



Industrial Control Systems

Chapter Six:

Dynamic Behavior and Stability of Closed-Loop Control Systems

Dr. Eng. Baha'eddin Alhaj Hasan
Department of Industrial Engineering

Effect of Zeros on the Transient Response

Reminder: for $H(s) = \frac{q(s)}{p(s)}$, zeros are the roots of $q(s) = 0$

Example: start with $H_1(s) = \frac{1}{s^2 + 2\zeta s + 1}$ ($\omega_n = 1$)

Let's add a zero at $s = -a$, $a > 0$ – LHP zero

To keep DC gain = 1, let's take the numerator to be $\frac{s}{a} + 1$:

$$\begin{aligned} H_2(s) &= \frac{\frac{s}{a} + 1}{s^2 + 2\zeta s + 1} \\ &= \underbrace{\frac{1}{s^2 + 2\zeta s + 1}}_{\text{this is } H_1(s)} + \frac{1}{a} \cdot \underbrace{\frac{s}{s^2 + 2\zeta s + 1}}_{\text{call this } H_d(s)} \\ &= H_1(s) + \frac{1}{a} H_d(s), \quad H_d(s) = sH_1(s) \end{aligned}$$

Effect of a LHP Zero

$$H_1(s) = \frac{1}{s^2 + 2\zeta s + 1} \xrightarrow{\text{add zero at } s = -a} H_2(s) = H_1(s) + \frac{1}{a} \cdot sH_1(s)$$

Step response:

$$\begin{aligned} Y_1(s) &= \frac{H_1(s)}{s} \\ Y_2(s) &= \frac{H_2(s)}{s} \\ &= \frac{H_1(s)}{s} + \frac{1}{a} \frac{sH_1(s)}{s} \\ &= Y_1(s) + \frac{1}{a} sY_1(s) \end{aligned}$$

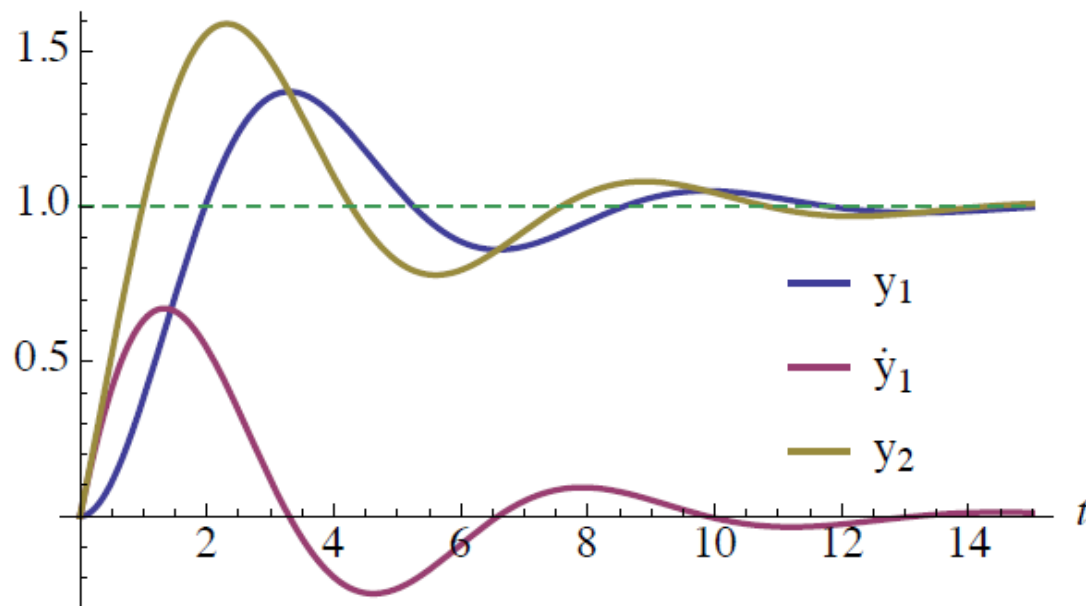
$$y_2(t) = \mathcal{L}^{-1}\{Y_2(s)\} = \mathcal{L}^{-1}\left\{Y_1(s) + \frac{1}{a} \cdot sY_1(s)\right\} = y_1(t) + \frac{1}{a} \dot{y}_1(t)$$

(assuming zero initial conditions)

Effect of a LHP Zero

Step response (zero at $s = -a$)

$$y_2(t) = y_1(t) + \frac{1}{a} \dot{y}_1(t) \quad \text{where } y_1(t) = \text{original step response}$$

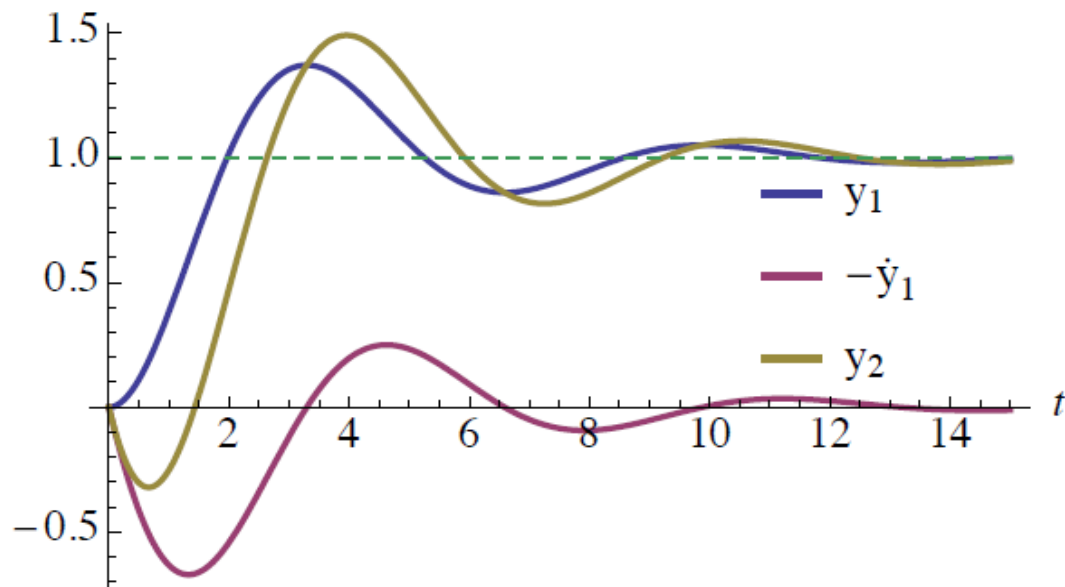


Effects of a LHP zero:

- ▶ increased overshoot (major effect)
- ▶ little influence on settling time
- ▶ what happens as $a \rightarrow \infty$? — effects become less significant

What About a RHP Zero?

$$H_1(s) = \frac{1}{s^2 + 2\zeta s + 1} \xrightarrow{\text{add zero at } s = a} H_2(s) = H_1(s) - \frac{1}{a} \cdot sH_1(s)$$
$$y_2(t) = y_1(t) - \frac{1}{a} \cdot \dot{y}_1(t)$$

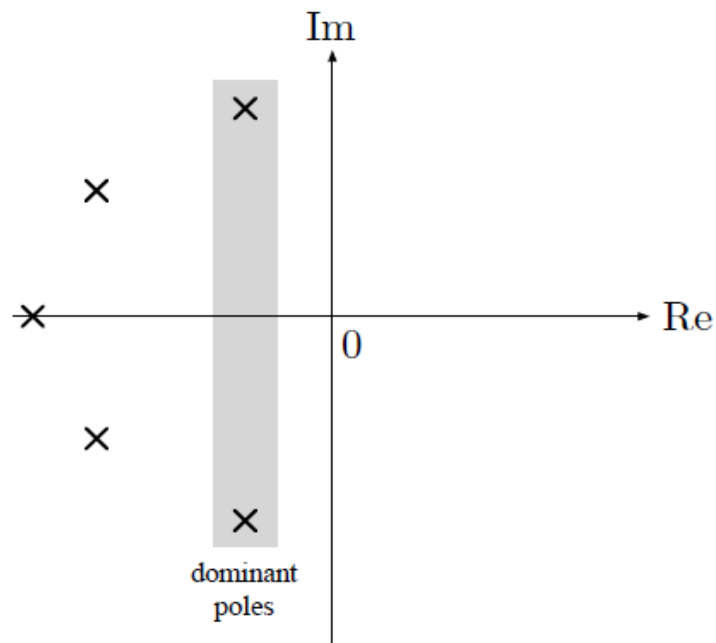


Effects of a RHP zero:

- ▶ slows down (delays) the response
- ▶ creates *undershoot* (at least, when a is small enough)

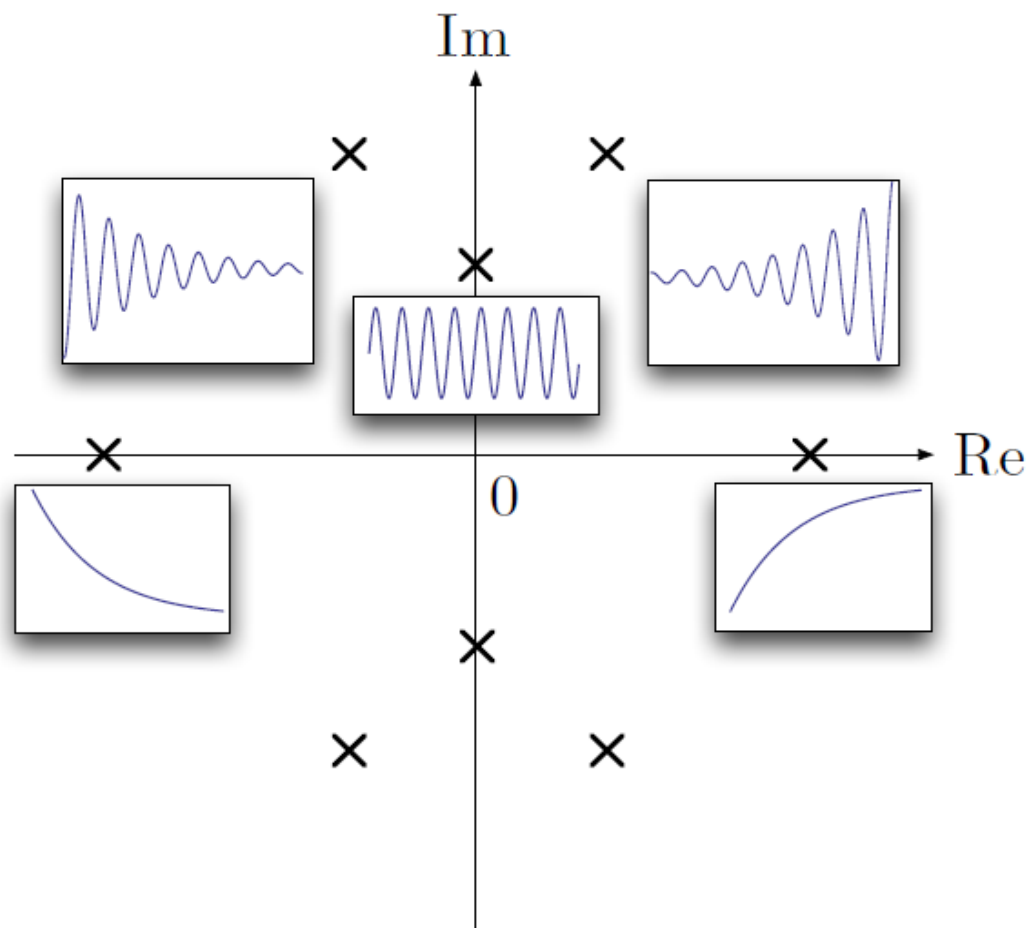
Effect of Extra Poles

A general n th-order system has n poles



- ▶ extra LHP poles are not significant if their real parts are at least $5\times$ the real parts of dominant LHP poles
- ▶ e.g., if dominant poles have $\text{Re}(s) = -2$ and we have extra poles with $\text{Re}(s) = -10$, their time-domain contributions will be e^{-2t} and $e^{-10t} \ll e^{-2t}$
- ▶ $5\times$ is just a convention, but we can really see the effect of extra poles that are closer (cf. Lab 2)

Effect of Pole Locations



- ▶ poles in open LHP ($\text{Re}(s) < 0$) — stable response
- ▶ poles in open RHP ($\text{Re}(s) > 0$) — unstable response
- ▶ poles on the imaginary axis ($\text{Re}(s) = 0$) — tricky case

Marginal Case: Poles on the Imaginary Axis

Let's consider the case of a pole at the origin: $H(s) = \frac{1}{s}$

Is this a stable system?

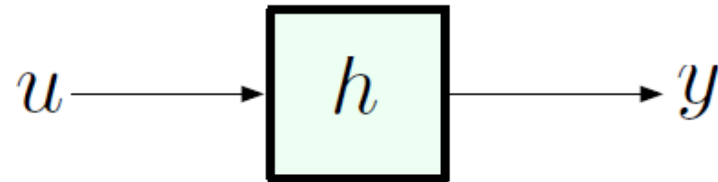
- ▶ impulse response: $Y(s) = \frac{1}{s} \implies y(t) = 1(t)$ (OK)
- ▶ step response: $Y(s) = \frac{1}{s^2} \implies y(t) = t, t \geq 0$ — *unit ramp!!*

What about purely imaginary poles? $H(s) = \frac{\omega^2}{s^2 + \omega^2}$

- ▶ impulse response: $Y(s) = \frac{\omega^2}{s^2 + \omega^2} \implies y(t) = \omega \sin(\omega t)$
- ▶ step response: $Y(s) = \frac{\omega^2}{s(s^2 + \omega^2)} \implies y(t) = 1 - \cos(\omega t)$

Systems with poles on the imaginary axis are *not stable*.

What Is Stability?



One reasonable definition is as follows:

A linear time-invariant system is *Bounded-Input, Bounded-Output (BIBO) stable* provided either one of the following three equivalent conditions is satisfied:

1. If every bounded input $u(t)$ results in a bounded output $y(t)$, regardless of initial conditions.
2. If the impulse response $h(t)$ is absolutely integrable:

$$\int_{-\infty}^{\infty} |h(t)| dt < \infty.$$

3. If all poles of the transfer function $H(s)$ are *strictly stable* (lie in open LHP).

Checking for Stability?

Consider a general transfer function:

$$H(s) = \frac{q(s)}{p(s)}$$

where q and p are polynomials, and $\deg(q) \leq \deg(p)$.

We need tools for checking stability: whether or not all roots of $p(s) = 0$ lie in OLHP.

For simple polynomials, can just factor them “by inspection” and find roots.

Now, this is hard to do for high-degree polynomials — it’s computationally intensive, especially symbolically.

But: often we *don’t need to know* precise pole locations, just need to know that they are **strictly stable**.

Checking for Stability

Problem: given an n th-degree polynomial

$$p(s) = s^n + a_1s^{n-1} + a_2s^{n-2} + \dots + a_{n-1}s + a_n$$

with real coefficients, check that the roots of the equation $p(s) = 0$ are strictly stable (i.e., have negative real parts).

Terminology: we often say that the polynomial p is (strictly) stable if all of its roots are.

A Necessary Condition for Stability

Terminology: we say that A is a necessary condition for B if

$$A \text{ is false} \implies B \text{ is false}$$

Important!! Even if A is true, B may still be false.

Necessary condition for stability: a polynomial p is strictly stable only if all of its coefficients are strictly positive.

Proof: suppose that p has roots at r_1, r_2, \dots, r_n with $\operatorname{Re}(r_i) < 0$ for all i . Then

$$p(s) = (s - r_1)(s - r_2) \dots (s - r_n)$$

— multiply this out and check that all coefficients are positive.

Routh–Hurwitz Criterion

Necessary & Sufficient Condition for Stability

Terminology: we say that A is a sufficient condition for B if

$$A \text{ is true} \implies B \text{ is true}$$

Thus, A is a necessary *and* sufficient condition for B if

$$A \text{ is true} \iff B \text{ is true}$$

— we also say that A is true if and only if (iff) B is true.

We will now introduce a necessary and sufficient condition for stability: the *Routh–Hurwitz Criterion*.

Routh's Test

Problem: check whether the polynomial

$$p(s) = s^n + a_1 s^{n-1} + a_2 s^{n-2} + \dots + a_{n-1} s + a_n$$

is strictly stable.

We begin by forming the **Routh array** using the coefficients of p :

$$\begin{array}{l} s^n : \quad 1 \quad a_2 \quad a_4 \quad a_6 \quad \dots \quad \text{(if necessary, add zeros in the} \\ s^{n-1} : \quad a_1 \quad a_3 \quad a_5 \quad a_7 \quad \dots \quad \text{second row to match lengths)} \end{array}$$

Note that the very first entry is always 1, and also note the order in which the coefficients are filled in.

Routh's Test

$$\begin{array}{l} s^n : \quad 1 \quad a_2 \quad a_4 \quad a_6 \quad \dots \\ s^{n-1} : \quad a_1 \quad a_3 \quad a_5 \quad a_7 \quad \dots \\ s^{n-2} : \quad b_1 \quad b_2 \quad b_3 \quad \dots \end{array}$$

Next, we form the third row marked by s^{n-2} :

$$s^{n-2} : \quad b_1 \quad b_2 \quad b_3 \quad \dots$$

$$\text{where } b_1 = -\frac{1}{a_1} \det \begin{pmatrix} 1 & a_2 \\ a_1 & a_3 \end{pmatrix} = -\frac{1}{a_1} (a_3 - a_1 a_2)$$

$$b_2 = -\frac{1}{a_1} \det \begin{pmatrix} 1 & a_4 \\ a_1 & a_5 \end{pmatrix} = -\frac{1}{a_1} (a_5 - a_1 a_4)$$

$$b_3 = -\frac{1}{a_1} \det \begin{pmatrix} 1 & a_6 \\ a_1 & a_7 \end{pmatrix} = -\frac{1}{a_1} (a_7 - a_1 a_6) \quad \text{and so on ...}$$

Note: the new row is 1 element shorter than the one above it

Routh's Test, continued

$$\begin{array}{l} s^n : \quad 1 \quad a_2 \quad a_4 \quad a_6 \quad \dots \\ s^{n-1} : \quad a_1 \quad a_3 \quad a_5 \quad a_7 \quad \dots \\ s^{n-2} : \quad b_1 \quad b_2 \quad b_3 \quad \dots \\ s^{n-3} : \quad c_1 \quad c_2 \quad \dots \end{array}$$

Next, we form the fourth row marked by s^{n-3} :

$$s^{n-3} : \quad c_1 \quad c_2 \quad \dots$$

$$\text{where } c_1 = -\frac{1}{b_1} \det \begin{pmatrix} a_1 & a_3 \\ b_1 & b_2 \end{pmatrix} = -\frac{1}{b_1} (a_1 b_2 - a_3 b_1)$$

$$c_2 = -\frac{1}{b_1} \det \begin{pmatrix} a_1 & a_5 \\ b_1 & b_3 \end{pmatrix} = -\frac{1}{b_1} (a_1 b_3 - a_5 b_1)$$

and so on ...

Routh's Test, continued

Eventually, we complete the array like this:

$$\begin{array}{l} s^n : \quad 1 \quad a_2 \quad a_4 \quad a_6 \quad \dots \\ s^{n-1} : \quad a_1 \quad a_3 \quad a_5 \quad a_7 \quad \dots \\ s^{n-2} : \quad b_1 \quad b_2 \quad b_3 \quad \dots \\ s^{n-3} : \quad c_1 \quad c_2 \quad \dots \\ \vdots \\ s^1 : \quad * \quad * \\ s^0 : \quad * \end{array} \quad \begin{array}{l} \\ \\ \\ \text{(as long as we don't get stuck with} \\ \text{division by zero: more on this later)} \end{array}$$

After the process terminates, we will have $n + 1$ entries in the first column.

The Routh–Hurwitz Criterion

Consider degree- n polynomial

$$p(s) = s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n$$

and form the Routh array:

$$\begin{array}{l} s^n : \quad 1 \quad a_2 \quad a_4 \quad a_6 \quad \dots \\ s^{n-1} : \quad a_1 \quad a_3 \quad a_5 \quad a_7 \quad \dots \\ s^{n-2} : \quad b_1 \quad b_2 \quad b_3 \quad \dots \\ s^{n-3} : \quad c_1 \quad c_2 \quad \dots \\ \vdots \\ s^1 : \quad * \quad * \\ s^0 : \quad * \end{array}$$

The Routh–Hurwitz criterion: Assume that the necessary condition for stability holds, i.e., $a_1, \dots, a_n > 0$. Then:

- ▶ p is stable if and only if all entries in the first column are positive;
- ▶ otherwise, $\#(\text{RHP poles}) = \#(\text{sign changes in 1st column})$

Example

Check stability of

$$p(s) = s^4 + 4s^3 + s^2 + 2s + 3$$

All coefficients strictly positive: necessary condition checks out.

$$\begin{array}{l} s^4 : \quad 1 \quad 1 \quad 3 \\ s^3 : \quad 4 \quad 2 \quad 0 \\ s^2 : \quad 1/2 \quad 3 \\ s^1 : \quad -22 \quad 0 \\ s^0 : \quad 3 \end{array}$$

Answer: p is unstable — it has 2 RHP poles (2 sign changes in 1st column)

Low-Order Cases ($n = 2, 3$)

$$n = 2 \quad p(s) = s^2 + a_1s + a_2$$

$$s^2 \quad : 1 \quad a_2$$

$$s^1 \quad : a_1 \quad 0$$

$$s^0 : \quad b_1$$

$$b_1 = -\frac{1}{a_1} \det \begin{pmatrix} 1 & a_2 \\ a_1 & 0 \end{pmatrix} = a_2$$

— p is stable iff $a_1, a_2 > 0$ (necessary *and* sufficient).

$$n = 3 \quad p(s) = s^3 + a_1s^2 + a_2s + a_3$$

$$s^3 \quad : 1 \quad a_2$$

$$s^2 \quad : a_1 \quad a_3$$

$$s^1 : \quad b_1 \quad 0$$

$$s^0 : \quad c_1$$

$$b_1 = -\frac{1}{a_1} \det \begin{pmatrix} 1 & a_2 \\ a_1 & a_3 \end{pmatrix} = \frac{a_1a_2 - a_3}{a_1}$$

$$c_1 = -\frac{1}{b_2} \det \begin{pmatrix} a_1 & a_3 \\ b_1 & 0 \end{pmatrix} = a_3$$

— p is stable iff $a_1, a_2, a_3 > 0$ (necc. cond.) and $a_1a_2 > a_3$

Stability Conditions for Low-Order Polynomials

The upshot:

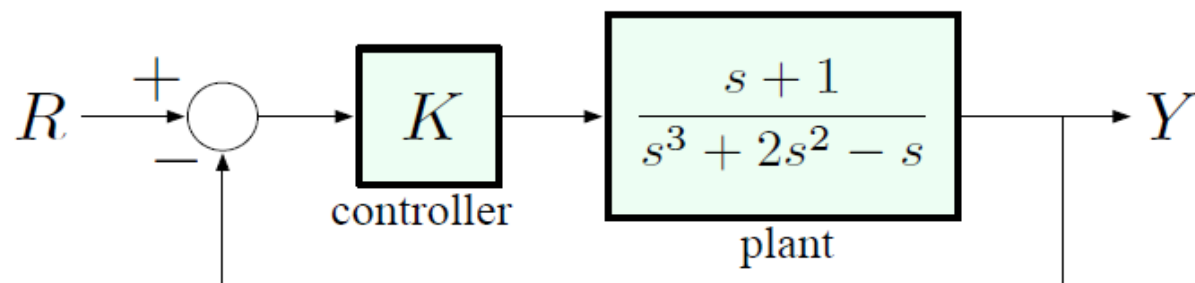
- ▶ A 2nd-degree polynomial $p(s) = s^2 + a_1s + a_2$ is stable if and only if $a_1 > 0$ and $a_2 > 0$
 - ▶ A 3rd-degree polynomial $p(s) = s^3 + a_1s^2 + a_2s + a_3$ is stable if and only if $a_1, a_2, a_3 > 0$ and $a_1a_2 > a_3$
-
- ▶ These conditions were already obtained by Maxwell in 1868.
 - ▶ In both cases, the computations were *purely symbolic*: this can make a lot of difference in *design*, as opposed to *analysis*.

Routh–Hurwitz as a Design Tool

Parametric stability range

We can use the Routh test to determine *parameter ranges* for stability.

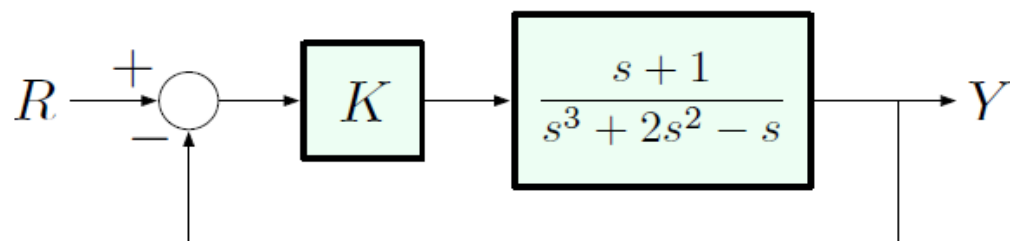
Example: consider the unity feedback configuration



Note that the plant is *unstable* (the denominator has a negative coefficient and a zero coefficient).

Problem: determine the range of values the scalar gain K can take, for which the closed-loop system is stable.

Example, continued



Problem: determine the range of values the scalar gain K can take, for which the closed-loop system is stable.

Let's write down the transfer function from R to Y :

$$\begin{aligned}\frac{Y}{R} &= \frac{\text{forward gain}}{1 + \text{loop gain}} \\ &= \frac{K \cdot \frac{s+1}{s^3+2s^2-s}}{1 + K \cdot \frac{s+1}{s^3+2s^2-s}} = \frac{K(s+1)}{s^3 + 2s^2 - s + K(s+1)} \\ &= \frac{Ks + K}{s^3 + 2s^2 + (K-1)s + K}\end{aligned}$$

Now we need to test stability of $p(s) = s^3 + 2s^2 + (K-1)s + K$.

Example, continued

Test stability of

$$p(s) = s^3 + 2s^2 + (K - 1)s + K$$

using the Routh test.

Form the Routh array:

$$\begin{array}{l} s^3 : \quad 1 \quad K - 1 \\ s^2 : \quad 2 \quad K \\ s^1 : \quad \frac{K}{2} - 1 \quad 0 \\ s^0 : \quad K \end{array}$$

For p to be stable, all entries in the 1st column must be positive:

$$K > 2 \quad \text{and} \quad K > 0 \quad (\text{already covered by } K > 1)$$

Note: The necessary condition requires $K > 1$, but now we actually know that we must have $K > 2$ for stability.

Some Comments on the Routh Test

- ▶ The result ($\#(\text{RHP roots})$) is not affected if we multiply or divide any row of the Routh array by an arbitrary *positive* number.
- ▶ If we get a zero element in the 1st column, we can't continue. In that case, we can replace the 0 by a small number ε and apply Routh test to that. When we are done with the array, take the limit as $\varepsilon \rightarrow 0$. (see Ex. 3.33 in FPE)
- ▶ For an *entire row of zeros*, the procedure is a more complicated (see Example 3.34 in FPE) – we will not worry about this too much.