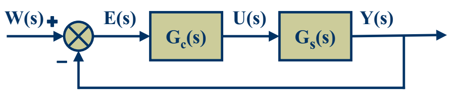
Fractional order systems and controllers

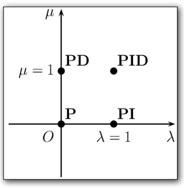
Fractional-order systems



$$G_s(s) = \frac{1}{a_n s^{\beta_n} + a_{n-1} s^{\beta_{n-1}} + \ldots + a_1 s^{\beta_1} + a_0 s^{\beta_0}}$$

$$a_n D^{\beta_n} y(t) + a_{n-1} D^{\beta_{n-1}} y(t) + \ldots + a_0 D^{\beta_0} y(t) = u(t)$$

$PI^{\lambda}D^{\mu}$ controllers



$$G_c(s) = \frac{U(s)}{E(s)} = K_p + K_I s^{-\lambda} + K_D s^{\mu}$$

$$u(t) = K_p e(t) + K_I D^{-\lambda} e(t) + K_D D^{\mu} e(t)$$

Bode's ideal loop transfer function

Hendrik Wade Bode (1905-1982)

H.W. Bode, Network Analysis and Feedback Amplifier Design. D.Van Nostrand Company, Inc., New York, 1945.

Bode's ideal loop transfer function:

$$L(s) = \left(\frac{s}{\omega_{gc}}\right)^{\alpha}$$

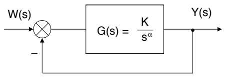
ω_{gc} is desired crossover frequency

α is slope of the ideal cut-off characteristic.

Phase margin is $\Phi_m = \pi(1+\alpha/2)$ for all values of the gain. The amplitude margin A_m is infinity. The constant phase margin 60^o, 45^o and 30^o correspond to the slopes $\alpha = -1.33$, -1.5 and -1.66.

The Nyquist curve for ideal Bode transfer function is simply a straight line through the origin with $\arg(L(j\omega)) = \alpha\pi/2$

Bode's ideal loop transfer function



Bode's ideal loop.

General characteristics of the Bode's ideal transfer function:

(a) Open loop:

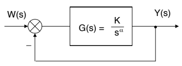
- Magnitude: constant slope of $-\alpha 20dB/dec.$;
- Crossover frequency: a function of K;
- Phase: horizontal line of $-\alpha\frac{\pi}{2}$;
- Nyquist: straight line at argument $-\alpha\frac{\pi}{2}$.

(b) Closed loop:

- Gain margin: A_m = infinite;
- Phase margin: constant : $\Phi_m = \pi\left(1 - \frac{\alpha}{2}\right)$;
- Step response: $y(t) = Kt^{\alpha} E_{\alpha,\alpha+1}(-Kt^{\alpha})$, where $E_{a,b}(z)$ is Mittag-Leffler function of two parameters

Bode's ideal loop transfer function

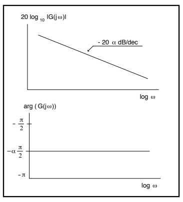
Can be used as a reference system in the form:



Bode's ideal loop.

$$G_c(s) = \frac{K}{s^{\alpha} + K}$$

$$G_o(s) = \frac{K}{s^{\alpha}}$$



Bode's ideal loop transfer function: example

The transfer function of a DC motor is

$$G(s) = \frac{K_m}{Js(s+1)}$$

J is payload inertia

Assume that we would like to have a closed loop system that is insensitive to gain variations with a constant phase margin of 60°. Bode's ideal loop transfer function that gives this phase margin is

$$G_o(s) = \frac{1}{s\sqrt[3]{s}}$$

Bode's ideal loop transfer function: example

Since $G_o(s) = C(s)G(s)$, the controller transfer function is

$$C(s) = \frac{J}{K_m}\left(s^{2/3} + \frac{1}{s^{1/3}}\right)$$

this is a particular case of fractional PID

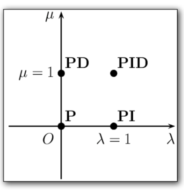
The obtained phase margin is 60° :

$$\Phi_m = \arg[C(j\omega)G(j\omega)] + \pi = \arg\left[\frac{1}{(j\omega)^{4/3}}\right] + \pi = \pi - \frac{4}{3}\frac{\pi}{2} = \frac{\pi}{3}.$$

Step response:

$$y(t) = L^{-1}\left\{\frac{1}{s\left(s^{1+1/3}+1\right)}\right\} = t^{1+1/3}E_{1+1/3,\,2+1/3}\left(-t^{1+1/3}\right)$$

PI$^{\lambda}$D$^{\mu}$ controllers

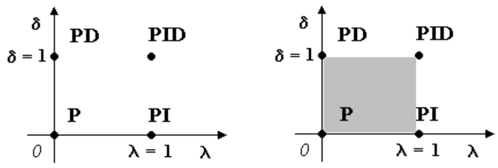


$$G_c(s) = \frac{U(s)}{E(s)} = K_p + K_I s^{-\lambda} + K_D s^{\mu}$$

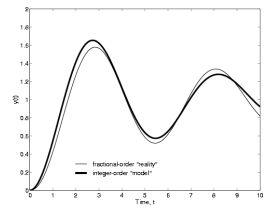
$$u(t) = K_p e(t) + K_I D^{-\lambda} e(t) + K_D D^{\mu} e(t)$$

PI$^{\lambda}$D$^{\mu}$ controllers

Fractional-order PID controllers:
from points to plane

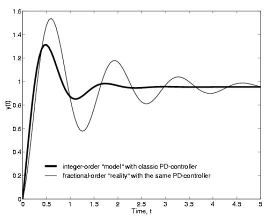


PI$^{\lambda}$D$^{\mu}$ controllers



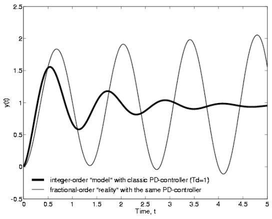
Comparison of unit step responses of the fractional-order "reality" and its integer-order "model"

PI$^{\lambda}$D$^{\mu}$ controllers



Control of the "reality" and the "model" using a classical PD controller, which is optimal for the "model"

$PI^{\lambda}D^{\mu}$ controllers



Control of the “reality” and the “model” using a classical PD controller, which is optimal for the “model”,

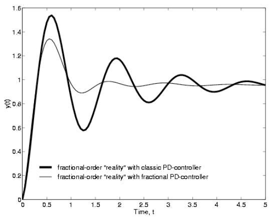
after detuning (aging) of the controller (when $T_D = 1$).

$PI^{\lambda}D^{\mu}$ controllers: design

The design of $PI^{\lambda}D^{\mu}$ controllers can be based on gain and phase margin specifications:

$$\begin{cases} \Re\left[C\left(j\omega_p\right)\right]\Re\left[P\left(j\omega_p\right)\right] - \Im\left[C\left(j\omega_p\right)\right]\Im\left[P\left(j\omega_p\right)\right] = -\frac{1}{A_m}, \\ \Re\left[C\left(j\omega_p\right)\right]\Im\left[P\left(j\omega_p\right)\right] + \Im\left[C\left(j\omega_p\right)\right]\Re\left[P\left(j\omega_p\right)\right] = 0, \\ \Re\left[C\left(j\omega_g\right)\right]\Re\left[P\left(j\omega_g\right)\right] - \Im\left[C\left(j\omega_g\right)\right]\Im\left[P\left(j\omega_g\right)\right] = -\cos\Phi_m, \\ \Re\left[C\left(j\omega_g\right)\right]\Im\left[P\left(j\omega_g\right)\right] + \Im\left[C\left(j\omega_g\right)\right]\Re\left[P\left(j\omega_g\right)\right] = -\sin\Phi_m, \end{cases}$$

$PI^{\lambda}D^{\mu}$ controllers



Control of the „reality“ using the PD controller, which is optimal for the „model“, and using the PD^{μ} controller.

$PI^{\lambda}D^{\mu}$ controllers: design

- The plant model is assumed to be

$$G(s) = \frac{1}{a_1 s^{\alpha} + a_2 s^{\beta} + a_3}$$

- and the fractional order PID controller is

$$G_c(s) = K_P + \frac{K_I}{s^{\lambda}} + K_D s^{\mu}$$

- It is expected that the gain and phase margin of the compensated systems are A_m and ϕ_m
- Question: how to choose $PI^{\lambda}D^{\mu}$ parameters

$PI^{\lambda}D^{\mu}$ controllers: design

Design of fractional-order PID controller parameters is determined by given requirements. These requirements can be, for example:

- the damping ratio,
- the steady-state error,
- dynamical properties,
- etc.

$PI^{\lambda}D^{\mu}$ controllers: design

Requiring $|G_c(j\omega_g)G_p(j\omega_g)| = 1, \quad \arg[G_c(j\omega_p)G_p(j\omega_p)] = -\pi$
it can be found that

$$K_P + \frac{K_I}{\omega_p^{\lambda}}\cos\frac{\pi\lambda}{2} + K_D\omega_p^{\mu}\cos\frac{\pi\mu}{2} = -\frac{a_1}{A_m}\omega_p^{\alpha}\cos\frac{\pi\alpha}{2} - \frac{a_2}{A_m}\omega_p^{\beta}\cos\frac{\pi\beta}{2} - \frac{a_3}{A_m}$$

$$K_P + \frac{K_I}{\omega_g^{\lambda}}\cos\frac{\pi\lambda}{2} + K_D\omega_g^{\mu}\cos\frac{\pi\mu}{2} = -a_1\omega_g^{\alpha}\cos\left(\frac{\pi\alpha}{2} + \phi_m\right) - a_2\omega_g^{\beta}\cos\left(\frac{\pi\beta}{2} + \phi_m\right) - a_3\cos\phi_m$$

$$K_P + \frac{K_I}{\omega_g^{\lambda}}\cos\frac{\pi\lambda}{2} + K_D\omega_g^{\mu}\cos\frac{\pi\mu}{2}$$
$$= -a_1\omega_g^{\alpha}\cos\left(\frac{\pi\alpha}{2} + \phi_m\right) - a_2\omega_g^{\beta}\cos\left(\frac{\pi\beta}{2} + \phi_m\right) - a_3\cos\phi_m$$

$$-K_I\frac{\sin\frac{\pi\lambda}{2}}{\omega_g^{\lambda}} + K_D\sin\frac{\pi\mu}{2}\omega_g^{\mu} = -a_1\omega_g^{\alpha}\sin\left(\frac{\pi\alpha}{2} + \phi_m\right) - a_2\omega_g^{\beta}\sin\left(\frac{\pi\beta}{2} + \phi_m\right) - a_3\sin\phi_m$$

PI$^\lambda$D$^\mu$ controllers: design

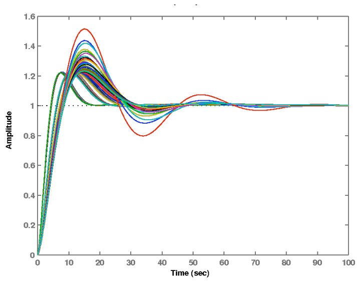
We have four equations with seven variables:

$$(\omega_p,\ \omega_g,\ \lambda,\ \mu,\ K_I,\ K_P,\ K_D)$$

The rest of the variables can be determined by minimizing the ISE criterion $\int_0^\infty e^2(t)\mathrm{d}t$

PI$^\lambda$D$^\mu$ controllers: design

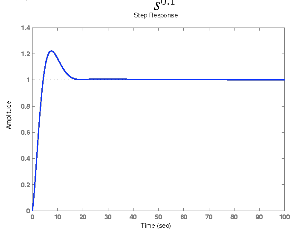
Sample plots for various combinations of λ and μ:



PI$^\lambda$D$^\mu$ controllers: design

The optimal PI$^\lambda$D$^\mu$ controller:

$$G_{Fc}(s) = 822.8093 + \frac{9.1806}{s^{0.1}} + 74.9213s^{0.5}$$



PI$^\lambda$D$^\mu$ control: MATLAB

I. General Description of Linear Fractional Order Systems

I.1 The normal form

$$G(s) = \frac{b_1 s^{\gamma_1} + b_2 s^{\gamma_2} + \cdots + b_m s^{\gamma_m}}{a_1 s^{\eta_1} + a_2 s^{\eta_2} + \cdots + a_{n-1} s^{\eta_{n-1}} + a_n s^{\eta_n}}$$

- Thus compared with IO LTI's, information on orders are also used
- A FOTF model class/object can be defined in MATLAB to describe the system model

Courtesy: Dingyü Xue, YangQuan Chen

PI$^\lambda$D$^\mu$ control: MATLAB

II.1 Create a @fotf directory

- The fotf class can be defined as

```
function G=fotf(a,na,b,nb)
if nargin==0,
   G.a=[]; G.na=[]; G.b=[]; G.nb=[];
   G=class(G,'fotf');
elseif isa(a,'fotf'),
   G=a;
else,
   G.a=a; G.na=na; G.b=b; G.nb=nb;
   G=class(G,'fotf');
end
```

Courtesy: Dingyü Xue, YangQuan Chen

PI$^\lambda$D$^\mu$ control: MATLAB

- Function call `G=fotf(a,η,b,γ)`
- A display function

```
function display(G)
strN=polydisp(G.b,G.nb); strD=polydisp(G.a,G.na);
nm=max([length(strN),length(strD)]); nn=length(strN); nd=length(strD);
disp([char(' '*ones(1,floor((nm-nn)/2))) strN]), disp(char('-'*ones(1,nm)));
disp([char(' '*ones(1,floor((nm-nd)/2))) strD])
function strP=polydisp(p,np)
P=''; [np,ii]=sort(np,'descend'); p=p(ii);
for i=1:length(p)
   P=[P,'+',num2str(p(i)),'s^{',num2str(np(i)),'}'];
end
P=P(2:end); P=strrep(P,'s^{0}',''); P=strrep(P,'+-','-');
P=strrep(P,'^{1}',''); P=strrep(P,'+1s','+s'); strP=strrep(P,'-1s','-s');
if length(strP)>=2, if strP(1:2)=='1s', strP=strP(2:end); end,end,
```

Courtesy: Dingyü Xue, YangQuan Chen

PI$^{\lambda}$D$^{\mu}$ control: MATLAB

- Example

$$G(s) = \frac{-2s^{0.63} - 4}{2s^{3.501} + 3.8s^{2.42} + 2.6s^{1.798} + 2.5s^{1.31} + 1.5}$$

- MATLAB command

```
>> b=[-2,-4]; nb=[0.63,0]; a=[2 3.8 2.6 2.5 1.5];
   na=[3.501,2.42,1.798,1.31,0]; G=fotf(a,na,b,nb)
```

- Display

```
                    -2s^{0.63}-4
---------------------------------------------------
2s^{3.501}+3.8s^{2.42}+2.6s^{1.798}+2.5s^{1.31}+1.5
```

Courtesy: Dingyü Xue, YangQuan Chen

PI$^{\lambda}$D$^{\mu}$ control: MATLAB

- Feedback function

```
function G=feedback(F,H)
b=kron(F.b,H.a); na=[]; nb=[];
a=[kron(F.b,H.b), kron(F.a,H.a)];
for i=1:length(F.b),
   nb=[nb F.nb(i)+H.nb]; na=[na,F.nb(i)+H.nb];
end
for i=1:length(F.a), na=[na F.na(i)+H.na]; end
G=unique(fotf(a,na,b,nb));
```

- These functions are suitable for interconnections of fractional order systems

Courtesy: Dingyü Xue, YangQuan Chen

PI$^{\lambda}$D$^{\mu}$ control: MATLAB

- Plus function for parallel connections

```
function G=plus(G1,G2)
a=kron(G1.a,G2.a); na=[]; nb=[];
b=[kron(G1.a,G2.b), kron(G1.b,G2.a)];
for i=1:length(G1.a),
   na=[na G1.na(i)+G2.na];
   nb=[nb, G1.na(i)+G2.nb];
end
for i=1:length(G1.b),
   nb=[nb G1.nb(i)+G2.na];
end
G=unique(fotf(a,na,b,nb));
```

Courtesy: Dingyü Xue, YangQuan Chen

PI$^{\lambda}$D$^{\mu}$ control: MATLAB

- A common function unique

```
function G=unique(G)
[a,n]=polyuniq(G.a,G.na); G.a=a; G.na=n;
[a,n]=polyuniq(G.b,G.nb); G.b=a; G.nb=n;
function [a,an]=polyuniq(a,an)
[an,ii]=sort(an,'descend'); a=a(ii);
ax=diff(an); key=1;
for i=1:length(ax)
   if ax(i)==0,
      a(key)=a(key)+a(key+1); a(key+1)=[];
      an(key+1)=[];
   else, key=key+1; end
end
```

Courtesy: Dingyü Xue, YangQuan Chen

PI$^{\lambda}$D$^{\mu}$ control: MATLAB

- Multiplication for series connection

```
function G=mtimes(G1,G2)
a=kron(G1.a,G2.a); b=kron(G1.b,G2.b); na=[]; nb=[];
for i=1:length(G1.na), na=[na,G1.na(i)+G2.na]; end
for i=1:length(G1.nb), nb=[nb,G1.nb(i)+G2.nb]; end
G=unique(fotf(a,na,b,nb));
```

- Minus

```
function G=minus(G1,G2)
G=G1+(-G2);
```

- Uminus

```
function G=uminus(G1)
G=fotf(G1.a,G1.na,-G1.b,G1.nb);
```

- Inv

```
function G=inv(G1)
G=fotf(G1.b,G1.nb,G1.a,G1.na);
```

Courtesy: Dingyü Xue, YangQuan Chen

PI$^{\lambda}$D$^{\mu}$ control: MATLAB

Example of interconnection:

- Unity negative feedback system with

$$G(s) = \frac{0.8s^{1.2} + 2}{1.1s^{1.8} + 0.8s^{1.3} + 1.9s^{0.5} + 0.4}$$

$$G_c(s) = \frac{1.2s^{0.72} + 1.5}{3s^{0.8}}$$

- The closed-loop system

```
>> G=fotf([1.1,0.8 1.9 0.4],[1.8 1.3 0.5 0],[0.8 2],[1.2 0]);
   Gc=fotf([3],[0.8], [1.2 1.5],[0.72 0]); H=fotf(1,0,1,0);
   GG=feedback(G*Gc,H)
```

$$G(s) = \frac{0.96s^{1.92} + 1.2s^{1.2} + 2.4s^{0.72} + 3}{3.3s^{2.6} + 2.4s^{2.1} + 0.96s^{1.92} + 5.7s^{1.3} + 1.2s^{1.2} + 1.2s^{0.8} + 2.4s^{0.72} + 3}$$

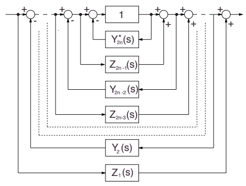
Continued fractions (CFE)

$$G(s) \simeq a_0(s) + \cfrac{b_1(s)}{a_1(s) + \cfrac{b_2(s)}{a_2(s) + \cfrac{b_3(s)}{a_3(s)+\ldots}}}$$

$$= a_0(s) + \frac{b_1(s)}{a_1(s)+}\,\frac{b_2(s)}{a_2(s)+}\,\frac{b_3(s)}{a_3(s)+}\,\ldots$$

CFEs and nested multiple loops

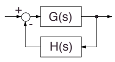
$$Z(s) = Z_1(s) + \cfrac{1}{Y_2(s) + \cfrac{1}{Z_3(s) + \cfrac{1}{Y_4(s) + \cfrac{1}{\cdots\cdots\cdots \cfrac{1}{Y_{2n-2}(s) + \cfrac{1}{Z_{2n-1}(s) + \cfrac{1}{Y_{2n}(s)}}}}}}}$$



Nested multiple-loop control system of the first type.

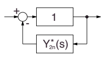
Nested loop of type I

CFEs and nested multiple loops



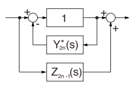
$$R(s) = \frac{G(s)}{1 + G(s)H(s)}.$$

A control loop with a negative feedback.



$$P_{2n}(s) = \frac{1}{1 + 1 \cdot Y_{2n}^{*}(s)} = \frac{1}{Y_{2n}(s)}$$

Nested multiple-loop control system – level 1.

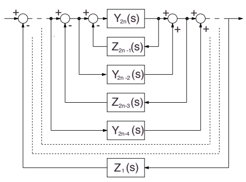


$$Q_{2n-1}(s) = Z_{2n-1}(s) + P_{2n}(s) = Z_{2n-1}(s) + \frac{1}{Y_{2n}(s)}.$$

Nested multiple-loop control system – level 2.

CFEs and nested multiple loops

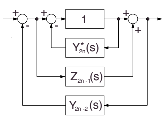
$$Z(s) = \cfrac{1}{Z_1(s) + \cfrac{1}{Y_2(s) + \cfrac{1}{Z_3(s) + \cfrac{1}{\cdots\cdots\cdots \cfrac{1}{Y_{2n-2}(s) + \cfrac{1}{Z_{2n-1}(s) + \cfrac{1}{Y_{2n}(s)}}}}}}}$$



Nested multiple-loop control system of the second type.

Nested loop of type II

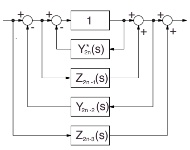
CFEs and nested multiple loops



$$P_{2n-2}(s) = \frac{Q_{2n-1}(s)}{1 + Q_{2n-1}(s)Y_{2n-2}(s)} = \cfrac{1}{Y_{2n-2}(s) + \cfrac{1}{Q_{2n-1}(s)}}$$

$$= \cfrac{1}{Y_{2n-2}(s) + \cfrac{1}{Z_{2n-1}(s) + \cfrac{1}{Y_{2n}(s)}}}$$

Nested multiple-loop control system – level 3.



$$Q_{2n-3}(s) = Z_{2n-3}(s) + P_{2n-2}(s)$$

$$= Z_{2n-3}(s) + \cfrac{1}{Y_{2n-2}(s) + \cfrac{1}{Z_{2n-1}(s) + \cfrac{1}{Y_{2n}(s)}}}$$

Nested multiple-loop control system – level 4.

General CFE method

A rational approximation for $G(s) = s^{-\alpha}, 0 < \alpha < 1$ can be obtained by CFE of the following expressions:

$$H_h(s) = \frac{1}{(1+sT)^{\alpha}} \qquad \text{high frequencies } (\omega T >> 1)$$

$$H_l(s) = \left(1 + \frac{1}{s}\right)^{\alpha} \qquad \text{low frequencies } (\omega << 1)$$

Example:

Performing the CFE of the function $H_h(s)$, with $T = 1$, $\alpha = 0.5$, we obtain:

$$H_1(s) = \frac{0.3513s^4 + 1.405s^3 + 0.8433s^2 + 0.1574s + 0.008995}{s^4 + 1.333s^3 + 0.478s^2 + 0.064s + 0.002844}$$

General Approach to Fractances

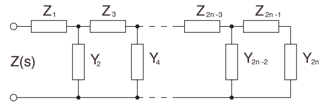
A devices or a circuit exhibiting fractional-order behaviour is called a fractance.

- domino ladder circuit network,
- a tree structure of electrical elements,
- transmission line circuit

Design of fractances can be using a truncated CFE, which gives a rational approximation.

General Approach to Fractances

Domino ladder circuit



$$Z(s) = Z_1(s) + \cfrac{1}{Y_2(s) + \cfrac{1}{Z_3(s) + \cfrac{1}{Y_4(s) + \cfrac{1}{\cdots\cdots\cdots \cfrac{1}{Y_{2n-2}(s) + \cfrac{1}{Z_{2n-1}(s) + \cfrac{1}{Y_{2n}(s)}}}}}}$$

General Approach to Fractances

S. C. Dutta Roy on Khovanskii's CFE for $x^{1/2}$:

> "... if x is replaced by the complex frequency variable s, then the realization would require a negative resistance. Thus, the [Khovanskii's] CFEs do not seem to be useful for realization of fractional inductor or capacitor."

However, the possibility of realization of negative impedances in electric circuits has been pointed out by H.W. Bode in 1945.

General Approach to Fractances

Example 1: design a domino ladder circuit with

$$Z(s) = \frac{s^4 + 4s^2 + 1}{s^3 + s}$$

Find a CFE: $Z(s) = \dfrac{s^4 + 4s^2 + 1}{s^3 + s} = s + \cfrac{1}{\cfrac{1}{3}s + \cfrac{1}{\cfrac{9}{2}s + \cfrac{1}{\cfrac{2}{3}s}}}$

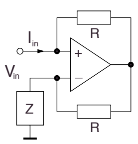
Therefore, $Z_1(s) = s; \quad Z_3(s) = \frac{9}{2}s; \quad Y_2(s) = \frac{1}{3}s; \quad Y_4(s) = \frac{2}{3}s$

For the first Cauer's canonic LC circuits we must take

$$L_1 = 1\ [H]; \quad L_3 = \frac{9}{2}\ [H]; \quad C_2 = \frac{1}{3}\ [F]; \quad C_4 = \frac{2}{3}\ [F].$$

General Approach to Fractances

Negative impedance converters are available:



Negative-impedance converter.

General Approach to Fractances

Example II: design a domino ladder circuit with

$$Z(s) = \frac{s^4 + 3s^2 + 8}{2s^3 + 4s}$$

We have: $Z(s) = \dfrac{s^4 + 3s^2 + 8}{2s^3 + 4s} = \dfrac{1}{2}s + \cfrac{1}{2s + \cfrac{1}{-\cfrac{1}{12}s + \cfrac{1}{-\cfrac{3}{2}s}}}$

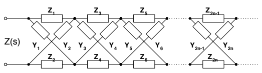
$$Z_1(s) = \frac{1}{2}s; \quad Z_3(s) = -\frac{1}{12}s; \quad Y_2(s) = 2s; \quad Y_4(s) = -\frac{3}{2}s.$$

$$L_1 = \frac{1}{2}\ [H]; \quad L_3 = -\frac{1}{12}\ [H]; \quad C_2 = 2\ [F]; \quad C_4 = -\frac{3}{2}\ [F].$$

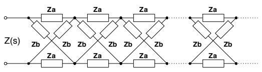
Notice negative values - can be done using OpAmps

General Approach to Fractances

Transmission lines circuit



General structure of transmission line.



Transmission line circuit composed of two inpedance Z_a and Z_b.

$$Z(s) = \sqrt{Z_a . Z_b}$$

Take

$$Z_a = R$$
$$Z_b = 1/sC$$

Then

$$Z(s) = \sqrt{\frac{R}{C}}\, s^{-1/2}$$

General Approach to Fractances

Sample implementations:

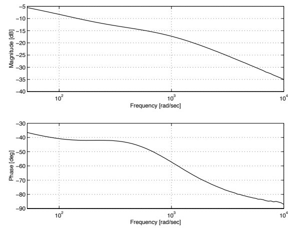
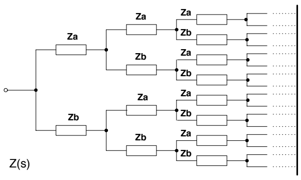


Figure 7.5: Bode plots of the $I^{1/2}$ controller where half order integral was approximated by transmission line depicted in Fig. 7.2.

General Approach to Fractances

Tree structure circuit



$$Z(s) = \sqrt{Z_a . Z_b}$$

Take

$$Z_a = R$$
$$Z_b = 1/sC$$

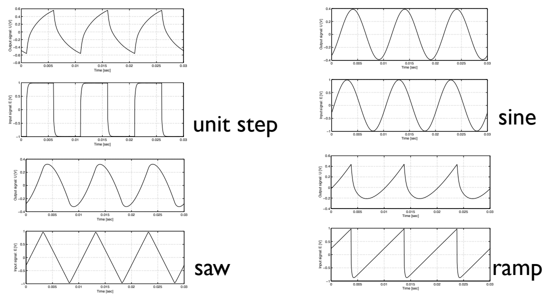
Then

$$Z(s) = \sqrt{\frac{R}{C}}\, s^{-1/2} = \sqrt{\frac{R}{C}}\, \omega^{-1/2} e^{-j\pi/4}|_{s=j\omega},$$

phase angle is constant $-\pi/4$, independent of the frequency

General Approach to Fractances

Sample implementations -- transmission lines -- responses:



General Approach to Fractances

Sample implementations:

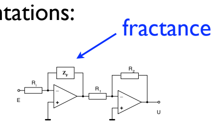
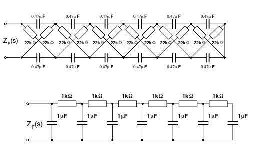
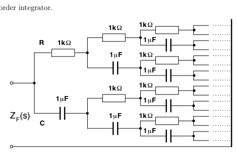


Figure 7.1: Analogue fractional-order integrator.





Stability of fractional order systems

Commensurate order systems:

$$G(s) = K \frac{\sum_{k=0}^{m} b_k (s^{\alpha})^k}{\sum_{k=0}^{n} a_k (s^{\alpha})^k} = K \frac{N(s^{\alpha})}{D(s^{\alpha})}$$

For example:

$$G(s) = \frac{1}{s^{2/3} - s^{1/2} + 1/2}$$

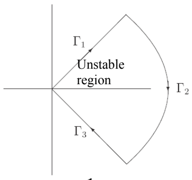
can be written as

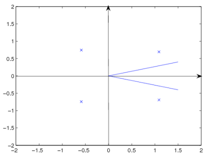
$$G(\lambda) = \frac{1}{\lambda^4 - \lambda^3 + 1/2} \qquad \lambda = s^{1/6}$$

Stability of fractional order systems

- The Γ curve
  - if $\lambda = s^{\alpha}$, then the stable condition for the system is

$$|\arg(\lambda_i)| > \alpha\pi/2$$





$$G(\lambda) = \frac{1}{\lambda^4 - \lambda^3 + 1/2}$$

with four poles in the stable region