

基于改进动态域遗传算法的神经元控制器参数优化*

鲜 阳,谭 飞
(四川轻化工大学自动化与信息工程学院,宜宾 644000)

摘要:针对传统遗传算法的选择、交叉与变异操作容易造成算法全局收敛概率低,求解精度不够等问题,提出了一种基于精英策略的动态域十进制整数编码遗传算法。算法以十进制整数编码遗传算法为基础,通过设计包括基因补全变异、重生变异等变异操作以及精英交叉操作,来提高算法的收敛全局性;设计搜索变量空间同步与独立动态调整策略,提升算法的收敛精度。仿真函数测试表明算法性能优异,同时该算法用于优化挖掘机伺服系统神经元 PID 控制参数,取得优良的控制品质,说明算法具有很好的工程适应性。
关键词:遗传算法;基因补全;变异;动态域;神经元 PID
中图分类号:TH165;TG659 **文献标识码:**A

Parameter Optimization of Neuron Controller Based on Improved Dynamic Domain Genetic Algorithm

XIAN Yang, TAN Fei

(School of Automation and Information Engineering, Sichuan University of Science Engineering, Yibin 644000, China)

Abstract: This paper puts forward a dynamic domain integer-coded genetic algorithm based on elitist strategy by analyzing the traditional genetic algorithm's limitation in selection, crossover and mutation operation that lead to algorithm's low global convergence probability and low accuracy. Based on the decimal integer-coded genetic algorithm with the elitist crossover strategy, the modified algorithm improves the algorithm's global performance of convergence by designing mutation operators that include gene completion mutation, rebirth mutation. The design of search domain for synchronous and independent dynamic adjustment improves the algorithm's convergence accuracy. The overall test of simulation function indicates that the algorithm has sound global performance of optimization, at the same time, the algorithm is used for optimizing neuron PID control system parameters, and good control quality is achieved. This proves that the algorithm has good adaptability to the project.

Key words: genetic algorithm; gene complement; mutation; dynamic domain; neuron PID

0 引言

遗传算法理论基本思想是基于进化论和遗传学说^[1-2]。由于遗传算法不依赖被求解的目标函数本身,具有较强的全局搜索能力、适用范围广、鲁棒性强,得到了较多的工程应用^[3-4]。HWANG、王剑楠等^[5-7]认为遗传算法和神经网络、模糊逻辑一起构成了“软计算”的三大基石,但算法还是存在编码、选择交叉、计算效率、精度等方面的一些问题需要解决,吸引了大量的科技工作者对它进行研究改进。杜嘉伟等^[8]提出一种基于矩阵二进制编码的改进遗传算法,通过对初始种群的编码以及降维保证了合理的初始适应度,使算法有优良的全局和局部搜索能力,但未在高维数据下验证可行性。周凯等^[9]设计了一种基于二进制和实数编码的混合编码多种群遗传算法,能有效提高计算效率和求解精度,但定义的优化模型的目标函数较少,显得算

法适应性不足。而本文以十进制编码遗传算法为基础,从提高算法收敛全局性、收敛精度两个方面进行相关设计,通过大量代表性测试函数对比分析,将其用于挖掘机的神经元控制系统参数优化,取得的结果说明算法具有很好的工程适应性。

1 归一化十进制编码与解码

讨论遗传算法的编码方法的文献有很多^[10-11],多数文献是以二进制和浮点编码进行讨论,而讨论十进制整数编码的文献有少量,且主要是变异问题存在缺陷,收敛效果不佳。

本算法设优化问题如式(1)所示。

$$z = \max f(\mathbf{x}) \quad \mathbf{x} = (x_1, x_2, \dots, x_m) \in S \subseteq \mathbf{R}^m \quad (1)$$

式中, $x_i \in [a_i, b_i], i = 1, 2, \dots, m; a_i, b_i \in \mathbf{R}$ 分别为 x_i 的下界和上界; m 为目标函数变量维数。

该十进制整数编码是把目标函数的每一维变量的

收稿日期:2022-03-19;修回日期:2022-04-29
* 基金项目:国家自然科学基金(61902268);四川省科技计划(2019YFSY0045);大学生创新创业训练计划项目(S202010622090, cx2021192, cx2019259)
作者简介:鲜阳(1996—),男,硕士研究生,研究方向为智能信息处理与控制,(E-mail)xy_mts@163.com。

定义域规范到 1, 域间的点属于 $[0, 1]$ 间, 然后放大 $(10^d - 1)$ 倍, 得到变量基因位的编码值, 如式(2)所示。

$$y_{xi} = \sum_{k=1}^d c_{ik} \cdot 10^{d-k} = \frac{x_i - a_i}{b_i - a_i} \cdot (10^d - 1)$$

$$x_i^c = (c_{i1}c_{i2}c_{i3}\cdots c_{ik}\cdots c_{id})$$

$$c_{ik} \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

$$k = 1, 2, \dots, d \quad i = 1, 2, \dots, m \quad (2)$$

式中, y_{xi} 为变量 x_i 编码 x_i^c 对应的整数; d 为编码精度, d 越大编码越长, 精度越高, 但会增加算法的计算复杂度。一个变量个体 x 的编码 $\{x_1^c, x_2^c, \dots, x_m^c\}$ 总长度为 $L = m \cdot d$ 。通常 d 以 $\log(b_i - a_i) + e$ 为参考, e 为变量 x_i 求解需要精确到的有效小数位。

编码是变量 x_i 离散过程, 其误差与 10^{-d} 成正比, d 越大, 误差越小。解码时通过式(3)可得变量值 x_i 。

$$x_i = (b_i - a_i) \cdot \sum_{j=1}^d c_{ij} \cdot 10^{d-j} / (10^d - 1) + a_i$$

$$i = 1, 2, \dots, m \quad (3)$$

编码取各变量的基因长度相同, 不需处理符号位和定义域的宽度。

2 动态域遗传算法设计

2.1 种群初始化

在目标函数定义域随机取 p 点构成变量矩阵 $V_{p \times m}$, 求出相应的适应值列向量 Z , 同时把 $V_{p \times m}$ 转换为编码阵 C 如式(4)所示, 并按适应值从大到小对各矩阵排序, 最优个体在第一位。编码阵 C 中, 第 j 行表示为 $\{x_{j1}^c, x_{j2}^c, \dots, x_{jm}^c\}$, x_{ji}^c 表示每一维变量的十进制编码。

$$Z_{p \times 1} = f(V_{p \times m}), V_{p \times m} \rightarrow C_{p \times md} \quad (4)$$

2.2 交叉与选择

算法交叉, 父代采用精英策略, 其中一个父代固定为备选库中的最优个体, 另一个通过二选一择优机制从库中随机抽取。采用两点交叉法, 如式(5)所示。

$$\begin{cases} x_{11}^c, x_{12}^c, \dots, x_{1m}^c \\ x_{k1}^c, x_{k2}^c, \dots, x_{km}^c \end{cases} \text{ 在第 2 点交叉得 } \begin{cases} x_{11}^c, x_{k2}^c, \dots, x_{1m}^c \\ x_{k1}^c, x_{12}^c, \dots, x_{km}^c \end{cases}$$

$$x_{12}^c \neq x_{k2}^c \quad (5)$$

若 x_{12}^c 与 x_{k2}^c 基因完全相同, 则分别随机取一个 d 维 $0 \sim 9$ 的整数向量替换子代中的 x_{12}^c 与 x_{k2}^c 部分作为新的子代。变量数少建议采用均匀交叉。

交叉完成规定子代的产生, 译码求解目标函数值, 然后子代与父代备选库一起按适应值从大到小排序, 并去除相似个体, 择优选择排在前的 n_1 条个体组成下一代备选库的一部分。

2.3 算法的变异策略

2.3.1 基因补全变异

在二进制遗传算法中, 变异是通过 0 和 1 的互变实现, 因为它的状态只有两个, 很易满足每个等位基因都存在基因 0 和 1 的全集, 而高进制算法互补运算很难实现等位基因为 $0 \sim n-1$ 的全集。本算法采用基因补进方法实现, 方法是, 检查备选库编码阵 C , 按列查找缺失的等位基因。比如, 查询哪些列缺失基因 0, 采用最优个体编码 c_e , 修改相应等位基因为 0, 得到补进基因值 0 的个体, 如式(6)所示。

$$c_e^0 = \begin{cases} 0, & c_{jl} \neq 0, j = 1, 2, \dots, p; l = 1, 2, \dots, md \\ c_{el}, & \text{otherwise} \end{cases} \quad (6)$$

把十进制的 $0 \sim 9$ 的基因值补完, 最多可得到 10 条染色体, 并译码加入到备选库中。该方法可大概率提高优秀个体与缺失基因的组合, 提高算法全局优化能力。

2.3.2 重生变异

重生变异就是以最优个体 x_e 为中心随机从当前定义域中生成多个点, 方法如式(7)所示, 译码加入到父代备选库中。重生策略可增加备选库基因模式的多样性。这种策略也相当于是一种变异, 可称为重生变异。

$$x'_e = x_e + \frac{(b - x_e)}{\rho} * r_1 - \frac{(x_e - a)}{\rho} * r_2, \rho \geq 1 \quad (7)$$

式中, x'_e 为补充的一个新个体; x_e 为最优个体译码值; a 为变量下界; b 为各变量上界; r_1 和 r_2 为两个随机生成的均匀分布于 $[0, 1]$ 间行向量; $*$ 为对应元素相乘, 不是矩阵乘; ρ 为一个常数, 通常 $\rho = 1$, 等效于全域范围随机取点, ρ 越大, 取点越靠近当前最优个体 x_e 。

这两种变异个体的补充, 构成了每列各种十进制基因都存在的完整父代备选库。

2.4 动态域调整

提高算法寻优精度的办法可以采用实时缩小算法搜索范围来解决, 搜索范围小, 精度就越高, 但全局性兼顾是个问题。解决这个矛盾本算法采用动态域方法, 结合到前面精英交叉策略, 把这算法称为动态域精英遗传算法 (dynamic domain and elite genetic algorithm, DDEGA)。

按式(8)检查最优个体的改善速度是否满足要求, 成立则进行动态域调整。

$$z_e - z_{e0} < \alpha \cdot |z_{e0}| \text{ 或 } z_e - z_{e0} < \alpha \quad (8)$$

式中, z_e 和 z_{e0} 分别为当前和前一代的最优适应值; $\alpha \in (0, 1)$, 通常 α 取 0.05 或再小一点。

动态域搜索调整如图 1 所示, 有两个阶段。第一阶段式(8)不满足时, 采用式(10)调整域边界, 这是动态界 1 法, 此法使各维变量边界同步整体缩小。

$$B_1 = \min(|b^0 - a^0|) \quad (9)$$

$$a^1 = a^0 + B_1/\beta, b^1 = b^0 - B_1/\beta, \beta > 2 \quad (10)$$

式中, a^0 和 b^0 分别为上代的下界和上界; a^1 和 b^1 为变量 x 调整后的下界和上界; β 为一个常数, 本算法取 10, 若取更大, 搜索域缩小很快, 不利于全局寻优。式(8)成立, 域进一步缩小。

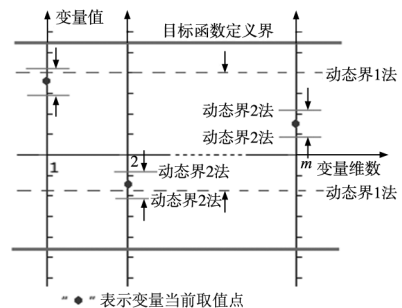


图 1 算法搜索动态域示意图

在域 1 里边界变化后, 首先复制最优个体, 备选库的其他个体用重生策略随机产生。若最优个体的任何一维值离域边界的距离小于容错变量 t_{ol} , 则进入动态界 2 调整, 以每维变量为中心, 缩小搜索域。

$$B_2 = \max(|b^1 - a^1|) \quad (11)$$

$$a^2 = x_e - B_2/\beta, b^2 = x_e + B_2/\beta, \beta > 2 \quad (12)$$

式中, a^2 和 b^2 分别为调整后变量下界和上界。调整后边界限制到原始定义域内。

动态界 2 方法, 以式 (8) 为准则, 使边界逐次缩小, 当边界宽度小于 t_{ol} 或最优个体离边界距离小于 t_{ol} 时, 恢复边界到最初定义域, 复制精英个体, 其他备选库个体重新初始化产生, 然后继续遗传搜索。

2.5 算法步骤

算法的停止条件可设置多个。当达到最大代数、或者达到寻优精度等条件都可停止寻优, 显示参数和结果, 完成计算。

按照以上所述, DDEGA 算法的流程图如图 2 所示。

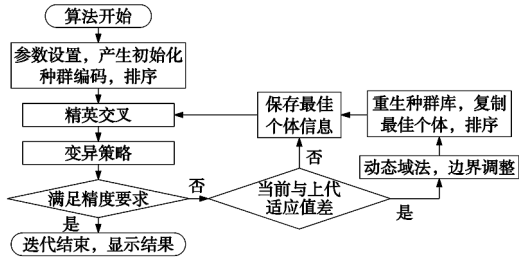


图 2 DDEGA 算法流程图

2.6 算法全局收敛性分析

研究算法时, 采用设置 $d=3$, 去除动态域法, 只应用前述的基于精英交叉策略的遗传算法 (elite genetic algorithm, EGA), 或者不用精英交叉, 两个父体都用联赛选择的遗传算法 (league matches genetic algorithm, LMGA), 比较它们的收敛特性。

由于算法随时从搜索域随机补充新个体, 且每次边界恢复时从全域重生新个体种群, 使种群丰富多样, 从而减小了算法陷入局部最优的概率。可以证明每一次重生策略和搜索域恢复时都从全局随机取点, 取到全局最优解的概率可以达到 1, 所以 DDEGA 具有全局最优收敛能力。

3 算法测试

为便于比较, 测试条件一致, 测试函数选择文献 [12] 所列部分 Benchmark 函数如表 1 所示。为了说明精英策略的有效性和快速性, 把 EGA 算法与 LMGA 算法进行测试比较; 同时把动态域算法 DDEGA 与文献 [12] 的自适应直接变异浮点编码遗传算法 ADMRCGA (adaptive directed mutation for real-coded genetic algorithm) 进行比较。文献 [12] 列了 6 种算法, ADMRCGA 是其最好的一种。

算法设种群为 60, 编码精度为 14 位, 交叉操作后通过排序择优选择得 45 个个体, 基因补全可最多得到 10 个个体, 重生操作添加补足总数, 它们一起构成下一代交叉备选库种群。选函数维数为 30, 最大评价 30 万次, 独立运行 30 次统计结果如表 2 所示。

本文算法采用 MATLAB2020a 在 RTX2060.8 G 内存、Win10 系统下编程实现。函数优化停止条件: ①达到最大代数; ②100 代适应值不变; ③与全局最优值绝对误差小于阈值 $1E-8$ 。只要满足任一条件就终止迭代进化。采用 30 代独立运行进行统计比较, 如表 2、表 3 所示。

表 1 BENCHMARK 测试函数

函数	变量域	全局极值
$f_1 = -20\exp(-0.02\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	$[-30, 30]$	$f_{1min} = 0$
$f_2 = 0.1 \frac{1}{n} \sum_{i=1}^n \cos(5\pi x_i) - \sum_{i=1}^n x_i^2 - 0.1 \cdot n$	$[-1, 1]$	$f_{2max} = 0$
$f_3 = \exp(-0.5 \sum_{i=1}^n x_i^2) - 1$	$[-1, 1]$	$f_{3max} = 0$
$f_4 = 1 + \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}})$	$[-600, 600]$	$f_{4min} = 0$
$f_5 = 0.1 \left \sin^2(3\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right $	$[-5, 5]$	$f_{5max} = 0$
$f_6 = 10 \cdot n + \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i)]$	$[-5.12, 5.12]$	$f_{6min} = 0$
$f_7 = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 - (x_i - 1)^2]$	$[-30, 30]$	$f_{7min} = 0$
$f_8 = \sum_{i=1}^n [x_i \sin(\sqrt{ x_i })] - 418.9828872724338 \cdot n$	$[-500, 500]$	$f_{8max} = 0$

表 2 函数 30 次独立运行的均值与方差

函数	DDEGA		ADMRCGA	
	均值	方差	均值	方差
f_1	8.33E-09	2.18E-09	3.28E-01	2.74E-02
f_2	7.75E-09	2.62E-09	3.33E-07	4.71E-07
f_3	8.33E-09	1.61E-09	2.20E-07	6.00E-08
f_4	8.50E-09	2.22E-09	4.43E-03	7.87E-03
f_5	6.13E-09	3.28E-09	2.23E-09	3.42E-09
f_6	8.28E-09	2.17E-09	1.36E+01	2.48E+00
f_7	4.31E-09	3.18E-09	6.58E+00	4.51E+00
f_8	7.53E-09	2.88E-09	6.51E+02	5.99E+02

函数	EGA		LMGA	
	均值	方差	均值	方差
f_1	7.76E-03	1.45E-03	9.99E-03	2.37E-02
f_2	1.36E-06	5.29E-07	2.56E-06	3.45E-07
f_3	5.58E-08	2.75E-08	9.52E-08	4.73E-08
f_4	1.66E-02	1.75E-02	1.64E-02	2.67E-02
f_5	8.24E-07	8.09E-07	1.86E-06	7.31E-07
f_6	7.05E-04	3.00E-04	1.41E-03	3.19E-03
f_7	9.13E+00	2.05E+01	2.12E+01	3.41E+01
f_8	1.24E-03	5.45E-04	1.91E-03	5.24E-03

表 3 函数的平均耗时及评价次数

30 次平均函数	DDEGA		ADMRCGA	
	耗时/s	评价数	耗时/s	评价数
f_1	4.05	45 133.0	36.25	300 000
f_2	2.17	24 442.6	38.05	300 000
f_3	1.86	20 075.3	36.37	300 000
f_4	2.74	30 548.0	20.47	155 700
f_5	1.82	20 614.8	11.27	82 800
f_6	2.99	33 882.9	22.03	170 000
f_7	5.77	84 922.8	37.46	300 000
f_8	2.77	31 404.3	18.46	156 000

30 次平均函数	EGA		LMGA	
	耗时/s	评价数	耗时/s	评价数
f_1	17.44	300 000	19.58	300 000
f_2	17.20	300 000	19.45	300 000
f_3	16.83	300 000	19.46	300 000
f_4	18.10	300 000	20.26	300 000
f_5	18.13	300 000	19.58	300 000
f_6	17.29	300 000	19.20	300 000
f_7	17.19	300 000	19.48	300 000
f_8	17.61	300 000	19.98	300 000

从表 3 可以看出 DDEGA 计算结果在函数 $f_1 \sim f_4$ 、

$f_6 \sim f_8$ 比算法 ADMRCGA 高很多,并且从表 3 可以看出 DDEGA 算法搜索计算时间很短,说明 DDEGA 算法收敛很快。但 DDEGA 算法要进一步提高精度,得适当增加编码长度,与浮点编码比较,在要求非常高精度时受到一定限制。结合表 2 和表 3 可以看出,在不用动态域方法时,很难达到 $1E-8$ 的要求精度。在相同评价计算次数下,基于精英策略的算法精度要优一些。由于篇幅限制,列了两个函数的收敛过程如图 3 和图 4 所示,可以看出,基于精英策略的算法 EGA 在初始阶段的收敛速度比联赛算法 LMGA 要快,但从表 3 看,后继的精确搜寻能力比较,LMGA 的能力比 EGA 相对稍微强点,平均来看还是 EGA 比 LMGA 好点。几种算法中,DDEGA 是精英策略与动态域方法结合,有了快速与精确的搜索能力,使遗传算法的速度得到了质的飞越。实验还发现 DDEGA 在函数 f_7, f_8 上的收敛效果比文献[13]的基于精英学习的动态差分算法还好,说明遗传算法的性能并不逊于现代差分进化算法,有它独有的魅力。

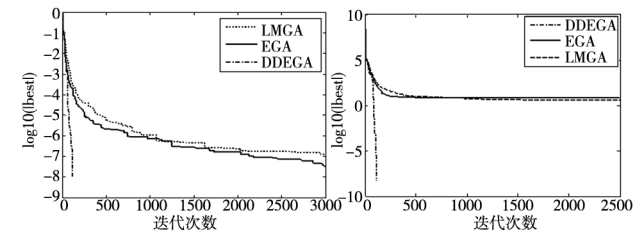


图 3 f_3 函数收敛曲线 图 4 f_7 函数收敛曲线

4 算法的应用

神经网络技术发展并应用到控制中,解决了很多常规 PID 控制无法解决的问题。增量 PID 控制算法和神经网络技术的结合,得到了神经元 PID 控制算法^[14-16]。一些神经元 PID 控制算法采用 BP 方法实现自适应控制,但效果不理想。虽然理论上证明,系统自主学习调整控制参数,可以达到稳定收敛,但仿真发现,还是存在长时间学习后失稳的情况,但是在确定控制权重等参数后,系统的鲁棒性和控制性能比常规 PID 好很多。所以神经元控制的应用最好是离线学习后用于在线参数整定。

神经元 PID 控制系统的结构如图 5 所示。采用动态域精英遗传算法优化神经元 PID 控制系统的权值。

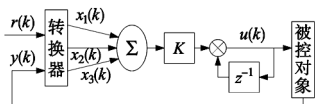


图 5 神经元 PID 控制系统

神经元控制单元算法如式(13)、式(14)所示,这种结构便于系统投运,控制的总增益 K 可根据实际情况及时在线调整,权值设置好后可不再调整。

$$x_1(k) = \Delta e(k) \quad (13)$$

$$x_2(k) = r(k) - y(k) = e(k)$$

$$x_3(k) = e(k) - 2e(k-1) + e(k-2)$$

$$u(k) = u(k-1) + K \tanh[\omega_1 x_1(k) + \omega_2 x_2(k) + \omega_3 x_3(k)] \quad (14)$$

式中, ω_i 为加权系数。

被控对象取某挖掘机伺服系统特性:

$$G_p(s) = \frac{1}{s(0.8s+1)(0.25s^2+0.4s+1)} \quad (15)$$

设采样时间 0.1 s 对 $G_p(s)$ 进行离散,对控制器的 3 个权值寻优。目标函数取 ITAE 和 ISE 如式(16)所示,优化到最小。

$$\min J_{ITAE} = \int_0^{t_m} t |e| dt, \min J_{ISE} = \int_0^{t_m} e^2 dt \quad (16)$$

式中, t_m 为仿真终止时间,取 30 s 。

该目标函数是一个复杂的多峰函数,寻优发现在 K 不定的情况下优化得到的权值解不唯一。本次优化把 K 固定等于 1.5 (连续系统的临界增益), ω_1 和 ω_3 取值范围 $[0, 1]$; 虽然对象由积分特性,但仿真发现 ω_2 不能为 0 , 否则对定值控制的稳态偏差将失去修正能力,所以寻优取值范围确定为 $[0.001, 1]$ 。本文算法对系统进行 10 次寻优,每次评价 5000 次得到的 ITAE 结果, $\omega_1 = 0.2687, \omega_2 = 0.001, \omega_3 = 0.1647$; ISE 的优化结果, $\omega_1 = 0.4515, \omega_2 = 0.0054, \omega_3 = 0.3649$ 。这些结果同 ZN 法整定的 PID 控制结果做比较如图 6 所示。可见算法取得了优良控制质量。可以验证在这权值下把 K 放大 3 倍或缩小到 $1/3$, 系统不会失去稳定,神经元 PID 控制系统具有很好的鲁棒稳定性。

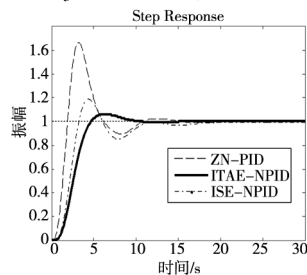


图 6 挖掘机伺服控制系统的阶跃响应曲线

5 结论

本文提出的基于精英策略动态域整数编码遗传算法,采用归一化十进制整数编码求解连续定义域函数最大值问题,它编码方便,不用考虑符号位和定义域的大小,同样情况下比二进制编码求解效率高。实验表明它算法具有全局寻优和求精的能力,并且快速稳定收敛于理论目标,具有良好的收敛特性,搜索性能优越。本算法缺陷在于采用整数编码,有离散量化误差,进一步精度的提高将加大编码译码的复杂性,但精英学习的快速收敛能力、重生策略的全局最优保证能力、动态域的求精能力值得其他算法设计的学习。算法用于优化神经元 PID 控制系统,取得了成功,表明能很好解决工程优化实际问题。

[参考文献]

- [1] 沈小伟, 王桂所, 王一云. 现代智能优化算法研究综述[J]. 山西建筑, 2009, 35(35): 30-31.
- [2] 范爽. 遗传算法理论研究及其应用[J]. 科技与创新, 2017(23): 21-22.
- [3] 李岩, 袁弘宇, 于佳乔, 等. 遗传算法在优化问题中的应用综述[J]. 山东工业技术, 2019, (12): 242-243, 180.
- [4] 李芳, 赵天洋. 遗传算法理论及其进展探析[J]. 技术与市场, 2016, 23(1): 87.
- [5] HWANG M, LIN J W. Neural fuzzy systems: A tutorial and an application[J]. Journal of Computer Information Systems, 2000, 40(4): 27-31.
- [6] 王剑楠, 崔英花. 基于自适应变异算子的实数编码遗传算法[J]. 北京信息科技大学学报(自然科学版), 2021, 36(2): 46-51.
- [7] 杨从锐, 钱谦, 王锋, 等. 改进的自适应遗传算法在函数优化中的应用[J]. 计算机应用研究, 2018, 35(4): 1042-1045.

用式(9)和式(10)构成单回路控制, k_d 取2,采用本文IPSGA算法进行优化。以ZN法参数的两倍为搜索上界,进行1000次评价,优化得到的结果如表4所示。与文献[16]中ZN-PID和GA-PID以及文献[17]的自适应二次变异差分ASMDE-PID比较。优化的PID阶跃响应曲线如图3所示,改进的模式遗传混合算法得到的参数响应比差分算法快,总体可见本算法取得了快速且超调小的满意控制质量。

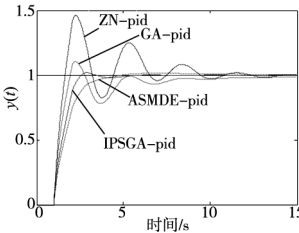


图3 系统最优PID控制的阶跃响应

表4 不同方法的PID参数优化结果比较

PID 控制器	ZN-PID	GA-PID	ASMDE-PID	IPSGA-PID
k_p	2.121 5	1.856 1	1.428 6	1.698 1
T_i	2.624 5	7.511 5	4.835 9	5.351 6
T_d	0.629 9	0.525	0.304 0	0.302 42
M_p	46.19%	10.66%	1.34%	1.94%
t_s	10.44	1.93	4.1	2.44
t_r	1.64	12.84	4.94	2.59
e_{ss}	0	0	0	0

5 结论

本文分析了模式搜索算法特点,结合它现存在的不足,提出了结合遗传交叉的混合算法,主要做了3方面改进:①引入信息向量,并改进了模式搜索半径,搜索点全域可达,便于全局寻优;②组合模式搜索改为方向增益组合,实验发现更有利于提高搜索速度;③引入遗传交叉学习,使算法具有更好的全局性。接着本文选取了典型的测试函数进行测试,取得了较好的计算结果。并把这种改进的模式搜索算法用于实际微分PID参数的优化,最后与ZN法、GA法和ASMDE法进行比较,结果表明本文算法具有快速而精确的寻优能力,也说明改进的模式遗传搜索算法的工程可用性。

[参考文献]

[1] 刘文斌,贺小林,梁栋,等. 基于模式搜索算法的三相无源PFC参数整定[J]. 日用电器,2021(7):77-80,106.
[2] 谢智平,吴剑,徐梦蛟,等. 遗传—广义模式搜索方法反演瑞雷波频散曲线[J]. 山西建筑,2016,42(31):70-72.
[3] KOESSLER E, ALMOMANI A. Hybrid particle swarm optimization and pattern search algorithm[J]. Optimization and Engineering, 2021(22):1539-1555.

[4] HOFMEISTER B, BRUNS M, ROLFES R. Finiteelement modelupdating using deterministic optimisation:a global pattern search approach[J]. Engineering Structures, 2019,195:373-381.
[5] 阳洋,李昌林,凌园,等. 基于广义模式搜索算法的框架结构损伤检测新方法研究[J]. 仪器仪表学报,2021,42(6):123-131.
[6] BEIGI A M, MAROOSI A. Parameter identification for solar cells and module using a hybrid firefly and pattern search algorithms[J]. Solar Energy, 2018,171:435-446.
[7] KHADANGA R K, PADHY S, PANDA S, et al. Design and analysis of tilt integral derivative controller for frequency control in an islanded microgrid:a novelhybrid dragonfly and pattern search algorithm approach[J]. Arabian Journal for Science and Engineering, 2018,43(6):3103-3114.
[8] 曲良东,何登旭,吴尽昭. 一种群模式全局搜索算法[J]. 模式识别与人工智能,2013,26(6):592-597.
[9] SONI V, PARMAR G, KUMAR M. A hybrid grey wolf optimisation and pattern search algorithm for automatic generation control of multi-area interconnected power systems[J]. International Journal of Advanced Intelligence Paradigms, 2021,18(3):265-293.
[10] 封京梅,刘三阳. 基于模式搜索的粒子群算法求解绝对值方程[J]. 兰州大学学报(自然科学版),2017,53(5):701-705.
[11] 黄光球,陆秋琴. 保护区种群迁移动力学优化算法[J]. 计算机科学,2020,47(2):186-194.
[12] 郭海燕,李英娜,李川. 基于改进人工鱼群优化BP的短期负荷预测[J]. 数据通信,2021(6):24-29.
[13] 厉康平,汪鹏君,张会红. 基于人口迁移算法的三值FPRM电路面积最佳极性搜索[J]. 华东理工大学学报(自然科学版),2016,42(1):104-109.
[14] DENG W, LIU H L, XU J J, et al. An improved quantum-inspired differential evolution algorithm for deep belief network[J]. IEEE Transactionson Instrumentation and Measurement, 2020,4:9724-9732.
[15] WANG Y L, YU Y Y, LI K, et al. A human-computer cooperation improved ant colony optimization for ship pipe route design[J]. Ocean Engineering, 2018,150:12-20.
[16] ZAHIR A A M, ALHADY S S N, OTHMAN W, et al. Genetic algorithm optimization of PID controller for brushed DC motor[J]. Intelligent Manufacturing & Mechatronics, 2018,38:427-437.
[17] 吴亮红,王耀南. 动态差分进化算法及其应用[M]. 北京:科学出版社,2014.

(编辑 祝智铭)

(上接第39页)

[8] 杜嘉伟,余粟. 基于矩阵二进制编码遗传算法的频繁项集挖掘[J]. 智能计算机与应用,2021,11(1):143-146,151.
[9] 周凯,许中,马智远,等. 基于混合编码多种群遗传算法的动态电压恢复器优化配置求解方法研究[J]. 电力电容器与无功补偿,2022,43(1):151-157.
[10] 董其欢. 用于多维尺寸可变装箱的遗传算法新编码[J]. 计算机时代,2018,(8):61-63,66.
[11] 张超群,郑建国,钱洁. 遗传算法编码方案比较[J]. 计算机应用研究,2011,28(3):819-822.
[12] TANG P H, TSENG M H. Adaptive directed mutation for re-

al-coded genetic algorithms[J]. Applied Soft Computing, 2013,13(1):600-614.
[13] 彭虎,吴志健,周新宇,等. 基于精英区域学习的动态差分进化算法[J]. 电子学报,2014,42(8):1522-1530.
[14] 乐旭东. 基于单神经元PID控制双缸同步液压系统[D]. 杭州:浙江工业大学,2020.
[15] 项思哲,周依涛,郑炜炀,等. 基于改进单神经元PID算法的平衡小车控制[J]. 电子测量技术,2021,44(13):68-72.
[16] 孙建飞,辛长范. 火箭弹电动舵机模糊单神经元PID控制算法[J]. 海军航空工程学院学报,2019,34(6):499-504,510.

(编辑 祝智铭)