ELEC 341 – Graded Assignments

Assignment A8
Optimal Control

100 Marks

Required Files

Available on Canvas

- **e341-a8.pdf** *Assignment description (this document)*
- **a8Submit.p** *Grading script (**LATEST** version)*
- **e341-APE.pdf** *Instructions for submitting graded work (for reference)*

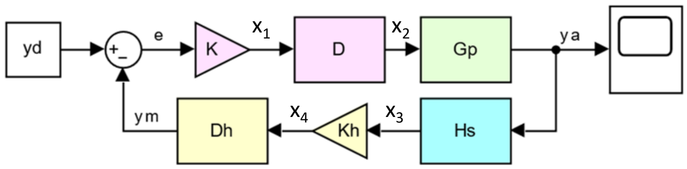
Topics

PD, PI & PID Control
- branch dynamics

Nyquist Contour
- gain & phase margin

Optimal Control
- PD, PI & PID control

Figure 1



Parameter	Value	Physical Units
τ_{pid}	#D / 2	ms
ω_n res	10^{-1}	rad/s
ζ res	10^{-2}	pure
Target PM	60	deg

Re-design the **PD**-controller developed in **A7** (see **Figure 1**).
Specify the forward path gain **G**: $\mathbf{G = y_a/x_2}$
Specify the feedback path gain **H**: $\mathbf{H = y_m/y_a}$

1. 0 mark(s) A7 Parameters

- Q1.G (m/V) LTI
- Q1.H (m/m) LTI

Derivative branch noise is compensated by an **IIR** filter with time constant $\boldsymbol{\tau_{pid}}$.
Find the filter **beta**, filter delay $\mathbf{N_f}$ and total delay multiple $\mathbf{N_{pid}}$.
Determine the associated pole frequency $\boldsymbol{\omega_p}$.

2. 10 mark(s) IIR Filter

- Q2.beta (pure) Scalar
- Q2.Nf (pure) Scalar
- Q2.Npid (pure) Scalar
- Q2.wp (rad/s) Scalar

Find the initial gain $\mathbf{K_0}$ and cross-over frequency $\boldsymbol{\omega_{xo}}$.
Use the search resolution ($\boldsymbol{\omega_n}$) to find the optimal zero **Z** and controller dynamics **D**.

3. 10 mark(s) PD Dynamics

- Q3.K0 (V/m) Scalar
- Q3.wxo (rad/s) Scalar
- Q3.Z (rad/s) Scalar
- Q3.D (pure) LTI

Find master gain **K** to satisfy the RCG (**Target PM**).
Find the normalized (**K=1**) proportional $\mathbf{K_p}$ and derivative $\mathbf{K_d}$ gains.

4. 10 mark(s) PD Gains

- Q4.K (V/m) Scalar
- Q4.Kp (pure) Scalar
- Q4.Kd (pure) Scalar

Find settle time $\mathbf{T_s}$ and steady-state error $\mathbf{E_{ss}}$.

5. 10 mark(s) PD Metrics

- Q5.Ts (s) Scalar
- Q5.Ess (%) Scalar

Re-design the **PI**-controller developed in **A7** (see **Figure 1**).
Find the initial gain $\mathbf{K_0}$ and cross-over frequency $\boldsymbol{\omega_{xo}}$.
Use the search resolution ($\boldsymbol{\omega_n}$) to find the optimal zero **Z** and controller dynamics **D**.

6. 10 mark(s) PI Dynamics

- Q6.K0 (V/m) Scalar
- Q6.wxo (rad/s) Scalar
- Q6.Z (rad/s) Scalar
- Q6.D (pure) LTI

Find master gain **K** to satisfy the RCG (**Target PM**).
Find the normalized (**K=1**) proportional $\mathbf{K_p}$ and integral $\mathbf{K_i}$ gains.

7. 10 mark(s) PI Gains

- Q7.K (V/m) Scalar
- Q7.Kp (pure) Scalar
- Q7.Ki (pure) Scalar

Find rise time $\mathbf{T_r}$ and percentage over-shoot $\mathbf{OS_u}$.

8. 10 mark(s) PI Metrics

- Q8.Tr (s) Scalar
- Q8.OSu (%) Scalar

Design a **PID**-controller using the specs from **A7** and the filter from **Q2** (see **Figure 1**).

Find the initial gain $\mathbf{K_0}$ and cross-over frequency $\boldsymbol{\omega_{xo}}$.

Use the search resolution to find the optimal zeros **Z** and controller dynamics **D**.

9. 10 mark(s) PID Dynamics

• Q9.K0	(V/m)	Scalar
• Q9.wxo	(rad/s)	Scalar
• Q9.Z	(rad/s)	Vector 1x2
• Q9.D	(pure)	LTI

Find master gain **K** to satisfy the RCG (**Target PM**).

Find the normalized (**K=1**) proportional $\mathbf{K_p}$, integral $\mathbf{K_i}$, and derivative $\mathbf{K_d}$ gains.

10. 10 mark(s) PID Gains

• Q10.K	(V/m)	Scalar
• Q10.Kp	(pure)	Scalar
• Q10.Ki	(pure)	Scalar
• Q10.Kd	(pure)	Scalar

Find rise time $\mathbf{T_r}$ and percentage over-shoot $\mathbf{OS_u}$.

11. 10 mark(s) PID Metrics

• Q11.Tr	(s)	Scalar
• Q11.OSu	(%)	Scalar

COW:

You've done all the work. Now is a perfect opportunity to practice Heuristic Tuning.

*For **EACH** controller, adjust the individual gains until the actual output most closely resembles the desired output (a step).*

Try to make T_r, T_p, and T_s, all approximately equal with minimal OS_u and/or OS_y.

Compare your tuned zero(s) to the zero(s) from the Newtonian search you did in the assignment.

Pay attention to which performance aspects are most difficult to control with each controller. This will help you to appreciate what the individual gains can do for you.

Increase τ_{pid} and re-tune the controller. In practice, this is something you will need to do if you find your derivative branch is too noisy to be useful.