

文章编号: 1009-4741 (2002) 04-0243-04

机构设计的遗传算法约束条件处理

竺志超

(浙江工程学院机械与自动控制学院 杭州 310033)

摘要: 将机构设计的约束条件分成边界约束和性能约束, 分别讨论在遗传算法中的约束处理方法, 特别提出改进惩罚策略, 使 GA 更有效、更普遍地处理带非线性约束的机构设计数值优化问题。

关键词: 机构设计; 遗传算法; 约束条件; 惩罚函数法

中图分类号: TH122 **文献标识码:** A

0 引言

机构设计优化属于非线性约束优化问题, 并且它的目标函数复杂, 采用传统的算法往往搜索不到全局最优解。遗传算法 (简称 GA) 不要求目标函数和约束是可微的, 计算效率比较高, 搜索结果的最好个体在概率意义上代表全局最优解, 因此, 这一算法在机构设计领域开始逐渐得到了应用。本文探讨遗传算法中约束条件的处理方法, 目的在于将 GA 更好地应用于机构设计。

1 机构优化设计策略

机构优化设计模型通常表示为

$$\begin{aligned} \min f(x) \quad & x \in R^n \\ \text{s. t. } g_u(x) & \leq 0 \quad u=1, 2, \dots, p \\ h_v(x) & = 0 \quad v=1, 2, \dots, q \end{aligned} \tag{1}$$

其中约束分为两类, 一类是边界约束, 由组成设计变量 x 的 n 个参数的允许变化的范围构成; 一类是性能约束, 如传动角要求、曲柄存在条件、机构成立几何条件、强刚度条件等等构成。当目标函数 $f(x)$ 代表着某种约束时, 上述模型并不是追求机构性能最佳, 而仅是一种数值设计方法, 即通过搜索寻找满足所有约束条件的机构设计方案。例如连杆机构运动轨迹设计就是如此, 因为可行的方案本来就不多, 按常规的方法, 设计变得很困难, 此时往往采用优化方法来实现。

机构设计优化最大特点是问题的非线性及多峰, 有效的解决方法是遗传算法。通常遗传算法用于求最大问题。因此, 在将目标函数值转换为适值前必须先将最小化目标函数转换为最大化目标函数, 并通过加法机制进行调整:

$$F(x) = C - f(x) \rightarrow \max \tag{2}$$

式中 C 为一正常数 (计算时取群体中的最大适值或某一足够大的数), 以便保证遗传算法搜索过程中目标函数值始终大于零。遗传算法中的染色体即为设计变量 x , 求解过程一般直接采用浮码表达, 每个染

染色体编码为一个和解向量维数相同的实向量,以避免二进制编码和解码过程,提高算法的运行效率。因为带有非线性约束条件,常采用惩罚策略,所以构造适值评价函数:

$$\text{eval}(x) = F(x) \cdot p(x) \quad (3)$$

其中 $p(x)$ 为自适应的惩罚函数,违反度越大惩罚越重。它类似于传统优化的惩罚函数法,通过惩罚不可行解将约束问题转化为无约束问题来处理。这样允许在每代的群体中保持部分不可行解,使遗传搜索可以从可行域和不可行域两侧来达到最优解。

2 基于约束分类的约束处理方法

约束条件处理是优化设计的关键,目前遗传算法对于违反约束的处理主要有四种^[1]。死亡惩罚策略,对于一些很难通过一般遗传因子产生可行解的问题,算法耗费大量的机时去评价非法个体;修复策略用特殊的修补算法来校正所有产生的不可行解,只对特定问题而言,同样耗费大量的计算;改进遗传算子策略通过设计专门的遗传算子来维持染色体的可行性。上述3种策略无法考虑可行域外的点,故又有常见的惩罚策略。这里提出基于约束分类的约束处理方法,即对于等式约束解决方案是降维法,使设计变量 x 的维数降低,并且消去等式约束,简化优化时约束条件的处理,至于不等式约束按边界约束和性能约束分别讨论如下。

2.1 边界约束的处理方法

如初始群体中各染色体依据设计参数的边界 (x_k^l, x_k^u) 约束随机产生,即取染色体 x 的基因为:

$$x_k = x_k^l + \text{rand}() \cdot (x_k^u - x_k^l); \quad k = 1, 2, \dots, n \quad (4)$$

其中随机函数 $\text{rand}()$ 取在区间 $[0, 1]$ 内的随机数。显然,这样产生的染色体均满足边界约束条件。那么遗传进化操作时,可以采用算术交叉算子和非均匀变异算子,使产生的后代仍满足边界约束条件,从而在惩罚函数中忽略对边界约束的惩罚。

理由是如采用算术交叉,设父代染色体 x_1, x_2 , 则交叉后产生的子代染色体为

$$\begin{aligned} x'_1 &= \lambda x_1 + (1 - \lambda) x_2 \\ x'_2 &= \lambda x_2 + (1 - \lambda) x_1 \end{aligned} \quad (5)$$

式中 λ 是 $(0, 1)$ 内的随机数。这种交叉是凸交叉,其特点为如父代 x_1, x_2 均属于凸集,则子代 x'_1, x'_2 也均属于该凸集。又采用非均匀变异,那么对于选定的父代 x , 如其基因 x_k 被选作变异,则生成的后代为

$$x = [x_1, x_2, \dots, x'_k, \dots, x_n] \quad (6)$$

其中 x'_k 随机地按如下两种可能的机会变换

$$\begin{aligned} x'_k &= x_k + \Delta(t, x_k^u - x_k) \quad \text{如随机数为 } 0; \\ x'_k &= x_k - \Delta(t, x_k - x_k^l) \quad \text{如随机数为 } 1; \end{aligned} \quad (7)$$

这里, x_k^u 和 x_k^l 分别是 x_k 的上下界。这种变异的特点是如父代在上下界域内,则变异产生的后代也在该域内。可见因交叉和变异前父代是满足边界约束的,而边界约束形成的空间是凸集,故通过上述所采用的交叉和变异算子产生的后代也必然满足边界约束。因此,这样处理把边界约束从约束中分离了出来,使整个搜索过程中可以不考虑边界约束而只需惩罚不满足性能约束的染色体即可。

2.2 性能约束的处理方法

性能约束中有一类机构存在约束条件,如不满足表示机构不成立,以致目标函数 $f(x)$ 本身无法计算,例如对于曲柄摇杆机构设计,机构成立条件可表示对于曲柄任意转角,在机构运动计算中有关式的根号内值必须大于零。因此,对于不满足机构存在约束条件的染色体,因机构不存在,故只能采用拒绝策略,即重新产生新的染色体代之。

至于其他性能约束不满足时采用惩罚策略,惩罚函数的设计各有不同^[2]。其中合理的惩罚因子取值是非常困难的,与讨论的问题有关,一般通过试验获得。有时调整不好,一些约束不满足的个体其适值比满足约束的其他个体适值还好,以致搜敛到可行域外。有特色的是 *Pavell*^[3] 等提出附加的惩罚项,对不可行个体惩罚增加,使它们的适值不好于可行个体中最差值。对于机构设计往往在约束界面上取得最优点,如图 1 所示,这样处理任何不可行个体都会丢失很多有用信息。文献[4] 也针对这一问题给出了一种算法。这里提出新的改进惩罚策略,是基于这样的思想:对于目标函数值好于可行域内任一点的不可行点如 x_i ,有可能由它们搜索到约束界面上最优点 x_0 ,因此这样的点更有价值,惩罚应小些;同样两个在可行域外的这样点,离约束界面越近越优。因此建立适值函数如下:

$$\text{eval}(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}) & \boldsymbol{x} \text{ 点在可行域内时;} \\ F(\boldsymbol{x}) p(\boldsymbol{x}) & \boldsymbol{x} \text{ 越界且 } F(\boldsymbol{x}) < F_{\max}; \\ \beta F_{\max} P(\boldsymbol{x}) & \boldsymbol{x} \text{ 超界且 } F(\boldsymbol{x}) \geq F_{\max}; \end{cases} \quad (8)$$

式中参数 β 为一常数, F_{\max} 为到目前为止可行域内最大目标函数值。为使惩罚少些,取 $\beta \geq 1$; 这样使可行域外、但目标函数值好的点惩罚小,搜索时最有效地保留了不可行解,即保留了更多搜索有用的信息,有利于算法的搜敛。惩罚函数 $p(\boldsymbol{x})$ 仍然按原先保证超界越多惩罚越大的原则建立,使搜索向约束界面移动。

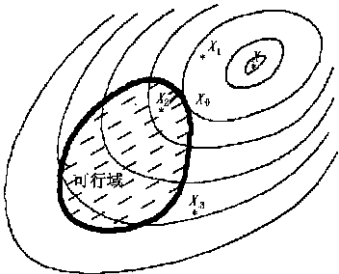


图 1 群体中个体散布
 X : 无约束最优点; X_0 : 约束最优点; X_1 : 可行域外较好点; X_2 : 当前可行域内最好点; X_3 : 可行域外较差点

3 设计实例

这里,以企业委托的某剑杆织机上采用的一种空间六杆引纬机构^[5]设计为例。剑头进出梭口时空动程 S_j 、 S_c 及相应时间角、总动程 S_m 和剑头进足交接纬时间角 Φ_m 、剑头最大加速度等是最重要的设计参数,除最大加速度影响动力学性能需要控制外,其他由织造工艺要求确定,这些和机构成立等其他条件一起构成性能约束。该机构有 8 个结构设计参数,其中各个参数对应有一定的结构所允许的边界约束条件 $x_i^l < x_i < x_i^u$ ($i = 1, 2, \dots, 8$)。因连杆机构的特性限制,剑头引纬规律不能随意设计,但是,可综合 8 个结构参数,以获得能满足织造工艺要求的剑头运动规律。所以,建立该引纬机构优化设计模型:在满足上述工艺要求的前提下,尽可能降低剑头的最大加速度 S_{\max} ,以减缓引纬机构的受力,减少整机的振动。然后按上述约束处理方法采用 *GA* 算法,遗传进化时采用精华模型,选择下一代新群体时采用比例选择,群体规模 $\text{popsize} = 40$,最大进化代数 $T = 100$,交叉概率 $p_c = 0.8$,变异率 $p_m = 0.15$ 。搜索结果如表 1 所示,当取 $\beta = 1.1$ 时剑头最大加速度为最小,好于惩罚策略改进前的搜索结果,且是在性能约束的 3 个界面上获得的,而改进前的搜索结果只落在一个界面上。根据优化方案生成的机构,其剑头引纬规律完全符合设计要求。这说明提出的改进惩罚策略是行之有效的。

表 1 两种惩罚策略搜索结果

工 艺 条 件(性 能)	惩罚策略	惩罚策略改进后				
	约束范围	改进前	$\beta= 0.5$	$\beta= 0.8$	$\beta= 1.0$	$\beta= 1.1$
$S_j= 0.300\sim 0.316\text{mm}$	0.316	0.315	0.315	0.316	0.316	0.306
$S_c= 0.300\sim 0.315\text{mm}$	0.311	0.314	0.311	0.314	0.315	0.312
$S_m= 1.440\sim 1.450\text{mm}$	1.448	1.441	1.440	1.442	1.440	1.418
$\Phi_m= 185^{\circ}\sim 195^{\circ}$	191	190	190	190	190	191
优化目标 \ddot{S}_{\max}	820.906	810.762	810.052	808.579	806.308	801.49
搜索结果点位置	1个界面上	可行域内部	1个界面上	1个界面上	3个界面上	可行域外

比较上表可知, β 的取值影响搜索结果, 取值较小时表示惩罚大, 则易搜索到可行域内部, 取值较大时表示惩罚小, 则易搜索到可行域外, 而对于诸如机构设计这样往往在性能约束界面上获得最优点的问题, 应取一合理的值。此外, 出于加快搜敛速度的考虑, β 可设计为一函数, 开始时取大些, 结束时取小些, 以便可以从可行域外搜敛到界面上。

4 结 论

约束条件处理是用 GA 求解机构约束优化设计的关键。由上述实际应用结果表明: a) 在遗传算法中初始群体在设计参数的上下边界范围内产生, 经所采用的遗传算子保证产生的后代仍满足边界约束, 因此惩罚时少去了边界约束引起的惩罚, 使惩罚函数更为简单, 提高优化计算效率。b) 提出的改进惩罚策略, 使遗传算法更有效、更普遍地处理带非线性约束的数值优化问题。这些优化策略, 可以简化算法过程, 加快 GA 搜索的收敛速度。

参考文献:

- [1] Michalewicz Z et al. Evolutionary algorithms for constrained engineering problems [J]. Computer ind. Engng., 1996, 30 (4).
- [2] 玄光男, 程润伟, 汪定伟. 唐加福、黄敏译. 遗传算法与工程设计 [M]. 北京: 科学出版社, 2000. 39—42.
- [3] Z 米凯利维茨, 周家驹、何险峰译. 演化程序 [M]. 北京: 科学出版社, 2000. 107.
- [4] 林丹、李敏强、冠纪淞. 基于遗传算法求解约束优化问题的一种算法 [J]. 软件学报, 2001, 12 (4): 628—632.
- [5] 竺志超, 李志祥. 引纬机构的遗传算法优化设计 [J]. 纺织学报, 2002, (3): 34—35.

A Constraint Handling in Mechanism Design Based on Genetic Algorithms

ZHU Zhi-chao

(College of Mechanical Engineering & Automation, zhejiang Institute of Science and Technology, Hangzhou 310033, China)

Abstract: Classifies the constrains into the boundary and performance constraints, discusses the handling method of the constraint in using genetic algorithms, proposes a new penalty tactic that makes GA to deal with numerical optimization subject in mechanism design with nonlinear constraints more effectively and easily.

Key words: Mechanism design; Genetic algorithms; Constraints; Penalty

(责任编辑: 杨元兆)