Control of Medical Instrumentation VU, SS2020
Matlab/Simulink Introduction - 2^{nd} Part

Lukas Knoflach

June 4^{th}, 2020

1 Discrete-Time Systems

Consider the following continuous-time transfer function

$$G_1(s) = \frac{4}{s^2 + 2s + 4}$$

(a) Define the given transfer function $G_1(s)$ and determine, if the the system is BIBO-stable. Plot the step response of $G_1(s)$ for $0 \leq t \leq 5$s
(Discussed in the 1^{st} Matlab/Simulink introduction)

(b) The transfer function should be transformed into a discrete-time system with the sampling time $T_d = 0.05$s. Compare the two different discretization methods *impulse invariant discretization* and *zero-order hold* for $0 \leq t \leq 5$s.
(Discussed in the 1^{st} Matlab/Simulink introduction)

(c) Create a Simulink-model for $G_1(s)$ and add a hold element at the input and a sample element at the output. Compare the discrete-time step response for $0 \leq t \leq 5$s with the discretization methods from task b).
(Matlab commands: `stairs`, `stem`)

(d) Compare the step response for $0 \leq t \leq 5$s for different sampling times

(i) $T_d = 0.05$s (ii) $T_d = 0.2$s (iii) $T_d = 0.55$s

and interpret the results. Use the Simulink-model from task c) and create a stairstep and a stem-graph for comparison.
Hint: use a `for` loop
(Matlab commands: `subplot`, `sprintf`)

2 State controller

Consider the following linear system

$$\frac{d\mathbf{x}}{dt} = \begin{bmatrix} 0 & -2 \\ 2 & 7 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 1 \\ -4 \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 3 \end{bmatrix} \mathbf{x}$$

with the initial state $x_0 = \begin{bmatrix} 3 & -2 \end{bmatrix}^T$ and the reference signal $r(t) = 30\sigma(t)$.

(a) Design a state controller $u = -\mathbf{k}^T\mathbf{x} + r$. The eigenvalues of the closed loop system matrix $(\mathbf{A} - \mathbf{b}\mathbf{k}^T)$ should be located at $\lambda_1 = \lambda_2 = -2$.
(Matlab command: `acker`)

(b) Create a Simulink-model of the system using a Matlab function block for the given plant. Run the simulation for 10 seconds and compare the reference signal $r(t)$ with the output of the system $y(t)$.

(c) (**optional**) Enhance the state controller with a PI state controller to compensate the constant offset. The new system has an additional state variable $\frac{d\varepsilon}{dt} = r - y$ and is defined as

$$\frac{d}{dt}\begin{bmatrix}\mathbf{x}\\ \varepsilon\end{bmatrix} = \tilde{A}\begin{bmatrix}\mathbf{x}\\ \varepsilon\end{bmatrix} + \tilde{b}u = \begin{bmatrix}\mathbf{A} & 0\\ -\mathbf{c^T} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \varepsilon\end{bmatrix} + \begin{bmatrix}\mathbf{b}\\ 0\end{bmatrix}u$$

Design a PI state controller of the form

$$u = -\mathbf{k}^T\mathbf{x} - k_i\varepsilon - k_p(r-y)$$

such that $\tilde{k}^T = \begin{bmatrix}\mathbf{k}^T - k_p\mathbf{c^T} & k_i\end{bmatrix}$ and the eigenvalues are located at $\lambda_1 = \lambda_2 = \lambda_3 = -2$, $k_p = \frac{1}{\mathbf{c}^T\mathbf{A}^{-1}\mathbf{b}}$ and $\mathbf{k}^T = \begin{bmatrix}\tilde{k}_1 & \dots & \tilde{k}_{n-1}\end{bmatrix} + k_p\mathbf{c^T}$.
Implement the PI-state controller in the Simulink model using a Matlab function block and compare the result with task b).
(Matlab commands: `acker`, `inv`)

3 Dealing with operating points

Consider the following nonlinear continuous-time mathematical model of the plant

$$\begin{aligned}
\dot{x}_1 &= x_2\\
\dot{x}_2 &= g - \frac{c}{m}\frac{x_3^2}{x_1^2}\\
\dot{x}_3 &= -\frac{R}{L}x_3 + \frac{2c}{L}\frac{x_2x_3}{x_1^2} + \frac{1}{L}u\\
y &= x_1
\end{aligned}$$

(a) Implement the given system in a Matlab function block with the parameters

$g = 9.81$ $\quad$ $L = 1.08$
$c = 0.00013632$ $\quad$ $R = 18$
$m = 0.06687$

A state controller should be designed for the operating point $x_e^T = \begin{bmatrix}y_e & 0 & y_e\sqrt{\frac{mg}{c}}\end{bmatrix}$ and $u_e = Ry_e\sqrt{\frac{mg}{c}}$ where $y_e = 0.013$ and the deviations of the operating point are denoted by $\Delta x = x - x_e$ and $\Delta u = u - u_e$. To allow the implementation of a state controller, the model is linearized around the operating point which results in the linear model (derivation not important):

$$\begin{aligned}
\frac{d\mathbf{\Delta x}}{dt} &= \begin{bmatrix}0 & 1 & 0\\ \frac{2g}{y_e} & 0 & -\frac{2}{y_e}\sqrt{\frac{cg}{m}}\\ 0 & \frac{2}{Ly_e}\sqrt{cmg} & -\frac{R}{L}\end{bmatrix}\mathbf{\Delta x} + \begin{bmatrix}0\\ 0\\ \frac{1}{L}\end{bmatrix}\Delta u\\
\Delta y &= \begin{bmatrix}1 & 0 & 0\end{bmatrix}\mathbf{\Delta x}
\end{aligned}$$

(b) Compute the corresponding discrete-time system of the linearized model with a sampling time of $T_d = 5$ms.
(Matlab command: `ss`)

(c) Compute a state controller for the discrete-time system $\Delta u = -\mathbf{k}^T\mathbf{\Delta x}$ where the eigenvalues are located at $\lambda_1 = \lambda_1 = \lambda_1 = 0.8$.
(Matlab command: `acker`)

(d) Enhance the Simulink model with the state controller. Run the simulation for one second and plot the output with the initial value $x_0 = 3x_e$ and the reference signal $r = 3y_e$.

(e) Constrain the actuating variable u in the Simulink model by

(i) $0 \le u \le 40$ $\quad$ (ii) $0 \le u \le 100$

and discuss the differences.

4 Addendum

- Matlab Live Script
  https://de.mathworks.com/videos/using-the-live-editor-117940.html
- Export Simulink model to previous version

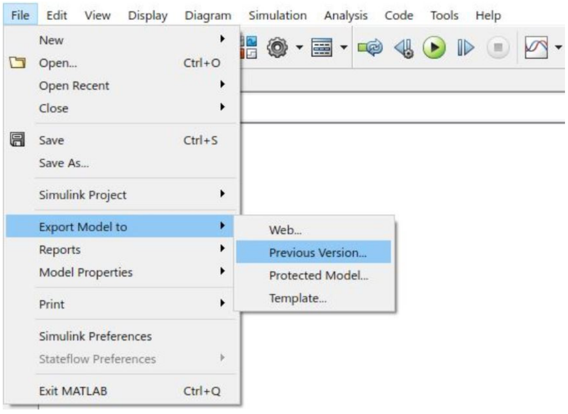

Figure 1