System Performance Analysis

transient specifications · steady-state error · Routh–Hurwitz · root locus

Abu Haqeem

MEC524 Control Systems · Handbook Chapter 4 · Syllabus topic 4 · Weeks 10–12

Semester 20264

How to use this chapter

Three documents, three different jobs. Use the right one.

These slides	Where every result *comes from*, and what it means. Follow the derivations here.
The handbook	The reference. Look up a result, a table, a worked example. Chapter 4, § 4.0–4.5.
The tutorial	Where you find out whether you actually understood it. § 4.6.

Every number here has been checked twice

Formula against simulation, Routh against the actual roots. The handbook carries the same checked numbers.

Where Chapter 4 is going

Chapter 3 told you what a system *is*. This chapter asks what it *does* — four questions, one per section.

1. **How fast, and how cleanly, does it respond?** Handbook § 4.1–4.2
2. **Where does it end up?** The steady-state error. Handbook § 4.3
3. **Does it stay bounded at all?** Stability, by Routh–Hurwitz. Handbook § 4.4
4. **Where do the poles go as I turn up the gain?** Root locus. Handbook § 4.5

PBL Milestone 2 and Chapter 5 are built on these

A PID design is judged on overshoot, settling time and steady-state error, inside a stable gain range. Chapter 5 asks the stability question again, in frequency.

You have already felt every specification

No formulas yet: five things you have lived through, and their names.

You notice...	**What it does**	**Its name here**
a shower takes a while to run hot	creeps up, never past	time constant τ
a lift passes your floor, sinks back	goes beyond, returns	overshoot, peak time
a car seat bounces after a bump	rings, dies away	damping ζ, settling time
a phone camera trails a walking face	arrives, not exactly	steady-state error
a microphone squeals louder and louder	grows, never dies	instability

Each row is a number you can read out of $G(s)$ *before* building anything.

Where the behaviour hides: one pole, one term

Partial fractions (Chapter 1) split any response into one term per pole p:

$$y(t) = y_{ss} + \sum_{\text{poles}} c\, e^{pt}$$

p real, negative

e^{-2t} dies away without ringing. The shower.

p complex, $\mathrm{Re} < 0$

$e^{-t} \sin 3t$ rings inside a shrinking envelope. The lift, the seat.

$\mathrm{Re}\, p > 0$

$e^{+0.5t}$ grows for ever. The squealing microphone.

So "how fast, how cleanly, does it stay put" are all questions about *where the poles sit*. This chapter turns each into a formula — and four animations let you watch it happen.

Three questions for every closed loop

A linear positioning stage: a motor, a leadscrew, a controller. You command a new position and watch the carriage move. Handbook § 4.0

Transient

Does it get there quickly? Does it overshoot and ring?

Steady state

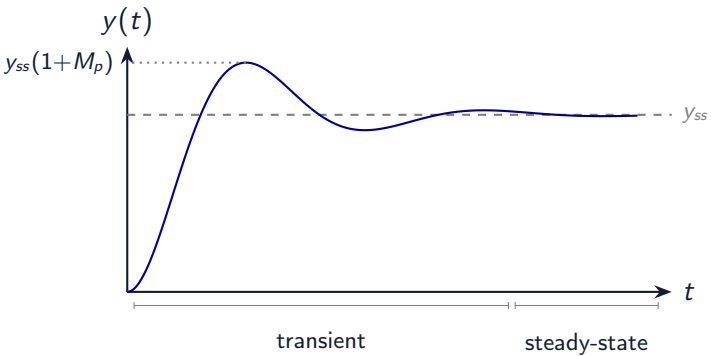
Once the motion dies away, is the carriage where you asked?

Stability

Turn the gain up for speed. Does it start to oscillate, then run away?

All three answers are already inside the transfer function.
This chapter teaches you to *read* them out.

Every response is transient plus steady state



Test inputs, and all three answers from $G(s)$

The standard inputs

Input	$R(s)$	**Probes**
step	$1/s$	overshoot, settling
ramp	$1/s^2$	tracking lag
parabola	$1/s^3$	acceleration lag

Any other command is built from these.

The mass–spring–damper of § 3.3, $m = 1$, $c = 2$, $k = 5$:

$$G(s) = \frac{1}{s^2 + 2s + 5}$$

Stable? Poles $s = -1 \pm j2$, both in the left half-plane.

Transient? $\omega_n = \sqrt{5} = 2.236\,\text{rad/s}$, $\zeta = 1/\sqrt{5} = 0.447$ — it will ring.

Steady state?
$y_{ss} = \lim_{s\to 0} sY(s) = 1/5 = 0.20$.

Handbook § 4.0.5

First order: one energy store, one exponential

An RC filter, a tank, a sensor, a motor's speed: one store, so $\tau\dot{y} + y = Ku$ and $G(s) = \dfrac{K}{\tau s + 1}$.

Handbook § 4.1

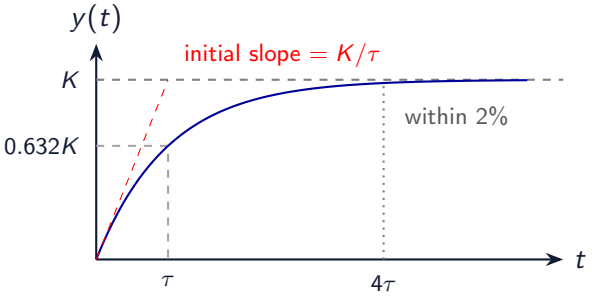
$$Y(s) = \frac{K}{s(\tau s + 1)} = \frac{K}{s} - \frac{K}{s + 1/\tau}$$

$$\boxed{y(t) = K\left(1 - e^{-t/\tau}\right)}$$

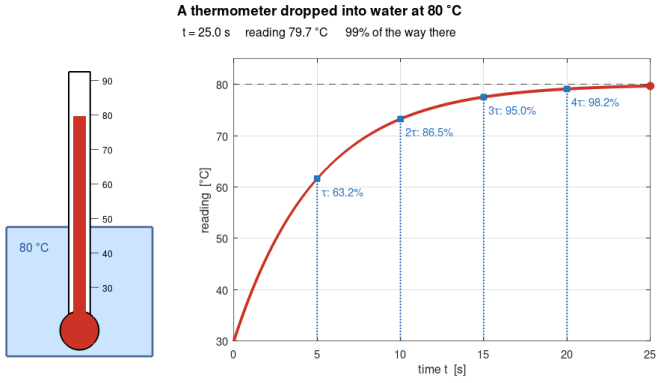
t	y/K
τ	$1 - e^{-1} = 63.2\,\%$
3τ	$95.0\,\%$
4τ	$98.2\,\%$ — inside the $2\,\%$ band

- Initial slope K/τ: the tangent at $t = 0$ meets K at $t = \tau$.
- One real pole at $-1/\tau$. It *cannot* overshoot.

The first-order step response, annotated



Watch a thermometer settle



A probe at 30°C dropped into water at 80°C, $\tau = 5\,\text{s}$. After τ it has covered 63.2 % of the gap (61.6°C); after 2τ, 86.5 %; 3τ, 95.0 %; 4τ, 98.2 % (79.1°C) — that is why $T_s(2\,\%) = 4\tau$. It never overshoots: one real pole.

GIF: `ch4-anim/MEC524-ch4-tau.gif`.

Ramp input: the output lags by τ

Command a steadily moving target, $R(s) = 1/s^2$, with $K = 1$:

$$Y(s) = \frac{1}{s^2(\tau s + 1)} = \frac{1}{s^2} - \frac{\tau}{s} + \frac{\tau}{s + 1/\tau}$$

$$y(t) = t - \tau + \tau e^{-t/\tau} \qquad \Longrightarrow \qquad e(t) = t - y(t) = \tau \left(1 - e^{-t/\tau}\right) \to \tau$$

Read it as an engineer

The output runs parallel to the command, τ seconds behind it, for ever. A slow loop cannot follow a fast reference — and no amount of waiting fixes that.

That constant lag is what the velocity error constant K_v of § 4.3 measures.

Handbook § 4.1.3

Worked examples — reading τ off a trace

Pressure transducer. Settles at 5 V; reads 4.2 V at 0.4 s.

Step 1 — normalise.
$4.2/5 = 0.84 = 1 - e^{-0.4/\tau}$

Step 2 — take logs. $e^{-0.4/\tau} = 0.16$, so $0.4/\tau = -\ln 0.16 = 1.833$

$$\tau = 0.218 \text{ s}$$

Step 3 — check. The trace should cross $0.632 \times 5 = 3.16$ V at $t \approx \tau$. Settling (2 %): $4\tau = 0.873$ s.

DC motor speed. $K = 5$ rad/s/V, $\tau_m = 0.05$ s, 12 V step.

Step 1 — final value. $12 \times 5 = 60$ rad/s (≈ 573 rpm).

Step 2 — the whole response.

$$\omega(t) = 60\left(1 - e^{-20t}\right) \text{ rad/s}$$

Step 3 — how long. 63 % at $\tau_m = 0.05$ s; within 2 % by $4\tau_m = 0.20$ s.

Handbook § 4.1.5–4.1.6

Second order: two energy stores, two poles

A positioning servo — inertia plus a controller — closed around its position sensor behaves like

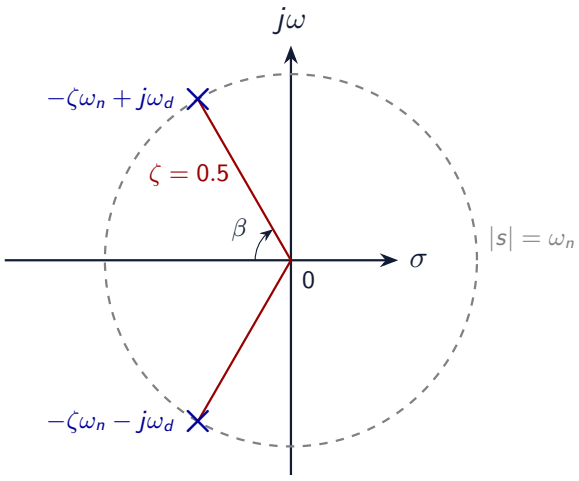
Handbook § 4.2.1

$$G(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \qquad s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$$

ζ	**Poles**	**Step response**
0	$\pm j\omega_n$ on the imaginary axis	rings for ever
$0 < \zeta < 1$	$-\zeta\omega_n \pm j\omega_d$, complex	decaying oscillation
1	$-\omega_n$, repeated	fastest with no overshoot
$\zeta > 1$	two negative reals	sluggish creep

$\omega_d = \omega_n\sqrt{1 - \zeta^2}$ — the frequency you actually see ringing

Pole geometry: radius and angle



Radius is speed, angle is damping

For $0 < \zeta < 1$ the pole is $-\zeta\omega_n + j\omega_d$. Measure it from the origin: Handbook § 4.2.2

$$|s| = \sqrt{\zeta^2\omega_n^2 + \omega_n^2(1-\zeta^2)} = \omega_n \qquad \cos\beta = \frac{\zeta\omega_n}{\omega_n} = \zeta$$

Move the pole radially out

ω_n rises, β fixed. Faster — *same* overshoot.

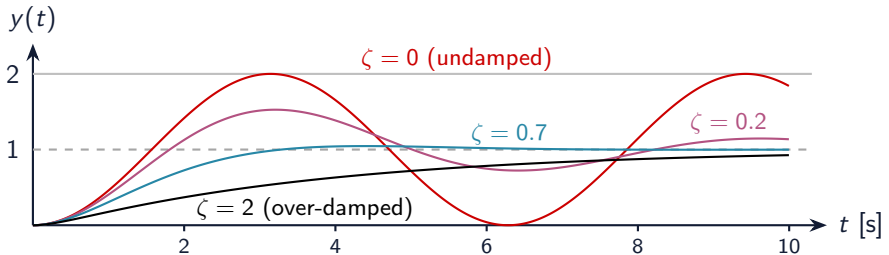
Rotate it toward the real axis

ζ rises. Less overshoot — but slower to rise.

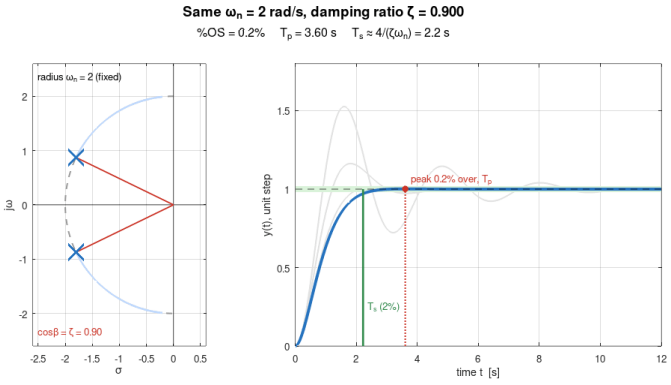
Every specification is a statement about where the poles may sit

Overshoot is a ray, settling time is a vertical line, peak time is a horizontal line. The next three slides show why.

Same natural frequency, four damping ratios



Watch the poles swing round the circle



$\omega_n = 2$ rad/s held fixed while ζ goes from 0.1 to 0.9: the poles slide round the circle of radius ω_n towards the real axis. Watch the overshoot shrink and the peak move later. No response from the controls? Your PDF viewer cannot animate — open `ch4-anim/MEC524-ch4-zeta.gif`.

Three frozen frames from the swing

$\omega_n = 2\,\text{rad/s}$, so $\omega_d = 2\sqrt{1-\zeta^2}$ and $\zeta\omega_n = 2\zeta$. Handbook § 4.2.3–4.2.4

ζ	ω_d [rad/s]	% OS	$T_p = \pi/\omega_d$	$T_s \approx 4/(\zeta\omega_n)$	**Feels like**
0.2	1.960	52.7 %	1.60 s	10.0 s	a bouncy seat
0.5	1.732	16.3 %	1.81 s	4.0 s	a brisk lift
0.8	1.200	1.52 %	2.62 s	2.5 s	a soft, sure stop

The trade you just watched

Turning the poles toward the real axis cut the overshoot *and* the settling time — but the peak arrives later (T_p 1.60 s $\to$ 2.62 s), because ω_d shrinks. Same radius, so the same ω_n throughout.

Deriving the under-damped step response

Unit step, $K = 1$, $0 < \zeta < 1$. Complete the square in the denominator:

$$Y(s) = \frac{\omega_n^2}{s\left(s^2 + 2\zeta\omega_n s + \omega_n^2\right)} = \frac{1}{s} - \frac{s + 2\zeta\omega_n}{(s + \zeta\omega_n)^2 + \omega_d^2}$$

Split the numerator so each piece is a shifted cosine or sine:

$$Y(s) = \frac{1}{s} - \frac{s + \zeta\omega_n}{(s + \zeta\omega_n)^2 + \omega_d^2} - \frac{\zeta\omega_n}{\omega_d}\,\frac{\omega_d}{(s + \zeta\omega_n)^2 + \omega_d^2}$$

$$\boxed{y(t) = 1 - e^{-\zeta\omega_n t}\left(\cos\omega_d t + \frac{\zeta}{\sqrt{1-\zeta^2}}\sin\omega_d t\right) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin(\omega_d t + \beta)}$$

An oscillation at ω_d, inside an envelope $e^{-\zeta\omega_n t}$, with $\beta = \cos^{-1}\zeta$.

Handbook § 4.2.3

Peak time and overshoot

Peak time. The derivative of a step response is the impulse response:

$$\dot{y}(t) = \frac{\omega_n}{\sqrt{1-\zeta^2}}\, e^{-\zeta\omega_n t} \sin \omega_d t$$

It is first zero again when $\omega_d t = \pi$: $\boxed{T_p = \dfrac{\pi}{\omega_d}}$

Overshoot. Put $t = T_p$ into $y(t)$: $\cos\pi = -1$, $\sin\pi = 0$, and $\zeta\omega_n T_p = \zeta\pi/\sqrt{1-\zeta^2}$:

$$y(T_p) = 1 + e^{-\zeta\pi/\sqrt{1-\zeta^2}} \qquad \Longrightarrow \qquad \boxed{\%\,\mathrm{OS} = 100\, e^{-\zeta\pi/\sqrt{1-\zeta^2}}}$$

ω_n cancelled

Overshoot depends on ζ alone. To go back from a specification:
$\zeta = -\ln(\mathrm{OS})/\sqrt{\pi^2 + \ln^2(\mathrm{OS})}$.

Settling time comes from the envelope

The oscillation stays below its envelope, so ask when the *envelope* is inside the band:

$$\frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}} = 0.02 \qquad \Longrightarrow \qquad t \approx \frac{-\ln 0.02}{\zeta\omega_n} = \frac{3.91}{\zeta\omega_n}$$

dropping the $\sqrt{1-\zeta^2}$ factor and rounding up: Handbook § 4.2.4

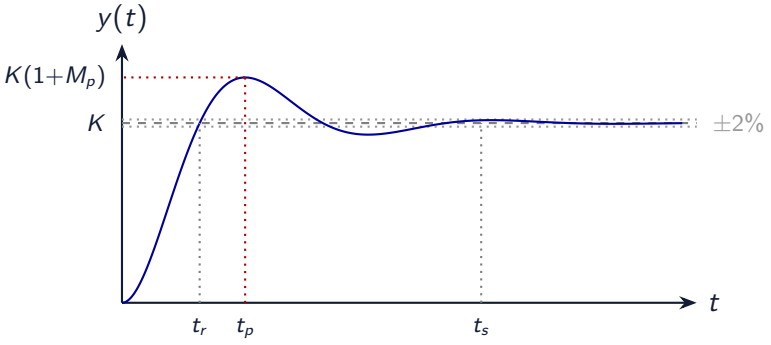
$$\boxed{T_s\,(2\,\%) \approx \frac{4}{\zeta\omega_n}} \qquad \boxed{T_s\,(5\,\%) \approx \frac{3}{\zeta\omega_n}} \qquad (-\ln 0.05 = 3.00)$$

Rise time (0–100 %). $y = 1$ first when $\sin(\omega_d t + \beta) = 0$: $\quad T_r = \dfrac{\pi - \beta}{\omega_d}$

Each specification pins one coordinate of the pole

T_s fixes the real part $\zeta\omega_n$. T_p fixes the imaginary part ω_d. % OS fixes the angle.

The specifications on one trace



Three specifications, one envelope

The settling-time slide bounded the error by its envelope. Turn that bound *into* the specification:

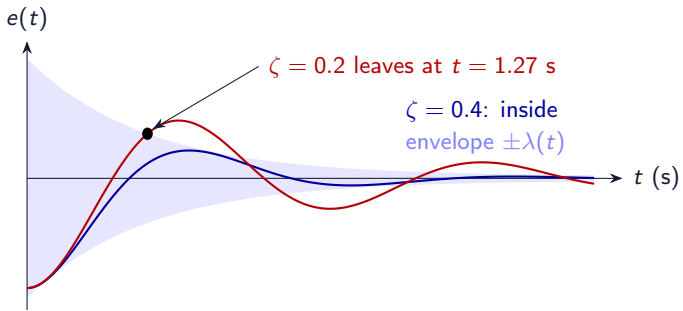
$$|e(t)| < \lambda(t) = (\lambda_0 - \lambda_\infty)\, e^{-t/\tau} + \lambda_\infty$$

Chapter 4 specification	Envelope parameter	Deck example ($\zeta = 0.4$, $\omega_n = 2$)
overshoot / initial error	$\lambda_0 = 1/\sqrt{1-\zeta^2}$	$\lambda_0 = 1.09$
settling time $T_s \approx 4\tau$	$\tau = 1/(\zeta\omega_n)$	$\tau = 1.25$ s
settling band, e_{ss}	λ_∞	$\lambda_\infty = 0.02$

A controller samples, so it runs the envelope in discrete time:
$\lambda_{k+1} = (1-\omega)\lambda_k + \omega\lambda_\infty, \quad \omega = 1 - e^{-T/\tau}$.

Enrichment, not assessed: this is the idea behind *prescribed performance control*.

One envelope, two designs



Reading three numbers off a trace becomes *one* yes/no check at every instant. The lighter-damped design fails it during its first overshoot.

The same envelope, enforced on a nano-stage

A piezoelectric stage (30 μm travel, 2 nm sensor) tracking a sine. Its controller is designed so that $|e_k| < \lambda_k$ at *every* sample.

- Measured at 1 Hz: $\max |e_k|/\lambda_k = 0.20$, comfortably inside.
- Measured at 30 Hz: $\max |e_k|/\lambda_k = 0.98$, inside, but only just.

Try it

`Get_start_Matlab/scripts/funnel_spec_demo.m` builds the envelope from the specifications, tests both designs, and plots the hardware runs.

Data: Prof. X. Chen's laboratory, Shibaura Institute of Technology (2016), used with permission for teaching only.

ζ against overshoot — the table worth memorising

ζ	% OS	ζ	% OS	ζ	% OS
0.1	72.9	0.4	25.4	0.7	4.60
0.2	52.7	0.5	16.3	0.707	4.33
0.3	37.2	0.6	9.48	0.8	1.52

- Non-linear: halving ζ from 0.4 to 0.2 roughly doubles the overshoot.
- $\zeta = 0.707$ is the usual compromise: 4.33 % overshoot, and $T_s \approx 5.66/\omega_n$.
- Most designs you will meet aim for ζ between 0.5 and 0.8.

Handbook § 4.2.5

Worked example W1 — specifications from the TF (1 of 2)

A position servo closes to

$$\frac{Y(s)}{R(s)} = \frac{25}{s^2 + 6s + 25}$$

Find ω_n, ζ, T_p, %OS and T_s (2%). Handbook §4.2.6

Step 1 — match the canonical form, term by term.

$$\omega_n^2 = 25 \Rightarrow \omega_n = 5 \text{ rad/s} \qquad 2\zeta\omega_n = 6 \Rightarrow \zeta = 0.6$$

Step 2 — the pole, before any formula.

$$\omega_d = 5\sqrt{1 - 0.36} = 5 \times 0.8 = 4.0 \text{ rad/s} \qquad s = -3 \pm j4$$

Step 3 — DC gain. $25/25 = 1$: the output ends where it was told to.

Worked example W1 — specifications from the TF (2 of 2)

Step 4 — now the formulas, each from its own coordinate.

$$T_p = \frac{\pi}{4} = 0.785 \text{ s} \qquad \%\,\mathrm{OS} = 100\, e^{-\pi(0.6)/0.8} = 100\, e^{-2.356} = 9.48\,\%$$

$$T_s\,(2\,\%) = \frac{4}{3} = 1.33 \text{ s} \qquad T_r = \frac{\pi - 53.13^\circ}{4} = 0.554 \text{ s} \quad (\beta \text{ in radians!})$$

Now check against the real trace (RK4 simulation)

Peak	9.48 % at 0.785 s	exact — the formulas are exact here
2 % settling	1.19 s	the estimate $4/(\zeta\omega_n)$ said 1.33 s

T_p and % OS are exact for two poles and no zero. T_s is an *estimate* — quote it as one.

Checking W1 in MATLAB/Octave

```
clear; clc; format compact
mec524_setup % loads control on Octave

G = tf(25, [1 6 25]); % W1 closed loop
p = pole(G) % -3 + 4i, -3 - 4i
[wn, zeta] = damp(G) % 5 and 0.6, per pole

S = stepinfo(G) % mec524lib on Octave
% S.Overshoot about 9.48
% S.PeakTime about 0.785
% S.SettlingTime about 1.19 (true trace)
% S.RiseTime about 0.37 (10-90 %!)
```

Primer § 4.2 recipe, unchanged.

Read the last two lines carefully.

- `SettlingTime` is measured off the trace: 1.19 s, not the 1.33 s estimate.
- `RiseTime` is 10–90 %. The handbook's $T_r = 0.554$ s is 0–100 %. Both are right; they are different quantities.

Values are read off a sampled trace, so the last digit can differ.

Worked example W2 — designing from the specification

A robot joint must have $\%\,\mathrm{OS} \le 5\,\%$ and $T_s\,(2\,\%) \le 0.5\,\mathrm{s}$. Handbook § 4.2.7

Step 1 — overshoot fixes ζ only. $\zeta \ge -\ln 0.05/\sqrt{\pi^2 + \ln^2 0.05} = 0.690$. Take $\zeta = 0.7$ (4.60 %).

Step 2 — settling fixes $\zeta\omega_n$. $4/(0.7\,\omega_n) \le 0.5 \Rightarrow \omega_n \ge 11.43\,\mathrm{rad/s}$. Take $\omega_n = 12$.

Step 3 — the target closed loop.

$$T(s) = \frac{144}{s^2 + 16.8\,s + 144} \qquad s = -8.4 \pm j8.57$$

Step 4 — verify every specification. $T_p = 0.367\,\mathrm{s}$, $T_s = 4/(0.7 \times 12) = 0.476\,\mathrm{s}$ ✓. Simulated: 4.60 % overshoot, inside the 2 % band before 0.5 s ✓.

What is still missing

A controller that actually puts the poles at $-8.4 \pm j8.57$. That is the job of the root locus (§ 4.5) and of PBL Milestone 2.

2 % or 5 %? 0–100 % or 10–90 %? Say which

The same W1 servo, four legitimate "answers":

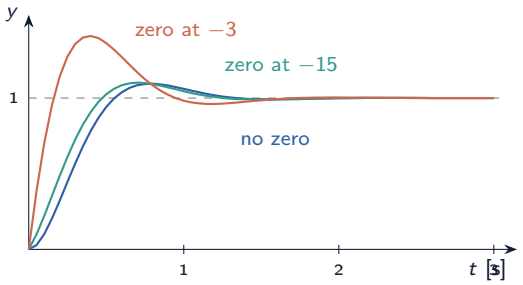
Quantity	Criterion	Estimate	True trace
T_s	2 % band	$4/(\zeta\omega_n) = 1.33\,\mathrm{s}$	1.19 s
T_s	5 % band	$3/(\zeta\omega_n) = 1.00\,\mathrm{s}$	1.05 s
T_r	0–100 %	$(\pi - \beta)/\omega_d = 0.554\,\mathrm{s}$	0.554 s
T_r	10–90 %	—	0.37 s

Write the criterion in the first line of your answer

A settling time with no band is not an answer. The handbook default is 2 %.

Note the 5 % estimate is *optimistic* here: an estimate is not a bound.

When the formulas lie: a closed-loop zero



Same poles as W1 ($\zeta\omega_n = 3$), same DC gain, one zero at $-z$:

$$T(s) = \frac{(25/z)(s+z)}{s^2 + 6s + 25}$$

Zero	$a = z/\zeta\omega_n$	%OS
none	—	9.48
-15	5	10.2
-3	1	41.0

The formula still says 9.48 % for all three.

Check for a zero before you use the canonical formulas

With *a* below about 2 the zero dominates the rise. Simulate, or say the formula does not apply.

Steady-state error: does it arrive?

A PV solar tracker is told to point at the sun. The transient dies away. Is it pointing where it was told? Handbook § 4.3.1

Unity feedback, forward path $G(s)$:

$$E(s) = R(s) - Y(s) = \frac{R(s)}{1 + G(s)}$$

The final-value theorem turns "as $t \to \infty$" into "as $s \to 0$":

$$\boxed{e_{ss} = \lim_{t \to \infty} e(t) = \lim_{s \to 0} \frac{s\,R(s)}{1 + G(s)}}$$

Only if the closed loop is stable

An unstable loop has no final value — but the limit still returns a tidy number. Check stability *first*.

K_p, K_v and K_a fall out of one limit

Put each test input into $e_{ss} = \lim_{s\to 0} sR(s)/[1 + G(s)]$: Handbook § 4.3.2

Step, $R = 1/s$:

$$e_{ss} = \lim_{s\to 0} \frac{1}{1 + G(s)} = \frac{1}{1 + K_p} \qquad K_p = \lim_{s\to 0} G(s)$$

Ramp, $R = 1/s^2$: one s is left over, so move it into the denominator:

$$e_{ss} = \lim_{s\to 0} \frac{1}{s + sG(s)} = \frac{1}{K_v} \qquad K_v = \lim_{s\to 0} sG(s)$$

Parabola, $R = 1/s^3$:

$$e_{ss} = \lim_{s\to 0} \frac{1}{s^2 + s^2 G(s)} = \frac{1}{K_a} \qquad K_a = \lim_{s\to 0} s^2 G(s)$$

System type: count the integrators

Write the open loop with its poles at the origin pulled out: Handbook § 4.3.3

$$G(s) = \frac{K(\tau_a s + 1)\cdots}{s^N(\tau_1 s + 1)\cdots} \qquad N = \text{system type}$$

Type	Step	Ramp	Parabola
0	$1/(1+K_p)$	∞	∞
1	0	$1/K_v$	∞
2	0	0	$1/K_a$

Why the staircase

$s^k G(s)$ stays finite at $s = 0$ only while $k \leq N$. Each integrator buys zero error against one more order of input — and costs stability margin, as Chapter 5 will show.

Worked examples — type 0 and type 1

Solar tracker. $G(s) = \dfrac{10}{(s+2)(s+5)}$

Step 1 — type. No pole at 0: type 0.

Step 2 — constant. $K_p = \dfrac{10}{2 \times 5} = 1.0$

Step 3 — error. $e_{ss} = \dfrac{1}{1+1} = 0.5$ — a 50 % offset.

Check. Closed loop $10/(s^2+7s+20)$ settles at 0.5 in simulation ✓

Motor position loop. $G(s) = \dfrac{KK_m}{s(\tau_m s+1)}$, $K = 4$, $K_m = 5$, $\tau_m = 0.05\,\text{s}$

Step 1 — type. One integrator: type 1, $e_{ss}^{\text{step}} = 0$.

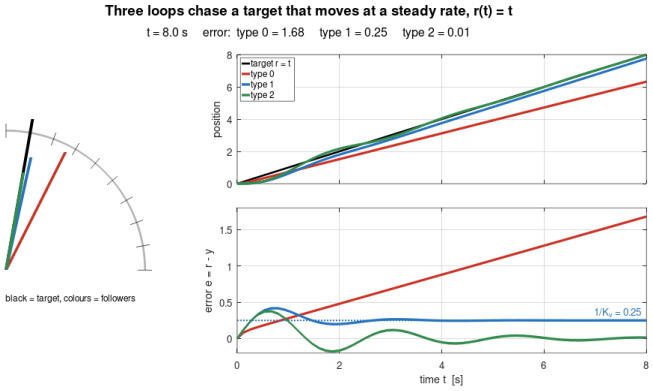
Step 2 — constant. $K_v = KK_m = 20\ \text{s}^{-1}$

Step 3 — error. $1/K_v = 0.05$ rad per rad/s of ramp ($2.86°$).

Check. Ramp into $400/(s^2+20s+400)$: error settles at 0.05 ✓

Handbook § 4.3.4–4.3.5

Watch three loops chase a ramp



A target moving at a steady rate, $r(t) = t$ — the sun for a tracker, a walking face for a camera. Gain 4 each, with 0, 1 and 2 integrators. Type 0 falls further behind for ever; type 1 settles a fixed distance behind; type 2 catches up. GIF: `ch4-anim/MEC524-ch4-ramp.gif`.

The chase, read off the error table

Unity feedback, ramp $r(t) = t$. Handbook § 4.3.2–4.3.3

Type	$G(s)$	**Constant**	e_{ss}, ramp	**At** $t = 8\,\mathbf{s}$
0	$\dfrac{4}{0.5s+1}$	$K_v = 0$	∞: grows at $1/(1+K_p) = 0.2$ per second	1.68
1	$\dfrac{4}{s(0.5s+1)}$	$K_v = \lim_{s\to 0} sG = 4$	$1/K_v = 0.25$	0.25
2	$\dfrac{4(s+1)}{s^2(0.5s+1)}$	$K_v = \infty$	0	0.013

Check stability first, as always

Closed loops: $8/(s+10)$; $8/(s^2+2s+8)$; $8(s+1)/(s^3+2s^2+8s+8)$. Routh on the last: $(2\cdot 8 - 8)/2 = 4 > 0$ — all three stable, so the final-value theorem may be used.

Two ways a correct formula gives a wrong e_{ss}

1. The final-value theorem on an unstable loop. A telescope tracker, Handbook § 4.3.6

$$G(s) = \frac{K(0.1s+1)}{s^2(s+4)} \qquad \text{type 2}, \quad K_a = \frac{K}{4}, \quad e_{ss} = \frac{4}{K}$$

"$K = 4000$ gives 0.001 rad." But the characteristic equation is $s^3 + 4s^2 + 0.1Ks + K = 0$, and its Routh s^1 entry is $(4 \times 0.1K - K)/4 = -0.15K < 0$ for every $K > 0$. At $K = 4000$ there are 2 right-half-plane poles. *There is no steady state to have an error.*

2. Counting the type on the wrong transfer function. Type is counted on the *open loop* $G(s)H(s)$ with unity feedback. The closed loop $T = G/(1+G)$ never has a pole at the origin — count there and every system is type 0.

Two more collisions of notation

The error constant K_p is not the proportional gain K_p. And § 4.3.6 in the handbook finishes this example: the Routh array, and a zero at -1 that makes the loop stable for every K.

Stability: will it stay put?

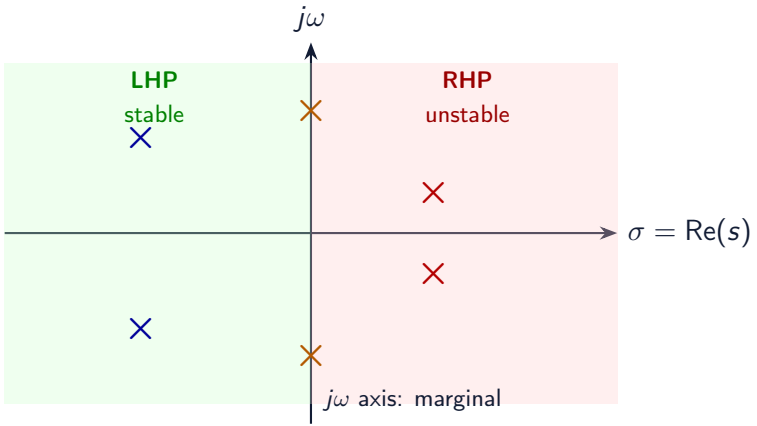
Turn up the gain on a servo for a faster response. At some gain it starts to hum, then shake itself apart. Where is that gain? Handbook § 4.4.1

Every natural mode is e^{pt} for a pole $p = \sigma + j\omega$, and $|e^{pt}| = e^{\sigma t}$. The sign of σ decides everything:

Poles	Mode	Verdict
all with $\sigma < 0$	decays	asymptotically stable
simple poles on $j\omega$, none right	sustains	marginally stable
any $\sigma > 0$, or repeated on $j\omega$	grows	unstable

BIBO stable $\iff$ every pole strictly in the left half-plane.

The s-plane stability map



Building the Routh array

$$D(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_0$$

Rows 1–2: the coefficients, alternately.

Every later entry: a 2×2 cross-multiplication of the two rows above, divided by the pivot:

$$b_1 = \frac{a_{n-1} a_{n-2} - a_n a_{n-3}}{a_{n-1}}$$

Before building it: every coefficient present and of one sign. A missing or negative coefficient means unstable already.

Try it: $s^3 + 4s^2 + 5s + 2$

s^3	1	5
s^2	4	2
s^1	$\frac{4 \cdot 5 - 1 \cdot 2}{4} = 4.5$	0
s^0	2	

First column $1, 4, 4.5, 2$: no sign change, **stable**.

Roots, for the record: $-1, -1, -2$.

Handbook § 4.4.4–4.4.5

Why a sign change counts a right-half-plane root

Routh–Hurwitz

The number of sign changes in the first column equals the number of roots in the right half-plane.

Why it has to be a sign change. Roots move continuously with the coefficients. To reach the right half-plane a root must *cross* the $j\omega$ axis — and at that instant a Routh row vanishes, so a first-column entry passes through zero and flips.

Watch it happen on $s^3 + 5s^2 + 6s + K$, s^1 entry $(30 - K)/5$:

K	First column	Sign changes	numpy: RHP roots
10	1, 5, 4, 10	0	0
31	1, 5, -0.2, 31	2	2

Handbook § 4.4.4

Worked example — the stable range of K

Find the $K > 0$ that keep $s^3 + 5s^2 + 6s + K = 0$ stable. Handbook § 4.4.6

s^3	1	6
s^2	5	K
s^1	$\frac{30 - K}{5}$	0
s^0	K	

Step 1 — every first-column entry positive.

$$\frac{30 - K}{5} > 0 \text{ and } K > 0 \quad \Longrightarrow \quad \boxed{0 < K < 30}$$

Step 2 — what happens at the edge. At $K = 30$ the s^1 row vanishes. The row above gives the auxiliary equation

$$5s^2 + 30 = 0 \;\Rightarrow\; s = \pm j\sqrt{6} = \pm j2.449$$

— it oscillates at 2.449 rad/s.

Check with the actual roots

numpy: stable at $K = 29.9$, unstable at $K = 30.1$. K_{cr} and ω_{cr} are the hand-off to root locus.

Special case 1: a zero in the first column

$$D(s) = s^4 + 2s^3 + s^2 + 2s + 1$$

Handbook § 4.4.7

s^4	1	1	1
s^3	2	2	
s^2	$0 \to \varepsilon$	1	
s^1	$\dfrac{2\varepsilon - 2}{\varepsilon} \to -\infty$		
s^0	1		

Step 1. The pivot is zero but the row is not. Replace it by a small positive ε and carry on.

Step 2. Let $\varepsilon \to 0^+$. Column: $+,\ +,\ +,\ -,\ +$.

Step 3. Two sign changes: **two RHP roots**. numpy agrees.

A zero pivot is not "marginally stable"

Stopping at the zero and writing "marginal" is the commonest wrong answer. Finish the array.

Special case 2: a whole row of zeros

$$D(s) = s^5 + 3s^4 + 5s^3 + 15s^2 + 4s + 12$$

Handbook § 4.4.8

s^5	1	5	4
s^4	3	15	12
s^3	12	30	
s^2	7.5	12	
s^1	10.8		
s^0	12		

Step 1. The s^3 row is all zero. Take the row above as the auxiliary polynomial: $A(s) = 3s^4 + 15s^2 + 12$.

Step 2. Replace the zero row by $A'(s) = 12s^3 + 30s$.

Step 3. Finish: 7.5, 10.8, 12. No sign changes — no RHP roots.

Step 4. But $A(s) = 3(s^2 + 1)(s^2 + 4)$: roots $\pm j$, $\pm j2$ sit *on* the axis.

A row of zeros means symmetric roots

No sign change here means **marginally** stable, not stable. Solve $A(s) = 0$ before you write the verdict.

Root locus: the angle and magnitude conditions

Routh says *whether* the loop is stable at a given K. The root locus shows *where* the closed-loop poles are for every K from 0 to ∞. Handbook § 4.5.2

$$1 + KG(s) = 0 \quad \Longleftrightarrow \quad KG(s) = -1$$

A complex number equals -1 only if its size is 1 and its angle is an odd multiple of 180°:

$$|KG(s)| = 1 \qquad \angle G(s) = \sum_i \angle(s - z_i) - \sum_j \angle(s - p_j) = (2k+1)\,180^\circ$$

Angle condition

Draws the locus. K does not appear in it at all.

Magnitude condition

Calibrates it: $K = 1/|G(s_0)|$ at a point already on the locus.

Sketching rules 1–4, and why they hold

1. Branches. One per open-loop pole: the characteristic polynomial has degree n.

2. Start and finish. Clear fractions: $D(s) + KN(s) = 0$. At $K = 0$ the roots are the poles; as $K \to \infty$ they are the zeros — or infinity, for the $n - m$ that have no zero to go to.

3. Real-axis segments. Test a real point. A complex pair's two angles cancel. A real pole or zero to its left contributes $0°$, to its right $180°$. So the point is on the locus when the count to its *right* is odd.

4. Asymptotes. From far away every pole and zero looks like one point, the centroid:

$$\theta_a = \frac{(2k+1)\,180°}{n-m} \qquad \sigma_a = \frac{\sum p_j - \sum z_i}{n-m}$$

Handbook § 4.5.3

Sketching rules 5–7, and why they hold

5. Breakaway. Along a real segment between two poles, $K = -1/G(s)$ rises from 0, peaks, and then no real root remains — the pair leaves the axis at the peak:

$$\frac{dK}{ds} = 0$$

Keep only roots that lie on a locus segment. Handbook § 4.5.3

6. $j\omega$ **crossing.** The gain that makes a Routh row vanish is K_{cr}; the auxiliary equation from the row above gives ω_{cr}.

7. Symmetry. Real coefficients give conjugate roots: sketch the top half, mirror it.

Rules sketch the shape; conditions give the numbers

Any value you quote — a design point, a gain — comes from the angle and magnitude conditions, not from where a sketch happens to cross a line.

Worked example — three poles (1 of 3)

Sketch the locus of $\dfrac{K}{s(s+2)(s+4)}$. Handbook § 4.5.5

Step 1 — poles and zeros. Poles $0,\ -2,\ -4$. No zeros. $n - m = 3$ branches go to infinity.

Step 2 — real axis, counting from the right.

Segment	Poles to the right	On the locus?
$-2 < s < 0$	1	yes
$-4 < s < -2$	2	no
$s < -4$	3	yes

Step 3 — asymptotes. $\theta_a = 60^\circ,\ 180^\circ,\ 300^\circ$ from $\sigma_a = (0 - 2 - 4)/3 = -2$.

Worked example — three poles (2 of 3)

Step 4 — breakaway. On $-2 < s < 0$, $K = -(s^3 + 6s^2 + 8s)$:

$$\frac{dK}{ds} = -(3s^2 + 12s + 8) = 0 \;\Rightarrow\; s = -2 \pm 1.155 = -0.845 \text{ or } -3.155$$

-3.155 lies on $-4 < s < -2$, which is not on the locus: reject it. The pair leaves the axis at $s = -0.845$, where $K = 3.08$. Handbook § 4.5.5

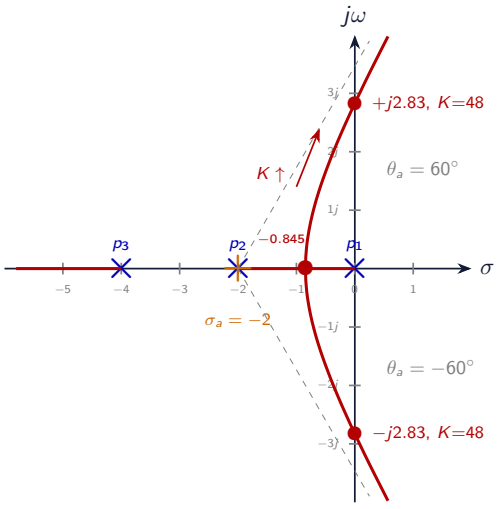
Step 5 — $j\omega$ crossing. Routh on $s^3 + 6s^2 + 8s + K$: s^1 entry $(48 - K)/6$, so

$$K_{cr} = 48 \qquad 6s^2 + 48 = 0 \Rightarrow s = \pm j2.83$$

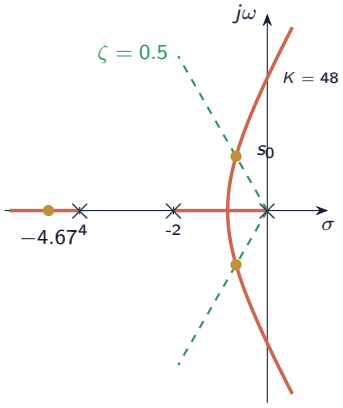
Two cross-checks, two different routes

A dense sweep of K finds the real pair merging at $K = 3.08$, $s = -0.845$. numpy is stable at $K = 47.9$ and unstable at $K = 48.1$.

The locus of K/[s(s+2)(s+4)], computed



Worked example — three poles (3 of 3): K for $\zeta = 0.5$



Step 6 — the angle condition places the point. On the ray $\beta = \cos^{-1} 0.5 = 60°$, search until

$$120° + 40.9° + 19.1° = 180°$$

at $s_0 = -0.667 + j1.155$. Handbook § 4.5.6

Step 7 — the magnitude condition gives K.

$$\begin{aligned} K &= |s_0|\,|s_0 + 2|\,|s_0 + 4| \\ &= (1.333)(1.764)(3.528) = 8.30 \end{aligned}$$

Step 8 — the pole you forgot. Third closed-loop pole at -4.67.

A point read off a sketch, $-1 + j1.73$, sums to $210°$: not on the locus. Its "$K \approx 14$" gives $\zeta \approx 0.31$.

Is the pair really dominant? Ask the third pole

At $K = 8.30$ the poles are $-0.667 \pm j1.155$ and -4.67: the third one is 7 times further from the axis. Handbook § 4.5.6

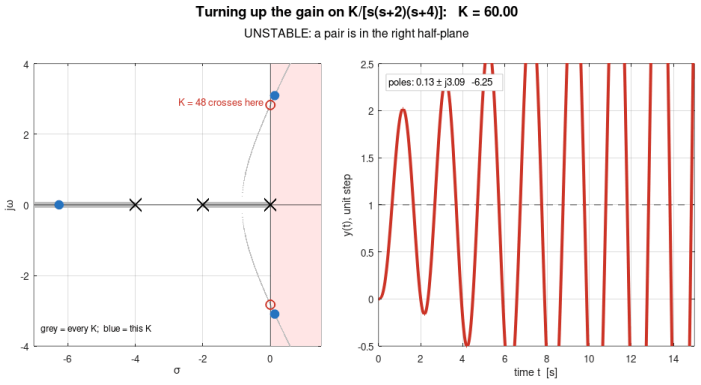
	Second-order formulas	RK4, all three poles
% OS	16.3 %	15.5 %
T_p	$\pi/1.155 = 2.72\,\text{s}$	2.96 s
T_s (2 %)	$4/0.667 = 6.0\,\text{s}$	6.24 s

The rule of thumb, and its limit

A pole at least about five times further left barely matters — the estimates above are within 10 %. Closer than that, or with a zero nearby, simulate.

The same caution as the closed-loop zero in § 4.2: formulas describe two poles and nothing else.

Watch the poles move as you turn up K



The worked example $K/[s(s+2)(s+4)]$, K from 0.3 to 60. Left: the three closed-loop poles riding the locus. Right: the step response at that K. The pair crosses the $j\omega$ axis at $K = 48$, exactly where Routh put it. GIF: `ch4-anim/MEC524-ch4-locus.gif`.

Five gains read off the animation

Roots of $s^3 + 6s^2 + 8s + K$. Handbook § 4.5.5–4.5.6

K	**Closed-loop poles**	**What the step does**
1	$-0.14,\ -1.75,\ -4.11$	creeps up, no overshoot
3.08	-0.845 (double), -4.31	breakaway: the pair meets; above this it rings
8.30	$-0.667 \pm j1.155,\ -4.67$	$\zeta = 0.5$: one overshoot, 15.5 %
48	$\pm j2.83,\ -6$	rings for ever, period $2\pi/2.83 = 2.22\,\mathrm{s}$
60	$+0.13 \pm j3.09,\ -6.25$	grows: unstable

One knob, every regime

Gain buys speed and costs damping. Root locus is the map of that trade; Routh tells you where the map ends.

Stable is not the same as good

The last *stable* frame of the animation is $K = 47.97$. Handbook § 4.4, 4.5

$$s^3 + 6s^2 + 8s + 47.97 = 0 \quad \Rightarrow \quad s = -0.000398 \pm j2.828,\ -6.00 \qquad \zeta = 0.00014$$

$$T_s\,(2\,\%) \approx \frac{4}{0.000398} \approx 10\,000 \text{ s} \quad \text{— nearly three hours of ringing.}$$

Answer the question that was asked

"Find the range of K for stability" $\Rightarrow 0 < K < 48$. "Choose K" $\Rightarrow$ a specification first (ζ, T_s), then the magnitude condition. Never quote a gain just inside K_{cr} as a design.

When gain alone cannot do it

A double integrator. Handbook §4.5.6a

$$KG(s) = \frac{K}{s^2(s+2)(s+5)}$$

Characteristic equation
$s^4 + 7s^3 + 10s^2 + 0 \cdot s + K = 0$.

The s^1 coefficient is *missing*: the necessary condition fails before any array. Unstable for every $K > 0$ (numpy: $K = 0.1$ to 100).

Asymptotes at $45°, 135°, 225°, 315°$ from -1.75 — two of them point straight into the right half-plane.

The fix is a zero.

Zeros attract branches; poles repel them. Add a compensator zero and a branch ends there instead of heading to infinity, and the asymptotes rearrange.

Example (tutorial Q11): adding a zero at -2 to $K/[s(s+1)(s+4)]$ leaves 2 asymptotes, at $\pm 90°$ from -1.5 — vertical, instead of bending into the right half-plane.

This is why PID has a D

A PD or lead term adds exactly that zero. PBL Milestone 2 is where you use it.

Checking the locus in MATLAB/Octave

```
G = tf(1, [1 6 8 0]); % s(s+2)(s+4), K pulled out
figure; rlocus(G); grid on
sgrid(0.5, []) % overlay the zeta = 0.5 ray

% Routh-derived K_cr, checked by sweep
Ks = linspace(1, 100, 5000);
unstable = false(size(Ks));
for k = 1:length(Ks)
    charpoly = [1 6 8 Ks(k)];
    unstable(k) = any(real(roots(charpoly)) >= 0);
end
K_max = Ks(find(unstable, 1, 'first') - 1)
% K_max = 47.99 (Routh: 48)

roots([1 6 8 8.30]) % -4.67, -0.67 +/- 1.15i
```

Primer § 7.1–7.2 recipes, unchanged.

Hand first, MATLAB second. Do the segments, centroid, breakaway and Routh by hand; then let the sweep confirm K_{cr} to better than 1 %.

Do not read the design gain off a mouse click on the plot and call it exact. The last line checks it: the complex pair sits on $\zeta = 0.5$.

Three ways to lose marks here

1. **A settling time with no band.**
   $4/(\zeta\omega_n)$ is 2 %, $3/(\zeta\omega_n)$ is 5 %, and `stepinfo`'s rise time is 10–90 %. Say which, in the first line.

2. **An e_{ss} for a loop that is not stable.**
   Routh first, final-value theorem second. And count the type on the open loop $G(s)H(s)$, never on $T(s)$.

3. **A gain read off a sketch.**
   Check that the angles sum to 180° at your point before you apply the magnitude condition. Reject breakaway roots that are not on a locus segment.

And one from § 4.2

The canonical % OS, T_p, T_s formulas describe two poles and no zero. Look for a zero and a third pole before you use them.

What the assessment will ask

Assessment	Week	Weight	What it asks from this chapter
Test 2	After Wk 14	30% CLO1+CLO2	Transient specifications; steady-state error from system type; Routh–Hurwitz ranges; root-locus sketch and gain.
Individual MATLAB Assignment (individual)	Wk 13	10% CLO3	Simulated specifications checked against your hand-derived ones.
PBL Milestone 2	Wk 14	10% CLO3	These specifications and stability ranges are the targets your PID design must meet.

What a full-mark answer looks like

Settling criterion stated; stability shown *before* any e_{ss}; the whole Routh array with its first column read; a design point that satisfies the angle condition, then its K.

What to take away

1. Pole position *is* performance: $\zeta\omega_n$ sets T_s, ω_d sets T_p, ζ alone sets the overshoot.
2. The formulas are for two poles and no zero. Otherwise, simulate.
3. Steady-state error: stability first, then the type of $G(s)H(s)$, then K_p, K_v or K_a.
4. Routh counts right-half-plane roots by sign changes; a vanishing row gives K_{cr} and ω_{cr}.
5. Root locus: the angle condition places the point, the magnitude condition prices it in K.

Handbook § 4.0–4.5. Worked examples: § 4.1–4.5 and the solutions in § 4.6.

Tutorial: § 4.6 — attempt Q1–Q11 before the tutorial class.