

被控过程的数学模型

学习目标

- (1) 掌握被控过程的基本概念；
- (2) 掌握被控过程的数学模型表示方法；
- (3) 掌握自衡和无自衡过程的机理建模方法；
- (4) 掌握被控过程的实验建模方法；
- (5) 掌握基于神经网络的数据建模方法；
- (6) 了解机理与数据相结合的混合建模方法。

3.1 概述

控制质量的优劣是工业过程自动控制中最重要的问题,它主要取决于自动控制系统的结构及组成控制系统的各个环节的特性。为了很好地控制一个过程,需要知道当控制量变化一个已知量时,被控量如何改变并最终将改变多少以及向哪个方向改变、被控量的变化将需要经历多长时间、被控量随时间变化的曲线形状等。这些均依赖于被控过程的数学模型。因此,建立被控过程的数学模型是自动控制系统分析与设计中的重要环节。

被控过程的数学模型是指被控过程的输出变量与输入变量之间的函数关系数学表达式。

在工业生产过程中,被控过程的种类多种多样,结构千差万别,影响被控过程特性的参数和条件也各不相同。因此,采用数学方法建立被控过程的数学模型,只适用于设备结构比较简单、生产过程机理比较清楚的情况。

被控过程可以按照被控量的个数不同分为单变量的被控过程和多变量的被控过程,也可以按照被控过程是否具有自平衡能力将其分为有自平衡能力和无自平衡能力的过程。

3.1.1 单变量与多变量的被控过程

单变量的被控过程也可以称为单输出的被控过程。

通常,被控过程的输入量不止一个,应该选择容易被控制又直接影响对象动态特性的一个作为控制器的输入,其他的输入均视为扰动作用,如图 3-1 所示。其中,被控量的拉普拉斯变换函数为

$$Y(s) = W_o(s)X(s) + W_{f1}(s)F_1(s) + \cdots W_{fn}(s)F_n(s) \quad (3-1)$$

式中, $X(s), F_1(s), \cdots, F_n(s)$ 为控制信号 x 及扰动信号 $f_1 \cdots f_n$ 的拉普拉斯函数; $W_{f1}(s), \cdots, W_{fn}(s)$ 为在扰动 $f_1 \cdots f_n$ 作用下对象的传递函数; $W_o(s)$ 为在 x 控制作用下对象的传递函数。

图 3-1 中,被控量 y 与输入量 x 之间的联系通道称为控制通道,被控量 y 与扰动 f 之间的联系通道称为扰动通道。

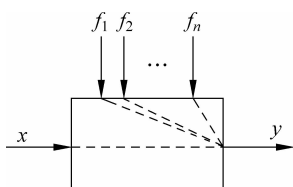


图 3-1 单变量对象及其信号通道示意图

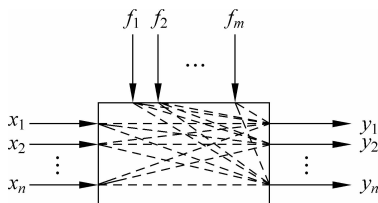


图 3-2 多变量对象及其信号通道示意图

多变量的被控过程也可以称为多输入多输出的被控过程。

当被控过程中存在多个输入量($x_1, x_2, \cdots, x_n, f_1, f_2, \cdots, f_m$)和多个输出量($y_1, y_2, \cdots, y_n$)时,输入量与输出量之间往往不是一对一的关系,也就是说,某个输入量将同时影响两个或两个以上的被控量,即某一个控制作用除了影响与其“对应”的被控量外,还对其他的被控量产生影响。多变量的被控过程可以表示为图 3-2 所示。其中,被控量的拉普拉斯变换函数可表示成如下的矩阵向量形式

$$\mathbf{Y}(s) = \mathbf{W}_o(s)\mathbf{X}(s) + \mathbf{W}_f(s)\mathbf{F}(s) \quad (3-2)$$

其中, $\mathbf{Y}(s) = [Y_1(s) Y_2(s) \cdots Y_n(s)]^T$ 为被控量的拉普拉斯函数向量; $\mathbf{X}(s) = [X_1(s) X_2(s) \cdots X_n(s)]^T$ 为调节量的拉普拉斯变换函数向量; $\mathbf{F}(s) = [F_1(s) F_2(s) \cdots F_m(s)]^T$ 为扰动量的拉普拉斯变换函数向量。 $\mathbf{W}_o(s), \mathbf{W}_f(s)$ 分别为被控过程在调节量和扰动量的作用下被控过程的传递函数矩阵,并可以表示为

$$\mathbf{W}_o(s) = \begin{bmatrix} W_{11}(s) & W_{12}(s) & \cdots & W_{1n}(s) \\ W_{21}(s) & W_{22}(s) & \cdots & W_{2n}(s) \\ \vdots & \vdots & \ddots & \vdots \\ W_{n1}(s) & W_{n2}(s) & \cdots & W_{nm}(s) \end{bmatrix} \quad \mathbf{W}_f(s) = \begin{bmatrix} F_{11}(s) & F_{12}(s) & \cdots & F_{1m}(s) \\ F_{21}(s) & F_{22}(s) & \cdots & F_{2m}(s) \\ \vdots & \vdots & \ddots & \vdots \\ F_{n1}(s) & F_{n2}(s) & \cdots & F_{nm}(s) \end{bmatrix} \quad (3-3)$$

在多变量的被控过程中,如果一个调节量只对一个被控量起作用,则不同控制区是彼此独立的,这时,可以认为多变量的被控过程是由多个彼此独立的单变量被控过程所组成的;

如果一个调节量同时影响多个被控量,并且这些影响是不利的,则采用解耦控制等方法来消除其对其余被控量的影响。在本书中,如果没有特别指出,仅讨论单变量的被控过程。

3.1.2 自衡过程和无自衡过程

当输入量发生变化破坏了被控过程的平衡状态时,如果在没有任何外界干扰的情况下,被控过程依靠自身的能力能够重新达到一个新的平衡状态,那么这个被控过程就被称为有自平衡能力的过程,也可以称其为自衡过程,该被控过程所具有的这个特性称为自衡特性。然而,当输入量发生变化破坏了被控过程的平衡状态时,如果在没有任何外界干扰的情况下,被控过程无法依靠其自身的能力重新达到一个新的平衡状态,那么这个被控过程就被称为无自平衡能力的过程或无自衡过程,该被控过程就不具有自衡特性。

被控过程的动态特性取决于生产过程本身的物理、化学特性,并与生产设备的结构和运行状态有关。原则上可以采用机理方法,根据过程的物料或能量平衡以及过程进行中的物理或化学变化,经过数学推导或简化而得到描述被控过程动态特性的数学表达式。由于生产过程大多比较复杂,要得到正确的表达式十分困难,即使得到也难于求解,有时又因为过于简化而失去了实际的意义。采用实验方法测取被控过程的响应曲线,是分析研究被控过程动态特性的常用方法。

一般常采用测取被控过程的阶跃响应曲线的方法来研究和过程的动态特性,图 3-3 中记录了有自平衡能力过程和无自平衡能力过程的阶跃响应曲线。从图 3-3(a)可以看出,当被控过程的输入量 x 发生一个阶跃变化 Δx 后,被控量 y 的平衡状态遭到破坏,但经过一段时间以后,被控量又达到了一个新的稳定值 $y(\infty)$,在这段时间里并没有外界的干扰,而是依靠其自身的能力重新恢复平衡,因此该曲线是有自平衡能力过程的阶跃响应曲线。图 3-3(b)中的被控量无法再平衡下来,因此是无自平衡能力过程的阶跃响应曲线。

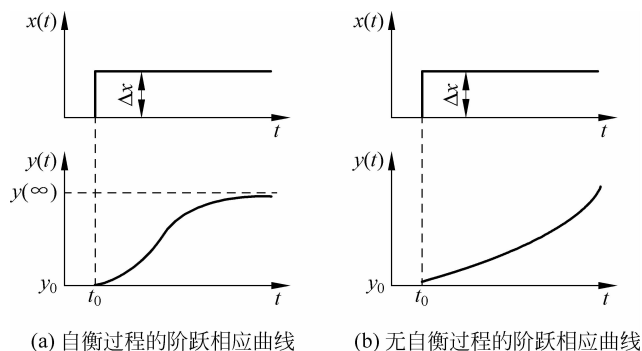


图 3-3 被控过程的阶跃响应曲线

3.1.3 被控过程数学模型的表示方法

求取被控过程数学模型的方法目前有三种。一是根据过程的内在机理,通过静态与动态物料平衡或能量平衡关系,采用数学推导的方法求取过程的数学模型,这种方法适用于内在机理比较清楚、简单的情况;二是根据过程的输入、输出数据,通过过程辨识和参数估计的方法建立被控过程的数学模型,这种方法仅依赖于过程的输入输出数据,不需要知道被控过程的先验知识;三是将上述两种方法相结合,利用机理分析的方法分析确定模型的结构,再通过过程的输入输出数据对模型参数进行估计。

机理推导的几类数学模型可见表 3-1。

表 3-1 数学模型的类型

过程类别	静态模型	动态模型
集中参数过程	代数过程	微分方程
分布参数过程	微分方程	偏微分方程
多级过程	差分方程	微分-差分方程

由系统辨识所得到的模型结构一般比较简单。以单输入单输出的过程模型为例,最常用的有如下形式。

1. 线性时间连续模型(微分方程或传递函数形式)

$$a_n y^{(n)}(t) + \cdots + a_1 y'(t) + y(t) = b_m u^{(m)}(t - \tau) + \cdots + b_1 u'(t - \tau) + b_0 u(t - \tau) \quad (3-4)$$

或

$$W_o(s) = \frac{Y(s)}{U(s)} = \frac{b_0 + b_1 s + \cdots + b_m s^m}{1 + a_1 s + \cdots + a_n s^n} e^{-\tau s} \quad (3-5)$$

式中, y 、 u 和 τ 分别代表被控过程的输出变量、输入变量和滞后时间。

2. 线性时间离散模型

$$\begin{aligned} & a_n y(k - n) + a_{n-1} y(k - n - 1) + \cdots + a_1 y(k - 1) + y(k) \\ & = b_m u(k - m - d) + \cdots + b_1 u(k - 1 - d) + b_0 u(k - d) \end{aligned} \quad (3-6)$$

即

$$y(k) = \frac{b_0 + b_1 q^{-1} + \cdots + b_m q^{-m}}{1 + a_1 q^{-1} + \cdots + a_n q^{-n}} q^{-d} u(k) \quad (3-7)$$

式中, d 、 q^{-1} 分别表示滞后时间(采样周期的整数倍)和后向差分算子, q^{-1} 与 z 变换的 z^{-1} 相当。

与线性系统建模方法不同,由于非线性系统的输入、输出之间具有更为复杂的时间关联性与非线性关系,输入输出非线性时间离散系统模型可以用一般表达式描述为

$$y(k) = f(y(k-1), y(k-2), \dots, y(k-n), u(k-d), u(k-d-1), \dots, u(k-d-m), \xi(k)) \quad (3-8)$$

式中, y 、 u 和 ξ 分别代表被控过程的输出变量、控制变量和扰动量, d 为滞后时间, $f(\cdot)$ 为可以描述过程输入输出关系的非线性函数。为了从理论上精确地描述一个非线性系统 f , 人们提出了多种类型的描述非线性模型的方法, 如, 块联模型、基于各种核函数描述的模型以及利用输入输出信号中隐含的过程动态特性所提出的神经网络、支持向量机等数据建模方法等。

3.2 被控过程的机理建模

对于一些比较简单的、物料或能量变化过程以及内在机理比较清楚的被控过程, 一般可以通过静态和动态的物料平衡或能量平衡关系来推导出被控过程的数学模型。

静态的物料(或能量)平衡关系是指单位时间内进入被控过程的物料量(或能量)等于单位时间内从被控过程流出的物料量(或能量)。动态的物料(或能量)平衡关系是指单位时间内进入被控过程的物料量(或能量)减去单位时间内从被控过程流出的物料量(或能量)等于被控过程中物料量(或能量)储存量的变化率。

3.2.1 建立自衡过程的数学模型

被控过程有无自平衡能力, 决定于对象本身的结构, 并与生产过程的特性有关。凡是受到干扰后, 不依靠外加控制作用就能重新达到平衡状态的对象都具有自平衡能力, 否则就是没有自平衡能力。

1. 单容自衡过程的数学模型

所谓单容过程, 是指只有一个储蓄能量的过程。

如图 3-4(a) 所示, 为一个单溶液位被控过程, 其流入量为 q_1 , 改变阀 1 的开度可以改变 q_1 的大小; 其流出量为 q_2 , 改变阀 2 的开度可以改变 q_2 。当流入量 q_1 和流出量 q_2 相等时, 液位 h 保持不变, 处于平衡状态。当流入阀门突然开大, 水的流入量阶跃增多, 液位便开始上升, 随着液位的升高, 水箱内液体的静压力增大, 使水的流出量跟着增多, 这一趋势会使流出量 q_2 和流入量 q_1 再次相等, 从而使得液位 h 达到一个新的平衡状态。可见, 单容水箱是一个自衡过程。自平衡是一种自然形式的负反馈, 好像在过程内部具有比例控制器的作用, 但对象的自平衡作用与系统的控制作用完全不同, 后者是靠控制器施加的控制作用, 而不是

被控过程自身所具有的能力。

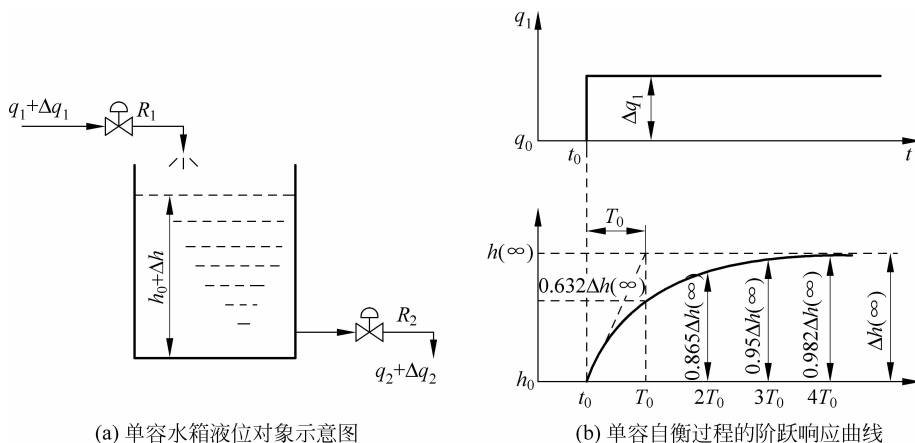


图 3-4 单容水箱过程及其阶跃响应曲线

若将前述水箱的流入量 q_1 作为液位对象的输入变量,液位 h 为其输出变量,则该被控过程的数学模型就是 h 与 q_1 之间的数学表达式。

上述水箱在平衡状态时,流入量 q_1 和流出量 q_2 相等,液位 h 保持不变,即

$$q_{10} = q_{20}, \quad h = h_0 \quad (3-9)$$

当控制阀突然开大一些,液位会逐渐上升,如果流出侧阀门开度不变,则随着液位的升高而流出量逐渐增大,这时根据动态物料平衡关系有

$$q_1 - q_2 = A \frac{dh}{dt} \quad (3-10)$$

或

$$(q_{10} + \Delta q_1) - (q_{20} + \Delta q_2) = A \frac{d(h_0 + \Delta h)}{dt} \quad (3-11)$$

由式(3-9)~式(3-11)可得

$$\Delta q_1 - \Delta q_2 = A \frac{d\Delta h}{dt} \quad (3-12)$$

上述各式中, Δq_1 、 Δq_2 、 Δh 分别表示偏离某一平衡状态 q_{10} 、 q_{20} 、 h_0 的增量; A 为水箱的横截面积。

由流体力学可知,流体在紊流情况下,液位 h 与流出量之间为非线性关系。但为简化起见,经线性化处理,可近似认为 Δq_2 和 Δh 成正比,而与流出阀门的阻力 R_2 成反比,即

$$\Delta q_2 = \frac{\Delta h}{R_2} \quad \text{或} \quad R_2 = \frac{\Delta h}{\Delta q_2} \quad (3-13)$$

式中, R_2 为流出阀门的阻力,称为液阻。

为了求取单容过程的数学模型,需要消去中间变量 q_2 。在这里,介绍两种消去中间变

量的方法,代数代换法和方框图法。

代数代换法 将式(3-13)代入式(3-12)并化简,即可得到微分方程形式的数学模型

$$AR_2 \frac{d\Delta h}{dt} + \Delta h = R_2 \Delta q_1 \quad (3-14)$$

对式(3-14)两端进行拉普拉斯变换,经整理即可得到传递函数形式的数学模型

$$W_o(s) = \frac{H(s)}{Q_1(s)} = \frac{R_2}{AR_2 s + 1} = \frac{K_0}{T_0 s + 1} \quad (3-15)$$

式中, $H(s)$ 、 $Q_1(s)$ 为水箱液位变化和进水流量变化的拉普拉斯变换函数; T_0 为被控过程的时间常数; K_0 为被控过程的放大系数。

由此可见,单容自衡过程的数学模型为一个一阶惯性环节。由式(3-15)可得单容过程的阶跃响应如下

$$\Delta h(t) = K_0 \Delta q_1 (1 - e^{-t/T_0}) \quad (3-16)$$

图 3-4(b)为单容水箱被控过程的阶跃响应曲线。

画方框图法 将式(3-12)、式(3-13)取拉普拉斯变换后,画出图 3-5 所示方框图。然后再利用画出的方框图求取被控过程传递函数形式的数学模型。

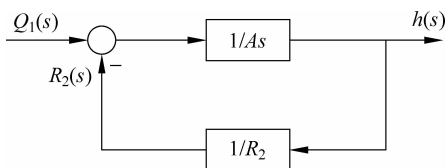


图 3-5 液位过程方框图

被控过程都具有一定储存物料或能量的能力,其储存能力的大小,称为容量,用容量系数来表示,其物理意义是:引起单位被控量变化时,被控过程储存量的变化量。上述单容水箱的容量系数为 A 。

从式(3-15)可知,液阻 R_2 不但影响过程的时间常数 T_0 ,而且影响过程的放大系数 K_0 ,而容量系数 A 仅影响过程的时间常数。

2. 有纯滞后单容自衡过程的数学模型

在工业生产过程中,过程具有纯滞后是经常碰到的问题。当被控量的检测点与产生扰动的地点之间有一段物料传输距离时,就会出现纯滞后。如图 3-6 为典型的具有纯滞后的皮带运输机过程。在该过程中,如果输送皮带秤的扰动发生在电动控制阀,与物料称重传感器相距为 l ,皮带必须经过这一段传输距离后,变化后的重量才会被传感器检测出来,显然流经距离 l 的时间完全是传输滞后造成的,故称其为传输滞后或纯滞后,以 τ_0 表示。

对于前述单容水箱,当进水阀门在距离水箱 l 的地方,如图 3-7 所示,则阀门开度变化产生扰动后,液体要经过流经 l 距离的时间后才流入水箱,使水位发生变化而被检测出来。显然图 3-7 所示的水箱为具有纯滞后的自衡过程,假设该过程的纯滞后为 τ_0 ,那么具有纯滞后 τ_0 的单容水箱的微分方程和传递函数形式的数学模型可分别表示为式(3-17)

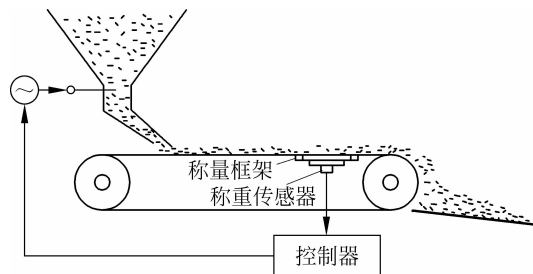


图 3-6 皮带运输机传输过程

和式(3-18)。

$$T_0 \frac{d\Delta h}{dt} + \Delta h = K_0 \Delta q_1 (t - \tau_0) \quad (3-17)$$

$$W_o(s) = \frac{H(s)}{Q_1(s)} = \frac{K_0}{T_0 s + 1} e^{-\tau_0 s} \quad (3-18)$$

在式(3-17)和式(3-18)中, T_0 、 K_0 和 τ_0 分别为被控过程的时间常数、被控过程的放大系数和被控过程的纯滞后时间。

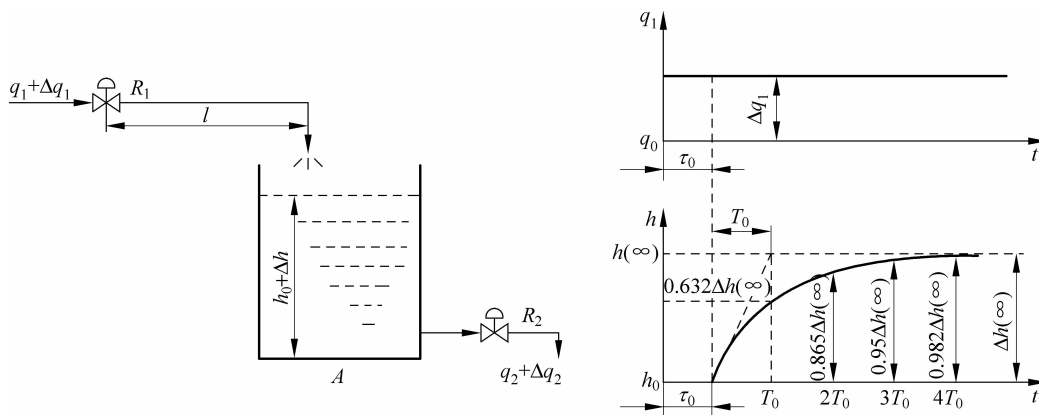


图 3-7 具有纯滞后的单容水箱过程及其阶跃响应曲线

3. 多容自衡过程的数学模型

在工业生产过程中,被控过程往往由多个容积和阻力构成,这种过程称为多容过程。下面以双容水箱为例,介绍建立多容过程数学模型的方法。

如图 3-8(a)所示的由两个单容水箱串联组成的双容过程,若其输入量为 q , 输出量为液位 h_2 , 根据物料平衡关系, 可以列出下列微分方程组

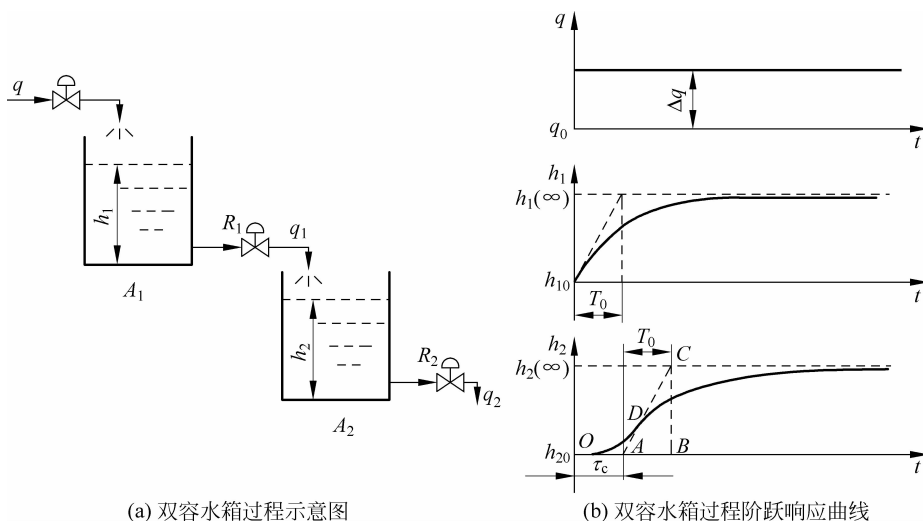


图 3-8 双容水箱过程及其阶跃响应曲线

$$\begin{cases} \Delta q - \Delta q_1 = A_1 \frac{d\Delta h_1}{dt} \\ \Delta q_1 - \Delta q_2 = A_2 \frac{d\Delta h_2}{dt} \\ \Delta q_1 = \frac{\Delta h_1}{R_1} \\ \Delta q_2 = \frac{\Delta h_2}{R_2} \end{cases} \quad (3-19)$$

消去中间变量后可得

$$T_1 T_2 \frac{d^2 \Delta h_2}{dt^2} + (T_1 + T_2) \frac{d\Delta h_2}{dt} + \Delta h_2 = K_0 \Delta q \quad (3-20)$$

式中, A_1 、 A_2 为两只水箱的容量系数; T_1 为第一容积的时间常数, $T_1 = A_1 R_1$; T_2 为第二容积的时间常数, $T_2 = A_2 R_2$; K_0 为双容过程的放大系数, $K_0 = R_2$ 。

将式(3-20)两端进行拉普拉斯变换, 可得双容过程传递函数形式的数学模型为

$$W_o(s) = \frac{H_2(s)}{Q(s)} = \frac{K_0}{T_1 T_2 s^2 + (T_1 + T_2)s + 1} = \frac{K_0}{(T_1 s + 1)(T_2 s + 1)} \quad (3-21)$$

由此可见, 双容自衡过程的数学模型为二阶惯性环节, 其 S 状阶跃响应曲线如图 3-8(b) 所示。由曲线可以看出, 双容过程受到扰动后, 其被控量 h_2 的变化速率并非一开始就最大, 而要经过一段时间以后, 变化速率才达到最大, 这段时间是由于被控过程的两个容积均存在着阻力而造成的, 称其为容量滞后, 用 τ_c 表示。 τ_c 可用作图法求得, 即通过多容被控过程的阶跃响应曲线拐点作切线, 与时间轴相交于 A 点, 与稳态值 $h(\infty)$ 交于 C 点, C 点在时间轴上的投影为 B, 则 AB 即为过程的时间常数, OA 为过程的容量滞后时间 τ_c 。

对于具有纯滞后的多容过程,其传递函数形式的数学模型一般表达式为

$$W_o(s) = \frac{K_0}{(T_1s + 1)(T_2s + 1) \cdots (T_ns + 1)} e^{-\tau_0 s} \quad (3-22)$$

在过程控制中,有些被控过程可以认为 $T_1 = T_2 = \cdots = T_n = T_0$,则多容过程的数学模型可以表示为

$$W_o(s) = \frac{K_0}{(T_0s + 1)^n} e^{-\tau_0 s} \quad (3-23)$$

3.2.2 建立无自衡过程的数学模型

1. 单容无自衡过程的数学模型

如果将图 3-4(a)所示水箱的出口阀门换成定量泵,如图 3-9 所示。这样,其流出量将与液位 h 的变化无关。当流入量 q_1 发生阶跃变化时,液位 h 即发生变化,但由于流出量是不变的,所以水箱液位或者等速上升直至液体溢出,或者等速下降直至液体被抽干,其阶跃响应曲线如图 3-9 所示。

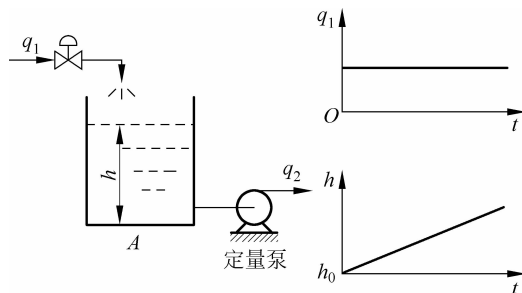


图 3-9 单容无自衡过程及其阶跃响应曲线

图 3-9 所示过程的微分方程为

$$A \frac{d\Delta h}{dt} = \Delta q_1 \quad (3-24)$$

式中, A 为水箱的容量系数。

过程传递函数形式的数学模型为

$$W_o(s) = \frac{1}{T_a s} \quad (3-25)$$

式中, T_a 为过程的积分时间常数, $T_a = A$ 。

当过程具有纯滞后时,其传递函数为

$$W_o(s) = \frac{1}{T_a s} e^{-\tau_0 s} \quad (3-26)$$

2. 多容无自衡过程的数学模型

同理,若图 3-8(a)中第二个水箱的出口阀门变成定量水泵,如图 3-10 所示,那么双容自衡过程就成为了双容无自衡过程,若其输入量为 q ,输出量为液位 h_2 ,则该过程的微分方程组为

$$\begin{cases} \Delta q - \Delta q_1 = A_1 \frac{d\Delta h_1}{dt} \\ \Delta q_1 = A_2 \frac{d\Delta h_2}{dt} \\ \Delta q_1 = \frac{\Delta h_1}{R_1} \end{cases} \quad (3-27)$$

消去式(3-27)的中间变量,得到微分方程形式的数学模型为

$$T_a T_1 \frac{d^2 \Delta h_2}{dt^2} + T_a \frac{d\Delta h_2}{dt} = \Delta q \quad (3-28)$$

式中, A_1 、 A_2 为两只水箱的容量系数; T_1 为第一只水箱的时间常数, $T_1 = A_1 R_1$; T_a 为双容过程的积分时间常数, $T_a = A_2$ 。

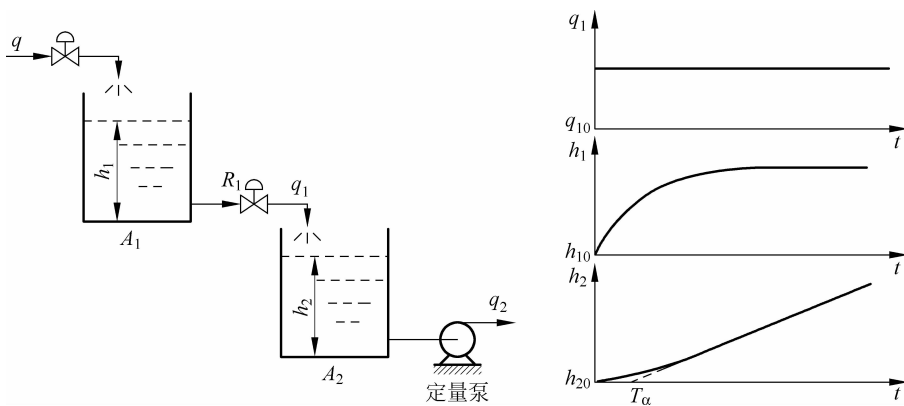


图 3-10 多容无自衡过程及其阶跃响应曲线

对式(3-28)进行拉普拉斯变换,可得上述双容无自衡过程的传递函数为

$$W_o(s) = \frac{1}{T_a s (T_1 s + 1)} \quad (3-29)$$

对于多容具有纯滞后的无自衡过程,则有

$$W_o(s) = \frac{1}{T_a s (T_1 s + 1)(T_2 s + 1) \cdots (T_n s + 1)} e^{-\tau_0 s} \quad (3-30)$$

$$W_o(s) = \frac{1}{T_a s (T_0 s + 1)^n} e^{-\tau_0 s} \quad (3-31)$$

3.3 被控过程的实验建模

上面介绍的机理建模方法具有较大的普遍性,但是多数工业过程的机理较复杂,建立其机理模型非常困难。同时在建模过程中虽然作了一些具有一定实际依据的近似和假设,但仍使得所建立的数学模型不能完全反映过程的实际情况。在这种情况下,利用实验方法——过程辨识与参数估计方法,建立其数学模型是一种有效的手段。实验建模方法主要有响应曲线法、相关统计法和最小二乘方法等。在这里,只介绍一种容易理解,又较常用的响应曲线实验建模法。响应曲线法包括阶跃响应曲线法和矩形脉冲响应曲线法。

3.3.1 测取阶跃响应曲线

将被控过程的输入量作一阶跃变化,同时记录其输出量随时间变化的曲线,称其为阶跃响应曲线。

阶跃响应曲线能直观、完全描述被控过程的动态特性。实验测试方法易于实现,只要使阀门开度作一阶跃变化即可。实验时必须注意以下几点:

(1) 扰动量的大小要合适。如果扰动量太大,会影响生产的正常进行,太小了又可能受干扰信号的影响而失去作用。通常是取控制阀门流入量最大值的10%左右为宜。

(2) 实验应在相同的条件下重复做几次,需获得两次以上的比较接近的响应曲线,减少干扰的影响。

(3) 实验应在阶跃信号作正、反方向变化时分别测出其响应曲线,以检验被控过程的非线性程度。

(4) 在实验前,必须保证被控过程处于稳定的工作状态。

3.3.2 测取矩形脉冲响应曲线

利用阶跃响应曲线可以方便地辨识出被控过程的模型结构,并估计出模型的参数。但在生产过程中,有时不允许存在长时间的阶跃扰动,这时可采用矩形脉冲法。另外,当阶跃信号幅值受生产条件限制而影响过程的模型精度时,也要改用矩形脉冲信号作为过程的输入信号,其响应曲线称为矩形脉冲响应曲线。为了利用阶跃响应曲线辨识模型的方便之处,在测得矩形脉冲曲线后,仍需要将其转换成阶跃响应曲线,如图3-11所示。

为了实现曲线转换,首先将幅值为 Δx ,宽度为 a 的矩形脉冲信号分解为两个方向相反幅值相等的阶跃信号,其一是从 $t=0$ 开始幅值为 Δx 的正阶跃 $x_1(t)$,另一是从 $t=a$ 开始的幅值为 Δx 的负阶跃 $x_2(t)$,即

$$x(t) = x_1(t) + x_2(t) = x_1(t) - x_1(t-a) \quad (3-32)$$

假设被控过程是线性的,则其矩形脉冲响应曲线 $y^*(t)$ 可以看成由 $x_1(t)$ 和 $x_2(t)$ 的阶跃响应曲线 $\bar{y}(t)$ 及 $\bar{y}(t-a)$ 叠加而成,即

$$y^*(t) = \bar{y}(t) - \bar{y}(t-a) \quad (3-33)$$

由上式可见,要获得阶跃信号 $x_1(t)$ 的阶跃响应曲线 $y_1(t)$,可以通过叠加矩形脉冲响应曲线 $y^*(t)$ 和阶跃响应曲线 $\bar{y}(t-a)$ 而实现。即

$$\bar{y}(t) = y^*(t) + \bar{y}(t-a) \quad (3-34)$$

从图 3-11 可以看出,在 $t=0 \sim a$ 时间段内, $\bar{y}(t-a)=0$,即 $\bar{y}(t)=y^*(t)$,阶跃响应曲线 $\bar{y}(t)$ 和脉冲响应曲线 $y^*(t)$ 重合。在 $t>a$ 时, $\bar{y}(t)=y^*(t)+\bar{y}(t-a)$,此时 $\bar{y}(t-a)$ 是已经获得的。这样,就可以由矩形脉冲响应曲线求得完整的阶跃响应曲线。

对于脉冲宽度 a 的选择,应视被控过程的惯性、滞后时间以及被控量的幅值而定。通常,在正式测试前,选几个不同脉冲宽度 a 的信号作几次实验,观察被控量的变化,选择最合适的一次进行测得。

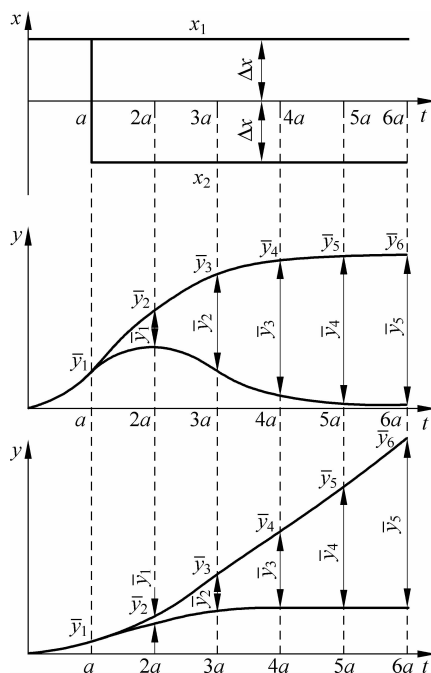


图 3-11 由脉冲响应曲线求取阶跃响应曲线

3.3.3 由阶跃响应曲线辨识被控过程的模型

在工业生产过程中,大多数自衡过程模型常常可以近似地以一阶、二阶以及一阶加时延、二阶加时延特性之一来描述,即

$$\begin{cases} W_o(s) = \frac{K_0}{T_0 s + 1} \\ W_o(s) = \frac{K_0}{T_0 s + 1} e^{-\tau s} \\ W_o(s) = \frac{K_0}{(T_1 s + 1)(T_2 s + 1)} \\ W_o(s) = \frac{K_0}{(T_1 s + 1)(T_2 s + 1)} e^{-\tau s} \end{cases} \quad (3-35)$$

而少数无自衡过程模型,可以描述为

$$\begin{cases} W_o(s) = \frac{1}{T_a s} \\ W_o(s) = \frac{1}{T_a s} e^{-\tau s} \\ W_o(s) = \frac{1}{T_1 s (T_2 s + 1)} \\ W_o(s) = \frac{1}{T_1 s (T_2 s + 1)} e^{-\tau s} \end{cases} \quad (3-36)$$

测取阶跃响应曲线的目的是为了得到表征所测对象的数学模型,为分析、设计控制系统,整定控制器参数或改进控制系统提供必要的参考依据。由阶跃响应曲线确定过程的数学模型,首先就要选定模型的结构,然后再由阶跃响应曲线确定过程的放大系数、时间常数以及时间滞后,就可以得到被控过程的数学模型。下面介绍几种常用的模型参数确定方法。

1. 确定无滞后一阶对象的模型参数

一阶非周期过程比较简单,只需确定放大系数 K_0 及时间常数 T_0 即可获得传递函数模型。

确定静态放大系数 K_0 。利用所测取的阶跃响应曲线估计并绘出被控量的最大稳态值 $y(\infty)$,如图 3-12 所示,放大系数 K_0 为

$$K_0 = [y(\infty) - y(0)] / \Delta x \quad (3-37)$$

确定时间常数 T_0 。由响应曲线起点作切线与 $y(\infty)$ 相交点在时间轴上的投影,就是时间常数 T_0 。由于切线不易作准,从式(3-16)可知 $\Delta y(T_0) = 0.632 K_0 \Delta x = 0.632 \Delta y(\infty)$,所以响应曲线 $\Delta y(t_1) = 0.632 \Delta y(\infty)$ 所对应的时间 t_1 就是时间常数 T_0 ,同理响应曲线 $\Delta y(t_2) = 0.865 \Delta y(\infty)$ 所对应的时间 t_2 是 2 倍时间常数,即 $2T_0$ 。

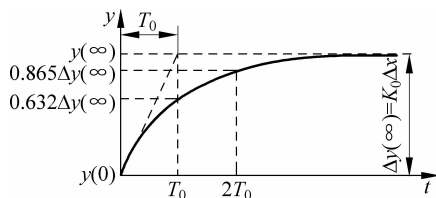


图 3-12 无滞后一阶对象的响应曲线

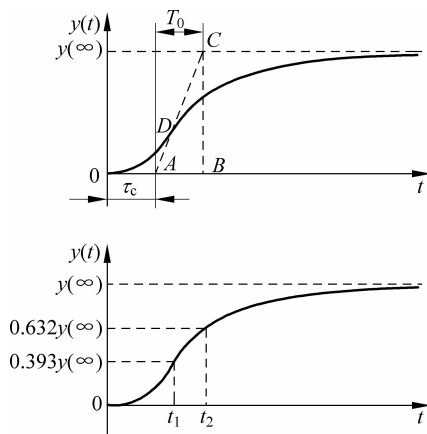


图 3-13 S 状阶跃响应曲线

2. 确定有滞后一阶对象的模型参数

当所测取的响应曲线起始速度较慢,曲线呈 S 状,可近似将此具有滞后的过程视为带纯滞后的一阶非周期过程,将对象的容量滞后也当纯滞后处理,则传递函数模型可以表示为

$$W_o(s) = \frac{K_0}{T_0 s + 1} e^{-\tau_c s} \quad (3-38)$$

对于 S 状的曲线的参数估计,常用两种方法处理。

切线法是一种比较简单的方法,即通过响应曲线的拐点 D 作一切线,在时间轴上的交点为 A ,则 OA 为滞后时间 τ_c ;切线与 $y(\infty)$ 线的交点在时间轴上的投影为 B ,则 AB 即为等效的时间常数 T_0 ,如图 3-13 所示。对象的放大系数 K_0 可按式(3-37)计算。

计算法被控量 $y(t)$ 以相对值表示,即 $y'(t) = \Delta y(t) / \Delta y(\infty)$,当 $t \geq \tau_c$ 或 $t < \tau_c$ 时有

$$y'(t) = \begin{cases} 0 & t < \tau_c \\ 1 - e^{-(t-\tau_c)/T_0} & t \geq \tau_c \end{cases} \quad (3-39)$$

在曲线上选择两个不同的点 $(t_1, y'(t_1))$ 、 $(t_2, y'(t_2))$,建立两个联立方程

$$\begin{cases} y'(t_1) = 1 - e^{-(t_1-\tau_c)/T_0} \\ y'(t_2) = 1 - e^{-(t_2-\tau_c)/T_0} \end{cases} \quad (t_2 > t_1 > \tau_c)$$

对上式两边取对数后,有

$$\begin{cases} -(t_1 - \tau_c)/T_0 = \ln[1 - y'(t_1)] \\ -(t_2 - \tau_c)/T_0 = \ln[1 - y'(t_2)] \end{cases}$$

求解得

$$\begin{cases} T_0 = \frac{t_2 - t_1}{\ln[1 - y'(t_1)] - \ln[1 - y'(t_2)]} \\ \tau_c = \frac{t_2 \ln[1 - y'(t_1)] - t_1 \ln[1 - y'(t_2)]}{\ln[1 - y'(t_1)] - \ln[1 - y'(t_2)]} \end{cases} \quad (3-40)$$

通常选择 $y'(t_1) = 0.393$, $y'(t_2) = 0.632$,因此可得

$$T_0 = 2(t_2 - t_1), \quad \tau = 2t_1 - t_2 \quad (3-41)$$

计算出 T_0 与 τ 后,放大系数 K_0 仍可按式(3-37)求取。

3. 确定无滞后二阶对象的模型参数

对于 S 状的阶跃响应曲线,若对模型精度要求较高,则应采用二阶对象的模型结构,即

$$W_o(s) = \frac{K_0}{(T_1 s + 1)(T_2 s + 1)}$$

式中, K_0 、 T_1 、 T_2 的求法如下:

第一,由式(3-37)求取过程的静态放大系数 K_0 。

第二, T_1 、 T_2 可根据阶跃响应曲线上的两个点来确定,如图 3-14 所示,首先读取 $y(t_1) = 0.4y(\infty)$ 和 $y(t_2) = 0.8y(\infty)$ 所对应的时间 t_1 和 t_2 值,然后利用下

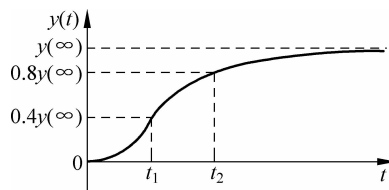


图 3-14 S 状阶跃响应曲线

式计算 T_1 、 T_2 。即

$$\begin{cases} T_1 + T_2 \approx \frac{t_1 + t_2}{2.16} \\ \frac{T_1 T_2}{(T_1 + T_2)^2} \approx 1.74 \frac{t_1}{t_2} - 0.55 \end{cases} \quad \left(0.32 < \frac{t_1}{t_2} < 0.46\right) \quad (3-42)$$

当 $t_1/t_2=0.32$ 时,可采用无滞后的一阶环节来近似,其时间常数为

$$T_0 = \frac{t_1 + t_2}{2.12} \quad (3-43)$$

当 $t_1/t_2=0.46$ 时,可采用下式所示的二阶环节近似,即

$$W_o(s) = \frac{K_0}{(T_0 s + 1)^2}$$

此时,时间常数为

$$T_0 = \frac{t_1 + t_2}{2 \times 2.18} \quad (3-44)$$

当 $t_1/t_2 > 0.46$ 时,则应采用高于二阶环节来近似,即

$$W_o(s) = \frac{K_0}{(T_0 s + 1)^n} \quad (3-45)$$

此时,仍采用上述两个点所对应的时间来确定式(3-45)中的时间常数

$$T_0 \approx \frac{t_1 + t_2}{2.16n} \quad (3-46)$$

式中, n 的值可由表 3-2 查得。

表 3-2 多容过程的 n 与 t_1/t_2 的关系

n	1	2	3	4	5	6	8	10	12	14
t_1/t_2	0.32	0.46	0.53	0.58	0.62	0.65	0.685	0.71	0.735	0.75

4. 确定有滞后二阶对象的模型参数

如式(3-47)所示有滞后二阶对象的数学模型,需要确定的模型参数有 K_0 、 T_1 、 T_2 和 τ 。

$$W_o(s) = \frac{K_0}{(T_1 s + 1)(T_2 s + 1)} e^{-\tau_0 s} \quad (3-47)$$

如图 3-15 所示,通过拐点 D 作一切线,在时间轴上的交点为 A ,则 OA 为滞后时间 τ_0 , $T_A = BF$, $T_B = EF$ 。 T_1/T_2 与 T_B/T_A 有以下关系

$$\frac{T_B}{T_A} = (1+x)x^{\frac{x}{1-x}} \quad (3-48)$$

式中, $x = T_1/T_2$, 且 $T_1 + T_2 = T_B$, 代入式(3-48)可得到 T_1 和 T_2 。

5. 确定无滞后积分环节的模型参数

当被控过程的阶跃响应曲线为一条如图 3-16 所示直线时,其模型结构可用一积分环节

表示,即

$$W_o(s) = \frac{1}{T_a s}$$

式中, T_a 为积分时间常数。 T_a 的值就是直线的斜率。

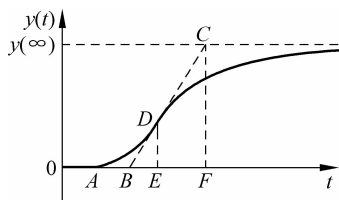


图 3-15 S 状阶跃响应曲线图

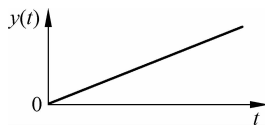


图 3-16 积分环节阶跃响应曲线

6. 确定有滞后积分环节的模型参数

如图 3-17 所示,若被控过程的阶跃响应曲线开始时,变化速度较慢,一段时间后开始等速上升,此时,过程的模型结构可由下式来近似,即

$$W_o(s) = \frac{1}{T_a s} e^{-\tau s}$$

式中, T_a 为积分时间常数, τ 为滞后时间。

在阶跃响应曲线变化速度最大处作切线,交时间轴于 A 点, OA 即为滞后时间 τ , T_a 为切线斜率。

7. 确定有滞后、一阶和积分环节的模型参数

若被控过程的阶跃响应曲线如图 3-18 所示,为了提高模型精度,可采用下式近似无自衡过程的数学模型,即

$$W_o(s) = \frac{1}{T_a s (T_0 s + 1)} e^{-\tau_0 s} \quad (3-49)$$

图 3-18 中, OA 即为纯滞后时间 τ_0 。在阶跃响应曲线变化速度最大处作切线,交时间轴于 B 点, AB 即为时间常数 T_0 , T_a 仍为切线斜率。



图 3-17 有滞后积分环节阶跃响应曲线

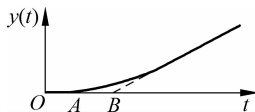


图 3-18 有滞后、一阶和积分环节阶跃响应曲线

3.4 基于神经网络的数据建模

机理建模是从过程系统内在的物理和化学规律出发,利用物料、能量或动量守恒等关系,得到关于过程的数学关系表达式。机理建模方法具有先验性、预估性等优点,然而机理建模要求对研究对象的机理有深刻的理解,如果过程非常复杂,研究过程机理需要投入相当大的资源和较长的时间,而且在多数情况下,必须提出简化假设条件,以使建模问题比较易于处理。对于实际工业过程,机理建模可能代价很高,引入的各种假设条件也会影响模型的精度。因此,对复杂工业过程应用机理建模有较大的局限性。

数据建模方法在了解过程机理的基础上,可在较少的先验知识和假设的条件下进行建模。由于过程的动态特性必然表现在过程的输入输出信号中,因此通过测量过程的输入输出数据,就可以直接利用输入输出数据所提供的信息建立数学模型。神经网络是一种常用的数据建模方法,在工业生产过程非线性建模中应用十分广泛。

3.4.1 人工神经元和人工神经网络

神经网络是一个高度复杂的非线性动力学系统,具有很强的自适应学习能力;具有联想、概括、类比和推理能力;具有大规模并行计算能力、较强的容错能力和鲁棒性;尤其具有独特的实时、并行和强大的信息处理能力。因此,可成功用于数据建模。

1. 人工神经元模型

神经生理学和神经解剖学的研究表明,人脑极其复杂,由一千多亿个神经细胞(也称神经元)交织在一起的网状结构构成。为了模拟生物神经元处理信息的特性,人们提出人工神经元模型。最典型的人工神经元模型如图 3-19 所示。

该神经元的输入输出关系为

$$s_j = \sum_{i=1}^n w_{ij} x_i - \theta_j \quad (3-50)$$

$$y_j = f(s_j) \quad (3-51)$$

式中各变量含义如下:

x_i 为人工神经元的输入,模仿生物神经元模型来自其他神经元的输入信号。

w_{ij} 为人工神经元之间的连结权值,表示第 i 个神经元对第 j 个神经元的影响程度,模仿生物神经元模型的神经链接强度。

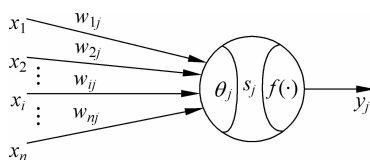


图 3-19 人工神经元模型示意图

θ_j 为人工神经元的阈值,模仿生物神经元模型的阈值。

y_j 为人工神经元的输出,模仿生物神经元模型的输出信号。

$f(\cdot)$ 为人工神经元的转换函数,通常取非线性函数,模拟生物神经元的非线性处理能力。图 3-20 表示了几种常见的转换函数。

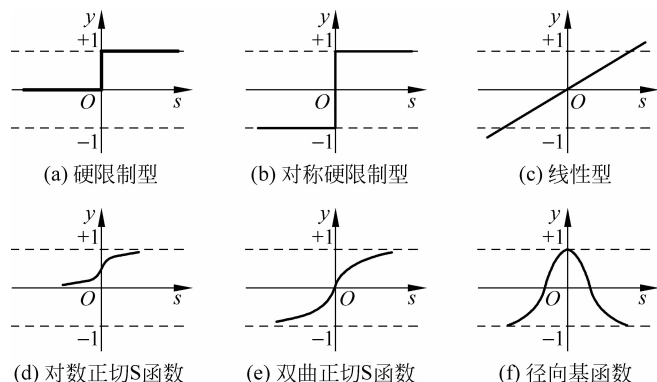


图 3-20 常用的转换函数

2. 人工神经网络

单个神经元本身就其功能而言并不强大,但由多个神经元以各种形式构成的人工神经网络则是一个并行和分布式的网络结构,具有强大的信息处理能力。神经元的不同连接形式构成了具有各种拓扑结构的神经网络,其中前馈网络和反馈网络是两种典型的结构模型。

(1) 前馈神经网络

前馈神经网络结构如图 3-21 所示。可以看出,在前馈网络中,神经元分层排列,可以有多层,同层神经元之间无连接。网络输入模式从输入层起,经过各层变换后传向输出层,信息流向由入到出,无反馈。前馈神经网络是应用最广泛的神经网络。

(2) 反馈神经网络

反馈神经网络如图 3-22 所示,在反馈神经网络中存在着信息的反馈,即输入结点会接受来自输出神经元的信息。这种神经网络是一种反馈动力学系统,它需要工作一段时间才能达到稳定。Hopfield 网络是最简单且应用最广泛的一种反馈网络。

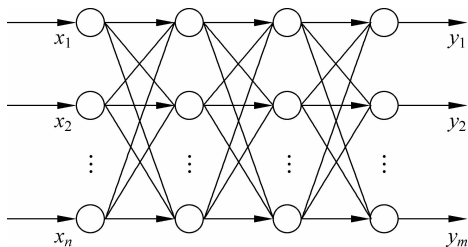


图 3-21 前馈神经网络

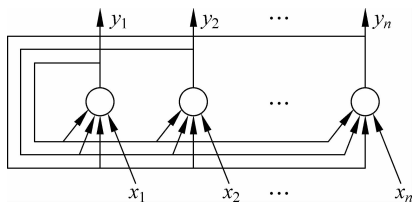


图 3-22 反馈神经网络

3. 神经网络的学习

神经网络的学习功能是神经网络智能特性的重要标志,神经网络通过学习算法,实现了自适应、自组织和自学习能力。目前神经网络的学习算法有多种,按有无“教师”来分类,可分为有监督学习、无监督学习、再励学习等几大类。在有监督学习方式中,网络的输出和期望的输出(即教师信号)进行比较,然后根据两者之间的差异调整网络的权值,最终使偏差减小。在无监督的学习方式中,输入模式进入网络后,网络按照预先设定的规则(如竞争规则)自动调整权值,使网络最终具有模式分类等功能。再励学习是介于上述两者之间的一种学习方式。下面介绍神经网络中常用的几种最基本的学习方法。

(1) Hebb 学习规则

Hebb 学习规则是一种联想式学习方法。生物学家 D. O. Hebb 基于对生物学和心理学的研究,提出了学习行为的突触联系和神经群理论。认为突触前与突触后二者同时兴奋,即两个神经元同时处于激发状态时,它们之间的连接强度将得到加强,这一论述的数学描述为

$$w_{ij}(k+1) = w_{ij}(k) + x_i x_j \quad (3-52)$$

式中, w_{ij} 是神经元 i, j 之间的连接权, x_i, x_j 是神经元 i, j 的状态。

Hebb 学习规则是一种无教师的学习方法,它只根据神经元连接间的激活水平改变权值,因此这种方法又称为相关学习或关联学习。

(2) δ 学习规则

令误差准则函数为

$$E = \frac{1}{2} \sum_{p=1}^P (d_p - y_p)^2 = \sum_{p=1}^P E_p \quad (3-53)$$

其中, d_p 代表期望的输出,为教师信号; y_p 为网络的实际输出。神经网络学习的目的是通过调整权值,使误差准则函数最小。

3.4.2 典型神经网络

1. BP 网络

BP 网络是一个多层前馈网络,因其学习算法采用误差反向传播算法(BP 算法)而得名。它具有和多层感知器相同的网络结构,但是神经元转换函数不再采用二值函数,而是采用连续的 S 型函数。BP 网络结构如图 3-23 所示。

第 1 层为输入层,第 Q 层为输出层,中间各层为隐层。设第 q 层的神经元个数为 n_q ,第 $q-1$ 层的第 j 个神经元与第 q 层的第 i 个神经元的连接权为 w_{ij}^q 。神经元模型如下

$$s_i^{(q)} = \sum_{j=0}^{n_{q-1}} w_{ij}^{(q)} x_j^{(q-1)} \quad (x_0^{q-1} = \theta_i^{(q)}, w_{i0}^{(q)} = -1) \quad (3-54)$$

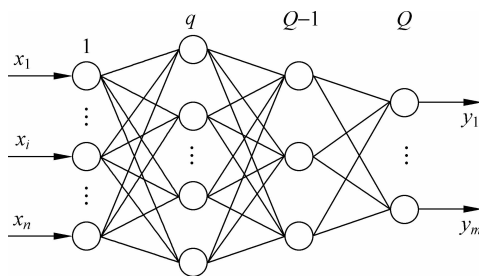


图 3-23 BP 神经网络结构

$$x_i^{(q)} = f(s_i^{(q)}) = \frac{1}{1 + e^{-\mu s_i^{(q)}}} \quad (3-55)$$

其中 $i=1,2,\dots,n_q$; $j=1,2,\dots,n_{q-1}$; $q=1,2,\dots,Q$ 。

给定输入输出样本集,如何利用该样本集对 BP 网络进行训练,即调整网络的连接权系数,使网络能够表达样本所给定的输入输出映射关系。Rumelhart 提出的误差反向传播(back propagation, BP)学习算法,解决了这一问题。BP 算法的学习过程由信息正向传播和误差反向传播两部分组成。在正向传播过程中,计算各层神经元的状态。输入信息从输入层经各隐含层逐层处理,并传向输出层,每层神经元(节点)的状态只影响下一层节点的状态。如果输出层的状态与期望的输出不一致,则转入反向传播过程,将误差信号沿原来的连接通路返回,同时修改各层神经元的权值,权值的不断调整会使网络误差越来越小。

BP 网络本质上具有对任意非线性映射关系进行逼近的能力,并且采用的是全局逼近,因而 BP 网络有较好的泛化能力。BP 网络在控制领域得到了非常广泛的应用,但是它也存在一些缺点,主要包括:①收敛速度慢。对于给定的样本集,目标函数 E 是全体连接权系数 $w_{ij}^{(q)}$ 的函数,因此要寻优的参数个数比较多,也就是说,目标函数 E 是关于连接权的一个非常复杂的超曲面,这就导致算法收敛速度慢。②局部极值问题。目标函数 E 关于连接权的超曲面可能存在多个极值点,按照上面的寻优算法,它一般收敛到初值附近的局部极值。③难以确定隐层数和各隐层节点数目。原理上只要有足够多的隐层和节点就可以实现复杂的映射关系,但是如何根据特定的问题来具体确定网络的结构尚无很好的方法,仍需凭借经验和试凑的方法。

2. RBF 网络

径向基函数(radial basis function, RBF)神经网络是由 J. Moody 和 C. Darken 在 20 世纪 80 年代末提出的一种神经网络。其结构如图 3-24 所示。它是具有单隐层的 3 层前馈网络,输入层节点只是传递信号到隐层,隐层节点的转换函数可采用高斯基函数、多二次函数、逆多二次函数、薄板

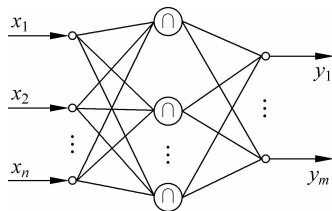


图 3-24 RBF 神经网络示意图

样条函数等。网络输出是隐层节点输出的线性加权和。

隐层节点的高斯基转换函数具有如下形式

$$\mu_j = \exp\left[-\frac{(X - C_j)^T(X - C_j)}{2\sigma_j^2}\right] \quad j = 1, 2, \dots, N_h \quad (3-56)$$

其中, μ_j 是第 j 个隐层节点的输出, $X = (x_1, x_2, \dots, x_n)^T$ 是输入样本, C_j 是高斯函数的中心值, σ_j 是标准差, N_h 是隐层节点数。可以看出, 隐层节点的输出范围在 0 和 1 之间, 输入样本越靠近节点的中心, 节点的输出越大。

RBF 网络的输出节点状态为隐层节点输出的线性组合, 即

$$y_i = \sum_{j=1}^{N_h} w_{ij} \mu_j \quad i = 1, 2, \dots, m \quad (3-57)$$

RBF 网络的学习过程分为两个阶段: ①根据所有的输入样本决定隐层节点的高斯函数的中心值 C_j 和标准差 σ_j 。可随机地选取 l 个输入样本作为隐层节点的中心。但更常用的方法是采用聚类分析的方法, 对输入样本聚类, 将聚类中心作为隐层节点高斯函数的中心值。最常用的一种聚类方法是 K-Means 法。②确定隐层参数后, 根据样本, 利用最小二乘原则, 求输出层的权值 w_{ij} 。

从理论上讲, RBF 网络和 BP 网络一样可近似任何的连续非线性函数。两者的主要差别在于使用不同的转换函数。BP 网络隐层节点使用 Sigmoid 函数, 其函数值在输入空间中无限大的范围内为非零值, 而 RBF 网络中的转换函数是径向基函数, 其值在输入空间中有限范围内为非零值, 因而 RBF 网络是局部逼近的神经网络, BP 网络则是全局逼近网络。BP 网络在训练过程中, 每一次样本学习都要重新调整网络的所有权值, 致使其收敛速度慢; 而 RBF 网络在训练时, 由于其局部逼近的特点, 在学习某一样本时, 只有少数影响该样本输出的权值进行调整, 因此学习速度快。因此 RBF 网络更适合于实时控制的要求。

3. 其他典型神经网络

CMAC 网络是 J. S. Albus 于 1975 年最先提出来的, 它是小脑模型关节控制器 (cerebellar model articulation controller) 的简称, 是仿照小脑如何控制肢体运动的原理而建立的神经网络模型。CMAC 本质上是一种用于映射复杂非线性函数的查表技术。CMAC 是基于局部学习的神经网络, 它把信息存储在局部结构上, 使每次修正的权很少, 在保证函数非线性逼近性能的前提下, 学习速度快, 适合于实时控制。CMAC 网络具有一定泛化能力, 即相近输入产生相近输出。CMAC 网络学习时, 对数据出现的次序不敏感。CMAC 网络在机器人控制中得到了广泛的应用。

1986 年美国物理学家 J. J. Hopfield 利用非线性动力学系统理论中的能量函数方法研究反馈人工神经网络的稳定性, 提出了 Hopfield 神经网络模型。Hopfield 网络是一个动态的反馈网络, 其输入是网络的状态初值, 输出是网络的稳定状态。根据网络的输出是离散量还是连续量, Hopfield 网络可分为离散和连续两种。离散 Hopfield 网络实质上是一个离散

的非线性动力学系统。它可以实现联想记忆的功能。若将系统稳态视为一个记忆样本,那么网络从初态朝稳态收敛的过程便是寻找记忆样本的过程。初态可认为是给定样本的部分信息,网络改变的过程可认为是从部分信息找到全部信息,从而实现联想记忆的功能。连续 Hopfield 网络可实现优化计算的功能。若将稳态与某种优化计算的目标函数相对应,并作为目标函数的极小点,那么初态朝稳态收敛的过程便是优化计算过程,该优化计算是在网络演变过程中自动完成的。

3.4.3 基于 RBF 神经网络的数据建模方法

在 RBF 神经网络中,输出层和隐含层所完成的任务是不同的,因而学习策略也不相同。输出层是对线性连接权进行调整,采用的是线性优化策略,因而学习速度较快;隐含层是对激励函数的参数进行调整,采用的是非线性优化策略,相对来说较慢。RBF 神经网络的主要训练方法有 Poggio 训练方法、局部训练方法、监督训练方法、正交最小二乘训练方法以及各种改进的训练方法,其中局部训练方法为 RBF 神经网络使用较多的学习算法之一。

局部训练方法是指 RBF 神经网络中每个隐含层单元的学习是独立进行的,RBF 的中心是可以移动的,并通过自组织学习确定其位置。输出层的线性连接权则通过有监督学习规则计算。因此,这是一种混合的学习方法。自组织学习部分是在某种意义上对网络资源进行分配,学习目的是使 RBF 的中心位于输入空间重要的区域。

RBF 中心的选择可以采用 K-均值聚类算法。这是一种无监督的学习方法,算法的具体步骤如下:

步骤 1 初始化聚类中心 $C_j (j=1,2,\dots,N_h)$,一般是从输入样本 $x_i (i=1,2,\dots,N)$ 中选择 N_h 个样本作为聚类中心。

步骤 2 将输入样本按最临近规则分组,即将 $x_i (i=1,2,\dots,N)$ 分配给中心 $C_j (j=1,2,\dots,N_h)$,即 $x_i \in \theta_j$,且满足

$$d_i = \min(\|x_i - C_j\|) \quad i=1,2,\dots,N; j=1,2,\dots,N_h \quad (3-58)$$

式中, d_i 是最小欧氏空间距离。

步骤 3 计算 θ_j 中样本的平均值(即聚类中心 C_j)

$$C_j = \frac{1}{m_j} \sum_{x_i \in \theta_j} x_i \quad (3-59)$$

式中, m_j 是 θ_j 中的输入样本数。按以上步骤计算,直到聚类中心的分布不再变化。RBF 的中心确定以后,如果 RBF 是高斯函数,则计算其均方差。这样隐含单元的输出就可以得到。

输出层线性连接权的计算可以采用误差校正学习算法,如最小二乘法(LS)。此时,隐含层的输出就是 LS 算法的输入。

下面以某生物发酵过程中的菌体浓度建模来说明 RBF 神经网络的应用。菌体浓度 X 是发酵过程中的关键生化变量,但很难在线测量,可以利用过程中的其他数据信息来进行建模以获得菌体浓度的估计值 $\hat{X}$ 。通过机理分析,可以选取溶解氧浓度 C_O 、溶解氧浓度变化率 $\dot{C}_O$ 、二氧化碳释放率 CER 、发酵液 pH 值 pH 、发酵液温度 T 、发酵罐进气流量 Q 及搅拌转速 N 为菌体浓度预测模型的输入信息,建立如式(3-60)所示的菌体浓度预测模型。即

$$\hat{X} = f(C_O, \dot{C}_O, CER, pH, T, Q, N) \quad (3-60)$$

将所选取的模型输入信息作为输入,菌体浓度的估计值 $\hat{X}$ 作为输出的基于 RBF 神经网络的菌体浓度数据模型结构如图 3-25 所示。

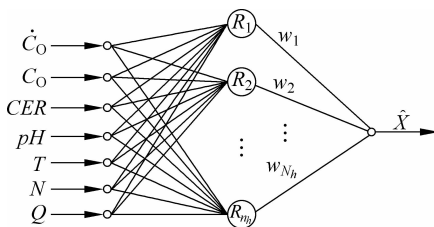


图 3-25 RBF 神经网络菌体浓度数据模型

菌体浓度数据模型的输入节点数为所选辅助变量个数 7,令输入变量表示为 $\mathbf{x} = [\dot{C}_O, C_O, CER, pH, T, N, Q]^T = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]^T$ 。隐含层径向基函数有多种选择,但理论和数值分析表明,隐含层径向基函数的选择对 RBF 神经网络的性能影响不大。高斯函数表示简单且解析性好,因此隐含层径向基函数选用高斯函数

$$\mu_j = \exp\left[-\frac{(\mathbf{x} - \mathbf{C}_j)^T(\mathbf{x} - \mathbf{C}_j)}{2\sigma_j^2}\right] \quad j = 1, 2, \dots, N_h \quad (3-61)$$

本例中,隐含层节点数 N_h 可由交叉验证法确定,RBF 神经网络的输出层节点个数 $n_o = 1$ (菌体浓度)。隐含层与输出层之间的连接权值向量用 $\mathbf{w}$ 表示,即 $\mathbf{w} = [w_1, w_2, \dots, w_{N_h}]^T$ 。令 $\mathbf{u} = [u_1, u_2, \dots, u_{N_h}]^T$,则可得数据模型的菌体浓度输出为

$$\hat{X} = \sum_{j=1}^{N_h} w_j u_j = \mathbf{u}^T \mathbf{w} \quad (3-62)$$

综上,在获得训练样本数据后,首先需要对数据进行预处理,数据预处理主要是指对辅助变量数据进行平滑滤波处理、归一化处理和对样本数据进行异常数据的检测、剔除等,进而提高数据质量,保证数据模型的描述精度;其次,利用神经网络的训练算法确定所建立的数据模型的结构,即确定网络隐含层节点数以及网络中各个未知参数等;最后,利用测试数据验证数据模型的泛化性,如模型精度无法满足预测精度要求,需要重新训练模型,直至满足要求为止。

3.5 数据与机理相结合的建模方法

单纯的数据建模不依赖于机理模型,而仅仅根据生产过程中积累的大量数据进行建模,对数据的依赖性较强,在模型训练数据之外又容易出现违背客观规律的情况;而另一方面,对于一部分复杂工业生产过程,我们并非对其机理知识一无所知,它们往往已经存在比较成熟的机理模型,例如:生物发酵过程,精苯精馏过程等,只是在已知的机理模型中有时会存在一些难以确定的未知参数,或者机理模型本身受各种因素限制与实际的工业生产过程特性之间存在一定的差异。因此将机理与数据结合进行混合建模成为许多复杂工业生产过程建模的理想方法。

混合建模过程中通常会采用简化的机理模型,结合各种观测数据处理的方法,以求得既能反映过程本质又能解释过程现象的模型。因此,此种模型在工业过程的建模中比较常用。在混合建模的实际应用中,根据机理模型与数据模型所起的作用与连接方式的不同,又可将混合模型分为串联混合模型、并联混合模型以及串并联混合模型。

1. 数据与机理模型相结合的串联混合模型

串联结构混合模型是以数学描述的复杂工业生产过程机理模型为基础,以数据驱动的经验模型去估计输出模型中不可测的中间变量或参数。它们的结构呈串连形式,串联结构混合模型示意如图 3-26 所示。该种串联结构混合模型适用于过程机理模型结构已知(精确),但存在未知模型参数的情况。

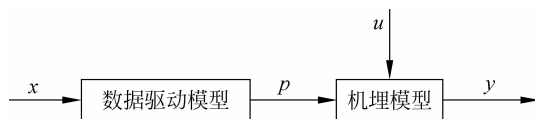


图 3-26 串联结构混合模型示意图

2. 数据与机理模型相结合的并联混合模型

并联结构混合模型是以数学描述的复杂工业生产过程机理模型为基础,以数据驱动的经验模型去建立机理模型无法或难于实现的未建模动态,补偿机理模型的建模误差。它们的结构呈并联形式,并联结构混合模型示意图如图 3-27 所示。该种并联结构混合模型适用于过程机理模型部分已知、存在未建模动态的情况。

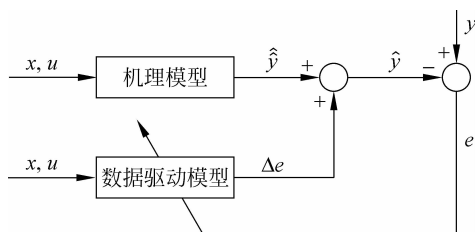


图 3-27 并联结构混合模型示意图

3. 数据与机理相结合的串并联混合模型

图 3-28 所示的串并联结构比串联结构或并联结构的模型更能精确描述复杂工业生产过程的特性。串并联结构的模型中,通过串接的数据驱动模型实现机理模型中不可测的中间变量或未知参数的估计,同时再利用并接的数据驱动模型来补偿机理模型的未建模动态。该种串并联结构混合模型适用于过程机理模型部分已知、存在未知模型参数,且存在未建模动态的情况。

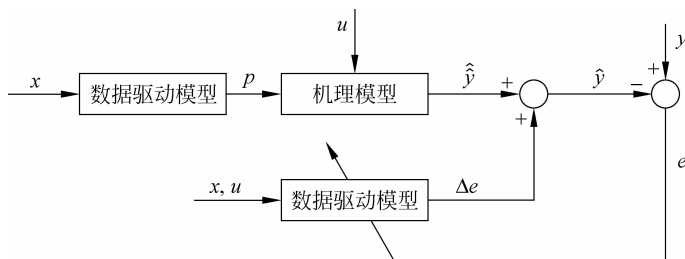


图 3-28 串并联结构混合模型示意图

总之,在并联或串并联的混合模型结构中,并联的数据驱动模型的作用是机理模型的附加校正,它的输出是机理模型估计输出与实际输出的偏差。

工业过程的机理模型具有明显的物理意义,但往往由于各种简化而导致模型描述精度大大下降;数据模型通常具较强的数据拟合能力,但却不具有明显的物理意义;机理和数据相结合的混合模型能够吸取两者的优点,既能反映实际系统的主要规律,又能体现未知扰动或不确定性对实际系统的影响。混合模型的实质是用成熟的机理模型保证混合模型的物理意义、工业可用性和外推能力,同时采用数据模型对机理模型未精确描述的过程特性部分进行修正,从而提高模型的准确性能,进而减少模型的预测误差。因此,数据与机理相结合的混合模型已经在很多领域得到了成功的应用。

本章小结

建立被控过程的数学模型,对正确设计过程控制系统的控制方案具有重要的指导意义。

在本章中,首先介绍了被控过程的基本概念、被控过程的分类、被控过程数学模型的表示方法等,然后以单容和多容水箱为例,讲解了自衡过程和无自衡过程的机理建模方法,并针对复杂工业过程难以实现机理建模的问题,介绍了比较常用又简单易懂的实验建模方法以及利用神经网络进行数据建模的方法,最后对基于机理与数据的混合建模方法进行了简单介绍。

思考题与习题

1. 举例说明什么是自衡过程和无自衡过程? 自衡过程和无自衡过程的阶跃响应曲线各有什么特点?

2. 什么是控制通道和扰动通道? 什么是单容过程和多容过程? 什么是纯滞后和容量滞后?

3. 什么是阶跃响应曲线和矩形脉冲响应曲线? 如何利用矩形脉冲响应曲线获得阶跃响应曲线?

4. 某被控过程的传递函数为 $W_o(s) = Y(s)/X(s) = 20/(T_0s + 1)$, 要求:

① 求出该被控过程的单位阶跃响应;

② 在同一坐标系中,分别画出 $T_0 = 1$ 和 $T_0 = 2$ 的单位阶跃响应曲线。

5. 图 3-29 所示单容水箱的流入量为 q , 流出量为 q_1 、 q_2 , 液位 h 为被控参数, A 为水箱的容量系数, 设 R 、 R_1 、 R_2 均为线性液阻。要求:

① 列出对象的微分方程组;

② 画出对象的方框图;

③ 求出被控对象的传递函数 $W_o(s) = H(s)/Q(s)$;

④ 求出被控对象的单位阶跃响应。

6. 图 3-30 所示为两只串联工作的水箱。该过程的流入量为 q , 流出量为 q_2 , 两只水箱的容量系数分别为 A_1 、 A_2 , 设 R 、 R_1 、 R_2 均为线性液阻。要求:

① 列出对象的微分方程组;

② 画出对象的方框图;

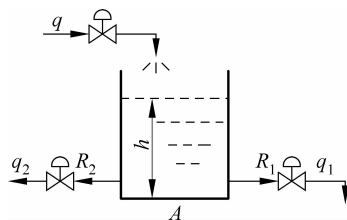


图 3-29 单容水箱过程

③ 求出被控对象的传递函数 $W_o(s) = H_2(s)/Q(s)$ 。

7. 某被控过程的阶跃响应曲线如图 3-31 所示, 试求出该被控过程的放大系数、时间常数以及纯滞后时间。

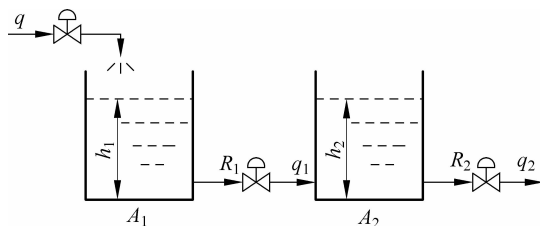


图 3-30 两只串联水箱过程

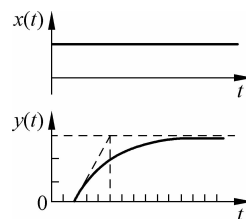


图 3-31 某被控过程阶跃响应曲线

8. 说明利用神经网络和机理知识进行串并联混合建模的基本原理。