

Smith 预估控制器算式的改进

王永初

(精密机械工程系)

摘 要

Smith 预估器是一种重要的控制系统结构, 可以应用于纯滞后补偿控制. 本文介绍一种精确、可用的 Smith 预估控制器算式. 这种算式由 PID 模块与 Smith 预估模块组成. PID 模块算式与 $G(s)$ 模块算式由虚拟零阶保持器 z 变换来拟合, $G(s)e^{-\tau}$ 模块算式由虚拟一阶保持器 z 变换来拟合, $e^{-\tau}$ 单元由具有双指针的循环器构成. 这种算式具有很高的拟合精度.

一、问题的提出

Smith 预估控制器在现代工业生产过程控制中有许多重要的应用, 如合成氨厂的氢/氮比控制, 脱硫厂的尾气成分控制, 火电厂的表面式过热器的温度控制. 1980 年作者曾主持 DDC 控制算式的研究, 整个 DDC 序列算式有 60 多个, 并出版了专辑^[1], 由于当时条件对 Smith 预估控制器的算式研究尚比较肤浅, 使用中发现存在拟合精度较低, 闭环控制引起非正常动作等缺点.

(1) 积分与微分功能削弱

由于具有大纯滞后的控制对象, 在调节过程容易引起超调, 并且从提高数字控制系统的精度看, 采样周期要求很短, 从系统的稳定性看, 积分器的积分时间必须较大, 以致预估控制器中的 PID 调节器失去积分功能. 例如在采样周期 $T_s = 1s$, 积分时间 $T_i = 10min$ 时, 在调节误差 $e(K) = 1\%$ 的情况下, 积分部分的运算输出为

$$m_i(K) = \frac{T_s}{T_i} e(K) = \frac{1}{60000}$$

即使是 16 位定点运算的机器 (或 8 位双倍字长定点运算), 均表示不出这个数, 机器就将这个数当作零处理.

控制中的微分项的作用在 1 至 2 个采样周期内消失. 如果要加强微分作用, 则在计算的第一个周期内很可能引起溢出, 造成在线运行机器的停机, 这在生产上被当作一个事故.

由于解决上述问题的一个根本办法是将常规的 PID 运算

$$PID(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (1)$$

修改为如下形式

$$PID(s) = K_P \left(1 + \frac{1}{T_i s} + \frac{T_D s + 1}{r s + 1} \right) \quad (2)$$

并在PID模块的积分运算中,采用特殊的处理方法。

(2) e^{-rs} 单元的计算占据大量的工作时间,使得一台微型机的大部分工作花费在 e^{-rs} 单元的运算上,所以 e^{-rs} 单元的实现方式必须改进。

(3) 由于目前纯数字设计Smith预估控制器的方法还不成熟,需要先经模拟系统的设计,然后转换成数字的方式来实现,这个特定问题存在着数字模式与模拟模式提高拟合度的问题。

二、Smith 预估控制器的构成

Smith 预估控制器的构成如图 1 所示,包括PID模式、 $G(s)$ 模式与 $G(s)e^{-rs}$ 模式。PID模式要求不产生积分整化误差,而且运算过程的量化误差应该为最小。 $G(s)$ 模式与 $G(s)e^{-rs}$ 应该具有较高的拟合度。根据图 1 的结构得到计算程序框图如图 2 所示。

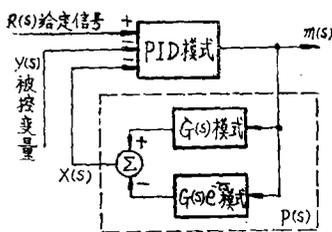


图 1

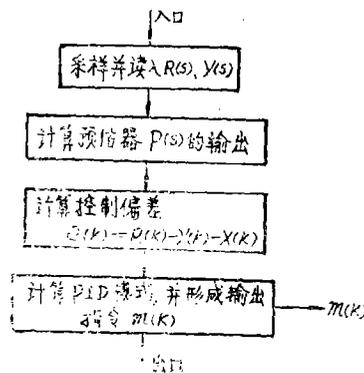


图 2

下面分析各个模块的具体实现问题:

1. PID 模块

(1) 结构选择:

对于慢速响应过程,虚拟零阶保持器 z 变换对模拟模式具有较高的拟合度。因此将式(2)转换成拟合 Z 变换式

$$D(Z) = z \left\{ \frac{1 - e^{-T_D s}}{s} K_P \left(1 + \frac{1}{T_i s} + \frac{T_D s + 1}{r s + 1} \right) \right\}$$

$$= K_P + \frac{\alpha Z^{-1}}{1 - Z^{-1}} + \frac{\beta_1 - \beta_2 Z^{-1}}{1 - \beta_3 Z^{-1}} \quad (3)$$

其中

$$\alpha = \frac{K_P T_s}{T_i}$$

$$\beta_1 = K_P r$$

$$\beta_2 = K_p \left(r - 1 + e^{-\frac{r}{T_D} T_s} \right)$$

$$\beta_3 = e^{-\frac{r}{T_D} T_s}$$

式(3)各项可以进一步合并为

$$D(Z) = \frac{(K_p + \beta_1) - [K_p(1 + \beta_3) - \alpha + \beta_1 + \beta_2]Z^{-1} + (\beta_2 + K_p\beta_3 - \alpha\beta_3)Z^{-2}}{(1 - Z^{-1})(1 - \beta_3 Z^{-1})} \quad (4)$$

根据总体量化误差最小，我们选择式(3)的数字模式。图 3 与图 4 分别表示式(3)与式(4)误差源及其传播结构方框。

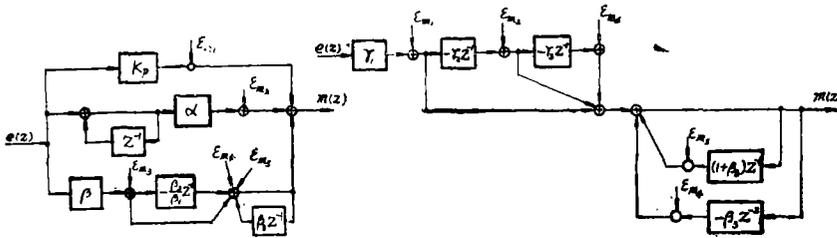


图 3

图 4

由图 3 求得 $m(Z)$ 同各系数量化误差的关系式

$$m(Z) = \varepsilon_{m_1} + \varepsilon_{m_2} + \frac{1}{1 - \beta_3 Z^{-1}} (\varepsilon_{m_4} + \varepsilon_{m_5}) + \frac{1 - \frac{\beta_2}{\beta_1} Z^{-1}}{1 - \beta_3 Z^{-1}} \varepsilon_{m_3} \quad (5)$$

由图 4 求得 $m(Z)$ 同各系数量化误差的关系式

$$m(Z) = \frac{1 - r_2 Z^{-1} + r_2 r_3 Z^{-2}}{1 - (1 + \beta_3) Z^{-1} + \beta_3 Z^{-2}} \varepsilon_{m_1} + \frac{1 - r_3 Z^{-1}}{1 - (1 + \beta_3) Z^{-1} + \beta_3 Z^{-2}} \varepsilon_{m_2} + \frac{1}{1 - (1 + \beta_3) Z^{-1} + \beta_3 Z^{-2}} (\varepsilon_{m_3} + \varepsilon_{m_4} + \varepsilon_{m_5}) \quad (6)$$

其中

$$r_1 = K_p + \beta_1$$

$$r_2 = [K_p(1 + \beta_3) - \alpha + \beta_1 + \beta_2] / r_1$$

$$r_3 = (\beta_2 + K_p\beta_3 - \alpha\beta_3) / r_1 r_2$$

若设

$$\bar{\varepsilon}_{m_1} = \bar{\varepsilon}_{m_2} = \bar{\varepsilon}_{m_3} = \bar{\varepsilon}_{m_4} = \bar{\varepsilon}_{m_5} = \bar{\varepsilon}$$

则由式(5)可以求得

$$\bar{m} = \bar{\varepsilon}_{m_1} + \bar{\varepsilon}_{m_2} + (\bar{\varepsilon}_{m_3} + \bar{\varepsilon}_{m_5}) \lim_{Z \rightarrow 1} \frac{1}{1 - \beta_3 Z^{-1}} + \bar{\varepsilon}_{m_3} \lim_{Z \rightarrow 1} \frac{1 - \frac{\beta_2}{\beta_1} Z^{-1}}{1 - \beta_3 Z^{-1}}$$

$$= \bar{\varepsilon} \left(\frac{5 - 2\beta_3 - \frac{\beta_2}{\beta_1}}{1 - \beta_3} \right) \quad (7)$$

由式(6)可以求得

$$\bar{m} = \bar{e} \frac{5 - \tau_2 - \tau_3 - \tau_2 \tau_3}{1 - (1 + \beta_3) + \beta_3} \rightarrow \infty \quad (8)$$

因此从稳态传播量化误差的角度看,图3的结构形式比图4好。由式(4)得到如下的差分方程式

$$m(K) = m_1(K) + m_2(K) + m_3(K) \quad (9)$$

其中

$$\begin{cases} m_1(K) = K_p e(K) \\ m_2(K) = m_2(K-1) + a e(K-1) \\ m_3(K) = \beta_3 m_3(K-1) + \beta_1 e(K) - \beta_2 e(K-1) \end{cases} \quad (10)$$

(2) 防止积分整化与舍位误差:

式(9)算式包括三个部分:比例、积分与微分。当 T_s 较大, T_s 较小时,积分部分不起作用,但是对于控制系统来说,积分可以消除调节余差是一个不可省略的部分,为此将积分的程序框图修改成图5。

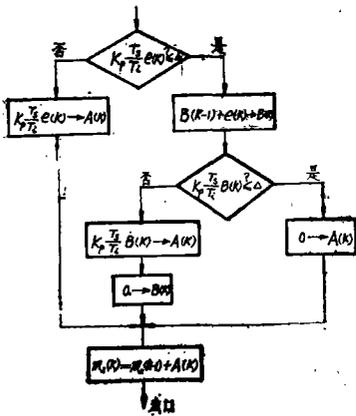


图 5

2. G(s) 模块

由于最典型的工业对象特性可以写成

$$G_0(s) = \frac{K_0 e^{-\tau s}}{(T_1 s + 1)(T_2 s + 1)} \quad (11)$$

因此 $G(s)$ 可记为

$$G(s) = \frac{K_0}{(T_1 s + 1)(T_2 s + 1)} \quad (12)$$

利用带零阶虚拟保持器的 Z 变换,可以将 $G(s)$ 变成 $G(Z)$,

比直接按差分方法(即利用 $s = \frac{1-Z^{-1}}{T_s}$ 关系代入 $G(s)$)有更高的拟合度。

$$\begin{aligned} G(z) &= z \left\{ \left(\frac{1 - e^{-T_s s}}{s} \right) \frac{K_0}{(T_1 s + 1)(T_2 s + 1)} \right\} \\ &= K_0 - \left(\frac{T_1}{T_1 - T_2} \right) \frac{K_0(1 - Z^{-1})}{1 - e^{-\frac{T_s}{T_1}} Z^{-1}} + \left(\frac{T_2}{T_1 - T_2} \right) \frac{K_0(1 - Z^{-1})}{1 - e^{-\frac{T_s}{T_2}} Z^{-1}} \end{aligned} \quad (13)$$

同式(13)相应的算式为

$$X_1(K) = X_{11}(K) - X_{12}(K) + X_{13}(K) \quad (14)$$

其中

$$\begin{cases} X_{11}(K) = K_0 m(K) \\ X_{12}(K) = e^{-\frac{T_s}{T_1}} X_{12}(K-1) + \frac{K_0 T_1}{T_1 - T_2} m(K) - \frac{K_0 T_1}{T_1 - T_2} m(K-1) \\ X_{13}(K) = e^{-\frac{T_s}{T_2}} X_{13}(K-1) + \frac{K_0 T_2}{T_1 - T_2} m(K) - \frac{K_0 T_2}{T_1 - T_2} m(K-1) \end{cases} \quad (15)$$

这里通过一个具体例子的计算来验证拟合度:设 $T_1 = 20s$, $T_2 = 10s$, $K_0 = 1$, $T_s = 1s$,得到的单位阶跃响应如图6的圈点所示的曲线。 $G(s)$ 在单位阶跃作用下的响应函数

为

$$X_1(t) = L^{-1} \left\{ \frac{1}{s} \cdot \frac{K_0}{(20s+1)(10s+1)} \right\}$$

$$= 1 - 2e^{-\frac{t}{20}} + e^{-\frac{t}{10}} \tag{16}$$

$X_1(t)$ 函数的变化曲线如图 6 的虚线所示。可见利用虚拟零阶保持器的Z变换法可以求得精确的 $G(s)$ 拟合函数。

3. $G(s)e^{-\tau s}$ 模块

$G(s)e^{-\tau s}$ 模块算式采用虚拟一阶保持器的Z变换方法，原因是 $G(s)e^{-\tau s}$ 包含有L个延迟节拍，而虚拟一阶保持器Z变换法可以使拟合曲线提前一拍拟合，这正好对 $G(s)e^{-\tau s}$ 拟合是有利的。如果此法应用于拟合 $G(s)$ 模块，则会使 $G(Z)$ 包含有超前因子，使得运算无法进行。所以 $G(s)e^{-\tau s}$ 采用虚拟一阶保持器Z变换法是合适的，即

$$\begin{aligned} & \mathcal{Z} \left\{ \frac{e^{-\frac{T_s s}{T_s}} (1 - e^{-T_s s})^2}{s} \frac{K_0 e^{-\tau s}}{(T_1 s + 1)(T_2 s + 1)} \right\} \\ &= \frac{K_0 Z(1 - Z^{-1})^2 Z^{-L}}{T_s T_1 T_2} \mathcal{Z} \left\{ \frac{1}{(s + \frac{1}{T_1})(s + \frac{1}{T_2}) s^2} \right\} \\ &= K_0 Z^{-L} - \frac{K_0(T_1 + T_2)(1 - Z^{-1})Z^{-(L-1)}}{T_s} \\ &+ \frac{K_0 T_1^2 (1 - Z^{-1})^2 Z^{-(L-1)}}{T_s(T_1 - T_2)(1 - e^{-\frac{T_s}{T_1}} Z^{-1})} - \frac{K_0 T_2^2 (1 - Z^{-1})^2 Z^{-(L-1)}}{T_s(T_1 - T_2)(1 - e^{-\frac{T_s}{T_2}} Z^{-1})} \end{aligned} \tag{17}$$

由式(17)求得 $G(s)e^{-\tau s}$ 模块的算式为

$$X_2(K) = X_{21}(K) + X_{22}(K) - X_{23}(K) \tag{18}$$

其中

$$\begin{aligned} X_{21}(K) &= \frac{K_0(T_s + T_1 + T_2)}{T_s} m(K-L) - \frac{K_0(T_1 + T_2)}{T_s} m(K-L+1) \\ X_{22}(K) &= e^{-\frac{T_s}{T_1}} X_{22}(K-1) + \frac{K_0 T_1^2}{T_s(T_1 - T_2)} m(K-L+1) \\ &\quad - \frac{2K_0 T_1^2}{T_s(T_1 - T_2)} m(K-L) + \frac{K_0 T_1^2}{T_s(T_1 - T_2)} m(K-L-1) \\ X_{23}(K) &= e^{-\frac{T_s}{T_2}} X_{23}(K-1) + \frac{K_0 T_2^2}{T_s(T_1 - T_2)} m(K-L+1) \\ &\quad - \frac{2K_0 T_2^2}{T_s(T_1 - T_2)} m(K-L) + \frac{K_0 T_2^2}{T_s(T_1 - T_2)} m(K-L-1) \end{aligned} \tag{19}$$

$$L = \frac{\tau}{T_s}$$

为了验证拟合结果，同样以 $T_1 = 20, T_2 = 10, \tau = 10s, T_s = 1s, K_0 = 1$ 为例，得到如图 7 中

的圆点所示结果，正确的响应曲线如图中虚线所示，可见拟合结果比较理想。

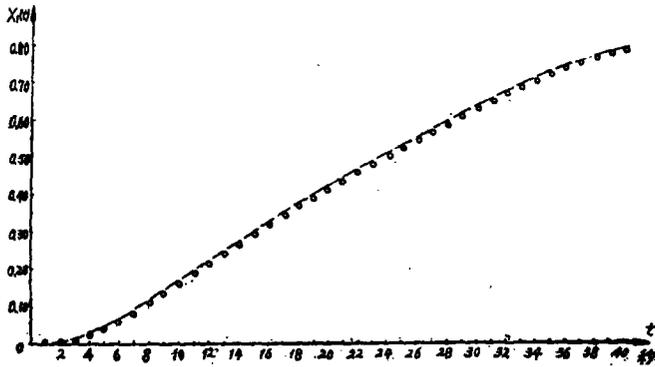


图 6

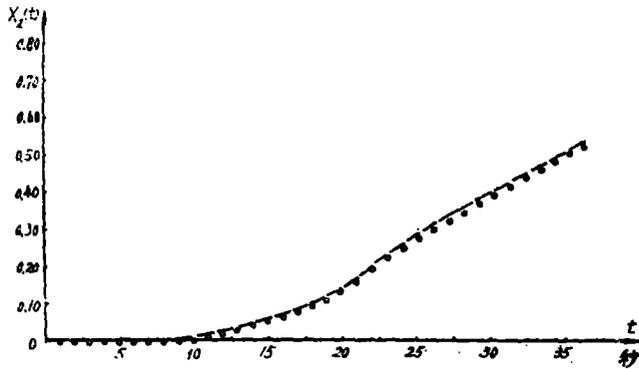


图 7

三、延迟 n 拍信息的实现

在计算式(15)时，要求存储的信息只有 $X_{12}(K-1)$ ， $X_{13}(K-1)$ 及 $m(K-1)$ ，即只需延迟一个节拍的信息。但是在计算式(19)时，要求有 $(L+1)$ 个延迟节拍的信息，而且制造成一个通用的模块， L 是可变的，有时 L 很长，有时很短，取得延迟 n 拍信息的做法，一般采用图 8 所示方法。

每采样读入一次数据， $0, 1, \dots, (n-1)$ 单元的内容，均沿着数据流的方向移到下一个单元。从 $(n-i)$ 号单元的信号是被延迟了 i 个采样周期。这种方法的缺点是浪费了大量的机器工作时间。因为每一个单元的内容移入下一个单元，都要通过CPU， L 越长，实现 e^{-s} 功能就愈占据机器的工作时间。我们提出一个节省机器工作时间的办法是由指针改变数据入口与出口的方法，在机器的内存中开辟一个专门的内存区域，并依次编号 $1, 2, \dots, n$ ，每采样读入一次数据，入口地

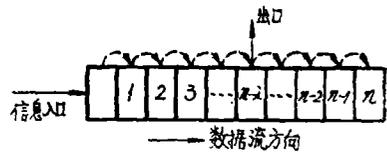


图 8

址指针 Z_i 自动加 1. 当 $Z_i > n$ 时, 自动执行 $(Z_i - n) \rightarrow Z_i$ 的运算, 这样入口地址指针象 时 针 一样从 $1 \rightarrow n \rightarrow 1 \rightarrow n$ 周而复始地循环移动. 出口地址指针 P_i 为

$$P_i = \begin{cases} Z_i - L, & \text{当 } Z_i > L \text{ 时} \\ n + Z_i - L, & \text{当 } Z_i < L \text{ 时,} \end{cases}$$

$$L = \text{integer} \left\{ \frac{\tau}{T_s} \right\}$$

采用双指针的程序结构设计, 可以方便地实现信息的延迟.

参 考 文 献

[1] 王永初、俞济成、程与仁、方国鹏、焦如兰、余德美, DDC控制算式研究报告(专辑), 工业自动化仪表, 4(1980).

[2] P.卡茨著, 王永初、邓志林、周立峰译, 微型机在数字控制中的应用, 四川省自动化与仪器仪表学会, (1985).

The Retrofitting of Algorithm for Smith Predicted Controller

Wang Yongchu

Abstract

Smith predictor is an important control system which can be used in dead time compensated control. An accurate algorithm suitable for Smith predicted controller is presented in this paper. The algorithm is composed of PID model and Smith predicted model. The PID model algorithm and $G(s)$ model algorithm are fitted by the virtual zero order holder Z transform, and the $G(s)e^{-\tau}$ model algorithm is fitted by the virtual first order holder Z transform. The $e^{-\tau}$ unit is made up of a circulator with dual pointer. This algorithm possesses a highly fitting accuracy.