

一种基于进化知识融合的多目标人工蜂群算法

沈艳霞[†], 陈 杰, 吴定会

(江南大学 物联网技术应用教育部工程研究中心, 江苏 无锡 214000)

摘 要: 提出一种基于进化知识融合的多目标人工蜂群算法. 首先, 采用精英群体知识和种群自身进化知识混合引导引领蜂进化, 保持种群的多样性和优异性; 然后, 将一种融合个体支配关系和种群分布关系的方法引入跟随蜂的概率选择中, 合理选择个体进行深度开发以改善算法收敛性能和分布性能; 最后, 提出一种更为严格的外部档案维护策略以降低外部档案维护成本, 提高解集的分布性能. 通过求解标准测试函数, 并与其他 3 种多目标优化算法进行比较, 仿真结果表明所提出算法具有良好的收敛性能和分布性能, 且解集的覆盖范围更广.

关键词: 多目标优化; 进化知识融合; 多目标人工蜂群算法; 智能优化

中图分类号: TP18

文献标志码: A

A multi-objective artificial bee colony based on evolutionary knowledge integrated

SHEN Yan-xia[†], CHEN Jie, WU Ding-hui

(Research Center of Engineering Applications for IOT, Jiangnan University, Wuxi 214000, China)

Abstract: A multi-objective artificial bee colony algorithm based on evolutionary knowledge integrated is proposed. The elite population knowledge and population itself knowledge are integrated to guide the evolution of employed bees and maintain the diversity and superiority of the population further. Then, a method of combining the individual dominance relationship and the population distribution relationship is introduced into the probability selection of onlooker bees. Finally, a more strict strategy of external file maintenance is proposed to reduce the cost of maintenance and improve the distribution performance of the solution set. Comparing with three kinds of multi-objective optimization algorithms in the standard functions, the results show that the proposed algorithm has good convergence performance and distributed performance, and the coverage of the solution set is more wider.

Keywords: multi-objective optimization; evolutionary knowledge integrated; multi-objective artificial bee colony algorithm; intelligent optimization

0 引 言

随着科学研究和工程应用的深入开展, 多目标优化问题(MOP)已成为理论研究和实践应用的重难点问题之一. 多目标优化问题通常存在多个相互冲突的目标, 其求解过程一般通过协调处理各目标使其尽可能达到最优, 从而得到Pareto最优解集. 目前, 进化算法已广泛应用于求解多目标优化问题, 并取得了大量研究成果, 先后出现了包括NSGA-II^[1]、SPEA2^[2]、MOPSO^[3]和MOEA/D^[4]等一批经典多目标进化算法, 主要解决MOP最优解集的分布性和收敛性两方面问题, 通过研究基于拥挤距离、自适应网格等^[5]外部档案维护策略以获得良好的分布性能, 通过改进如

部分占优、 ϵ -占优等占优机制^[6-7], 引入如粒子群算法、差分进化以及免疫算法等智能算法, 以进一步提高Pareto最优解集的收敛性能.

土耳其学者Karaboga^[8-9]提出的人工蜂群算法(ABC), 作为一种新型群体智能算法, 其结构简单、设置参数较少、求解性能优异, 受到人们广泛关注, 现已成功应用于神经网络训练、电网调度、路径规划等领域. 目前, ABC算法的研究领域大多集中于单目标优化, 性能表现优异, 将ABC算法应用于求解多目标优化问题既能拓宽ABC算法的应用领域, 也为解决多目标优化问题提供了新思路^[10]. 而关于人工蜂群算法在多目标优化方面的研究和应用相对较少, 相比

收稿日期: 2016-10-19; 修回日期: 2017-01-12.

基金项目: 国家自然科学基金项目(61573167, 61572237); 高等学校博士学科点专项科研项目(20130093110011).

作者简介: 沈艳霞(1973-), 女, 教授, 博士, 从事人工智能、风电系统优化等研究; 陈杰(1992-), 男, 硕士生, 从事智能优化算法及应用的研究.

[†]通讯作者. E-mail: shenyx@jiangnan.edu.cn

于单目标优化,多目标优化所面临的进化知识更为复杂多样.现有多目标人工蜂群算法(MOABC)大多结合ABC算法的种群协作进化策略和经典多目标进化算法的占优机制求解MOP.其中:文献[11]以基于向量评价的遗传算法(VEGA)为框架,构建VEABC并成功应用于复合结构的多目标优化;文献[12]提出一种基于自适应网络的MOABC,并通过外部精英档案促进种群协作进化;文献[13]在引领蜂阶段,采用精英引导结合正弦搜索提高MOABC的求解能力,同时利用临时集合保证精英个体进化.现有文献所提出的MOABC虽然在一定程度上提高了求解MOP的能力,但在引领蜂阶段未能充分利用精英解引导种群进化,在跟随蜂的概率选择阶段也未能全面评价个体质量.此外,对于外部档案维护策略还可进一步完善以获得分布性能良好的Pareto前沿.

基于以上分析,本文提出一种基于进化知识融合的多目标人工蜂群算法(EKMOABC).在引领蜂阶段,通过混合外部精英档案集和种群自身进化知识引导群体进化;在跟随蜂阶段,提出一种融合个体支配关系和种群分布关系的概率选择策略,合理选择个体深度开发;在外部档案维护时,采用一种更为严格的策略以降低维护成本,并改善解集的分布性能.通过求解标准测试函数,并对比其他几种算法,验证了本文所提出算法的有效性.

1 多目标优化问题

多目标优化问题一般存在多个优化目标.考虑最小化多目标问题,其数学表达式定义为

$$\min F(x) = [f_1(x), f_2(x), \dots, f_m(x)]. \quad (1)$$

其中: $x = [x_1, x_2, \dots, x_d] \in \Omega$ 为决策变量, d 为决策变量个数, Ω 为决策变量可行解空间; m 为优化目标个数; $f_i(x)$ 为第*i*个目标函数值, $i = 1, 2, \dots, m$.

为求解多目标优化问题,不失一般性,给出以下定义.

定义1 对于决策变量 $x_a, x_b \in \Omega$,称 x_a 占优 x_b ,当且仅当 $\forall f_i(x_a) \leq f_i(x_b)$,且 $\exists f_i(x_a) < f_i(x_b)$,记作 $x_a \prec x_b$.

定义2 称决策变量 $\nu \in \Omega$ 为Pareto最优解,当且仅当 $\neg \exists x \in \Omega : x \prec \nu$.相应地,由Pareto最优解组成的集合称为Pareto最优解集,该解集对应的目标函数值所构成的空间称为Pareto最优前沿.

2 人工蜂群算法

人工蜂群算法(ABC)是一种通过群体合作完成目标优化的智能算法.在人工蜂群算法中,蜂群被分为3类子种群:引领蜂、跟随蜂和侦查蜂.3类蜂群相

互协作,共同促进种群进化.

采用ABC算法求解单目标优化问题时,首先,需对引领蜂进行更新,其公式为

$$v_{ij} = x_{ij} + r \times (x_{ij} - x_{kj}). \quad (2)$$

其中: $i = 1, 2, \dots, N/2; j = 1, 2, \dots, d; N$ 为种群个数,引领蜂和跟随蜂各占一半; d 为决策变量维数; x 为决策变量.其次,引领蜂更新完成后,根据其蜜源质量计算跟随蜂的选择概率,计算公式如下:

$$P_i = F_i / \sum_{i=1}^{N/2} F_i, \quad (3)$$

其中 F_i 为适应度值.以最小化问题为例,其计算公式为

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

其中 f_i 为优化目标函数值.再次,跟随蜂按照式(3)所求概率选择优异蜜源,按式(2)对其进行深度开发.引领蜂和跟随蜂完成更新后,贪婪选择是否保留新解.最后,当某个个体更新达到一定次数,其解仍未得到改善时,引领蜂将转化为侦查蜂,随机产生一个新个体代替原来个体.

ABC算法在求解单目标优化问题时,其性能主要取决于引领蜂的引导作用和跟随蜂的概率选择作用两个方面,这两个方面对于求解多目标优化问题同样重要.但不同于单目标优化问题,多目标优化问题的解是一个Pareto最优集合,且对解集中个体解的评价需综合考虑支配关系和分布关系.因此,在引领蜂阶段,如何通过支配关系和分布关系从最优解集中选取精英解并充分利用;在跟随蜂阶段,如何通过综合指标评价蜜源质量,计算个体更新概率,这两个问题对于将ABC算法应用于求解多目标问题至关重要.此外,还应考虑更为严格的Pareto最优解集维护策略以提高解集分布性能.下面将详细论述解决以上问题的具体做法.

3 基于进化知识融合的多目标人工蜂群算法

3.1 引领蜂的精英群体引导策略

在精英个体选取方面,传统的方法主要包括从Pareto最优解集中随机选取、根据拥挤距离选取以及通过适应度加权方式选取等^[14].本文采用一种半随机方式选取精英个体,具体方法如下:

Step 1: 初次迭代时,按照下式计算Pareto最优解集的拥挤距离,选取除两端点外拥挤距离最大的解作为精英个体:

$$d_i = \begin{cases} \frac{1}{2} \sum_{k=1}^m ((f_i^k - f_i^{k-1})^2 + (f_i^k - f_i^{k+1})^2), & 1 < i < s; \\ \text{inf}, & \text{else.} \end{cases} \quad (5)$$

其中: f_i^k 为第 i 个个体第 k 个目标值, m 为目标数, s 为最优解集个数.

Step 2: 当引领蜂通过 Step 1 选出的精英个体更新后, 对产生的新个体进行贪婪选择, 当新个体支配原个体时替换原个体, 此时, 所产生的新个体具有成为精英个体的潜质. 如果新个体能够支配原精英个体, 则将其选为新的精英个体; 如果互不支配, 则通过轮盘赌法决定是否替换原精英个体, 如果新个体被支配, 则不替换.

Step 3: 当迭代次数达到总次数的 10%, 20%, \dots , 90% 时, 重新采用 Step 1 方式选取精英个体.

以上方法通过支配关系和拥挤距离交替选取精英个体. 一方面, 基于拥挤距离的精英个体选取有利于引导种群均匀进化; 另一方面, 通过支配关系方式选取能够选择更有潜力的新个体以保持种群多样性, 防止种群陷入局部最优.

在精英个体引导种群方面, 本文将种群分为非支配个体和支配个体两部分分别进化, 其公式为

$$\begin{cases} v_{tj} = x_{tj} + r_1(x_{tj} - x_{kj}) + r_2(x_{tj} - x_{bj}), & s_i = 0; \\ v_f = x_f + r_3(x_f - x_b), & s_i = 1. \end{cases} \quad (6)$$

其中: s_i 表示第 i 个个体是否为非支配解, 是则为 1, 否则为 0; x_b 为精英个体, x_{bj} 为精英个体的第 j 维; x_{tj} 和 x_{kj} 分别为第 t 和第 k 个支配个体的第 j 维, 且 $t \neq j$; x_f 为第 f 个非支配解; r_1, r_2, r_3 为 $[-1, 1]$ 之间的随机数. 当某个个体为非支配解时, 将其与精英个体全维交叉, 更容易产生优秀个体, 保持精英群体活力; 而当某个个体为支配解时, 通过精英个体和普通个体混合引导种群进化, 相比于单一的精英引导, 更有利于保持种群多样性.

3.2 融合支配关系和种群分布的概率选择作用

与单目标优化问题不同, 多目标优化问题因同时存在多个互不支配的解, 且需考虑解集的分布性能, 如果仅依靠单一准则计算种群个体适应度值, 则难以获得分布均匀的 Pareto 最优解集.

本文提出一种基于支配关系和种群分布融合的概率选择作用, 充分考虑个体的支配关系和分布关系, 综合评价个体质量以获得具有良好收敛性和分布性的 Pareto 最优解集. 改进后的概率计算公式为

$$P_i = \begin{cases} \frac{1}{1 + (1 - s_i)e^{\eta_1 l_i}}, & s_i = 0; \\ \frac{1}{1 + s_i e^{-\eta_2 d_i}}, & s_i = 1. \end{cases} \quad (7)$$

其中: s_i 表示第 i 个个体是否为非支配解; d_i 为非支配解的拥挤距离, 按式 (5) 计算; η_1 和 η_2 为综合惩罚系数; l_i 为支配个体到 Pareto 最优前沿的欧氏距离, 其计算公式为

$$l_i = \min_{f=1}^s \sqrt{\sum_{k=1}^m (f_{ik} - f_{fk})^2}. \quad (8)$$

这里, f_{ik} 为第 i 个支配个体的第 k 个目标值, f_{fk} 为第 f 个非支配个体的第 k 个目标值, m 为目标个数, s 为最优解集个数.

由式 (7) 可以看出, 当某个个体被支配时, 通过其与最优解集距离的大小对其进行概率惩罚, 距离越近, 惩罚越小, 该个体进一步进化的概率也越大. 但由于其为支配个体, 其进化概率被限制在 $[0, 0.5]$ 之间, 即当个体无限接近于最优前沿时, 该个体的支配概率仅为 0.5. 而对于非支配个体, 则通过非支配解的拥挤距离对其进行概率惩罚, 拥挤距离越大, 惩罚越小, 其进化概率越大. 但考虑到其个体本身为非支配解, 具有先天优越性, 其选择概率区间为 $[0.5, 1]$, 因此, 当拥挤距离最小时, 其进一步进化的概率也比支配个体要大. 这样既能够合理考虑非支配解的优越性, 也能够根据群体分布情况有效区分同一支配关系下, 不同个体的优良程度, 有利于同时改善种群进化的收敛性能和分布性能.

3.3 改进外部档案维护策略

传统的外部档案维护策略通常直接或间接地基于支配关系进入外部档案和基于拥挤距离从外部档案中删除. 当外部档案存储个数达到最大时, 种群进化将产生更多的非支配解, 如果仍仅按照支配关系选取非支配解进化, 则将需要删除更多拥挤距离较小的解, 一定程度上增加了外部档案的维护成本, 且过多的删减也增加了不合理操作的可能性, 不利于获得分布均匀的 Pareto 最优解集. 基于以上考虑, 当外部档案个数达到设定值时, 本文采用一种支配关系和拥挤距离综合评价个体适应度并据此判断该解能否进入外部档案. 具体做法如下.

Step 1: 在外部档案中解的个数未达到最大值时, 选取互不支配的解进入外部档案.

Step 2: 当达到外部档案个数设定值时, 按式 (5) 计算外部档案中解的拥挤距离, 通过循环删减法删除外部档案中拥挤距离最小的解, 并记录每次已删减解的拥挤距离值, 取最大值为 d_t^{\max} , t 为当前迭代次数.

Step 3: 当某个新解与外部档案中任一解互不支配时,需要进入外部档案.此时,先计算该解与外部档案中所有解的欧氏距离,取其最小值记为 d_t^{\min} ,当且仅当 $d_t^{\min} > d_t^{\max}$ 时,选择该解进入外部档案.

改进后的外部档案维护策略对新解进入外部档案增加了基于拥挤距离的限制条件,相比于传统策略更为严格,能够在某个解未进入外部档案之前就对该解进行一轮删减,减少了过多解同时进入外部档案再删减的过程,有效降低了外部档案维护成本,也在一定程度上改善了 Pareto 最优解集的分布性能.

3.4 EKMOABC 算法流程

根据以上分析,本文所提出的基于进化知识融合的多目标人工蜂群算法步骤如下.

Step 1: 初始化种群数量 N ,其中引领蜂和跟随蜂各占一半,设置最大迭代次数 T_m ,外部档案最大存储个数 I_m ,侦查蜂的最大淘汰次数 D_m ,概率惩罚系数 η_1 和 η_2 ,随机初始化蜜源并计算各蜜源的适应度值,根据支配关系选择互不支配的解进入外部档案.

Step 2: 按照 3.1 节精英选取策略选择精英个体,通过式 (6) 对引领蜂进行更新引导种群进化.更新后的蜜源如果支配原蜜源则保留,如果互不支配则对种群中其他支配解进行贪婪选择.

Step 3: 根据 3.2 节计算跟随蜂的选择概率,并依据所计算概率选择是否对该蜜源进行深度进化,进化

完成后,其新蜜源保留策略与引领蜂相同.

Step 4: 当引领蜂和跟随蜂更新完成后,若所得新蜜源未成功保留,则其淘汰次数加 1.当达到最大淘汰次数 D_m 时,引领蜂转化为侦查蜂,重新产生一个蜜源将其替换.

Step 5: 按 3.3 节的策略对外部档案进行维护.

Step 6: 迭代次数加 1,判断是否达到最大迭代次数.若达到则结束并输出外部档案,否则转 Step 2.

4 算法性能测试及分析

4.1 性能指标

本文选取收敛性指标 GD、分布性指标 Δ 以及覆盖率指标 HV 评价算法性能.

1) 收敛性指标 GD. 该指标用于衡量算法所求非支配解与真实最优前沿的距离大小,其公式为

$$GD = \frac{1}{n} \sqrt{\sum_{i=1}^n d_i^2}. \quad (9)$$

其中: n 为算法所求非支配解的个数, d_i 为第 i 个非支配解到真实最优前沿的最小距离. GD 值越小,表明所求解集越接近真实最优前沿,其收敛性能越好.

2) 分布性能指标 Δ . 该指标用于衡量算法所求解集的分布性能,其公式为

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{s-1} |d_i - \bar{d}|}{d_f + d_l + (s-1)\bar{d}}. \quad (10)$$

表 1 各算法 GD 值比较

算法		测试函数							
		ZDT1	ZDT2	ZDT3	ZDT4	ZDT6	DTLZ1	DTLZ2	DTLZ6
NSGA-II	Worse	3.35e-03	9.52e-03	5.24e-02	6.91e-03	7.52e-01	5.59e-01	7.58e-03	8.02e-01
	Best	9.02e-04	6.37e-03	8.26e-03	3.28e-03	9.24e-02	1.24e-01	4.04e-03	3.27e-01
	Mean	1.60e-03	8.25e-03	2.15e-02	4.55e-03	2.15e-01	3.47e-01	5.26e-03	6.02e-01
	Std	3.25e-04	3.55e-04	6.26e-02	1.49e-02	8.35e-02	4.43e-02	3.63e-04	6.71e-02
CMOPSO	Worse	1.18e-03	6.25e-04	8.45e-03	2.57e-02	9.35e-03	1.51e-01	1.25e-02	8.54e-01
	Best	4.43e-04	3.58e-04	1.54e-03	5.38e-03	1.24e-03	5.24e-02	3.24e-03	8.21e-02
	Mean	7.55e-04	4.35e-04	3.26e-03	8.69e-03	4.25e-03	9.03e-02	7.33e-03	2.54e-01
	Std	2.55e-05	6.26e-05	8.11e-03	5.25e-04	7.27e-04	5.25e-03	5.90e-04	3.30e-02
GMOABC	Worse	1.04e-03	1.15e-03	7.54e-03	8.81e-03	7.58e-03	1.44e-01	1.51e-02	7.22e-01
	Best	4.19e-04	6.24e-04	1.82e-03	9.24e-04	8.51e-04	6.21e-02	2.15e-03	6.24e-02
	Mean	8.05e-04	8.23e-04	4.25e-03	3.60e-03	3.57e-03	8.34e-02	7.57e-03	1.02e-01
	Std	4.97e-05	5.22e-05	5.68e-03	6.25e-04	4.22e-04	2.60e-03	3.91e-04	3.34e-02
PMOABC	Worse	5.58e-04	4.55e-04	4.11e-03	7.57e-04	8.24e-04	9.24e-03	5.21e-03	5.29e-02
	Best	1.83e-04	1.57e-04	6.24e-04	4.18e-04	3.24e-04	3.25e-03	1.54e-03	9.05e-03
	Mean	2.57e-04	2.36e-04	1.05e-03	5.22e-04	6.55e-04	6.28e-03	3.85e-03	3.28e-02
	Std	5.68e-05	8.33e-05	2.58e-04	4.20e-05	4.11e-05	7.24e-04	4.57e-04	5.22e-03
EKMOABC	Worse	1.91e-04	1.61e-04	8.28e-04	3.24e-04	6.21e-04	4.25e-03	1.94e-03	2.14e-02
	Best	1.28e-04	1.02e-04	4.29e-04	1.86e-04	3.01e-04	1.74e-03	1.12e-03	8.21e-03
	Mean	1.39e-04	1.14e-04	6.67e-04	2.52e-04	4.59e-04	2.36e-03	1.40e-03	9.84e-03
	Std	4.59e-05	3.22e-05	1.58e-05	1.38e-05	3.55e-05	4.20e-04	4.63e-04	4.57e-04

表2 各算法Δ值比较

算法		测试函数							
		ZDT1	ZDT2	ZTD3	ZDT4	ZDT6	DTLZ1	DTLZ2	DTLZ6
NSGA-II	Worse	5.24e-01	4.47e-01	9.28e-01	7.58e-01	8.27e-01	1.08	8.28e-01	6.82e-01
	Best	2.14e-01	1.67e-01	3.36e-01	3.82e-01	4.08e-01	4.28e-01	3.24e-01	3.13e-01
	Mean	3.68e-01	2.37e-01	6.60e-01	5.22e-01	5.63e-01	7.59e-01	5.42e-01	5.27e-01
	Std	5.26e-02	4.63e-02	6.36e-02	3.57e-02	4.33e-02	4.67e-02	6.86e-02	8.36e-02
CMOPSO	Worse	5.84e-01	5.44e-01	8.54e-01	8.27e-01	8.22e-01	8.28e-01	7.58e-01	7.28e-01
	Best	2.28e-01	2.04e-01	4.27e-01	4.41e-01	3.34e-01	3.69e-01	5.38e-01	3.27e-01
	Mean	4.50e-01	3.42e-01	6.57e-01	6.21e-01	5.62e-01	5.82e-01	6.21e-01	5.63e-01
	Std	4.93e-02	6.36e-02	7.69e-02	6.26e-02	7.06e-02	5.33e-02	4.51e-02	5.20e-02
GMOABC	Worse	7.51e-01	7.41e-01	1.05	8.77e-0	7.51e-01	8.61e-01	9.41e-01	1.05
	Best	4.24e-01	3.24e-01	5.29e-01	4.39e-01	3.28e-01	3.19e-01	5.16e-01	5.18e-01
	Mean	6.26e-01	5.94e-01	8.14e-01	6.15e-01	5.33e-01	6.49e-01	7.28e-01	7.59e-01
	Std	5.24e-02	3.22e-02	4.61e-02	8.27e-02	4.26e-02	3.16e-02	4.36e-02	2.12e-02
PMOABC	Worse	1.97e-01	2.44e-01	6.28e-01	5.47e-01	5.27e-01	4.59e-01	6.19e-01	5.08e-01
	Best	1.58e-01	1.62e-01	4.24e-01	2.54e-01	2.47e-01	2.88e-01	3.14e-01	2.64e-01
	Mean	1.84e-01	1.94e-01	5.27e-01	3.55e-01	3.91e-01	3.18e-01	5.22e-01	3.84e-01
	Std	5.27e-02	4.11e-02	2.28e-02	8.14e-02	5.22e-02	4.22e-02	1.08e-02	3.22e-02
EKMOABC	Worse	1.84e-01	1.78e-01	6.84e-01	3.24e-01	3.95e-01	2.49e-01	3.84e-01	3.28e-01
	Best	1.51e-01	1.44e-01	3.15e-01	1.99e-01	1.89e-01	1.64e-01	1.89e-01	2.04e-01
	Mean	1.60e-01	1.52e-01	4.37e-01	2.31e-01	2.72e-01	1.97e-01	2.26e-01	2.67e-01
	Std	3.28e-02	4.15e-02	8.37e-02	2.35e-02	6.29e-02	3.87e-02	5.81e-02	3.42e-02

表3 各算法HV值比较

算法		测试函数							
		ZDT1	ZDT2	ZTD3	ZDT4	ZDT6	DTLZ1	DTLZ2	DTLZ6
NSGA-II	Worse	3.572	3.254	4.755	3.535	2.754	7.485	7.286	4.912
	Best	3.618	3.295	4.838	3.586	2.858	7.596	7.315	5.028
	Mean	3.599	3.268	4.802	3.558	2.806	7.534	7.305	4.99
	Std	2.55e-02	2.81e-02	1.80e-01	6.36e-02	5.49e-02	3.52e-02	4.22e-02	4.21e-02
CMOPSO	Worse	3.614	3.284	4.758	3.544	2.827	7.588	7.256	4.925
	Best	3.645	3.301	4.902	3.596	2.905	7.69	7.319	5.096
	Mean	3.632	3.302	4.856	3.573	2.872	7.648	7.284	5.032
	Std	2.02e-02	3.69e-02	7.25e-02	7.92e-02	3.59e-02	2.52e-02	3.93e-02	2.52e-02
GMOABC	Worse	3.601	3.275	4.781	3.568	2.795	7.624	7.286	4.968
	Best	3.642	3.308	4.935	3.605	2.907	7.725	7.324	5.102
	Mean	3.625	3.297	4.883	3.589	2.865	7.689	7.291	5.057
	Std	3.56e-02	3.48e-02	4.42e-02	5.22e-02	2.85e-02	3.15e-02	2.55e-02	6.23e-02
PMOABC	Worse	3.635	3.304	4.856	3.584	2.857	7.692	7.286	4.993
	Best	3.651	3.318	4.943	3.618	2.937	7.765	7.322	5.107
	3.643	3.311	4.924	3.606	2.914	7.715	7.295	5.025	
	Std	2.48e-02	1.47e-02	4.28e-02	6.35e-02	2.58e-02	8.21e-02	5.35e-02	4.87e-02
EKMOABC	Worse	3.644	3.31	4.925	3.607	2.905	7.759	7.326	5.036
	Best	3.658	3.318	4.992	3.628	2.984	7.793	7.388	5.202
	Mean	3.650	3.315	4.976	3.619	2.945	7.781	7.359	5.187
	Std	1.79e-02	1.51e-02	2.36e-02	1.49e-02	1.56e-02	2.24e-02	3.32e-02	2.14e-02

其中: $\bar{d} = \frac{1}{s-1} \sum_{i=1}^{s-1} d_i$, d_i 为非支配解间的欧氏距离, d_f 和 d_i 为边界解与真实前沿极端解间的欧氏距离. Δ 值越小, 表明算法所求解集的分布性越好.

3) 覆盖率指标 HV. 该指标用于评价算法所求解集分布的广泛性, 且与非支配解集的收敛性和分布性

关系密切, 能够综合评价算法性能, 其公式为

$$HV = \frac{|X|}{h_i} \quad (11)$$

其中: X 为算法所求 Pareto 前沿个数, h_i 为第 i 个解到参考点的超体积. 该指标越大, 表明所得解集越能够广阔地覆盖在其前端.

4.2 结果分析

为验证本文算法求解多目标优化问题的有效性,将本文算法EKMOABC与文献[1]中的NSGA-II、文献[3]中的CMOPSO、文献[12]中的GMOABC以及文献[15]中的PMOABC作对比实验.选取5个ZDT系列(ZDT1-4,ZDT6)两目标优化问题,3个DTLZ系列(DTLZ1-2,DTLZ6)三目标优化问题进行测试.

为保证对比实验的公平性,将各算法的共同参数设置为:种群个数为100,外部档案存储个数为100,两目标和三目标最大迭代次数分别为500和1000,最大评价次数分别为50000和100000;本文算法EKMOABC中概率惩罚参数 η_1 和 η_2 分别为50和10,侦查蜂最大淘汰次数为50;另外4种算法的其他参数参考原文献.5种不同算法分别独立运行30次,统计并分别比较3种性能指标的最优值、最差值、平均值和标准差.表1~表3分别为收敛性指标GD、分布性指标 Δ 和覆盖率指标HV(两目标和三目标优化

问题的参考点分别取(2,2)和(2,2,2))的统计结果.

由表1的收敛性指标GD可以看出,相比于其他3种算法,本文所提出的EKMOABC算法无论对两目标优化问题还是三目标优化问题,都能够得到更为接近真实Pareto前沿的解集,且具有良好的稳定性能,尤其对ZDT3-4、DTLZ1-2测试问题,EKMOABC算法相比于其他算法,其收敛性指标有较大提高,这表明本文算法不仅能简单且具有凹凸前沿的优化问题尽可能收敛至Pareto前沿,而且对于复杂不连续前端问题也能够取得较好的收敛效果.

由表2的分布性指标可以看出,本文所提出的EKMOABC算法相比于其他3种算法,具有良好的分布性能,且分布性能指标稳定性良好.对于两目标优化问题,EKMOABC算法的分布性指标最好,其次分别为PMOABC、CMOPSO、NSGA-II和GMOABC.其中:对于ZDT3优化问题的分布性指标,4种算法虽都有所下降,但EKMOABC算法性能下降较少且仍

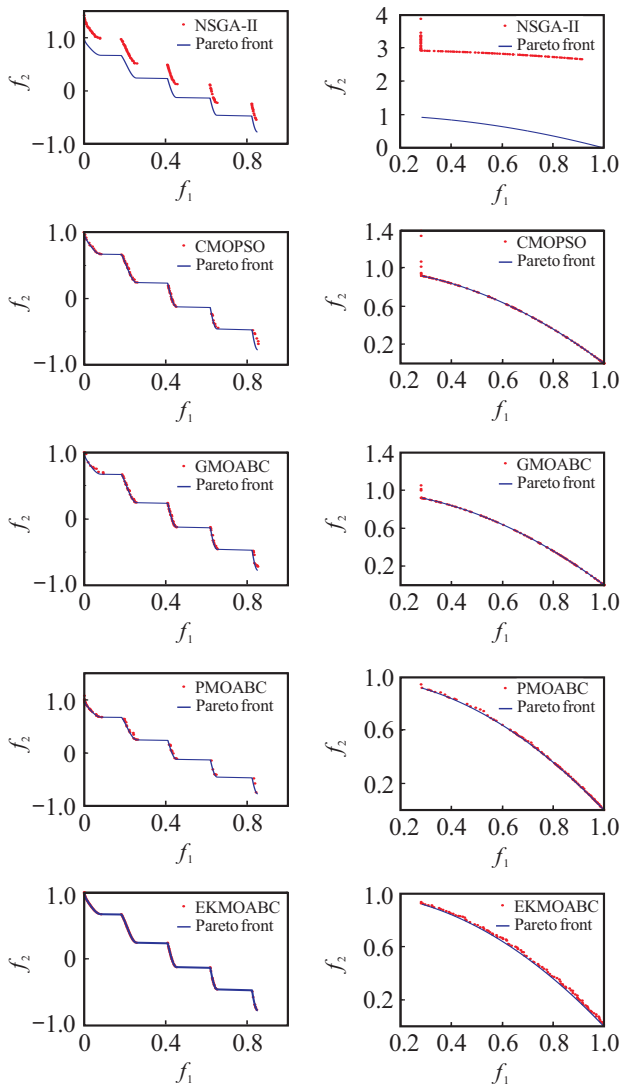


图1 ZDT3

图2 ZDT6

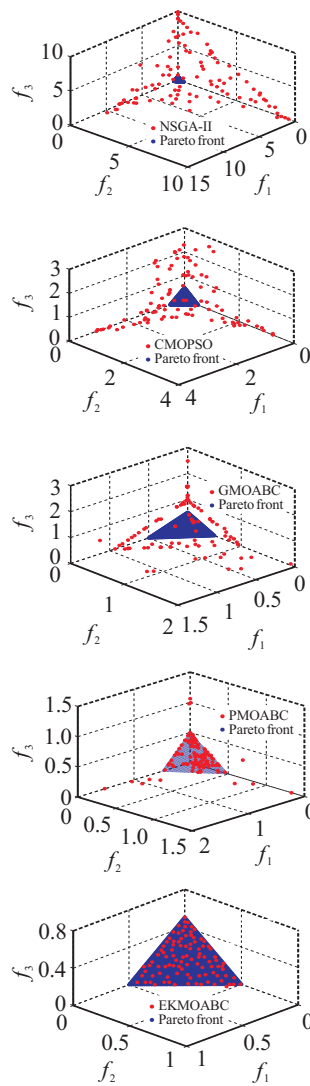


图3 DTLZ1

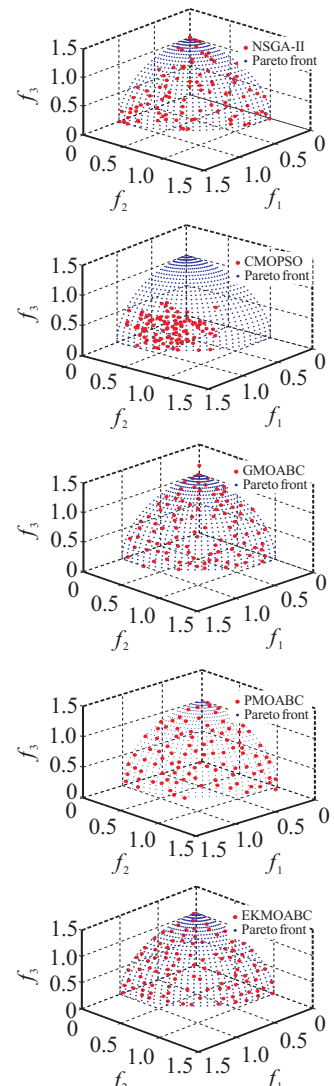


图4 DTLZ2

优于其他4种算法;对于ZDT4和ZDT6优化问题,EKMOABC算法分布性指标保持良好,而其他4种算法分布性能下降较为明显.在三目标优化问题中,EKMOABC和PMOABC算法的分布性能只略微变差,而其他3种算法的分布性能变差较为明显,尤其是CMOABC.

由表3的覆盖性指标HV可以看出,在两目标优化问题和三目标优化问题上,EKMOABC算法的覆盖性指标都大于其他3种算法,表明其所求解集的覆盖范围更广,且覆盖性指标的波动性较小,具有良好的稳定性.

为了更直观地表现算法性能,图1~图4中分别给出了5种算法求解ZDT3、ZDT6、DTLZ1和DTLZ2优化问题的结果对比.由图1~图4可以更直观地看出,EKMOABC算法相比于其他4种算法,无论是两目标不连续的ZDT3和多局部极值点的ZDT6,三目标的DTLZ1和DTLZ2都具有更好的收敛性能,能够更接近于真实的Pareto前沿,且其分布性能和覆盖性能也更为优异,尤其是对于DTLZ1问题,EKMOABC算法的4种性能指标都具有较大改善,进一步表明了本文所提改进算法的有效性.

5 结论

为了提高多目标优化问题的求解性能,本文提出了一种基于进化知识融合的多目标人工蜂群算法,充分考虑了外部精英档案集知识和种群自身进化知识之间的关系以及个体支配关系和拥挤距离之间的关系.在引领蜂阶段,充分利用外部精英档案集来引导种群进化,同时,在跟随蜂的概率选择阶段,引入一种综合考虑支配关系和拥挤距离的概率计算公式,合理评价个体适应度值以促进种群高效进化,提高算法收敛性能和分布性能.此外,还提出了一种更为严格的外部档案维护机制以降低档案维护成本,一定程度上改善了所求解集的分布性能.分别对两目标和三目标测试函数进行测试,仿真结果表明相比于其他4种算法,本文所提算法不仅具有良好的收敛性能和分布性能,而且其解集的覆盖范围更广.

参考文献(References)

[1] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multi-objective genetic algorithm: NSGAI[J]. IEEE Trans on Evolutionary Computation, 2002, 6(2): 182-197.

[2] Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm[R]. Lausanne:

Swiss Federal Institute of Technology Computer Engineering and Networks Laboratory, 2001.

- [3] Coello C A, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization[J]. IEEE Trans on Evolutionary Computation, 2004, 8(3): 256-279.
- [4] Zhang Q F, Li H. MOEA/D: A multi-objective evolutionary algorithm based on decomposition[J]. IEEE Trans on Evolutionary Computation, 2007, 11(6): 712-731.
- [5] 公茂果, 焦李成, 杨咚咚, 等. 进化多目标优化算法研究[J]. 软件学报, 2009, 20(2): 271-289. (Gong M G, Jiao L C, Yang D D, et al. Research on evolutionary multi-objective optimization algorithms[J]. J of Software, 2009, 20(2): 271-289.)
- [6] Hernandez-Daz A G, Santana-Quintero L V, Coello Coello C A, et al. Pareto-adaptive dominance[J]. Evolutionary Computation, 2007, 15(4): 493-517.
- [7] 刘衍民, 赵庆祯, 牛奔, 等. 基于 ϵ 占优的自适应多目标粒子群算法[J]. 控制与决策, 2011, 26(1): 89-95. (Liu Y M, Zhao Q Z, Niu B, et al. Adaptive multi-objective particle swarm optimization based on dominance[J]. Control and Decision, 2011, 26(1): 89-95.)
- [8] Karaboga D. An idea based on honey bee swarm for numerical optimization[R]. Turkey: Computers Engineering Department, Erciyes University, 2005.
- [9] Karaboga D, Basturk B. On the performance of artificial bee colony(ABC) algorithm[J]. Applied Soft Computing, 2008, 8(1): 687-797.
- [10] 秦全德, 程适, 李丽, 等. 人工蜂群算法研究综述[J]. 智能系统学报, 2014, 9(2): 127-135. (Qin Q D, Cheng S, Li L, et al. Artificial bee colony algorithm: A survey[J]. CAAI Trans on Intelligent Systems, 2014, 9(2): 127-135.)
- [11] Omkar S N, Senthilnath J, Khandelwal R, et al. Artificial bee colony(ABC) for multi-objective design optimization of composite structures[J]. Applied Soft Computing, 2011, 11(1): 489-499.
- [12] Akbari R, Hedayatzadeh R, Ziarati K, et al. A multi-objective artificial bee colony algorithm[J]. Swarm and Evolutionary Computation, 2012, 2: 39-52.
- [13] 葛宇, 梁静. 一种多目标人工蜂群算法[J]. 计算机科学, 2015, 42(9): 257-262. (Ge Y, Liang J. Multi-objective artificial bee colony algorithm[J]. Computer Science, 2015, 42(9): 257-262.)
- [14] 胡旺, Yen G G, 张鑫. 基于Pareto熵的多目标粒子群优化算法[J]. 软件学报, 2014, 25(5): 1025-1050. (Hu W, Yen G G, Zhang X. Multi-objective particle swarm optimization based on Pareto entropy[J]. J of Software, 2014, 25(5): 1025-1050.)

(责任编辑: 李君玲)