

基于改进粒子群优化算法的PID控制器整定

杨 智, 陈志堂, 范正平, 李晓东

(中山大学 信息科学与技术学院, 广东 广州 510275)

摘要: 由传统的Z-N(Ziegler-Nichols)整定公式得出的PID参数, 不能得到最佳的控制性能. 为此, 本文提出一种基于适应度指数定标和边界缓冲墙相结合的改进型粒子群算法, 应用于PID参数的整定. 首先采用适应度指数定标的选中概率, 挑选出粒子进行随机变异; 其次对越界的粒子进行缓冲, 保证粒子落在寻优空间内以增加粒子种群多样性, 同时调整种群粒子个数、社会和认知因子以提高寻优效率. 在仿真实验中, 将改进的粒子群算法分别应用于5种不同的工业过程, 整定他们的PID参数. 对误差绝对值乘以时间积分的性能指标(ITAE)做最小化, 得到了相应的PID参数, 验证了这里提出的改进型粒子群算法的有效性.

关键词: PID参数整定; 粒子群算法; 过程控制; 通用过程模型; 系统仿真

中图分类号: TP18 **文献标识码:** A

Tuning of PID controller based on improved particle-swarm-optimization

YANG Zhi, CHEN Zhi-tang, FAN Zheng-ping, LI Xiao-dong

(School of Information Science and Technology, Sun Yat-sen University, Guangzhou Guangdong 510275, China)

Abstract: Because the classical PID parameter settings obtained by Z-N(Ziegler-Nichols) method usually fail to achieve the best control performances, we propose an improved particle swarm optimization(IPSO) algorithm with fitness exponential scale and border buffer wall for tuning the PID parameters. First, by the selection-probability of the fitness exponential scale, we select the underbred particles for random mutations. Secondly, we employ the border buffer wall to block the slopping-over particles, making them to fall in the explored space of optima to enhance the diversity of the particle swarm. Meanwhile, by modifying the number of swarm particles as well as the social and cognitive factors, we improve the efficiency of optimum searching. In the simulation experiments, we apply the IPSO algorithm to the PID parameter tuning for 5 different industrial process models, the corresponding optimal PID parameters are obtained under the criterion of minimal integral-time-weighted-absolute error(ITAE). The effectiveness of the proposed improved particle swarm optimization(IPSO) algorithm is validated.

Key words: PID controller parameters tuning; particle swarm optimization; process control; general process model; system simulation

1 引言(Introduction)

在工业过程控制中, PID控制器因其结构简单、鲁棒性强等优点得到广泛应用. 据统计, PID控制策略在实际控制系统中所占的比例超过90%^[1], 自Ziegler和Nichols提出PID参数整定问题(简称Z-N法)^[2], 已经引起许多学者研究的广泛兴趣, 目前存在许多传统的PID参数整定方法^[3], 如经验整定法、飞升曲线法、临界比例度法、衰减曲线法等等. 尽管这些方法已在实际控制系统中普遍使用, 但传统的PID参数整定表控制性能指标不理想^[4], 通常超调量大、调节时间长. 随着智能计算科学的

逐步兴起, 智能优化算法在PID参数整定中也取得了传统优化方法无法比拟的结果, 如采用遗传算法^[5]、蚁群算法^[6]进行参数寻优得到了较好的结果, 但遗传算法存在早熟、参数依赖性强的缺点; 而蚁群算法算法编程复杂, 一定程度上限制了其应用, 且目前利用这两类优化算法进行PID参数整定的研究的被控过程模型类型较少, 通用性不高. 粒子群优化(particle swarm optimization, 简称PSO)算法是Eberhart和Kennedy在1995年提出的一类基于群智能的随机全局优化进化算法^[7]. 该算法以其实现容易、精度高、收敛速度快和适应性强等优点引起了

学术界的高度重视^[8~11]. 本文针对传统的粒子群算法早熟、易陷入局部极值点及维数灾难等缺点, 提出了一种基于适应度指数定标和边界缓冲墙相结合的粒子群算法, 并对3种典型函数进行了性能测试, 将改进的粒子群算法(IPSO)用于控制系统PID参数的优化整定中, 对工业过程控制中通用的5类被控过程模型进行优化计算和仿真实验, 进一步验证基于适应度指数定标和边界缓冲墙相结合的改进粒子群算法(IPSO)的性能.

2 改进的粒子群算法(Improved particle swarm optimization)

2.1 标准粒子群优化问题(Basic particle swarm optimization)

定义 \mathbb{R}^n 为维数为 n 的寻优空间, \mathbf{x} 是 $1 \times n$ 的行向量. 优化算法就是在 \mathbb{R}^n 中找到一个位置向量 \mathbf{x}_0 , 使得 $f(\mathbf{x}) \geq f(\mathbf{x}_0)$ 成立. 标准粒子群算法表述如下: $\mathbf{x} \in \mathbb{R}^n$ $\mathbf{x}_0 \in \mathbb{R}^n$. 标准粒子群算法表述如下: 首先初始化一个随机粒子群体, 每个粒子每一维上都赋予一个初始随机位置和初始随机速度, 然后通过不断迭代更新粒子的位置使其更接近优化目标, 直到达到最大寻优次数或设置精度时迭代停止, 输出最优解. 标准的粒子群算法可用以下的迭代公式表示:

$$v_{ij}(k+1) = \chi[w(k+1)v_{ij}(k) + c_1r_1(pb_{ij} - x_{ij}(k)) + c_2r_2(gbest_j - x_{ij}(k))], \quad (1)$$

$$x_{ij}(k+1) = x_{ij}(k) + v_{ij}(k+1). \quad (2)$$

其中: $x_{ij}(k)$ 和 $v_{ij}(k)$ 分别代表第 i 个粒子在第 k 时刻所处的第 j 维位置分量和速度分量; $c_1 > 0$ 和 $c_2 > 0$ 分别是认知因子和社会因子; χ 是限制因子影响算法的收敛性, Clerc^[12]认为限制因子取

$$\chi = \frac{2}{|2 - \phi - \sqrt{\phi^2 - 4\phi}|}$$

能起到较好结果, 这里 $\phi = c_1 + c_2$; r_1 和 r_2 是两个 $[0, 1]$ 区间的随机因子; pb_{ij} 和 $gbest_j$ 分别是第 i 个粒子和全局粒子当前所经过的第 j 维最好位置分量; $w(k)$ 为惯性因子通常采用随迭代次数线性下降的方法^[13], 即

$$w(k) = w_{\max} - \frac{w_{\max} - w_{\min}}{\text{iter}_{\max}} \text{iter}, \quad (3)$$

式中: iter_{\max} 是算法设置的最大迭代次数, iter 是目前的迭代次数. 这里将待整定的参数集 $(b_{n-1}, b_{n-2}, \dots, b_1)$ 作为粒子在寻优空间位置的坐标 $P(b_{n-1}, b_{n-2}, \dots, b_1)$.

2.2 改进的粒子群算法(Improved particle swarm optimization)

2.2.1 监测邻域(Supervision region)

标准粒子群算法往往因寻优空间维数扩大而容易引起维数灾难、种群陷入局部极值点, 寻优结果大打折扣, 为了改善上述问题, 本文利用监测邻域策略监测全局最优邻域的拓扑. 为此, 定义监测邻域为

$$\phi = \{Pbest \mid \|Pbest - Gbest\|_2 < \varepsilon\},$$

其中 $\|Pbest - Gbest\|_2$ 表示粒子个体最优与全局最优的欧氏空间距离, ε 为欧氏距离设定值, 表征监测邻域大小. $\|Pbest - Gbest\|_2$ 定义为

$$\|Pbest - Gbest\|_2 = \sqrt{\sum_{j=1}^M (Pbest_{ij} - Gbest_j)^2}, \quad (4)$$

式中: M 表示寻优空间的维数, $Pbest_{ij}$ 和 $Gbest_j$ 分别是第 i 个粒子和全局粒子当前所经过的第 j 维最佳位置. 为了监测 ϕ 中粒子数量, 这里记 N_ϕ 为 ϕ 中粒子的个数, N 为种群粒子总个数, 并设定 δ 为监测邻域粒子浓度上限, 当式 $N_\phi > N \times \delta$ 成立时, 表明监测邻域中粒子的浓度太高, 种群多样性有损失太多的危险, 必须增加种群的多样性.

2.2.2 基于适应度指数定标的概率变异(Probability mutation based on fitness exponential scale)

当监测到种群多样性受到威胁时, 必须挑选出粒子进行变异, 对于如何挑选粒子进行变异, 本文提出基于指数定标适应度概率变异的方法, 将粒子适应度定义为目标函数值:

$$\text{fitness}_i = f(x)|_{(b_{n-1,i}, b_{n-2,i}, \dots, b_{1,i})}, \quad (5)$$

式中下标 i 表示第 i 个粒子. 在最小值优化问题中粒子的适应度越大, 该粒子被选中进行变异的概率越大. 可采用选中概率与适应度成比例的方式, 即

$$P_{si} = \text{fitness}_i / \sum_j \text{fitness}_j,$$

但当种群绝大部分粒子距离非常接近, 而只有极少数粒子严重偏离群体即出现超常粒子时, 该方法有很大缺陷. 为了降低超常粒子的适应度. 这里定义适应度指数定标为

$$\text{fitness}'_i = e^{K \times \text{fitness}_i}, \quad (6)$$

式中 K 为适当大小的常数. 通过指数定标能有效地降低超常粒子的适应度. 当监测邻域中粒子比较集中时, 粒子的适应度十分接近, 基于选中概率与适应

度成比例的方式挑选粒子进行变异可行性不高, 为了提高粒子选中概率的差异, 定义粒子选中变异概率为

$$P_{si} = 10^{\tau_c} \times \frac{\text{fitness}'_i}{\sum_j \text{fitness}'_j} - \left[10^{\tau_c} \times \frac{\text{fitness}'_i}{\sum_j \text{fitness}'_j} \right], \quad (7)$$

式中: τ_c 为适当大小的常数, 符号 $\lfloor \cdot \rfloor$ 表示向下取整. 为监测邻域中的每个粒子产生一个随机数 rand_i , 即 $\text{rand}_i \in [0, P_{si}]$, 此时, 由式(8)对被选粒子进行随机变异:

$$P_{\text{best}_{ij}} = a_j + (b_j - a_j) * \text{rand}(0, 1). \quad (8)$$

式中: $P_{\text{best}_{ij}}$ 是第 i 个粒子当前所经过的第 j 维最好位置, a_j 和 b_j 分别为粒子第 j 维寻优范围的下限和上限. 这里采用随机变异, 有利于提高种群的多样性. 实验结果表明, 上述方法确实有效的解决了粒子陷入局部极值点的问题, 提高了算法的精度.

2.2.3 边界缓冲墙(Border buffer wall)

利用粒子群算法进行寻优, 必须为粒子的位置进行限定, 使得粒子能够在有效空间内进行探索, 这些位置的限定值称为寻优空间的边界. 在算法迭代过程中, 粒子可能试图飞越边界, 针对上述问题, 传统的方法是将突破边界的粒子的位置强制为边界值. 但上述方法在寻优初期影响了种群有效个数, 降低种群多样性, 不利于算法的收敛. 为此, 本文提出边界缓冲墙策略, 根据粒子突破边界的速度进行有差别的缓冲. 定义越界粒子的缓冲为

$$\begin{cases} x_{ij}(k) = a_j((1 - \text{sgn } a_j L) + \\ \quad \text{sgn } a_j L(|v_{ij}(k)|/v_{j \max})\text{rand}), \\ \quad x_{ij}(k) > a_j, \\ x_{ij}(k) = b_j((1 + \text{sgn } b_j L) - \\ \quad \text{sgn } b_j L(|v_{ij}(k)|/v_{j \max})\text{rand}), \\ \quad x_{ij}(k) < b_j. \end{cases} \quad (9)$$

式中: sgn 为符号函数, $L \in [0, 1]$ 为一适当大小的常数, 它与缓冲墙的厚度有关, a_j 和 b_j 分别为粒子第 j 维寻优范围的下限和上限. 由式(9)可以看到, 对试图突破边界的粒子, 缓冲墙能够根据粒子的速度进行有差别的缓冲, 这样做的目的是考虑到粒子运动的规律性、目的性, 通过基于速度的缓冲能够在一定程度上保留粒子运动的有利趋势. 实验表明, 通过增加边界缓冲墙对粒子种群多样性和收敛速度精度有明显的改善.

2.3 改进的粒子群算法步骤(Steps of the improved particle swarm optimization)

综上所述, 可将适应度指数定标和边界缓冲墙的

改进粒子群算法(IPSO)步骤归纳如下:

Step 1 随机初始化粒子群中粒子的位置和速度;

Step 2 由式(1)和式(2)更新粒子位置和速度;

Step 3 根据式(9)启动边界缓冲墙对越界粒子进行缓冲;

Step 4 更新粒子个体最优值和个体最优位置, 粒子全局最优值和全局最优位置;

Step 5 计算监测邻域中粒子个数, 若满足 $N_\varphi > N \times \delta$, 转Step 6; 否则, 判断是否达到设定迭代数, 若满足, 则算法终止, 输出寻优结果, 否则, 转Step 2;

Step 6 计算监测邻域中粒子的适应度值, 根据式(6)进行适应度指数标定, 根据式(7)计算粒子的选中概率, 对选中的式(8)进行变异, 转Step 2.

3 经典测试函数测试(Classical test functions verification)

为了验证改进的粒子群算法的有效性, 本文选用了3种经典函数进行测试, 同时与惯性因子随迭代次数线性递减的PSO算法进行比较. 3种经典测试函数表示为

1) Sphere函数

$$f_1(\mathbf{X}) = \sum_{i=1}^D x_i^2, \quad (10)$$

式中 $x_i \in [-10, 10]$. 该函数由K. A. De Jong提出, 在 $x_i = 0, i = 1, 2, 3, \dots, D$ 时达到全局最小0.

2) Rastrigin函数

$$f_2(\mathbf{X}) = \sum_{i=1}^D \{x_i^2 - 100 \cos(2\pi x_i) + 10\}, \quad (11)$$

式中 $x_i \in [-5.12, 5.12]$. 在 $x_i = 0, i = 1, 2, 3, \dots, D$ 时达到全局最小0, 大约存在 $10D$ 个局部极值点.

3) Rosenbrock函数

$$f_3(\mathbf{X}) = \sum_{i=1}^{D-1} \{100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2\}, \quad (12)$$

式中 $x_i \in [-10, 10]$. Rosenbrock是很难极小化的病态函数, 理论全局最小值为0. 测试中, 测试函数维数 $D = 10$; IPSO与PSO算法参数设置如下: 种群个数 $N = 30$, 迭代次数 $\text{iter}_{\max} = 1000$, 认知因子 $c_1 = 2.05$, 社会因子 $c_2 = 2.05$, 限制因子 $\chi = 0.73$, 惯性因子最大值 $w_{\max} = 0.9$, 惯性因子最小值 $w_{\min} = 0.4$, 每一维的寻优范围为 $[-10, 10]$ (Rastrigin函数测试时为 $[-5.12, 5.12]$), 粒子最大速度 $v_{\max} = 6$ (Rastrigin函数测试时为3). 对于IPSO, 欧氏距离设定值 $\varepsilon = 0.1$, 监测邻域粒子浓度上限 $\delta = 0.16$, 适应度定标常数 $K = 0.005$, $\tau_c = 2$, 缓冲墙厚度相关因子 $L = 0.3$.

两种算法分别连续运行10次, 结果如表1所示. 为了更直观地了解IPSO和PSO的性能差异, 这里以Rosenbrock函数为基准函数, 图1为两种算法在 $x_i \in$

$[-10, 10]$ 时最优适应度进化比较图. 图2和图3分别为IPSO及PSO算法迭代过程中第1, 250, 750, 1000次迭代粒子种群分布图(第一、二维度).

表1 IPSO & PSO优化结果比较

Table 1 Comparison of optimization results for IPSO & PSO

函数名称	理论极值	PSO算法适应值			IPSO算法适应值		
		平均值	最优值	最差值	平均值	最优值	最差值
Sphere	0	0.652	0.024	2.77	3.027e-07	1.036e-07	8.384e-07
Rastrigin	0	24.76	7.886	62.8	5.374	1.99	8.955
Rosenbrock	0	297.6	14.32	882.4	6.967	5.186	9.461

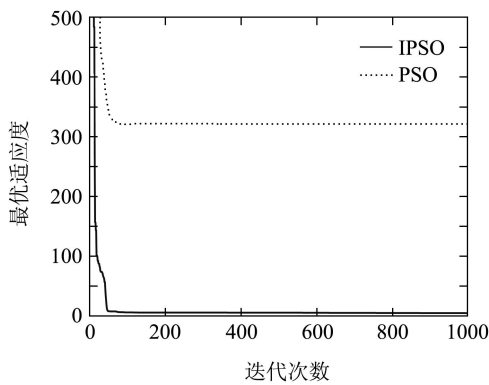
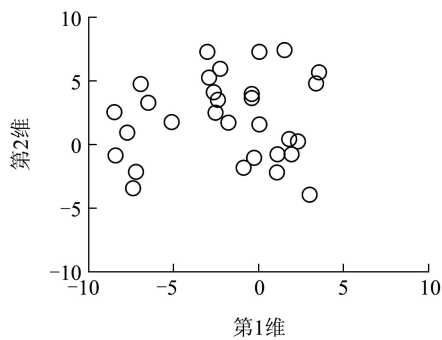


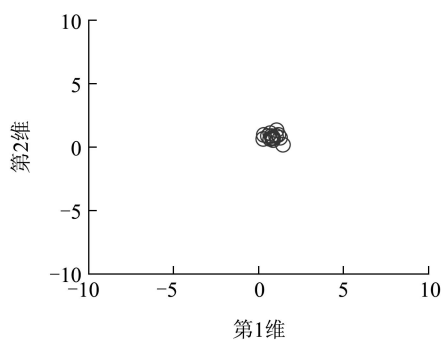
图1 Rosenbrock函数最优适应度进化曲线

Fig. 1 Evolutionary curves of optimum fitness for

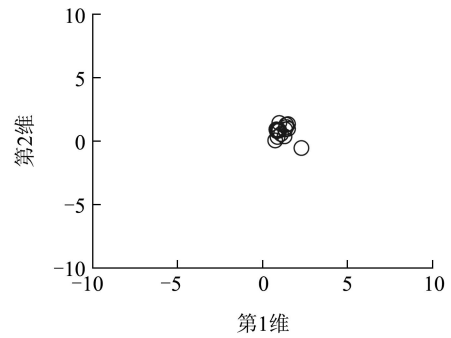
Rosenbrock



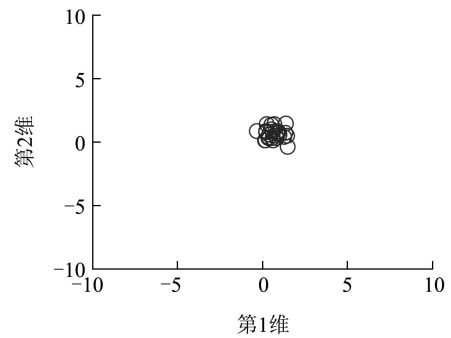
(a) 迭代为1时



(b) 迭代为250时



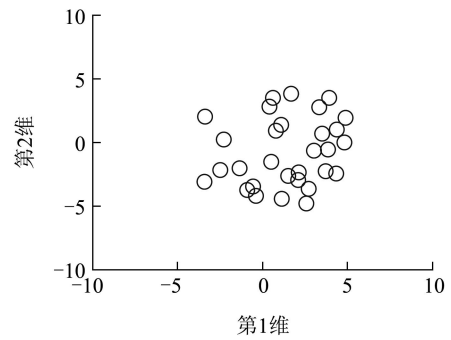
(c) 迭代为750时



(d) 迭代为1000时

图2 IPSO算法迭代过程中粒子种群分布图

Fig. 2 Particles distribution during the iterations for IPSO



(a) 迭代为1时

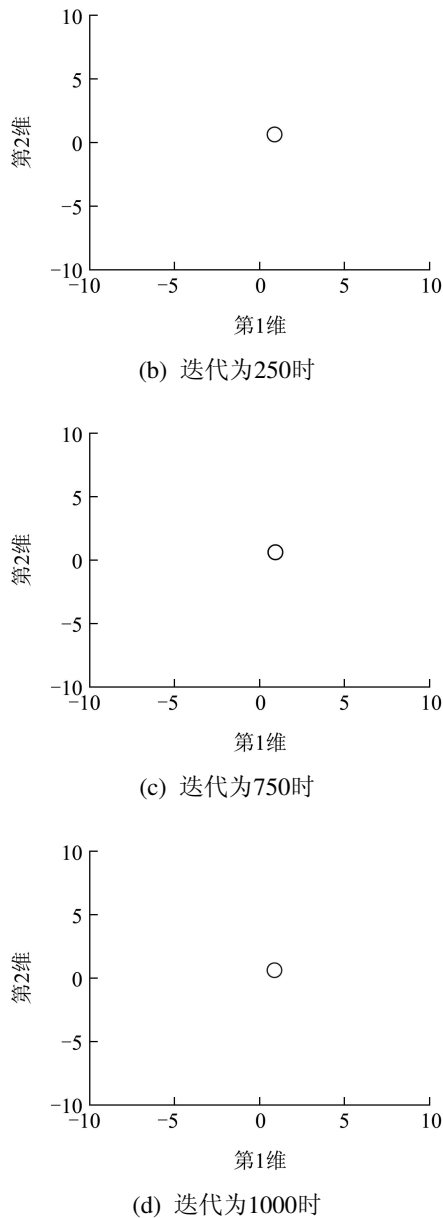


图 3 PSO算法迭代过程中粒子种群分布图

Fig. 3 Particle distribution during the iterations for PSO

由以上测试结果可以非常明显地看出IPSO算法的性能确实比PSO算法性能有了很大的改进. 从图2中看出, IPSO在整个算法迭代过程中, 粒子种群有向全局最优聚集的趋势, 但当种群过分集聚监测邻域的情况下会被驱散开来, 因此算法在整个迭代过程中粒子种群多样性得到很好的维护. 但反观PSO, 图3反映了PSO在算法中后期种群多样性急剧衰减, 整个种群陷入了局部极值点, 寻优结果很不理想. 上述实验证明了本文所提出的改进方法: 适应度指数定标, 边界缓冲墙确实能够克服传统粒子群算法早熟, 易陷入局部极值点的缺点.

4 IPSO 应用实例—PID 参数优化整定 (IPSO application example—PID parameter settings tuning)

4.1 性能指标(Performance index)

考查如图4所示的一个单位负反馈控制系统.

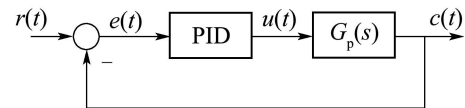


图 4 PID 控制系统

Fig. 4 PID control system

图中 $G_p(s)$ 为被控过程, PID控制器的传递函数为

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s, \quad (13)$$

式中 K_p, K_i, K_d 分别为比例增益、积分系数和微分系数. 通常控制系统的性能指标函数主要有误差绝对值积分IAE、误差平方积分ISE和误差绝对值乘以时间积分ITAE几种, IAE和ISE不受时间的约束, 易引起减小超调和缩短调节时间的矛盾, 而ITAE由于考虑了时间的约束, 是控制系统设计中最常用的性能指标之一. ITAE指标定义为

$$ITAE = \int_0^T t |e(t)| dt. \quad (14)$$

如果把误差 $e(t)$ 组成的泛函看成是一种“损失函数”, 那么式(14)的积分指标就可以看成是控制系统由一个状态转移到另一个状态时, 以某种损失为最小代价实现它的控制目标, 这时就认为这个系统有某种性能指标最小的最优控制律. 一般用采用5%或2%误差带所对应的调节时间 t_s 作为该积分的上限 T 参数. 由于误差绝对值 $|e(t)|$ 的存在, 式(14)难于用解析法进行求解, 通常选择数值方法进行求解. 具体步骤如下: 将连续时间以时间步长 Δt 进行离散化, 同时使时间 $T = m\Delta t$, m 为适当大小的整数, 得到离散时间向量 $t[n] (n = 0, 1, \dots, m)$. 测量出对应于离散时间向量 $t[n]$ 的误差向量 $e[n]$. 则当 Δt 足够小时, ITAE指标可以按照式(15)求得:

$$ITAE = \sum_{i=0}^m t[i] |e[i]| \Delta t. \quad (15)$$

利用改进的粒子群算法进行PID参数整定的过程实际上是一个优化问题. 这里, 令待整定的参数集 (K_p, K_i, K_d) 作为粒子在寻优空间的位置向量 $P(K_p, K_i, K_d)$, 而 K_p, K_i, K_d 分别被约束在 $[0, K_{pm}]$, $[0, K_{im}]$, $[0, K_{dm}]$, 并选取ITAE性能指标作

为算法的适应度函数. 通过IPSO算法的迭代, 找出一组参数(K_p, K_i, K_d), 使得系统响应的ITAE值达到最小.

4.2 典型被控过程及PID参数整定结果(Typical controlled processes and PID tuning results)

为了验证IPSO算法的控制性能效果, 首先选取5类典型工业过程控制模型 $G_p(s)$ 作为研究对象, 见表2所示, 即自衡的非振荡一阶和二阶时滞过程, 无自衡的非振荡过程, 具有反向特性的非最小相位过程和高阶过程.

表2 通用被控过程数学模型

Table 2 Models for general controlled processes

模型	数学模型	模型性质
I	$\frac{e^{-\tau s}}{1+s}$ $\tau = 0.5, 1, 2$	一阶时滞过程 本文取 $\tau = 0.5$
II	$\frac{e^{-\tau s}}{(1+s)^2}$ $\tau = 0.5, 1, 2$	二阶时滞过程 本文取 $\tau = 1$
III	$\frac{e^{-s}}{s(1+s)}$	无自衡过程
IV	$\frac{1-\alpha s}{(1+s)^2}$ $\alpha = 0.5, 1$	非最小相位过程 本文取 $\alpha = 0.5$
V	$\frac{1}{(1+s)^n}$ $n = 4, 6, 8$	高阶过程 本文 $n = 6$

利用本文改进的粒子群算法(IPSO)对表2中5个典型过程模型算例进行PID控制器参数整定, 其中IPSO算法的参数设置如下: 种群个数 $N = 60$; 迭代次数 $iter_{max} = 100$; 粒子群算法参数中, 限制因子 χ 影响算法的收敛性; 根据Clerc提供的计算方法, 这里取限制因子 $\chi = 0.73$ 能起到较好结果; c_1 和 c_2 的选取对寻优结果也会产生影响, 认知因子 c_1 的大小主要影响粒子个体在寻优空间进行全局探索的能力, 为了保证粒子群能够找到全局极值点, 本文取 $c_1 = 2.05$. 社会因子 c_2 的大小主要影响粒子个体在当前全局最优点附近进行局部开发的能力. 为了提高寻优精度, 本文取 $c_2 = 2.05$; 惯性因子 w 的大小主要影响粒子的探索和开发能力, 在寻优前期 w 须比较大, 保证粒子能在全局进行搜索, 寻优后期 w 必须比较小, 使粒子进行局部开发, 本文的算法采用惯性因子随迭代次数线性下降的方法, 惯性因子最大值 $w_{max} = 0.9$, 惯性因子最小值 $w_{min} = 0.4$; 每一维的寻优范围为 $[0, 10]$;

粒子最大速度 $v_{max} = 6$; 经过大量仿真实验证明下列设置能够取得较好的算法性能: 欧氏距离设定值 $\varepsilon = 0.1$; 监测邻域粒子浓度上限 $\delta = 0.16$; 适应度定标常数 $K = 0.005$; $\tau_c = 2$; 缓冲墙厚度相关因子 $L = 0.4$. 表3给出了5种典型被控过程的经典PID参数Z-N整定和基于IPSO的PID参数整定值及性能指标ITAE, 包括超调量 $\sigma\%$ 和2%误差带准则计算的调节时间 t_s . 从表3的结果看出基于改进粒子群算法(IPSO)的性能指标相当优越.

表3 经典Z-N与IPSO法PID参数整定值和性能指标对比

Table 3 PID parameters for Z-N & IPSO and their performances

参数	模型				
	I	II	III	IV	V
k_p	2.2800	1.6200	0.6780	2.3880	1.4220
	1.7945	1.2876	0.8098	1.9988	1.0420
k_i	2.6576	0.6707	0.1855	1.6959	0.2613
	1.2426	0.4497	0.1400	0.7449	0.1967
k_d	0.4890	0.9782	0.6169	0.8406	1.9346
	0.3142	0.8685	0.5938	0.8908	1.6709
$\sigma\%$	38.2000	27.6000	71.3000	43.5000	22.7000
	3.2000	2.8000	63.5000	1.4000	3.4000
t_s	3.0100	8.7500	20.6000	5.7400	20.5100
	2.1200	6.3500	15.0900	2.8900	15.0600
ITAE	0.8683	4.4524	30.2060	2.7027	26.1630
	0.3797	3.0419	16.7180	1.0048	17.5900

4.3 仿真实验(Simulation experiment)

为了进一步验证IPSO算法的效果, 利用MATLAB仿真在30 s时测试加入了幅度为0.2的干扰, 考察5种典型模型控制系统的单位阶跃响应, 结果如图5~图9所示.

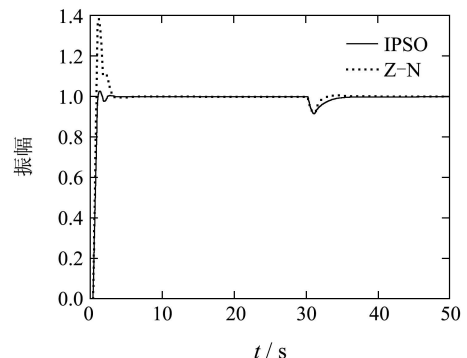


图5 一阶时滞过程闭环控制系统的阶跃响应
Fig. 5 Closed-loop step response with process I

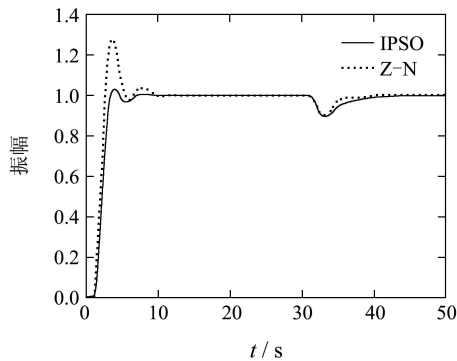


图 6 二阶时滞过程闭环控制系统的阶跃响应
Fig. 6 Closed-loop step response with process II

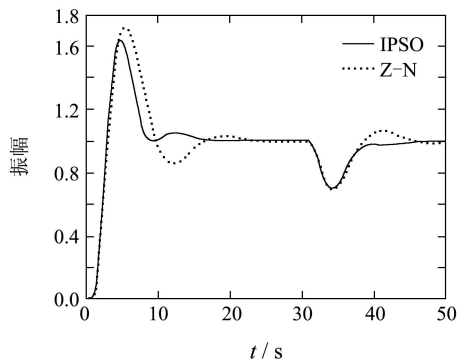


图 7 无自衡过程闭环控制系统的阶跃响应
Fig. 7 Closed-loop step response with process III

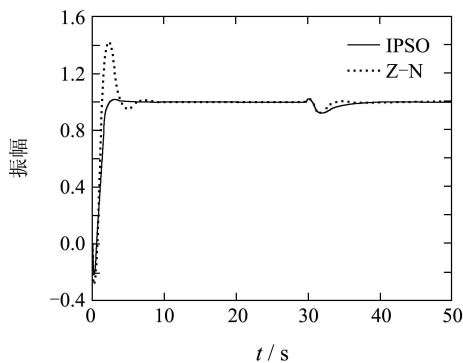


图 8 非最小相位过程闭环控制系统的阶跃响应
Fig. 8 Closed-loop step response with process IV

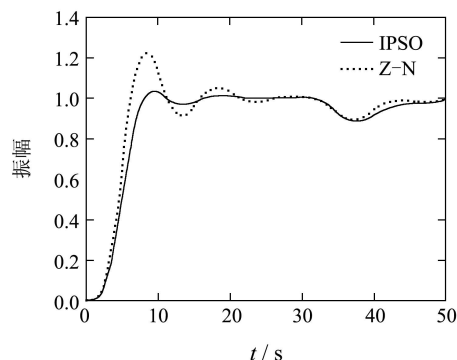


图 9 高阶过程闭环控制系统的阶跃响应
Fig. 9 Closed-loop step response with process V

由仿真结果可以看出, 通过适应度定标和边界缓冲墙的改进粒子群算法(IPSO)得到的控制结果的性能是相当满意的。

5 结论(Conclusion)

本文指出了传统PID参数整定方法和标准粒子群算法在高维复杂函数寻优过程中的不足. 通过引入监测邻域的概念提出了一种基于指数定标适应度概率变异和边界缓冲墙相结合的粒子群算法(IPSO), 对传统粒子群算法存在易陷入局部极值点、种群多样性削减、维数灾难等问题进行了有效的改进, 对3种典型测试函数的性能测试结果看出IPSO算法的优越性, 并将该算法应用于PID控制器参数整定的优化设计过程中. 智能计算和仿真实验表明该算法得到的结果性能指标优越, 控制效果显著. 而较之其他优化的PID整定方法而言, 采用改进的混合粒子群算法实现较简单、参数设置经验性要求不高、寻优速度快, 经过改进能有效避免陷入局部极值点. 另外, 基于IPSO的PID参数整定结果鲁棒性好, 有较强的抗干扰能力, 对系统的特征属性依赖性较少, 同时也适用于非线性控制系统的参数整定, 因而具有更大的优越性. 因此在PID参数优化整定方面, 粒子群算法可以发挥更大的作用. 作者下一阶段还需要进一步研究算法收敛速度和精度与缓冲墙厚度的关系。

参考文献(References):

- [1] KNOSPE C. PID control[J]. *IEEE Control System Magazine*, 2006, 26(1): 30 – 31.
- [2] ZIEGLER J G, NICHOLS N B. Optimum settings for automatic controllers[J]. *Transaction on American Society of Mechanical Engineers, Journal of Dynamic Systems, Measurement, and Control*, 1942, 64(8): 759 – 768.
- [3] HAGGLUND T, ASTROM K J. Revisiting the Ziegler-Nichols tuning rulers for PI control[J]. *Asian Journal of Control*, 2002, 4(4): 364 – 380.
- [4] ASTROM K J, HAGGLUND T. Revisiting the Ziegler-Nichols step response method for PID control[J]. *Journal of Process Control*, 2004, 14(6): 635 – 650.
- [5] WANG P, KWOK D P. Auto-tuning of classical PID controllers using an advanced genetic algorithm[C] // *Proceedings of International Conference on Power Electronics and Motion Control*. San Diego, CA, USA: IEEE, 1992, 23: 1224 – 1229.
- [6] VAROL H A, BINGUL Z. A new PID tuning technique using ant algorithm[C] // *Proceedings of American Control Conference*. Boston, Massachusetts, USA: IEEE, 2004: 2154 – 2159.

- [7] KENNEDY J, EBERHART R C. Particle swarm optimization[C] //IEEE International Conference on Neural Network. New Jersey, USA: IEEE, 1995, 4: 1942 – 1948.
- [8] LIN C J, WANG J G, LEE C Y. Pattern recognition using neural-fuzzy networks based on improved particle swarm optimization[J]. *IEEE Transaction on Systems, Man and Cybernetics*, 2009, 36(3): 5402 – 5410.
- [9] LAI C C, WU C H, TSAI M C. Feature selection using particle swarm optimization with application in spam filtering[J]. *International Journal of Innovative Computing, Information and Control*, 2009, 5(2): 423 – 432.
- [10] SRINIVAS J, GIRI R, YANG S H. Optimization of multi-pass turning using particle swarm intelligence[J]. *International Journal of Advanced Manufacturing Technology*, 2009, 40(1/2): 56 – 66.
- [11] PARROT D, LI X D. Locating and tracking multiple dynamic optima by a particle swarm model using speciation[J]. *IEEE Transaction on Evolutionary Computation*, 2006, 10(4): 440 – 458.
- [12] CLERC M. The swarm and the queen: towards a deterministic and adaptive particle swarm optimization[C] //Proceedings of the Congress on Evolutionary Computation. New Jersey, USA: IEEE, 1999: 1951 – 1957.
- [13] SHI Y, EBERHART R C. A modified particle swarm optimizer[C] //IEEE International Conference of Evolutionary Computation. Anchorage, Alaska, USA: IEEE, 1998: 69 – 73.

作者简介:

杨智 (1961—), 男, 教授, 主要研究方向为复杂系统的建模与控制、医学信号处理等, E-mail: issyz@mail.sysu.edu.cn;

陈志堂 (1987—), 男, 现为中山大学信息科学与技术学院自动化系本科生, 目前研究方向为粒子群算法、遗传算法等, E-mail: ztangchen@gmail.com;

范正平 (1977—), 男, 讲师, 研究方向为复杂网络的动力学分析等, E-mail: zhpfan@yahoo.com.cn;

李晓东 (1977—), 男, 教授, 主要研究方向为优化控制理论及应用等, E-mail: lixd@mail.sysu.edu.cn.