Model Predictive Control

Lecture 2: Modeling and Simulation

John Bagterp Jørgensen

Technical University of Denmark
Department of Applied Mathematics and Computer Science

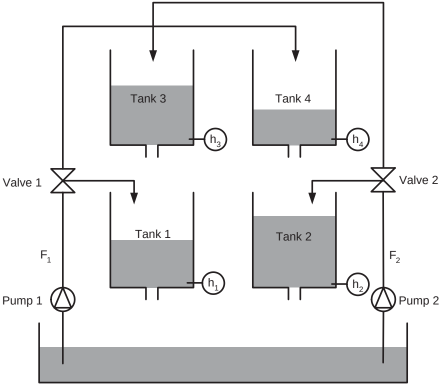
02619 Model Predictive Control

Learning Objectives

After this lecture you should be able to

1. Apply conservation of mass to develop simple first-principle models
2. Simulate systems described by ODEs using Matlab
3. Compute steady-states and linearize a system around a steady state
4. Discretize a linear continuous-time state space system
5. Derive transfer functions for linear state-space systems
6. Do simulations of stochastic systems
7. Discretize a continuous-time stochastic system

4-Tank System - Motivating Example



Conservation Principle

Physical models are based on conservation principles.

1. Conservation of mass
2. Conservation of energy
3. Conservation of momentum (force)

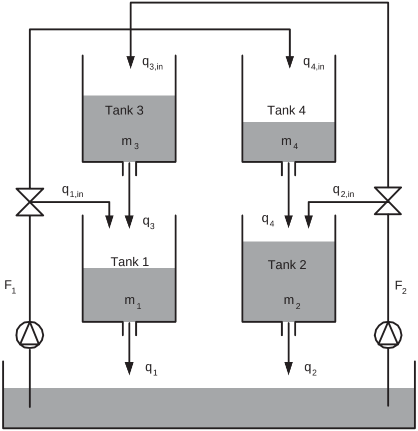
The general derivation of the system equations have the form

$$\text{Accumulated} = \text{Influx} - \text{Outflux} + \overbrace{\text{Produced} - \text{Consumed}}^{=\text{Generated}}$$

For non-reactive systems the generation term is absent

$$\text{Accumulated} = \text{Influx} - \text{Outflux}$$

Example - Tank 1



$$\text{Accumulated} = \text{In - Out}$$

with

$$\text{Accumulated} = m_1(t+\Delta t) - m_1(t)$$
$$\text{In} = \rho q_{1,in}(t)\Delta t + \rho q_3(t)\Delta t$$
$$\text{Out} = \rho q_1(t)\Delta t$$

$$\underbrace{m_1(t+\Delta t) - m_1(t)}_{\text{Accumulated}} = \underbrace{\rho q_{1,in}(t)\Delta t + \rho q_3(t)\Delta t}_{\text{In}} - \underbrace{\rho q_1(t)\Delta t}_{\text{Out}}$$

Example - Tank 1

1. Conservation of mass

$$\underbrace{m_1(t+\Delta t) - m_1(t)}_{\text{Accumulated}} = \underbrace{\rho q_{1,in}(t)\Delta t + \rho q_3(t)\Delta t}_{\text{In}} - \underbrace{\rho q_1(t)\Delta t}_{\text{Out}}$$

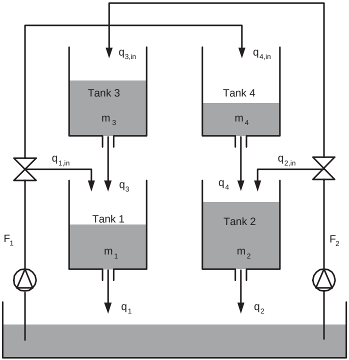
2. Divide by Δt

$$\frac{m_1(t+\Delta t) - m_1(t)}{\Delta t} = \rho q_{1,in}(t) + \rho q_3(t) - \rho q_1(t)$$

3. Let $\Delta t \to 0$

$$\frac{dm_1(t)}{dt} = \rho q_{1,in}(t) + \rho q_3(t) - \rho q_1(t)$$

4-Tank System - Model



Mass balances

$$\frac{dm_1}{dt}(t) = \rho q_{1,in}(t) + \rho q_3(t) - \rho q_1(t) \qquad m_1(t_0) = m_{1,0}$$

$$\frac{dm_2}{dt}(t) = \rho q_{2,in}(t) + \rho q_4(t) - \rho q_2(t) \qquad m_2(t_0) = m_{2,0}$$

$$\frac{dm_3}{dt}(t) = \rho q_{3,in}(t) - \rho q_3(t) \qquad m_3(t_0) = m_{3,0}$$

$$\frac{dm_4}{dt}(t) = \rho q_{4,in}(t) - \rho q_4(t) \qquad m_4(t_0) = m_{4,0}$$

Inflows

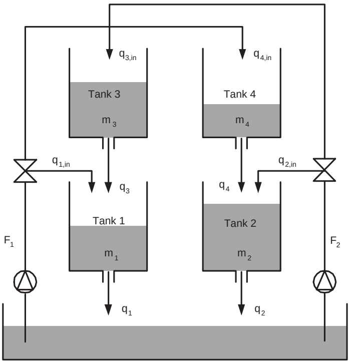
$$q_{1,in}(t) = \gamma_1 F_1(t) \qquad q_{2,in}(t) = \gamma_2 F_2(t)$$

$$q_{3,in}(t) = (1-\gamma_2) F_2(t) \qquad q_{4,in}(t) = (1-\gamma_1) F_1(t)$$

Outflows

$$q_i(t) = a_i \sqrt{2 g h_i(t)} \quad h_i(t) = \frac{m_i(t)}{\rho A_i} \quad i \in \{1,2,3,4\}$$

4-Tank System - Model



System of ordinary differential equations

$$\dot{x}(t) = f(x(t), u(t)) \qquad x(t_0) = x_0$$

with the vectors defined as

$$x = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} \qquad u = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

This is a non-stiff ODE system
as all processes take place on the same time-scale

Generic Input-Output Model

$$\frac{dx(t)}{dt} = f(x(t), u(t)) \qquad x(t_0) = x_0$$ Process model

$$y(t) = g(x(t))$$ Sensor function

$$z(t) = h(x(t))$$ Output function

Simulation in Matlab

The model

$$\dot{x}(t) = f(t, x, u, p) \qquad x(t_0) = x_0$$

may be implemented in Matlab as

```
function xdot = ProcessModel(t,x,u,p)
% Process Model   dx/dt = f(t,x,u,p)
%
% Computes the RHS of dx/dt = f(t,x,u,p)
% It stores the result in xdot.
```

...

and called using

```
[T,X] = ode15s(@ProcessModel,[t0 tf],x0,odeOptions,u,p)
```

Model for the 4-Tank System

```
function xdot = FourTankSystem(t,x,u,p)
% FOURTANKSYSTEM   Model dx/dt = f(t,x,u,p) for 4-tank System
%
% This function implements a differential equation model for the
% 4-tank system.
%
% Syntax: xdot = FourTankSystem(t,x,u,p)

% Unpack states, MVs, and parameters
m     = x;                        % Mass of liquid in each tank [g]
F     = u;                        % Flow rates in pumps [cm3/s]
a     = p(1:4,1);                 % Pipe cross sectional areas [cm2]
A     = p(5:8,1);                 % Tank cross sectional areas [cm2]
gamma = p(9:10,1);                % Valve positions [-]
g     = p(11,1);                  % Acceleration of gravity [cm/s2]
rho   = p(12,1);                  % Density of water [g/cm3]

% Inflows
qin = zeros(4,1);
qin(1,1) = gamma(1)*F(1);         % Inflow from valve 1 to tank 1 [cm3/s]
qin(2,1) = gamma(2)*F(2);         % Inflow from valve 2 to tank 2 [cm3/s]
qin(3,1) = (1-gamma(2))*F(2);     % Inflow from valve 2 to tank 3 [cm3/s]
qin(4,1) = (1-gamma(1))*F(1);     % Inflow from valve 1 to tank 4 [cm3/s]

% Outflows
h = m./(rho*A);                   % Liquid level in each tank [cm]
qout = a.*sqrt(2*g*h);            % Outflow from each tank [cm3/s]

% Differential equations
xdot = zeros(4,1);
xdot(1,1) = rho*(qin(1,1)+qout(3,1)-qout(1,1));   % Mass balance Tank 1
xdot(2,1) = rho*(qin(2,1)+qout(4,1)-qout(2,1));   % Mass balance Tank 2
xdot(3,1) = rho*(qin(3,1)-qout(3,1));             % Mass balance Tank 3
xdot(4,1) = rho*(qin(4,1)-qout(4,1));             % Mass balance Tank 4
```

Define Simulation Parameters

```
% ---------------------------------------------------------------
% Parameters
% ---------------------------------------------------------------
  a1 = 1.2272     %[cm2] Area of outlet pipe 1
  a2 = 1.2272     %[cm2] Area of outlet pipe 2
  a3 = 1.2272     %[cm2] Area of outlet pipe 3
  a4 = 1.2272     %[cm2] Area of outlet pipe 4

  A1 = 380.1327   %[cm2] Cross sectional area of tank 1
  A2 = 380.1327   %[cm2] Cross sectional area of tank 2
  A3 = 380.1327   %[cm2] Cross sectional area of tank 3
  A4 = 380.1327   %[cm2] Cross sectional area of tank 4


  gamma1 = 0.45;  % Flow distribution constant. Valve 1
  gamma2 = 0.40;  % Flow distribution constant. Valve 2

  g = 981;        %[cm/s2] The acceleration of gravity
  rho = 1.00;     %[g/cm3] Density of water

  p = [a1; a2; a3; a4; A1; A2; A3; A4; gamma1; gamma2; g; rho];
% ---------------------------------------------------------------
```

Simulation Scenario and Simulation

```
% ---------------------------------------------------------------
% Simulation scenario
% ---------------------------------------------------------------
  t0 =    0.0;        % [s] Initial time
  tf = 20*60;         % [s] Final time

  m10 = 0.0;          % [g] Liquid mass in tank 1 at time t0
  m20 = 0.0;          % [g] Liquid mass in tank 2 at time t0
  m30 = 0.0;          % [g] Liquid mass in tank 3 at time t0
  m40 = 0.0;          % [g] Liquid mass in tank 4 at time t0

  F1 = 300;           % [cm3/s] Flow rate from pump 1
  F2 = 300;           % [cm3/s] Flow rate from pump 2

  x0 = [m10; m20; m30; m40];
  u = [F1; F2]
% ---------------------------------------------------------------
```

Simulate the system

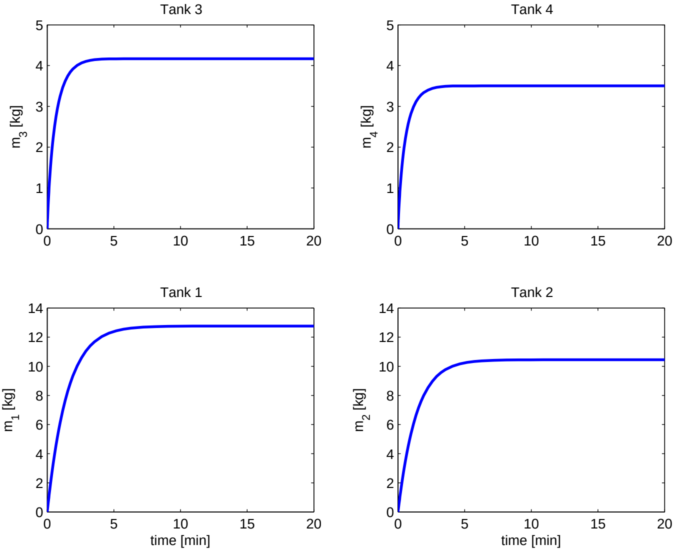
```
% ---------------------------------------------------------------
% Compute the solution / Simulate
% ---------------------------------------------------------------
% Solve the system of differential equations
[T,X] = ode15s(@FourTankSystem,[t0 tf],x0,[],u,p);
```

Computation of additional variables

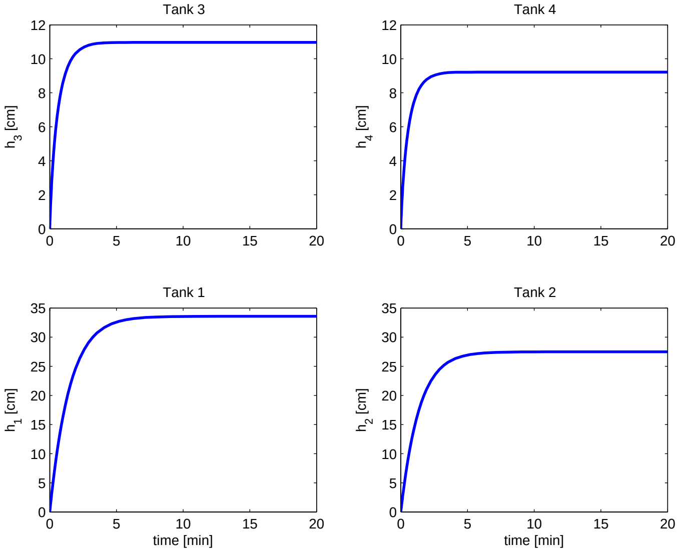
Compute additional variables for plotting

```
% ----------------------------------------------------------------
% help variables
[nT,nX] = size(X);
a = p(1:4,1)';
A = p(5:8,1)';

% Compute the measured variables
H = zeros(nT,nX);
for i=1:nT
    H(i,:) = X(i,:)./(rho*A);
end

% Compute the flows out of each tank
Qout = zeros(nT,nX);
for i=1:nT
    Qout(i,:) = a.*sqrt(2*g*H(i,:));
end
% ----------------------------------------------------------------
```

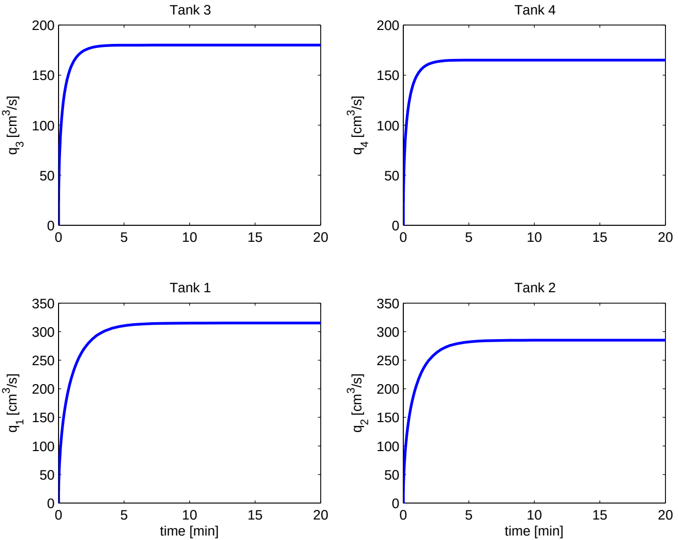
Masses in the Tanks



Levels in the Tanks



Outflow Rates



Discrete-Time Generic Input-Output Model

$$x_{k+1} = F(x_k, u_k) \qquad \text{Discrete-time process model}$$
$$y_k = g(x_k) \qquad \text{Sensor function}$$
$$z_k = h(x_k) \qquad \text{Output function}$$

Zero-order-hold for MVs

$$u(t) = u_k \qquad t_k \leq t < t_{k+1}$$

Continuous-time process model to discrete-time process model

$$F(x_k, u_k) = x_k + \int_{t_k}^{t_{k+1}} f(x(t), u_k) dt$$

Difference Equation

The difference equation

$$x_{k+1} = F(x_k, u_k)$$

with

$$F(x_k, u_k) = x_k + \int_{t_k}^{t_{k+1}} f(x(t), u_k) dt$$

can be computed by numerical solution of

$$\frac{dx}{dt}(t) = f(x(t), u_k) \qquad x(t_k) = x_k \qquad t_k \leq t < t_{k+1}$$

such that

$$x_{k+1} = x(t_{k+1})$$

Discrete-Time Simulation using Matlab

The discrete-time model

$$
\begin{aligned}
x_{k+1} &= F(x_k, u_k) \qquad F(x_k, u_k) = x_k + \int_{t_k}^{t_{k+1}} f(x(t), u_k)dt \\
y_k &= g(x_k) \\
z_k &= h(x_k)
\end{aligned}
$$

may be simulated in Matlab using

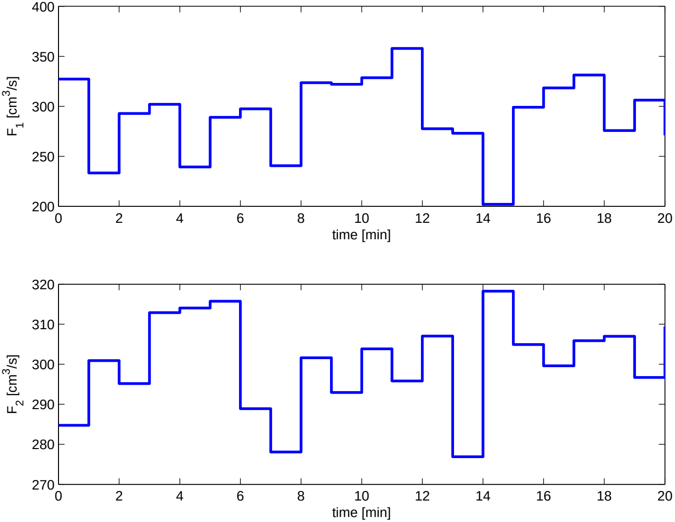
```
for i=0:N
  k=i+1;
  y(:,k) = g(x(:,k));
  z(:,k) = h(x(:,k));
  [Tk,Xk] = ode15s(@f,[t(k) t(k+1)],x(:,k),odeOptions,u(:,k));
  x(:,k+1) = Xk(end,:)';
  T = [T; Tk];
  X = [X; Xk];
end
```

Example - 4-Tank System

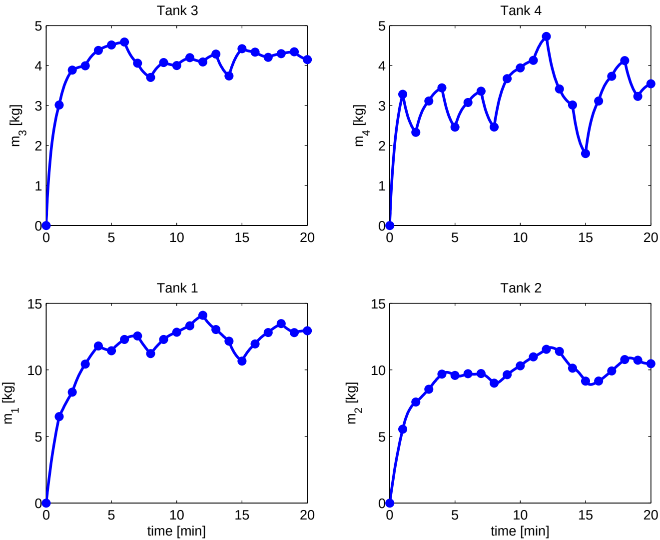
$$x_{k+1} = F(x_k, u_k)$$
$$y_k = g(x_k)$$
$$z_k = h(x_k)$$

```
X = zeros(0,nx);
T = zeros(0,1);

x(:,1) = x0;
for k = 1:N-1
    y(:,k) = FourTankSystemSensor(x(:,k),p); % Sensor function
    z(:,k) = FourTankSystemOutput(x(:,k),p); % Output function

    [Tk,Xk] = ode15s(@FourTankSystem,[t(k) t(k+1)],x(:,k),[],u(:,k),p);
    x(:,k+1) = Xk(end,:)';

    T = [T; Tk];
    X = [X; Xk];
end
k = N;
    y(:,k) = FourTankSystemSensor(x(:,k),p);
    z(:,k) = FourTankSystemOutput(x(:,k),p);
```

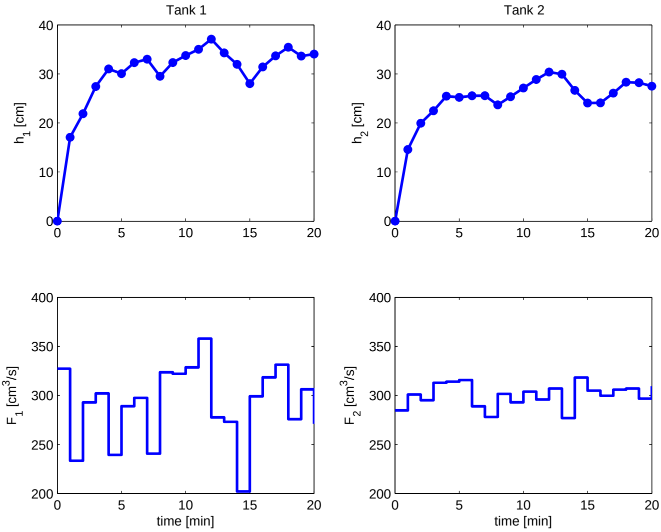
Flow Rate Scenario



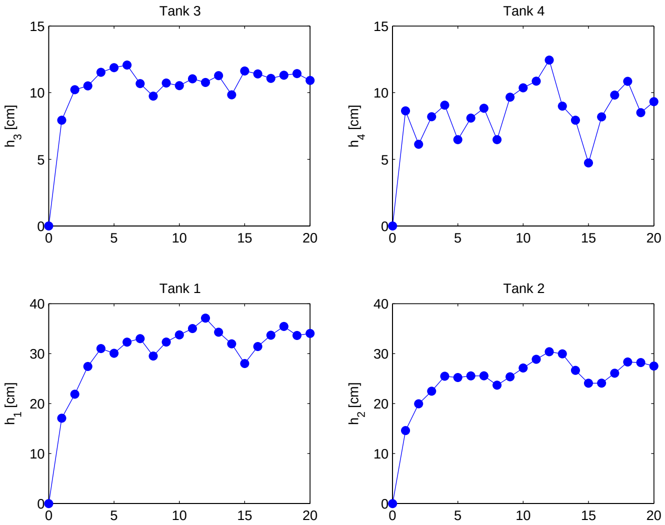
Quadruple Tank Process - States



Quadruple Tank Process - Inputs and Outputs



Quadruple Tank Process - Sensors



Stochastic Simulation

$$\begin{aligned} \boldsymbol{x}_{k+1} &= F(\boldsymbol{x}_k, u_k, \boldsymbol{w}_k) && \text{Process model} \\ \boldsymbol{y}_k &= g(\boldsymbol{x}_k) + \boldsymbol{v}_k && \text{Sensor function} \\ \boldsymbol{z}_k &= h(\boldsymbol{x}_k) && \text{Output function} \end{aligned}$$

$$\begin{aligned} &\boldsymbol{x}_0 \sim N(\bar{x}_0, P_0) && \text{Unknown initial state} \\ &\boldsymbol{w}_k \sim N_{iid}(0, Q) && \text{Process noise} \\ &\boldsymbol{v}_k \sim N_{iid}(0, R) && \text{Measurement (sensor) noise} \end{aligned}$$

A Stochastic Realization

The stochastic variables

$$\boldsymbol{w}_k \sim N_{iid}(0, Q) \qquad Q = LL'$$
$$\boldsymbol{e}_k \sim N_{iid}(0, I)$$

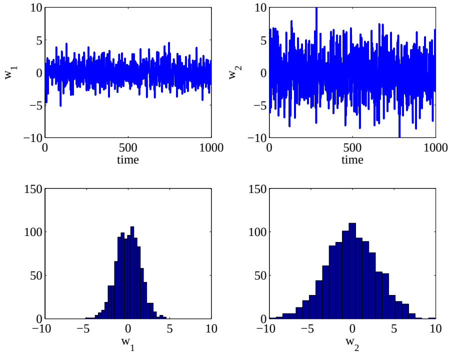
are related by

$$\boldsymbol{w}_k = L\boldsymbol{e}_k$$

As a normal distribution is completely characterized by its means and covariance, this relation can be proved by

$$E\{\boldsymbol{w}_k\} = E\{L\boldsymbol{e}_k\} = LE\{\boldsymbol{e}_k\} = 0$$
$$V\{\boldsymbol{w}_k\} = \langle \boldsymbol{w}_k, \boldsymbol{w}_k \rangle = \langle L\boldsymbol{e}_k, L\boldsymbol{e}_k \rangle = L\underbrace{\langle \boldsymbol{e}_k, \boldsymbol{e}_k \rangle}_{=I} L' = LL' = Q$$

Stochastic Realization in Matlab



$$w_k \sim N_{iid}(0, Q)$$

$$Q = LL'$$

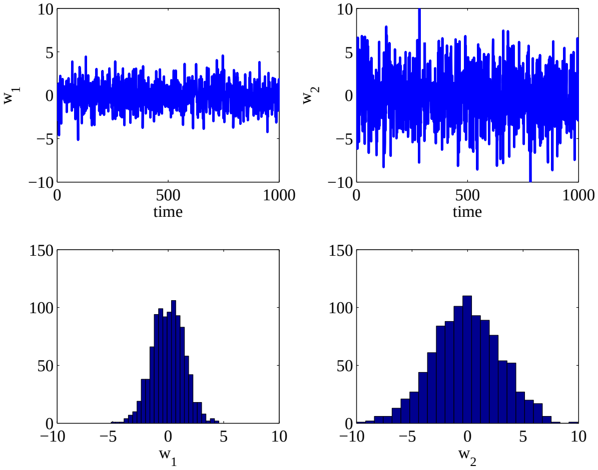
$$e_k \sim N_{iid}(0, I)$$

$$w_k = Le_k$$

```
Q = [2 1; 1 10];
L = chol(Q)';

MySeed = 100;                % You can use any number
randn('state', MySeed);      % Makes the random sequence reproducable
w = L*randn(2,1000);         % Generation of the random sequence
```

Stochastic Process Noise

$$w_k \sim N_{iid}(0, Q)$$



Process noise

$$\begin{bmatrix} \boldsymbol{F}_1 \\ \boldsymbol{F}_2 \end{bmatrix} = \begin{bmatrix} F_{1s} \\ F_{2s} \end{bmatrix} + \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix}$$

$$\begin{bmatrix} F_{1s} \\ F_{2s} \end{bmatrix} = \begin{bmatrix} 300 \\ 300 \end{bmatrix} \qquad \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} \sim N_{iid}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 30^2 & 0 \\ 0 & 10^2 \end{bmatrix} \right)$$

Measurement noise

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3 \\ \boldsymbol{y}_4 \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix} + \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \boldsymbol{v}_3 \\ \boldsymbol{v}_4 \end{bmatrix} \qquad \boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \boldsymbol{v}_3 \\ \boldsymbol{v}_4 \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1^2 & 0 & 0 & 0 \\ 0 & 1^2 & 0 & 0 \\ 0 & 0 & 1^2 & 0 \\ 0 & 0 & 0 & 1^2 \end{bmatrix} \right)$$

Outputs

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{z}_1 \\ \boldsymbol{z}_2 \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}$$

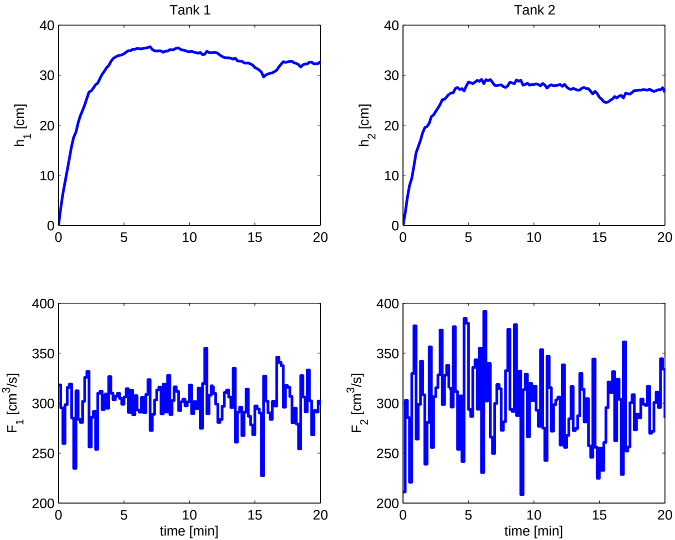
Stochastic Simulation - Definition of Simulation Scenario

```
t0 =    0.0;        % [s] Initial time
tf = 20*60;         % [s] Final time
Ts = 10;            % [s] Sample Time
t = [t0:Ts:tf]';    % [s] Sample instants
N = length(t);

m10 = 0;           % [g] Liquid mass in tank 1 at time t0
m20 = 0;           % [g] Liquid mass in tank 2 at time t0
m30 = 0;           % [g] Liquid mass in tank 3 at time t0
m40 = 0;           % [g] Liquid mass in tank 4 at time t0

F1 = 300;           % [cm3/s] Flow rate from pump 1
F2 = 300;           % [cm3/s] Flow rate from pump 2

x0 = [m10; m20; m30; m40];
u = [repmat(F1,1,N); repmat(F2,1,N)];

% Process Noise
Q = [20^2 0;0 40^2];
Lq = chol(Q,'lower');
w = Lq*randn(2,N);

% Measurement Noise
R = eye(4);
Lr = chol(R,'lower');
v = Lr*randn(4,N);
```

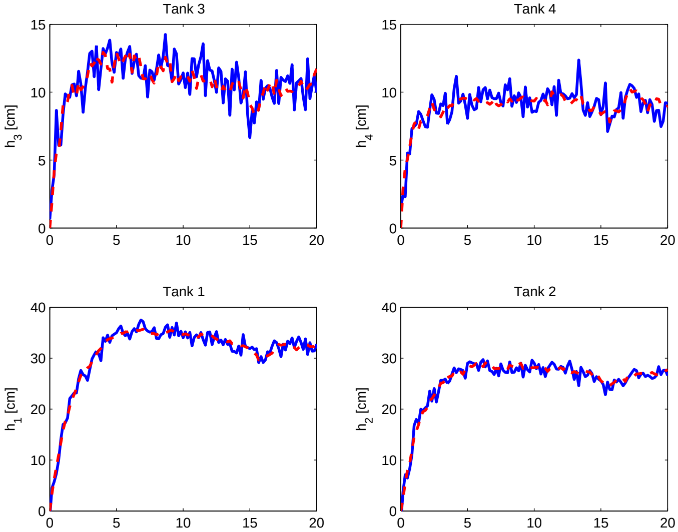
Stochastic Simulation - Matlab Script

```
nx = 4; nu = 2; ny = 4; nz = 2;
x = zeros(nx,N);
y = zeros(ny,N);
z = zeros(nz,N);

X = zeros(0,nx);
T = zeros(0,1);

x(:,1) = x0;
for k = 1:N-1
    y(:,k) = FourTankSystemSensor(x(:,k),p)+v(:,k); % Sensor function
    z(:,k) = FourTankSystemOutput(x(:,k),p);        % Output function

    [Tk,Xk] = ode15s(@FourTankSystem,[t(k) t(k+1)],x(:,k),[],...
                        u(:,k)+w(:,k),p);
    x(:,k+1) = Xk(end,:)';

    T = [T; Tk];
    X = [X; Xk];
end
k = N;
    y(:,k) = FourTankSystemSensor(x(:,k),p)+v(:,k); % Sensor function
    z(:,k) = FourTankSystemOutput(x(:,k),p);        % Output function
```

Stochastic Process Simulation - Input-Output



Stochastic Process Simulation - Measurements



Steady-State (Equilibrium Point)

$$\dot{x}(t) = f(x(t), u(t)) \qquad x(t_0) = x_0$$

Steady state (equilibrium point) (x_s, u_s). Given $u(t) = u_s$ solve

$$0 = f(x_s, u_s)$$

`fsolve` in Matlab

```
xs0 = 5000*ones(4,1);    % Initial guess of steady state
us  = [300; 300];        % Steady-state inputs
xs = fsolve(@FourTankSystemWrap,xs0,[],us,p)
```

with

```
function xdot = FourTankSystemWrap(x,u,p)
xdot = FourTankSystem(0,x,u,p);
```

Lineariztion of Continuous-Time Model

$$\dot{x}(t) = f(x(t), u(t)) \qquad x(t_0) = x_0$$

Steady state (equilibrium point) (x_s, u_s). Given $u(t) = u_s$ solve

$$0 = f(x_s, u_s)$$

Taylor expansion

$$\begin{aligned} f(x(t), u(t)) &\approx \overbrace{f(x_s, u_s)}^{=0} + \overbrace{\left(\frac{\partial f}{\partial x}(x_s, u_s)\right)}^{=A} \overbrace{(x(t) - x_s)}^{=X(t)} + \overbrace{\left(\frac{\partial f}{\partial u}(x_s, u_s)\right)}^{=B} \overbrace{(u(t) - u_s)}^{=U(t)} \\ &= AX(t) + BU(t) \end{aligned}$$

$$\dot{X}(t) = \frac{d}{dt}\left(x(t) - x_s\right) = \frac{dx(t)}{dt} - \frac{dx_s}{dt} = \frac{dx(t)}{dt} - 0 = \dot{x}(t)$$

Then

$$\dot{X}(t) = AX(t) + BU(t) \qquad X(t_0) = X_0 = x_0 - x_s$$

$$\dot{x}(t) = f(x(t), u(t)) \qquad x(t_0) = x_0$$
$$y(t) = g(x(t))$$
$$z(t) = h(x(t))$$

Steady-state (x_s, u_s, y_s, z_s)

$$0 = f(x_s, u_s) \qquad y_s = g(x_s) \qquad z_s = h(x_s)$$

Deviation variables

$$X(t) = x(t) - x_s \qquad X_0 = x_0 - x_s$$
$$U(t) = u(t) - u_s \qquad Y(t) = y(t) - y_s \qquad Z(t) = z(t) - z_s$$

Linear system

$$\dot{X}(t) = AX(t) + BU(t) \quad X(t_0) = X_0$$
$$Y(t) = CX(t)$$
$$Z(t) = C_z X(t)$$

$$A = \frac{\partial f}{\partial x}(x_s, u_s) \quad B = \frac{\partial f}{\partial u}(x_s, u_s)$$
$$C = \frac{\partial g}{\partial x}(x_s, u_s) \quad C_z = \frac{\partial h}{\partial x}(x_s, u_s)$$

$$\dot{x}(t) = f(x(t), u(t)) \qquad x(t_0) = x_0$$
$$y(t) = g(x(t), u(t))$$

Steady-state (x_s, u_s, y_s)

$$0 = f(x_s, u_s) \qquad y_s = g(x_s, u_s)$$

Deviation variables

$$X(t) = x(t) - x_s \qquad X_0 = x_0 - x_s$$
$$U(t) = u(t) - u_s$$
$$Y(t) = y(t) - y_s$$

Linear system

$$\dot{X}(t) = AX(t) + BU(t) \quad X(t_0) = X_0$$
$$Y(t) = CX(t) + DU(t)$$

$$A = \frac{\partial f}{\partial x}(x_s, u_s) \quad B = \frac{\partial f}{\partial u}(x_s, u_s)$$
$$C = \frac{\partial g}{\partial x}(x_s, u_s) \quad D = \frac{\partial g}{\partial u}(x_s, u_s)$$

Example - Linearization of 4-Tank System

$$\dot{X}(t) = AX(t) + BU(t) \qquad X(t_0) = X_0$$
$$Y(t) = CX(t)$$
$$Z(t) = C_z X(t)$$

$$X(t) = x(t) - x_s \qquad U(t) = u(t) - u_s$$
$$Y(t) = y(t) - y_s \qquad Z(t) = z(t) - z_s$$

$$u_s = \begin{bmatrix} F_{1s} \\ F_{2s} \end{bmatrix} = \begin{bmatrix} 300\ \mathrm{cm}^3/\mathrm{s} \\ 300\ \mathrm{cm}^3/\mathrm{s} \end{bmatrix} \qquad z_s = \begin{bmatrix} h_{1s} \\ h_{2s} \end{bmatrix} = \begin{bmatrix} 33.6\ \mathrm{cm} \\ 27.5\ \mathrm{cm} \end{bmatrix}$$

$$x_s = \begin{bmatrix} m_{1s} \\ m_{2s} \\ m_{3s} \\ m_{4s} \end{bmatrix} = \begin{bmatrix} 12765\ \mathrm{g} \\ 10449\ \mathrm{g} \\ 4168\ \mathrm{g} \\ 3502\ \mathrm{g} \end{bmatrix} \qquad y_s = \begin{bmatrix} h_{1s} \\ h_{2s} \\ h_{3s} \\ h_{4s} \end{bmatrix} = \begin{bmatrix} 33.6\ \mathrm{cm} \\ 27.5\ \mathrm{cm} \\ 11.0\ \mathrm{cm} \\ 9.2\ \mathrm{cm} \end{bmatrix}$$

Example - Linearization of 4-Tank System

$$\begin{aligned}
\dot{X}(t) &= AX(t) + BU(t) \qquad X(t_0) = X_0 \\
Y(t) &= CX(t) \\
Z(t) &= C_z X(t)
\end{aligned}$$

with the matrices

$$A = \begin{bmatrix} -\frac{1}{T_1} & 0 & \frac{1}{T_3} & 0 \\ 0 & -\frac{1}{T_2} & 0 & \frac{1}{T_4} \\ 0 & 0 & -\frac{1}{T_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{T_4} \end{bmatrix} \qquad B = \begin{bmatrix} \rho\gamma_1 & 0 \\ 0 & \rho\gamma_2 \\ 0 & \rho(1-\gamma_2) \\ \rho(1-\gamma_1) & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} \frac{1}{\rho A_1} & 0 & 0 & 0 \\ 0 & \frac{1}{\rho A_2} & 0 & 0 \\ 0 & 0 & \frac{1}{\rho A_3} & 0 \\ 0 & 0 & 0 & \frac{1}{\rho A_4} \end{bmatrix} \qquad C_z = \begin{bmatrix} \frac{1}{\rho A_1} & 0 & 0 & 0 \\ 0 & \frac{1}{\rho A_2} & 0 & 0 \end{bmatrix}$$

and time constants defined by

$$T_i = \frac{A_i\sqrt{2gh_{i,s}}}{a_i g} = \frac{A_i}{a_i}\sqrt{\frac{2h_{i,s}}{g}} \qquad i = \{1, 2, 3, 4\}$$

Example - Linearization 4-Tank System

```
% -------------------------------------------------------------
% Parameters
% -------------------------------------------------------------
ap = [a1; a2; a3; a4];    % [cm2] Pipe cross sectional areas
At = [A1; A2; A3; A4];    % [cm2] Tank cross sectional areas
gam = [gamma1; gamma2];   % [-] Valve constants
g = 981;                  %[cm/s2] The acceleration of gravity
rho = 1.00;               %[g/cm3] Density of water

p = [ap; At; gamma; g; rho];

% -------------------------------------------------------------
% Steady State
% -------------------------------------------------------------
us   = [300; 300]                 % [cm3/s] Flow rates
xs0 = [5000; 5000; 5000; 5000]   % [g] Initial guess on xs

xs = fsolve(@FourTankSystemWrap,xs0,[],us,p)
ys = FourTankSystemSensor(xs,p)
zs = FourTankSystemOutput(xs,p)

% -------------------------------------------------------------
% Linearization
% -------------------------------------------------------------
hs = ys;
T = (At./ap).*sqrt(2*hs/g)

A=[-1/T(1) 0 1/T(3) 0;0 -1/T(2) 0 1/T(4);0 0 -1/T(3) 0;0 0 0 -1/T(4)]
B=[rho*gam(1) 0;0 rho*gam(2); 0 rho*(1-gam(2)); rho*(1-gam(1)) 0]
C=diag(1./(rho*At))
Cz=C(1:2,:)
```

Example - Linearization 4-Tank System

```
A =
    -0.012338          0   0.021592          0
            0  -0.013637          0   0.023555
            0          0  -0.021592          0
            0          0          0  -0.023555

B =
         0.45          0
            0        0.4
            0        0.6
         0.55          0

C =
    0.0026307          0          0          0
            0  0.0026307          0          0
            0          0  0.0026307          0
            0          0          0  0.0026307

Cz =
    0.0026307          0          0          0
            0  0.0026307          0          0
```

Continuous-Time Transfer Function

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad x(0) = x_0$$
$$y(t) = Cx(t) + Du(t)$$

Define the LaPlace transform

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^\infty e^{-st} f(t) dt$$

such that

$$\mathcal{L}\{x(t)\} = X(s) \qquad \mathcal{L}\{u(t)\} = U(s) \qquad \mathcal{L}\{y(t)\} = Y(s)$$
$$\mathcal{L}\{\dot{x}(t)\} = sX(s) - x_0$$

Then

$$sX(s) - x_0 = AX(s) + BU(s) \qquad \Leftrightarrow$$
$$X(s) = (sI - A)^{-1} x_0 + (sI - A)^{-1} BU(s)$$

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad x(0) = x_0$$
$$y(t) = Cx(t) + Du(t)$$

The LaPlace transforms of this system are

$$X(s) = (sI - A)^{-1}x_0 + (sI - A)^{-1}BU(s)$$

$$\begin{aligned} Y(s) &= CX(s) + DU(s) \\ &= C(sI - A)^{-1}x_0 + \overbrace{\left[C(sI - A)^{-1}B + D\right]}^{=G(s)}U(s) \\ &= C(sI - A)^{-1}x_0 + G(s)U(s) \end{aligned}$$

Let $x_0 = 0$. Then

$$Y(s) = G(s)U(s) \qquad G(s) = C(sI - A)^{-1}B + D$$

$G(s)$ is the transfer function of the system

Continuous-Time Transfer Function

The continuous-time linear time-invariant system

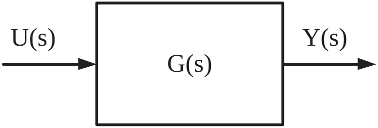
$$\dot{x}(t) = Ax(t) + Bu(t) \qquad x(0) = 0$$
$$y(t) = Cx(t) + Du(t)$$

can be represented as the input-output function in the LaPlace domain

$$Y(s) = G(s)U(s)$$

with the transfer function

$$G(s) = C(sI - A)^{-1}B + D$$



Poles-Zero Representation - SISO system

$$
\begin{aligned}
G(s) &= C(sI-A)^{-1}B + D \\
&= K_{zp}\frac{(s-z_1)(s-z_2)\dots(s-z_{n_z})}{(s-p_1)(s-p_2)\dots(s-p_n)} = K_{zp}\frac{\prod(s-z_i)}{\prod(s-p_i)}
\end{aligned}
$$

- The poles are computed as the eigenvalues of A: `p = eig(A)`
- The zeros are computed as the generalized eigenvalues of

$$
\overbrace{\begin{bmatrix} A & B \\ C & D \end{bmatrix}}^{=M} \begin{bmatrix} x \\ u \end{bmatrix} = z \overbrace{\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}}^{=N} \begin{bmatrix} x \\ u \end{bmatrix}
$$

`z = eig(M,N)` $(|z| < \infty)$

- The constant K_{zp} is computed by

$$
K_{zp} = G(s)\frac{\prod(s-p_i)}{\prod(s-z_i)} \qquad s \neq p_i, z_i
$$

Example

$$G(s) = 2\frac{(s-3)}{(s-0)(s+4)(s+(2+3i))(s+(2-3i))}$$

Constant: $K_{zp} = 2$
Zero: $z = 3$
Poles: $p_1 = 0$, $p_2 = -4$, $p_3 = -2 \pm 3i$

This is equivalent with

$$G(s) = 2\frac{(s-3)}{s(s+4)(s^2+4s+13)} = K\frac{(\beta s+1)}{s(\tau_1 s+1)(\tau^2 s^2+2\zeta\tau s+1)}$$

$$\beta = -1/3$$

$$\tau_1 = 1/4$$

$$\tau = \frac{1}{\sqrt{13}} \quad \zeta = \frac{4/13}{2\sqrt{13}} = \frac{2}{13\sqrt{13}}$$

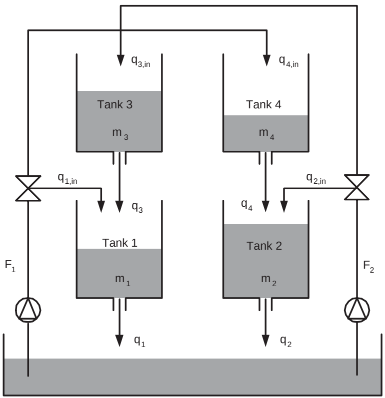
$$K = 2(-3)/(4\cdot 13) = -3/26$$

Some Elementary Continuous-Time Transfer Functions

$$Y(s) = G(s)U(s)$$

$G(s) = \frac{1}{s}$	Integrator
$G(s) = \frac{K}{\tau s + 1}$	First order system
$G(s) = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)}$	Second order system - damped
$G(s) = \frac{K}{(\tau s + 1)^n}$	n'th order system
$G(s) = \frac{K}{\tau^2 s^2 + 2\tau\zeta s + 1}$	Second order system - underdamped
$G(s) = \frac{K(\beta s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)}$	Second order system with zero

4-Tank System - Model



$$T_1 = \frac{A_1}{a_1}\sqrt{\frac{2h_{1,s}}{g}} = 81.0\,\mathrm{s} = 1.35\,\mathrm{min}$$

$$T_2 = \frac{A_2}{a_2}\sqrt{\frac{2h_{2,s}}{g}} = 73.3\,\mathrm{s} = 1.22\,\mathrm{min}$$

$$T_3 = \frac{A_3}{a_3}\sqrt{\frac{2h_{3,s}}{g}} = 46.3\,\mathrm{s} = 0.77\,\mathrm{min}$$

$$T_4 = \frac{A_4}{a_4}\sqrt{\frac{2h_{4,s}}{g}} = 42.5\,\mathrm{s} = 0.71\,\mathrm{min}$$

$$K = G(0) = -C_z A^{-1} B$$

$$K_{11} = 0.0959\,\mathrm{s/cm^2}$$
$$K_{21} = 0.1061\,\mathrm{s/cm^2}$$
$$K_{12} = 0.1279\,\mathrm{s/cm^2}$$
$$K_{22} = 0.0772\,\mathrm{s/cm^2}$$

$$\begin{bmatrix} Z_1(s) \\ Z_2(s) \end{bmatrix} = \begin{bmatrix} \frac{K_{11}}{T_1 s+1} & \frac{K_{12}}{(T_3 s+1)(T_1 s+1)} \\ \frac{K_{21}}{(T_4 s+1)(T_2 s+1)} & \frac{K_{22}}{T_2 s+1} \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix}$$

Linear System of First-Order Differential Equations

The linear system of first order differential equations

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad x(t_0) = x_0$$

has the solution

$$x(t) = e^{A(t-t_0)}x_0 + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau$$

You can convince yourself about this solution by reviewing the solution of

$$\frac{dp(t)}{dt} + ap(t) = q(t)$$

This equation is known from calculus.

Discretization

Discrete time

$$t_k = t_0 + kT_s \qquad k = 0, 1, 2 \ldots$$

States at discrete times

$$x_k = x(t_k)$$

Zero-order-hold of the inputs

$$u(t) = u_k \qquad t_k \leq t < t_{k+1}$$

Consider the linear differential equation

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad x(t_k) = x_k$$

Then

$$\begin{aligned} x_{k+1} = x(t_{k+1}) &= e^{A(t_{k+1}-t_k)} x_k + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} Bu(\tau) d\tau \\ &= \left[e^{AT_s}\right] x_k + \left[\int_0^{T_s} e^{A\eta} B d\eta\right] u_k \end{aligned}$$

Discrete-Time Linear Model

The continuous linear time-invariant model

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \qquad x(t_0) = x_0 \\ y(t) &= Cx(t) + Du(t) \end{aligned}$$

is equivalent with the discrete linear time-invariant model

$$\begin{aligned} x_{k+1} &= A_d x_k + B_d u_k \\ y_k &= C_d x_k + D_d u_k \end{aligned}$$

with

$$\begin{aligned} A_d &= e^{AT_s} & B_d &= \int_0^{T_s} e^{A\tau} B d\tau \\ C_d &= C & D_d &= D \end{aligned}$$

(A_d, B_d) can be computed by

$$\begin{bmatrix} A_d & B_d \\ 0 & I \end{bmatrix} = \exp\left(\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} T_s \right)$$

ZOH Discretization of Linear System

$$\begin{bmatrix} A_d & B_d \\ 0 & I \end{bmatrix} = \exp\left(\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} T_s\right)$$

Matlab implementation

```
function [Abar,Bbar]=c2dzoh(A,B,Ts)

[nx,nu]=size(B);
M = [A B; zeros(nu,nx) zeros(nu,nu)];
Phi = expm(M*Ts);
Abar = Phi(1:nx,1:nx);
Bbar = Phi(1:nx,nx+1:nx+nu);
```

```
A =
    -0.012338            0     0.021592            0
            0    -0.013637            0     0.023555
            0            0    -0.021592            0
            0            0            0    -0.023555

B =
         0.45            0
            0          0.4
            0          0.6
         0.55            0
```

$T_s = 10.$ `[Ad,Bd]=c2dzoh(A,B,10)`

```
Ad =
      0.8839         0    0.1823         0
           0    0.8725         0    0.1957
           0         0    0.8058         0
           0         0         0    0.7901

Bd =
      4.2335    0.5791
      0.5729    3.7392
           0    5.3965
      4.9002         0
```

Discrete-Time Transfer Function

$$
\begin{aligned}
x_{k+1} &= Ax_k + Bu_k \\
y_k &= Cx_k + Du_k
\end{aligned}
$$

Z-transform (unilateral)

$$F(z) = \mathcal{Z}\{f_k\} = \sum_{k=0}^{\infty} z^{-k} f_k$$

$$X(z) = \mathcal{Z}\{x_k\} \quad U(z) = \mathcal{Z}\{u_k\} \quad Y(z) = \mathcal{Z}\{y_k\}$$

$$\mathcal{Z}\{x_{k+1}\} = zX(z) - zx_0$$

$$
\begin{aligned}
&zX(z) - zx_0 = AX(z) + BU(z) \qquad \Leftrightarrow \\
&X(z) = (zI - A)^{-1} z x_0 + (zI - A)^{-1} BU(z)
\end{aligned}
$$

$$
\begin{aligned}
x_{k+1} &= Ax_k + Bu_k \\
y_k &= Cx_k + Du_k
\end{aligned}
$$

$$
X(z) = (zI - A)^{-1} z x_0 + (zI - A)^{-1} B U(z)
$$

$$
\begin{aligned}
Y(z) &= CX(z) + DU(z) \\
&= C(zI - A)^{-1} z x_0 + \overbrace{\left[C(zI - A)^{-1}B + D\right]}^{=G(z)} U(z) \\
&= C(zI - A)^{-1} z x_0 + G(z)U(z)
\end{aligned}
$$

Let $x_0 = 0$. Then

$$
Y(z) = G(z)U(z) \qquad G(z) = C(zI - A)^{-1}B + D
$$

Discrete-Time Transfer Function

The discrete-time linear time-invariant system

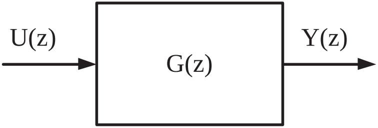
$$x_{k+1} = Ax_k + Bu_k \qquad x_0 = 0$$
$$y_k = Cx_k + Du_k$$

can be represented as the input-output function in the Z-domain

$$Y(z) = G(z)U(z)$$

with the transfer function

$$G(z) = C(zI - A)^{-1}B + D$$



Poles-Zero Representation - SISO system

$$
\begin{aligned}
G(z) &= C(zI - A)^{-1}B + D \\
&= K_{zp}\frac{(z-z_1)(z-z_2)\dots(z-z_{n_z})}{(z-p_1)(z-p_2)\dots(z-p_n)} = K_{zp}\frac{\prod(z-z_i)}{\prod(z-p_i)}
\end{aligned}
$$

- The poles are computed as the eigenvalues of A: `p = eig(A)`
- The zeros are computed as the generalized eigenvalues of

$$
\overbrace{\begin{bmatrix} A & B \\ C & D \end{bmatrix}}^{=M} \begin{bmatrix} x \\ u \end{bmatrix} = z \overbrace{\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}}^{=N} \begin{bmatrix} x \\ u \end{bmatrix}
$$

`z = eig(M,N)` $(|z| < \infty)$

- The constant K_{zp} is computed by

$$
K_{zp} = G(z)\frac{\prod(z-p_i)}{\prod(z-z_i)} \qquad z \neq p_i, z_i
$$

Stochastic Systems

- Physical systems evolve in continuous time
- Some of the phenomena occurring are stochastic
- Question: How to convert a continuous-time stochastic model to a discrete time stochastic model

In particular, we cannot extend a system of ODEs

$$\frac{dx}{dt}(t) = f(x(t), u(t)) \qquad x(t_0) = x_0$$

to a system of stochastic differential equations in the following form

$$\frac{d\boldsymbol{x}}{dt}(t) = f(\boldsymbol{x}(t), u(t)) + \sigma(\boldsymbol{x}(t), u(t))\boldsymbol{w}(t) \qquad \text{Not well-defined}$$

as the last expression is not well-defined mathematically

Stochastic Systems

Hence, the following is not well-defined

$$\frac{d\boldsymbol{x}}{dt}(t) = f(\boldsymbol{x}(t), u(t), \boldsymbol{w}(t)) \qquad x(t_0) = x_0 \qquad \text{Not well-defined}$$

for continuous stochastic disturbances $\boldsymbol{w}(t)$.

We can assume that

$$\boldsymbol{w}(t) = \boldsymbol{w}_k \qquad t_k \leq t < t_{k+1} \qquad t_{k+1} = t_k + T_s$$

Then a specific realization $\{w_k\}$ leads to

$$\frac{dx}{dt}(t) = f(x(t), u(t), w_k) \qquad t_k \leq t < t_{k+1} \qquad x(t_k) = x_k$$

from which $x(t)$ is computed. The problem with this construct is that the stochastic properties depends on T_s and does not have the correct properties in the limit $T_s \to 0$.

Stochastic Differential Equation - with Ito interpretation

The approximation of deterministic differential equation

$$x(t+\delta t) - x(t) = f(x(t), u(t))\delta t + o(\delta t)$$

is extended to the stochastic case

$$\boldsymbol{x}(t+\delta t) - \boldsymbol{x}(t) = f(\boldsymbol{x}(t), u(t))\delta t + \sigma(\boldsymbol{x}(t), u(t))\left[\boldsymbol{w}(t+\delta t) - \boldsymbol{w}(t)\right] + o(\delta t)$$

using increments $\Delta\boldsymbol{w}(t)$ that are independent and normally distributed

$$\Delta\boldsymbol{w}(t) = \left[\boldsymbol{w}(t+\delta t) - \boldsymbol{w}(t)\right] \sim N_{iid}(0, I\delta t) = \sqrt{\delta t} N_{iid}(0, I)$$

In the limit $\delta t \to 0$

$$d\boldsymbol{x}(t) = f(\boldsymbol{x}(t), u(t))dt + \sigma(\boldsymbol{x}(t), u(t))d\boldsymbol{w}(t)$$

Stochastic Differential Equation - with Ito interpretation

$$dx(t) = f(x(t), u(t))dt + \sigma(x(t), u(t))dw(t)$$

with $\{w(t)\}$ being a standard Wiener Process (Brownian motion)

1. $w(t)$ is normally distributed
2. $w(t)$ is independent of $w(s)$ for all $s \neq t$
3. $E\{w(t)\} = 0$
4. $E\{dw(t)dw(t)'\} = Idt$

The integral equation corresponding to the Stochastic Differential Equation (SDE) is

$$x(t) = x(t_0) + \int_{t_0}^{t} f(x(s), u(s))ds + \int_{t_0}^{t} \sigma(x(s), u(s))dw(s)$$

with the integrals interpreted as Ito integrals (forward Euler integrals).

Stochastic Systems - Discrete Time

$$
\begin{aligned}
\boldsymbol{x}_{k+1} &= F(\boldsymbol{x}_k, u_k, \boldsymbol{w}_k) \quad && \boldsymbol{x}_0 \sim N(\bar{x}_0, P_0) \quad \boldsymbol{w}_k \sim N(0, Q) \\
\boldsymbol{y}_k &= g(\boldsymbol{x}_k) + \boldsymbol{v}_k && \boldsymbol{v}_k \sim N(0, R) \\
\boldsymbol{z}_k &= h(\boldsymbol{x}_k)
\end{aligned}
$$

Stochastic Systems - Discrete Time

$$\begin{aligned} \boldsymbol{x}_{k+1} &= F(\boldsymbol{x}_k, u_k, \boldsymbol{w}_k) & \boldsymbol{x}_0 &\sim N(\bar{x}_0, P_0) & \boldsymbol{w}_k &\sim N(0, Q) \\ \boldsymbol{y}_k &= g(\boldsymbol{x}_k) + \boldsymbol{v}_k & \boldsymbol{v}_k &\sim N(0, R) \\ \boldsymbol{z}_k &= h(\boldsymbol{x}_k) \end{aligned}$$

Steady-state (equilibrium point): Given $u_s = 0$ and $w_s = 0$, compute

$$x_s = F(x_s, u_s, 0) \qquad R(x_s) = x_s - F(x_s, u_s, 0) = 0$$

and

$$\begin{aligned} y_s &= g(x_s) \\ z_s &= h(x_s) \end{aligned}$$

Deviation variables

$$\begin{aligned} \boldsymbol{X}(t) &= \boldsymbol{x}(t) - x_s & U(t) &= u(t) - u_s \\ \boldsymbol{Y}(t) &= \boldsymbol{y}(t) - y_s & \boldsymbol{Z}(t) &= \boldsymbol{z}(t) - z_s \end{aligned}$$

Stochastic Systems - Discrete Time

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= F(\boldsymbol{x}_k, u_k, \boldsymbol{w}_k) & \boldsymbol{x}_0 &\sim N(\bar{x}_0, P_0) \quad \boldsymbol{w}_k \sim N(0, Q)\\
\boldsymbol{y}_k &= g(\boldsymbol{x}_k) + \boldsymbol{v}_k & \boldsymbol{v}_k &\sim N(0, R)\\
\boldsymbol{z}_k &= h(\boldsymbol{x}_k)
\end{aligned}$$

can be approximated by the linear discrete-time stochastic system

$$\begin{aligned}
\boldsymbol{X}_{k+1} &= A\boldsymbol{X}_k + BU_k + G\boldsymbol{w}_k & \boldsymbol{X}_0 &\sim N(\bar{X}_0, P_0),\ \boldsymbol{w}_k \sim N(0, Q)\\
\boldsymbol{Y}_k &= C\boldsymbol{X}_k + \boldsymbol{v}_k & \boldsymbol{v}_k &\sim N(0, R)\\
\boldsymbol{Z}_k &= C_z\boldsymbol{X}_k
\end{aligned}$$

around the steady-state (x_s, u_s, y_s, z_s). The matrices (A, B, G, C, C_z) are defined by

$$\begin{aligned}
A &= \frac{\partial F}{\partial x}(x_s, u_s, 0) & B &= \frac{\partial F}{\partial u}(x_s, u_s, 0) & G &= \frac{\partial F}{\partial w}(x_s, u_s, 0)\\
C &= \frac{\partial g}{\partial x}(x_s) & C_z &= \frac{\partial h}{\partial x}(x_s)
\end{aligned}$$

Stochastic Systems - Linear Continuous-Discrete Time

$$d\boldsymbol{x}(t) = A\boldsymbol{x}(t)dt + Gd\boldsymbol{w}(t)$$

Let $\boldsymbol{x}(t_0) \sim N(\bar{x}_0, P_0)$ and let $\{\boldsymbol{w}(t)\}$ be a standard Wiener process. The solution to this system is

$$\boldsymbol{x}(t) = e^{A(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{A(t-s)}Gd\boldsymbol{w}(s)$$

and has the distribution

$$\boldsymbol{x}(t) \sim N(\bar{x}(t), P(t))$$

$$\begin{aligned}
\bar{x}(t) &= E\left\{\boldsymbol{x}(t)\right\} = e^{A(t-t_0)}\bar{x}(t_0) \\
P(t) &= V\left\{\boldsymbol{x}(t)\right\} = e^{A(t-t_0)}P(t_0)e^{A'(t-t_0)} + \int_{t_0}^{t} e^{A(t-s)}GG'e^{A'(t-s)}ds
\end{aligned}$$

Mean

$$
\begin{aligned}
\bar{x}(t) &= E\left\{\boldsymbol{x}(t)\right\} \\
&= E\left\{e^{A(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{A(t-s)} G d\boldsymbol{w}(s)\right\} \\
&= e^{A(t-t_0)} E\left\{\boldsymbol{x}(t_0)\right\} + E\left\{\int_{t_0}^{t} e^{A(t-s)} G d\boldsymbol{w}(s)\right\} \\
&= e^{A(t-t_0)} \bar{x}(t_0)
\end{aligned}
$$

Covariance

$$
\begin{aligned}
P(t) &= E\left\{(\boldsymbol{x}(t) - \bar{x}(t))(\boldsymbol{x}(t) - \bar{x}(t))'\right\} \\
&= e^{A(t-t_0)} E\left\{(\boldsymbol{x}(t_0) - \bar{x}(t_0))(\boldsymbol{x}(t_0) - \bar{x}(t_0))'\right\} e^{A'(t-t_0)} \\
&\quad + e^{A(t-t_0)} E\left\{(\boldsymbol{x}(t_0) - \bar{x}(t_0))\left(\int_{t_0}^{t} e^{A(t-s)} G d\boldsymbol{w}(s)\right)'\right\} \\
&\quad + E\left\{\left(\int_{t_0}^{t} e^{A(t-s)} G d\boldsymbol{w}(s)\right)(\boldsymbol{x}(t_0) - \bar{x}(t_0))'\right\} e^{A'(t-t_0)} \\
&\quad + E\left\{\left(\int_{t_0}^{t} e^{A(t-s)} G d\boldsymbol{w}(s)\right)\left(\int_{t_0}^{t} e^{A(t-s)} G d\boldsymbol{w}(s)\right)'\right\} \\
&= e^{A(t-t_0)} P(t_0) e^{A'(t-t_0)} + \int_{t_0}^{t} e^{A(t-s)} G G' e^{A'(t-s)} ds
\end{aligned}
$$

Stochastic Systems - Linear Continuous-Discrete Time

$$
\begin{aligned}
\boldsymbol{x}_{k+1} = \boldsymbol{x}(t_{k+1}) &= e^{A(t_{k+1}-t_k)}\boldsymbol{x}(t_k) + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-s)} G d\boldsymbol{w}(s) \\
&= e^{AT_s}\boldsymbol{x}_k + \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-s)} G d\boldsymbol{w}(s) \\
&= e^{AT_s}\boldsymbol{x}_k + \boldsymbol{w}_k
\end{aligned}
$$

with

$$
\boldsymbol{w}_k = \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-s)} G d\boldsymbol{w}(s)
$$

$\boldsymbol{w}_k \sim N_{iid}(0, \bar{Q})$

$$
\begin{aligned}
\bar{Q} = E\left\{\boldsymbol{w}_k \boldsymbol{w}_k'\right\} &= E\left\{\left(\int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-s)} G d\boldsymbol{w}(s)\right)\left(\int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-s)} G d\boldsymbol{w}(s)\right)'\right\} \\
&= \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-s)} G G' e^{A'(t_{k+1}-s)} ds = \int_0^{T_s} e^{A\tau} G G' e^{A'\tau} d\tau
\end{aligned}
$$

Stochastic Systems - Linear Continuous-Discrete Time

$$d\boldsymbol{x}(t) = A\boldsymbol{x}(t)dt + Gd\boldsymbol{w}(t)$$

is equivalent to

$$\boldsymbol{x}_{k+1} = \bar{A}\boldsymbol{x}_k + \boldsymbol{w}_k \qquad \boldsymbol{x}_0 \sim N(\bar{x}_0, P_0),\ \boldsymbol{w}_k \sim N_{iid}(0, \bar{Q})$$

with

$$\bar{A} = e^{AT_s}$$
$$\bar{Q} = \int_0^{T_s} e^{A\tau} G G' e^{A'\tau} d\tau$$

The matrices $(\bar{A}, \bar{Q})$ can be computed by

$$\begin{bmatrix} \Phi_{11} & \Phi_{12} \\ 0 & \Phi_{22} \end{bmatrix} = \exp\left(\begin{bmatrix} -A & GG' \\ 0 & A' \end{bmatrix} T_s \right)$$
$$\bar{A} = \Phi_{22}'$$
$$\bar{Q} = \Phi_{22}'\Phi_{12}$$

Stochastic Systems - Linear Continuous-Discrete Time

The continuous-discrete stochastic system with ZOH input $u(t)$

$$
\begin{aligned}
d\boldsymbol{x}(t) &= (A\boldsymbol{x}(t) + Bu(t))\,dt + Gd\boldsymbol{w}(t) \\
\boldsymbol{y}(t_k) &= C\boldsymbol{x}(t_k) + \boldsymbol{v}(t_k) \\
\boldsymbol{z}(t_k) &= C_z\boldsymbol{x}(t_k)
\end{aligned}
$$

$$
\begin{aligned}
&\boldsymbol{x}_0 \sim N(\bar{x}_0, P_0) \\
&\{\boldsymbol{w}(t)\} \quad \text{standard Wiener process} \\
&\boldsymbol{v}(t_k) = \boldsymbol{v}_k \sim N_{iid}(0, R)
\end{aligned}
$$

is equivalent to the discrete-time stochastic system

$$
\begin{aligned}
\boldsymbol{x}_{k+1} &= \bar{A}\boldsymbol{x}_k + \bar{B}u_k + \bar{G}\boldsymbol{w}_k \\
\boldsymbol{y}_k &= C\boldsymbol{x}_k + \boldsymbol{v}_k \\
\boldsymbol{z}_k &= C_z\boldsymbol{x}_k
\end{aligned}
$$

$$
\begin{aligned}
&\boldsymbol{x}_0 \sim N(\bar{x}_0, P_0) \quad \boldsymbol{w}_k \sim N_{iid}(0, \bar{Q}) \\
&\boldsymbol{v}_k \sim N(0, R)
\end{aligned}
$$

$$
\bar{A} = e^{AT_s} \quad \bar{B} = \int_0^{T_s} e^{A\tau}d\tau B
$$

$$
\bar{G} = I \qquad \bar{Q} = \int_0^{T_s} e^{A\tau}GG'e^{A'\tau}d\tau
$$

$$
\begin{bmatrix} \bar{A} & \bar{B} \\ 0 & I \end{bmatrix} = \exp\left(\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} T_s\right)
$$

$$
\begin{bmatrix} \Phi_{11} & \Phi_{12} \\ 0 & \Phi_{22} \end{bmatrix} = \exp\left(\begin{bmatrix} -A & GG' \\ 0 & A' \end{bmatrix} T_s\right)
$$

$$
\bar{Q} = \Phi_{22}'\Phi_{12}
$$

Learning Objectives

After this lecture you should be able to

1. Apply conservation of mass to develop simple first-principle models
2. Simulate systems described by ODEs using Matlab
3. Compute steady-states and linearize a system around a steady state
4. Discretize a linear continuous-time state space system
5. Derive transfer functions for linear state-space systems
6. Do simulations of stochastic systems
7. Discretize a continuous-time stochastic system

Exercise 1

Consider the 4-tank system with the parameters given in p. 12. Let $F_{1s} = 250$ cm^3/s and $F_{2s} = 325$ cm^3/s.

1. Compute the steady-state of this system
2. Simulate the response for a 5, 10, and 25% step increase in F_1, respectively.
3. Linearize the model at steady state. What is (A, B, C, D) for the continuous-time system?
4. Discretize the system using a sample time of $T_s = 4$ seconds. What is (A, B, C, D) for the discrete time system.
5. Simulate step responses for a 5, 10, and 25% step increase in F_1, respectively using the linear model. Compare these responses to the responses of the nonlinear model.
6. Compute the continuous-time and discrete-time transfer functions for this 4-tank system. What are the gain, poles and zeros for these systems?

Exercise 2

Consider the continuous-time system

$$\begin{aligned} d\boldsymbol{x}(t) &= A\boldsymbol{x}(t)dt + B\left(u(t)dt + \sigma d\boldsymbol{w}(t)\right) \\ &= \left(A\boldsymbol{x}(t) + Bu(t)\right)dt + B\sigma d\boldsymbol{w}(t) \end{aligned}$$

with σ being 10% of the mean value (steady state value) of u(t). Consider the above model applied for the 4-tank system (see Exercise 1).

- Discretize this system (for the 4-tank system example)
- Simulate the discrete time stochastic system. Make time series plot and histograms of the outputs.

Questions and Comments

John Bagterp Jørgensen
jbjo@dtu.dk

Technical University of Denmark
Department of Applied Mathematics and Computer Science