

一种快速遗传算法及其收敛性

马永杰^{1,2}, 马义德³, 蒋兆远¹, 孙启国¹

(1. 兰州交通大学机电技术研究所, 甘肃 兰州 730070; 2. 西北师范大学物理与电子工程学院, 甘肃 兰州 730070; 3. 兰州大学信息科学与工程学院, 甘肃 兰州 730000)

摘要: 针对遗传算法局部搜索能力较差、全局优化速度缓慢、搜索的效率对约束惩罚因子的选择有明显依赖性等问题, 设计了能够从可行解空间和不可行解空间同时搜索的交叉算子、具有前期的快速搜索能力和后期的最优解保持能力的变异算子和具有“精英”保持能力的选择算子, 并采用已搜索解集避免了子代的“返祖”和退化现象。在此基础上, 设计了具有全局搜索能力的遗传算法, 并证明了算法的收敛性。仿真结果表明, 该算法能够快速找到全局极值点。

关键词: 遗传算法; 收敛性; 快速收敛; 约束优化

中图分类号: TP 18; TP 301.6

文献标志码: A

Fast genetic algorithm and its convergence

MA Yong-jie^{1,2}, MA Yi-de³, JIANG Zhao-yuan¹, SUN Qi-guo¹

(1. Inst. of Mechatronics Technology, Lanzhou Jiaotong Univ., Lanzhou 730070, China;

2. Coll. of Physics and Electronic Engineering, Northwest Normal Univ., Lanzhou 730070, China;

3. Coll. of Information Science & Engineering, Lanzhou Univ., Lanzhou 730000, China)

Abstract: Aiming at problems of genetic algorithm, such as a worse local search, a slower global optimization, and search efficiency depending on the selection of penalty function obviously, a crossover operator which searches from both feasible and infeasible solution space simultaneously, a mutation operator which can rapidly search prophase and hold global optimal solution anaphase, and a selection operator which can hold optimal solution are designed. Moreover, the reversion and degradation are avoided via using the searched solution space. Based on all these, a novel effective genetic algorithm for global optimization is proposed and its global convergence is proved. At last, the simulation result shows that this algorithm can rapidly find the global extremum point.

Keywords: genetic algorithm; convergence; fast convergence; constrain optimization

0 引言

遗传算法是解复杂优化问题的一种新型有效方法, 具有全局收敛性、鲁棒性、简单通用等优点, 但局部搜索能力较差, 全局优化速度缓慢, 易出现早熟收敛等问题。对于约束的处理方法目前主要采用惩罚函数法^[1-3], 但优化搜索的效率对惩罚因子的选择有明显的依赖性, 同时, 惩罚因子没有统一的选择标准, 使得惩罚因子选择非常困难。但若采用拒绝策略, 完全拒绝不可行解则会大大减少搜索范围, 很难收敛到最优解。基于惩罚函数的方法, 选择是对可行解与不可行解的混合种群进行的。如果种群中只有可行解, 就不会出现惩罚因子的问题, 从而将会避免进行惩罚因子的选择。

文中, 设计了一种既利用不可行解扩大搜索范围, 又不引入惩罚因子的约束处理方法^[3]; 设计了一种具有“记忆”能力的交叉算子, 阻止了个体的“返祖”现象, 避免了重复搜索, 提高了搜索效率; 设计了搜索步长和方向可变的变异算子, 保证了算法在前期的快速搜索能力和后期的最优解的保持能力; 采用“精英”保持策略, 将父代的最优个体合并到子代, 使算法具有较强的鲁棒性。

考虑如下全局优化问题

$$\begin{aligned} \min f(x) \\ \text{s.t. } L \leq x \leq U \end{aligned} \quad (1)$$

式中, $x = (x_1, \dots, x_m) \in D$, $D \subseteq R^m$ 为搜索空间, $f: R^m \rightarrow R$ 为目标函数, $L = (l_1, l_2, \dots, l_m)$, $U = (u_1, u_2, \dots, u_m)$, $[L, U] = \{(x_1, x_2, \dots, x_m) | l_i \leq x_i \leq u_i, i = 1, 2, \dots, m\}$ 表示

收稿日期: 2007-10-16; 修回日期: 2008-02-20.

基金项目: 国家自然科学基金(60572011); 甘肃省科技基金(2GS066-A52-001-04)资助课题

作者简介: 马永杰(1967-), 男, 副教授, 博士研究生, 主要研究方向为软件技术、智能控制。E-mail: myjmyj@163.com

可行解空间, $[L, U] \subseteq D, Q = D \setminus [L, U]$ 为不可行解空间。

定义 1^[4] 一个向量 $u = (u_1, u_2, \dots, u_m)$ 称为非劣于 (Dominated) 向量 $v = (v_1, v_2, \dots, v_m)$, 当且仅当对于 $\forall i \in \{1, 2, \dots, m\}, u_i \leq v_i \wedge \exists i \in \{1, 2, \dots, m\}$, 使得 $u_i < v_i$ 。

1 遗传算法设计

本文采用实数编码, 用向量 $x = (x_1, \dots, x_m)^T$ 作为一个个体 (或染色体) 来表示问题 (1) 的一个解。记 $D_0 = \{x | x \in D\}$ 为已搜索空间, 记 $D_3 = \{x | x \in D\}$ 为交叉子集, 记 $D_4 = \{x | x \in D\}$ 为变异子集, 算法开始赋 $D_0 = \Phi, D_3 = \Phi, D_4 = \Phi$ 。

1.1 初始种群

在可行解空间 $[L, U]$ 上, 随机生成 pop 个均匀分布的初始个体 $x^i(0) (i=1, 2, \dots, pop)$; 在不可行解空间 Q 上, 随机生成 N_2 个均匀分布的初始个体 $x^i(0) (i=pop+1, \dots, pop+N_2)$ 。记 $D_1 = \{x^i | i=1, 2, \dots, pop\}$ 为种群集, pop 为种群规模, $D_1 \subseteq [L, U], D_2 = \{x^i | i=1, 2, \dots, N_2\}$ 为不可行解集, N_2 为不可行解数量, $D_2 \subseteq Q$, 且 $pop \gg N_2$ 。赋 $D_0 = D_0 \cup D_1 \cup D_2$ 。

1.2 适应度函数

由于所讨论的问题 (1) 是最小化问题, 因此设定适应度越小, 结果越好。

设适应度函数 $fit(x)$ 表示为

$$fit(x) = f(x) \quad (2)$$

1.3 交叉算子

设 $r^1, r^2 \in D_1 \cup D_2, p_c$ 是交叉概率, t 为迭代次数, 生成一个随机数 $\lambda_1 \in [0, 1]$, 若 $\lambda_1 \leq p_c$, 则

(1) 生成一个随机数 $\lambda_2 \in [0, 1]$, 若 $\lambda_2 \leq N_2 / (pop + N_2)$, 从 D_2 中随机选取一个个体 $x^i(t)$, 并使 $r^1 = x^i(t) (l=1, 2; i=1, 2, \dots, pop, pop+1, \dots, pop+N_2)$; 若 $\lambda_2 > N_2 / (pop + N_2)$, 从 D_1 中随机选取 $x^j(t)$, 并使 $r^1 = x^j(t) (l=1, 2; j=1, 2, \dots, pop, pop+1, \dots, pop+N_2)$;

(2) 重复 (1), 直到获得两个个体 r^1, r^2 。

然后由式 (3) 产生后代

$$\begin{cases} c^1 = \lambda_1 r^1 + (1 - \lambda_1) r^2 \\ c^2 = (1 - \lambda_1) r^1 + \lambda_1 r^2 \end{cases} \quad (3)$$

若 $r^1, r^2 \in D_1$, 则在可行解空间搜索, 若 $r^1 \in D_1, r^2 \in D_2$, 则在可行解空间与不可行解空间之间, 即约束的边界上搜索, 使算法从可行解空间内部和可行解空间边界两个方向的搜索, 从而保证了算法的遍历性, 如图 1 所示。

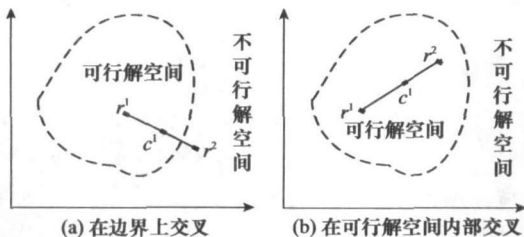


图 1 算术交叉

由于 λ_1 是一个随机值, 交叉后的新的个体可能存在着“返祖”现象, 这将影响算法的效率, 所以对于交叉得到的 pop 个新的个体 $c^i (i=1, 2, \dots, pop)$:

(1) 若 $c^i \in D_0$, 则抛弃;

(2) 若 $c^i \notin D_0$, 则 $D_3 = D_3 \cup \{c^i\}$ 。

最后得到具有 $N_3 (N_3 \leq pop)$ 个个体的交叉子集 D_3 , $D_3 = \{c^i | c^i \in D; c^i \notin D_0; i=1, 2, \dots, N_3\}$ 。

1.4 变异算子

设 $z^i \in D, p_m$ 是变异概率, G_{MAX} 为最大迭代次数, f_{best} 为 D_3 中最好个体的适应度, f_{max} 为 D_3 中最差个体的适应度。生成一个随机数 $\lambda_3 \in [0, 1]$, 若 $\lambda_3 \leq p_m$, 则从 D_3 中随机选取 $c^i (i=1, 2, \dots, N_3)$, 生成一个随机数 $\lambda_4 \in [0, 1]$, 并按下式产生变异后代

$$z^i = \begin{cases} c^i \left(1 + \frac{f(c^i)}{f_{max} - f_{best}} \right) e^{-t}, & \text{当 } \lambda_4 \leq 0.5 \text{ 且 } f(z^i) < f(c^i) \\ c^i \left(1 - \frac{f(c^i)}{f_{max} - f_{best}} \right) e^{-t}, & \text{当 } \lambda_4 > 0.5 \text{ 且 } f(z^i) < f(c^i) \\ \text{抛弃 } z^i, & \text{当 } f(z^i) > f(c^i) \end{cases} \quad (4)$$

式 (4) 保证了变异操作从 $x^i(t) + \Delta x^i(t)$ 和 $x^i(t) - \Delta x^i(t)$ 两个方向的搜索, 并选择适应值较小的方向作为搜索方向; 当变异操作不能保证适应值减小时, 放弃变异操作, 采取精英保持策略, 维持父代个体不变, 避免了子代的退化; 适应值较小的个体以较小的步长 $\Delta x^i(t)$ 搜索; 在搜索的前期, 采取较大的搜索步长, 而在搜索的后期, 采取较小的搜索步长 $x^i(t) (1 + \Delta x^i(t)) e^{-t}$, 使个体向最优解收敛。

对于变异得到的 pop 个新的个体 $z^i (i=1, 2, \dots, pop)$:

(1) 若 $z^i \in D_0$, 则抛弃;

(2) 若 $z^i \notin D_0$, 则 $D_4 = D_4 \cup \{z^i\}$ 。

最后得到具有 $N_4 (N_4 \leq pop)$ 个个体的变异子集 D_4 , $D_4 = \{z^i | z^i \in D; z^i \notin D_0; i=1, 2, \dots, N_4\}$ 。

1.5 选择算子

合并父代种群 $D_1 \cup D_2$ 、交叉子集 D_3 、变异子集 D_4

$$D_5 = D_1 \cup D_2 \cup D_3 \cup D_4 \quad (5)$$

得到临时种群 D_5

$$D_5 = \{x^i | i=1, 2, \dots, pop + N_2 + N_3 + N_4\}$$

式 (5) 采取精英保持策略, 将父代合并到子代, 避免了子代的退化; 将交叉和变异操作产生的新个体合并到子代, 提高种群的多样性。

在种群 D_5 中, 计算个体的适应度, 按适应度的升序排列, 并按式 (6) 选择前 pop' 个个体, 得到子代种群

$$D_1 = \{x^i(t+1) | i=1, 2, \dots, pop'\};$$

$$t=1, 2, \dots, G_{MAX}; x^i(t+1) \in [L, U]; x^i(t+1) \notin Q \quad (6)$$

如果 $pop' < pop$, 再从 $[L, U]$ 中随机产生 $pop - pop'$ 个个体, 把这两部分个体合起来作为可行解集 D_1 。

在种群 D_5 中, 按式 (7) 选择前 N' 个个体, 得到不可

行解集

$$D_2 = \{x^i(t+1) \mid (i = pop + 1, pop + 2, \dots, pop + N'_2; t = 1, 2, \dots, G_{MAX}; x^i(t+1) \notin [L, U]; x^i(t+1) \in Q\} \quad (7)$$

如果 $N'_2 < N_2$, 再从 Q 中随机产生 $N_2 - N'_2$ 个个体, 把这两部分个体合起来作为不可行解集 D_2 。

最后得到具有 $pop + N_2$ 个个体的下一代种群 $D_1 \cup D_2$ 。

1.6 终止规则

若算法达到了最大进化代数 G_{MAX} 或 $D_3 \cup D_4 = \Phi$ (或 $N_3 + N_4 = 0$), 则停止, 所得最好解集 D_1 作为近似全局最优解。

2 算法流程

步骤 1 初始化 确定杂交概率 p_c , 变异概率 p_m , 种群规模 pop, N_2 , 最大进化代数 G_{MAX} 。置 $D_0 = \Phi, D_3 = \Phi, D_4 = \Phi$, 产生初始种群 $x^i(0), D_0 = D_0 \cup D_1 \cup D_2$ 。对每一个个体 $x^i(0)$, 按式(2)计算适应度;

步骤 2 交叉 按 1.3 节方法产生交叉子集 D_3 ;

步骤 3 变异 按 1.4 节方法产生变异子集 D_4 ;

步骤 4 选择 按 1.5 节给出的选择算子, 在当前临时种群 D_5 中选出具有 $pop + N_2$ 个个体的下一代种群 $D_1 \cup D_2$ 。

步骤 5 判断 若满足终止条件, 则停止, 输出保留的最好解作为问题(1)的近似全局最优解; 否则, 令 $t = t + 1, D_3 = \Phi, D_4 = \Phi$, 转步骤 2。

3 算法的收敛性分析^[5-8]

定义 2 称个体 x' 是从 x 通过杂交和变异可达的, 如果 $P\{MC(x) = x'\} > 0$, $MC(x)$ 表示由 x 通过杂交和变异算子产生的点, $P\{\cdot\}$ 表示随机事件 $\{\cdot\}$ 的概率。

定义 3 若 $P\{\|MC(x) - x'\|_\infty \leq \epsilon\} > 0, \|\cdot\|_\infty$ 是 R^m 上定义的范数, $\epsilon > 0$ 是给定的任意小正数, 则称 x' 是从 x 通过杂交和变异为 ϵ -精度可达的。

引理 1 若一个进化算法(EAs)满足下面两个条件^[9]:

(1) 对可行域中任意两个点 x' 和 x, x' 是从 x 通过杂交和变异可达的;

(2) 种群序列 $p(1), p(2), \dots, p(t)$ 是单调的, 即对 $\forall t, p(t+1)$ 中任意解非劣于 $p(t)$ 中任意解或者至少不差于 $p(t)$ 中任意解。则 EAs 以概率 1 收敛到问题的全局最优解集。

若将引理 1 中的第一个条件“可达”改为“ ϵ -精度可达”, 则和文献[10]第二章中定理 4 的证明完全一样, 可证明进化算法 EAs 以概率 1 收敛到问题具有 ϵ -精度的最优解集。

定理 1 对充分小的 $\epsilon > 0$, 若 $f(x)$ 在搜索空间 D 上连续, 则本文的算法以概率 1 收敛到问题(1)具有 ϵ -精度的最优解集。即

$$P\{\lim_{t \rightarrow \infty} \|p(t) - p_{true}\|_\infty \leq \epsilon\} = 1 \quad (8)$$

式中, p_{true} 为问题(1)的 Pareto 最优解集, $\|p(t) - p_{true}\|_\infty \leq \epsilon$ 表示 $\{\|x - x^*\|_\infty \leq \epsilon \mid \forall x \in p(t), \exists x^* \in p_{true}\}$ 。

证明 首先证对任意两个点 x' 和 $x \in D, x'$ 是从 x 通过杂交和变异为 ϵ -精度可达, 即

$$P\{\|MC(x) - x'\|_\infty \leq \epsilon\} > 0 \quad (9)$$

式中, $MC(x)$ 为 x 通过杂交和变异算子产生的点。设 \bar{x} 是 x 通过杂交算子产生的任意点, 即 $C(x) = \bar{x}$ 。由式(4)知, 当 x' 采用式(4)的变异算子变为 $M(\bar{x})$ 时, 总有

$$\lim_{t \rightarrow \infty} |c^i - z^i| = \lim_{t \rightarrow \infty} \left| \frac{f(c^i)}{f_{max} - f_{best}} e^{-t} \right| \leq \epsilon \quad (10)$$

即

$$P\{\|M(\bar{x}) - x'\|_\infty \leq \epsilon\} > 0 \quad (11)$$

式(11)证明, x' 是 \bar{x} 通过变异算子为 ϵ -精度可达, 故 x' 是由 x 通过杂交和变异为 ϵ -精度可达的。

由变异算子和选择算子的最优解保持策略知, 由本算法产生的子代 $p(t+1)$ 中任意解一定非劣于 $p(t)$ 中的解, 或改进了 $p(t)$ 中的解 (至少不劣于 $p(t)$ 中的解), 于是 $p(1), p(2), \dots, p(t)$ 是单调的。由文献[10]中的第二章的定理 4 知, 式(8)成立, 即本算法以概率 1 收敛到问题(1)的具有 ϵ -精度的最优解集 p_{true} 。证毕

4 算法测试及仿真结果

4.1 测试函数

本文对 3 个标准函数进行了测试, 并和 3 种已有算法的数值结果进行了比较, 关于测试函数的解析性质及所比较的算法见文献[11], 所用的测试函数如下, 其中 n 为问题的维数

$$F1 : f_1(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

式中, $-5.12 \leq x_i \leq 5.12, i = 1 \sim 5, n = 5$, 最优值为 $f_1^* = 0$ 。

$$F2 : f_2(x) = -20 \exp \left[-0.2 \sqrt{\frac{1}{30} \sum_{i=1}^{30} x_i^2} \right] - \exp \left[\frac{1}{30} \sum_{i=1}^{30} \cos 2\pi x_i \right] + 20 + e$$

式中, $-32 \leq x_i \leq 32, i = 1 \sim 30, n = 30$, 最优值为 $f_2^* = 0$ 。

$$F3 : f_3(x) = \frac{1}{4000} \sum_{i=1}^{30} x_i^2 - \prod_{i=1}^{30} \cos \left[\frac{x_i}{\sqrt{i}} \right]$$

式中, $-600 \leq x_i \leq 600, i = 1 \sim 30, n = 30$, 最优值为 $f_3^* = 0$ 。

4.2 仿真结果

本文算法记为 FGA, 在计算中, 取种群规模为 $pop = 100, N_2 = 20$, 交叉概率 $p_c = 0.8$, 变异概率 $p_m = 0.3$, 最大迭代次数 $G_{MAX} = 30$, 对每个函数独立运行 50 次, 记录所得最好函数值的平均值 (记作 Mean.B)、标准差 (记作 Std.D) 及平均迭代次数 (记作 Num), 并和文献[5]中方法 NEP、文献[12]中

方法 FEP、CEP 的已有计算结果进行比较, 结果见表 1。

本文算法的性能跟踪曲线如图 2 所示。

表 1 4 种方法的比较

f		FGA	NEP	FEP	CEP
f_1	$D=[-7, 7]$	Num	10, {5}, [5]	400	400
	$[L, U] = [-5.12, 5.12]$	Mean. B	5.433 9E-11	0.033 2	4.08
		Std. D	1.115 4E-10	0.031 9	3.08
		Num	15, {9}, [13]	1 500	1 500
f_2	$D=[-35, 35]$	Mean. B	1.433 9E-15	1.32E-6	1.8E-2
	$[L, U] = [-32, 32]$	Std. D	8.436 6E-14	1.37E-12	2.1E-3
		Num	30, {5}, [6]	328	2 000
		Mean. B	1.111 8E-131	0.004 1	1.6E-2
f_3	$D=[-700, 700]$	Std. D	5.006 1E-131	5.36E-5	2.2E-2
	$[L, U] = [-600, 600]$				0.12

注: FGA 的 Num 栏的 {5}, [5] 分别表示当 Mean. B、Std. D 的值小于 NEP 对应值时的迭代次数

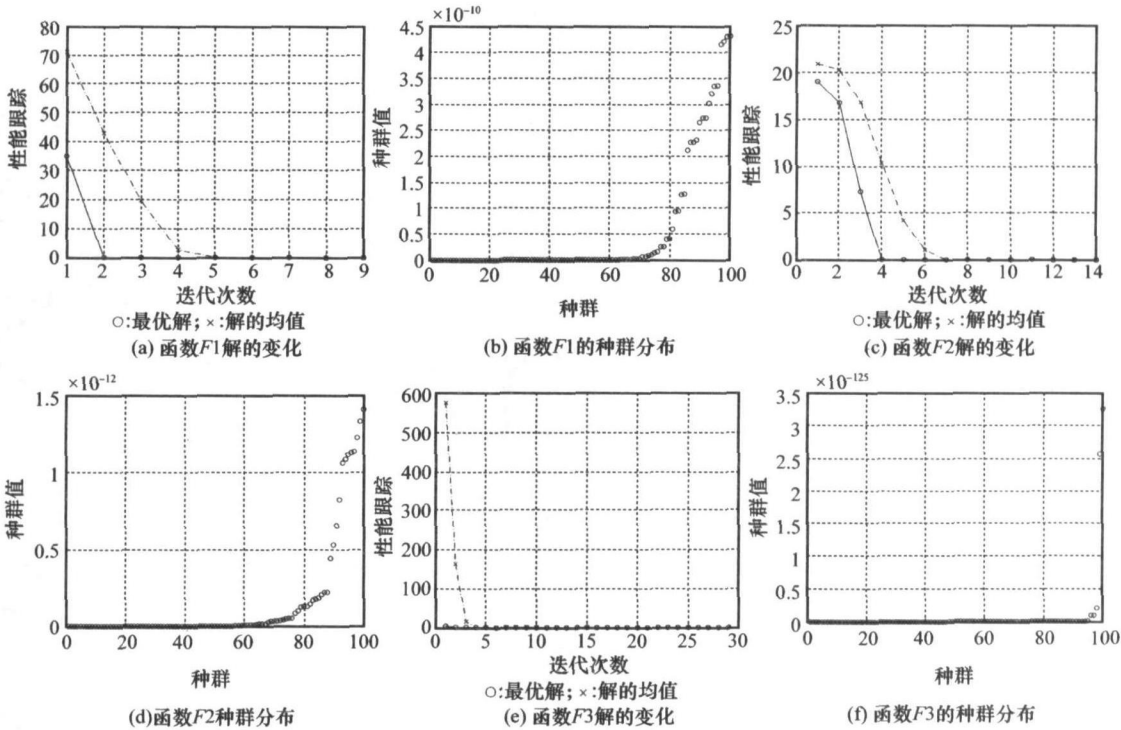


图 2 性能跟踪

4.3 结果讨论

由表 1、图 2 可以明显地看到, 与文献[5, 12] 中 3 种方法相比, 本文的算法能更迅速找到更精确的全局极值点, 这一特点是基于算法自身的设计。文献[13-14] 的评估方法, 以函数 F_3 为例, 通过多次计算取平均值, 可以得到本算法为 1 阶线性收敛算法, 收敛系数为 $0.016\ 6 \ll 1$, 所以保证算法能以较快的速度收敛。

本文采用的精英保持策略和非均匀变异策略, 保证了算法的快速收敛, 但可能造成算法陷入局部收敛。1.3 节中的基于边界搜索的算术交叉算子及其“记忆”能力保证总是能够找到新的解并使之参与选择, 因此有效地阻止了局部收敛现象的发生。

5 结束语

本文设计的算术交叉算子避免了惩罚因子的引入, 变异算子具有前期的快速搜索能力和后期的最优解的保持能力, 选择算子具有“精英”保持能力, 因此, 算法具有了快速的收敛能力, 而且保持了解的多样性。仿真结果说明了算法的有效性。

参考文献:

[1] Deb K, Pratap A, Meyarivan T. Constrained test problems for multi-objective evolutionary optimization. Zitzler E, Deb K, Lothar Thiele et al eds. *Proc. of the 1st Int' l Conf on Evolutionary Multi-Criterion Optimization*. Berlin: Springer Verlag, 1994-2014 China Academic Journal Electronic Publishing House. All rights reserved. <http://www.cnki.net>

- 2001; 284-298.
- [2] 玄光男, 程润伟. 遗传算法与工程优化[M]. 北京: 清华大学出版社, 2004; 26-29.
- [3] 苏勇彦, 王攀, 范衡. 一种基于新约束处理方法的遗传算法[J]. 计算机工程与应用, 2007, 43(14): 71-73.
- [4] Lu Hai-ming, Yen G G. Rank-density-based multi-objective genetic algorithm and benchmark test function study[J]. *IEEE Trans. on Evolutionary Computation*, 2003, 7(4): 325-342.
- [5] 王宇平, 刘大莲. 基于平滑技术和一维搜索的全局优化进化算法及其收敛性[J]. 计算机学报, 2006, 29(4): 670-676.
- [6] 刘淳安, 王宇平. 动态多目标优化的进化算法及其收敛性分析[J]. 电子学报, 2007, 35(6): 1118-1121.
- [7] 王宇平, 焦永昌, 张福顺. 解无约束非线性全局优化的一种新进化算法及其收敛性[J]. 电子学报, 2002, 30(12): 1868-1871.
- [8] 刘淳安, 王宇平. 约束多目标优化问题的进化算法及其收敛性[J]. 系统工程与电子技术, 2007, 29(2): 277-280. (Liu Chun'an, Wang Yuping. Evolutionary algorithm for constrained multi-objective optimization problems and its convergence[J]. *Systems Engineering and Electronics*, 2007, 29(2): 227-280.)
- [9] Bäck T. Evolutionary algorithms in theory and practice[M]. New York: Oxford University Press, 1996; 21-28.
- [10] Rudolph G, Agapie A. Convergence properties of some multi-objective evolutionary algorithm[C]. Ali Zalala Eds. *Proc. of the Congress on Evolutionary Computation*, Piscataway, NJ: IEEE Press, 2000; 1010-1016.
- [11] Yao Xin, Liu Yong, Lin Guang-ming. Evolutionary programming made faster[J]. *IEEE Trans. on Evolutionary Computation*, 1999, 3(2): 82-102.
- [12] Lucidi S, Piccialli V. New classes of globally convexized filled functions for global optimization[J]. *Journal of Global Optimization*, 2002, 24(1): 219-236.
- [13] 周育人, 岳喜顺, 周继香. 演化算法的收敛速率与效率分析[J]. 计算机学报, 2004, 27(11): 1485-1491.
- [14] 明亮, 王宇平. 关于一类遗传算法的收敛速度研究[J]. 计算数学, 2007, 29(1): 15-26.

(上接第 660 页)

益矩阵来设计全维和降维状态观测器, 仿真结果说明观测器的设计是合理准确的。

参考文献:

- [1] Muller P, Hou M. On the observer design for descriptor systems[J]. *IEEE Trans. on Automatic Control*, 1993, 38: 1666-1671.
- [2] Hou M, Muller P. Observer design for descriptor systems[J]. *IEEE Trans. on Automatic Control*, 1999, 44: 164-169.
- [3] Darouach M, Zasadzinski M, Hayar M. Reduced-order observer design for descriptor systems with unknown inputs[J]. *IEEE Trans. on Automatic Control*, 1996, 41: 1068-1072.
- [4] Darouach M, Boutayeb M. Design of observers for descriptor systems[J]. *IEEE Trans. on Automatic Control*, 1995, 40: 1323-1327.
- [5] Koenig D and Marx B. Design of observers for descriptor systems with delayed state and unknown inputs[C]. *Proc. of the 2004 American Control Conference*, 1992; 981-982.
- [6] Symos V. Observer design for descriptor systems with unmeasurable disturbances[C]. *Proc. of 31st Conference on Decision and Control*, 1992; 981-982.
- [7] Gao Z W, Ding S X. PMID observer design for unknown input generalized dynamical systems[C]. *Proc. of the 16th IFAC World Congress*, 2005.
- [8] Liu G P, Daniel W C Ho. Full-Order and reduced-order observers for Lipschitz descriptor systems: the unified LMI approach[J]. *IEEE Trans. on Circuits Systems*, 2006, 53: 563-567.
- [9] Liu G P, Wang D, Sun Y. Observer design for descriptor systems with Lipschitz constraint[C]. *International Conference on Control and Automation*, 2005; 220-223.
- [10] Zimmera G, Meierb J. On observing nonlinear descriptor systems[J]. *Systems & Control Letters*, 1997, 32: 43-48.
- [11] Boutayeb M, Darouach M. Observers design for nonlinear descriptor systems[C]. *Proc. of 34th Conference on Decision and Control*, 1995; 2369-2374.
- [12] Kaprielian S and Turi J. An observer for a nonlinear descriptor system[C]. *Proc. of the 34th Conference on Decision and Control*, 1992; 975-976.
- [13] Aslund J, Frisk E. An observer for non-linear differential-algebraic systems[J]. *Automatica*, 42, 2006; 959-965.
- [14] Boutayeb M, Darouach M, Rafaralahy H, et al. Asymptotic observers for a class of nonlinear singular systems[C]. *Proc. of the American Control Conference*, 1994; 1440-1441.
- [15] Gao Z W, Ding S X. Fault estimation for nonlinear descriptor systems with lipschitz constraints via LMI approach[C]. *Proc. of IFAC Symposia Safe-process*, 2006.
- [16] Gao Z W, Ho D W C. State/noise estimator for descriptor systems with application to sensor fault diagnosis[J]. *IEEE Trans. on Signal Processing*, 2006, 54: 1316-1326.
- [17] Gao Z W, Ding S X. Actuator fault robust estimation and fault-tolerant control for a class of nonlinear descriptor systems[J]. *Automatica*, 2007, 43: 912-920.
- [18] Zemouche A, Boutayeb M, Bara G. Observer design for nonlinear systems: An approach based on the differential mean value theorem[C]. *Proc. of the 44th IEEE Conference on Decision and Control and European Control Conference*, 2005; 6353-6358.
- [19] Yue D, Lam J, Ho D W C. Reliable control of uncertain descriptor systems with multiple time delays[J]. *IEE Proc. of Control Theory Appl.*, 2003, 150: 557-564.