

改进选择策略的有约束多目标优化算法^①

杨景明^② 王成浩^③ 吴绍坤

(燕山大学工业计算机控制工程河北省重点实验室 秦皇岛 066004)

摘 要 针对一般多目标优化算法在搜索过程中存在无法有效判断搜索方向和有效利用个体信息的问题,提出了改进选择策略的有约束多目标优化算法(CMOP-ISS)。该算法首先对 Deb 教授提出的约束占优直接支配选择策略进行改进,使得算法在向可行 Pareto 前沿进化的同时,可以保留更多优秀不可行解;另外采用双种群策略,利用优秀的可行与不可行个体,共同产生新个体,加快算法的收敛速度。将 CMOP-ISS 与其他算法在经典测试函数上进行对比实验,结果表明 CMOP-ISS 不仅提升了算法的收敛性能,而且保证了 Pareto 解集良好的分布性。

关键词 约束多目标优化(CMOP); 选择策略; 收敛性; 分布性

0 引 言

在实际科研与工程实践中,很多问题的优化都涉及到优化算法^[1]。然而在优化领域,许多问题都是有约束条件限制的,因此开展约束优化算法的研究具有重要的实际意义。

近年来,研究者提出了大量将约束条件结合到进化算法中的约束处理方法^[2]。根据约束优化进化算法中约束处理方式的不同可以将它们划分为 6 类:罚函数法^[3]、可行性法则^[4]、随机排序法^[5]、 ϵ 约束处理法^[6]、多目标优化法^[7] 和混合法^[8]。罚函数法通过添加惩罚因子的方式将约束优化问题转化为无约束优化问题,是最为传统的约束处理方法之一,该类方法的主要不足在于惩罚参数的选取比较困难;可行性法则将可行解和不可行解分离处理,其中可行解优于不可行解,因此不可行解难以进入种群,这使得该方法在处理最优解位于可行域边界上的约束优化问题时可能存在问题;随机排序法采

用概率参数评判个体优劣,在一定程度上平衡了目标函数和约束违反度的关系,但稳定性较差; ϵ 约束处理法通过设定 ϵ 值将约束划分为不同的区间,解的约束违反度在不同的约束区间内采用不同的处理方法,它是在牺牲时间的前提下对搜索空间进行充分的搜索;多目标优化法将约束优化问题转换为多目标优化问题,并采用多目标优化技术来处理问题,然而没有引入偏好的多目标优化方法可能并不像人们所想象的那样高效;混合法求解约束优化问题是近些年来研究的热点,该方法是通过将不同的约束处理机制相结合来处理约束优化问题。

本文采用双种群存储的思想^[9],将可行解和不可行解分开存储,避免了可行解和不可行解的直接比较,从而能够较好地保留优秀解。对于有约束多目标优化问题,优秀的不可行解对于种群进化非常重要,因此利用改进后的选择策略,可以保留种群中优秀的不可行解,从而为种群的进化提供更加有利的方向信息。

① 河北省自然科学基金(F2016203249)资助项目。
② 男,1957 年生,博士,教授;研究方向:冶金机械综合自动化,先进控制及工程应用;E-mail: yangjm6188@sina.com
③ 通信作者,E-mail: 2505042904@qq.com
(收稿日期:2019-01-02)

1 约束多目标优化问题

不失一般性,以目标函数最小化为例,约束多目标优化问题可以定义为^[10]

$$\begin{aligned} \min F(x) &= [f_1(x), f_2(x), \cdots, f_m(x)] \\ x &\in R^n \\ \text{s. t. } &\begin{cases} g_i(x) \leq 0 & i = 1, 2, \cdots, p \\ h_j(x) = 0 & j = 1, 2, \cdots, q \end{cases} \end{aligned} \tag{1}$$

其中, $F(x)$ 称为 m 维目标函数, $x = (x_1, x_2, \cdots, x_n) \in R^n$ 称为 n 维决策变量, $h_j(x) = 0$ 为第 j 个不等式约束条件, p 为不等式约束条件的个数, $h_j(x) = 0$ 为第 j 个等式约束条件, q 为等式约束条件的个数。将满足所有约束条件的解空间 S 称为式(1)的可行域,否则称为不可行域。

约束违反程度函数如式(2)所示,其大小称为约束违反度。

$$G(x) = \sum_{i=1}^p \max(0, g_i(x)) + \sum_{j=1}^q \max(0, |h_j(x)|) \tag{2}$$

2 改进的选择策略

进化算法中选择策略是算法前进的动力,在有约束多目标优化问题中,目前应用最广泛的是 Deb 教授等人^[11]提出的约束占优直接支配选择策略,它由 3 条比较准则构成:

准则 1 如果 2 个个体都是可行解时,选择非支配个体作为较优解,如果 2 个个体互不支配,则选择拥挤度较大的个体作为较优解;

准则 2 如果一个个体是可行解,另一个个体是不可行解时,则选择可行解作为较优解;

准则 3 如果 2 个个体均为不可行解时,则选择约束违反程度较小的个体作为较优解;

对上述准则进行分析,可知该选择策略存在 2 个明显的问题:(1)当可行解 x_F 的目标函数值很差,而不可行解 x_{IF} 的目标函数值较优同时约束违反度较小时(如图 1 所示),此时显然不可行解 x_{IF} 更接近真实的 Pareto 前沿,因此保留不可行解 x_{IF} 更利于搜索到更优可行解,但根据约束占优直接支配选择

策略准则 2 会选择保留 x_F 而不是 x_{IF} ; (2)当不可行解 x_{IF1} 的目标函数值很差而约束违反度较小,不可行解 x_{IF2} 的目标函数值较优而约束违反度相对较大时(如图 2 所示),保留 x_{IF2} 更有利于搜索到靠近真实 Pareto 前沿的个体,从而改善搜索效率以及加快收敛速度,但根据约束占优直接支配选择策略准则 3,会选择保留 x_{IF1} 而不是 x_{IF2} 。

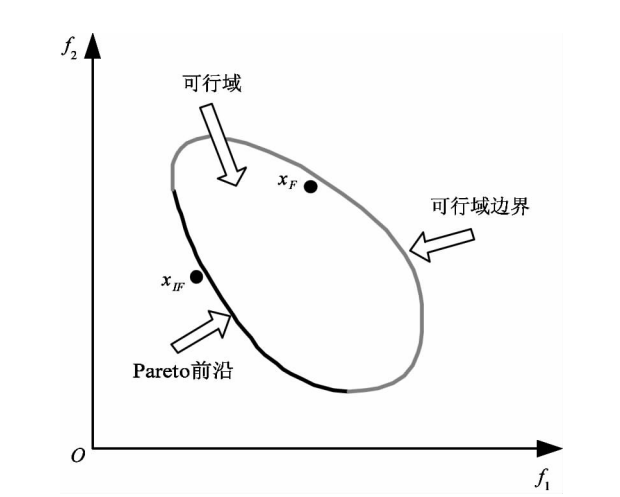


图 1 可行解优于不可行解

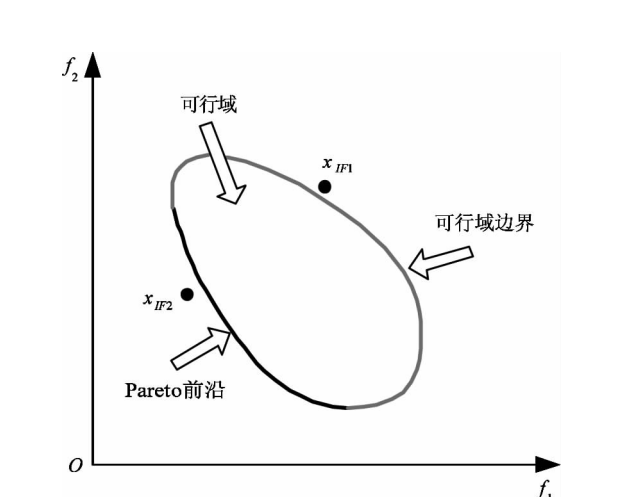


图 2 不可行解优于不可行解

为解决上述问题,对约束占优直接支配选择策略的准则 2、3 进行改进,改进后的选择策略为:

准则 4 如果 2 个个体都是可行解时,选择非支配个体作为较优解,如果 2 个个体互不支配,则选择拥挤度较大的个体作为较优解;

准则 5 一个个体是可行解,另一个个体是不可行解时,分 2 种情况讨论:

情况 1:如果不可行解是允许解,那么选择非支配个体作为较优解,如果 2 个个体互不支配,则选择拥挤度较大的个体作为较优解;

情况 2:如果不可行解是不允许解,则选择可行解为较优解。

准则 6 如果 2 个个体均为不可行解时,选择非支配个体作为较优解,如果 2 个个体互不支配,则选择违反约束程度低的个体作为较优解。

定义如下概念:

(1) 种群约束允许放松程度 $\varepsilon(\text{gen})$ ^[12]。 $\varepsilon(\text{gen})$ 表示种群进化到第 gen 代时个体违反约束程度 $G(x)$ 的界限。其表达式如下:

$$\varepsilon(\text{gen}) = \begin{cases} \varepsilon(\text{gen} - 1)/\theta & \varepsilon > 10^{-6} \\ 0 & \varepsilon \leq 10^{-6} \end{cases} \quad (3)$$

其中, θ 表示种群每进化一代种群约束允许放松程度减小的比例。

(2) 允许解。种群进化到第 gen 代时,如果 $0 < G(x) \leq \varepsilon(\text{gen})$, 个体 x 就是允许解。

(3) 不允许解。种群进化到第 gen 代时,若 $G(x) > \varepsilon(\text{gen})$, 个体 x 就是不允许解。

(4) 拥挤度

$$w(i) = \begin{cases} \infty & \text{个体 } i \text{ 处于边界} \\ \sum_{j=1}^k \left| \frac{f_j(i+1) - f_j(i-1)}{f_j(N) - f_j(1)} \right| & \text{其他个体} \end{cases} \quad (4)$$

其中, k 为目标函数的个数, N 是种群的个体数, $f_j(1), f_j(2), \dots, f_j(N)$ 是全部个体在目标函数上从小到大的排列。

改进的选择策略中采用约束放松的目的是为了利用不可行解, ε 采用线性收缩,每次收缩程度都相同,随着进化代数的增加 ε 不断减小,种群中不可行解的数量不断减少,直到种群约束允许放松程度减为 0。在种群进化前期,种群内保持一定数量的不可行解可以维持种群多样性,而随着进化代数的增加,不可行解提供的有效信息越来越少,故在种群进化后期,更多的是利用可行个体的信息。

改进后的选择策略如图 3 ~ 图 5 所示。

通过与约束占优直接支配选择策略相比较,可知改进后的选择策略准则 5 中结合了种群约束允许

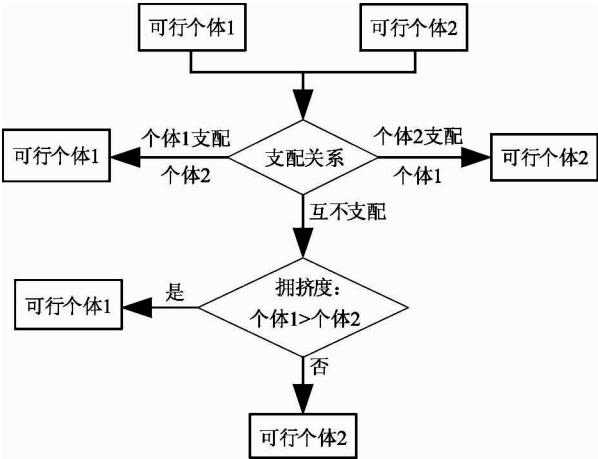


图 3 改进选择策略准则 4

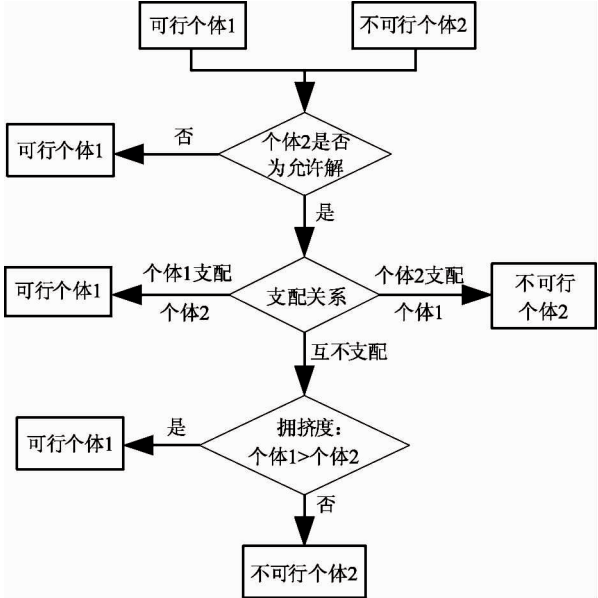


图 4 改进选择策略准则 5

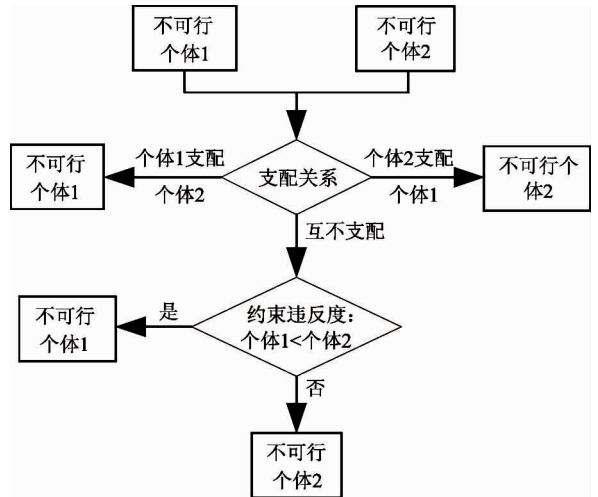


图 5 改进选择策略准则 6

放松程度的概念,不再以个体是否可行作为选择的唯一标准,使得可行解与不可行解竞争时,目标函数值较优且约束违反度较小的不可行解得以胜出,从而为变异、交叉等操作提供更有利的信息。准则 6 增加了支配关系的比较,将其作为首要判断标准,不再以约束违反度作为唯一的选择标准,可以保留目标函数值较优但约束违反度相对较大的不可行解,从而有利于搜索到靠近真实 Pareto 前沿的个体。

3 其他操作算子的改进

3.1 外部种群更新策略

约束处理技术在约束多目标优化算法中起着重要的作用,好的约束处理技术能较好地协调可行解与不可行解的关系,从而提高约束多目标优化算法的求解效率。本文采用的双种群存储策略在处理约束多目标优化问题时,建立了 2 个种群,分别存储可行解和不可行解,避免了两者的直接比较,在每个种群里面的个体按照不同方式进行更新,最后得到最优解集。该策略充分促进了可行解与不可行解之间的信息交流,能够较好地保留优秀的不可行解,从而扩大种群的搜索范围,提高种群的多样性。可行解集和不可行解集的更新方式如下所述。

3.1.1 可行解集的更新

将本代新产生的可行解与可行解集合并成一个新的可行解种群,选出其中的非支配个体,如果所选出的个体数小于 N_1 (可行解集的规模),则直接构成新的可行解集;如果个体数大于 N_1 , 再利用式(4)计算个体的拥挤度,删除拥挤度较小的个体,直到剩余的个体数为 N_1 , 然后将其作为新的可行解集。

3.1.2 不可行解集的更新

将本代新产生的不可行解与不可行解集合并成新的不可行解集,再将新的不可行解集与可行解集合并成一个新的种群,筛选出其中的非支配个体,然后删除选出的非支配个体中的可行解,剩下的个体即为选出的优秀不可行解,如果所选出的不可行解个数小于 N_2 (不可行解集的规模),则直接构成新的不可行解集;如果不可行解个体数大于 N_2 , 则利用式(4)计算个体的拥挤度,然后删除拥挤度小的个

体,直到数量达到 N_2 , 并将其作为新的不可行解集。

上述不可行解集的更新方式中,选出的不可行解支配可行解或与可行解互不支配,因此选出的不可行解具有较优的目标函数值,这样的不可行解有助于产生更优的可行解。

3.2 变异策略

有约束多目标优化问题有它自身的特点,可行域内大多数可行解的适应度值可能没有那些处于约束边界上或附近的不可行解的适应度值优秀,因此不可行解在种群进化中不能被简单地舍弃,利用优秀不可行个体的方向信息,可以增强算法的局部开发能力,提高算法的搜索效率。

在进化的过程中,随着进化代数的增加,不可行解提供的信息越来越少,本文提出如下变异策略:

$$v^{t+1} = \begin{cases} r_1 \times x_g + (1 - r_1) \times x_{IF} + r_2 \times (x_{r_1} - x_{r_2}) & t \leq 0.5G_{\max} \\ r_1 \times x_g + r_2 \times (x_{r_1} - x_{r_2}) & t > 0.5G_{\max} \end{cases} \quad (5)$$

其中, x_g 为可行解集中的随机个体, x_{IF} 为不可行解集中的随机个体, x_{r_1}, x_{r_2} 为在 t 代中不同的 2 个随机个体, r_1, r_2 为 $[0, 1]$ 之间的随机数, t 为迭代次数, $G_{\max}$ 为最大迭代次数。

上式中,以 $t = 0.5G_{\max}$ 为分界点,在种群进化前期,即 $t \leq 0.5G_{\max}$, 充分利用种群中优秀不可行解的方向信息对于提高种群多样性和保证种群收敛性起着重要的作用。而在种群进化后期,即 $t > 0.5G_{\max}$, 随着进化代数的增加,不可行解的有效信息越来越少,故此后续算法更多地利用可行解的信息,忽略不可行解。

3.3 交叉策略

交叉操作,如下式^[13]所示:

$$u_{ij}(t+1) = \begin{cases} v_{ij}(t+1) & rand(j) \leq CR \\ x_{ij}(t+1) & \text{其他} \end{cases} \quad (6)$$

式中, t 为进化代数, v_{ij} 为个体 v_i 的第 j 维分量, x_{ij} 为个体 x_i 的第 j 维分量, $rand(j)$ 为 $[0, 1]$ 之间随机数,交叉算子 $CR = CR_{\min} + (1 - \alpha) \times (CR_{\max} - CR_{\min})$, 其中 $\alpha = (G_{\max} - t)/G_{\max}$, $CR_{\max}, CR_{\min}$ 分别为最大、最小交叉算子,与一般采用固定值交叉算子有所不同的是,改进选择策略的有约束多目标优化

算法 (constrained multi-objective optimization algorithm with improved selection strategy, CMOP-ISS) 中所采用的交叉算子在算法搜索初始阶段较小,提高种群的收敛速度;而随着搜索代数逐渐增大,交叉算子也逐渐增大,有利于种群的多样性。

4 算法实现流程

CMOP-ISS 的基本流程如表 1 所示。

表 1 CMOP-ISS 的基本流程

改进选择策略的有约束多目标优化算法	
1	初始化种群 Pop,设置最大迭代次数;
2	计算种群 Pop 的目标函数值,约束违反度;
3	选取可行解集 FeaPop,不可行解集 InfeaPop;
4	While $t < G$
5	对本代个体进行变异,交叉,选择操作,选出下一代个体;
6	更新可行解集 FeaPop 和不可行解集 InfeaPop;
7	$t++$;
8	End

5 实验分析

本文采用 CTP3、CTP4、CTP7、CF1 4 个经典标准测试函数对算法的性能进行测试,其中 CTP3、CTP4、CTP7 的测试函数表达式如下:

$$\begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = c(x) \left(1 - \frac{f_1(x)}{c(x)}\right) \\ g(x) = a | \sin\{b\pi[\sin(\theta)[f_2(x) - e] \\ \quad + \cos(\theta)f_1(x)]^c\} |^d - \cos(\theta)[f_2(x) - e] \\ \quad + \sin(\theta)f_1(x) \leq 0 \end{cases}$$

其中, $c(x) = 41 + \sum_{i=2}^5 [x_i^2 - 10 \cos(2\pi x_i)]$, $0 < x_1 < 1$, $-5 < x_i < 5$, $i = 2, 3, 4, 5$, 参数 a, b, c, d, e, θ 的取值如表 2 所示。CF1 的表达式如下所示:

$$\min f_1(x) = x_1 + \frac{2}{|J_1|} \sum_{j \in J_1} (x_j - x_1^{0.5(1+\frac{3(j-2)}{n-2})})^2$$

$$\min f_2(x) = 1 - x_1 + \frac{2}{|J_2|} \sum_{j \in J_2} (x_j - x_1^{0.5(1+\frac{3(j-2)}{n-2})})^2$$
$$g(x) = 1 - f_1 - f_2 + a | \sin[N\pi(f_1 - f_2 + 1)] | \leq 0$$
其中: $0 \leq x_j \leq 1$ ($j = 1, 2, \dots, n$), $n = 10$, $J_1 = \{j | j = 3, 5, \dots, n\}$, $J_2 = \{j | j = 2, 4, \dots, n\}$, $N = 10, a = 1$.

表 2 CTP3、CTP4、CTP7 参数列表

	θ	a	b	c	d	e
CTP3	-0.2π	0.1	10	1	0.5	1
CTP4	-0.2π	0.75	10	1	0.5	1
CTP7	-0.05π	40	5	1	6	0

在相同的测试函数及硬件条件下,将 CMOP-ISS 与 NSGA-II^[14]、CMODE^[15] 和 DPDE^[16] 算法分别独立运行 30 次。采用常用的性能评价指标 IGD^[17] 来测试 4 种算法的性能,分别对 4 种算法所得结果进行统计比较,分析算法的优劣。本文算法参数设置如表 3 所示。

表 3 CMOP-ISS 参数列表

参数	数值
种群大小	200
可行解集大小	100
不可行解集大小	50
进化代数	1 000
最大交叉算子	1
最小交叉算子	0.4

图 6 ~ 图 9 是本文算法与其他 3 种算法从 30 次运行中随机选择的一次运行结果,从图中可以看出,4 种算法都具有较好的分布性,均能找到真实的 Pareto 前沿的大致位置。

表 4 ~ 表 7 为本文算法与其他 3 种算法在 CTP3、CTP4、CTP7、CF1 4 个测试函数的 IGD 值比较。通过比较可以看出,本文算法的 IGD 值均优于其他 3 种算法,并且具有最小的标准差,这说明本文算法所得的非劣解集具有更好的分布性和收敛性,更加接近真实的 Pareto 前沿,且算法更加稳定。

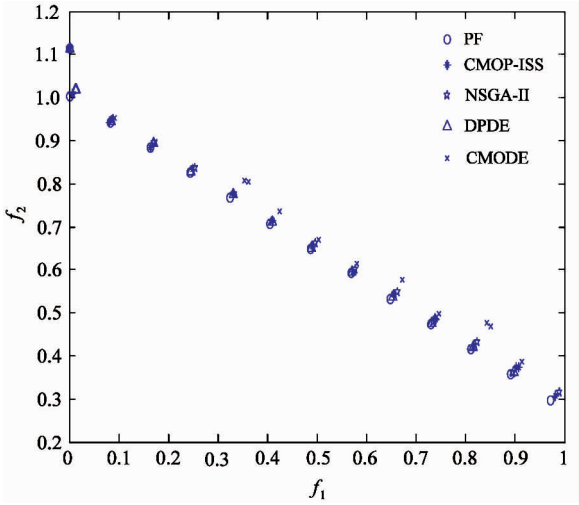


图 6 测试函数 CTP3 的 Pareto 前沿

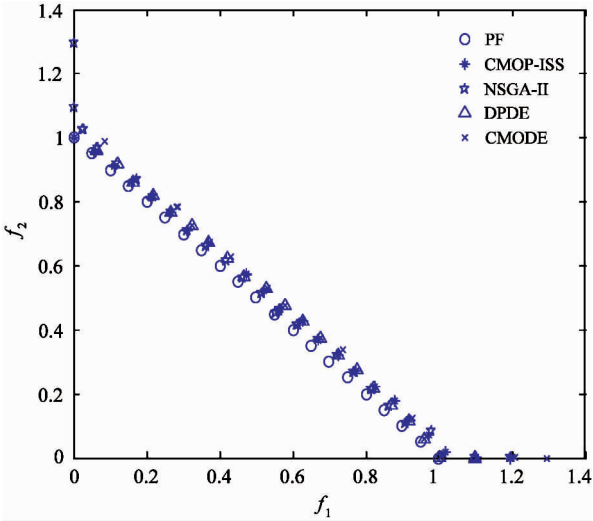


图 9 测试函数 CF1 的 Pareto 前沿

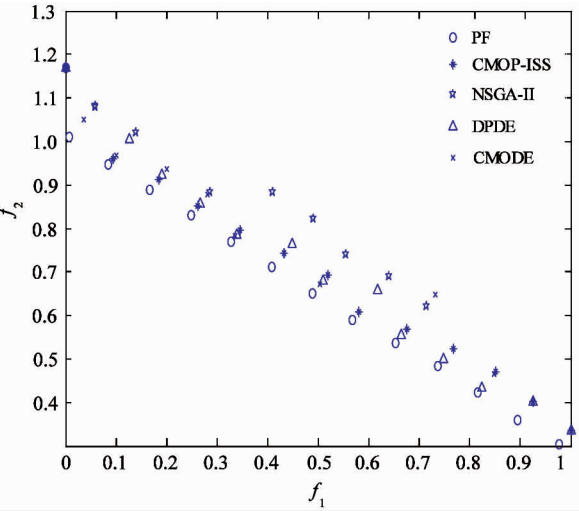


图 7 测试函数 CTP4 的 Pareto 前沿

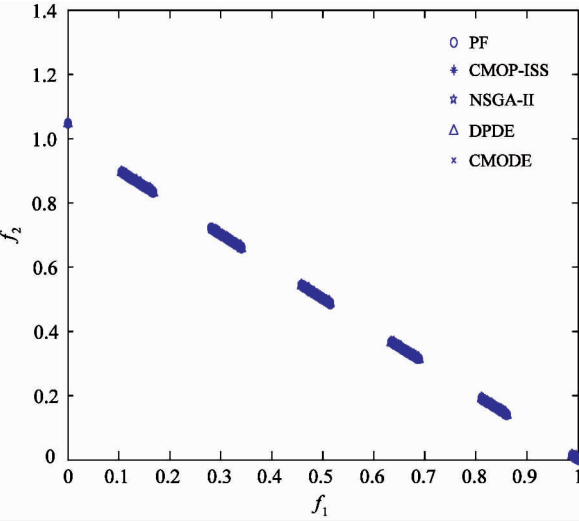


图 8 测试函数 CTP7 的 Pareto 前沿

表 4 测试函数 CTP3 的 IGD 值比较

	CMODE	NSGA-II	DPDE	CRDSOA
最优解	1.27e-3	1.29e-4	6.78e-5	5.12e-5
最差解	2.16e-3	1.36e-1	2.46e-4	1.56e-4
平均值	6.37e-4	8.66e-3	1.50e-4	9.13e-5
标准差	3.16e-4	2.66e-2	4.05e-3	2.50e-5

表 5 测试函数 CTP4 的 IGD 值比较

	CMODE	NSGA-II	DPDE	CRDSOA
最优解	1.69e-3	3.22e-3	1.59e-3	1.03e-3
最差解	2.43e-2	1.38e-1	7.36e-3	5.71e-3
平均值	9.33e-3	3.02e-2	3.01e-3	2.86e-3
标准差	6.21e-3	3.61e-2	1.37e-3	1.08e-3

表 6 测试函数 CTP7 的 IGD 值比较

	CMODE	NSGA-II	DPDE	CRDSOA
最优解	1.99e-6	2.16e-6	3.23e-6	1.87e-6
最差解	1.35e-4	8.62e-6	6.65e-1	2.20e-6
平均值	1.39e-5	2.86e-6	1.07e-1	1.99e-6
标准差	2.94e-5	2.27e-6	2.45e-1	8.54e-8

表 7 测试函数 CF1 的 IGD 值比较

	CMODE	NSGA-II	DPDE	CRDSOA
最优解	7.67e-3	4.50e-4	1.16e-3	4.37e-4
最差解	2.05e-3	1.22e-3	2.15e-3	9.97e-4
平均值	1.32e-3	8.01e-4	1.68e-3	6.17e-4
标准差	3.20e-4	1.86e-4	2.49e-4	1.33e-4

6 结 论

本文在处理有约束多目标优化问题时,考虑到可行解和不可行解在算法的寻优过程中均具有重要的作用,采用2个种群分别保存可行解和不可行解,避免两者的直接比较,通过改进选择策略,使得优秀不可行解参与种群进化;利用新的不可行解的更新方式,保留目标函数值较优的不可行解,这样的不可行解对于种群进化是至关重要的;通过改进的变异、交叉策略,提高种群的全局搜索和局部搜索能力。综合分析本文算法与NSGA-II、CMODE、DPDE算法在CTP3、CTP4、CTP7、CF1经典测试函数中的比较结果,表明本文算法具有更好的收敛精度。对于约束多目标优化问题,如何权衡各个目标函数,如何处理约束条件和目标函数的关系,如何协调解集的分布性和收敛性仍是未来急需研究的内容。

参考文献

- [1] 禹鑫焱,朱熠琛,詹益安,等. SLAM 过程中的机器人位姿估计优化算法研究[J]. 高技术通讯, 2018, 28(8):48-54
- [2] 李智勇,黄滔,陈少森,等. 约束优化进化算法综述[J]. 软件学报, 2017, 28(6):1529-1546
- [3] Gong W, Cai Z H, Liang D W. Adaptive ranking mutation operator based differential evolution for constrained optimization [J]. *IEEE Transactions on Cybernetics*, 2015, 45(4):716-727
- [4] Saha A, Datta R, Deb K. Hybrid gradient projection based genetic algorithms for constrained optimization[C] // 2010 IEEE Congress on Evolutionary Computation, Barcelona, Spain, DOI:10.1109/CEC.2010.5586303
- [5] Jan M A, Khanum R A. A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D[J]. *Applied Soft Computing Journal*, 2013, 13(1):128-148
- [6] 毕晓君,王珏,李博,等. 基于动态迁移的 ε 约束生物地理学优化算法[J]. 计算机研究与发展, 2014, 51(3):580-589
- [7] 蔡自兴,江中央,王勇,等. 一种新的基于正交实验设计的约束优化进化算法[J]. 计算机学报, 2010, 33(5):855-864
- [8] Mallipeddi R, Suganthan P N. Ensemble of constraint handling techniques[J]. *IEEE Transactions on Evolutionary Computation*, 2010, 14(4):561-579
- [9] 孟红云,张小华,刘三阳. 用于约束多目标优化问题的双群体差分进化算法[J]. 计算机学报, 2008, 31(2):228-235
- [10] Daneshyari M, Yen G G. Constrained multiple-swarm particle swarm optimization within a cultural framework [J]. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 2012, 42(2):475-490
- [11] Deb K, Pratap A, Meyarivan T. Constrained test problems for multi-objective evolutionary optimization [C] // The 1st International Conference on Evolutionary Multi-Criterion Optimization, Zurich, Switzerland, 2001:284-298
- [12] 郑建国,王翔,刘荣辉. 求解约束优化问题的 ε -DE 算法[J]. 软件学报, 2012, 23(9):2374-2387
- [13] 俞国燕,李鹏,何真,等. 一种用于多目标约束优化的改进进化算法[J]. 计算机集成制造系统, 2009, 15(6):1172-1178
- [14] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2):182-197
- [15] 毕晓君,刘国安. 基于云差分进化算法的约束多目标优化实现[J]. 哈尔滨工程大学学报, 2012, 33(8):1022-1031
- [16] 孟红云,张小华,刘三阳. 用于约束多目标优化问题的双群体差分进化算法[J]. 计算机学报, 2008, 31(2):228-235
- [17] Venske S M, Gonçalves R A, Delgado M R. ADEMO/D: Multiobjective optimization by an adaptive differential evolution algorithm [J]. *Neurocomputing*, 2014, 127(127):65-77

Constrained multi-objective optimization algorithm with improved selection strategy

Yang Jingming, Wang Chenghao, Wu Shaokun

(Key Laboratory of Industrial Computer Control Engineering of Hebei Province,

Yanshan University, Qinhuangdao 066004)

Abstract

Aiming at the problem that the general multi-objective optimization algorithm can not effectively judge the search direction and utilize individual information in the search process, a constrained multi-objective optimization algorithm with improved selection strategy (CMOP-ISS) is proposed. Firstly, the algorithm improves Professor Deb's constrained dominant direct dominant selection strategy, so that the algorithm can retain more excellent infeasible solutions while evolving to the feasible Pareto frontier. In addition, a dual-population strategy is adopted to generate new individuals by utilizing excellent feasible and infeasible individuals, so as to accelerate the convergence speed of the algorithm. Comparing CMOP-ISS with other algorithms on classical test functions, the results show that CMOP-ISS not only improves the convergence performance of the algorithm, but also guarantees the good distribution of Pareto solutions.

Key words: constrained multi-objective optimization (CMOP), selection strategy, convergence, distribution