Homework3

Control Theory

March 14, 2020

Deadline is March 29, 2020, 23:59 (MSK)

All results should be beautiful packed in one pdf and loaded to Github in PR. Plots should be signed. Do not forget to describe and explain your results.

.

1. Select your variant.

   Open link: Link to variants

   Change name and email and press button "run".

   You will see your variant on the right side. Use it for all tasks.

   Put name and email that you use for generation and your variant in the report.

2. In this task use python for calculations.

   (A) Design PD-controller that tracks time varying reference states i.e. $[x^*(t), \dot{x}^*(t)]$ as closely as possible. Test your controller on different trajectories, at least two. System: $\ddot{x} + \mu\dot{x} + kx = u$, see variants below.

   (B) Tune controller gains k_p and k_d . Find gains that provide no ocsillations and no overshoot. Prove it with step input.

   (C) Prove that controlled oscillator dynamics is stable for your choice of k_p and k_d .

   (D) Think of how you would implement PD control for a linear system:

   $$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 10 & 3 \\ 5 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

   (E) Implement a PI/PID controller for the system: $\ddot{x} + \mu\dot{x} + kx + 9.8 = u$ (variants are below) Test your controller on different trajectories, at least two.

   Variants for the task:

(a) $\mu = 3, k = 45$
(b) $\mu = 44, k = 1$
(c) $\mu = 13, k = 2$
(d) $\mu = 35, k = 5$
(e) $\mu = 14, k = 85$
(f) $\mu = 63, k = 15$
(g) $\mu = 7, k = 25$

3. Design a PID controller. Use step input function and try to improve rise time, overshoot and steady-state error, comparing with no controller system. Describe your actions. Use pidTuner in Matlab.

   Variants for the task:

   (a) $W = \frac{s+4}{s^2+3s+15}$
   (b) $W = \frac{s^2+3s+8}{s^4+2s^3+3s^2+13s+7}$
   (c) $W = \frac{s+1}{s^3+2s^2+9s+20}$
   (d) $W = \frac{3s^2+4s+10}{s^4+3s^3+7s^2+19s+30}$
   (e) $W = \frac{s+2}{2s^2+7}$
   (f) $W = \frac{s+3}{2s^3+4s^2+7s+13}$
   (g) $W = \frac{s+2}{s^2+4s+11}$

4. Design a lag or lead compensator (if applicable), play with zero and pole to find optimal values (of overshoot, peak time, transient process time, stationary error, etc.) for transient process. Use editors in Matlab Control System Designer.

   Variants for the task:

   (a) $W(s) = \frac{s+2}{2s^2+7}$
   (b) $W(s) = \frac{s+3}{2s^3++4s^2+7s+13}$
   (c) $W(s) = \frac{s+2}{s^2+4s+11}$
   (d) $W(s) = \frac{s^2+3s+8}{s^4+2s^3+3s^2+13s+7}$
   (e) $W(s) = \frac{s+4}{s^2+3s+15}$
   (f) $W(s) = \frac{s^2+3s+8}{s^4+2s^3+3s^2+13s+7}$
   (g) $W(s) = \frac{s+1}{s^3+2s^2+9s+20}$

Some suggestions(the same):

All results should be in the PullRequest(PR) in a separate branch.

Do not forget to add Readme.md with description what the repo is, how to run the code, all dependencies installation, maybe other instructions. Put matlab/python versions, operating system that you use.

Report should contains calculations, pictures, maybe some descriptions, all - but only useful information.

For pdf generation, i suggest practising in usage LaTeX with overleaf.com for example.

Merge PR only after Mike approve it.

Do not leave the task for the last day. You can meet difficulties that you will not manage to solve in 30 minutes.