浙江理工大学学报(自然科学版),第 31 卷,第 3 期,2014 年 5 月
Journal of Zhejiang Sci-Tech University (Natural Sciences)
Vol. 31, No. 3, May 2014

文章编号: 1673-3851 (2014) 03-0301-05

基于改进粒子群算法的约束隐式广义预测控制

吴密密[1],**戴文战**[2]
(1. 浙江理工大学机械与自动控制学院,杭州 310018;2. 浙江工商大学信息与电子工程学院,杭州 310018)

摘　要: 针对大多数工业系统的控制输入输出都存在约束的情况,提出了一种基于改进粒子群算法的广义预测控制算法。广义预测控制算法采用隐式算法,用最小二乘法直接辨识控制增量表达式中的参数,避免求解丢番图方程;为避免粒子群算法陷入早熟,提高精度,在广义预测控制算法的滚动优化环节采用速度变异的粒子群优化算法,克服了受约束优化问题处理的缺陷,更快更准确地寻到最优目标函数值。仿真结果表明了该方法的有效性和良好的控制性能。

关键词: 隐式广义预测控制;粒子群算法;滚动优化
中图分类号: TP273.2　　**文献标志码:** A

0　引　言

工业过程的复杂化迫使人们去寻找对模型要求低、控制效果好、在线实现较为方便的控制算法。1984 年,Clarke 提出广义预测控制(GPC)[1],它具有鲁棒性好、模型要求低、在线实现方便等特点,引起了国内外控制理论界的广泛重视。

传统的广义预测控制假设对象的输入输出是线性无约束的,采用梯度寻优法获得控制增量输入。然而,实际工业中的输入、输出和控制变化率往往是有约束的,单纯地使用传统的广义预测控制不能满足实际要求。针对该问题,学者们做了大量研究,提出了多种解决方法。Demircioğlu[2] 将约束优化问题转化为二次规划问题,但是要求目标函数约束可微。金元郁[3]引入了柔化输入的概念,并且对性能指标进行了简化,解决了控制过程输入受约束的情况。童朝南等[4]采用遗传算法(GA)来解决带约束的优化问题,但是遗传算法需要进行二进制编码,不易于程序编写及仿真。一些研究采用了隐式广义预测算法,李国勇[5]引入隐式广义预测算法,避免了求解丢番图方程,减少计算量,刘福才等[6]不但采用隐式广义预测控制算法,同时还引入了输入输出柔化系数解决计算量和约束问题。还有一些研究引进了粒子群算法(PSO)[7-9],Wang 等[7]和蒋朝辉等[8]使用粒子群算法解决存在约束问题的 GPC 的滚动优化问题,肖本贤等[9]则结合了广义预测控制与 PSO 优化算法解决约束问题,该类算法可同时应用于无约束和有约束的控制过程,然而缺点是采用显式算法,计算量较大,且粒子群算法容易陷入局部最优。针对上述文献所提出的基于粒子群算法的广义预测控制算法的缺陷,本文研究将速度变异的粒子群算法和隐式广义预测控制算法相结合,提高优化精度和响应速度。

1　隐式广义预测控制基本算法

广义预测控制采用受控自回归积分滑动平均(CARIMA)模型,可用下式表示:

$$A(z^{-1})y(k)=B(z^{-1})u(k-1)+C(z^{-1})\xi(k)/\Delta \quad (1)$$

式中 $A(z^{-1})$、$B(z^{-1})$、$C(z^{-1})$分别为 n、m 和 n 阶的 z^{-1} 的多项式,$\Delta=1-z^{-1}$(后移算子),$y(k)$、$u(k)$、$\xi(k)$为实际对象的输出、输入和均值为 0 的白噪声干扰。为计算方便,令 $C(z^{-1})=1$。

GPC 的实质问题是求取最优控制增量序列 ΔU,使目标函数值最小,达到输出跟踪系统设定值

收稿日期:2013－11－06
基金项目:国家自然科学基金(N61374022);国家高新技术研究发展项目(2009AA04Z139)
作者简介:吴密密(1989－),女,浙江温州人,硕士研究生,主要从事智能检测与控制等方面的研究。

的目标。目标函数可用下式表示：

$$J=\sum_{j=1}^{n}[(y(k+j)-w(k+j)]^2+\sum_{j=1}^{m}\lambda(j)[\Delta u(k+j-1)]^2 \tag{2}$$

式中 n 和 m 分别为预测时域和控制时域，为控制加权系数，通常为常数，取值在(0,1)之间。参考轨迹为：

$$w(k+j)=a^j y(k)+(1-a^j)y_r \tag{3}$$

式中 y_r 和 $y(k)$ 分别为设定值和系统当前输出。a 为柔化系数，取值在(0,1)之间。引入丢番图方程，推出最优控制率为：

$$\Delta U=(G^{\mathrm{T}}G+\lambda I)^{-1}G^{\mathrm{T}}(W-f) \tag{4}$$

式中：

$$\begin{aligned}&\Delta U=[\Delta u(k)\quad \Delta u(k+1)\quad \cdots\quad \Delta u(k+m-1)]^{\mathrm{T}}\\&W=[w(k+1)\quad w(k+2)\quad \cdots\quad w(k+n)\quad]^{\mathrm{T}}\\&f=[f(k+1)\quad f(k+2)\quad \cdots\quad f(k+n)\quad]^{\mathrm{T}}\end{aligned} \tag{5}$$

$$G=\begin{bmatrix} g_0 & \cdots & \cdots & \cdots & 0\\ g_1 & g_0 & & & \\ \vdots & & \ddots & & \\ \vdots & & & \ddots & \\ g_{n-1} & g_{n-2} & \cdots & \cdots & g_0\end{bmatrix}_{n\times m} \tag{6}$$

其中 g_0、g_1、…、g_{n-1} 为系统单位阶跃响应的前 n 项。

下一刻控制量输入为：

$$u(k)=u(k-1)+g^{\mathrm{T}}(W-f) \tag{7}$$

式中 g^{T} 为 $(G^{\mathrm{T}}G+\lambda I)^{-1}G^{\mathrm{T}}$ 的第一行。

由公式(4)可知，在 λ 和 W 已知的情况下，只需要计算 G 和 f，即可求得 ΔU。根据参考文献[5]可知，隐式广义预测控制利用系统的输入输出数据，根据预测方程，可直接得出 f，并使用最小二乘法估计 G 中的元素。

1.1 隐式广义预测控制矩阵 G 的求取

广义预测控制的预测方程为：

$$\hat{\mathbf{y}}=G\Delta U+f \tag{8}$$

其中 $\hat{\mathbf{y}}=[\hat{y}(k+1|k)\cdots\hat{y}(k+N|k)]^{\mathrm{T}}$。

由公式(8)得：

$$y(k+n)=g_{n-1}\Delta u(k)+\cdots+g_0\Delta u(k+n-1)+f(k+n)$$

令：

$$\begin{cases}X(k)=[\Delta u(k),\Delta u(k+1),\cdots,\Delta u(k+n-1),1]\\ \theta(k)=[g_{n-1},g_{n-2},\cdots,g_0,f(k+n)]^{\mathrm{T}}\end{cases}$$

$\theta(k)$ 用下式递推最小二乘法估计：

$$\begin{cases}\hat{\theta}(k)=\hat{\theta}(k-1)K(k)[\Delta y(k)-\hat{X}(k-1)\hat{\theta}(k-1)]\\ K(k)=P(k-1)\hat{X}^{\mathrm{T}}(k-n)[\hat{X}(k-1)P(k-1)\hat{X}^{\mathrm{T}}\\ \qquad (k-1)+\lambda_1]^{-1}\\ P(k)=\dfrac{1}{\lambda_1}[I-K(k)\hat{X}(k-1)]P(k-1)\end{cases} \tag{9}$$

式中，λ_1 为遗忘因子，取值范围在[0,1]之间。由式(9)可以估计出 $\theta(k)$，即可求得 G 中的元素。

1.2 隐式广义预测控制矩阵 f 的求取

$$f=\begin{bmatrix}f(k+1)\\ f(k+2)\\ \vdots\\ f(k+n)\end{bmatrix}=\begin{bmatrix}\hat{y}(k+2|k)\\ \hat{y}(k+3|k)\\ \vdots\\ \hat{y}(k+n+1|k)\end{bmatrix}+\begin{bmatrix}1\\1\\ \vdots\\1\end{bmatrix}\times e(k+1) \tag{10}$$

式中，$e(k+1)=y(k+1)-y(k+1|k)$ 为预测误差。

以上是控制对象无约束情况下的隐式广义预测控制。实际上，广义预测控制中存在各种各样的约束限制，文献[10]列出了广义预测控制中对象存在的九种约束限制，本文采用的约束条件为：

$$\Gamma\Delta u_{\min}\leqslant\Delta U\leqslant\Gamma\Delta u_{\max} \tag{11}$$

其中 $\Delta u_{\min}$ 和 $\Delta u_{\max}$ 为控制增量的下限与上限，$\Gamma=\begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix}_{m\times 1}$。

2 粒子群基本算法

粒子群基本算法是由美国的 Kennedy 和 Eberhart 两位博士在 1995 年共同提出来的一种基于群体智能的演化计算技术。

PSO 描述了一个群体中有 m 个粒子，在 D 维空间中以一定的速度飞行即搜索，每个粒子在搜索的时候，都在自己曾到过的最优位置和群体中所有个体历史最优的最优位置的基础上搜索。第 i 个粒子的当前位置为 $x_i=(x_{i1},x_{i2},\cdots,x_{iD})$，速度为 $v_i=(v_{i1},v_{i2}\cdots v_{iD})$，曾到过的最优位置为 $pbesti=(p_{i1},p_{i2}\cdots p_{iD})$，群体中所有粒子所经过的最优位置为 $gbest=(p_{g1},p_{g2},\cdots p_{gD})$。粒子位置和速度更新公式为：

$$v_{id}^{k+1}=wv_{id}^{k1}+c_1\xi(p_{id}^{k}-x_{id}^{k})+c_2(p_{gd}^{k}-x_{id}^{k}) \tag{12}$$

$$x_{id}^{k+1}=x_{id}^{k}+v_{id}^{k+1} \tag{13}$$

其中，c_1、c_2 称为学习因子，通常取 $c_1=c_2=2$。w 为惯性权重，决定继承当前速度的比例，ξ、η 为[0,1]区间内均匀分布的伪随机数，粒子的最大速度为 $v_{\max}$。

每个粒子的个体历史最佳位置更新方程为[8]：

$$pbesti(k+1)=\begin{cases}pbesti(k),J_1(x_i(k+1))\geqslant \\ J_1(pbesti(k)) \\ x_i(k+1),J_1(x_i(k+1))< \\ J_1(pbesti(k))\end{cases} \tag{14}$$

其中 J_1 为粒子群的适应度函数。

群体中所有粒子所经过的最优位置更新方程为：

$$gbest=\arg\min_{pbesti} J_1(pbesti(k+1)) \tag{15}$$

3　基于改进粒子群算法的广义预测控制

求解不等式约束的优化问题有两类粒子群算法，一种是利用罚函数先把约束优化问题转为无约束优化问题，然后用标准的粒子群算法进行优化。另一种是直接在适应度函数中体现不等式约束。单纯使用粒子群算法对高维问题进行优化会发生早熟而导致收敛性能差和寻优速度相对较慢[9]的问题。针对这些问题，本文将对粒子群算法进行改进。

3.1　粒子群算法的适应度函数

公式(2)综合考虑了系统输出值与给定值的偏差以及控制增量的变化情况。采用公式(2)作为粒子群算法的适应度函数，目的是使得系统优化时系统的输出与给定值尽量接近，控制增量变化不大。

3.2　粒子群算法的初始化

由文献[9]可知，PSO 中粒子的初始位置是随机产生的，所以初始种群中个体位置分布过于分散，从而导致了种群中大多数粒子的初始位置与最优位置之间相距很大，算法收敛速度较慢，无法实现当今工业化所需的高效性。因此可以选择利用系统的信息选取质量优异的初始种群，加快算法的迭代速度。本文在粒子群的初始化过程中，利用在无约束条件下对公式(2)采用梯度寻优获得最优序列 ΔU，代入约束条件(11)式中，将超出约束条件的序列元素按约束边界值赋值。然后将 ΔU 作为初值赋值给粒子群中随机选取出来的 $\alpha\%$ 的粒子，剩余的粒子在约束范围内随机赋予初始值，粒子位置的维数为控制时域 m。这样既保证了粒子的多样性，又保证了粒子的优质性，从而加快了粒子群算法的收敛速度，减少了计算搜索的时间。

$\alpha\%$ 的种群位置初始化过程为：

$$x_i^{\mathrm{T}}=\begin{cases}\Gamma\Delta u_{\min},\Delta U\Gamma\Delta u_{\min} \\ \Delta U,\Gamma\Delta u_{\min}\leqslant\Delta U\leqslant\Gamma\Delta u_{\max} \\ \Gamma\Delta u_{\max},\Delta U>\Gamma\Delta u_{\max}\end{cases} \tag{16}$$

种群中粒子的速度在其规定范围内随机初始化。

3.3　粒子群算法的速度变异策略

针对粒子群算法的早熟问题，本文提出一个变异策略：假设 m 个粒子，速度向量为 n 维。粒子的速度为 $v_i=(v_{i1},v_{i2}\cdots v_{in})$。每次迭代对每个粒子的 n 维速度向量的各个分量的绝对值进行比较，求 $v_{it}=\min\{|v_{i1}|,|v_{i2}|,\cdots,|v_{in}|\}$，其中最小的速度以一定的概率 ρ 进行变异[11]，增强了每个粒子的搜索能力。变异概率 ρ 的取值会影响算法的寻优能力，而它的取值主要取决于算法的收敛程度。经过实验调试取 $\rho=0.5$。该变异算法的速度更新公式为：

$$v_{id}^{k+1}=wv_{id}^{k}+c_1\xi(p_{id}^{k}-x_{id}^{k})+c_2\eta(p_{gd}^{k}-x_{id}^{k}) \tag{17}$$

$$v_{it}^{k+1}=0.5*rand-1 \tag{18}$$

其中 $d=1,2,\cdots n,t$ 为速度分量最小的维度。

改进的粒子群算法具体操作：

Step 1：对种群中粒子的速度与位置进行初始化；

Step 2：计算每个粒子的适应值，根据公式(14)、(15)更新 $pbesti$ 和 $gbest$。

Step 3：根据公式(12)、(13)更新粒子群的速度和位置。

Step 4：寻找每个粒子中速度绝对值最小速度分量。

Step 5：产生一个随机数 $rand$，服从(0－1)分布。比较随机数与变异概率 ρ 的大小，如果随机数大于变异概率，则粒子速度按公式(17)，公式(18)进行变异，否则不变化。

Step 6：终止条件判断。若未达到终止条件(迭代最大迭代次数或迭代精度)，则转至 Step 2。否则，输出 $gbest$，终止计算。

基于改进 PSO 优化的隐式 GPC 算法步骤：

Step 1：设置算法初值，构造模型公式(1)。

Step 2：利用广义预测控制最优控制率计算控制增量 ΔU，判断其值是否满足约束条件。如果满足条件，则输出最优控制增量 ΔU，取其第一个元素。否则，利用改进的 PSO 算法寻优，将控制增量作为优化变量，即粒子群优化算法中的粒子位置，获得最优控制增量 ΔU 序列，取其第一个元素。

Step 3：根据公式(7)计算系统下一时刻输入量 $u(k)$，反馈到控制系统中。

Step 4：转至第二步，反复进行滚动优化，直到控制系统停止工作。

4 仿真研究

隐式广义预测算法中引进变异的粒子群算法，由于其增加了粒子的多样性，解决普通的粒子群易陷入局部最优的缺陷，更快更准确地找到系统最优值。这从下面两个仿真实例可以看出。

a) 参考文献[12]，研究带有约束的线性离散系统，由公式(1)得系统模型：

$$y(k)-1.001\,676y(k-1)+0.241\,714y(k-2)=0.235\,69u(k-1)+\xi(k)/\Delta$$

隐式广义预测控制参数：$n=6$，$m=2$，$\lambda(j)=0.5$，$a=0.35$，$\xi(k)$为$[-0.2,0.2]$均匀分布的白噪声，约束$\Delta u(k)$在$[-1,1]$之间[12]。改进的PSO参数：粒子群数为20，迭代次数为100，惯性权重为0.9，学习因子均为2。经过MATLAB仿真，三种方法的跟踪曲线结果如下图1所示。

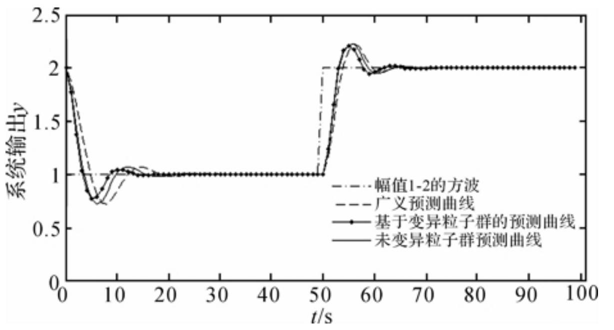

图1 三种方法系统输出的仿真结果

比较图1中三条曲线，得出在输入控制增量受限的情况下，采用基于速度变异粒子群算法的隐式广义预测控制算法具有更强的适应能力和更好的跟踪性能。在图1中可以看出，一般的广义预测控制算法曲线第一次到达最大峰值点为(8，−0.718 8)，稳定点为(25，1)；基于未变异的粒子群算法的隐式广义预测曲线第一次到达最大峰值点为(6，−0.718 7)，稳定点为(22，1)；基于变异粒子群算法的隐式广义预测曲线第一次到达最大峰值点为(5，−0.769 2)，稳定点为(18，1)。由此可见基于变异粒子群算法的隐式广义预测控制算法更快达到稳定值，输出变化平缓，更加符合实际要求。

b) 研究车用发动机的模型[5]，由公式(1)得系统模型：

$$y(k)-0.496\,585y(k-1)=0.5u(k-2)+\xi(k)/\Delta$$

隐式广义预测控制参数：$n=6$，$m=2$，$\lambda(j)=0.8$，$\alpha=0.3$，$\xi(k)$为$[-0.2,0.2]$均匀分布的白噪声。约束$\Delta u(k)$在$[-1,1]$之间。

改进的PSO参数：粒子群数为20，迭代次数为100，惯性权重为0.8，学习因子均为2。经过MATLAB仿真结果如下图2、图3所示。

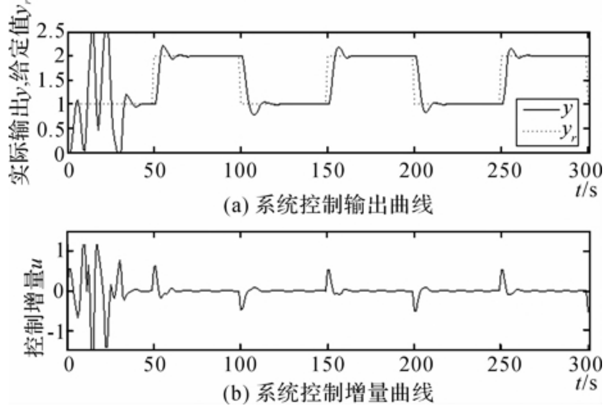

图2 隐式广义预测控制仿真结果

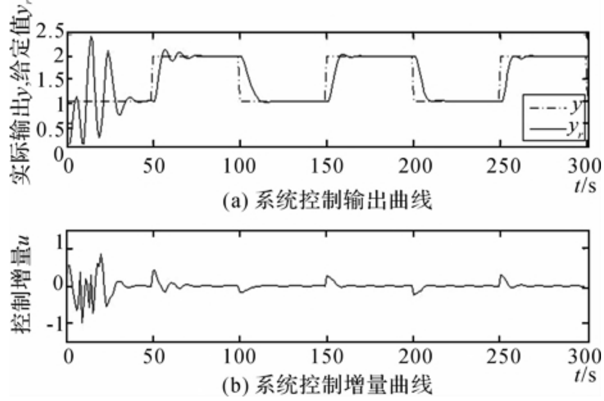

图3 变异粒子群的隐式广义预测控制仿真结果

比较图2与图3可知，采用基于速度变异粒子群算法的隐式广义预测控制，其控制输出的超调更小，响应速度更快，稳定性更高，控制增量的变化浮动值更小。

5 结 语

本文针对受约束的工业过程控制问题，提出了混合改进的粒子群与隐式广义预测控制的算法，用于解决带有约束条件的广义预测控制问题。通过仿真结果的比较，验证了基于改进的粒子群算法的隐式广义预测控制的可行性和有效性。该方法具有适用范围广，响应速度快，稳定性高等特点，对指导实际工业生产有较高的实用价值。

由于仿真系统的受限，在设计GPC算法中的各个参数如控制加权系数、柔化系数、预测长度等，其具体的取值需要通过不断的试验才能找到最佳值，目前还没有完整的理论体系，有待进一步研究。

参考文献：

[1] Clarke D W, Mohtadi C, Tuffs P S. Generalized predictive control: part I the basicalgorithm[J]. Automatica,

1987，23(2)：137-148.

[2] Demircioğlu H. Constrained continuous-time generalised predictive control[J]. IEE Proceedings-Control Theory and Applications，1999，146(5)：470-476.

[3] 金元郁. 一种约束输入的广义预测控制新算法[J]. 控制与决策，2002，17(4)：506-508.

[4] 童朝南，肖　磊，彭开香，等. 基于遗传算法的结晶器液位约束广义预测控制[J]. 控制与决策，2009，24(11)：1735-1739.

[5] 李国勇. 输入受限的隐式广义预测控制算法的仿真研究[J]. 系统仿真学报，2004，16(7)：1533-1535.

[6] 刘福才，贺浩博. 一种约束输入输出的隐式广义预测控制新算法[J]. 计算机仿真，2007，24(6)：301-303.

[7] Wang Z，Sun Y. Generalized predictive control based on particle swarm optimization for linear/nonlinear process with constraints [C]//Computational Intelligence and Natural Computing Proceedings (CINC)：2010 2nd International Conference on. IEEE，2010：303-306.

[8] 蒋朝辉，李学明，桂卫华. 大时滞系统全参数自适应预测控制策略[J]. 中南大学学报，2012，43(1)：195-201.

[9] 肖本贤，朱志国，刘一福. 基于粒子群算法混合优化的广义预测控制器研究[J]. 系统仿真学报，2007，19(4)：821-824.

[10] Guzman J L，Berenguel M，Dormido S. Interactive teaching of constrained generalized predictive control [J]. Control Systems，IEEE，2005，25(2)：52-66.

[11] 付国江，王少梅，刘淑燕，等. 改进的速度变异粒子群算法[J]. 计算机工程与应用，2006，13(3)：48-50.

[12] 李国勇. 智能预测控制及其 MATLAB 实现[M]. 北京：电子工业出版社，2005：258-261，309-324.

A Kind of Constrained Implicit Generalized Predictive Control Based on Improved Particle Swarm Optimization

WU Mi-mi[1]，*DAI Wen-zhan*[2]

(1. School of Mechanical Engineering & Automation，Zhejiang Sci-Tech University，Hangzhou 310018，China；2. School of Information and Electronic Engineering，Zhejiang Gongshang University，Hangzhou 310018，China)

Abstract：With respect to constrained control input and output of most industrial systems，this paper proposes a kind of generalized predictive control based on improved particle swarm optimization. Generalized predictive control is an implicit algorithm to identify directly parameters in control increment expression by least square method，to avoid solving Diophantine equations；to avoid particle swarm optimization getting into local convergence and improve precision，speed variant particle swarm optimization is used in rolling optimization of generalized predictive control to overcome defect in processing constrained optimization problem and seek optimal objective function value more quickly and precisely. Simulation results show effectiveness and good control performance of this method.

Key words：implicit generalized predictive control；particle swarm optimization；rolling optimization

(责任编辑：康　锋)