# A Fractional-Order Control Approach to Ramp Tracking with Memory-Efficient Implementation

Christoph Weise*, Rafael Tavares†, Kai Wulff*, Michael Ruderman†, Johann Reger*◇

***Abstract*— We investigate the fractional-order (FO) control of arbitrary order LTI systems. We show that, for ramp tracking or input disturbance rejection, it is advantageous to include an FO integrator to the open-loop if we have to increase the order of integration further than one. With the lower phase-loss of the FO integrator it is easier to guarantee a desired phase margin. Furthermore the flat phase response around the crossover-frequency (iso-damping property) can be achieved for a wider frequency range such that the closed-loop is more robust wrt. amplitude and phase margins.**

**The drawback of the FO approach is the increased implementation effort and the algebraic decay, which slows down the transient response for larger times. The algebraic decay can be reduced by placing the fractional closed-loop poles to the corresponding integer-order poles. The remaining FO transfer zeros are compensated by an additional filter. We acheieve a more efficient implementation by reducing the memory needed by a direct discretization of the Grünwald-Letnikov definition. As the controller design is done in the frequency domain, we investigate the effect of the different memory truncations. All strategies are demonstrated by simulation.**

## I. Introduction

Since the late seventies fractional-order (FO) approaches gained increasing attention to control SISO systems. Generalizing the PID controller to the FO domain allows to introduce additional constraints to controller design, e.g. the iso-damping property increases the robustness against gain variations. Many papers concern the tuning of such controllers, see [1], [2], [3]. In [4] the pole-placement technique is used with an additional filter. Other two degree of freedom approaches to FO control are given in [5], [6], [7].

Although the increased adjustability of the controllers is enormous, these controllers show two main drawbacks: First, the implementation requires large memory. Secondly, the algebraic decay of the introduced FO operators slows the closed-loop response for large times even for large gains.

We discuss the advantages of the FO control over integer-order (IO) approaches in the case of open-loop shaping. The slow convergence is, however, still limiting the system performance. Therefore we discuss the possibility to use the pole-placement approach to introduce IO dynamics to the close-loop, despite the FO dynamics of the open loop.

The required memory for implementation is reduced using modified memory weights of the Grünwald-Letnikov (GL) discretization. We investigate the effect of the different memory truncations and compare the results to standard methods, e.g. higher-order IO approximations [2].

The paper is organized as follows. Section II holds definitions of FO derivatives and FO LTI system stability criteria. Section III describes FO controller design for ramp tracking, for open-loop design and the 2-DOF structure with pole-placement to IO poles and pre-filter shaping. The real-time memory implementation of the discrete GL version is given in Section IV, along with frequency and time response comparisons. Simulations of the implemented controller are detailed in Section V, followed by conclusions in Section VI.

## II. Preliminary Results and Definitions

### A. Fractional-Order Derivatives

There are different approaches to generalize the IO derivative to the non-IO case. An intuitive approach is the generalization of the difference quotient leading to the definition of Grünwald and Letnikov [8]

$$ {}^{\mathrm{G}}_{t_0}\mathcal{D}^{\alpha}_{t} f(t) = \lim_{h\to 0} \frac{1}{h^{\alpha}} \sum_{k=0}^{\lfloor (t-t_0)/h \rfloor} (-1)^k \binom{\alpha}{k} f(t-kh) \quad (1) $$

with $\alpha \in \mathbb{R}$, the floor integer part $\lfloor \cdot \rfloor$ and the generalized binomial coefficient given by Euler's Gamma function, i.e.

$$ \binom{\alpha}{k} = \frac{\Gamma(\alpha+1)}{k!\Gamma(\alpha-k+1)}. \quad (2) $$

This definition gives a natural approach to implement FO derivatives in a discrete time setting. However, it is difficult to use, so differently defined operators are of interest. In [8] it is shown, that for $f \in \mathcal{C}^1$ the GL approach coincides with the definition of Riemann and Liouville:

$$ {}^{\mathrm{R}}_{t_0}\mathcal{D}^{\alpha}_{t} f(t) = \frac{\mathrm{d}^m}{\mathrm{d}t^m} \left( \frac{1}{\Gamma(m-\alpha)} \int_{t_0}^{t} \frac{f(t)}{(t-\tau)^{\alpha-m+1}} \mathrm{d}\tau \right). \quad (3) $$

For engineering praxis, yet Caputo's FO derivative [2], [8] is used very often. It is given by

$$ {}_{t_0}\mathcal{D}^{\alpha}_{t} f(t) = \frac{1}{\Gamma(m-\alpha)} \int_{t_0}^{t} \frac{f^{(m)}(t)}{(t-\tau)^{\alpha-m+1}} \mathrm{d}\tau, \quad (4) $$

where $\alpha \in \mathbb{R}^+$ is the differentiation order and $m$ is an integer with $m-1 < \alpha < m$. The initial conditions of this operator have a similar interpretation as in the classical IO case. The Laplace transform with $F(s) = \mathcal{L}\{f(t)\}$, see [8], [2], is

$$ \mathcal{L}\left\{ {}^{\mathrm{R}}_{0}\mathcal{D}^{\alpha} f(t) \right\} = s^{\alpha} F(s) - \sum_{k=0}^{m-1} s^{k} \left[ {}^{\mathrm{R}}_{0}\mathcal{D}^{\alpha-k-1} f(t) \right]_{t=0}, \quad (5) $$

$$ \mathcal{L}\left\{ {}_{0}\mathcal{D}^{\alpha} f(t) \right\} = s^{\alpha} F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1} f^{(k)}(0). \quad (6) $$

* Control Engineering Group, Technische Universität Ilmenau, P.O. Box 10 05 65, D-98684, Ilmenau, Germany

† Department of Engineering Science, University of Agder, P.O. Box 422, 4604 Kristiansand, Norway, michael.ruderman@uia.no

◇ Corresponding author: johann.reger@tu-ilmenau.de

**Author´s accepted manuscript**
**The manuscript accepted to publication in**
**IEEE International Workshop on Advanced Motion Control (AMC2020)**

Hence the operators only lead to similar results if the initial conditions are zero. If we are only considering transfer functions, any of the presented operators can be applied. Due to available numerical tools ([9]) we refer to Caputo's operator for simulation studies, and show an approximation of the GL operator for real-time implementation.

Due to the contained FO integral with its integration limits, the online evaluation of an FO controller needs the complete past of the controller input, i.e. $f(t)$ for $t \in [t_0, t]$. As this is not possible, only the recent past is taken into account by applying the short-memory principle [8] and reducing the time interval to $t \in [t-L, t]$. This leads to the following error bound for $|f(t)| \leq M$, $L < t < t_1$:

$$\varepsilon(t) = |{}_{t_0}D^{\alpha} f(t) - {}_{t-L}D^{\alpha} f(t)| \leq \frac{ML^{\alpha}}{\Gamma(1-\alpha)}. \quad (7)$$

### B. Fractional-Order LTI System

The FO SISO LTI system using Caputo's operator with $t_0 = 0$ is given by

$$\Sigma_{\text{FO}} : \begin{cases} \mathcal{D}^{\alpha} x(t) &= Ax(t) + Bu(t), \quad x(0) = x_0 \\ y(t) &= Cx(t) + Du(t) \end{cases}$$

with the (pseudo) state $x(t) \in \mathbb{R}^n$, input $u(t) \in \mathbb{R}$, output $y(t) \in \mathbb{R}$, order of differentiation $\alpha \in (0, 1]$ and real-valued matrices $A, B, C$ and $D$ of matching dimensions. The solution of this initial value problem can be formulated in terms of the Mittag-Leffler Function $\mathcal{E}_{\alpha,\beta}(\cdot)$, i.e.

$$x(t) = \mathcal{E}_{\alpha,1}(At^{\alpha})x_0 + \int_0^t \tau^{\alpha-1}\mathcal{E}_{\alpha,\alpha}(A\tau^{\alpha})Bu(t-\tau)\mathrm{d}\tau \quad (8)$$

with
$$\mathcal{E}_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \beta)}. \quad (9)$$

Note that the scalar Mittag-Leffler Function converges with an algebraic decay, i.e. for $t \to \infty$ with $\alpha \in (0, 1)$ we have $\mathcal{E}_{\alpha,1}(-t^{\alpha}) \sim t^{-\alpha}/\Gamma(1-\alpha)$ [8].

**Theorem 1** (FO-LTI System Stability [10])**.** *The origin of the system $\Sigma$ is asymptotically stable iff*

$$|\arg(\lambda_i)| > \alpha\frac{\pi}{2}, \quad \forall i = 1, 2, \ldots, n, \quad (10)$$

*where $\lambda_i$ denotes the $i$-th eigenvalue of $A$.*

For $\alpha < 1$ the suitable region for pole-placement is enlarged compared to IO systems, because eigenvalues with positive real part are possible. This region however is not convex any more. For zero initial conditions ($x(t) = 0,\ \forall t < t_0$) its transfer function is given by

$$G(s) = C\left(s^{\alpha}I - A\right)^{-1}B + D = \frac{B_G(s^{\alpha})}{A_G(s^{\alpha})} \quad (11)$$

with the pseudo-polynomials

$$A_G(s^{\alpha}) = a_n s^{n\alpha} + a_{n-1}s^{(n-1)\alpha} + \cdots + a_1 s^{\alpha} + a_0 \quad (12)$$

$$B_G(s^{\alpha}) = b_n s^{n\alpha} + b_{n-1}s^{(n-1)\alpha} + \cdots + b_1 s^{\alpha} + b_0. \quad (13)$$

**Theorem 2** (BIBO Stability of FO-LTI systems [10])**.** *The fractional-order system with transfer function $G(s) = B_G(s^{\alpha})/A_G(s^{\alpha})$ and commensurate order $\alpha$ is bounded-input-bounded-output stable iff*

$$|\arg(p_i)| > \alpha\frac{\pi}{2} \quad \forall p_i \in \mathbb{C} \quad A_G(p_i) = 0$$

*with $p = s^{\alpha}$ and $p_i$ the roots of $A_G(p)$.*

## III. Controller Design

### A. Improved Robustness using FO Loop-Shaping

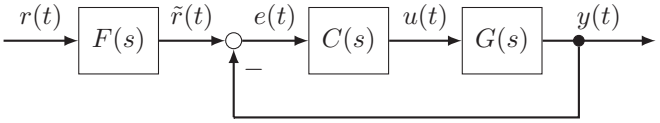


Fig. 1: Controller structure including the prefilter $F(s)$.

We consider the regulator design problem as shown in Fig. 1 (with $F(s) = 1$ in this section). We will compare simple IO and FO controller designs to shape the open-loop, based on the crossover frequency $\omega_s$, phase margin $\Phi_R$ and iso-damping property [2]:

$$\frac{\mathrm{d}}{\mathrm{d}\omega}\arg\left(C(j\omega)G(j\omega)\right)\Big|_{\omega=\omega_s} = 0. \quad (14)$$

We require more than one free integrator in the open-loop design, such that increasing signals (ramps) can be tracked or compensated. Furthermore the additional integrating action increases the robustness against input-disturbances if the process already shows a free integrator. The open-loop transfer function can be parametrized as

$$L(s) = C(s)G(s) = V\frac{N(s)}{s^{\rho}\tilde{D}(s)}, \quad N(0) = \tilde{D}(0) = 1, \quad (15)$$

with $\rho \in \mathbb{R}$. For tracking ramp-shaped reference signals, e.g. $r(t) = t$, the open loop $L(s) = C(s)G(s)$ has to contain more than one free integrator. This is evident by investigating the error $e(t)$ for large times. Assuming a stable closed-loop, we apply the final value theorem of the Laplace transform

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s(R(s) - Y(s))$$

with the closed-loop $T(s) = L(s)/(1 + L(s))$ and the ramp $R(s) = \frac{k}{s^2}$. We end up with

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} s\frac{R(s)}{1+L(s)} = \lim_{s\to 0} s\frac{s^{\rho}\tilde{D}(s)\ R(s)}{s^{\rho}\tilde{D}(s) + N(s)}$$
$$= \left(\frac{0^{\rho-1}\tilde{D}(0)}{0^{\rho}\tilde{D}(0) + N(0)}\right)k,$$

which is only zero if $\rho > 1$. For IO controllers this is problematic due to the loss of phase with each integrator. In order to stabilize the closed-loop, the phase-loss has to be compensated by introducing minimum phase zeros to the controller $C(s)$. There are methods like the symmetrical optimum, but bandwidth is limited and the performance might be poor, especially for non-minimum phase systems.

If we consider FO integrators in the open-loop, however, we can reduce the initial phase loss, because the phase of the fractional-order operator scales with its order, hence

$$\arg\left((j\omega)^{-\alpha}\right) = -\alpha\frac{\pi}{2}. \quad (16)$$

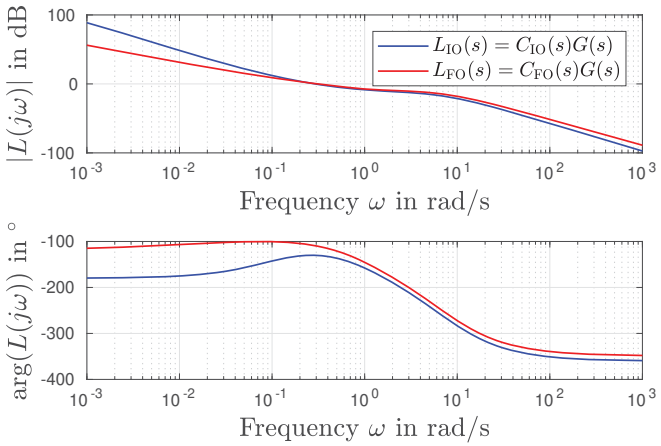


Fig. 2: Bode plot of the integer- and fractional-order open-loops, for $\omega_s = 0.3\,\mathrm{rad\,s^{-1}}$.

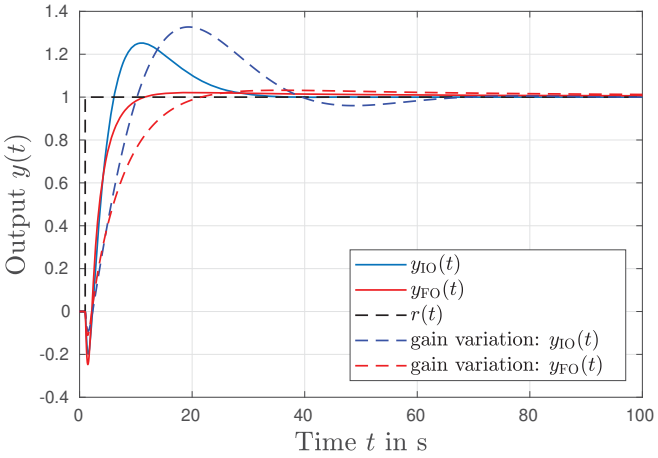


Fig. 3: Step responses of both controllers: nominal gain (solid line), wrong gain (dashed line).

So for $1 < \rho < 2 - \epsilon$ we have less phase loss and require less additional transfer-zeros.

**Example 1.** *We consider the IO non-minimum phase plant*

$$G(s) = V\frac{-s + s_0}{s(s + s_{p,1})(s + s_{p,2})} \tag{17}$$

*with $s_0 < s_{p,1} < s_{p,2}$. The IO controller for ramp tracking needs to compensate the non-minimum phase zero to maintain a phase of $-180°$. For a positive phase margin, an additional lead-lag part is needed. Yet, we have to limit the center frequency of this additional pole-zero pair to maintain a monotonically decreasing amplitude response of the open loop. The overall controller with $\tilde{s}_0 < s_0$ is given by*

$$C_{\mathrm{IO}}(s) = K\frac{(s + \tilde{s}_0)}{s} \tag{18}$$

*where gain $K$ is adapted to the required crossover frequency.*

*The FO approach, however, starts with a higher phase for the low frequency-range. With an order $\alpha$ close to zero, the tracking performance is sluggish, for $\alpha$ close to one, the phase loss is too drastic. Therefore we shall choose $\alpha$ in a reasonable range: $\alpha \in [\frac{1}{3}, \frac{2}{3}]$. Note that an additional FO (pseudo) zero $(s^\alpha - s_0)$ can be placed arbitrarily, since it only increases the slope of the amplitude response by $20\alpha\,\mathrm{dB}$. An additional-fractional-order lead-lag pair with $\beta < 1 - \alpha$ can be added in order to increase the controller bandwidth. The resulting fractional-order controller is given by*

$$C_{\mathrm{FO}}(s) = K\frac{(s^\alpha + s_0)}{s^\alpha}\frac{(\tau_1 s^\beta + 1)}{(\tau_2 s^\beta + 1)}. \tag{19}$$

*The results of this tuning approaches are illustrated in Fig. 2 with the parameter set given in Section V. Both controllers are tuned to have a crossover frequency of $\omega_s = 0.5\,\mathrm{rad\,s^{-1}}$ and a phase margin of $\Phi_R = 45°$. We can see that the FO approach leads to a better phase response for the lower frequency range. The phase is almost constant (iso-damping property), therefore this controller is more robust against variations of the closed-loop gain than the IO controller. Fig. 3 shows the step responses of both closed-loop systems. The FO approach can reduce the overshooting and its performance is not disturbed by the variation of the stationary gain of 50 %, $\tilde{V} = \frac{V}{2}$. The overshooting only changes slightly.*

**Remark 3.** *Note that in the FO loop shaping we are also able to partially compensate non-minimum phase zeros [11], by the FO decomposition in pseudo zeros, e.g. we can rewrite the positive zeros at $s_0 = 1$ with $\alpha = \frac{1}{2}$ as per*

$$(s - 1) = (s^\alpha + 1)(s^\alpha - 1).$$

**Remark 4** (Discrete time implementation)**.** *For small orders $\alpha$ the controller's pseudo zero/pole $s^*$ should not be chosen too fast, since its corner frequency in the bode plot is shifted by the order, i.e.*

$$\omega^* = |s^*|^{\frac{1}{\alpha}}, \tag{20}$$

*which leads to oscillations and instability, if the sampling frequency is not taken into account.*

All in all we see that the FO integrators have an advantage in the open loop. Yet, they introduce an algebraic decay to the closed loop response, even for IO processes to be controlled.

### *B. Pole-Placement to Integer-order poles*

FO systems show an algebraic decay, which reduces the performance for large times. The stationary behavior seems like a stationary deviation. This effect is counteracted increasing the controller gains. But this is limited due to stability, input saturation and noise amplification. To avoid this slow algebraic convergence, the memory reset of the FO controller is proposed in [12]. This method, however, still is limited by the initial design of the FO controller. Furthermore, the processes need to be integer.

In the following section we explore the possibility to design the controller based on a pole-placement approach in order to increase the controller bandwidth and reduce the effect of the algebraic convergence.

Similar to [4], we propose the output measurement based pole-placement for commensurate Fo systems with the aim of increasing the convergence. In [4] a slightly different controller structure is applied and the computed controllers show

a negative relative degree which restricts the implementation to higher-order approximations.

The basic idea of the pole-placement is to reduce the FO dynamics to an IO one. In [13] the connection of the eigenvalues of an IO system and its FO representation is discussed. Using a pole-placement approach we set the pseudo poles wrt. the closed-loop transer function

$$T(s^\alpha) = \frac{P_T(s^\alpha)}{Q_T(s^\alpha)} \tag{21}$$

such that the remaining pseudo polynomial is reduced to an ordinary polynomial.

We consider the FO system given by its transfer function (11) with the commensurate order $\alpha$. We can rewrite the pseudo polynomials $A_G(s^\alpha)$ and $B_G(s^\alpha)$ as polynomials in $p$ by substituting $s^\alpha = p$. With the controller of appropriate order and the same commensurate order $\alpha$ we have

$$C(s) = \frac{P(s^\alpha)}{Q(s^\alpha)} = \frac{P(p)}{Q(p)} \tag{22}$$

and are left to solve the Diophantine equation in $p$ to place the poles of the closed-loop with the desired polynomial $Q_T$:

$$Q_T(p) = A(p)Q(p) + B(p)P(p). \tag{23}$$

For these problem we can use standard methods to place the poles of the closed-loop, e.g., algebraic methods based on the Silvester's matrix. For the order of these pole-placement compensator $n_c$ there are in general two options [14]. With $n_c = n + \kappa$, the controller is strictly proper. The controller order is increased by $\kappa$ which reflects the number of FO integrators contained in the controller. The minimal order $n_c = n+\kappa-1$ typically is bi-proper. Hence, there is a choice to adjust the order $m = n + n_c$ of the desired denominator of the closed-loop transfer funtion, i.e.

$$Q_T(p) = c_m p^m + ... + c_1 p + c_0. \tag{24}$$

We propose to choose the poles such that all the coefficients of non-integer power of $s$ vanish. For $\alpha^{-1} \in \mathbb{N}$, for the desired poles we have

$$\lambda_{\mathrm{IO}} = \lambda_{FO}^{1/\alpha}, \tag{25}$$

where each IO pole $\lambda_{\mathrm{IO}}$ leads to $\alpha^{-1}$ corresponding FO pseudo poles $\lambda_{\mathrm{FO}}$.

Note that the order $m$ has to match the order of differentiation $\alpha$ such that

$$m\alpha = (n + n_c)\alpha \in \mathbb{N}, \tag{26}$$

otherwise not all FO pseudo poles can be moved to their IO counterparts.

If the order of the process $n$ and the commensurate order $\alpha^{-1} \in \mathbb{N}$ do not match, the controller needs to fix a number of additional open-loop pseudo poles $n_{\mathrm{add}}$ in advance, i.e.

$$(n + 2n_{\mathrm{add}} + n_c)\alpha \in \mathbb{N}. \tag{27}$$

Altogether there are three possibilities to match the orders. First of all, we can chose a full or minimal order compensation structure. In addition, we can split the additional pseudo poles between the integrator constraints $\kappa$ and additional filters added to the plant $n_{\mathrm{add}}$. For controlling IO systems with $\alpha = \frac{1}{2}$ with a minimal controller order aiming for $n_c = n$, we can chose $\kappa = 1$ and $n_{\mathrm{add}} = 0$ resulting in $n_c = 2n + 1 - 1$ and $m = 4n$. For $\alpha = \frac{1}{3}$, however, we might also choose $\kappa = 0$ and $n_{\mathrm{add}} = 2$.

Solving the pole-placement problem leads to a closed-loop denominator of (21) which is IO, but the numerator is still fractional. The remaining FO zeros slow down the input-response of the system, as the solution is still given by the Mittag-Leffler function with its algebraic decay. Hence, the controller structure is extended by a prefilter $F(s)$ (see Fig. 1) shaping the reference, such that the effect of the FO transfer zeros is canceled (c.f. Fig. 4). Thus, we split the numerator $P_T(s^\alpha)$ into a minimum and non-minimum phase part

$$P_T(s^\alpha) = P_{Tb}(s^\alpha) \cdot P_{Tg}(s^\alpha) \tag{28}$$

with $P_{Tb}(\lambda_i) = 0, |\arg(\lambda_i)| < \alpha\pi/2$ and $P_{Tg}(\lambda_i) = 0, |\arg(\lambda_i)| \geq \alpha\pi/2$. The part $P_{Tg}$ can be easily compensated while the components of $P_{Tb}$ can only be moved to its IO non-minimum phase counterparts. Additional IO poles $Q_{\mathrm{add}}(s)$ are needed to guarantee the causality of the filter

$$F(s) = \frac{N_F(s)}{D_F(s)} = \frac{\bar{P}_{Tb}(s^\alpha)}{P_{Tg}(s^\alpha)Q_{\mathrm{add}}(s)}. \tag{29}$$

Finally, the input response of the closed-loop decays exponentially since the control removes any FO memory effect:

$$\frac{Y(s)}{R(s)} = F(s)T(s) = \frac{P_{Tb}(s^\alpha)\bar{P}_{Tb}(s^\alpha)}{Q_T(s)Q_{\mathrm{add}}(s)}. \tag{30}$$

**Example 2.** *Consider the non-minimum phase FO process*

$$G(s) = \frac{-0.2s^\alpha + 1}{s + 2s^\alpha + 1}, \quad \alpha = \frac{1}{2} \tag{31}$$

*and set the closed-loop poles to $\lambda_{\mathrm{IO}} = -2 \pm j$. This results in the closed-loop given by*

$$T(s) = \frac{-1.06s^{3\alpha} + 4.89s^{2\alpha} + 0.94s^\alpha + 5}{s^{4\alpha} + 0s^{3\alpha} + 4s^{2\alpha} + 0s^\alpha + 5}. \tag{32}$$

*The numerator shows one nonminimum-phase pseudo zero at $s_0^\alpha = 5$ which cannot be compensated. Therefore the filter is extended by an additional pseudo zero $\bar{s}_0^\alpha = -5$ such that the filter $F(s)$ is given by*

$$F(s) = \frac{0.94737s^\alpha + 4.7368}{5s^{2\alpha} + 1.8421s^\alpha + 4.7368}. \tag{33}$$

*Finally all FO coefficients are are removed from the closed-loop input-output behavior, i.e.*

$$F(s)T(s) = \frac{-0.2s + 5}{s^2 + 4s + 5}. \tag{34}$$

*Fig. 4 shows the step responses of the FO process $G(s)$, the closed-loop with IO poles $T(s)$ and filtered version $F(s)T(s)$. The increased convergence is clearly visible. The effect of the non-minimum phase zero is also reduced.*

Note, as in all prefilter design, the performance is achieved for reference response only. Disturbances and nonzero initial conditions slow down the decay to an algebraic one. Therefore, the proposed methodology is suited best for Fo processes to improve their input response. The disturbance rejection remains fractional-order.

**Remark 5** (Filter requirements for ramp tracking)**.** *A causal prefilter introduces a delay into the filtered reference signal*

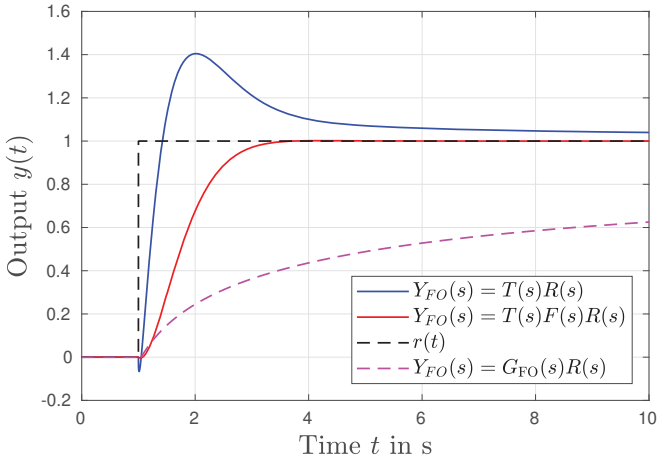


Fig. 4: Step responses of the closed-loop system with integer-order assigned poles.

$\tilde{r}(t)$. *This leads to the tracking error* $\tilde{e}(t) = \tilde{r}(t) - r(t)$. *For ramp-shaped references* $R(s) = ks^{-2}$ *we hence have for* $t \to \infty$*:*

$$\begin{aligned} &\lim_{t\to\infty} \tilde{e}(t) = \lim_{s\to 0} s\tilde{E}(s) = \lim_{s\to 0} s\left(1 - F(s)\right)\frac{k}{s^2} \\ &= \lim_{s\to 0}\left(\frac{A_F(s) - B_F(s)}{A_F(s)}\right)\frac{k}{s} \\ &= \frac{(a_n - b_n)s^{n\alpha} + \cdots + (a_1 - b_1)s^{\alpha} + (a_0 - b_0)}{A(s)}\frac{k}{s}. \end{aligned} \tag{35}$$

*For perfect tracking we require*

$$a_i = b_i \quad \textit{for} \quad i = 0, 1, \ldots, \alpha^{-1} \tag{36}$$

*such that* $\lim_{t\to\infty} \tilde{e}(t) = 0$. *This is a hard constraint, e.g. for* $\alpha = 1/2$ *we might have to use* $b_1 \neq 0$*, however, this introduces an FO zero to the closed-loop behaviour, which slows the system response down.*

## IV. Real-Time Implementation with Reduced Memory

A proposal for real-time implementation of discrete FO dynamical systems and controllers oriented to minimize the computational efforts, consists of applying the GL method and incorporate the short-memory principle [15]

$$\mathcal{D}^{\alpha} f(t) \approx {}_{t-L}\mathcal{D}^{\alpha} f(t), \quad t > L. \tag{37}$$

This implementation allows to select the memory length $L$ to approximate the FO operator. The explicit numerical computation of the FO derivative of a sampled function $f(t)$ uses the approximation [15]

$${}_{t_0}\mathcal{D}^{\alpha} f(t) \approx \frac{1}{h^{\alpha}} \sum_{k=0}^{\lfloor (t-t_0)/h \rfloor} w_k^{(\alpha)} f(t - kh) \tag{38}$$

where $h$ is the sampling interval. The recursive computation of the coefficients $w_k^{(\alpha)}$ is given by

$$w_0^{(\alpha)} = 1, \quad w_k^{(\alpha)} = \left(1 - \frac{\alpha + 1}{k}\right) w_{k-1}^{(\alpha)}, \quad k = 1, 2, ..., \frac{L}{h}.$$

Note that the IO integrator needs to be implemented separately if the (38) should approximate an FO integrator:

$$G_{GL}(s) = \begin{cases} s^{\alpha}, & \text{if} \quad \alpha \in [0, 1] \\ s^{\alpha} = \underbrace{s^{1+\alpha}}_{\text{FO}} \underbrace{s^{-1}}_{\text{Integrator}}, & \text{if} \quad \alpha \in [-1, 0[. \end{cases} \tag{39}$$

To compute the frequency response of the GL operator, consider the discrete-time state space LTI system representation

$$\begin{aligned} \bar{x}(k+1) &= \bar{A}\bar{x}(k) + \bar{B}f(k) \\ \bar{y}(k) &= \bar{C}\bar{x}(k) + \bar{D}f(k) \end{aligned}$$

where $\bar{A} \in \mathbb{R}^{L/h \times L/h}$ is a superdiagonal matrix, the state vector $\bar{x}(k)$ corresponds to the operator history vector and

$$B = \begin{pmatrix} 0 & \ldots & 0 & 1 \end{pmatrix}^T$$

$$C = \begin{pmatrix} w_{\frac{L}{h}}^{(\alpha)} & \ldots & w_2^{(\alpha)} & w_1^{(\alpha)} \end{pmatrix} h^{-\alpha} \quad D = \frac{1}{h^{\alpha}} = \frac{w_0^{(\alpha)}}{h^{\alpha}}.$$

### A. Memory extension

Adaptive time step memory methods have been suggested to achieve efficient computation of the GL derivative with smaller errors during numerical simulations [16], [17]. Although the focus of these works are in efficiently achieve a smaller approximation error, one should acknowledge that the design of FO controllers is mostly based on shaping the response in the frequency domain. Here we focus on the effect of efficient memory methods in the phase margin of real-time implementation of these discrete operators. Consider the accumulated history of the operator output $y$, i.e.

$$y = \left[f(t - kh), ..., f(t - L)\right]^T, \quad k = 1, 2, ..., \frac{L}{h}. \tag{40}$$

With higher sampling times and higher memory lengths, real-time implementation of a full memory operator is not feasible. In order to reach the same memory length $L$ with less required memory, we propose a memory extension combining a short term memory and a long term memory. This approach requires the sampling of $y$ with $T_s$ for the short memory term and with higher sample time $\lambda T_s$ for the long memory term. The approximation is given by combining both memory terms with the same weighting vector $w_k^{(\alpha)}$ as

$$\frac{h_{st}}{L} \sum_{k=1}^{L/h_{st}} w_k^{(\alpha)} y_k^{h_{st}} + \frac{h_{lt}}{L} \sum_{k=1}^{L/h_{lt}} w_k^{(\alpha)} y_k^{h_{lt}}. \tag{41}$$

This approach is simpler and feasible to implement in most real-time hardware, since it requires only two different samplings of the history vector $y$. For minimizing the error, a modification where the long term memory is based on the average instead is also considered. Adaptive memory approaches introduced in [16] focus on efficient memory use through geometric progressions, such as

$$\sum_{k=1}^{\infty} (2i - 1) w_k^{(\alpha)} y_k, \tag{42}$$

and reaching a minimal adaptive memory implementation for the power law

$$\sum_{k=1}^{\infty} (2^{i-1}) w_k^{(\alpha)} y_k. \tag{43}$$

The visualization of the different memory approaches is shown in Fig. 5.

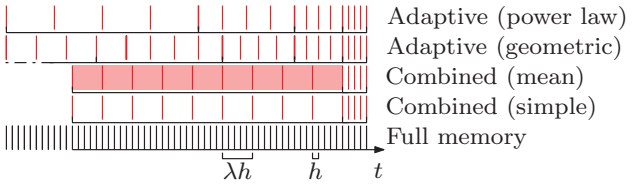


Fig. 5: Memory methods for GL approximation.

The magnitude and phase response of the GL operator with adaptive memory proposed in eq. (41) for $\alpha = 0.5$ is shown in Fig. 6, in comparison with $G(s) = s^{0.5}$, the full memory, and the short memory for $h_1 = T_s$ and $h_2 = 10T_s$ respectively. The effect of periodification is visible for the short memory terms.

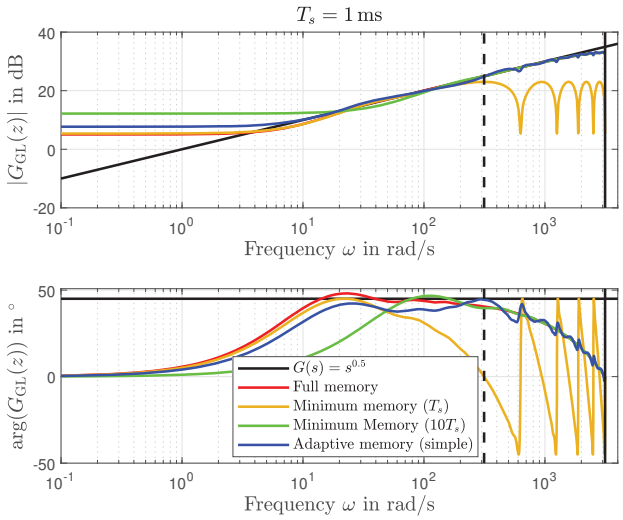


Fig. 6: Comparison of magnitude and phase response of combined memory approach.

For comparison, we consider Oustaloup's recursive filter [18], which approximates the FO operator in the frequency range $[\omega_b, \omega_h]$ with an continuous time IO model of order $2\nu + 1$, hence

$$\hat{G}(s) = \omega_h^{\alpha} \prod_{k=-\nu}^{\nu} \frac{s + \omega_k^-}{s + \omega_k^+}, \; \omega_k^{\pm} = \omega_b \left( \frac{\omega_h}{\omega_b} \right)^{\frac{k+\nu+0.5\pm0.5\alpha}{2\nu+1}}. \tag{44}$$

The magnitude and phase response comparison between the GL operator with adaptive memory (simple) proposed in equation (41), its mean value modification for the long term memory, the minimal adaptive power law equation (43) and the Oustaloup filter equation (44) are shown in Fig. 7.

The combined memory extension shows a similar phase response to the power law implementation and has the advantage of only requiring two different sampling times of the input history. The adaptive power law [16] is the most efficient implementation but requires a non-linear sampling of the history.

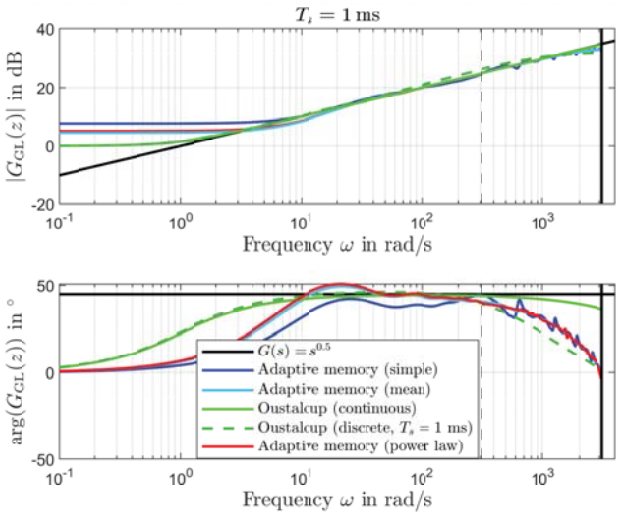


Fig. 7: Comparison of magnitude and phase response of different memory implementations. Parameters: GL-Implementation $T_s = 1\,\mathrm{ms}, N_{\mathrm{short}} = 10, N_{\mathrm{long}} = 100$, Oustaloup: $\omega_l = 1\,\mathrm{rad\,s^{-1}}, \omega_h = 1 \times 10^3\,\mathrm{rad\,s^{-1}}, \nu = 7$.

Even though the Oustaloup continuous filter shows a better approximation for $s^{\alpha}$, the disadvantage of its use is on the online change of parameters in comparison with the GL approach where only the history of the function is stored. The introduced pairs of zeros of poles add additional states which do not necessary have a physical interpretation. Furthermore, the Oustaloup filter performs poor at high frequencies when it is sampled.

In order to compare the different memory approaches in terms of approximation error and computation time, let us consider the fractional derivative of a polynomial $f(t) = t^{\beta-1}$, with analytic solution given by

$$\mathcal{D}^{\alpha} f(t) = \frac{\Gamma(\beta)}{\Gamma(\beta - \alpha)} t^{\beta-\alpha-1}, \quad \beta > 0. \tag{45}$$

The comparison of relative error of approximation (y-axis, in %) versus normalized computation time (x-axis) is shown in Fig. 8, for the different memory methods. Simulations for several memory lengths $L$ are performed for $t \in [0, 10]$ considering $\beta = 2$, $\alpha = 0.5$, $h = 1\,\mathrm{ms}$ and $\lambda = 10$. Increasing $L$ and decreasing $h$ results in an improved error of approximation at cost of increased computation times.

## V. Simulation Studies

We consider the previously discussed example

$$G(s) = \frac{1}{2} \frac{-s + 1}{s(s + 5)(s + 10)} \tag{46}$$

and compare the controller with $\alpha = \frac{1}{3}$ and $\beta = \frac{1}{6}$, i.e.

$$C_{\mathrm{FO}} = K_{\mathrm{FO}} \frac{(s^{\alpha} + 0.2)(s^{\beta} + 0.2)}{s^{\alpha}(s^{\beta} + 2)} \tag{47}$$

with the IO compensator

$$C_{\mathrm{IO}} = K_{\mathrm{IO}} \frac{(s + 0.1)}{s}. \tag{48}$$

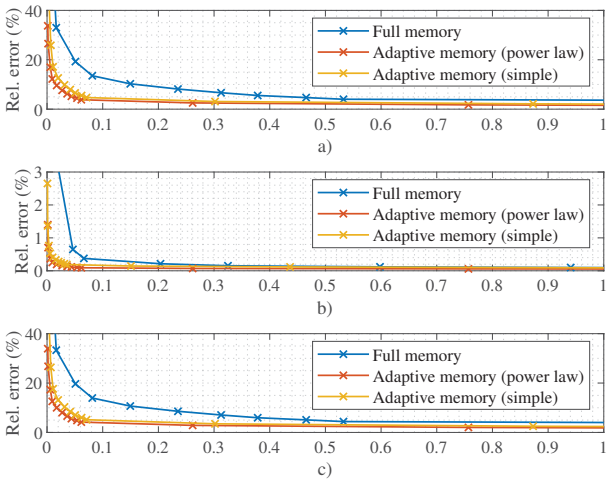


Fig. 8: Relative approximation error (%) versus computation time of different memory implementations: a) Total error; b) Error inside memory window; c) Error outside memory window.

The gains $K_{\mathrm{FO}} = 124.25$ and $K_{\mathrm{IO}} = 27.32$ are set to achieve a crossover frequency of $\omega_{s,\mathrm{IO}} = 0.3\,\mathrm{rad\,s^{-1}}$, $\omega_{s,\mathrm{FO}} = 0.7\,\mathrm{rad\,s^{-1}}$. The FO controller achieves a higher bandwidth while maintaining a similar phase margin of $\Phi_R \approx 60°$. With the introduced integrating action, these controllers are designed for tracking ramps.

Fig. 9 shows the tracking error achieved by each controller. Despite the higher bandwidth of the FO controller, the IO approach performs better. In the FO case the error only decays slowly which is caused by the algebraic decay introduced to the closed-loop system.

Hence, a design based on pole-placement is essential to enhance the convergence. On the one hand, we can increase the controller gains while maintaining stability, on the other hand we might move the dynamics back to the IO closed loop dynamics such that the signals converge exponentially.

Since the system is IO, the pole placement technique cannot be applied directly or would lead to an IO controller. So an additional FO pseudo pole is introduced with $\alpha = \frac{1}{2}$ to partially compensate the non-minimum phase zero of

$$\tilde{G}(s) = \frac{G(s)}{\tau_{\mathrm{add}} s^{\alpha} + 1}, \quad \tau_{\mathrm{add}} = 1\,\mathrm{s^{-1}}. \tag{49}$$

The resulting system shows an FO numerator such that the pole-placement leads to an FO controller as well. As the process already shows an IO integrator, we introduce a half-order integrator as a constraint to the pole-placement design. The IO poles are set to $\lambda_{\mathrm{IO}} = \{-5 \pm 0.1j, -10 \pm 1j, -10 \pm 2j\}$ resulting in the controller of order 7 (due to the FO integrator and the additional pole to force an FO control law). Furthermore the pre-filter cannot be designed to achieve exponential convergence as well as the ramp tracking capability. Therefore we set the filter to $F(s) = 1$. The additional pole needed to compute the IO controller

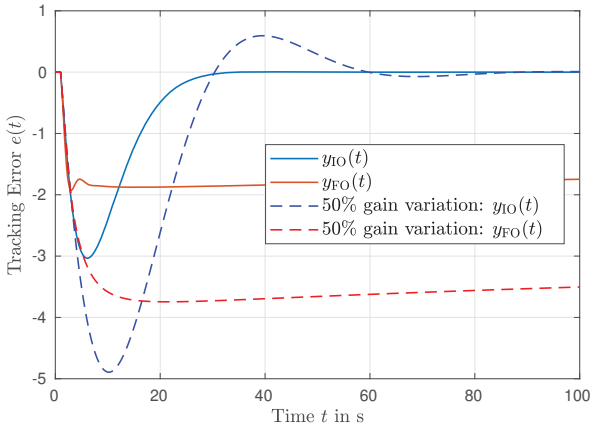


Fig. 9: Tracking error for open loop design.

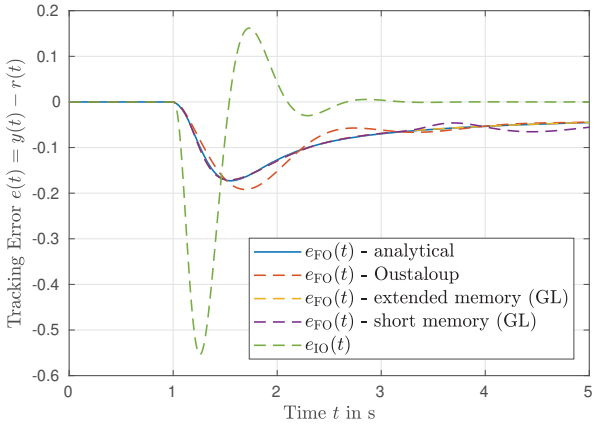


Fig. 10: Tracking error for pole-placement design.

with a second integrator is set to $\lambda* = -5$. The resulting controller of order three with the same IO closed-loop poles is applied to the process for comparison. Fig. 10 shows the tracking error of the closed loop system. Compared to the direct open-loop approach, the tracking error is reduced, but still converges slowly. The simulation uses the proposed discretization method, as well as a standard implementation using an Oustaloup filter of order 16 for $\omega \in (1 \times 10^{-2}\,\mathrm{rad\,s^{-1}}, 1 \times 10^{3}\,\mathrm{rad\,s^{-1}})$ with the fixed-step solver `ode3`. The discrete time implementation is much closer to the analytical solution within the full memory window $N_{\mathrm{short}} = 2000$. After this time period, the extended memory outperforms the short-memory GL implementation.

Compared to the IO the effect of the non-minimum phase zero is also reduced. To apply the method to IO systems, one has to balance between the reduced non-minimum phase behaviour and a desired exponential convergence.

## VI. Conclusions

An FO control strategy for IO or FO systems was investigated. It has been shown that it is advantageous to include an FO integrator to the open-loop by pole-placement in order to partially compensate non-minimum phase transfer zeros or to increase the order of integration further than one would need for ramp tracking and input disturbance rejection. With this

controller structure it is easier to guarantee a desired phase margin, due to the lower phase-loss of the FO integrator compared to an IO integrator. In addition, the iso-damping property is achieved for a wider frequency range, such that closed-loop is robust against gain variations.

The control structure was extended to 2-DOF with the addition of a pre-filter shaping the reference in such way that FO transfer zeros are cancelled, which leads to an exponential convergence of the input-output dynamics. This is especially suited to control FO systems and achieve certain convergence rates without increasing the gains too much

The effect of different memory truncations suitable for real-time implementation of such discrete operators with $\alpha \in [-1, 1]$ was investigated in the frequency domain. Combined memory extension shows similar phase response to the adaptive power law, while the last one being the most efficient implementation, but requires non-linear sampling of history. Approximation error due to discretization and memory truncation was also evaluated in terms of computation effort. Control and memory truncated strategies have been demonstrated by numeric simulations.

## ACKNOWLEDGMENT

This research was supported by IS-DAAD program under RCN project number 294835 (DAAD project-ID 57458791). The first author gratefully acknowledges support by Deutsche Forschungsgemeinschaft (DFG) in the framework of the Research Training Group "Tip- and laser-based 3D-Nanofabrication in extended macroscopic working areas" (GRK 2182) at Technische Universität Ilmenau, Germany. The last author acknowledges financial support from European Union Horizon 2020 research and innovation program, Marie Skłodowska-Curie grant agreement No. 734832.

## REFERENCES

[1] B. Saidi, S. Najar, M. Amairi, and M. Abdelkrim, "Design of a robust fractional pid controller for a second order plus dead time system," in *10th International Multi-Conference on Systems, Signals Devices*, 2013, pp. 1–6.

[2] C. Monje, Y. Chen, B. Vinagre, D. Xue, and V. Feliu-Batlle, *Fractional-order Systems and Controls: Fundamentals and Applications*. Springer, 2010.

[3] P. Lanusse, J. Sabatier, and A. Oustaloup, "Extension of pid to fractional orders controllers: a frequency-domain tutorial presentation," in *IFAC World Congress*, Cape Town, South Africa, 2014, pp. 7436–7442.

[4] H. Rasouli, A. Fatehi, and H. Zamanian, "Design and implementation of fractional order pole placement controller to control the magnetic flux in damavand tokamak," *Review of Scientific Instruments*, vol. 86, no. 3, p. 033503, 2015.

[5] K. Bingi, R. Ibrahim, M. N. Karsiti, S. M. Hassan, and V. R. Harindran, "A comparative study of 2dof PID and 2dof fractional order PID controllers on a class of unstable systems," *Archives of Control Sciences*, vol. 28, no. 4, 2018.

[6] S. Pandey, P. Dwivedi, and A. Junghare, "A novel 2-DOF fractional-order $PI^{\lambda}$ - $D^{\mu}$ controller with inherent anti-windup capability for a magnetic levitation system," *AEU - International Journal of Electronics and Communications*, vol. 79, pp. 158–171, May 2017.

[7] M. Jain, A. Rani, N. Pachauri, V. Singh, and A. P. Mittal, "Design of fractional order 2-DOF PI controller for real-time control of heat flow experiment," *Engineering Science and Technology, an International Journal*, vol. 22, no. 1, pp. 215–228, Feb. 2019.

[8] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*. Acad. Press, 1999.

[9] K. Diethelm and A. D. Freed, "The FracPECE subroutine for the numerical solution of differential equations of fractional order," *Forschung und wissenschaftliches Rechnen 1998*, pp. 57–71, 1999.

[10] D. Matignon, "Stability results for fractional differential equations with applications to control processing," in *Multiconference on Computational Engineering in Systems Applications*, 1996, pp. 963–968.

[11] N. Mahani, A. Sedigh, and F. Merrikh Bayat, "Performance evaluation of non-minimum phase linear control systems with fractional order partial pole-zero cancellation," in *Asian Control Conference*, Istanbul, Turkey, 2013, pp. 1–4.

[12] C. Weise, K. Wulff, and J. Reger, "Fractional-order memory reset control for integer-order LTI systems," in *Conference on Decision and Control*, Nice, France, 2019, pp. 5710–5715.

[13] ——, "Fractional-order observer for integer-order LTI systems," in *Asian Control Conference*, Gold Coast, Australia, 2017, pp. 2101–2106.

[14] J. P. How, "Dynamic Output-Feedback Control Architectures Matter [Focus on Education]," *IEEE Control Systems Magazine*, vol. 36, no. 6, pp. 88–117, Dec. 2016.

[15] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu-Batlle, *Fractional-order systems and controls: fundamentals and applications*. Springer, 2010.

[16] C. L. MacDonald, N. Bhattacharya, B. P. Sprouse, and G. A. Silva, "Efficient computation of the Grünwald–Letnikov fractional diffusion derivative using adaptive time step memory," *Jurnal of Computational Physics*, vol. 297, pp. 221–236, 2015.

[17] D. Clemente-López, J. M. Muñoz-Pacheco, O. G. Félix-Beltrán, and C. Volos, "Efficient computation of the Grünwald-Letnikov method for arm-based implementations of fractional-order chaotic systems," in *International Conference on Modern Circuits and Systems Technologies*, May 2019, pp. 1–4.

[18] A. Tepljakov, *Fractional-order modeling and control of dynamic systems*. Springer, 2017.