Discrete Time Optimal Control with Delay

Yang Xu

Department of Automatic Control
Lund University

Outline

1. Problem formulation
2. Optimal controller design
3. Jitterbug toolbox

Outline

1. Problem formulation
2. Optimal controller design
3. Jitterbug toolbox

Problem formulation

- Continuous time plant

$$\dot{x}(t) = Ax(t) + Bu(t-\tau)$$

- Discrete time controller
- Minimize the loss function

$$J = \int_0^{Nh} \begin{pmatrix} x^T(t) & u^T(t) \end{pmatrix} Q_c \begin{pmatrix} x(t) \\ u(t) \end{pmatrix} dt + x^T(Nh) Q_{0c} x(Nh)$$

Sampling a time delay system

Sample the continuous-time system

$$
\begin{aligned}
x(kh+h) =& e^{Ah}x(kh) + \int_{kh+\tau}^{kh+h} e^{A(kh+h-s)} ds Bu(kh) \\
&+ \int_{kh}^{kh+\tau} e^{A(kh+h-s)} ds Bu(kh-h) \\
=& \Phi x(kh) + \Gamma_0 u(kh) + \Gamma_1 u(kh-h)
\end{aligned}
$$

Extended state space model:

$$
\begin{pmatrix} x(kh+h) \\ u(kh) \end{pmatrix} = \begin{pmatrix} \Phi & \Gamma_1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x(kh) \\ u(kh-h) \end{pmatrix} + \begin{pmatrix} \Gamma_0 \\ I \end{pmatrix} u(kh)
$$

Sampling the loss function

Discrete time loss function

$$J = \sum_{k=0}^{N-1} \begin{pmatrix} x^T(kh) & u^T(kh) \end{pmatrix} Q \begin{pmatrix} x(kh) \\ u(kh) \end{pmatrix} + x^T(Nh) Q_{0c} x(Nh)$$

where

$$Q_1 = \int_{kh}^{kh+h} \Phi^T(s,kh) Q_{1c} \Phi(s,kh) ds$$

$$Q_{12} = \int_{kh}^{kh+h} \Phi^T(s,kh) (Q_{1c} \Gamma(s,kh) + Q_{12c}) ds$$

$$Q_2 = \int_{kh}^{kh+h} (\Gamma^T(s,kh) Q_{1c} \Gamma(s,kh) + 2\Gamma^T(s,kh) Q_{12c} + Q_{2c}) ds$$

Outline

Optimal controller design

The cost from the last step to the first step:

$$\begin{aligned} J_N^*(x_N) =& x_N^T Q_{0c} x_N \\ J_{N-1}^*(x_{N-1}, u_{N-1}) =& \min_{u_{N-1}} (x_{N-1}^T Q_{1c} x_{N-1} + 2x_{N-1}^T Q_{12} u_{N-1} \\ & + u_{N-1}^T Q_2 u_{N-1} + x_N^T Q_{0c} x_N) \\ ... =& ... \end{aligned}$$

In order to get the optimal value, we have

$$\frac{\partial J_{N-1}^*}{\partial u_{N-1}} = 0$$

$$\frac{\partial^2 J_{N-1}^*}{\partial u_{N-1}^2} > 0$$

Optimal controller design

From the previous conditions,

$$u^*_{N-1} = -(Q_2+\Gamma^T Q_{0c}\Gamma)^{-1}(2Q_{12}^T+\Gamma^T Q_{0c}\Phi)x_{N-1} = -L_{N-1}x_{N-1}$$

$$Q_2 + \Gamma^T Q_{0c}\Gamma > 0$$

The cost function can be written

$$\begin{aligned} J^*_{N-1}(x_{N-1}) =& x^T_{N-1}(Q_1 + 2Q_{12}L_{N-1} + L^T_{N-1}Q_2 L_{N-1} \\ &+ (\Phi - \Gamma L_{N-1})^T Q_{0c}(\Phi - \Gamma L_{N-1}))x_{N-1} \\ =& x^T_{N-1}P_{N-1}x_{N-1} \end{aligned}$$

Calculation procedure

$$
\begin{aligned}
P_N =& Q_{0c} \\
L_k =& (Q_2 + \Gamma^T Q_{0c} \Gamma)^{-1} (2Q_{12}^T + \Gamma^T Q_{0c} \Phi) \\
P_k =& Q_1 + 2Q_{12} L_{N-1} + L_{N-1}^T Q_2 L_{N-1} \\
& + (\Phi - \Gamma L_{N-1})^T Q_{0c} (\Phi - \Gamma L_{N-1})
\end{aligned}
$$

Eliminate L_k:

$$
\begin{aligned}
P_k =& Q_1 + \Phi^T P_{k+1} \Phi \\
& - (\Phi^T P_{k+1} \Gamma + Q_{12})(\Gamma^T P_{k+1} \Gamma + Q_2)^{-1} (\Gamma^T P_{k+1} \Phi + Q_{12}^T)
\end{aligned}
$$

Let $P_k = P_{k+1}$ to get discrete time algebraic Riccati equation

$$
Q_1 + \Phi^T P_k \Phi - (\Phi^T P_k \Gamma + Q_{12})(\Gamma^T P_k \Gamma + Q_2)^{-1} (\Gamma^T P_k \Phi + Q_{12}^T) \\
- P_k = 0
$$

Outline

Jitterbug

A MATLAB-based toolbox that allows the computation of a quadratic performance criterion for a linear control system under various timing conditions

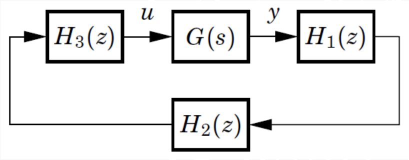

Figure: Signal model

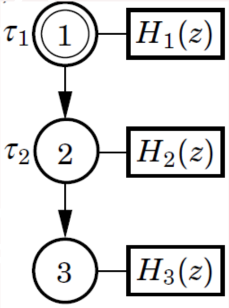

Figure: Timing model

Jitterbug command

Command	Description
initjitterbug	Initialize a new JITTERBUG system.
addtimingnode	Add a timing node.
addcontsys	Add a continuous-time system.
adddiscsys	Add a discrete-time system to a timing node.
adddiscexec	Add an execution of a previously defined discrete-time system.
adddisctimedep	Add time-dependence to a previously defined discrete-time system.
calcc2d	Internal function used to sample continuous-time dynamics, cost, and noise
calcdynamics	Calculate the internal dynamics of a JITTERBUG system.
calccost	Calculate the total cost of a JITTERBUG system and, for periodic systems, calculate the spectral densities of the outputs.
lqgdesign	Design a discrete-time LQG controller for a continuous-time plant with a constant or random time delay and a continuous-time cost function.

An example

```
s = tf('s');
P = 1/(s^2-1);                       % The process (inverted pendulum)

R1c = 1;                             % Continuous-time input noise
R2 = 0.01;                           % Discrete-time measurement noise
Qc = diag([1 0.01]);                 % Cost J = E(y^2 + 0.001*u^2)

h = 0.3;                             % Sampling period
tau = 0.15;                          % Assumed input-output delay

S = eye(1);                          % Sampler system
CA = lqgdesign(P,Qc,R1c,R2,h,tau);   % LQG controller

dt = h/10;                           % Time-grain

% Ptau = 1;                            % Zero delay
Ptau = [0 0 0 0 0 1.0 0 0 0 0 0];    % Constant delay of h/2
% Ptau = ones(1,round(h/dt));          % Uniform random delay in [0,h]
% Ptau = Ptau/sum(Ptau);

N = initjitterbug(dt,h);             % Initialize Jitterbug
N = addtimingnode(N,1,Ptau,2);       % Add node 1 (the periodic node)
N = addtimingnode(N,2);              % Add node 2
N = addcontsys(N,1,P,3,Qc,R1c,R2);   % Add sys 1 (P), input from sys 3
N = adddiscsys(N,2,S,1,1);           % Add sys 2 (S), input from 1
N = adddiscsys(N,3,CA,2,2);          % Add sys 3 (CA), input from 2
N = calcdynamics(N);                 % Calculate the internal dynamics
J = calccost(N);                     % Calculate the cost
```

Thank you.