Int. J. Advance Soft Compu. Appl, Vol. 13, No. 1, March 2021
Print ISSN 2710-1274, Online ISSN: 2074-8523

Optimal design of fractional-order PID controllers using bacterial foraging optimization algorithm

Radwan M. Batyha

Department of Computer Science, Faculty of Science and Technology
Irbid National University, Irbid, P.O.Box 2600 Jordan
e-mail:rbatiha@inu.edu.jo

Abstract

This paper introduces a new design method of Fractional-order PID (FoPID or $PI^{\lambda}D^{\delta}$) controller using Bacterial Foraging Optimization (BFO) algorithm via El-Khazali's biquadratic approximation. The integro-differential fractional-order Laplacian operators, $s^{(\pm\alpha)}$, of order α, for $0 < \alpha \leq 1$, is approximated using El-Khazali's approach by finite-order rational transfer functions. The significance of this approach lies in developing an algorithm that only depends on the fractional-order α, which allows one to reduce the number of the controller parameters to be tuned. To illustrate the influence and the efficiency of the proposed design method, the BFO-FoPID controller via El-Khazali's approach is carried out and employed on some systems and compared with that of the same using two well-known approximations for $s^{(\pm\alpha)}$; i.e., Carlson's and Oustaloup's approximations. The main results of this work are verified via numerical simulations.

Keywords: *Fractional calculus, Fractional-order PID controller, Bacterial foraging optimization algorithm, Laplacian operator, El-Khazali's approach.*

1 Introduction

Many physical systems are modeled by integer-order dynamics, which may not fully describe their behavior over wide spectrum. In reality, however, such systems could be accurately described by fractional-order dynamics [1, 2], where the total transient and steady-state system responses are fully achieved. Developing appropriate mathematical models of the system to be controlled is the first step in the control design process. Such models may be derived either from physical laws or experimental data [3, 4, 5]. The purpose of a controller is to enhance systems' behavior to meet desired performance specifications [6]. In feedback control, the plant can be monitored and its response can be measured using sensors and transducers [6]. Then, the controller compares the sensed signal with a preferred response as specified externally, and uses the error information to generate a suitable control signal. Generally, a high-quality control system should be: stable; fast; accurate; insensitive to noise, external disturbances, modeling errors and parameter variations; sufficiently sensitive to control inputs; and be free of undesirable coupling and dynamic interactions [6].

Proportional-Integral-Derivative PID controller is commonly used in industries and manufacturing applications. The advantage of using such controller lies in its design simplicity and good performance including low percentage overshoot and small settling time for slow industrial processes [7, 8]. This controller includes three parameters to choose; a proportional gain, K_p, an integral time constant, K_i, and a derivative time K_d. Observe that PID controller could not perform properly for higher order systems [9]. Especially in complex systems that mimics real systems [10, 11, 12]. The performance of systems when using PID controller can be further improved by using fractional-order Proportional-Integral-Derivative (FoPID or $PI^{\lambda}D^{\delta}$) controller [7, 13, 14, 15, 16, 17]. In FoPID-controller, there are two additional parameters; λ and δ, which increase the number of parameters to tune from three to five parameters; i.e., $\{K_p,\ K_i,\ K_d,\ \lambda,\ \delta\}$ [9, 18, 19, 20]. To obtain the best FoPID-controller, the optimum set of these parameters should be found [21]. Tuning five parameters of a FoPID-controller adds more flexibility to the design but with an increase complexity. A compromise should be made between adopting a straightforward design procedure that meets most of the design requirements, and the minimum number of controller parameters that needs to be optimized. Several optimization methods, such Genetic Algorithm (GA), Artificial Bee Colony (ABC) algorithm, Zeigler-Nichols (ZN) method, Nelder-Mead (NM) method, Particle Swarm Optimization (PSO) technique, Bacterial Foraging Optimization (BFO) algorithm and many others where successfully used to obtain the optimum set of such five parameters [22, 23, 24, 25, 26]. The main differences between such methods could be appeared in the implementation (i.e., the realization of the integral and the differential components of the controller).

In this work, the BFO algorithm is used to determine the optimum controller parameters. The BFO is a swarm intelligence algorithm that has been widely approved as a global optimization algorithm to meet some current interests associated with some distributed and control optimizations [27, 28]. This algorithm, which is inspired by the social foraging behavior of Escherichia coli, has already attracted the consideration of many scholars due to its capability in dealing with many real-life optimization problems arising in various application fields [27, 28]. More particularly, the BFO algorithm is implemented, here, to optimize the five parameters of the FoPID-controller. The realization of the integro-differential components of the controller is implemented by replacing the corresponding Laplacian operators by finite-order rational transfer functions. The objective of the BFO algorithm is to minimize systems' integral time square error (ITSE) and/or the integral square error (ISE) subject to time or frequency domain constraints, such as the maximum overshoot, rise time and settling time, or system's gain and phase margins. The role of these specifications relatively measures the robustness of the controlled system. This paper is organized as follows: Section 2 includes some necessary definitions and preliminaries related to the fractional calculus. Section 3 presents the concept of the FoPID-controller. Section 4 introduces El-Khazali's approach for the fractional-order Laplacian operator, $s^{(\pm\alpha)}$, followed by Section 5 that exhibits some numerical simulations, and finally, the conclusion of this work is summarized in the last section.

2 Fractional Calculus

Fractional calculus is essentially a non-integer-order calculus, in which the order of differentiation or integration can be real or complex numbers. The basic operation of fractional calculus is a fractional-order differentiation, ${}_aD_t^\alpha$, which denotes the fractional-order differential operator [29], i.e.;

$$ {}_aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \text{if } \Re(\alpha) > 0 \\ 1 & \text{if } \Re(\alpha) = 0 \\ \int_a^t (d\tau)^{-\alpha} & \text{if } \Re(\alpha) < 0 \end{cases} $$

where a and t are the upper and lower bounds of the operators, α is the order which can be any complex number, and $\Re(\alpha)$ is the real part of α.

The following definitions illustrate the Riemann-Liouville fractional differentiator and integrator of a function $f(t)$ of order α followed by the Caputo definition of the fractional differentiator of $f(t)$ of order α.

Definition 2.1 *For a positive integer m, let $\alpha \in \mathbb{R}^+$ be such that $m-1 \leq \alpha \leq m$. The Riemann-Liouville fractional derivative of a function $f(t)$ of order α is defined by [30]:*

$$ D_a^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)} \left(\frac{d}{dt}\right)^m \int_a^t (t-\tau)^{m-\alpha-1} f(\tau) d\tau, \tag{1} $$

where $\Gamma(\cdot)$ is the Euler's Gamma function that generalizes the factorial and allows the operator, $D_a^\alpha(\cdot)$, to take non-integer values.

Definition 2.2 *Let $f(t)$ be an integrable piecewise continuous function on any finite subinterval of $t \in (0, +\infty)$, then the Riemann-Liouville fractional integral of $f(t)$ of order α is defined as [30]:*

$$J^\alpha f(t) = \frac{t^{\alpha-1}}{\Gamma(\alpha)} * f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau) d\tau, \tag{2}$$

where $t > 0$ and $0 < \alpha \leq 1$.

Definition 2.3 *Let $\alpha \in \mathbb{R}^+$ and $m \in \mathbb{N}$ such that $m - 1 < \alpha < m$, then the Caputo fractional derivative of order α is defined by [30]:*

$$D_a^\alpha f(t) = \frac{1}{\Gamma(m-\alpha)} \int_a^t \frac{f^m(\tau)}{(t-\tau)^{\alpha+1-m}} d\tau, \;\; f^m(\tau) = \frac{d^m f(\tau)}{d\tau^m}. \tag{3}$$

The frequency response of dynamical systems is a popular approach to realize fractional-order controllers. Hence, Laplace transform is generalized to include systems of non-integer order dynamics [31]. The following two most popular definitions of factional-order derivatives in the frequency domain are stated for completeness.

Definition 2.4 *The Laplace transform of the Riemann-Liouville fractional-order derivative is given by [30]:*

$$\mathcal{L}\left\{D^\alpha f(t)\right\} = s^\alpha F(s) - \sum_{k=0}^{m-1} s^k \left[D^{\alpha-k-1} f(t)\right]|_{t=0}, \tag{4}$$

where $m - 1 \leq \alpha < m$; $m \in \mathbb{N}$, $t > 0$ and $F(s)$ is the Laplace transform of $f(t)$.

Definition 2.5 *The Laplace transform of the Caputo fractional-order derivative is given by [30]:*

$$\mathcal{L}\left\{D^\alpha f(t)\right\} = s^\alpha F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1} f^{(k)(0)}, \tag{5}$$

where $m - 1 \leq \alpha < m$; $m \in \mathbb{N}$, $t > 0$ and $F(s)$ is the Laplace transform of $f(t)$. If the derivatives of the function $f(t)$ are all equal 0 at $t = 0$ in (5), then [30]:

$$\mathcal{L}\left\{D^\alpha f(t)\right\} = s^\alpha \mathcal{L}\left\{f(t)\right\} = s^\alpha F(s). \tag{6}$$

The so-called fractional-order Laplacian operator, $s^{(\pm\alpha)}$, is expressed in the frequency domain by letting $s = j\omega$. Hence, $(j\omega)^{(\pm\alpha)}$ can be expressed as [32]:

$$s^{(\pm\alpha)} = (j\omega)^{(\pm\alpha)} = \omega^{(\pm\alpha)} \left[\cos\left(\frac{\alpha\pi}{2}\right) \pm j\sin\left(\frac{\alpha\pi}{2}\right)\right], \tag{7}$$

where $\omega \in (0, 1)$ and $j = \sqrt{-1}$.

Definition 2.6 *The Laplace transform of the fractional-order integral, $J^\alpha f(t)$, is given by [30]:*

$$\mathcal{L}\{J^\alpha f(t)\} = s^{-\alpha}\mathcal{L}\{f(t)\} = s^{-\alpha}F(s). \tag{8}$$

3 FoPID Controllers

The FoPID-controller was first introduced by Podlubny in 1997 [33]. The integration action enjoys fractional-order dynamics of order λ, while the differentiator is of order δ. The FoPID-controller is used for industrial application to improve systems' performance. It provides extra degrees of freedom by adding two more parameters to tune (λ and δ) to the original three parameters, (K_p, K_i, K_d), thus increasing the complexity of parameter tuning [22]. The fractional-order integro-differential equation that describes the FoPID-controllers is given by [22, 34]:

$$G(s) = K_p + K_i\frac{1}{s^\lambda} + K_d s^\delta, \tag{9}$$

where K_p, K_i, K_d, λ, and δ are real constants to be designed.

Obviously, increasing the number of controller parameters from three, in the case of integer-order PID controllers, to five parameters for the case of FoPID-controllers increases the complexity of the controller design. This yields a set of special cases that can be considered to simplify the design process. To overcome this problem, and to assign the five parameters all at once, optimization methods, such as the BFO algorithm, are usually adopted to find the best candidate of FoPID-controllers [7, 8, 9, 10]. For completeness, equation (10) represents the class of linear time-invariant (LTI) systems that are considered in this work [35]:

$$a_n D^{\alpha_n} y(t) + \cdots + a_0 D^{\alpha_0} y(t) = b_m D^{\beta_m} u(t) + \cdots + b_0 D^{\beta_0} u(t), \tag{10}$$

where $u(t)$ and $y(t)$ are the systems' input and output signals, while D^α defines the Caputo fractional-order differential operator of arbitrary constant orders, α_k; $k = 0, 1, 2, \cdots, n$, and β_l; $\ell = 0, 1, 2, \cdots, m$, and $n, m \in \mathbb{N}$. The transfer function of system (10) is given by [36]:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + b_{m-1}s^{\beta_{m-1}} + \cdots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1}s^{\alpha_{n-1}} + \cdots + a_0 s^{\alpha_0}}, \tag{11}$$

and the output signal of the FoPID-controller, $m(t)$, can be written as:

$$m(t) = K_p e(t) + K_i J^{\lambda} e(t) + K_d D^{\delta} e(t). \tag{12}$$

Clearly, the integer-order PID controller when $\lambda = \delta = 1$ represents one case from the set of special cases of the FoPID-controllers.

4 Implementation of FoPID Controllers

Many researchers investigated the design of FoPID-controllers using the some well-known optimization algorithms. The implementation and the effectiveness of such controllers depend on the type of approximation used to replace the fractional-order integro-differential Laplacian operators $s^{(\pm\alpha)}$ [34, 37, 38, 39]. Besides Oustaloup's and Carlson's approximations, El-Khazali introduced in [34] a biquadratic approximation that only depends on the order of integration or differentiation.

Whence the best five parameters of the FoPID-controller are found using the BFO algorithm, and in order to decide the way of implementing these controllers, the effectiveness of the design is investigated using the aforementioned three approximations. A modular representation of El-Khazali's approximation is introduced here for completeness [40]. A single module of El-Khazali's approximation enjoys a flat phase response at its center frequency, which can be considered as a constant phase element (CPE). To widen the spectrum of approximation, one may cascade several 2^{nd}-order biquadratic forms to generate higher order approximation [40, 41] as given by the following form:

$$\left(\frac{s}{\omega_g}\right)^{\alpha} = \prod_{i=1}^{n} H_i(s/\omega_i) = \prod_{i=1}^{n} \frac{N_i\left(\frac{s}{\omega_i/\omega_g}\right)}{D_i\left(\frac{s}{\omega_i/\omega_g}\right)}, \tag{13}$$

where ω_i, $i = 1, 2, ..., n$, is the center frequency of each biquadratic module, and where $\omega_g = \sqrt[n]{\prod_{i=1}^{n} \omega_i}$ is their geometric mean. If one selects the first center frequency, ω_1, of the first section, then to obtain a constant phase element, the subsequent center frequencies of each section can be calculated from the following recursive formula [40]:

$$\omega_i = \omega_x^{2(i-1)} \omega_1;\ i = 2, 3, ..., n, \tag{14}$$

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

$$a_0 a_2 \eta \gamma^4 + a_1(a_2 - a_0)\gamma^3 + (a_1^2 - a_2^2 - a_0^2)\eta\gamma^2 + a_1(a_2 - a_0)\gamma + a_0 a_2 \eta = 0, \tag{15}$$

and where $\eta = \tan(\frac{\alpha\pi}{4})$. Each biquadratic module in (13) is given by:

$$\left(\frac{s}{\omega_i}\right)^{\alpha} = H_i\left(\frac{s}{\omega_i}\right) = \frac{N_i\left(\frac{s}{\omega_i}\right)}{D_i\left(\frac{s}{\omega_i}\right)} \cong \frac{a_0(\frac{s}{\omega_i})^2 + a_1(\frac{s}{\omega_i}) + a_2}{a_2(\frac{s}{\omega_i})^2 + a_1(\frac{s}{\omega_i}) + a_0}, \tag{16}$$

where $i = 1, 2, 3, \cdots$, and where

$$\begin{aligned} a_0 &= \alpha^\alpha + 2\alpha + 1 \\ a_2 &= \alpha^\alpha - 2\alpha + 1 \\ a_1 &= (a_2 - a_0)\tan\left(\frac{(2+\alpha)\pi}{4}\right) = -6\alpha\tan\left(\frac{(2+\alpha)\pi}{4}\right). \end{aligned} \tag{17}$$

Observe that (16) is the only approximation that yields

$$H_i\left(\frac{s}{\omega_i}\right) = \left(\frac{s}{\omega_i}\right) \text{ as } \alpha \to 1. \tag{18}$$

Moreover, the reciprocal of (13) approximates a fractional-order integrator [40], or simply:

$$\left(\frac{s}{\omega_g}\right)^{-\alpha} = \prod_{i=1}^{n} \underline{\mathrm{H}}_i(s/\omega_i) = \prod_{i=1}^{n} \frac{D_i\left(\frac{s}{\omega_i/\omega_g}\right)}{N_i\left(\frac{s}{\omega_i/\omega_g}\right)}. \tag{19}$$

This means that whence the order of the integrators and differentiators are found, the implementation of the FoPID becomes straightforward.

However, in this work, the FoPID-controller will be taken along with some transfer functions of several industrial applications. In other words, we will try to optimize the performance of some systems by improving their unit-step response. This optimization will be performed by employing the BFO algorithm through using El-Khazali's approximation. For a complete description of the BFO algorithm, one may find more details in [27, 28, 42, 43, 44, 45]. The improvement of control system performance in the time domain is equivalent to the problem of minimizing $e(t)$ [36]. For proper tuning of controller in this domain and to evaluate their performance, there are several performance criteria that might be taken into consideration such as [42]. In particular, the minimization of several error functions; ITAE, IAE, ISE and ITSE, will be the main goal of our optimization technique. These error functions are of the form:

- Integral Time-Absolute Error (ITAE)

$$ITAE = \int_0^\infty t|e(t)|dt. \tag{20}$$

- Integral Absolute Error (IAE)

$$IAE = \int_0^\infty |e(t)|dt. \tag{21}$$

- Integral Square Error (ISE)

$$ISE = \int_0^{\infty} e^2(t)dt. \tag{22}$$

- Integral Time Square Error (ITSE)

$$ITSE = \int_0^{\infty} te^2(t)dt. \tag{23}$$

The performance indices given by (20-23) are very important in measuring system performance [36]. Each one shows different aspects of the system response [46, 47, 48]. The design procedure of the FoPID-controller through the BFO and algorithm is described by the block diagram shown in Figure 1.

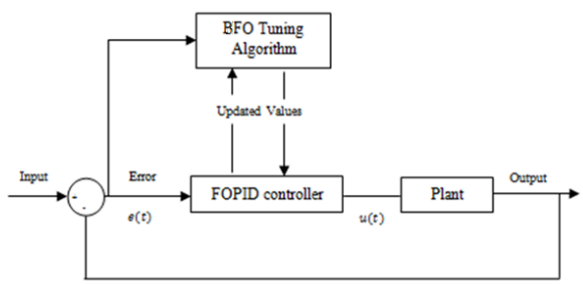


Figure 1: Block diagram of BFO tuned FoPID-controller

The Laplace transform of (12) yields the following dynamics of the FoPID-controller in the frequency domain:

$$G_c(s) = K_p + \frac{K_i}{s^{\lambda}} + K_d s^{\delta}. \tag{24}$$

Whence, the best parameters of (24) are found, one may replace $s^{(-\lambda)}$, and s^{δ} by realizable rational transfer functions using any of the existing approximation algorithms [34, 37, 38, 39]. A detailed comparative study will be carried out in the next section to highlight the effect of the approximation methods on the performance of the controlled systems.

5 Simulation Results

As will be shown, the effectiveness of any BFO algorithm used to design FoPID-controllers lies in the approximation method used to approximate the integro-differential Laplacian operators. In this section, two numerical examples are investigated to highlight the effect of approximating $s^{(\pm\alpha)}$ to the same set of optimum parameters. Table 1 shows a list of values used for the parameters of the BFO algorithm (see [27, 28, 42, 43, 44, 45]). One may use different values when necessary.

Table 1: Parameters of BFO algorithm

Parameter	Value
Number of bacteria (n)	100
The probability that each bacteria will be eliminated/dispersed	0.25
Number of chemotactic steps	4
Maximum number of reproductions to be undertaken	6
Maximum number of elimination-dispersal events to be imposed over the bacteria	2
Swimming length after which tumbling of bacteria will be done in a chemotactic step	3

In this work, FoPID-controller design using El-Khazali's approach is carried out and compared with Oustaloup's, and Carlson's approximations for $s^{(\pm\alpha)}$, $0 < \alpha \leq 1$ [35, 38, 39, 49, 50].

Remark 5.1 *For the case when the BFO-FoPID controller in (24) is tuned to $n < (\lambda, \delta) < n+1$, for $n \geq 1$, and to avoid implementing non-realizable controllers, one may cascade set of fractional-order integrators (differentiators) each of order $(\lambda/(n+1))$, $(\delta/(n+1))$, respectively; i.e, equation (24) can be rewritten as:*

$$G_c(s) = K_p + \frac{K_i}{\prod_{i=1}^{n+1}} s^{(\lambda/(n+1))} + K_d \prod_{i=1}^{(n+1)} s^{\delta/(n+1)}. \tag{25}$$

For example, when $n = 1$, equation (25) can be written as:

$$G_c(s) = K_p + \frac{K_i}{s.s^{\lambda-1}} + K_d \left(s^{\delta/2}\right)^2. \tag{26}$$

The representation of the BFO-FoPID controller in (26) can still yield realizable controllers (see Example 5.2 for further details).

Example 5.1 *Consider the transfer function of a servo motor [50]:*

$$G(s) = \frac{0.3}{s(s+0.07)}. \tag{27}$$

The best values for the fractional order integrator and differentiator using Carlson's approximation were found to be $\lambda = 0.49966$, and $\delta = 0.50196$. Thus, from (24), the best dynamics of the FoPID is given by [50]:

$$C_c(s) = 4.1267 + \frac{0.44093}{s^{0.49966}} + 1.1309s^{0.50196}, \tag{28}$$

where the following approximations were used to realize the FoPID-controller given by (28) [50], i.e.,

$$\frac{1}{s^{0.49966}} = \frac{0.1579s^4 + 0.5712s^3 + 0.2009s^2 + 0.01346s + 0.0001449}{s^4 + 0.938s^3 + 0.1414s^2 + 0.00406s + 1.133e - 005}, \quad (29)$$

and

$$s^{0.50196} = \frac{3.158s^4 + 2.963s^3 + 0.4466s^2 + 0.01282s + 3.58e - 005}{s^4 + 3.619s^3 + 1.273s^2 + 0.08526s + 0.0009181}. \quad (30)$$

Substituting from (29) and (30) into (28) yields the following 8^{th}*-order FoPID using Carlson's approximation:*

$$C_c(s) = \frac{\begin{array}{l} 5.7s^8 + 22.7s^7 + 24.3s^6 + 9.5s^5 + 1.6s^4 + 0.12s^3 \\ \qquad + 0.0038s^2 + 4.74e - 05s + 1.946e - 07 \end{array}}{\begin{array}{l} s^8 + 4.6s^7 + 4.8s^6 + 1.8s^5 + 0.28s^4 + 0.018s^3 \\ \qquad + 0.0005s^2 + 4.69e - 06s + 1.04e - 08 \end{array}} \quad (31)$$

On the other hand, the BFO algorithm is carried out using El-Khazali's approximation for $50 \leq n \leq 200$ *to minimize the four different objective functions, i.e., ITAE, IAE, ISE, and ITSE, respectively. As depicted, the values of the four fitness functions, shown in Figure 2, decrease with the increase of the number of iterations, where ITSE yields the best result.*

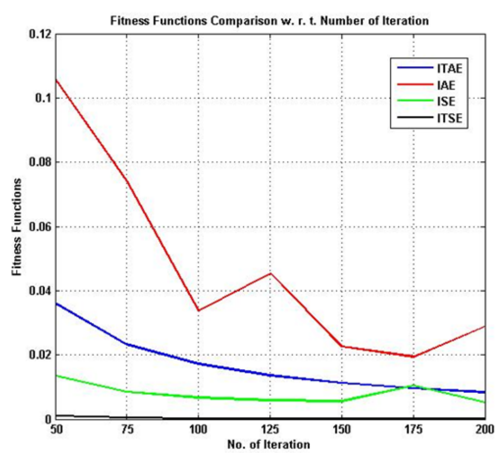


Figure 2: Fitness functions vs. number of bacteria (n).

For more insight, the different parameters that minimizes the ITSE fitness function is listed in Table 2, where the best result corresponds to $n = 200$. *The corresponding parameters for the FoPID-controller are found to be* $K_p = 13.03841$, $K_i = 1$, *and* $K_d = 200$ *with* $\lambda = 0.001$, *and* $\delta = 0.9983754$*; i.e.,*

$$C_K(s) = 13.03841 + \frac{1}{s^{0.001}} + 200s^{0.9983754}. \quad (32)$$

Table 2: Values of ITSE fitness function for different number of bacteria

n	K_p	K_i	K_d	λ	δ	ITSE
200	13.0382	1.0	200	0.001	0.998375	$6.161e-05$
175	1.0	11.253	175	0.001	0.999547	$8.281e-05$
150	9.6297	1.0	150	0.001	0.999982	$1.155e-04$
125	7.8457	1.0	125	0.001	0.999748	$1.697e-04$
100	1.0	6.0805	100	0.001	0.999994	$2.696e-04$
75	4.3032	1.0	75	0.0019	0.999867	$4.857e-04$
50	2.6271	1.03166	50	0.001	0.999989	0.001102

To develop a realizable form for (32), one needs to replace $s^{0.001}$ *and* $s^{0.9983754}$ *by the following forms, respectively [40]:*

$$s^{0.001} = \frac{1.995s^2 + 5.093s + 1.991}{1.991s^2 + 5.093s + 1.995}, \tag{33}$$

and

$$s^{0.9983754} = \frac{3.995s^2 + 4.004s + 0.001627}{0.001627s^2 + 4.004s + 3.995}. \tag{34}$$

Substituting from (33) and (34) into (32) yields the following 4^{th}*-order transfer function of the FoPID-controller:*

$$C_K(s) = \frac{1594s^4 + 5779s^3 + 6068s^2 + 1994s + 112.3}{0.003247s^4 + 7.996s^3 + 28.36s^2 + 28.32s + 7.955}. \tag{35}$$

Figure 3 shows the step response of the closed-loop system with unity feedback using $C_C(s)$ *and* $C_K(s)$ *controllers given by (31) and (35), respectively.*

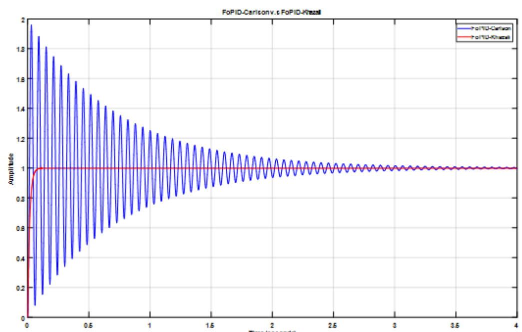


Figure 3: Closed-Loop system step response using $C_C(s)$ (blue color) and $C_K(s)$ (red color).

Obviously, a 50% order reduction is obtained (4^{th}*-order instead of* 8^{th}*-order) and a significant improvement of the step response have been achieved using El-Khazali's approximation over that of Carlson. To see this, take a look at Table 3 given below.*

Table 3: System step response using (31) and (35).

Step Response Specifications	$C_C(s)$	$C_K(s)$
Rise Time	0.0102	0.0357
Settling Time	2.8249	0.0640
Settling Min.	0.0801	0.9034
Settling Max.	1.9593	1.000
Overshoot	95.9334	9.294e-4
Peak	1.9593	1.000
Peak Time	0.0304	0.16067

Example 5.2 *Consider the following open-loop fractional-order system reported in [49]:*

$$G(s) = \frac{1}{0.8s^{2.2} + 0.5s^{0.9} + 1} = \frac{1}{0.8s^2.s^{0.2} + 0.5s^{0.9} + 1}. \tag{36}$$

The objective is to design a FoPID-controller using the BFO algorithm based on Oustaloup's and El-Khazali's approximations. The fractional-order Laplacian operators $s^{0.2}$ and $s^{0.9}$ are first replaced by integer-order rational functions using Oustaloup's and El-Khazali's methods; i.e.,

$$s_O^{0.2} = \frac{2.512s^5 + 98.83s^4 + 531.7s^3 + 442.3s^2 + 56.87s + 1}{s^5 + 56.87s^4 + 442.3s^3 + 531.7s^2 + 98.83s + 2.512}, \tag{37}$$

$$s_O^{0.9} = \frac{63.1s^5 + 1303s^4 + 3679s^3 + 1606s^2 + 108.4s + 1}{s^5 + 108.4s^4 + 1606s^3 + 3679s^2 + 1303s + 63.1}, \tag{38}$$

and

$$s_K^{0.2} = \frac{2.125s^2 + 5.051s + 1.325}{1.325s^2 + 5.051s + 2.125}, \tag{39}$$

$$s_K^{0.9} = \frac{3.71s^2 + 4.215s + 0.1095}{0.1095s^2 + 4.215s + 3.71}. \tag{40}$$

Substituting from (37-38) and (39-40) into (36) yields the following two transfer functions $G_O(s)$ and $G_K(s)$, respectively:

$$G_O(s) = \frac{\begin{array}{c} s^{10} + 165.3s^9 + 8213s^8 + 1.44e005s^7 \\ +9.79e005s^6 + 2.57e006s^5 + 2.69e006s^4 \\ + 1.09e006s^3 + 1.72e005s^2 + 9509s + 158.5 \end{array}}{\begin{array}{c} 2.01s^{12} + 296.9s^{11} + 1.23e004s^{10} + 1.83e005s^9 \\ +1.08e006s^8 + 2.80e006s^7 + 4.12e006s^6 + 4.6e006s^5 \\ + 3.4e006s^4 + 1.2e006s^3 + 1.8e005s^2 + 9695s + 159.8 \end{array}} \tag{41}$$

and

$$G_K(s) = \frac{0.1451s^4 + 6.138s^3 + 26.44s^2 + 27.7s + 7.884}{0.1862s^6 + 7.608s^5 + 26.06s^4 + 37.76s^3 + 45.03s^2 + 32.45s + 8}. \tag{42}$$

The optimal BFO-FoPID controller using Oustaloup's approximation denoted by $C_O(s)$ is given by [49]:

$$C_O(s) = 82.3532 + \frac{40.888}{s^{1.5733}} + 27.1914 s^{1.8797}. \tag{43}$$

It was reported in [49] that the integro-differential operators in (43) were simplified as $s^{1.5733} = s \times s^{0.5733}$, and $s^{1.8797} = s \times s^{0.8797}$, respectively. Consequently, the following Oustaloup's approximations were reported in [49] to realize $s^{0.5733}$, and $s^{0.8797}$; i.e.,

$$s^{0.5733} = \frac{14.02s^5 + 391s^4 + 1492s^3 + 879.7s^2 + 80.21s + 1}{s^5 + 80.21s^4 + 879.7s^3 + 1492s^2 + 391s + 14.02}, \tag{44}$$

and

$$s^{0.8797} = \frac{57.46s^5 + 1209s^4 + 3478s^3 + 1547s^2 + 106.4s + 1}{s^5 + 106.4s^4 + 1547s^3 + 3478s^2 + 1209s + 57.46}. \tag{45}$$

Remark 5.2 *Notice that convolving an integer-order differentiator with (45) yields an unrealizable FoPID-controller, (i.e., the order of the numerator is greater than the order of the denominator). This is not acceptable even if the algebraic manipulation of the closed-loop system transfer function yields a proper and stable closed-loop system. Since $1 < \delta < 2$, one may overcome this problem by decomposing $s^{\delta} = s^{(\delta/2)} \times s^{(\delta/2)}$, or $s^{1.8797} = s^{0.9398} \times s^{0.9398}$.*

Now, to conduct a fair comparison between Oustaloup's and El-Khazali's approximations, a 3^{rd}-order Oustaloup's approximation to $s^{0.9398}$ and $s^{0.5733}$ is used for $\omega \in (0.01, 100)$ rad/s, i.e.,

$$s^{0.9398} = \frac{659.8s^3 + 7655s^2 + 879.4s + 1}{s^3 + 879.4s^2 + 7655s + 659.8}, \tag{46}$$

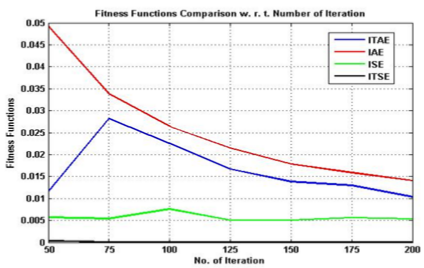
and

$$s^{0.5733} = \frac{52.47s^3 + 1416s^2 + 378.2s + 1}{s^3 + 378.2s^2 + 1416s + 52.47}. \tag{47}$$

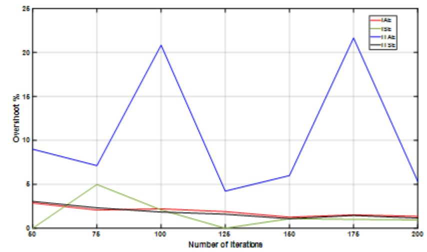
Substituting from (46) and (47) into (43) yields the following 7^{th}-order realizable FoPID-Oustaloup's controller:

$$C_O(s) = \frac{\begin{array}{r}9.5e05s^7 + 4.02e07s^6 + 4.4e08s^5 + 1.05e09s^4 \\ + 4.9e08s^3 + 4.76e08s^2 + 5.5e07s + 1.4e06\end{array}}{\begin{array}{r}52.47s^7 + 4.76e04s^6 + 1.65e06s^5 + 1.12e07s^4 \\ + 3.83e06s^3 + 2.57e05s^2 + 659.8s\end{array}} \tag{48}$$

Repeating the same procedure used in the previous example implies Figure 4 (a) that shows the values of the four fitness functions for $50 \leq n \leq 200$. Figure 4 (b), however, shows the corresponding values of the percentage overshoot of the system step-response.



(a)



(b)

Figure 4:
(a) Values of the four objective functions vs. no. of bacteria, (b) Overshoot vs. no. of bacteria.

Obviously, the ISE fitness function yields a zero overshoot for $n = 125$*, while the ITSE and the IAE are competitive to each other. One may be tempted to choose the controller parameters using the ISE fitness function for* $n = 125$*, which gives* $K_p = 125$*,* $K_i = 0.98837$*,* $K_d = 125$*,* $\lambda = 0.8304$ *and* $\delta = 1.66$*. From Remark 5.2, this case should be avoided because it may yield an unrealizable controller if it is not treated carefully. Therefore, to fairly compare between El-Khazali's and Oustaloup's approximations, the case of a maximum range of* $n = 100$ *is considered here, which gives the following FoPID-BFO controller:*

$$C_K(s) = 100 + \frac{100}{s^{0.9944475}} + 66.38609s^{0.9721622}. \tag{49}$$

Replacing the integro-differential operators in (49) using El-Khazali's approximation gives the following controller:

$$C_K(s) = \frac{1047s^4 + 3761s^3 + 5925s^2 + 4773s + 1563}{0.1139s^4 + 16.3s^3 + 31.91s^2 + 15.74s + 0.02187}. \tag{50}$$

Figure 5 shows the step-response of the closed-loop system using the two controllers $C_O(s)$ *and* $C_K(s)$ *given by (48) and (50), respectively, while Table 4 shows the corresponding information of the closed-loop system step response.*

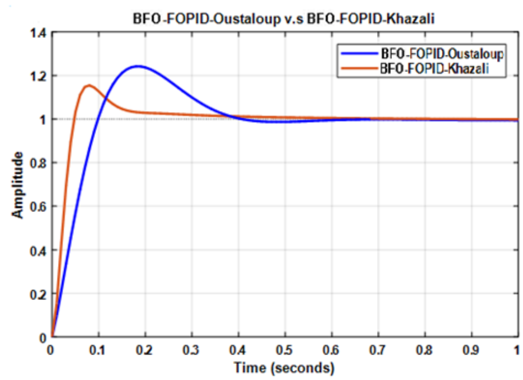


Figure 5: Closed-loop step response using (48) and (50).

Table 4: System step response using (48) and (50).

Step Response Specifications	$C_O(s)$	$C_K(s)$
Rise Time	0.074	0.0326
Settling Time	0.371	0.291
Settling Min.	0.9227	0.905
Settling Max.	1.2423	1.155
Overshoot	24.23%	15.5%
Peak	1.2423	1.155
Peak Time	0.1847	0.0787

Clearly, El-Khazali's approximation method provides a lower order controller and yields a better system performance than the case when using Outsaloup's approximation.

6 Conclusion

A new design method is proposed to synthesize FoPID-controllers using the BFO algorithm that minimizes several objective functions. The BFO algorithm is mainly used to minimize the well-known fitness functions; ITAE, IAE, ITSE and ISE, which eventually improves the time domain characteristics of the controlled systems. The proposed method ensures the implementation of lower order and realizable FoPID-controllers. The biquadratic form of El-Khazali's approximation alleviates the need of using additional filters that is usually considered when implementing integer-order PID controllers. Moreover, it exhibits an adaptive nature since the structure of the controller is fully dependent on the orders of the integro-differential components of the FoPID-controllers.

References

[1] Chen, L., He, Y., Chai, Y. & Wu, R. (2014). New results on stability and stabilization of a class of nonlinear fractional-order systems, *Nonlinear Dyn., 75(4)*, 633-641.

[2] Podlubny, I. (1999). *Fractional Differential Equations.* San Diego, California, USA: Academic Press.

[3] Das, S., Saha, S., Gupta, A. & Das, S. (2011). Analog realization of fractional Order hybrid differentiators via Carlson's approach, In *2011-International Conference on Multimedia Signal Processing and Communication Technologies, Zakir Hussain College of Engineering and Technology, AMU, Algeria.* IEEE.

[4] Dorf, R.C. & Bishop, R.H. (2011). *Modern Control Systems.* New Jersey, USA: Prentice Hall.

[5] Xue, D., Chen, Y.Q. & Atherton D.P. (2007). *Linear Feedback Control: Analysis and Design with MATLAB.* Philadelphia, USA: Siam.

[6] De Silva, C.W. (2009). *Modeling and Control of Engineering Systems.* Boca Raton-London-New York: CRC Press.

[7] Maiti, D., Biswas, S. & Konar, A. (2008). Design of a fractional order PID controller using particle swarm optimization technique, In *2nd-National Conference on Recent Trends in Information Systems (ReTIS-08), Jadavpur University, Kolkata, India.* IEEE.

[8] Merrikh-Bayat, F. & Karimi-Ghartemani, M. (2010). Method for designing $PI^{\lambda}D^{\delta}$ stabilisers for minimum-phase fractional-order systems, *IET Control Theory Appl. A., 4(1)*, 61–70.

[9] Jain, R.V., Aware, M.V. & Junghare, A.S. (2016). Tuning of fractional order PID controller using particle swarm optimization technique for DC motor speed control, In *1st-IEEE International Conference on Power Electronics: Intelligent Control and Energy Systems (ICPEICES-2016), Delhi Technological University, India.* IEEE.

[10] Qu, L., Hu, H. & Huang, Y. (2010). Fractional order PID controller based on particle swarm optimization implemented with FPGA, In *2010 International Conference on Artificial Intelligence and Computational Intelligence, Hangzhou, China.* IEEE.

[11] Gouari, Y., Dahmani, Z. & Jebril, I. (2020). Application of fractional calculus on a new differential problem of duffing type, *Advances in Mathematics: Scientific Journal, 9(12)*, 10989–11002.

[12] Batiha, I.M., Albadarneh, R.B., Momani, S. & Jebril I.H., Dynamics analysis of fractional order hopfield networks, *International Journal of Biomathematics, 13(08)*, 2050083.

[13] Jin, Y.S., Chen, Y.Q. & Xue, D.Y. (2011). Time constant robust analysis of fractional order [proportional derivative] controller, *IET Control Theory & Applications, 5(1)*, 164-172.

[14] Rinu, P.S. (2015). Design and analysis of optimal fractional-Order PID controller, *International Journal of Applied Engineering Research, 10(3)*, 23000-23002.

[15] Batiha, I.M., El-Khazali, R. & Momani, S. (2019). Dynamic responses of PI^λ, PD^δ, and $PI^\lambda D^\delta$ controllers for prosthetic hand model using PSO algorithm, *The IEEE International Symposium on Signal Processing and Information Technology (ISSPIT19), Ajman, United Arab Emirates, December 10-12*. IEEE.

[16] Momani, S., Batiha, I.M. & El-Khazali, R. (2019). Design of $PI^\lambda D^\delta$-heart rate controllers for cardiac pacemaker, *The IEEE International Symposium on Signal Processing and Information Technology (ISSPIT19), Ajman, United Arab Emirates, December 10-12*. IEEE.

[17] Momani, S., El-Khazali, R. & Batiha I.M. (2019). Tuning PID and $PI^\lambda D^\delta$ controllers using particle swarm optimization algorithm via El-Khazali's approach, *AIP Conference Proceedings, 2172*.

[18] Rajasekhar, A., Das, S. & Abraham, A. (2013). Fractional order PID controller design for speed control of chopper fed DC motor drive using artificial bee colony algorithm, In *2013 World Congress on IEEE, in Nature and Biologically Inspired Computing (NaBIC), Fargo, North Dakota, USA*. IEEE.

[19] El-Khazali, R., Batiha, I.M. & Momani, S. (2019). Approximation of fractional-order operators, *Springer Proceedings in Mathematics and Statistics, 303*, 121-151.

[20] El-Khazali, R., Batiha, I.M. & Momani S. (2022). The drug administration via fractional-order $PI^\lambda D^\delta$-controller, *Progress in Fractional Differentiation and Applications, 8(1)*.

[21] Mohamed, M.J. & Khashan, A. (2014). Comparison between PID and FOPID controllers based on particle swarm optimization, In *The 2nd-Engineering Conference of Control, Computers and Mechatronics Engineering (ECCCM2, 2014), University of Technology, Bagdad, Iraq*. University of Technology.

[22] Roy, A. & Srivastava, S. (2016). Design of optimal controller for speed control of DC motor using constrained particle swarm optimization, In *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Tamil Nadu, India*. IEEE.

[23] Wang, P. & Kwok, D.P. (1994). Optimal design of PID process controllers based on genetic algorithms, *Control Engineering Practice, 2(4)*, 641-648.

[24] Tan, C. & Liang Z.S. (2014). Modeling and simulation analysis of fractional order Boost converter in pseudo-continuous conduction mode, *Acta Phys. Sin., 63(7)*.

[25] Cech, M. & Schlegel, M. (2006). The fractional-order PID controller outperforms the classical one, In *Process control 2006, Pardubice Technical University, Czech Republic*. Technical University.

[26] Ogata, K. (2010). *Modern Control Engineering*. New Jersey, USA: Prentice Hall.

[27] Das, S., Biswas, A., Dasgupta, S. & Abraham, A. (2009). Bacterial foraging optimization algorithm: Theoretical foundations, analysis, and applications, *Foundations of Computational Intelligence Volume 3. Studies in Computational Intelligence, vol. 203*. Springer. Berlin, Heidelberg.

[28] Hernández-Ocaña, B., Hernández-Torruco, J., Chááavez-Bosquez, O., Canul-Reich, J. & Montané-Jiménez L.G. (2019). Bacterial foraging optimization algorithm with mutation to solve constrained problems, *Acta Universitaria, 29*, 1-16.

[29] Petras, I. (2011). *Fractional order nonlinear systems-modeling: analysis and simulation*. New Jersey, USA: Springer publication.

[30] Podlubny, I. (1999). *Fractional Differential Equations*. San Diego, California, USA: Academic Press.

[31] Atherton, D.P., Tan, N. & Yüce A. (2015). Methods for computing the time response of fractional-order systems, *IET Control Theory and Applications, 9*, 817-830.

[32] Tepljakov, A. (2011). *Fractional-order calculus based identification and control of linear dynamic systems* (Master thesis, Tallinn University of Technology).

[33] Monje, C.A., Chen, Y.Q., Vinagre, B.M., Xue, D. & Feliu V. (2010). *Fractional-Order Systems and Controls: Fundamentals and Applications*. London-New York: Springer.

[34] El-Khazali, R. (2014). Discretization of fractional-order differentiators and integrators, *IFAC Proceedings Volumes, 47(3)*, 2016-2021.

[35] Das, S. & Pan, I. (2012). *Fractional Order Signal Processing: Introductory Concepts and Applications*. Berlin, Germany: Springer Briefs in Applied Sciences and Technology.

[36] Tepljakov, A. (2015). *Fractional-order modeling and control of dynamic systems* (Doctoral dissertation, Tallinn University of Technology).

[37] Mahmood, A.K. & Saleh, S.A. (2018). Realization of fractional-order proportional-integral-derivative controller using fractance circuit, *Journal of Electrical Engineering, 2*, 22-32.

[38] Shrivastava, N. & Varshney, P. (2015). Rational approximation of fractional order systems using Carlson's method, In *2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI), Faridabad, India*. IEEE.

[39] Oustaloup, A., Levron, F., Mathieu, B. & Nanot, F.M. (2000). Frequency band complex non-integer differentiator: characterization and synthesis, *IEEE Trans. on Circuit and Systems – I, Fund. Theory and Appl., 47*, 25–39.

[40] El-Khazali, R. (2015). On the biquadratic approximation of fractional-order Laplacian operators, *Analog Int. Circuits and Sig. Proc., 82*, 503-517.

[41] Gil'mutdinov, A.K., Ushakov, P.A. & El-Khazali, R. (2017). *Fractal Elements and their Applications*. Switzerland: Springer.

[42] Hernández-Ocaña, B., Chávez-Bosquez, O., Hernández-Torruco, J., Canul-Reich, J. & Pozos-Parra, P. (2018). Bacterial foraging optimization algorithm for menu planning, *IEEE Access, 6*, 8619-8629.

[43] Halilu, B.D., Anene, E.C., Omigzegba, E.E., Maijama'a, L. & Baraza S.A. (2019). Optimization of PID controller gains for identified magnetic levitation plant using bacteria foraging algorithm, *International Journal of Engineering and Modern Technology, 5*, 12-18.

[44] Muñoz, M.A., Halgamuge, S.K., Alfonso, W. & Caicedo, E.F. (2010). Simplifying the bacteria foraging optimization algorithm, *IEEE Congress on Evolutionary Computation, Barcelona, Spain*. IEEE.

[45] Gole, H., Barve, P., Kesarkar, A.A. & Selvaganesan, N. (2012). Investigation of fractional control performance for magnetic levitation experimental set-up, *2012 Conference on Emerging Trends in Science, Engineering and Technology, Tiruchirapalli, India*. IEEE.

[46] Khubalkar, S.W., Chopade, A.S., Junghare, A.S. & Aware, M.V. (2016). Design and tuning of fractional order PID controller for speed control of permanent magnet brushless DC motor, *2016 IEEE First International Conference on Control, Measurement and Instrumentation (CMI), Kolkata, India*. IEEE.

[47] Maiti, D., Acharya, A., Chakraborty, M., Konar, A. & Janarthanan, R. (2008). Tuning PID and $PI^{\lambda}D^{\delta}$ controllers using the integral time absolute error criterion, *4th IEEE Conference on Information and Automation for Sustainability, Colombo, Sri Lanka*. IEEE.

[48] Ramezanian, H. & Balochian, S. (2013). Optimal design a fractional-order PID controller using particle swarm optimization algorithm, *International Journal of Control and Automation, 6*, 55-67.

[49] Yeroglu, C. & Tan, N. (2011). Note on fractional-order proportional integral differential controller design, *IET Control Theory Appl., 5*, 1978–1989.

[50] Shah, P., Agashe, S. & Kulkarni, A.J. (2018). Design of a fractional $PI^{\lambda}D^{\mu}$ controller using the cohort intelligence method, *Frontiers of Information Technology & Electronic Engineering, 19*, 437-445.