Control Lab
www.a-lab.ee

TALLINN UNIVERSITY OF TECHNOLOGY

ISS0023 Intelligent Control Systems

Fractional-order Calculus based Modeling and Control of Dynamic Systems

Aleksei Tepljakov, Ph.D.

December 1, 2017

Lecture overview

- Mathematical basis of fractional-order calculus;
- Fractional-order calculus in modeling and control:
  - Analysis of fractional models;
  - Implementations of fractional-order systems;
  - $\text{PI}^{\lambda}\text{D}^{\mu}$ controllers and their design.
- Overview of CACSD tools and examples of practical applications:
  - Introduction to FOMCON toolbox for MATLAB;
  - Control design and implementation examples.

Part I: Mathematical Basis of Fractional-order Calculus

Introduction: Historical facts

- The concept of the differentiation operator $\mathscr{D} = \mathrm{d}/\mathrm{d}\,x$ is a well-known fundamental tool of modern calculus. For a suitable function f the n-th derivative is well defined as

$$\mathscr{D}^n f(x) = \mathrm{d}^{\,n} f(x)/\mathrm{d}x^n, \tag{1}$$

  where n is a positive integer.
- What happens if we extend this concept to a situation, when the order of differentiation is arbitrary, for example, fractional?
- That was the very same question L'Hôpital addressed to Leibniz in a letter in 1695. Since then the concept of fractional calculus has drawn the attention of many famous mathematicians, including Euler, Laplace, Fourier, Liouville, Riemann, Abel.

Fractional derivative of a power function: An approach based on intuition

For the power function $f(x) = x^k$ the fractional derivative can be shown to be

$$\frac{\mathrm{d}^\alpha f(x)}{\mathrm{d}x^\alpha} = \frac{\Gamma(k+1)}{\Gamma(k-\alpha+1)} x^{k-\alpha}. \tag{2}$$

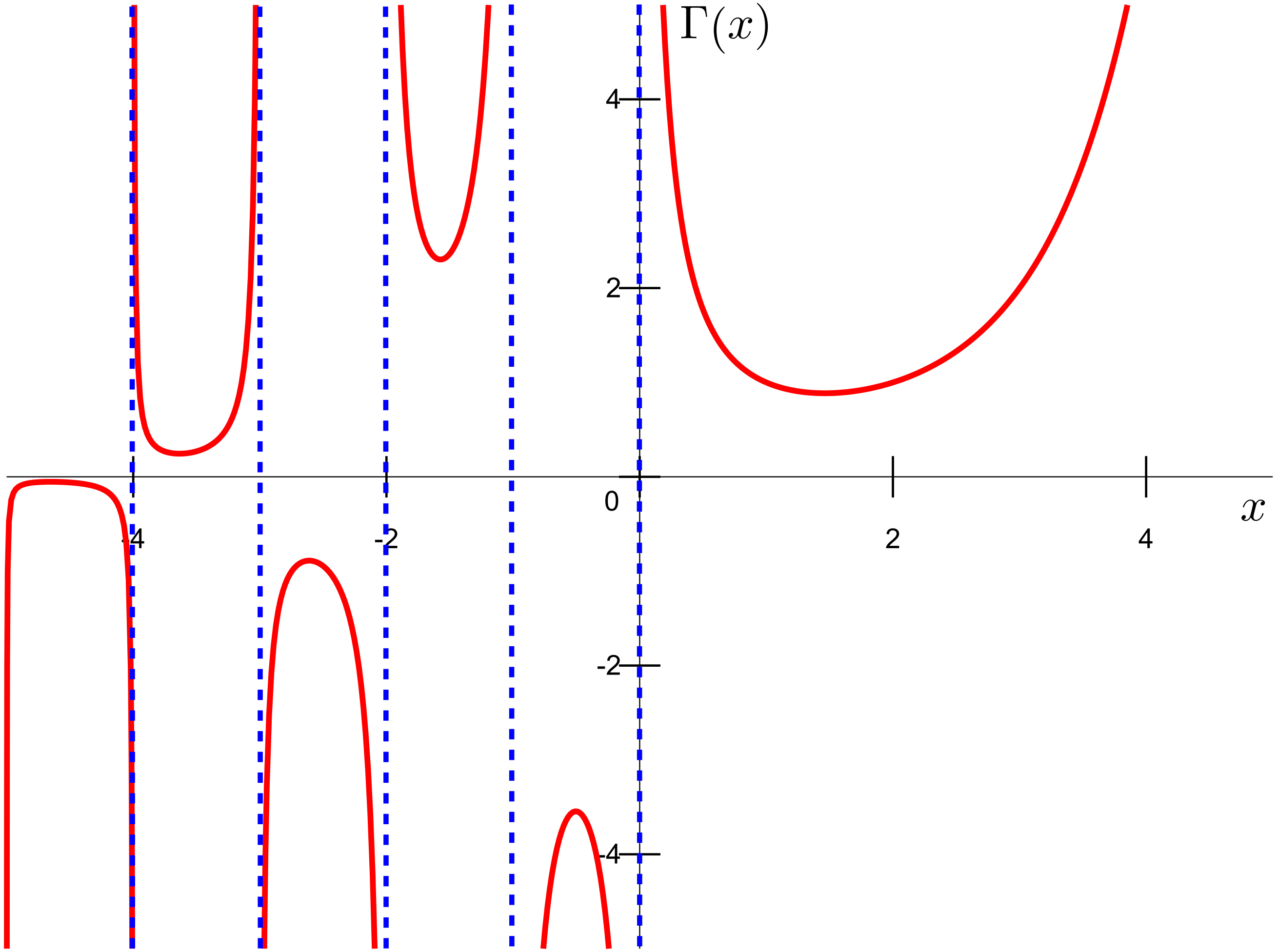
The function $\Gamma(\cdot)$ above is the Gamma function—the generalization of the factorial function:

$$\Gamma(x) = \int_0^\infty t^{x-1} \mathrm{e}^{-t} \mathrm{d}t, \quad x > 0. \tag{3}$$

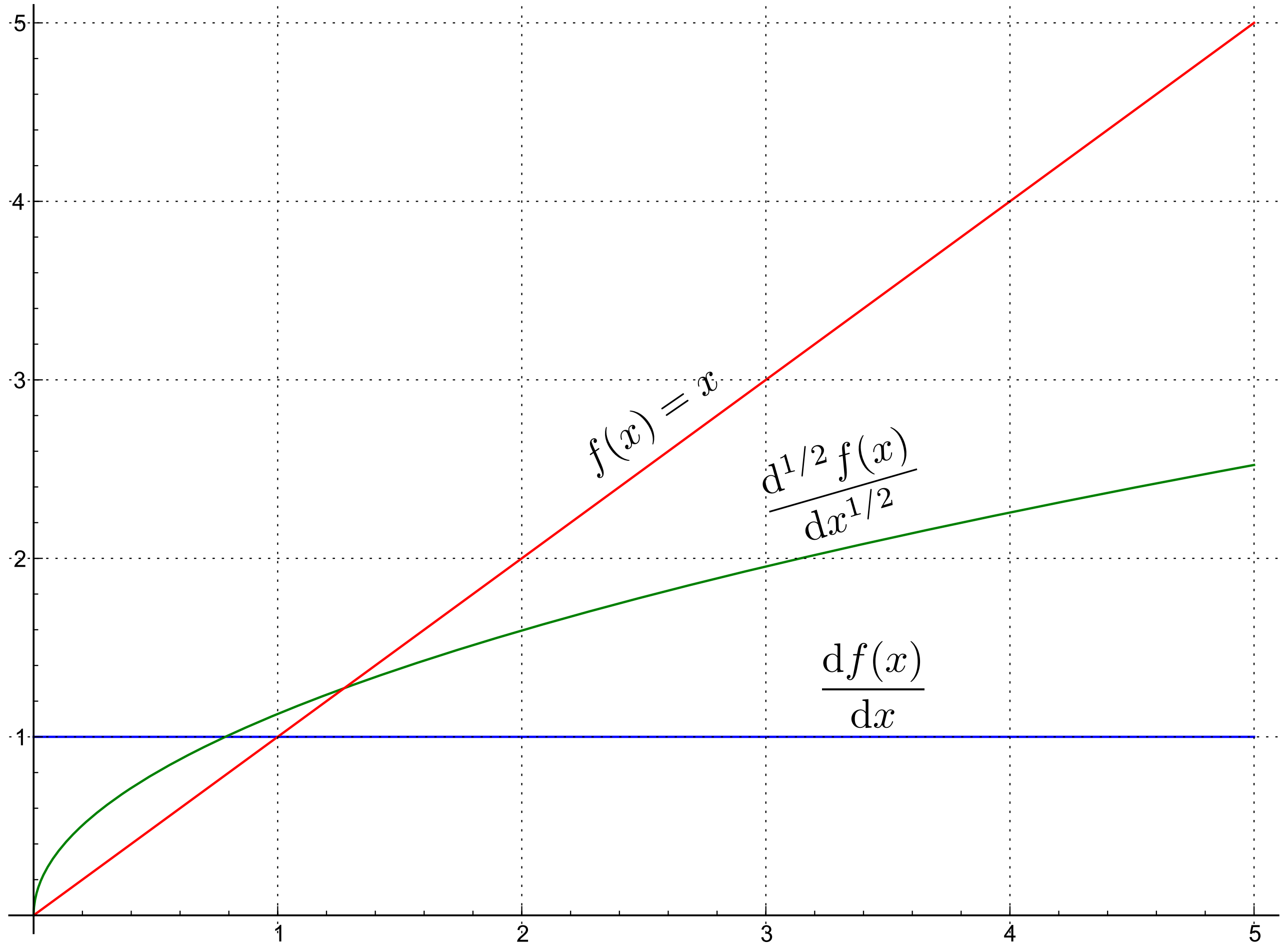
Example:

$$\frac{\mathrm{d}^{1/2}(x^2)}{\mathrm{d}x^{1/2}} = \frac{\Gamma(3)}{\Gamma(5/2)} x^{3/2} = \frac{8x^{3/2}}{3\sqrt{\pi}}.$$

The Gamma function



Example: fractional-order derivative of a function $f(x) = x$



Fractional derivative of a trigonometric function: An approach based on intuition

We observe, what happens when we repeatedly differentiate the function $f(x) = \sin x$:

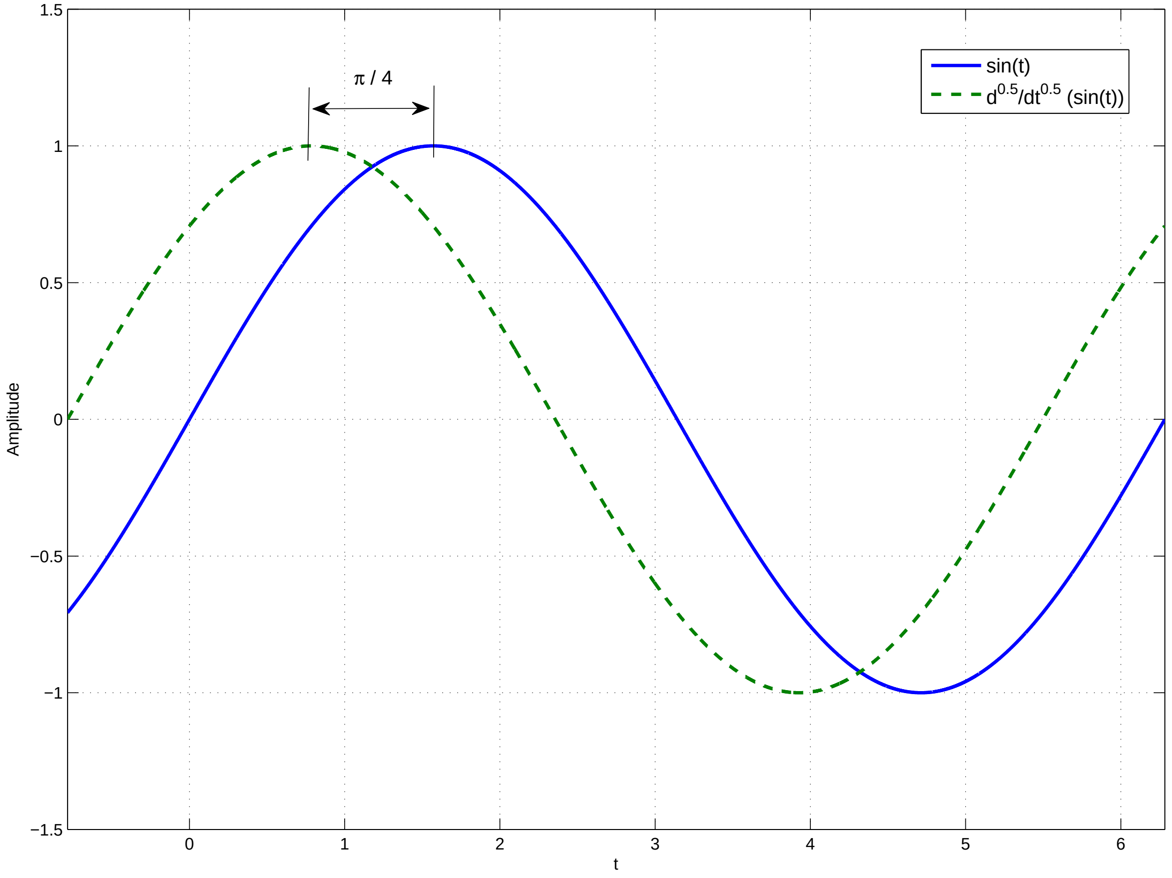
$$\frac{\mathrm{d}}{\mathrm{d}x}\sin x = \cos x,\ \frac{\mathrm{d}^2}{\mathrm{d}x^2}\sin x = -\sin x,\ \frac{\mathrm{d}^3}{\mathrm{d}x^3}\sin x = -\cos x,\ \ldots$$

The pattern can be deduced: for the nth derivative, the function $\sin x$ is shifted by $n\pi/2$ radians. This can be observed from studying the graph of the function. Thus, if we replace n by $\alpha \in \mathbb{R}_+$, we have

$$\frac{\mathrm{d}^\alpha}{\mathrm{d}x^\alpha}\sin x = \sin\left(x + \frac{\alpha\pi}{2}\right). \tag{4}$$

Obviously, a similar equation holds for the cosine function as well.

Half derivative of a sine function



Repeated differentiation: Backward difference equation

Recall the backward difference definition of $f'(x)$ given by

$$f'(x) = \lim_{h\to 0} \frac{f(x) - f(x-h)}{h}. \tag{5}$$

It follows, that

$$f''(x) = \lim_{h\to 0} \frac{f'(x) - f'(x-h)}{h} = \lim_{h\to 0} \frac{f(x) - 2f(x-h) + f(x-2h)}{h^2}.$$

Furthermore,

$$f'''(x) = \lim_{h\to 0} \frac{f(x) - 3f(x-h) + 3f(x-2h) - f(x-3h)}{h^3}.$$

And in general

$$f^{(n)}(x) = \lim_{h\to 0} \frac{1}{h^n} \sum_{k=0}^{n} (-1)^k \binom{n}{k} f(x-kh). \tag{6}$$

Repeated differentiation: Backward difference equation based generalization

Can we generalize this to the case $n \in \mathbb{R}_+$?

Of course! All we need to do is to consider the factorial formula for the binomial coefficient and use the ever so kind Gamma function to lend a helping hand in case we have $\alpha \in \mathbb{R}_+$:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad \rightarrow \quad \binom{\alpha}{k} = \frac{\Gamma(\alpha+1)}{\Gamma(k+1)\Gamma(\alpha-k+1)}. \tag{7}$$

We find that this approach is the very basis for Grünwald-Letnikov's definition of the fractional-order derivative. In fact, here it is:

Definition 1. *(Grünwald-Letnikov)*

$${}^{GL}\mathscr{D}^{\alpha} f(t)|_{t=nh} = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{k=0}^{n} (-1)^k \binom{\alpha}{k} f(nh - kh). \tag{8}$$

Fractional-order derivative: Important alternative definitions

Definition 2. *(Riemann-Liouville)*

$$ {}_a^R\mathscr{D}_t^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)} \left(\frac{\mathrm{d}}{\mathrm{d}t}\right)^m \left[\int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-m+1}}\,\mathrm{d}\tau\right], \tag{9} $$

where $m-1<\alpha<m$, $m\in\mathbb{N}$,$\alpha\in\mathbb{R}_+$.

Definition 3. *(Caputo)*

$$ {}_0^C\mathscr{D}_t^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)} \int_0^t \frac{f^{(m)}(\tau)}{(t-\tau)^{\alpha-m+1}}\mathrm{d}\tau, \tag{10} $$

where $m-1<\alpha<m$, $m\in\mathbb{N}$.

The generalized operator

Fractional calculus is a generalization of integration and differentiation to non-integer order operator ${}_a\mathscr{D}_t^\alpha$, where a and t denote the limits of the operation and α denotes the fractional order such that

$$
{}_a\mathscr{D}_t^\alpha = \begin{cases} \frac{\mathrm{d}^\alpha}{\mathrm{d}t^\alpha} & \Re(\alpha) > 0, \\ 1 & \Re(\alpha) = 0, \\ \int_a^t (\mathrm{d}\tau)^{-\alpha} & \Re(\alpha) < 0, \end{cases} \tag{11}
$$

where generally it is assumed that $\alpha \in \mathbb{R}$, but it may also be a complex number. We restrict our attention to the former case.

Properties of fractional-order differentiation

Fractional-order differentiation has the following properties:

1. If $\alpha = n$ and $n \in \mathbb{Z}_+$, then the operator ${}_0\mathscr{D}_t^\alpha$ can be understood as the usual operator $\mathrm{d}^n/\mathrm{d}t^n$.
2. Operator of order $\alpha = 0$ is the identity operator: ${}_0\mathscr{D}_t^0 f(t) = f(t)$.
3. Fractional-order differentiation is linear; if a, b are constants, then

$$ {}_0\mathscr{D}_t^\alpha \left[af(t) + bg(t) \right] = a\,{}_0\mathscr{D}_t^\alpha f(t) + b\,{}_0\mathscr{D}_t^\alpha g(t). \tag{12} $$

4. If $f(t)$ is an analytic function, then the fractional-order differentiation ${}_0\mathscr{D}_t^\alpha f(t)$ is also analytic with respect to t.

Properties of fractional-order differentiation (continued)

5. For the fractional-order operators with $\Re(\alpha) > 0, \Re(\beta) > 0$, and under reasonable constraints on the function $f(t)$ it holds the additive law of exponents:

$$ {}_0\mathscr{D}_t^{\alpha}\left[{}_0\mathscr{D}_t^{\beta} f(t)\right] = {}_0\mathscr{D}_t^{\beta}\left[{}_0\mathscr{D}_t^{\alpha} f(t)\right] = {}_0\mathscr{D}_t^{\alpha+\beta} f(t) \qquad (13) $$

6. The fractional-order derivative commutes with integer-order derivative

$$ \frac{\mathrm{d}^n}{\mathrm{d}t^n}\left({}_a\mathscr{D}_t^{\alpha} f(t)\right) = {}_a\mathscr{D}_t^{\alpha}\left(\frac{\mathrm{d}^n f(t)}{\mathrm{d}t^n}\right) = {}_a\mathscr{D}_t^{\alpha+n} f(t), \qquad (14) $$

and if $t = a$ we have $f^{(k)}(a) = 0,\ (k = 0,\ 1,\ 2,\ \ldots,\ n-1)$.

On the meaning of the fractional-order derivative

We shall call $\mathscr{F}(f_t(\cdot), t)$ a hereditary operator acting on a cause process $f_t(\cdot)$ to produce a time-shifted effect $g(t)$ which depends on the history of the process $\{f_t(\tau); \tau < t\}$:

$$g(t) = \mathscr{F}\left[f_t(\cdot); t\right]. \tag{15}$$

We can replace $g(t)$ by the function $f(t)$ or its derivatives, i.e.

$$\frac{\mathrm{d}f(t)}{\mathrm{d}t} = \mathscr{F}\left[f_t(\cdot); t\right] \tag{16}$$

and so on. (Again we see repeated differentiation/integration.)

Some hereditary process examples from physics: Brownian motion; Viscoelasticity; Heat transfer; Long transmission.

Exercise: Integration

Compute a fractional-order derivative of order $1/2$ for the function $f(t) = t^2$ using the Caputo definition. Hint: $\Gamma(1/2) = \sqrt{\pi}$.

$$ {}_0^C\mathscr{D}_t^{1/2} t^2 = \frac{1}{\Gamma(1-1/2)} \int_0^t \frac{(\tau^2)'}{(t-\tau)^{1/2-1+1}} \mathrm{d}\tau = \quad ? $$

Solution: Compute the indefinite integral

$$ \begin{aligned} \int \frac{(\tau^2)'}{(t-\tau)^{1/2-1+1}} \mathrm{d}\tau &= \int \frac{2\tau}{\sqrt{t-\tau}} \mathrm{d}\tau = \\ \overset{u=t-\tau}{=} 2\int \frac{u-t}{\sqrt{u}} \mathrm{d}u &= 2\int \sqrt{u}\mathrm{d}u - 2t\int 1/\sqrt{u}\mathrm{d}u = \\ 4/3 u^{3/2} - 4t\sqrt{u} + C &= 4/3 (t-\tau)^{3/2} - 4t\sqrt{t-\tau} + C. \end{aligned} $$

The answer is

$$ \frac{1}{\sqrt{\pi}} \cdot \left(4/3 (t-\tau)^{3/2} - 4t\sqrt{t-\tau} + C \right)\Big|_0^t = \frac{1}{\sqrt{\pi}} \cdot \left(-4/3 t^{3/2} + 4t^{3/2} \right) = \frac{8t^{3/2}}{3\sqrt{\pi}}. $$

Part II: Factional-order Modeling of Dynamic Systems

Laplace transform

A function $F(s)$ of the complex variable s is called the Laplace transform of the original function $f(t)$ and defined as

$$F(s) = \mathscr{L}\left[f(t)\right] = \int_0^\infty \mathrm{e}^{-st} f(t)\mathrm{d}t \tag{17}$$

The original function $f(t)$ can be recovered from the Laplace transform $F(s)$ by applying the inverse Laplace transform

$$f(t) = \mathscr{L}^{-1}\left[F(s)\right] = \frac{1}{j2\pi}\int_{c-j\infty}^{c+j\infty} \mathrm{e}^{st} F(s)\mathrm{d}s, \tag{18}$$

where c is greater than the real part of all the poles of $F(s)$.

Fractional-order derivative definitions: Laplace transform

Definition 4. *(Riemann-Liouville)*

$$\mathscr{L}\left[{}^{R}\mathscr{D}^{\alpha}f(t)\right] = s^{\alpha}F(s) - \sum_{k=0}^{m-1} s^{k}\left[\mathscr{D}^{\alpha-k-1}f(t)\right]_{t=0}. \tag{19}$$

Definition 5. *(Caputo)*

$$\mathscr{L}\left[{}^{C}\mathscr{D}^{\alpha}f(t)\right] = s^{\alpha}F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1}f^{(k)}(0). \tag{20}$$

Definition 6. *(Grünwald-Letnikov)*

$$\mathscr{L}\left[{}^{L}\mathscr{D}^{\alpha}f(t)\right] = s^{\alpha}F(s). \tag{21}$$

For the first two definitions we have $(m-1 \leqslant \alpha < m)$.

Fractional-order models

A linear, fractional-order continuous-time dynamic system can be expressed by a fractional differential equation of the following form

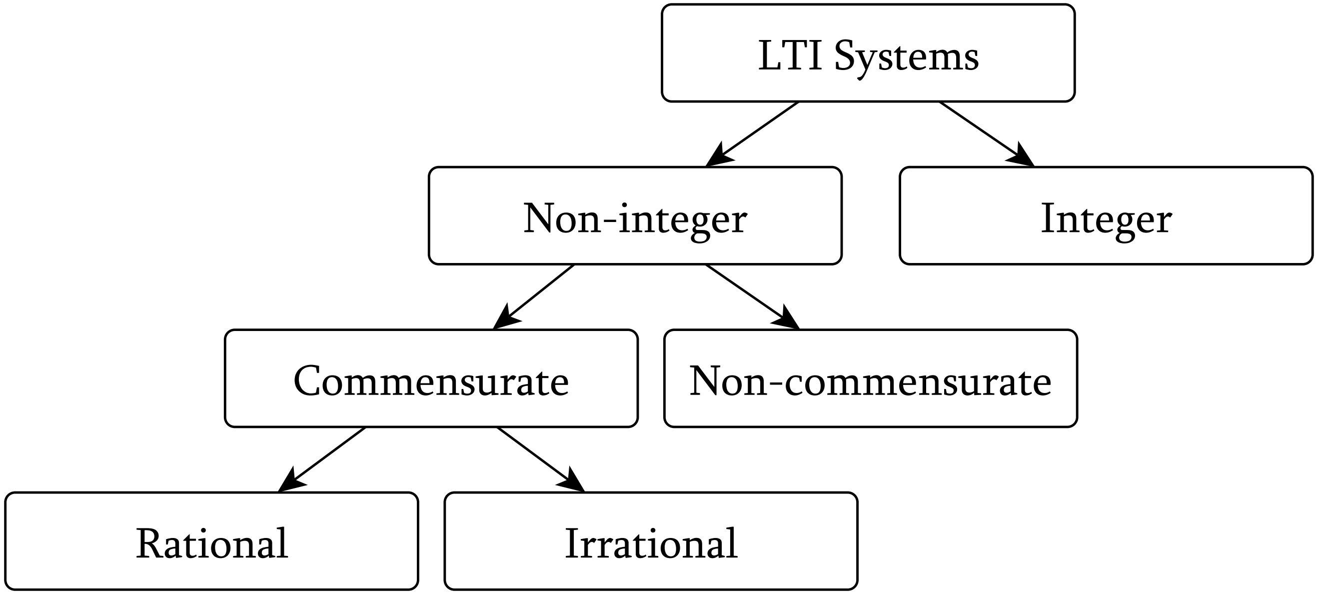
$$\begin{aligned} a_n \mathscr{D}^{\alpha_n} y(t) + a_{n-1} \mathscr{D}^{\alpha_{n-1}} y(t) + \cdots + a_0 \mathscr{D}^{\alpha_0} y(t) &= \\ b_m \mathscr{D}^{\beta_m} u(t) + b_{m-1} \mathscr{D}^{\beta_{m-1}} u(t) + \cdots + b_0 \mathscr{D}^{\beta_0} u(t), & \end{aligned} \tag{22}$$

where $a_k,\, b_k \in \mathbb{R}$. The system is said to be of *commensurate-order* if in (22) all the orders of derivation are integer multiples of a base order γ such that $\alpha_k,\, \beta_k = k\gamma,\, \gamma \in \mathbb{R}_+$. The system can then be expressed as

$$\sum_{k=0}^{n} a_k \mathscr{D}^{k\gamma} y(t) = \sum_{k=0}^{m} b_k \mathscr{D}^{k\gamma} u(t). \tag{23}$$

Linear, time invariant fractional-order system classification

If in (23) the order is $\gamma = 1/q$, $q \in \mathbb{Z}_+$, the system will be of rational order. The diagram with linear time-invariant (LTI) system classification is given in the following diagram.



Fractional-order transfer functions

Applying the Laplace transform to (22) with zero initial conditions the input-output representation of the fractional-order system can be obtained in the form of a transfer function:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \cdots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \cdots + a_0 s^{\alpha_0}}. \qquad (24)$$

In the case of a system with commensurate order γ we have

$$G(s) = \frac{\sum\limits_{k=0}^{m} b_k \left(s^{\gamma}\right)^k}{\sum\limits_{k=0}^{n} a_k \left(s^{\gamma}\right)^k}. \qquad (25)$$

Fractional-order transfer functions and state-space representation

Taking $\lambda = s^\gamma$ the function (25) can be viewed as a pseudo-rational function $H(\lambda)$:

$$H(\lambda) = \frac{\sum\limits_{k=0}^{m} b_k \lambda^k}{\sum\limits_{k=0}^{n} a_k \lambda^k}. \tag{26}$$

Based on the concept of the pseudo-rational function, a state-space representation can be established in the form:

$$\begin{aligned} \mathscr{D}^\gamma x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t). \end{aligned} \tag{27}$$

Example: From a FO transfer function to the FO state-space form

Suppose that we are given a fractional-order transfer function

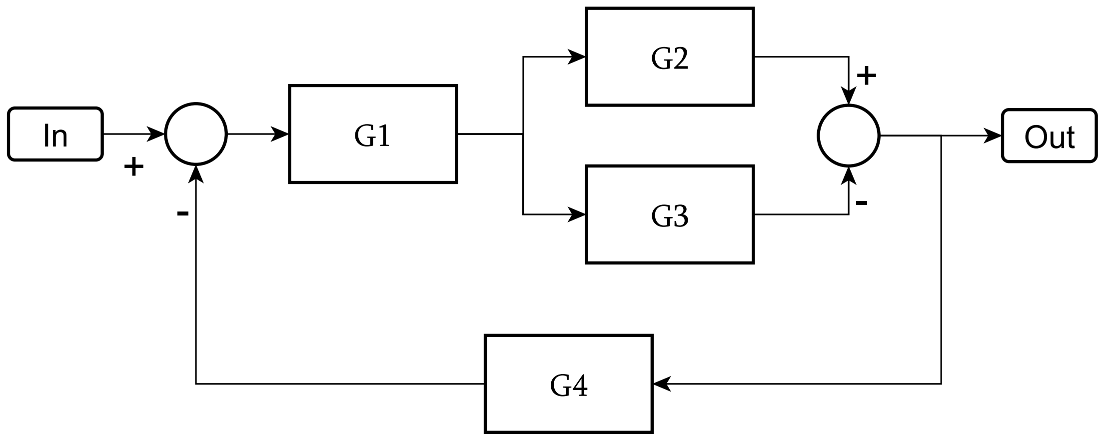
$$G(s) = \frac{s^{0.25} + 2.5}{3s^{1.75} + 2s^{0.5} + 1}.$$

We find, that the commensurate order for this system is $\gamma = 0.25$. Then we use $H(s) = C(sI - A)^{-1}B + D$ and arrive at the following state-space matrices

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & -0.66 & 0 & -0.33 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0.33 & 0.83 \end{bmatrix}, \quad D = 0.$$

Let us assume that a fractional system is given by a block diagram



Here

$$G_1(s) = \frac{1}{s^{0.5}+1}, \quad G_2(s) = \frac{s^{0.3}+1}{s^{2.5}+s+1},$$
$$G_3(s) = \frac{2}{s^{0.1}+1}, \quad G_4(s) = \frac{1}{15s+1}.$$

Compute the transfer function resulting from the interconnection above.

Example: Fractional system composition (solution)

The fractional-order systems we consider are linear. Therefore, the usual operations for computing system interconnections hold. In this case the complete system is given by

$$G(s) = \frac{\begin{gathered}-30s^{3.5} - 2s^{2.5} - 30s^{2} + 15s^{1.4} + 15s^{1.3} \\ +15s^{1.1} - 17s + s^{0.4} + s^{0.3} + s^{0.1} - 1\end{gathered}}{\begin{gathered}15s^{4.1} + 15s^{4} + 15s^{3.6} + 15s^{3.5} + s^{3.1} + s^{3} + 16s^{2.6} \\ +14s^{2.5} + 15s^{2.1} + 15s^{2} + 16s^{1.6} + 16s^{1.5} \\ +16s^{1.1} + 14s + s^{0.6} + s^{0.5} + s^{0.4} + s^{0.3} + 2s^{0.1}\end{gathered}}.$$

It can be seen from this example that from relatively simple initial systems a fairly complicated fractional-order transfer function was obtained. In this case we find, that the commensurate order of the system is $\gamma = 0.1$.

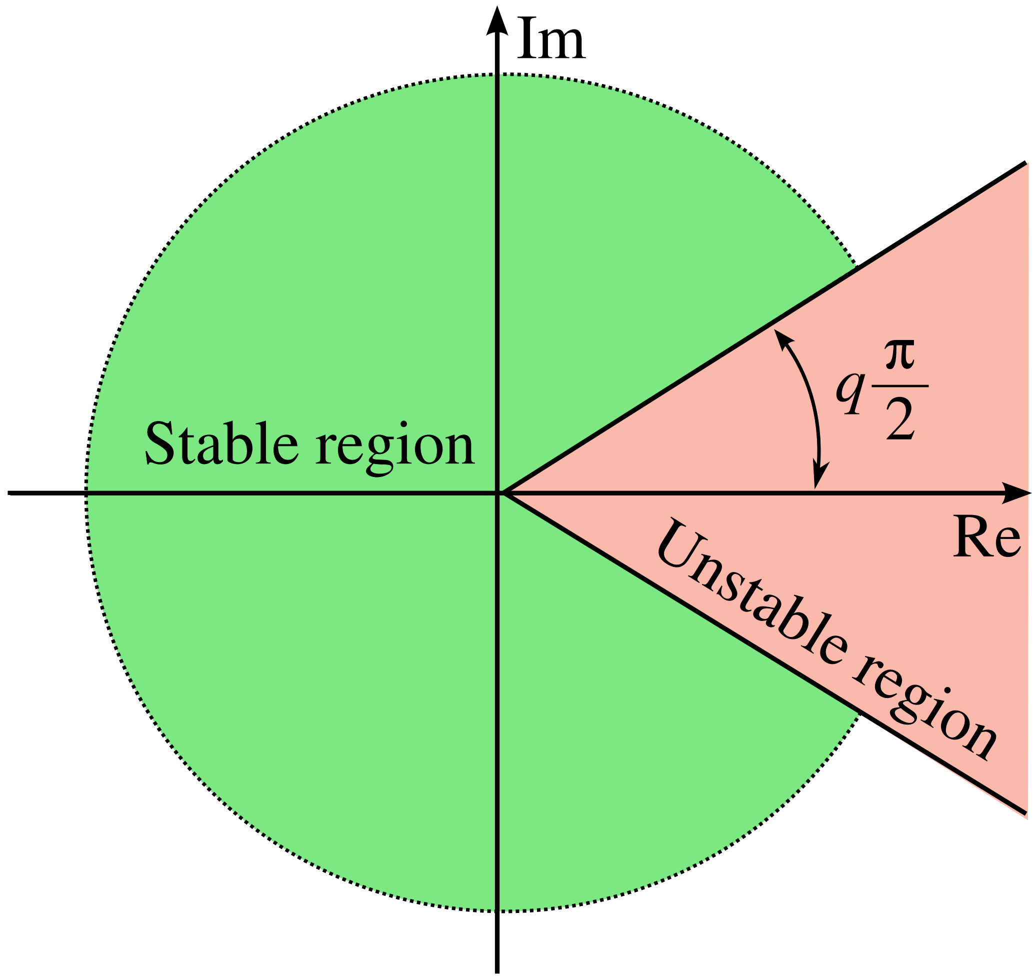
Theorem 1. *(Matignon's stability theorem) The fractional transfer function $G(s) = Z(s)/P(s)$ is stable if and only if the following condition is satisfied in σ-plane:*

$$\left|\arg(\sigma)\right| > q\frac{\pi}{2}, \forall\sigma \in \mathbb{C}, P(\sigma) = 0, \tag{28}$$

where $\sigma := s^q$. When $\sigma = 0$ is a single root of $P(s)$, the system cannot be stable. For $q = 1$, this is the classical theorem of pole location in the complex plane: no pole is in the closed right plane of the first Riemann sheet.

Algorithm summary: Find the commensurate order q of $P(s)$, find $a_1, a_2, \ldots a_n$ in (25) and solve for σ the equation $\sum_{k=0}^{n} a_k \sigma^k = 0$. If all obtained roots satisfy the condition (28), the system is stable.

Stability regions



Exercise: Stability

Determine the commensurate order γ of the fractional-order system given below. Then, write out and solve the characteristic equation $P(\lambda) = 0$. Hint: $\lambda = s^\gamma$.

$$G(s) = \frac{s+1}{s - 2s^{0.5} + 5}.$$

Solution: The commensurate order is $\gamma = 0.5$, so we have $\lambda = s^{0.5}$. Therefore, the characteristic equation is

$$P(\lambda) = \lambda^2 - 2\lambda + 5.$$

Solving $P(\lambda) = 0$ yeilds complex roots $\lambda_{1,2} = 1 \pm j2$. Notice, that in case of a classical integer-order system this result would immediately imply instability. However, in case of this system we have

$$|\arg(1 \pm j2)| \approx 1.1071 > 0.7854 \approx \frac{0.5\pi}{2},$$

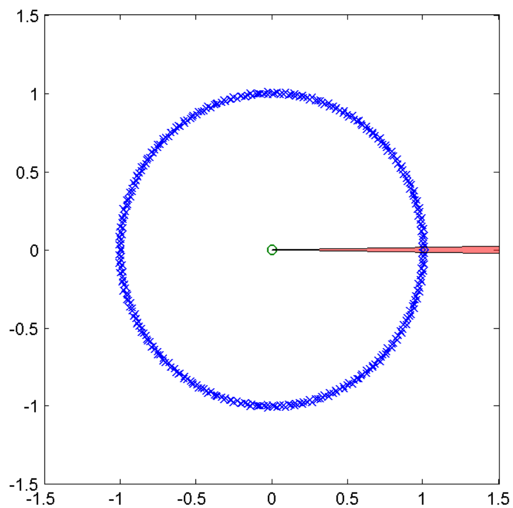
hence the system under analysis is stable.

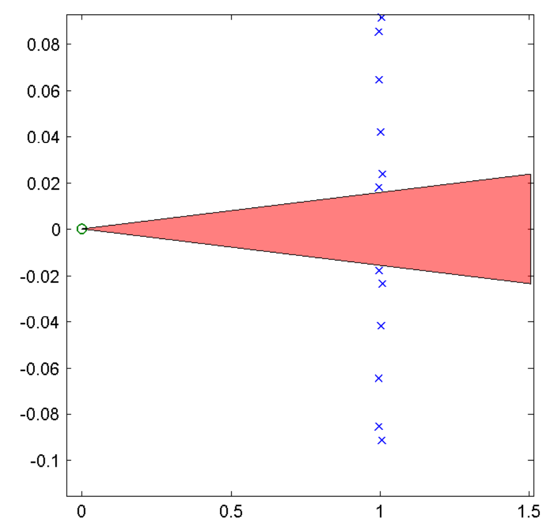
Example: Stability evaluation of a relatively complex system

The transfer function is

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

and the commensurate order $q = 0.01$. It is found to be stable.





Time-domain analysis

Consider a revised Grünwald-Letnikov definition rewritten as

$$ {}_a\mathscr{D}_t^{\alpha} f(t) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\left[\frac{t-a}{h}\right]} w_j^{(\alpha)} f(t - jh), \tag{29} $$

where h is the computation step-size and $w_j^{(\alpha)} = (-1)^j \binom{\alpha}{j}$ can be evaluated recursively from

$$ w_0^{(\alpha)} = 1,\ w_j^{(\alpha)} = \left(1 - \frac{\alpha + 1}{j}\right) w_{j-1}^{(\alpha)},\ j = 1,\ 2,\ \cdots . \tag{30} $$

Further manipulations provide an algorithm for fixed-step numerical time-domain evaluation of fractional-order transfer functions. Please see [3] for details.

Frequency-domain analysis

Frequency-domain response may be obtained by substituting $s = j\omega$ in (24). The complex response for a frequency $\omega \in (0; \infty)$ can then be computed as follows:

$$G(\omega) = \frac{b_m(j\omega)^{\beta_m} + b_{m-1}(j\omega)^{\beta_{m-1}} + \cdots + b_0(j\omega)^{\beta_0}}{a_n(j\omega)^{\alpha_n} + a_{n-1}(j\omega)^{\alpha_{n-1}} + \cdots + a_0(j\omega)^{\alpha_0}}, \qquad (31)$$

where j is the imaginary unit.

It should be noted, that frequency-domain analysis is a very important tool where fractional-order modeling and control design are concerned.

Approximation of fractional operators

The Oustaloup recursive filter gives a very good approximation of fractional operators in a specified frequency range and is widely used in fractional calculus. For a frequency range (ω_b, ω_h) and of order N the filter for an operator s^γ, $0 < \gamma < 1$, is given by

$$s^\gamma \approx K \prod_{k=-N}^{N} \frac{s + \omega'_k}{s + \omega_k}, \quad K = \omega_h^\gamma, \quad \omega_r = \frac{\omega_h}{\omega_b}, \tag{32}$$

$$\omega'_k = \omega_b (\omega_r)^{\frac{k+N+\frac{1}{2}(1-\gamma)}{2N+1}}, \quad \omega_k = \omega_b (\omega_r)^{\frac{k+N+\frac{1}{2}(1+\gamma)}{2N+1}}.$$

The resulting model order is $2N + 1$.

A modified Oustaloup filter has been proposed in literature [3].

Approximation of fractional-order models

A general method for approximating a fractional-order model by an integer-order one may be proposed. Recall the property in (14):

- The fractional-order derivative commutes with integer-order derivative

$$\frac{\mathrm{d}^n}{\mathrm{d}t^n}\left({}_a\mathscr{D}_t^{\alpha} f(t)\right) = {}_a\mathscr{D}_t^{\alpha}\left(\frac{\mathrm{d}^n f(t)}{\mathrm{d}t^n}\right) = {}_a\mathscr{D}_t^{\alpha+n} f(t).$$

Thus, for fractional orders $\alpha \geq 1$ it holds

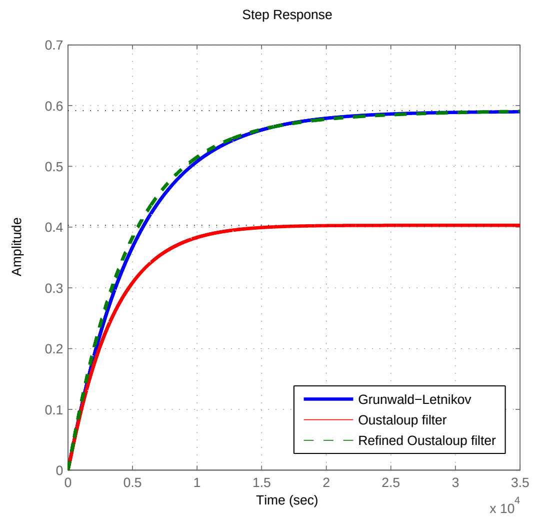
$$s^{\alpha} = s^n s^{\gamma}, \tag{33}$$

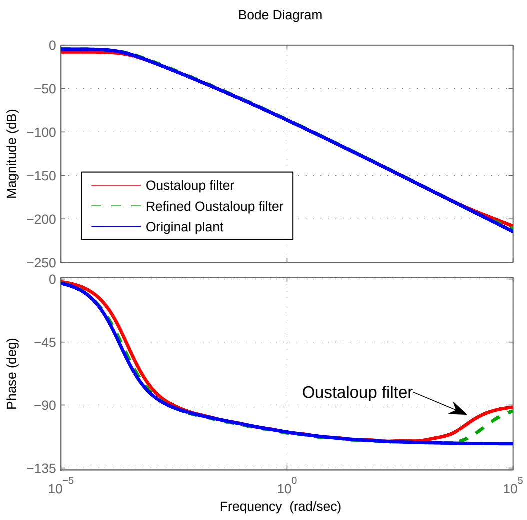
where $n = \alpha - \gamma$ denotes the integer part of α and s^{γ} is obtained by the Oustaloup approximation in (32).

The fractional-order transfer function is

$$G(s) = \frac{1}{14994s^{1.31} + 6009.5s^{0.97} + 1.69},$$

and approximation parameters $\omega = [10^{-4}; 10^{4}]$, $N = 5$.





Discrete-time approximation: The zero-pole matching equivalents method

Continuous zeros and poles, obtained using the Oustaloup recursive filter, are directly mapped to their discrete-time counterparts by means of the relation

$$z = \mathrm{e}^{sT_s}, \tag{34}$$

where T_s is the desired sampling interval. The gain of the resulting discrete-time system $H(z)$ must be corrected by a proper factor.

For the synthesis of continuous zeros and poles using the Oustaloup method with the intent to obtain a discrete-time approximation the transitional frequency ω_h may be chosen such that

$$\omega_h \leqslant \frac{2}{T_s}. \tag{35}$$

Fractional-order integrator: Implementation considerations

We now address the issue of implementing the fractional-order integrator component. A continuous-time integrator of order λ has to be implemented as

$$G_I(s) = \frac{1}{s^\lambda} = \frac{s^{1-\lambda}}{s}$$

to ensure a nice control effect at lower frequencies. Its discrete-time equivalent is given by

$$H_I(z^{-1}) = H^{1-\lambda}(z^{-1}) \cdot H_I(z^{-1}), \tag{36}$$

where $H^{1-\lambda}(z)$ is computed using the method presented above, and

$$H_I(z^{-1}) = \frac{T_s}{(1 - z^{-1})} \tag{37}$$

is a simple discrete-time integrator.

Time-domain identification: Output error minimization

Given the transfer function model in (24)

$$G(s) = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \cdots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \cdots + a_0 s^{\alpha_0}}$$

we search for a parameter set $\theta = [\ a_p \quad \alpha_p \quad b_z \quad \beta_z\ ]$, such that

$$\begin{aligned} a_p &= [\ a_n \quad a_{n-1} \quad \cdots \quad a_0\ ],\ \alpha_p = [\ \alpha_n \quad \alpha_{n-1} \quad \cdots \quad \alpha_0\ ], \\ b_z &= [\ b_m \quad b_{m-1} \quad \cdots \quad b_0\ ],\ \beta_z = [\ \beta_n \quad \beta_{n-1} \quad \cdots \quad \beta_0\ ], \end{aligned}$$

by employing numerical optimization with an objective function given by an output error norm $\left\| e\left(t\right) \right\|_2^2$, where $e(t) = y(t) - \tilde{y}(t)$ is obtained by taking the difference of the original model output $y(t)$ and simulated model output $\tilde{y}(t)$.

Time-domain identification: Process models

Consider the following generalizations of conventional process models used in industrial control design.

(FO)FOPDT	$G(s) = \frac{K}{1+Ts}\mathrm{e}^{-Ls}$	$G(s) = \frac{K}{1+Ts^{\alpha}}\mathrm{e}^{-Ls}$
(FO)IPDT	$G(s) = \frac{K}{s}\mathrm{e}^{-Ls}$	$G(s) = \frac{K}{s^{\alpha}}\mathrm{e}^{-Ls}$
(FO)FOIPDT	$G(s) = \frac{K}{s(1+Ts)}\mathrm{e}^{-Ls}$	$G(s) = \frac{K}{s(1+Ts^{\alpha})}\mathrm{e}^{-Ls}$

Therefore, due to additional parameters K (gain) and L (delay) we may update the identified parameter set discussed previously to

$$\theta = [\ K \quad L \quad a_p \quad \alpha_p \quad b_z \quad \beta_z\ ].$$

Residual Analysis

Denote by y_r the experimental plant output, and by y_m the identified model output. We consider the SISO case, so both y_r and y_m should be vectors of size $N \times 1$. In the following, we address the problem of statistical analysis of modeling residuals. Residuals are given by a vector containing the model output error

$$\varepsilon = y_r - y_m. \tag{38}$$

The percentage fit may be expressed as

$$Fit = \left(1 - \frac{\|\varepsilon\|}{\|y_r - \bar{y}_r\|}\right) \cdot 100\%, \tag{39}$$

where $\|\cdot\|$ is the Euclidean norm, and $\bar{y}_r$ is the mean value of y_r.

Residual Analysis: Basic Statistical Data

- Maximum absolute error

$$\varepsilon_{max} = \max_k |\varepsilon(k)|, \tag{40}$$

shows the maximum deviation from the expected behavior of the model over the examined time interval; however, it may be misleading in case of disturbances or strong noise;

- The mean squared error

$$\varepsilon_{MSE} = \frac{1}{N}\sum_{k=1}^{N}\varepsilon_k^2 = \frac{\|\varepsilon\|_2^2}{N} \tag{41}$$

may serve as a general measure of model quality. The lower it is, the more likely the model represents an adequate description of the studied process.

Residual Analysis: Autocorrelation of Residuals

Additional useful information is given by an estimate for autocorrelation of residuals for lag $\tau = 1, 2, \ldots, \tau_{max} < N$, which may be computed by means of

$$R_{\varepsilon}(\tau) = \frac{1}{(N-\tau)} \sum_{k=1}^{N-\tau} \varepsilon(k)\varepsilon(k+\tau). \tag{42}$$

The vector $r^{\varepsilon} = \begin{bmatrix} R_{\varepsilon}(1) & R_{\varepsilon}(2) & \cdots & R_{\varepsilon}(\tau_{max}) \end{bmatrix}$ is constructed and is normalized such that $r^{\varepsilon,norm} = r^{\varepsilon}/R_{\varepsilon}(1)$. Assuming normal distribution of residuals the confidence band $\hat{\eta}$ is then approximated for a confidence percentage $p_{conf} \in (0, 1]$ around zero mean as an interval

$$\hat{\eta} = \left[\left(0 - \Phi^{-1}(c_p)\right) / \sqrt{N}, \left(0 + \Phi^{-1}(c_p)\right) / \sqrt{N} \right], \tag{43}$$

where $c_p = 1 - 0.5(1 - p_{conf})$ and $\Phi^{-1}(x) = \sqrt{2}\,\mathrm{erf}^{-1}(2x - 1)$ is the quantile function. If the residual samples represent uncorrelated white noise, then ideally:

$$r_i^{\varepsilon,norm} \in \hat{\eta} \quad \forall i = 1, 2, \ldots, \tau_{max}. \tag{44}$$

Time domain identification: Different optimization algorithms: Example

Identification data is collected from a system

$$\Psi = \Psi_G + \mathfrak{N}, \tag{45}$$

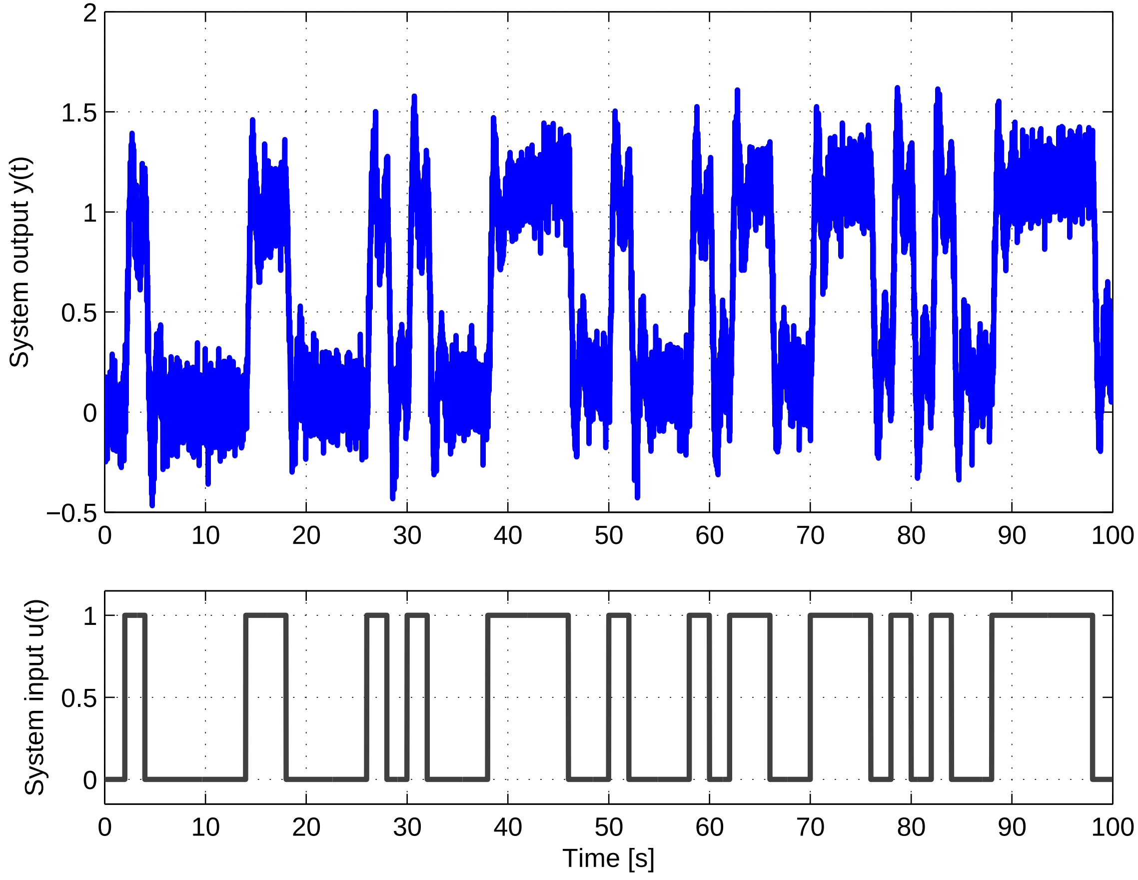
where Ψ_G is given by a continuous-time fractional-order transfer function of the form

$$\Psi_G(s) = \frac{1.5}{0.11s^{1.93} + 0.79s^{0.31} + 1}, \tag{46}$$

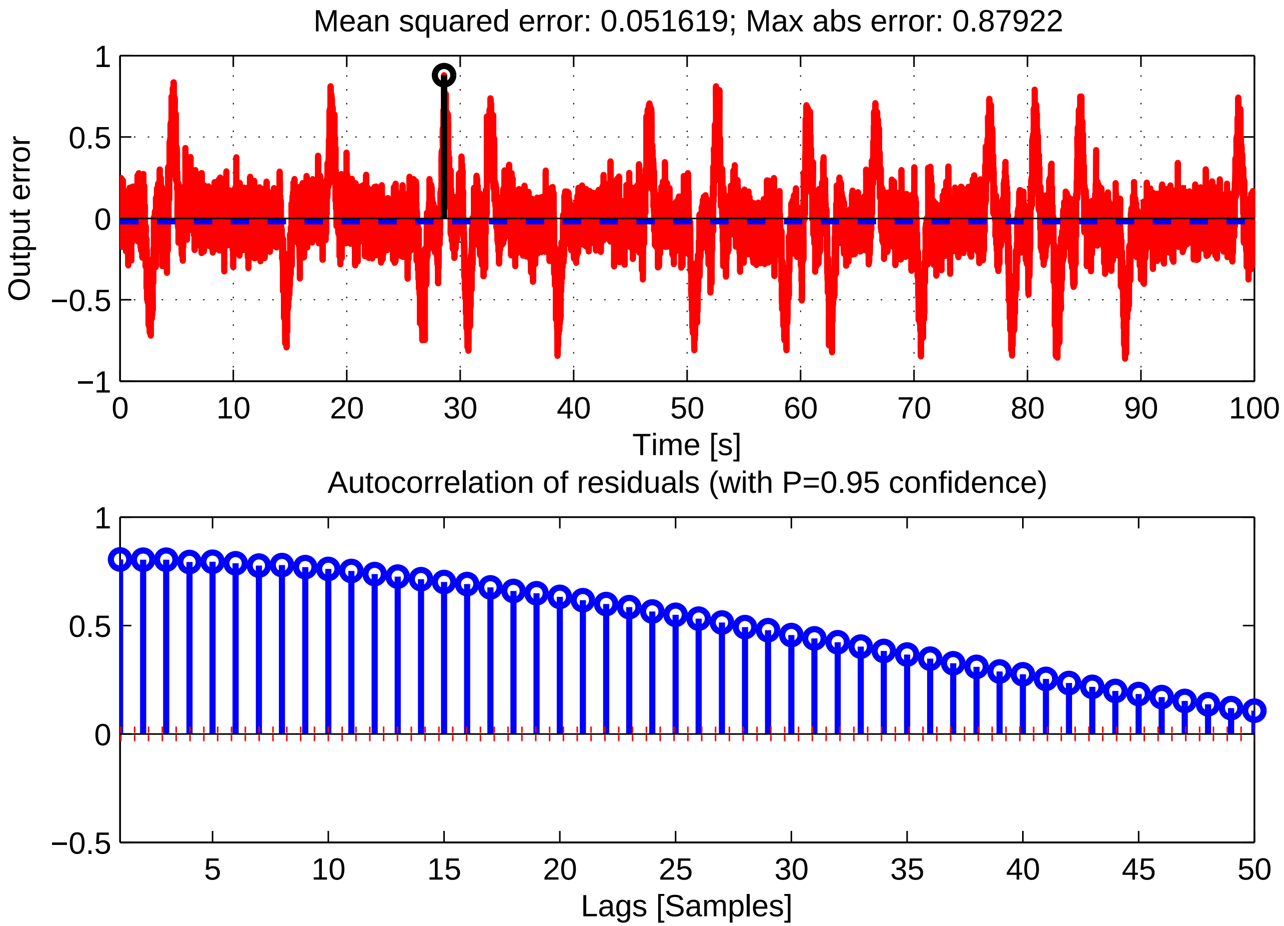
and the noise term has an amplitude of $\mathfrak{N} = \pm 0.05$. A pseudo-random binary sequence is used as the excitation signal for obtaining the transient response with a sample time of 0.01s.

In this example, the initial model structure is chosen such that its pseudo-order is $n = 2$ and commensurate order $\gamma = 1$.

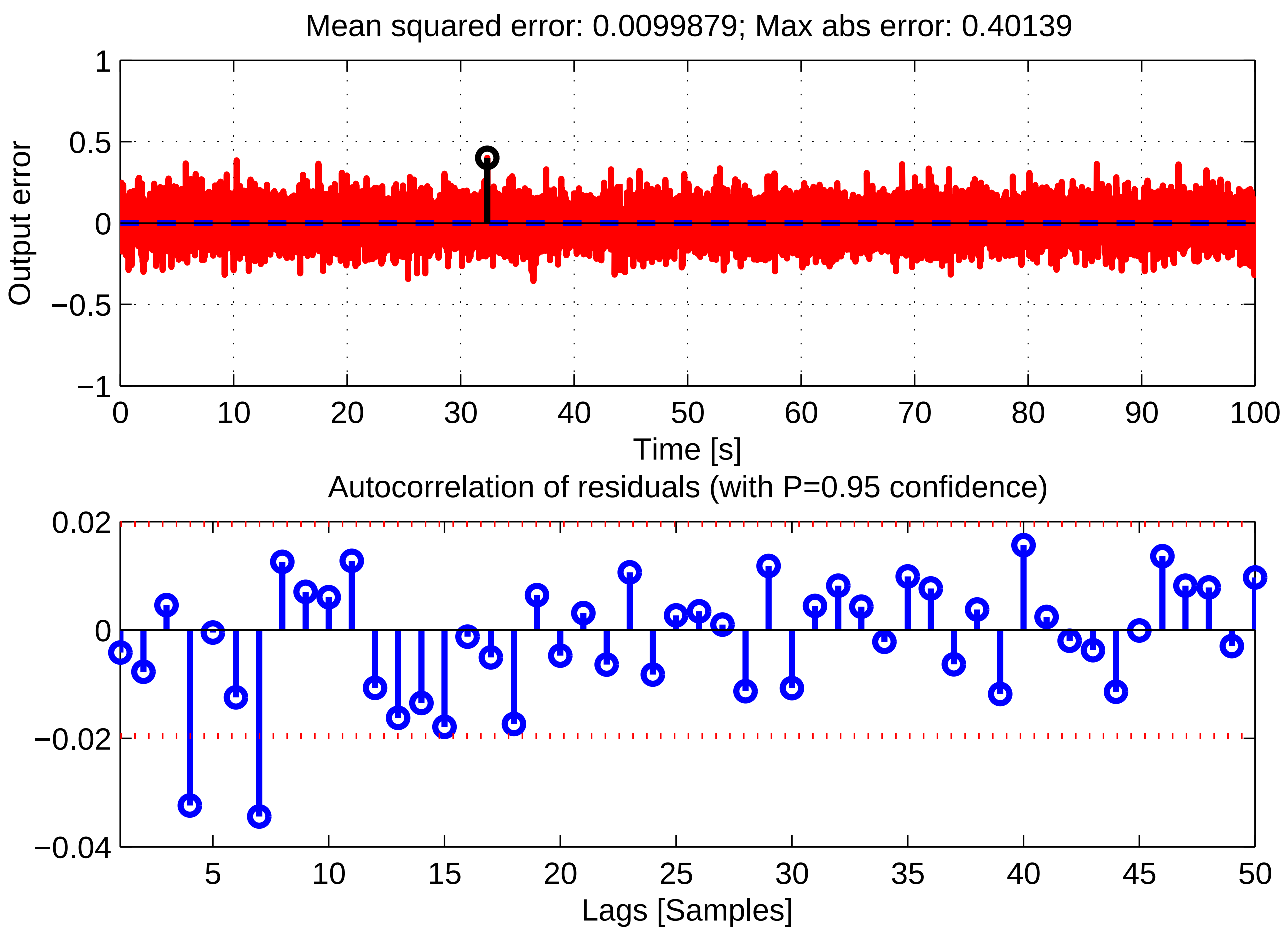
Time domain identification: Excitation signal



Time domain identification: Trust-Region-Reflective identification



Time domain identification: Levenberg-Marquardt algorithm



Part III: Factional-order PID Controllers

Fractional-order Control: PI$^\lambda$D$^\mu$ controller

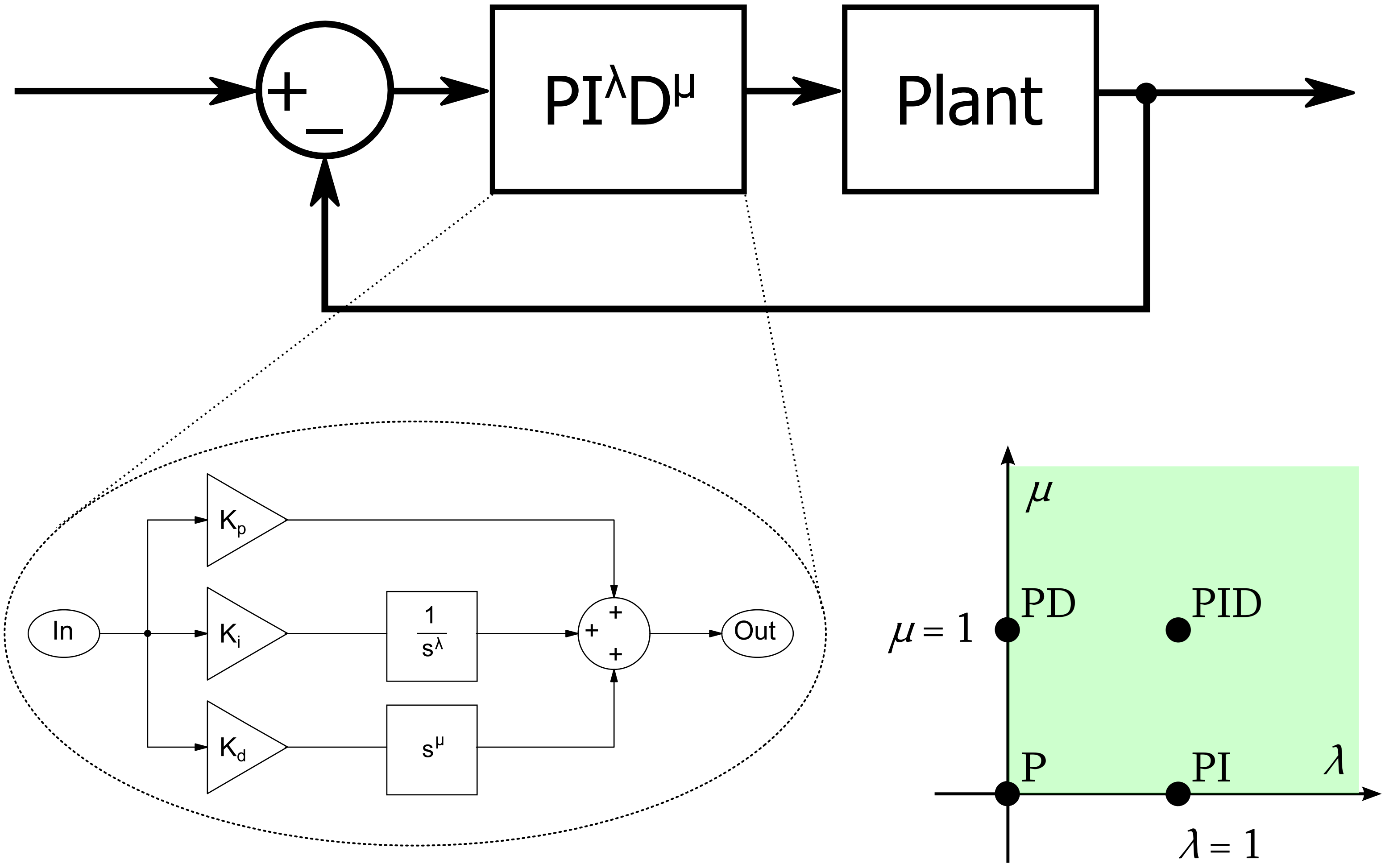
The control law of the PI$^\lambda$D$^\mu$ controller can be expressed as follows:

$$u(t) = K_p e(t) + K_i \mathscr{D}^{-\lambda} e(t) + K_d \mathscr{D}^{\mu} e(t), \tag{47}$$

where $e(t) = y_{sp}(t) - y(t)$ is the error signal. After applying the Laplace transform to (47) assuming zero initial conditions, the following equation is obtained:

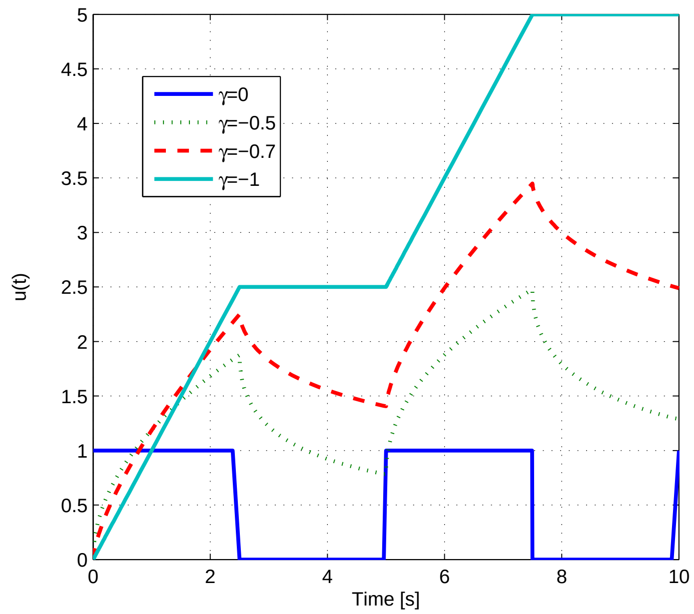
$$C(s) = K_p + \frac{K_i}{s^\lambda} + K_d s^\mu \tag{48}$$

Obviously, when taking $\lambda = \mu = 1$ the result is the classical integer-order PID controller.

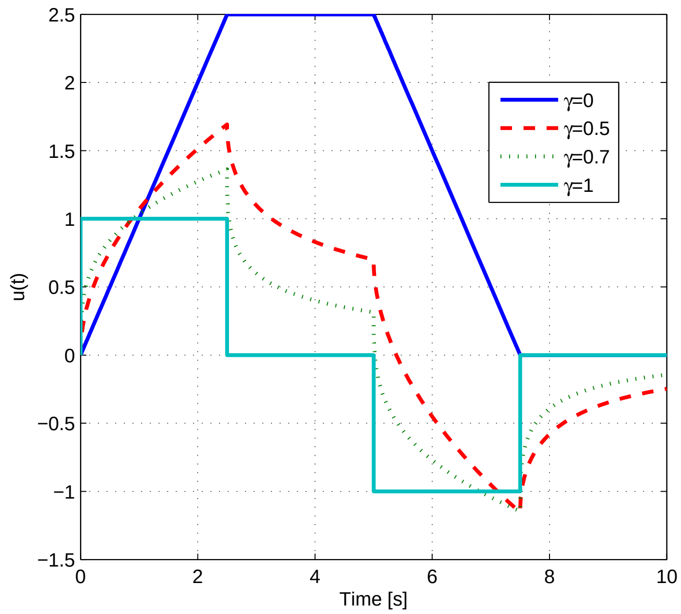

PIλDμ
Plant
+
_
In
Kp
Ki
Kd
1
sλ
sμ
+
+
+
Out
μ
PD
PID
μ = 1
P
PI
λ
λ = 1

Basics of fractional control: Fractional control actions

Let a basic fractional control action be defined as $C(s) = K \cdot s^{\gamma}$. The control actions in the time domain for $\gamma \in [-1, 1]$ with $K = 1$ under different input signals are given below.

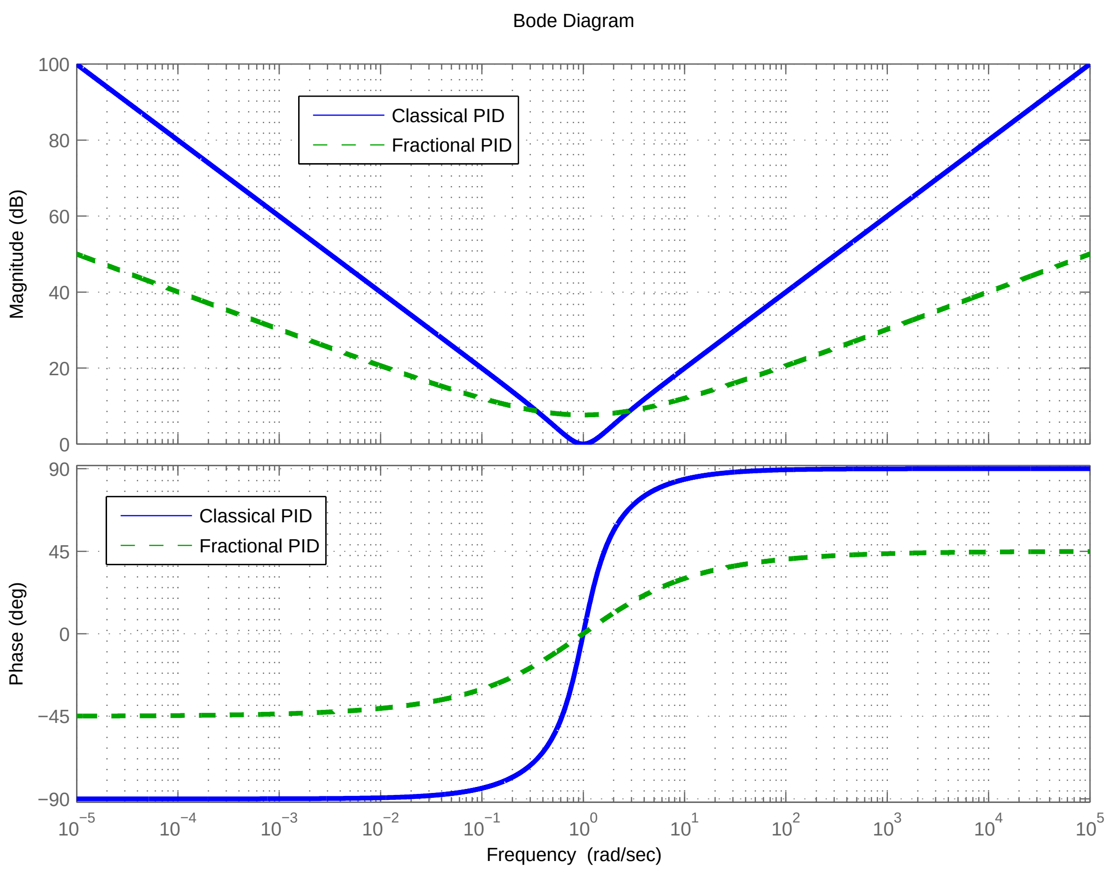


Fractional integrator $s^{-\gamma}$



Fractional differentiator s^{γ}

PID controller vs. $PI^{0.5}D^{0.5}$ controller: frequency-domain characteristics



Fractional-order Control: PI$^\lambda$D$^\mu$ controller tuning methods (F-MIGO)

We would like to establish tuning methods for the FOPID controller similar to conventional ones (e.g. Ziegler-Nichols tuning formulae). Several methods have been proposed in literature so far. Consider the F-MIGO method suitable for tuning PI$^\lambda$ controllers [3]. Suppose we are given a FOPDT process model

$$G(s) = \frac{K}{Ts+1}\mathrm{e}^{-Ls}, \quad \tau = \frac{L}{L+T}, \tag{49}$$

where τ is the relative dead-time of the system. Then

$$\lambda = \begin{cases} 1.1, & \text{if } \tau \geqslant 0.6 \\ 1.0, & \text{if } 0.4 \leqslant \tau < 0.6 \\ 0.9, & \text{if } 0.1 \leqslant \tau < 0.4 \\ 0.7, & \text{if } \tau < 0.1. \end{cases}$$

and

$$K_p = \frac{1}{K}\left(\frac{0.2978}{\tau + 0.000307}\right), \quad K_i = \frac{K_p(\tau^2 - 3.402\tau + 2.405)}{0.8578T}.$$

Optimization based PI$^\lambda$D$^\mu$ tuning

Optimization provides general means of tuning a fractional-order PID controller given a cost function and suitable optimization constraints. There are several aspects to the problem of designing a proper controller using constrained optimization:

- The type of plant to be controlled (integer or noninteger order, nonlinear);
- Optimization criterion (cost function);
- Fractional controller design specifications;
- Specific parameters to optimize in the set $\{K_p, K_i, K_d, \lambda, \mu\}$;
- Selection of initial controller parameters.

Optimization based PI$^\lambda$D$^\mu$ tuning: Cost function

In case of a linear model we use time-domain simulation of a typical negative unity feedback loop

$$G_{cs}(s) = \frac{C(s)G(s)}{1 + C(s)G(s)}. \tag{50}$$

For the cost function we consider performance indicies:

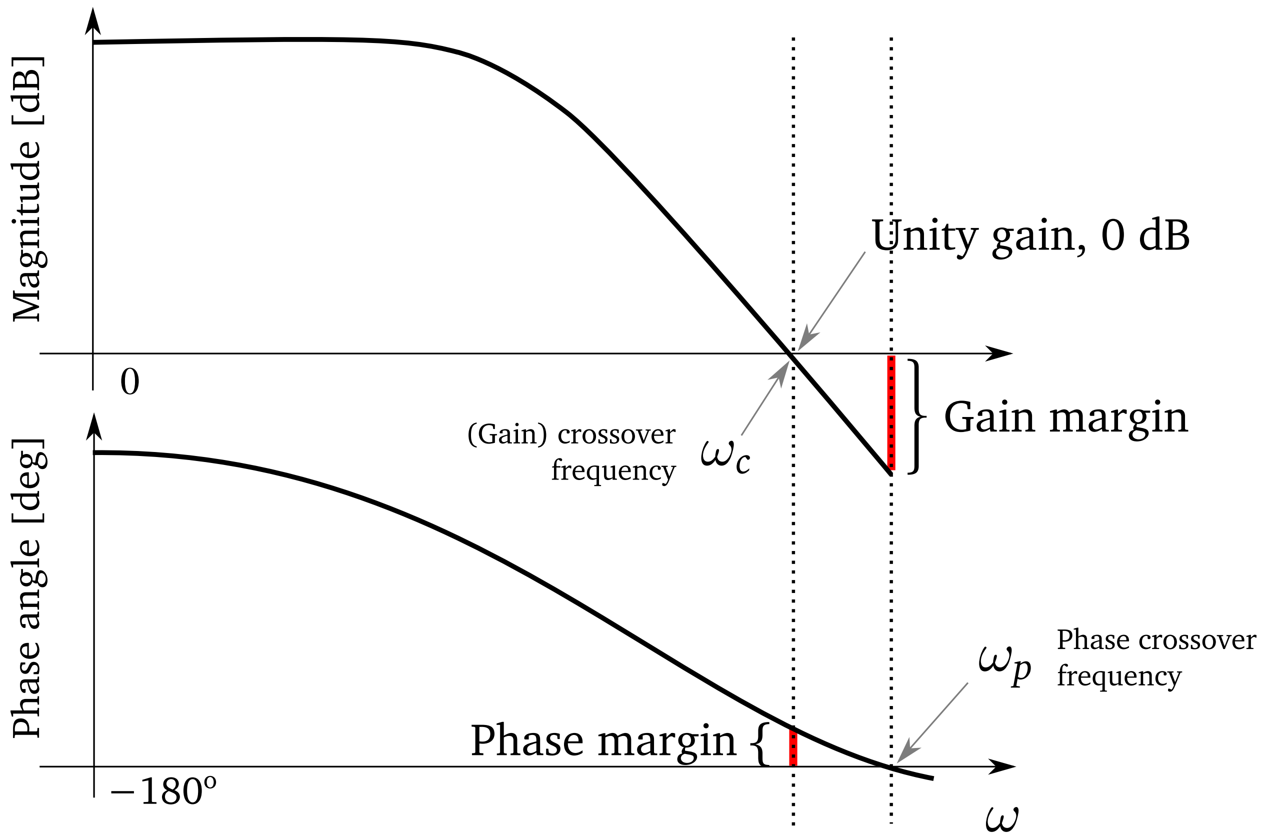
- integral square error $ISE = \int_0^\tau e^2(t)\mathrm{d}t$,
- integral absolute error $IAE = \int_0^\tau \left|e(t)\right| \mathrm{d}t$,
- integral time-square error $ITSE = \int_0^\tau te^2(t)\mathrm{d}t$,
- integral time-absolute error $ITAE = \int_0^\tau t \left|e(t)\right| \mathrm{d}t$.

Optimization based PI$^\lambda$D$^\mu$ tuning: Constraints

The design specifications include:

- Gain margin G_m and phase margin φ_m specifications;
- Complementary sensitivity function $T(j\omega)$ constraint, providing A dB of noise attenuation for frequencies $\omega > \omega_t$ rad/s;
- Sensitivity function $S(j\omega)$ constraint for output disturbance rejection, providing a sensitivity function of B dB for frequencies $\omega < \omega_s$ rad/s;
- Robustness to plant gain variations: a flat phase of the system is desired within a region of the system critical frequency ω_{cg};
- For practical reasons, a constraint on the control effort $u(t)$ may also be set.

Gain and phase margin specifications



(See `http://a-lab.ee/edu/ajs/freq/` for details.)

The FOPID Controller Retuning Method

Consider the original integer-order PID controller of the form

$$C_{PID}(s) = K_P + K_I s^{-1} + K_D s. \tag{51}$$

Let $C_R(s)$ be a controller of the form

$$C_R(s) = \frac{K_2 s^{\beta} + K_1 s^{\alpha} - K_D s^2 + (K_0 - K_P)s - K_I}{K_D s^2 + K_P s + K_I}, \tag{52}$$

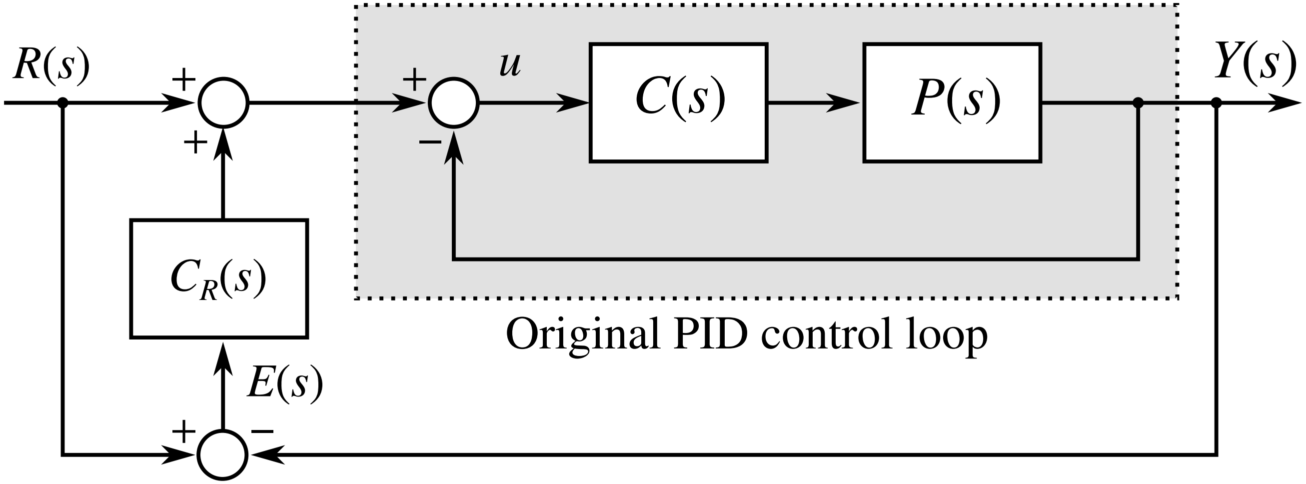
where the orders α and β are such, that $-1 < \alpha < 1$ and $1 < \beta < 2$. The PI$^{\lambda}$D$^{\mu}$ controller resulting from a classical PID controller will have the following coefficients

$$K_P^{\star} = K_0, \quad K_I^{\star} = K_1, \quad K_D^{\star} = K_2, \tag{53}$$

and the orders will be

$$\lambda = 1 - \alpha, \quad \mu = \beta - 1. \tag{54}$$

The FOPID Controller Retuning Method: Illustration



It can be shown, that this structure may be replaced by a negative unity feedback where the controller is

$$C(s) = (C_R(s) + 1) \cdot C_{PID}(s). \tag{55}$$

Discrete-time approximation of fractional-order controllers

After acquiring a set of discrete-time zeros and poles by means of (34), the fractional-order controller may be implemented in form of a IIR filter represented by a discrete-time transfer function $H(z^{-1})$. In general, one has two choices:

1. Implement each fractional-order component approximation of the controller in (48) separately as $H^{\lambda}(z^{-1})$ and $H^{\mu}(z^{-1})$; this method offers greater flexibility, since the components may be reused in the digital signal processing chain, but requires more memory and is generally more computationally expensive;
2. Compute a single LTI object approximating the whole controller; this method is suitable when there is a need for a static description of a fractional-order controller.

Digital controller implementation: IIR filters

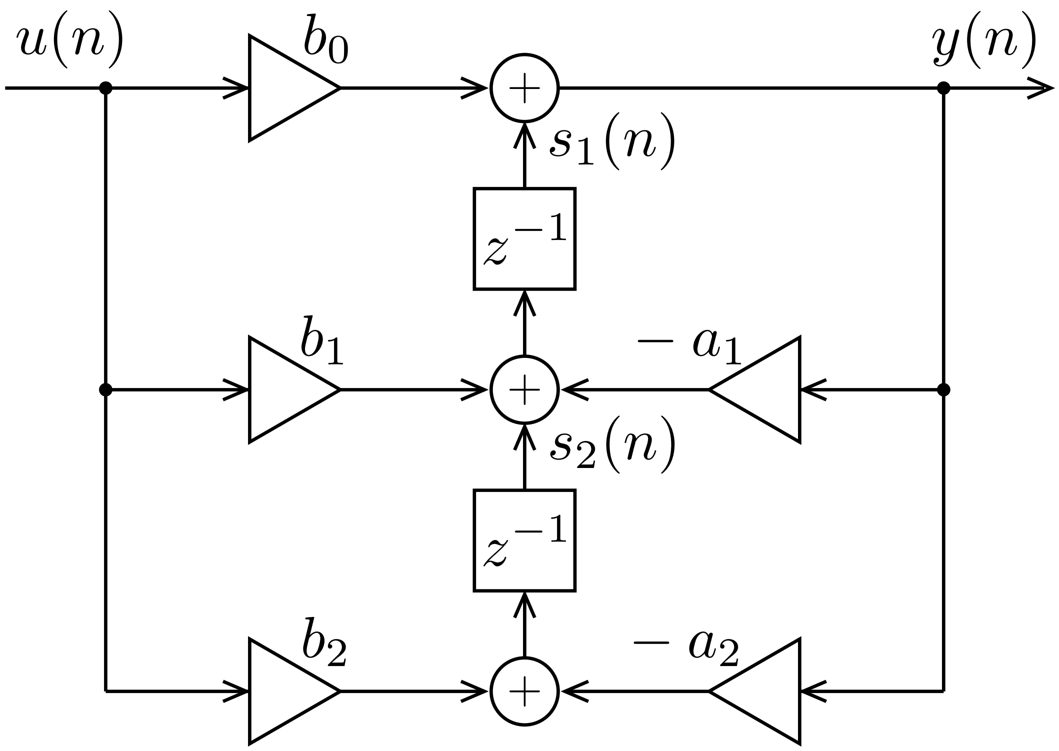
In this particular work we choose the second option, that is we seek a description of the controller in the form

$$H(z^{-1}) = K\frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}}. \tag{56}$$

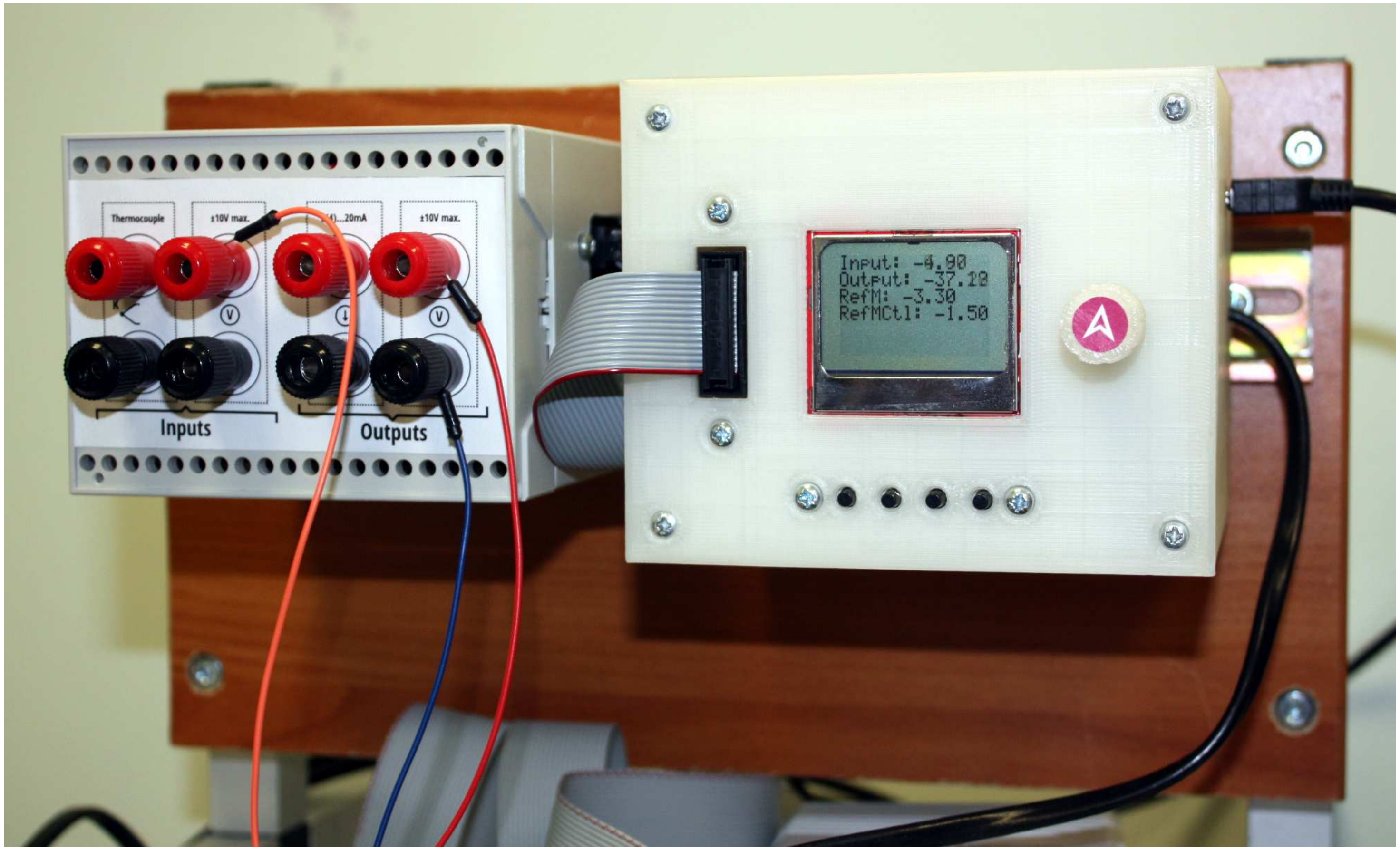
For practical reasons, the equivalent IIR filter should be comprised of second-order sections. This allows to improve computational stability when the target signal digital processing hardware has limited DSP capabilities. Thus, the discrete-time controller must be transformed to yield

$$H(z^{-1}) = K_c \prod_{k=1}^{N} \frac{b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}}. \tag{57}$$

Biquad IIR filter: Transposed form II



FOPID Controller Hardware Prototype

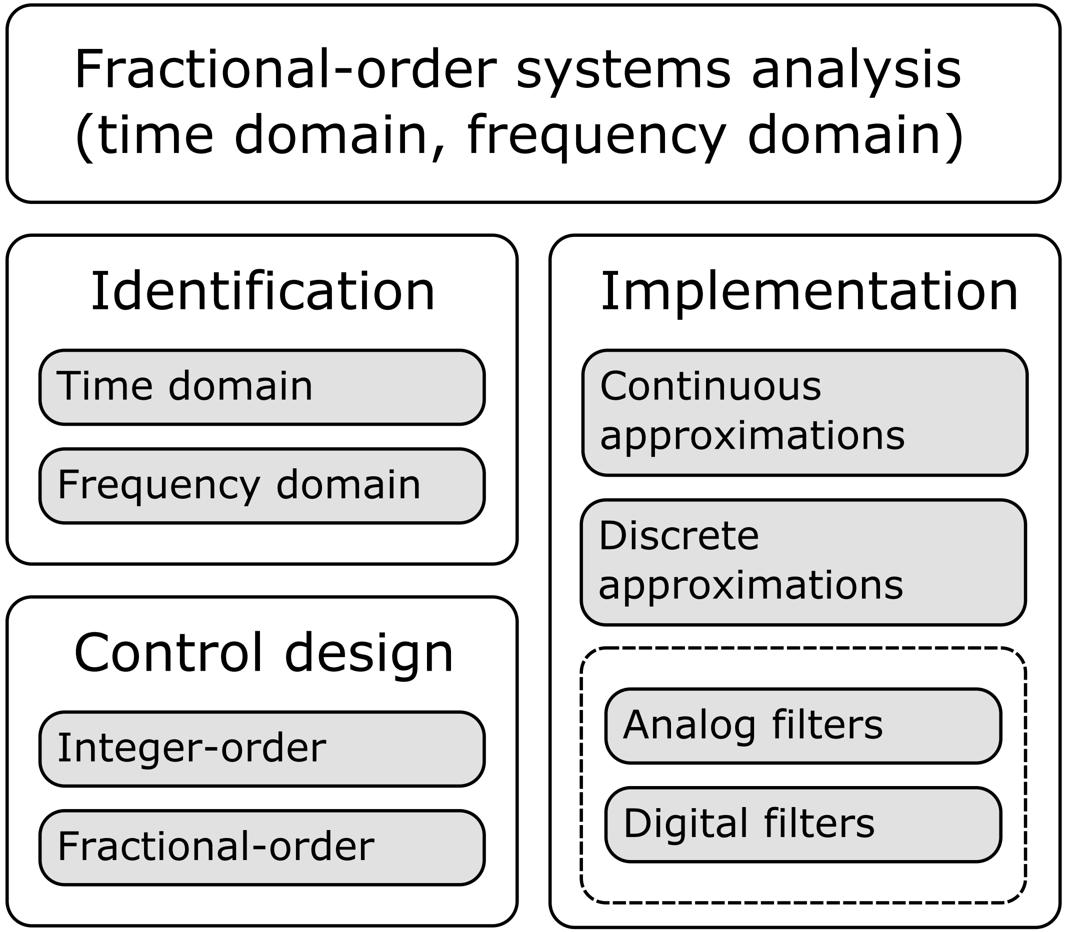


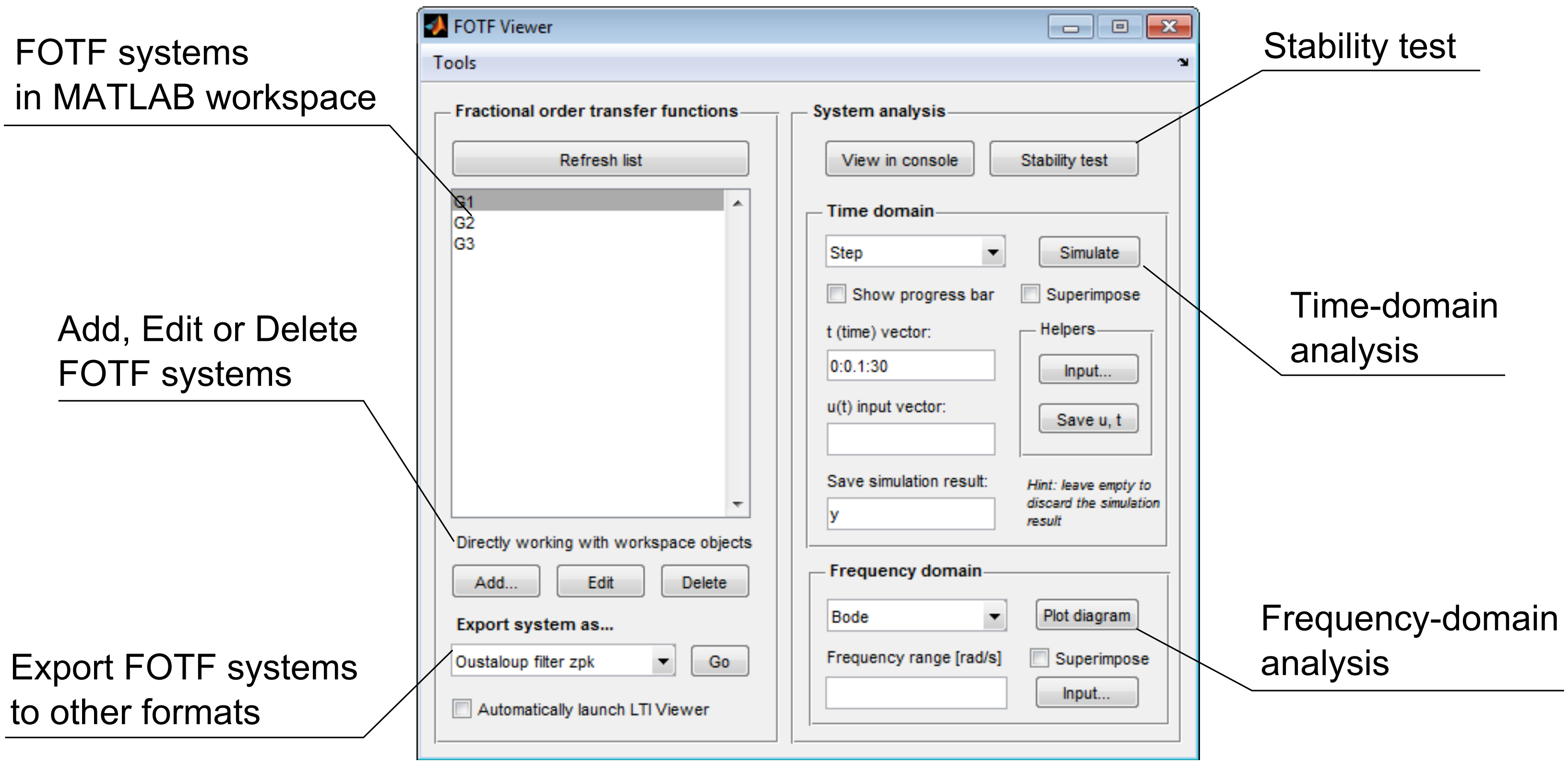
Part IV: CACSD Tools: FOMCON

FOMCON project: Fractional-order Modeling and Control

- Official website: `http://fomcon.net/`
- Toolbox for MATLAB available, development via GitHub: `https://github.com/AlekseiTepljakov/fomcon-matlab`
- Recently: Added initial support for studying FO MIMO systems.

FOMCON toolbox: Structure




FOTF systems in MATLAB workspace
Add, Edit or Delete FOTF systems
Export FOTF systems to other formats
Stability test
Time-domain analysis
Frequency-domain analysis
FOTF Viewer
Tools
Fractional order transfer functions
Refresh list
G1
G2
G3
Directly working with workspace objects
Add...
Edit
Delete
Export system as...
Oustaloup filter zpk
Go
Automatically launch LTI Viewer
System analysis
View in console
Stability test
Time domain
Step
Simulate
Show progress bar
Superimpose
t (time) vector:
0:0.1:30
Helpers
Input...
u(t) input vector:
Save u, t
Save simulation result:
y
Hint: leave empty to discard the simulation result
Frequency domain
Bode
Plot diagram
Frequency range [rad/s]
Superimpose
Input...

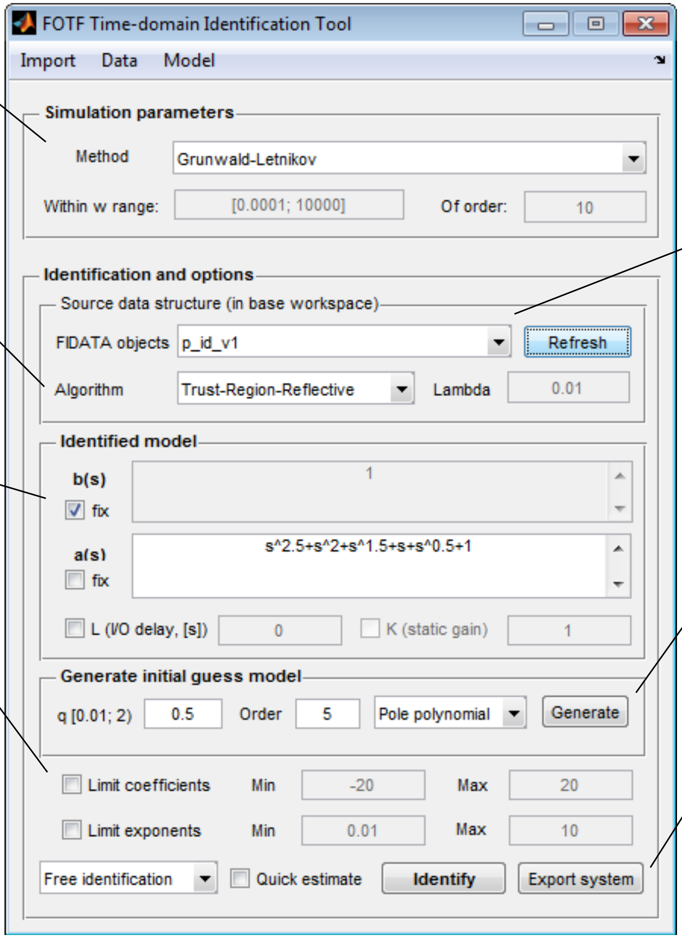
FOMCON toolbox: Time-domain identification

Time-domain simulation parameters

Optimization algorithm selection

Identified model; Structure selection

Other available identification options

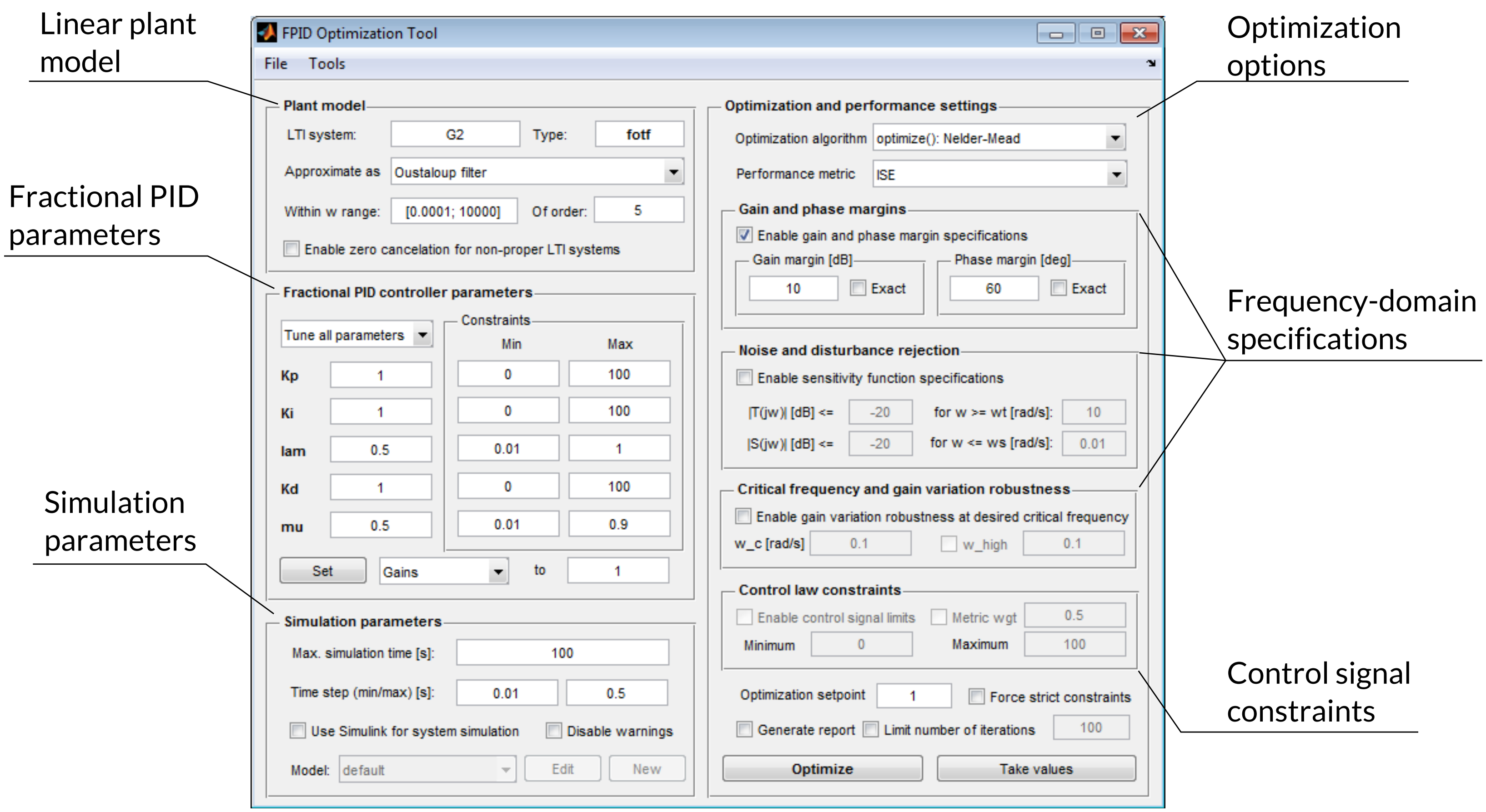


Experimental data

Initial guess model generation

Identified model export to workspace

FOMCON toolbox: Optimization based PI$^\lambda$D$^\mu$ tuning



Part V: Applications of Fractional-order Control

Case study (1): Fractional-order control of the coupled tank system

The system is modeled in continuous time in the following way:

$$\dot{x}_1 = \frac{1}{A}u_1 - d_{12} - w_1 c_1 \sqrt{x_1}, \qquad (58)$$

$$\dot{x}_2 = \frac{1}{A}u_2 + d_{12} - w_2 c_2 \sqrt{x_2},$$

where x_1 and x_2 are levels of fluid, A is the cross section of both tanks; c_1, c_2, and c_{12} are flow coefficients, u_1 and u_2 are pump powers; valves are denoted by $w_i : w_i \in \{0, 1\}$ and

$$d_{12} = w_{12} \cdot c_{12} \cdot \mathrm{sign}(x_1 - x_2)\sqrt{|x_1 - x_2|}.$$

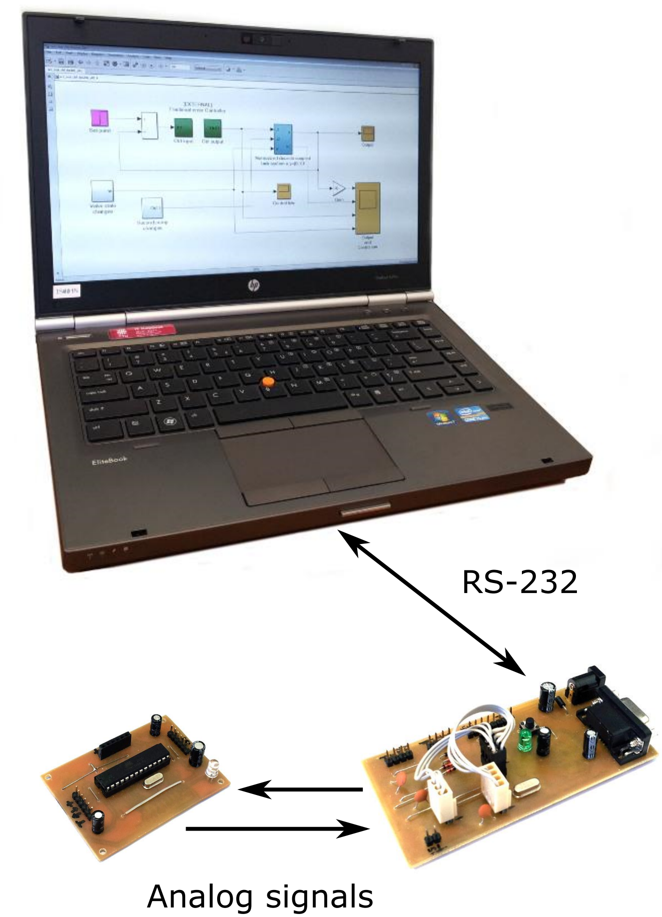
Case study (1): Fractional-order control of the coupled tank system (continued)

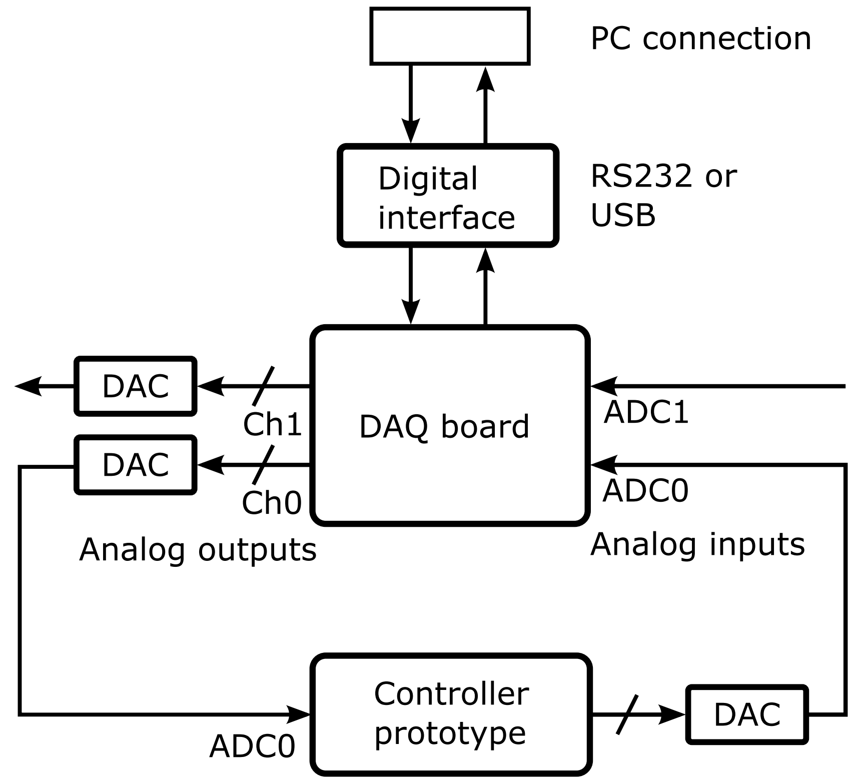
Our task is to control the level in the first tank. We identify the real plant from a step experiment with $w_1 = w_{12} = 1, w_2 = 0$ in (58). The resulting fractional-order model is described by a transfer function

$$G_2 = \frac{2.442}{18.0674s^{0.9455} + 1}\mathrm{e}^{-0.1s}. \quad (59)$$

We notice, that this model does not tend to exhibit integer-order dynamics. Due to the value of the delay term the basic tuning formulae for integer-order PID tuning do not provide feasible results. It is possible to select some starting point manually and run optimization several times. However, it is important to choose the correct frequency domain specifications to ensure control system stability.

Case study (1): Experiments with controller implementation: Hardware platform



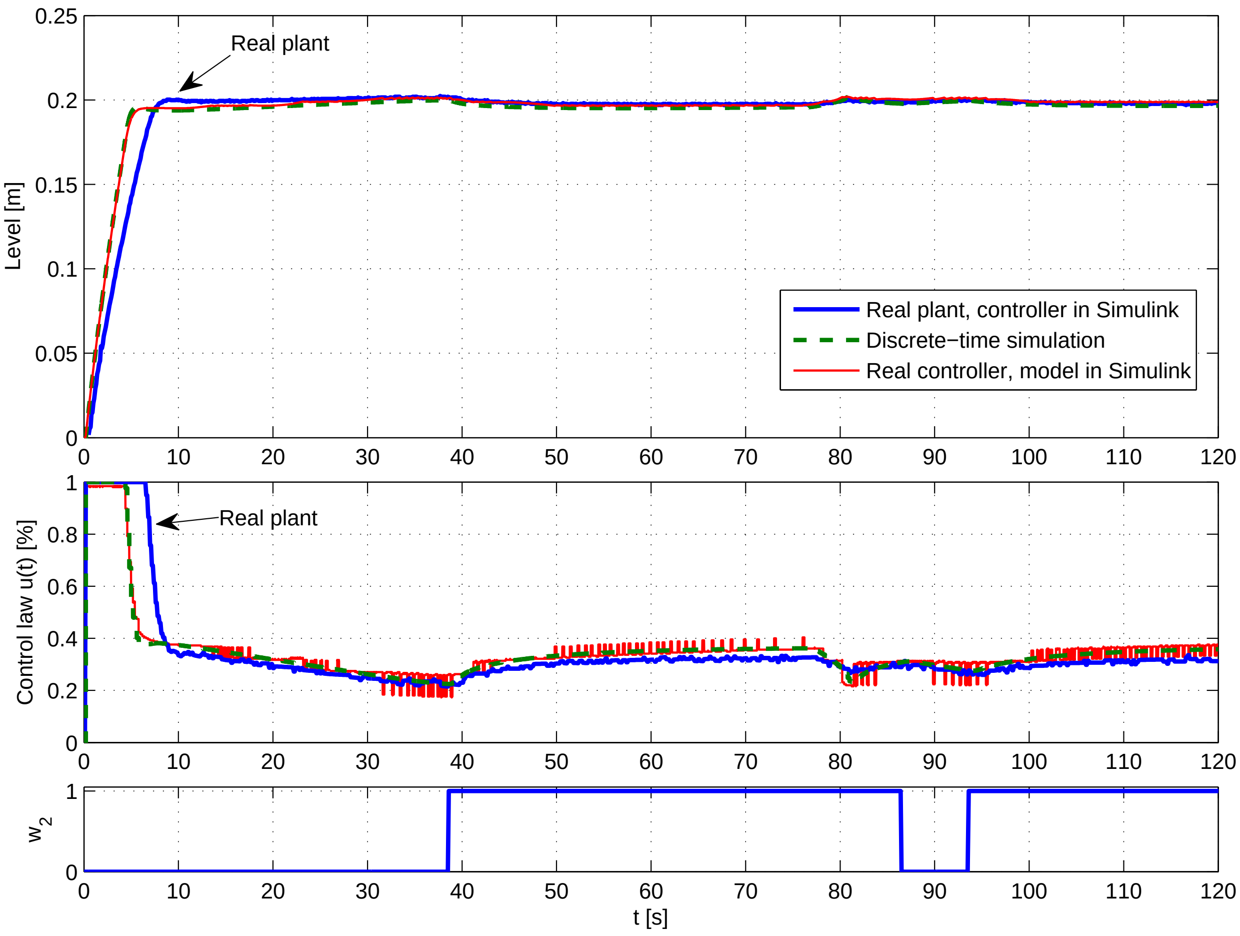


Case study (1): Fractional-order control of the coupled tank system (continued)

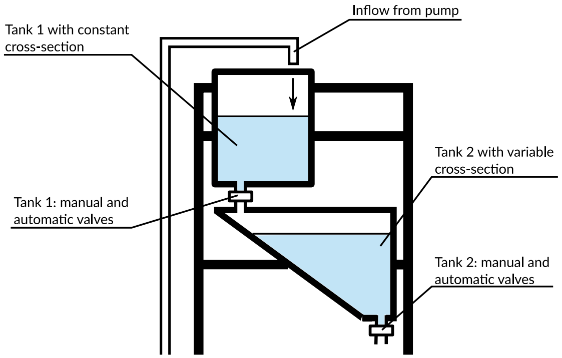
In our case the goal is to minimize the impact of disturbance, so constraints on the sensitivity functions could be imposed. Our choice is such that $\left|T(j\omega)\right| \leq -20\, dB, \forall \omega \geq 10\,\mathrm{rad/s}$ and $\left|S(j\omega)\right| \leq -20\, dB, \forall \omega \leq 0.1\,\mathrm{rad/s}$. The gain and phase margins are set to $G_m = 10\,\mathrm{dB}$ and $\varphi = 60^\circ$, respectively. Additionally, in order to limit the overshoot, the upper bound of control signal saturation was lowered from 100% to 60%. Using the IAE performance metric we finally arrive at the following PI$^\lambda$D$^\mu$ controller parameters by optimizing the response of the nonlinear system in Simulink:

$$K_p = 6.9514, \quad K_i = 0.13522, \quad K_d = -0.99874, \quad \lambda = 0.93187, \quad \mu = 0.29915. \tag{60}$$

Case study (1): Fractional-order control of the coupled tank system (continued)



Case study (2): Control of a Multi-Tank System



This system can be described by the following differential equations:

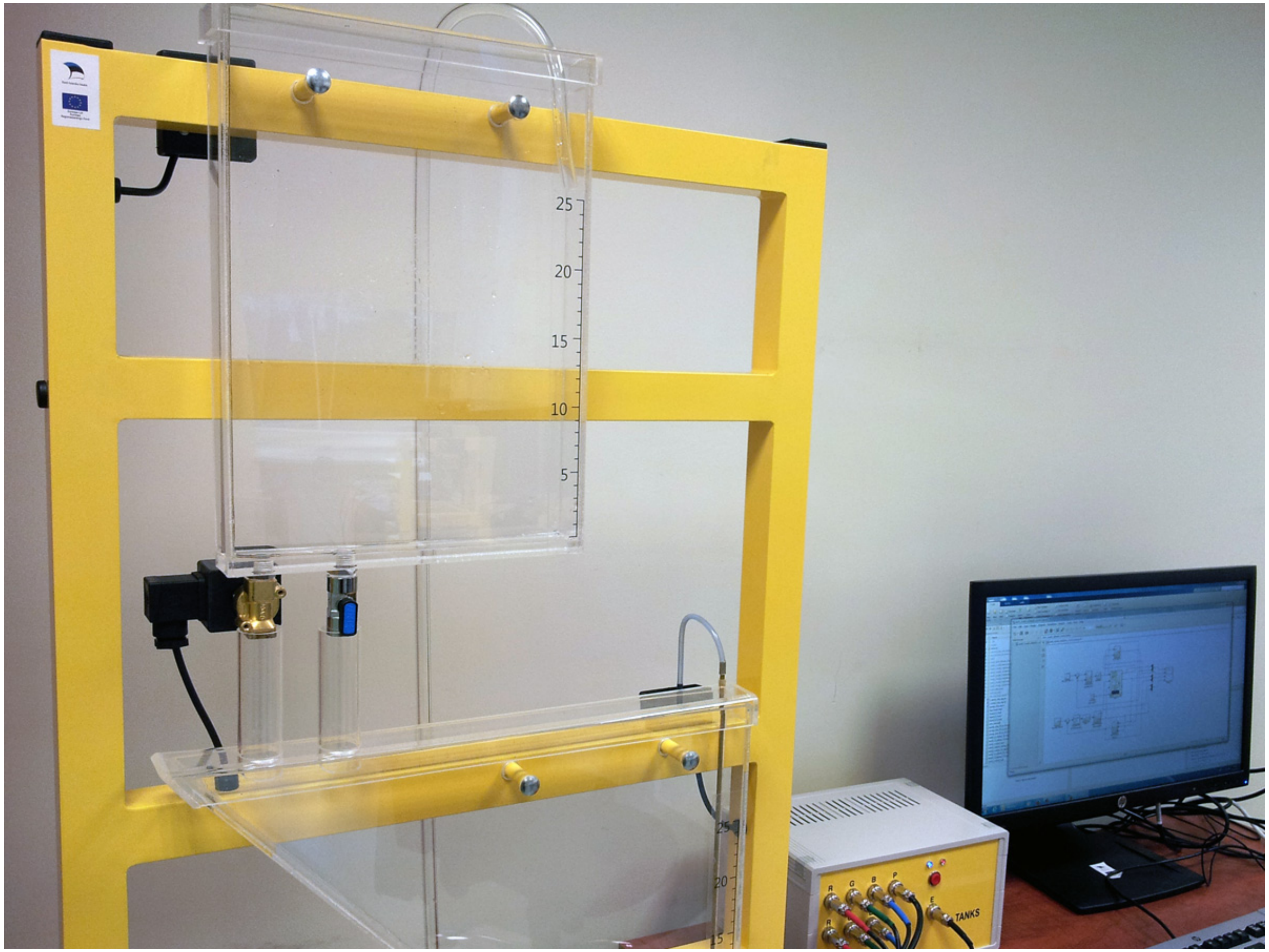
$$\dot{x}_1 = \frac{1}{\eta_1(x_1)}\left(u_p(v) - C_1 x_1^{\alpha_1} - \zeta_1(v_1) x_1^{\alpha_{v1}}\right),$$
$$\dot{x}_2 = \frac{1}{\eta_2(x_2)}\left(q + r - C_2 x_2^{\alpha_2} - \zeta_2(v_2) x_2^{\alpha_{v2}}\right),$$

where x_1 and x_2 are levels in the upper tank and middle tank, respectively, $\eta_1(x_1) = A = aw$ and $\eta_2(x_2) = cw + x_2 bw / x_{2max}$ are cross-sectional areas of the upper and middle tank, respectively, $u_p(v)$ is the pump capacity, such that depends on the normalized input $v(t) \in [0, 1]$; $\zeta_1(v_1)$ and $\zeta_2(v_2)$ are variable flow coefficients of the automatic valves controlled by normalized inputs $v_1(t), v_2(t) \in [0, 1]$, $q = C_1 x_1^{\alpha_1}$ and $r = \zeta_1(v_1) x_1^{\alpha_{v1}}$.

Case study (2): Statement of the control problem

- The task is to design a controller for the upper tank such that would keep the level of fluid within reasonable bounds at the desired set point in the presense of disturbances caused by the controlled output valve.
- It is required to design a controller for the middle tank, such that would keep the level of fluid at the desired set point using controlled valves of the upper tank and also its own valve.
- The tanks are, in fact, coupled, so only a limited range of fluid level values is achievable in the middle tank and it is related to the level in the upper tank.
- The outflow of liquid from the upper tank through the automatic valve forms part of the control for the middle tank and is considered a disturbance from the perspective of level control in the upper tank.

Case study (2): The real-life Multi-Tank system



Case study (2): Linear approximations

First, linear approximations are obtained from the nonlinear model by means of time-domain identification at system working points $(0.7029, 0.1)$ and $(0.7879, 0.2)$. The following models are found:

$$G_1(s) = \frac{0.14464}{18.728s^{0.91746} + 1}$$

and

$$G_2(s) = \frac{0.25881}{27.859s^{0.9115} + 1}.$$

Next, controllers are designed for level control in the upper tank using the FOPID optimization tool of FOMCON toolbox. For this a nonlinear model of the system is used for simulations in the time domain, the set value corresponds to the particular operating point. Linear approximations, corresponding to the working points, are used to constrain the optimization by means of frequency-domain specifications.

Case study (2): Tuning the FOPID controller for the upper tank

We use a two-point GOS scheme, therefore we have two controllers. The specifications are as follows:

- In case of the first controller, a phase margin is set to $\varphi_m \geqslant 60^\circ$, sensitivity and complementary sensitivity function constraints are set such that $\omega_t = 0.02$ and $\omega_s = 0.1$ with $A = B = -20$ dB. Robustness to gain variations specification is also used with the critical frequency $\omega_c = 0.1$.
- For the second controller, the phase margin specification is changed to $\varphi_m = 85^\circ$ and the bandwidth limitation specified by ω_c is removed.

Due to the flexibility of the tuning tool, it is possible to retune the controllers by considering the composite control law during the controller optimization process.

Case study (2): Composite control law and stability test

As a result, two FOPID controllers are obtained:

$$C_1(s) = 6.1467 + \frac{1.0712}{s^{0.9528}} + 0.8497 s^{0.8936}$$

and

$$C_2(s) = 5.1524 + \frac{0.3227}{s^{1.0554}} + 2.4827 s^{0.010722}.$$

The composite control law

$$C(s) = \frac{\left(1 - \gamma(x_1)\right) C_1(s) + \gamma(x_1) C_2(s)}{2}$$

is then verified with both models $G_1(s)$ and $G_2(s)$ using the stability test with step size of $\Delta\gamma = 0.01$ and minimum commensurate order $q_{min} = 0.01$. The result of the test is that the closed-loop systems are stable in case of both fractional models.

Case study (2): Tuning the FOPID controller for the middle tank

Once the gain and order scheduled composite controller is designed, it is plugged into the simulated control system, and a FOPID controller is designed for the second tank using the same optimization tool. In addition, we consider the following:

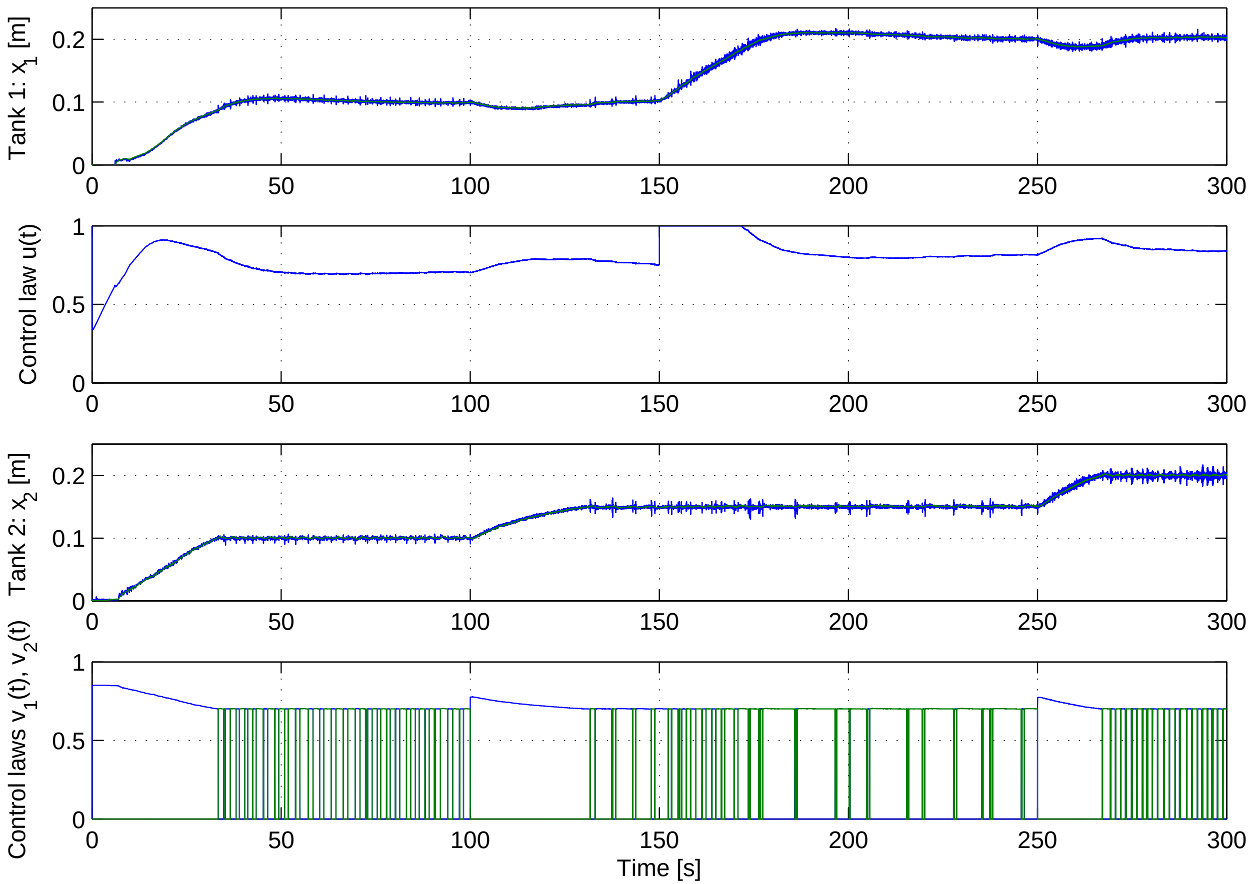
- Frequency-domain specifications are not applicable, since we do not have a linear model of this process.
- The application of the D^{μ} component is not very desirable in this case due to higher levels of noise.

Therefore we design a FOPI controller based only on optimization of the transient response of the control system in the time domain. The following controller is obtained:

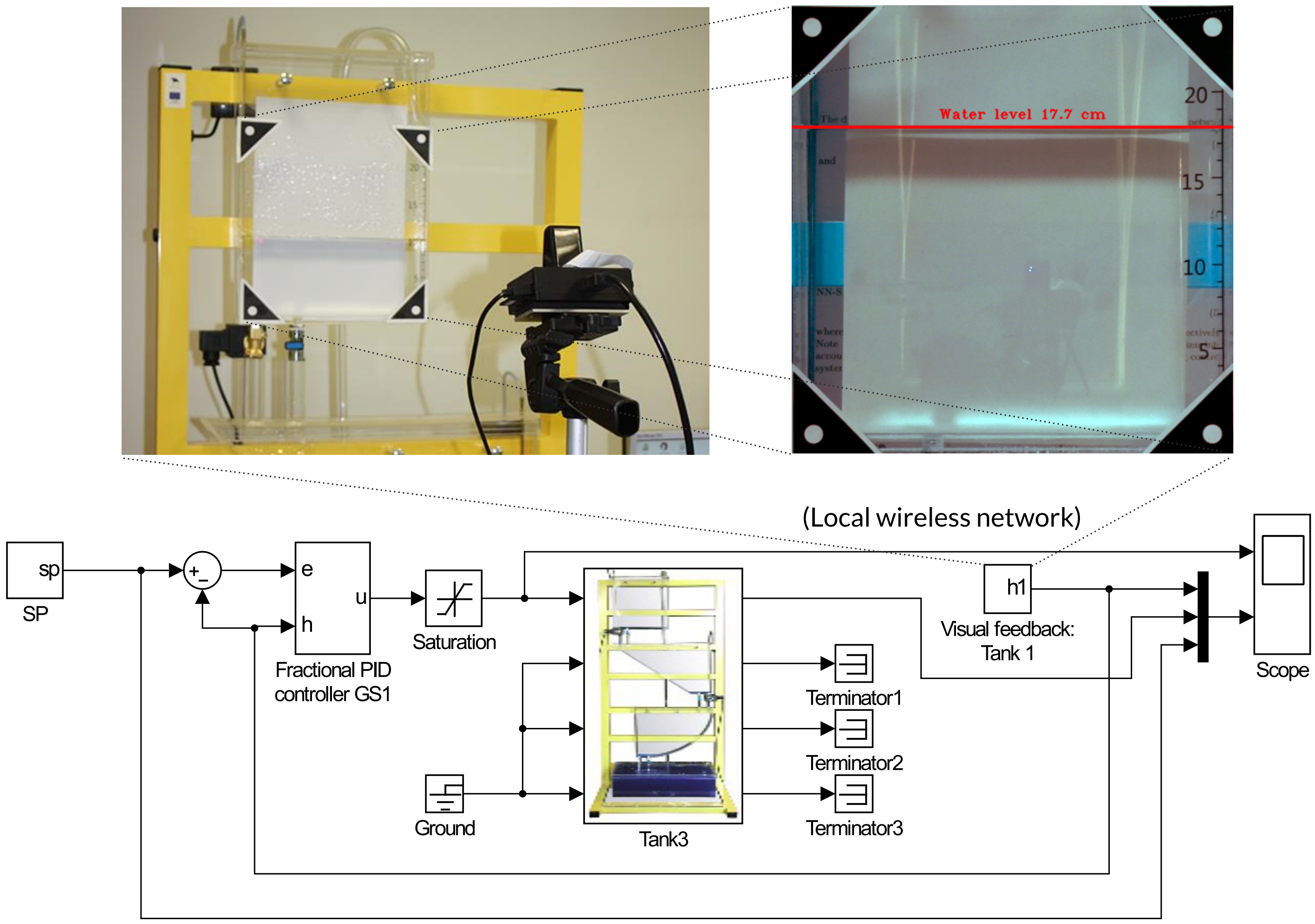
$$C_3(s) = 5.0000 + \frac{0.06081}{s^{0.1029}}$$

which is essentially a proportional controller with a weak fractional-order integrator.

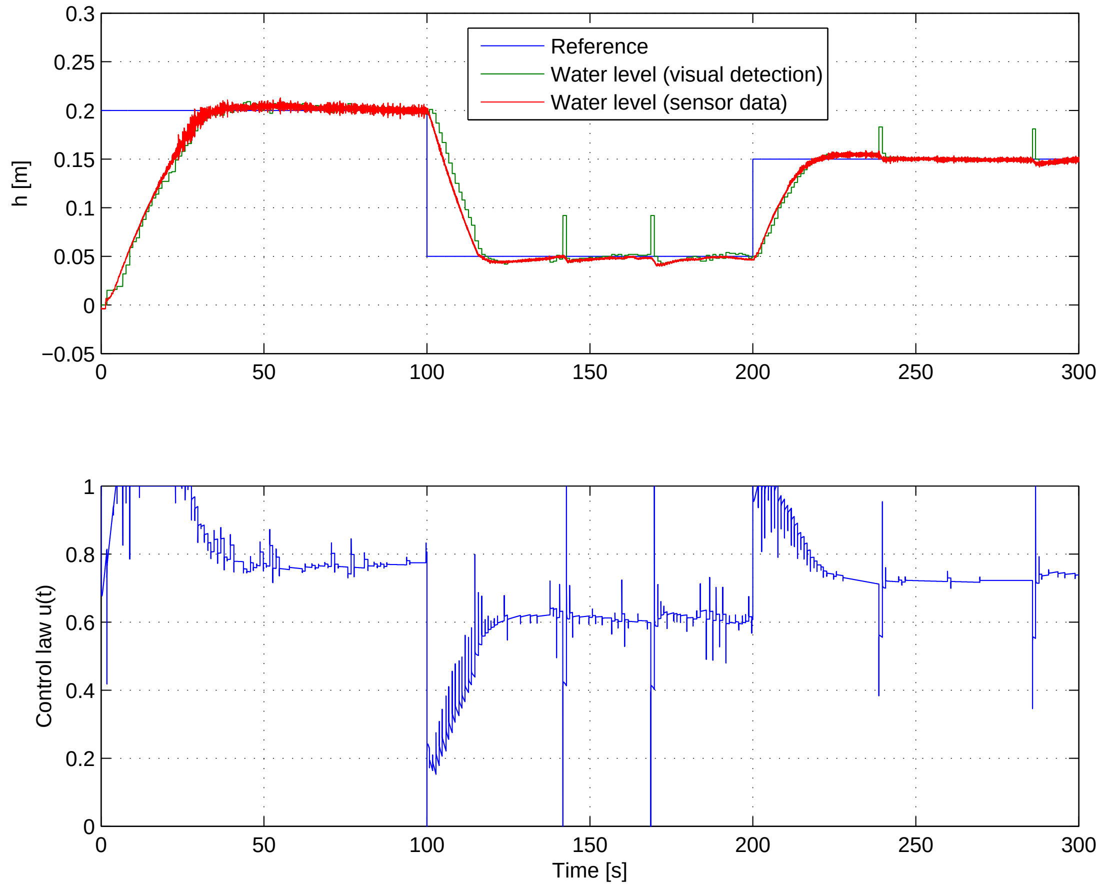
Case study (2): Control system performance

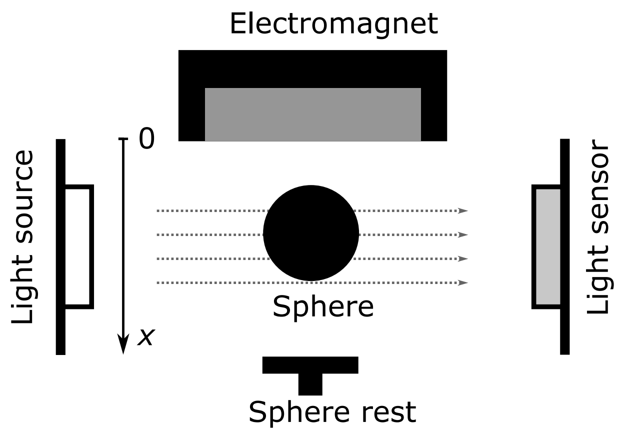


Case study (2): GOS FOPID control of level in the first tank via visual feedback



Case study (2): GOS FOPID control of level in the first tank via visual feedback: Results



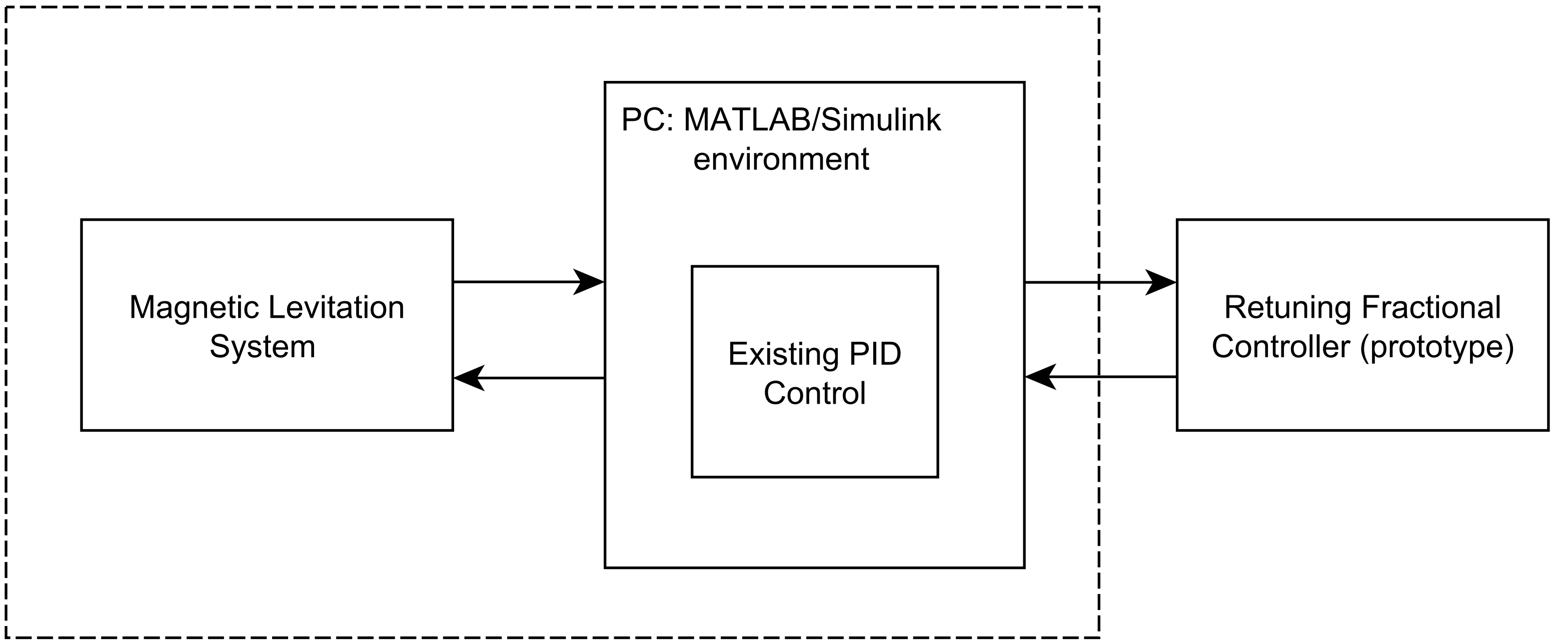


We use the following model of the MLS:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -\frac{c(x_1)}{m}\frac{x_3^2}{x_1^2} + g, \\ \dot{x}_3 &= \frac{f_{ip2}}{f_{ip1}}\frac{i(u) - x_3}{\mathrm{e}^{-x_1/f_{ip2}}}, \end{aligned} \tag{61}$$

where x_1 is the position of the sphere, x_2 is the velocity of the sphere, and x_3 is the coil current, f_{ip1} and f_{ip2} are constants, $c(x_1)$ is a 4th order polynomial and $i(u)$ is a 2nd order polynomial.

Case study (3): Experimental Setup



A real-life MLS is used in this experiment. The MATLAB/Simulink environment acts as an interface between the two devices.

Case study (3): Experimental Results

The following FOPID controller was implemented in the retuning configuration:

$$C_1^*(s) \quad = \quad -45.839 - 18.504 s^{-1.06} - 3.0559 s^{0.94},$$

The parameters of the retuning controller in (52) were computed, and an implementation of the form (57) was obtained using the Oustaloup method with $N = 4$ and $\omega = [0.001, 2/T_s]$, where $T_s = 0.001$s is the desired sample rate.

Case study (3): Experimental Results

Video: `https://youtu.be/NXbqjK6oIcw`

Case study (4): Network approximation of a FO lead compensator

Recall the example, where our goal was to obtain an analog implementation a fractional controller for a model of a position servo

$$G(s) = \frac{1.4}{s(0.7s+1)} e^{-0.05s}.$$

We now provide the results of approximating the controller

$$C(s) = \left(\frac{2.0161s+1}{0.0015s+1} \right)^{0.7020}$$

by an electrical network by using a deterministic method, implemented as part of the unified network generation framework in FOMCON, for obtaining the parameters of the network.

Case study (4): Electrical network approximations

In order to implement it, the following steps are carried out:

- We choose $R_1 = 200k\Omega$ and $C_1 = 1\mu$F due to the time constant τ.
- The basic structure is the Foster II form RC network and the implementation is done by means of the mentioned algorithm.
- To obtain the differentiator, we use the property $Z_d(s) = 1/Z_i(s)$, where $Z_d(s)$ and $Z_i(s)$ correspond to impedances of a differentiator and an integrator, respectively.
- This is done by setting the impedances in the active filter circuit such that $Z_1(s) = Z_i(s)$ and $Z_2(s) = R_k$, where R_k serves as a gain correction resistor.

Case study (4): Electrical network approximations (continued)

```
b  = 2.0161; wz = 1/b;
alpha = 0.702;
Gc = fotf('s')^alpha / wz^alpha;

params = struct; params.R1 = 200e3;
params.C1 = 1e-6; params.N = 4;
params.varphi = 0.01;

imp2 = frac_rcl(1/Gc, ...
       'frac_struct_rc_foster2', ...
       'frac_imp_rc_foster2_abgen', ...
       logspace(-2,3,1000), ...
       params);
```

Case study (4): Electrical network approximations (continued)

The controller is obtained from the object using

```
C = 1/zpk(imp2);
```

Now we set the resistor values to the preferred series with 5% tolerance, and the capacitor values are substituted for closest components out of the 10%-series:

```
imp2 = imp2.prefnum('5%','10%',[],'5%');
```

Finally, the bill of materials can be generated using `engnum()`:

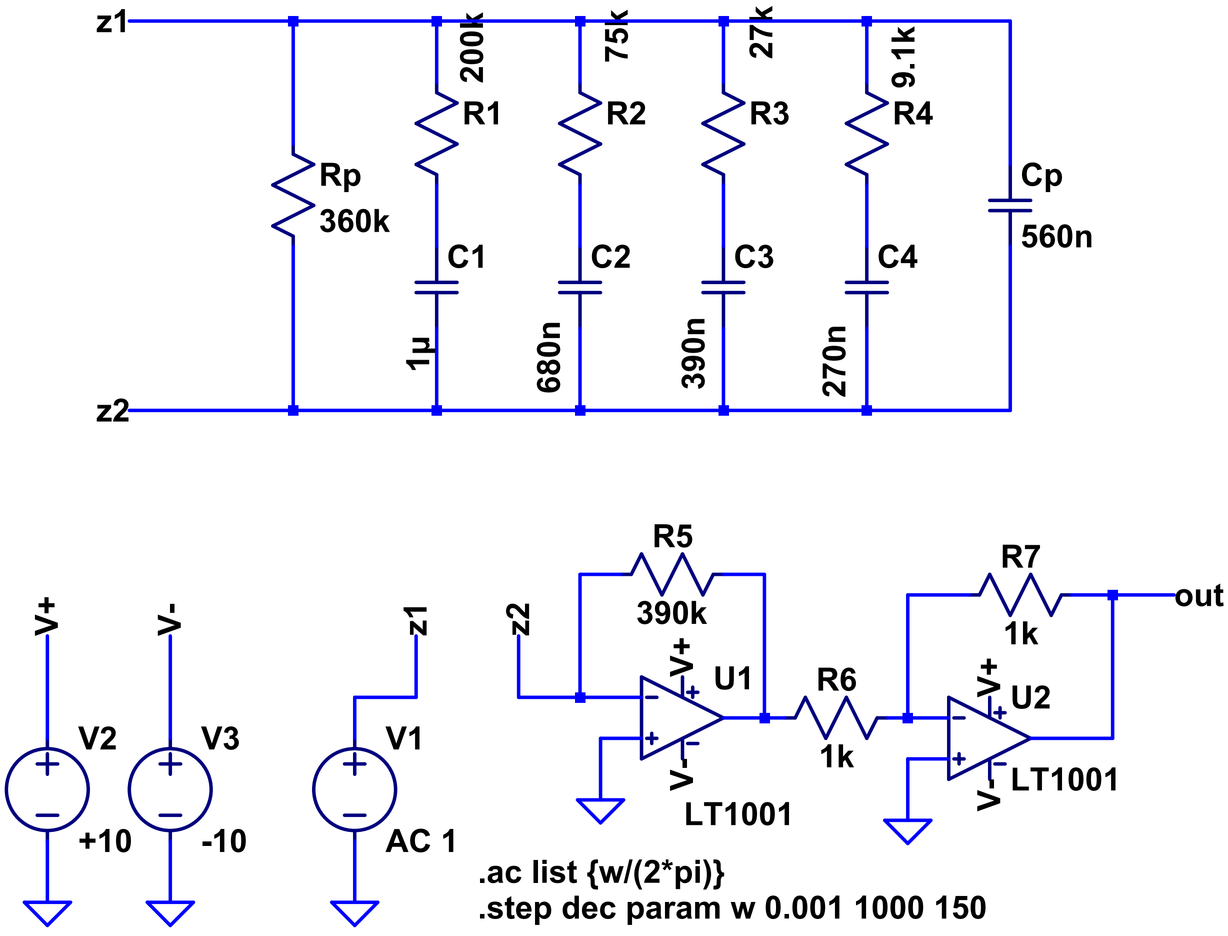
```
[vals, str] = engnum(imp2.R);
```

The variable `str` will contain the following:

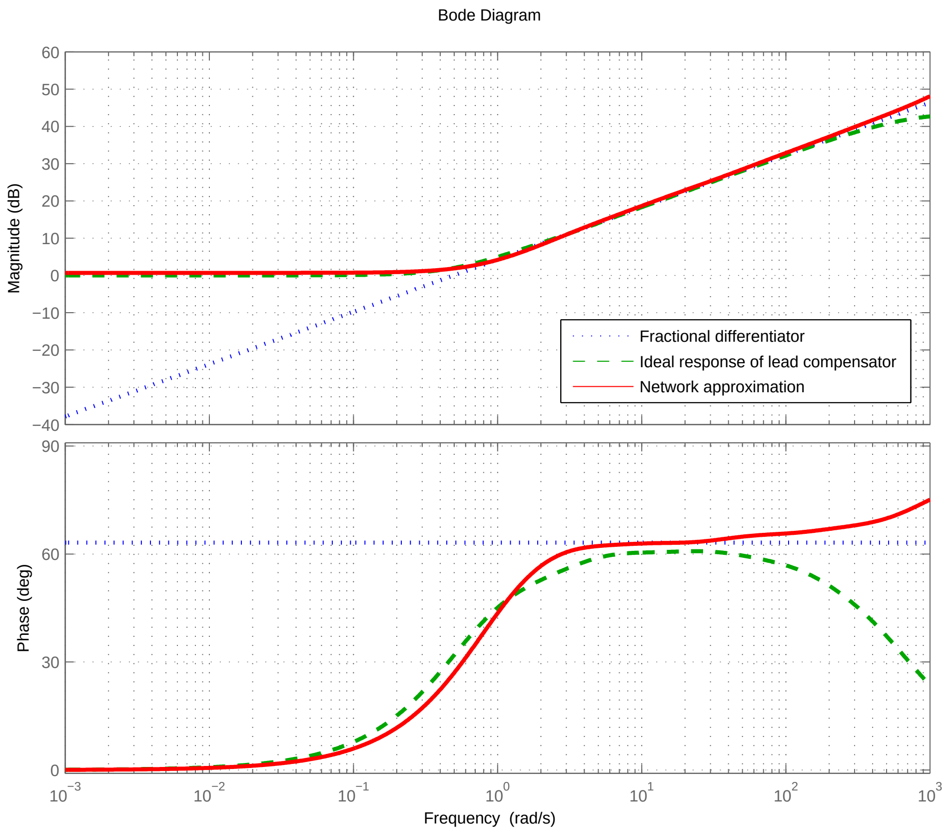
```
'360 k'   '200 k'   '75 k'   '27 k'   '9.1 k'
```

The gain setting resistor R_k has the preferred value of $390k\Omega$.

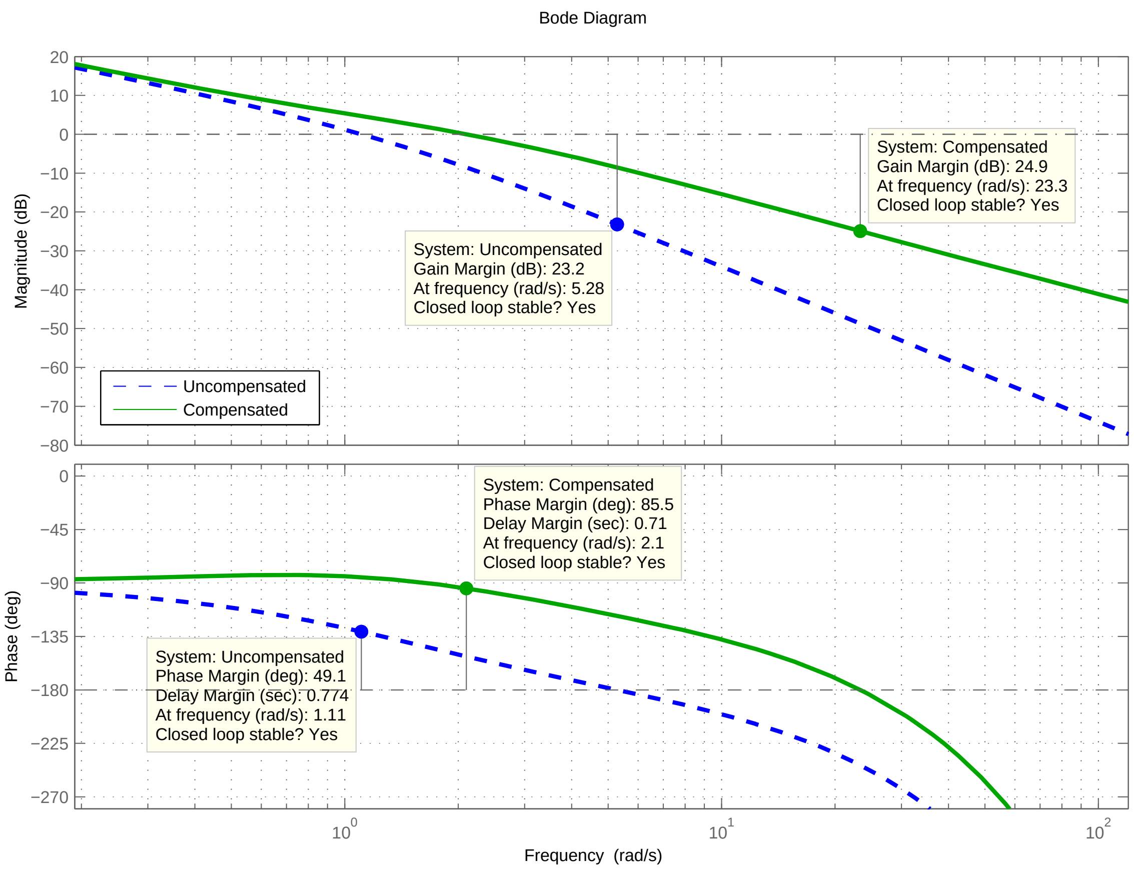
Case study (4): Electrical network approximations (continued)



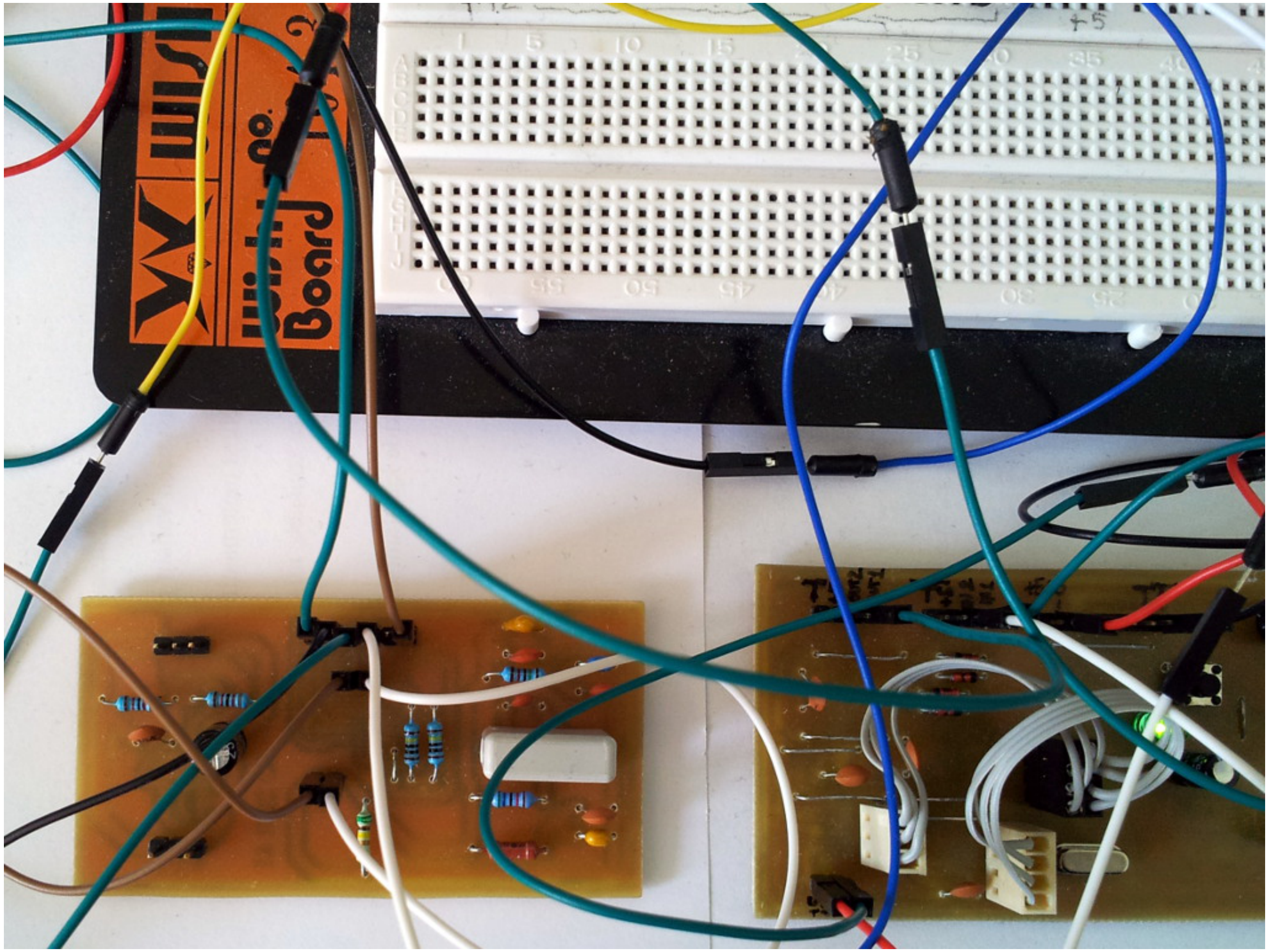
Case study (4): Electrical network approximations (continued)



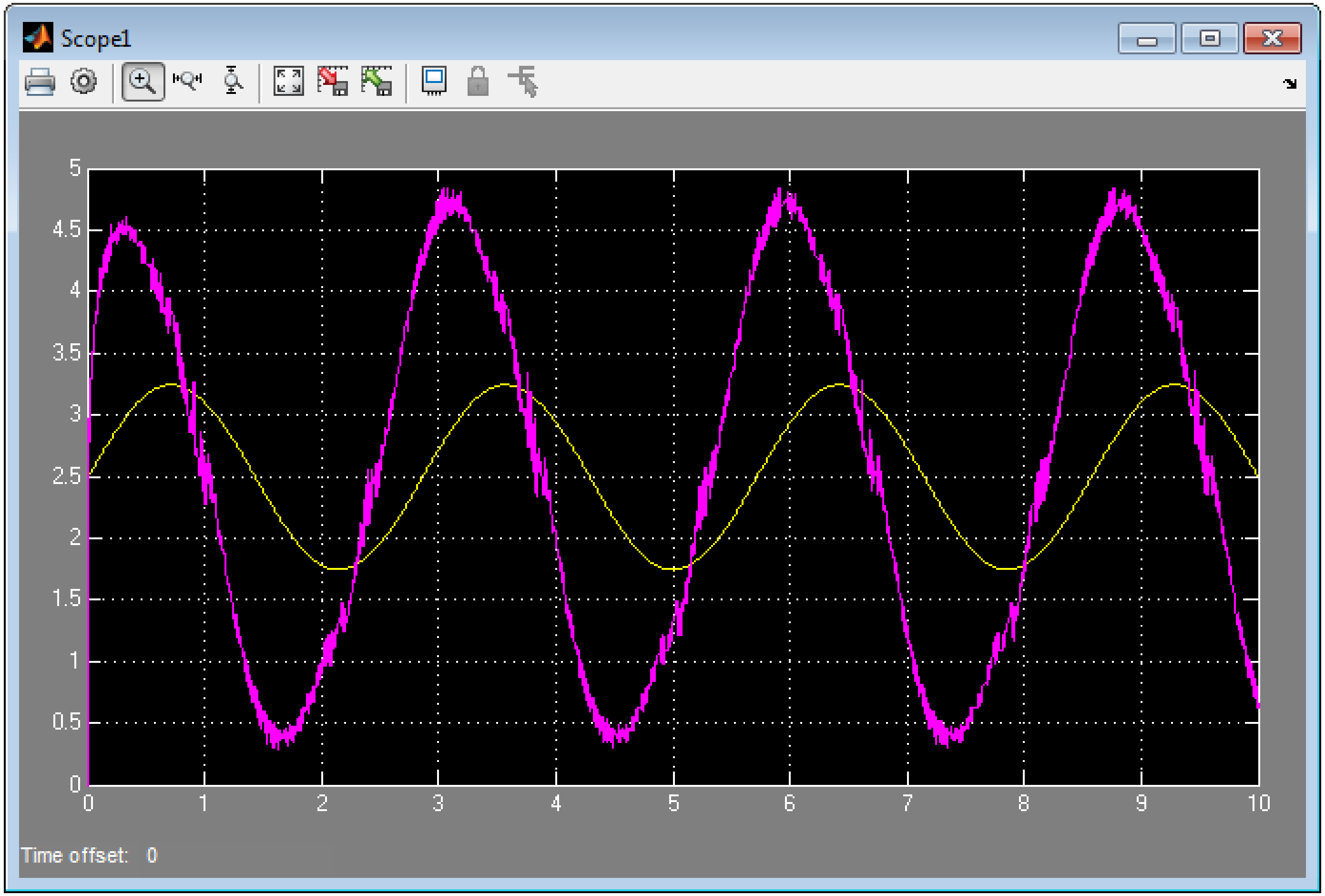
Case study (4): Electrical network approximations (continued)



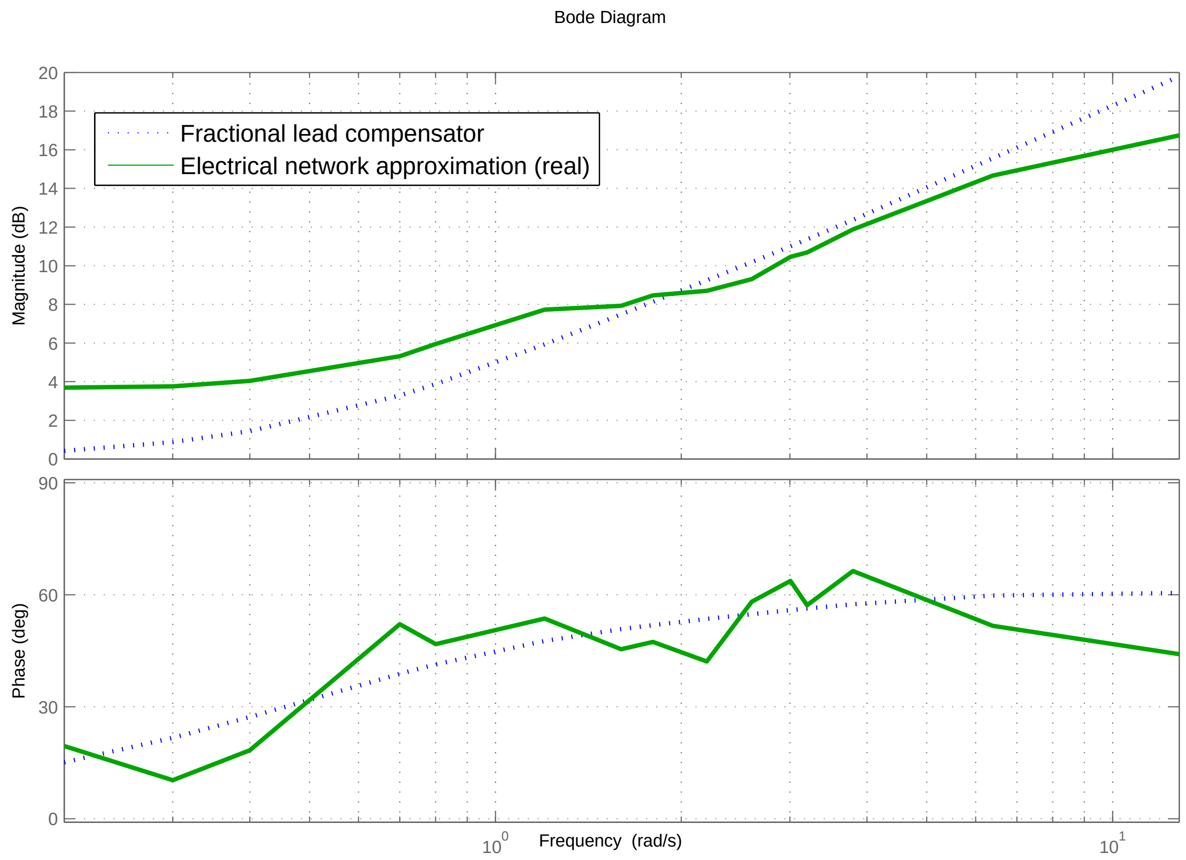
Case study (4): Electrical network approximations (continued)



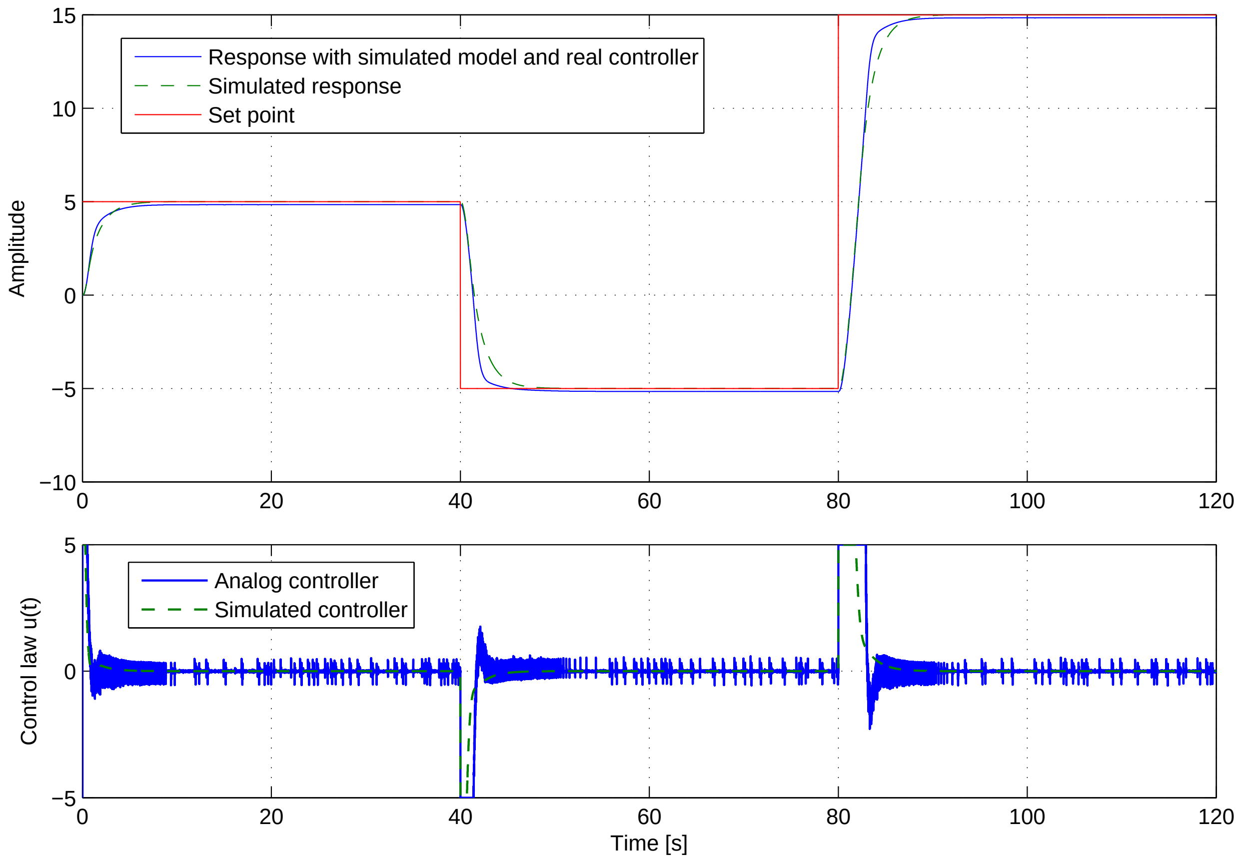
Case study (4): Frequency response around $\omega_{cg} = 2.2$ rad/s



Case study (4): Frequency response around $\omega_{cg} = 2.2$ rad/s



Case study (4): Electrical network approximations: Results



References

[1] K.S. Miller, B. Ross, *An Introduction to the Fractional Calculus and Fractional Differential Equations*, Wiley, New York, 1993.

[2] I. Podlubny, *Fractional Differential Equations*, Academic Press, San Diego, CA, 1999.

[3] C.A. Monje, Y.Q. Chen, B.M. Vinagre, D. Xue, V. Feliu, *Fractional-order Systems and Controls Fundamentals and Applications*, Springer-Verlag, London, 2010.

[4] R. Hilfer, *Applications of Fractional Calculus in Physics*, World Scientific, Singapore, 2000.

[5] I. Podlubny, *Geometric and Physical Interpretation of Fractional Integration and Fractional Differentiation, Fractional Calculus and Applied Analysis*, vol. 5, no. 4, pp. 367-386, 2002.