

基于改进粒子群优化算法的预测控制

姜苏英

(宝鸡文理学院 电子电气工程学院, 陕西 宝鸡 721013)

摘要: 在实际工业过程中预测控制算法应用广泛, 但是对于多变量预测控制算法其参数较多, 且各个参数之间相互耦合, 故整定其参数比较复杂, 鉴于此提出一种基于改进粒子群算法的预测控制参数优化算法; 该算法的基本思想是将生物寄生行为机制引入到粒子群优化算法中, 形成双种群粒子群优化算法, 使用该改进粒子群算法对多变量预测控制算法的参数进行离线优化, 从而确定预测控制算法参数的最优取值; 最后, 将文章算法用于冷热水系统液位和温度的控制, 并通过仿真将该算法与标准粒子群优化算法相比较, 仿真结果表明使用该算法对多变量预测控制的参数进行优化整定时, 系统的阶跃响应应具有抗干扰性能好、超调量小、调节时间短等优点。

关键词: 预测控制; 标准粒子群优化; 参数优化; 多变量; 耦合

Predictive Control Based on Improved PSO Algorithm

Jiang Suying

(College of Electrical & Electronic Engineering, Baoji University of Arts and Sciences, Baoji 721013, China)

Abstract: In the actual industrial process, the predictive control algorithm is widely used, but the parameters of the multivariable predictive control algorithm are more and the parameters are coupled with each other, so tuning the parameters are complicated. In view of this, a predictive control parameters optimization algorithm based on improved particle swarm optimization is proposed. The basic idea of this algorithm is that the mechanism of parasitic behavior is introduced into the particle swarm optimization algorithm to form a two-population particle swarm optimization algorithm, the improved particle swarm algorithm is used to off-line optimization the parameters of the multivariable predictive control algorithm, so as to determine the optimal values of predictive control algorithm parameters. Finally, the algorithm is used to control the liquid level and temperature of a cold and hot water system, and the algorithm is compared with the standard particle swarm optimization algorithm through simulation. The simulation results show that the proposed algorithm is used to optimize the parameters of multivariable predictive control, the step response of system has advantages such as good anti-jamming performance, little overshoot, short adjusting time and so on.

Keywords: predictive control; standard particle swarm optimization; parameter optimization; multivariable; coupling

0 引言

预测控制是一种先进的计算机控制技术^[1], 该算法自提出以来在工业过程中得到了广泛应用。动态矩阵控制 (dynamic matrix control, DMC) 算法属于预测控制算法中的一种, 该算法适用于多变量过程的控制。

与传统控制算法相比, 采用动态矩阵算法对多变量过程进行控制时, 可以减小超调, 获得良好的跟踪性和较强的鲁棒性, 但是需要整定的动态矩阵控制参数较多, 且各个参数之间存在较强的关联性, 不同参数组合对系统控制效果影响较大。因此对动态矩阵控制算法的参数整定方法进行研究很有必要。

对动态矩阵控制参数整定大多采用试凑法结合仿真的方法, 但是采用这种方法需要花费大量的时间, 而且有可能得到的控制参数并不是最优的参数, 不能达到最佳的控制效果^[2]。因此, 需要寻找一种快速、简单的算法优化动态矩阵参数。

随着智能控制算法的快速发展, 很多智能优化算法被引入到动态矩阵控制的参数优化, 如: 标准粒子群算法^[2-5]、遗传

算法^[6]等。文献 [2] 采用标准粒子群算法优化动态矩阵控制的参数; 文献 [6] 采用遗传算法优化灰色 PI 预测控制参数。虽然这些算法可以实现对动态矩阵控制参数的优化, 但各自都存在一定的局限性, 如: 遗传算法迭代过程比较复杂, 迭代速度慢, 易产生早熟收敛等问题; 标准粒子群算法存在早熟、易于陷入局部最优值及维数灾难等问题。

为了克服手动调节动态矩阵控制参数的不确定性和繁琐性, 粒子群算法在搜索后期易于陷入局部最优等问题, 本文采用一种改进粒子群算法优化动态矩阵控制参数。最后将该算法应用于冷热水系统进行仿真研究, 仿真结果表明该算法优于其他几种算法, 具有更好的控制性能, 可以有效应用于多变量动态矩阵控制参数的优化。

1 基于改进粒子群优化的 DMC 算法

1.1 多变量动态矩阵控制

DMC 算法包括 3 个部分: 预测模型、滚动优化、反馈校正。

1) 预测模型。

对于 m 个输入 r 个输出的系统, 假定已测得每一个输出 y_i 对每一个输入 u_j 的阶跃响应 $a_{ij}(t)$, 则由它们在采样点的值可构成模型向量 $a_{ij} = [a_{ij}(1), a_{ij}(2), \dots, a_{ij}(N)]^T$, 其中 $i = 1, 2, \dots, r, j = 1, 2, \dots, m, N$ 为建模时域。

收稿日期: 2017-10-18; 修回日期: 2018-03-11。

基金项目: 宝鸡文理学院硕士科研启动项目 (ZK2017096)。

作者简介: 姜苏英 (1990-), 女, 陕西商南人, 助教, 硕士研究生, 主要从事工业过程智能控制算法方向的研究。

假设 u_j 从 k 时刻依次有 M 个增量变化 $\Delta u_j(k), \Delta u_j(k+1), \dots, \Delta u_j(k+M-1)$ 时, 则 y_i 在未来 P 个时刻的输出预测值为:

$$\tilde{Y}_{i,PM}(k) = \tilde{Y}_{i,P_0}(k) + \sum_{j=1}^m A_{ij} \Delta U_{j,M}(k) \quad (1)$$

其中:

$$\tilde{Y}_{i,PM}(k) = \begin{bmatrix} \tilde{y}_{i,M}(k+1|k) \\ \tilde{y}_{i,M}(k+2|k) \\ \vdots \\ \tilde{y}_{i,M}(k+P|k) \end{bmatrix};$$

$$\tilde{Y}_{i,P_0}(k) = \begin{bmatrix} \tilde{y}_{i,0}(k+1|k) \\ \tilde{y}_{i,0}(k+2|k) \\ \vdots \\ \tilde{y}_{i,0}(k+P|k) \end{bmatrix};$$

$$A_{ij} = \begin{bmatrix} a_{ij}(1) & 0 & \cdots & 0 \\ a_{ij}(2) & a_{ij}(1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{ij}(M) & a_{ij}(M-1) & \cdots & a_{ij}(1) \\ a_{ij}(M+1) & a_{ij}(M) & \cdots & a_{ij}(2) \\ \vdots & \vdots & \ddots & \vdots \\ a_{ij}(P) & a_{ij}(P-1) & \cdots & a_{ij}(P-M+1) \end{bmatrix};$$

$$\Delta U_{j,M}(k) = \begin{bmatrix} \Delta u_j(k) \\ \Delta u_j(k+1) \\ \vdots \\ \Delta u_j(k+M-1) \end{bmatrix};$$

M 为控制时域长度; P 为预测时域长度; $(k+i|k)$ 表示在 k 时刻对 $k+1$ 时刻的预测。

因此, 多变量系统预测模型为:

$$\tilde{Y}_{PM}(k) = \tilde{Y}_{P_0}(k) + \mathbf{A} \Delta \mathbf{U}(k) \quad (2)$$

其中:

$$\tilde{Y}_{PM}(k) = \begin{bmatrix} \tilde{Y}_{1,PM}(k) \\ \tilde{Y}_{2,PM}(k) \\ \vdots \\ \tilde{Y}_{r,PM}(k) \end{bmatrix};$$

$$\tilde{Y}_{P_0}(k) = \begin{bmatrix} \tilde{Y}_{1,P_0}(k) \\ \tilde{Y}_{2,P_0}(k) \\ \vdots \\ \tilde{Y}_{r,P_0}(k) \end{bmatrix};$$

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{r1} & A_{r2} & \cdots & A_{rm} \end{bmatrix};$$

$$\Delta \mathbf{U}(k) = \begin{bmatrix} \Delta U_{1,M}(k) \\ \Delta U_{2,M}(k) \\ \vdots \\ \Delta U_{m,M}(k) \end{bmatrix}$$

2) 滚动优化。

要求每一个输出 y_i 在未来 P 个时刻紧密跟踪相应的期望值 w_i , 为了避免控制增量变化剧烈, 对 m 个控制增量的大小均加以软约束。 \mathbf{K} 时刻的性能指标可写为:

$$\min J(k) = [\mathbf{W}(k) - \tilde{\mathbf{Y}}_{PM}(k)]^T \mathbf{Q} [\mathbf{W}_p(k) - \tilde{\mathbf{Y}}_{PM}(k)] +$$

$$\Delta \mathbf{U}^T(k) \mathbf{R} \Delta \mathbf{U}(k) \quad (3)$$

其中:

$$\mathbf{W}(k) = \begin{bmatrix} W_1(k) \\ W_2(k) \\ \vdots \\ W_r(k) \end{bmatrix};$$

$$W_i(k) = \begin{bmatrix} w_i(k+1) \\ w_i(k+1) \\ \vdots \\ w_i(k+P) \end{bmatrix}, i = 1, 2, \dots, r;$$

$$\mathbf{Q} = \text{diag}\{Q_1, Q_2, \dots, Q_r\},$$

$Q_i = \text{diag}\{q_i(1), q_i(2), \dots, q_i(P)\}, i = 1, 2, \dots, r, Q_i$ 为误差矩阵;

$\mathbf{R} = \text{diag}\{R_1, R_2, \dots, R_m\}, R_j = \text{diag}\{r_j(1), r_j(2), \dots, r_j(M)\}, j = 1, 2, \dots, m, R_j$ 为控制权矩阵。

当不考虑约束时, 由预测模型可求出使性能指标最优的控制增量为:

$$\Delta \mathbf{U}(k) = (\mathbf{A}^T \mathbf{Q} \mathbf{A} + \mathbf{R})^{-1} \mathbf{A}^T \mathbf{Q} [\mathbf{W}(k) - \tilde{\mathbf{Y}}_{P_0}(k)] \quad (4)$$

所求的 M 个控制增量中只执行第一个控制增量:

$$\Delta \bar{\mathbf{U}}(k) = \mathbf{D} [\mathbf{W}(k) - \tilde{\mathbf{Y}}_{P_0}(k)] \quad (5)$$

其中:

$$\mathbf{D} = \mathbf{L} (\mathbf{A}^T \mathbf{Q} \mathbf{A} + \mathbf{R})^{-1} \mathbf{A}^T \mathbf{Q}$$

$$\triangleq \begin{bmatrix} d_{11}^T & d_{12}^T & \cdots \\ d_{21}^T & d_{22}^T & \cdots \\ \cdots & \cdots & \ddots \\ d_{m1}^T & d_{m2}^T & \cdots \end{bmatrix};$$

$$\mathbf{L} = \begin{bmatrix} \theta & 0 & \cdots & 0 \\ 0 & \theta & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \theta \end{bmatrix}, \mathbf{L} \text{ 为 } m \times mM \text{ 维矩阵, } \theta = [1, 0,$$

$\dots, 0]$ 是一个 M 维向量;

$d_{ji}^T (j = 1, 2, \dots, r; i = 1, 2, \dots, m)$ 均为 P 维向量。

3) 反馈校正。

由于存在模型失配、环境干扰等因素, 故上式的预测值可能会偏离实际值, 因此及时利用实时信息进行反馈校正。在 k 时刻实施控制后, 对象在未来时刻的各输出值:

$$\tilde{Y}_{N_1}(k) = \tilde{Y}_{N_0}(k) + \bar{\mathbf{A}} \Delta \bar{\mathbf{U}}(k) \quad (6)$$

$k+1$ 时刻测得对象实际输出 $y_i(k+1)$, 则预测误差为:

$$\mathbf{E}(k+1) = \begin{bmatrix} e_1(k+1) \\ e_2(k+1) \\ \vdots \\ e_r(k+1) \end{bmatrix} = \begin{bmatrix} y_1(k+1) - \tilde{y}_{1,1}(k+1|k) \\ y_2(k+1) - \tilde{y}_{2,1}(k+1|k) \\ \vdots \\ y_r(k+1) - \tilde{y}_{r,1}(k+1|k) \end{bmatrix}$$

经校正的预测向量为:

$$\tilde{Y}_{cor}(k+1) = \tilde{Y}_{N_1}(k) + \mathbf{H} \mathbf{E}(k+1) \quad (7)$$

其中: \mathbf{H} 为误差校正矩阵; 通常取:

$$\mathbf{H} = \begin{bmatrix} h_{11} & 0 & \cdots & 0 \\ 0 & h_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{rr} \end{bmatrix}; h_{ii} = \begin{bmatrix} h_{ii}(1) \\ h_{ii}(2) \\ \vdots \\ h_{ii}(N) \end{bmatrix}$$

经过校正的预测向量 $\tilde{Y}_{cor}(k+1)$ 可通过移位作为 $k+1$ 时刻的初始预测值。

$$\tilde{Y}_{N0}(k+1) = S_0 \tilde{Y}_{cor}(k+1) \quad (8)$$

其中:

$$S_0 = \begin{bmatrix} S & 0 & 0 & 0 \\ 0 & S & \cdots & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & S \end{bmatrix}; S = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

有了 $\tilde{Y}_{N0}(k+1)$, 则可进行 $k+1$ 时刻的优化计算, 求出 $\Delta \bar{U}(k+1)$ 。

通过上面分析可知, 使用多变量动态矩阵控制算法对多变量系统控制时, DMC 需要整定的参数较多, 如果手动调节参数将会比较费时, 而采用标准粒子群算法, 容易陷入局部最优, 因此本文将采用一种改进 PSO 算法即基于生物寄生行为的改进 PSO 算法优化整定多变量 DMC 参数。

1.2 基于生物寄生行为的改进 PSO 算法

1) 基本粒子群优化算法。

基本粒子群算法的思想: 设在 D 维搜索空间中, 由 m 个粒子构成一个种群。第 i 个粒子的当前位置为 $x_i^t = [x_{i1}^t, x_{i2}^t, \dots, x_{iD}^t]$, 粒子的飞行速度为 $v_i^t = [v_{i1}^t, v_{i2}^t, \dots, v_{iD}^t]$ 。其中, $x_{ij}^t \in [L_d, U_d]$, L_d, U_d 分别为搜索空间的下限和上限。 $v_{ij}^t \in [v_{\min,j}, v_{\max,j}]$, $v_{\min,j}, v_{\max,j}$ 分别为 j 维粒子飞行的最大速度和最小速度, 通常取 $v_{\max,j} = -v_{\min,j}$ 。该粒子所经历的最优位置为 $p_i^t = [p_{i1}^t, p_{i2}^t, \dots, p_{iD}^t]$, 种群的全局最优位置为 $p_g^t = [p_{g1}^t, p_{g2}^t, \dots, p_{gD}^t]$, 则第 i 个粒子根据下式更新自己的位置和速度。

$$v_{ij}^{t+1} = v_{ij}^t + c_1 r_1 (p_{ij}^t - x_{ij}^t) + c_2 r_2 (p_{gd}^t - x_{ij}^t) \quad (9)$$

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \quad (10)$$

其中: $i = 1, 2, \dots, m$ 为粒子的个数; $j = 1, 2, \dots, D$ 为粒子的维数; t 为当前进化代数; v_{ij}^t 为粒子 i 在第 t 次迭代中第 j 维的速度; x_{ij}^t 为粒子 i 在第 t 次迭代中第 j 维的当前位置; r_1, r_2 为介于 $[0, 1]$ 的随机数; c_1, c_2 为学习因子, 为非负常数。

2) 改进 PSO 算法。

将生物寄生行为的规律引入到粒子群算法中形成基于生物寄生行为的双种群改进粒子群算法 PSOPB (particle swarm optimization based on parasitic behavior)。在 PSOPB 算法中将粒子分为两个种群, 一个为寄生群 (Swarm^P), 其粒子数量为 N^P , 另一个为宿主群 (Swarm^H), 其粒子数量为 N^H 。两个种群之间会发生兼性寄生行为, 两个种群间隔一定的迭代次数 k , 寄生群从宿主群中获取适应度好的粒子, 宿主群将从寄生群中得到同样数量的适应度差的粒子, 之后宿主群中适应度较差的粒子以一定的比率 γ 淘汰。为了使宿主群粒子数量不变, 将淘汰的粒子使用重新初始化的粒子代替^[7-8]。两个种群交换的粒子数量设为 $\text{round}(\mu * N^P)$ 。 μ 的值设定为随着迭代次数的增加而呈线性递减变化:

$$\mu = (\mu_{start} - \mu_{end}) \times \frac{(t_{max} - t)}{t_{max}} + \mu_{end} \quad (11)$$

其中: t 为当前迭代次数; t_{max} 为最大迭代次数; μ_{start} 为 μ 的初始值, 也是最大值; μ_{end} 为迭代结束时的 μ 值, 也是最小值。

为了提高该算法的搜索精度和收敛速度, 寄生群的粒子采用带收缩因子的 PSO 算法更新速度, 速度更新公式如下:

$$v_{id}^{t+1} = K(v_{id}^t + c_1 r_1 (p_{id}^t - x_{id}^t) + c_2 r_2 (p_{gd}^t - x_{id}^t)) \quad (12)$$

式中, 收缩因子 K 由下式确定:

$$K = \frac{2}{|2 - \varphi + \sqrt{\varphi^2 - 4\varphi}|} \quad (13)$$

其中: $\varphi = c_1 + c_2, \varphi > 4$ 。

宿主群中的粒子根据下式更新速度:

$$v_{id}^{t+1} = \begin{cases} \lambda (v_{id}^t + c_{11} r_{11} (p_{id}^t - x_{id}^t) + c_{12} r_{12} (p_{gd}^t - H - x_{id}^t) + c_{13} r_{13} (p_{gd}^t - P - x_{id}^t)) \\ \lambda (v_{id}^t + c_1 r_{21} (p_{id}^t - x_{id}^t) + c_2 r_{22} (p_{gd}^t - H - x_{id}^t)) \\ fp_{g,H}^t \geq fp_{g,P}^t \\ fp_{g,H}^t \geq fp_{g,P}^t \end{cases} \quad (14)$$

其中: $c_{11}, c_{12}, c_{13}, c_1, c_2$ 是加速因子; $r_{11}, r_{12}, r_{13}, r_{21}, r_{22}$ 是介于 $[0, 1]$ 之间的随机数; $p_{g,H}^t = [p_{g1,H}^t, p_{g2,H}^t, \dots, p_{gD,H}^t]$ 是 t 次迭代时宿主群的群体最优位置; $p_{g,P}^t = [p_{g1,P}^t, p_{g2,P}^t, \dots, p_{gD,P}^t]$ 是 t 次迭代时寄生群的群体最优位置; $fp_{g,H}^t$ 和 $fp_{g,P}^t$ 分别表示 t 次迭代时宿主群和寄生群最优个体的适应值。

当寄生群的群体最优位置在连续 η 次迭代中都没有进化时, 随机选择寄生群的一个粒子 i 在任一维 d 上的速度按下式发生变异。

$$v_{id}^{t+1} = \begin{cases} 0.5 \times v_{\max} \times r_3 & r_4 < 0 \\ -0.5 \times v_{\max} \times r_3 & otherwise \end{cases} \quad (15)$$

其中: r_3, r_4 为 $[0, 1]$ 之间的随机数。

3) 适应度函数。

DMC 参数的优化整定, 实际就是利用基于生物寄生行为的改进 PSO 算法整定 DMC 的参数。其本质是基于一定目标函数的参数寻优问题, 目标函数也就是适应度函数, 适应度函数是随着迭代的运行而逐渐减小的, 适应度函数值越小, 则表明粒子越靠近全局最优解。为了获取对象满意的过渡过程动态特性, 采用 ITAE 性能指标作为参考的最小目标函数; 为了避免控制量多大, 在目标函数中加入控制输入的平方项; 为了避免超调, 将超调量作为目标函数的一项^[9-10]。本文取下式作为目标函数为:

$$J = \int_0^{\infty} (\omega_1 |e(t)| + \omega_2 u^2(t) + \omega_3 |e(t)|) dt + \omega_4 t_u \quad (16)$$

式中, $e(t)$ 为系统误差; $u(t)$ 为控制器输出; t_u 为上升时间; $\omega_1, \omega_2, \omega_3, \omega_4$ 均为权值, 且 $\omega_1 \geq \omega_1$ 。

本文取: $\omega_1 = 0.999, \omega_2 = 0.001, \omega_3 = 2.0, \omega_4 = 100$ 。

1.3 算法实现

本文使用改进 PSO 算法优化多变量 DMC 参数, 具体步骤如下:

1) 基本参数设置。根据需要优化的 DMC 参数的个数设定粒子维数; 设定群体规模 M , 最大迭代次数 t_{max} , 加速因子 $c_{11}, c_{12}, c_{13}, c_1, c_2$, 搜索范围。

2) 将群体等分为寄生群和宿主群两个种群, 并初始化两个种群。

3) 粒子性能评估。根据式 (16) 计算两个种群粒子的适

应度值，并确定宿主群和寄生群中的最优位置。

4) 粒子状态更新。根据式 (12)、(10) 对寄生群中所有个体进行速度、位置更新；根据式 (14)、(10) 对宿主群中所有个体进行速度、位置更新。

5) 更新宿主群和寄生群中的最优位置。计算两个种群每个粒子的适应度值，并将其适应度值与其经过的最好位置进行比较，如果目前适应度值好，则将其作为当前的最优位置。每个种群的粒子将其适应度值与其所在种群的最好位置进行比较，如果较好，那么将其设置为所在种群最优位置。

6) 两个种群间隔一定的迭代次数 k ，寄生群从宿主群中获取适应度好的粒子，宿主群将从寄生群中得到同样数量的适应度差的粒子，之后宿主群中适应度较差的粒子将以一定的比率 γ 淘汰，并使用重新初始化的粒子代替淘汰掉的粒子。

7) 判断寄生群的群体最优位置是否在连续 η 次迭代中都没有进化，如果是则随机选择寄生群的一个粒子发生变异。

8) 记录当前整个群体中的最佳个体，如果满足精度要求或整个进化已达到最大迭代次数，则终止算法。否则转至步骤 4)。

2 冷热水混合系统

冷热水混合系统是一套实验装置件，它可以真是的模拟常见的典型工业过程，其结构图如图 1 所示。它主要由大水槽、加热水槽、混合水槽、开关电磁阀、水泵、变频器 1，变频器 2，铂热电阻、压力传感器、计算机等组成。

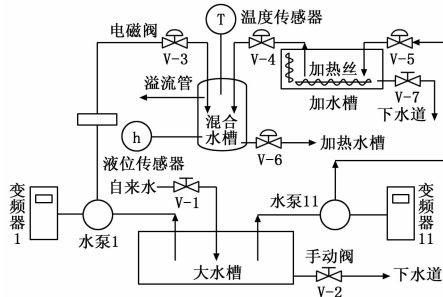


图 1 冷热水混合系统

冷水注入大水槽中，并保证其液位维持在安全工作范围。水泵 1 将冷水从大水槽中泵到上面，进入混合水槽；水泵 2 将冷水从大水槽泵入热水槽，接着有一部分进入混合水槽和冷水混合，另一部分由水道排出^[12]。液位和温度是该系统的主要被控参数，本文主要研究冷热水混合系统的液位和温度控制。这是一个双输入双输出具有耦合时滞特性的系统，输入为冷、热水流量，输出为液位和温度。

本文采用文献 [13] 中建立的冷热水系统数学模型，该系统的传递函数模型为：

$$\begin{bmatrix} G_{11}(s) \\ G_{21}(s) \end{bmatrix} = \begin{bmatrix} \frac{1.7e^{-30s}}{7s+1} & \frac{0.5e^{-27s}}{8s+1} \\ -\frac{0.6e^{-25s}}{10s+1} & \frac{1.5e^{-28s}}{9s+1} \end{bmatrix} \quad (17)$$

3 系统仿真与分析

以上述冷热水混合系统作为仿真对象，进行仿真研究。在模型匹配的情况下，使用本文算法对系统进行控制，输入为单

位阶跃信号。参数设置如下：种群规模为 100，粒子维度为 8，最大迭代次数为 100；加速因子 $c_1 = 2.05, c_2 = 2.05, c_{11} = 1.367, c_{12} = 1.367, c_{13} = 1.367$ ；寄生行为发生后宿主群淘汰一半的粒子，即 $\gamma = 0.5$ ；搜索空间, M 取 1, $P \in [60, 150], h_1 \in [0.1, 1], q_1 \in [0.1, 1], r_1 \in [0.1, 1], h_2 \in [0.1, 1], q_2 \in [0.1, 1], r_2 \in [0.1, 1]$ 。

为了验证本文算法的有效性，将本文算法与参考文献 [2] 中的 SPSO 算法控制效果进行比较。使用 SPSO 对 DMC 参数进行优化时，粒子搜索范围、粒子维度，种群规模及最大迭代次数均与本文算法对应参数取值一致。其他参数设置为：惯性权重 $\omega_{\max} = 0.9, \omega_{\min} = 0.4$ ；学习因子 $c_1 = 2, c_2 = 2$ 。

两种算法独立运行 20 次，取 20 次优化结果的平均值作为试验结果，试验结果如表 1 所示。

表 1 不同算法的优化结果

算法	M	P	h_1	q_1	r_1	h_2	q_2	r_2
SPSO	1	60	0.995	1	0.1	1	0.1	1
本文算法	1	76	1	1	1	1	0.1	0.1

使用表 1 中的优化结果作为 DMC 参数对冷热水混合系统进行控制，系统的阶跃响应曲线如图 2 所示。由图 2 可以看出使用本文算法对系统进行控制，系统的超调更小，调节时间更短，更快进入稳态。

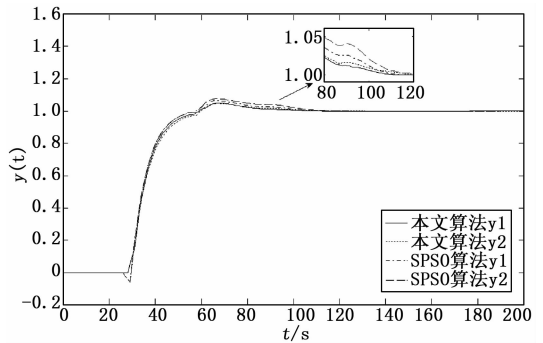


图 2 本文算法与 SPSO 响应曲线

当被控对象的增益、时间常数同时增大 20% 时，分别使用本文算法和参考文献 [2] 对系统进行控制，控制器参数不变，系统阶跃响应曲线如图 3 所示。由局部图可见，当对象参数摄动时，使用本文算法对系统进行控制时，系统响应曲线波动小于 SPSO 算法，且系统更快恢复稳态，因此本文算法系统的鲁棒性优于参考文献 [2] 中的算法。

为了验证算法的抗干扰性能，在 200 s 时，在系统的输入端引入值为 0.2，时间长度为 50 s 的干扰信号，系统响应曲线如图 4 所示。有图 4 中局部曲线可见，当系统中有干扰时，使用本文算法得到的控制器参数对系统进行控制，系统可以更快的恢复稳定。本文算法的抗干扰性能优于参考文献 [2] 中的 SPSO 算法。

4 结论

本文针对多变量 DMC 参数整定复杂、手动调节费时、标准粒子群算法容易陷入局部最优解等缺点，提出使用改进 PSO 算法优化多变量动态矩阵控制参数。该改进 PSO 算法将

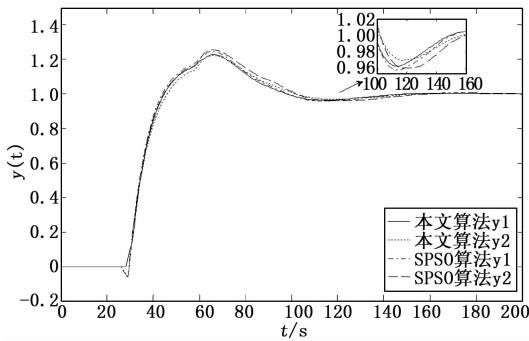


图 3 对象参数增大时本文算法与 SPSO 算法响应曲线

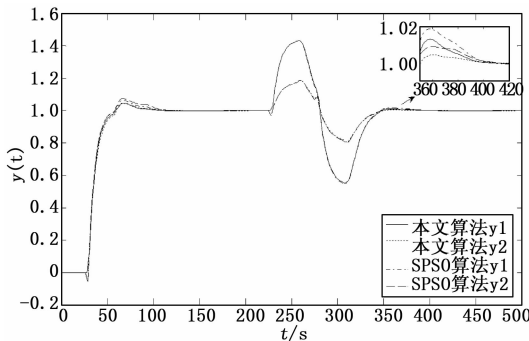


图 4 有干扰时本文算法与 SPSO 响应曲线

生物的兼性寄生行为机制引入到 PSO 算法中，将粒子分为两个种群即宿主群和寄主群，两个种群每隔一定的迭代次数根据适应度的大小，按一定的比率交换粒子，最终将宿主群中适应度差的粒子淘汰掉，并使用重新初始化的粒子代替淘汰掉的粒子。该改进 PSO 算法通过将兼性寄生行为机制引入到 PSO 中，有助于避免个体陷入局部最优，可提高 PSO 算法的收敛速度和搜索精度。本文使用该改进 PSO 算法优化 DMC 的参数，克服了 DMC 参数多、整定复杂的问题，通过本文方法得到的 DMC 参数更优，DMC 的控制效果更好。为了验证本文

算法的有效性，将本文算法用于具有大时滞、强耦合的冷热水混合系统的控制，仿真结果表明使用本文算法对冷热水系统进行控制时，系统具有超调量小、响应速度快抗干扰性强、鲁棒性好等优点。

参考文献：

[1] 唐贤伦, 李洋, 李鹏, 等. 多群体粒子群优化的 SVR 模型预测控制 [J]. 控制决策, 2014, 29 (4): 593-597.

[2] 谷泽坤, 何同祥. 基于粒子群算法的动态矩阵参数优化 [J]. 仪器仪表用户, 2015, 22 (3): 35-36.

[3] Gesner A. et al. A PSO - based optimal tuning strategy for constrained multivariable predictive controllers with model uncertainty [J]. ISA transactions, 2014, 53: 560-567.

[4] Susuki R, Kawai F, Nakazawa C, et al. Parameter optimization of model predictive control by PSO [J]. Electrical Engineering in Japan, 2012, 178 (1): 40-49.

[5] 黄昌远. 基于粒子群优化算法的反应釜温度预测控制研究与应用 [D]. 杭州: 浙江理工大学, 2014.

[6] 仇宝玉, 苏成利. 基于遗传算法优化因子的灰色 PI 预测控制 [J]. 科学技术与工程, 2011, 11 (22): 5312-5316.

[7] 秦全德. 粒子群算法研究及应用 [D]. 广州: 华南理工大学, 2011.

[8] 秦全德. 基于生物寄生行为的双种群粒子群算法 [J]. 控制与决策, 2011, 26 (4): 548-552.

[9] 姜苏英. 基于模型预测的多变量过程控制方法研究 [D]. 西安: 陕西科技大学, 2016.

[10] 刘金琨. 先进 PID 控制 MATLAB 仿真 (第 2 版) [M]. 北京: 电子工业出版社, 2004.

[11] 姜苏英, 郑恩让. 抄纸过程水分定量系统的预测控制 [J]. 计算机仿真, 2015, 32 (10): 350-353.

[12] 杨浩, 郑恩让. 液位温度时滞耦合系统自抗扰控制仿真研究 [J]. 计算机仿真, 2013, 30 (1): 356-359.

[13] 张玲. 液位温度耦合系统的智能控制 [J]. 陕西科技大学学报, 2007, 25 (6): 94-98.

(上接第 80 页)

[2] 康家玉, 王旭, 王素娥, 等. 一种基于 H6 桥单相光伏并网逆变器控制策略的研究 [J]. 可再生能源, 2016, 34 (8): 1144-1150.

[3] 于子捷, 魏晨曦, 田芳芳, 等. 一种改进型扰动观察法在最大功率跟踪中的应用 [J]. 电测与仪表, 2017, 54 (15): 113-119.

[4] 王要强, 张美玲, 齐歌. 光伏发电系统改进型扰动观察最大功率跟踪研究 [J]. 电力电容器与无功补偿, 2016, 37 (4): 111-116.

[5] 薛阳, 汪莎. 基于扰动观察法的模糊控制应用于光伏发电最大功率跟踪 [J]. 太阳能学报, 2014, 35 (9): 1622-1626.

[6] 王军成, 杨旭红, 王严龙. 改进自适应变步长光伏系统最大功率跟踪 [J]. 现代电力, 2014, 31 (5): 70-73.

[7] 何人望, 邱万英, 吴迅, 等. 基于 PSIM 的新型扰动观察法的 MPPT 仿真研究 [J]. 电力系统保护与控制, 2012, 40 (7): 56-59.

[8] 荣德生, 刘凤. 改进型扰动观察法在光伏 MPPT 中的研究 [J]. 电力系统及其自动化学报, 2017, 29 (3): 104-109.

[9] 杨奕, 王建山, 杨元培, 等. 基于外推追赶迭代法的 MPPT 算法仿真 [J]. 高电压技术, 2017, 43 (7): 2355-2361.

[10] 马昊, 张庆超, MAHao, 等. 基于粒子群优化算法和变步长扰动观察法的局部阴影情况下 MPPT 控制 [J]. 电源学报, 2016, 14 (3): 94-101.

[11] 黄勤, 石国飞, 凌睿, 等. 基于滑模控制的光伏系统 MPPT 控制方案 [J]. 计算机工程, 2012, 38 (6): 253-255.

[12] 许慧一. 基于 Fibonacci 变步长搜索的光伏发电最大功率跟踪算法 [J]. 广东电力, 2015 (12): 25-29.

[13] 刘翼, 荆龙, 童亦斌. 基于 Simulink 的光伏电池组件建模和 MPPT 仿真研究 [J]. 科技导报, 2010, 28 (18): 94-97.

[14] 江渝, 姜琦, 黄敏, 等. 单相单级光伏 LCL 并网逆变器系统控制策略 [J]. 电网技术, 2015, 39 (2): 312-319.