Electric systems: power and frequency control

control of Electric system

Islanded generator case

Multiple generators (network) case

A few words on swinging

Preliminaries

gen

g — net — L load

(from Islanded case: 1 generator feeding 1 load)

- We start from a generator model in the known form

Generator modelling ⇒

$$\left[G(s) = \frac{\Delta P_g(s)}{\Delta u(s)} = P_n g(s) \right.$$

variation of gen power

unity Gain T.F

{ depending on control policy }

variation of gen command (turbine / fuel valve)

where P_n is the nominal power and $g(s)$ depends on the generator dynamics time scales, in turn tied to both size and management policy.

→ • We consistently take an energy-centred approach, i.e., there is no explicit evidence of voltages and currents, and powers are treated as signals. → summ powers for en balance

- We assume the electric power demand P_e – precisely for our context, its *variation* ΔP_e – to be exogenous.
- We consider a single generator feeding its own load, which is named the islanded case.

Preliminaries

- We recall the balance at the alternator shaft, that replacing for generality the m (mechanical) subscript with the g (generated) one, reads

normalized variation of the frequency

normalized var

$$\delta\dot{\omega} = \frac{P_n}{J\omega_o^2}(\delta P_g - \delta P_e)$$

gen power

elect power → all normalized respect Pn

where lowercase δ's denote normalised variations, i.e.,

$$\delta\omega = \frac{\Delta\omega}{\omega_o} = \frac{\omega - \omega_o}{\omega_o}, \quad \delta P_{g,e} = \frac{\Delta P_{g,e}}{P_n} = \frac{P_{g,e} - P_n}{P_n};$$

- notice that we normalise both P_g and P_e w.r.t. P_n.

network balance

- We finally recall that $J\omega_o^2/P_m$ has the dimensions of time, and is denoted by T_A; also, $\underbrace{u \in [0,1]}_{\text{already } [0,1]}$, hence $\underbrace{\delta u = \Delta u}_{\text{same}}$.

Controlled system

for a single generator

same normalized, $u \in [0,1]$ already!

$\delta u(s) = u(s)$

- As a result, in the Laplace domain we can write

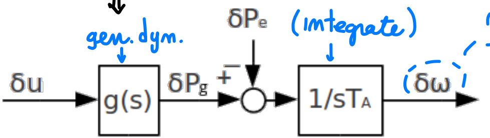
Laplace transform

unity Gain (normalized variation)

$$\delta\omega(s) = \underbrace{\frac{1}{sT_A}}_{\text{INTEGRAL}}\left(\underbrace{\delta P_g(s)}_{=g(s)u(s)} - \delta P_e(s)\right) = \frac{1}{sT_A}\left(g(s)u(s) - \delta P_e(s)\right),$$

- and therefore in block diagram form

for ISLANDED case system



where the disturbance role of δP_e is evidenced.

- Recall that $g(s)$ has low order, and above all is asymptotically stable.

(I/II ORD @ MAX) simple computation

structurally A.S as found previously from T.F matrices

Control scheme

Integrator in the process, so..

{ INTEGRATING power into energy for the Rotating mass! }

INTEGRATOR IN THE PROCESS (power control)

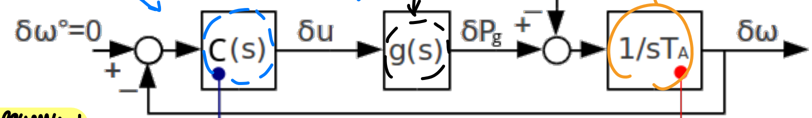
@ st. state

0 input → balance! (control)

second integrator needed!

another 1/s integrat.

(simple model)



BALANCE freq!

We need to control frequency!

Inherent power balance @ steady state ∀ C(s) control

This integrator (intrinsic of the system) guarantees zero steady-state power error

@ steady state all constant, so δω = const means input 0 of block

however I want 0 steady-state freq. error

(δω° - δω)... set point

δω° = 0 so I δω = 0,

We need another integrator in C(s)!

To nullify also the steady-state frequency error another integrator is required here, as g is asymptotically stable; C can be a PI or PID.

Control scheme

③ BUT I want also @ steady state a 0 freq error → $(\delta\omega^\circ - \delta\omega = 0)$

② so independently on what is C(s) the power balance is inherent in steady state ⇐

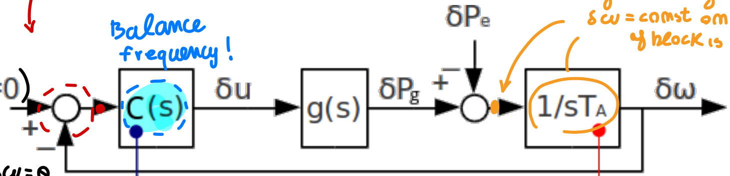
① integrator in the process (so @ steady state that signal $\delta P_g - \delta P_e = 0$) $\delta\omega = \text{const}$ only if input of block is 0

error set point = 0

⇓ $(\delta\omega^\circ = 0)$

so to have $\delta\omega^\circ - \delta\omega = 0$ we want $\delta\omega = 0$

Balance frequency!



This integrator (intrinsic of the system) guarantees zero steady-state power error

To nullify also the steady-state frequency error another integrator is required here, as g is asymptotically stable; C can be a PI or PID.

④ I need a second integrator on C(s)

⇓

since g(s) usually is simple, normally

→ C(s) := PI / PID controller

"I" action needed on C(s)!

$0 \rightarrow \boxed{\frac{1}{s}} \rightarrow \bar{y}$

@steady state → INPUT of 1/s goes to 0

Networked generators

multiple gen. case

(not same scheme or we lose controllability...)

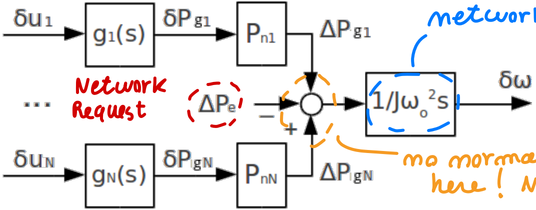
No sense of summ % of different things...

- In the case of multiple generators, at our system level we introduce the rigid synchronous network hypothesis: (all the masses rotate together at the same speed, no swinging.)

(because you cannot summ normalised quantities!!)

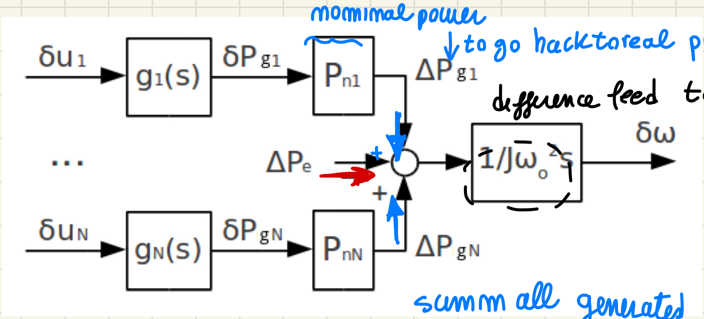
- In this case all the mechanical powers (*not the normalised powers*, beware) sum together, while there is still a *single* electric power demand (the total for the network) subtracted from them. The system under control is thus

having N generators... all same behaviour, summ toghether



where J is the total network inertia (there is no overall T_A as each generator has its own P_n). → (Pn / Jωo² = TA if oney one gen.! each gen. has its own Pn !!)

When 2 or multiple (N) generators



With J total inertia of all generators connected together...

here NO overall TA, each gen has its own Pm

$$\left\{ \frac{P_m}{J\omega_0^2} = T_A \text{ with one gen} \right\}$$ not for multiple

Networked generators

(*) $C_i(s)$ cannot contain the INTEGRATORS!
(overall not more than 1 integrator)
↓ otherwise lose controllability!

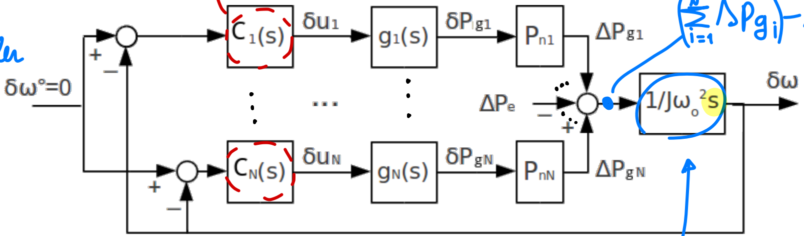
parallel integrator...

- The scheme for the islanded generator is easily extended to multiple generators as
⇓

1 local controller above each generator!

It can be simplified...

$\left(\sum_{i=1}^{N} \Delta P_{g_i}\right) - \Delta P_e = 0$
@ steady state as before!

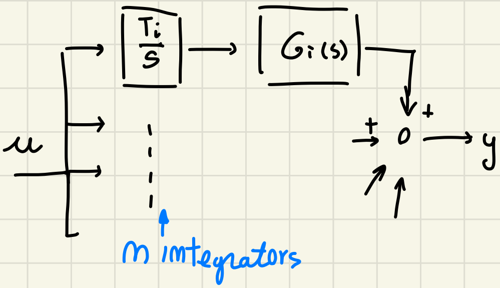


- Here too the network intrinsic integrator $(1/J\omega_o^2 s)$ guarantees zero steady-state power error. 0 POWER ERROR! @steady-state as before
- However the regulators in this scheme cannot encompass integrators, (*) because in that case the generation distribution would not be controllable.

$C_1 \dots C_N$ cannot contain overall more than 1 integrator OR the system will NOT be controllable

NOT GOOD ! ☹

If you have more 1/s integrators in parallel ... on the controllers...

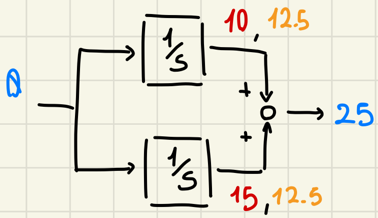


$$\frac{Y}{U} = \frac{1}{s}\sum_{i=1}^{m} T_i G_i(s)$$

↓ on Y/S

⟹ [1 integrator ! (visible)

(m − 1) in a hidden part of the system (hidden dynamic)]

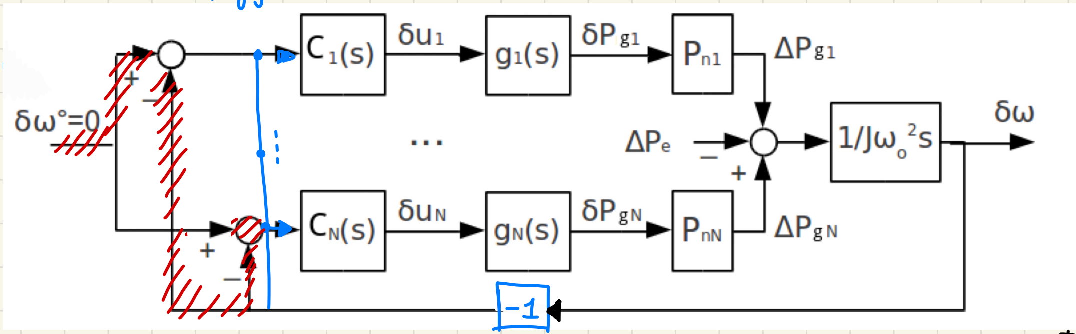


you can have different state, only summ guarantee

↓

{ NON CONTROLLABLE system }

I could simplify the scheme...



This scheme can be simplified... ↑

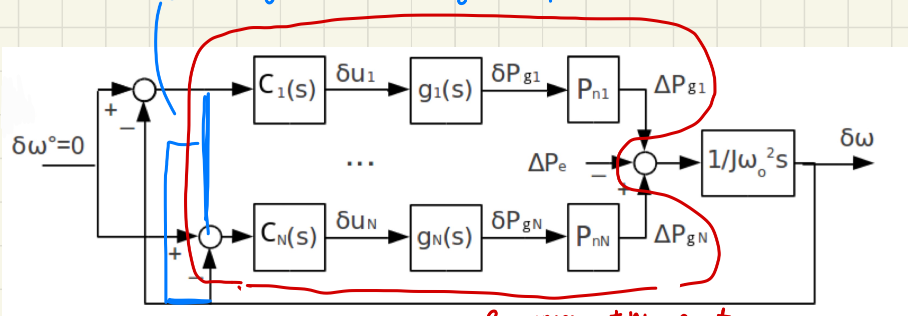
Connect all signals toghether

set point is the (STRANGE situation) set point for the error so ↳ intrinsically ZERO ($\delta\omega^0 = 0$)
you can just close the feedback with a (−1) on feedback path

---

to obtain the scheme on slide 182

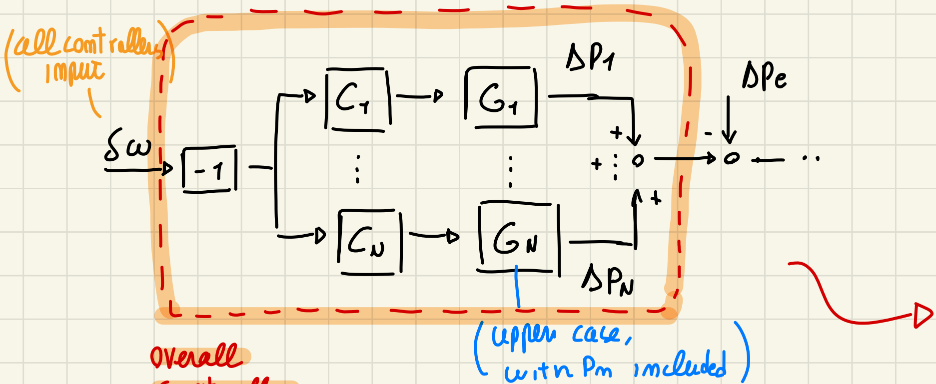
considering all as a single input δω



lumping this part...

↳ { lumping all toghether } ⇒

PARALLEL!
all Ci feed by same signal
and all outputs summ toghether
before (−Δpe) subtraction
[(*)(*)]



Overall controller $F(s)$! $\Rightarrow$ summation of parallel blocks...

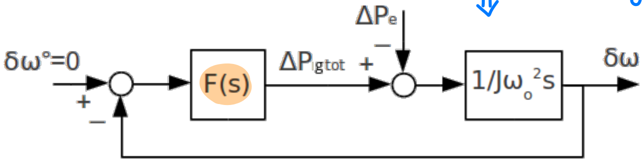
Networked generators

↓ we need a different solution

- To understand why, observe that the scheme is equivalent to ⇓

[(*) (*)]

Just lumping togheter each (SISO)



- where ΔP_{mtot} is the total mechanical power variation (*not normalised*), and

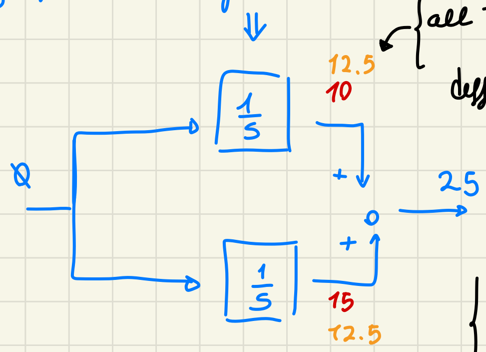
$$F(s)=\sum_{i=1}^{N} C_i(s)\underbrace{g_i(s)P_{ni}}_{G_i(s)}$$

- Possible integrators in the C_i regulators would thus be in parallel, whence the controllability loss.

If each $C_i(s)$ has an integrator (or just more than one!)

If having parallel integrators

⇓



{all those states produce the same output}

different possibilites with same output

↓

{you can only guarantee the summ..}

↓

non controllable (not the single values)

Networked generators

we need a different solution on $C_i(s)$ to guarantee $\delta\omega = 0$ at steady-state!

($\delta\omega = 0$)

- Solution: to have zero steady-state frequency error there must be one integrator.
- Therefore
  - we employ for the *primary regulators* $C_i(s)$ a type 0 structure (most frequently a pure proportional term K, whence the name "$K\Delta f$" frequently encountered for them),
  - and we introduce a *secondary frequency control* in the form of a single integrator per network, having as input the frequency error;
  - the output of the secondary regulator acts as an additive correction on the output of each $K\Delta f$ controller via a gain β_i that can be different for each generator, and dictates how much that generator will be asked to participate into secondary control.

Networked generators

Control scheme

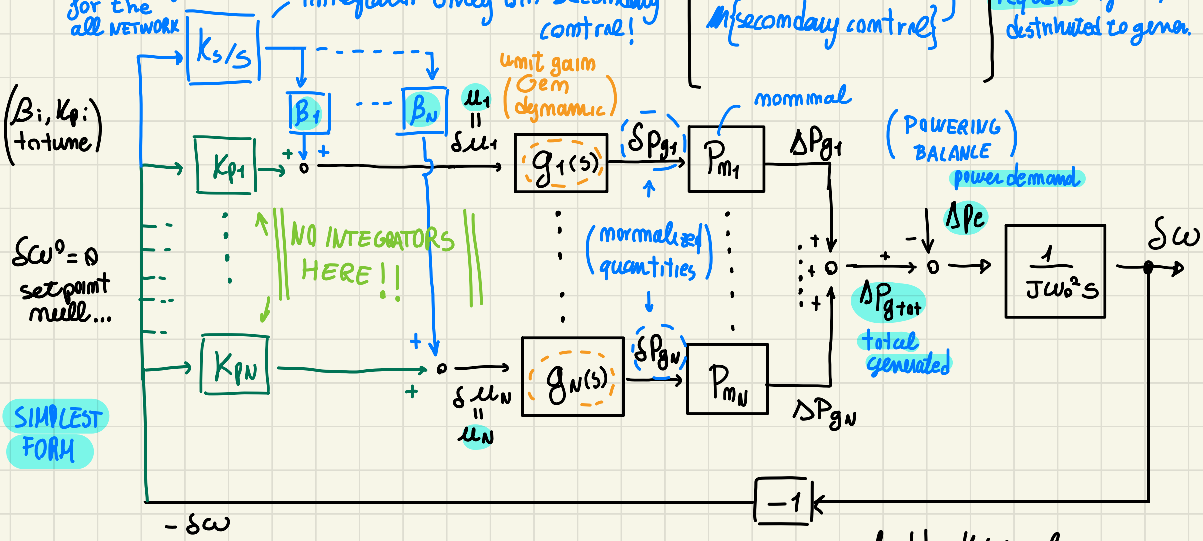
SCHEME

BUILD toghether

Scheme for multiple generators

{ "P": primary
"S": secondary: produces additional action } (SIMPLEST FORM)

- (PROPORTIONAL) K_{pi} {Primary control} or "$K\Delta f$" control → (proportional to freq error)
- {PROCESS part}
- {secondary control} → providing a "secondary request" signal, distributed to gener.



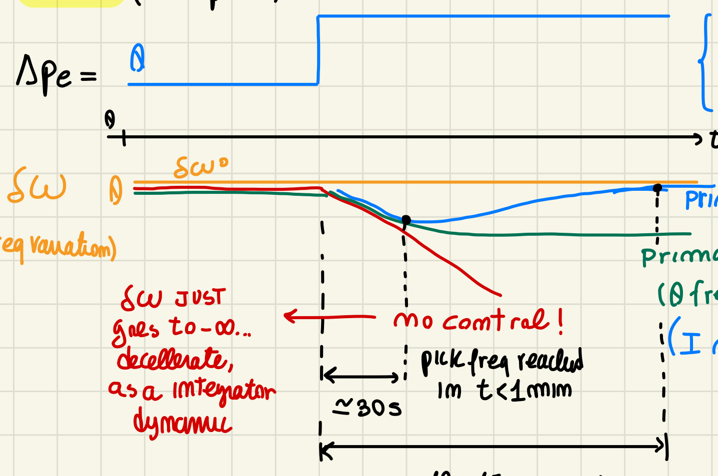
(hyp) rigid + synchr. network ⇒ only one feedback signal independently from where we measure!

(to get rid of parallel integrators...) ↑ (so to have controllability) (sometimes we add additional dynamic)

using the primary control ⇝ { need second control! need integral action to have 0 error $\delta\omega$ } → β_i correction to bring freq $\delta\omega = 0$ back to correct $\bar{\omega}$

⇓

RESULT (t responce)



{positive step Pe... generating less than how much we consume}

(reach $\delta\omega^0 = 0$) Primary + secondary control

Primary control only (0 freq but not @ original)

(I need secondary control!)

↓

not enough to go to 0 at steady state $\delta\omega$ freq don't come back to original value

(idea of "time" quantities) FAST-SLOWER action network wide

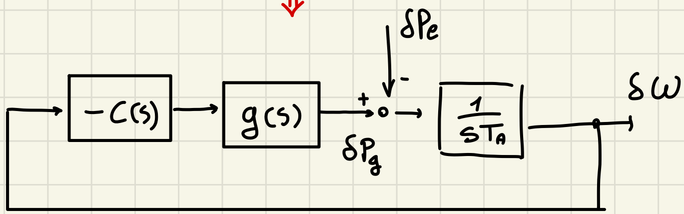
- in some cases we can add dynamics on the PRIMARY block
  BUT that primary block MUST NOT HAVE INTEGRATORS !

  $\downarrow$

  primary block of type 0 ! $\rightarrow$ nothing in the ORIGIN!

ANALYSIS & TUNING – {Islanded Case}

⇓



EXAMPLE 1 let $g(s) = \dfrac{1}{1+s\tau}$ (I ORD dynamic)

? which kind of controller should we use ?

- we need 0 steady-state error ($\delta\omega = 0$) @ steady state → no PROP controller

⇓

$\Leftarrow$ { simple PI controller, we need integrator! } "I" needed

⟹ Advised $C(s)$: PI, that is

$$C(s) = K\frac{1+sT_i}{sT_i} = K_p + \frac{K_s}{s}$$

form for TUNING

PRIMARY — SECONDARY ("I" action)

$\underbrace{K_p + \frac{K_s}{s}}$ form for develope (deploy) is similar to multiple gen case → reduced version...

"I" action: single gen as reduced version of the complex case multiplegen

↓

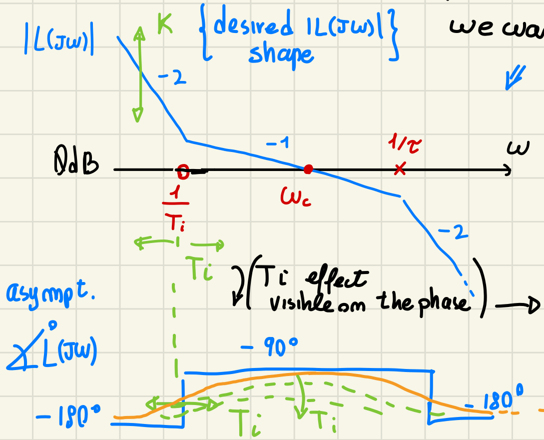
so the LOOP transfer function is $L(s) = C(s)\,g(s)\,\dfrac{1}{sT_A} = ..$

$$... = K\frac{1+sT_i}{sT_i}\,\frac{1}{1+s\tau}\,\frac{1}{sT_A} = \frac{K}{T_iT_A}\,\frac{1}{s^2}\,\frac{1+sT_i}{1+s\tau}$$

→ to synthesize the controller? in electric domain → STANDARD CONTROL structure

(standard CONTROL synthesys) → (tune by cancellation? NO $\varphi_m \simeq 0$) ↩ to get $\varphi_m > 0$ we study from bode plot

we wanna achieve ... $|L(s)|$ like:



true ∡ diag

{ We have 2 DOF (K, T_i) to obtain desired $L(s)$ so the requested requisites }

(by T_i you moove phase shift frequency) ⟹ so you act on φ_m by T_i!

if T_i → less evident phase mountain

• For **TUNING** (quantitatively)

let us write

$$L(s) = \frac{K}{T_i T_A} \frac{1}{s^2} \frac{1+sT_i}{1+s\tau} = \frac{\mu}{s^2} \frac{1+sT_i}{1+s\tau}$$

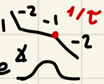
(constant parameter) μ

gen time const τ, T_A = $P_m / J\omega_0^2$ KNOWN parameter

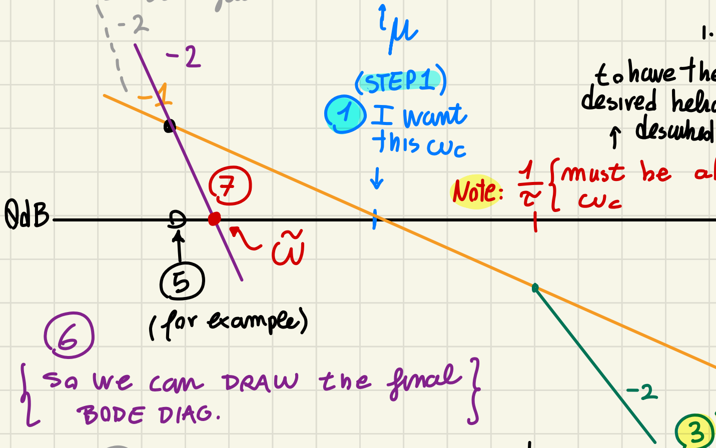
find μ, T_i

↳ get K (equivalent)

{ IF I know μ and T_i I know all PI tuning parameter }



to have the desired behav. ↑ described of Bode



(STEP 1) ① I want this ω_c

Note: $\frac{1}{\tau}$ { must be above ω_c }

⑦

⑤ (for example)

⑥ { so we can DRAW the final BODE DIAG. }

STEP 2 ② Draw line at slope (-1) across ω_c (cutting 0dB axis @ ω_c with -1 slope)

STEP 3 ③ $|L(j\omega)|$ will be like this! (slope (-2) after $1/\tau$)

STEP ④ phase at ω_c without the PI ZERO BUT with the double integrator ← still to introduce

$$\measuredangle L(j\omega_c) = -180° - \mathrm{atan}(\omega_c \tau)$$ ← double integrator (-180° contribution)

(we locate the ZERO respect needed phase) (we should compute how much phase gain we need, so position the ZERO)

STEP ⑤ compute the lead required from the PI ZERO and than locate that ZERO

(USING a phase ruler!) ⟶ you see the $\measuredangle$ gain necessary and so position that ZERO from log paper and phase ruler

knowing where the ZERO will be...

⑥ Draw −2 slope through slope change due to PI ZERO

⑦ (I need μ...) get this $\tilde{\omega}$

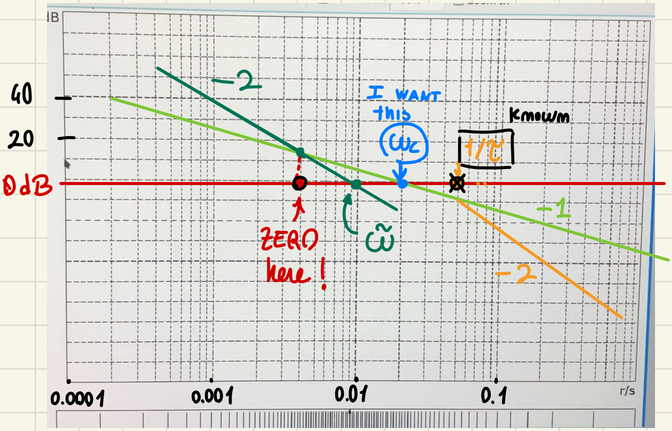
⑧ I know that we must impose $\frac{\mu}{\tilde{\omega}^2} = 1 \Rightarrow \boxed{\mu = \tilde{\omega}^2}$

{ I proceede tuning the PI, acting on $L(s)$ shape on steps respecting the control requirements }

example... ⟹

EXAMPLE from a log paper + phase ruler

⇓



0dB axis

desired ω_c ... (known from the model)

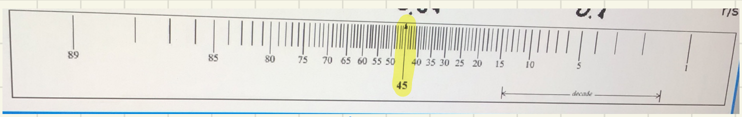
suppose $1/\tau$ on that point...

1. I need -1 on ω_c slope
2. -2 slope after $1/\tau$

↓

than we need to use a phase ruler!

to position the ZERO, respect the desired phase gain



phase ruler realize the "atan" function in logarithmic scale

to use it:

1) check that the decade scale on phase ruler match the log paper scale!

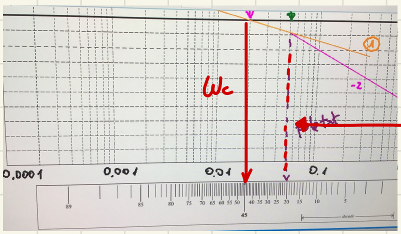
2) to know ∡ @ a certain frequency $\omega^* \to$ in ω_c,

you put 45° arrow in correspondence to the $\omega^* = \omega_c$ (freq where you want to compute the phase)

3) and if you have a singularity @ lower freq
↳ that pole/zero give you a contribute of phase on phase ruler ($\omega < \omega^* \to$ contribution ∡° > 45°)

↳ than summ all! (depending on type of singularity +/−)

in our case



(pole takes 21°)

IF I want $\varphi_m = 55°$

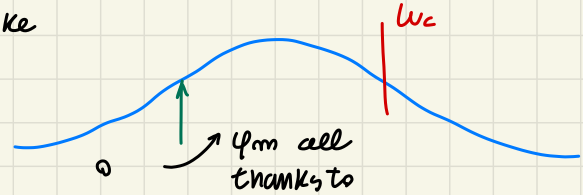
I need 76° given by the zero ⇓

ZERO MUST give 76°

(I can position it depending on that request)

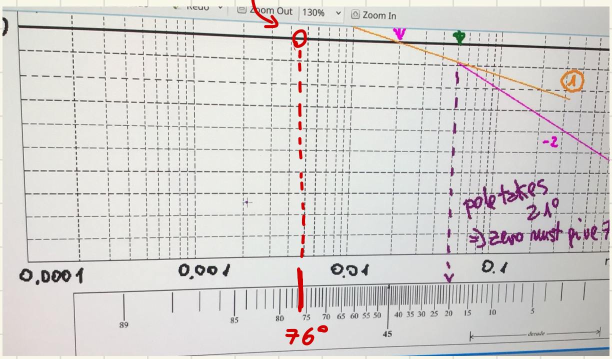
↳ obtaining a phase like

phase margin gain thanks to the ZERO!



φ_m all thanks to the ZERO!

ZERO here!



so to position the ZERO we look where we have 76° of contribution

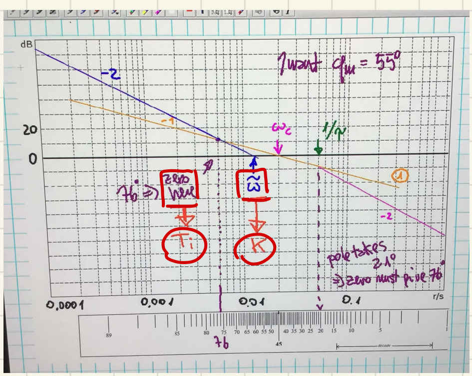
so we can draw -2 line until the ZERO.. (where the slope change $-2 \Rightarrow -1$) and find $\tilde{\omega}$! (here we are not considering constrain optimization)

↳ STANDARD sinthesys procedure → { given responce speed requirement and stability degree (φ_m) }

so once placed
{ ZERO → T_i ; $\tilde{\omega}$ → K } we tune the CONTROLLER !

... so in the end...

↳



limitation:
{ ω_c must be below $1/\tau$ }
↓
ω_c < inverse of generator time const ($1/\tau$)
⇓ Limitation

how to solve this?
to push above ω_c, because you have a fast actuator....
↳

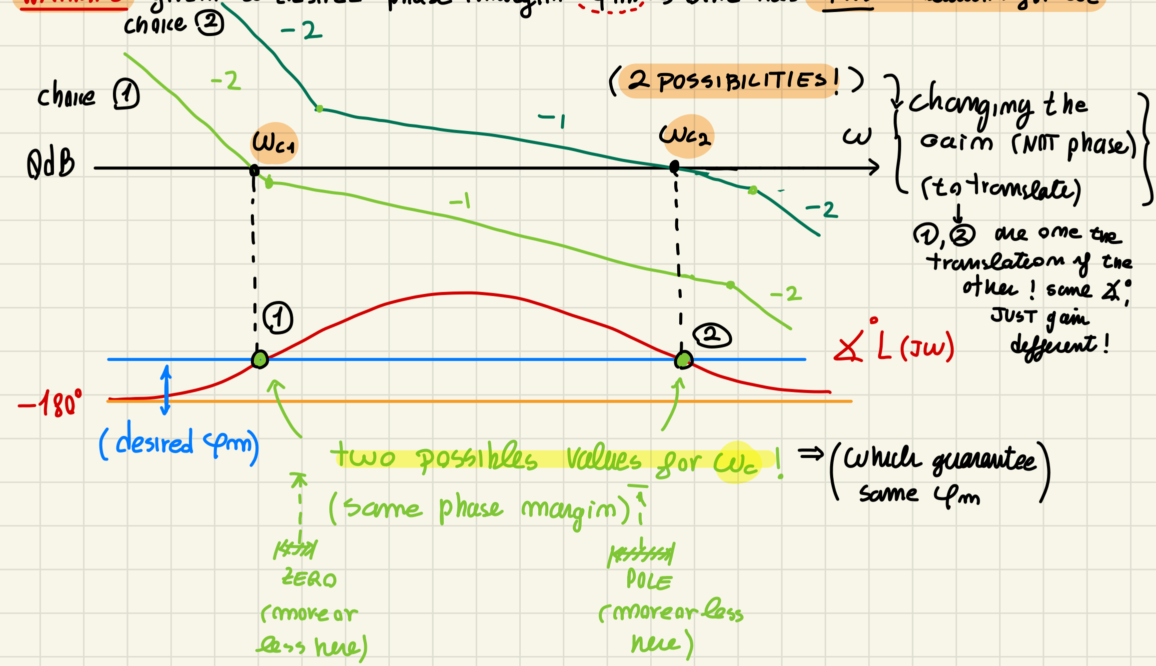
to push above ω_c! $\Rightarrow$ we can use a PID instead of PI,
(so cancel the pole with a ZERO!)

$\left\{ \text{PID} := \text{additional ZERO you can use to cancel generator pole} \right.$
so you can increase ω_c! → FASTER CONTROL $\left. \right\}$

⇒ (solution) use a Real PID with the second ZERO to cancel the generator pole and the 2^{nd} pole located above the desired ω_c

⇓

• WARNING: given a desired phase margin φ_m → one has TWO solutions for ω_c



⇓

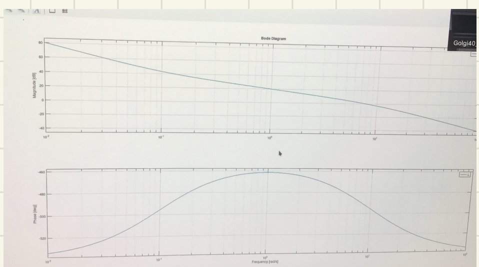
we can see this in "octave" (precise Bode)

pkg load "control"

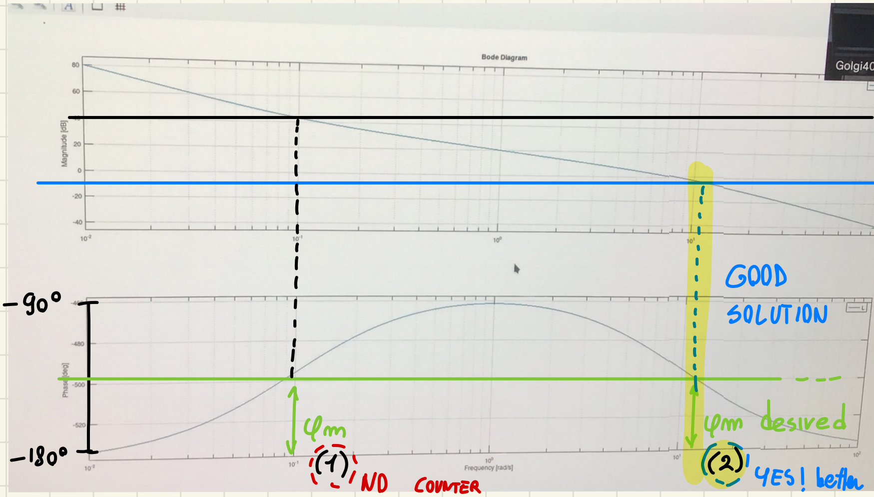
$L = 1/s^2 * (1+s/0.1)/(1+s/10)$ →

(general loop TF)

Bode (L)



↳ as you can see, fixed φ_m desired...



↳ you prefer? Bode criterion... ($\varphi_m > 0$, $K > 0$) OK

{ NOT consider the rule that it can't cut with -2 slope! } ! even IF cutting 0dB axis with (-2) slope, don't care, is import. the φ_m!

• IF system oscillate too much → I must reduce the gain (calm down) → normally reducing the gain increase φ_m...

usually

① ↓ here if $K\downarrow$ $\varphi_m\downarrow$ (counter intuitive ?!)

while here $\varphi_m \downarrow$ IF reduce gain... COUNTER INTUITIVE

normally mean ω_c LOW PASS behaviour $\delta\downarrow$ mean...

You can have $\varphi\uparrow$ here! STRANGE behav...

increasing gain, $\varphi\uparrow$ → COUNTER INTUITIVE during tuning / testing

① IF cut off is (1) → reducing the loop (controller) gain reduces the phase margin φ_m, which is totally counter intuitive

BAD choice!

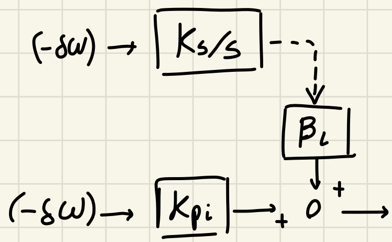
② "good" solution → more intuitive solution!

- **RECAP** → for **ISLANDED case** easy to have good controller configuration given the specification (responce speed / stability)

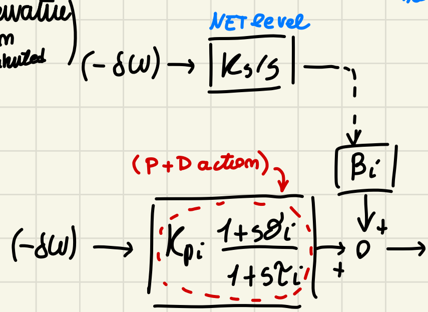
  - more DIFFICULT for a set of Generators BUT conceptually the same! → from full Gen syst. (compact T.F of syst) (you compute the overall T.F and you get $\boxed{K_p, K_s, \beta}$ coefficents)
    - [more τ to consider...]
    - ↳ (more time constants to deal with τ)

PRIMARY + SECONDARY (I can view it as...) $\simeq$ centralized I (integral action centralized) → one for all
+
Distributed P (D) (more proportional action) → each gen. has it own

(also derivative action distributed)



OVERALL (PI)



OVERALL (PID)

⇓

$K \frac{(1+sT)}{s}$ (PI) controller $\begin{cases} 1 \text{ ZERO} \\ 1 \text{ POLE @ origin} \end{cases}$

PI multiplyed by $\left(\frac{\text{ZERO}}{\text{POLE}}\right)$ dynamic becomes: (PID)!

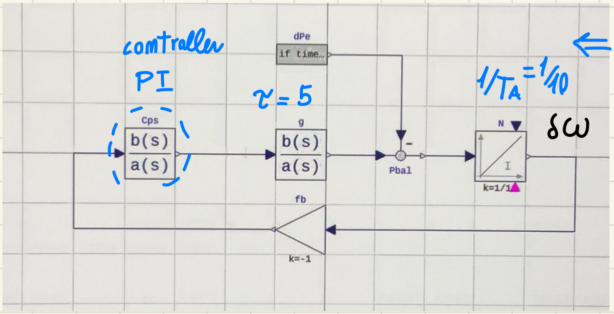
$K \frac{(1+sT_1)(1+sT_2)}{s(1+sT_p)}$ (PID) $\begin{cases} 2 \text{ POLES (1 @ ORIGIN)} \\ 2 \text{ ZEROS} \end{cases}$

↳ (II ORD system)

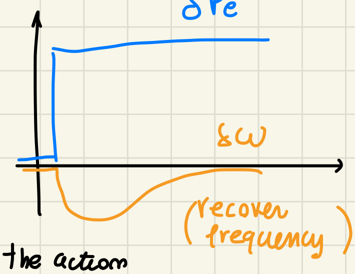
derivative action introduced locally ⇒ @ generator level I add zero/pole
(i can do for some gen and other NO...)

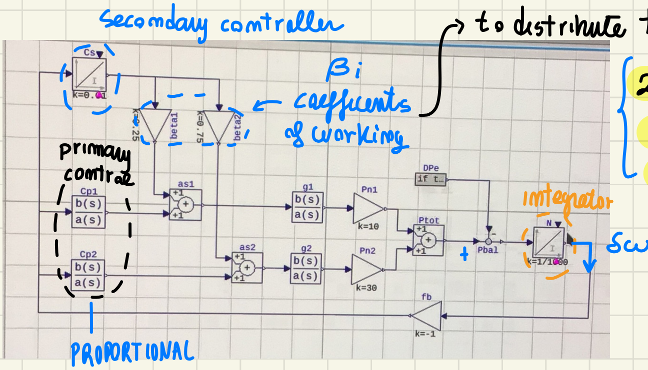
(PRINCIPLE!)

on MODELICA



{1 islanded generator}



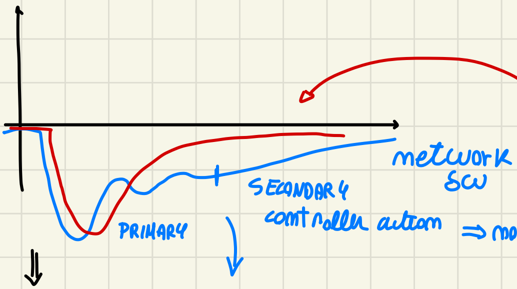


→ to distribute the action

{2 generator with primary and secondary control}

controllable system desired

⇒ our objective is to set up the controller and achieve the goal specification



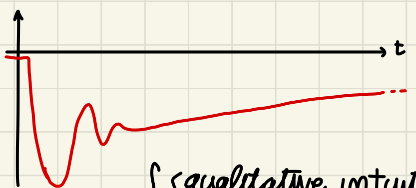
(Better action of $\delta\omega$)

{decreasing the gain of I, (primary) and increasing II gain (secondary)}

⇒ not so well tuned

⇓

on it

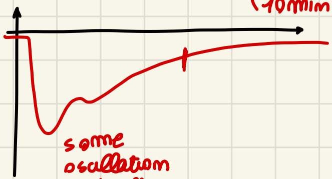


excess of proportional action, (initially too nervous)

+ lack of integral action (takes too much time to reach the steady state $\delta\omega = 0$)

{(qualitative intuitive the control leakage) idea of what happen to tune better!}

IMPORTANT to understand! → (tuning capability)



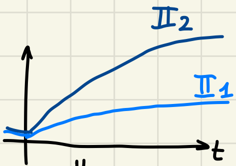
(10min for steady state)

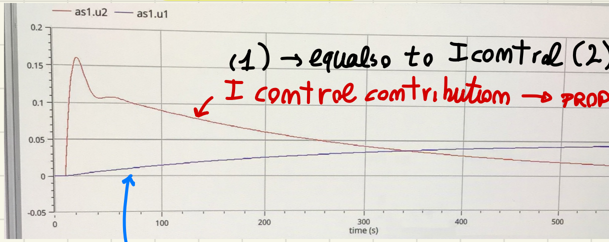
← intermediate...

same K_p, K_s

some oscillation initially

control action





(1) → equalso to I control (2)

I control contribution → PROPORTIONAL, @ the end goes to 0 when $\delta\omega \to 0$

II control (1) → INTEGRAL control ⇝ keeps going reaching steady value

⇓

I partition the controller secondary II by B action ⇝ { 75% → to 2 gen; 25% → to the 1 gen } ask different contribution to sustain the system on integral action B

↳ you can distribute the action as you wish (proportionally to gensize) → || and acting on CONTROL ACTION of $K_{p1}, K_{p2}, K_s, \beta$ || ⤵ you can

- achieve steady state, (cover the transient)
- govern the transition

↓ Integral action acts later first you act only by proportional action + distribute steady state contribute by gen. through B

⇓

properly tune primary / secondary control

- TUNING islanded case (same procedure)
- TUNE more generator

} ⤶ ability of:

Example 1

Islanded generator with (dominantly) first-order dynamics

Foreword

here we don't want to model electrical phenomen. → is enough to measure frequency! so a phasor model, BUT ← {this is because the coupling is electrical between gen!}

this happen because of how you transfer power from gen to NET, or necessary frequency measurements

↑ (NO mech coupling, BUT electrical) between gen for small parts of time

- It is sometimes useful to account for a non-rigid network;
- This means that generators can (transiently) rotate at different frequencies.
- To represent the involved phenomena, a mechanical equivalent of the generators' interconnection is the most adequate level of detail.
- To address the matter, we need however to first discuss how power is transferred in AC networks with controlled voltage.
- As usual (and with no didactic detriment) we assume a single voltage throughout the network, and a single-phase system.
- We also assume the network frequency to be controlled (now we know how), hence we can use phasors.
- We finally assume a prevailing network, i.e., that each generator is individually so small, compared to the union of the others, to see the network voltage as a fixed phasor. ~> assume net as still phasor! ⇒ (*)

every gen sees net as fixed phasor

Writing a model with phasors

coupling gen... we don't want models with electric phenomenon

JUST use freq measurement (automatic power balance)

till Now we only measure freq

(no elect quantities)

Sometimes We have transient with NON equal freq! and we wanna set-up a mechanical equivalent model!

to do this mech equivalent model we MUST discuss how AC network transfer power with controlled voltage sometimes we want to face probeem with transient NON equal frequency → set up a mech. equivalent model

↓

to do this we need to understand something else

↓

sometimes we need a NONRIGID NETWOR

because of how we transfer power from a gen to the network

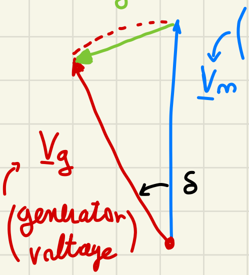
⇓

$\underbrace{\text{gen power}}_{}$ << network power

(*) ↳ each gen see the Net as a fixed phasor!

(difference phasor) responsable of {POWER TRANSFER across the generator-to-network admittance: ($\underline{Y}_{gm}$)}

$\underline{V}_g - \underline{V}_m$



⇓ on complex domain

$$\begin{cases} \underline{I}_{gm} = \underline{Y}_{gm}(\underline{V}_g - \underline{V}_m) \\ \underline{S} = \underline{V}_g \cdot \underline{I}_{gm}^* \end{cases}$$

↑ (POWER complex)

↑

{ $\underline{I}_{gm}$: current from gen to network
$\underline{S}$: complex power }

{ to transfer power we want the gen. phasor to be different from the network phasor }

↓

voltage control

↑

we want to preserve voltage control

Foreword

- In the above hypotheses one could see a contradiction between ideally controlled frequency, and generators accelerating/decelerating.
- What we precisely mean is that a generator can *transiently* accelerate or decelerate to change its voltage angle with respect to the network one (motivations follow),
- but then the generator recovers synchronisation with the network frequency,
- and the entity of such relative angle movements are individually associated to so small powers with respect to the total, to still allow to see the network voltage phasor as fixed.

- This said, we can proceed.

Power transfer

- Consider a generator connected to the network through an admittance $\underline{Y}_{gn}$.
- Let $\underline{V}_g$ and $\underline{V}_n$ be the generator and network voltage phasors.
- The current flowing from generator to network is therefore

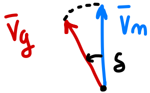
$$\underline{I}_{gn} = \underline{Y}_{gn} \left(\underline{V}_g - \underline{V}_n \right),$$

- and the complex power leaving the generator is

$$\underline{S}_{gn} = \underline{V}_g \left(\underline{Y}_{gn} \left(\underline{V}_g - \underline{V}_n \right) \right)^*,$$

where, recall, * denotes the complex conjugate.

Power transfer



$(\underline{V}_g - \underline{V}_n)$ angle

computation

- Denoting by δ_{gn} the angle between phasors $\underline{V}_g$ and $\underline{V}_n$,some Maxima gives us the active and reactive power (P_g and Q_g, respectively) leaving the generator as

```
Vn  : Vnm;                        /* magnitude Vnm, phase 0 (reference)  */
Vg  : Vgm*(cos(dgn)+%i*sin(dgn)); /* magnitude Vgm, phase dgn            */
Ygn : Ggn-%i*Bgn;
Ign : (Vg-Vn)*Ygn;
Sgn : Vg*conjugate(Ign);
Pg  : trigsimp(realpart(Sgn));
Qg  : trigsimp(imagpart(Sgn));
```

$$\begin{cases} \mathrm{Re}(\underline{Y}_{gm}) = G_{gm} \\ \mathrm{Im}(\underline{Y}_{gm}) = B_{gm} \end{cases}$$

$\Downarrow$

- We get

active power

$$\begin{aligned} P_g &= G_{gn}|\underline{V}_g|^2 - |\underline{V}_g||\underline{V}_n|\left(G_{gn}\cos\delta_{gn} - B_{gn}\sin\delta_{gn}\right) \\ Q_g &= B_{gn}|\underline{V}_g|^2 - |\underline{V}_g||\underline{V}_n|\left(B_{gn}\cos\delta_{gn} + G_{gn}\sin\delta_{gn}\right) \end{aligned}$$

reactive power

- As can be seen, acting on $|\underline{V}_g|$ and δ_{gn} allows to control both active and reactive power.

$\hookrightarrow$ BUT to control P, Q I have to give up voltage control... fixed $\rightarrow$ Voltage would be fixed to control P,Q

Power transfer

[we assume ideal internal voltage control]

control active power, considering voltage controlled by gen, ideally

- For our purpose let us however assume ideal voltage control, i.e., $|\underline{V}_g| = |\underline{V}_n| = V$; this reduces the previous equations for P_g and Q_g to

from previous example... computing..

$P_g \simeq \sin(\delta_{gn})$

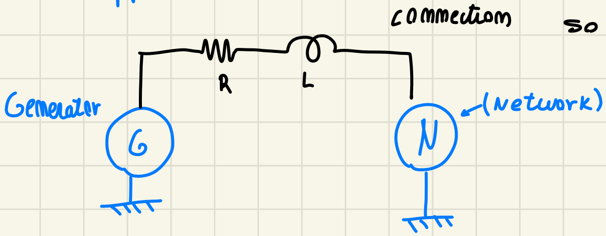
$$\begin{cases} P_g = V^2\left(\cancel{G_{gn}(1-\cos\delta_{gn})} + B_{gn}\sin\delta_{gn}\right) \\ Q_g = V^2\left(B_{gn}(1-\cos\delta_{gn}) - G_{gn}\sin\delta_{gn}\right) \end{cases}$$

$B_{gn} \gg G_{gn}$

- Assuming $B_{gn} \gg G_{gn}$, the active power transferred to the network is proportional to the sine of the angle between the generator and the network voltage. (*) ⇒
- This is called the $\boxed{\text{machine angle.}}$ = δ_{gn}

to control the amount of delivered active power transmitted you change machine angle (this is how we transfer power...)

suppose to have



(*)

$\Rightarrow$ I don't want to dissipate active power!

so I can assume

{ by construction the Im part is larger than Re part lost on connection! }

Im >> Re

by construction!

$B_{gm} >> G_{gm}$

Power transfer

- We can easily check this with a bit of Maxima:

```
Vn   : V;
Vg   : V*cos(dgn)+%i*V*sin(dgn);
Ygn  : Ggn-%i*Bgn;
Ign  : (Vg-Vn)*Ygn;
Sgn  : Vg*conjugate(Ign);
Pg   : factor(trigsimp(realpart(Sgn)));
Qg   : factor(trigsimp(imagpart(Sgn)));
```

simplified model

we wanna work with a block diagram (we want an equivalent model without electrical quantity and connection)

A mechanical equivalent model

Rationale and interface

how to obtain a simplified model without having to work on electric domain

total power must go to 0 when connected

- We can represent the connection between generators and network by means of a Modelica connector carrying an angle as effort variable and a power as flow variable (similar to a mechanical rotational flange, just with power instead of torque for convenience).

rigidly connected, all angles equals ! on connection

- This is specified as

```
connector PowerAnglePort
        Modelica.SIunits.Angle          theta;
  flow Modelica.SIunits.ActivePower P;
end PowerAnglePort;
```

effort variable → when connected 2 or more, done rigidly

power as flow var. if more connection ΣP = 0, as done with voltage/current

- Notice the use of SI units through the Modelica Standard Library.

block diagram with power & signal → no electric quantities

A mechanical equivalent model

Equations — we stick here to the m subscript for the generated power

θ_{port}	=	port.theta	Connector equations for angle
P_e	=	port.P	and power ← own internal angle
ω_{port}	=	$\dot{\theta}_{port}$	Angular velocities ← network frequency
ω_g	=	$\dot{\theta}_g$	← my gen freq → transiently different!
$J_g\dot{\omega}_g$	=	$\tau_m - \tau_e$	Momentum balance for generator inertia
P_e	=	$\omega_{port}\tau_e$	Power/torque/velocity equation, electrical ← (mechanically expression)
P_m	=	$\omega_g\tau_m$	Same, mechanical power
δ_g	=	$\theta_g - \theta_{port}$	Machine angle w.r.t. port (network)
τ_e	=	$K\delta_g + F\dot{\delta}_g$	Equivalent spring/damper
P_m	=	$G(s)u$	Mechanical power from generator dynamics

can be transiently different

torque -power

→ respect a momentum balance

momentum Balance → elastically, increase τ → increase δ → increase power!

MODEL A GENERATOR

FRICTION term, proportional to $\dot{\delta}_g$) damping factor

A mechanical equivalent model

Equations — we stick here to the m subscript for the generated power

(angle seem at the port)

θ_{port}	$=$	port.theta	Connector equations for angle
P_e	$=$	port.P	and power
ω_{port}	$=$	$\dot{\theta}_{port}$	Angular velocities
ω_g	$=$	$\dot{\theta}_g$	
$J_g\dot{\omega}_g$	$=$	$\tau_m - \tau_e$	Momentum balance for generator inertia
P_e	$=$	$\omega_{port}\tau_e$	Power/torque/velocity equation, electrical
P_m	$=$	$\omega_g\tau_m$	Same, mechanical side
δ_g	$=$	$\theta_g - \theta_{port}$	Machine angle w.r.t. port (network) → definition
τ_e	$=$	$K\delta_g + F\dot{\delta}_g$	Equivalent spring/damper
P_m	$=$	$G(s)u$	Mechanical power from generator dynamics

network frequency

my generator freq → (transiently $\dot{\theta}_g \neq \dot{\theta}_{PORT}$) → acc/dec to obtain vel desired

little acc/dec around $\dot{\theta}_{port}$

(where transfer ...)

(mechanical expression)

P_e electrical power given by $\tau \cdot \omega$ (torque × angular vel)

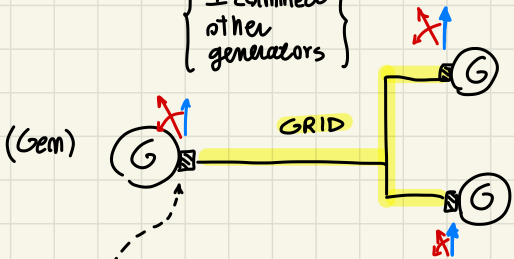
(mech torque)

instead of power balance

from momentum balance: $\tau_e = K\delta_g + F\dot{\delta}_g$ (τ_e proportional to δ_g and $\dot{\delta}_g$)

G phasor → inject different power!

{ I commect other generators }



{ I commect other generators }

port: { δ_{port}
P_e (electrical power) }

• I see same network phasor ↑
• and each gen has its own phasor respect net one ↓ (swinging)
[each one to inject different power]

in my P.O.V

• all equal frequencies → every gen can transiently swing respect SAME Network phasor!

(sporadic moovement in time!) all swing of same quantities ⇐ { angle change with oscillation BUT ALL respect SAME phasor! }

↓

{ some constant reference phasor respect to which all phasor swings }

[commect all togheter]

(any gen. can change the angle)
↳ and in changing the angle you can have oscillation, but all respect the same Network phasor

I move myself mooving a bit respect network phasor → sporadic moovement ←
⇒ all reffered to one reference!

τ depends on $K \delta_g$ (angle diff)
IF $\delta_g \uparrow \rightarrow \tau \uparrow \rightarrow P_m \uparrow$ power increase } modelling aspect!

assuming $\omega = \omega_m$ everywhere,
you can compute

P/δ transfer function!
↓
with proper oscillation + demping factor..

$\tau_e \sim \delta_g, \dot{\delta}_g$

(Elastic component)

because

$(\tau \sim \delta_g) \Rightarrow \begin{cases} \text{means that @ same Velocity } \dot{\delta}_g \\ \delta_g \uparrow \rightarrow \tau \uparrow \rightarrow P \uparrow \text{ more power!} \end{cases}$

fixing $\dot{\delta}_g$, the torque depends as $\tau \sim K \delta_g \uparrow \Rightarrow \tau \uparrow \Rightarrow P_e = \omega \tau \uparrow$ more power!

(modelling)

to turn a generator shaft you need torque... mechanical P.O.V

DISSIPATIONS! $\Rightarrow$ FRICTION term (proportional to $\dot{\delta}_g$)

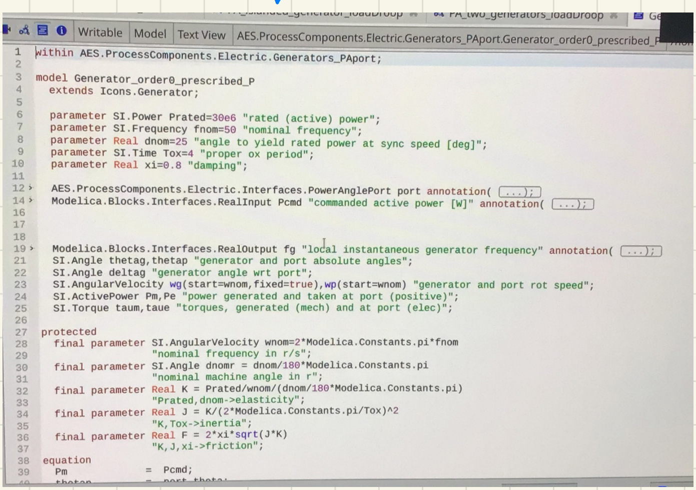
$\tau_e \sim F \dot{\delta}_g$ (otherwise ∞ oscillation)

damping factor

FRICTION!

you see a transient

Modelica model:



Writable | Model | Text View | AES.ProcessComponents.Electric.Generators_PAport.Generator_order0_prescribed_P

```
within AES.ProcessComponents.Electric.Generators_PAport;

model Generator_order0_prescribed_P
  extends Icons.Generator;

  parameter SI.Power Prated=30e6 "rated (active) power";
  parameter SI.Frequency fnom=50 "nominal frequency";
  parameter Real dnom=25 "angle to yield rated power at sync speed [deg]";
  parameter SI.Time Tox=4 "proper ox period";
  parameter Real xi=0.8 "damping";

  AES.ProcessComponents.Electric.Interfaces.PowerAnglePort port annotation( ...);
  Modelica.Blocks.Interfaces.RealInput Pcmd "commanded active power [W]" annotation( ...);



  Modelica.Blocks.Interfaces.RealOutput fg "local instantaneous generator frequency" annotation( ...);
  SI.Angle thetag,thetap "generator and port absolute angles";
  SI.Angle deltag "generator angle wrt port";
  SI.AngularVelocity wg(start=wnom,fixed=true),wp(start=wnom) "generator and port rot speed";
  SI.ActivePower Pm,Pe "power generated and taken at port (positive)";
  SI.Torque taum,taue "torques, generated (mech) and at port (elec)";

protected
  final parameter SI.AngularVelocity wnom=2*Modelica.Constants.pi*fnom
                  "nominal frequency in r/s";
  final parameter SI.Angle dnomr = dnom/180*Modelica.Constants.pi
                  "nominal machine angle in r";
  final parameter Real K = Prated/wnom/(dnom/180*Modelica.Constants.pi)
                  "Prated,dnom->elasticity";
  final parameter Real J = K/(2*Modelica.Constants.pi/Tox)^2
                  "K,Tox->inertia";
  final parameter Real F = 2*xi*sqrt(J*K)
                  "K,J,xi->friction";
equation
  Pm             =  Pcmd;
```

A mechanical equivalent model

Parametrisation

from that equations...
thats how we model a generator

given to the model

- Parameters (we omit Modelica.SIunits.) are given as

```
ApparentPower   Srated  "rated power";
Frequency       fnom    "nominal frequency";
Real            dnom    "angle to yield rated power at sync speed [deg]";
Time            Tox     "proper oscillation period";
Real            xi      "oscillation damping factor";
```

assuming $\omega = \omega_m$ for linearity you can compute δ to P_e T.F

- and those in the model just written are computed as

```
// nominal frequency in r/s
    AngularVelocity            wnom  = 2*Modelica.Constants.pi*fnom;
// nominal machine angle in r
    Angle                      dnomr = dnom/180*Modelica.Constants.pi;
// Srated,dnom->elasticity
    RotationalSpringConstant   K     = Srated/wnom/(dnom/180*Modelica.Constants.pi);
// K,Tox->inertia
    Inertia                    J     = K/(2*Modelica.Constants.pi/Tox)^2;
// K,J,xi->friction
    RotationalDampingConstant  F     = 2*xi*sqrt(J*K);
```

nom angle in radiant

coefficients

you see that damping factor and oscillation period are related to phisical parameter

for Generators

A mechanical equivalent model

to MODEL LOADS

So with Re, Im part! complex domain

Loads (just a few words here) → in electric domain, a load will be modelled as a complex Impedance

BUT here, interested only in active power → we introduce 2 kinds of load

1) constant active power acting on port with fixed angle!
2) [illegible]

- As for loads, a constant-power one is just composed of an input P for the active power to draw, and of the equation

(1)

$$\text{port.P} = P.$$

- A linear droop one with power variation limit is conversely written as

(2)

```
parameter Power     Pnom   "nominal (active) power";
parameter Frequency fnom   "nominal frequency";
parameter Real      droop  "DP/Pnom = droop * Df/fnom";
parameter Real      beta   "max per-unit P variation wrt Pnom";
...
final parameter SI.AngularVelocity wnom = 2*Modelica.Constants.pi*fnom;
...
w    = der(port.theta);
Pact = port.P;
Pact = max((1-beta)*Pnom,min((1+beta)*Pnom,P+droop*(w-wnom)/wnom*Pnom));
```

(power var. drift away)

↳ parameter represent that you ammit the power variation to drift away from nom. value

2 kinds of loads

1) constant active power, put on port const active power,
with δ_g determined, $\underline{I}$ prescribed $\rightarrow$ fix $\underline{V}$

2) linear droop load

Definition of DROOP:

"DROOP" := by definition

$$\frac{\Delta P}{P_{nom}} = \text{DROOP} \cdot \frac{\Delta f}{f_{nom}}$$

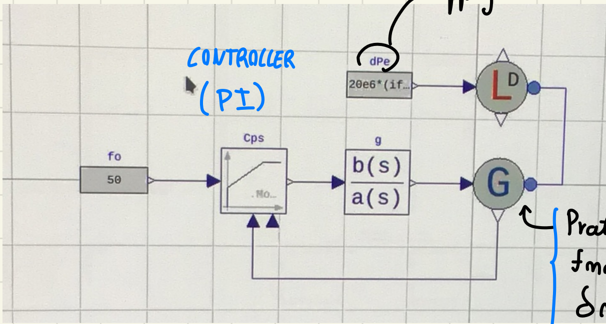
{ IF you vary $P \rightarrow \Delta P$ leads to Δf
and opposite also! }
varying $f \rightarrow$ also P vary

Example 1

Two generators and four loads (two with droop)

Example
simple gen. case

apply a load variation 20 MW
and 0.1 increase at 600s

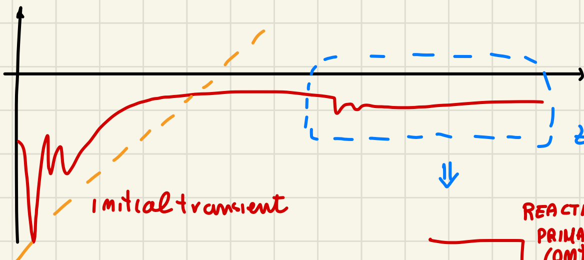


Generator Value

$P_{rated} = 30\ MW$
$f_{nom} = 50\ Hz$
$\delta_{nom} = 25°$
$T_{ox} = 0.8\ s$
$F = 0.3$ damping

⇓

simulating...

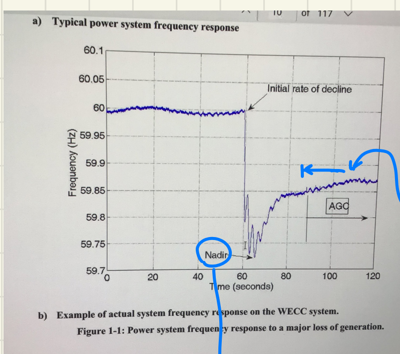


zoom on interested range

⇓

IEEE → document technical (real actuation)

documentation **technical report:** (real behav.)



(high freq. components neglected)

slow recovery after oscillation

↕

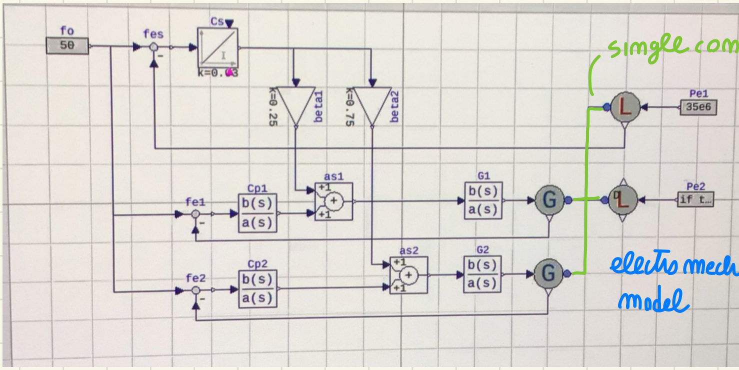
similar to our simple model!

time drop peak

all relevant phenomenon well modelled

We decently reproduce the behaviour of a real gen through a simple model! (similar shape)

2 LOADS, 2 GEN EXAMPLE

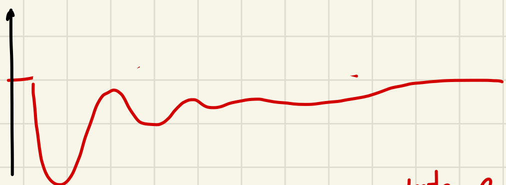


single connection... I/O between G, L

{ single wire connecting all toghether }

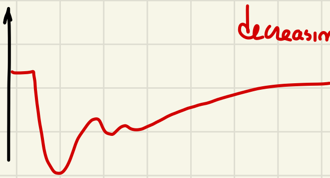
electromech model

{ Primary + secondary control... } ⇒



Integral time reduction!

decreasing K_s sequen integral time, better damper



if low damp

almost equal..

fast dynamic invariant...

↓

{ dominated by network dynamic }

IF low damping on one Gen

⇓

- necessary knowledge about it work!