



Article

An Approximation Method for Fractional-Order Models Using Quadratic Systems and Equilibrium Optimizer

Ali Yüce

Department of Electrical and Electronics Engineering, Faculty of Engineering and Natural Sciences, Malatya Turgut Ozal University, Malatya 44200, Turkey; ali.yuce@ozal.edu.tr

Abstract: System identification is an important methodology used in control theory and constitutes the first step of control design. It is known that many real systems can be better characterized by fractional-order models. However, it is often quite complex and difficult to apply classical control theory methods analytically for fractional-order models. For this reason, integer-order models are generally considered in classical control theory. In this study, an alternative approximation method is proposed for fractional-order models. The proposed method converts a fractional-order transfer function directly into an integer-order transfer function. The proposed method is based on curve fitting that uses a quadratic system model and Equilibrium Optimizer (EO) algorithm. The curve fitting is implemented based on the unit step response signal. The EO algorithm aims to determine the optimal coefficients of integer-order transfer functions by minimizing the error between general parametric quadratic model and objective data. The objective data are unit step response of fractional-order transfer functions and obtained by using the Grünwald-Letnikov (GL) method in the Fractional-Order Modeling and Control (FOMCON) toolbox. Thus, the coefficients of an integer-order transfer function most properly can be determined. Some examples are provided based on different fractional-order transfer functions to evaluate the performance of the proposed method. The proposed method is compared with studies from the literature in terms of time and frequency responses. It is seen that the proposed method exhibits better model approximation performance and provides a lower order model.



Citation: Yüce, A. An

Approximation Method for Fractional-Order Models Using Quadratic Systems and Equilibrium Optimizer. *Fractal Fract.* **2023**, *7*, 460. <https://doi.org/10.3390/fractalfract7060460>

Academic Editors: Thach Ngoc Dinh, Shyam Kamal, Rajesh Kumar Pandey, Bijnan Bandyopadhyay and Jun Huang

Received: 27 April 2023

Revised: 25 May 2023

Accepted: 31 May 2023

Published: 3 June 2023



Copyright: © 2023 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: fractional-order models; integer-order approximation methods; quadratic systems; Grünwald-Letnikov; equilibrium optimizer algorithm

1. Introduction

A physical system where the orders of the derivatives are any real number in its differential equation is called a fractional-order system. In recent years, there has been a strong interest in control systems whose mathematical models are expressed by fractional differential equations [1]. Such control systems are called fractional-order control systems. In practice, models of many systems can be described using fractional-order differential equations because systems that are defined by fractional-order describe the real-world more accurately [2]. Therefore, modeling a real-world system in fractional-order yields better results than classical integer-order modeling [3]. For example, viscoelastic materials, electromechanical processes, long transmission lines, dielectric polarizations, colored noise, cardiac behavior, bioengineering problems, and chaos are modelled by fractional-order differential equations [4–9]. However, many calculations in fractional-order control systems are quite complex and difficult due to the long memory effect [10]. For example, analytical time responses of fractional-order transfer functions are not possible, except in special cases, such as half derivatives [11]. Moreover, some classical control theory computations, such as State-Space form, Root-Locus and Routh-Hurwitz analysis, are not suitable for fractional-order systems. Classical control theory calculations are based on integer-order

systems. Therefore, studies of the integer-order approximation of a fractional process are very important, provided the approximation is accurate [12].

In the absence of analytical solutions, approximations or numerical methods are used for the above-mentioned computations. Many methods have been developed for the approximate time response computation. Some of the numerical based methods are the Grünwald-Letnikov (GL) method in the FOMCON toolbox [13], the Fourier Series Method (FSM), the Inverse Fourier Transform Method (IFTM), and Mittag-Leffler and Gamma functions [14–16]. Integer-order approximation-based methods are Continuous Fraction Expansion (CFE), Oustaloup's, Carlson's, Matsuda's, and Chareff's methods [16–23]. In addition to these methods, Stability Boundary Locus (SBL) fitting and its modified version, M-SBL fitting, and time response-based curve fitting, have been used in recent years [24–27]. Although there are many numerical methods, integer-order approximation methods are frequently preferred because they produce an integer-order transfer function. Thus, all control theory computation can be directly applied to the approximate integer-order transfer function. However, integer-order approximation methods mentioned above are operator-based methods (s^α , $\alpha \in R$), and it is necessary to use higher-order approximations to increase accuracy. Namely, the methods produce higher order approximate integer-order transfer functions for higher accuracy. Calculations with very high order transfer function structures can cause computational errors in control system analysis and are not preferred. The differences between the proposed method and the literature are delineated in the examples. In the literature, there are operator-based approximations that give rise to undesirable higher order approximate transfer functions, which complicate calculation. However, the proposed method results in maximum fifth order transfer functions due to direct approximation. Additionally, error margins are much smaller in time response analysis of the proposed method compared to the existing literature.

Casagrande et al. proposed an integer-order approximation using the interpolation technique in the Loewner framework [28]. An approximation procedure by means of integer-order state-space models is presented in [29]. R. Mansouri proposed a method, which deals with the fractional approximation of systems by integer reduced models [30]. Two other methods for simulating a fractional dynamical system in state space form are presented in [31]. Mohammed Saleh suggested a technique, which is related to the use of rational approaches in fractional-order systems and practical applications of controllers [32]. In [33], researchers proposed and offered an algorithm for a method of approximation useful in robust controller synthesis. Abdelaziz et al. contributed to the literature by presenting a stable approximation method for fractional-order systems based on the Gray Wolf Optimizer hybridized with the Cuckoo Search Algorithm (GWO-CS) in [34]. Apart from these studies, to overcome difficulties induced by operator-based approximations, some direct approximation methods have been proposed by the researchers. The studies are generally based on model reduction, optimization, and algorithms. Some of the most common studies include Particle Swarm Optimization (PSO) [35], the Genetic Algorithm (GA) [36], and Colliding Bodies Optimization (CBO) [37].

The main purpose of this study is to calculate the direct integer-order approximation of a fractional-order transfer function without using operator-based approximation methods. The contributions of this study to the literature are listed below.

- This study is the first to use a curve fitting method in the time domain using a quadratic standard model structure and EO optimizer. The proposed method allows us to achieve integer approximations with a lower order model and smaller margin of error. Thus, using the proposed method, classical control theory computations, such as State-Space form, Root-Locus and Routh-Hurwitz analysis, can be easily implemented.
- Using the proposed direct method, an integer-order approximate transfer function is achieved with a maximum of five orders. Additionally, lower-order approximation is obtained compared to operator-based methods.
- It has been shown that the proposed method is more accurate than operator-based methods in terms of frequency and time response performances.

- These are applicable to closed-loop fractional-order systems.

The remainder of the study is organized as follows. In Section 2, the EO optimizer is described. In Section 3, the curve fitting algorithm and determination of the approximate transfer function for proposed direct integer-order approximation method is explained. Some illustrative numerical examples using the proposed method are given in Section 4, as well as the calculation of direct approximations of fractional-order transfer functions and their comparison with the methods in the literature are presented. Results are given in Section 5.

2. Equilibrium Optimizer Algorithm

The Equilibrium Optimizer (EO) is a physics-based meta-heuristic optimization algorithm introduced to the literature in 2020 [38]. The idea of this algorithm is based on mimicking the dynamic mass balance on a control volume. EO optimization can be used to solve problems in many studies, such as feature selection, photovoltaic models, medical data and image classification, image segmentation, scheduling, sentiment analysis, and technical problems. In the study, Farmrazzi et al. explained the superiority of the EO algorithm over other optimization methods, such as the Genetic Algorithm (GA), Gravity Search Algorithm (GSA), Grey Wolf Optimizer (GWO), and Particle Swarm Optimization (PSO).

The EO algorithm tries to determine the equilibrium state of the volume on a mass-balance model. The mass-balance equation [39], which is described in the form of the first-order ordinary differential equation, is written as in Equation (1).

$$V \frac{dC}{dt} = QC_{eq} - QC + G \quad (1)$$

In Equation (1), the mass change rate in the control volume is defined as VdC/dt , and Q indicates the volume flow rate into or out of the control volume, C_{eq} denotes the concentration at equilibrium, C is the concentration in the control volume (V), and G denotes the mass generated in this system in the control volume. Here, Equation (2) is obtained when Equation (1) is rearranged by writing λ instead of Q/V .

$$\frac{dC}{\lambda C_{eq} - \lambda C + G/V} = dt \quad (2)$$

For the solution of the ordinary differential equation, Equation (3) is obtained by integrating both sides of the Equation (2).

$$\int_{C_0}^C \frac{dC}{\lambda C_{eq} - \lambda C + G/V} = \int_{t_0}^t dt \quad (3)$$

As a result, Equation (4) is obtained by solving the ordinary differential equation.

$$C = C_{eq} + (C_0 - C_{eq})F + \frac{G}{\lambda V}(1 - F) \quad (4)$$

where; C_0 denotes to the concentration values of the control volume at the initial start time t_0 . λ means flow rate. C represents the solution obtained at the current iteration, and C_{eq} indicates the best solution found so far. The structure given in Equation (4) forms the basis of the EO algorithm, and the EO algorithm performs iterative optimization search, as in the PSO algorithm.

F is the exponential term, and it is computed, as given in Equation (8).

The early step of EO is the random initiation of concentration, as expressed in Equation (5):

$$C_i^d = C_{\min} + rand_i^d(C_{\max} - C_{\min}) \quad i = 1, 2, 3, \dots, N \text{ and } d = 1, 2, 3, \dots, D \quad (5)$$

where; N is the number of particles in the population, D is the number of dimensions, C_i^d is the initial concentration vector for each particle, C_{max} and C_{min} represent the upper and lower bound of the optimization variables, respectively, and $rand_d^i$ is a random vector between $[0, 1]$.

$\vec{C}_{eq, pool}$ is called equilibrium pool. It consists of five elements and is shown in Equation (6). The value $\vec{C}_{eq(ave)}$ is the average of four best-so-far candidates.

$$\vec{C}_{eq, pool} = \left\{ \vec{C}_{eq(1)}, \vec{C}_{eq(2)}, \vec{C}_{eq(3)}, \vec{C}_{eq(4)}, \vec{C}_{eq(ave)} \right\} \tag{6}$$

$$\vec{C}_{eq(ave)} = \frac{\vec{C}_{eq(1)} + \vec{C}_{eq(2)} + \vec{C}_{eq(3)} + \vec{C}_{eq(4)}}{4} \tag{7}$$

When the optimization is finished, all particles are updated with the same number of updates relative to all candidate solutions. The main concentration values are updated using Equation (8):

$$\vec{F} = \exp \left[-\vec{\lambda} (t - t_0) \right] \tag{8}$$

where; λ in Equation (8) denotes a random vector between $[0, 1]$, t denotes the time and decreases, depending on the number of iterations [40]. t and t_0 are computed by Equations (10) and (11), respectively.

$$t = \left(1 - \frac{Iter}{Max_Iter} \right)^{\left(\alpha \frac{Iter}{Max_Iter} \right)} \tag{9}$$

$$t_0 = \frac{1}{\vec{\lambda}} \ln \left(-\beta sign \left(\vec{r} - 0.5 \right) \left[1 - e^{-\vec{\lambda} t} \right] \right) + t \tag{10}$$

Equation (11) is determined by substituting t_0 in Equation (8), and it is the equation that enables the algorithm to reach the optimal value.

$$\vec{F} = \beta sign \left(\vec{r} - 0.5 \right) \left[e^{-\vec{\lambda} t} - 1 \right] \tag{11}$$

The generation rate, \vec{G} , is one of the important roles in EO, and it improves the algorithm’s ability to work or to use. The mathematical model of \vec{G} is given in Equation (12).

$$\vec{G} = \vec{G}_0 e^{-\vec{k} (t-t_0)} \tag{12}$$

where; k represents the attenuation constant, and \vec{G}_0 is the initial value. For restricting the random variables, suppose that $k = \lambda$, and utilize the formerly derived exponential expression. Generation rate \vec{G} is calculated by using Equation (13).

$$\vec{G} = \vec{G}_0 e^{-\vec{\lambda} (t-t_0)} = \vec{G}_0 \vec{F} \tag{13}$$

where; \vec{G}_0 is computed by Equation (14).

$$\begin{aligned} \vec{G}_0 &= GCP \left(\vec{C}_{eq} - \vec{\lambda} \vec{C} \right) \\ GCP &= \begin{cases} 0.5r_1 & r_2 \geq GP \\ 0 & r_2 < GP \end{cases} \end{aligned} \tag{14}$$

The *GCP* is the generation rate control parameter that contains the potential of the generation expression assistance for the update procedure. r_1 and r_2 are random numbers in $[0, 1]$. *GP* is a specified value, which is called the generation probability. The updating of the individual solution of the EO is given in Equation (15). The flowchart of the EO is presented in Figure 1.

$$\vec{C} = \vec{C}_{eq} + \left(\vec{C} - \vec{C}_{eq} \right) \vec{F} + \frac{\vec{G}}{\lambda V} \left(1 - \vec{F} \right) \tag{15}$$

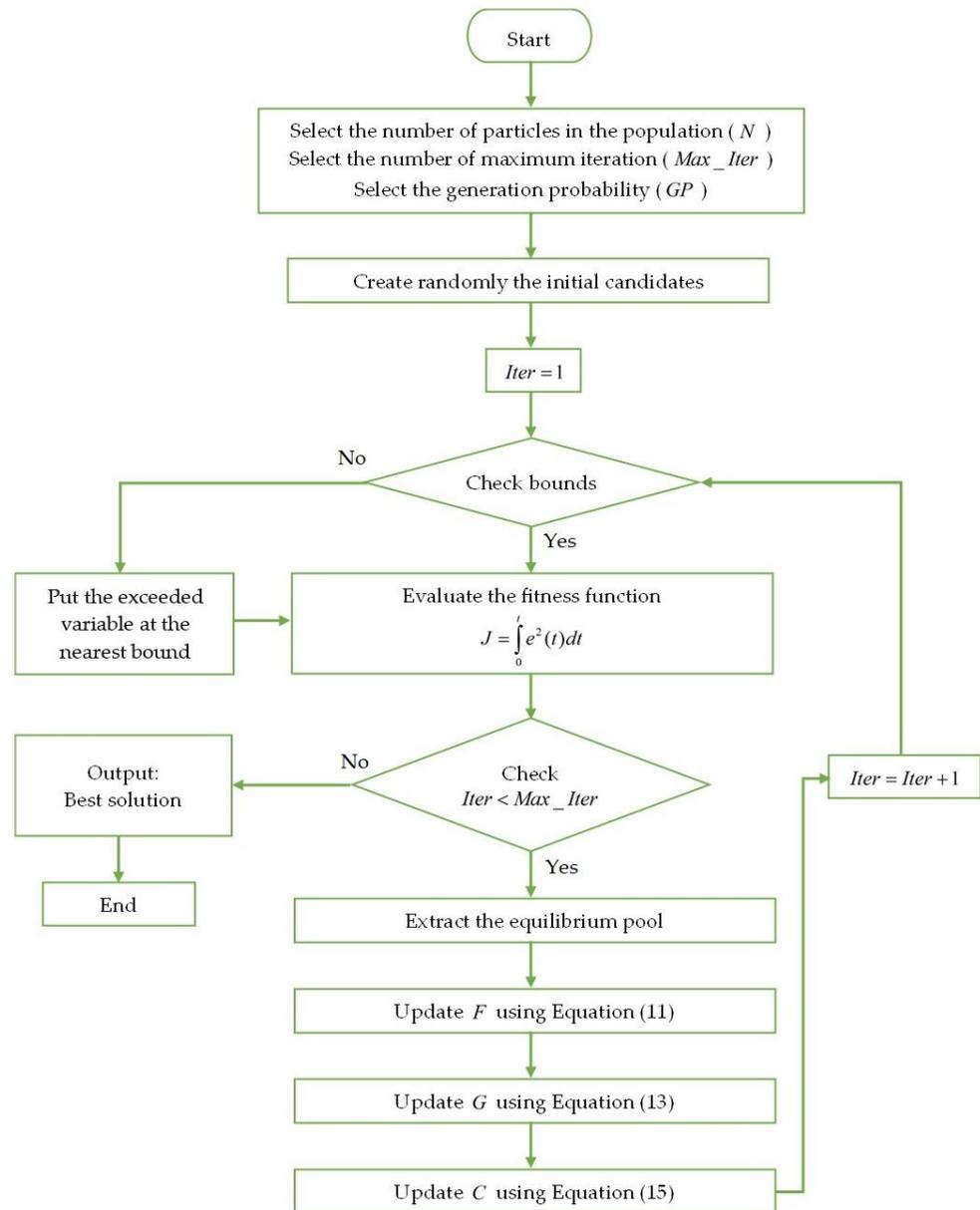


Figure 1. The flowchart of the Equilibrium Optimizer.

3. The Methodology of the Proposed Direct Integer-Order Approximation Method

In this section, the block diagram of the proposed direct approximation method and the integral performance criteria that we frequently encounter in optimization calculations will be introduced.

3.1. Block Scheme for Curve Fitting Computation with the EO

The block diagram of the proposed direct approximation method and the integral performance criteria that we frequently encounter in optimization calculations are as follows.

In the diagram, a unit step signal, $u(t)$, is applied to the inputs of the fractional-order transfer function and proposed model transfer function. The unit step time response data of the fractional transfer function defined in the form of Equation (16) are calculated with the GL method in the time step interval $\Delta t = 0.01$. Thus, the output $y_f(t)$ is obtained.

$$G_f(s) = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \dots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \dots + a_0 s^{\alpha_0}} \tag{16}$$

$$\begin{aligned} \{\alpha_i, \beta_j\} &\in R_+ \\ \{a_i, b_j\} &\in R \\ \alpha_0 < \alpha_1 < \dots < \alpha_n, \beta_0 < \beta_1 < \dots < \beta_m, i = 0, 1, 2, \dots, n \text{ and } j = 0, 1, 2, \dots, m \end{aligned}$$

The standard quadratic systems are common in practice and are the simplest type of dynamical systems to exhibit oscillations. Mass spring damper systems and RLC circuits can be given as examples for these type systems [41]. In fact, many higher-order systems may be approximated as second-order systems [42].

The canonical form of the second-order differential equation is defined, as in Equation (17).

$$\ddot{y} + 2\zeta\omega_n \dot{y} + \omega_n^2 y = \omega_n^2 u \tag{17}$$

The canonical second-order transfer function has the following form in Equation (18), in which it has two poles and no zeros.

$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{18}$$

In Equation (17), u denotes the input, y denotes the output, ζ is the damping ratio, and ω_n is the natural frequency. The unit step response of a standard quadratic transfer function is written in the time domain, as in Equation (19).

$$y(t) = 1 - e^{(-\zeta\omega_n t)} (\cos(\omega_n \sqrt{(1 - \zeta^2)} t) + (\zeta / \sqrt{(1 - \zeta^2)}) \sin(\omega_n \sqrt{(1 - \zeta^2)} t)) \tag{19}$$

Equation (19) can be written in generalized form as Equation (20), and it is proposed as a model transfer function for the curve-fitting computation. Thus, the error function is written as $e(t) = [y_f(t) - y_m(t)]$, and it is used as a fitness function in the EO optimizer.

$$y_m(t) = 1 - e^{-q_1 t} \cos(q_2 t) - q_3 e^{-q_4 t} \sin(q_5 t) \tag{20}$$

The Laplace transform of Equation (20) is given in Equation (21), where q_1, q_2, q_3, q_4 , and q_5 are the unknown coefficients. Equation (21) is an integer-order approximate transfer function to be obtained after using the EO curve fitting.

$$Y_m(s) = \frac{1}{s} - \frac{s + q_1}{(s + q_1)^2 + q_2^2} - \frac{q_3 q_5}{(s + q_4)^2 + q_5^2} \tag{21}$$

The structure of the transfer function with integer-order given in Equation (23) can be obtained from Equation (21). Depending on the structure of the fractional-order transfer function, the order of $G_m(s)$ can be 5 or less at the maximum.

$$G_m(s) = s Y_m(s) \tag{22}$$

$$G_m(s) = \frac{n_4 s^4 + n_3 s^3 + n_2 s^2 + n_1 s + n_0}{s^5 + m_4 s^4 + m_3 s^3 + m_2 s^2 + m_1 s + m_0} \tag{23}$$

$$\{m_i, n_i\} \in R, i = 0, 1, 2, \dots, n$$

3.2. Determination of the Proposed Model-Transfer-Function Parameters

The coefficients of optimal model transfer function can be obtained by minimizing the error signal by using integral performance criteria. Integral performance criteria are defined as cost functions in optimization algorithms and provide minimization. First, the integral of the square of the error (ISE) and the integral of the absolute value of the error (IAE) performance criteria were used by Graham and Lathrop [43] in 1953. Then, the absolute value of the time-weighted error (ITAE) and the integral of the square of the time-weighted error (ITSE) were developed by [44]. Some performance criteria and their mathematical expressions in the literature are given in Equations (25)–(30) [45].

- The Integral of Error (IE) is defined as:

$$IE = \int_0^t e(t) dt \quad (24)$$

- The Mean Square Error (MSE) is defined as:

$$MSE = \frac{\int_0^t e^2(t) dt}{t} \quad (25)$$

- The Integral Squared Error (ISE) is defined as:

$$ISE = \int_0^t e^2(t) dt \quad (26)$$

- The Integral Absolute Error (IAE) is defined as:

$$IAE = \int_0^t |e(t)| dt \quad (27)$$

- The Integral of the Square of the Time-Weighted Error (ITSE) is defined as:

$$ITSE = \int_0^t te^2(t) dt \quad (28)$$

- The Integral of the Absolute of the Time-Weighted Error (ITAE) is defined as:

$$ITAE = \int_0^t t|e(t)| dt \quad (29)$$

In the diagram in Figure 2, the cost function is defined as the square of the difference between the unit step responses of fractional-order and the model transfer function. This definition of the cost function (30) is the ISE performance criterion.

$$J = \int_0^t e^2(t) dt \quad (30)$$

In the optimization, various initial values, such as the lower and upper bounds of the model transfer function parameters, the maximum number of iterations, and the number of particles, are entered. Then, the algorithm is run for the cost function adapted to the EO algorithm. In this paper, the particle number is selected as 30, and the maximum iteration number is selected as 200. The lower and upper bounds of the transfer function coefficients are considered in the range of $[-5, 5]$. The error signal is obtained by subtracting the parametric unit step response of the model transfer function from the unit step response

of the fractional-order transfer function. Then, the error is updated according to the new values, and the cost function gradually decreases. Thus, the cycle continues until the desired criterion is met. Optimal model transfer function parameters (q_1, q_2, q_3, q_4, q_5) are obtained when the desired criterion is met. Thus, the proposed direct integer-order approximation of the fractional-order transfer function is calculated, as in the form of Equation (23).

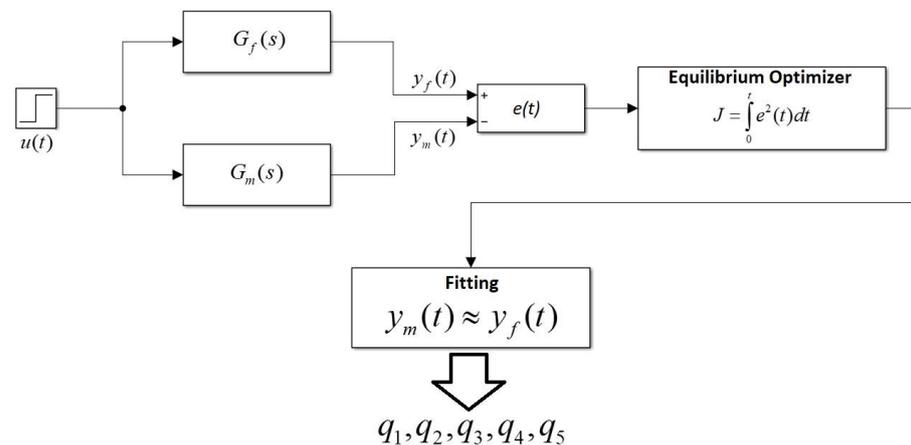


Figure 2. Block diagram of the proposed direct integer-order approximation method.

4. Numerical Examples

In this section, numerical examples are given, and the advantages of the proposed method over classical integer-order approximations are presented, as well as the use of the proposed method in closed-loop fractional-order control systems, and the comparison of two articles in the literature on direct integer-order approximation is given.

4.1. Examining the Proposed Method in Various Transfer Function Forms

Example 1: Let us obtain the integer-order approximation of the fractional-order transfer function given in (31).

$$G(s) = \frac{0.6s^{1.24} + 0.7}{0.5s^{4.16} + 3.2s^{3.28} + 4.4s^{2.12} + 5.5s^{1.57} + 1.2s^{0.94} + 1} \quad (31)$$

The fractional-order transfer function given in Equation (31) includes the terms of fractional derivative operators $s^{0.12}, s^{0.16}, s^{0.24}, s^{0.28}, s^{0.57}$, and $s^{0.94}$. Approximations of each fractional derivative term can be easily calculated by using Continuous Fraction Expansion (CFE), Oustaloup's method, and Matsuda's method, which are very common in the literature, as well as M-SBL methods, which were developed using the SBL technique. Finally, by substituting these approximations in Equation (31), the integer-order approximation of the $G(s)$ is obtained. Since Oustaloup can produce odd-order approximations, comparisons are made on odd-order approximations. The integer-order approximations of $G(s)$ calculated using the 3rd order approximations of the CFE, Oustaloup, Matsuda, and M-SBL methods are given in Equations (32)–(35), respectively. It can be seen from Equations (32)–(35) that $G(s)$ is obtained from 22 degrees for the 3rd order operator-based approximation. If we use fifth order approximations for each fractional derivative term, the order of the approximate transfer function will be 34. As observed, higher order approximations of the fractional derivative terms result in much higher orders in the total transfer function.

$$G_{CFE}(s) = \frac{1.463s^{19} + 387.1s^{18} + 2.298 \times 10^4 s^{17} + 6.228 \times 10^5 s^{16} + 9.441 \times 10^6 s^{15} + 8.774 \times 10^7 s^{14} + 5.264 \times 10^8 s^{13} + 2.116 \times 10^9 s^{12} + 5.881 \times 10^9 s^{12} + 1.158 \times 10^{10} s^{10} + 1.642 \times 10^{10} s^9 + 1.691 \times 10^{10} s^8 + 1.261 \times 10^{10} s^7 + 6.753 \times 10^9 s^6 + 2.551 \times 10^9 s^5 + 6.653 \times 10^8 s^4 + 1.161 \times 10^8 s^3 + 1.287 \times 10^7 s^2 + 8.161 \times 10^5 s + 2.249 \times 10^4}{0.9019s^{22} + 249.3s^{21} + 1.695 \times 10^4 s^{20} + 5.442 \times 10^5 s^{19} + 1.001 \times 10^7 s^{18} + 1.154 \times 10^8 s^{17} + 8.814 \times 10^8 s^{16} + 4.64 \times 10^9 s^{15} + 1.747 \times 10^{10} s^{14} + 4.853 \times 10^{10} s^{13} + 1.016 \times 10^{11} s^{12} + 1.619 \times 10^{11} s^{11} + 1.958 \times 10^{11} s^{10} + 1.788 \times 10^{11} s^9 + 1.223 \times 10^{11} s^8 + 6.245 \times 10^{10} s^7 + 2.379 \times 10^{10} s^6 + 6.744 \times 10^9 s^5 + 1.405 \times 10^9 s^4 + 2.085 \times 10^8 s^3 + 2.071 \times 10^7 s^2 + 1.225 \times 10^6 s + 3.235 \times 10^4} \quad (32)$$

$$G_{Ous}(s) = \frac{1.812s^{19} + 465s^{18} + 4.719 \times 10^4 s^{17} + 2.493 \times 10^6 s^{16} + 7.534 \times 10^7 s^{15} + 1.341 \times 10^9 s^{14} + 1.403 \times 10^{10} s^{13} + 8.601 \times 10^{10} s^{12} + 3.208 \times 10^{11} s^{11} + 7.619 \times 10^{11} s^{10} + 1.193 \times 10^{12} s^9 + 1.251 \times 10^{12} s^8 + 8.708 \times 10^{11} s^7 + 3.856 \times 10^{11} s^6 + 1.02 \times 10^{11} s^5 + 1.598 \times 10^{10} s^4 + 1.492 \times 10^9 s^3 + 8.13 \times 10^7 s^2 + 2.388 \times 10^6 s + 2.918 \times 10^4}{1.045s^{22} + 285.3s^{21} + 3.148 \times 10^4 s^{20} + 1.849 \times 10^6 s^{19} + 6.385 \times 10^7 s^{18} + 1.346 \times 10^9 s^{17} + 1.75 \times 10^{10} s^{16} + 1.416 \times 10^{11} s^{15} + 7.353 \times 10^{11} s^{14} + 2.566 \times 10^{12} s^{13} + 6.305 \times 10^{12} s^{12} + 1.118 \times 10^{13} s^{11} + 1.429 \times 10^{13} s^{10} + 1.288 \times 10^{13} s^9 + 8.016 \times 10^{12} s^8 + 3.429 \times 10^{12} s^7 + 1.022 \times 10^{12} s^6 + 2.08 \times 10^{11} s^5 + 2.794 \times 10^{10} s^4 + 2.389 \times 10^9 s^3 + 1.238 \times 10^8 s^2 + 3.529 \times 10^6 s + 4.235 \times 10^4} \quad (33)$$

$$G_{Mat}(s) = \frac{2.341s^{19} + 3780s^{18} + 1.045 \times 10^6 s^{17} + 1.173 \times 10^8 s^{16} + 6.564 \times 10^9 s^{15} + 1.935 \times 10^{11} s^{14} + 2.967 \times 10^{12} s^{13} + 2.314 \times 10^{13} s^{12} + 9.979 \times 10^{13} s^{11} + 2.58 \times 10^{14} s^{10} + 4.229 \times 10^{14} s^9 + 4.509 \times 10^{14} s^8 + 3.096 \times 10^{14} s^7 + 1.289 \times 10^{14} s^6 + 2.919 \times 10^{13} s^5 + 3.654 \times 10^{12} s^4 + 2.608 \times 10^{11} s^3 + 1.058 \times 10^{10} s^2 + 2.271 \times 10^8 s + 2.004 \times 10^6}{1.234s^{22} + 2024s^{21} + 6.002 \times 10^5 s^{20} + 7.458 \times 10^7 s^{19} + 4.751 \times 10^9 s^{18} + 1.654 \times 10^{11} s^{17} + 3.168 \times 10^{12} s^{16} + 3.332 \times 10^{13} s^{15} + 2.05 \times 10^{14} s^{14} + 7.963 \times 10^{14} s^{13} + 2.096 \times 10^{15} s^{12} + 3.892 \times 10^{15} s^{11} + 5.107 \times 10^{15} s^{10} + 4.614 \times 10^{15} s^9 + 2.787 \times 10^{15} s^8 + 1.119 \times 10^{15} s^7 + 3.062 \times 10^{14} s^6 + 5.431 \times 10^{13} s^5 + 5.986 \times 10^{12} s^4 + 3.999 \times 10^{11} s^3 + 1.565 \times 10^{10} s^2 + 3.29 \times 10^8 s + 2.866 \times 10^6} \quad (34)$$

$$G_{M-SBL}(s) = \frac{2.197s^{19} + 3729s^{18} + 1.318 \times 10^6 s^{17} + 1.957 \times 10^8 s^{16} + 1.454 \times 10^{10} s^{15} + 5.583 \times 10^{11} s^{14} + 1.054 \times 10^{13} s^{13} + 8.936 \times 10^{13} s^{12} + 3.939 \times 10^{14} s^{11} + 1.013 \times 10^{15} s^{10} + 1.623 \times 10^{15} s^9 + 1.664 \times 10^{15} s^8 + 1.075 \times 10^{15} s^7 + 4.067 \times 10^{14} s^6 + 7.86 \times 10^{13} s^5 + 7.903 \times 10^{12} s^4 + 4.348 \times 10^{11} s^3 + 1.323 \times 10^{10} s^2 + 2.094 \times 10^8 s + 1.345 \times 10^6}{1.182s^{22} + 2040s^{21} + 7.659 \times 10^5 s^{20} + 1.239 \times 10^8 s^{19} + 1.029 \times 10^{10} s^{18} + 4.577 \times 10^{11} s^{17} + 1.064 \times 10^{13} s^{16} + 1.231 \times 10^{14} s^{15} + 7.873 \times 10^{14} s^{14} + 3.084 \times 10^{15} s^{13} + 8.085 \times 10^{15} s^{12} + 1.485 \times 10^{16} s^{11} + 1.912 \times 10^{16} s^{10} + 1.678 \times 10^{16} s^9 + 9.733 \times 10^{15} s^8 + 3.683 \times 10^{15} s^7 + 9.189 \times 10^{14} s^6 + 1.406 \times 10^{14} s^5 + 1.262 \times 10^{13} s^4 + 6.57 \times 10^{11} s^3 + 1.943 \times 10^{10} s^2 + 3.024 \times 10^8 s + 1.923 \times 10^6} \quad (35)$$

The integer-order approximation of $G(s)$, calculated by the proposed direct approximation method, is obtained as a 4th order approximation, as seen in Equation (36).

$$G_m(s) = \frac{0.0027s^3 + 0.1923s^2 + 0.1691s + 0.0481}{s^4 + 2.05s^3 + 1.391s^2 + 0.3591s + 0.0687} \quad (36)$$

The GL method gives near-accurate results at low step intervals in time response calculations [46]. In this example, the comparisons are based on the GL method, and the step time interval for GL computations is chosen as 0.01. In operator-based approximation methods, the frequency range [0.01, 100] rad/s is preferred. The unit step response of the transfer functions in Equations (32)–(36) and unit step response of $G(s)$ calculated using the GL method are given in Figure 3.

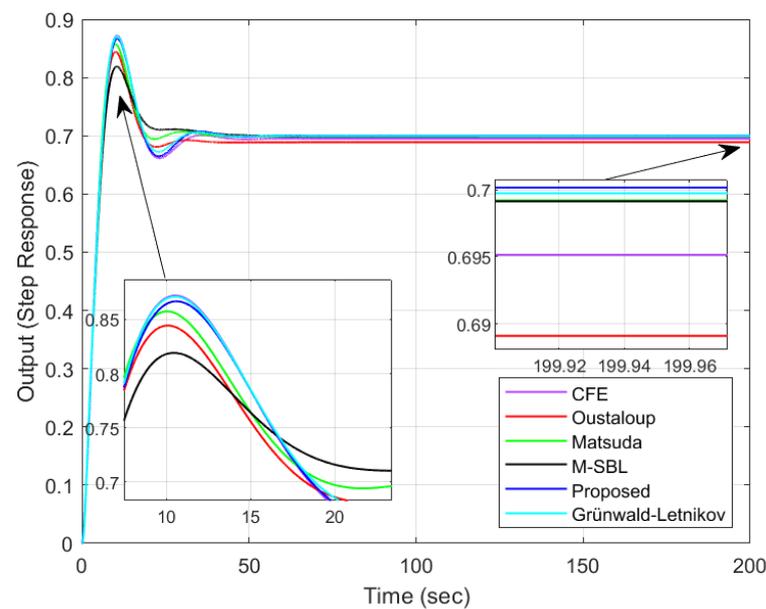


Figure 3. Unit step responses of Equations (31)–(36).

It can be seen, from Figure 3, that, although the transfer function calculated by the proposed direct approximation method is of the 4th order, it gives quite close results to the time response calculated by the GL method. To compare the operator-based approximation and the proposed method, approximations of different orders were obtained, and their errors were calculated. The mean squared errors (MSE) of these approximations, which derived from different orders, are presented in Table 1. Since operator-based approximations are calculated from odd orders in the Oustaloup method, approximations of the first-, third-, and fifth-orders are considered for the Oustaloup method. For the other methods, first-, second-, and third-order operator approximation are considered. The transfer function, $G_m(s)$, calculated with the proposed method, has lower error than the 28th order $G(s)$ obtained by CFE, Matsuda, and M-SBL methods. However, Table 1 shows that these methods have better results than the proposed direct approximation method for 34th-order approximation, except for Oustaloup.

Table 1. Errors based on time response for different methods and approximations of different orders.

Method	Order of Approximation	MSE (10^{-5})
CFE	22	2.5229
	28	0.7592
	34	0.2811
Oustaloup	22	13.9400
	34	11.5180
	48	11.6350
Matsuda	22	3.5524
	28	0.6710
	34	0.0852
M-SBL	22	13.5050
	28	0.4819
	34	0.2716
Proposed	4	0.3864

Table 2 shows the time response performance specifications, such as rise time, settling time, peak time, peak value, overshoot, and steady state error of Equations (31)–(36). The unit step response of Equation (31) is directly plotted numerically by the GL method. Unit step

responses are plotted using the approximate transfer functions given in Equations (32)–(36). Here, the approximation methods are compared with the GL method. The first thing to notice is that the steady-state error of the proposed direct approximation method is zero. It is observed that the method with the farthest performance specifications is Oustaloup. CFE, one of the operator-based methods, gives the closest result to the performance of the GL method. The transfer function calculated from the 4th degree with the proposed direct approximation method shows acceptable performance compared to the operator-based methods calculated from the 22nd order.

Table 2. Performance specifications for unit step responses of Equations (31)–(36).

Methods	Rise Time (s)	Settling Time (5%) (s)	Peak Time (s)	Peak Value	Overshoot (%)	Steady State Error
GL Method	4.4106	16.9845	10.5200	0.8711	24.4432	0
$G_{Proposed}(s)$	4.4276	23.3043	10.6000	0.8667	23.8176	0
$G_{CFE}(s)$	4.4009	25.0050	10.5300	0.8720	24.5749	0.0048
$G_{Ous}(s)$	4.3822	15.9932	10.0800	0.8444	20.6241	0.0110
$G_{Mat}(s)$	4.3098	16.4427	10.0400	0.8575	22.4929	0.0008
$G_{M-SBL}(s)$	4.5750	17.2698	10.4400	0.8191	17.0191	0.0006

Example 2: Let us consider an open loop unstable and stable fractional model given in (37) and (39), respectively.

$$G_f(s) = \frac{1}{0.75s^{2.42} + 1.47s^{0.8} + 3.2} \tag{37}$$

Equation (37) shows an open loop unstable model structure with two fractional-orders. When approximations are calculated with operator-based methods for this unstable model, an 8th-order transfer function is obtained. When the proposed method is used, a 5th-order approximation, such as Equation (38), is calculated. The unit step response performance of the operator-based approximations and the proposed method are given in Figure 4.

$$G_m(s) = \frac{0.2980s^3 + 5.5703s^2 + 40.1592s + 182.3607}{s^5 + 10.6963s^4 + 60.4775s^3 + 167.3740s^2 + 251.5915s + 583.2891} \tag{38}$$

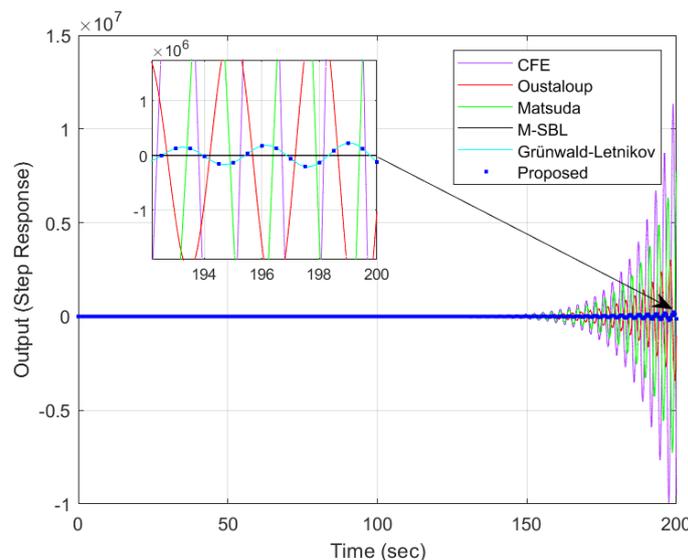


Figure 4. Unit step responses of unstable transfer function for operator-based, GL, and proposed methods.

The proposed method gives very robust approximations for open loop unstable transfer functions. It is observed from Figure 4 that the proposed method gives the same results

as the GL method. Although operator-based methods have 8th-order approximations, it is seen that they have different oscillations. For this example, it was seen that the Oustaloup and M-SBL method gave the closest result among the operator-based methods.

A stable model was obtained by changing the fractional-order term in the middle in the denominator in Equation (37) to 1.35. The approximation calculated with the proposed method for the open loop stable fractional-order model is given in Equation (40). Bode and Nyquist curves for frequency response analysis of this model are given in Figures 5 and 6, respectively.

$$G_f(s) = \frac{1}{0.75s^{2.42} + 1.47s^{1.35} + 3.2} \tag{39}$$

$$G_m(s) = \frac{0.1671s^3 + 6.4425s^2 + 4.5125s + 22.2325}{s^5 + 9.0625s^4 + 14.8300s^3 + 53.8000s^2 + 34.4750s + 71.1500} \tag{40}$$

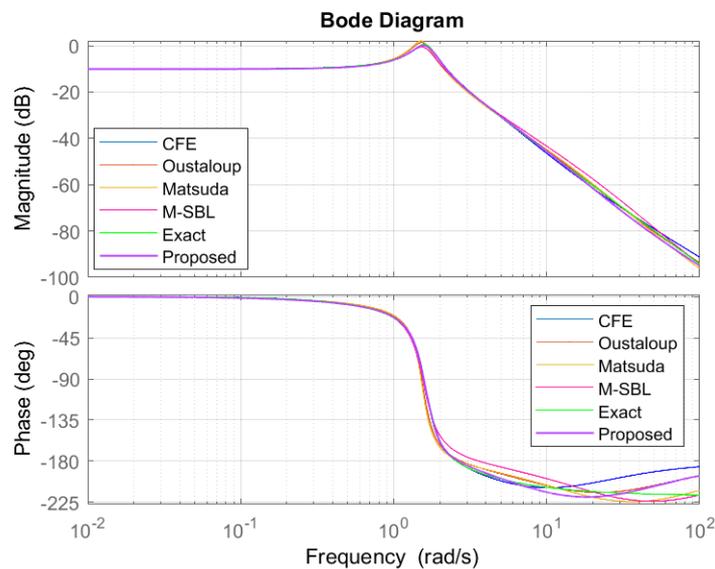


Figure 5. Bode diagram of stable transfer function for the operator-based, Exact, and Proposed methods.

The phase margin and gain margin values are important points on the bode diagram in control systems. It is seen in Figure 5 that, although the proposed method offers a 5th-degree approximation, it gives near-accurate results in data, such as phase margin and gain margin, compared to operator-based methods. The data measured on the Bode diagram are presented in Table 3. The proposed method is the best result in the phase margin. In the gain margin, it is the second-best result.

Table 3. Performance data of Bode and Nyquist diagrams.

Methods	Order of Approximation	Gain Margin (dB)	Phase Margin (Degree)	Critical Point
CFE	8	14.533	68.9600	−0.1877
Oustaloup	8	17.489	58.3900	−0.1337
Matsuda	8	17.100	51.9000	−0.1398
M-SBL	8	23.200	None	−0.0693
Proposed	5	15.713	99.4540	−0.1641
Exact	-	14.406	109.8400	−0.1904

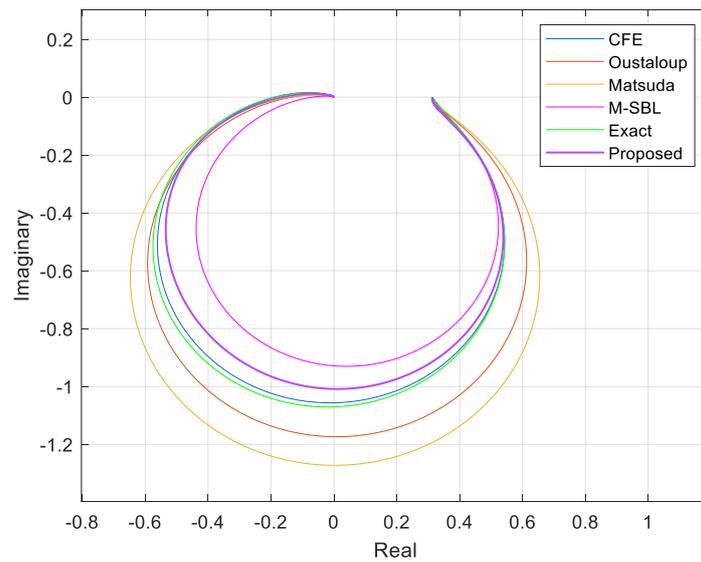


Figure 6. Nyquist diagram of stable transfer function for the operator-based, Exact, and Proposed methods.

Nyquist diagrams of operator-based approximation methods, for the Exact and the Proposed method, are shown in Figure 6. Here, the critical points where the curves intersect the real axis are examined. These points are also given in Table 3. As can be seen, the critical gain value of the CFE method is closer to the exact value. However, the CFE approximation is 8th-order. The proposed approximation is 5th-order, and it is the best result after CFE. Although the proposed method is of low order, it has an acceptable frequency characteristic compared to the other methods.

Example 3: Examination of the unity feedback control system with the fractional-order controller and the plant. Consider the fractional-order control system given in Figure 7.

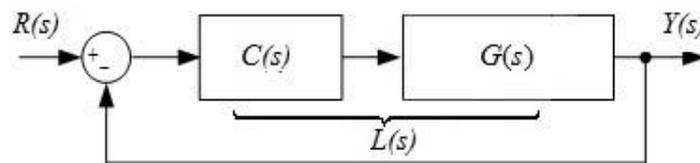


Figure 7. Closed loop control system.

In this example, it is shown how to compute integer-order approximation of a closed loop fractional-order control system by using the proposed method. Here, the fractional-order PI controller and plant transfer function are given in Equation (41). Additionally, the open loop transfer function is written, as shown in Equation (42). Thus, the closed loop transfer function can be computed, as shown in Equation (43).

$$G(s) = \frac{1.2s^{0.72} + 1.42}{2s^{0.72}} \text{ and } G(s) = \frac{0.64s^{1.17} + 3}{1.8s^{1.62} + 0.4s^{1.21} + 2s^{0.58} + 0.42} \tag{41}$$

$$L(s) = C(s)G(s) = \frac{0.768s^{1.89} + 0.9088s^{1.17} + 3.6s^{0.72} + 4.26}{3.6s^{2.34} + 0.8s^{1.93} + 4s^{1.3} + 0.84^{0.72}} \tag{42}$$

$$T(s) = \frac{L(s)}{1 + L(s)} = \frac{0.768s^{1.89} + 0.9088s^{1.17} + 3.6s^{0.72} + 4.26}{3.6s^{2.34} + 0.8s^{1.93} + 0.768s^{1.89} + 4s^{1.3} + 0.9088s^{1.17} + 4.44^{0.72} + 4.26} \tag{43}$$

As seen in (43), $T(s)$ has many fractional-orders, such as 0.89, 0.17, 0.72, 0.34, 0.93, and 0.30. When we use the 3rd-order operator-based approximation methods in this transfer function, the 29th-order integer approximation is obtained. Furthermore, when the 4th-order approximation methods are used, the 38th-order integer approximation is obtained. Namely, operator-based integer-order approximation methods produce very high-order transfer functions. Such higher order transfer functions cannot be suitable for real-world applications. However, when the proposed direct approximation method is used for $T(s)$, 4th-order integer approximation is obtained, as shown in (44).

$$T_m(s) = \frac{0.624s^3 + 1.933s^2 + 1.697s + 0.571}{s^4 + 2.611s^3 + 3.362s^2 + 2.115s + 0.571} \quad (44)$$

The unit step response and error plot of $T(s)$ and $T_m(s)$ are plotted in Figure 8 for the performance analysis of the proposed method. The error curve was obtained, according to the GL method. As is seen in Figure 8, the error margin of the proposed method is almost zero. Only at the first moment there is a maximum error margin of 0.034. It is shown that a low-order integer approximation is obtained due to the use of the proposed method. It also has a very low margin of error. This example also shows that very complex fractional-order transfer functions, such as $T(s)$, can be directly converted as a low-order integer-order transfer function by using the proposed method. The proposed method is also a helpful tool for applying classical control theory techniques to fractional-order control systems.

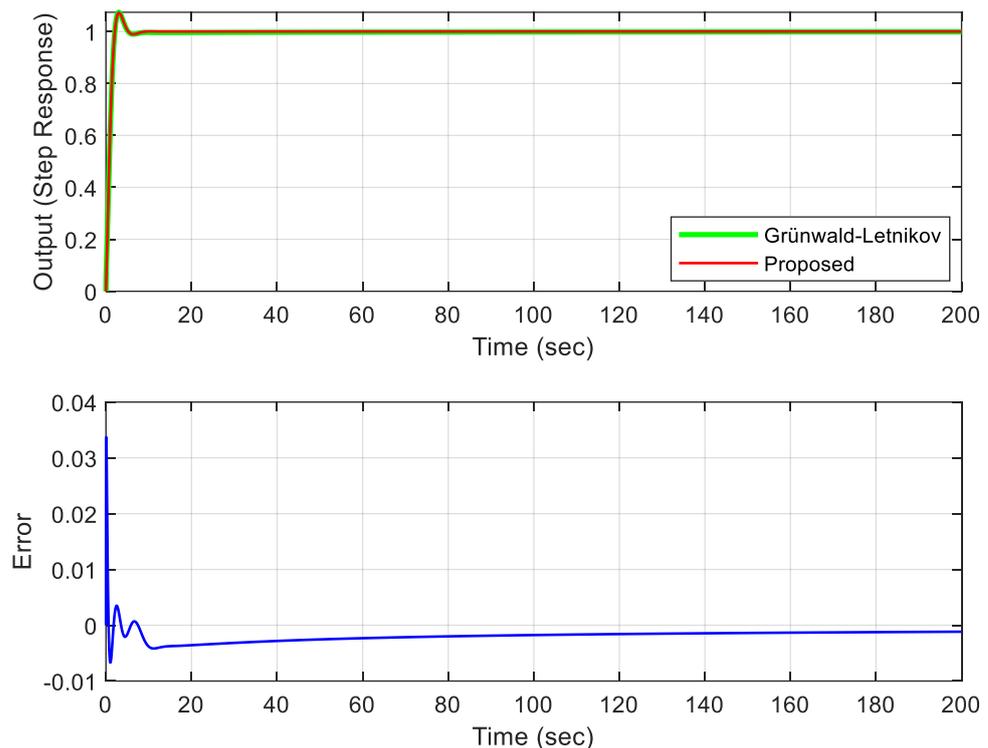


Figure 8. Unit step responses and error plot of $T_m(s)$ and $T(s)$.

4.2. Comparison with Direct Approximation Methods in the Literature

Example 4: Equation (45) is used widely in the literature for integer-order approximation. Techniques, such as PSO algorithm [35], block pulse functions and generalized operational matrices [47], Oustaloup approximation [48], and integer-order-approximation calculations using different model reduction [49] are often used.

$$G(s) = \frac{1}{0.8s^{2.2} + 0.5s^{0.9} + 1} \quad (45)$$

Integer-order approximations of Equation (45) are computed by above methods [35,47–49] and are given in Equations (46)–(51), respectively.

$$G_{PSO}(s) = \frac{1}{0.1772s^3 + 0.7329s^2 + 0.4463s + 1.0265} \quad (46)$$

$$G_{Block_Pulse1}(s) = \frac{1}{s^2 + 0.327s + 1.048} \quad (47)$$

$$G_{Block_Pulse2}(s) = \frac{-0.3213s + 1.525}{s^2 + 0.3255s + 1.571} \quad (48)$$

$$G_{BT}(s) = \frac{-0.2121s + 1.409}{s^2 + 0.1838s + 1.41} \quad (49)$$

$$G_{SPA}(s) = \frac{0.1777s^2 - 0.2739s + 1.437}{s^2 + 0.1932s + 1.438} \quad (50)$$

$$G_{HNA}(s) = \frac{-0.1072s^2 - 0.3346s + 1.421}{s^2 + 0.1876s + 1.422} \quad (51)$$

The integer-order approximate transfer function is computed by the proposed method, and the approximate transfer function is shown in Equation (52). Unit step response of the transfer functions is computed by the literature [35,47–49], and the proposed methods are superimposed in Figure 9.

$$G_{Proposed}(s) = \frac{0.8844s^3 + 1.8085s^2 + 1.6777s + 2.1596}{s^5 + 1.5691s^4 + 3.4532s^3 + 3.8213s^2 + 2.7957s + 2.1596} \quad (52)$$

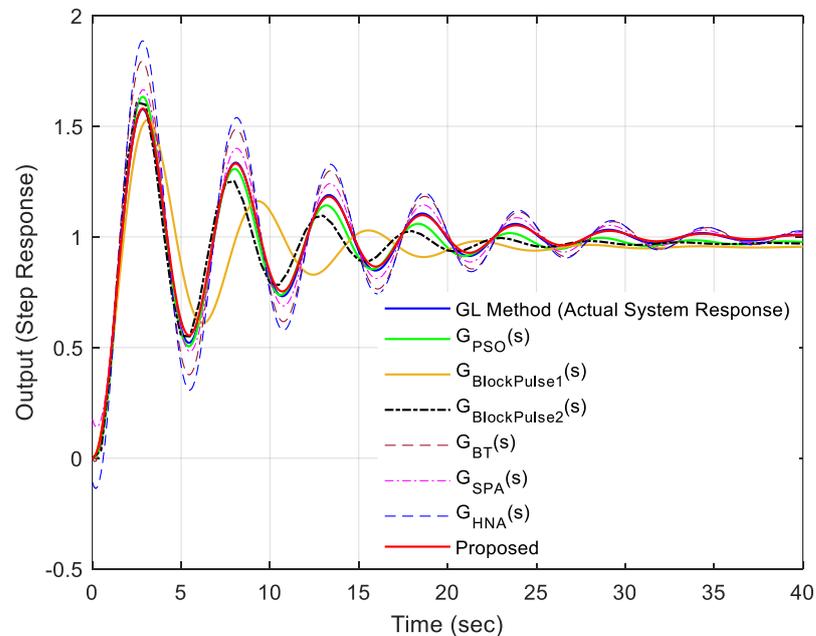


Figure 9. Comparison of unit step responses in Equations (45)–(52).

As shown in Figure 9, the proposed method follows the actual system response more accurately than the others. Furthermore, performance characteristics (rise time, settling time, peak time, peak value, overshoot, and integral square error (ISE)) are compared

among the methods in Table 4. The proposed method has the closest response to the GL method when compared to transfer functions with positive coefficients. The steady-state error of the transfer function obtained by the proposed method is exact compared to other methods. Three fundamental properties of the proposed method can be observed in Figure 9 and Table 4: it can compute the positive coefficient approximate transfer function, it gives nearly exact results in time response performance characteristics, and it results in the same steady-state-error as GL.

Table 4. Time response performance characteristics of Equations (45)–(52).

Methods	Rise Time (s)	Settling Time (5%) (s)	Peak Time (s)	Peak Value	Overshoot (%)	Steady State Error
GL Method	1.0630	21.9606	2.8400	1.5658	56.5764	0
$G_{Proposed}(s)$	1.0719	24.0946	2.8500	1.5799	57.9923	0
$G_{PSO}(s)$	0.9951	26.6459	2.8400	1.6332	63.3222	0.0258
$G_{Block_Pulse1}(s)$	1.1777	31.8859	3.1100	1.5282	52.8180	0.0458
$G_{Block_Pulse2}(s)$	0.8756	21.1973	2.7300	1.6351	63.5147	0.0293
$G_{BT}(s)$	0.8883	32.3321	2.8000	1.7945	79.4513	0.0007
$G_{SPA}(s)$	0.8388	29.6601	2.8800	1.6662	66.6186	0.0007
$G_{HNA}(s)$	0.8679	32.1206	2.8400	1.8865	88.6464	0.0007

Example 5: In another study, Oustaloup's recursive filter and a refined Oustaloup's filter, examined in [46,48], are considered for integer-order approximations. Integer-order transfer functions are obtained by using the filters. Then, using some model reduction techniques, lower-order transfer functions are obtained by preserving the dominant behavior of higher-order integer transfer functions. One of the applications in [48] is given in Equation (53), and the equation is considered in this paper, also.

$$G(s) = \frac{5}{s^{2.3} + 1.3s^{0.9} + 1.25} \quad (53)$$

The proposed method offers a 5th-order approximation. For the accuracy of the comparison, the transfer function of order 5 or more is chosen in [48], which is given in Equation (54).

$$G_{pade46}(s) = \frac{4.334s^4 + 3.488s^3 + 0.6366s^2 + 0.03234s + 0.0004762}{s^6 + 1.504s^5 + 2.152s^4 + 1.091s^3 + 0.173s^2 + 0.008392s + 0.000121} \quad (54)$$

The fractional-order transfer function in Equation (53) results an integer-order transfer function given in Equation (55) obtained by the proposed method.

$$G_{proposed}(s) = \frac{1.2751s^3 + 15.5456s^2 + 13.9474s + 4.7263}{s^5 + 3.6807s^4 + 6.5421s^3 + 8.0526s^2 + 5.0702s + 1.1816} \quad (55)$$

The unit-step-responses of the proposed method, reduced order method, and GL method are superimposed in Figure 10. The proposed method follows the exact method more accurately than the other as shown in Figure 10. Moreover, the order of the approximate-integer-order transfer function computed by the proposed method is less than the order of the approximate-integer-order transfer function in [48].

In Table 5, it is seen that the proposed method is more accurate than the reduced order model given in [48] in terms of performance characteristics, such as rise time, peak time, peak value, overshoot, and steady-state error.

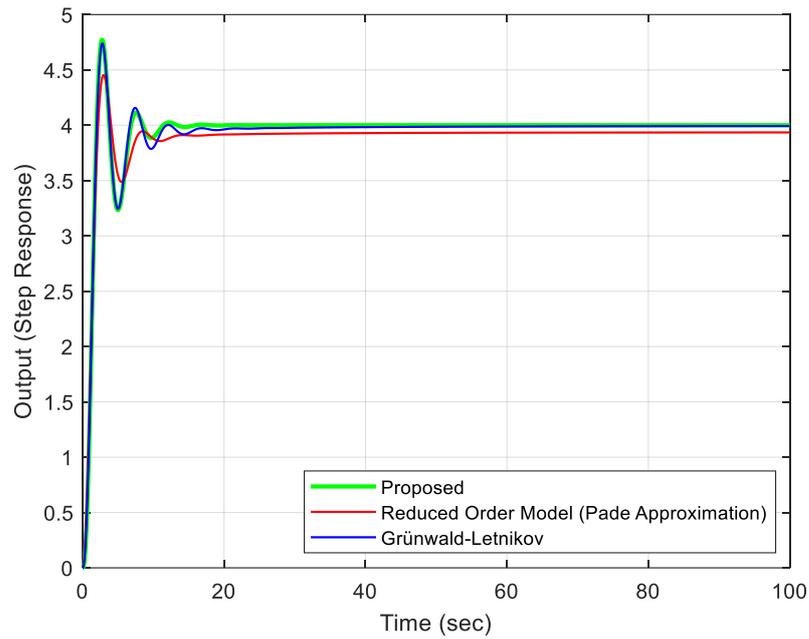


Figure 10. Comparison of step responses of the integer-order approximate transfer functions.

Table 5. Time response performance characteristics of Equations (53)–(55).

Methods	Rise Time (s)	Settling Time (5%) (s)	Peak Time (s)	Peak Value	Overshoot (%)	Steady State Error
GL Method	1.2786	10.0216	2.7800	4.7386	18.4639	0
$G_{Proposed}(s)$	1.2342	6.3591	2.7500	4.7723	19.3085	0
$G_{pade46}(s)$	1.4545	7.1269	2.9600	4.4537	11.3415	0.07

Design of a PID controller for the fractional-order transfer function as in Equation (53) is as follows: Routh-Hurwitz stability criterion cannot be applied for fractional-order transfer functions. However, the Routh-Hurwitz criterion can be successfully applied to the integer-order approximate transfer function $G_{proposed}(s)$ derived by using the proposed method. Additionally, PID controller design can be done easily using Ziegler-Nichols method. For the design method, the process is taken to closed-loop proportional control (K), as in Figure 11, and the closed-loop transfer function of the system is written as Equation (56).

$$H(s) = \frac{KG_{proposed}(s)}{1+KG_{proposed}(s)}$$

$$H(s) = \frac{1.2751Ks^3+15.5456Ks^2+13.9474Ks+4.7263K}{s^5 + 3.6807s^4 + (6.5421 + 1.2751K)s^3 + (8.0526 + 15.5456K)s^2 + (5.0702 + 13.9474K)s + (1.1816 + 4.7263K)} \tag{56}$$

$$\Delta(s) = s^5 + 3.6807s^4 + (6.5421 + 1.2751K)s^3 + (8.0526 + 15.5456K)s^2 + (5.0702 + 13.9474K)s + (1.1816 + 4.7263K) \tag{57}$$

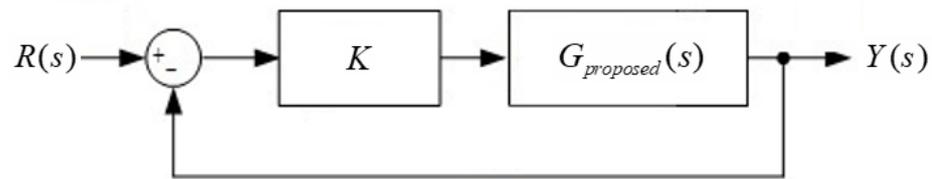


Figure 11. Closed loop control system with proportional controller.

For the characteristic equation given in Equation (57), the Routh-Hurwitz stability criterion method is applied as given in Equation (58). The critical gain K_{cr} and critical frequency ω_{cr} and critical period T_{cr} of the system are easily calculated using the Routh-Hurwitz method.

$$\begin{array}{l|lll}
 s^5 & 1 & (6.5421 + 1.2751K) & (5.0702 + 13.9474K) \\
 s^4 & 3.6807 & (8.0526 + 15.5456K) & (1.1816 + 4.7263K) \\
 s^3 & A & B & 0 \\
 s^2 & C & D & 0 \\
 s^1 & E & 0 & 0 \\
 s^0 & F & 0 & 0
 \end{array}$$

$$\begin{aligned}
 A &= 4.3543 - 2.9484K \\
 B &= 12.6633K + 4.7491 \\
 C &= \frac{[(45.8353K^2 + 2.6621K - 17.5832)]}{[(2.9484K - 4.3543)]} \\
 D &= \frac{[(47,263K + 11,816)(2.9484 \times 10^{-4}K - 4.3543 \times 10^{-4})]}{[(2.9484K - 4.3543)]} \\
 E &= -\frac{[(2.9484K - 4.3543)(-3.1076 \times 10^{34}K^3 - 7.0154 \times 10^{33}K^2 + 7.5374 \times 10^{33}K + 3.0551 \times 10^{33})]}{[(1.4742 \times 10^{32}K - 2.1772 \times 10^{32})(45.8353K^2 + 2.6621K - 17.5832)]} \\
 F &= \frac{[(47,263K + 11,816)(1.4742 \times 10^{65}K - 2.1772 \times 10^{65})(45.8353K^2 + 2.6621K - 17.5832)]}{\left[\begin{array}{l} (-1.5537 \times 10^{33}K^3 - 3.5077 \times 10^{32}K^2 + 3.7687 \times 10^{32}K + 1.5276 \times 10^{32}) \\ (2.9484K - 4.3543)(2.2917 \times 10^{33}K^2 + 1.3311 \times 10^{32}K - 8.7916 \times 10^{32}) \\ (-1.5537 \times 10^{70}K^3 - 3.5076 \times 10^{69}K^2 + 3.7687 \times 10^{69}K + 1.5275 \times 10^{69}) \end{array} \right]}
 \end{aligned} \tag{58}$$

In order to calculate the critical gain of the system, it is determined as $K_{cr} = K = 0.5467$ by using the Matlab `fminsearch` command at $E = 0$ in Equation (58). Using the K gain value, the equations C and D are calculated as 0.8855 and 3.7654 , respectively. Thus, from the auxiliary equation $0.8855s^2 + 3.7654 = 0$, the critical frequency is obtained as $\omega_{cr} = 2.0621$ rad/s. In this case, the period of the oscillating signal is calculated as $T_{cr} = 3.0470$ s. According to the Ziegler-Nichols Method, the parameters of the PID controller are calculated using Equation (59).

$$\begin{aligned}
 C(s) &= K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right) \\
 K_p &= 0.5882K_{cr} \\
 \tau_i &= 0.5 * T_{cr} \\
 \tau_d &= 0.125T_{cr}
 \end{aligned} \tag{59}$$

When the critical gain and critical period values are substituted in Equation (59), the PID controller parameters are calculated as $K_p = 0.3216$, $\tau_i = 1.5235$, and $\tau_d = 0.3809$. Thus, the PID controller transfer function is obtained as Equation (60). Closed-loop unit step responses of $G_{proposed}(s)$, $G_{pade}(s)$, and $G(s)$, using the PID controller, are presented in Figure 12. The closed-loop unit step response of the fractional transfer function $G(s)$ is obtained using the IFTM method. In Figure 12, it is seen that the closed loop unit step response of $G_{proposed}(s)$ successfully tracks of $G(s)$. This shows that $G_{proposed}(s)$ is a successful integer-order approximation. It is seen that $G_{pade}(s)$, which is the approximation obtained using the Pade method, obviously has higher error.

$$C(s) = 0.3216 + \frac{0.2111}{s} + 0.1225s \tag{60}$$

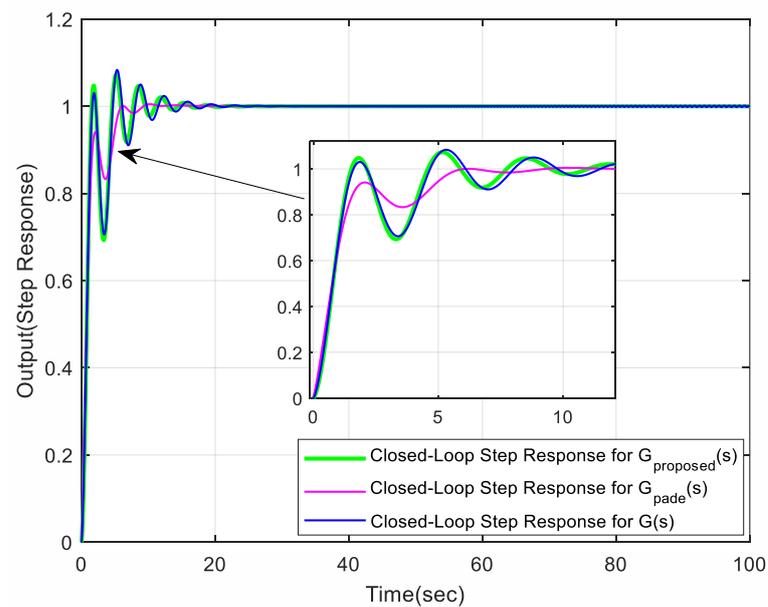


Figure 12. Comparison of closed-loop step responses for $G_{proposed}(s)$, $G_{pade}(s)$ and $G(s)$.

5. Conclusions

Integer-order approximations are very important for applying classical control theory rules and simulations on fractional-order transfer functions. In this study, an alternative method has been developed to directly obtain the integer-order approximation of fractional-order transfer functions. The proposed method applied to perform an approximation in both open-loop stable and unstable fractional-order transfer functions. The method is based on curve fitting with the EO algorithm, which has been developed in recent years. Operator-based integer-order approximation methods, such as Oustaloup, Matsuda, Carlson, Chareff, etc. can produce very large order transfer functions. The proposed method not only provides a low order transfer function approximation, but also has a lower error margin compared to existing methods with higher order transfer function approximation. It has been shown, in examples, that the proposed method can easily transform a fractional-order transfer function directly to approximate integer-order transfer function. The proposed method produces a maximum 5th-order approximate transfer function. In this way, the computations, such as time response, frequency response, stability analysis, controller design, etc. can be easily performed in control theory. In addition, analytical controller designs can be made by converting the approximate transfer function to standard forms by using model reduction methods. The Routh-Hurwitz stability criterion can be easily applied. In this way, the P/PI/PID controller can be designed using the Ziegler-Nichols method. In addition, the proposed method can be used successfully in closed loop control systems which include both fractional controller and plant. Thus, fractional-order closed-loop control systems can be investigated as an integer-order transfer function. Comparisons with other methods are presented on widely used applications in the literature. In Example 4, among the positive coefficient approximate transfer functions, it is seen that the proposed method is closest to the GL method in performance characteristics, such as rise time, settling time, peak value, overshoot, and steady-state error. In Example 5, a 6th-order integer approximation presented in the literature is compared with the 5th-order integer-order approximation, which was computed using the proposed method. It is seen that the proposed method is closest to the GL method in terms of rise time, peak time, peak value, over-shoot, and steady-state error performance characteristics. In both examples, five of the six performance criteria are very close to the performance criteria of the GL method, which give accurate results. In addition, Example 5, a successful PID controller design has been made using the Routh-Hurwitz stability criterion. It has been understood that the proposed method gives more efficient and accurate results than the

methods presented in Examples 4 and 5. This study can be used widely as a more efficient tool in fractional control systems calculations. In addition, the transfer function structure obtained by the proposed method is suitable for electronic circuit implementations, such as fractional-order filter circuit. In the future, the study of the electronic implementations of filters with two fractional-orders on these results can be considered.

Funding: This research received no external funding.

Data Availability Statement: The study did not report any data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Monje, C.A.; Chen, Y.; Vinagre, B.M.; Xue, D.; Feliu-Batlle, V. *Fractional-Order Systems and Controls: Fundamentals and Applications*; Springer: London, UK, 2010. [CrossRef]
2. Yu, W.; Luo, Y.; Pi, Y. Fractional order modeling and control for permanent magnet synchronous motor velocity servo system. *Mechatronics* **2013**, *23*, 813–820. [CrossRef]
3. Das, S. *Functional Fractional Calculus for System Identification and Controls*; Springer-Verlag: Berlin/Heidelberg, Germany, 2008.
4. Zhao, K. Existence, stability and simulation of a class of nonlinear fractional Langevin equations involving nonsingular Mittag-Leffler kernel. *Fractal Fract.* **2022**, *6*, 469. [CrossRef]
5. Rendón, J.E.; Mejía, C.E. Fractional order modeling of a nonlinear electromechanical system. *Enfoque UTE* **2018**, *9*, 45–56.
6. Hartley, T.T.; Lorenzo, C.F. Dynamics and control of initialized fractional-order systems. *Nonlinear Dyn.* **2002**, *29*, 201–233. [CrossRef]
7. Magin, R.L. *Fractional Calculus in Bioengineering*; Begell House Inc.: Redding, CT, USA, 2006; Volume 2.
8. Zhang, T.; Xiong, L. Periodic motion for impulsive fractional functional differential equations with piecewise Caputo derivative. *Appl. Math. Lett.* **2020**, *101*, 106072. [CrossRef]
9. Zhang, T.; Li, Y. Exponential Euler scheme of multi-delay Caputo–Fabrizio fractional-order differential equations. *Appl. Math. Lett.* **2022**, *124*, 107709. [CrossRef]
10. Rauh, A. Exponential Enclosures for the Verified Simulation of Fractional-Order Differential Equations. *Fractal Fract.* **2022**, *6*, 567. [CrossRef]
11. Chen, Y.; Petras, I.; Vinagre, B. A list of Laplace and Inverse Laplace Transforms Related to Fractional Order Calculus. 2001, pp. 1–3. Available online: http://ivopetras.tripod.com/foc_laplace.pdf (accessed on 6 January 2023).
12. Iqbal, A.; Shekh, R.R. A comprehensive study on different approximation methods of Fractional order system. *Int. Res. J. Eng. Technol. IRJET* **2016**, *3*, 1848–1863.
13. Tepljakov, A.; Petlenkov, E.; Belikov, J. FOMCON: Fractional-order modeling and control toolbox for MATLAB. In Proceedings of the 2011 18th International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES), Gliwice, Poland, 16–18 June 2011; pp. 684–689.
14. Podlubny, I. *Fractional Differential Equations*; Academic Press: Cambridge, MA, USA, 1998; Volume 198.
15. Polubny, I. Fractional-order systems and PID μ controller. *IEEE Trans. Autom. Control.* **1999**, *44*, 208–214. [CrossRef]
16. Atherton, D.P.; Tan, N.; Yüce, A. Methods for computing the time response of fractional-order systems. *IET Control. Theory Appl.* **2014**, *9*, 817–830. [CrossRef]
17. Vinagre, B.; Podlubny, I.; Hernandez, A.; Feliu, V. Some approximations of fractional order operators used in control theory and applications. *Fract. Calc. Appl. Anal.* **2000**, *3*, 231–248.
18. Krishna, B. Studies on fractional order differentiators and integrators: A survey. *Signal Process.* **2011**, *91*, 386–426. [CrossRef]
19. Oustaloup, A.; Levron, F.; Mathieu, B.; Nanot, F.M. Frequency-band complex noninteger differentiator: Characterization and synthesis. *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **2000**, *47*, 25–39. [CrossRef]
20. Carlson, G.; Halijak, C. Approximation of fractional capacitors $(1/s)^{(1/n)}$ by a regular Newton process. *IEEE Trans. Circuit Theory* **1964**, *11*, 210–213. [CrossRef]
21. Matsuda, K.; Fujii, H. H \sim Optimized Wave-Absorbing Control: Analytical and Experimental Results. *J. Guid. Control. Dyn.* **1993**, *16*, 1146. [CrossRef]
22. Charef, A.; Sun, H.; Tsao, Y.; Onaral, B. Fractal system as represented by singularity function. *IEEE Trans. Autom. Control.* **1992**, *37*, 1465–1470. [CrossRef]
23. Khanra, M.; Pal, J.; Biswas, K. Rational approximation of fractional operator—A comparative study. In Proceedings of the 2010 International Conference on Power, Control and Embedded Systems, Allahabad, India, 29 November–1 December 2010; pp. 1–5.
24. Deniz, F.N.; Alagoz, B.B.; Tan, N.; Atherton, D.P. An integer order approximation method based on stability boundary locus for fractional order derivative/integrator operators. *ISA Trans.* **2016**, *62*, 154–163. [CrossRef]
25. Yüce, A.; Tan, N. A New Integer Order Approximation Table for Fractional Order Derivative Operators. In Proceedings of the IFAC 2017 World Congress, Toulouse, France, 9–14 July 2017; pp. 9736–9741.
26. Shrivastava, N.; Varshney, P. Frequency band implementation of non-integer order functions. In Proceedings of the 5th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 22–23 February 2018; pp. 852–857.

27. Deniz, F.N.; Alagoz, B.B.; Tan, N.; Koseoglu, M. Revisiting four approximation methods for fractional order transfer function implementations: Stability preservation, time and frequency response matching analyses. *Annu. Rev. Control.* **2020**, *49*, 239–257. [[CrossRef](#)]
28. Casagrande, D.; Krajewski, W.; Viaro, U. The Integer–Order Approximation of Fractional–Order Systems in the Loewner Framework. *IFAC-PapersOnLine* **2019**, *52*, 43–48. [[CrossRef](#)]
29. Krajewski, W.; Viaro, U. A method for the integer-order approximation of fractional-order systems. *J. Frankl. Inst.* **2014**, *351*, 555–564. [[CrossRef](#)]
30. Mansouri, R.; Bettayeb, M.; Djennoune, S. Optimal reduced-order approximation of fractional dynamical systems. In Proceedings of the AIP Conference Proceedings, Annaba, Algeria, 30 June–2 July 2008; pp. 127–132.
31. Rachid, M.; Maamar, B.; Said, D. Comparison between two approximation methods of state space fractional systems. *Signal Process.* **2011**, *91*, 461–469. [[CrossRef](#)]
32. Tavazoei, M.S.; Haeri, M. Rational approximations in the simulation and implementation of fractional-order dynamics: A descriptor system approach. *Automatica* **2010**, *46*, 94–100. [[CrossRef](#)]
33. Mihaly, V.; Şuşcă, M.; Dulf, E.H.; Dobra, P. Approximating the Fractional-Order Element for the Robust Control Framework. In Proceedings of the 2022 American Control Conference (ACC), Atlanta, GA, USA, 8–10 June 2022; pp. 1151–1157.
34. Mouhou, A.; Badri, A. Low Integer-Order Approximation of Fractional-Order Systems Using Grey Wolf Optimizer-Based Cuckoo Search Algorithm. *Circuits Syst. Signal Process.* **2022**, *41*, 1869–1894. [[CrossRef](#)]
35. Maiti, D.; Konar, A. Approximation of a fractional order system by an integer order model using particle swarm optimization technique. *arXiv* **2008**, arXiv:0811.0077.
36. Soloklo, H.N.; Bigdeli, N. Direct approximation of fractional order systems as a reduced integer/fractional-order model by genetic algorithm. *Sādhanā* **2020**, *45*, 277. [[CrossRef](#)]
37. Mahata, S.; Saha, S.; Kar, R.; Mandal, D. Optimal integer-order rational approximation of α and $\alpha + \beta$ fractional-order generalised analogue filters. *IET Signal Process.* **2019**, *13*, 516–527. [[CrossRef](#)]
38. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl. Based Syst.* **2020**, *191*, 105190. [[CrossRef](#)]
39. Nazaroff, W.W.; Alvarez-Cohen, L. *Environmental Engineering Science*; John Wiley & Sons: Hoboken, NJ, USA, 2000.
40. Dogruer, T. Design of I-PD Controller Based Modified Smith Predictor for Processes with Inverse Response and Time Delay Using Equilibrium Optimizer. *IEEE Access* **2023**, *11*, 14636–14646. [[CrossRef](#)]
41. Haidekker, M.A. *Linear Feedback Controls: The Essentials*; Elsevier: Amsterdam, The Netherlands, 2020.
42. Sundaram, S. ECE 380: Control systems. In *Course Notes*; Department of Electrical and Computer Engineering, University of Waterloo: Waterloo, ON, Canada, 2013; Available online: https://engineering.purdue.edu/~sundara2/misc/ece380_notes.pdf (accessed on 10 January 2023).
43. Graham, D.; Lathrop, R.C. The synthesis of “ optimum ” transient response: Criteria and standard forms. *Trans. Am. Inst. Electr. Eng. Part II Appl. Ind.* **1953**, *72*, 273–288. [[CrossRef](#)]
44. Atherton, D.; Boz, A. Using standard forms for controller design. In Proceedings of the UKACC International Conference on Control’98 (Conf. Publ. No. 455), Swansea, UK, 1–4 September 1998; pp. 1066–1071.
45. Joseph, S.B.; Dada, E.G.; Abidemi, A.; Oyewola, D.O.; Khammas, B.M. Metaheuristic algorithms for PID controller parameters tuning: Review, approaches and open problems. *Heliyon* **2022**, e09399. [[CrossRef](#)]
46. Xue, D.; Chen, Y.; Atherton, D.P. *Linear Feedback Control: Analysis and Design with MATLAB*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2007.
47. Wang, C.-H.; Chen, C.-Y. Finding the integer order systems for fractional order systems via fractional operational matrices. In Proceedings of the 2012 9th IEEE International Conference on Networking, Sensing and Control, Beijing, China, 11–14 April 2012; pp. 267–270.
48. Senol, B.; Yeroglu, C. Filter approximation and model reduction comparison for fractional order systems. In Proceedings of the ICFDA’14 International Conference on Fractional Differentiation and Its Applications 2014, Catania, Italy, 23–25 June 2014; pp. 1–6.
49. Verma, S.K.; Nagar, S.K. Approximation and order reduction of fractional order SISO system. In Proceedings of the 2016 IEEE Annual India Conference (INDICON), Bangalore, India, 16–18 December 2016; pp. 1–6.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.