

计算机控制系统的设计,是指在给定系统性能指标的前提下,设计出控制器的控制规律和相应的数字控制算法。本章主要介绍计算机控制系统常用的设计技术,包括数字 PID 控制器的设计方法、数字控制器的直接设计方法、纯滞后控制技术、串级控制技术、前馈-反馈控制技术和解耦控制技术设计方法。数字控制器的直接设计方法主要介绍最少拍数字控制器设计方法。对大多数系统,采用这些常用的设计技术均能达到满意的控制效果,但对于复杂及有特殊控制要求的系统,则需要采用复杂控制技术,甚至采用现代控制和智能控制技术。

5.1 数字 PID 控制器的设计方法

PID 是 Proportional(比例)、Integral(积分)、Differential(微分)三词的缩写。PID 调节的实质就是根据偏差的比例、积分、微分的函数关系进行运算,并将运算结果进行控制。PID 调节是连续系统中技术最成熟、应用最广泛的一种调节方式。PID 调节器结构简单、参数易于整定,在长期应用中积累了丰富的经验,特别是在工业过程中,由于控制对象的精确数学模型难以建立,系统的参数又经常发生变化,运用现代控制理论进行分析综合要耗费很大代价进行模型辨识,但往往不能得到预期的效果,所以人们常采用 PID 调节器,并根据经验进行在线整定。由于计算机系统的灵活性,PID 算法修正可更加完善。本节将重点介绍数字 PID 控制算法以及与此有关的问题。

5.1.1 模拟 PID 调节器

PID 调节器是一种线性调节器,这种调节器是将设定值 r 与实际输出值 y 进行比较构成偏差 e ,即 $e=r-y$,并将其比例、积分、微分通过线性组合构成控制量 u ,所以简称 PID 调

节器。在实际应用中,根据对象的特性和控制要求,也可灵活地改变其结构,取其中一部分环节构成控制规律,例如,比例(P)调节器、比例积分(PI)调节器、比例微分(PD)调节器等。

如图 5.1 所示的 PID 控制,其模拟调节器的传递函数为

$$D(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s} + K_D s = K_P \left(1 + \frac{1}{T_I s} + T_D s \right) \quad (5-1)$$

其控制规律为

$$u(t) = K_P \left[e(t) + \frac{1}{T_I} \int_0^t e(t) dt + T_D \frac{de(t)}{dt} \right] \quad (5-2)$$

式中, K_P 为比例增益,与比例带 δ 成倒数关系,即 $K_P = 1/\delta$; T_I 为积分时间常数; T_D 为微分时间常数; $u(t)$ 为控制量; $e(t)$ 为偏差。

1. 比例调节器

比例调节器是最简单的一种调节器,其控制规律为

$$u = K_P e + u_0 \quad (5-3)$$

式中, u_0 为控制量的基准,也就是 $e=0$ 时的控制作用(如阀门起始开度、基准电信号等)。

图 5.2 显示了比例调节器对于偏差阶跃变化的时间响应。

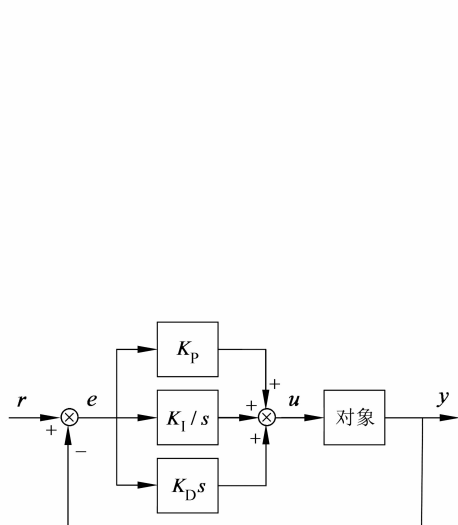


图 5.1 模拟 PID 控制

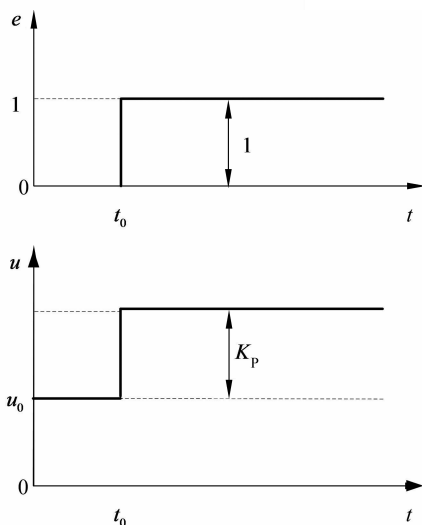


图 5.2 比例调节器的阶跃响应

比例调节器对于偏差 e 是即时反应的,偏差一旦产生,调节器立即产生控制作用使被控量朝着减小偏差的方向变化,控制作用的强弱取决于比例系数 K_P 。

比例调节器虽然直观快速,但有可能存在跟踪静差,如对于具有自平衡性(即系统的阶跃响应值为一定有限值)的控制对象,就可能存在静差。加大比例系数 K_P 可以减小静差,但当 K_P 过大时,会使动态响应质量变差,引起被控量震荡,甚至导致闭环不稳定。

2. 比例积分调节器

为了消除在比例调节器中残存的静差,可以在比例调节器的基础上增加积分调节,形成

积分调节器,其控制规律为

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt \right] + u_0 \quad (5-4)$$

从图 5.3 中可看出,PI 调节器对于偏差的阶跃响应除按比例变化的成分外,还带有累积的成分。只要偏差 e 不为零,它将通过累积作用影响控制量 u ,并减小偏差,直至偏差为零,控制作用不再变化,系统才能达到稳态。因此,积分环节的加入有助于消除系统静差。

显然,如果积分时间 T_i 大,则积分作用弱;反之,积分作用强。增大 T_i 将减慢消除静差的过程,但可减小超调,提高稳定性。 T_i 可根据对象特性选定,对于管道压力、流量等滞后不大的对象, T_i 可以选小一些;对温度等滞后较大的对象, T_i 可以选大一些。

3. 比例积分微分调节器

积分调节作用的加入虽然可以消除静差,但付出的代价是降低了响应速度。为了加快控制过程,有必要在偏差出现或变化的瞬间,不但对偏差量做出即时反应(即比例调节作用),而且对偏差量的变化做出反应,或者说按偏差变化的趋向进行控制,使偏差消灭于萌芽状态。为了达到这一目的,可以在上述 PI 调节器的基础上再增加微分调节,以得到 PID 调节器,其控制规律为

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_D \frac{de(t)}{dt} \right] + u_0 \quad (5-5)$$

理想的 PID 调节器对偏差阶跃变化的响应如图 5.4 所示,它在偏差 e 阶跃变化的瞬间 $t=t_0$ 处有一冲击式瞬时响应,这是由附加的微分环节引起的。

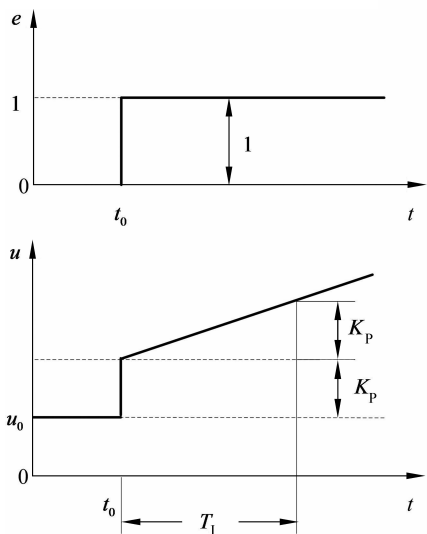


图 5.3 比例积分调节器的阶跃响应

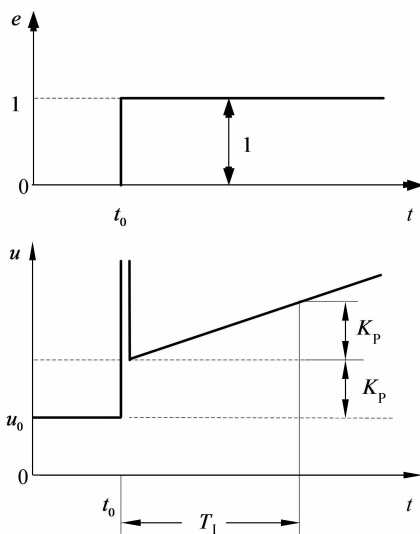


图 5.4 理想的 PID 调节器对偏差阶跃变化的响应

由加入的微分环节 $u_d = K_p T_D \frac{de(t)}{dt}$ 可见,它对偏差的任何变化都产生一个控制作用

u_d ,以调整系统输出,阻止偏差的变化。偏差变化越快, u_d 越大,反馈校正作用也越大,故微分作用的加入将有助于减小超调,克服震荡,使系统趋于稳定。它加快了系统的动作速度,减少了调整时间,从而改善了系统的动态性能。

在工业控制过程中,模拟 PID 调节器有电动、气动、液压等多种类型。这类模拟调节仪表是用硬件实现 PID 调节规律的,PID 参数之间的选择有一定约束。自从计算机进入控制领域以来,用计算机软件(包括 PLC 的指令)实现 PID 调节算法不但成为可能,而且具有更大的灵活性。下面介绍数字 PID 控制算法。

5.1.2 数字 PID 控制算法

由于计算机控制是一种采样控制,它只能根据采样时刻的偏差值计算控制量,因此式(5-5)中的积分和微分项不能直接准确计算,只能用数值计算的方法逼近,用求和代替积分、用后向差分代替微分,使模拟 PID 离散化为差分方程。在采样周期足够短时,可作如下近似:

$$u(t) \approx u(k) \quad (5-6)$$

$$e(t) \approx e(k) \quad (5-7)$$

$$\int_0^t e(t) dt = \sum_{i=0}^k e(i) \Delta t = \sum_{i=0}^k T e(i) \quad (5-8)$$

$$\frac{de(t)}{dt} \approx \frac{e(k) - e(k-1)}{\Delta t} = \frac{e(k) - e(k-1)}{T} \quad (5-9)$$

式中, T 为采样周期; k 为采样序号, $k=0,1,2,\dots,n$;用这种方法近似,可以得到两种形式的数字 PID 控制算法。

1. 位置式 PID 算法

由式(5-5)~式(5-9)可得离散化之后的表达式为

$$u(k) = K_P \left\{ e(k) + \frac{T}{T_I} \sum_{i=0}^k e(i) + \frac{T_D}{T} [e(k) - e(k-1)] \right\} \quad (5-10)$$

式(5-10)表示的控制算法提供了第 k 次采样时调节器的输出 $u(k)$ 。在数字控制系统中,表示第 k 个采样时刻执行机构应达到的位置。如果执行机构采用调节阀,则 $u(k)$ 就对应阀门的开度,所以式(5-10)通常被称为数字 PID 位置型控制算法。

由式(5-10)可以看出,数字调节器的输出 $u(k)$ 与过去的所有偏差信号有关,计算机需对 $e(i)$ 进行累加,运算工作量很大,而且计算机的故障可能使 $u(k)$ 做大幅度的变化,这种情况往往使控制很不方便,有些场合还可能造成严重事故。因此,在实际控制系统中不常用这种方法。

2. 增量式 PID 算法

由于位置式 PID 控制算法使用起来不够方便,不仅对偏差进行累加,占用较多的存储单元,而且不方便编写程序,所以需要做一些改进,对式(5-10)取增量 $\Delta u(k)$,方法如下。

根据递推原理,写出位置式 PID 算法的第 $(k-1)$ 次输出 $u(k-1)$ 的表达式为

$$u(k-1) = K_P \left\{ e(k-1) + \frac{T}{T_i} \sum_{i=0}^{k-1} e(i) + \frac{T_D}{T} [e(k-1) - e(k-2)] \right\} \quad (5-11)$$

用式(5-10)减去式(5-11),可得数字 PID 增量式控制算法为

$$\begin{aligned} \Delta u(k) &= u(k) - u(k-1) \\ &= K_P \left\{ e(k) - e(k-1) + \frac{T}{T_i} e(k) + \frac{T_D}{T} [e(k) - 2e(k-1) + e(k-2)] \right\} \\ &= K_P [e(k) - e(k-1)] + K_I e(k) + K_D [e(k) - 2e(k-1) + e(k-2)] \end{aligned} \quad (5-12)$$

式中, K_P 为比例系数; $K_I = \frac{K_P T}{T_i}$ 为积分系数; $K_D = \frac{K_P T_D}{T}$ 为微分系数。

式(5-12)的输出 $\Delta u(k)$, 表示第 k 次与第 $(k-1)$ 次调节器的输出差值, 即在第 $(k-1)$ 次的基础上增加(或)减少的量。在很多控制系统中, 由于执行机构是步进电动机或多圈电位器进行控制的, 所以只给出一个增量信号即可。

增量式算法和位置式算法本质上没有大的区别, 虽然增量式算法只是在位置式算法的基础上做了一点改动, 但却具有以下 3 个优点。

(1) 增量算法不需要做累加, 控制量增量的确定仅与最近几次误差采样值 $e(k)$ 、 $e(k-1)$ 、 $e(k-2)$ 有关, 计算误差或计算精度对控制量的计算影响较小。位置式控制算法要用到过去的误差的累加, 容易产生大的累加误差。

(2) 增量式算法得出的是控制量的增量, 如阀门控制中只输出阀门开度的变化部分, 误动作影响小, 必要时通过逻辑判断限制或禁止本次输出, 不会严重影响系统的工作。位置式算法的输出是控制量的全量输出, 误动作影响大。

(3) 在位置式控制算法中, 由手动到自动切换时, 必须首先使计算机的输出值等于阀门的原始开度, 即 $u(k-1)$, 才能保证手动到自动的无扰动切换, 这将给程序设计带来困难。增量式设计只与本次的偏差值有关, 与阀门原来的位置无关, 因而可以实现手动/自动的无扰动切换。

5.1.3 数字 PID 控制算法的改进

数字 PID 控制是应用最普遍的一种控制规律。但如果单纯地用数字 PID 控制器模仿模拟调节器, 不会获得很好的控制效果。为了得到更好的控制效果, 人们在实践中不断总结经验, 不断改进, 提出了不同的改进方法, 因而产生了一系列的改进型 PID 控制算法。本节介绍几种常用的 PID 改进算法。

1. 积分项的改进

在 PID 控制中, 积分的作用是消除残差。为了提高控制性能, 对积分项可以采取以下 4 条改进措施。

(1) 积分分离。

在一般的 PID 控制中, 在过程的启动、结束、大幅度增减设定值或出现较大的扰动时, 短时间内系统的输出会出现较大的偏差。由于系统的惯性和滞后性, 在积分项的作用下, 往往

引起系统的输出产生较大的超调量和长时间的波动。特别是对于温度、成分等变换缓慢的过程,这一现象更为严重。但是,在某些生产过程中,这是绝对不允许的。为了防止这种现象发生,可以引进积分分离算法。

积分分离 PID 算法的基本思想是: 设置一个积分分离阈值 β , 当 $|e(k)| \leq \beta$ 时, 采用 PID 控制, 以便于消除静差, 提高控制精度; 当 $|e(k)| > \beta$ 时, 采用 PD 控制, 以使超调量大幅度降低。采用积分分离 PID 算法, 既可以保持积分的作用, 又减小了超调量, 使得控制性能有较大的改善。

积分分离 PID 算法可以表示为

$$u(k) = K_P e(k) + \alpha K_I \sum_{i=0}^k e(i) + K_D [e(k) - e(k-1)] \quad (5-13)$$

或

$$\Delta u(k) = K_P [e(k) - e(k-1)] + \alpha K_I e(k) + K_D [e(k) - 2e(k-1) + e(k-2)] \quad (5-14)$$

式(5-13)和式(5-14)中, α 为逻辑变量, 其取值为

$$\alpha = \begin{cases} 1, & |e(k)| \leq \beta \\ 0, & |e(k)| > \beta \end{cases} \quad (5-15)$$

积分分离阈值 β 的选取要依据具体控制对象及控制要求而定。若 β 过大, 则达不到积分分离的目的; 若 β 过小, 则一旦被控制量 $y(t)$ 无法跳出积分分离区, 就只能进行 PD 控制, 将会出现静差, 如图 5.5 中的曲线 b 所示。

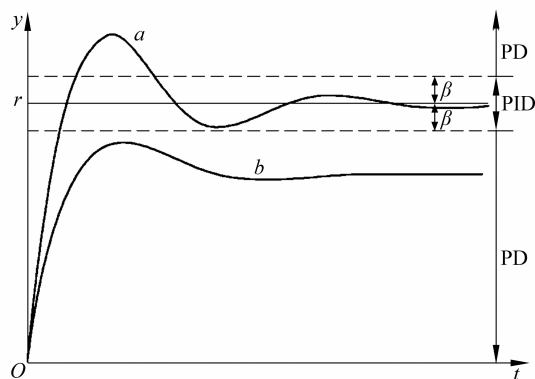


图 5.5 积分分离曲线

(2) 抗积分饱和。

因长时间出现偏差或偏差较大, 计算出的控制量有可能溢出, 或小于零。所谓溢出, 就是计算机运算得出的控制量 $u(k)$ 超出 D/A 转换器所能表示的数值范围。例如, 8 位 D/A 转换器的转换范围为 00H~FFH (H 表示十六进制)。一般执行机构有两个极限位置, 如调节阀全开或全关。设 $u(k)$ 为 FFH 时, 调节阀全开; $u(k)$ 为 00H 时, 调节阀全关。为了提高

运算精度,通常采用双字节或浮点数计算 PID 差分方程式。如果执行机构已到极限位置,仍然不能消除偏差时,由于积分作用,尽管计算 PID 差分方程式所得的运算结果继续增大或减小,但执行机构已无相应的动作,这就称为积分饱和。当出现积分饱和时,势必使超调量增加,控制品质变坏。作为防止积分饱和的办法之一,可对计算出的控制量 $u(k)$ 限幅,同时把积分作用切除掉。若以 8 位 D/A 为例,则有

当 $u(k) < 00\text{H}$, 取 $u(k) = 00\text{H}$;

当 $u(k) > \text{FFH}$, 取 $u(k) = 0\text{FFH}$ 。

(3) 梯形积分。

在 PID 控制中,积分项的作用是消除静差。为了减小静差,应提高积分项的运算精度。为此,可将矩形积分改为梯形积分,其计算公式为

$$\int_0^t e(t) dt \approx \sum_{i=0}^k \frac{e(i) + e(i-1)}{2} \cdot T \quad (5-16)$$

(4) 消除积分不灵敏区。

由式(5-12)可知,数字 PID 的增量型控制算式中的积分项的输出为

$$\Delta u_1(k) = K_I e(k) = K_P \frac{T}{T_1} e(k) \quad (5-17)$$

由于计算机字长的限制,当运算结果小于字长所能表示的数的精度时,计算机就作为“零”将此数丢掉。从式(5-17)可以看出,当计算机的运行字长较短,采样周期 T 也短,而积分时间常数 T_1 又较长时, $\Delta u_1(k)$ 容易出现小于字长的精度而丢数,此积分作用消失,这就称为积分不灵敏区。例如,某温度控制系统,温度量程为 $0 \sim 1275^\circ\text{C}$, D/A 转换为 8 位,并采用 8 位字长定点运算。设 $K_P = 1$, $T = 1\text{s}$, $T_1 = 10\text{s}$, $e(k) = 50^\circ\text{C}$, 根据式(5-17)得

$$\Delta u_1(k) = K_P \frac{T}{T_1} e(k) = \frac{1}{10} \times \left(\frac{255}{1275} \times 50 \right) = 1$$

上式说明,如果偏差 $e(k) < 50^\circ\text{C}$, 则 $\Delta u_1(k) < 1$, 计算机就作为“零”将此数丢掉,控制器就没有积分作用。只有当偏差达到 50°C 时,才会有积分作用。这样势必造成控制系统的残差。

为了消除积分不灵敏区,通常采用以下措施。

① 增加 D/A 和 A/D 转换位数,加长运算字长,这样可以提高运算精度。

② 当积分项 $\Delta u_1(k)$ 连续 n 次出现小于输出精度 ϵ 的情况时,不要把它们作为“零”舍掉,而是把它们一次次累加起来,即

$$S_1 = \sum_{i=1}^n \Delta u_1(i) \quad (5-18)$$

直到累加值 S_1 大于 ϵ 时,才输出 S_1 , 同时把累加单元清零。

2. 微分项的改进

(1) 不完全微分 PID 控制算法。

微分环节的引入是为了改善系统的动态性能,但对于具有高频扰动的生产过程,微分作用的响应过于灵敏,容易引起控制过程振荡,反而会降低控制品质。例如,当被控制量突然

变化时,正比于偏差变化率的微分输出就会很大,而计算机对每个控制回路输出时间是短暂的,且驱动执行器动作又需要一定的时间,如果输出较大,在短暂的时间内,执行器可能达不到控制量的要求值,实质上是丢失了控制信息,致使输出失真,这就是所谓的微分失控。为了克服这一缺点,同时又要使微分作用有效,可以在 PID 控制器的输出端再串联一阶惯性环节(如低通滤波器),抑制高频干扰,平滑控制器的输出,这样就组成了不完全微分 PID 控制器,如图 5.6 所示。

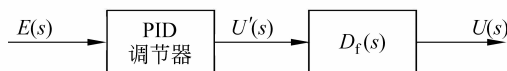


图 5.6 不完全微分 PID 控制器

一阶惯性环节 $D_f(s)$ 的传递函数为

$$D_f(s) = \frac{1}{T_f s + 1} \quad (5-19)$$

因为

$$u'(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_D \frac{de(t)}{dt} \right] \quad (5-20)$$

$$T_f \frac{du(t)}{dt} + u(t) = u'(t) \quad (5-21)$$

所以

$$T_f \frac{du(t)}{dt} + u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_D \frac{de(t)}{dt} \right] \quad (5-22)$$

对式(5-22)进行离散化处理,可得到不完全微分 PID 位置式控制算法。

$$u(k) = \alpha u(k-1) + (1-\alpha)u'(k) \quad (5-23)$$

式中, $u'(k) = K_p \left\{ e(k) + \frac{T}{T_i} \sum_{i=0}^k e(i) + \frac{T_D}{T} [e(k) - e(k-1)] \right\}$, $\alpha = \frac{T_f}{T_f + T}$ 。

与普通 PID 控制算法一样,不完全微分 PID 控制算法也有增量式控制算法,即

$$\Delta u(k) = \alpha \Delta u(k-1) + (1-\alpha) \Delta u'(k) \quad (5-24)$$

式中, $\Delta u'(k) = K_p [e(k) - e(k-1)] + K_I e(k) + K_D [e(k) - 2e(k-1) + e(k-2)]$ 。

在单位阶跃输入作用下,普通 PID 控制算法和不完全微分 PID 控制算法的阶跃响应比较如图 5.7 所示。由图 5.7 可见,普通 PID 控制中的微分作用只在第一个采样周期内起作用,而且作用较强。一般的执行机构无法在较短的采样周期内跟踪较大的微分作用输出,而且理想微分容易引起高频干扰;不完全微分 PID 控制中的微分作用能缓慢地维持多个采样周期,使得一般的工业执行机构能较好地跟踪微分作用的输出。又由于其中含有一个低通滤波器,因此抗干扰能力较强。

(2) 微分先行 PID 控制算法。

为了避免给定值的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 5.8 所示的微分先行 PID 控制方案。它和普通 PID 控制的不同之处在于,它只对被控

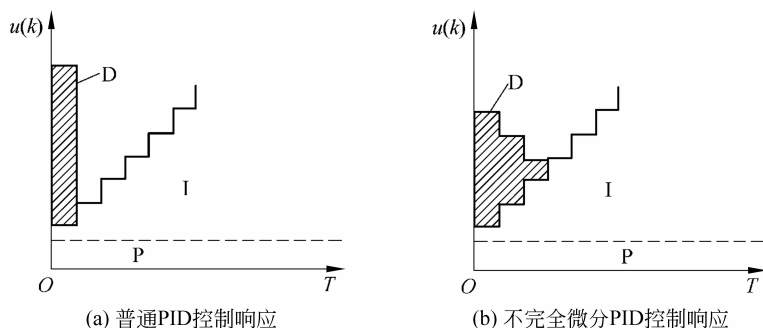


图 5.7 PID 控制的阶跃响应比较

量 $y(t)$ 微分, 不对偏差 $e(t)$ 微分, 也就是对给定值 $r(t)$ 无微分作用。被控量微分 PID 控制算法称为微分先行 PID 控制算法, 该算法对给定值频繁升降的系统无疑是有效的。图 5.8 中, γ 为微分增益系数。

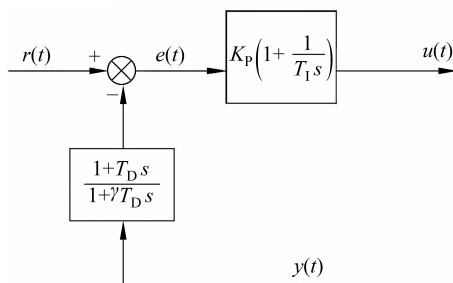


图 5.8 微分先行 PID 控制框图

3. 时间最优 PID 控制

最大值原理是庞特里亚金(Pontryagin)于 1956 年提出的一种最优控制理论。最大值原理也称快速时间最优控制原理, 它是研究满足约束条件下获得允许控制的方法。用最大值原理可以设计出控制变量只在 $u(t) \leq 1$ 范围内取值的时间最优控制系统。而在工程上, 设 $u(t) \leq 1$ 都只取 ± 1 两个值, 而且依照一定法则加以切换, 使系统从一个初始状态转到另一个状态所经历的时间最短, 这种类型的最优切换系统称为开关控制(Bang-Bang 控制)系统。

在工业控制应用中最有发展前途的是 Bang-Bang 控制与反馈控制相结合的系统, 这种控制方式在给定值升降时特别有效, 具体形式为

$$|e(k)| = |r(k) - y(k)| \begin{cases} > \alpha, & \text{Bang-Bang 控制} \\ \leq \alpha, & \text{PID 控制} \end{cases}$$

时间最优位置随动系统理论上应采用 Bang-Bang 控制, 但 Bang-Bang 控制很难保证足够的定位精度, 因此对于高精度的快速伺服系统, 宜采用 Bang-Bang 控制和线性控制相结合的方式, 在定位线性控制段采用数字 PID 控制就是可选的方案之一。

4. 带死区的 PID 控制算法

在计算机控制系统中,某些生产过程对控制精度要求不是很高,但希望系统工作平稳,执行机构不要频繁动作。针对这类系统,人们提出了一种带死区的 PID 控制算法,即在计算机中人为地设置一个不灵敏区,如图 5.9 所示,相应的算式为

$$p(k) = \begin{cases} e(k), & \text{当 } |r(k) - y(k)| = |e(k)| > \epsilon \\ 0, & \text{当 } |r(k) - y(k)| = |e(k)| \leq \epsilon \end{cases}$$

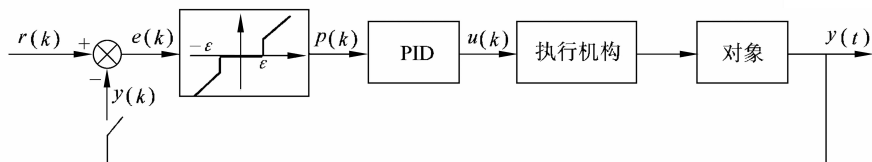


图 5.9 带死区的 PID 控制系统框图

在图 5.9 中,死区 ϵ 是一个可调参数,其具体数值可根据实际控制对象由实验确定。 ϵ 值太小,使调节过于频繁,达不到稳定被调节对象的目的;如果 ϵ 取得太大,则系统将产生很大的滞后;当 $\epsilon=0$ 时,即常规 PID 控制。

该系统实际上是一个非线性控制系统,即当偏差绝对值 $|e(k)| \leq \epsilon$ 时, $p(k)$ 为 0;当 $|e(k)| > \epsilon$ 时, $p(k) = e(k)$,输出值 $u(k)$ 以 PID 运算结果输出。

5.1.4 数字 PID 控制器的参数整定

在 PID 控制器的结构确定以后,系统的性能好坏主要取决于参数的选择是否合理。因此,PID 算法参数的整定非常重要。数字 PID 算法参数整定的任务主要是确定 K_P 、 T_I 、 T_D 和采样周期 T 。

1. 采样周期 T 的确定

根据香农采样定理,采样周期应满足 $T \leq \pi/\omega_{\max}$, $\omega_{\max}$ 为被采样信号的上限角频率。

香农采样定理给出了采样周期的上限 $T_{\max}$ 。采样周期的下限 $T_{\min}$ 为计算机执行控制程序和输入输出所耗费的时间,系统的采样周期只能在 $T_{\min}$ 和 $T_{\max}$ 之间选择。采样周期 T 既不能太大,也不能太小, T 太大时,信号失真比较大; T 太小时,一方面增加了微型计算机的负担,不利于发挥计算机的功能;另一方面,两次采样间的偏差变化太小,数字控制器的输出值变化不大,计算机将失去调节作用。因此,在选择采样周期时,必须综合考虑。

一般应考虑如下因素。

1) 给定值的变化频率

加到被控对象上的给定值变化频率越高,采样频率越高,这样给定值的变化就可以迅速得到反映。

2) 被控对象的特性

若被控对象是慢速变化的对象时,如热工或化工对象,采样周期一般取得较大;若被控