \quad j \in \{1, 2, \dots, n\} \quad (7)$$

将新解 Y_i 与解 X_i 的适应度值进行比较,从中选择较优者。这进一步增强了蜂群算法的寻优能力。

综合上面 3 方面的改进,形成本文增强寻优能力的改进蜂群算法(Modified artificial bee colony algorithm to enhance the ability of search, soa-ABC)。

随机产生 N 个蜜源(解),每个引领蜂在其对应的蜜源附近进行一次邻域搜索产生一新蜜源,通过其适应度值的比较对蜜源进行更新。所有引领蜂搜索完后,将更新解的适应度值信息传达给 $2N$ 个跟随蜂。前 N 个跟随蜂根据蜜源的花蜜数量采用原始的轮盘赌选择机制来选择引领蜂,并在相应的邻域进行更新,随后选出所有未更新计数器次数大于预设阈值的蜜源,将其对应的引领蜂变成侦察蜂,对蜜源进行更新搜索;对采用轮盘赌选择机制搜索出的 N 个蜜源(解),再利用后 N 个跟随蜂采用反向的轮盘赌选择机制来选择引领蜂,并对蜜源进行更新搜索,保持种群的多样性,这样就形成了一个双向轮盘赌的选择机制;当算法按照双向轮盘赌的选择机制迭代一轮结束后得到 N 个蜜源,对这 N 个蜜源求其中心位置,通过中心位置与每个蜜源所在的邻域内产生一新解,再比较适应度值的大小,选择较优者。按照上述方式反复循环迭代,直至达到终止条件为止,并输出当前最优解,较好地接近全局最优解。

2.4 soa-ABC 算法步骤

本文指出的 soa-ABC 算法步骤如下:

(1) 参数初始化。设置种群规模为 $3N$, 控制参数 limit, 最大循环迭代次数 $\text{max}_{\text{cycle}}$ 。

(2) 随机产生 N 个蜜源, 每个引领蜂对应一个蜜源, 并计算每个引领蜂所在蜜源的适应度值。

(3) 每个引领蜂按照式(2)更新蜜源位置, 比较更新前后蜜源的适应度值, 若新位置的适应度值更优, 则引领蜂更新到该位置; 否则, 引领蜂的位置不变, 未更新计数器加 1。

(4) 前 N 个跟随蜂根据蜜源的花蜜数量依概率式(3)采用原始的轮盘赌选择机制来选择引领蜂, 并根据式(2)按步骤(3)方式更新蜜源位置。记录当前最优解及最优值。

(5) 判断是否出现侦察蜂。若某个蜜源 X_i ($1 \leq i \leq N$) 的未更新计数器次数大于预设的阈值 (limit), 则相应引领蜂变成侦察蜂并根据式(4)随机搜索一个新蜜源 V_i 。将新蜜源 V_i 与蜜源 X_i 进行比较, 从中选取较优者。若存在多个要放弃的解, 则同上依次进行更新。

(6) 后 N 个跟随蜂根据蜜源的花蜜数量依概率式(5)采用反向的轮盘赌选择机制来选择引领蜂, 并根据式(2)按步骤(3)方式更新蜜源位置。记录当前最优解及最优值。

(7) 同步骤(5), 判断是否出现侦察蜂。

(8) 利用式(6)对当前较优的 N 个解 X_1, X_2, \dots, X_N 计算中心位置 X_0 。

(9) 根据式(7), 在中心与该 N 个解 X_1, X_2, \dots, X_N 所在的邻域范围内搜索产生 N 个新解 Y_1, Y_2, \dots, Y_N , 并将该 N 个新解的适应度值与先前 N 个解 X_1, X_2, \dots, X_N 的适应度值进行两两比较, 选取较优的 N 个解。

(10) 如果迭代次数小于最大迭代数 \max_{cycle} , 转到步骤(3); 否则, 迭代终止, 输出最优值。

2.5 soa-ABC 算法的计算量和时间复杂度分析

与传统的人工蜂群算法相比, 本文算法在选择方式、搜索方式上都有所变化, 有必要考虑蜂群规模对算法的计算量和时间复杂度的影响^[22], 设种群规模为 $3N$, 则引领蜂和跟随蜂的个数分别为 N 和 $2N$ 个。

2.5.1 soa-ABC 算法的计算量

算法的计算量通常由迭代次数与每次迭代所消耗的乘法和加法次数的乘积来衡量, 以一次迭代分析改进前后算法的计算量。

传统蜂群算法的计算量如下:

(1) 引领蜂、跟随蜂搜索共需计算 $4N$ 次加法, $2N$ 次乘法;

(2) 跟随蜂选择较优蜜源时采用轮盘赌选择方式, 此时累加适应值需计算 $N-1$ 次加法, 各适应值所占概率需计算 N 次乘法;

(3) 侦察蜂搜索需计算 2 次加法, 1 次乘法。

soa-ABC 算法的计算量如下:

(1) 引领蜂、跟随蜂搜索共需计算 $6N$ 次加法, $3N$ 次乘法;

(2) 跟随蜂选择较优蜜源时采用轮盘赌和反向轮盘赌两种选择方式, 此时累加适应值需计算 $2(N-1)$ 次加法, 各适应值所占概率需计算 $2N$ 次乘法;

(3) 自适应动态调整的侦察蜂搜索最多需计算 $4N$ 次加法, $2N$ 次乘法。

(4) 利用中心解邻域求新解的过程需 $3N-1$ 次加法, $N+1$ 次乘法。

通过上述分析表明 soa-ABC 算法的计算量和计算时间略有增加。

2.5.2 soa-ABC 算法的时间复杂度

soa-ABC 算法迭代一次的时间复杂度可计算如下:

(1) 引领蜂对蜜源进行邻域搜索操作的时间复杂度为 $o(N)$;

(2) 计算蜜源适应度值和每个蜜源被选择概率的时间复杂度均为 $o(N)$;

(3) 蜂群更新选择蜜源的最差时间复杂度为 $o(N)$;

(4) 跟随蜂根据轮盘赌选择法和反向轮盘赌选择法随机选择蜜源的最差时间复杂度均为 $o(N^2)$;

(5) 侦察蜂阶段蜜源被替换的最差时间复杂度为 $o(N)$;

(6) 新增迭代中心解邻域处更新搜索的中心解计算和蜜源更新的时间复杂度均为 $o(N)$ 。

由此可知, 将以上 6 点中所有的时间复杂度按照顺序依次相加即得算法迭代一次所需的最差时间复杂度为

$$o(N) + o(N) + o(N) + o(N) + o(N^2) + o(N^2) + o(N) + o(N) + o(N)$$

因为时间复杂度是取次数最高的且不带系数的一项, 所以上述复杂度可记为 $o(N^2)$, 则本文算法的最差时间复杂度为 $o(N^2)$ 。同理, 对传统的人工蜂群算法进行分析, 其时间复杂度为 $o(N^2)$ 。所以本文改进的人工蜂群算法虽然计算量和计算时间略有增加, 但它只是对蜜源 N 呈整数倍的线性方式增加, 并且它的时间复杂度与传统的人工蜂群

算法的时间复杂度相当,采用两种选择机制后,使算法朝着两个方向同时寻优,这样比单向寻优提高了寻优的进度,所以该改进算法在计算时间上没有太大的影响,并且改进的蜂群算法在解的寻优能力上有显著的提高。

3 实验仿真

用双向轮盘赌的选择机制替换传统蜂群算法的轮盘赌得到的蜂群算法记为(Artificial bee colony algorithm of two-way roulette, TABC);在 TABC 算法的基础上,将“大于预设阈值的未更新计数器 Bas_j 中最大者所对应的引领蜂变成侦察蜂”改为“将所有 $Bas_i > limit$ 的相应引领蜂全部转为侦察蜂”,得到的蜂群算法记为(Artificial Bee Colony algorithm of two-way roulette and changed scout bees, TSABC);为验证本文改进蜂群算法——soa-ABC 算法的寻优效果,将其与传统蜂群算法(ABC 算法)、TABC 算法、TSABC 算法进行比较,用上述 4 种算法分别对常用的 Sphere, Rosenbrock, Griewank, Rastrigin, Schwefel 和 Ackley 6 个标准测试函数做仿真实验。

(1) Sphere 函数为单峰值函数,是一个基本函数优化问题,其表达式为

$$f_1(x) = \sum_{i=1}^n x_i^2, \quad x_i \in [-100, 100] \quad (8)$$

在 $x_i = 0, (i = 1, 2, \dots, n)$ 时达到极小值 0。

(2) Rosenbrock 函数为单峰值、非凸、病态函数,是一个经典的复杂优化问题,在 $x_i = 1, (i = 1, 2, \dots, n)$ 时达到极小值 0。其表达式为

$$f_2(x) = \sum_{i=1}^n [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \\ x_i \in [-30, 30] \quad (9)$$

(3) Griewank 函数为多峰值函数,全局极小值 0 在 $x_i = 0 (i = 1, 2, \dots, n)$ 时达到,且有众多局部极小值点,变量之间有明显的相互作用,表达式为

$$f_3(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \\ x_i \in [-600, 600] \quad (10)$$

(4) Rastrigin 函数为多峰值函数,全局极小值 0 在 $x_i = 0 (i = 1, 2, \dots, n)$ 时达到,其表达式为

$$f_4(x) = \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i) + 10) \\ x_i \in [-5.12, 5.12] \quad (11)$$

(5) Schwefel 函数为多峰值函数,在 $x_i = 420.9687 (i = 1, 2, \dots, n)$ 时达到全局极小值

-418.9829n,表达式为

$$f_5(x) = - \sum_{i=1}^n (x_i \sin(\sqrt{|x_i|})), \quad x_i \in [-500, 500] \quad (12)$$

(6) Ackley 函数是一个多峰值函数,变量之间相互独立,有许多局部极小值,且寻找该函数全局最优值时容易陷入局部最优,在 $x_i = 0, (i = 1, 2, \dots, n)$ 时达到全局极小值 0。其表达式为

$$f_6(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e, \quad x_i \in [-32, 32] \quad (13)$$

实验仿真中参数具体设置为:种群大小取 90,其中,引领蜂和跟随蜂的个数分别为 30 和 60,测试函数都取 30 维,控制参数 limit 为 100,最大循环迭代次数 \max_{cycle} 取为 1000 次。4 种蜂群算法的仿真结果如表 1 所示;4 种算法对 6 个函数的进化过程曲线如图 1 所示。

表 1 为将 ABC, TABC, TSABC 和 soa-ABC 4 种算法独立实验 30 次,每次迭代 1000 轮后所得优化结果。理论值表示目标函数理论上取得的最小值;最大值和最小值分别为每种算法独立实验 30 次后,所得最优值的最差结果和最好结果;均值为算法实验 30 次的最优值结果取平均,可用来反映算法的优化效果;方差反映了算法的稳定性;方差越小,表示该算法的稳定性越好。

从表 1 中可以看出, TABC, TSABC 和 soa-ABC 三种算法的优化结果明显优于 ABC 算法,而 soa-ABC 算法又优于 TABC 和 TSABC 两种算法,并具有更好的稳定性;图 1 给出了该 4 种算法对上述 6 个测试函数实验一次的进化过程曲线。从图形中可以看出,在 Schwefel 函数中,4 种算法优化效果相当,而 soa-ABC 略显优势。除了该函数外,soa-ABC 算法明显优于其他 3 种算法,具有更高的收敛精度,而 TABC 和 TSABC 算法效果相当,且优于原始的 ABC 算法。

为进一步验证本文提出的增强寻优能力的改进蜂群算法(soa-ABC 算法)的有效性以及相对于传统蜂群优化算法的优越性,仍采用上述 6 种标准测试函数并在相同的参数设置条件下,将 soa-ABC 算法的实验结果与文献[21]中的基于轮盘赌反向选择机制的蜂群优化算法(Modified artificial bee colony based on reverse selection of roulette,

MABC)和蜂群算法^[23](Run artificial bee colony algorithm,runABC),将基本蜂群算法中的选择概

率公式改为适应值与最大适应值的比)的结果进行了对比分析。见表 2 及图 2。

表 1 6 个标准测试函数的仿真结果比较

函数	算法	理论值	最大值	最小值	均值	方差
Sphere	ABC	0	2.006 46e-11	3.900 08e-14	4.383 31e-12	2.173 47e-23
	TABC		9.969 54e-16	5.292 35e-16	7.405 63e-16	1.669 63e-32
	TSABC		7.570 43e-16	4.402 27e-16	6.053 03e-16	9.575 33e-33
	soa-ABC		7.371 86e-16	4.003 02e-16	5.482 05e-16	7.157 78e-33
Rosenbrock	ABC	0	8.034 16	3.296 14e-02	1.469 08	2.911 82
	TABC		2.224 56	6.484 1e-03	0.383 683	0.245 931
	TSABC		1.740 05	1.094 69e-02	0.396 433	0.222 099
	soa-ABC		1.478 22	9.354 2e-04	0.177 354	0.078 252 3
Griewank	ABC	0	1.358 42e-08	3.996 8e-15	4.604 2e-10	6.145 15e-18
	TABC		7.945 07e-09	1.110 22e-16	3.057 49e-10	2.113 68e-18
	TSABC		3.832 27e-12	0	1.278 75e-13	4.895 07e-25
	soa-ABC		9.269 25e-13	0	3.162 66e-14	2.860 42e-26
Rastrigin	ABC	0	0.101 507	5.343 28e-12	3.383 55e-03	3.434 52e-04
	TABC		1.875 83e-12	0	2.728 48e-13	1.683 33e-25
	TSABC		2.842 17e-13	0	9.284 43e-14	3.673 14e-27
	soa-ABC		1.136 87e-13	0	2.463 21e-14	1.266 47e-27
Schwefel	ABC	-125 69.5	-121 86.6	-12 559.7	-12 341.6	1 144.96e+1
	TABC		-12 451	-12 569.5	-12 535.5	2 583.87
	TSABC		-12 444.5	-12 569.5	-12 533.5	3 072.65
	soa-ABC		-12 451	-12 569.5	-12 539.1	2 613.17
Ackley	ABC	0	1.00639e-06	1.29918e-07	3.40477e-07	3.67638e-14
	TABC		1.020 07e-11	1.961 99e-12	4.738 67e-12	3.663 97e-24
	TSABC		6.122 21e-12	1.656 45e-12	3.565 33e-12	1.993 76e-24
	soa-ABC		7.549 52e-14	4.707 35e-14	5.915 27e-14	5.884 36e-29

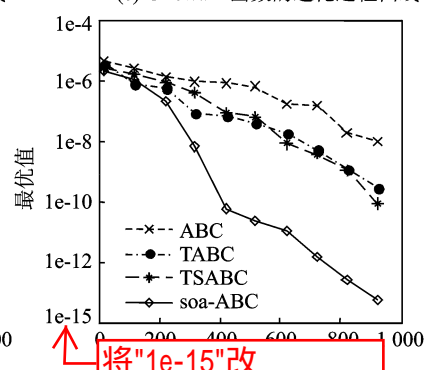
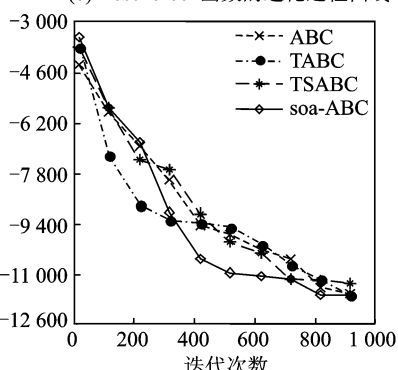
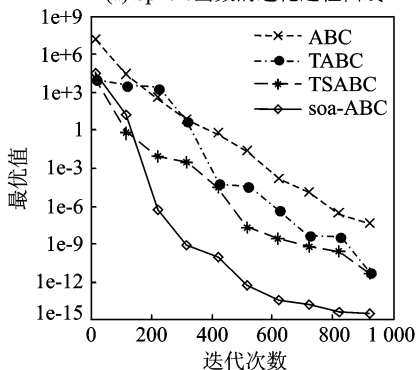
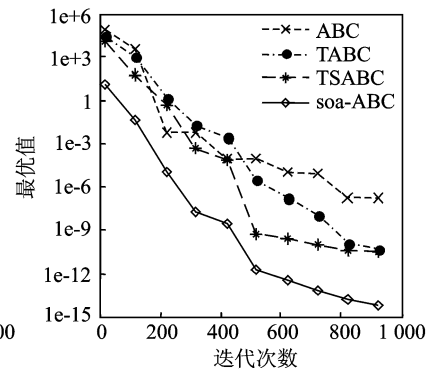
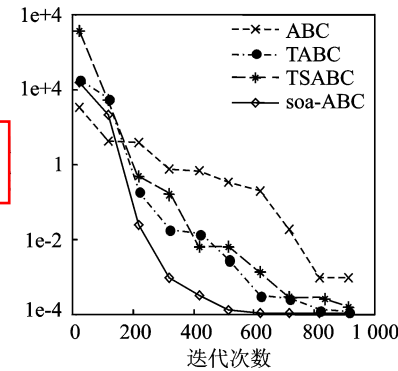
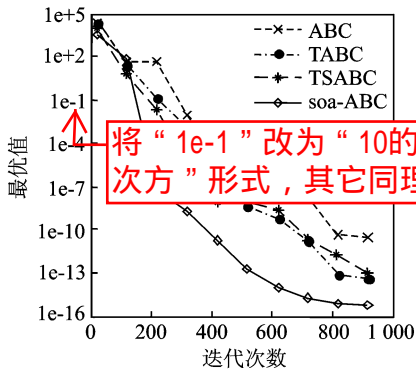


图 1 6 个测试函数的进化过程曲线

表 2 6 个标准测试函数的仿真结果比较

函数	算法	理论值	最大值	最小值	均值	方差
Sphere	runABC		4.075 07e-14	1.464 60e-15	1.069 10e-14	1.010 89e-28
	MABC	0	9.011 0e-15	9.924 61e-16	3.395 95e-15	3.887 85e-30
	soa-ABC		7.605 79e-16	4.303 23e-16	6.018 04e-16	1.147 04e-32
Rosenbrock	runABC		4.143 18	9.480 16e-03	6.498 31e-01	8.322 8e-01
	MABC	0	8.377 88	1.041 04	4.447 46	3.810 13
	soa-ABC		1.305 75	2.973 42e-04	2.025 03e-01	6.821 72e-02
Griewank	runABC		3.666 53e-07	1.332 27e-15	1.406 45e-08	4.504 73e-15
	MABC	0	6.344 61e-05	2.566 84e-13	2.116 37e-06	1.341 74e-10
	soa-ABC		4.130 03e-14	0	1.935 49e-15	6.054 82e-29
Rastrigin	runABC		0.109 46	1.165 29e-11	3.847 43e-3	3.989 22e-04
	MABC	0	1.056 46	8.488 06e-05	3.202 95e-01	1.956 72e-01
	soa-ABC		2.671 64e-12	0	1.117 92e-13	2.347 58e-25
Schwefel	runABC		-12 180	-12 569.5	-12 331	9 763.22
	MABC	-12 569.5	-12 095	-12 569.4	-12 323	1 339.57e+1
	soa-ABC		-12 451	-12 569.5	-12 526.8	3 143.63
Ackley	runABC		9.413 23e-07	8.082 72e-08	3.389 48e-07	5.355 26e-14
	MABC	0	6.55164e-07	7.064 07e-08	3.324 43e-07	1.691 57e-14
	soa-ABC		7.549 52e-14	3.996 8e-14	5.950 8e-14	5.984 46e-29

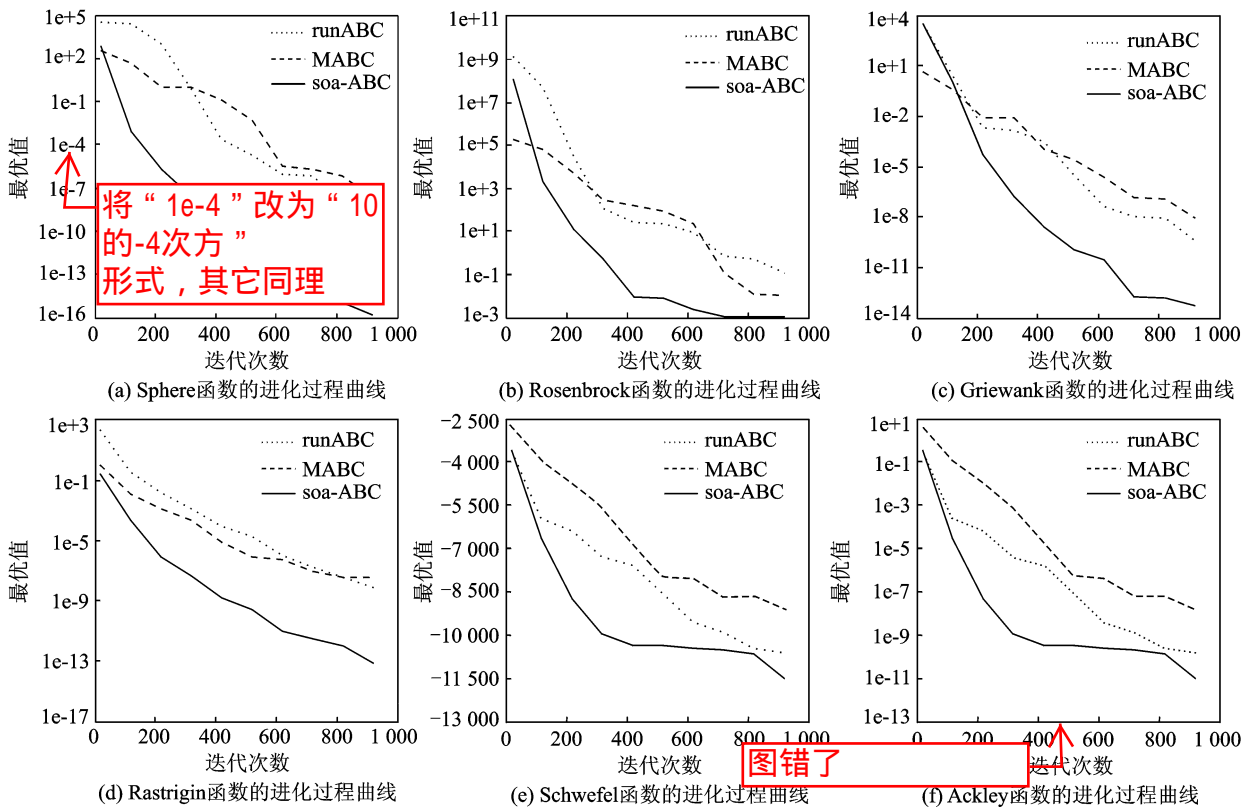


图 2 6 个测试函数的进化过程曲线

从表 2 中可以看出,对上述所有测试函数,不管是最大值、最小值、均值还是方差,改进算法 soa-ABC 的优化结果明显优于 runABC 和 MABC 这两种算法,而且有很好的稳定性。图 b 给出了上述 6 个测试函数实验一次的进化过程曲线。由图可

知,改进的蜂群优化算法 soa-ABC 随着迭代次数的增大,下降的幅度也不断增加,使其更快地向理论全局最优解逼近。从而说明了改进算法具有很强的进化速度,并且能够获得更高的收敛精度,进一步增强了蜂群算法的进化能力。

为验证 soa-ABC 算法的鲁棒性,以 Sphere 函数为例,取维数 $D=20$,蜜源个数分别为 $N=10$, $N=20$, $N=30$,通过固定维数来对不同群体规模进行对比分析。从图 3 中的进化过程曲线可以看出,种群规模越大,进化速度越快,随着进化过程的进行,3 种规模下的改进算法最终均能获得相同的收敛精度,这说明 soa-ABC 算法较好地保持了种群的多样性,从而降低了算法早熟收敛的概率,避免了算法陷入局部最优;并在一定程度上体现出 soa-ABC 算法对群体规模的鲁棒性较好,进而在选择较小群体规模情况下起到降低算法运行时间的效果。

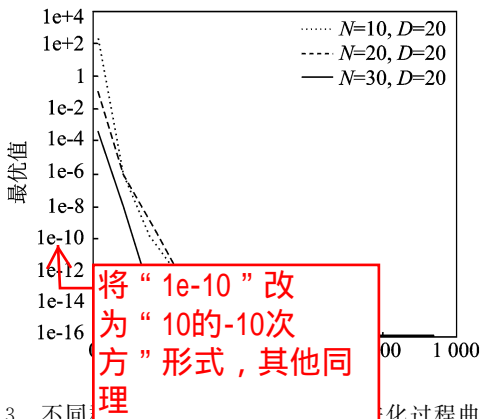


图 3 不同进化过程曲线

4 结束语

针对蜂群算法易陷入局部最优的缺陷,本文所提出的增强寻优能力的改进蜂群算法 soa-ABC 采用两种概率选择机制,增加了种群多样性的同时,降低了陷入局部最优的概率。通过将侦察蜂的个数进行动态的自适应调整,增加了跳出局部最优的机会。另外,在算法每轮结束后,利用中心解邻域的更新搜索进一步增强了算法的寻优能力,具有更高的寻优精度。本文对 6 个标准测试函数进行仿真实验,仿真结果表明本文提出的 soa-ABC 算法能有效跳出局部最优解,具有良好的优化性能,在解决复杂的优化问题上具有明显优势。

参考文献:

[1] Karaboga D. An idea based on honey bee swarm for numerical optimization[D]. Kayseri: Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[2] Karaboga D, Basturk B. On the performance of artificial bee colony (ABC) algorithm[J]. Applied Soft Computing, 2008(1): 687-697.

[3] Karaboga D, Akay B. A comparative study of artificial bee colony algorithm[J]. Applied Mathematics and Computation, 2009, 214(1): 108-132.

[4] 胡中华, 赵敏. 基于人工蜂群算法的机器人路径规划[J]. 电焊机, 2009, 39(4): 93-96.
Hu Zhonghua, Zhao Min. Research on robot path planning based on ABC algorithm[J]. Electric Welding Machine, 2009, 39(4): 93-96.

[5] Karaboga D, Akay B B. Artificial bee colony algorithm on training artificial neural networks[C]//2007 IEEE 15th Signal Processing and Communications Applications Conference. New York: IEEE, 2007: 818-821.

[6] Karaboga D, Akay B B, Ozturk C. Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks[C]//LNCS: Modeling Decisions for Artificial Intelligence. Berlin: Springer-Verlag, 2007: 318-329.

[7] Karaboga D. A new design method based on artificial bee colony algorithm for digital IIR filters[J]. Journal of the Franklin Institute, 2009, 346(4): 328-348.

[8] Srinivasa Rao R, Narasimham S V L, Ramalingaraju M. Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm[J]. International Journal of Electrical Power and Energy Systems Engineering, 2008, 1(2): 709-715.

[9] 李端明, 程八一. 基于人工蜂群算法求解不同尺寸工件单批调度问题[J]. 四川大学学报(自然科学版), 2009, 46(3): 657-662.
Li Duanming, Cheng Baiyi. Artificial bee colony algorithm for scheduling a single batch-processing machine with non-identical job sizes[J]. Journal of Sichuan University: Natural Science Edition, 2009, 46(3): 657-662.

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

[11] 段玉倩, 贺家李. 遗传算法及其改进[J]. 电力系统及其自动化学报, 1998, 10(1): 39-52.
Duan Yuqian, He Jiali. Genetic algorithm and its modification[J]. Proceedings of the EPSA, 1998, 10(1): 39-52.

[12] 李妍, 陈希, 费树岷, 等. 基于遗传算法初始群体优化的台风中心自动定位[J]. 数据采集与处理, 2011, 26(4): 425-429.
Li Yan, Chen Xi, Fei Shuming, et al. Automation locating typhoon center based on improved genetic al-

- gorithm from satellite cloud image[J]. *Journal of Data Acquisition & Processing*, 2011, 26(4):425-429.
- [13] 李亚文,于凤芹. 一种改进选择算子的遗传匹配追踪算法[J]. *数据采集与处理*, 2011, 26(2):177-180.
Li Yawen, Yu Fengqin. Genetic matching pursuit algorithm with improved selection operators[J]. *Journal of Data Acquisition & Processing*, 2011, 26(2):177-180.
- [14] 李奕晨,王毅,牛奕龙,等. 基于标准差的自适应激素调节遗传算法[J]. *数据采集与处理*, 2012, 27(3):333-339.
Li Yichen, Wang Yi, Niu Yilong, et al. Adaptive genetic algorithm based on hormone regulation[J]. *Journal of Data Acquisition & Processing*, 2012, 27(3):333-339.
- [15] Wu Tiejun, Lou Peihuang, Qin Guohua. Novel approach to locator layout optimization based on genetic algorithm[J]. *Transactions of Nanjing University of Aeronautics & Astronautics*, 2011, 28(2):176-182.
- [16] 李志,陈年生,郭小珊,等. 粒子群算法及其改进技术研究[J]. *湖北师范学院学报(自然科学版)*, 2011, 31(2):104-108.
Li Zhi, Chen Niansheng, Guo Xiaoshan, et al. Research on particle swarm algorithm and improved technology[J]. *Journal of Hubei Normal University: Natural Science Edition*, 2011, 31(2):104-108.
- [17] 张民,陈欣,陆宇平. 基于改进 PSO 算法的导弹控制参数优化[J]. *南京航空航天大学学报*, 2009, 41(4):445-449.
Zhang Min, Chen Xin, Lu Yuping. Missile control parameter optimization based on improved PSO algorithm[J]. *Transactions of Nanjing University of Aeronautics & Astronautics*, 2009, 41(4):445-449.
- [18] 陈庆芳,吴小俊. 基于分块互信息和量子粒子群算法的图像配准[J]. *数据采集与处理*, 2011, 26(4):473-477.
Chen Qingfang, Wu Xiaojun. Image registration based on block mutual information and quantum-behaved particle swarm optimization[J]. *Journal of Data Acquisition & Processing*, 2011, 26(4):473-477.
- [19] 胡珂,李迅波,王振林. 改进的人工蜂群算法性能[J]. *计算机应用*, 2011, 31(4):1107-1110.
Hu Ke, Li Xunbo, Wang Zhenlin. Performance of an improved artificial bee colony algorithm [J]. *Journal of Computer Applications*, 2011, 31(4):1107-1110.
- [20] 丁海军,冯庆娴. 基于 boltzmann 选择策略的人工蜂群算法[J]. *计算机工程与应用*, 2009, 45(31):53-55.
Ding Haijun, Feng Qingxian. Artificial bee colony algorithm based on Boltzmann selection policy [J]. *Computer Engineering and Applications*, 2009, 45(31):53-55.
- [21] 向万里,马寿峰. 基于轮盘赌反向选择机制的蜂群优化算法[J]. *计算机应用研究*, 2013(1):86-89.
Xiang Wanli, Ma Shoufeng. Artificial bee colony based on the reverse selection of roulette[J]. *Application Research of Computers*, 2013(1):86-89.
- [22] Gong Maoguo, Jiao Licheng, Du Haifeng, et al. Multiobjective immune algorithm with nondominated neighbor based selection[J]. *Evolutionary Computation*, 2008, 16(2):225-255.
- [23] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm[J]. *Journal of Global Optimization*, 2007, 39(3):459-471.

作者简介:易正俊(1963-),男,教授,研究方向:智能算法、信息融合, E-mail: yizhengjun2006@163.com; 韩晓晶(1988-),女,硕士研究生,研究方向:智能算法。