

文章编号: 1001-0920(2008)12-1435-06

一种保持群体多样性的多目标遗传算法

申晓宁, 郭 毓, 陈庆伟, 胡维礼

(南京理工大学 自动化学院, 南京 210094)

摘要: 提出一种保持群体多样性的多目标遗传算法. 该算法采用一种基于信息熵的测度, 以衡量群体在多目标空间下的多样性, 并利用该测度将群体当前的进化状态(多样性)与算法的运行机制相关联, 设计了若干种增强算法探索力度的策略, 有效地开拓了算法的搜索范围, 提高了进化过程中群体的多样性, 防止了算法早熟收敛. 对所提出算法的计算复杂度进行了理论分析. 仿真实验表明, 所提出的算法具有较好的收敛性能和分布特性.

关键词: 多样性; 多目标优化; 遗传算法; 熵

中图分类号: TP18 **文献标识码:** A

Multi-objective optimization genetic algorithm keeping diversity of population

SHEN Xiaoning, GUO Yu, CHEN Qingwei, HU Weili

(School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China. Correspondent: SHEN Xiaoning, E-mail: sxnystyt@sina.com.cn)

Abstract: A multi-objective genetic algorithm keeping diversity of the population is proposed. The algorithm uses a metric based on entropy to measure the diversity of the population in the case of multi-objective space. The evolving state of the current population is associated with the running mechanism of the algorithm by the diversity metric, and several strategies are designed to enhance the extent of exploration of the algorithm, which widens the searching range of the algorithm, and increases the diversity of the evolving population and prevents premature convergence. The computational complexity of the algorithm is analyzed theoretically. Simulation results indicate that the proposed algorithm has good performance of convergence and distribution.

Key words: Diversity; Multi-objective optimization; Genetic algorithm; Entropy

1 引言

遗传算法基于群体操作能在一次运行结束后并行搜索到多个解, 因而它特别适合于求解具有多个解的多目标优化问题. 多目标遗传算法有两个关键性的技术问题需要解决^[1]: 一是通过选取适当的适应值评价和选择策略, 引导算法收敛到问题真实的 Pareto 最优前沿; 二是在进化过程中维持群体的多样性, 并使得算法搜索到的近似 Pareto 最优前沿具有良好的分布特性且散布在较广的范围.

目前, 维护进化群体及最终解集多样性的方案有多种: NSGA^[2]等算法采用小生境法, 使得与其余个体相似性较小的个体有更多的机会被遗传到下一代的群体中; PAES^[3]采用网格法对搜索空间进行

划分, 个体邻域的密度定义为与它处于同一网格中的个体的数目; SPEA^[4]采用聚类法对相似个体进行合并; SPEA2^[5]和 NSGA-^[6]分别采用“最近邻居法”和“排挤距离”估计个体邻域的密度; 文献[7]将与某个体距离最近的两个体的平均距离作为该个体的密度估计指标. 上述方法各有特点, 但在某些情况下, 它们只能求得问题全局 Pareto 最优前沿的子集, 或陷入局部最优前沿. 究其原因, 一是算法进行选择操作的首要依据是个体的 Pareto 非支配级别, 只有当级别相同时才将密度信息作为选择的标准. 该做法只能在当前搜索到的非支配解集中维持较好的分布性能, 而不能推动算法进一步探寻到更宽广或更优的区域; 二是算法在进化选优的过程中, 对精

收稿日期: 2007-09-05; 修回日期: 2007-12-25.

基金项目: 国家自然科学基金项目(60174019, 60474034); 江苏省自然科学基金项目(BK2007210).

作者简介: 申晓宁(1981—), 女, 南京人, 博士生, 从事进化算法、多目标优化的研究; 胡维礼(1941—), 男, 江苏东台人, 教授, 博士生导师, 从事智能控制、网路控制系统等研究.

英个体过度“利用”,适应度较高的个体(当前非支配解)将逐渐占据整个群体,新个体难以进入,导致交叉、变异等繁殖算子因支配池中的个体模式缺乏多样性而无法产生新区域中的解;三是在算法运行过程中,未将群体当前的进化状态与选择及搜索机制相关联,算法不能依据群体状态的动态变化而及时地调整搜索策略,具有一定的盲目性。

近年来,有一些学者将信息熵的概念引入多目标优化方法的研究.文献[8]利用熵度量 Pareto 解集分布的平坦性,但它需将问题的可行空间划分为多个网格单元,对每个网格均需计算密度函数,当网格数目很大时计算量将大大增加,且当优化目标数大于 2 时,需要将解集在目标空间上构成的超平面投影到二维平面上再计算熵.文献[9]引入相对熵作为目标间均衡性的评价指标,以避免极端不均衡解的产生,为寻找偏好解提供了途径.然而,该方法需要确定均衡性重要程度系数和小生境参数,这些参数对算法的结果有较大的影响。

针对现有方法的不足,本文提出一种保持群体多样性的多目标遗传算法,简称 KDMOGA.该算法基于信息熵度量多目标空间下群体的多样性,无需设定任何额外的参数,整个求解过程方便可行;采用外部集合保存历代搜索到的精英个体,并依据该集合中个体的多样性自适应地调整算法参数,以指导选择操作和搜索机制,防止算法因对当前精英的过度依赖而产生停滞现象或早熟收敛.仿真算例表明了所提算法的有效性。

2 一种新型的基于熵的群体多样性测度

定义 1(多样性测度 E) 设群体 pop 中的个体依据目标向量间的相似性可划分为 K 个不同种类的子集 C_1, C_2, \dots, C_K , 各子集中的个体数目分别为 $|C_1|, |C_2|, \dots, |C_K|$, 且 $\forall i, j \in \{1, 2, \dots, K\}$, $C_i \cap C_j = \phi, C_i \cup C_j = \text{pop}$, 则 pop 的多样性测度 E 定义为

$$E = \frac{-\sum_{j=1}^K P_j \log(P_j)}{D^{\max}} \quad (1)$$

其中: $P_j = |C_j| / N$, N 为群体的规模, P_j 为个体属于第 j 类子集的概率; D^{\max} 为 $-\sum_{j=1}^K P_j \log(P_j)$ 可能取得的最大值,该值当 $K = N$ 时取得.对于十进制编码, $D^{\max} = \log(N)$;对于染色体长度为 L 的二进制编码, $D^{\max} = \log(\min(N, 2^L))$. $0 \leq E \leq 1$, 当 $K = 1$ 时, E 取最小值 0.

在定义 1 中,判断目标向量间相似性的方法如

下:

Step1: 求取群体 pop 分别在 m 个目标上的最大值 f_i^{\max} 和最小值 $f_i^{\min}, i = 1, 2, \dots, m$.

Step2: 将 pop 中的每个个体 x 在目标 f_i 上的值 $f_i(x)$ 分别依据 f_i^{\max} 和 f_i^{\min} 进行归一化,记 x 在所有目标上均归一化后构成的目标向量为

$$F^{\text{norm}}(x) = (f_1^{\text{norm}}(x), f_2^{\text{norm}}(x), \dots, f_m^{\text{norm}}(x)),$$

其中 $f_i^{\text{norm}}(x) = (f_i(x) - f_i^{\min}) / (f_i^{\max} - f_i^{\min})$.

Step3: 依次对 pop 中的每一对个体 x 和 x' , 计算经归一化后的目标向量间的欧拉距离 $F^{\text{norm}}(x) - F^{\text{norm}}(x')$, 若 $F^{\text{norm}}(x) - F^{\text{norm}}(x') < \epsilon$, 则将 x 和 x' 归为同一子集,直至处理完所有个体.分类产生的各子集 C_j 中,满足 $\forall x \in C_j, \exists x' \in C_j$ 且 $x \sim x'$, 有 $F^{\text{norm}}(x) - F^{\text{norm}}(x') < \epsilon$.

Step3 中参数 ϵ 的取值如下:

$$\epsilon = \sqrt{(1/N)^2 * m} \quad (2)$$

依据式(2)取值的目的是期望当前群体中的所有个体沿着各目标(归一化后取值范围为 $[0, 1]$) 均匀分布,从而在目标空间中保持较好的多样性。

由定义 1 可见,群体中个体类型越多,分配得越平均, E 的值越大。

3 一种保持群体多样性的多目标遗传算法

3.1 外部集合的使用

所提出算法 KDMOGA 设置一个外部集合保存历代搜索到的非支配解,并将它作为问题的近似 Pareto 最优解集。

在 KDMOGA 的每一进化代中,将从外部集合中取出若干 (N_{1t} 个) 邻域密度较小(采用排挤距离^[6] 估计密度)的个体,组成新一代的部分父代群体,并对这部分群体进行扰动(变异操作),生成相应的子代群体.该做法的目的是利用位于稀疏区域内精英个体的优良遗传特性,在它们的附近进一步探索到质量更优或潜力更大的解。

在第 t 代中,从外部集合中选取的个体数目 N_{1t} 由当前外部集合的多样性测度 E_t 自适应决定,即

$$N_{1t} = a * (1 - E_t)^2 * N + b \quad (3)$$

其中: $\lfloor \cdot \rfloor$ 表示小于 \cdot 的最大整数, N 为父代群体的总规模;本文取参数 $a = 0.5, b = 0.1 * N$.

当 E_t 较小,即当前外部集合的多样性较差时,需从中选取较多个邻域密度较小的个体进行探索,以提高外部集合中个体的分布特性;当 E_t 较大时,可适当地减少进行探索的个体数目。

3.2 对精英个体的控制

为避免由于对精英的过度利用而产生过大的选择压力,借鉴 controlled NSGA-^[10] 对精英个体

进行控制的策略,所提出算法对新一代的部分父代群体(另一部分由外部集合产生)采用如下生成方式:

设第 t 代中,父代与子代合并后的群体共被划分为 L_t 层非支配级别,对每一级别中的个体,依据排挤距离从大到小进行排列.从各级别中总共选取 N_{2t} 个个体 ($N_{2t} = N - N_{1t}$) 构成新一代的部分父代群体.设在第 i ($i = 1, 2, \dots, L_t$) 层级别中允许选取的最大个体数目为 n_{it} ,取 n_{it} 为等比数列,满足

$$n_{it} = r_i n_{(i-1)t}, \quad i = 2, \dots, L_t, \quad (4)$$

其中 $0 < r_i < 1$ 为衰减率.由等比公式可得

$$n_{it} = N_{2t} * \frac{1 - r_i}{1 - r_i^{L_t}} * r_i^{i-1}, \quad i = 1, 2, \dots, L_t. \quad (5)$$

因为 $0 < r_i < 1$,所以在第 1 层级别中允许选取的最大个体数目最多,在其余级别中允许选取的数目按指数依次衰减.这使得每一层非支配级别均有个体存在于新群体,从而在保留一定数目的精英个体的同时,增加了群体中个体模式的多样性.

Controlled NSGA- 中的衰减率 r 在整个进化过程中保持不变.对于不同的问题, r 的最优取值不同,用户难以预先确定,因此本文令 r_i 随当前代外部集合的多样性测度 E_t 自适应变化,即

$$r_i = 1 - E_t^2. \quad (6)$$

当 E_t 较大,即当前外部集合的多样性较好时, r_i 取较小的值,从而在较好的非支配级别中取较多的个体,以增强对精英个体的利用;当 E_t 较小时, r_i 取较大的值,使得较差的级别中也有较多的个体进入新群体,以加大探索的力度,期望搜索到新区域中的优良个体,促进算法的收敛.

3.3 “探索”与“利用”之间的模式转换

所提出算法 KDMOGA,依据当前外部集合的多样性测度 E_t 调节算法在“探索新个体”与“利用精英个体”两种搜索模式之间的转换.设置 2 个阈值参数 C_{good} 和 C_{bad} .当 $E_t > C_{good}$ 时,认为当前外部集合具有较好的多样性,算法进入“利用精英个体”的模式,在继承现有个体优良遗传特性的基础上,进一步发掘更优的个体;当 $E_t < C_{bad}$ 时,认为当前外部集合的多样性过差,算法转入“探索新个体”的模式,在新一代的群体中插入若干随机生成的个体,使用随机个体可能存在的潜在的优良模式探索未知区域中的精英个体,开拓算法的搜索范围,增加群体的多样性,从而跳出算法可能已经陷入的“局部最优”状态.本文取 $C_{good} = 0.7, C_{bad} = 0.5$.

3.4 对排挤距离的改进

个体的排挤距离^[6]定义为:在各个经归一化的目标函数上,排列在该个体的左侧和右侧的 2 个体

间距离的平均值.排挤距离的计算无需引入任何小生境参数,简单方便,但它也存在一定的问题.如图 1 所示,个体 e 和 f 虽然在 f_1 上的值较好,但在 f_2 上的值很差,它们与个体 a, b, c, d 本不属于同一 Pareto 前沿.然而,在进化过程中,当支配 e 和 f 的个体未产生时, a, b, c, d, e, f 由于互不支配而被归为同一级别中.由定义可得,个体 e 的排挤距离为无穷大, f 的排挤距离为 0.8720,而 a 的排挤距离为 0.4103, b 的排挤距离为 0.7850, c 的排挤距离为 0.8373.因此,在排挤选择时, a, b 和 c 将由于排挤距离较小而依次被淘汰.如果算法一直未能搜索到支配 e 和 f 的个体,则当它运行结束后, e 和 f 将被保留在非支配解集中. e 和 f 这类个体的存在,占据了算法的搜索空间,降低了搜索效率.

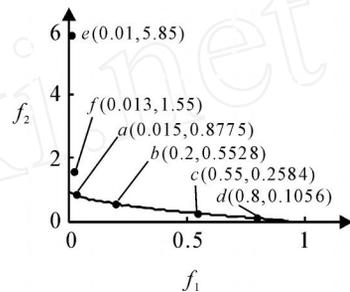


图 1 排挤距离引发的问题

为解决上述问题,本文设定算法的计算精度为一适当的常数 P .当 2 个体 x 和 x' 进行 Pareto 支配的比较时,如果在第 j 个经归一化的目标上,有 $|f_j^{norm}(x) - f_j^{norm}(x')| > P$,则认为 x 和 x' 在 f_j 上的值相等.如令 $P = 0.007$,则在图 1 中, e, f, a 在 f_1 上相等,显然 e 和 f 将因受 a 支配而遭淘汰.由此可见,通过适当地降低计算精度,算法能在搜索过程中逐渐移除那些在某个或某些目标上与其余个体差别很小,但在其他目标上较差的个体,使最终结果中出现不合理个体(如图 1 中的 e 和 f)的概率减少,促进算法的收敛,同时也使得解集具有更好的分布性能.

3.5 算法步骤

KDMOGA 算法的详细步骤如下:

Step 1: 令进化代数 $t = 1$,随机产生一个规模为 N 的初始父代群体 $parpop_t$,并创建一个空的外部集合 $archive_t$;对初始父代群体进行快速非支配排序、选择、交叉和变异操作,产生子代群体 $chipop_t$;设模式变量 $mode = \text{“探索”}$,最大进化代数 $= T$.

Step 2: 令 $mixedpop_t = parpop_t \cup chipop_t$,对 $mixedpop_t$ 进行快速非支配排序,同时对每一级别中的个体计算排挤距离,并按该值由大到小进行排序.

Step 3: 将 $mixedpop_t$ 中的所有非支配解加入 $archive_t$,将 $archive_t$ 中存在的重复解及受其余解支

配的解删除. 如果 $archive_t$ 中解的个数超过了其最大容量 M , 则移除那些排挤距离较小的解.

Step4: 计算 $archive_t$ 的多样性测度 E_t .

Step5: 如果 $mode = \text{“探索”}$ 且 $E_t > C_{good}$, 则 $mode = \text{“利用”}$; 如果 $mode = \text{“利用”}$ 且 $E_t < C_{bad}$, 则 $mode = \text{“探索”}$.

Step6: 由式(3) 计算 N_{1t} , 再求得 $N_{2t} = N - N_{1t}$. 如果 $mode = \text{“探索”}$, 则随机产生规模为 $0.1 * N$ 的群体 $randpop_{t+1}$, 并令 $N_{2t} = N_{2t} - 0.1 * N$.

Step7: 依据 3.1 节从 $archive_t$ 中选择规模为 N_{1t} 的群体 $parpop_{1(t+1)}$, 并对其进行变异操作, 生成子代群体 $chipop_{1(t+1)}$; 依据 3.2 节从 $mixedpop_t$ 中选择规模为 N_{2t} 的群体 $parpop_{2(t+1)}$, 并对其进行选择、交叉和变异操作, 生成子代群体 $chipop_{2(t+1)}$.

Step8: 令新一代父代群体 $parpop_{t+1} = parpop_{1(t+1)} \cup parpop_{2(t+1)}$. 如果 $mode = \text{“探索”}$, 则 $chipop_{t+1} = chipop_{1(t+1)} \cup chipop_{2(t+1)} \cup randpop_{t+1}$; 否则, $chipop_{t+1} = chipop_{1(t+1)} \cup chipop_{2(t+1)}$.

Step9: 如果 $t \geq T$, 则算法终止, 将当前外部集合 $archive_t$ 作为近似 Pareto 最优解集输出; 否则, $archive_{t+1} = archive_t, t = t + 1$, 转 Step2.

3.6 算法的计算复杂度

3.6.1 时间复杂度

设目标数为 m , 算法的时间复杂度取决于:

- 1) 快速非支配排序, $O(mN^2)^{[6]}$.
- 2) 计算个体的排挤距离, $O(mN \log(N))^{[6]}$.
- 3) 对外部集合更新时, 判断其中是否存在重复的解或若干解被其余解支配最多需进行 $4mNM$ 次比较, 因此该步骤的时间复杂度为 $O(mNM)$.
- 4) 外部集合在更新后最多有 $2N + M$ 个个体, 将它们分别依 m 个目标进行排序, 以计算个体的排挤距离进行裁减操作. 采用快速排序法, 该步骤的时间复杂度为 $O(m(2N + M) \log(2N + M))$.
- 5) 求取外部集合的多样性测度时, 将所有个体划分为不同子集所需的时间复杂度为 $O(mM^2)$.

一般情况下, 取 $M = N$, 则 KDMOGA 总的时间复杂度取上述分析结果中的最大值 $O(mN^2)$.

3.6.2 空间复杂度

KDMOGA 的空间复杂度主要取决于:

- 1) 快速非支配排序, $O(N^2)^{[6]}$.
- 2) 外部集合在更新后最多有 $2N + M$ 个个体, 保存排挤距离, 空间复杂度为 $O(2N + M)$.

算法总的空间复杂度取上述最大值 $O(N^2)$.

3.6.3 与其他算法的比较

NSGA- 是一种经典的快速非支配排序多目

标精英遗传算法, controlled NSGA- 是 NSGA- 的改进算法. 设它们的群体规模为 L , 则其时间复杂度均为 $O(mL^2)$, 空间复杂度均为 $O(L^2)^{[6]}$. 在 KDMOGA 中, 令群体规模 $N = L$, 则它与这两种算法在最坏情况下具有相同的时间与空间复杂度.

4 仿真实验

4.1 优化问题描述

$$T_1^{(1)} \begin{cases} \text{Minimize } f_1(x) = x_1, \\ \text{Minimize } f_2(x) = g(x) \cdot h(x). \end{cases}$$

其中

$$g(x) = 1 + 10(n - 1) + \prod_{i=2}^n (x_i^2 - 10 \cos(4 \cdot x_i)),$$

$$h(x) = 1 - \sqrt{f_1(x) / g(x)},$$

$$n = 10, x_1 \in [0, 1],$$

$$x_2, \dots, x_n \in [-5, 5]. \tag{7}$$

T_1 包括 21^9 个局部 Pareto 最优前沿. 当 $g(x) = 1$ 时, T_1 具有全局 Pareto 最优前沿.

$T_2^{(1)}$: 如图 2 所示, 设计目标是 minimized 构架杆的体积(即生产成本) 和构件 AC, BC 上的压力:

$$\begin{cases} \text{Minimize } f_1(x) = x_1 \sqrt{16 + y^2} + x_2 \sqrt{1 + y^2}, \\ \text{Minimize } f_2(x) = \max(AC, BC); \\ \text{s. t. } \max(AC, BC) \leq 10^5, \\ 1 \leq y \leq 3 \text{ 且 } 0 \leq x_1, x_2 \leq 0.01. \end{cases} \tag{8}$$

其中: y 表示 B 与 C 之间的垂直距离, x_1 和 x_2 分别为 AC 和 BC 的长度 (y, x_1 和 x_2 的单位均为 m). AC 和 BC 上的压力 F_{AC} 和 F_{BC} 分别计算如下:

$$F_{AC} = \frac{20 \sqrt{16 + y^2}}{yx_1}, \quad F_{BC} = \frac{80 \sqrt{1 + y^2}}{yx_2}.$$

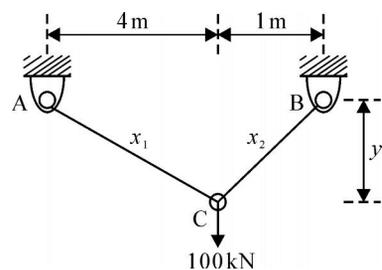


图 2 构架杆示意图

4.2 实验结果

使用收敛性测度 $M_1^{(1)}$ (度量算法搜索到的解到问题真实的 Pareto 最优解集在目标空间上的平均最短距离)、测度 $C^{(1)}$ (度量两两解集间相互支配的程度)、分布性能测度 $SP^{[12]}$ (度量解集中的每个解到其余解的最短距离的标准偏差, 即解集分布的均匀性)、本文给出的多样性测度 E 以及作图法, 将

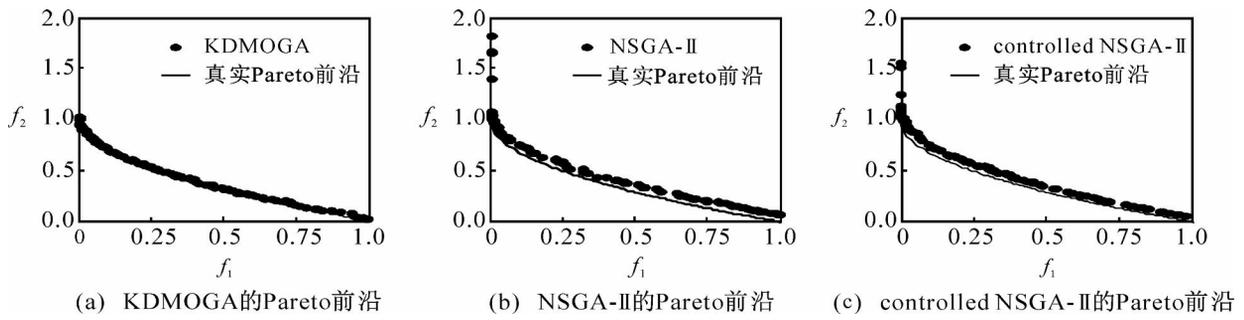


图 3 T_1 中 Pareto 前沿的比较

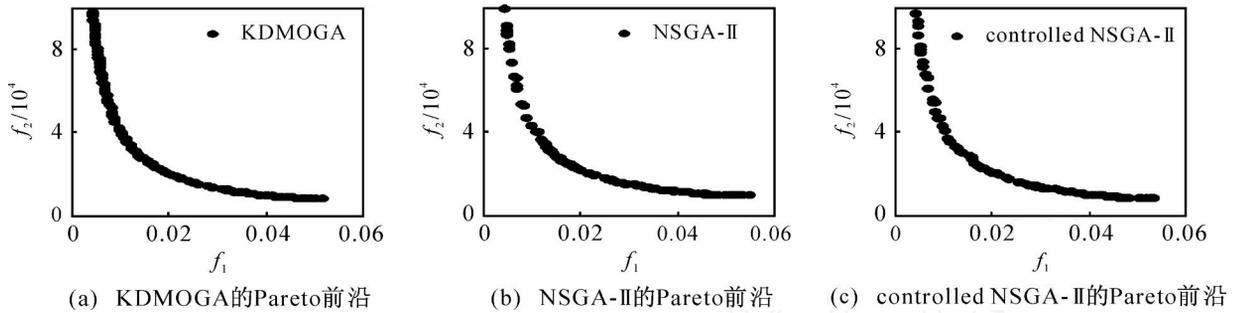


图 4 T_2 中 Pareto 前沿的比较

所提出算法与两种已有的多目标遗传算法 NSGA- ,controlled NSGA- 在 T_1 和 T_2 中进行比较. 测度 M_1 越小, 说明算法的收敛性能越好; SP 越小, E 越大, 说明解集的分布性和多样性越好. 对于有序解集对 (X_1, X_2) , $C(X_1, X_2) > C(X_2, X_1)$ 说明集合 X_1 在 C 测度上的性能优于 X_2 .

3 种算法均采用实数编码, 使用模拟二进制交叉算子和多项式变异算子^[13], 交叉概率取 0.9, 变异概率取 $1/n$ (n 为变量的个数). T_1 中最大进化代数 T 取 250, T_2 中 T 取 100. KDMOGA 中, $N = 100, M = 100$, 其余两种算法的群体规模 L 取 100.

将 3 种算法在 T_1 和 T_2 中分别独立运行 10 次, 10 次运行结果的平均值分别如表 1 ~ 表 3 所示(在 T_2 中, 真实的 Pareto 最优解集无法获取, 因此不使用测度 M_1). 图 3 和图 4 分别给出了各算法其中一次运行搜索到的 Pareto 前沿的比较.

表 1 T_1 中 3 种算法在 M_1, SP, E 测度上的比较

	M_1	SP	E
KDMOGA	0.0396	0.0057	0.8256
NSGA-	0.0870	0.0237	0.6969
controlled NSGA-	0.0746	0.0192	0.7346

表 2 T_2 中 3 种算法在 SP 和 E 测度上的比较

	SP	E
KDMOGA	0.0046	0.8789
NSGA-	0.0114	0.7399
controlled NSGA-	0.0098	0.7696

由上述图表可见, 所提出算法 KDMOGA 在 T_1

表 3 T_1 和 T_2 中 3 种算法在 C 测度上的比较

	$C(K, N)$	$C(N, K)$	$C(K, c)$	$C(c, K)$	$C(N, c)$	$C(c, N)$
T_1	0.5314	0.3139	0.4024	0.3696	0.4388	0.4886
T_2	0.3100	0.1071	0.3586	0.0729	0.2943	0.2100

注: K 表示 KDMOGA, N 表示 NSGA-, c 表示 controlled NSGA-.

和 T_2 中的收敛性能 (M_1, C), 分布性能 (SP) 及多样性 (E) 均明显优于已有两种算法, 说明所提出算法采用的基于多样性测度 E 自适应地调整选择操作和搜索机制的策略是可行而有效的. 它能够开拓算法的搜索范围, 增加进化群体的多样性, 从而搜索到区域内的优良个体, 跳出局部最优, 提高算法的收敛性能和分布特性. 由图 3 可见, NSGA- 和 controlled NSGA- 由于排挤距离定义的缺陷, 在搜索到的 Pareto 前沿上出现了若干个不合理的个体(位于 f_1 接近于 0 的位置), 而 KDMOGA 则因为采用改进的策略避免了不合理个体的出现.

5 结 论

本文基于信息熵的概念提出一种多目标空间下群体的多样性测度. 依据该测度自适应地调整算法的选择操作和搜索机制, 从而避免了对精英个体的过度利用, 加大了对新个体的探索力度. 基于上述策略, 提出了一种新型的保持群体多样性的多目标遗传算法, 并对算法的计算复杂度进行了理论分析. 仿真实验表明, 与已有算法相比, 本文算法具有更优的搜索性能, 它能有效地收敛到问题的全局 Pareto 最优前沿, 并能保证解集具有较好的多样性.

参考文献(References)

- [1] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: Empirical results [J]. *Evolutionary Computation*, 2000, 8(2): 173-195.
- [2] Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms[J]. *Evolutionary Computation*, 1994, 2(3): 221-248.
- [3] Knowles J D, Corne D W. The pareto archived evolution strategy: A new baseline algorithm for pareto multiobjective optimization[C]. *Congress on Evolutionary Computation*. Piscataway: IEEE, 1999: 98-105.
- [4] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach [J]. *IEEE Trans on Evolutionary Computation*, 1999, 3(4): 257-271.
- [5] Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength pareto evolutionary algorithm[R]. Zurich: Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, 2001.
- [6] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. *IEEE Trans on Evolutionary Computation*, 2002, 6(2): 182-197.
- [7] 雷德明, 吴智铭. 基于个体密集距离的多目标进化算法 [J]. *计算机学报*, 2005, 28(8): 1320-1326.
(Lei D M, Wu Z M. Crowding-measure based multi-objective evolutionary algorithm [J]. *J of Computers*, 2005, 28(8): 1320-1326.)
- [8] Farhang-Mehr A, Azarm S. Diversity assessment of pareto optimal solution sets: An entropy approach[C]. *Proc of the 2002 Congress on Evolutionary Computation*. Piscataway: IEEE, 2002: 723-728.
- [9] 陈昌巨, 武秀文. 求解多目标优化问题基于相对熵的 Pareto 解演化算法 [J]. *华中农业大学学报*, 2003, 22(1): 65-69.
(Chen C J, Wu X W. A relative entropy-based Pareto evolutionary algorithm for multi-objective optimization [J]. *J of Huazhong Agricultural University*, 2003, 22(1): 65-69.)
- [10] Deb K, Goel T. Controlled elit non-dominated sorting genetic algorithms for better convergence[C]. *1st Int Conf on Evolutionary Multi-criterion Optimization*. Zurich: Springer-Verlag, 2001: 67-81.
- [11] Deb K, Pratap A, Moitra S. Mechanical component design for multiple objectives using elitist non-dominated sorting GA[R]. Kanpur: Indian Institute of Technology, 2000.
- [12] Coello C A, Sierra M R. A coevolutionary multi-objective evolutionary algorithm[C]. *Proc of the 2003 Congress on Evolutionary Computation*. Australia: IEEE, 2003: 482-489.
- [13] Deb K, Agrawal R B. Simulated binary crossover for continuous search space[J]. *Complex Systems*, 1995, 9(2): 115-148.
- (上接第 1434 页)
- [3] Bliman P A. A convex approach to robust stability for linear systems with uncertain scalar parameters [J]. *SIAM J of Control Optimization*, 2004, 42(6): 2016-2042.
- [4] Oliveira Ricardo C L F. LMI conditions for robust stability analysis based on polynomially parameter-dependent Lyapunov functions[J]. *Systems and Control Letters*, 2006, 55(2): 52-61.
- [5] Boyd S, Ghaoui E L, Feron E, et al. *Linear matrix inequalities in system and control theory* [M]. New York: SIAM, 1994.
- [6] Oliveira M De. A new discrete-time robust stability condition[J]. *Systems and Control Letters*, 1999, 37(3): 261-265.
- [7] Xie L H, Lu L, Zhang D. Improved robust H_2 and H_∞ filtering for uncertain discrete-time systems [J]. *Automatica*, 2004, 40(6): 873-880.
- [8] Duan Z S, Zhang J X, Zhang C S. Robust H_2 and H_∞ filtering for uncertain linear systems [J]. *Automatica*, 2006, 42(11): 1919-1926.
- [9] Geromel J C, Korogui R H. Analysis and synthesis of robust control systems using linear parameter dependent Lyapunov functions [J]. *IEEE Trans on Automatic Control*, 2006, 51(12): 1984-1989.
- [10] Iwasaki T, Meinsma G, Fu M Y. Generalized S-procedure and finite frequency KYP lemma [J]. *Mathematical Problems in Engineering*, 2000, 6(3): 305-320.
- [11] Skelton R E, Iwasaki T, Grigoriadis K. *A unified approach to linear control design* [M]. New York: Taylor and Francis, 1997.
- [12] Scherer C W. LPV control and full block multipliers [J]. *Automatica*, 2001, 37(3): 361-375.
- [13] Cao Y Y, Lin Z. A descriptor system approach to robust stability analysis and controller synthesis [J]. *IEEE Trans on Automatic Control*, 2004, 49(11): 2081-2084.