Numerical Analysis of the Discrete, Fractional Order PID Controller Using FOBD Approximation

Krzysztof Oprzędkiewicz
AGH University of Science and Technology, Faculty of Electrical Engineering, Automatic Control, Informatics and Biomedical Engineering, Department of Automatic Control and Robotics, al. A. Mickiewicza 30, 30-059 Kraków, Poland

Abstract: This paper proposes a methodology of the numerical testing of the discrete, approximated Fractional Order PID Controller(FOPID). The fractional parts of the controller are approximated using the Fractional Order Backward Difference (FOBD) operator. The goal of the analysis is to find the memory length optimum from point of view both accuracy and duration of computations. To do it new cost functions describing both accuracy and numerical complexity were proposed and applied. Results of tests indicate that the optimum memory length lies between 200 and 400. The proposed approach can be also useful to examine of another discrete implementations of a fractional order operator using FOBD.

Keywords: FOPID controller, FOBD approximation, Grünwald-Letnikov definition, accuracy, ISE, IAE, numerical complexity

1. Introduction

One of main areas of application fractional order calculus in automation is a FOPID control. Results presented by many Authors, e.g. [2, 4, 12, 13], show that FOPID controller is able to assure better control performance than its integer order PID analogue.

Each digital implementation of FOPID controller (PLC, microcontroller) requires to apply integer order, finite dimensional, discrete approximant. The most known are: FOBD and CFE (*Continuous Fraction Expansion*) approximations [1]. They allow to estimate a non-integer order element with the use of a digital filter. The detailed comparison of both methods was done e.g. in [8]. The use of these methods in the FOPID controller were also considered in the paper [10].

For elementary fractional-order integrator/differentiator an analytical form of the step response is known [4]. It can be applied as the reference in a cost function describing an accuracy of an approximation. Typically such a cost function describes an accuracy only and it does not inform about another properties of an approximant, for example its numerical complexity.

This paper deals with the numerical analysis of the accuracy and numerical complexity of the discrete implementation of the FOPID controller. This implementation uses the FOBD approximation to express of the fractional parts of the controller. Unfortunately, the good accuracy of this approximation requires to apply of long memory. Of course, this improves a numerical complexity of approximant and consequently increases a duration of calculations. This implies that an implementation of such an algorithm should be preceded by an analysis of both accuracy and numerical complexity. To do it various cost functions containing these both factors need to be used.

The paper is organized as follows. Preliminaries draw theoretical background to presenting of main results. Next the proposed cost functions are proposed and employed to testing of the considered, approximated FOPID algorithm. Finally results are discussed.

Autor korespondujący:
Krzysztof Oprzędkiewicz, kop@agh.edu.pl

Artykuł recenzowany
nadesłany 22.03.2024 r., przyjęty do druku 28.06.2024 r.

2. Preliminaries

2.1. Elementary ideas

Elementary ideas from fractional calculus can be found in many books, for example: [5, 6, 12, 14]. Here only some definitions necessary to explain of main results are recalled.

Firstly the fractional-order, integro-differential operator is given [5, 7, 14]:

Definition 1 (The elementary fractional order operator)
The fractional-order integro-differential operator is defined as follows:

$$ {}_aD_t^\alpha f(t) = \begin{cases} \dfrac{d^\alpha f(t)}{dt^\alpha} & \alpha > 0 \\ f(t) & \alpha = 0 \\ \int_a^t f(\tau)(d\tau)^\alpha & \alpha < 0 \end{cases} \tag{1} $$

where a and t denote time limits for operator calculation, $\alpha \in \mathbb{R}$ denotes the non-integer order of the operation.

Next remember an idea of Gamma Euler function [7]:

Definition 2 (The Gamma function)

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt. \tag{2}$$

Furthermore recall an idea of Mittag-Leffer functions. The two parameter Mittag-Leffer function is defined as follows:

Definition 3 (The two parameter Mittag-Leffer function)

$$E_{\alpha,\beta}(x) = \sum_{k=0}^{\infty} \frac{x^k}{\Gamma(k\alpha+\beta)}. \tag{3}$$

For $\beta = 1$ we obtain the one parameter Mittag-Leffer function:

Definition 4 (The one parameter Mittag-Leffer function)

$$E_{\alpha}(x) = \sum_{k=0}^{\infty} \frac{x^k}{\Gamma(k\alpha+1)}. \tag{4}$$

The fractional-order, integro-differential operator (1) can be described by many definitions. The "classic" have been proposed by Grünwald and Letnikov (GL Definition), Riemann and Liouville (RL Definition) and Caputo (C Definition). In this paper C and GL definition will be employed. They are recalled beneath [4, 11].

Definition 5 (The Caputo definition of the FO operator)

$$ {}_0^C D_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^{\infty} \frac{f^{(n)}(\tau)}{(t-\tau)^{\alpha+1-n}} d\tau \tag{5}$$

where $n - 1 < \alpha <$ denotes the non-integer order of operation and $\Gamma(..)$ is the complete Gamma function expressed by (2).

For the Caputo operator the Laplace transform can be defined [6]:

Definition 1. (The Laplace transform of the Caputo operator)

$$\begin{gathered} L\left({}_0^C D_t^{\alpha} f(t)\right) = s^{\alpha} F(s), \quad \alpha < 0 \\ L\left({}_0^C D_t^{\alpha} f(t)\right) = s^{\alpha} F(s) - \sum_{k=0}^{n-1} s^{\alpha-k-1} {}_0 D_t^k f(0), \\ \alpha > 0,\ n-1 < \alpha \le n \in N. \end{gathered} \tag{6}$$

Definition 2. (The Grünwald-Letnikov definition of the FO operator)

$$ {}_0^{GL} D_t^{\alpha} f(t) = \lim_{h \to 0} h^{-\alpha} \sum_{l=0}^{\left[\frac{t}{h}\right]} (-1)^l \binom{\alpha}{j} f(t - lh). \tag{7}$$

In (7) $\binom{\alpha}{l}$ is the binomial coefficient:

$$\binom{\alpha}{l} = \begin{cases} 1, & l = 0 \\ \dfrac{\alpha(\alpha-1)\ldots(\alpha-l+1)}{l!}, & l > 0 \end{cases}. \tag{8}$$

2.2. The FOBD operator

The GL definition is limit case for $h \to 0$ of the Fractional Order Backward Difference (FOBD), commonly employed in discrete FO calculations (see e.g. [12], p. 68):

Definition 3. (The Fractional Order Backward Difference-FOBD)

$$FOBD = (h^{\alpha} x)(t) = \frac{1}{h^{\alpha}} \sum_{l=0}^{L} (-1)^l \binom{\alpha}{l} x(t - lh). \tag{9}$$

Denote coefficients $(-1)^l \binom{\alpha}{l}$ by $d_l(\alpha)$:

$$d_l(\alpha) = (-1)^l \binom{\alpha}{l}. \tag{10}$$

The coefficients (10) are functions of order α. They can be also calculated with the use of the following, equivalent recursive formula (see e.g. [4], p. 12), useful in numerical calculations:

$$\begin{aligned} & d_0(\alpha) = 1, \\ & d_l(\alpha) = \left(1 - \frac{1+\alpha}{l}\right) d_{l-1}(\alpha), \quad l = 1, \ldots, L. \end{aligned} \tag{11}$$

It is proven in [3] that:

$$\sum_{l=1}^{\infty} d_l(\alpha) = 1 - \alpha. \tag{12}$$

From (11) and (12) we obtain at once that:

$$\sum_{l=2}^{\infty} d_l(\alpha) = 1. \tag{13}$$

In (9) L denotes a memory length necessary to correct approximation of a non-integer order operator. Unfortunately good accuracy of approximation requires to use a long memory L what can make difficulties during implementation.

The approximator FOBD (9) can be described by the $G(z^{-1})$ transfer function in the form of the FIR filter containing only zeros:

$$G_{FOBD}(z^{-1}, \alpha) = \frac{1}{h^{\alpha}} \sum_{l=0}^{L} d_l(\alpha) z^{-l}. \tag{14}$$

where $d_l(\alpha)$ are expressed by (10) or equivalently by (11), h is the sample time and α is the fractional order. The transfer function (14) is typically applied to approximate of the fractional operator (1).

2.3. The FOPID controller

The FOPID controller is described by the following transfer function (see e.g. [4], p. 33):

$$G_c(s) = k_P + k_I s^{-\alpha} + k_D s^{\beta}. \tag{15}$$

where $\alpha, \beta \in \mathbb{R}$ are fractional orders of the integration and derivative actions and k_P, k_I and k_D are the coefficients of the proportional, integral and derivative actions respectively.

The analytical formula of the step response of the controller (15) takes the following form [9]:

$$y_a(t) = k_P + \frac{k_I t^{\alpha}}{\Gamma(\alpha+1)} + \frac{k_D t^{-\beta}}{\Gamma(1-\beta)}. \quad (16)$$

where Γ(..) is the complete Gamma function (2). This analytical formula will be used as the reference to estimate of the accuracy of the approximation.

The discrete implementation of the controller (15) using approximator (14) is as beneath:

$$G_{cFOBD}(z^{-1}) = k_P + k_I G_{FOBD}(z^{-1}, -\alpha) + k_D G_{FOBD}(z^{-1}, \beta). \quad (17)$$

The step response of the approximated controller (17) takes the following form:

$$y_{FOBD}(k) = Z^{-1}\left\{\frac{1}{1-z^{-1}} G_{cFOBD}(z^{-1})\right\}. \quad (18)$$

The formula (18) can be computed numerically with the use of step function from MATLAB. It is the function of a time and memory length L. The memory length determines also the accuracy of the implementation. The accuracy as a function of L was analyzed e.g. in the paper [9]. Here both accuracy and numerical complexity will be examined. Cost functions used to do it are proposed in the next section.

3. The considered cost functions

At the beginning consider the accuracy of approximation. It will be tested using known IAE and ISE cost functions, calculated at the discrete time grid and for finite time interval:

$$IAE(L) = h\sum_{k=1}^{K_f} \left|e(k)\right|. \quad (19)$$

$$ISE(L) = h\sum_{k=1}^{K_f} e^2(k). \quad (20)$$

where $k = 1, ..., K_f$ are the discrete time instants, h is the sample time. Consequently the final time of computing is equal:

$$t_f = hK_f. \quad (21)$$

Error $e(k)$ describes the difference between analytical and approximated step responses (16) vs (18) in the same time moment k:

$$e(k) = y_a(kh) - y_{FOBD}(k), \quad k = 1, \ldots, K_f. \quad (22)$$

Next the numerical complexity should be tested. Its simpliest estimation is the experimentally measured duration of calculation of the step response (18). It is a function of memory length L and it can be measured during working of software MATLAB. Of course, such an estimation is strongly determined by a hardware-software platform, but it contains more general information too.

A platform-independent measure of the numerical complexity of the considered approximation is the memory length L. It will be employed in the new proposed cost functions, describing accuracy and complexity associated together. They take the following form:

$$D_{IAE}(L) = \omega_1 IAE(L) + \omega_2 L. \quad (23)$$

$$D_{ISE}(L) = \omega_1 ISE(L) + \omega_2 L. \quad (24)$$

where $IAE(L)$ and $ISE(L)$ are described by (19) and (20) respectively, $\omega_1 + \omega_2 = 1.0$ are the normalized weight coefficients.

4. Simulations

Simulations were executed at the MATLAB platform using the function *step* to compute of the step response and functions *tic*, *toc* to measure the duration of computations. For each tested value of L the mean value of 100 tests was examined. Tests were done for fixed parameters k_P, k_I and k_D of controller and various fractional orders α and β and memory length L. Values of all parameters applied during tests are collected in the table 1.

Table 1. Values of all parameters used during the tests
Tabela 1. Zestawy wartości parametrów regulatora zastosowane do testów

Parameters	1	2	3
α	–0.25	–0.50	–0.75
β	0.25	0.50	0.75
k_P	1	1	1
k_I	1	1	1
k_D	1	1	1
h [s]	0.1	0.1	0.1
t_f [s]	100.0	100.0	100.0
Range of L	100–1000	100–1000	100–1000
Incrementation of L	100	100	100
No of tests for single L	100	100	100
Final time t_f [s]	100	100	100
Weights ω_1, ω_2	0.5, 0.5	0.5, 0.5	0.5, 0.5

At the beginning the accuracy described by the cost functions (19) and (20) was examined. Comparing of step responses for Parameters 2 and L = 100, 500 and 1000 are shown in the Figure 1, the cost functions IAE and ISE as functions of memory length L are presented in the Figure 2.

In Figures 1 and 2 it can be seen that the accuracy of the approximation is determined by the memory length L as well as the fractional orders of the controller α and β. The accuracy is generally better for smaller orders and decreases for their values tending to one.

Furthermore the average duration of computing was tested. Results are illustrated by the Figure 3.

The Figure 3 allows to conclude that the duration of calculation does not depend on the orders α and β and it is approximately a linear function of the memory length L. This allows to use the memory length L as a direct measure of a numerical complexity.

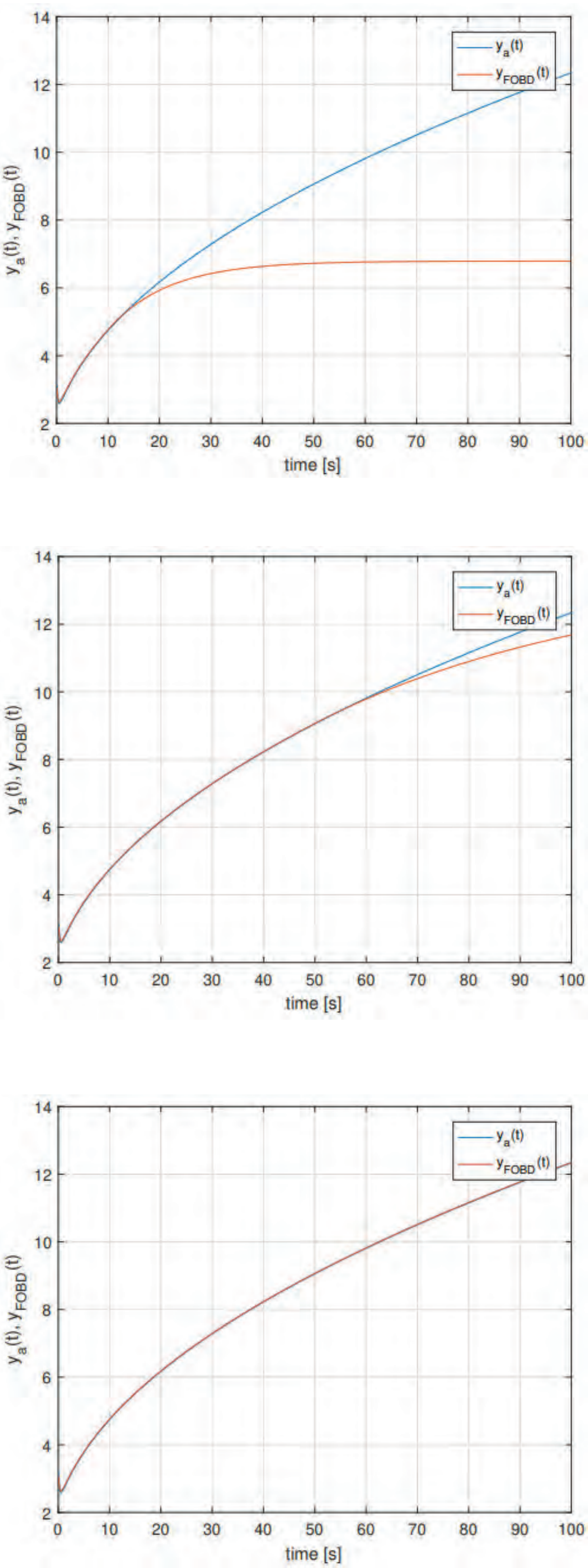

Fig. 1. The step responses $y_a(t)$ vs $y_{FOBD}(kh)$ for Parameters 2 and memory lengths: L = 100 – top, L = 500 – middle and L = 1000 bottom
Rys. 1. Porównanie odpowiedzi skokowej analitycznej i aproksymowanej dla zestawu parametrów nr 2 oraz długości pamięci: L = 100 – góra, L = 500 – środek i L = 1000 – dół

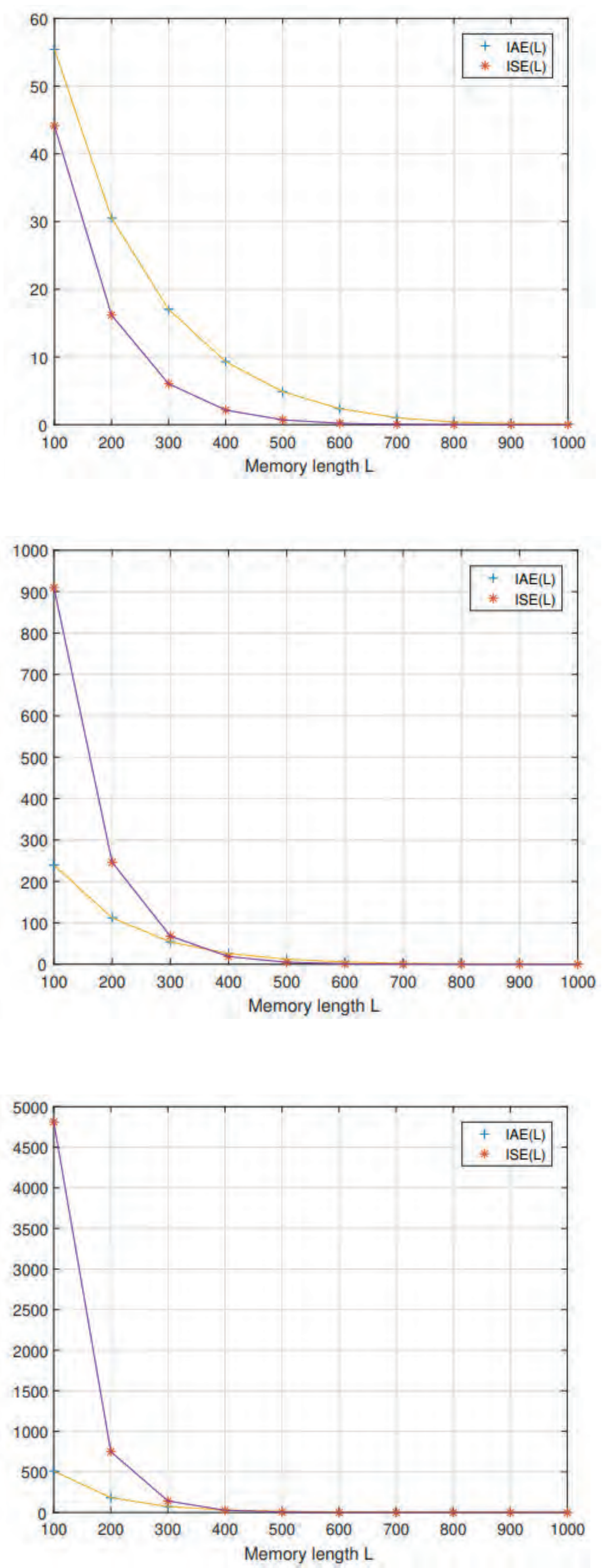

Fig. 2. The cost functions (19) and (20) for the approximated step response (18) and various fractional orders given in the table 1: Parameters 1 – top, Parameters 2 – middle, Parameters 3 – bottom
Rys. 2. Wskaźniki jakości (19) i (20) dla aproksymowanej odpowiedzi skokowej i różnych rzędów ułamkowych podanych w tabeli 1: zestaw parametrów 1 – góra, zestaw parametrów 2 – środek i zestaw parametrów 3 – dół

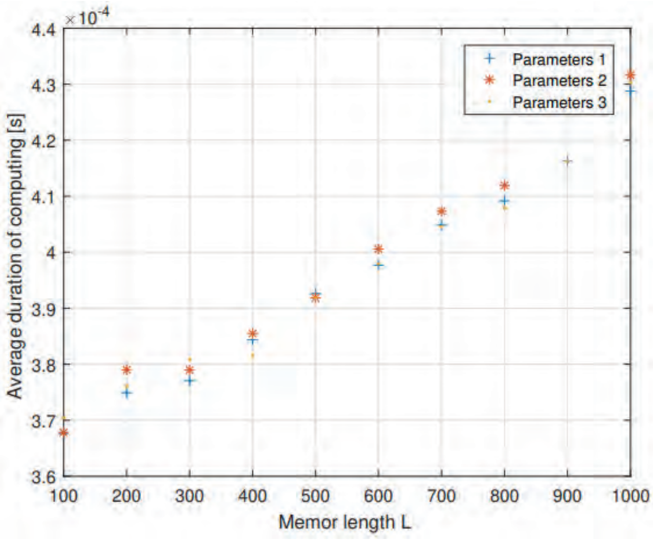

Fig. 3. Duration of calculations of the step response of the FOPID approximated by FOBD for various fractional orders

Rys. 3. Czasy obliczeń podczas wyznaczania aproksymowanej odpowiedzi skokowej dla różnych rzędów ułamkowych

Finally the complex cost functions (23) and (24) were examined. During tests the weights ω_1 and ω_2 were set equal 0.5 for each test. Such values well describe balancing between accuracy and numerical complexity.

The diagrams in the Figure 4 show that the best compromise between accuracy and numerical complexity is achieved for memory length L located in the range between 200 and 400.

5. Final Conclusions

The main result from this paper is that the minimum memory length $L = 100$ is able to assure the good accuracy only for lower fractional orders in the approximated FOPID controller. For higher orders, greater than 0.5 it is required to apply memory length not smaller than 200. The numerical tests shown in this paper can be expanded to find many other dependencies between accuracy, memory length, fractional orders α and β, coefficients $k_{P,I,D}$ of controller, sample time, value of final time t_f and so on.

Interesting can be also formulating of general analytical conditions associating an accuracy and a convergence of an approximation to a duration of computations at industrial device, e.g. at a PLC.

Acknowledgments

This paper was sponsored by AGH UST project no 16.16.120.773.

References

1. Al-Alaoui M.A., *Novel digital integrator and differentiator*. "Electronics Letters", Vol. 29, No. 4, 1993, 376–378, DOI: 10.1049/el:19930253.
2. Vinagre B.M., Podlubny I., Hernandez A., Feliu V., *Some approximations of fractional order operators used in control theory and applications*, "Fractional Calculus and Applied Analysis", Vol. 3, No. 3, 2000, 231–248.
3. Buslowicz M., Kaczorek T., *Simple conditions for practical stability of positive fractional discrete-time linear systems.*

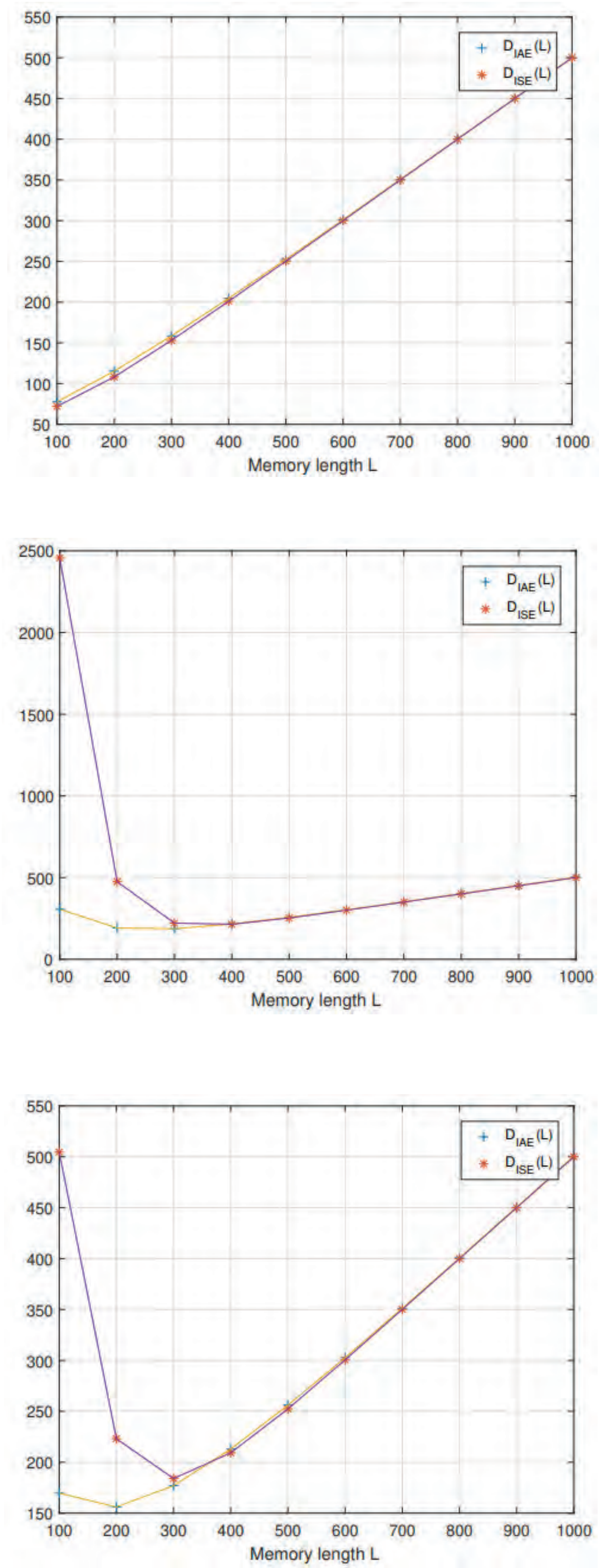

Fig. 4. The cost functions (23) and (24) for the approximated step response (18) and various fractional orders given in the table 1: Parameters 1 – top, Parameters 2 – middle, Parameters 3 – bottom

Rys. 4. Wskaźniki jakości (23) i (24) dla aproksymowanej odpowiedzi skokowej (18) oraz rzędów ułamkowych z tabeli 1: zestaw parametrów 1 – góra, zestaw parametrów 2 – środek i zestaw parametrów 3 – dół

"International Journal of Applied Mathematics and Computer Science", Vol. 19, No. 2, 2009, 263–269, DOI: 10.2478/v10006-009-0022-6.
4. Caponetto R., Dongola G., Fortuna L., Petráš I., *Fractional order systems: Modeling and Control Applications*. [In:] World Scientific Nonlinear Science Series A, Vol. 72, University of California, Berkeley, 2010.
5. Das S., *Functional Fractional Calculus for System Identification and Control*. Springer, Berlin, 2010, DOI: 10.1007/978-3-540-72703-3.
6. Kaczorek T., *Selected Problems of Fractional Systems Theory*. Springer, Berlin, 2011, DOI: 10.1007/978-3-642-20502-6.
7. Kaczorek T., Rogowski K., *Fractional Linear Systems and Electrical Circuits*. Springer, Bialystok University of Technology, Bialystok, 2015, DOI: 10.1007/978-3-319-11361-6.
8. Dorcák L., Petráš I., Terpák J., Zbrojovan M. *Comparison of the methods for discrete approximation of the fractional-order operator*. [In:] Proceedings of the ICCC'2003 Conference, Slovak Republic, 851–856.
9. Oprzędkiewicz K., *Accuracy estimation of digital fractional order PID controller*. [In:] Theory and applications of non-integer order systems : 8th conference on Non-integer order calculus and its applications, Zakopane, Poland, Lecture Notes in Electrical Engineering; Vol. 407, 2017, 265–275, DOI: 10.1007/978-3-319-45474-0_24.
10. Oprzędkiewicz K., Gawin E., *A non-integer order, state space model for one dimensional heat transfer process*. "Archives of Control Sciences", Vol. 26, No. 2, 2016, 261–275, DOI: 10.1515/acsc-2016-0015.
11. Ostalczyk P., *Equivalent descriptions of a discrete-time fractional-order linear system and its stability domains*. "International Journal of Applied Mathematics and Computer Science", Vol. 22, No. 3, 2012, 533–538, DOI: 10.2478/v10006-012-0040-7.
12. Ostalczyk P., *Discrete Fractional Calculus. Applications in Control and Image Processing*. World Scientific, New Jersey, London, Singapore, 2016, DOI: 10.1142/9833.
13. Petráš I., *Fractional order feedback control of a DC motor*. "Journal of Electrical Engineering", Vol. 60, No. 3, 2009, 117–128.
14. Podlubny I., *Fractional Differential Equations*. Academic Press, San Diego, 1999.

Analiza numeryczna dyskretnego regulatora PID niecałkowitego rzędu na bazie aproksymacji FOBD

Streszczenie: W artykule zaproponowano metodologię analizy numerycznej dyskretnego, aproksymowanego regulator PID niecałkowitego rzędu (regulator FOPID). Ułamkowe części regulatora są aproksymowane z wykorzystaniem aproksymacji FOBD (Fractional Order Backward Difference). Celem analizy jest znalezienie długości pamięci (wymiaru aproksymacji) optymalnej z punktu widzenia zarówno dokładności, jak i złożoności obliczeniowej. W tym celu zaproponowano i zastosowano nowe funkcje kosztu, opisujące oba te czynniki. Wynik testów wskazują, że optymalna długość pamięci w rozważanej sytuacji powinna leżeć w zakresie między 200 i 400. Proponowane podejście może też być wykorzystane do analizy innych dyskretnych implementacji operatora niecałkowitego rzędu, wykorzystujących operator FOBD.

Słowa kluczowe: regulator FOPID, aproksymacja FOBD, definicja Grunwalda-Letnikova, dokładność, ISE, ISA, złożoność numeryczna

Prof. Krzysztof Oprzędkiewicz, PhD DSc
kop@agh.edu.pl
ORCID: 0000-0002-8162-0011

He was born in Krakow in 1964. He obtained MSc in electronics in 1988, PhD and DSc in Automatics and Robotics in 1995 and 2009 at AGH University of Science and Technology (Krakow, Poland). He has been working at AGH University in Department of Automatics since 1988, recently as a professor. In 2012–2016 he was a deputy dean of faculty of Electrotechnics, Automatics, Informatics and Biomedical Engineering at AGH University. Recently he is the head of the Department of Automatic Control and Robotlics at AGH University. Since 2020 he is also a member of Committee on Automatic Control and Robotics of the Polish Academy of Sciences. His research covers infinite dimensional systems, fractional order modeling and control, uncertain parameter systems, industrial automation, PLC and SCADA systems.