AIMS Mathematics, 8(1): 657–675.
DOI: 10.3934/math.2023031
Received: 14 July 2022
Revised: 22 August 2022
Accepted: 29 August 2022
Published: 09 October 2022

http://www.aimspress.com/journal/Math

Research article

Optimal design of $PI^{\rho}D^{\mu}$-controller for artificial ventilation systems for COVID-19 patients

Iqbal M. Batiha[1,2], Reyad El-Khazali[3,*], Osama Y. Ababneh[4], Adel Ouannas[5], Radwan M. Batyha[6] and Shaher Momani[2,7]

[1] Department of Mathematics, Al Zaytoonah University of Jordan, Amman 11733, Jordan
[2] Nonlinear Dynamics Research Center (NDRC), Ajman University, Ajman, UAE
[3] EECS Department, Khalifa University, Abu-Dhabi, UAE
[4] Department of Mathematics, Zarqa University, Zarqa, Jordan
[5] Department of Mathematics and Computer Science, University of Larbi Ben M'hidi, Oum El Bouaghi, Algeria
[6] Department of Computer Science, Faculty of Science and Technology, Irbid National University, Irbid, Jordan
[7] Department of Mathematics, Faculty of Science, University of Jordan, Amman 11942, Jordan

* **Correspondence:** Email: reyad.elkhazali@ku.ac.ae; Tel: +971507563559; Fax: +97123123978.

Abstract: In light of the COVID-19 pandemic, many patients have suffered from Acute Respiratory Distress Syndrome (ARDS) in Intensive Care Units (ICUs) around the world. In the medical field, it is known that the so-called artificial ventilation device, which has become the mainstay of treatment of such syndrome, decreases mortality in critically ill COVID-19 patients. Due to the high reliability of this device, there is an emergency need to follow up the progress made on designing a robust controller for improving its performance. From this perspective, this work introduces different control design schemes for obtaining an optimal Fractional-order PID controller (or simply $PI^{\rho}D^{\mu}$-controller) of the Artificial Ventilation (AV) system through two optimization algorithms: the Bacteria Foraging Optimization (BFO) and the Particle Swarm Optimization (PSO) algorithms. The realization of the controller is accomplished using four approximations: Oustaloup's approximation, the Continued Fractional Expansion (CFE) approximation and the 1^{st}- and 2^{nd}-order El-Khazali approximations. The validation of the controller design and the AV system behavior are verified via numerical simulation in order to demonstrate the effectiveness and the potency of all proposed schemes.

Keywords: $PI^{\rho}D^{\mu}$-controller; particle swarm optimization algorithm; Bacteria Foraging Optimization; Laplacian operator

Mathematics Subject Classification: 47N70, 65K10

1. Introduction

Nowadays, all hospitals around the world have confronted a huge surge of COVID-19 patients that require emergency health-care and services. This matter, however, has caused an enormous pressure on the available resources in many hospitals, especially with limited resources [1]. Meanwhile, worldwide knowledge and expertise have rapidly developed new and innovative ideas to reduce the spread of COVID-19 and treat infected patients. Acute Respiratory Distress Syndrome (ARDS) is one of the likely symptoms of coronavirus patients that can possibly cause death [2]. This intractable symptom can be overcome using one of the significant medical devices called an Artificial Ventilator (AV). In general, such a device is put forward to help individuals to regain healthy breathing when normal breathing is not possible. The principle of its operation is based on introducing oxygen into the lungs and withdrawing extra carbon dioxide from the body. This would enhance the gas exchange rate and hence provide comfortable breathing of sufferers. In view of technological advancement and continual modernization in introducing some technical procedures associated with control theory for medical devices, an AV system is one of these typical devices (see, e.g., [3–8]). Hence, for complementing this work, this paper proposes several optimal proportional-integral-derivative fractional-order controllers (or simply $PI^{\rho}D^{\mu}$-controllers), where $0 < \rho < 1$ and $0 < \mu < 1$ represent the orders of the integral and the differential parts of the controller, respectively. This is for the purpose of developing dynamic stabilization capabilities of the Volume-Controlled Artificial Ventilation (VCAV) system, see Figure 1.

Figure 1. Modern high class artificial ventilator needed by most COVID-19 patients in ICU.

It is known that the optimized $PI^{\rho}D^{\mu}$-controller has better performance than the traditional PID controller. As is known to all, it can offer more degrees of freedom by adding two parameters to the construction of the traditional one, as was reported in numerous references. For instance, it was demonstrated in [9] that one can obtain better simulation results with the use of the $PI^{\rho}D^{\mu}$-controller than with the integer-order PID controllers when we deal with some time-delay systems. In addition,

the same result was presented in [10], when the authors designed a multivariable decoupling $PI^{\rho}D^{\mu}$-controller for variable air volume systems.

In this work, all proposed controllers have been optimally designed by carrying out two optimization techniques, namely, Particle Swarm Optimization (PSO) [11] and Bacteria Foraging Optimization (BFO) [12], for the purpose of fulfilling the high performance of the VCAV system. In particular, the key role of these two techniques is minimizing the objective function's value in light of specific constraints related to time or frequency domain, such as the system's rise time, overshoot and settling time. These constraints have a major task in quantifying the robustness of the controlled system. It is entirely normal that, once the two algorithms generate successfully the five optimal parameters (κ_p, κ_i, κ_d, ρ and μ) of the $PI^{\rho}D^{\mu}$-controller, the focus would be oriented towards the needs of approximating the so-called fractional-order Laplacian operator s^{ρ} and/or s^{μ} (or simply $s^{\pm\gamma}$, where $\gamma = \{\rho,\ \mu\}$ and $0 < \gamma < 1$). Such approximation would be in the form of a finite integer-order rational transfer function due to the arbitrariness of s^{γ} in its original form [13–18]. Actually, this new analytical function of s^{γ}, which permits one to design and analyze the system without a need to address some hard time-domain compositions, can be typically derived by applying multiple approximations, like Oustaloup's approximation [19, 20], the Continued Fraction Expansion (CFE) [19, 21] approximation and more recently the 1^{st}- and 2^{nd}-order El-Khazali approximations [22, 23]. In this work, only these four approximations will be used in order to provide the operators $s^{\pm\gamma}$ with their corresponding transfer functions. Thus, we intend to tune eight $PI^{\rho}D^{\mu}$-BFO/PSO controllers by generating them through implementing the two aforesaid optimization algorithms. Numerical simulations of these eight controllers provide sufficient information to adopt the best performance of the controlled AV system, such as settling time, minimal overshoot and least rise time.

The rest of the paper is arranged as follows: An overview of the fractional-order linear time-invariant system is presented in the next section, followed by presenting some basic concepts of designing $PI^{\rho}D^{\mu}$-controllers in Section 2. Four approximations of finite-order rational transfer functions of the fractional-order integro-differential Laplacian operators are outlined in Section 3, while the final section includes the main results and all numerical outcomes of the proposed design methods.

2. Fractional-order linear time-invariant systems

Fundamentally, the elementary principles of fractional calculus are typically utilized as a tool to transmit a class of control systems called the integer-order linear-time invariant (LTI) system into its fractional-order version, which is called the fractional-order LTI system (or simply FoLTI system) [24, 25]. In light of some manifestations presented in [24, 26], it has been confirmed that the FoLTI systems exceed the other integer-order counterparts by virtue of their flexibility in taking into account more additional parameters. In general, a FoLTI system can be expressed by the following fractional-order differential equation [24, 27]:

$$p_n D^{\delta_n} y(t) + p_{n-1} D^{\delta_{n-1}} y(t) + ... + p_1 D^{\delta_1} y(t) + p_0 D^{\delta_0} y(t) = q_m D^{\nu_m} u(t) + q_{m-1} D^{\nu_{m-1}} u(t) + ... + q_1 D^{\nu_1} u(t) + q_0 D^{\nu_0} u(t), \tag{2.1}$$

where $y(t)$ and $u(t)$ are two variables over the time t that indicate the control output and input of the system, respectively, and where $D^{\{\delta_i, \nu_k\}}$ represents the Caputo fractional differential operator of orders

$\delta_i, i = 1, 2, 3, ..., n$, and $\nu_k, k = 1, 2, 3, ..., m$, such that $n, m \in \mathbb{N}$.

Modeling system (2.1) can be accomplished by replacing $s^{\pm}\gamma$ by finite-order rational transfer functions, where $0 < \gamma < 1$. This would permit one to design and analyze the controlled system with finite-order controllers. The numerical method that generates such an approximate transfer function determines the bounds on the order of the controller. In a similar context, the frequency response of the LTI system can be represented by a transfer function which is the ratio of the Laplace transform of the system's output to the Laplace transform of its input for zero initial conditions [27], i.e.,

$$T(s) = \frac{Y(s)}{U(s)} = \frac{q_m s^{\nu_m} + q_{m-1} s^{\nu_{m-1}} + ... + q_1 s^{\nu_1} + q_0 s^{\nu_0}}{p_n s^{\delta_n} + p_{n-1} s^{\delta_{n-1}} + ... + p_1 s^{\delta_1} + p_0 s^{\delta_0}}, \tag{2.2}$$

where $Y(s) = \mathscr{L}\{y(t)\},\ U(s) = \mathscr{L}\{u(t)\}$ are the Laplace transforms of $y(t)$ and $u(t)$, respectively.

3. Design of $PI^{\rho}D^{\mu}$ controllers

The principal structure of the $PI^{\rho}D^{\mu}$-controller was proposed by Podlubny et al. [28]. It was shown that the $PI^{\rho}D^{\mu}$-controller outperforms the traditional PID-controller. From this vantage point, many real-life technical applications have been improved by applying this controller. The structure of this controller is based upon appending two further parameters (ρ and μ) to the primary parameters ($\kappa_p, \kappa_i, \kappa_d$) of the traditional PID-controller. Those two additional parameters would offer additional degrees of freedom in the controller algorithms. Nevertheless, the $PI^{\rho}D^{\mu}$-controller is definitely inferred from the following fractional-order integro-differential equation [24, 28]:

$$u(t) = \kappa_p e(t) + \kappa_i I^{\rho} e(t) + \kappa_d D^{\mu} e(t), \tag{3.1}$$

where $e(t)$ is the error signal, D^{μ} is the Caputo operator of order μ, and I^{ρ} is the Riemann-Liouville operator of order ρ. The transfer function of the $PI^{\rho}D^{\mu}$-controller is given by

$$C(s) = \frac{U(s)}{E(s)} = \kappa_p + \kappa_i s^{-\rho} + \kappa_d s^{\mu}, \tag{3.2}$$

where $E(s) = \mathscr{L}\{e(t)\}$.

The subsequent mission concentrates on using the $PI^{\rho}D^{\mu}$-controller with one of the most significant industrial systems during this coronavirus time: the AV system. The key purpose of implementing such a controller is to further improve the process control of the VCAV system by enhancing its dynamic performance. This demands implementing a robust optimization algorithm to enhance the system's step response by means of optimally designing the five parameters of the $PI^{\rho}D^{\mu}$-controller. For this purpose, the BFO and PSO algorithms will be applied to determine the optimum values of these parameters through different approximations of $s^{\pm\gamma}$ that are encountered in (3.2), where $\gamma = \{\rho, \mu\}$, $0 < \gamma < 1$. It is necessary to establish the main objective function of the optimization algorithms. Notice that there are multiple standard objective functions that could be engaged for tuning the best parameters for the $PI^{\rho}D^{\mu}$-controller, like the Integral Absolute Error (IAE), Integral Time Square Error (ITSE), Integral Time-Absolute Error (ITAE) and Integral Square Error (ISE). For instance, minimizing any objective function value is the key goal of the chosen optimization algorithm with the aim of accomplishing the best values of the $PI^{\rho}D^{\mu}$-controller. The motivation of using the PSO and BFO algorithms comes back to their simple concepts, easy

implementations, powerful abilities to control parameters and computational efficiencies when compared with other mathematical algorithms and other heuristic optimization techniques [29]. Nevertheless, to get a full overview about the BFO and PSO algorithms, one may consult [27, 30–34]. The design procedure of the $PI^{\rho}D^{\mu}$-controller through the BFO and the PSO algorithms is described by the block diagram shown in Figure 2.

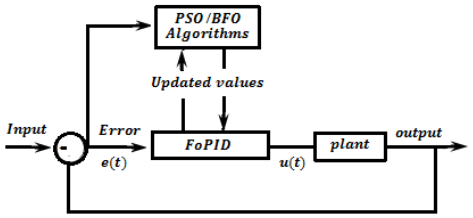

Figure 2. Block diagram of BFO/PSO running to tune the $PI^{\rho}D^{\mu}$-controller.

To obtain the best parameters of the $PI^{\rho}D^{\mu}$-controller, an objective function is established based on four terms to be minimized: the rise time, steady-state error, peak overshoot and the settling time of the controlled system. It takes the following form [32, 35]:

$$J = (1 - e^{-\beta})(M_p + e_{ss}) + e^{-\beta}(T_s - T_r), \tag{3.3}$$

where e_{ss} is the steady state error, T_s is the settling time, T_r is the rise time, M_p is the peak overshoot and β is a scaling factor. It is relevant to note that although the scaling factor β is typically chosen by a designer, it can definitely identify the roles of the four aforementioned items in the basic objective function [32, 35]. In this work, this factor has been selected to be 0.5 for the same reasons as reported in [32].

4. Rational approximations of fractional-order Laplacian operators

This part briefly introduces four approximations formulated as finite-order rational transfer functions of the fractional-order integro-differential Laplacian operators, $s^{\pm\gamma}$, where $0 < \gamma < 1$. The need to employ such approximations promptly emerges after determining the best values of the $PI^{\rho}D^{\mu}$-controller by applying some optimization algorithms. Those four approximations permit one to design the target system without the need to treat some hard time-domain compositions [13, 14].

4.1. El-Khazali Approximations

El-Khazali proposed two practical approximations of 1^{st}- and 2^{nd}-orders, respectively. The 1^{st}-order approximation is represented by a rational transfer function to replace s^{γ}, where $0 < \gamma < 1$, and given by [22]:

$$s^{\gamma} = \frac{N(s,\gamma)}{D(s,\gamma)} \cong \frac{\frac{s\tau}{\omega_{cn}} + 1}{\frac{s}{\omega_{cn}} + \tau}, \tag{4.1}$$

where ω_{cn} is a corner frequency, and

$$\tau = \tan\frac{\gamma\pi}{2} + \sec\frac{\gamma\pi}{2}. \tag{4.2}$$

It is worth mentioning that the fractional-order integral Laplacian operator $s^{-\gamma}$ could be simply found from the reciprocal of (4.1) [22]. MATLAB Code C1, to calculate (4.1), is given in Appendix A for completeness.

In [23], a 2^{nd}-order approximation was proposed for setting up a finite-order rational transfer function of the operators s^{γ}, where $0 < \gamma < 1$. The reciprocal of such approximation yields $s^{-\gamma}$. It provides an exact phase response at its center frequency and can be represented by the following modular structure [23]:

$$\Big(\frac{s}{\omega_g}\Big)^{\gamma} = \prod_{i=1}^{n} F_i(s/\omega_i) = \prod_{i=1}^{n} \frac{N_i\Big(\frac{s}{(\omega_i/\omega_g)}\Big)}{D_i\Big(\frac{s}{(\omega_i/\omega_g)}\Big)}, \tag{4.3}$$

where $\omega_i, i = 1, 2, ..., n$, is the center frequency of each biquadratic module and $\omega_g = \sqrt[n]{\prod_{i=1}^{n} \omega_i}$ is their geometric mean. If the leading center frequency ω_1 of the beginning section is determined, then to detect the constant phase element, the next center frequencies of every section could be evaluated from the following recursive equation [19]:

$$\omega_i = \omega_x^{2(i-1)}\omega_1, \quad i = 2, 3, ..., n, \tag{4.4}$$

where ω_x is the maximum real solution of the following equation:

$$a_0a_2\eta\lambda^4 + a_1(a_2 - a_0)\lambda^3 + (a_1^2 - a_2^2 - a_0^2)\eta\lambda^2 + a_1(a_2 - a_0)\lambda + a_0a_2\eta = 0, \tag{4.5}$$

where $\eta = \tan\frac{\gamma\pi}{4}$. Every biquadratic component in (4.3) can be determined by

$$\Big(\frac{s}{\omega_g}\Big)^{\gamma} = F_i\Big(\frac{s}{\omega_i}\Big) = \frac{N_i(\frac{s}{\omega_i})}{D_i(\frac{s}{\omega_i})} \cong \frac{a_0(\frac{s}{\omega_i})^2 + a_1(\frac{s}{\omega_i}) + a_2}{a_2(\frac{s}{\omega_i})^2 + a_1(\frac{s}{\omega_i}) + a_0}, \tag{4.6}$$

where $i = 1, 2, 3, ...,$ and

$$\begin{cases} a_0 &= \gamma^{\gamma} + 2\gamma + 1 \\ a_2 &= \gamma^{\gamma} - 2\gamma + 1 \\ a_1 &= (a_2 - a_0)tan\Big(\frac{(2+\gamma)\pi}{4}\Big) = -6\gamma\ tan\Big(\frac{(2+\gamma)\pi}{4}\Big). \end{cases} \tag{4.7}$$

Note that (4.6) is the exclusive approximation that leads to $F_i(\frac{s}{\omega_i}) = (\frac{s}{\omega_i})$ as $\gamma \to 1$. In other words,

$$F_i\Big(\frac{s}{\omega_i}\Big) = \frac{a_0(\frac{s}{\omega_i})^2 + a_0(\frac{s}{\omega_i})}{a_0(\frac{s}{\omega_i}) + a_0} = \Big(\frac{s}{\omega_i}\Big). \tag{4.8}$$

Similarly, Code C2 in Appendix A simulates the approximation given by (4.6).

4.2. Oustaloup's approximation

This approximation is the most widespread one among several approximations that are employed to generate approximate finite-order rational transfer functions for the operators $s^{\pm\gamma}$, where $0 < \gamma < 1$. The bandwidth could be defined to offer a proper adaptation to such operators by predetermining the

frequency band. Typically, in order to obtain a finite-order approximation of s^{γ} over the frequency range (ω_b, ω_h), the following rational transfer function is implemented [36]:

$$s^{\gamma} \cong K \prod_{k=-N}^{N} \frac{s + \omega_k^{'}}{s + \omega_k}, \tag{4.9}$$

where the gain, zeros and poles can be calculated from the following states:

$$\omega_k = \omega_b \Big(\frac{\omega_h}{\omega_b}\Big)^{\frac{k+N+0.5(1+\gamma)}{2N+1}} \tag{4.10a}$$

$$\omega_k^{'} = \omega_b \Big(\frac{\omega_h}{\omega_b}\Big)^{\frac{k+N+0.5(1-\gamma)}{2N+1}} \tag{4.10b}$$

$$K = \Big(\frac{\omega_h}{\omega_b}\Big)^{\frac{-\gamma}{2}} \prod_{k=-N}^{N} \frac{\omega_k}{\omega_k^{'}}. \tag{4.10c}$$

It should be pointed out that the following equation allows one to compute the unity-gain geometric frequency ω_u [36]:

$$\omega_u = \sqrt{\omega_b \cdot \omega_h}. \tag{4.11}$$

Having the form of the transfer function given in (4.9) in mind, one could observe that such a function will be always of odd order ($n = 2N + 1$). Anyhow, Code C3, given in Appendix A, illustrates suitable MATLAB code for calculating a finite-order rational transfer function corresponding to $s^{\pm\gamma}$ using Oustaloup's approximation, where $0 < \gamma < 1$.

4.3. The CFE approximation

This scheme is regarded as the principal mathematical method for offering approximate rational transfer functions for $s^{\pm\gamma}$, where $0 < \gamma < 1$. This approximation was set as follows [21]:

$$(1+z)^{\gamma} = \cfrac{1}{1 - \cfrac{\gamma z}{1 + \cfrac{(1+\gamma)z}{2 + \cfrac{(1-\gamma)z}{3 - \cfrac{(2+\gamma)z}{2 + \cfrac{(2-\gamma)z}{5 + \cfrac{\ldots+(n+\gamma)z}{2 + \cfrac{(n-\gamma)z}{2n+1+\ldots}}}}}}}}, \tag{4.12}$$

where $0 < \gamma < 1$, and $n \in \mathbb{N}$. For the sake of simplification, (4.12) can be rewritten in the following form [21]:

$$(1+z)^{\gamma} = \frac{1}{1-}\,\frac{\gamma z}{1+}\,\frac{(1+\gamma)z}{2+}\,\frac{(1-\gamma)z}{3+}\,\frac{(2+\gamma)z}{2+}\,\frac{(2-\gamma)z}{5+}\,\frac{(n+\gamma)z}{2+}\,\frac{(n-\gamma)z}{2n+1+}. \tag{4.13}$$

In order to find a proper finite-order approximation for s^{γ} using (4.12), one may replace $z = (s^{\gamma} - 1)$ in (4.12) or (4.13). This replacement step permits the n^{th}-order approximation of such operator to be found, about the center frequency $\omega_0 = 1\ rad/sec$, in the following form [26]:

$$s^{\gamma} \cong \frac{\gamma_0 s^n + \gamma_1 s^{n-1} + \ldots + \gamma_{n-1} s + \gamma_n}{\gamma_n s^n + \gamma_{n-1} s^{n-1} + \ldots + \gamma_1 s + \gamma_0}, \tag{4.14}$$

where $\gamma_i \in (0,\ 1)$, which can be found in [37] for $i = 0, 1, ..., 5$. Two more observations should be made at the end of this part. The first one is associated with the operator $s^{-\gamma}$, which could be easily achieved by inverting upside down the formula given in (4.14). The second one is related to computing the finite-order rational transfer function corresponding to $s^{\pm\gamma}$ using the CFE approximation, which can be illustrated by tracking the MATLAB Code C4 given in Appendix A.

5. Optimum design for VCAV System

It is common knowledge in the biomedical engineering field that the 1^{st}-order lumped parameter model is the most streamlined model for the mechanics of the breathing system. In this model, the pressure in the upper airways is typically assumed to equal zero throughout spontaneous ventilation. The diaphragm, if it is active, may add subatmospheric pressure [38, 39]. In general, there are commonly two actuators within many breathing systems of ventilators employed in ICU or anesthesia. These actuators are responsible for exhaling and inhaling processes. Since the AV system is just one of those breathing systems, its major components consist of these two actuators. In many circumstances, the exhaling actuator is actuated by a Positive End Expiratory Pressure (PEEP) valve, whereas the inhalator actuator, on the other hand, can be actuated by a piston drive mechanism. Also, the inhalator actuator system is linked with piston driven and lung mechanisms [38, 39]. The overall system could be modeled by a 3^{rd}-order differential equation by taking into account the linear friction and the electrical time constant of the motor (see [38, 39]). Through such system, the two input variables T_l and V_a, which respectively indicate the load torque and the applied motor voltage, are fed back to the subsystem of the lung mechanics. In particular, in view of possible estimations for all required parameters of the VCAV system, an open-loop transfer function of this system has been recently reported in [38] as follows:

$$G(s) = \frac{Y(s)}{V_i(s)} = \frac{14.471}{s^3 + 76.43s^2 + 109.76s + 0.129}, \tag{5.1}$$

where $Y(s) = \mathscr{L}\{y(t)\}$ and $V_i(s) = \mathscr{L}\{v_i(t)\}$ are respectively the Laplace transforms of the output variable, $y(t)$, which indicates the piston position, and the input variable of the system, $v_i(t)$, which indicates the voltage source of the motor.

The key target of this work is to enhance the dynamic behavior of the VCAV system by considering the transfer function given in (5.1). However, to show the efficiency of all feedback tuning schemes of $PI^{\rho}D^{\mu}$-controllers, the objective function given in (3.3) is minimized by running the BFO and PSO algorithms, which take into account the four approximations of $s^{\pm\gamma}$ in Section 3. The maximum value of iterations and the population size in both algorithms have been taken as 100 and 20, respectively. The optimum parameters of the $PI^{\rho}D^{\mu}$-controllers for $\beta = 0.5$ and $\beta = 0.9$, which are used to generate the corresponding approximation schemes, are given in Tables 1 and 2, respectively.

Table 1. Tuning results of the $PI^{\rho}D^{\mu}$-controllers using the PSO and the BFO algorithms for $\beta = 0.5$.

Algorithm	Type of approximation	κ_p	κ_i	κ_d	ρ	μ
PSO	1^{st}-order El-Khazali	31.4119	0.54000	25.4162	0.31300	0.90600
	2^{nd}-order El-Khazali	41.0000	15.4404	45.0000	0.09000	0.91100
	Oustaloup	59.0000	0.76000	61.0000	0.82100	0.92503
	CFE	0.16000	45.5900	51.0000	0.17700	0.86700
BFO	1^{st}-order El-Khazali	2.71690	2.50680	7.71930	0.91510	0.57810
	2^{nd}-order El-Khazali	20.8839	12.9392	21.7497	0.86760	0.90610
	Oustaloup	25.4065	18.5733	32.2031	0.54090	0.15850
	CFE	2.56650	10.0268	15.2471	0.18520	0.33260

Table 2. Tuning results of the $PI^{\rho}D^{\mu}$-controllers using the PSO and the BFO algorithms for $\beta = 0.9$.

Algorithm	Type of approximation	κ_p	κ_i	κ_d	ρ	μ
PSO	1^{st}-order El-Khazali	31.6262	0.5400	25.5702	0.9030	0.9060
	2^{nd}-order El-Khazali	41.000	0.3200	45.000	0.7990	0.8557
	Oustaloup	59.000	24.5393	61.0000	0.1650	0.9760
	CFE	41.5691	0.3100	51.000	0.1770	0.8670
BFO	1^{st}-order El-Khazali	4.5230	4.3728	6.7855	0.8750	0.2870
	2^{nd}-order El-Khazali	13.7804	2.0452	16.5894	0.2662	0.9468
	Oustaloup	17.5025	11.2505	21.6484	0.8129	0.2091
	CFE	2.2440	2.1768	8.1292	0.9151	0.5781

Remark 1. *The scaling factor β is chosen to define a trade-off between the overshoot and the steady-state error ($M_p + e_{ss}$) and the time difference between the settling and the rise times ($T_s - T_r$). For example, for $\beta > 0.7$ more emphasis will be given to ($M_p + e_{ss}$) than ($T_s - T_r$). The impact of β is highlighted by several numerical results of the proposed system model for $\beta = 0.5$ and $\beta = 0.9$, reported below for completeness.*

Clearly, Figures 3 and 4 show the performances of the closed-loop controlled system using both the PSO and BFO optimization algorithms by utilizing the four different approximations of the fractional-order operator repeated many times, coupled with assuming $\beta = 0.5$. In addition, we also plot in Figures 5 and 6 the performances of the closed-loop controlled system using both algorithms by assuming $\beta = 0.9$.

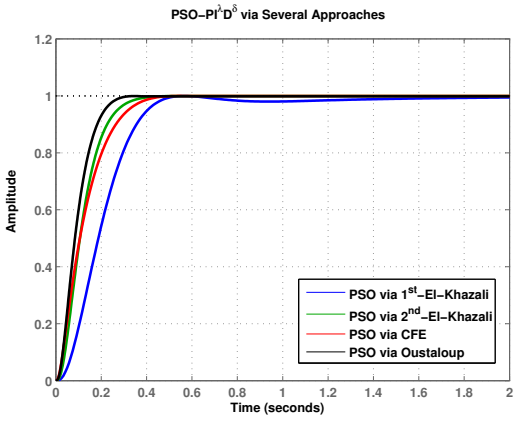

Figure 3. Step responses of the controlled system using the PSO algorithm for $\beta = 0.5$.

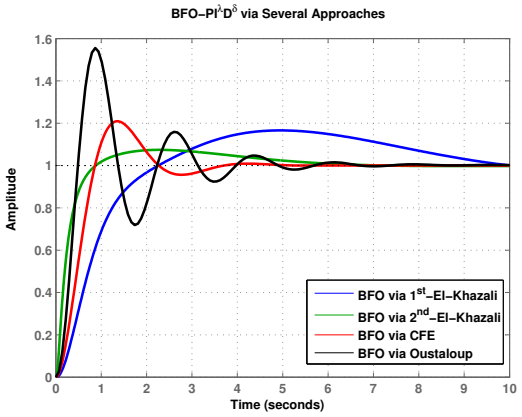

Figure 4. Step responses of the controlled system using the BFO algorithm for $\beta = 0.5$.

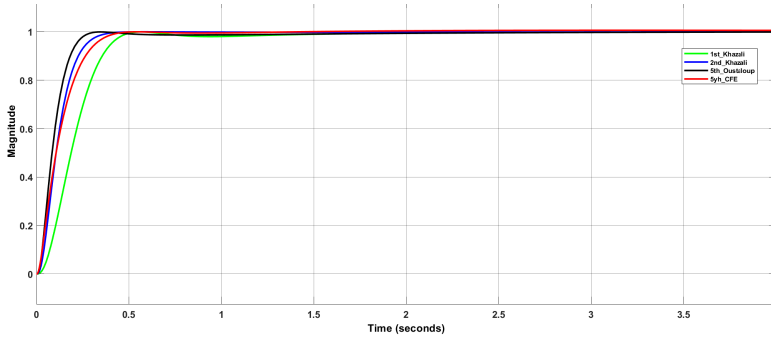

Figure 5. Step responses of the controlled system using the PSO algorithm for $\beta = 0.9$.

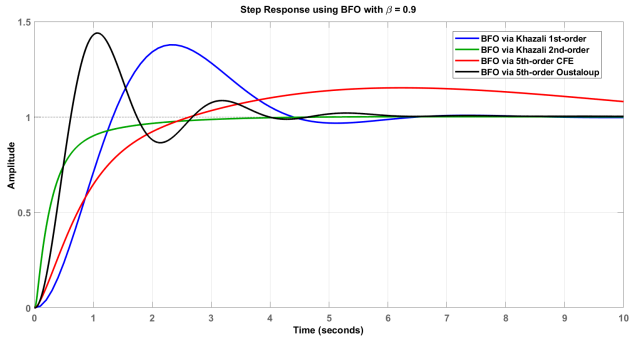

Figure 6. Step responses of the controlled system using the BFO algorithm for $\beta = 0.9$.

In both optimization techniques, the size of the controllers when using El-Khazali approximation is much less than that of the other two ones, as depicted in Tables 3 and 4. Moreover, there is a significant improvement in the step responses when using the PSO algorithm over that of the BFO algorithm. More precisely, one could notice that the minimum overshoot of the closed-loop VCAV system has been achieved when the $PI^{\rho}D^{\mu}$-controller was applying the PSO algorithm.

Table 3. The transfer functions of the $PI^{\rho}D^{\mu}$-controllers for $\beta = 0.5$.

Algorithm	Type of approximation	Controller transfer function, $C(s)$
PSO	1^{st}-order El-Khazali approximation	$C_{\{1^{st}-Kh\}}(s) = \frac{626.9s^2+1135s+462.3}{1.67s^2+23.58s+13.52}$
	2^{nd}-order El-Khazali approximation	$C_{\{2^{nd}-Kh\}}(s) = \frac{344.4s^4+1705s^3+2852s^2+1810s+371}{0.1917s^4+8.814s^3+28.9s^2+25.83s+6.079}$
	Oustaloup approximation	$C_{Ous}(s) = \frac{\begin{array}{c}1.92e05s^{10}+8.374e06s^9+1.191e08s^8+6.29e08s^7+1.468e09s^6+1.569e09s^5\\ +7.846e08s^4+1.793e08s^3+1.675e07s^2+6.184e05s+6599\end{array}}{\begin{array}{c}43.85s^{10}+5837s^9+1.847e05s^8+2.14e06s^7+9.029e06s^6+1.54e07s^5\\ +9.936e06s^4+2.592e06s^3+2.463e05s^2+8567s+70.81\end{array}}$
	The CFE approximation	$C_{CFE}(s) = \frac{\begin{array}{c}1.637e04s^{10}+5.268e05s^9+5.684e06s^8+2.671e07s^7+6.385e07s^6+8.343e07s^5\\ +6.143e07s^4+2.489e07s^3+5.182e06s^2+4.744e05s+1.467e04\end{array}}{\begin{array}{c}2.254s^{10}+543.4s^9+1.462e04s^8+1.312e05s^7+5.014e05s^6+9.102e05s^5\\ +8.165e05s^4+3.595e05s^3+7.334e04s^2+6038s+142\end{array}}$
BFO	1^{st}-order El-Khazali approximation	$C_{\{1^{st}-Kh\}}(s) = \frac{379.1s^2+303.8s+124.7}{14.97s^2+44.52s+2.907}$
	2^{nd}-order El-Khazali approximation	$C_{\{2^{nd}-Kh\}}(s) = \frac{301.3s^4+1018s^3+1314s^2+772.4s+186.4}{0.3702s^4+15.65s^3+31.49s^2+16.57s+0.555}$
	Oustaloup approximation	$C_{Ous}(s) = \frac{\begin{array}{c}1132s^{10}+8.422e04s^9+1.968e06s^8+1.638e07s^7+5.564e07s^6+7.554e07s^5\\ +4.426e07s^4+1.068e07s^3+1.106e06s^2+4.292e04s+550.1\end{array}}{\begin{array}{c}12.07s^{10}+1008s^9+2.53e04s^8+2.234e05s^7+7.698e05s^6+1.02e06s^5\\ +5.413e05s^4+1.104e05s^3+8797s^2+246.4s+2.075\end{array}}$
	The CFE approximation	$C_{CFE}(s) = \frac{\begin{array}{c}184.2s^{10}+7443s^9+1.039e05s^8+5.885e05s^7+1.588e06s^6+2.15e06s^5\\ +1.499e06s^4+5.243e05s^3+8.724e04s^2+5874s+137.9\end{array}}{\begin{array}{c}2.342s^{10}+141.1s^9+2552s^8+1.726e04s^7+5.29e04s^6+7.875e04s^5\\ +5.864e04s^4+2.135e04s^3+3575s^2+232.4s+4.71\end{array}}$

Table 4. The transfer functions of the $PI^{\rho}D^{\mu}$-controllers for $\beta = 0.9$.

Algorithm	Type of approximation	Controller transfer function, $C(s)$
PSO	1^{st}-order El-Khazali approximation	$C_{\{1^{st}-Kh\}}(s) = \frac{4944s^2+6329s+548.8}{13.1s^2+178.1s+13.52}$
	2^{nd}-order El-Khazali approximation	$C_{\{2^{nd}-Kh\}}(s) = \frac{577.3s^4+2012s^3+2207s^2+778.2s+40.67}{0.5624s^4+15.49s^3+31.31s^2+16.83s+0.8531}$
	Oustaloup approximation	$C_{Ous}(s) = \frac{\begin{array}{c}1.529e06s^{10}+7.401e08s^9+9.569e10s^8+1.819e12s^7+1.058e13s^6+1.81e13s^5\\ +1.14e13s^4+1.881e12s^3+8.537e10s^2+5.842e08s+1.088e06\end{array}}{\begin{array}{c}3.126s^{10}+3.135e04s^9+1.393e07s^8+1.702e09s^7+2.953e10s^6+1.378e11s^5\\ +1.149e11s^4+1.94e10s^3+7.952e08s^2+4.859e06s+8017\end{array}}$
	The CFE approximation	$C_{CFE}(s) = \frac{\begin{array}{c}9.315e04s^{10}+2.947e06s^9+3.076e07s^8+1.401e08s^7+3.212e08s^6+3.971e08s^5\\ +2.709e08s^4+9.988e07s^3+1.848e07s^2+1.468e06s+3.426e04\end{array}}{\begin{array}{c}2.254s^{10}+2084s^9+6.151e04s^8+5.866e05s^7+2.343e06s^6+4.418e06s^5\\ +4.103e06s^4+1.87e06s^3+3.946e05s^2+3.376e04s+809.4\end{array}}$
BFO	1^{st}-order El-Khazali approximation	$C_{\{1^{st}-Kh\}}(s) = \frac{134.8s^2+187.8s+66.78}{10.15s^2+13.45s+1.226}$
	2^{nd}-order El-Khazali approximation	$C_{\{2^{nd}-Kh\}}(s) = \frac{144.4s^4+613.8s^3+875.3s^2+475.1s+80.65}{0.1251s^4+9.486s^3+29.32s^2+24.11s+4.499}$
	Oustaloup approximation	$C_{Ous}(s) = \frac{\begin{array}{c}4.56e04s^{10}+6.061e07s^9+6.876e09s^8+1.95e11s^7+1.091e12s^6+1.394e12s^5\\ +6.238e11s^4+7.936e10s^3+2.854e09s^2+2.559e07s+2.134e04\end{array}}{\begin{array}{c}274.6s^{10}+6.801e05s^9+9.577e07s^8+3.703e09s^7+2.237e10s^6+3.132e10s^5\\ +9.348e09s^4+4.765e08s^3+6.737e06s^2+1.324e04s+6.861\end{array}}$
	The CFE approximation	$C_{CFE}(s) = \frac{\begin{array}{c}3.669e04s^{10}+9.368e05s^9+8.288e06s^8+3.18e07s^7+6.075e07s^6+6.215e07s^5\\ +3.635e07s^4+1.26e07s^3+2.545e06s^2+2.556e05s+9712\end{array}}{\begin{array}{c}262.6s^{10}+2.016e04s^9+3.028e05s^8+1.752e06s^7+4.556e06s^6+5.762e06s^5\\ +3.585e06s^4+1.068e06s^3+1.365e05s^2+6128s+16.91\end{array}}$

Tables 5 and 6, however, list the performance results of the unit-step responses of the controlled system using the four approximations for $\beta = 0.5$ and $\beta = 0.9$, respectively. It can be seen from Table 5 that the three approximations, i.e., Oustaloup, CFE, and the 2^{nd}-order El-Khazali approximations, gave

the same performance. However, the 2^{nd}-order El-Khazali approximation yields a 4^{th}-order $PI^{\rho}D^{\mu}$, while the other two yield 10^{th}-order controllers. This is a significant reduction in the size of controller. A closer look at the rise time, settling time and the maximum overshoot justifies the use of the El-Khazali 2^{nd}-order approximation. Similarly, when using the BFO optimization algorithm, the 2^{nd}-order El-Khazali approximation was superior to the other three approximations. Therefore, both remarks satisfy the use of the biquadratic approximation described by Eqs (4.3) to (4.8).

Table 5. Performance of the controlled system step response using the PSO and the BFO algorithms for $\beta = 0.5$.

Step Response	$H^{PSO}_{\{1^{st}-Kh\}}$	$H^{PSO}_{\{2^{nd}-Kh\}}$	H^{PSO}_{CFE}	H^{PSO}_{Ous}	$H^{BFO}_{\{1^{st}-Kh\}}$	$H^{BFO}_{\{2^{nd}-Kh\}}$	H^{BFO}_{CFE}	H^{BFO}_{Ous}
Rise Time	0.2829	0.1858	0.2333	0.1516	1.3512	0.4828	0.5938	0.3225
Settling Time	0.4526	0.3260	0.3783	0.2514	14.6911	5.2044	3.3122	4.7305
Settling Min.	0.9001	0.9010	0.9012	0.9008	0.9017	0.9000	0.9079	0.7189
Settling Max.	0.9997	0.9998	0.9999	0.9994	1.1660	1.0741	1.2092	1.5552
Overshoot	5.2705e-04	0.0000	0.0000	0.0000	16.6236	7.4173	20.9616	55.5270
Peak	0.9997	0.9998	0.9999	0.9994	1.1660	1.0741	1.2092	1.5552
Peak Time	0.5622	0.5414	0.5430	0.3460	4.9523	2.2783	1.3415	0.8683

Table 6. Performance of the controlled system step response using the PSO and the BFO algorithms for $\beta = 0.9$.

Step Response	$H^{PSO}_{\{1^{st}-Kh\}}$	$H^{PSO}_{\{2^{nd}-Kh\}}$	H^{PSO}_{CFE}	H^{PSO}_{Ous}	$H^{BFO}_{\{1^{st}-Kh\}}$	$H^{BFO}_{\{2^{nd}-Kh\}}$	H^{BFO}_{CFE}	H^{BFO}_{Ous}
Rise Time	0.2817	0.1854	0.2200	0.1387	0.9100	0.9252	1.6685	0.4199
Settling Time	0.4506	0.3016	0.3499	0.2433	5.4000	2.5069	18.9348	3.9376
Settling Min.	0.9018	0.9005	0.9022	0.9014	0.9075	0.8997	0.9006	0.8633
Settling Max.	0.9998	0.9998	0.9996	0.9997	1.2930	0.9982	1.1533	1.4392
Overshoot	2.4138e-004	0.0000	0.0000	0.0000	29.3215	0.0000	15.3356	43.9248
Peak	0.9998	0.9998	0.9996	0.9997	1.2930	0.9982	1.1533	1.4392
Peak Time	0.5620	0.3872	0.4728	0.3791	2.3246	4.3336	6.2231	1.1085

Figures 7 and 8 shows the control signals of the controlled system using the PSO and the BFO optimization algorithms for $\beta = 0.5$, while Figures 9 and 10 show the control signals for $\beta = 0.9$. In addition, Table 7 shows the values of the objective function (3.3) using the two optimization algorithms for $\beta = 0.5$ and $\beta = 0.9$. One concludes that the PSO algorithm yields a better result than that of the BFO algorithm. Even though the initial magnitude of the control signal of the PSO algorithm is larger than that of the BFO algorithm, the value of the objective function J described by (3.3) is smaller when using the PSO algorithm. This is also verified by the step responses of the controlled systems, as shown by Figures 5 and 6. It is clear that the 2^{nd}-order El-Khazali approximation yields a smaller controller signal than the other approximations and provides a competitive steady-state error compared to its other counterparts in both algorithms and for both $\beta = 0.5$ and $\beta = 0.9$.

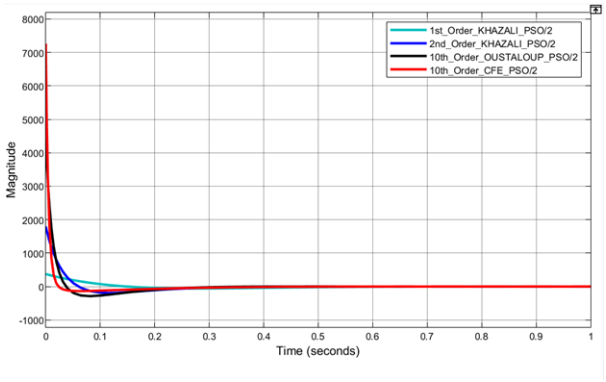

Figure 7. $PI^{\rho}D^{\mu}$-control signal using PSO algorithm for $\beta = 0.5$.

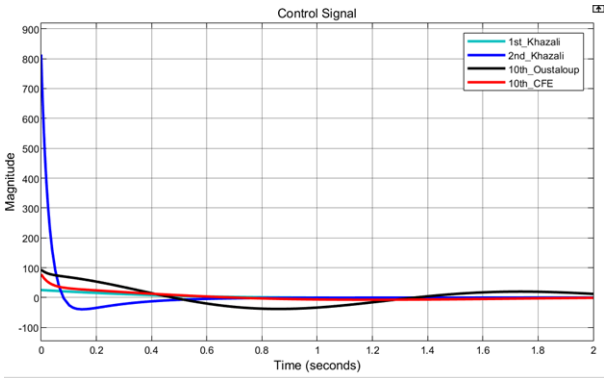

Figure 8. $PI^{\rho}D^{\mu}$-control signal using BFO algorithm for $\beta = 0.5$.

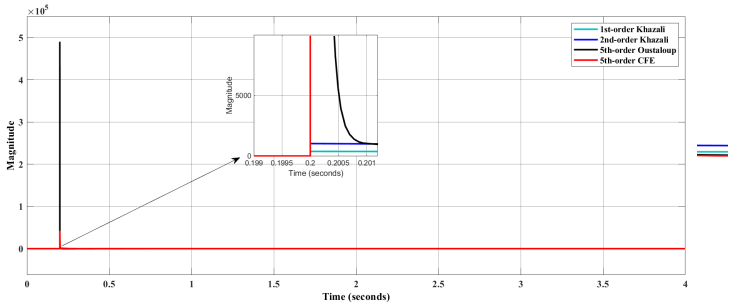

Figure 9. $PI^{\rho}D^{\mu}$-control signal using PSO algorithm for $\beta = 0.9$.

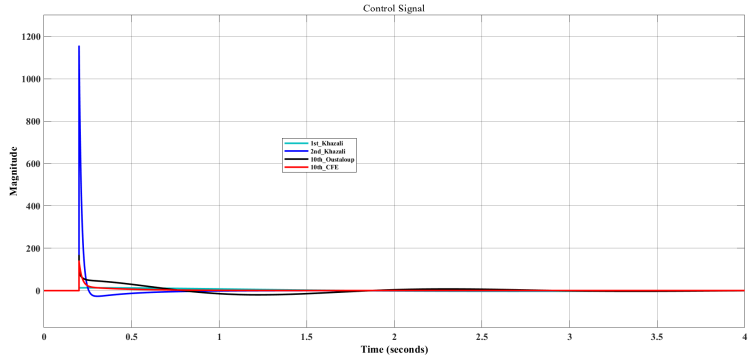

Figure 10. $PI^{\rho}D^{\mu}$-control signal using BFO algorithm for $\beta = 0.9$.

Table 7. The value of the objective function (3.3) using PSO and BFO algorithms for $\beta = 0.5$ and $\beta = 0.9$.

Algorithm	Type of approximation	When $\beta = 0.5$	When $\beta = 0.9$
PSO	1^{st}-order El-Khazali approximation	0.2190	0.2199
	2^{nd}-order El-Khazali approximation	0.1247	0.1662
	Oustaloup approximation	0.1131	0.0949
	The CFE approximation	0.1336	0.1732
BFO	1^{st}-order El-Khazali approximation	0.8197	1.403
	2^{nd}-order El-Khazali approximation	0.3074	0.4393
	Oustaloup approximation	0.7388	0.8126
	The CFE approximation	0.6341	0.9395

6. Conclusions

A new $PI^{\rho}D^{\mu}$-controller is developed to control and improve the behavior of the Volume-Controlled Artificial Ventilation (VCAV) system. Two optimization algorithms, Bacteria Foraging Optimization (BFO) and Particle Swarm Optimization (PSO), are successfully used in conjunction with the use of four different approximations of the fractional-order integro-differential Laplacian operators, $s^{\pm\gamma}$, where $0 < \gamma < 1$. These approximations are the 1^{st}- and the 2^{nd}-order El-Khazali approximations, Oustaloup's approximation and the Continued Fractional Expansion (CFE) approximation. The PSO algorithm yields a better performance compared to that of the BFO algorithm, especially when using the 2^{nd}-order El-Khazali approximations. It provided a significant improvement in terms of a smaller controller size, i.e., a 4^{th}-order one, and in terms of the overall performance of the controlled system step response.

Acknowledgments

This work is supported by Ajman University grant 2020-COVID-19-08.

Conflict of interest

The authors declare no conflict of interest.

References

1. E. Fan, J. R. Beitler, L. Brochard, C. S. Calfee, N. D. Ferguson, A. S. Slutsky, et al., COVID-19-associated acute respiratory distress syndrome: is a different approach to management warranted? *Lancet Resp. Med.*, **8** (2020), 816–821. https://doi.org/10.1016/S2213-2600(20)30304-0
2. H. D. Poor, C. E. Ventetuolo, T. Tolbert, G. Chun, G. Serrao, A. Zeidman, et al., COVID-19 critical illness pathophysiology driven by diffuse pulmonary thrombi and pulmonary endothelial dysfunction responsive to thrombolysis, *Lancet Resp. Med.*, **10** (2020), e44. https://doi.org/10.1002/ctm2.44
3. J. Guay, E. A. Ochroch, S. Kopp, Intraoperative use of low volume ventilation to decrease postoperative mortality, mechanical ventilation, lengths of stay and lung injury in adults without acute lung injury, *Cochrane Database Syst. Rev.*, **2018** (2018), CD011151. https://doi.org/10.1002/14651858.CD011151.pub3
4. T. Yoshida, S. Nakahashi, M. A. M. Nakamura, Y. Koyama, R. Roldan, V. Torsani, et al., Volume-controlled ventilation does not prevent injurious inflation during spontaneous effort, *Am. J. Resp. Cri. Care*, **196** (2017), 590–601. https://doi.org/10.1164/rccm.201610-1972OC
5. Y. Shi, B. Zhang, M. Cai, X. D. Zhang, Numerical simulation of volume-controlled mechanical ventilated respiratory system with 2 different lungs, *Int. J. Numer. Meth. Bio.*, **33** (2017), e2852. https://doi.org/10.1002/cnm.2852
6. S. S. Vidhya, K. Keerthana, M. Janaki, J. Kanimozhi, A survey of control algorithms used in physiological closed loop control for oxygen therapy, *Int. J. Appl. Eng. Res.*, **14** (2019), 694–702.
7. C. Kirakli, I. Naz, O. Ediboglu, D. Tatar, A. Budak, E. Tellioglu, A randomized controlled trial comparing the ventilation duration between adaptive support ventilation and pressure assist/control ventilation in medical patients in the ICU, *Chest*, **147** (2015), 1503–1509. https://doi.org/10.1378/chest.14-2599
8. S. Mesic, R. Babuska, H. C. Hoogsteden, A. F. Verbraak, Computer-controlled mechanical simulation of the artificially ventilated human respiratory system, *IEEE T. Biomed. Eng.*, **50** (2003), 731–743. https://doi.org/10.1109/TBME.2003.812166
9. M. E. Shahri, S. Balochian, H. Balochian, Y. Zhang, Design of fractionalorder PID controllers for time delay systems using differential evolution, *Indian J. Sci. Technol.*, **7** (2014), 1307–1315. https://doi.org/10.17485/ijst/2014/v7i9.28
10. N. Nasirpour, S. Balochian, Optimal design of fractional-order PID controllers for multi-input multi-output (variable air volume) air-conditioning system using particle swarm optimization, *Intell. Build. Int.*, **9** (2017), 107–119. https://doi.org/10.1080/17508975.2016.1170659
11. J. Kennedy, R. Eberhart, Particle swarm optimization, In: *Proceedings of ICNN'95-international conference on neural networks*, 1995, 1942–1948. https://doi.org/10.1109/ICNN.1995.488968

12. K. M. Passino, Distributed optimization and control using only a germ of intelligence, In: *The 2000 IEEE international symposium on intelligent control. Held jointly with the 8th IEEE Mediterranean conference on control and automation (Cat. No.00CH37147)*, 2000, 5–13. https://doi.org/10.1109/ISIC.2000.882888

13. S. Momani, I. M. Batiha, R. El-Khazali, Design of $PI^{\lambda}D^{\delta}$-heart rate controllers for cardiac pacemaker, In: *2019 IEEE international symposium on signal processing and information technology (ISSPIT)*, 2019. https://doi.org/10.1109/ISSPIT47144.2019.9001785

14. I. M. Batiha, R. El-Khazali, S. Momani, Dynamic responses of PI^{λ}, PD^{δ}, and $PI^{\lambda}D^{\delta}$ controllers for prosthetic hand model using PSO algorithm, In: *2019 IEEE international symposium on signal processing and information technology (ISSPIT)*, 2019. https://doi.org/10.1109/ISSPIT47144.2019.9001835

15. I. M. Batiha, S. A. Njadat, R. M. Batyha, A. Zraiqat, A. Dababneh, S. Momani, Design fractional-order PID controllers for single-joint robot arm model, *Int. J. Advance Soft Compu. Appl.*, **14** (2022), 96–114. https://doi.org/10.15849/IJASCA.220720.07

16. S. Momani, I. M. Batiha, Tuning of the fractional-order PID controller for some real-life industrial processes using particle swarm optimization, *Prog. Fract. Differ. Appl.*, **8** (2022), 377–391. https://doi.org/10.18576/pfda/080303

17. R. El-Khazali, I. M. Batiha, S. Momani, The drug administration via fractional-order $PI^{\lambda}D^{\delta}$-controller, *Prog. Fract. Differ. Appl.*, **8** (2022), 53–62. https://doi.org/10.18576/pfda/080103

18. I. M. Batiha, J. Oudetallah, A. Ouannas, A. A. Al-Nana, I. H. Jebril, Tuning the fractional-order PID-controller for blood glucose level of diabetic patients, *Int. J. Advance Soft Compu. Appl.*, **13** (2021), 1–10.

19. A. K. Gil'mutdinov, P. A. Ushakov, R. El-Khazali, *Fractal elements and their applications*, Switzerland: Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-45249-4

20. R. El-Khazali, I. M. Batiha, S. Momani, Approximation of fractional-order operators, In: *Fractional calculus*, Singapore: Springer, 2019, 121–151. https://doi.org/10.1007/978-981-15-0430-3_8

21. B. T. Krishna, Studies on fractional order differentiators and integrators: A survey, *Signal Process.*, **91** (2011), 386–426. https://doi.org/10.1016/j.sigpro.2010.06.022

22. R. El-Khazali, Fractional-order $LC^{\alpha}L$ filter-based grid connected PV systems, In: *2019 IEEE 62nd international midwest symposium on circuits and systems (MWSCAS)*, 2019, 533–536. https://doi.org/10.1109/MWSCAS.2019.8885133

23. R. El-Khazali, On the biquadratic approximation of fractional-order Laplacian operators, *Analog Integr. Circ. Sig. Procss.*, **82** (2015), 503–517. https://doi.org/10.1007/s10470-014-0432-8

24. I. Podlubny, Fractional-order systems and $PI^{\lambda}D^{\mu}$-controllers, In: *IEEE transactions on automatic control*, 1999, 208–214. https://doi.org/10.1109/9.739144

25. I. M. Batiha, R. El-Khazali, A. AlSaedi, S. Momani, The general solution of singular fractional-order linear time-invariant continuous systems with regular pencils, *Entropy*, **20** (2018), 400. https://doi.org/10.3390/e20060400

26. R. El-Khazali, Fractional-order $PI^{\lambda}D^{\mu}$ controller design, *Comput. Math. Appl.*, **66** (2013), 639–646. https://doi.org/10.1016/j.camwa.2013.02.015

27. S. Momani, R. El-Khazali, I. M. Batiha, Tuning PID and $PI^{\lambda}D^{\delta}$ controllers using particle swarm optimization algorithm via El-Khazali's approach, *AIP Conf. Proc.*, **2172** (2019), 050003. https://doi.org/10.1063/1.5133522

28. I. Podlubny, L. Dorcak, I. Kostial, On fractional derivatives, fractional-order dynamic systems and $PI^{\lambda}D^{\mu}$ -controllers, In: *Proceedings of the 36th IEEE conference on decision and control*, 1997, 4985–4990. https://doi.org/10.1109/CDC.1997.649841

29. T. Hamadneh, I. Abu-Falahah, M. AlQudah, Numerical optimization and positivity certificates for polynomials and rationals over simplices, *J. Comput. Appl. Math.*, **414** (2022), 114430. https://doi.org/10.1016/j.cam.2022.114430

30. W. Lin, Z. Chongquan, Design of optimal fractional-order PID controllers using particle swarm optimization algorithm for DC motor system, In: *2015 IEEE advanced information technology, electronic and automation control conference (IAEAC)*, 2015, 175–179. https://doi.org/10.1109/IAEAC.2015.7428542

31. S. Das, A. Biswas, S. Dasgupta, A. Abraham, Bacterial foraging optimization algorithm: Theoretical foundations, analysis, and applications, In: *Foundations of computational intelligence*, Heidelberg: Springer, 2009, 23–55. https://doi.org/10.1007/978-3-642-01085-9_2

32. M. Nasri, H. Nezamabadi-pour, M. Maghfoori, A PSO-based optimum design of PID controller for a linear brushless DC motor, *Proceedings of World Academy of Science, Engineering and Technology*, **20** (2007), 211–215.

33. B. D. Halilu, E. C. Anene, E. E. Omigzegba, L. Maijama'a, S. A. Baraza, Optimization of PID controller gains for identified magnetic levitation plant using bacteria foraging algorithm, *Int. J. Eng. Modern Technol.*, **5** (2019), 12–18.

34. M. A. Muñoz, S. K. Halgamuge, W. Alfonso, E. F. Caicedo, Simplifying the bacteria foraging optimization algorithm, In: *IEEE congress on evolutionary computation*, 2010, 1–7. https://doi.org/10.1109/CEC.2010.5586025

35. K. E. Dagher, Design of an auto-tuning PID controller for systems based on slice genetic algorithm, *IJCCCE*, **13** (2013), 1–9.

36. A. Oustaloup, F. Levron, B. Mathieu, F. M. Nanot, Frequency band complex noninteger differentiator: characterization and synthesis, *IEEE T. Circuits Syst. I*, **47** (2000), 25–39. https://doi.org/10.1109/81.817385

37. I. Dimeas, *Design of an integrated fractional-order controller*, Patras: University of Patras, 2017.

38. A. T. Yimchunger, D. Acharya, D. K. Das, Particle swarm optimization based PID-controller design for volume control of artificial ventilation system, In: *2020 IEEE calcutta conference (CALCON)*, 2020, 278–282. https://doi.org/10.1109/CALCON49167.2020.9106480

39. M. Walter, S. Leonhardt, Control applications in artificial ventilation, In: *2007 Mediterranean conference on control & automation*, 2007, 1–6. https://doi.org/10.1109/MED.2007.4433762

Appendix A

Code C1: MATLAB code for calculating the 1^{st}-order El-Khazali approximation for $s^{\pm\gamma}$, where $0 < \gamma < 1$.

```
% This program is written to find the first-order El-Khazali's approximation for
%fractional-order integrators and differentiators.
clear all
gamma=input ('Enter the fractional-order value= ')
Wcn= input ('Enter the center frequency (in rad/s)=  ')
tau = tan(gamma*pi/2)+sec(gamma*pi/2);
N= [tau  Wcn]; D=[1  tau*Wcn];
G1_diff = tf(N,D)  % Differentiator
G1_Int = 1/G1_diff % Integrator
```

Code C2: MATLAB code for calculating the 2^{nd}-order El-Khazali approximation for $s^{\pm\gamma}$, where $0 < \gamma < 1$.

```
% This program is written to find the second-order El-Khazali's approximation for
% fractional-order integrators and differentiators.
clear all
gamma=input ('Enter the fractional-order value= ')
Wcn= input ('Enter the center frequency (in rad/s)=  ')
co=gamma^gamma+2*gamma+1;
c2=gamma^gamma-2*gamma+1;
c1=(co-c2)*tan((2-gamma)*pi/4);
N2=[co   c1*Wcn   c2*Wcn^2];D2 =[c2  c1*Wcn   co*Wcn^2 ];
G2_diff =tf(N2,D2)  % Differentiator
G2_int = 1/G2_diff  % Integrator
```

Code C3: MATLAB code for calculating the Oustaloup approximation for $s^{\pm\gamma}$, where $0 < \gamma < 1$.

```
% This program is written to find the Oustaloup's approximation for fractional-order
%integrators and differentiators.
clear all
close all
gamma=input ('Enter the fractional-order value= ')
N=input ('Enter the value of N= ')
w_L=1e-1;
w_H=1e1;
w_L=w_L*0.1;w_H=w_H*10;
mu=w_H/w_L;
w_u=sqrt(w_H*w_L);
alpha=(mu)^(gamma/(2*N+1));
eta=(mu)^((1-gamma)/(2*N+1));
k=-N:N;
w_kp=(mu).^((k+N+0.5-0.5*gamma)/(2*N+1))*w_L;
w_k=(mu).^((k+N+0.5+0.5*gamma)/(2*N+1))*w_L;
D_N_K=(w_u/w_H)^gamma*prod(w_k)/prod(w_kp);
D_N_P=-w_k;D_N_Z=-w_kp;
```

```
[num,den]=zp2tf(D_N_Z',D_N_P',D_N_K);
G3_diff=tf(num,den)   % Differentiator
G3_Int = 1/G3_diff    % Integrator
```

Code C4: MATLAB code for calculating the CFE approximation for $s^{\pm\gamma}$, where $0 < \gamma < 1$.

```
% This program is written to find the CFE approximation for fractional-order integrators
%and differentiators.
clear all
close all
gamma=input ('Enter the fractional-order value= ')
err=10;
w=logspace(-3,3,300);
% Continued Fraction Expansion 5th-order approximation
a5=(-(gamma^5)-15*(gamma^4)-85*(gamma^3)-225*(gamma^2)-274*gamma-120);
a4=(5*(gamma^5)+45*(gamma^4)+5*(gamma^3)-1005*(gamma^2)-3250*gamma-3000);
a3=(-10*(gamma^5)-30*(gamma^4)+410*(gamma^3)+1230*(gamma^2)-4000*gamma-12000);
a2=(10*(gamma^5)-30*(gamma^4)-410*(gamma^3)+1230*(gamma^2)+4000*gamma-12000);
a1=(-5*(gamma^5)+45*(gamma^4)-5*(gamma^3)-1005*(gamma^2)+3250*gamma-3000);
ao=((gamma^5)-15*(gamma^4)+85*(gamma^3)-225*(gamma^2)+274*gamma-120);
b5=((gamma^5)-15*(gamma^4)+85*(gamma^3)-225*(gamma^2)+274*gamma-120);
b4=(-5*(gamma^5)+45*(gamma^4)-5*(gamma^3)-1005*(gamma^2)+3250*gamma-3000);
b3=(10*(gamma^5)-30*(gamma^4)-410*(gamma^3)+1230*(gamma^2)+4000*gamma-12000);
b2=(-10*(gamma^5)-30*(gamma^4)+410*(gamma^3)+1230*(gamma^2)-4000*gamma-12000);
b1=(5*(gamma^5)+45*(gamma^4)+5*(gamma^3)-1005*(gamma^2)-3250*gamma-3000);
bo=(-(gamma^5)-15*(gamma^4)-85*(gamma^3)-225*(gamma^2)-274*gamma-120);
num_5=[a5/b5 a4/b5 a3/b5 a2/b5 a1/b5 ao/b5];
den_5=[1 b4/b5 b3/b5 b2/b5 b1/b5 bo/b5];
G4_diff=tf(num_5, den_5)  % Differentiator
G4_Int = 1/G4_diff        % Integrator
```

AIMS Press