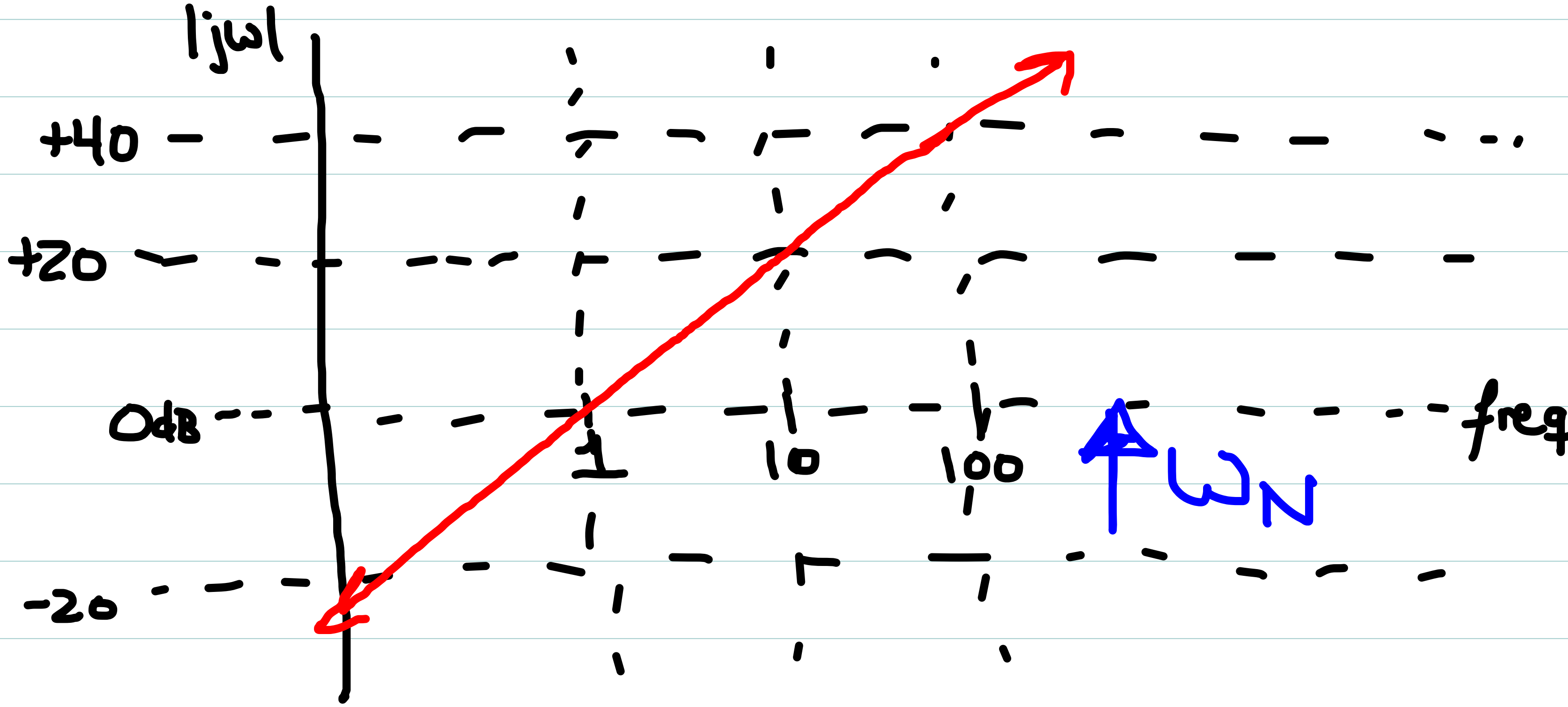


Differentiation amplifies the effect of noise

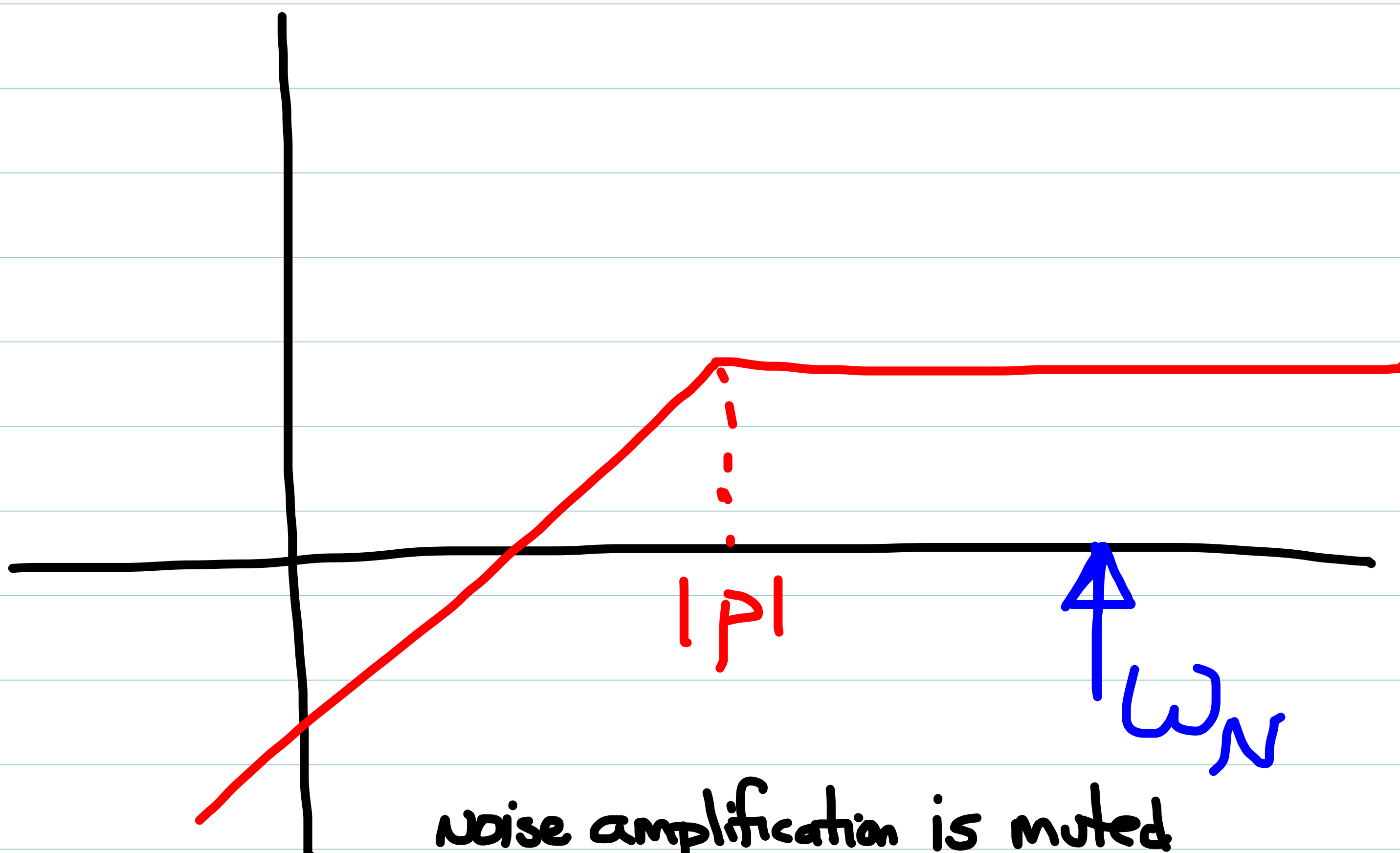
explicitly: if again $n(t) = \varepsilon \sin(\omega_N t)$, $\omega_N \gg 1$
then

$$z(t) = \frac{d}{dt}\left[y(t) + n(t)\right] = \dot{y}(t) + \underline{\varepsilon \omega_N} \cos(\omega_N t)$$

Not small!
(potentially larger than $\dot{y}$)

Note that if we added a pole to our derivative estimation scheme

$$Z(s) = \left[\frac{s}{s-p}\right] Y_m(s)$$



Noise amplification is muted and may be tolerable.

If we used this strategy to replace the derivative information needed for implementation an ideal zero:

$$H(s) = K(s-z) \Longrightarrow H(s) = K\left[\frac{s}{s-p} - z\right]$$

Then:

$$H(s) = K\left[\frac{(1-z)s + pz}{s-p}\right]$$

which is a <u>lead compensator</u> (for typical case $p<z$).

So really, a lead compensator is effectively a "practical" implementation of an ideal zero, which acknowledges the imperfect nature of the measurement process.

The most basic (and essential) task of the control engineer — achieving a stable closed-loop system with nominal performance characteristics — is straightforward to approach.

However, it is tricky to also incorporate and balance the competing constraints of

- Implementation constraints
- Tracking accuracy
- Disturbance rejection
- Noise rejection
- Model uncertainty
- Sensor/Actuator/Computation delays
- Actuator Limits/Control saturation
- Power/weight/cost demands

The "best" design is one which achieves an <u>acceptable</u> trade-off among these competing factors.

There is no "one true design" which makes the "ideal" tradeoff — so don't waste time looking for it!

Find something that works acceptably well, and <u>move on</u>

Major, common families of compensators

① $H(s) = K \Rightarrow u(t) = K e(t)$ "Proportional" control

② $H(s) = K_P + K_D s = K(s-z)$ $\left(K = K_D,\ z = -K_P/K_D\right)$

$\Rightarrow u(t) = K_P e(t) + K_D \dot{e}(t)$ "Prop. + Derivative (PD) control"

Note: implementable if both $y(t)$ and $\dot{y}(t)$ measured directly)

③ $H(s) = K_P + \frac{K_I}{s} = K\left[\frac{s-z}{s}\right]$ $\left(K = K_P,\ z = -K_I/K_P\right)$

$$\Rightarrow u(t) = K_P e(t) + K_I x_1(t)$$
$$\dot{x}_1(t) = e(t)$$

Equivalently: $u(t) = K_P e(t) + K_I \int_0^t e(\tau)\,d\tau$

"prop. + integral (PI) control")

4. $H(s) = K_P + K_D s + K_I/s = K\left[\frac{(s-z_1)(s-z_2)}{s}\right]$

$(K = K_D$; z_1, z_2 roots of $K_D s^2 + K_P s + K_I)$

$\Rightarrow u(t) = K_P e(t) + K_D \dot{e}(t) + K_I \int_0^t e(\tau)\,d\tau$

"Prop/Int/Deriv (PID) control"

Notes: a.) Very popular. Special purpose chips which do this computation are commonly available

b.) 1)-3) above are special cases of this more general form.

c.) Provides 2 zeros to help meet margin/xover requirements, and pole at origin to help with tracking/dist. rejection requirements.

d.) Like PD, requires direct measurement of $\dot{y}(t)$

(5) $H(s) = K\left[\frac{(s-z)}{s-p}\right]$, $|z|<|p|$

"Lead compensator"

Notes: a.) "Implementable" form of PD control when only $y(t)$ measured

b.) Using minimal values of $\beta = |p|/|z|$ helps with noise rejection and control saturation

(6.) $H(s) = K\left[\frac{(s-z_1)(s-z_2)}{s(s-p)}\right]$ $|p|>|z_1|,|z_2|$

"PI/Lead": implementable form of PID when only $y(t)$ measured

---

Of course, a designer is free to choose $H(s)$ as desired. These are common "go to" starting points which can be modified or added to as needed.

Alternate Design Perspectives

Our correlation between phase margin/crossover and the poles of $T(s)$ [hence its transient response characteristics] is approximate and tenuous at best.

It would be nice if we could specifically target the desired closed-loop poles, and design $H(s)$ to obtain them.

There are, in fact, techniques for this, although in using them we give up many of the insights afforded by the freq. response design methods . . .

(Everything is a trade-off! There are no magic bullets in this game!)

Recall the Characteristic Equation:

All closed-loop poles satisfy: $1 + L(s) = \emptyset$

$\Rightarrow L(s) = -1$

Let $L_o(s) = [L(s)]_{K=1}$ (K= compensator gain - real!)

Then s is a CL pole if: $K L_o(s) = -1$

which requires $L_o(s)$ to be real.

For any such s: $K = \dfrac{-1}{L_o(s)}$

is the gain which would make this s a CL pole

<u>Moreover:</u> $L_o(s)$ is real (hence s a possible CL pole)

if: $\measuredangle L_o(s) = \ell(180^\circ)$ (ℓ = any integer)

In particular:

$$L_o(s) = (1+2\ell)180^\circ$$ (odd multiple of 180°)

$\Rightarrow$ $L_o(s)$ is a <u>negative</u> real number

$\Rightarrow$ corresponding $K = \dfrac{-1}{L_o(s)}$ is positive

and:

$$L_o(s) = 2\ell(180^\circ)$$ (even multiple of 180°)

$\Rightarrow$ corresponding K is <u>negative</u>

"Angle condition", $K > 0$

If we restrict ourself initially to $K > 0$, we need

$$\boxed{\measuredangle L(s) = \measuredangle L_0(s) = (1+2\ell)180^\circ}$$

for s to be a CL pole. This is the "angle condition".

Any value of s satisfying this condition will be a CL pole for an appropriate positive value of K.

Suppose that we want a specific CL pole, s_{Des}.

We need $\measuredangle L(s_{Des}) = (1+2\ell)180^\circ$

But recall: $\measuredangle L(s) = \measuredangle G(s) + \measuredangle H(s)$ for any $s \in \mathbb{C}$

Thus, to make s_{Des} a CL pole we need

$$(1+2\ell)180^\circ = \measuredangle G(s_{Des}) + \measuredangle H(s_{Des})$$

Hence, must design compensator $H(s)$ so that:

$$\boxed{\measuredangle H(s_{Des}) = (1+2\ell)180^\circ - \measuredangle G(s_{Des})}$$

Similar to Bode design approach, define:

$$\boxed{\varphi_{req} = (1+2\ell)180^\circ - \measuredangle G(s_{Des})}$$

(choose ℓ to get φ_{req} in range $[-180^\circ, +180^\circ]$)

Then choose poles/zeros in $H(s)$ so that

$$\boxed{\measuredangle H(s_{Des}) = \varphi_{req}}$$

Example:

Suppose $G(s) = \dfrac{3}{s(s+2)}$

and we want $T(s)$ to have a pole at $s_{Des} = -3+3j$

$$\measuredangle G(s_{Des}) = 116.56^\circ$$

(In Matlab: angle(evalfr(G, -3+3j)))

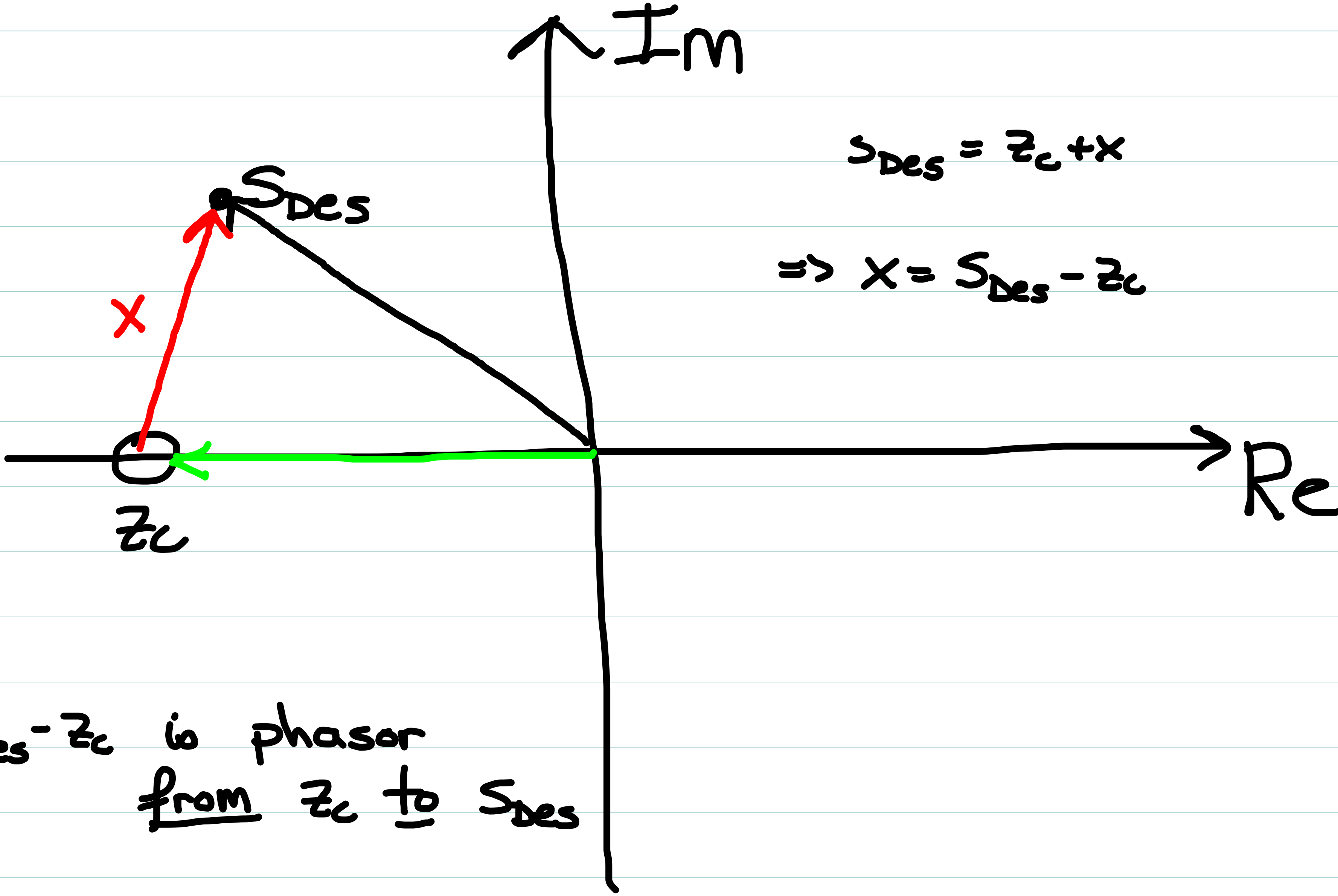
$$\Rightarrow \varphi_{req} = 180^\circ - 116.56^\circ = 63.43^\circ \text{ here}$$

Assume initially: $H(s) = K(s - z_c)$, $K > 0$

(Not implementable: for illustration only!)

Then we need $\measuredangle(s_{Des} - z_c) = 64.43^\circ$

Visualization – Phasor Interpretation

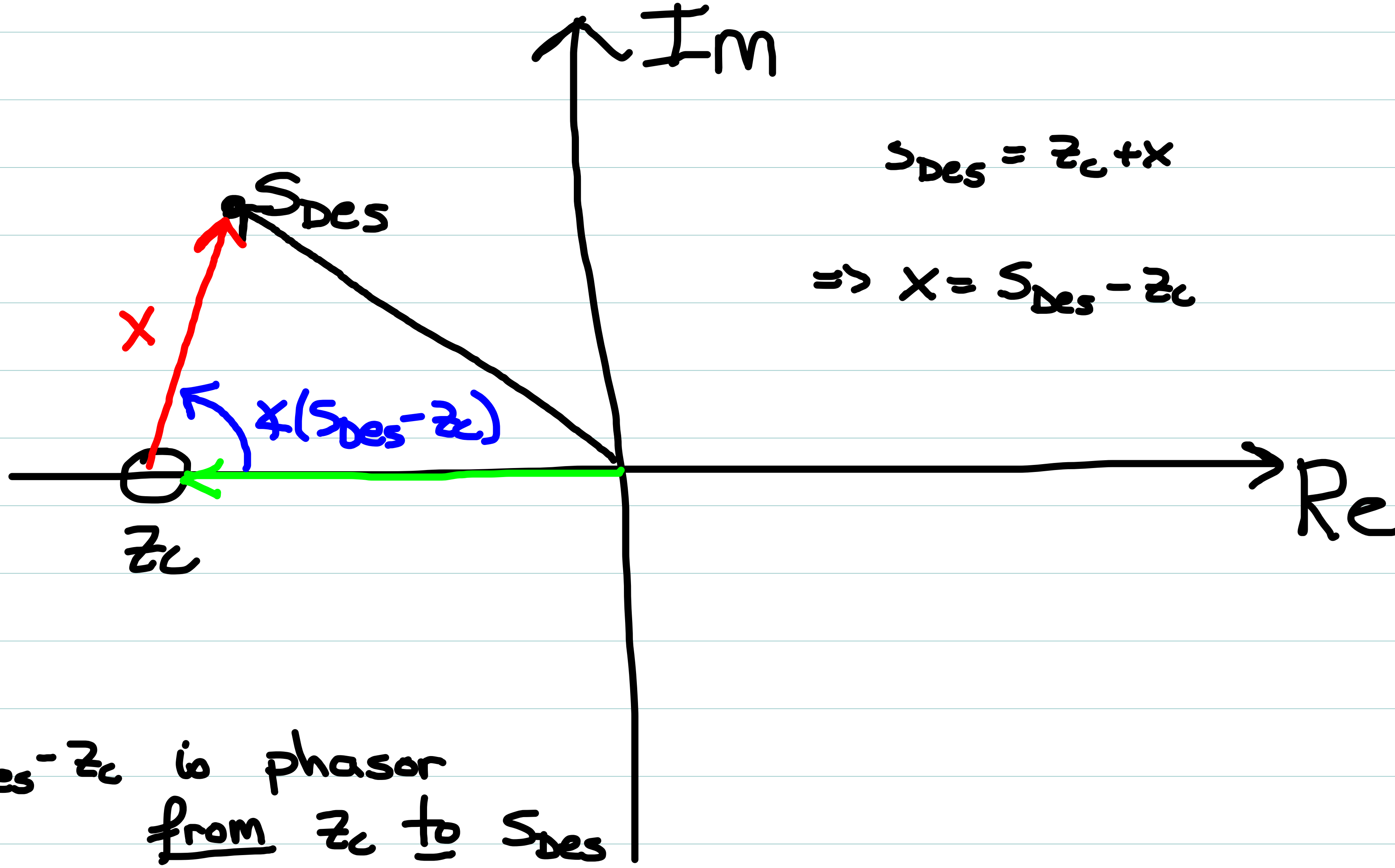


$s_{Des} = z_c + x$

$\Rightarrow x = s_{Des} - z_c$

$s_{Des} - z_c$ is phasor <u>from</u> z_c <u>to</u> s_{Des}

Visualization – Phasor Interpretation



$s_{Des} = z_c + x$

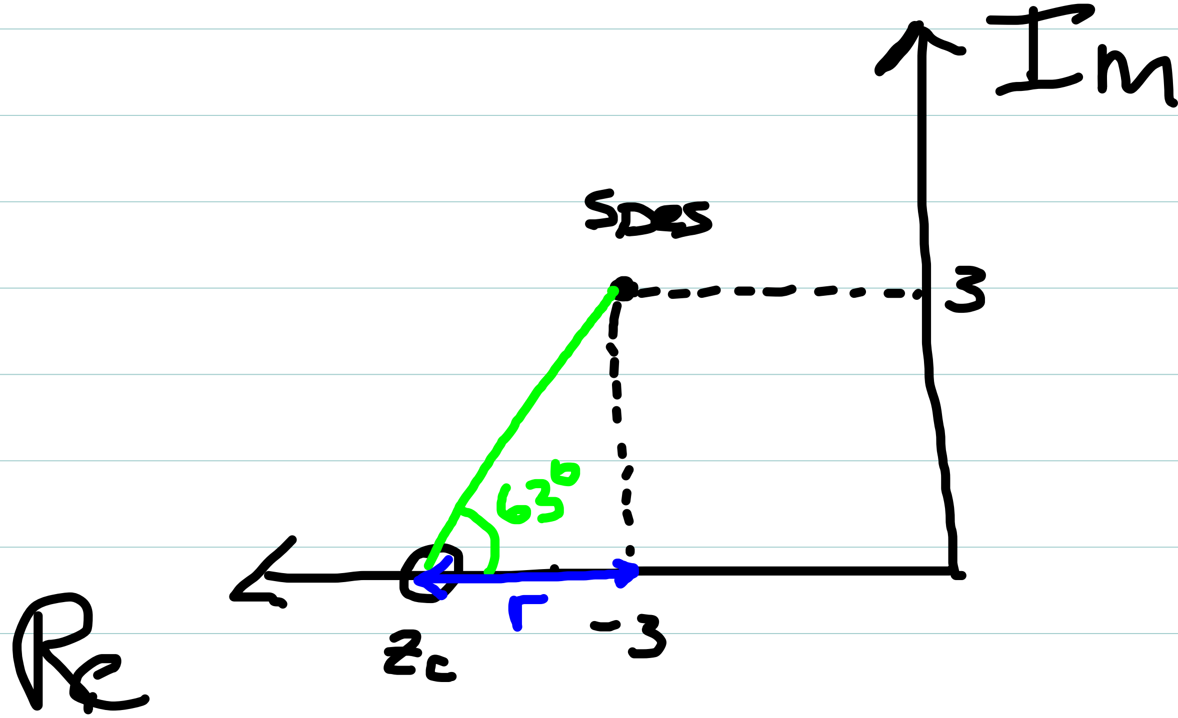
$\Rightarrow x = s_{Des} - z_c$

$s_{Des} - z_c$ is phasor <u>from</u> z_c <u>to</u> s_{Des}

<u>Note</u>: unlike Bode designs we can get <u>up to +180°</u> at s_{Des} from a <u>single</u> zero.

Example cont'd

If we need $\measuredangle(s_{Des} - z_c) = 63.43°$ at $s_{Des} = -3+3j$:



$$\tan(63.43°) = \frac{3}{r} \Rightarrow r = 1.5$$

$$\Rightarrow z_c = -4.5$$

Example, cont'd

So $H(s) = K(s+4.5)$ and then

$$L_o(s) = \frac{3(s+4.5)}{s(s+2)}$$

$$|L_o(s_{Des})| = 3/4$$ (Matlab: abs(evalfr(Lo, -3+3j)))

so $K = 4/3$.

Check:

$$T(s) = \frac{4(s+4.5)}{s^2+2s+4(s+4.5)} = \frac{4(s+4.5)}{s^2+6s+18} \checkmark$$

roots at $-3 \pm 3j$

Notes

1.) To be implementable $H(s)$ needs a pole. Choose pole p_c so that

$$\measuredangle(s_{Des} - p_c) \approx 5^\circ$$

Then $\measuredangle H(s_{Des}) = \measuredangle(s_{Des} - z_c) - \measuredangle(s_{Des} - p_c) = \measuredangle(s_{Des} - z_c) - 5^\circ$

Add $+5^\circ$ to φ_{req} to account for required pole.

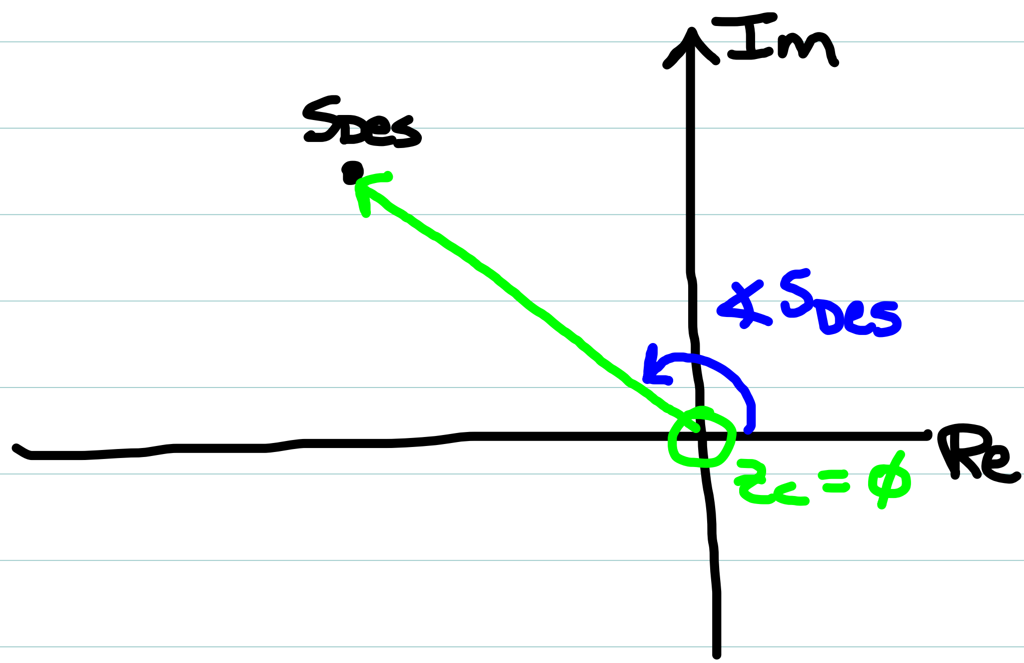
("β-minimizing" principle is quite messy here).

2.) Keep $\varphi_{req} < 90^\circ$, pref. below $60^\circ - 70^\circ$, or else zero will be closer to imag Axis than s_{Des}, creating substantial additional overshoot. "Split" large φ_{req} over multiple zeros if necessary.

Notes (cont).

3.) Do not choose z_c in RHP! (We'll see why later)

⇒ places practical limit on maximum angle contribution from a zero



(effective max for $z_c \leq \phi$)

Notes (cont)

4.) Design method guarantees s_{Des} is a CL pole, but says <u>nothing</u> about location of other CL poles.

These might actually be unstable!

<u>Suppose:</u> $G(s) = \frac{2}{s^2(s+1)}$, $s_{Des} = -2+0j$

$G(-2) = -\frac{1}{2} \Rightarrow \varphi_{req} = \phi \Rightarrow H(s) = K > \phi$ sufficient

$K = \frac{-1}{-\frac{1}{2}} = 2$ and here

$$T(s) = \frac{4}{s^3+s^2+4}$$

With poles at $\boxed{-2}$, $\boxed{\frac{1}{2} \pm \frac{4}{3}j}$ unstable!

To use these ideas effectively as a design tool, we must have <u>some</u> idea where the <u>other</u> poles of $T(s)$ will be; i.e. at least if they are stable.

Requires us to more generally understand <u>all</u> <u>possible</u> solutions of $1 + L(s) = \emptyset$

Or, equivalently, the "locus" of points in the complex plane which satisfy the angle condition(s):

$$\measuredangle L_o(s) = (1+2\ell)180° \qquad (\text{if } K > \emptyset)$$

$$\measuredangle L_o(s) = (2\ell)180° \qquad (\text{if } K > \emptyset).$$