Common Process Types

CSE 425 Industrial Process Control

Lecture 3

Outline

- Common process types
  - Self-regulating
    - First order
    - Higher order
  - Non self-regulatory
    - Integrating
    - Runaway
  - Dead time
- Steady-state characteristics
- Dynamic characteristics

Objective

- Predict the response of common processes to typical inputs (impulse, step) without any control
- This gives us idea about what performance we could expect from the controller
- Example: Bus vs. bicycle
  - Bus difficult to control, bicycle easy to control

Self-Regulating Processes

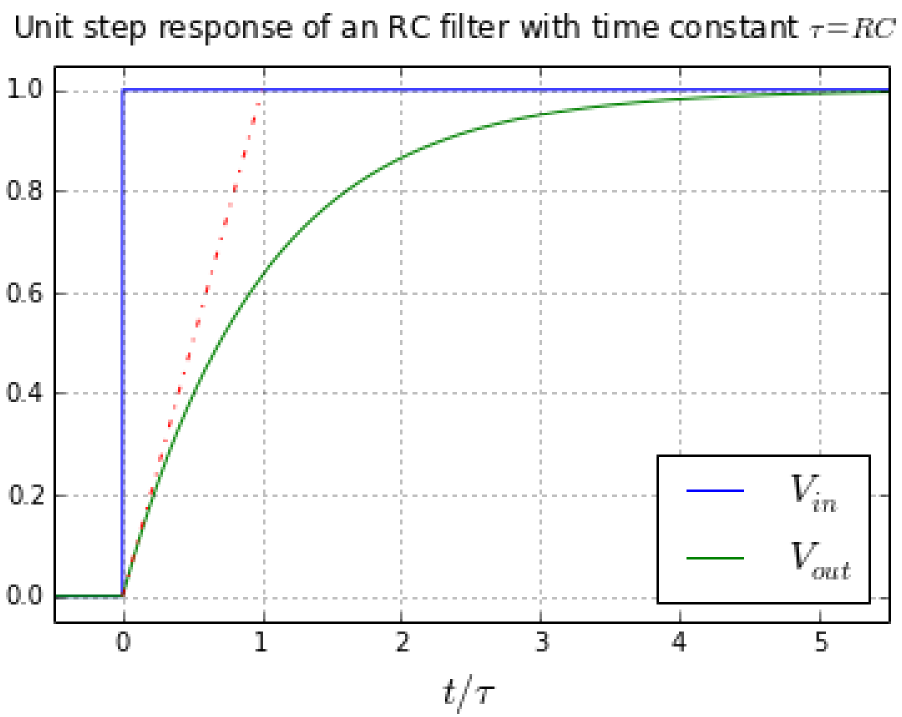
- The process by itself (without any control) reaches equilibrium.
  - Most processes are self-regulating (car, motor speed)
  - No overshoot
- First-order system is self-regulating as its step response reaches a steady state.

$$\tau_p \frac{dy}{dt} + y = K_p u(t) \qquad \frac{Y(s)}{U(s)} = \frac{K_p}{\tau_p s + 1}$$

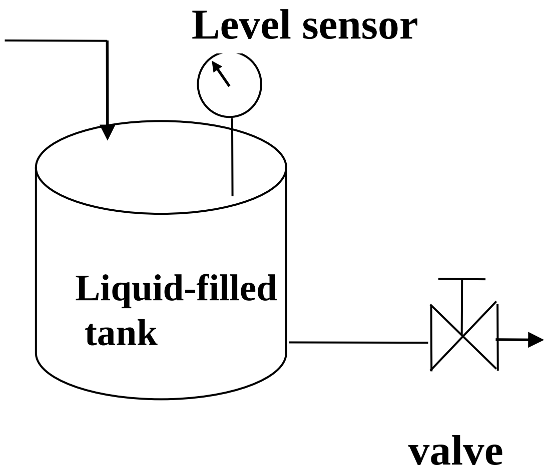
- where τ_p is the time constant and K_p is the steady-state gain.

Self-Regulating Processes

- First-order process is also called single capacity or first-order lag.
- Capacity is a volume where mass or energy is stored.
- First-order lag is the favorable dynamics: easy to control



Example



$$\frac{dV}{dt} = A\frac{dh}{dt} = F_{in} - F_{out}$$

$$F_{in}(t) \neq f(h)$$ i.e. no controller yet

$$F_{out}(t) = bh$$

$$\frac{H}{F_{in}} = \frac{K}{\tau s + 1}$$

$$\tau : residence \quad time = A/b$$

$$K = 1/b$$

Steady-State Process Characteristics

- **Process Graph** (for self-regulating processes): describes the process input (MV) to process output (PV) relationship **in steady state**
- Changes with the load

PV

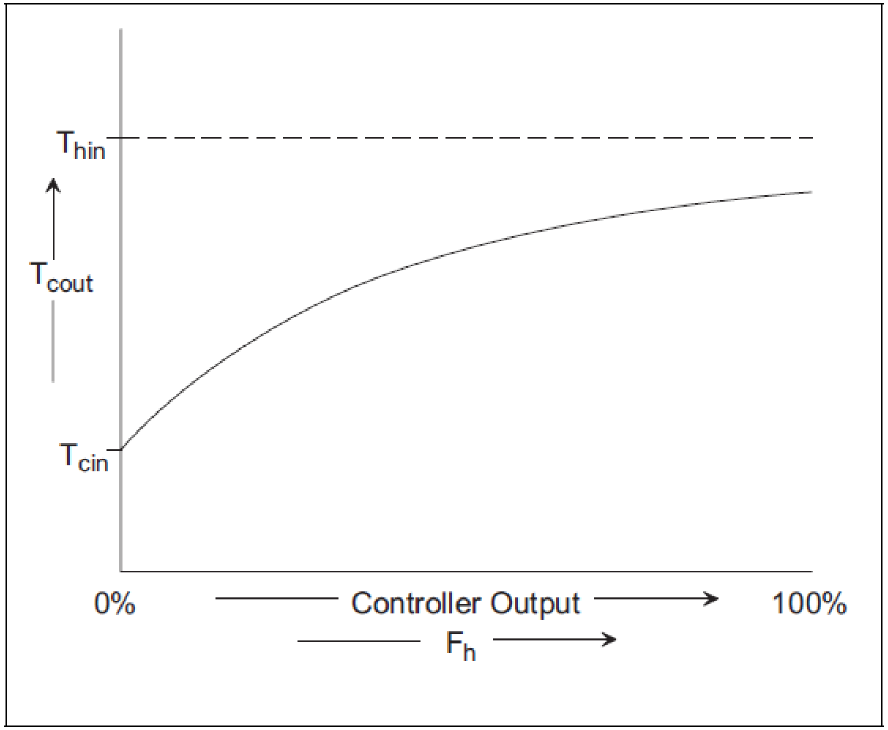


Figure 3-2. The Process Graph

MV: Valve opening

Steady-State Characteristics

- **Operating point** (in terms of PV):
  - it will be the function of the controller to find the valve opening (MV) required to achieve the desired PV value

- **Process Gain:**
  - It equals the slope of the line or tangent to process graph at the operating point.
  - May be +ve or –ve

Process Gain

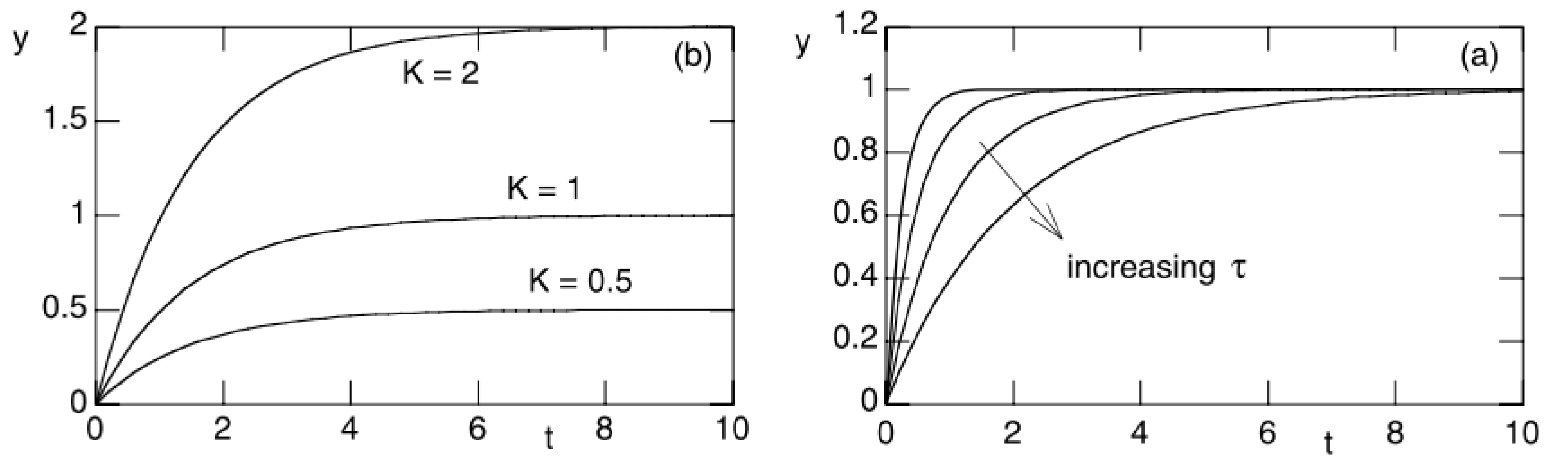
- The process gain (also called dc gain) is the percent (fractional) change in PV divided by the percent (fractional) change in MV after the process has fully responded to the change

$$K_p = \frac{\left(\dfrac{\Delta PV}{PV \text{ range}}\right)}{\left(\dfrac{\Delta MV}{MV \text{ range}}\right)}$$

- The gain is dimensionless (%/%)
- The process gain depends on the operating point.
    - This makes controller tuning more difficult.

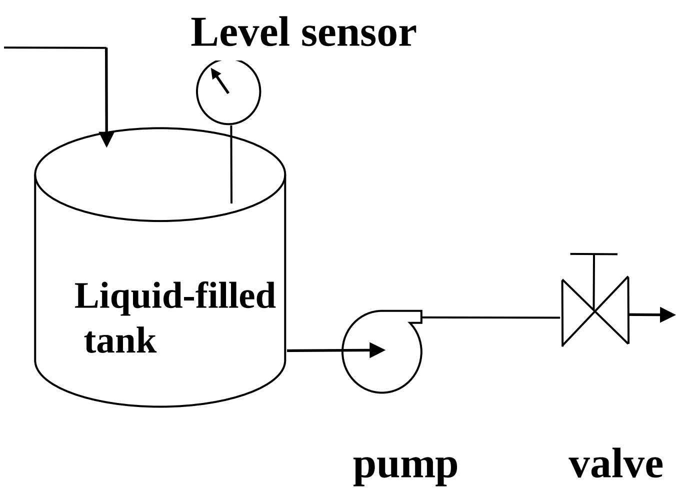
Step response of a first-order model

- The time constant τ determines the speed of the response.
- If it is negative, the process is unstable.



Integrating processes

- This is a type of non-self regulating process because the output variable tends to “drift” linearly far from desired values without reaching equilibrium (fixed rate of increase).
- For example, consider a tank whose flows are un-manipulated (The pump at the outlet keeps the outlet flow fixed regardless of the level (i.e. we do not apply control or manual correction).



$$\frac{dV}{dt} = A\frac{dh}{dt} = F_{in} - F_{out}$$

$$F_{in}(t) \neq f(h)$$

$$F_{out}(t) \neq f(h)$$

Integrating Processes

$$\frac{dV}{dt} = A\frac{dh}{dt} = F_{in} - F_{out}$$

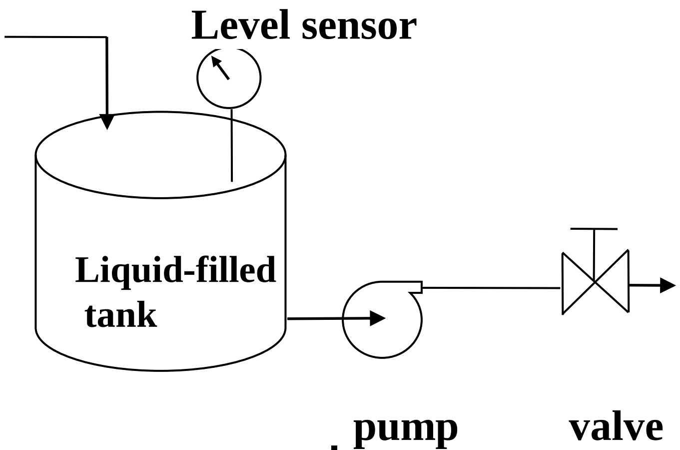
$$h(t) = \frac{1}{A}\int_0^t (F_{in}(\lambda) - F_{out}(\lambda))d\lambda$$

$$\boxed{H(s) = \frac{1}{As}(F_{in}(s) - F_{out}(s))}$$

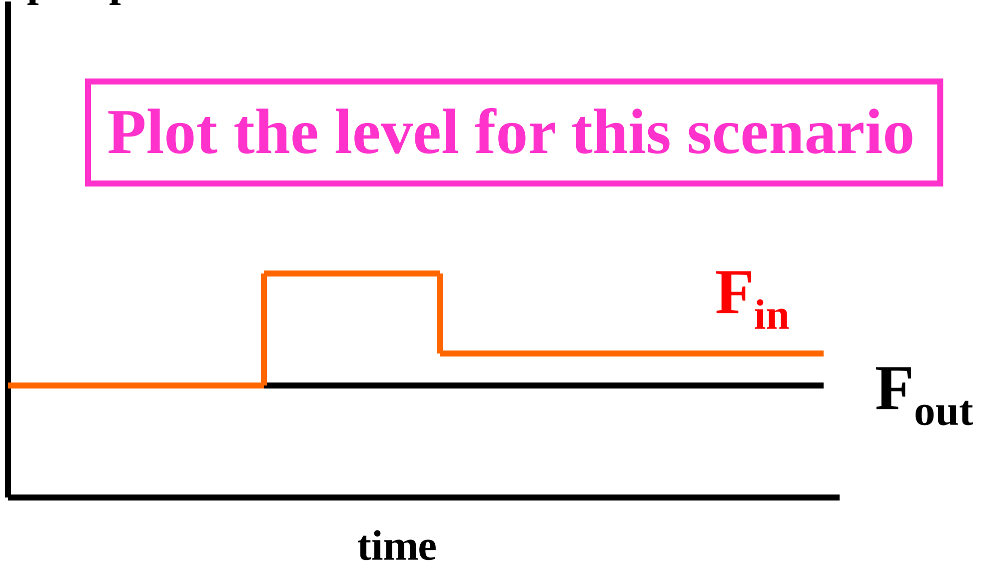
Integrating process has a pole at $s = 0$.

- These systems are termed "pure integrators" because they integrate the difference between **in** and **out** flows.
- Feedback control is necessary for these processes.
- To tune a controller for such processes, we try to stabilize the level first before applying a step in the inlet valve

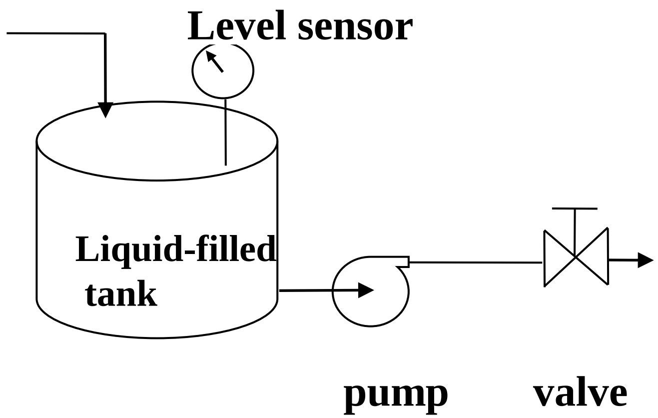
Question



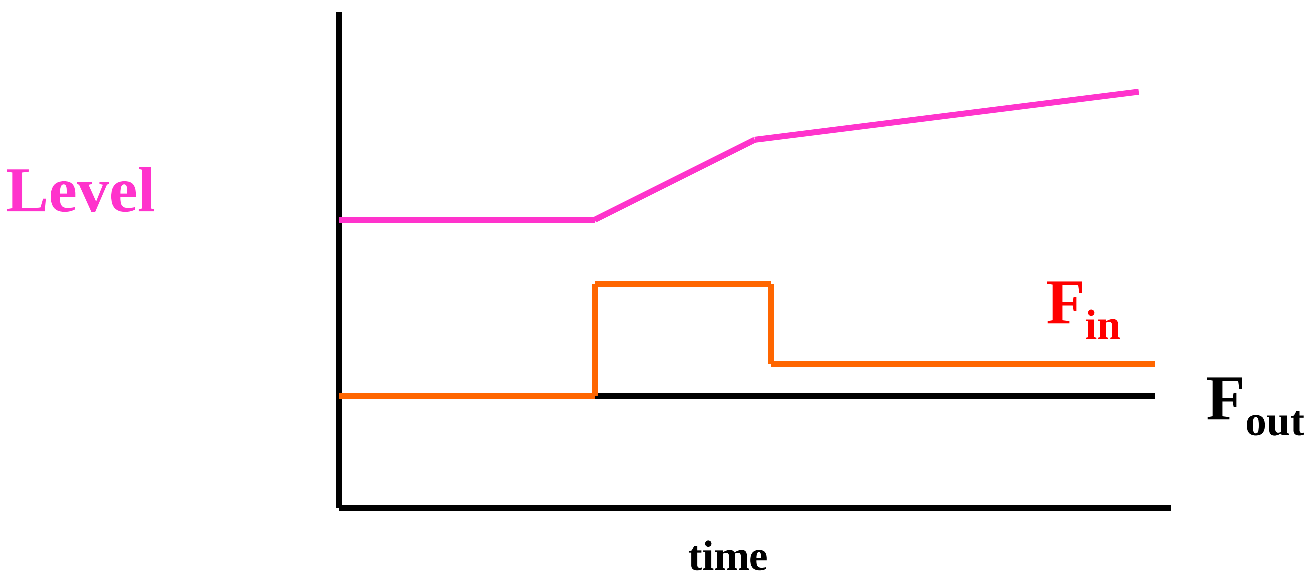
$$\frac{dV}{dt} = A\frac{dh}{dt} = F_{in} - F_{out}$$



Answer



$$\frac{dV}{dt} = A\frac{dh}{dt} = F_{in} - F_{out}$$



Second-order models

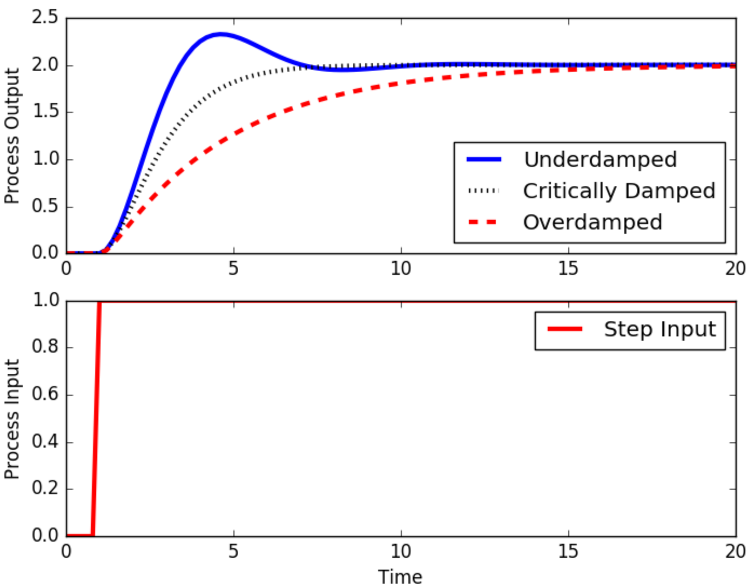
$$\frac{Y(s)}{U(s)} = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

- where ω_n is the natural (undamped) frequency, ζ is the damping ratio, and K is the steady-state gain.
- The system has two poles

$$p_{1,2} = -\xi\omega_n \pm \omega_n\sqrt{\xi^2 - 1}.$$

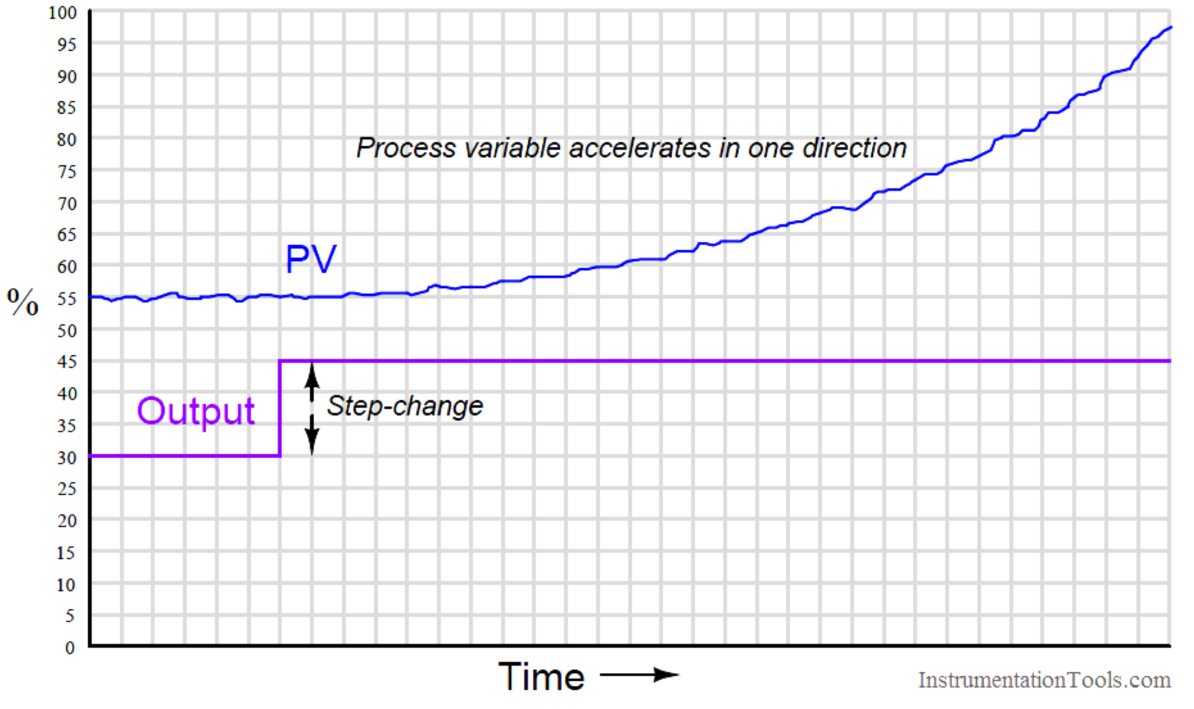
Second-order models

- Usually, processing equipment *doesn't have* oscillatory behavior. They are first-order, second-order over-damped, or high-order).
- A 2nd process that exhibits oscillatory behavior (under-damped response) is most often the result of implementing a controller. Exception is the car shock absorber (mechanical systems)



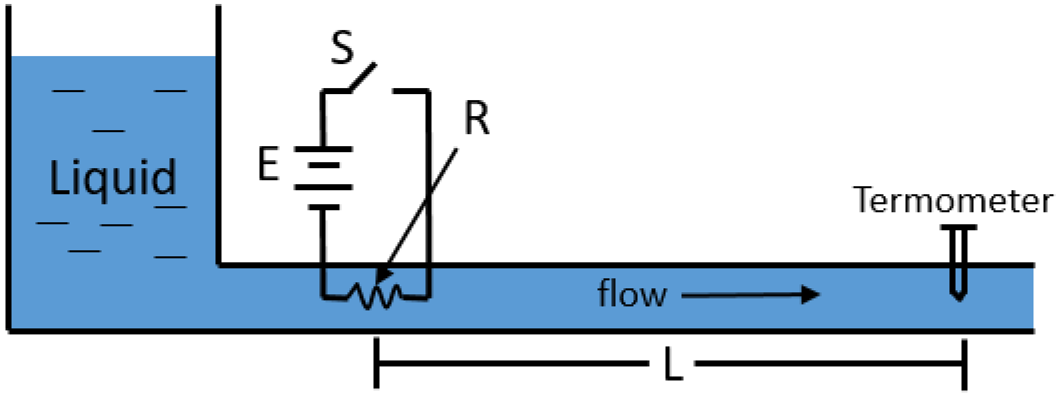
Runaway (Unstable) Processes

- The process output keeps increasing very rapidly without reaching equilibrium (the rate is itself increasing).
- Example: some exothermic reactions (Wade, Chap. 3)



Processes with dead time

- Many chemical processes involve a time delay between input and output (mostly measured in minutes).
- This may be due to the time required for a slow chemical sensor (e.g. analyzer) to respond or for a fluid to travel down a pipe.
- **Time delay** is also called **dead time** or **transport delay**.

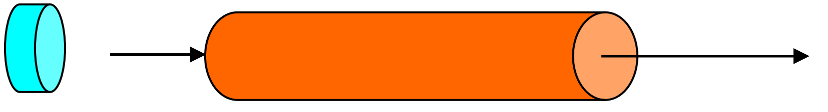


Processes with dead time

- The location of the sensor is determined by the process engineer, not the control engineer.
- In feedback control of a process with time delay, the process will not respond quickly to control actions. This causes the controller to *over-react* resulting in oscillatory response
- Systems with long dead time are more difficult to control. You have to be careful (conservative) when you tune a controller for these processes. You should use lower controller gains.
- The larger the time delay in a control loop, the less stable it becomes.

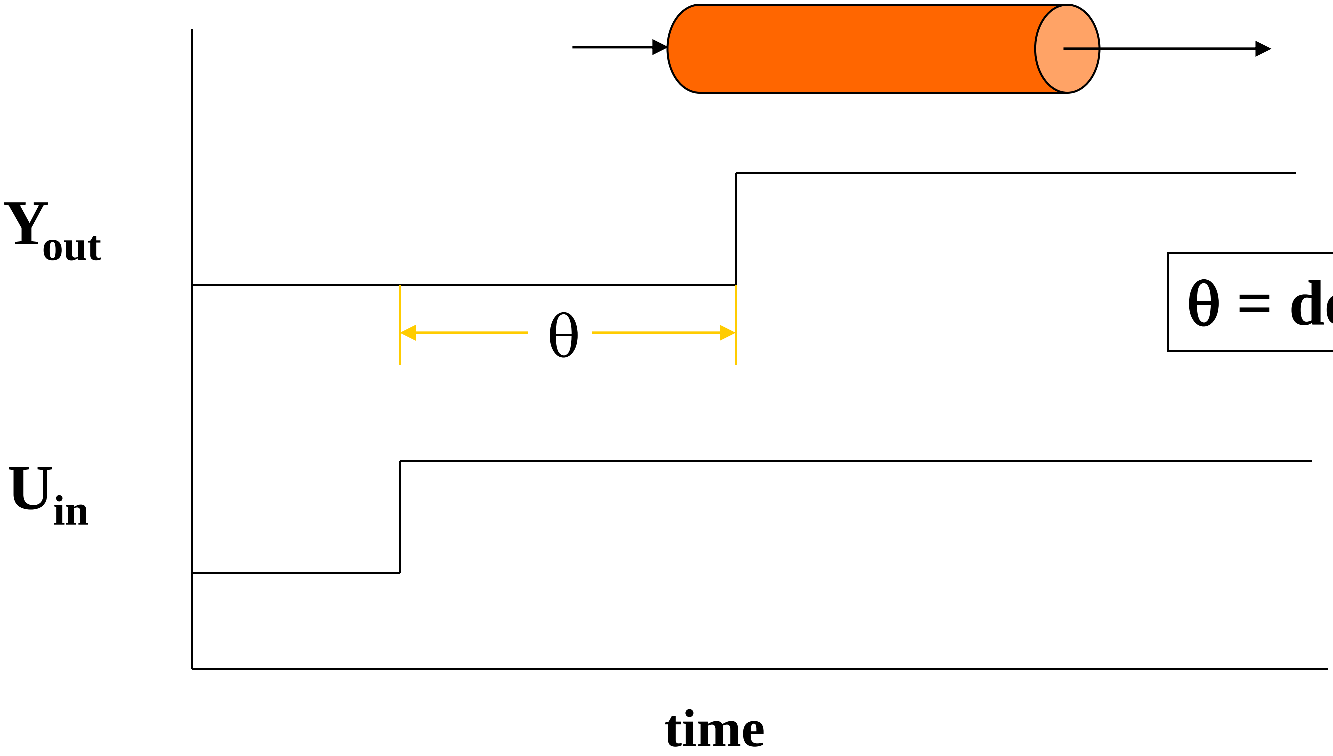
Example

Let's consider plug flow through a pipe. Plug flow has no backmixing; we can think of this as a hockey puck traveling in a pipe.



What is the dynamic response of the outlet fluid property (e.g., concentration) to a step change in the inlet fluid property?

Laplace Transform of a dead time element



θ = dead time

The dynamic model for dead time is

$$y(t) = u(t-\theta) \quad \Rightarrow \quad Y(s) = e^{-\theta s} U(s)$$

Pade Approximation of Time Delay

- Dead time can be approximated as a ratio of two polynomials in s as follows:

$$e^{-\theta\, s} = e^{-\frac{\theta}{2}\, s} e^{-\frac{\theta}{2}\, s} = \frac{e^{-\frac{\theta}{2}\, s}}{e^{\frac{\theta}{2}\, s}}.$$

- Recall Taylor series expansion

$$e^{x} = 1 + x + \frac{x^2}{2!} + \cdots$$

- The first-order Pade approximation is obtained by keeping only first order terms:

$$e^{-\theta\, s} \approx \frac{1 - \frac{\theta}{2} s}{1 + \frac{\theta}{2} s}.$$

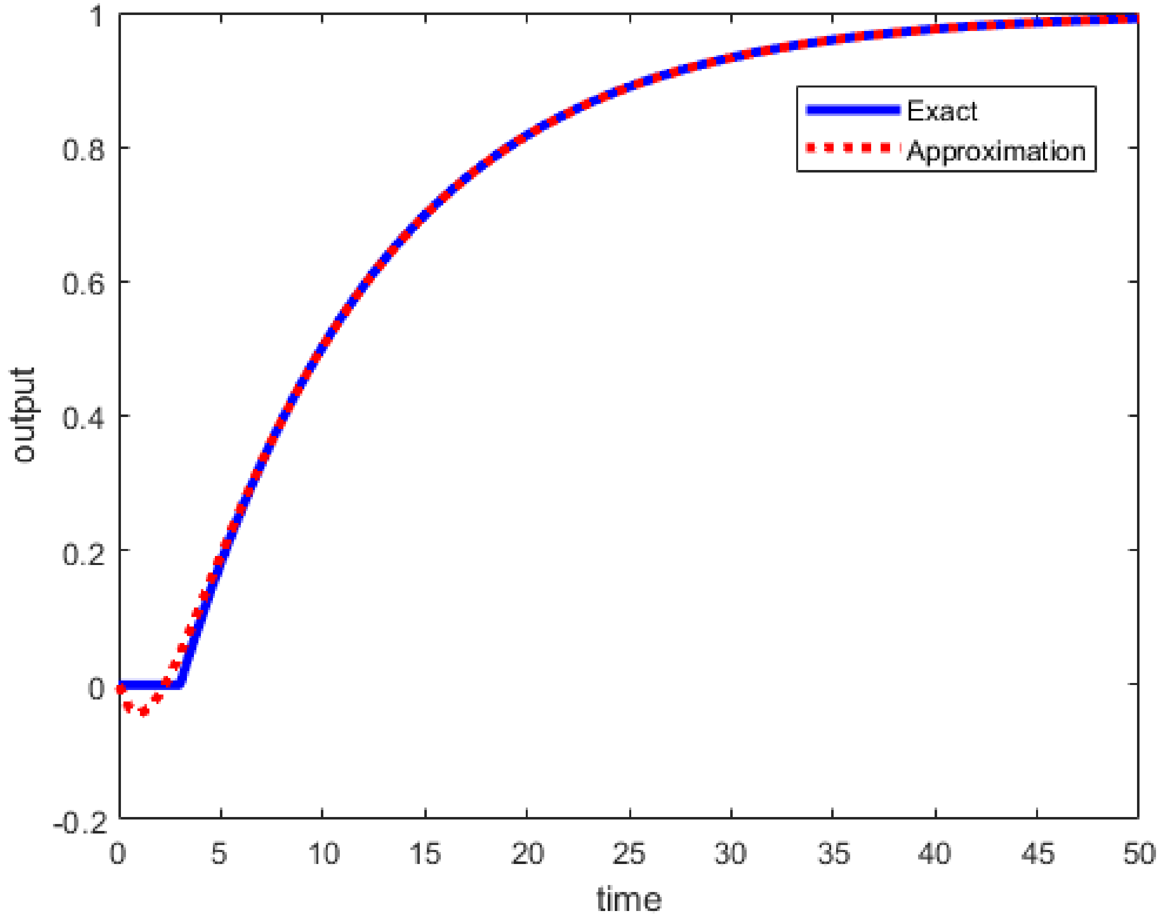
Example

- Use first-order Pade approximation to plot the unit-step response of first order with a dead-time transfer function:

$$\frac{Y(s)}{U(s)} = \frac{e^{-3s}}{10s+1}$$

- Making use of the first order Pade approximation, we can construct a plot with the approximation

$$\frac{Y(s)}{U(s)} = \frac{1-1.5s}{(10s+1)(1+1.5s)}.$$



The approximation is very good except near t = 0, where the approximate response dips below. A better approximation can be obtained with, e.g., a second-order Pade approximation.

MATLAB Code

```
% Process with dead time
G = tf([1],[10 1],'iodelay',3)
t = 0:0.5:50;
y1 = step(G,t);

% First-order Padé approximation
th = 3;
P1 = tf([-th/2 1],[th/2 1]);
G1 = tf(1,[10 1]);
y2 = step(G1*P1,t);

plot(t,y1,'b',t,y2,':r','Linewidth',3);
xlabel('time')
ylabel('output')
legend('Exact','Approximation')
```