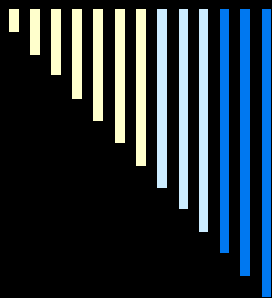




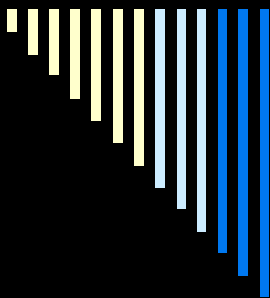
第4章 计算机控制系统的控制算法

- 控制算法决定着控制规律
- 控制规律是反映计算机控制系统性能的核心
- 本章介绍目前常用的几种控制规律



本章要点

- 数字滤波和数据处理
- PID控制算法
- 史密斯预估控制原理与控制器
- 解耦控制
- 模糊控制

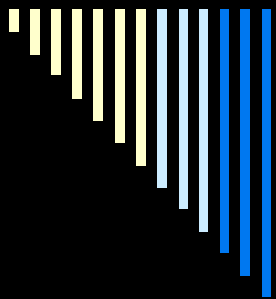


4.1 数字滤波和数据处理

计算机控制系统采集生产过程的各种物理参数，如温度、压力、流量、料位和成分等。

这些原始数据中可能混杂了干扰噪声，需要进行数字滤波。

这些原始数据可能与实际物理量成非线性关系，需要进行线性化处理。



4.1.1 数字滤波

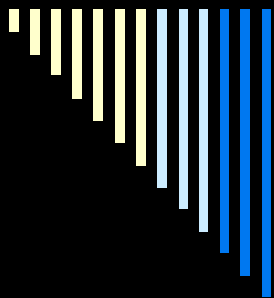
噪声分为两类：

- 一类为周期性，典型代表为50Hz的工频干扰，对于这类信号，采用积分时间等于20ms的整数倍的双积分A/D转换器，就可以有效地消除；
- 另一类是不规则的随机信号，对此可以用数字滤波方法予以消除或过滤。



数字滤波的优点

- 数字滤波是用程序实现的不需要增加硬设备，所以稳定性好、可靠性高，成本低；
- 数字滤波器可以对极低频（如0.01Hz）干扰信号进行滤波，克服了模拟滤波器的不足；（一般的高频和低频干扰可以采用RC低通和高通滤波器加以抑制。）
- 数字滤波器可以根据信号的不同，采用不同的滤波方法或滤波参数，具有灵活、方便、功能强的特点。



什么是数字滤波？

在计算机中用某种计算方法对输入的信号进行数学处理，以便减少干扰在有用信号中的比重，提高信号的真实性和可靠性。



1. 程序判断法（限幅滤波）

由于工业现场测控系统存在随机脉冲干扰，通过变送器将尖脉冲干扰引入输入端，从而造成测量信号的严重失真。

$$\Delta y_n = \left| y_n - \bar{y}_{n-1} \right|$$
$$\bar{y}_n = \begin{cases} y_n, & \Delta y_n \leq \alpha \\ \bar{y}_{n-1}, & \Delta y_n > \alpha \end{cases}$$

α 表示相邻两个采样值之差的最大可能变化范围。



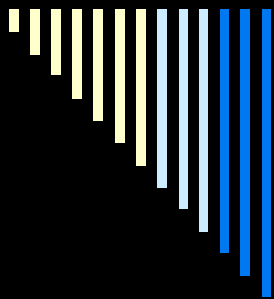
α 值的选择

- 过程的动态特性决定其输出参数的变化速度。通常按照参数可能的最大变化速度 v_{max} 及采样周期 T ，决定 a 值。

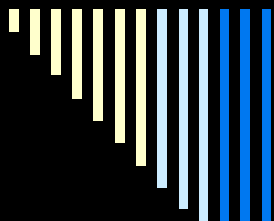


2. 中位值滤波法

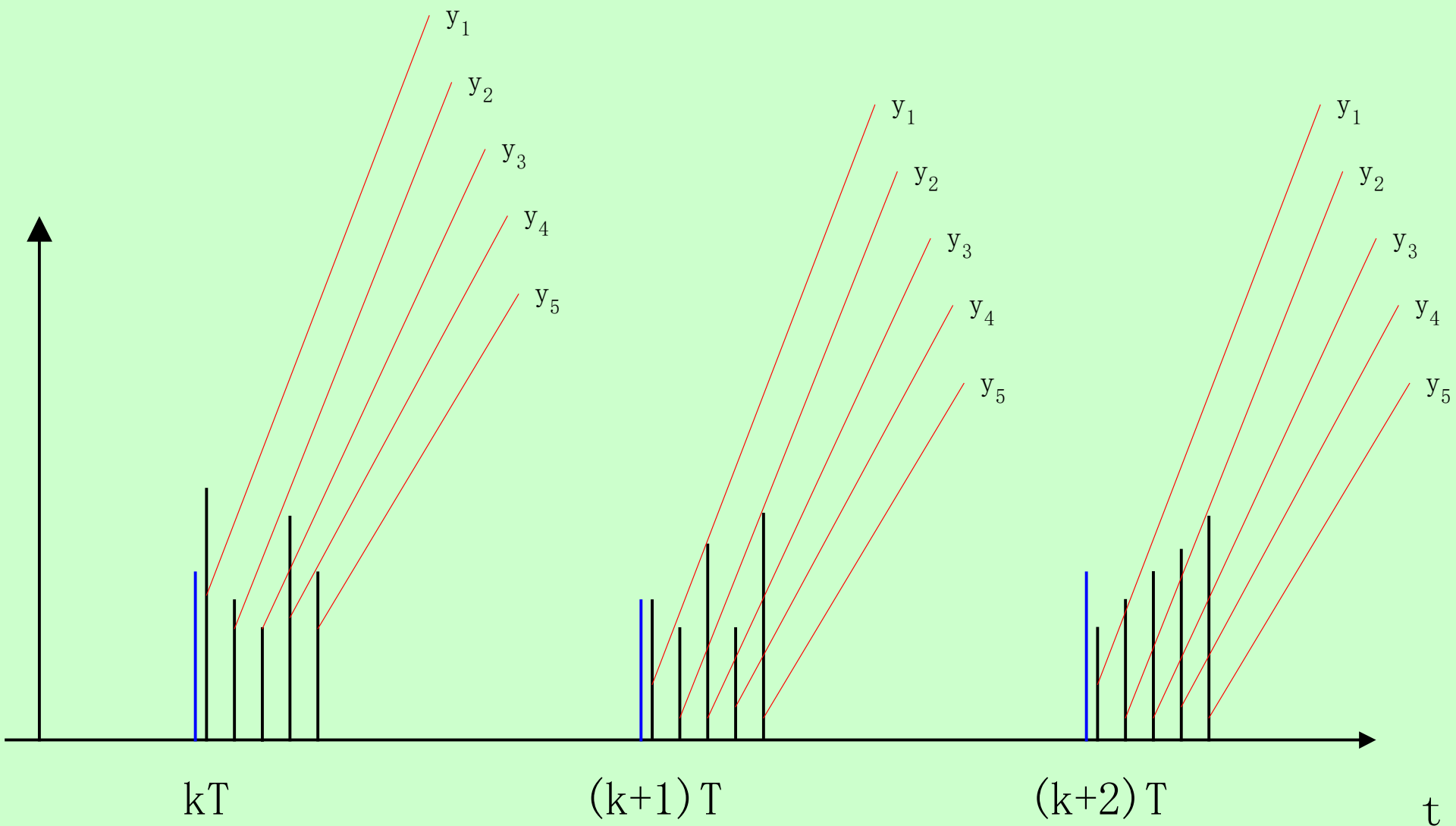
在某一采样时刻（采样点），对某一被测参数连续采样 n 次(一般 n 取奇数)，然后把 n 次采样值按大小排队，取中间值为本次采样值。



- 中位值滤波能有效地克服因偶然因素引起的波动或采样器不稳定引起的误码等造成的脉冲干扰。
- 对缓慢变化的过程采用中位值滤波有良好的效果。
- n 越大，排序算法所占的时间越长。



中位值滤波法

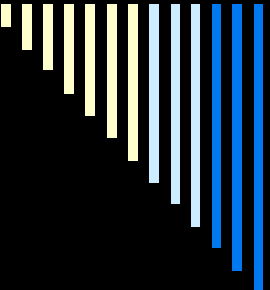




3. 算术平均滤波法

在某一采样时刻（采样点），对某一被测参数连续采样 **N** 次，然后把 **N** 个测量值进行算术平均。

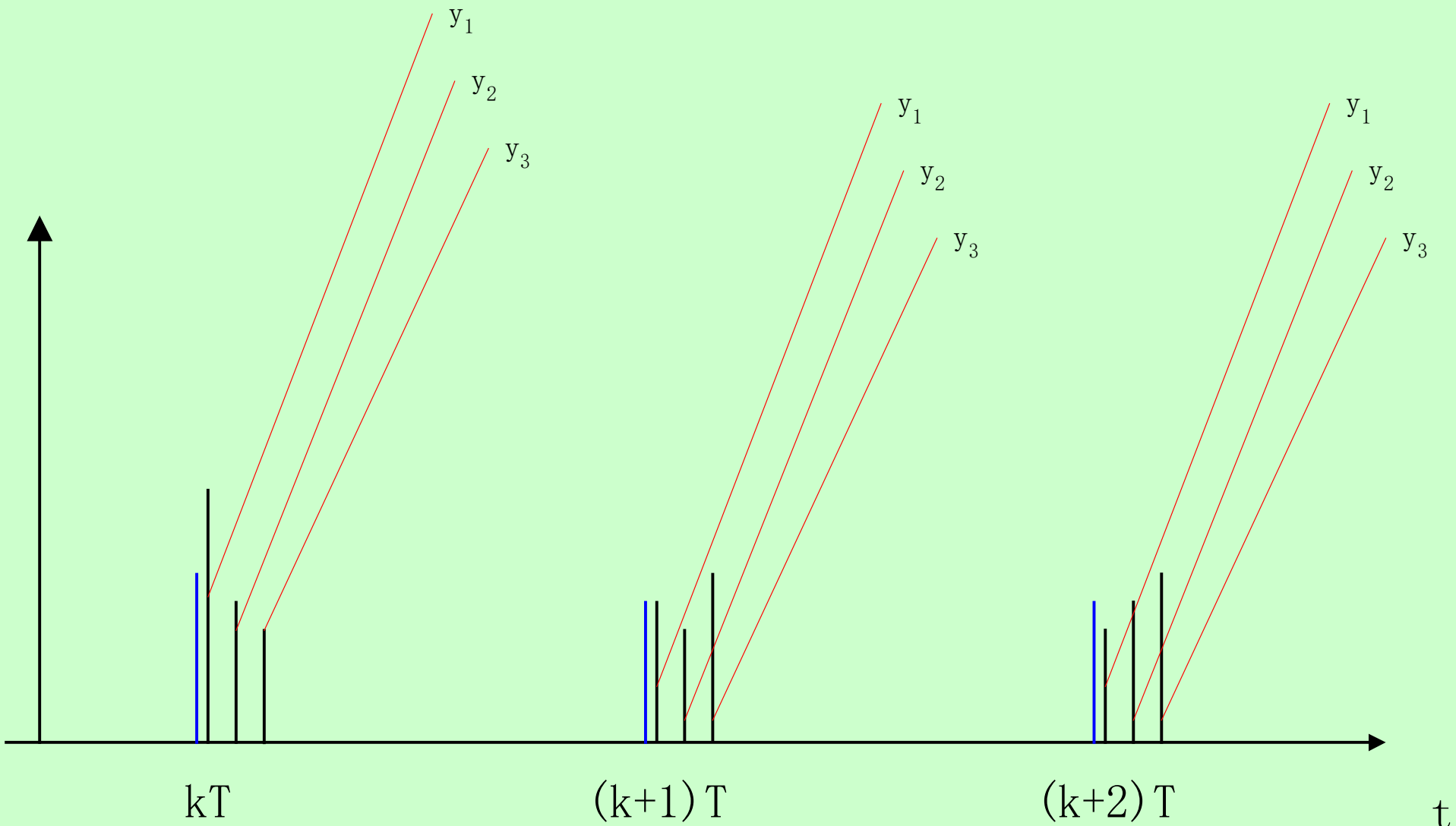
$$y_n = \frac{1}{N} \sum_{i=1}^N y_{ni}$$

- 
- 算术平均滤波法适用于对一般具有随机干扰的信号进行滤波。
 - 这种信号的特点是有有一个平均值，信号在某一数值范围附近作上下波动，在这种情况下仅取一个采样值为依据显然是不准确的。

流量测量，通常取 $N=8\sim 12$

压力测量，通常取 $N=4\sim 8$

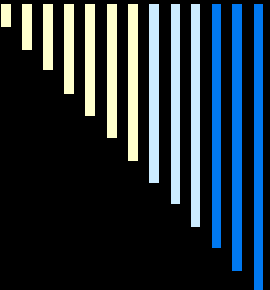
算术平均滤波法





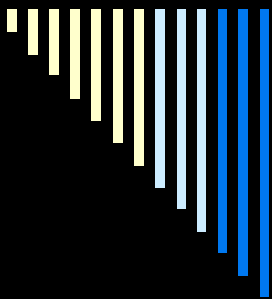
4. 递推平均滤波法

算术平均滤波法，每计算一次数据，需采样 N 次。对于采样速度较慢或要求数据计算速度较高的系统，该方法是无法使用的。例如某A / D芯片转换速率为10次 / s，而要求每秒输入4次数据时，则 N 不能大于2。



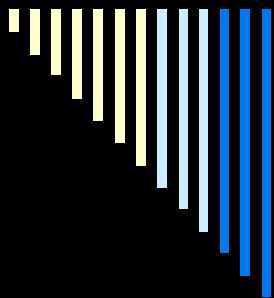
递推平均滤波法，只需进行一次测量，就能得到当前算术平均滤波值。

把 N 个采样数据看成一个队列，队列的长度固定为 N ，每进行一次新的采样，把采样结果放入队尾，而扔掉原来队首的一次数据。



在队列中始终有 N 个的数据，计算滤波值时，只要把队列中的 N 个数据进行算术平均，就可得到新的滤波值。

这样每进行一次采样，就可计算得到一个新的平均滤波值。



$$\overline{y}_n = \frac{1}{N} \sum_{i=0}^{N-1} y_{n-i}$$

\overline{y}_n

第n次采样值经滤波后的输出

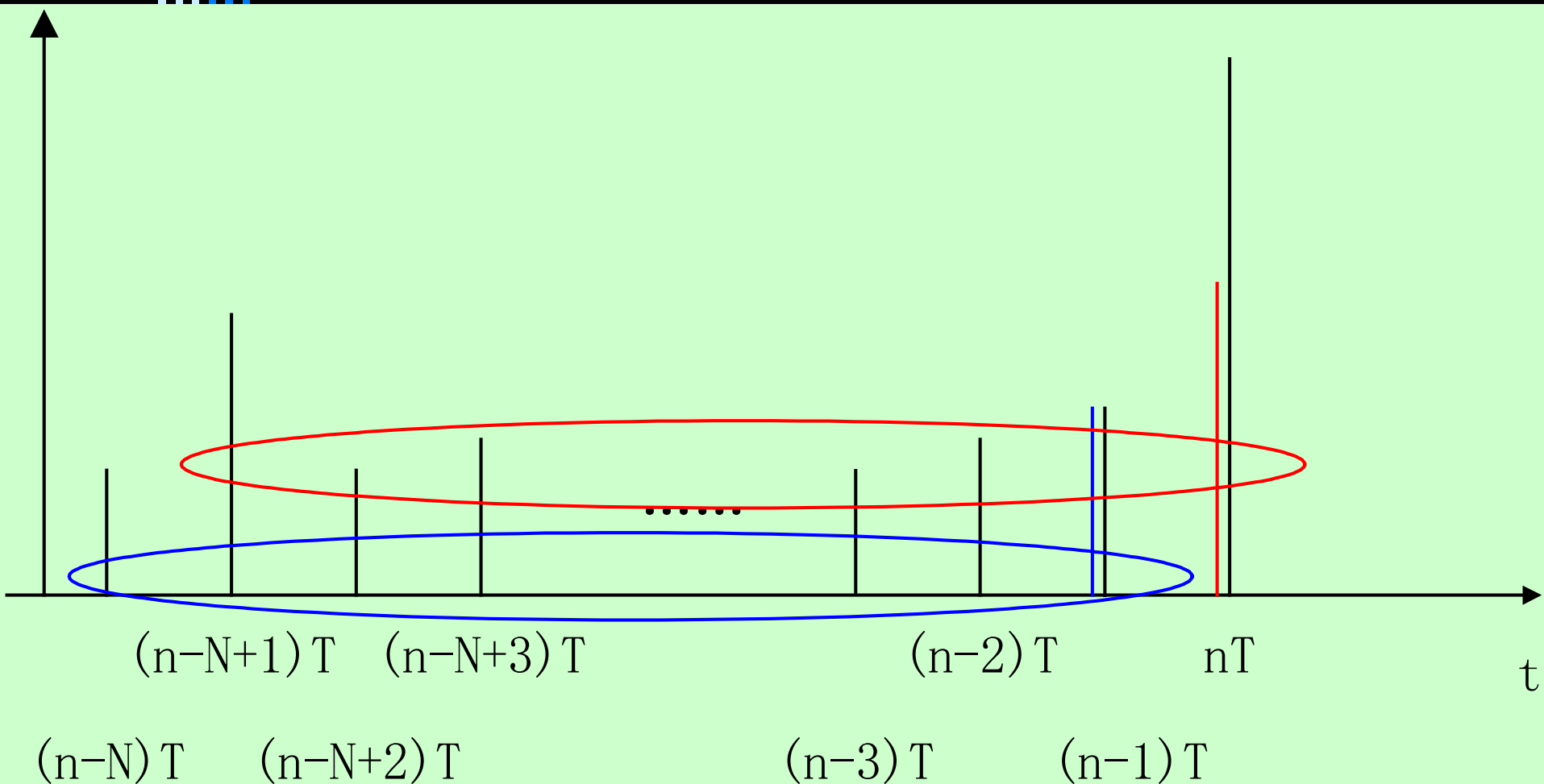
y_{n-i}

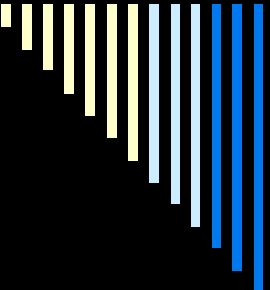
未经滤波的第 n-i 次采样值

N

递推平均项数

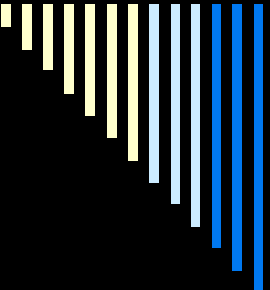
递推平均滤波法





递推平均滤波算法对周期性干扰有良好的抑制作用，平滑度高，灵敏度低；

但对偶然出现的脉冲性干扰的抑制作用差，不易消除由于脉冲干扰引起的采样值偏差，因此它不适用于脉冲干扰比较严重的场合，而适用于高频振荡的系统。



通过观察不同N值下递推平均的输出响应来选取N值，以便既少占用计算机时间，又能达到最好的滤波效果。

N 的工程经验值如下：

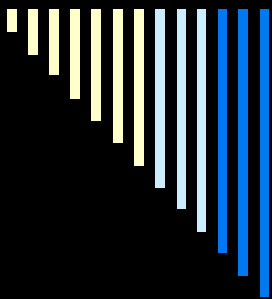
- 流量 12
- 压力 4
- 液面 4 ~ 12
- 温度 1 ~ 4



5. 加权递推平均滤波法

算术平均滤波法和递推平均滤波法中， N 次采样值在输出结果中的比重是均等的，即 $1 / N$ 。用这样的滤波算法，对于测量信号会引入滞后， N 越大，滞后越严重。

为了增加最新采样数据在递推平均中的比重，以提高系统对当前采样值的灵敏度，可以采用加权递推平均滤波算法。



加权递推平均滤波算法是递推平均滤波算法的改进，即不同时刻的数据加以不同的权，通常越接近当前时刻的数据，权取得越大，N项加权递推平均滤波算法为

$$\overline{y}_n = \frac{1}{N} \sum_{i=0}^{N-1} C_i y_{n-i}$$



上式中, C_0, C_1, \dots, C_{N-1} 为常数, 且满足如下条件:

$$C_0 + C_1 + \dots + C_{N-1} = 1$$

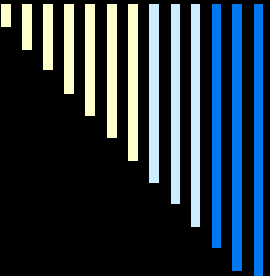
$$C_0 > C_1 > \dots > C_{N-1} > 0$$

用加权系数法选取常系数 C_i

$$R = 1 + e^{-\tau} + e^{-2\tau} + \dots + e^{-(N-1)\tau}$$

$$C_0 = \frac{1}{R}, C_1 = \frac{e^{-\tau}}{R}, \dots, C_{N-1} = \frac{e^{-(N-1)\tau}}{R}$$

其中 τ 为对象的纯迟后时间。



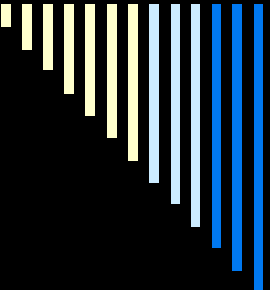
因为 T 越大， R 越小，则给予新的采样值的权系数就越大，而给予先前采样值权系数就越小，从而提高了新的采样值在平均过程中的比重。

所以加权递推平均滤波算法适用于有较大纯滞后时间常数的对象和采样周期较短的系统，而对于纯滞后时间常数较小、采样周期较长、变化缓慢的信号，则不能迅速反映系统当前所受干扰的严重程度，滤波效果差。



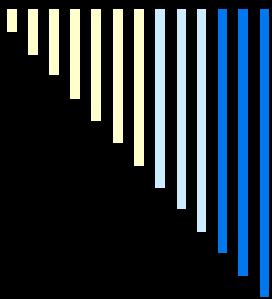
6. 一阶惯性滤波法

在模拟量输入通道等硬件电路中，常用一阶惯性 RC 模拟滤波器来抑制干扰，当用这种模拟方法来实现对低频干扰的滤波时，首先遇到的问题是要求滤波器有大的时间常数和高精度的 RC 网络。时间常数 T 越大，要求 RC 值越大，其漏电流也随之增大，从而使 RC 网络的误差增大，降低了滤波效果。



一阶惯性滤波算法是一种以数字形式通过算法来实现动态的 RC 滤波方法，它能很好克服上述模拟滤波器的缺点，在滤波常数要求大的场合，此法更为实用。一阶惯性滤波算法为

$$\overline{y_n} = (1 - \alpha)y_n + \alpha \overline{y_{n-1}}, \alpha = \frac{T_f}{T + T_f}$$

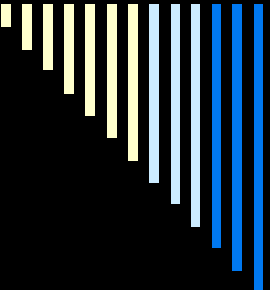


y_n 未经滤波的第 n 次采样值

T_f 滤波时间常数

T 采样周期

根据一阶惯性滤波的频率特性，若滤波系数 α 越大，则带宽越窄，滤波频率也越低。因此，需要根据实际情况，适当选取 α 值，使得被测参数既不出现明显的波纹，反应又不太迟缓。



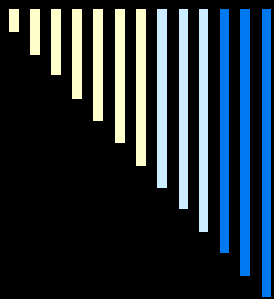
以上讨论了六种数字滤波方法，在实际应用中，究竟选取哪一种数字滤波方法，应视具体情况而定。

平均值滤波法适用于周期性干扰；

中位值滤波法和限幅滤波法适用于偶然的脉冲干扰；

惯性滤波法适用于高频及低频的干扰信号；

加权平均值滤波法适用于纯迟延较大的被控制对象。



如果同时采用几种滤波方法，一般先用中位值滤波法或限幅滤波法，然后再用平均值滤波法。

如果应用不恰当，非但达不到滤波效果，反而会降低控制品质。



4.1.2 数据处理

采用了上述数字滤波方法，虽然可以得到比较真实的被测参数，但有时并不能直接使用这些采样数据，还需要对它们作某些数学处理。例如，对孔板差压信号进行开方运算、热电偶信号的线性化处理等。



1. 线性化处理

计算机从模拟量输入通道得到的检测信号与该信号所代表的物理量之间不一定成线性关系。例如，差压变送器输出的孔板差压信号同实际的流量之间成平方根关系；热电偶的热电势与其所测温度的关系成非线性等。

而在计算机内部参与运算和控制的二进制数希望与被测参数之间成线性关系，其目的是既便于运算又便于数字显示。为此，必须对非线性参数进行线性化处理。



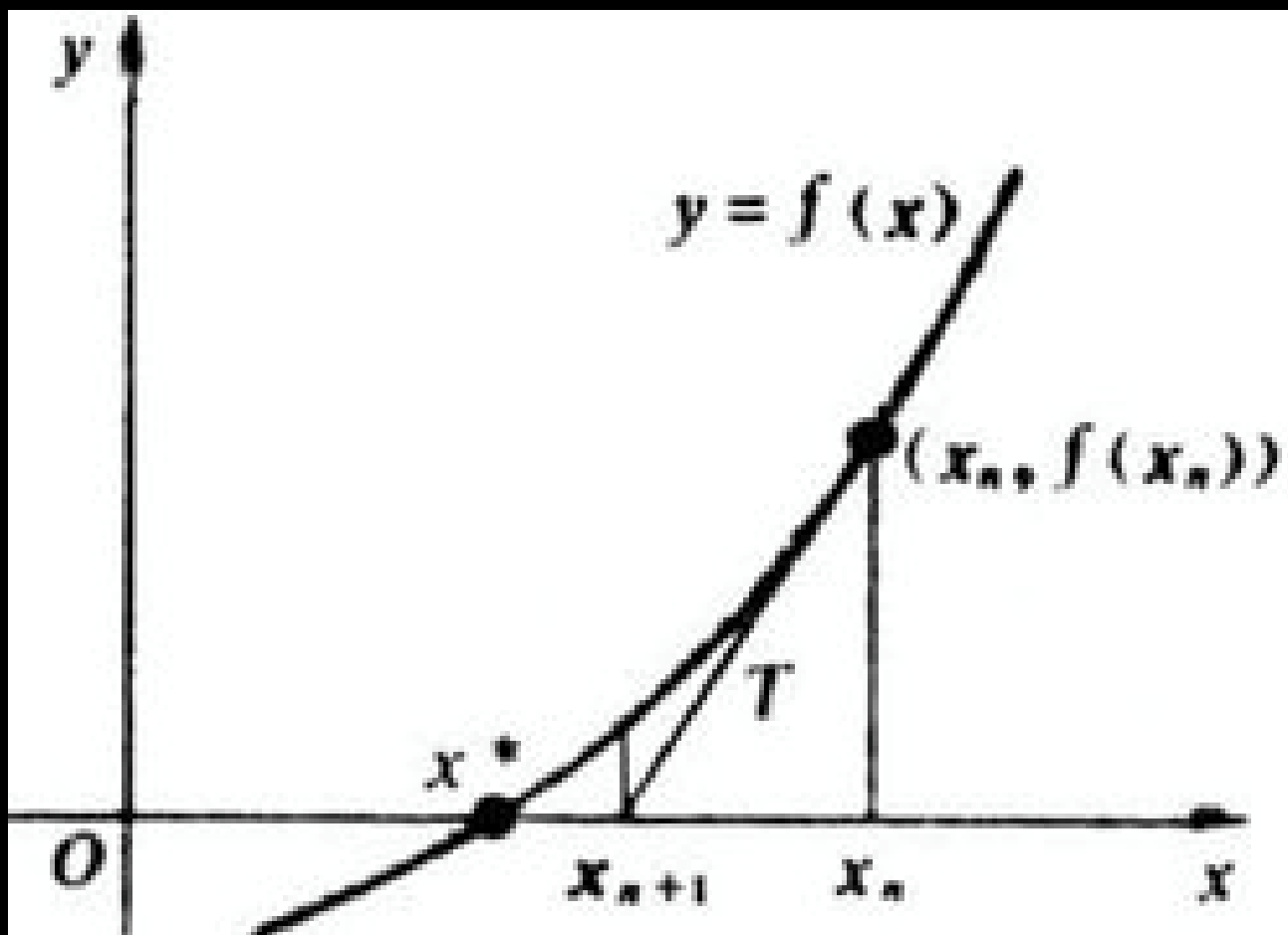
(1) 孔板差压与流量

用孔板测量气体或液体的流量，差压变送器输出的孔板差压信号 ΔP 同实际流量 Q 之间是平方根关系，即

$$Q = K \sqrt{\Delta P}$$

式中， K 是流量系数。

当计算机内没有开方函数时，则软件开发者用数值分析的方法计算平方根，可采用牛顿(Newton)迭代法。

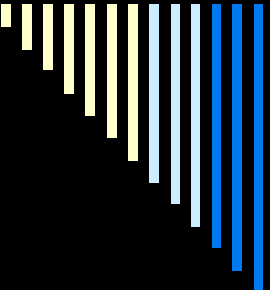




(2) 热电偶的热电势与温度

热电偶是常见的测温元件，但热电势与温度成非线性关系，因而需要线性化。常用热电偶的热电势 E 与温度 T 存在如下式所示的关系

$$T = a_4 E^4 + a_3 E^3 + a_2 E^2 + a_1 E + a_0$$



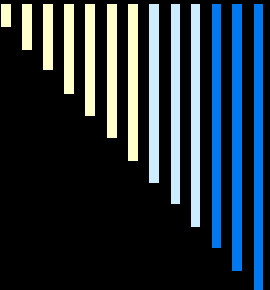
当计算机不能实时进行上述运算时，则对于上式表示的热电偶的热电势 E 与温度 T 的非线性关系，可以采取分段的办法处理；线性化时首先判断测量数据处于哪一折线区间内，然后按相应的线性化公式计算出线性值。折线段数越多，线性化精度就越高。

除此之外，还可将热电偶分度表以表格形式存在计算机内，在线的工作量便仅仅是根据采样值查表。(P111)



2. 标度变换

生产过程中的各种参数都具有不同的量纲和数值变化范围。如电压的单位为V、电流的单位为A、温度的单位为 $^{\circ}\text{C}$ 等。而且经一次检测仪表输出信号的变化范围也不相同，如热电偶的输出为毫伏信号，电压互感器的输出为 $0\sim 100\text{V}$ ；电流互感器的输出为 $0\sim 5\text{A}$ 等。



所有这些具有不同量纲和数值范围的信号又都经各种形式的变送器转化为统一信号范围，如 $0\sim 5\text{ V}$ ，可经A / D转换成数字量(如 8 位A / D，则数字可以从 $00\sim\text{FFH}$)。为了进行显示、打印、记录或报警，又必须把这些数字量转换成具有不同量纲的数值，以使操作人员进行监视和管理，这就是所谓的标度变换，也称为工程量转换。



(1) 线性参数的标度变换

线性参数：指一次仪表测量值与 A / D 转换结果具有线性关系，或者说一次仪表是线性刻度的。其标度变换公式为

$$A_x = A_0 + (A_m - A_0) \frac{N_x - N_0}{N_m - N_0}$$

A_0 一次测量仪表的下限

A_m 一次测量仪表的上限

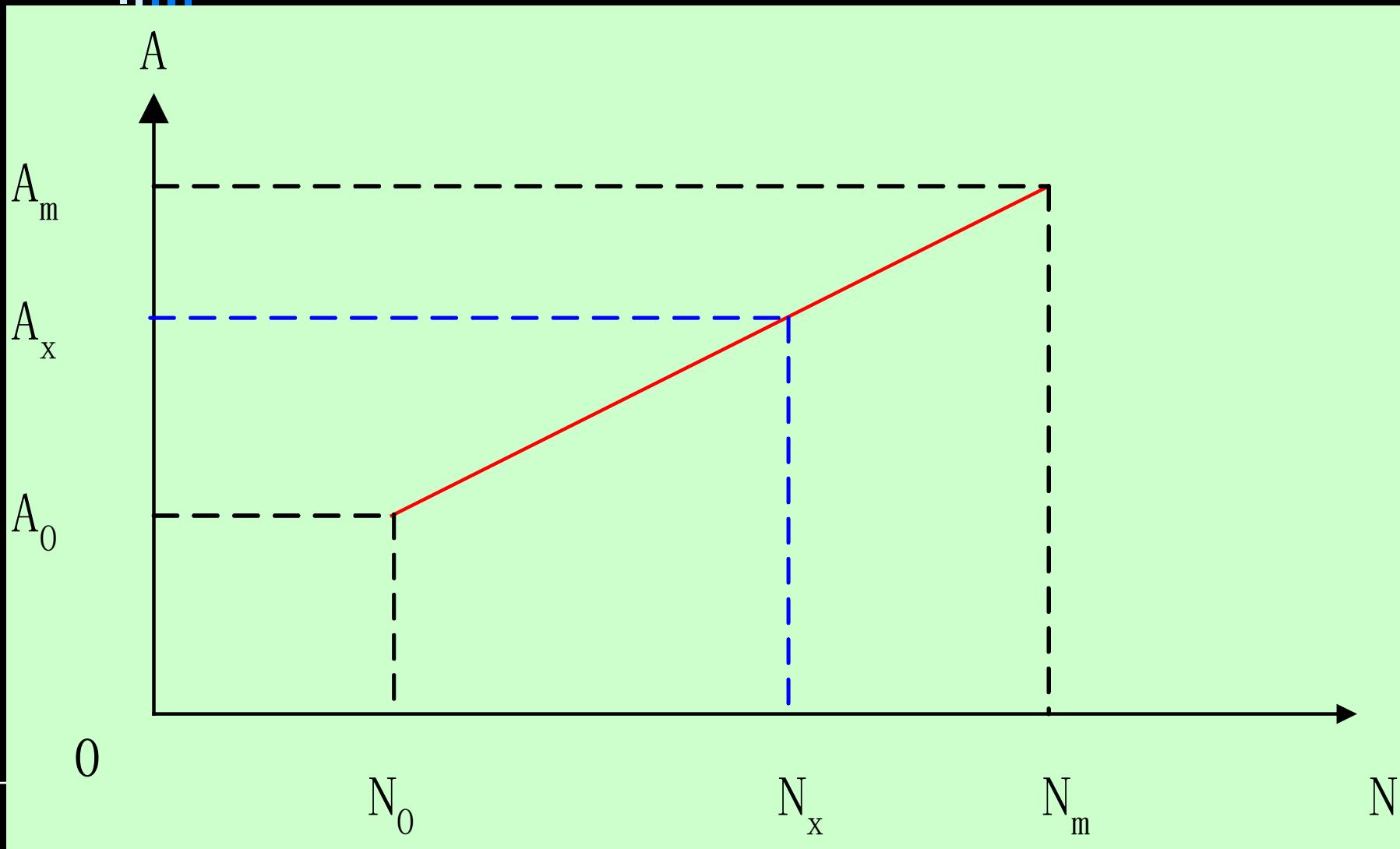
A_x 实际测量值(工程量)

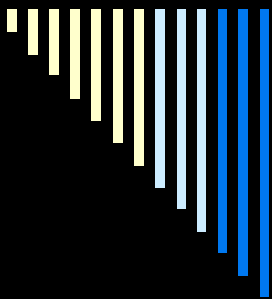
N_0 仪表下限对应的数字量

N_m 仪表上限对应的数字量

N_x 测量值所对应的数字量

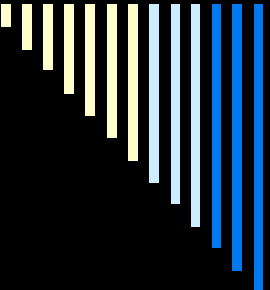
线性参数的标度变换





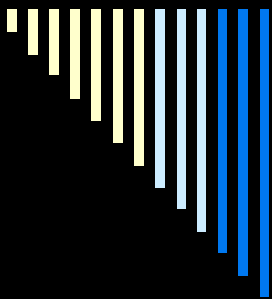
为使程序简单，一般把被测参数的起点 A_0 (输入信号为 0) 所对应的 A / D 输出值为 0，即 $N_0 = 0$ ，这样，上式可化为

$$A_x = \frac{N_x}{N_m} (A_m - A_0) + A_0$$



有时，工程量的实际值还需经过一次变换。如电压测量值是电压互感器的二次侧的电压，其一次侧的电压还有一个互感器的变化问题，这时上式应再乘上一个比例系数，即

$$A_x = K \cdot \left[\frac{N_x}{N_m} (A_m - A_0) + A_0 \right]$$



例：某热处理炉温度测量仪表的量程为 $200^{\circ}\text{C} \sim 800^{\circ}\text{C}$ ，在某一时刻计算机采样并经数字滤波后的数字量为0CDH，求此时温度值为多少？
(设仪表量程为线性的)



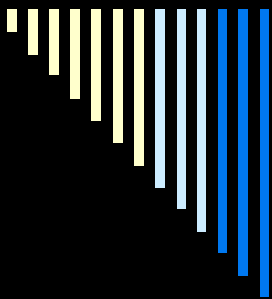
解：已知 $A_0 = 200^{\circ}\text{C}$, $A_m = 800^{\circ}\text{C}$,

$N_0 = 0$, $N_m = 0$ FFH = 255

$N_x = 0$ CDH = 205

此时温度为

$$A_x = K \cdot \left[\frac{N_x}{N_m} (A_m - A_0) + A_0 \right]$$
$$= \left[\frac{205}{255} (800 - 200) + 200 \right]^{\circ}\text{C} = 682^{\circ}\text{C}$$



在微机控制系统中，为了实现上述转换，可把它设计成专门的子程序，把各个不同参数所对应的 A_0 、 A_m 、 N_0 、 N_m 存放在存储器中，然后当某一参数要进行标度变换时，只要调用标度变换子程序即可。



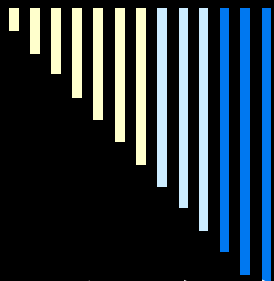
(2) 非线性参数的标度变换

在过程控制中，最常见的非线性关系是差压变送器信号 ΔP 与流量 Q 的关系

$$Q = K \sqrt{\Delta P}$$

据此，可得测量流量时的标度变换式为

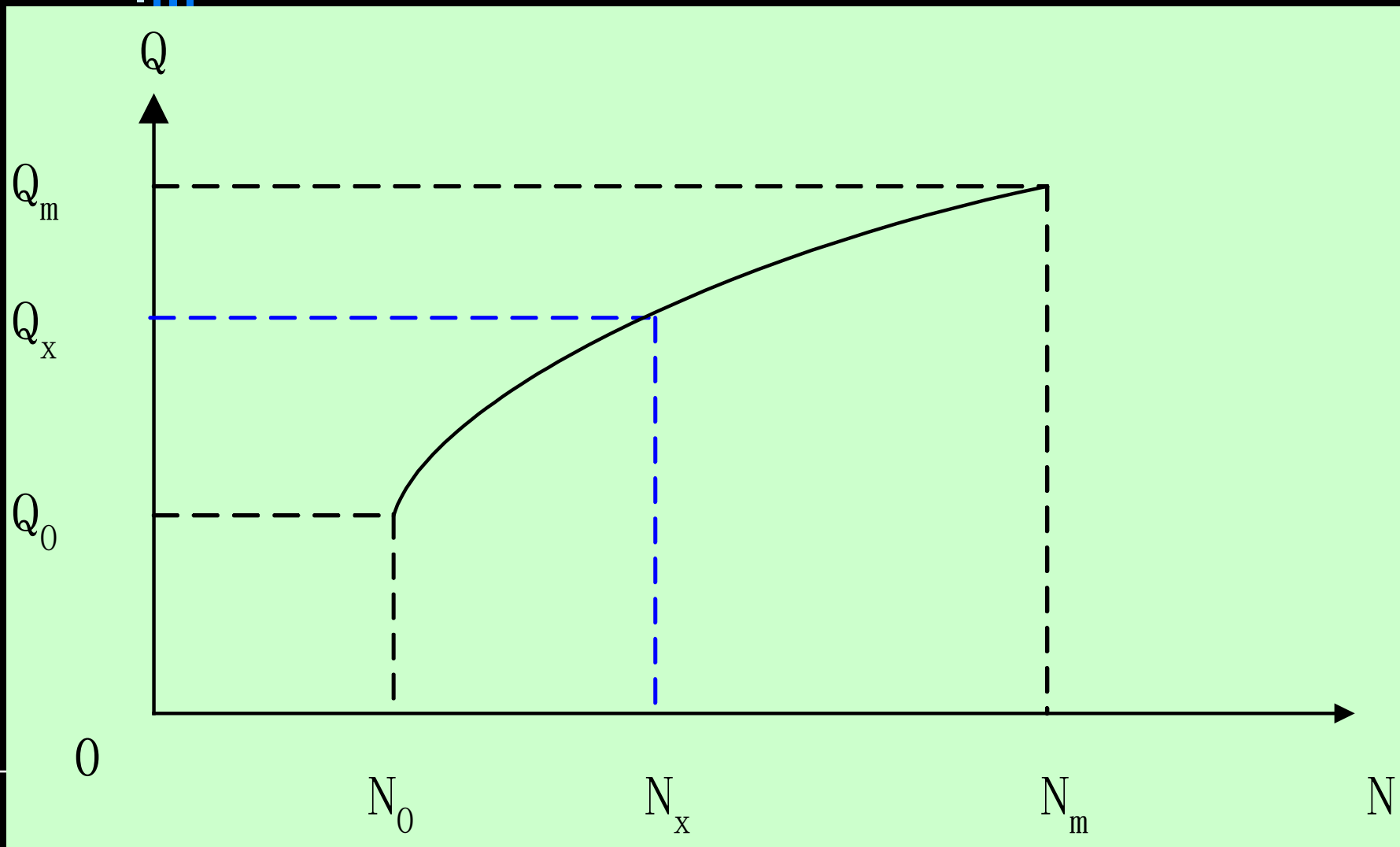
$$\frac{Q_x - Q_0}{Q_m - Q_0} = \frac{K \sqrt{N_x} - K \sqrt{N_0}}{K \sqrt{N_m} - K \sqrt{N_0}}$$

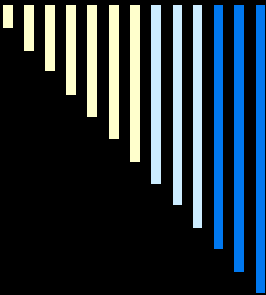


上式中,

Q_x	被测量的流量值
Q_m	流量仪表的上限值
Q_0	流量仪表的下限值
N_x	差压变送器所测得的差压值(数字量)
N_m	差压变送器上限所对应的数字量
N_0	差压变送器下限所对应的数字量

非线性参数的标度变换





对于流量测量仪表，一般下限为取0，此时 $Q_0 = 0$ ， $N_0 = 0$ ，故上式变为

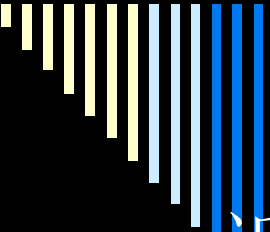
$$Q_x = Q_m \frac{\sqrt{N_x}}{\sqrt{N_m}}$$

同样，可用子程序调用方法实现非线性参数的标度变换。



4.2 数字PID控制

在过程控制中，*PID控制算法是应用最为广泛的一种控制规律*。它具有原理简单、易于实现、鲁棒性强和适用面广等优点，在计算机用于生产过程控制以前，模拟PID调节器几乎一直占垄断地位。计算机的出现和它在过程控制中的应用使这种情况开始有所改变。



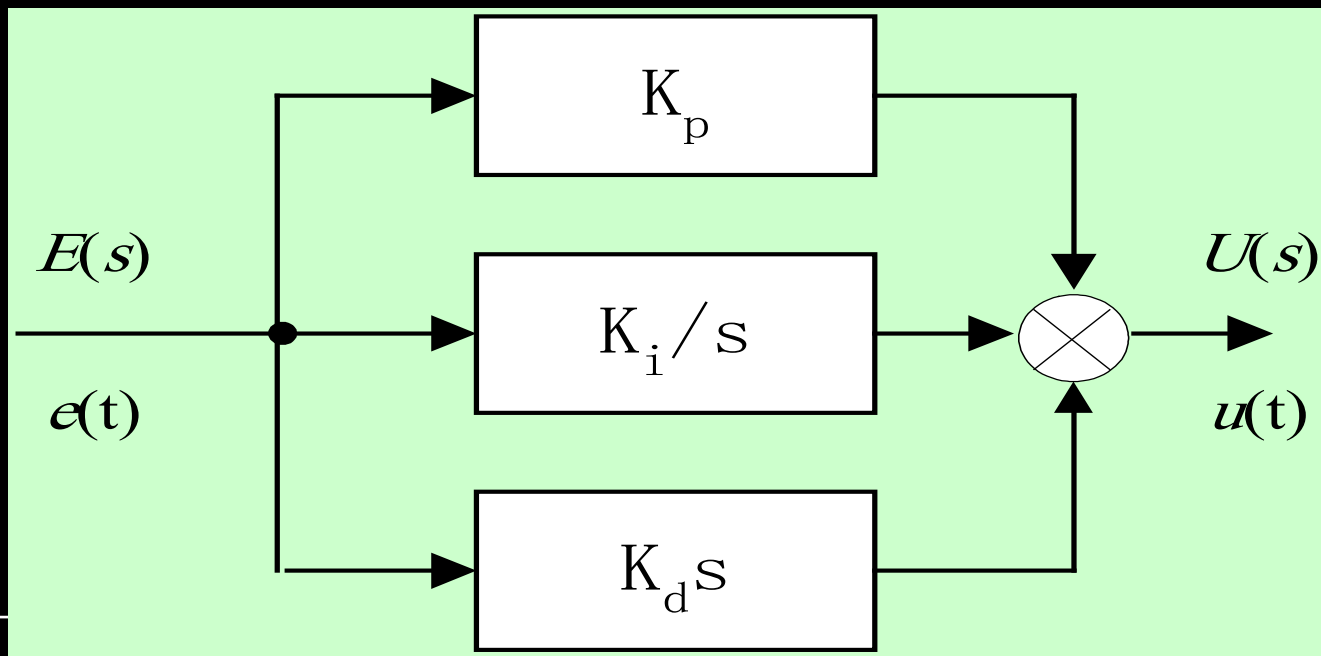
近20年来相继出现一批复杂的，只有计算机才能实现的控制算法。然而，目前即使在过程计算机控制中，PID控制仍然是应用最广泛的控制算法。

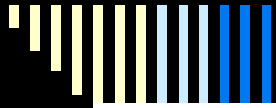
不过，用计算机实现PID控制，就不仅仅是简单地把PID控制规律数字化，而是进一步与计算机的逻辑判断功能结合起来，使PID控制更加灵活多样，更能满足生产过程提出的各种要求。

4.2.1 基本PID控制

PID 控制表示

比例(*Proportional*)—积分(*Integral*)—微分(*Differential*)控制。设 PID 调节器如图所示。






$$U(s) = K_p E(s) + K_i \frac{E(s)}{s} + K_d s E(s)$$

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right]$$

$$D(s) = \frac{U(s)}{E(s)} = K_p + K_i \frac{1}{s} + K_d s$$


$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right]$$

上式中，

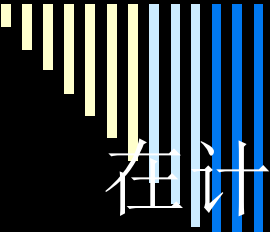
T_i 积分时间常数

T_d 微分时间常数

K_p 比例系数

$K_i = K_p / T_i$ 积分系数

$K_d = K_p * T_d$ 微分系数



在计算机控制系统中使用的是数字PID 调节器，就是对 $u(t)$ 式子离散化，离散化时，令：

$$\left. \begin{aligned} u(t) &\approx u(k) \\ e(t) &\approx e(k) \\ \int_0^t e(t) dt &\approx T \sum_{j=0}^k e(j) \\ \frac{de(t)}{dt} &\approx \frac{e(k) - e(k-1)}{T} \end{aligned} \right\}$$

上式中： T 采样周期 K 为采样序号

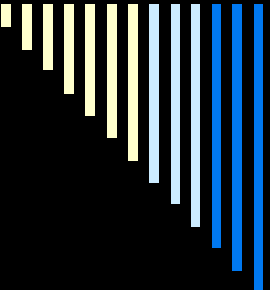
显然，上述离散化过程中，采样周期 T 必须足够短，才能保证有足够的精度。



位置式数字PID控制系统

$$u(k) = K_p \left[e(k) + \frac{T}{T_i} \sum_{j=0}^k e(j) + \frac{T_d}{T} [e(k) - e(k-1)] \right]$$

位置式PID数字调节器 的输出 $u(k)$ 是全量输出，是**执行机构所应达到的位置(如阀门的开度)**，计算机的故障有可能使 $u(k)$ 作大幅度的变化，这种情况往往是生产实践中不允许的，而且有些场合可能会造成严重的事故。



位置式PID数字调节器 除本身安全性、相对控制精度外，在PID位置控制算式中存在累加，不仅要占用许多存储单元，也不利于编写程序及执行速度。



增量式数字 PID 控制系统

$$\Delta u(k) = u(k) - u(k-1) = \\ K_p[e(k) - e(k-1)] + K_i e(k) + \\ K_d[e(k) - 2e(k-1) + e(k-2)]$$

或者

$$\Delta u(k) = K_p \Delta e(k) - K_i e(k) + \\ K_d[\Delta e(k) - \Delta e(k-1)]$$

式中

$$\Delta e(k) = e(k) - e(k-1) \\ \Delta e(k-1) = e(k-1) - e(k-2)$$



上式可整理为

$$\Delta u(k) = Ae(k) + Be(k-1) + Ce(k-2)$$

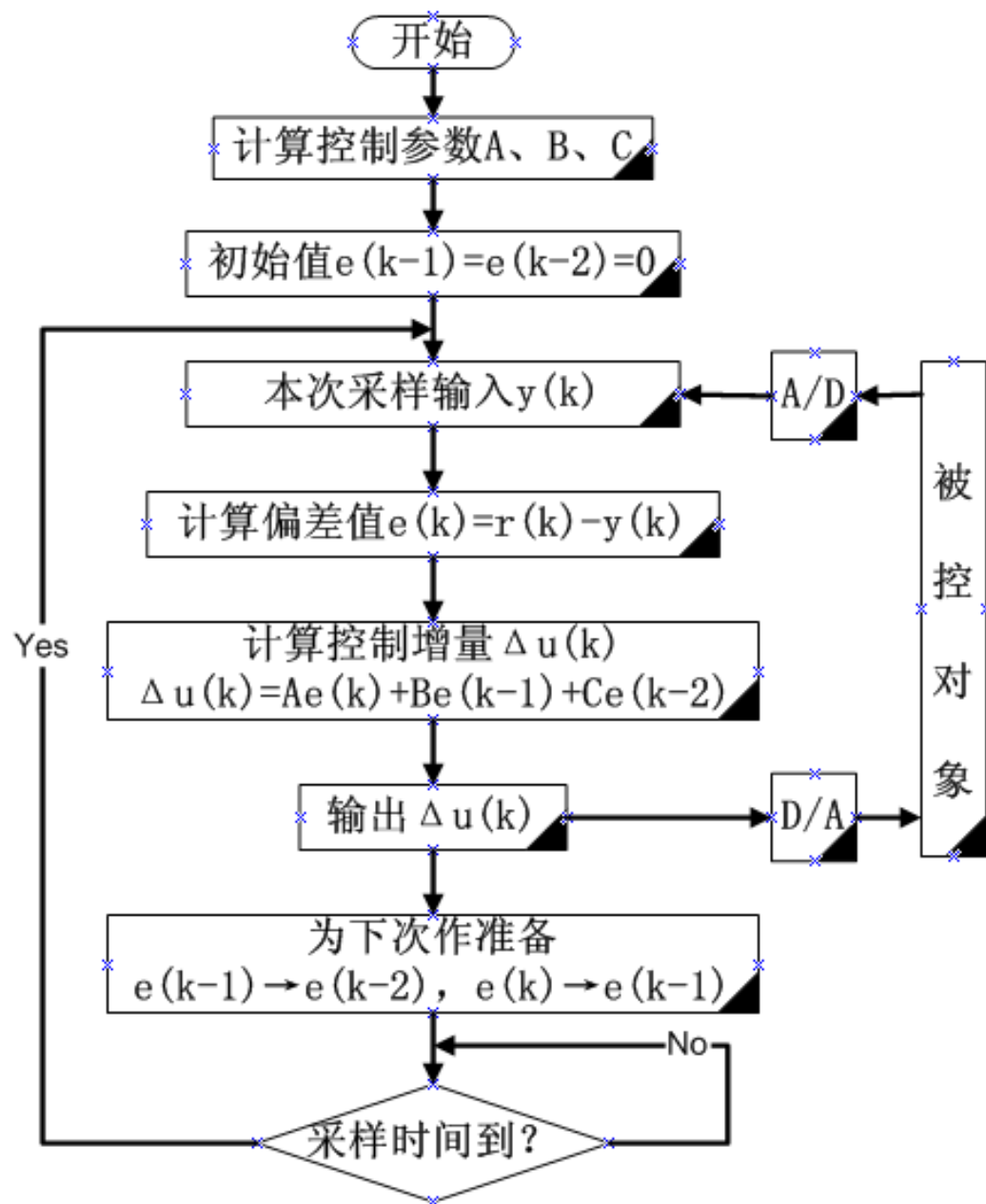
其中：A、B、C均为“常数”

$$A = K_p + K_I + K_D = K_p \left(1 + \frac{T}{T_I} + \frac{T_D}{T}\right)$$

$$B = -(K_p + 2K_D) = -K_p \left(1 + \frac{2T_D}{T}\right)$$

$$C = K_D = K_p \frac{T_D}{T}$$

增量式PID 控制 算法 程序 框图





增量式数字 PID 控制系统的执行机构是步进电机。

增量式算法和位置式算法本质上无大的差别，增量式算法虽然只是算法上的一点改动，却带来了不少的优点。

- ① 计算机(数字调节器)只输出增量，计算机误动作时造成的影响比较小。
- ② 手动—自动切换的冲击小。
- ③ 算式中不需要作累加，累加误差较小。



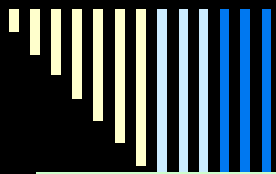
4.2.2 数字PID控制的改进

数字PID控制是应用最普遍的一种控制规律，人们在实践中不断总结经验，不断改进，使得PID控制日臻完善。下面介绍几种数字PID的改进算法：

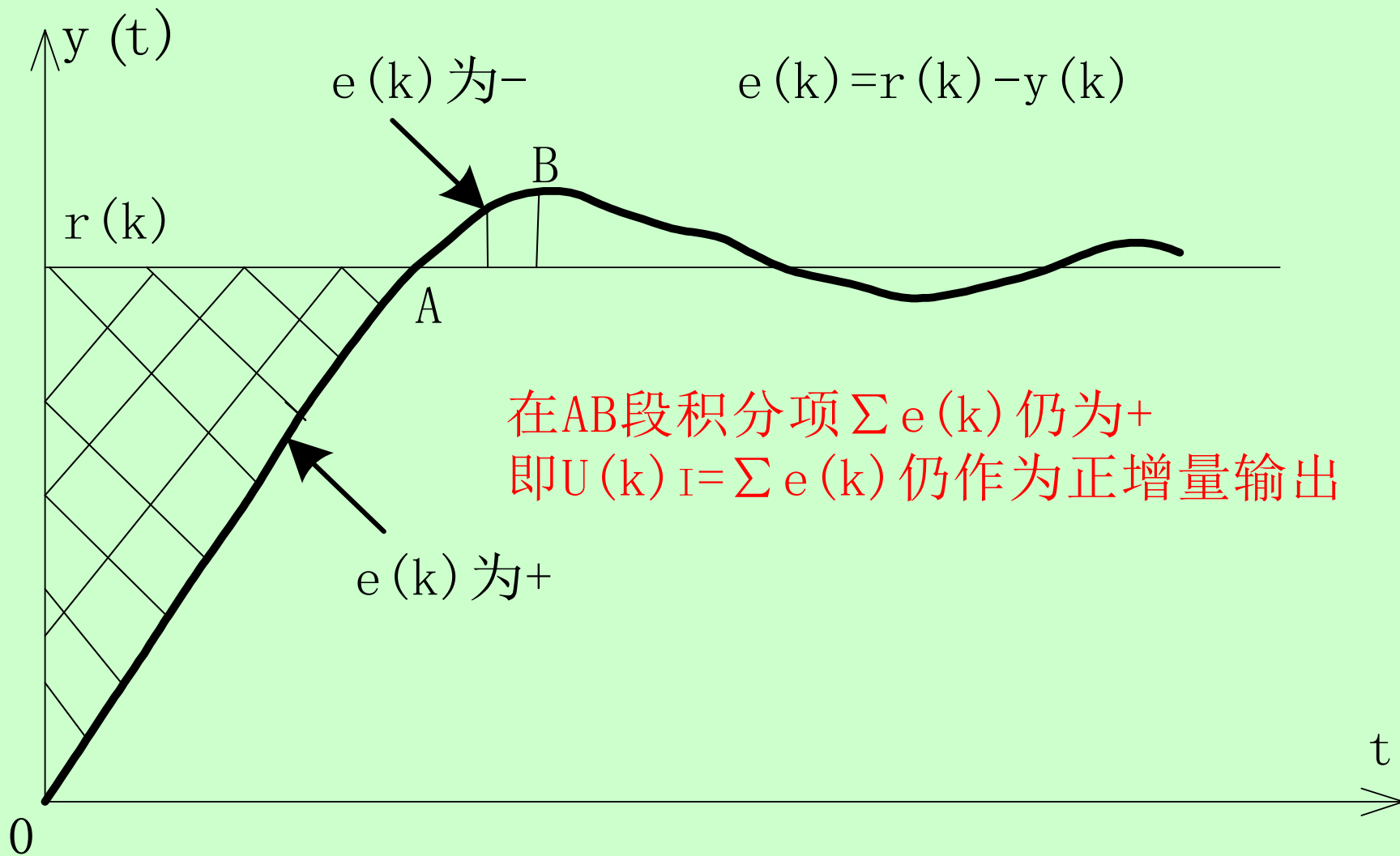
积分分离算法

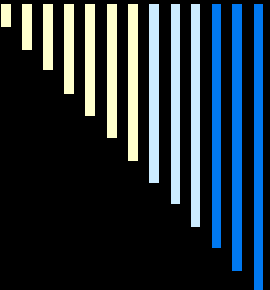
不完全微分算法

带死区的PID算法



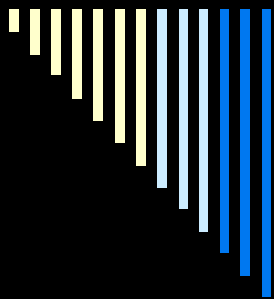
1. 积分分离PID控制算法





系统中加入积分校正以后，会产生过大的超调量，这对某些生产过程是绝对不允许的。

引进积分分离算法，既保持了积分的作用，又减小了超调量，使得控制性能有了较大的改善。



积分分离算法要设置积分分离阈 E_0 ，当 $|e(k)| \leq E_0$ 时，也即偏差值 $|e(k)|$ 比较小时，采用 PID 控制，可保证系统的控制精度。

当 $|e(k)| > E_0$ 时，也即偏差值 $|e(k)|$ 比较大时，采用 PD 控制，可使超调量大幅度降低。

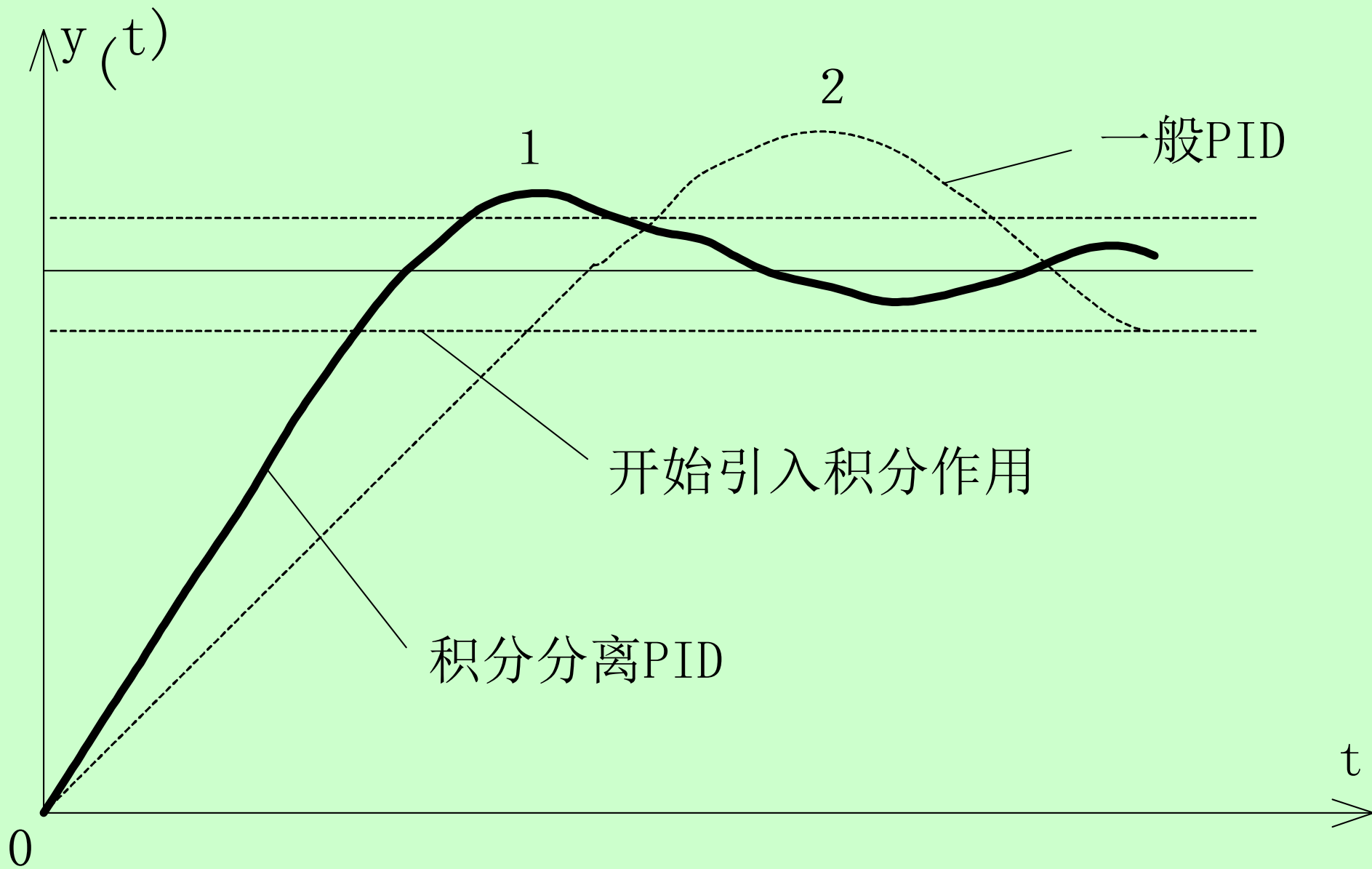


积分分离PID控制算法

$$u(k) = K_p e(k) + K_l K_i \sum_{j=0}^k e(k) + K_d [e(k) - e(k-1)]$$

$$K_l = \begin{cases} 1, & |e(k)| \leq E_0 \\ 0, & |e(k)| > E_0 \end{cases}$$

K_l 称为逻辑系数。



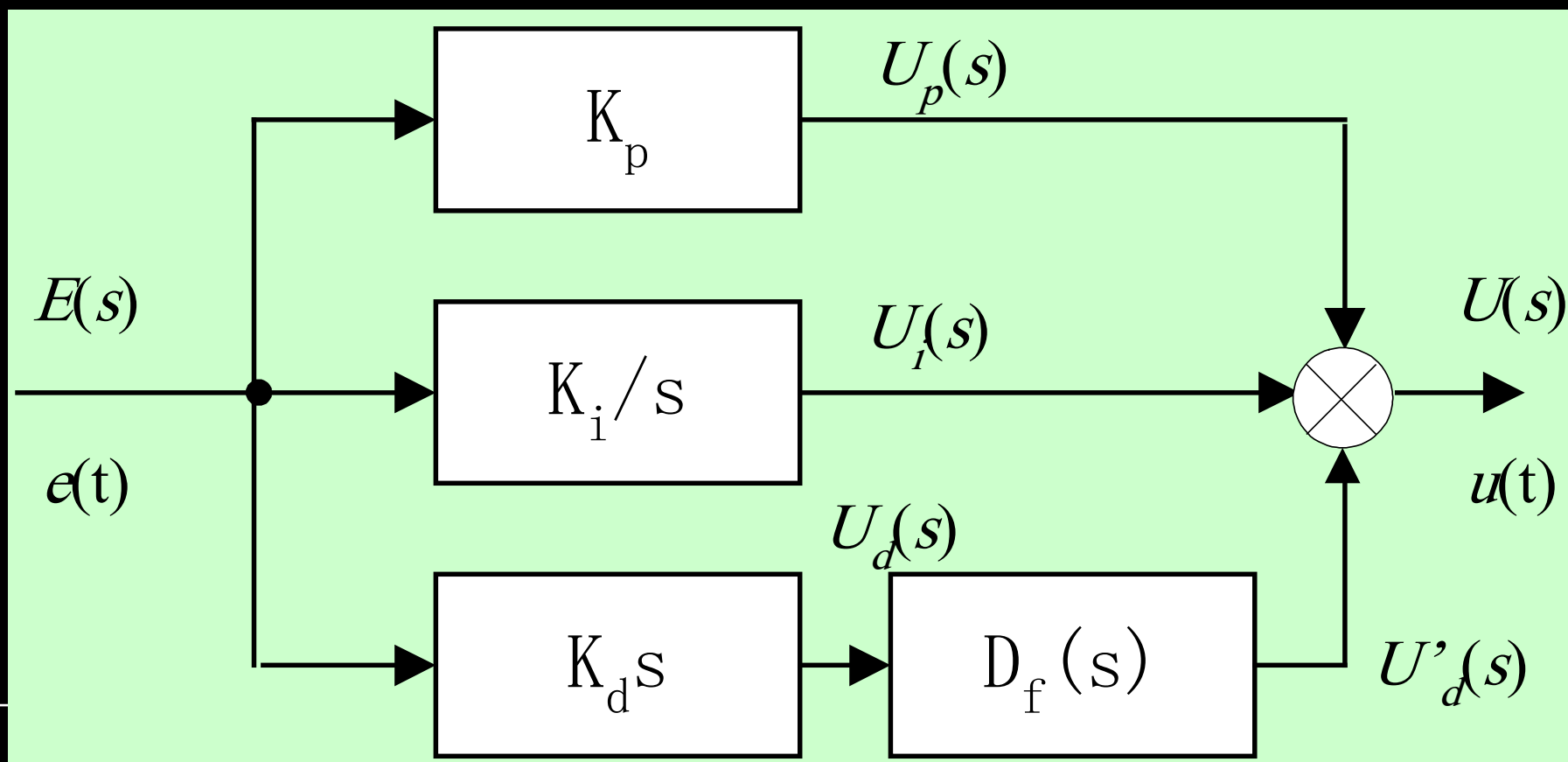


2. 不完全微分PID控制算法

标准的 PID 控制算式，对具有高频扰动的生产过程，微分作用响应过于灵敏，容易引起控制过程振荡，降低调节品质。

尤其是计算机对每个控制回路输出时间是短暂的，而驱动执行器动作又需要一定时间，如果输出较大，在短暂时间内执行器达不到应有的相应开度，会使输出失真。

为了克服这一缺点，同时又要使微分作用有效，可以在 PID 控制的微分输出串联一阶惯性环节，这就组成了不完全微分PID控制器。






串联一阶惯性环节 $D_f(s)$ 的传递函数为

$$D_f(s) = \frac{U'_d(s)}{U_d(s)} = \frac{1}{T_f s + 1}$$

移位则有 $T_f s U'_d(s) + U'_d(s) = U_d(s)$

对上式拉氏逆变换，得

$$T_f \frac{du'_d(t)}{dt} + u'_d(t) = u_d(t)$$



用差分 $\frac{u'_d(k) - u'_d(k-1)}{T}$

代替微分 $\frac{du'_d(t)}{dt}$

并对上式进行离散化，得

$$T_f \frac{u'_d(k) - u'_d(k-1)}{T} + u'_d(k) = u_d(k)$$

整理得：

$$u'_d(k) = \frac{T_f}{T_f + T} u'_d(k-1) + \frac{T}{T_f + T} u_d(k)$$



上式令:

$$a = \frac{T_f}{T_f + T}$$

则可整理为:

$$u'_d(k) = au'_d(k-1) + (1-a)u_d(k)$$

$$u(k) = u_p(k) + u_i(k) + u'_d(k)$$

$$= K_p \left[e(k) + \frac{T}{T_i} \sum_{j=0}^k e(j) \right] + u'_d(k)$$



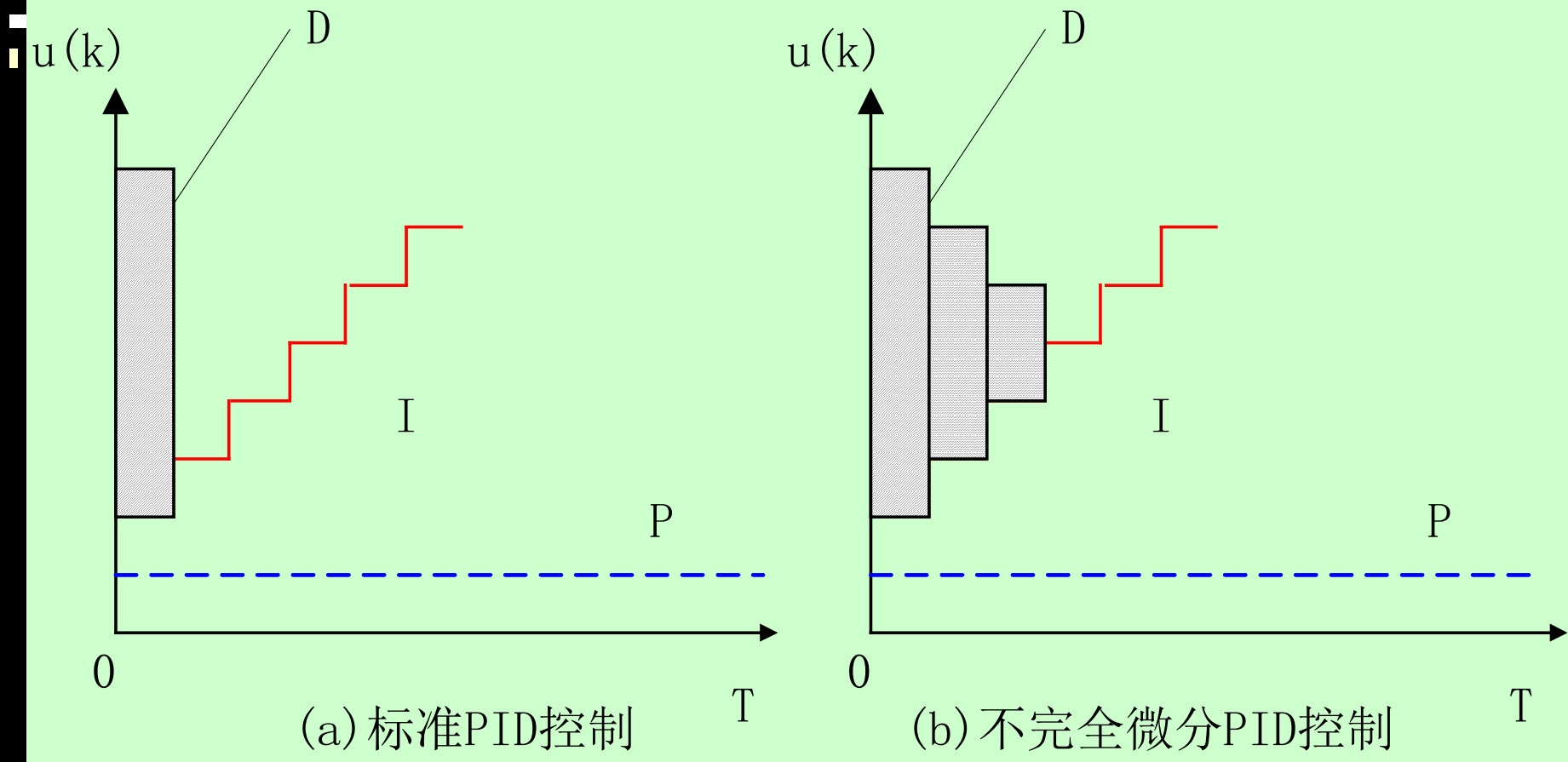
不完全微分PID 增量型控制算式

$$\Delta u'_d(k) = a\Delta u'_d(k-1) + (1-a)\Delta u_d(k)$$

$$\begin{aligned}\Delta u(k) &= \Delta u_p(k) + \Delta u_i(k) + \Delta u'_d(k) \\ &= K_p [e(k) - e(k-1)] + K_i e(k) + \Delta u'_d(k)\end{aligned}$$

其中, $K_i = K_p T / T_i$ 积分系数

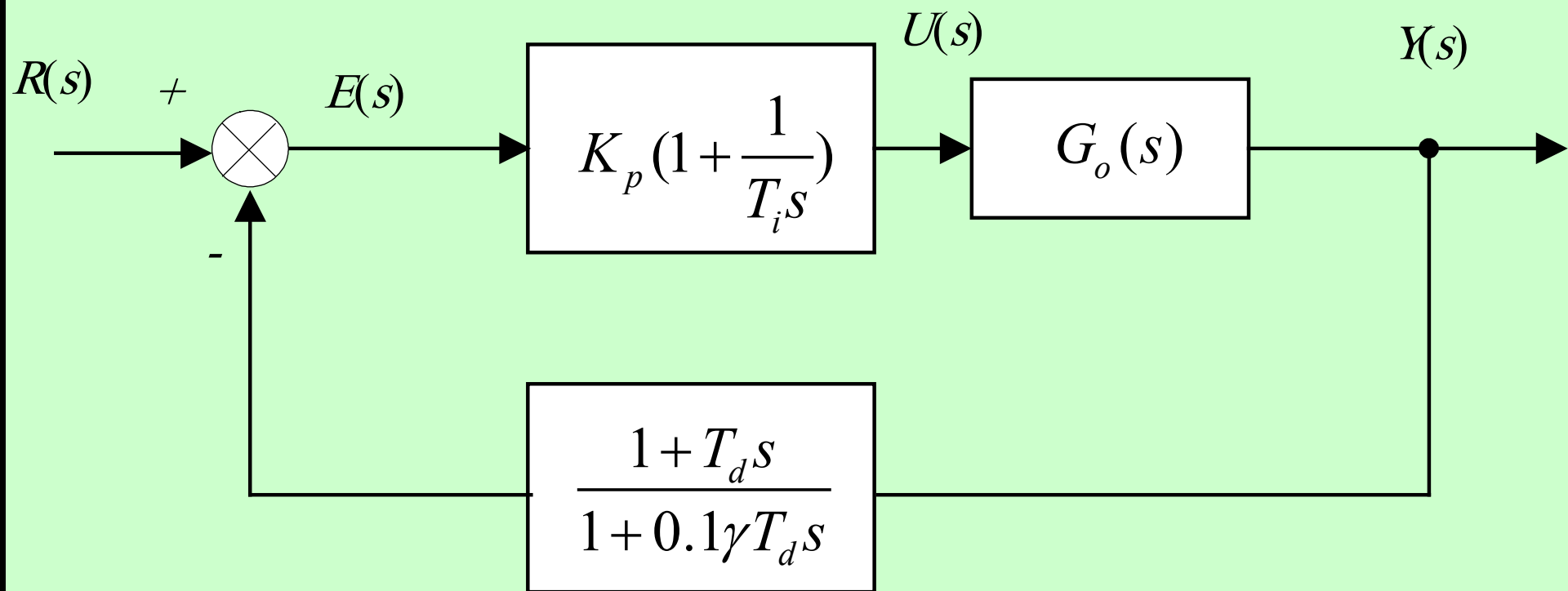
$K_d = K_p T_d / T$ 微分系数

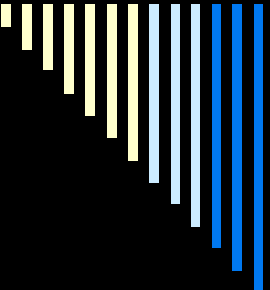


标准PID控制算式中的微分作用只在第一个采样周期内起作用，而且作用很强。而**不完全微分PID控制算式**的输出在较长时间内仍有微分作用，因此可获得较好的控制效果。

3. 微分先行PID控制算式

为了避免给定值频繁升降给控制系统带来冲击，如超调量过大，调节阀动作剧烈，可采用如图所示的**微分先行PID**控制方案。



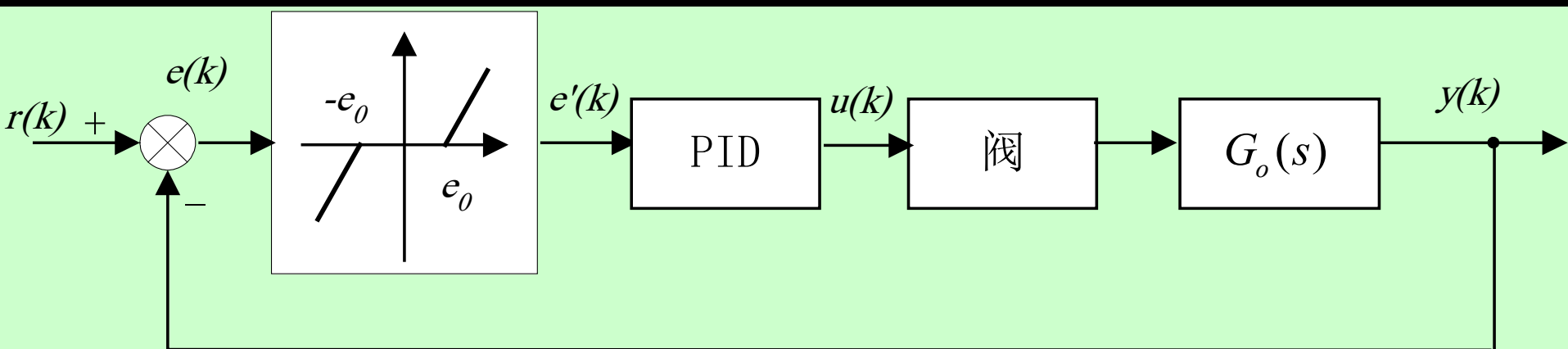


微分先行PID控制算式 和标准PID控制不同之处在于，只对被控量 $y(t)$ 微分，不对偏差 $e(t)$ 微分，也就是说对给定值 $r(t)$ 无微分作用。

该算法对给定值频繁升降的系统无疑是有效的。图中， γ 为微分增益系数。

4. 带死区的PID控制

在要求控制作用少变动的场合，可采用带死区的PID，带死区的PID实际上是非线性系统。



当 $|e(k)| \leq e_0$ 时， $e'(k) = 0$;

当 $|e(k)| > e_0$ 时， $e'(k) = e(k)$

数字调节器输出值 $u(k)$ 以 $e'(k)$ 的 PID 运算结果输出。



Question

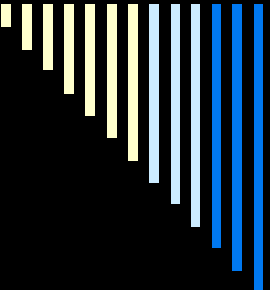
- 1.数字滤波有哪几种方式，分别在什么场合下使用？
- 2.标度变换有什么意义？
- 3.PID控制算法有哪两种实现形式？分别有什么特点？
- 4.什么是积分分离PID算法？算法的优点是什么？
- 5.什么是微分先行PID算法？在何种场合下使用？
- 6.什么是不完全微分PID算法，在何种场合下使用？
- 7.带死区的PID算法如何实现？有什么作用？



4.2.3 数字 PID 控制器的工程实现

前面两节分别讨论了数字 PID 控制算法及其改进措施，为编制 PID 程序提供了算式，这是数字PID控制器的核心问题。

如果计算机仅实现此算式，并不能满足实际控制的需要，还必须考虑其他工程实际问题，才能使 PID 控制程序具有通用性和实用价值。



众所周知，模拟 PID 调节器是一台**硬设备**。与之相比，计算机中的数字PID 控制却是**软设备**，也就是说，它是由一段 PID 程序来实现的。

一台模拟调节器只能控制一个回路，而一段PID控制程序却可以作为一台计算机所控制的所有回路的公共子程序。

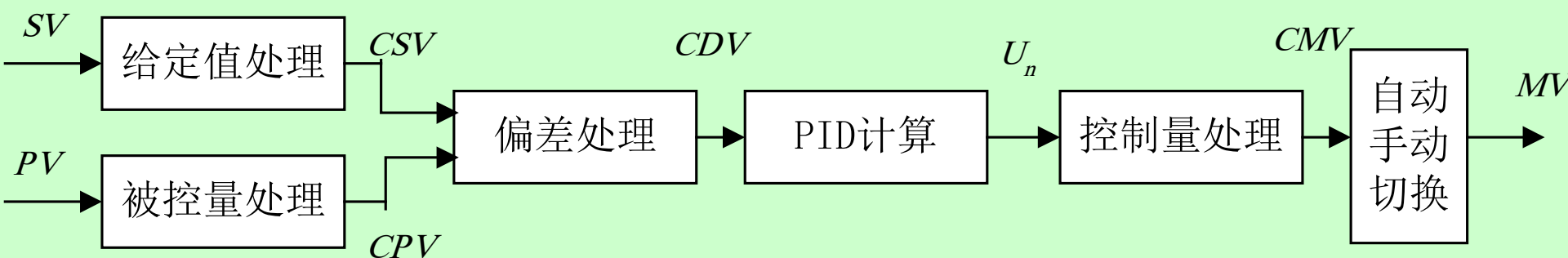


所不同的只是各个控制回路提供的原始数据不一样，输入输出通道也不一样。

为此，必须给每个 PID 控制回路提供一段内存数据区(亦称线性表)，以便存放各种信息参数。

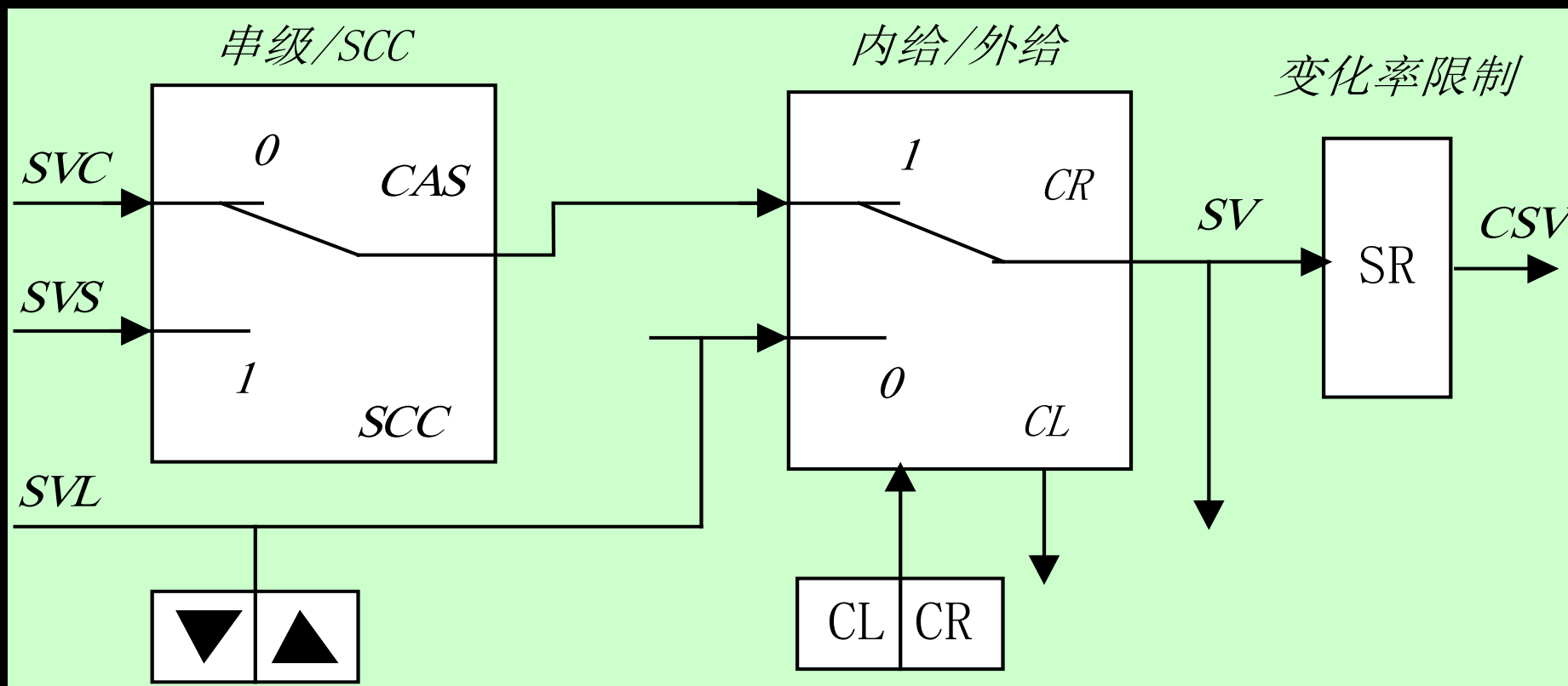
既然 PID 控制程序是公共子程序，就应该具有通用性和工程实用价值。在设计PID控制程序时，必须考虑各种工程实际情况，并含有多种功能，以便用户选择。

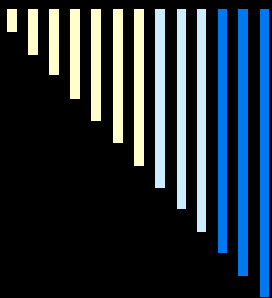
本节将PID控制程序分成如书图4.6所示的六部分来讨论。此外，为了便于数字PID 控制器的操作显示，通常给每个 PID 控制模块配置一个回路操作显示器，它与模拟调节器的面板操作显示相类似。



1. 给定值处理

给定值处理包括选择给定值 **SV** 和给定值变化率限制 **SR** 两部分，如图所示。





通过选择软开关 **CL / CR**，可以构成内给定状态或外给定状态；通过选择软开关 **CAS / SCC**，可以构成串级控制或 **SCC** 控制。



(1) 内给定状态

当软开关 CL / CR 切向 **CL 位置** 时，选择操作员设置的给定值 **SVL**。这时，系统处于**单回路控制的内给定状态**，利用给定值按键可以改变给定值。

(2)外给定状态

当软开关 CL / CR切向 **CR** 位置时，给定值来自上位计算机、主回路或运算模块。这时，系统处于外给定状态。在此状态下，可以实现以下两种控制方式：

- ① **SCC 控制**：当软开关CAS / SCC切向 **SCC** 位置时，接收来自上位计算机的给定值 **SVS** 以便实现两级计算机控制。
- ② **串级控制**：当软开关 CAS / SCC切向 **CAS** 位置时，给定值 **SVC** 来自主调节模块，实现串级控制。



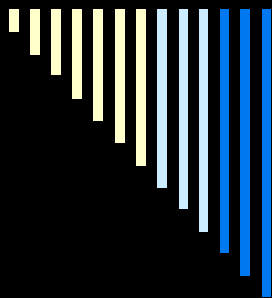
(3)给定值变化率限制

为了减少给定值突变对控制系统的扰动，防止比例、微分饱和，以实现平稳控制，需要对给定值的变化率 SR 加以限制。变化率的选取要适中，过小会使响应变慢，过大则达不到限制的目的。



综上所述，在给定值处理图中，共具有三个输入量 (SVL, SVC, SVS)、两个输出量 (SV, CSV)、两个开关量 (CL / CR, CAS / SCC)、一个变化率 (SR)。

为了便于 PID 控制程序调用这些量，需要给每个 PID 控制模块提供一段内存数据区，并采用相对地址进行调用和存放，如下图所示。

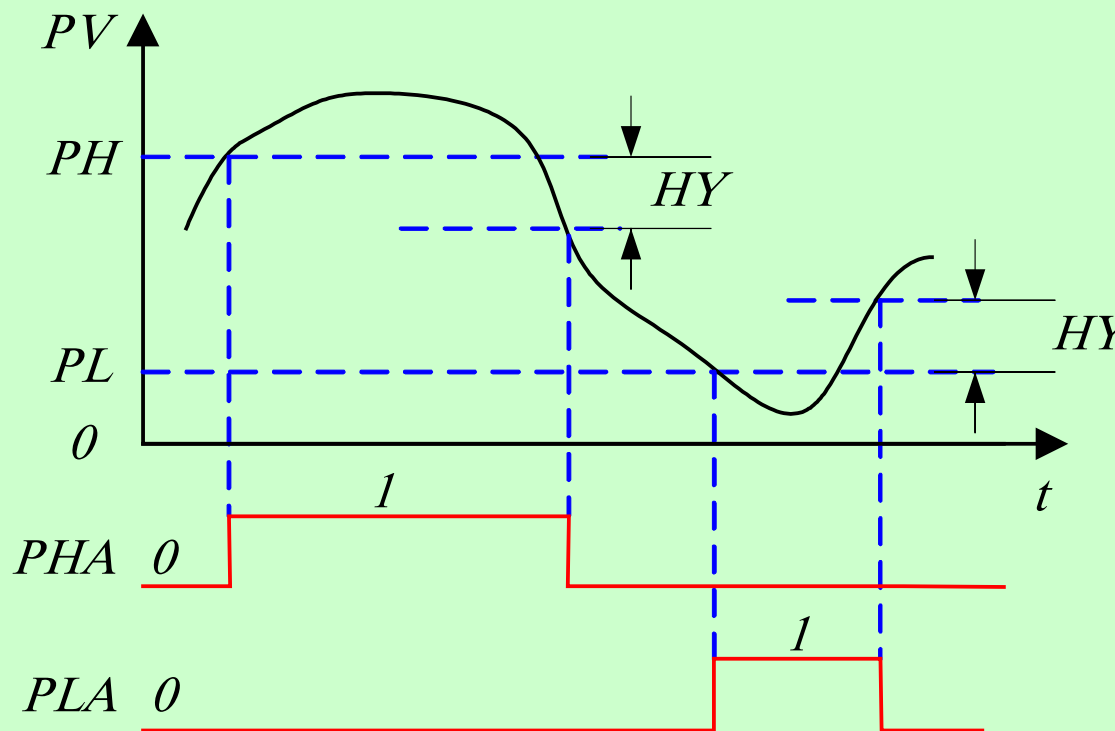
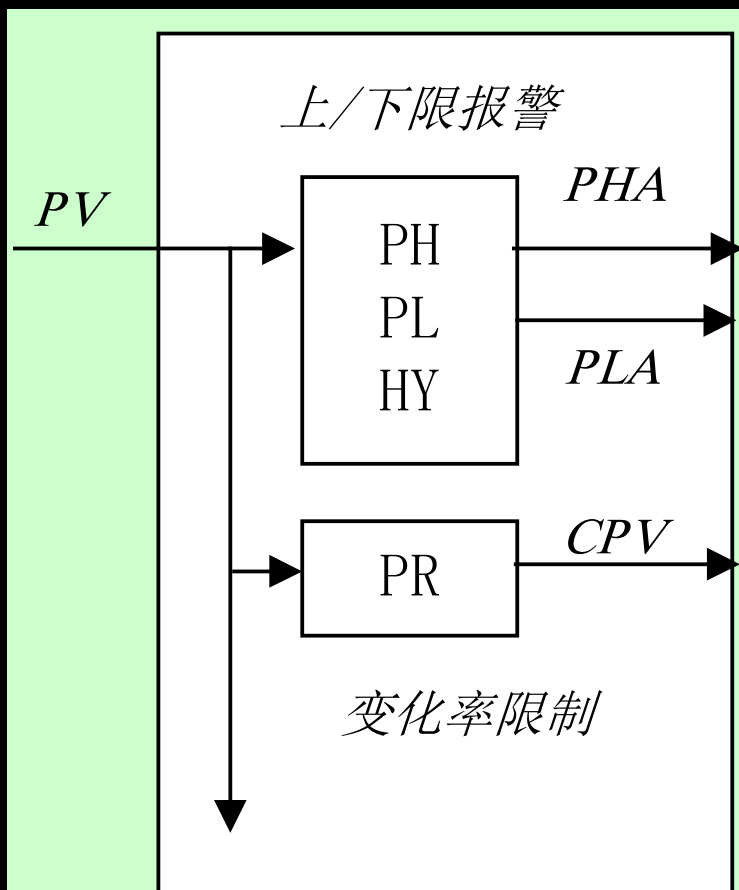


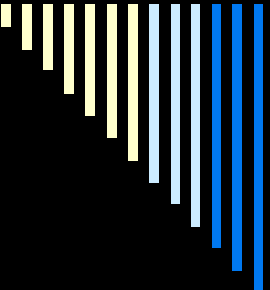
IX+00H	SVL
01	
02	SVC
03	
04	SVS
05	
06	SV
07	
08	CSV
09	
0A	SR
0B	
0C	CL/CR
IX+0DH	CAS/SCC

图4. 8 给定值处理数据区

2. 被控量处理

为了安全运行，需要对被控量 PV 进行上限或下限报警处理，其原理如图所示。

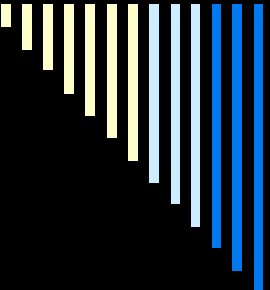




当 $PV > PH$ (上限报警值) 时，
则上限报警状态 (PHA) 为1；

当 $PV < PL$ (下限报警值) 时，则
下限报警状态 (PLA) 为1；

当出现上、下限报警状态(PHA, PLA)时，它们通过驱动电路发出声或光，以便提醒操作员注意。为了不使PHA / PLA的状态频繁改变，可以设置一定的报警死区 (HY)。

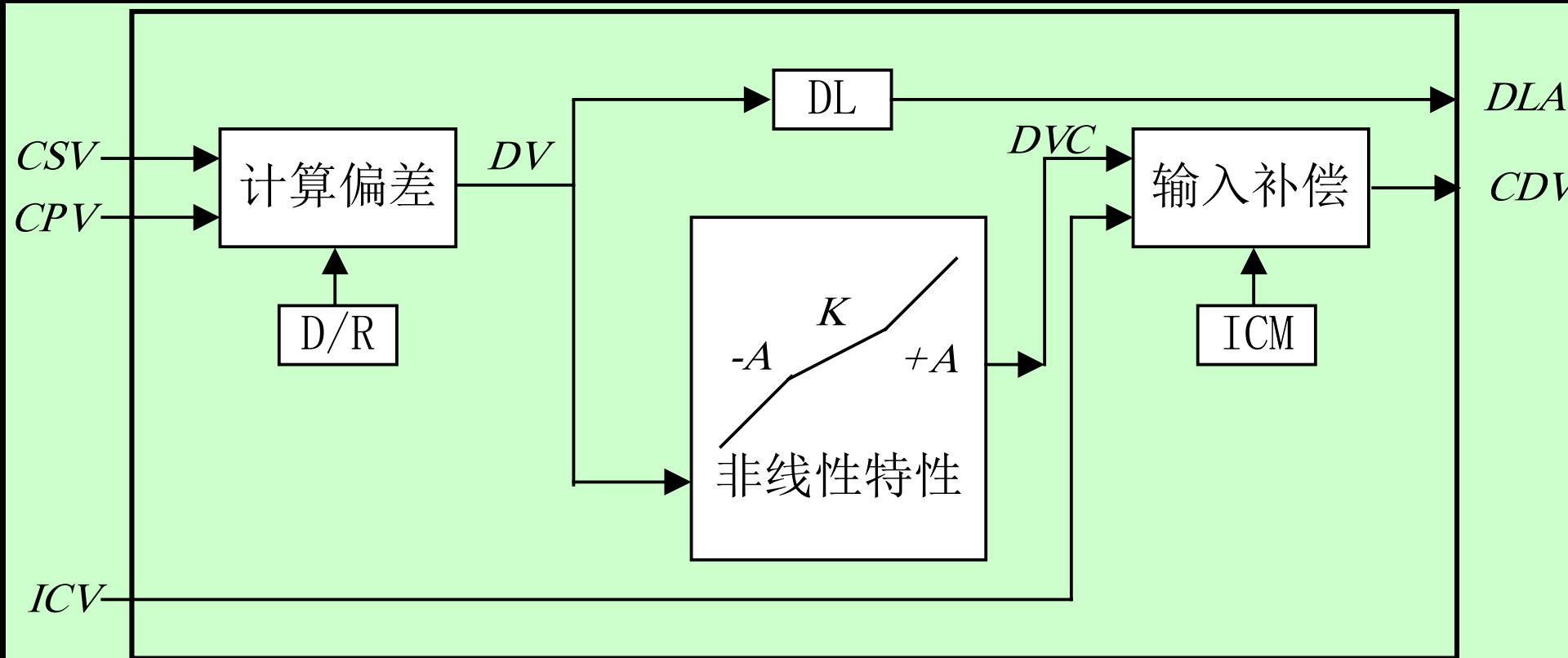


为了实现平稳控制，需要对参与控制的被控量的**变化率 PR** 加以限制。变化率的选取要适中，过小会使响应变慢，过大则达不到限制的目的。

被控量处理数据区存放一个输入量 **PV**，三个输出量 **PHA**、**PLA** 和 **CPV**，四个参数 **PH**、**PL**、**HY** 和 **PR**。数据存放方式类似于图4.8。

3. 偏差处理

偏差处理分为计算偏差、偏差报警、非线性特性和输入补偿四部分，如图所示。





(1) 计算偏差

根据正 / 反作用方式 (D / R) 计算偏差DV，即：

当 $D / R = 0$ ，代表正作用：

此时偏差 $DV+ = CPV - CSV$ ；

当 $D / R = 1$ ，代表反作用：

此时偏差 $DV- = CSV - CPV$ 。

(PID中的正反作用是针对：“现场值的变化趋势”与“PID控制输出值变化趋势”之间的关系。

例一：用PID调节器控制恒温炉的炉温。这是一个典型的反作用调节。如果炉温下降（现场值下降趋势），我们PID控制输出要增加输出量，来增加功率。是增加的趋势。两个趋势相反。所以，是反作用。

例二：用PID调节器来控制冰库的温度，如果冰库的温度在上升，我们PID调节的输出也要增加，增加制冷的功率。两个趋势是相同的。这就是正作用。

根据现场控制要求，来选择PID调节器的正反作用。)

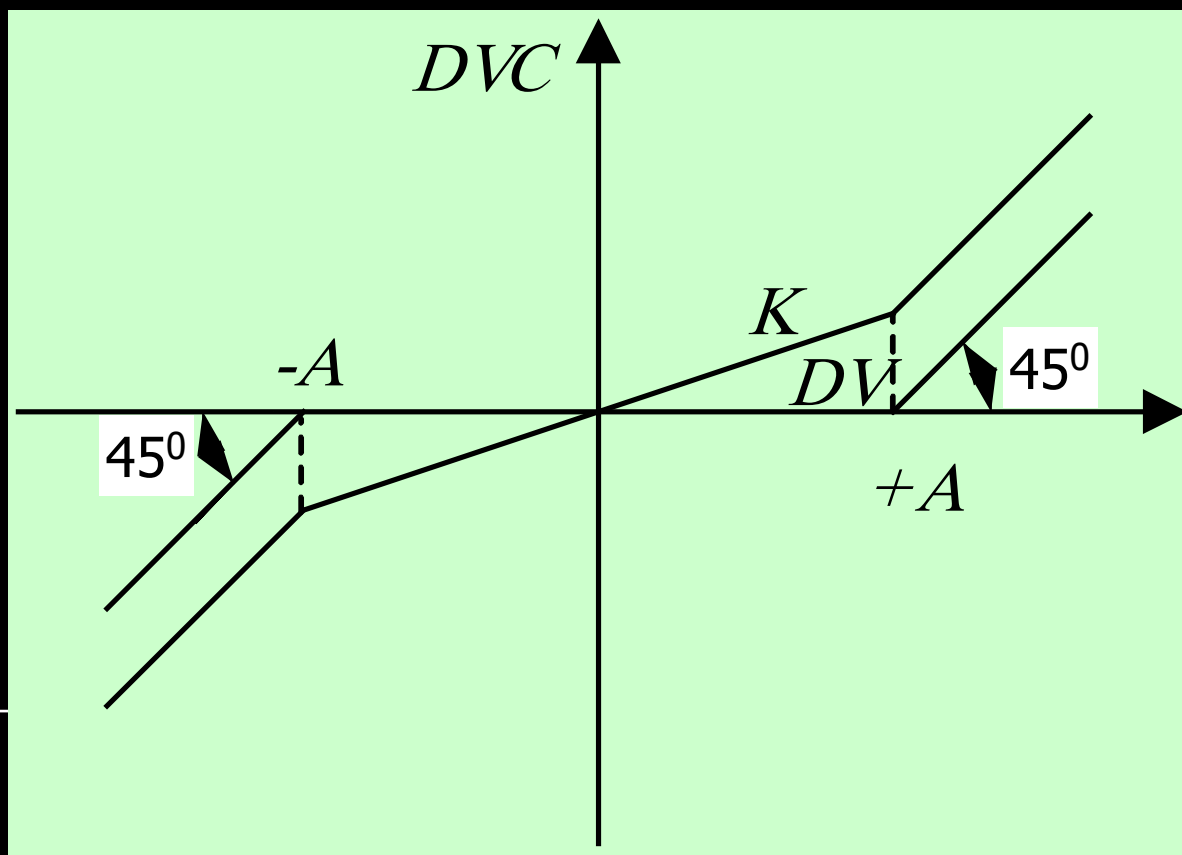


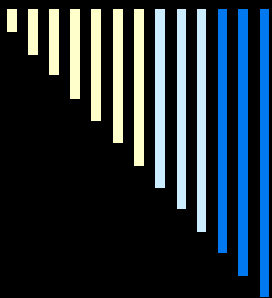
(2) 偏差报警

对于控制要求较高的对象，不仅要设置被控量 PV 的上、下限报警，而且要设置偏差报警。当偏差绝对值 $|DV| > DL$ 时，则偏差报警状态 DLA 为 1。

(3) 非线性特性

为了实现非线性 PID 控制或带死区的PID控制，设置了非线性区 $-A$ 至 $+A$ 和非线性增益 K ，如图所示。





上图中，当增益 $K = 0$ 时，则为带死区的 PID 控制；

当 $0 < K < 1$ 时，则为非线性 PID 控制；

当 $K = 1$ 时，则为正常的 PID 控制。



(4) 输入补偿

根据输入补偿方式 **ICM** 的状态，决定偏差 **DVC** 与输入补偿量 **ICV** 之间的关系，即：

ICM = 0，无补偿，**CDV = DVC**

ICM = 1，加补偿，**CDV = DVC + ICV**

ICM = 2，减补偿，**CDV = DVC - ICV**

ICM = 3，置换补偿，**CDV = ICV**

利用加、减输入补偿，可以分别实现前馈控制
和纯迟延补偿控制。

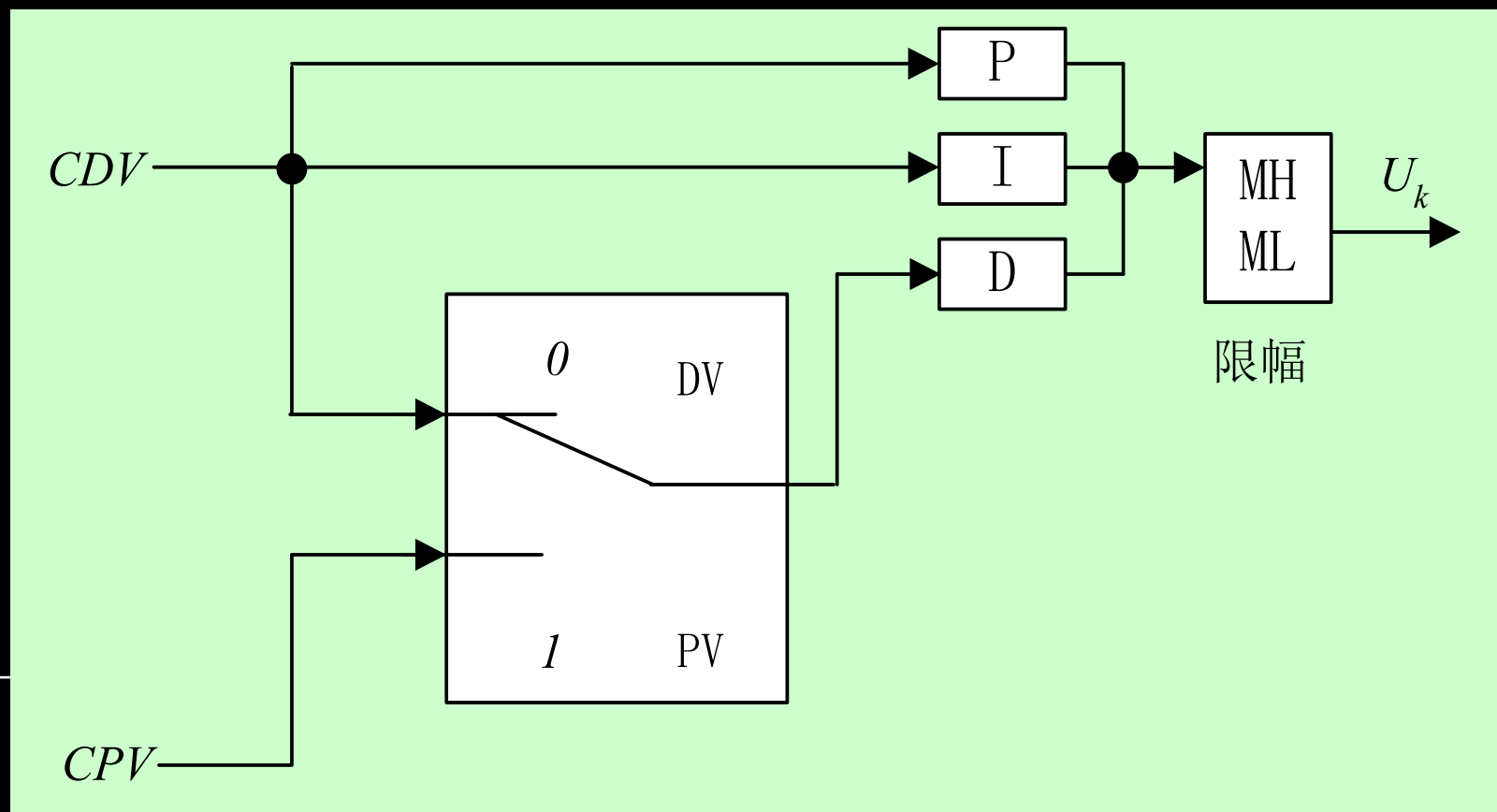


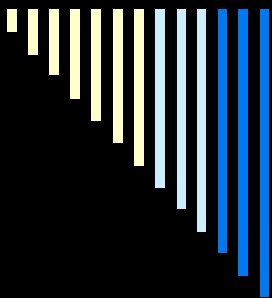
偏差处理数据区共存放

- 一个输入补偿量 ICV ;
- 两个输出量 DLA 和 CDV ;
- 两个状态量 D / R 和 ICM ;
- 四个参数 DL 、 $-A$ 、 $+A$ 和 K 。

4. PID计算

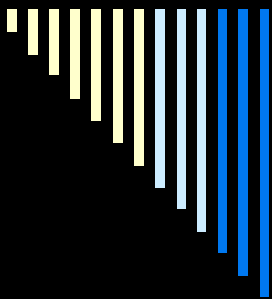
在自动状态下，需要进行PID计算，即按照PID控制的各种差分方程，计算控制量 U_k ，并进行上、下限限幅处理，如图所示。





当软开关 DV / PV 切向 **DV** 位置时，选用偏差微分方式，即**常规PID**控制；

当软开关 DV / PV切向 **PV** 位置时，则选用测量值（即被控量）微分方式，即**微分先行PID**控制。

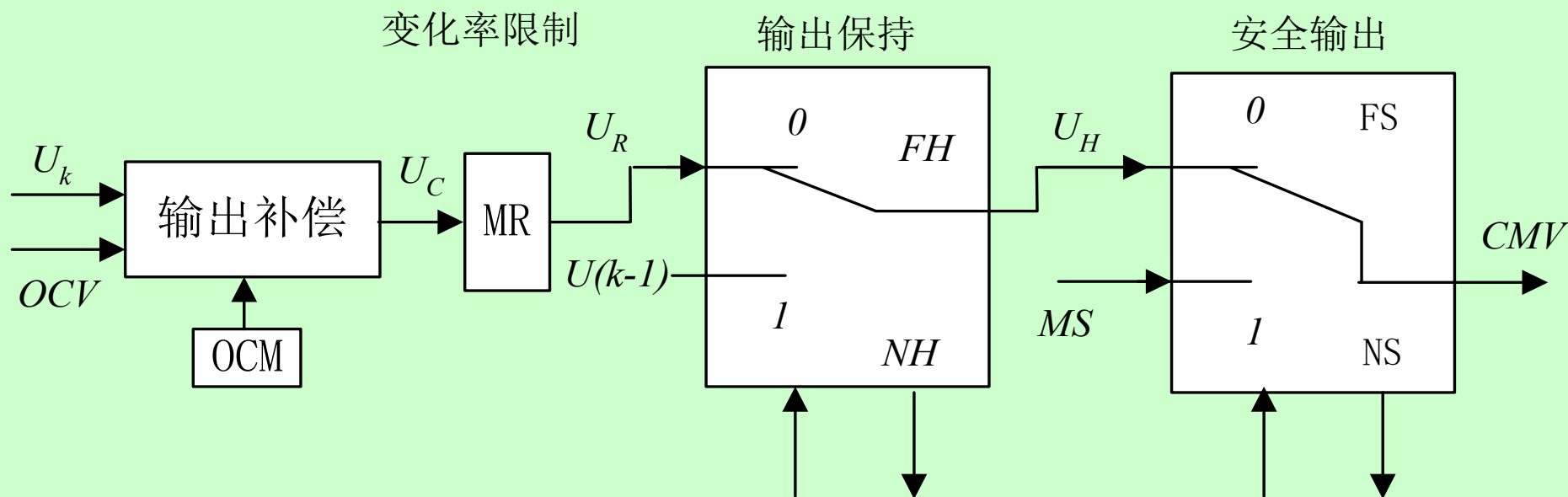


在PID计算数据区，不仅要存放PID参数(K_i 、 T_i 、 T_d)和采样控制周期 T ，还要存放微分方式 DV / PV 、积分分离值 E_0 、控制量上限限值 MH 和下限限值 ML 以及控制量 U_k 。

为了进行递推运算，还应保存历史数据的 $e(k-1)$ 、 $e(k-2)$ 和 $u(k-1)$ 。

5. 控制量处理

一般情况下，在输出控制量 U_k 以前，还应经过下图所示的各项处理和判断，以便扩展控制功能，实现安全平稳操作。





(1) 输出补偿

根据输出补偿方式 OCM 的状态，决定控制量 U_k 与输出补偿量 OCV 之间的关系，即：

OCM = 0，无补偿， $U_c = U_n$

OCM = 1，加补偿， $U_c = U_n + OCV$

OCM = 2，减补偿， $U_c = U_n - OCV$

OCM = 3，置换补偿， $U_c = OCV$

利用输出和输入补偿，可以扩大 PID 控制功能，以便组成复杂的 PID 控制系统。



(2) 变化率限制

为了实现平稳操作，需要对控制量的变化率 MR 加以限制。变化率的选取要适中，过小会使操作缓慢，过大则达不到限制的目的。



(3) 输出保持

当软开关FH / NH切向NH位置时，现时刻的控制量 $U(k)$ 等于前一时刻的控制量 $U(k - 1)$ ，也就是说，输出控制量保持不变。当软开关FH / NH切向FH位置时，又恢复正常输出方式。软开关FH / NH状态一般来自系统安全报警开关。



(4) 安全输出

当软开关 FS / NS 切向 **NS** 位置时，现时刻的控制量等于预置的安全输出量**MS**。当软开关 FS / NS 切向 **FS** 位置时，又恢复正常输出方式。软开关 FS / NS 状态一般来自系统安全报警开关。

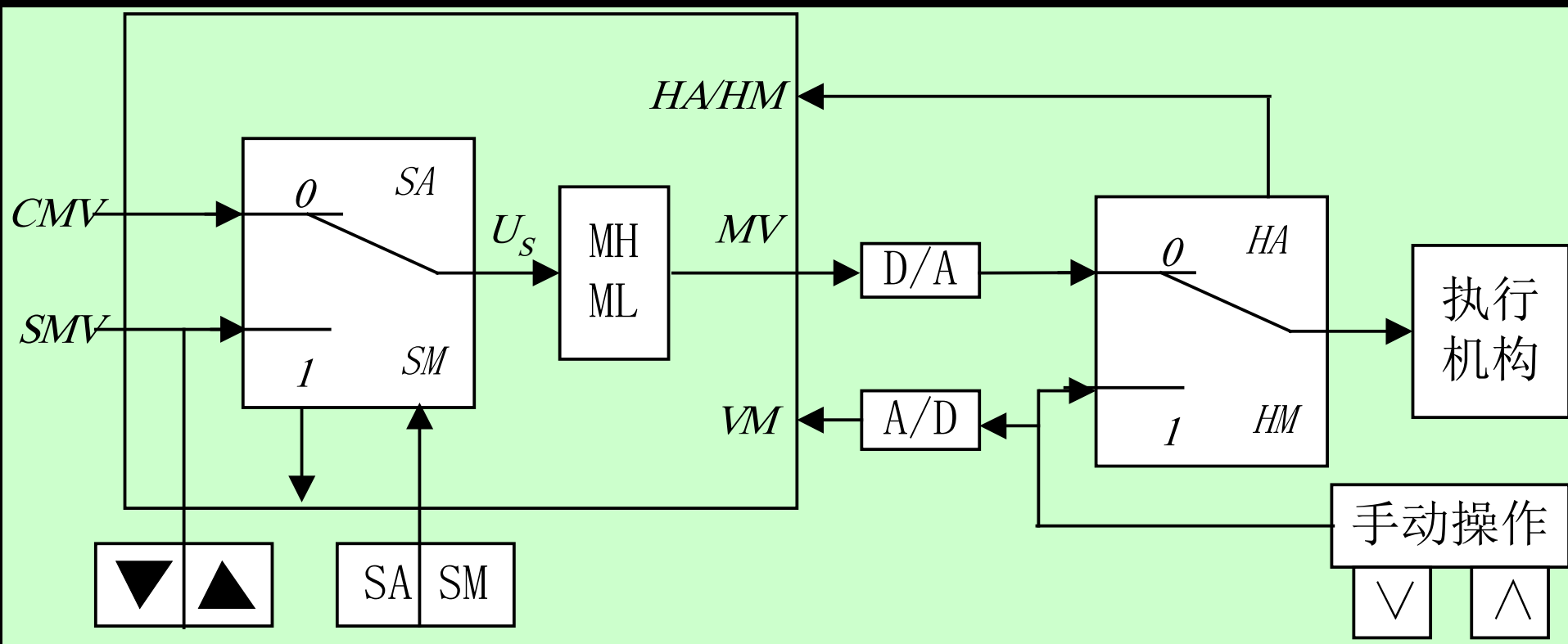


控制量处理数据区需要存放

- 输出补偿量 OCV;
- 补偿方式 OCM;
- 变化率限制值 MR;
- 软开关 FH / NH 和 FS / NS;
- 安全输出量 MS;
- 控制量 CMV。

6. 自动手动切换

在正常运行时，系统处于自动状态；而在调试阶段或出现故障时，系统处于手动状态。下图为自动 / 手动切换处理框图。





(1) 软自动 / 软手动

当软开关 SA / SM 切向 SA 位置时，系统处于正常的自动状态，称为软自动（SA）；反之，切向 SM 位置时，控制量来自操作键盘或上位计算机，此时系统处于计算机手动状态，称为软手动（SM）。

一般在调试阶段，采用软手动（SM）方式。



(2) 控制量限幅

为了保证执行机构工作在有效范围内，需要对控制量 U_s 进行上、下限限幅处理，使得 $MH \leq MV \leq ML$ ，再经D / A 转换器输出 DC0~10 mA 或DC4~20mA。



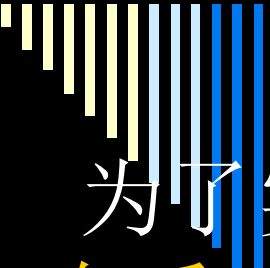
(3) 自动 / 手动

对于一般的过程计算机控制系统，可采用手动操作器作为计算机的后备操作。当切换开关处于HA位置时，控制量MV 通过D / A输出，此时系统处于正常的计算机控制方式，称为自动状态(HA状态)；反之，若切向HM位置，则计算机不再承担控制任务，由运行人员通过手动操作器输出DC0~10mA或DC4~20mA信号，对执行机构进行远方操作，这称为手动状态(HM状态)。

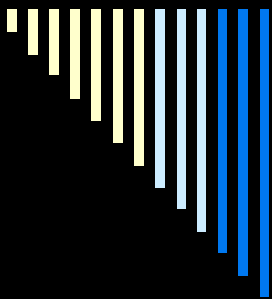


(4) 无平衡无扰动切换

所谓无平衡无扰动切换，是指在
进行手动到自动或自动到手动的切
换之前，无须由人工进行手动输出
控制信号与自动输出控制信号之间的
对位平衡操作，就可以保证切换
时不会对执行机构的现有位置产生
扰动。为此，应采取以下措施。



为了实现从手动到自动的无平衡无扰动切换，在手动(SM或HM)状态下，尽管并不进行PID计算，但应使给定值(CSV)跟踪被控量(CPV)，同时也要把历史数据，如PID算式中的 $e(k-1)$ 和 $e(k-2)$ 清零。还要 $u(k-1)$ 跟踪手动控制量(MV或VM)。这样，一旦切向自动状态(SA或HA)时，由于 $CSV = CPV$ ，因而偏差为0，而 $u(k-1)$ 又等于切换瞬间的手动控制量，这就保证了PID控制量的连续性。当然，这一切需要有相应的硬件电路配合。



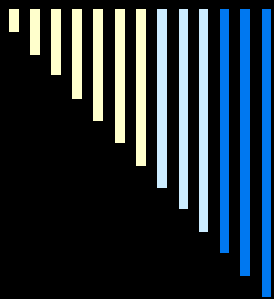
当从自动 (SA 与 HA) 切向软手动(SM) 时，只要计算机应用程序工作正常，就能自动保证无扰动切换。

从输出保持状态或安全输出状态切向正常的自动工作状态时，同样需要进行无扰动切换，为此可采取类似的措施，不再赘述。



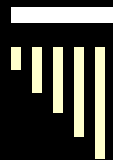
自动手动切换数据区需要存放

- 软手动控制量SMV;
- 软开关SA / SM状态;
- 控制量上限限值MH;
- 控制量下限限值ML;
- 控制量MV;
- 切换开关HA / M状态;
- 手动操作器输出VM。

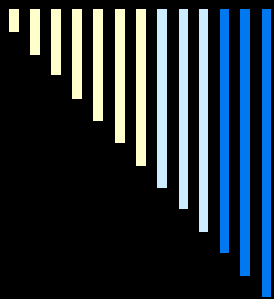


完整的PID控制模块数据区除了上述六部分外，还有被控量量程上限RH和量程下限RL、工程单位代码、采样(控制)周期等，如下图所示。

该数据区是PID控制模块存在的标志，可把它看作是数字PID控制器的实体。只有正确地填写PID数据区后，才能实现PID控制系统。



IX+00H	模块号
IX+01H	量程上限 RH
	量程下限 RL
	工程单位代码
	控制周期 T
IX+07H	给定值处理数据
	被控量处理数据
	偏差处理数据
	PID 计算数据
	控制量处理数据
	自动 / 手动切换数据
	其他
IX+7FH	Lnext



采用上述PID控制模块，不仅可以组成单回路控制系统，而且可以组成串级、前馈、纯迟延补偿(Smith)等复杂控制系统，对于后面两种系统还应增加补偿器运算模块。

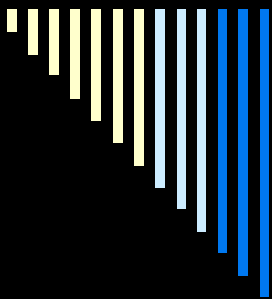
利用PID控制模块和各种功能运算模块的组合，可以组成各种控制系统来满足生产过程控制的要求。



4.2.4 PID参数整定

模拟PID调节器的整定是按照工艺对控制性能的要求，决定调节器的参数 K_p 、 T_i 、 T_d ，这是工程中使用最普遍，最为广大工程技术人员所熟知的。

数字PID调节器参数的整定，除了需要确定 K_p 、 T_i 、 T_d 外，还需要确定系统的采样周期 T 。



生产过程（对象）通常有较大的惯性时间常数，而大多数情况，采样周期与对象的时间常数相比要小得多，所以数字调节器参数的整定可以仿照模拟PID调节器参数 K_p 、 T_i 、 T_d 整定的各种方法。



1. PID调节器参数对控制性能的影响

K_p 、 T_i 、 T_d 对系统控制性能的作用。



(1) 比例控制参数 K_p 对系统性能的影响

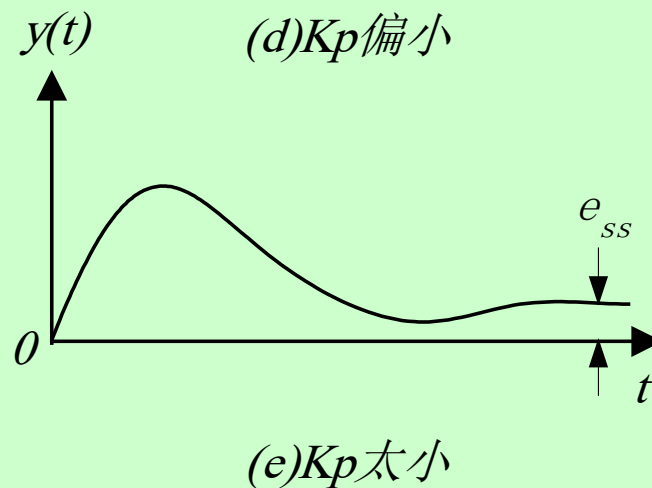
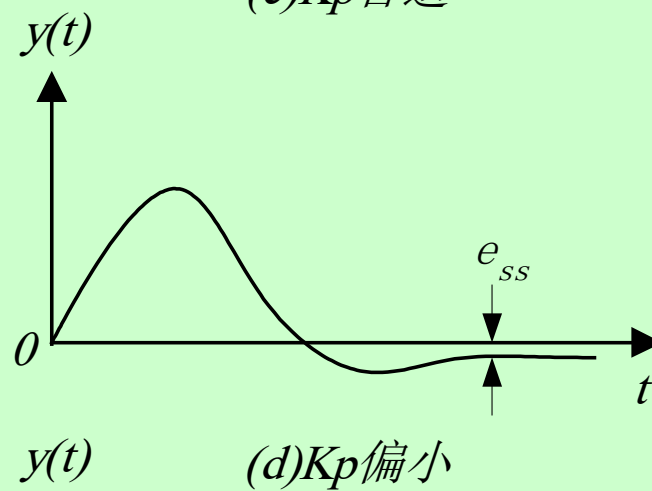
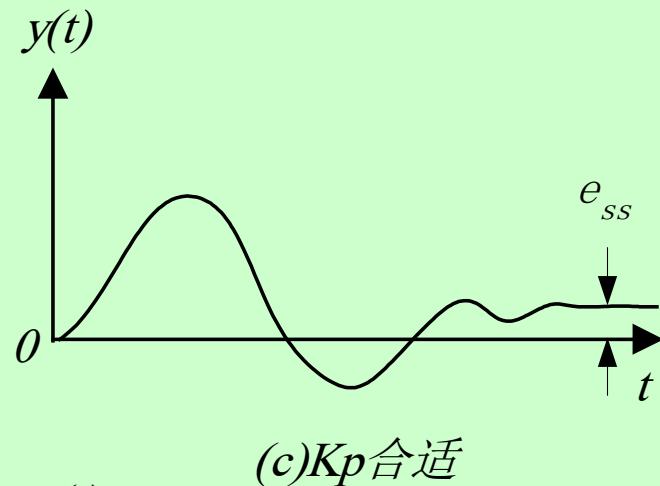
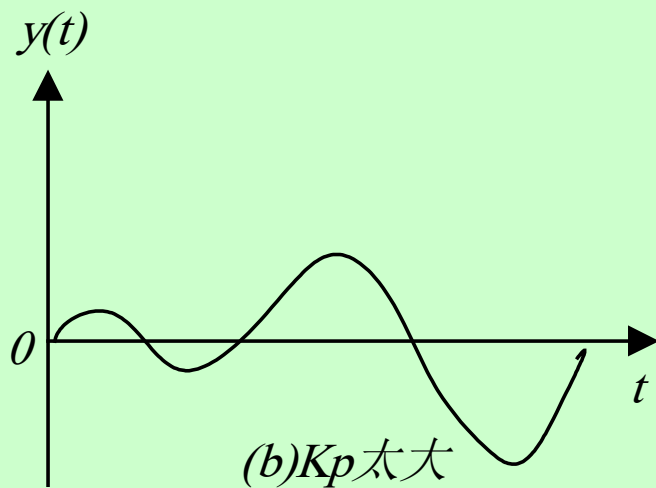
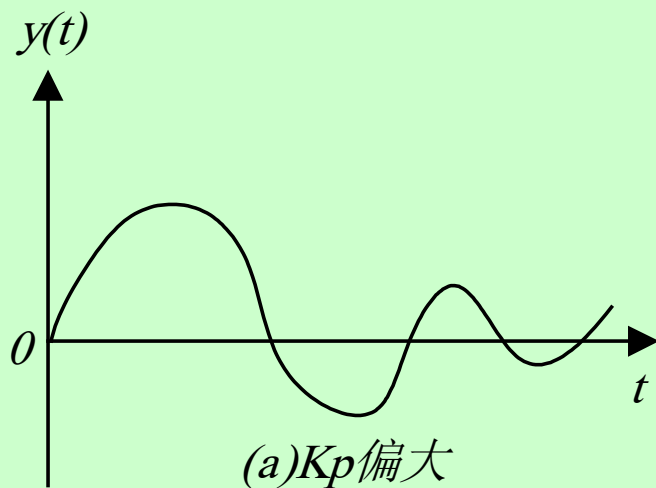
- 对动态性能的影响
- 对稳态特性的影响



①对动态特性的影响

比例控制 K_p 加大，使系统的动作灵敏提高，速度加快； K_p 偏大，振荡次数加多，调节时间加长；当 K_p 太大时，系统会趋于不稳定。若 K_p 太小，又会使系统的动作缓慢。下图比较了不同 K_p 对动态性能的影响。

图4.16
不同 K_p 对
动态性能
的影响





②对稳态特性的影响

在系统稳定的情况下，加大比例控制 K_p ，可以减小稳态误差 ess ，提高稳态精度。但是加大 K_p 只是减少 ess ，却不能完全消除稳态误差。



(2) 积分控制参数 T_i 对系统性能的影响

积分控制通常与比例控制或微分控制联合作用，构成**PI控制**或**PID控制**。积分控制对性能的影响如图4.17所示。

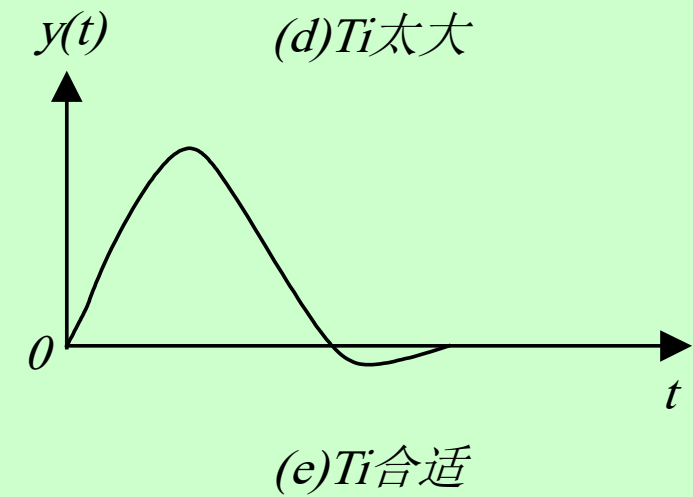
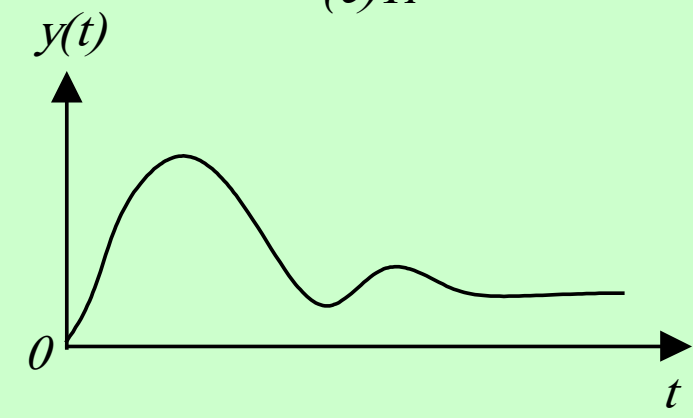
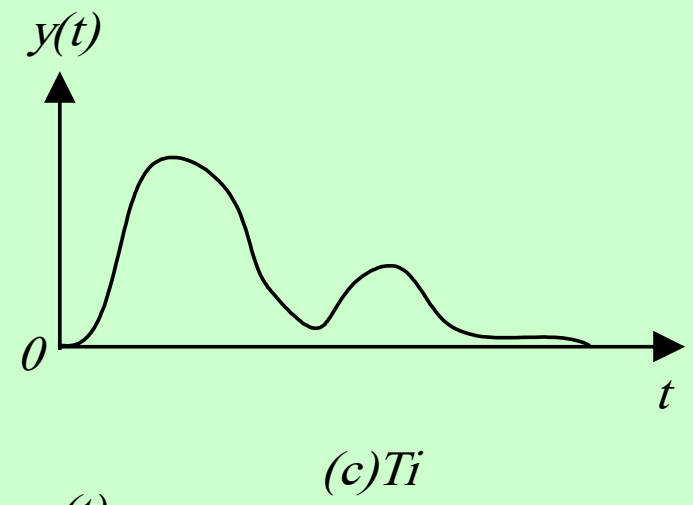
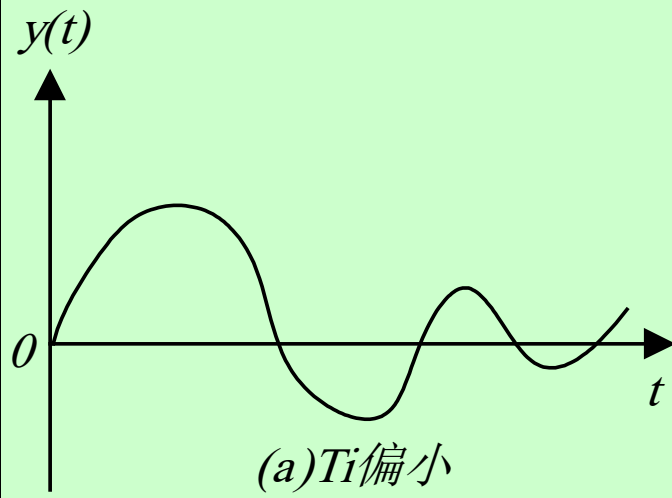
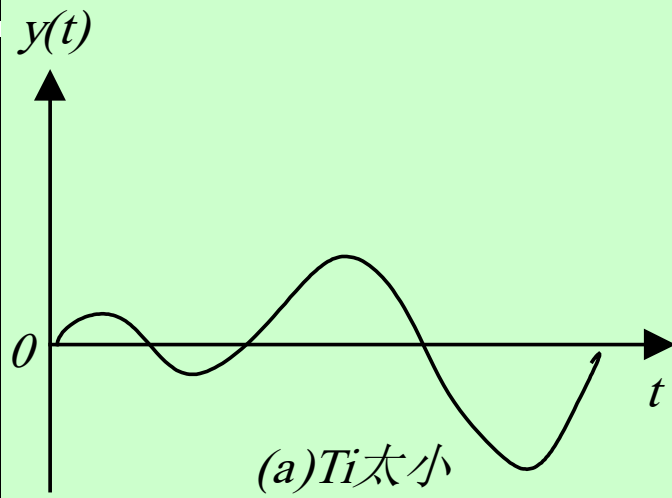
$$K_i = \frac{K_p T}{T_i}$$



①对动态特性的影响

积分控制通常使系统的稳定性下降。 T_i 太小（即 K_i 过大）系统将不稳定； T_i 偏小，振荡次数较多； T_i 太大，对系统性能的影响减少；当合适时，过渡特性比较理想。

图4.17
积分控制
Ti对控制
性能的影响





②对稳态特性的影响

积分控制 T_i 能消除系统的稳态误差，提高控制系统的控制精度。但是若 T_i 太大时，积分作用太弱，以至不能减小稳态误差。

理论上讲，对于I型以上的系统来说，其稳态误差均为0。



(3) 微分控制参数 T_d 对系统性能的影响

微分控制通常与比例控制或积分控制联合作用，构成**PD控制**或**PID控制**。

微分控制可以改善动态特性，如超调量 σ_p 减少，调节时间 t_s 缩短，允许加大比例控制，使稳态误差减小，提高控制精度。当 T_d 偏大或偏小时，都会使超调量 σ_p 较大，调节时间 t_s 较长。只有 T_d 合适时，可以得到比较满意的过渡过程。

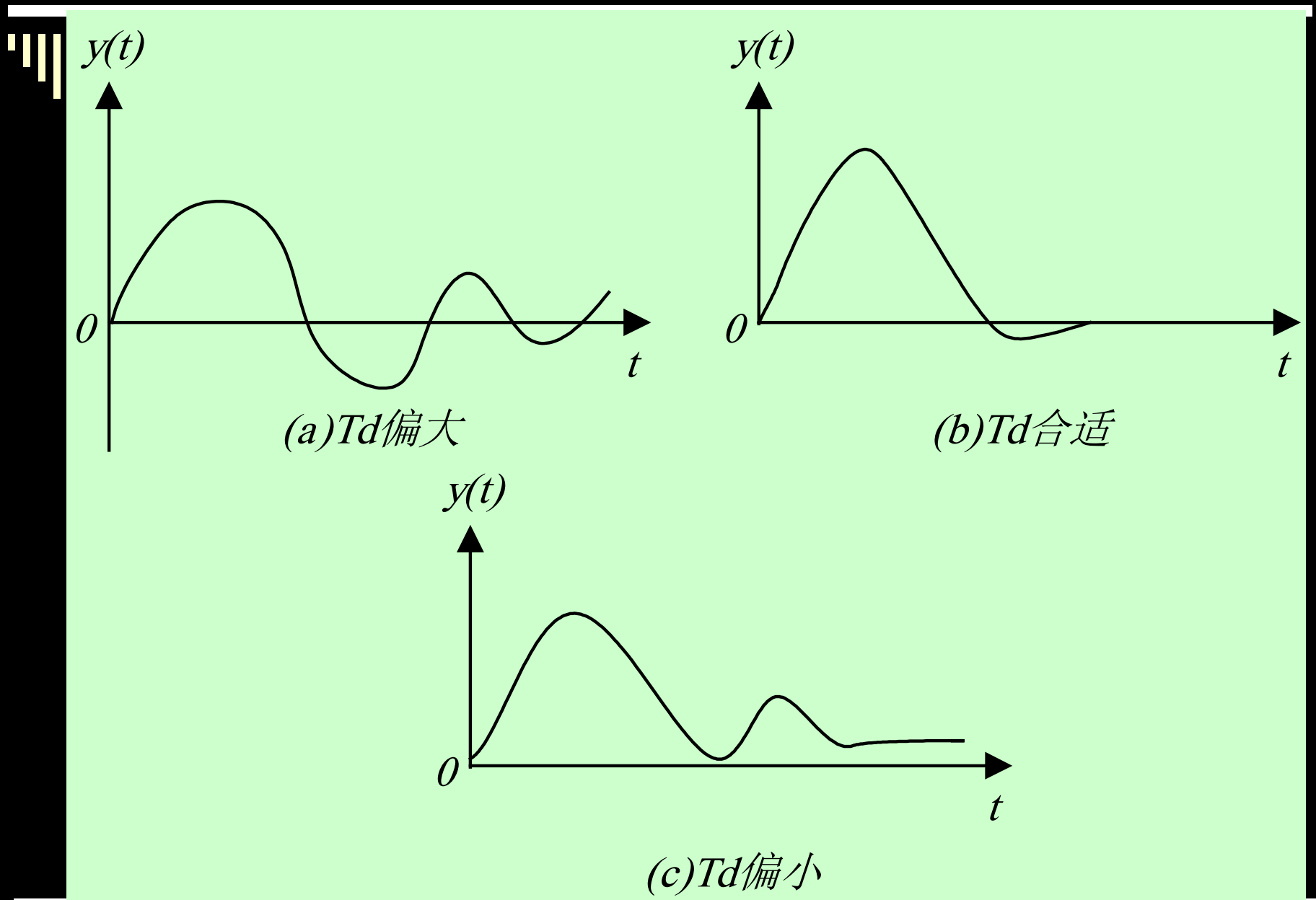
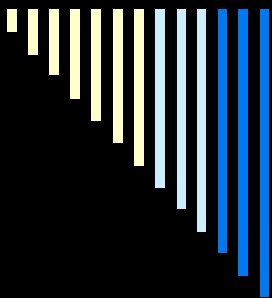
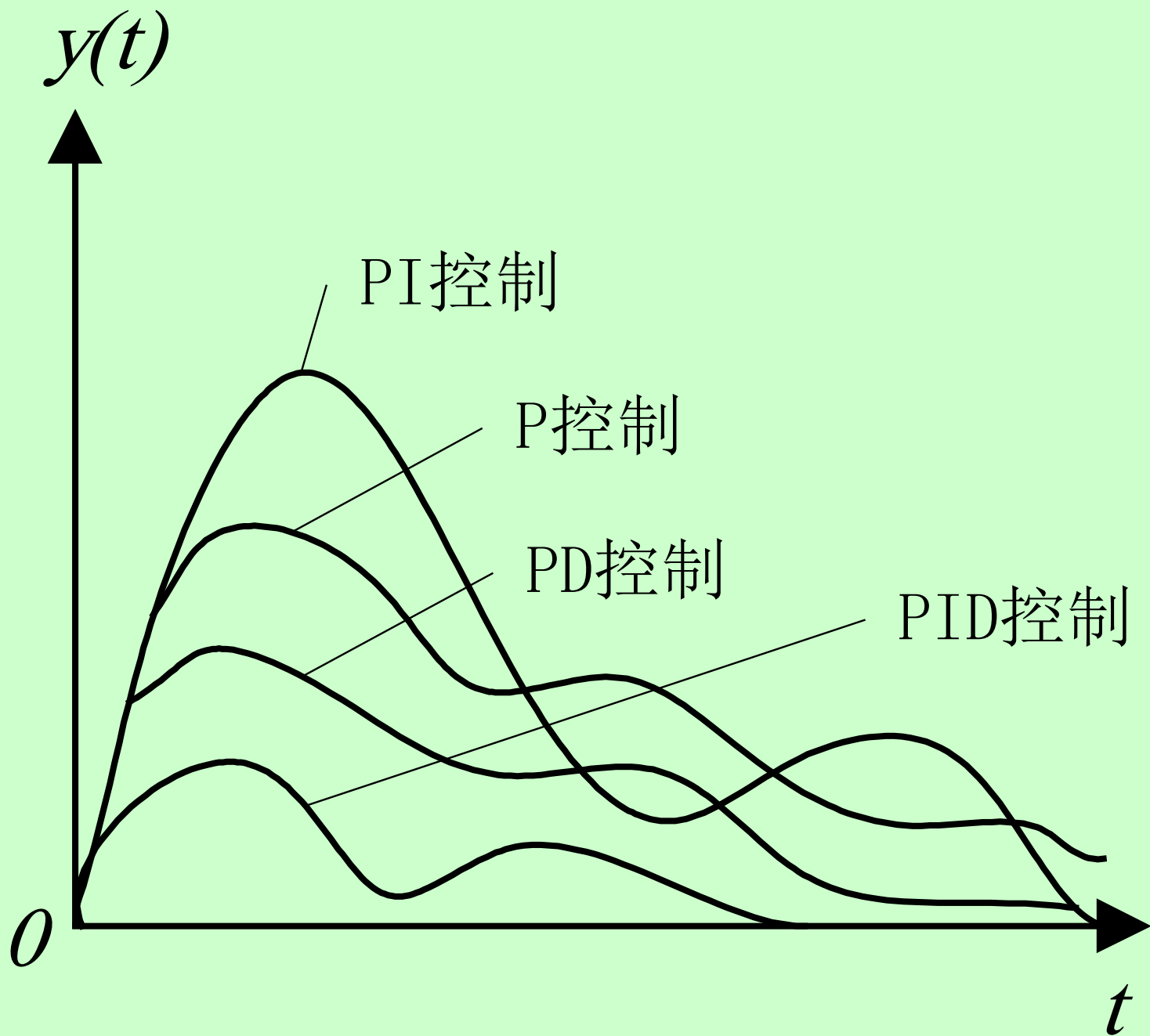


图4.18 微分控制 T_d 规律对控制性能的影响¹³²



不同的控制规律各有特点，*对于相同的控制对象，不同的控制规律有不同的控制效果*，下图曲线是不同控制规律时的过渡过程曲线。





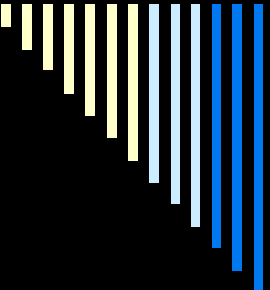
2. 控制规律的选择

长期以来PID调节器应用十分普遍，为广大工程技术人员所接受和警悉。究其原因，可以证明对于特性为一阶惯性纯滞后、二阶惯性纯滞后的控制对象，PID控制是一种最优的控制算法。PID控制参数 K_p 、 T_i 、 T_d 相互独立，参数整定比较方便，PID算法比较简单，计算工作量比较小，容易实现多回路控制。



几点结论

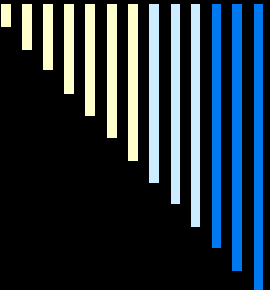
- ①对于一阶惯性的对象，负荷变化不大，工艺要求不高，可采用比例(P)控制。例如，用于压力、液位、串级副控回路等。
- ②对于一阶惯性与纯滞后环节串联的对象，负荷变化不大，要求控制精度较高，可采用比例积分(PI)控制。例如，用于压力、流量、液位的控制。

- 
- ③对于*纯滞后时间 τ 较大*，负荷变化也较大，控制性能要求高的场合，可采用*比例积分微分(PID)控制*。例如，用于过热蒸汽温度控制、PH值控制。
- ④当对象为*高阶(二阶以上)惯性环节又有纯滞后特性*，负荷变化较大，控制性能要求也高时，应*采用串级控制，前馈—反馈、前馈—串级或纯滞后补偿控制*。



3. 采样周期 T 的选择

采样周期 T 在计算机控制系统中是一个重要参量，从信号的保真度来考虑，采样周期 T 不宜太长，也就是采样角频率 ω ($\omega = 2\pi / T$) 不能太低，**采样定理**给出了下限频率即 $\omega_s \geq 2\omega_m$ ， ω_m 是原来信号的最高频率。从控制性能来考虑，采样周期 T 应尽可能地短，也即 ω_s 应尽可能地高。



但是采样频率越高，对计算机的运算速度要求越快，存储器容量要求越大，计算机的工作时间和工作量随之增加。另外，采样频率高到一定程度，对系统性能的改善已经不显著了。所以，对每个回路都可以找到一个最佳的采样周期 T 。下图是从功能和经济角度分析的最佳采样周期。

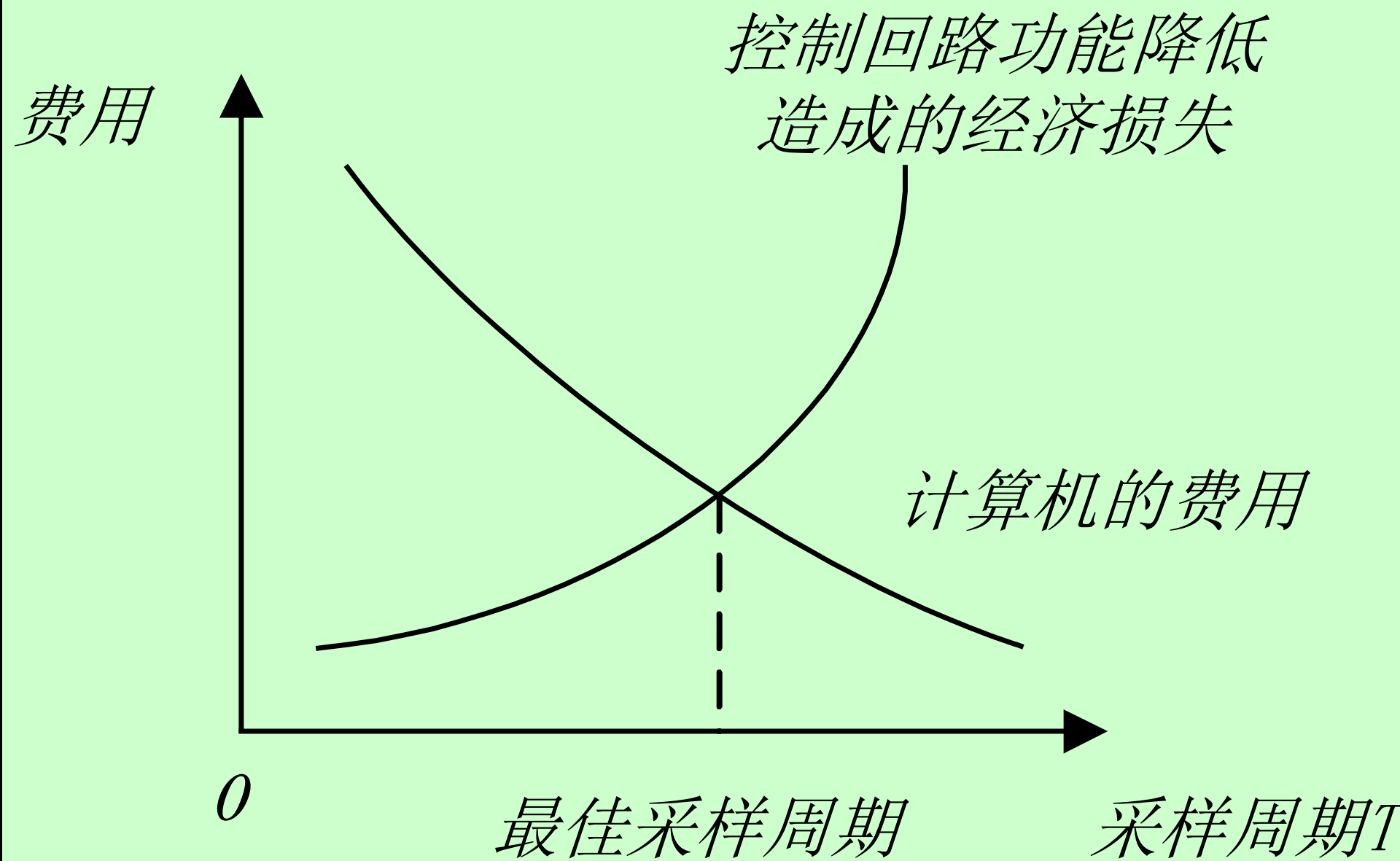


图4.20 从功能和经济上选最佳采样周期¹⁴⁰



与采样周期 T 的选择相关的因素

- ① 作用于系统的扰动信号频率 f_n ：通常 f_n 越高，要求采样频率 f_s 也要相应提高，即采样周期 ($T = 2\pi / f_s$) 缩短。



②对象的动态特性:

当系统中仅是惯性时间常数起作用时, $\omega_s \geq 10\omega_m$, ω_m 为系统的通频带;

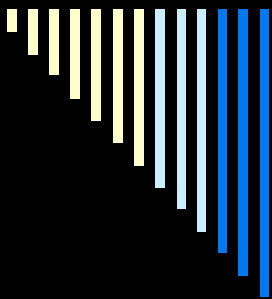
当系统中纯滞后时间 τ 占有一定分量时, 应该选择 $T \approx \tau / 10$;

当系统中纯滞后时间 τ 占主导作用时, 可选择 $T \approx \tau$ 。



表4.2 常见对象选择采样周期的经验数据

受控物理量	采样周期 / s	备 注
流量	1~5	优先选用1~2 s
压力	3~10	优先选用6~8 s
液位	6~8	优先选用7 s
温度	15~20	取纯滞后时间常数
成分	15~20	优先选用18 s



在实际应用中，用理论计算来确定采样周期存在一定的困难。如信号最高频率、噪声干扰源频率都难以确定，所以*应根据实际情况选择合适的采样周期。*

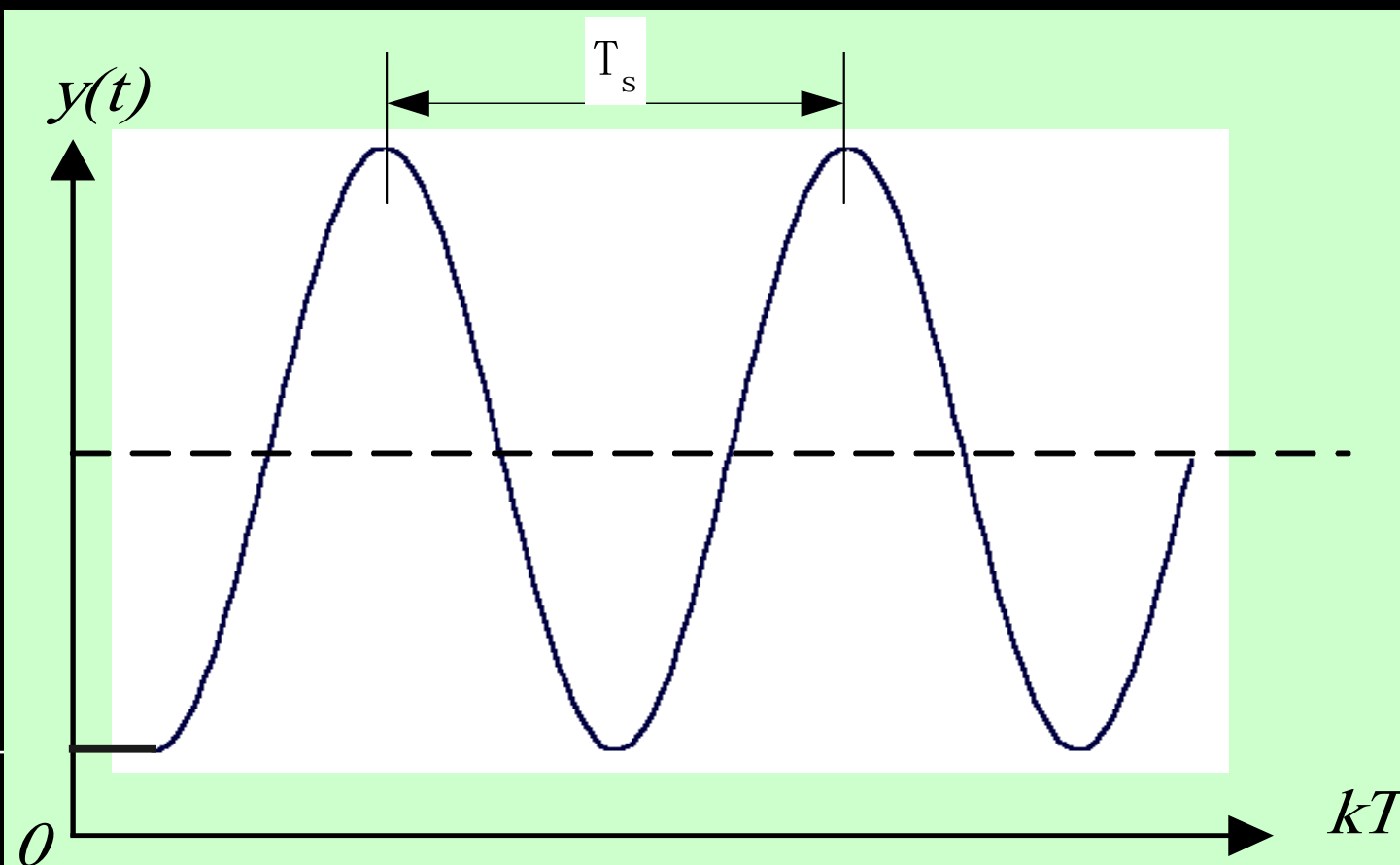


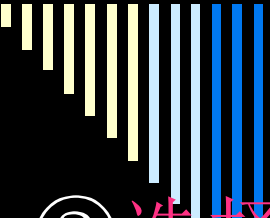
4. 扩充临界比例度法选择PID参数

扩充临界比例度法是以模拟调节器中使用的 **临界比例度法** 为基础的一种PID数字调节器参数的整定方法。整定步骤如下：

①选择合适的采样周期 T ，调节器作纯比例 K_p 控制。

②逐渐加大比例参数 K_p ，使控制系统出现临界振荡，如图所示。由临界振荡过程求得相应的临界振荡周期 T_s ，并记下临界振荡增益 K_s 。





③选择控制度，控制度 Q 的定义是数字调节器和模拟调节器所对应的过渡过程的误差平方的积分之比。

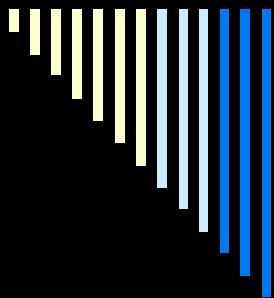
$$Q = \frac{\left[\min \int_0^{\infty} e^2 dt \right]_D}{\left[\min \int_0^{\infty} e^2 dt \right]_A}$$

实际上，只是一个大概的 **性能的评价函数**，不必具体计算，通常控制度取 1.05。



④选择控制制度以后，按下表选择采样周期T、Kp、Ti和Td。

控制度	控制规律	T / Ts	Kp / Ks	Ti / Ts	Td / Ts
1.05	PI	0.03	0.55	0.88	—
	PID	0.014	0.63	0.49	0.14
1.20	PI	0.05	0.49	0.91	—
	PID	0.043	0.47	0.47	0.16
1.50	PI	0.14	0.42	0.99	—
	PID	0.09	0.34	0.43	0.20
2.00	PI	0.22	0.36	1.05	—
	PID	0.16	0.27	0.40	0.22
模拟调节器	PI	—	0.57	0.83	—
	PID	—	0.70	0.50	0.13
Ziegler-Nichols 整定式	PI	—	0.45	0.83	—
	PID	—	0.60	0.50	0.125



⑤按照求得的整定参数，进行系统投运，观察控制效果，再适当调整参数，直到获得比较满意的控制效果。



5. PID归一参数的整定法

调节器参数的整定乃是一项繁琐而又费时的的工作，当一台计算机控制数十乃至数百个控制回路时，整定参数是十分浩繁的工作。因此，近年来国内外在数字PID调节器参数的工程整定方面作了大量的研究工作，*PID归一参数的整定法*是一种简易的整定法。



PID增量式为:

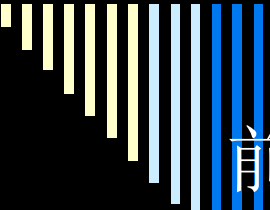
$$\Delta u(k) = Ae(k) + Be(k-1) + Ce(k-2)$$

其中: A、B、C均为“常数”

$$A = K_p + K_I + K_D = K_p \left(1 + \frac{T}{T_i} + \frac{T_D}{T}\right)$$

$$B = -(K_p + 2K_D) = -K_p \left(1 + \frac{2T_D}{T}\right)$$

$$C = K_D = K_p \frac{T_d}{T}$$



前面介绍的数字PID调节器参数的整定，就是要确定式中， T 、 K_p 、 T_i 和 T_d 四个参数，为了减少在线整定参数的数目，根据大量实际经验的总结，人为假设约束的条件，以减少独立变量的个数。

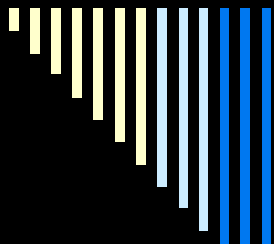
根据表4.3中的Ziegler—Nichols整定式可得

$T \approx 0.1 T_s$ (根据P.D.Roberts提出的简化扩充临界比例带法，这个约束条件人为规定。)

$$T_i \approx 0.5 T_s$$

$$T_d \approx 0.125 T_s$$

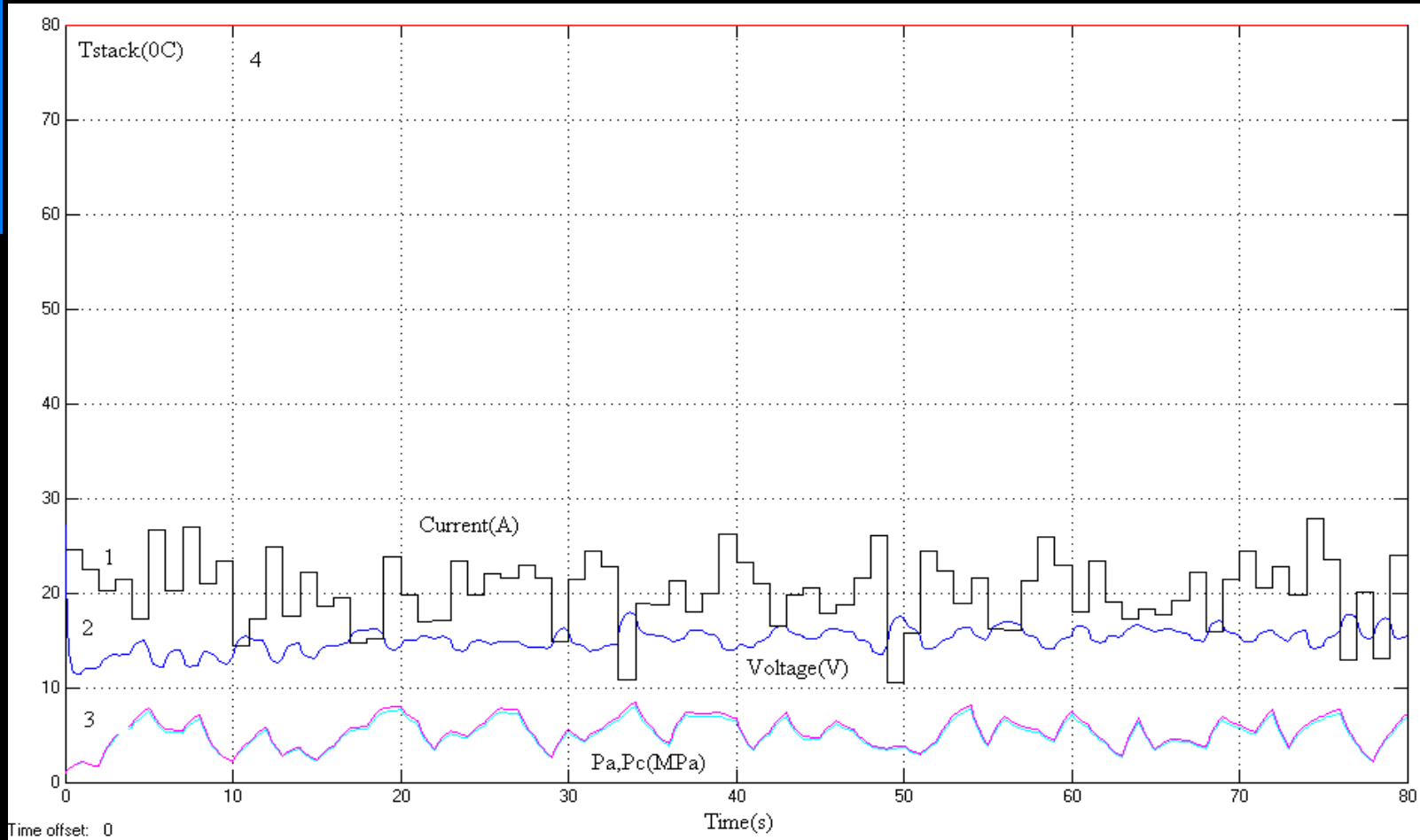
式中， T_s 是纯比例控制时的临界振荡周期。



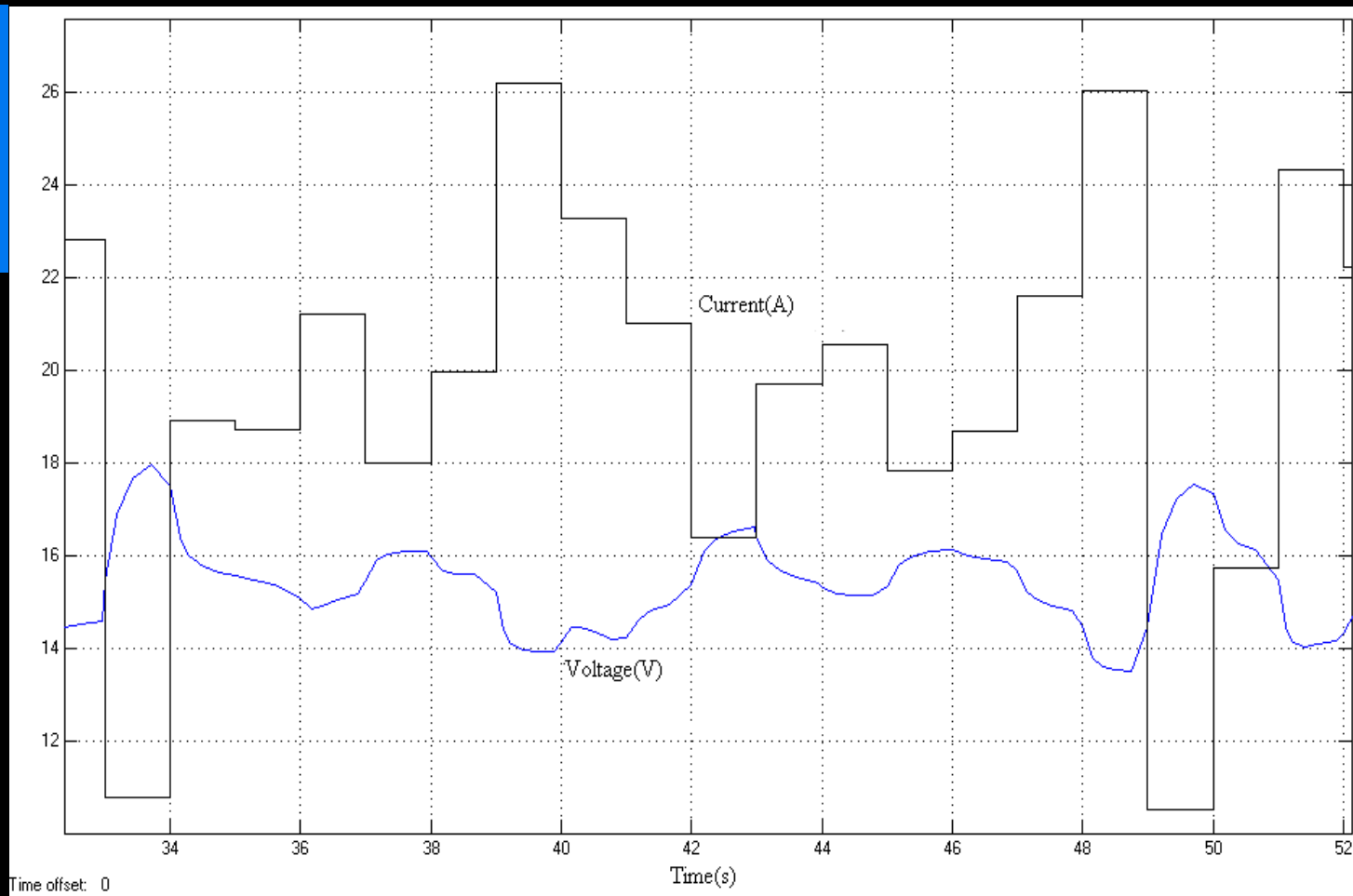
$$\Delta u(k) = K_p [2.45e(k) - 3.5e(k-1) + 1.25e(k-2)]$$

对四个参数的整定简化成对一个参数 K_p 的整定，问题明显地简化。

应用约束条件减少整定参数数目的归一参数整定法是有发展前途的，因为它不仅对数字PID调节器的整定有意义，而且对实现PID自整定系统也将带来许多方便。

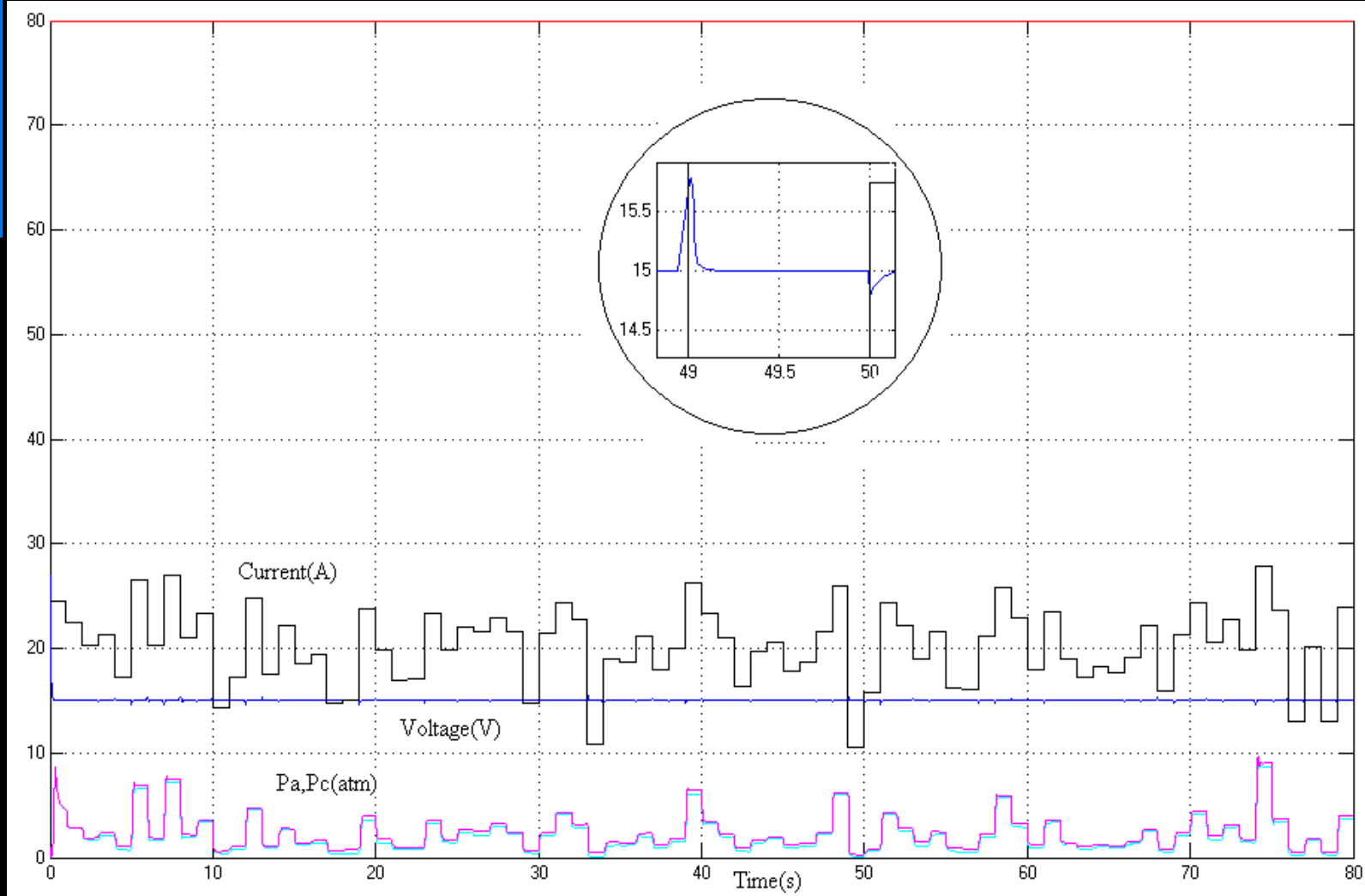


未加PID控制器时PEMFC电堆的输出



周期性较强的一段波形

选取的这段曲线大约有五个周期, $T_p = 3$, 则 $K_c = 7500$, $T_i = 1.5$, $T_d = 0.38$;

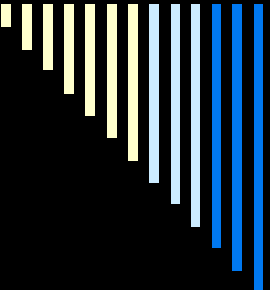


参数整定后的波形

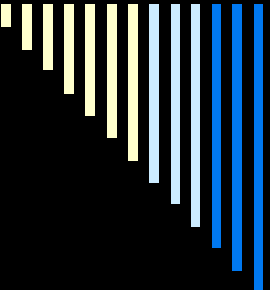


6. 凑试法

由于实际系统错综复杂，参数千变万化，对象的动态特性并非容易，同时在不同的工况下参数可能也不一样，因此可以采用凑试法，其步骤如下。



1) 先将 K_d 、 K_i 设为0，逐渐加大比例参数 K_p （或先取大，然后用0.618黄金分割法选择 K_p ）观察系统的响应，直到反应快、超调小的响应曲线。如果系统没有静差或静差很小已小到允许的范围，并响应曲线已属满意，则只须用比例调节器即可。



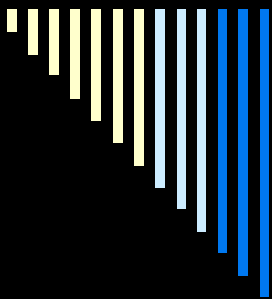
2) 如果比例调节时系统的静差不能满足设计要求，可以加入积分环节，同样 K_i 先选小，然后逐渐加大（或先取大，然后用0.618黄金分割法选择 K_i ），直到得到较满意的响应曲线。这个过程也可以改变 K_p 。

3) 若动态性能不好，这可以加大 K_d 以提高响应速度，减少超调；但对于干扰较敏感的系统，则要谨慎，加大 K_d 可能反而加大系统的超调量。



4.3 史密斯(Smith)预估控制

在实际生产过程中，大多数工业对象具有较大的纯滞后时间。对象的纯滞后时间 τ 对控制系统的控制性能极为不利。当对象的纯滞后时间 τ 与对象的时间常数 T_c 之比，即 $\tau / T_c \geq 0.5$ 时，采用常规的PID控制来克服大纯滞后是很难适应的，而且还会使控制过程严重超调，稳定性变差。



长期以来，人们对纯滞后对象的控制作了大量的研究。但在工程实践上有效的方法还是不多。比较有代表性的方法有 **大林算法和史密斯预估算法**。

4.3.1 史密斯预估控制原理

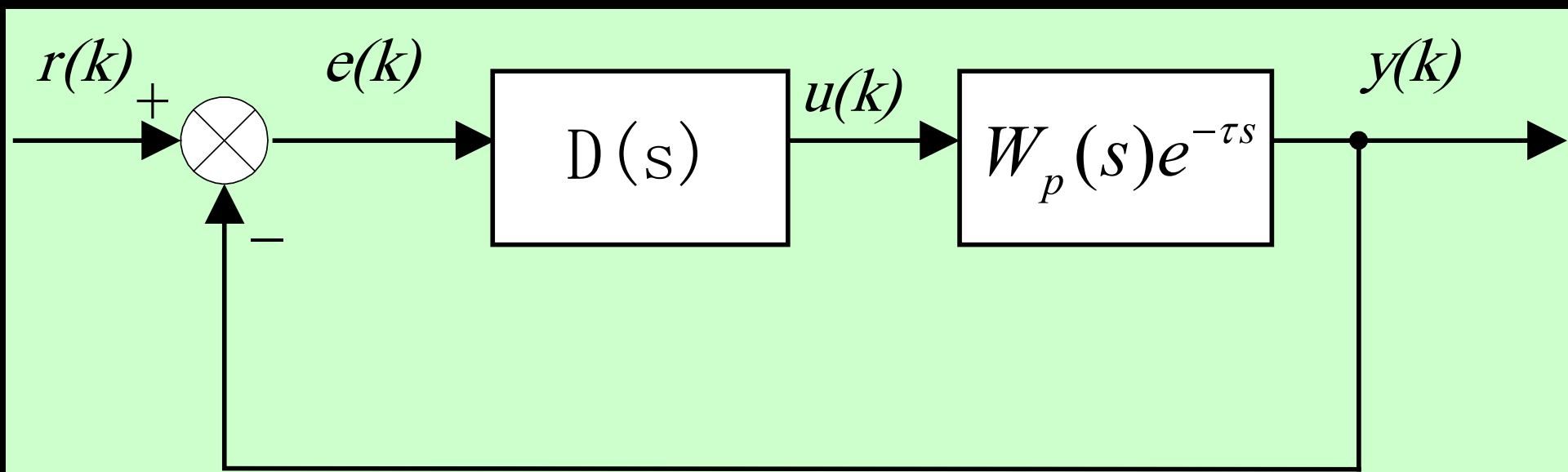
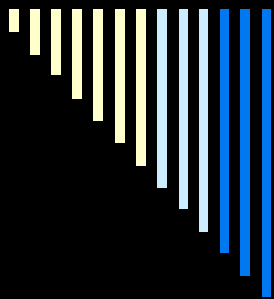


图4.22 带纯滞后环节的控制系统



在上图所示的单回路控制系统中，
 $D(s)$ 表示调节器(控制器)的传递函数；

$W_p(s)e^{-\tau s}$ 表示被控对象的传递函数；

$W_p(s)$ 为被控对象中不包含纯滞后部分的
传递函数；

$e^{-\tau s}$ 为被控对象纯滞后部分的传递函数。

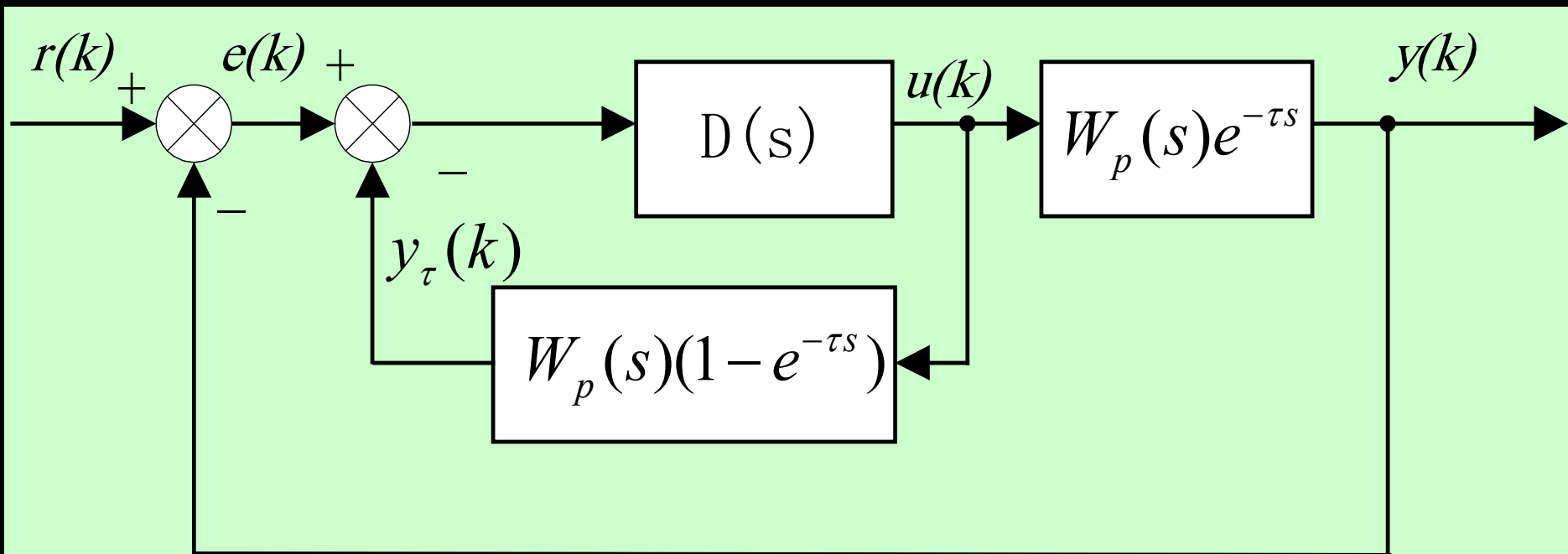


则其闭环传递函数为:

$$\Phi(s) = \frac{D(s)W_p(s)e^{-\tau s}}{1 + D(s)W_p(s)e^{-\tau s}}$$

在闭环传递函数的分母中包含有纯滞后环节，它降低了系统的稳定性。如果 τ 足够大的话，系统将是不稳定的，这就是大纯滞后过程难以控制的本质。

史密斯预估控制原理是：与 $D(s)$ 并接一补偿环节，用来补偿被控对象中的纯滞后部分。这个补偿环节称为预估器，其传递函数为 $W_p(s)(1 - e^{-\tau s})$ 如下图所示



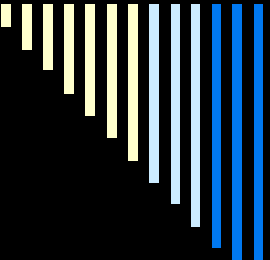


新的控制器闭环传递函数为:

$$D'(s) = \frac{D(s)}{1 + D(s)W_p(s)(1 - e^{-\tau s})}$$

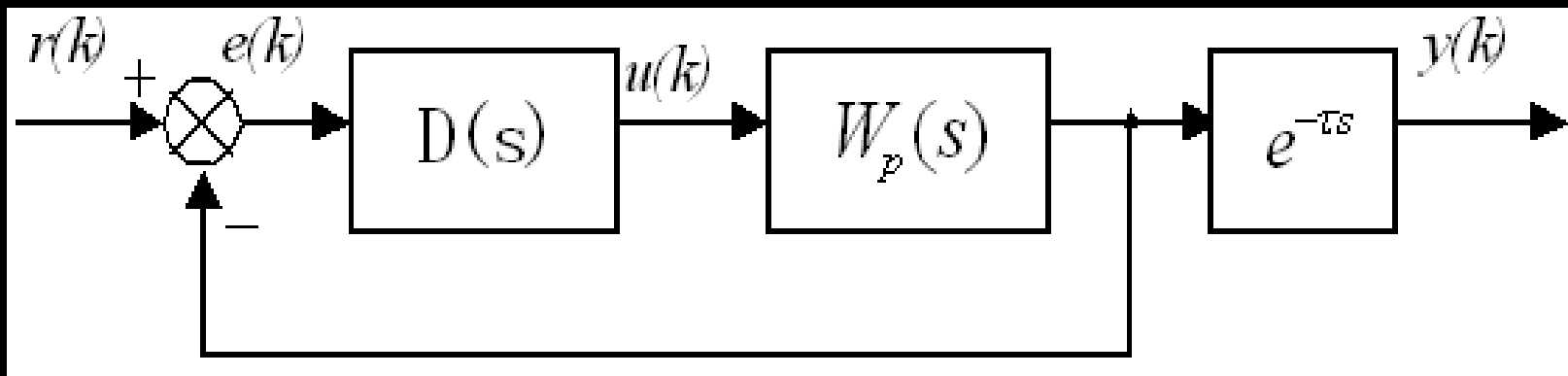
则其总的闭环传递函数为:

$$\Phi'(s) = \frac{D'(s)W_p(s)e^{-\tau s}}{1 + D'(s)W_p(s)e^{-\tau s}} = \frac{D(s)W_p(s)}{1 + D(s)W_p(s)} e^{-\tau s}$$



上式可知，经补偿后，消除了纯滞后部分对控制系统的影响，因为式中的 $e^{-\tau s}$ 在闭环控制回路之外，不影响系统的稳定性。拉氏变换的位移定理说明：

$e^{-\tau s}$ 仅将控制作用在时间坐标上推移了一个时间 τ ，控制系统的过渡过程及其他性能指标都与对象特性为 $W_p(s)$ 时完全相同。



史密斯预估控制系统等效框图



前页下图表明，带纯滞后补偿的控制系统就相当于控制器为 $D(s)$

被控对象为 $W_p(s)(1 - e^{-\tau s})$

反馈回路串上一个 $e^{\tau s}$ 的反馈控制系统，即检测信号通过超前环节 $e^{\tau s}$ 后进入控制器。

因此，从形式上可把纯滞后补偿视为具有超前控制作用，而实质上是对被控参数的预估。

这也就是为什么称史密斯补偿器为 **史密斯预估器** 的原因。

4.3.2 具有纯滞后补偿的数字控制器

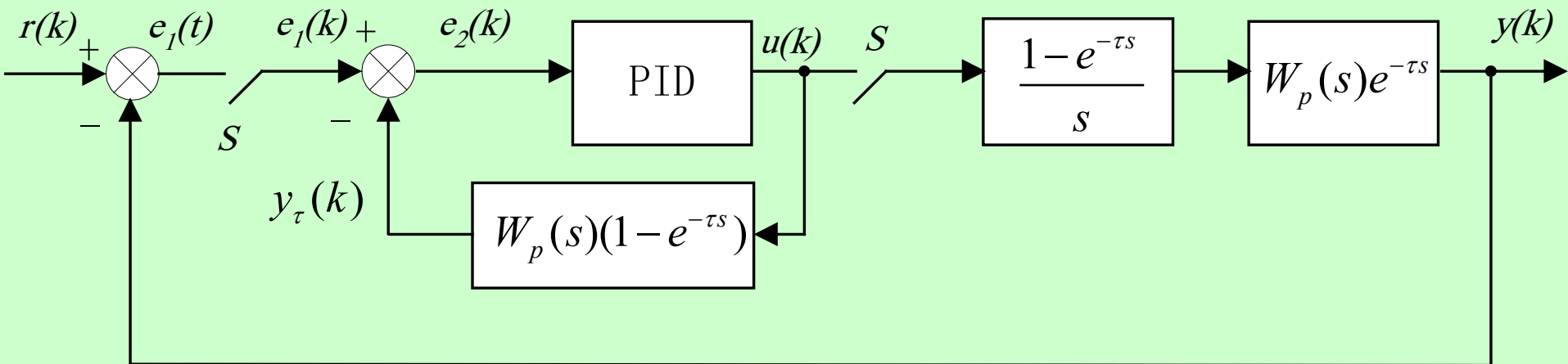


图4.24 具有纯滞后补偿的控制系统

由上图可见，纯滞后补偿的数字控制器由两部分组成：一部分是**数字PID控制器**(由 $D(s)$ 离散化得到)；一部分是**史密斯预估器**。



1. 史密斯预估器

系统中的滞后环节使信号延迟，为此，在内存中专门设定 N 个单元存放信号 $m(k)$ 的历史数据。存储单元的个数 N 由下式决定。

$$N = \tau / T$$

式中， τ —纯滞后时间， T —采样周期。

每采样一次，把 $m(k)$ 记入 0 单元，同时把 0 单元原来存放数据移到 1 单元，1 单元原来存放数据移到 2 单元……以此类推。从 N 单元输出的信号，就是滞后 N 个采样周期的 $m(k-N)$ 信号。

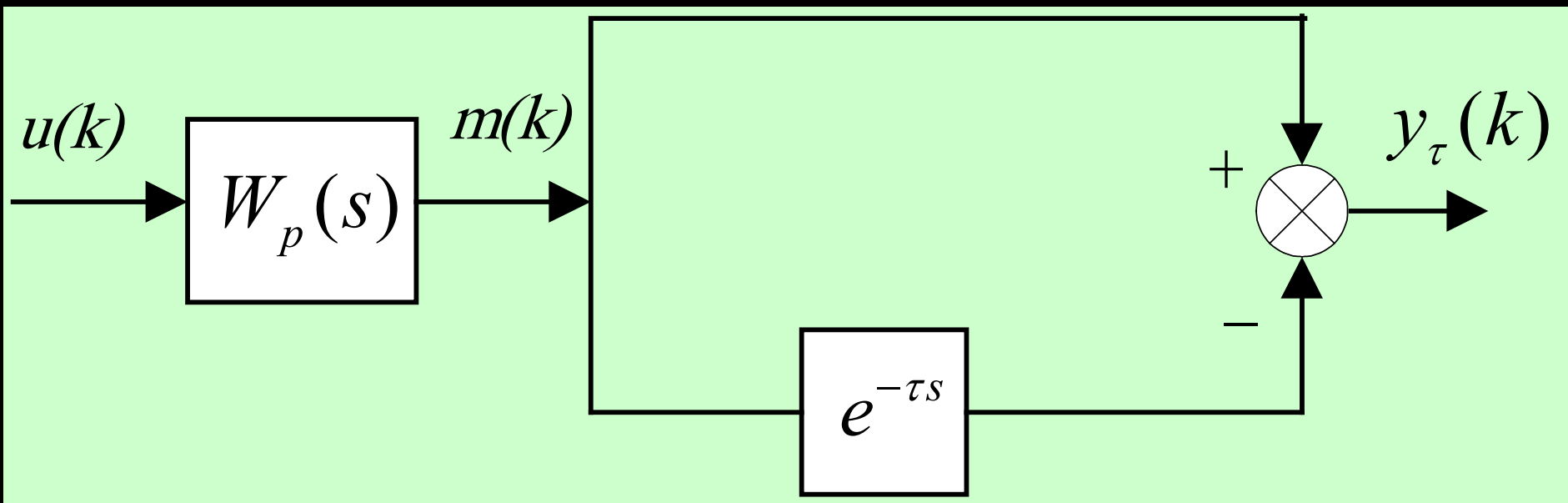


图4.25 史密斯预估器方框

史密斯预估器的输出可按上图的顺序计算。图中， $u(k)$ 是 PID 控制器的输出； $y_{\tau}(k)$ 是史密斯预估器的输出。



许多工业对象可近似用一阶惯性环节和纯滞后环节的串联来表示

$$W_c(s) = W_p(s)e^{-\tau s} = \frac{K_f}{1 + T_f s} e^{-\tau s}$$

式中， K_f ——被控对象的放大系数；

T_f ——被控对象的时间常数；

τ ——纯滞后时间。



预估器的传递函数

$$W_{\tau}(s) = W_p(s)(1 - e^{-\tau s}) = \frac{K_f}{1 + T_f s} (1 - e^{-\tau s})$$



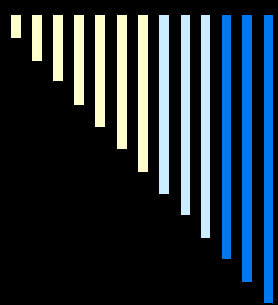
2. 纯滞后补偿控制算法步骤

(1) 计算反馈回路的偏差 $e_1(k)$

$$e_1(k) = r(k) - y(k)$$

(2) 计算纯滞后补偿器的输出 $y_\tau(k)$

$$\frac{Y_\tau(s)}{U(s)} = W_p(s)(1 - e^{-\tau s}) = \frac{K_f}{1 + T_f s} (1 - e^{-\tau s})$$



设

$$N = \frac{\tau}{T}$$

,并 移位:

$$T_f s Y_\tau(s) + Y_\tau(s) = K_f U(s)(1 - e^{-NTs})$$

化成微分方程式, 则可写成

$$T_f \frac{dy_\tau(t)}{dt} + y_\tau(t) = K_f [u(t) - u(t - N)]$$

微分变差分方程:

$$T_f \frac{y_\tau(k) - y_\tau(k-1)}{T} + y_\tau(k) = K_f [u(k) - u(k - N)]$$

$$(T_f + T)y_\tau(k) - T_f y_\tau(k-1) = TK_f [u(k) - u(k - N)]$$

$$(T_f + T)y_\tau(k) = T_f y_\tau(k-1) + TK_f[u(k) - u(k-N)]$$



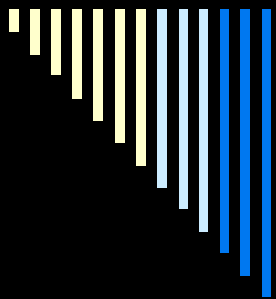
$$y_\tau(k) = \frac{T_f}{T_f + T} y_\tau(k-1) + \frac{TK_f}{T_f + T} [u(k) - u(k-N)]$$

设 $\alpha = \frac{T_f}{T_f + T}$ 则相应的差分方程为:

$$y_\tau(k) = \alpha y_\tau(k-1) + K_f(1-\alpha)[u(k) - u(k-N)]$$

实际计算时写为:

$$y_\tau(k) = \alpha y_f(k-1) + K_f(1-\alpha)[u(k-1) - u(k-N-1)]$$



(3) 计算偏差 $e_2(k)$

$$e_2(k) = e_1(k) - y_\tau(k)$$

(4) 计算控制器的输出 $u(k)$

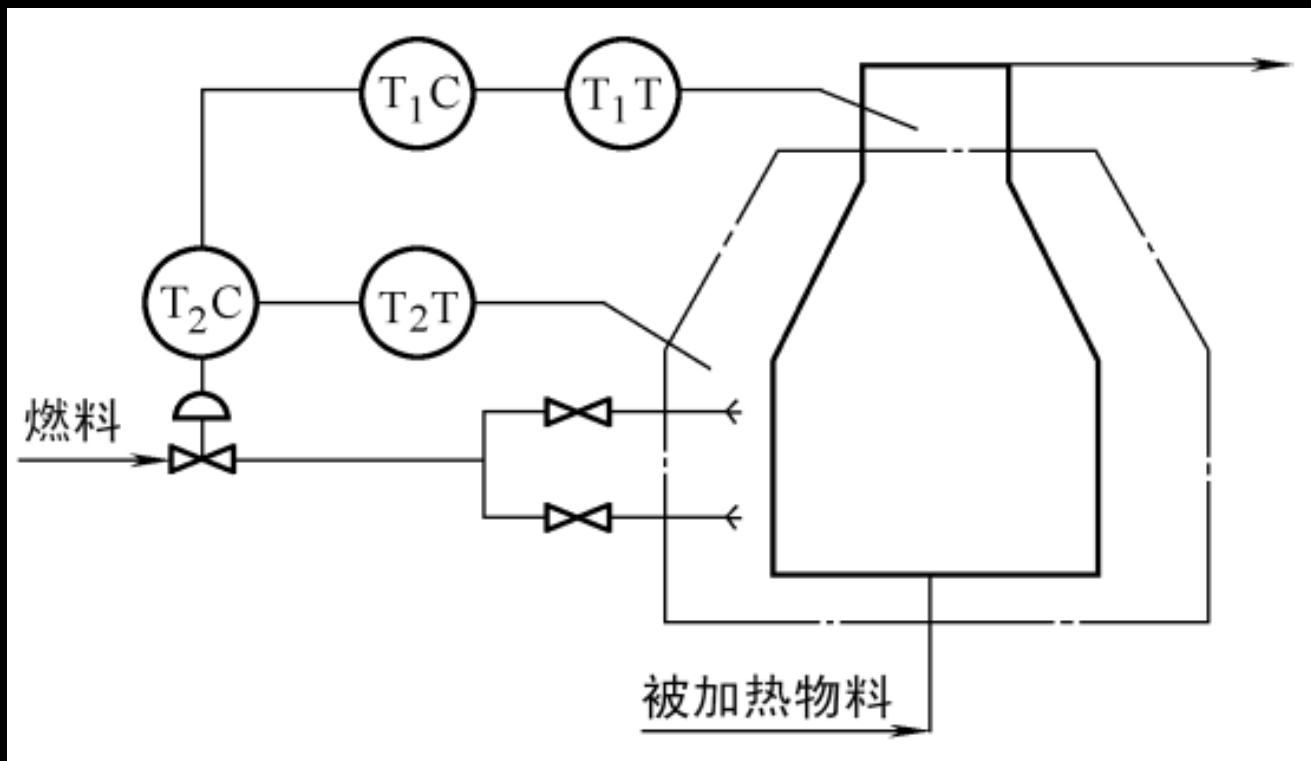
当控制器采用 PID 控制算法时，则

$$u(k) = u(k-1) + \Delta u(k)$$

其中

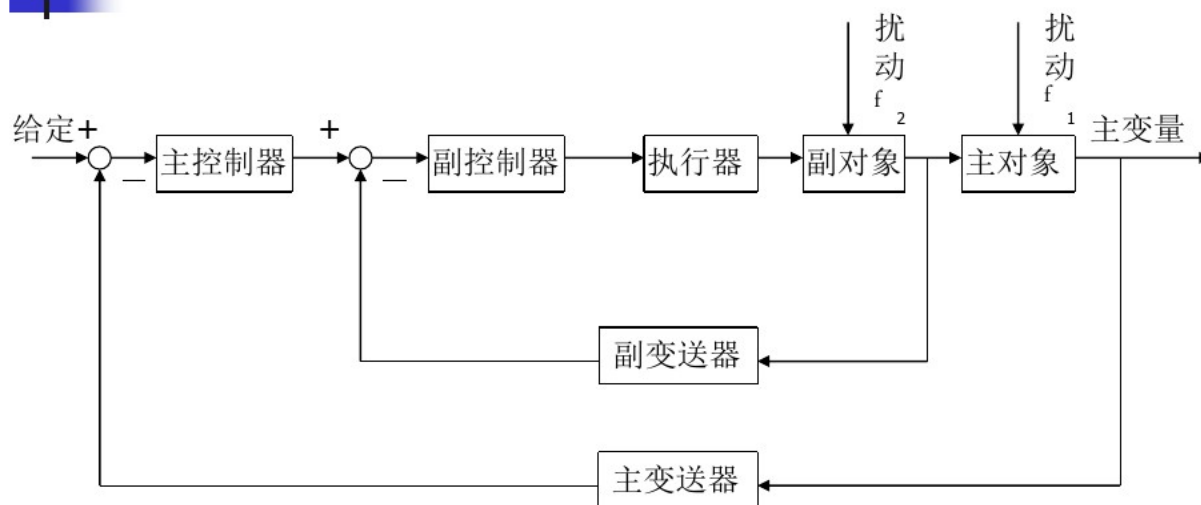
$$\Delta u(k) = Ae(k) + Be(k-1) + Ce(k-2)$$

串级控制



串级控制

串级调节系统的方块图



S203