



由图 1 可见,所增加的三个环节可理解为在模糊控制器内部引进一个“软件反馈”,即由软件实现对控制器自身控制性能的反馈,通过这个反馈不断地调整和改善控制器的性能,以使控制系统的控制效果达到最佳状态

控制器的性能如何得到呢?通常是将直接测量系统的输出特性与预定的特性进行比较,看其偏差多大,然后计算出输出特性应该进行多大的校正,用  $p(nT)$  表示校正量,其中  $T$  是采样周期。

在一般的模糊控制器中,通常用偏差和偏差的变化值来衡量输出特性与给定特性之间的偏离情况,然后根据偏差和偏差变化的采样值  $e(nT)$  和  $\dot{e}(nT)$  来计算出对输出特性所需的校正量  $p(nT)$ ,即根据输出值在设定点下还是在设定点上,以及输出值是呈减小趋势还是呈增加趋势来确定校正量的大小和符号。根据性能测量规则可归纳出模糊校正表(表 1)。

表 1 校正表

Table 1 Tuning table

$e$	不同 $e_c$ 的 $p$ 值 $p$ values in different $e_c$ values														
	-6	-5	-4	-3	-2	-1	0	+1	+2	+3	+4	+5	+6	+7	+8
设	-6	-6	-6	-6	-6	-6	-6	-5	-4	-3	-2	-1	0	0	
定	-4	-6	-6	-6	-5	-5	-4	-3	-3	-3	-2	-1	0	0	
点	-3	-6	-6	-5	-5	-4	-4	-3	-2	-1	-1	-1	0	0	
以	-2	-5	-5	-4	-3	-3	-3	-2	-1	-1	-1	0	0	0	
上	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
设	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
定	2	0	0	0	1	1	1	2	3	3	3	4	4	4	
点	3	0	0	1	1	1	2	3	3	4	4	4	4	4	
以	4	0	0	1	2	3	3	4	4	4	5	6	6	6	
下	5	0	0	1	2	3	3	4	4	4	5	6	6	6	
	6	0	0	1	2	3	4	6	6	6	6	6	6	6	

$e_c$ : 误差变化率 Rate of change of error,  $e_c = de/dt$ ;

$e$ : 给定值与输出值之差 Difference between the fixed and output values;

$p$ : 校正量 Tuning quantity.

## 1.2 自组织模糊控制器控制规则修正算法的实现

### 1.2.1 模糊控制表的建立

一个二维模糊控制器的控制规则可写成如下的条件语句:

$$\text{if } E = \underline{A}_i \text{ and } Ec = \underline{B}_j,$$

then

$$U = \underline{C}_{ij} (i = 1, 2, \dots, n; j = 1, 2, \dots, m)$$

其中  $\underline{A}_i, \underline{B}_j, \underline{C}_{ij}$  是定义在误差、误差变化和控制量论域  $X, Y, Z$  上的模糊集,上述模糊条件语句可用  $\underline{R} = \bigcup_{ij} \underline{A}_i \times \underline{B}_j \times \underline{C}_{ij}$  的模糊关系  $\underline{R}$  来描述,  $\underline{R}$  的隶属函数

$$\mu_{\underline{R}}(x, y, z) = \bigvee_{i=1, j=1}^{i=n, j=m} \mu_{\underline{A}_i}(x) \wedge \mu_{\underline{B}_j}(y) \wedge \mu_{\underline{C}_{ij}}(z),$$

式中  $x \in X, y \in Y, z \in Z$

当误差、误差变化分别取模糊集  $\underline{A}, \underline{B}$  时,根据模糊推理合成规则可得到输出控制量

$$\underline{U} = (\underline{A} \times \underline{B}) \cdot \underline{R}, \underline{U} \text{ 的隶属函数}$$

$$\mu_{\underline{U}}(z) = \bigvee_{x \in X, y \in Y} \mu_{\underline{R}}(x, y, z) \wedge \mu_{\underline{A}}(x) \wedge \mu_{\underline{B}}(y),$$

设论域  $X = (x_1, x_2, \dots, x_n), Y = \{y_1, y_2, \dots, y_m\}, Z = \{z_1, z_2, \dots, z_l\}$ , 则  $X, Y, Z$  上的模糊集分别为一个  $n, m$  和  $l$  元的模糊向量,模糊关系  $\underline{R}$  为一个  $n \times m$  行  $l$  列的矩阵

根据采样得到的误差  $x_i$  和误差变化  $y_j$  可计算出相应的控制量  $U_{ij}$ ,对所有  $X, Y$  中元素的所有组合计算出相应的控制量值  $(U_{ij})^{n \times m}$ ,将  $(U_{ij})^{n \times m}$  矩阵制成如表 2 所示形式的模糊控制表

表 2 模糊控制表

Table 2 Fuzzy control table

$e$	不同 $e_c$ 的 $u$ 值 $u$ values in different $e_c$ values														
	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8
-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-5	-4	-3	
-5	-6	-6	-6	-6	-6	-6	-5	-5	-5	-4	-4	-3	-2		
-4	-5	-5	-4	-3	-4	-3	-3	-2	-1	-2	-1	-1	-1		
-3	-4	-4	-3	-3	-3	-2	-2	-1	-1	0	0	0	0		
-2	-4	-3	-3	-2	-2	-1	-1	0	0	0	0	0	0	1	
-1	-3	-2	-2	-1	-1	-1	-1	0	0	0	0	0	1	2	
0	-3	-2	-1	-1	-1	-1	0	1	1	1	1	2	3		
1	-2	-1	0	0	0	0	1	1	1	1	2	2	3		
2	-1	0	0	0	0	0	2	1	2	2	3	3	4		
3	0	0	0	0	1	1	3	2	3	3	3	4	4		
4	1	1	1	1	2	2	3	3	3	3	4	5	5		
5	2	3	4	4	5	5	5	6	6	6	6	6	6		
6	3	4	5	6	6	6	6	6	6	6	6	6	6		

$e_c$ : 误差变化率 Rate of change of error,  $e_c = de/dt$ ;

$e$ : 给定值与输出值之差 Difference between the fixed and output values;

$p$ : 校正量 Tuning quantity.

### 1.2.2 模糊控制规则修正算法

#### (1) 模糊控制表的校正

表 1 可表示为  $U(nT) = x [e(nT), e_c(nT)]$ , 式中  $x$  表示模糊控制表,  $nT$  表示当前采样时刻。

设系统有一定滞后, 即过去  $m$  个采样周期的控制作用对目前 (即  $nT$  时刻) 比较差的性能有影响, 这里称  $nT$  时刻为校正点, 称  $(nT - mT)$  时刻为被校正点, 现要根据校正点的性能测量值求得校正量  $r(nT)$  值。  $r(nT)$  称为控制量的校正量) 去校正被校正点的控制量  $U(nT - mT)$

若初始规则反映在控制表上为  $x [e(nT - mT), e_c(nT - mT)] = U(nT - mT)$ , 经过  $nT$  时刻的性能测试后, 将上述规则修正为  $x [e(nT - mT), e_c(nT - mT)] = V(nT - mT)$ , 其中  $V(nT - mT) = U(nT - mT) + r(nT)$ , [对于纯滞后较小的系统可取  $r(nT) = P(nT)$ ] 但通过仿真发现: 若连续两次测得的误差和误差变化量都相等, 即  $e(nT - mT) = e(nT - mT + T)$ ,  $e_c(nT - mT) = e_c(nT - mT + T)$ , 由于在  $(nT)$  时刻已将  $x [e(nT - mT), e_c(nT - mT)] = U(nT - mT)$  改为  $x [e(nT - mT), e_c(nT - mT)] = U(nT - mT) + r(nT)$  而到了  $(nT + T)$  时刻, 原规则已不存在了, 必须将新规则  $x [e(nT - mT + T), e_c(nT - mT + T)] = x [e(nT - mT), e_c(nT - mT)] = V(nT - mT)$  再改为  $x [e(nT - mT + T), e_c(nT - mT + T)] = x [e(nT - mT), e_c(nT - mT)] = U(nT - mT) + r(nT) + r(nT + T)$ , 考虑到上述情况, 总结出模糊控制表的修正方法为: a. 若连续两次测得的被校正点的误差或误差变化量不相等, 则将控制表  $x$  中的控制量修正为  $x [e(nT - mT), e_c(nT - mT)] = V(nT - mT) = U(nT - mT) + r(nT)$ ; b. 若连续两次测得的被校正点的误差和误差变化量都相等, 那么在第二个被校正点将控制表  $x$  中的控制量修正为  $x [e(nT - mT), e_c(nT - mT)] = \text{int} \{ \alpha x [e(nT - mT), e_c(nT - mT)] + (1 - \Gamma) V(nT - mT) \}$ ,  $\text{int}$  表示取一个不大于  $x$  的最大整数,  $\Gamma$  为加权因子,  $0 < \Gamma \leq 1$ ,  $\Gamma$  越大, 对前一点的加权值就越大

#### (2) 模糊控制表校正算法的实现

由以上分析可得出校正算法步骤为

- ① 根据输出特性计算出  $e(nT)$  和  $e_c(nT)$ ,
- ② 查表 2 可得  $U(nT - mT)$ ,
- ③ 查表 1 可得  $P(nT)$ , 且  $r(nT) = P(nT)$ ,
- ④ 计算  $V(nT - mT) = U(nT - mT) + r(nT)$ ,
- ⑤ 如果  $e(nT - mT) = e(nT - mT - T)$  且  $e_c$

$(nT - mT) = e_c(nT - mT - T)$ , 则  $x [e(nT - mT), e_c(nT - mT)] = \text{int} \{ \alpha x [e(nT - mT), e_c(nT - mT)] + (1 - \Gamma) V(nT - mT) \}$  否则  $x [e(nT - mT), e_c(nT - mT)] = V(nT - mT)$

利用这种算法可实现对控制表的自动校正, 但如何选择校正点和如何建立校正表问题仍存在许多实际困难

为了克服校正算法实现过程中的困难, 我们采用特殊点优先校正算法

#### (3) 特殊点优先校正算法

从图 2 响应曲线可知: 在  $\theta - P_1$  线段上, 系统在控制信号作用下, 由静态到动态再向稳态转变, 由于系统具有惯性, 决定了这段曲线呈倾斜上升趋势, 在  $P_1$  点系统的输出达到给定值并开始偏离给定值向误差增大的方向变化, 到了  $P_2$  点, 误差达到最大值。对于一般的惯性系统, 如果  $P_1 \sim P_2$  这段时间延续很长, 不但带来很大的超调, 而且系统后面的状态影响很坏。

为了抑制系统响应在  $P_1 \sim P_2$  段上的延续, 应首先考虑在这个时间段上选择校正点, 本文定义  $P_1 \sim P_2$  段上前三个采样点为特殊点并首先选择这三个特殊点为校正点, 进行性能测量, 以修正控制规则。

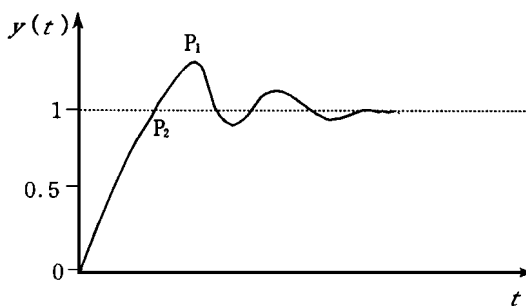


图 2 响应曲线

Fig. 2 Response curve

利用特殊点优先校正法, 对一个具有大滞后、大惯性、可用高阶方程  $y(t) = L^{-1} [1/s(T_m S + 1)]$  描述的对象——工业电炉进行了计算机仿真研究, 经过反复逼近, 其数学模型可简化为  $G(s) = k \times \exp(-zs) / (T_m S + 1)$ , 其中  $k$  为飞升曲线中的稳态值  $Y_\infty$ ,  $z$  为对象的纯滞后时间,  $T_m$  为惯性时间常数

根据用户提供的资料, 可画出飞升曲线, 然后用一阶惯性环节飞升曲线反复逼近高阶飞升曲线, 最终求得工业电炉模型参数如下: 炉型 100 kW 箱式炉,  $z = 482 s$ ,  $T_m = 1085 s$ ,  $k = 682$ ,  $z/T_m = 0.444$ , 仿真步骤如下: 使用表 2 作出模糊控制器的初始控制表,

对该炉进行控制,其阶跃响应曲线如图 3 (a) 中曲线①所示,性能测量结果超调很大,过渡时间很长,经过 4 个多小时,仍未进入稳态过程,这说明初始控制表 2 对该系统是不理想的,需要校正。

先用一般方法,对初始控制表进行校正,经过反复调试,取校正周期为 50 s,校正后的阶跃响应曲线如图 3 (a) 的曲线②所示,经过 200 多次校正,使得系统趋于稳定,但过渡时间仍很长,约需 3 h,且超调量达 20% 之多。

采用特殊点优先校正法对初始控制表进行校正后的阶跃响应曲线如图 3 (b) 中曲线②所示,只要经过 25 次校正,系统的性能明显优于图 3 (a) 中曲线②的性能,完全消除了超调,过渡时间加快,调节时间缩短了 73.2%。

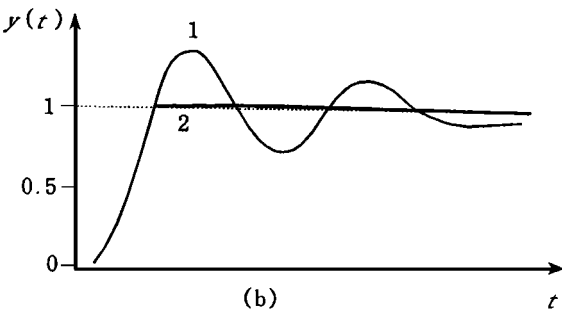
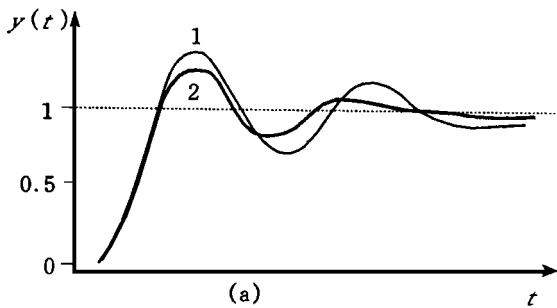


图 3 100 kW 电炉的响应曲线

Fig. 3 Response curve of 100 kW furnace

(a) 固定间隔选择校正点

(a) Choose tuning point by fixed interval

(b) 特殊点优先校正

(b) Giving priority to tuning in special point

特殊点优先校正算法不但对大滞后,大惯性系统控制效果很好,而且对一些非线性对象的控制情况也较令人满意。

我们采用  $y(k+1) = y(k) / [1 + y^2(k) + u^3(k)]$  式描述的非线性对象进行计算机仿真,在毫无先验知识的情况下,先用表 2 做初始模糊控制表,并细心选择其他控制参数,系统的响应曲线如图 4 中的曲线①所示,响应曲线开始有较大的振盈,而且超调也很大,用特殊点优先校正法对初始模糊控制表校正后,其响应曲线如图 4 曲线②所示,图 4 表明振盈和超调已消除,响应特性很平稳。

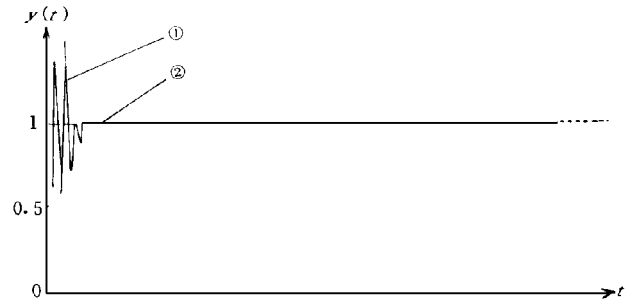


图 4 非线性对象的响应曲线

Fig. 4 Response curve of nonlinear subject

## 2 结论

采用特殊点优先校正算法对模糊控制表进行校正比一般校正算法控制超调量要小得多,而且总的收敛时间快得多。

对于大于 0.5 的不同  $z/T_m$  值进行仿真均能实现无超调或小超调,调整时间短,鲁棒性强等特点。

采用特殊点优先校正算法对于大滞后,大惯性,非线性系统的控制效果是十分满意的。

## 参考文献

- 1 Procyk T J, Mamdani E H. A linguistic self-organizing process controller. Automatica. 1970, 15 (1): 15~ 30.
- 2 Mamdani E H. Advances in the linguistic synthesis of fuzzy controllers. Int J. Manmach stnd, 1976, (8): 669 ~ 678.
- 3 洪钟威,邵世煌.用 Z-80对自组织模糊控制器进行仿真研究.微型计算机, 1981, (6).
- 4 冯宏娟,王守觉.直接修改控制规则的自调整模糊控制器.电子学报, 1992, (2): 10~ 15.
- 5 王学慧,田成方.微机模糊控制理论及其应用.北京:电子工业出版社, 1987. 135~ 185.

(责任编辑: 蒋汉明 邓大玉)