

Dead-time Processes

Dead times appear in many processes in industry and in other fields, including economical and biological systems. They are caused by some of the following phenomena: (a) The time needed to transport mass, energy or information; (b) the accumulation of time lags in a great number of low-order systems connected in series; and (c) the required processing time for sensors, such as analysers; controllers that need some time to implement a complicated control algorithm or process.

Dead times introduce an additional lag in the system phase, thereby decreasing the phase and gain margin of the transfer function making the control of these systems more difficult.

This chapter gives an introduction to the modelling, analysis and control of dead-time systems. Several ideas will be introduced and will be discussed in the following chapters.

2.1 Dead-time Systems: Some Case Studies

In this first section, some examples of dead-time systems are presented. These examples come from different fields and show that dead times are present almost everywhere.

2.1.1 A Heated Tank with a Long Pipe

Consider a water heater system such as the one shown in Fig. 2.1. The water is heated in the tank using an electric resistor and driven by a pump along a thermally insulated pipe to the output of the system. The control input is the power W at the resistor and the plant output is the temperature T at the end of the pipe. A linear model of the process can be obtained using a simple step-test identification procedure close to an operation point W_0, T_0 . When a positive step is applied at W , the temperature inside the tank starts to

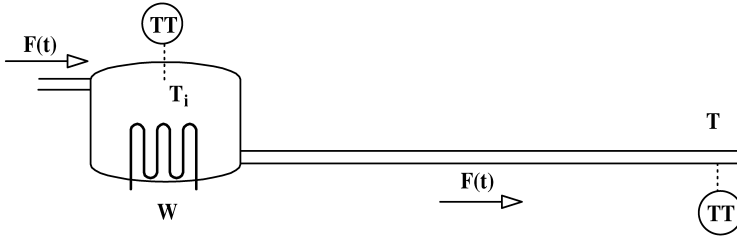


Fig. 2.1. A heated tank and a long pipe

increase. As the pipe is full of water at the initial temperature T_0 , this change is not immediately perceived at the output and it is necessary to wait until the hot water reaches the end of the pipe before it is noticed. Thus, after a dead time, defined by the flow and the length of the pipe, the output temperature T starts to rise with the same dynamics as the temperature inside the tank.

When a constant flow of water F is used, the dead time L can be estimated using F and the volume of the pipe V as

$$L = \frac{V}{F}.$$

Figure 2.2 shows the behaviour of T when a step is applied at W . In this simulated situation, the power W (dashed line) changes from 40% to 50% at $t = 1$ and the temperature (solid line) increases from 55% to 65%. Note that the temperature inside the tank T_i (dotted-dashed line) starts rising at $t = 1$ s, while the temperature at the end of the pipe only reacts at $t = 6$ s, thus, there is a dead time of 5 s due to the time needed for mass transportation. Therefore, it is possible to relate the two temperatures $T_i(t) = T(t + 5)$.

Suppose now that a linear model is used to represent the dynamic relationship between the variations on T_i (ΔT_i) and the variations on W (ΔW). The transfer function between ΔT_i and ΔW is given by

$$G(s) = \frac{\Delta T_i(s)}{\Delta W(s)} \Rightarrow \Delta T_i(s) = G(s)\Delta W(s).$$

If a generic dead time L is considered, and the Laplace transform is used ($\mathcal{L}\{x(t + L)\} = e^{Ls}\mathcal{L}\{x(t)\}$), it follows that

$$\Delta T_i(s) = e^{Ls} \Delta T(s) \Rightarrow \Delta T(s) = \Delta T_i(s)e^{-Ls}.$$

Thus

$$\frac{\Delta T(s)}{\Delta W(s)} = G(s)e^{-Ls} \quad L > 0,$$

which is the linear model most used to represent the behaviour of dead-time processes.

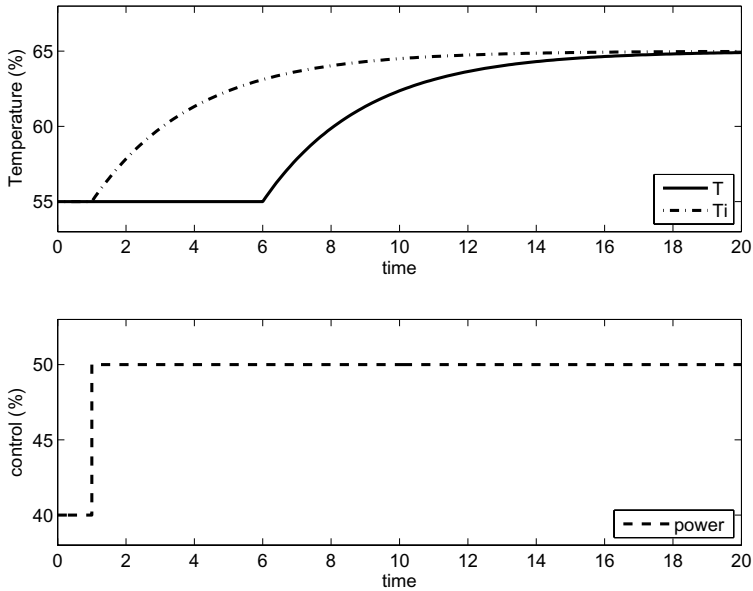


Fig. 2.2. Step response of the system: T_i (dotted-dashed line), T (solid line) and W (dashed line)

2.1.2 Variable Dead Time: Temperature Control at a Solar Plant

The transportation of fluids is a very common dead-time process in industry. An interesting dead-time control example is the temperature control in a distributed solar collector field such as the ACUREX field of the Solar Energy Platform in Almería (Spain). A schematic diagram of the field is given in Fig. 2.3.

Solar radiation is used to heat oil inside a long pipe that passes through the focal point of a set of solar collectors with parabolic mirrors. The heated oil is then pumped to a storage tank. The objective of the control system is to maintain the outlet oil temperature at a desired level in spite of disturbances such as changes in the solar irradiance level (caused by daily variations and passing clouds), the mirror reflectivity or the inlet oil temperature. Since solar radiation cannot be controlled, this can only be achieved by adjusting the flow of the oil and the daily solar power cycle characteristic is such that the oil flow has to change substantially during operation. This leads to significant variations in the dynamic characteristics of the field such as the response rate and the dead time, which cause difficulties in obtaining adequate performance over the operating range with a fixed parameter controller.

Some of the plant operating modes require the temperature of the oil entering the top of the thermal storage tank to be controlled. The considerable length of the pipe joining the output of the collector field to the top of the

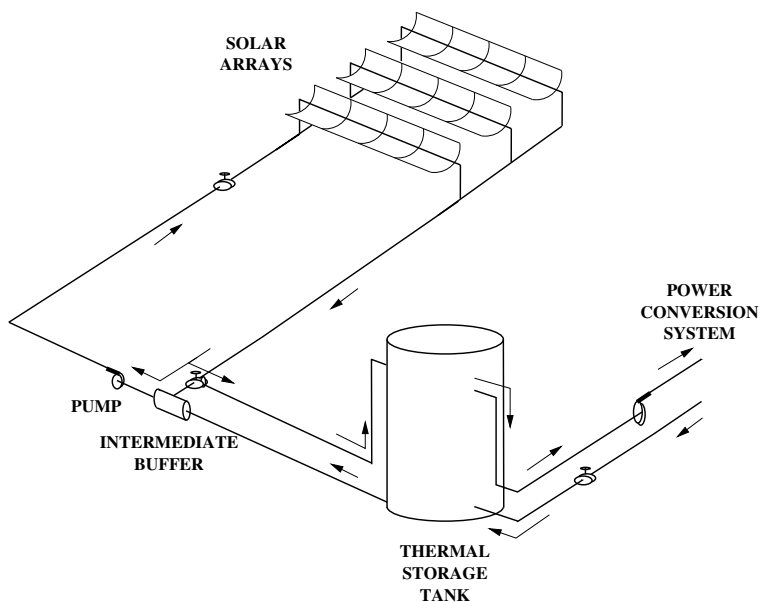


Fig. 2.3. Schematic diagram of the ACUREX field

tank introduces a large dead time within the control loop, which depends on the value of the flow.

The plant can be described by a set of nonlinear distributed parameter equations describing energy and mass balance [16, 12]. Thus, the dead time in this process is also caused by the effect of the distributed dynamics.

The dynamic behaviour of the process can be seen in Fig. 2.4 for a sequence of steps in the control action. Both the outlet temperature of the collector field and the inlet temperature at the top of the storage tank are shown. As can be seen, a varying dead time is present in this process because the dead time, as in the previous example, is a function of the oil flow (see Exercise 2.1). For example, at low temperature close to 225°C the flow is around 8 l/s and the dead time is small while when the system is operating around 255°C , the corresponding flow is approximately 3 l/s and the dead time increases.

2.1.3 High-order Systems

In many cases dead time is caused by the effect produced by the accumulation of a large number of low-order systems. Consider, for instance, a set of n equal cylinder atmospheric tanks as shown in Fig. 2.5. In this system the output flow of tank i (F_{iO}) feeds tank $i + 1$; that is, the input flow of tank $i + 1$ is $F_{(i+1)I} = F_{iO}$. When the tank levels are close to an operating point the

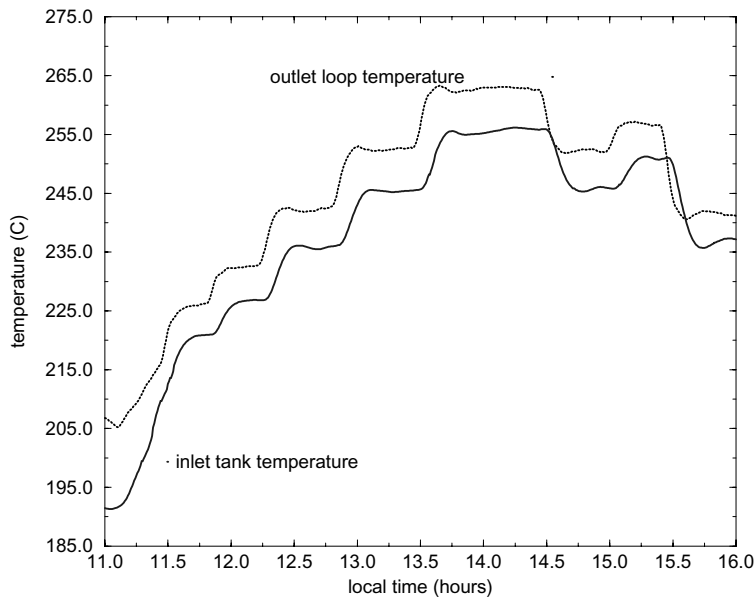


Fig. 2.4. Behaviour of the outlet and inlet tank temperatures

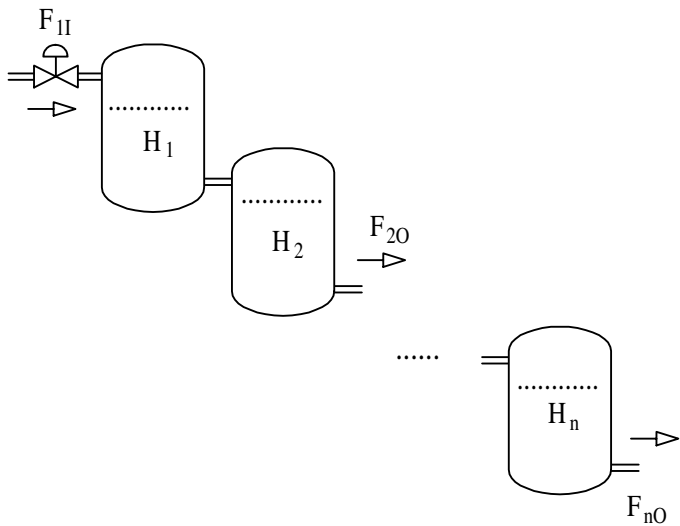


Fig. 2.5. A series of tanks

dynamic behaviour of the level in each tank H_i can be modelled by a linear system

$$A \frac{dH_i}{dt} = F_{iI} - F_{iO},$$

$$F_{iO} = K H_i,$$

where A is the area of the base of the tank and K is a constant that depends on the tank characteristics. Thus, the transfer function relating the input flow in tank i and its level is

$$H_i(s) = \frac{1/K}{Ts + 1} F_{iI}(s), \quad T = A/K.$$

For tank 1

$$H_1(s) = \frac{1/K}{Ts + 1} F_{1I}(s),$$

and for tank 2

$$H_2(s) = \frac{1/K}{Ts + 1} F_{2I}(s) = \frac{1/K}{Ts + 1} F_{1O}(s) = \frac{1/K}{Ts + 1} K H_1(s).$$

Then, using the expression of $H_1(s)$ it follows

$$H_2(s) = \frac{1}{Ts + 1} \frac{1/K}{Ts + 1} F_{1I}(s) = \frac{1/K}{(Ts + 1)^2} F_{1I}(s).$$

If this procedure is applied recursively the transfer function ($P(s)$) relating $F_{1I}(s)$ with the level in tank n ($H_n(s)$) is

$$H_n(s) = P(s) F_{1I}(s) = \frac{K_e}{(Ts + 1)^n} F_{1I}(s), \quad K_e = 1/K.$$

As a numerical example, consider the step response of a system with $n = 8$, $K = 2$ and $T = 1$ s. In this case

$$P(s) = \frac{0.5}{(s + 1)^8}.$$

Figure 2.6 shows the dynamic behaviour when a change of 20% in the level variation is desired (from 60% to 80%). As can be seen, an apparent dead time of approximately 2 s appears in the step response of the system. The MATLAB® code for this example is:

MATLAB® code for the computation of $P(s)$ and the step response <pre>% data n=8; Ke=1/2; T=1; % transfer function Pb=tf(1,[T 1]); P=Ke*Pb ^ n; % step response from operating point 60%, input 40% y0=60; u0=40; [y,t]=u0*step(P); y=y0+y; plot(t,y);</pre>

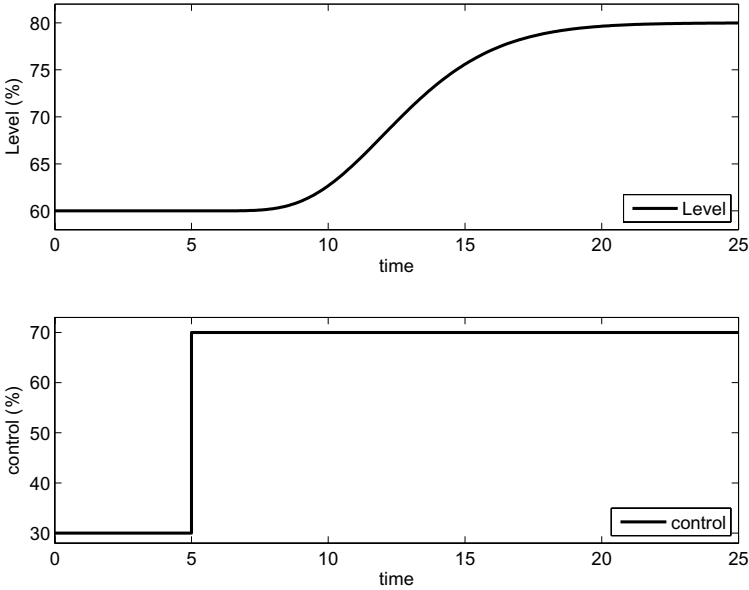


Fig. 2.6. Step response of the level of the 8th tank

We can generalise the previous analysis for any process having N first-order elements in series, each having a time constant L/N [30]. That is, the resulting transfer function (a unitary gain is considered for simplicity) is

$$G(s) = \frac{1}{(1 + \frac{L}{N}s)^N}.$$

Changing the value of N from 1 to ∞ the response shifts from exact first-order to pure dead time (equal to L)

$$e^{-Ls} = \lim_{N \rightarrow \infty} \frac{1}{(1 + \frac{L}{N}s)^N}.$$

When one time constant is much larger than the others (as in many processes), the smaller time constants work together to produce a lag that acts as pure dead time. In this situation the dynamical effects are mainly due to this larger time constant. It is therefore possible to approximate the model of a very high-order, complex, dynamic process with a simplified model consisting of a first-order process combined with a dead-time element. This type of model is analysed in detail in Sect. 2.3.

Several industrial processes have the “dead-time effect” produced by the accumulation of a great number of low-order systems. An interesting case is presented in the following example.

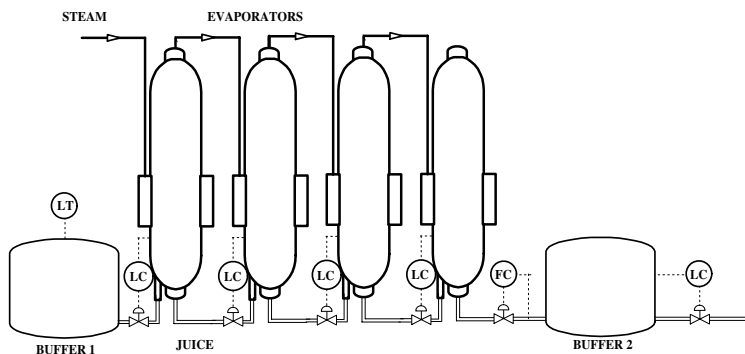


Fig. 2.7. Structure of the evaporator unit

2.1.4 Control Level in an Evaporator Section of a Sugar Factory

The first stage in sugar production is the extraction of sucrose from beets or cane after which a juice with impurities is obtained. These impurities must be removed before pure sucrose can be crystallised. The purified juice contains less than 20% solids, and consequently the juice must be concentrated by evaporating as much water as possible. Later, sugar is crystallised from the concentrated juice by continuing to evaporate water in vacuum pans. Finally, this sugar is dried and packed.

Evaporation is the stage in which the water contained in a juice with low sugar concentration is eliminated in order to obtain a juice with a higher sugar concentration. Evaporation can be carried out in one or several evaporators. When working with high flows and the cost of the steam is high, a chain of evaporators is usually used. In this configuration, the product to be concentrated passes in series from one evaporator to the next. The steam produced in the evaporating process of one of the evaporators is used for heating the next one; only the first evaporator receives steam directly from the boiler. This is known as a multiple effect system. This configuration requires decreasing pressures in order to have decreasing boiling points. The advantage of this multiple stage system is basically the more efficient use of the steam. Consider the arrangement shown in Fig. 2.7. This configuration presents four evaporators. An important control objective in this configuration is to stabilise the level of the buffer tank at the inlet of juice [89]. To achieve this, the extraction flow is controlled at the last stage. The process presents a large dead time because there are four evaporators between the actuating point and the controlled level. Note that changes in the extraction flow cause a variation in the tank level after modifying all the intermediate stages, as each stage has a local control. Furthermore, the storage tank has integral dynamics. Thus, it is possible to describe the dynamic behaviour of the level as a slow process with integral action. Figure 2.8 shows the evolution of the level of the storage tank when a step change in the flow juice from

40% to 45% has been applied at $t = 100$ s. Although the complete model of the evaporator unit is a complex nonlinear system, a simple model can be used to approximate the behaviour close to the operating point using an integrator, a velocity gain and a dead time.

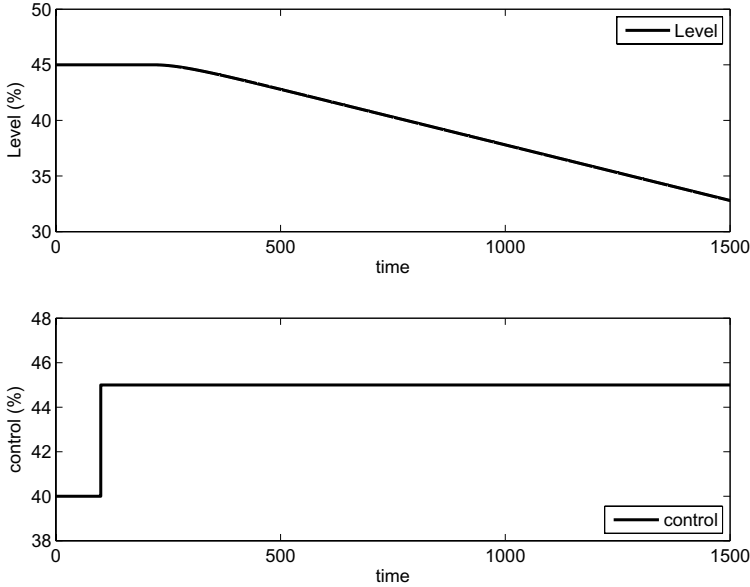


Fig. 2.8. Step response of the buffer tank 1 level

2.1.5 Traffic Systems

Consider a queue of cars at a traffic light as in Fig. 2.9. After the light changes to green the first car in the queue will start to move. A simple model can be obtained considering that each driver will try to follow the speed of the car in front (the first car will try to follow the desired speed v_0). Consider also that the drivers use a proportional control law

$$\frac{dv_i(t)}{dt} = K[v_{i-1}(t) - v_i(t)] \quad i = 1, 2, 3, \dots, N.$$

Applying Laplace transforms

$$sV_i(s) = K[V_{i-1}(s) - V_i(s)] \Rightarrow (1 + s/K)V_i(s) = V_{i-1}(s), \quad i = 1, 2, \dots, N.$$

This can be considered as a series of first-order system transfer functions:

$$\frac{V_i(s)}{V_{i-1}(s)} = G(s) = \frac{1}{1 + s/K}.$$

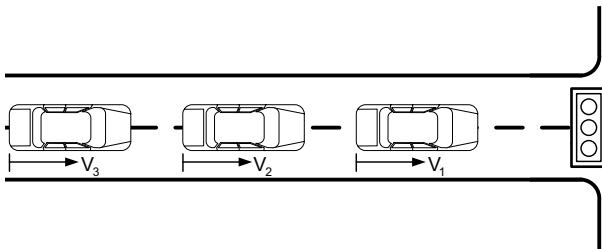


Fig. 2.9. A queue of cars

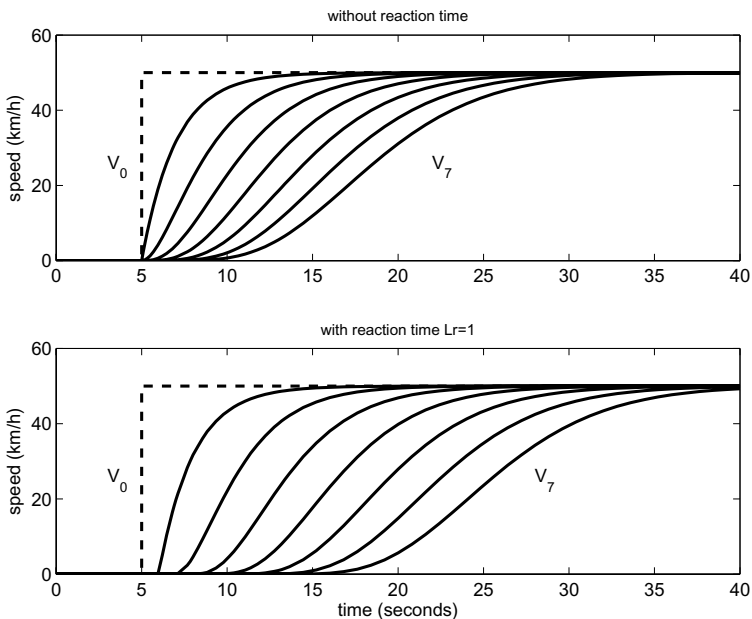


Fig. 2.10. The step response of a queue of seven cars. Case without reaction dead time and with reaction dead time

As in the case of the level in the set of tanks

$$V_N(s) = \frac{1}{(1 + s/K)^N} V_0(s),$$

the speed of the last car reacts with an apparent dead time caused by the accumulative effect of the N time constants. Figure 2.10a shows the speed behaviour of a queue of 7 cars, where $K = 0.5$ and v_0 is a step of 50 km/h applied at $t = 5$ s. Note that the speed of the last car exhibits a dead time of approximately 5 s.

A more complete model can also consider that each driver has a “reaction-dead time” given by $L_{r,i}$, therefore

$$V_i(s) = \frac{e^{-L_r s}}{1 + s/K} V_{i-1}(s), \quad i = 1, 2, 3, \dots, N.$$

Therefore, the behaviour of the last car is determined by the combination of N real dead times and an apparent dead time caused by the N time constants

$$V_N(s) = \frac{e^{-N L_r s}}{(1 + s/K)^N} V_0(s).$$

Figure 2.10b shows the speed behaviour of the same situation as in Fig. 2.10a but considering a reaction dead time of 1 s ($L_r = 1$) in each car model. As can be seen in the figure, the effective dead time of the last car in the queue is approximately 12 s while the real dead time caused by the reaction of the drivers is 7 s. The additional 5 s are due to the accumulative effect of the set of first-order systems.

2.2 Dynamic Behaviour of Dead-time Systems

The effect of dead time is easy to understand in the continuous time domain; however, for the analysis and design of control systems it is sometimes necessary to use a frequency response or a discrete representation of the process.

2.2.1 Representation of Dead Time in the Frequency Domain

Consider the linear model of a pure dead time L given by $G(s) = e^{-Ls}$ with $L > 0$. The frequency response is obtained by computing $G(j\omega) = e^{-j\omega L}$ for $\omega \in \mathbb{R}, \omega > 0$. The gain and phase lags of G are given by

$$|G(j\omega)| = |e^{-j\omega L}| = 1, \quad \varphi_{G(j\omega)} = \varphi_{e^{-j\omega L}} = -\omega L \quad \forall \omega > 0. \quad (2.1)$$

Note that, as the magnitude is equal to one, the dead time will only affect the phase diagram.

The frequency response of a dead-time system can also easily be deduced from Fig. 2.11 where a sine wave of period T_p and the same signal delayed by L are shown. It can be seen that the amplitude of the delayed signal is equal to the amplitude of the original signal, thus the gain for all frequencies is one. The phase can easily be computed as

$$\phi = -\frac{L}{T_p} 2\pi = -\omega_p L.$$

Consider the normalised frequency

$$\omega_n = \omega L,$$

where L is given in seconds, ω in rad/s and ω_n in rad. Figure 2.12a illustrates a normalised phase diagram showing the value of the phase for different

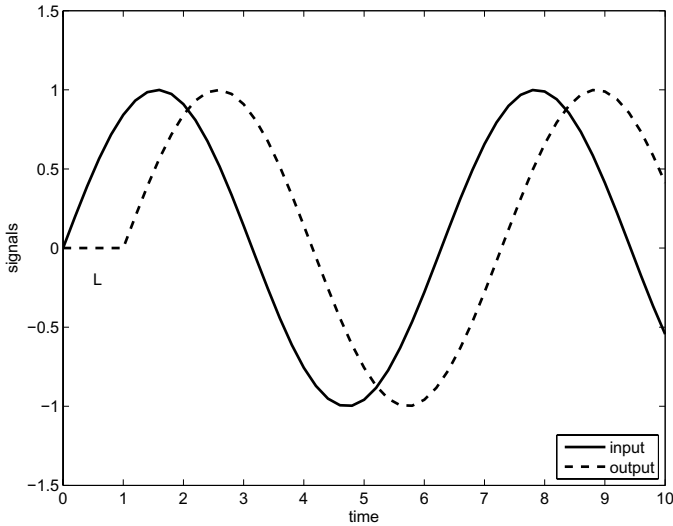


Fig. 2.11. A sine wave and the same signal delayed by L

frequencies. Note that $\phi = -\omega_n$; that is, the relationship is linear and has an exponential shape in the logarithmic scale of the graphics.

Example 2.1: Consider a second-order system with a variable dead time such that

$$P(s) = \frac{1}{(1+s)^2} e^{-Ls}; \quad L \in \{0, 0.1, 1\}.$$

The phase of the dead-time-free system $\phi \rightarrow -180^\circ$ when $\omega \rightarrow \infty$. Note that with a small value of the dead time $L = 0.1$ this value is almost reached at $\omega \approx 5$ rad/s. In Fig. 2.12b the phase diagram with the variable dead time is shown for $L = 0$, $L = 0.1$ and for $L = 1$. Note the fast decrease of the phase for high values of L . The MATLAB[®] code for this example is:

MATLAB [®] code to compute the phase
<pre>% data T=1;K=1;L1=0;L2=.1;L3=1; % phase without dead time [m,phg,w]=bode(K,[T*T 2*T 1]); % dead-time phase phd1=-L1*w*180/pi; phd2=-L2*w*180/pi; phd3=-L3*w*180/pi; % total phase ph1=phg+phd1; ph2=phg+phd2; ph3=phg+phd3; % plot semilogx(w,ph1,w,ph2,'--',w,ph3,'-.'); legend('L=0','L=0.1','L=1');</pre>

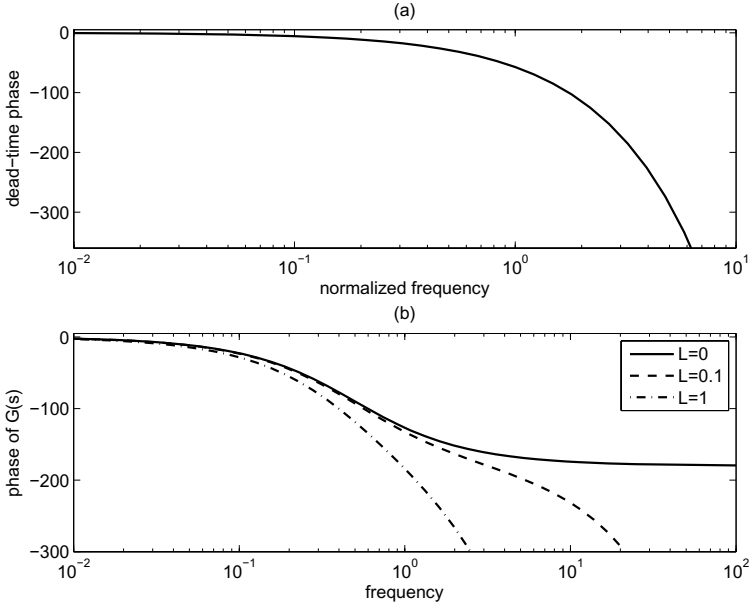


Fig. 2.12. (a) Phase diagram of the dead-time factor e^{-Ls} . (b) Phase diagram of $P(s) = \frac{5}{(1+s)^2} e^{-Ls}$ in Example 2.1 for different values of L

Example 2.2: Consider a heated tank such as the one shown in Fig. 2.1 represented by the model $P(s) = \frac{5}{1+2s} e^{-s} = G(s)e^{-s}$. The frequency response is obtained by computing

$$|P(j\omega)| = \frac{5}{\sqrt{1+(2\omega)^2}}; \quad \varphi_{P(j\omega)} = -\arctan(2\omega) - \omega.$$

It is possible to see the effect of the dead time on the phase of the system. As can be seen in Fig. 2.13 the effect of dead time on the phase decreases the phase margin (PM) of the system. In this particular case, the phase margin of the system without dead time is positive while that with dead time is negative, which shows the important negative effect of dead time on stability.

2.2.2 Polynomial Approximations of Dead Time

In the frequency domain dead time can be directly represented, thus frequency methods of analysis and design can be used without approximations in dead-time processes. However, because the transfer function of a dead time is not rational, when pole-zero representations are needed, as in root-locus or pole-placement methods, polynomial approximations of dead time are used.

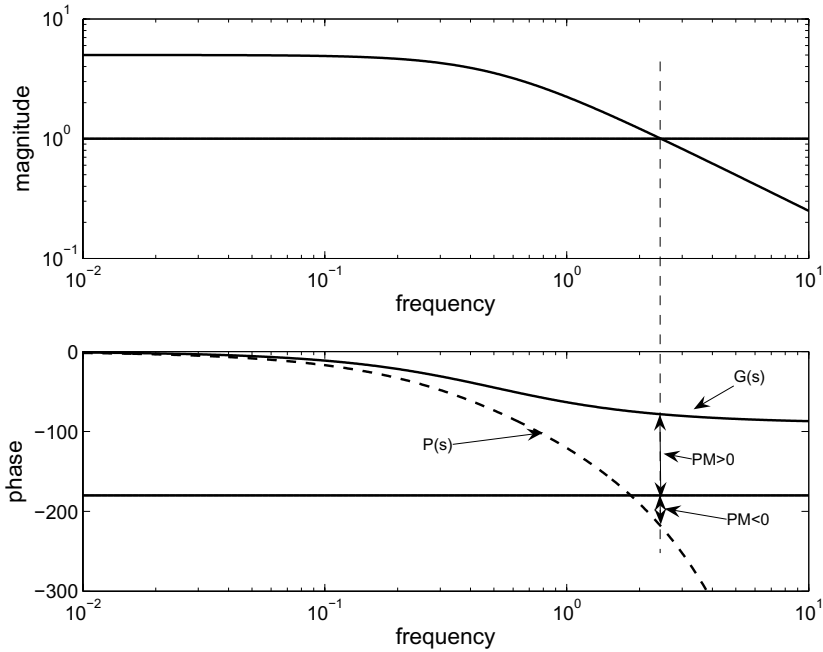


Fig. 2.13. Frequency diagram of the heated tank. Effect of dead time on the phase margin

The nonrational representation of dead time e^{-Ls} can be approximated to a rational transfer function of the form $F(s) = \frac{N(s)}{D(s)}$ using different approaches. Some of these are:

- a Taylor series expansion of $e^{-Ls} = \frac{1}{e^{Ls}}$

$$T_i(s) = \frac{1}{1 + \sum_{i=1}^i \frac{(sL)^i}{i!}}, \quad i = 1, 2, \dots,$$

- a multiple lag transfer function

$$G_i(s) = \frac{1}{(1 + \frac{Ls}{i})^i}, \quad i = 1, 2, \dots,$$

that is an i -order truncation of expression

$$e^{-Ls} = \lim_{i \rightarrow \infty} \frac{1}{(1 + Ls/i)^i},$$

- a Padé representation of ij -order. Although these approximations can be computed for a generic order, the ones most used in practice are the $P_{11}(s)$ and $P_{22}(s)$ given by

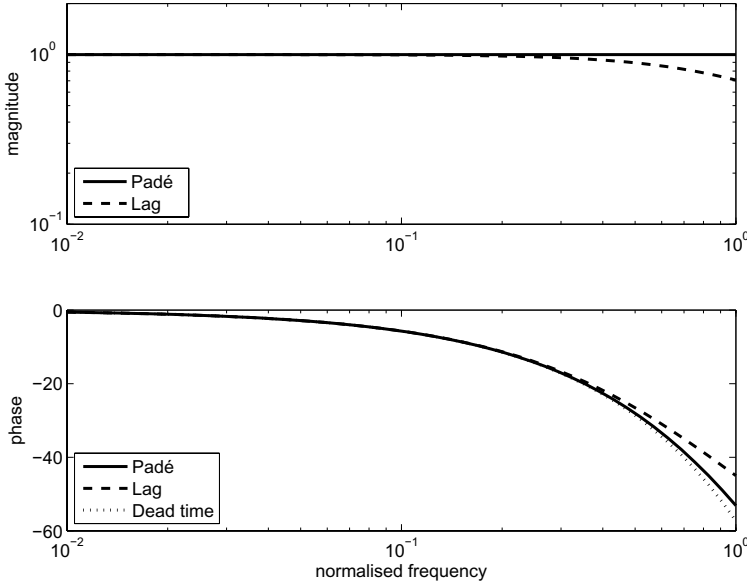


Fig. 2.14. Normalised frequency response of G_1 (dashed lines) and P_{11} (solid lines). The phase of the real dead time is by dotted line

$$P_{11}(s) = \frac{1 - \frac{L}{2}s}{1 + \frac{L}{2}s}, \quad P_{22}(s) = \frac{1 - \frac{L}{2}s + \frac{L^2}{12}s^2}{1 + \frac{L}{2}s + \frac{L^2}{12}s^2}.$$

For control purposes, when simple models are necessary to compute controllers such as the PID analysed in Chap. 4, low-order polynomial approximations of dead time are used. A simple study is presented here for order 1, which is the case most used in classical approaches (see Exercise 2.5 for the analysis of other cases). Note that when simple models are used to represent dead-time-free dynamics, a complete low-order model of the process is also obtained. Also note that $G_1 = T_1$.

In the frequency domain P_{11} verifies

$$|P_{11}(s)|_{s=j\omega} = 1 \quad \forall \omega,$$

that is, the error in the magnitude is zero for all frequencies. In the phase analysis the error is ∞ when $\omega \rightarrow \infty$ for all the approximation methods; however it is possible to define a maximum admissible error e_m and compute the range of frequencies when the approximation error is lower than e_m .

Figure 2.14 shows the normalised frequency response of the lag and Padé approximations. As can be seen from the figure, P_{11} presents better results than G_1 . For example, for an error of 10% in the phase, the approximation P_{11}

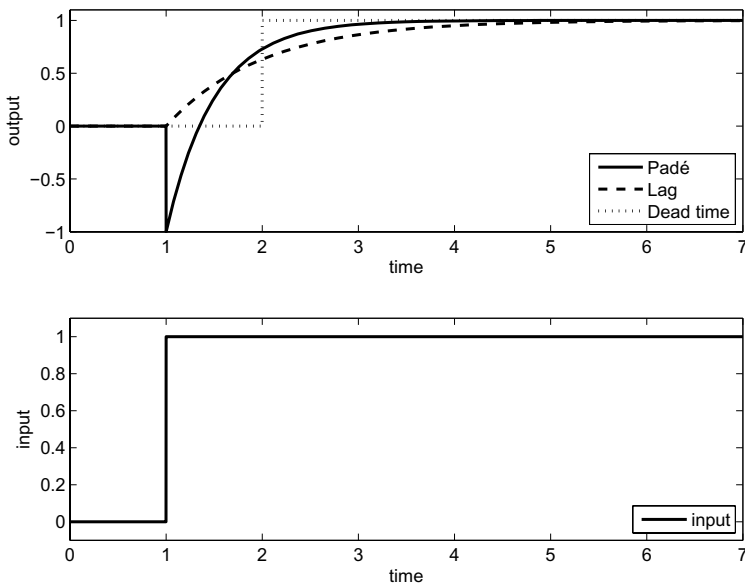


Fig. 2.15. Normalised step response of the lag approximation (dashed line) and the Padé approximation (solid line). The response of the real dead time is the step at $t = 2$

can be considered acceptable up to $\omega_n \cong 1$ rad; while G_1 is only acceptable up to $\omega_n \cong 0.6$ rad.

The analysis can be performed in the time domain using the process response to a unitary step test and, for instance, the integral of the square error (ISE) between the approximate response and the real one can be used to compare the different models. In this case, model P_{11} has an ISE= 5.23 and model G_1 an ISE= 2.22. Figure 2.15 shows the step response of G_1 and P_{11} for a unitary dead time. The step input is applied at $t = 1$. Furthermore, as can be seen from the figure, P_{11} presents nonminimal phase behaviour. The selection of the most appropriate approximation will depend on the type of analysis to be performed. In this analysis, G_1 is clearly superior if a simulation model is needed, but P_{11} seems to be better for a frequency based control design.

Example 2.3: Consider the heat exchanger of Fig. 2.16. In this process steam is used for heating water. An increment in the steam flow (F_s) produces an increment in the outlet water temperature T . On the other hand, an increment in the water flow F_w , regulated by V_1 , produces a decrement in T . Because of the pipe length a significant dead time is observed in the dynamics.

Consider that the steam flow is constant and that the water flow F_w is used as a manipulated variable to control the temperature T . The transfer function between F_w and T can be represented by $P(s) = \frac{-e^{-s}}{s+1}$.

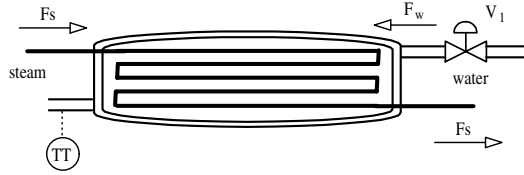


Fig. 2.16. Heat-exchanger

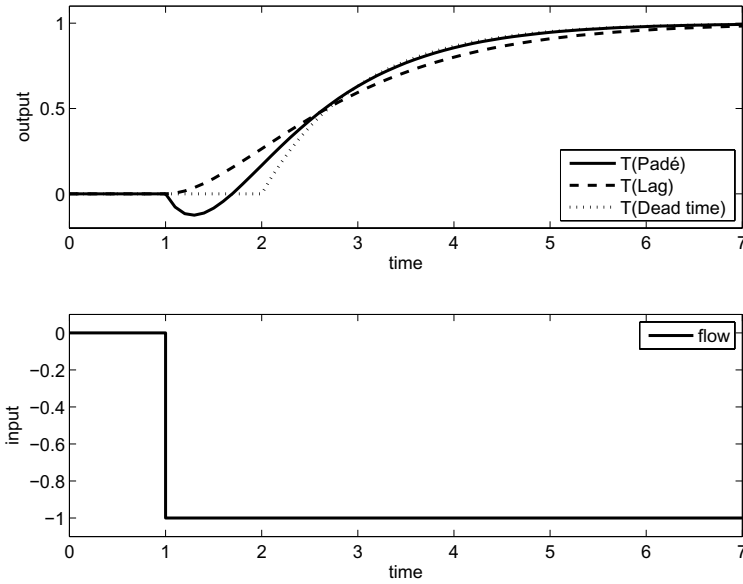


Fig. 2.17. Step response of the model with the lag approximation (dashed line) and the Padé approximation (solid line) and the response of the real dead-time system (dotted line)

Figure 2.17 shows the step response of the heat exchanger represented by $P(s)$ (dotted line) and the one obtained when the lag approximation (dashed line) and the Padé approximation (solid line) are used to approximate the dead time. For the simulations, a negative step input is applied at $t = 1$ in F_w . Note that the step response obtained when using the Padé approximation better reproduces the real behaviour for $t > 2$ but has an undesirable negative response caused by the zero introduced at $s = 0.5$.

2.2.3 Discrete Representation of Dead Time

There are some cases where, because of the nature of the process, the model description can be made directly in the discrete time domain. In this type

of process, time is a discrete variable; that is, if a signal $x_2(t)$ is obtained by delaying $x_1(t)$ the relationship is given by

$$x_2(t) = x_1(t - d),$$

where both t and d are integer multiples of a certain period of time that we call sampling time.

In this book we use the variable t to represent time for continuous systems ($t \in \mathbb{R}$). For discrete systems ($t \in \mathbb{Z}$) represents the number of sampling instants. The real time instant for discrete systems is therefore, tT_s where T_s is the sampling time.

As an example consider the dynamic model of a manufacturing supply chain, where the time is measured in days [119]. The dead time is an important part of the dynamic model of this process. A simple model for this system is

$$y(t) = y(t - 1) + Ku(t - d - 1) - q(t), \quad (2.2)$$

where y is the stock level, u represents the factory starts, q is the demand and t is expressed in days. In this model K is the factory production yield and d the delay time of the factory. Ideally $K = 1$ but in practice it could assume different values because of a mistake in estimating the quantity of material u arriving from the factory on the correct day.

Applying the $\mathcal{Z}$ transform, the representation of this model is given by

$$Y(z) = \frac{Kz^{-1}}{1 - z^{-1}}z^{-d}U(z) - \frac{1}{1 - z^{-1}}Q(z), \quad (2.3)$$

that is, the model shows integral behaviour with a dead time.

Another case where discrete models of dead time may be important is when digital equipment is used to control continuous time systems. This is a very common situation because, in practice, on many occasions controllers are implemented using microprocessors. For the analysis and design of these discrete controllers two different approaches can be made: (i) Using a continuous design and then computing a discrete approximation of the continuous controller or (ii) using a direct discrete design based on the discrete representation of the continuous process. In the latter case, it is always necessary to use a discrete model of the dead time.

Consider that the dynamic behaviour of the process is described by the continuous transfer function

$$P(s) = G(s)e^{-Ls}, \quad (2.4)$$

where $G(s)$ is the dead-time-free part of the process and L is the effective dead time and a sampling period T_s . A discrete description of the process is given by

$$P(z) = \mathcal{Z}\{B_o(s)P(s)\}, \quad (2.5)$$

where $P(z)$ is the discrete transfer function relating the $\mathcal{Z}$ transform of the sampled output of the process and the $\mathcal{Z}$ transform of the discrete input that passes through a zero-order hold block ($B_o(s)$).

First suppose that T_s is chosen as an integer submultiple of L , that is, an integer d exists such that $L = dT_s$. In this case the discrete model can be computed as

$$P(z) = G(z)z^{-d}, \quad (2.6)$$

where $G(z)$ represents the discrete dead-time-free dynamics of the process and d represents the dead time, that is the dead time L is represented by d samples

$$L = dT_s, \quad e^{-Ls} \rightarrow z^{-d}$$

and

$$G(z) = \mathcal{Z}\{B_o(s)G(s)\}.$$

It is clear that in general the real dead time L will not be a multiple of T_s ; thus,

$$L = dT_s + \delta L, \quad -T_s/2 \leq \delta L \leq T_s/2,$$

where δL is the error introduced by the discrete representation of dead time. This error can be neglected when $\delta L \ll T_s$. Note that the error in the estimation of dead time when computing the continuous model can, in practice, be greater than δL .

If the error δL needs to be considered, a polynomial approximation can be used. In this case, one of the models presented in the previous section is included in the representation of the process, giving

$$P(s) = G(s)e^{-Ls} = G(s)A(s)e^{-dT_s s},$$

where $A(s)$ is the rational function used to approximate $e^{-\delta L s}$. In this case the complete model is given by

$$P(z) = G(z)z^{-d}, \quad G(z) = \mathcal{Z}\{B_o(s)G(s)A(s)\}. \quad (2.7)$$

Example 2.4: Consider the heat exchanger in Fig. 2.16 where the flow of steam is used as a manipulated variable. Performing a step test close to the operating point, the following continuous model is obtained

$$P(s) = G(s)e^{-Ls} = \frac{5}{1+2s}e^{-4s}.$$

Using a sample time $T_s = 0.4$, the discrete model of $P(s)$ is then given by

$$P(z) = \frac{5(1-0.81)}{z-0.81}z^{-10} = \frac{0.95}{z-0.81}z^{-10}.$$

The MATLAB[®] function `cp2dp` can be used for this discretisation:

MATLAB[®] code for the discretisation of dead-time systems

```
% define data
num=5; den=[2 1]; Ts=0.4; L=4;
% compute numdis, dendis
[numdis, dendis]=cp2dp(num, den, Ts, L)
```

Example 2.5: Now consider the process given by

$$P(s) = \frac{2}{1+5s} e^{-7.2s}$$

and a sample time $T_s = 0.5$. The dead time $L = 7.2$ is not a multiple of 0.5, thus writing

$$L = 0.5d + \delta L,$$

gives $d = 14$ and $\delta L = 0.2$. Approximating $e^{-\delta Ls} = e^{-0.2s}$ by $\frac{1}{1+0.2s}$, the model can be written as

$$P(s) \simeq \frac{2}{(1+5s)(1+0.2s)} e^{-7s}.$$

The discrete representation, computed using a zero-order-holder, is

$$P(z) = \frac{0.1218z + 0.0529}{z^2 - 0.9869z + 0.0743} z^{-14}.$$

Figure 2.18 shows the first part of the step response of the process $P(s)$ and the discrete model $P(z)$ for the example. As can be seen, the differences between the responses of the model and the process are very small. In this case any small error in the estimation of the value of L could cause larger errors than the ones shown in Fig. 2.18. These errors are normally related to the use of linear low-order models to represent the nonnecessary linear high-order dynamics of the process and also to measurement errors produced by noise. These issues are analysed later in this chapter and in Chap. 3.

2.2.4 State-space Representation of Dead-time Systems

A state-space representation equivalent to the input–output representation given by the transfer function of a dead-time-free process ($G(s)$) can be obtained using different realisations [71]. Consider

$$G(s) = \frac{Y(s)}{U'(s)},$$

where $u'(t)(U'(s))$ is the input and $y(t)(Y(s))$ is the output. Using

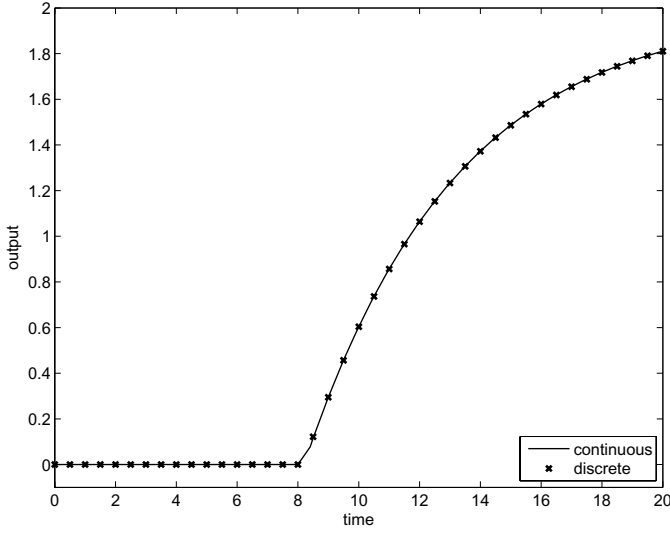


Fig. 2.18. Step response of the process P and the discrete model for Example 2.5

$$G(s) = \frac{B(s)}{A(s)},$$

where $A(s)$ is monic and has degree n and $B(s)$ has degree $n - 1$

$$A(s) = s^n + a_{n-1}s^{n-1} + \dots + a_1s + a_0,$$

$$B(s) = b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_1s + b_0.$$

If a controllable canonical form is used to represent the system

$$\frac{d\mathbf{x}(t)}{dt} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ \dots \\ 0 \\ 1 \end{bmatrix} u'(t),$$

$$y(t) = [b_0 \ b_1 \ \dots \ b_{n-1}] \mathbf{x}(t),$$

where $\mathbf{x}(t)$ is the state vector. In a matrix form

$$\begin{aligned} \frac{d\mathbf{x}(t)}{dt} &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}u'(t), \\ y(t) &= \mathbf{c}\mathbf{x}(t). \end{aligned} \tag{2.8}$$

Consider now that a dead time L is introduced at the input

$$u'(t) = u(t - L)$$

gives

$$\begin{aligned} \frac{d\mathbf{x}(t)}{dt} &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t - L), \\ y(t) &= \mathbf{c}\mathbf{x}(t), \end{aligned} \quad (2.9)$$

that is represented in the s -domain as

$$U'(s) = U(s)e^{-Ls},$$

$$Y(s) = G(s)U'(s) = G(s)e^{-Ls}U(s) = P(s)U(s).$$

Therefore the model (2.9) represents the model

$$\frac{Y(s)}{U(s)} = P(s) = G(s)e^{-Ls}.$$

The differential equation in (2.9) is not an ordinary differential equation but a differential-difference equation that belongs to the class of functional differential equations that are infinite dimensional [112]. The infinite dimensional characteristic can be understood considering the Taylor expansion of e^{-Ls} . To represent exactly the dead time

$$e^{-Ls} = \frac{1}{1 + \sum_{i=1}^{\infty} \frac{(sL)^i}{i!}},$$

which has a denominator of ∞ degree. Therefore, the correspondent state-space representation has an infinite dimensional state.

This is not the case when a discrete representation of the dead-time process is used. In this case the transfer function is

$$P(z) = G(z)z^{-d} = \frac{B(z)z^{-d}}{A(z)} = \frac{B(z)}{A(z)z^d}, \quad (2.10)$$

where $A(z)$ is monic and has degree n and $B(z)$ has degree $n - 1$

$$A(z) = z^n + a_{n-1}z^{n-1} + \dots + a_1z + a_0,$$

$$B(z) = b_{n-1}z^{n-1} + b_{n-2}z^{n-2} + \dots + b_1z + b_0.$$

Thus, the dead time only increases the order of the model in d . Note that the new denominator $A(z)z^d$ has degree $n + d$

$$A(z)z^d = z^{d+n} + a_{n-1}z^{d+n-1} + \dots + a_1z^{d+1} + a_0z^d.$$

Using the canonical form to represent the system gives

$$\mathbf{x}(t+1) = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & -a_0 & \dots & -a_{n-1} \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 0 \\ \dots \\ 0 \\ 1 \end{bmatrix} u(t),$$

$$y(t) = [b_0 \ b_1 \ \dots \ b_{n-1} \ 0 \ \dots \ 0] \mathbf{x}(t),$$

where $t \in \mathbb{Z}$ represents the discrete time. In this case, the state description maintains the characteristics of the dead-time-free case but the dimension of the system is greatly increased. For instance, for an FOPDT model $P(s) = \frac{e^{-Ls}}{sT+1}$ with a dominant dead time $L = 5T$ and a sampling time $T_s = 0.1T$, we have $n = 1$ and $d = 50$. Thus, while the dead-time-free system has a state vector of dimension 1, the dead-time system has a state vector of dimension 51.

The state-space representation of the dead-time system is used in several works to analyse stability and other properties (see the Further Reading section for references). In some cases more complex state space descriptions are used, with different dead times in the input and the state

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}_1\mathbf{x}(t) + \mathbf{A}_2\mathbf{x}(t - L_1) + \mathbf{b}u(t - L_2), \quad (2.11)$$

$$y(t) = \mathbf{c}\mathbf{x}(t). \quad (2.12)$$

Since this book is orientated to control applications where the process model is, usually obtained from input–output data, the transfer function model is used in most of the system representations.

2.3 Simple Models for Typical Dead-time Systems

Simple models are widely used in industry to represent the dynamic behaviour of many processes. Most of these models include a dead time as part of the process representation. Some linear low-order input–output models are presented in this section. Normally, they are obtained by identification procedures, for example, using the well known step-test method close to an operating point of the process. This method, as well as other identification techniques for dead-time processes, is presented in Chap. 3.

2.3.1 Linear Models Close to an Operating Point

Linear models and linear controllers are very common in industry. Although the real dynamic behaviour of many industrial processes presents nonlinear characteristics, when the processes operate close to an operating point the nonlinear behaviour can be adequately approximated by a linear model. Consider a stable¹ process with input $u(t)$ and output $y(t)$ operating in steady state

$$u(t) = U_0, \quad y(t) = Y_0,$$

where (U_0, Y_0) represents the operating point. Considered now that the input is changed to drive the system to a different operating point (U_1, Y_1) . Figure 2.19 shows an example of the (U, Y) plane of operating points of the process where (U_0, Y_0) and (U_1, Y_1) are represented.

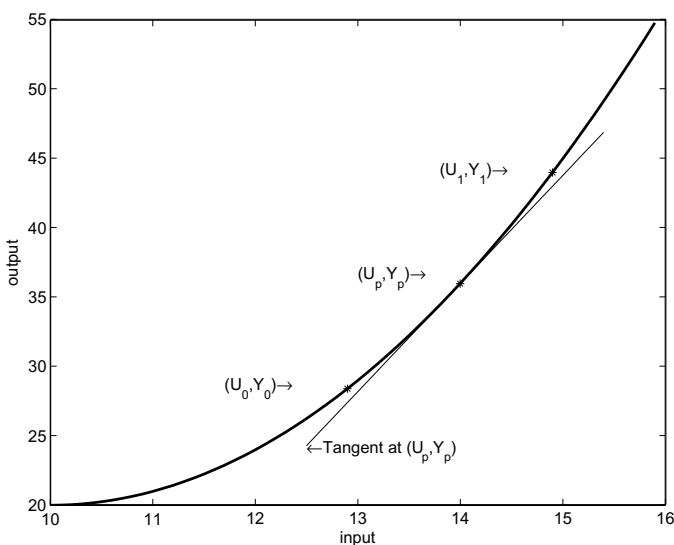


Fig. 2.19. (U, Y) plane of operating points

This procedure can be repeated for all the operating points obtaining a function $Y = f(U)$ that represents the static characteristics of the process. In general $f(U)$ is a nonlinear function. However, if only the behaviour of the process close to one operating point (U_p, Y_p) is considered, $f(U)$ can be approximated by a linear function that uses the tangent of $f(U)$ at the desired

¹In this book we consider stability in the bounded-input–bounded-output sense, that is, a system is considered stable if for any bounded input its output is also bounded.

point (see Fig. 2.19). In this situation, the relationship between $u(t)$ and $y(t)$ can be approximated by a linear system considering an incremental model

$$u(t) = U_p + \Delta u(t), \quad y(t) = Y_p + \Delta y(t).$$

To obtain the linear dynamics of the incremental model we can use two procedures. Analytically, we can substitute the values of $y(t)$ and $u(t)$ in the nonlinear model (for example, a nonlinear differential equation) and eliminate the constant and high-order terms. For example, the dynamics of the level of a tank $H(t)$ can be represented by the nonlinear differential equation

$$\frac{dH(t)}{dt} = k_1 F(t) - k_2 \sqrt{H(t)},$$

where $F(t)$ is the input flow and k_1, k_2 are constants that depend on the tank characteristics. When the system is at the equilibrium corresponding to the operating point (F_p, H_p) , $\frac{dH(t)}{dt} = 0$ and therefore $k_1 F_p = k_2 \sqrt{H_p}$. Using the incremental variables

$$h(t) = H(t) - H_p, \quad f(t) = F(t) - F_p$$

and the first-order Taylor approximation of $\sqrt{H(t)}$ (the terms of order greater than 2 are neglected)

$$\sqrt{H(t)} \cong \sqrt{H_p} + \frac{1}{2\sqrt{H_p}} h(t),$$

and substituting in the nonlinear differential equation

$$\frac{d(H_p + h(t))}{dt} = k_1(F_p + f(t)) - k_2 \left[\sqrt{H_p} + \frac{1}{2\sqrt{H_p}} h(t) \right],$$

this gives

$$\frac{dh(t)}{dt} = k_1 F_p + k_1 f(t) - k_2 \sqrt{H_p} - k_2 \frac{1}{2\sqrt{H_p}} h(t).$$

Noting that $k_1 F_p - k_2 \sqrt{H_p} = 0$, it follows that

$$\frac{dh(t)}{dt} = k_1 f(t) - k_2 \frac{1}{2\sqrt{H_p}} h(t),$$

which is a linear differential equation in the incremental variables $h(t)$ and $f(t)$.

The other method that is the most common in practice, consists of the use of a step test that considers a fairly small step change in $u(t)$ close to U_p which produces a small change in the output near Y_p ($Y_p = 36$ in Fig. 2.20) in such a way that the linear behaviour approximates the nonlinear dynamics with

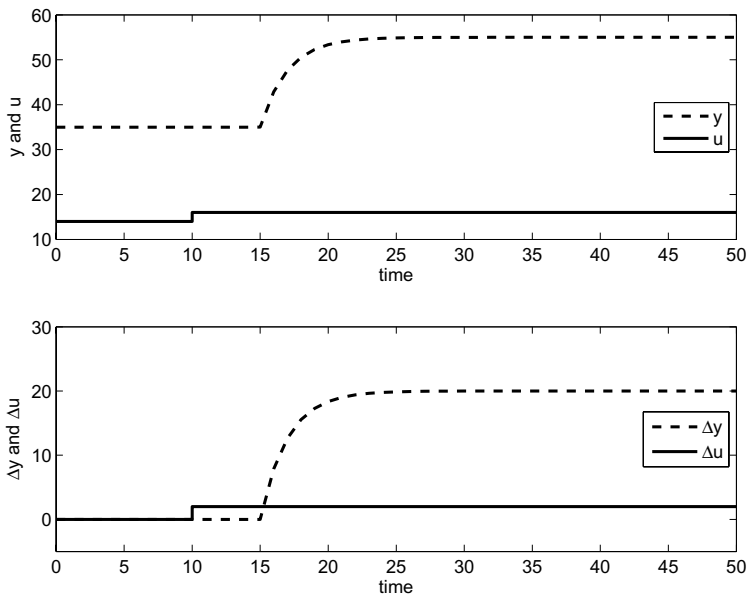


Fig. 2.20. Step test close to an operating point

a small error. Figure 2.20 illustrates this procedure when a step of amplitude $\Delta U_p = 2$ is applied in $u(t)$. Note that the final value of the output is $Y_p + \Delta Y_p$ (56 in Fig. 2.20). In the same figure the incremental variables are shown. Note that in this case the initial point is $(0, 0)$ and the final point is $(\Delta U_p, \Delta Y_p)$ ((2, 20) in Fig. 2.20). From this data, a transfer function can be computed to represent the relationship between $\Delta U(s)$ and $\Delta Y(s)$

$$P(s) = \frac{\Delta Y(s)}{\Delta U(s)},$$

as will be explained in Chap. 3. Normally, $P(s)$ is of low order. The following simple transfer function models are used in industry to describe the behaviour of stable and unstable processes exhibiting a dead time.

Open-loop Stable Models with a Dead Time

In this case two models are used, the first-order-plus-dead-time (FOPDT) model and the second-order-plus-dead-time (SOPDT) model:

- the FOPDT model is represented by

$$P(s) = \frac{K_p}{1 + Ts} e^{-Ls}, \quad (2.13)$$

where K_p , T and L are real numbers. $T > 0$ is the equivalent time constant of the plant and K_p is the static gain. $L > 0$ is the equivalent dead time.

This model is, perhaps, the one most commonly used in industry and it is also the basis for most of the classical PID tuning rules. Figure 2.21 shows the step response of this model. Note that the step response has a nonnull derivative in the first instant after the dead time (see Fig. 2.21).

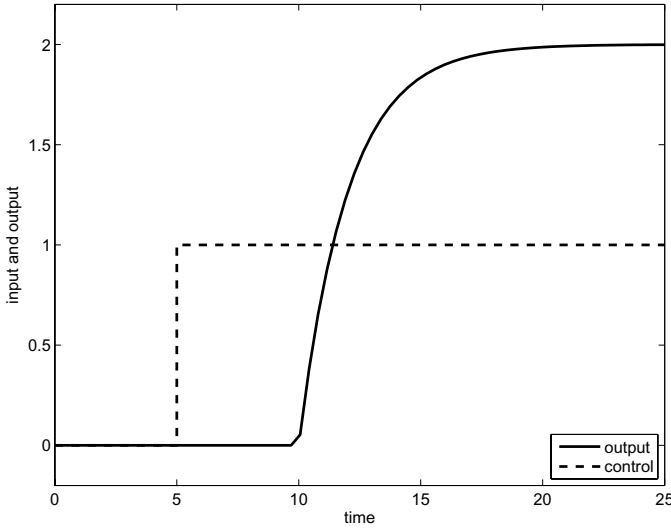


Fig. 2.21. Unitary step response of the FOPDT model

- When it is desirable to represent a smoother step response in the first part of the transients or an oscillatory step response, a second-order process with a dead time is used

$$P(s) = \frac{K_p e^{-Ls}}{(1 + T_1 s)(1 + T_2 s)} = \frac{K_p e^{-Ls}}{1 + \frac{2\xi s}{\omega_n} + \frac{s^2}{\omega_n^2}}, \quad (2.14)$$

where $K_p, T_1, T_2, \xi, \omega_n$ and L are real numbers. As in the FOPDT model K_p is the static gain and $L > 0$ the equivalent dead time. $T_1 > 0$ and $T_2 > 0$ are time constants of the plant in the case of a nonoscillatory response while $\xi \in (0, 1)$ (the damping coefficient) and $\omega_n > 0$ (the natural frequency) are used when the process exhibits an oscillatory step response. Figure 2.22 shows the step response of this type of model for both cases.

Integrative Models with Dead Time

When the process exhibits integrative behaviour, two simple models can be used, the integrative-plus-dead-time (IPDT) model and the second-order-integrative-plus-dead-time (SOIPDT) model

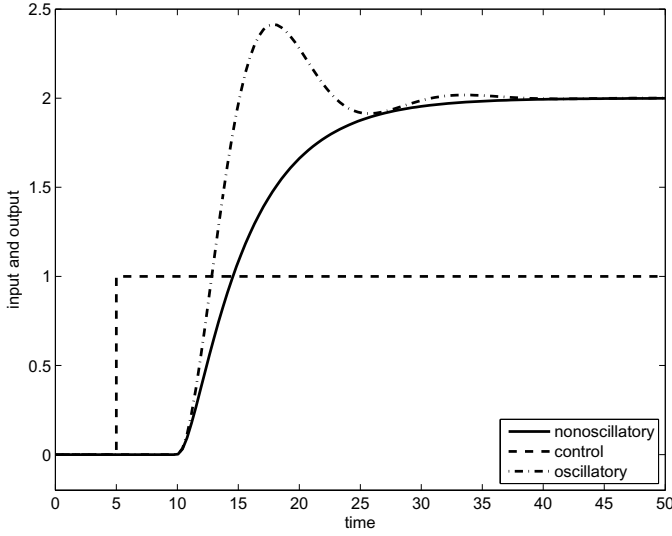


Fig. 2.22. Unitary step response of the nonoscillatory and oscillatory SOPDT model

- The IPDT model is represented by

$$P(s) = \frac{K_v}{s} e^{-Ls}, \quad (2.15)$$

where $L > 0$ is the equivalent dead time and K_v is the velocity gain.

- As in the stable case, a SOIPDT model is used to represent a smoother step response in the first part of the transients

$$P(s) = \frac{K_v}{s(1 + Ts)} e^{-Ls}, \quad (2.16)$$

where the new parameter $T > 0$ is the equivalent time constant of the nonintegrative part of the plant.

Figure 2.23 shows the step response of these two models.

Open-loop Unstable Model with a Dead Time

In some situations the process can exhibit unstable behaviour that can be represented by

$$P(s) = \frac{K_p}{1 - Ts} e^{-Ls}, \quad (2.17)$$

where $L > 0$ is the equivalent dead time and K_p and $T > 0$ are used to describe the increasing exponential process output. Figure 2.24 shows the step response of this model.

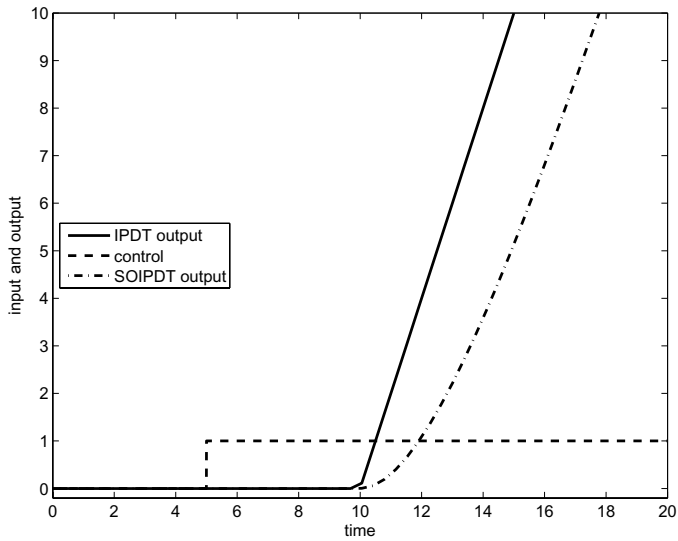


Fig. 2.23. Unitary step response of the IPDT and SOIPDT models

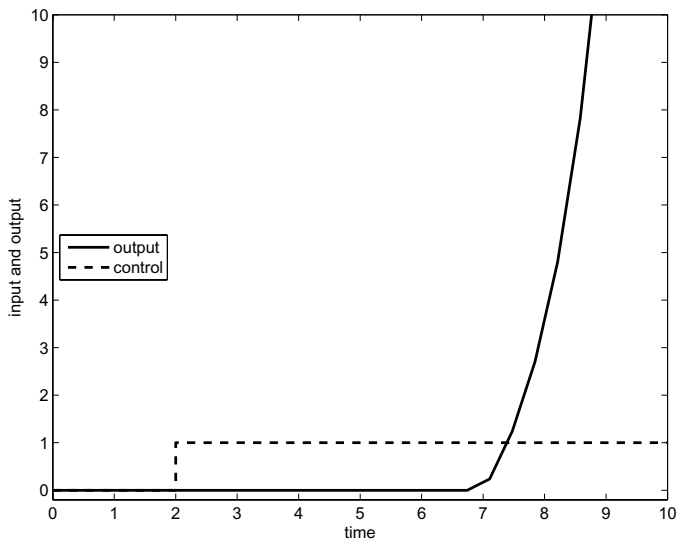


Fig. 2.24. Step response of the unstable model with a dead time

The following example illustrates how these simple models can represent the dynamics of a high-order process in the step test (see also Exercises 2.7 and 2.8).

Example 2.6: Consider: (i) an FOPDT model with $K_p = 1$, $T = 1.5$ and $L = 10.5$, and the “real” process

$$P(s) = \frac{e^{-10s}}{(1+s)(1+.5s)(1+.25s)(1+.125s)}, \quad (2.18)$$

(ii) an IPDT model with $K_v = 2$ and $L = 6.7$, and the “real” process

$$P(s) = \frac{2e^{-5s}}{s(1+s)(1+0.5s)(1+0.1s)}. \quad (2.19)$$

Figure 2.25 shows the step responses of the models and the processes in order to appreciate how the dead time represents the effect of the high-order dynamics. The input is a unitary step applied at $t = 0$.

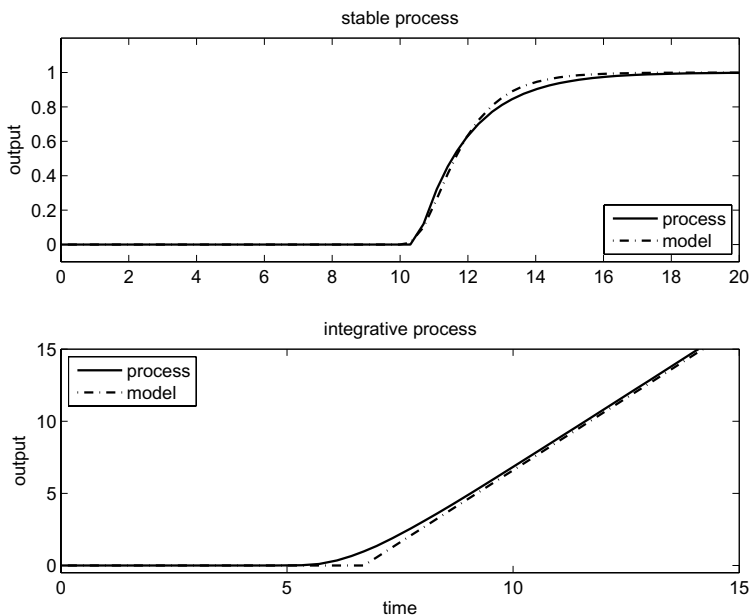


Fig. 2.25. Step response of the “real” processes and models

In these simple models the relationship between the dead time and the time constant associated with the dead-time-free part of the process is an important measurement of the process characteristics. Thus, usually, the dead time is normalised.

Dead time can be normalised in different ways. The most common way is the normalised dead time (τ) defined as

$$\tau = \frac{L}{L+T}, \quad 0 \leq \tau \leq 1, \quad (2.20)$$

using the time constant and dead time of the FOPDT model. Note that dead-time-dominant processes have a value of τ close to 1 and processes with τ close to 0 are lag dominant. A rule of thumb is to consider a process to be dead-time-dominant when $\tau > 2/3$ ($L > 2T$). Normally, τ is used to define the difficulties associated with the control design of a given process (greater values of τ mean more difficulties) and it is also usually called the “controllability ratio”.

The previous models are normally obtained using some measurement from the output of the real process or using some simplifications in a complex set of differential equations that describe the process dynamics. Therefore, it is necessary to study how to represent the modelling errors introduced in the determination of the process model.

2.4 Analysis of Modelling Errors

Many approximations are made when modelling a process: high-order dynamics are neglected, nonlinear dynamics are represented by linearised equations, etc. As the model is an approximation of the real process, an analysis of the modelling errors is necessary if a robust controller is to be designed.

2.4.1 Modelling Error Representation

Modelling errors can be represented in a parametric form or in the frequency response domain. In this book, the second approach is used based on the results of [70, 124]

To account for model uncertainty we will assume that the process is described by a family of transfer functions in such a way that the magnitude and phase of $P(j\omega)$ can vary in a disk with radius $\overline{\Delta P}(\omega)$, as shown in Fig. 2.26 [70]. Thus, each model $P(s)$ in the family can be written in the frequency domain as

$$P(j\omega) = P_n(j\omega) + \Delta P(j\omega), \quad |\Delta P(j\omega)| \leq \overline{\Delta P}(\omega) \quad \forall \omega \geq 0,$$

where $P_n(j\omega)$ is the nominal model and $\Delta P(j\omega)$ is defined as the additive description of the modelling errors. Equivalently

$$P(j\omega) = P_n(j\omega)(1 + \delta P(j\omega)), \quad |\delta P(j\omega)| \leq \overline{\delta P}(\omega) \quad \forall \omega \geq 0,$$

where $\delta P(j\omega)$ is a multiplicative description of the modelling errors and

$$\delta P(j\omega) = \frac{\Delta P(j\omega)}{P_n(j\omega)}, \quad \overline{\delta P}(\omega) = \frac{\overline{\Delta P}(\omega)}{|P_n(j\omega)|} \quad \forall \omega \geq 0.$$

This type of uncertainty model is usually called “unstructured description” (additive or multiplicative) because it is not linked to the variation of specific parameters (gain, dead time, etc.) and it is specially suitable to describe

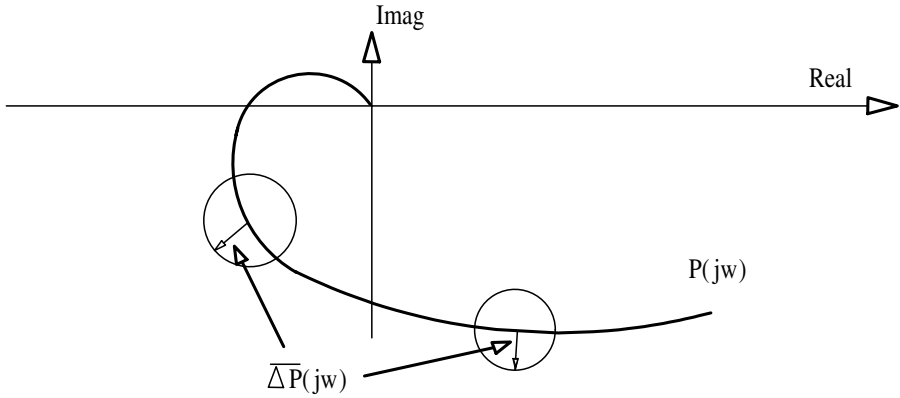


Fig. 2.26. Model uncertainty in the frequency domain

the typical unmodelled dynamics found in industrial processes. However, it must be noted that when considering disks to represent the uncertainty bound of the plant we are including some other plants not present in the original uncertainty set.

$\overline{\Delta P}(\omega)$ and $\overline{\delta P}(\omega)$ are the bounds for the additive and multiplicative uncertainties, respectively. Also, it is possible to write

$$\Delta P(j\omega) = \Delta_A \overline{\Delta P}(\omega); \quad \delta P(j\omega) = \delta_I \overline{\delta P}(\omega).$$

The phase of Δ_A and δ_I can vary arbitrarily but $|\Delta_A| \leq 1$ and $|\delta_I| \leq 1$.

The uncertainty usually increases with frequency and $\overline{\delta P}(\omega)$ can exceed 1 at high frequencies, as shown in the following examples. This is reasonable because, in practice, models tend to describe the low frequency behaviour well, but become inaccurate for high frequency inputs [70].

Note that if one plant in the family has a pole at $s = j\omega_1$ in the imaginary axis

$$|P(j\omega_1)| = \infty \Rightarrow \overline{\Delta P}(\omega_1) = \infty \Rightarrow \overline{\delta P}(\omega_1) = \infty.$$

Thus the additive and multiplicative uncertainty are unsuitable for describing sets of plants where poles can cross the imaginary axis. Therefore, all the plants in the family should have the same number of poles on the right side of the s -plane [70, 124]. This condition is considered valid in all the robustness analysis of this book. Furthermore, in this book, we use the multiplicative uncertainty for the analysis of robustness in most of cases. Typical simple transfer functions are used to represent the bound for the multiplicative uncertainty

$$\overline{\delta P}(s) = \frac{T_\delta s + r_0}{(T_\delta/r_\infty)s + 1},$$

where r_0 is the relative uncertainty at steady state, $1/T_\delta$ is approximately the frequency at which the relative uncertainty reaches 100% and r_∞ is the

relative uncertainty at high frequencies. In some situations a more complex $\overline{\delta P}(s)$ is necessary. In practice, we start using a simple weight and then, if necessary, try a higher-order transfer function.

2.4.2 Modelling Errors in Dead-time Processes

In the dead-time process case, uncertainties can be associated with the error in the estimation of the dead time L and to the simplifications used to approximate the dead-time-free dynamics of the process by a transfer function. This analysis can be made for the general case but in this chapter the attention is focussed on the case where the nominal models are the simple first-order-plus-dead-time and integrative-plus-dead-time transfer functions, as they are the most widely used in industry. It is shown that the main factor affecting frequency domain uncertainties is dead-time-estimation error.

The following types of uncertainties will be considered:

- Errors in the estimation of the gains K_p or K_v ;
- errors in estimating the dead time;
- errors in estimating the dominating time constant in the stable case;
- unmodelled dynamics. As is customary in literature, poles and zeros that are faster than the dominant pole are approximated by a first-order model with an equivalent time constant T_u , smaller than the dominant T [4].

These considerations are related to the most common effects found in industrial processes. Thus, for the stable case the “real” process ($P(s)$) and the model ($P_n(s)$) are represented by

$$P(s) = e^{-Ls} \frac{K}{(1+sT)(1+sT_u)}, \quad P_n(s) = e^{-L_n s} \frac{K_n}{1+sT_n}$$

and for the integrative case by

$$P(s) = e^{-Ls} \frac{K}{s(1+sT_u)}, \quad P_n(s) = e^{-L_n s} \frac{K_n}{s},$$

where from now on the suffix n is used to represent the nominal value of a parameter or a model.

In the following, a frequency analysis of the multiplicative error is presented for errors in all the parameters (see also Exercise 2.10). The effect of each type of modelling error can be seen in [70, 124].

Stable Models

Using the transfer functions of $P(s)$ and $P_n(s)$ for the stable case, the general expression of $\delta P(s)$ is

$$\delta P(s) = \frac{\frac{Ke^{-Ls}}{(1+sT)(1+sT_u)} - \frac{K_ne^{-L_ns}}{1+sT_n}}{\frac{K_ne^{-L_ns}}{1+sT_n}} = \frac{K}{K_n} \frac{(1+sT_n)e^{-(L-L_n)s}}{(1+sT)(1+sT_u)} - 1.$$

Using a parametrisation of the models with the value of the nominal dead time, the error can be written as

$$\delta P(j\omega_n) = (1 + \delta K) \frac{1 + j\omega_n T'_n}{1 + j\omega_n (T'_n + \delta T)} \frac{1}{1 + j\omega_n T'_u} e^{-j\omega_n \delta L} - 1,$$

where $\omega_n = \omega L_n$, $L' = L/L_n$, $T' = T/L_n$, $T'_u = T_u/L_n$, $T'_n = T_n/L_n$ and $T' = T/L_n$. The incremental (or relative) errors are

$$\delta L' = \delta L = \frac{L - L_n}{L_n},$$

$$\delta T' = \delta T = \frac{T - T_n}{T_n},$$

$$\delta K = \frac{K - K_n}{K_n}.$$

To combine the effects of the errors, some assumptions are made: (i) dead-time-dominant processes are considered, thus $L \geq T$ and (ii) the neglected dynamics represented by T_u are considered at high frequencies if compared to T . Figure 2.27 shows the shape of δP for the case $T'_n = 0.5$, $T_u = 0.2$ and $\delta L = 30\%$. The following cases are considered: $\delta T = 0.1$, $\delta K = 0.1$ (solid line), $\delta T = 0.1$, $\delta K = 0.2$ (dashed line), $\delta T = 0.2$, $\delta K = 0.1$ (dotted line) and $\delta T = 0.2$, $\delta K = 0.2$ (dashed-dotted line). Several conclusions can be derived from these curves (see also Exercises 2.10 and 2.11):

- The error in the gain has an important effect on the multiplicative errors at low frequencies but almost no effect at medium and high frequencies;
- from medium to high frequencies the curves are dominated by the dead-time error;
- the maximum value of the error ($\delta P_{\max}$) is always near 2. Note that if only δL is considered $\delta P_{\max} = 2$.

Thus, special attention must be paid to the dead-time-estimation error in this type of process.

The following MATLAB® code is used for the computation of $\delta P(s)$.

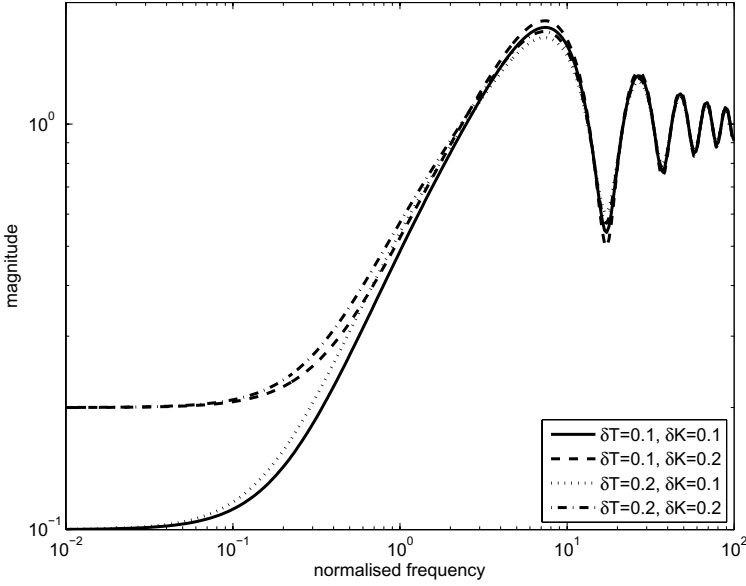


Fig. 2.27. Multiplicative modelling error in the general case for stable processes: $T'_n = 0.5$, $T_u = 0.2$ and $\delta L = 30\%$

MATLAB[®] code for the computation of $\delta P(s)$ <pre> %define errors dL=0.3; Tn=0.5; Tu=Tn/5; dK=[.1 .2]; dT=[.1 .2]; n=0; % loop for jj=1:2 dt=dT(jj); for kk=1:2 dk=dK(kk); n=n+1; ww=logspace(-2,2,300); for i=1:300 w=ww(i); % error e=(1+dk)*((1+w*j*Tn)/((1+w*j*(Tn+dt))*(1+w*j*Tu))); deltaP(n,i)=abs(e*exp(-w*j*dL)-1); end; end; end; </pre>

Integrative Models

In this case the general expression of $\delta P(s)$ is

$$\delta P(s) = \frac{e^{-Ls} \frac{K}{s(1+sT_u)} - e^{-L_n s} \frac{K_n}{s}}{e^{-L_n s} \frac{K_n}{s}} = e^{-(L-L_n)s} \frac{K}{K_n} \frac{1}{1+sT_u} - 1.$$

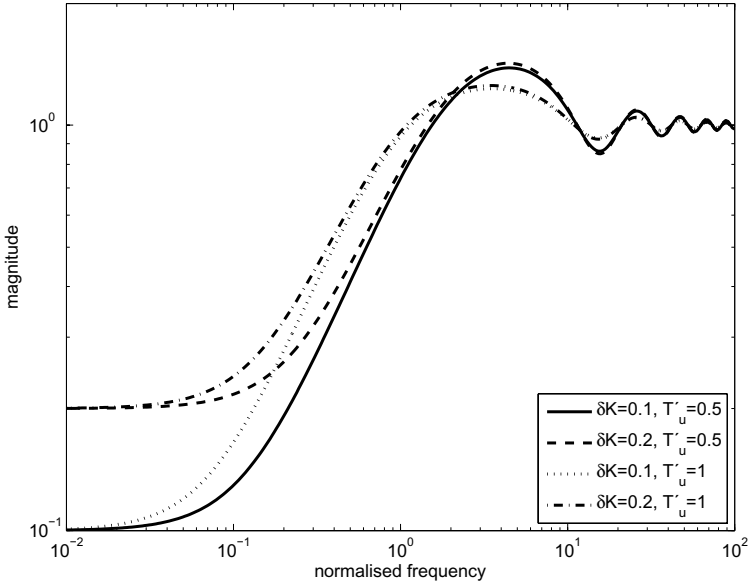


Fig. 2.28. Multiplicative modelling error in the general case for integrative processes for a dead-time error of 30%

Using the same parametrisation as in the stable case, the error can be written as

$$\delta P(j\omega_n) = (1 + \delta K) \frac{1}{1 + j\omega_n T'_u} e^{-j\omega_n \delta L} - 1.$$

Here, only one assumption is made: The dead time is considered such that $L \geq T_u$. Using this condition, two situations are simulated: $T_u = 0.5$ and $T_u = 1$. Figure 2.28 shows the shape of δP for a dead-time error of 30%. The following cases are considered: $\delta K = 0.1, T'_u = 0.5$ (solid line), $\delta K = 0.2, T'_u = 0.5$ (dashed line), $\delta K = 0.1, T'_u = 1$ (dotted line) and $\delta K = 0.2, T'_u = 1$ (dashed-dotted line). Again, the maximum error here is almost defined by the errors in the dead time, while the error in the gain and T_u are more important at low and medium frequencies. Note that the effect of T_u is more important here than in the stable case. This can be explained because an integrative process can be interpreted as a lag-dominant stable process where $\frac{e^{-Ls}}{1+sT} \cong \frac{e^{-Ls}}{sT}$ if $T \gg L$. Furthermore, note that as T_u increases, the maximum value of $\delta P(j\omega)$ decreases and the frequency at which this maximum occurs also decreases.

The previous analysis allows us to conclude that the errors in the estimation of dead time can have a great effect on the unstructured description of the modelling errors of the process. If the dead time is dominant, or at least important, this error will be responsible for driving the closed-loop system

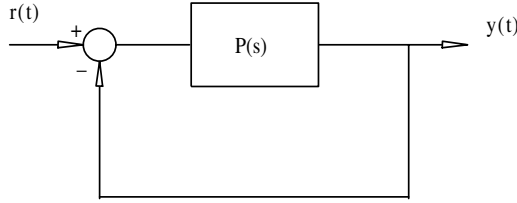


Fig. 2.29. Closed-loop system

to instability. In the next section, and also in the following chapters, this problem is discussed in detail.

2.5 Dead-time Uncertainties and Delay Margin

Because of the important effect of errors in dead-time estimation on robustness, it is usual to define the delay margin of the system as the largest variation in the dead time that can occur in the process $P(s)$ before the closed-loop system in Fig. 2.29 becomes unstable. Using the Nyquist diagram of the system under study, if the Nyquist curve intersects the unitary circle at frequencies ω_i with the corresponding phase margin ϕ_i , then the delay margin D_m is defined as

$$D_m = \min_i \left[\frac{\phi_i}{\omega_i} \right].$$

The following example illustrates the importance of the D_m .

Example 2.7: Consider two systems with open-loop transfer function $P_i(s)$ such that

$$P_i(s) = \frac{K_i}{s(1 + sT_i)}, \quad i = 1, 2,$$

where K_i is tuned to achieve at $\omega_i = 1/T_i$ a phase margin of 45° ($\frac{\pi}{4}$ rad) and a gain margin of ∞ . As the value of the phase of $P_i(j\omega)$ is -135° ($-\frac{3\pi}{4}$ rad) at ω_i , it is easy to compute the value of K_i to achieve the specifications

$$|P_i(j\omega_i)| = \frac{K_i}{\sqrt{2}\omega_i} = 1 \Rightarrow K_i = \sqrt{2}\omega_i.$$

Thus, as these two systems have the same phase margin and gain margin, the same robustness characteristics are expected. Now, consider that a dead time $e^{-L_i s}$ has been neglected in the modelling procedure. What is the maximum value of the neglected dead time L_i that can be considered without loosing closed-loop stability? Or equivalently, what is the value of the D_m of these two systems? Using the definition of D_m , it follows that

$$D_{mi} = \frac{\pi T_i}{4} \quad i = 1, 2.$$

Therefore, the system with higher bandwidth (small value of T_i) becomes unstable with a small value of the dead time. Theoretically, as T_i can be as small as desired, an infinitesimal value of L_i could drive the closed-loop system to instability even when the phase margin is 45° and the gain margin is ∞ .

This analysis can also be made using a time domain approach. In previous cases, the bandwidth of the closed-loop system is related to ω_i , thus to T_i . It is clear then that if T_1 is smaller than T_2 a small value of L can be considered dominant for system 1 and not for system 2. Thus, it is expected that in case 1 the effect of the dead time will be more important.

Even when L is precisely known, the controller design for dead-time processes is not easy. The following section presents the main difficulties of this problem and also indicates a way to deal with it.

2.6 Control Problems Associated with Dead Times

As has been previously mentioned, dead times cause a deterioration of the closed-loop response because of the negative effect on the phase of the system. To illustrate this effect, consider the following example.

Example 2.8: A second-order-plus-dead-time model is considered to be the process model of the heated tank with the long pipe presented in Sect. 2.1.1

$$P(s) = \frac{1}{(1 + 1.5s)(1 + 0.4s)} e^{-Ls} \quad (2.21)$$

First, it is assumed that $L = 0$; that is, the temperature sensor is located in the tank. A simple proportional integral PI controller $C(s) = K_c \frac{T_i s + 1}{T_i s}$ can be tuned for this process to obtain a step response with an overshoot of less than 5% and faster than the open-loop one. Using a root locus diagram and a pole placement of the dominant poles the following tuning is chosen: $T_i = 1.2$, $K_c = 1$.

Figure 2.30 (solid line) shows the response obtained for a unitary step applied at $t = 1$ (dotted line). Consider now that the temperature sensor is located at the end of the pipe causing a dead time $L = 1.5$. There are two main effects caused by the dead time on the closed-loop response: (a) The dead time also appears in the plant output closed loop (the controller cannot avoid this); (b) the response of the closed-loop after the dead time also deteriorates (something can be done about this).

Notice that with the same parameters the PI now gives a very oscillatory response as can be seen in Fig. 2.30 (dashed line) and it is necessary to use a high value of T_i and/or a lower K_c to avoid oscillations. In this case the PI is

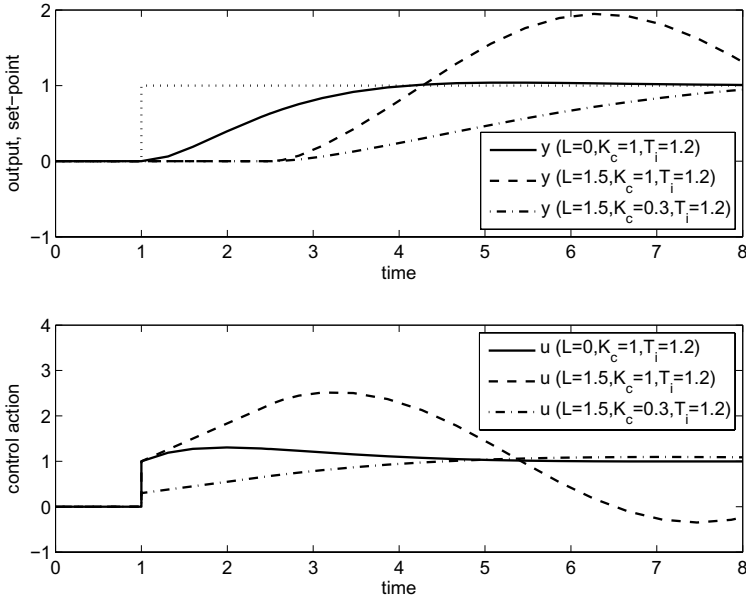


Fig. 2.30. Step response and control action for $K_c = 1$ and $T_i = 1.2$ with $L = 0$; $K_c = 1$ and $T_i = 1.2$ with $L = 1.5$ and $K_c = 0.3$ and $T_i = 1.2$ with $L = 1.5$

re-tuned with $K_c = 0.3$ and $T_i = 1.2$. Note that the oscillations disappear and the price for this is a slower step response (dashed-dotted line in Fig. 2.30); that is, even considering the dead time and knowing its value, the controller cannot give a good closed-loop response.

This can also be interpreted using the frequency domain approach. For the chosen tuning parameters of the PI controller, the open-loop transfer function is given by

$$C(s)P(s) = \frac{K_c(1 + 1.2s)e^{-Ls}}{1.2s(1 + 0.4s)(1 + 1.5s)}.$$

The phase margin PM and cross-over frequency (w) of the system are shown in Fig. 2.31 for the previous cases. Note that with $L = 0$ the system has a $PM \cong 70^\circ$ and $w = w_1 > 0.7$ rad/s. When $L = 1.5$ the $PM \cong 5^\circ$, giving a poor damped closed-loop system. To obtain a nonoscillatory response, gain K_c has to be decreased until a $PM > 60^\circ$ is obtained, which causes a smaller value of the cross-over frequency $w_2 < w_1$. This new value of $w \cong 0.25$ rad/s gives a slower transient after the dead time and is the result of the phase introduced by the dead time. Notice that a PID could be used to improve the performance obtained by the PI for the case of this small dead time. This is expected because a PID controller can be interpreted as a lead-lag controller where the derivative action is tuned to advance the phase of the open-loop

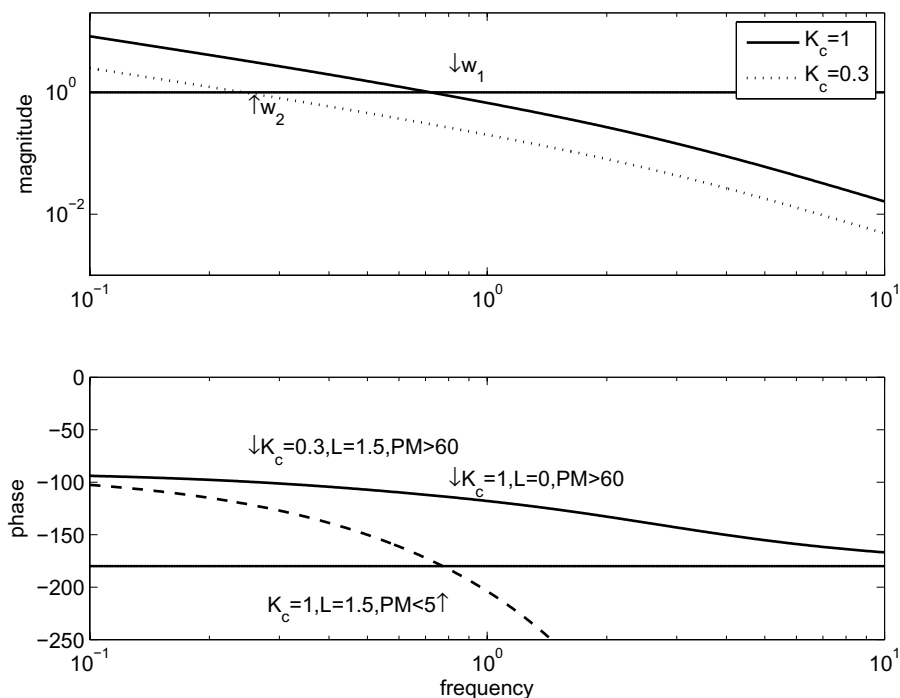


Fig. 2.31. Frequency response $C(j\omega)P(j\omega)$ for $K_c = 1$ and $T_i = 1.2$ with $L = 0$; $K_c = 1$ and $T_i = 1.2$ with $L = 1.5$ and $K_c = 0.3$ and $T_i = 1.2$ with $L = 1.5$

transfer function $C(s)P(s)$ close to the cross-over frequency, giving a better phase margin. However, if a faster closed-loop response is specified or a process with a higher dead time is considered, the phase advance of the PID will not be sufficient to attempt the desired closed-loop performance. Then, the order of the controller could again be increased, to improve the performance of the PID, but this process has no end. A different approach to this problem can be considered, as shown in the next section.

2.7 A Predictor-based Solution: The Smith Predictor

Let us return to the simple example of the heated tank. If the temperature sensor is installed at the beginning of the pipe the new process model has no dead time and a simple PI controller could be tuned to achieve good performance. This is obviously the ideal solution for a dead-time process: “To feed the plant output without the dead time”. Figure 2.32 shows this ideal situation in a general case, where the controller $C(s)$ is tuned using only $G(s)$ and the real output $y(t)$ is the output of $G(s)$ (y_1) delayed L units of time

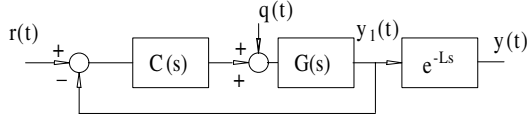


Fig. 2.32. “Ideal” control for dead-time processes

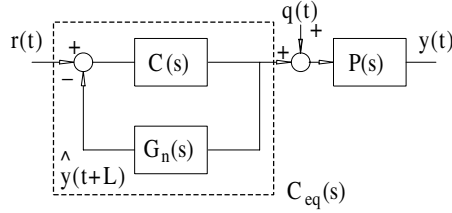


Fig. 2.33. Open-loop predictor

($y(t) = y_1(t - L)$). In this situation the dead time has no effect on the closed-loop transients, as the closed-loop transfer function is

$$\frac{Y(s)}{R(s)} = \frac{C(s)G(s)e^{-Ls}}{1 + C(s)G(s)}.$$

The real implementation of this solution is, in general, not possible in practice mainly because the sensor cannot be installed in the desired position and/or the process dead time is not caused by mass transportation. A simple solution for this problem can be obtained using the idea of prediction and will be applied here to a stable process. If a dead-time-free model $G_n(s)$ of the plant $P(s) = G(s)e^{-Ls}$ is considered, it is possible to feed the output of this model to the controller as shown in Fig. 2.33. In this structure if $G_n(s) = G(s)$ the primary controller $C(s)$ can be tuned considering only $G(s)$ and the obtained closed-loop performance is the same as in the ideal case: $\frac{Y(s)}{R(s)} = \frac{C(s)G(s)e^{-Ls}}{1 + C(s)G(s)}$. The equivalent controller for this system is

$$C_{eq}(s) = \frac{C(s)}{1 + C(s)G_n(s)},$$

which, as can be seen in Fig. 2.33, operates in an open-loop manner. This strategy is known as open-loop predictor based control and it is clear that it cannot be used in practice because the controller does not “see” the effect of the disturbances and also model mismatches are not taken into account and, therefore, all the beneficial properties of feedback disappears.

A better solution for this problem was proposed by Smith in the 1950s [126] based on a closed-loop predictor structure of the open-loop-stable process. This structure is shown in Fig. 2.34 and is known in literature as the “Smith predictor” (SP). In this strategy, the prediction at time t is computed

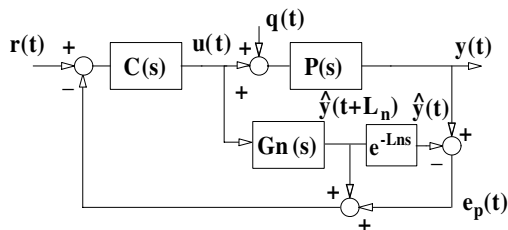


Fig. 2.34. The Smith predictor structure

by the use of a model of the plant without dead time ($G_n(s)$) and, in order to correct the modelling errors, the difference between the output of the process and the model (including the dead time $P_n(s) = G_n(s)e^{-L_n s}$) is fed back, as can be seen in Fig. 2.34. With this structure, if there are no modelling errors or disturbances, the error between the current process output and the model output ($e_p(t)$) will be null and the controller can be tuned as if the plant had no dead time. Thus, in the nominal case this structure gives the same performance as the ideal solution. The fundamental properties of the SP will be analysed in detail in Chap. 5, but to illustrate the use of this strategy an example is given here.

Example 2.9: The SP will be tuned for the process $P(s) = \frac{1}{(1+1.5s)(1+0.4s)}e^{-Ls}$ but considering two different response $L = 0.5$ and $L = 5$. Considering perfect modelling a PI with $K_c = 3$ and $T_i = 1.5$ can be used in both cases, because the predictor can cope with the dead time. Figure 2.35 shows the closed-loop response for both cases that have, after dead time, better transient than the one obtained when $L = 0$ in Example 2.8. Also note that the control action is the same in both cases because the controller acts as if the process had no dead time.

This analysis shows that the control of dead-time processes can be solved efficiently using predictive control structures. It is also clear that this simple example does not consider other important objectives in control design like robustness or disturbance rejection. These points will be analysed in detail in the following chapters (see Exercise 2.12).

2.8 Summary

Dead-time processes are common in industry and it has been shown that the controller design for systems exhibiting dead time is more difficult than for dead-time-free systems. The difficulties associated with the design are mainly due to the effects of the dead time on the phase lag of the process and the difficulties of estimating the dead time.

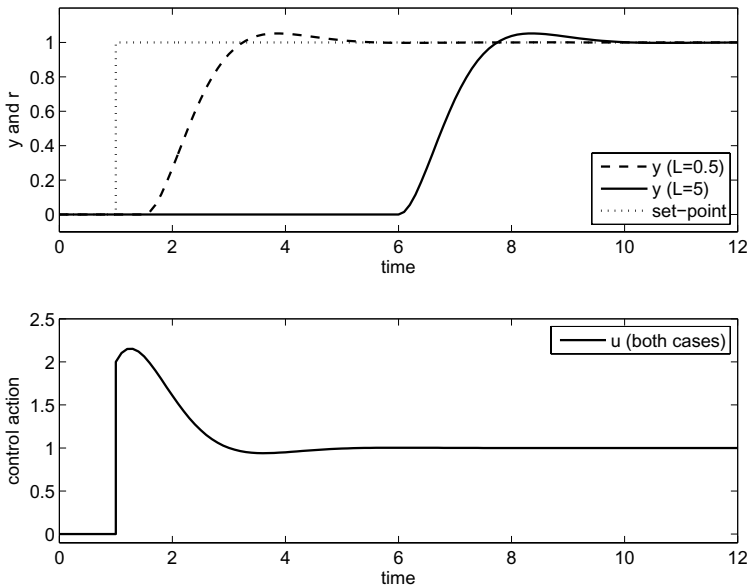


Fig. 2.35. Step response and control action for the SP with $K_c = 3$ and $T_i = 1.5$ for a step change in the set-point with $L = 0.5$ and $L = 5$

This chapter shows how dead-time systems can be modelled and what the difficulties for controlling dead-time systems with classical controllers are. The chapter also shows that a predictor based control structure is an appropriate strategy for use with dead-time systems; in fact, with perfect modelling this approach gives an ideal solution. All the ideas and concepts of this chapter are used in the controller design of the following chapters, from the more simple PID to model predictive controllers.

2.9 Further Reading

Modelling of processes exhibiting a dead time can be found in several process control books [93]. In particular, details of the solar plant described in Sect. 2.1.2 are given in [16, 12]. The industrial sugar process presented in this chapter is described in [89]. A more complete analysis of the polynomial approximations of dead time can be found in [58]. Other frequency domain approaches for dead-time systems can be found in [104]. The simple models used for the representation of dead-time systems are the same as those used in the PID approach [4]. The books of Morari and Zafirou [70] and Skogestad and Postlethwaite [124] present a complete analysis of the model uncertainty description used in this chapter. The control problems associated to

dead-time processes are discussed in several papers and books, for instance, [99, 4, 70].

2.10 Exercises

2.1. For the heated tank and the long pipe in Fig. 2.1 consider that the distance from the tank to temperature sensor is 5 m and the diameter of the pipe is 0.05 m. Assume that the flow can vary between 10 and 30 l/min. Compute the maximum and minimum values of the transport dead time.

2.2. Consider the process with transfer function $P(s) = \frac{3e^{-Ls}}{(1+5s)^2}$. Compute the gain and phase margin of the system for the following values of the dead time $L = 0, 2, 5, 10$.

2.3. For the process with transfer function $P(s) = \frac{2e^{-Ls}}{1+3s}$, compute at least two polynomial approximate models for the dead time. Compare the behaviour of the models and the process in the time and frequency domains.

2.4. For the process in the previous exercise, compute a first-order Padé approximation of the dead time. Compare the gain and phase margin of the model and the process for the following values of the dead time $L = 0, 2, 5, 10$.

2.5. Perform a comparative analysis in the time domain and frequency domain of the lag, Taylor and Padé approximations of second order.

2.6. For the process $P(s) = \frac{2}{1+2s}e^{-6.2s}$ and a sample time $T_s = 0.5$ compute two discrete models. $P_1(z)$ considering the approximate model of the dead-time error introduced by the discretisation and $P_2(z)$ neglecting this error. Consider a dead-time-estimation error of 15%. Compare the behaviour of the process with different dead times and the two models in the time and frequency domain.

2.7. Consider the process with transfer function $P(s) = \frac{1}{(1+2s)^5}$. Compute an FOPDT and an SOPDT model to approximate the step response of $P(s)$. Compare the behaviour of the models and the process in the time and frequency domains.

2.8. Given the process with transfer function $P(s) = \frac{1}{s(1+5s)^2}$, compute an IPDT and an SOIPDT model to approximate the step response of $P(s)$. Compare the behaviour of the models and the process in the time and frequency domains.

2.9. For the process $P(s) = \frac{2e^{-Ls}}{1+3s}$, first consider that $L = 0$ and tune a PI controller to achieve a step response with an overshoot of less than 10% and faster than the open-loop response of the process. With the PI obtained find the value of $L > 0$ that drives the closed-loop system to instability.

2.10. Analyse the multiplicative modelling error in the general case presented in Sect. 2.4.2. Consider different intervals for the variation of $T_n, T'_u, \delta L, \delta T$ and δK . Verify the importance of the dead-time-estimation error. Compare to the case where only dead-time errors are considered. Analyse the effects on the frequency $\omega_{\max}$ where $\delta P_{\max}$ is achieved. Show that as δL increases $\delta P_{\max}$ increases and $\omega_{\max}$ decreases. Compute an uncertainty bound for the two cases.

2.11. Analyse the modelling error for the following processes and models. In each case compute the maximum error and the corresponding frequency. Use normalised graphics.

- (a) $P(s) = \frac{2.2e^{-5s}}{(1+5s)(1+s)^3}$ and $P_n(s) = \frac{2e^{-7s}}{(1+6s)}$.
- (b) $P(s) = \frac{1.1e^{-4s}}{(1+3s)s}$ and $P_n(s) = \frac{e^{-6.5s}}{s}$.

2.12. Using simulation, analyse the behaviour of the controller in Example 2.9 for an error in the dead time of 10% and 30%. Try to re-tune the primary controller to improve performance.



<http://www.springer.com/978-1-84628-828-9>

Control of Dead-time Processes

Normey-Rico, J.E.

2007, XXV, 462 p., Softcover

ISBN: 978-1-84628-828-9