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 $P^{\lambda}D^{\delta}$) 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 $\alpha$, 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 $\alpha$, 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; \(\lambda\) and \(\delta\), 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, \( \alpha \) is the order which can be any complex number, and \( \Re(\alpha) \) is the real part of \( \alpha \).

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

**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 \( \alpha \) is defined by [30]:

\[
D_a^\alpha f(t) = \frac{1}{\Gamma(m - \alpha)} \left( \frac{d}{dt} \right)^m \int_a^t (t - \tau)^{m-\alpha-1} f(\tau)d\tau,
\] (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 $\alpha$ is defined as [30]:

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

(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 $\alpha$ is defined by [30]:

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

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]:

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

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]:

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

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]:

$$\mathcal{L} \{D^\alpha f(t)\} = s^\alpha \mathcal{L} \{f(t)\} = s^\alpha F(s). \quad \text{(6)}$$
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]:

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

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]:

\[
\mathcal{L}\{J^\alpha f(t)\} = s^{-\alpha}\mathcal{L}\{f(t)\} = s^{-\alpha}F(s).
\]

### 3 FoPID Controllers

The FoPID-controller was first introduced by Podlubny in 1997 [33]. The integration action enjoys fractional-order dynamics of order \( \lambda \), while the differentiator is of order \( \delta \). 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 (\( \lambda \) and \( \delta \)) 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]:

\[
G(s) = K_p + \frac{K_i}{s^\lambda} + K_d s^\delta,
\]

where \( K_p, K_i, K_d, \lambda, \) and \( \delta \) 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]:

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

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

\[
G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^{\beta_m} + b_{m-1}s^{\beta_{m-1}} + \cdots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1}s^{\alpha_{n-1}} + \cdots + a_0 s^{\alpha_0}},
\]
and the output signal of the FoPID-controller, $m(t)$, can be written as:

$$m(t) = K_p e(t) + K_i \int_0^t e(t) + K_d D^\delta e(t). \quad (12)$$

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 2nd-order biquadratic forms to generate higher order approximation [40, 41] as given by the following form:

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

where $\omega_i$, $i = 1, 2, ..., n_i$, 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, $\omega_1$, 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]:

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

where $\omega_x$ is the maximum real solution of the following polynomial:

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

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

$$\left(\frac{s}{\omega_i}\right)^\alpha = H_i\left(\frac{s}{\omega_i}\right) = \frac{N_i\left(\frac{s}{\omega_i}\right)}{D_i\left(\frac{s}{\omega_i}\right)} \approx \frac{a_0\left(\frac{s}{\omega_i}\right)^2 + a_1\left(\frac{s}{\omega_i}\right) + a_2}{a_2\left(\frac{s}{\omega_i}\right)^2 + a_1\left(\frac{s}{\omega_i}\right) + a_0}, \quad (16)$$
where $i = 1, 2, 3, \ldots$, and where
\[
\begin{align*}
    a_0 &= \alpha^\alpha + 2\alpha + 1 \\
    a_2 &= \alpha^\alpha - 2\alpha + 1 \\
    a_1 &= (a_2 - a_0) \tan\left(\frac{(2 + \alpha\pi)}{4}\right) = -6\alpha \tan\left(\frac{(2 + \alpha\pi)}{4}\right). 
\end{align*}
\] (17)

Observe that (16) is the only approximation that yields
\[
H_i\left(\frac{s}{\omega_i}\right) = \left(\frac{s}{\omega_i}\right) \quad \text{as} \quad \alpha \to 1. \quad (18)
\]

Moreover, the reciprocal of (13) approximates a fractional-order integrator [40], or simply:
\[
\left(\frac{s}{\omega_g}\right)^{-\alpha} = \prod_{i=1}^{n} H_i(s/\omega_i) = \prod_{i=1}^{n} D_i\left(\frac{s}{\omega_i/\omega_g}\right) N_i\left(\frac{s}{\omega_i/\omega_g}\right). \quad (19)
\]

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)
\[
ITAE = \int_{0}^{\infty} t|e(t)|dt. \quad (20)
\]

- Integral Absolute Error (IAE)
\[
IAE = \int_{0}^{\infty} |e(t)|dt. \quad (21)
\]
Optimal Design of Fractional-order PID Controllers ...

- Integral Square Error (ISE)
  \[ ISE = \int_{0}^{\infty} e^2(t) dt. \]  
  (22)

- Integral Time Square Error (ITSE)
  \[ ITSE = \int_{0}^{\infty} te^2(t) dt. \]  
  (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](image)

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

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

Whence, the best parameters of (24) are found, one may replace \( s^{(-\lambda)} \), and \( s^\delta \) 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

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

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:

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

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

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

$$G(s) = \frac{0.3}{s(s + 0.07)}.$$  \quad (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]:

$$C_c(s) = 4.1267 + \frac{0.44093}{s^{0.49966}} + 1.1309s^{0.50196},$$  \quad (28)
where the following approximations were used to realize the FoPID-controller given by (28) [50], i.e.,

\[
\frac{1}{s^{0.49966}} = \frac{0.1579s^4 + 0.5712s^3 + 0.2009s^2 + 0.01346s + 0.0001449}{s^4 + 0.938s^3 + 0.1414s^2 + 0.00406s + 1.133e-05},
\]

(29)

and

\[
s^{0.50196} = \frac{3.158s^4 + 2.963s^3 + 0.4466s^2 + 0.01282s + 3.58e-05}{s^4 + 3.619s^3 + 1.273s^2 + 0.08526s + 0.0009181}.
\]

(30)

Substituting from (29) and (30) into (28) yields the following 8th-order FoPID using Carlson’s approximation:

\[
C_c(s) = \frac{5.7s^8 + 22.7s^7 + 24.3s^6 + 9.5s^5 + 1.6s^4 + 0.12s^3 + 0.0038s^2 + 4.74e-05s + 1.946e-07}{s^8 + 4.6s^7 + 4.8s^6 + 1.8s^5 + 0.28s^4 + 0.018s^3 + 0.0005s^2 + 4.69e-06s + 1.04e-08}.
\]

(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).](image)

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.,

\[
C_K(s) = 13.03841 + \frac{1}{s^{0.001}} + 200s^{0.9983754}.
\]

(32)
Table 2: Values of ITSE fitness function for different number of bacteria

<table>
<thead>
<tr>
<th>( n )</th>
<th>( K_p )</th>
<th>( K_i )</th>
<th>( K_d )</th>
<th>( \lambda )</th>
<th>( \delta )</th>
<th>ITSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>13.0382</td>
<td>1.0</td>
<td>200</td>
<td>0.001</td>
<td>0.998375</td>
<td>6.161e-05</td>
</tr>
<tr>
<td>175</td>
<td>9.6297</td>
<td>1.0</td>
<td>175</td>
<td>0.001</td>
<td>0.999547</td>
<td>8.281e-05</td>
</tr>
<tr>
<td>150</td>
<td>7.8457</td>
<td>1.0</td>
<td>150</td>
<td>0.001</td>
<td>0.999982</td>
<td>1.155e-04</td>
</tr>
<tr>
<td>125</td>
<td>6.0805</td>
<td>1.0</td>
<td>125</td>
<td>0.001</td>
<td>0.999748</td>
<td>1.697e-04</td>
</tr>
<tr>
<td>100</td>
<td>4.3032</td>
<td>1.0</td>
<td>100</td>
<td>0.0019</td>
<td>0.999867</td>
<td>4.857e-04</td>
</tr>
<tr>
<td>75</td>
<td>2.6271</td>
<td>1.03166</td>
<td>50</td>
<td>0.001</td>
<td>0.999989</td>
<td>0.001102</td>
</tr>
</tbody>
</table>

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

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

and

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

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

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

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 (4th-order instead of 8th-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.
Substituting from (37-38) and (39-40) into (36) yields the following two trans-
operators

\[ G(s) = \frac{1}{0.8s^{2.2} + 0.5s^{0.9} + 1} = \frac{1}{0.8s^2s^{0.2} + 0.5s^{0.9} + 1}. \] (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.,

\[ s^{0.2}_O = \frac{2.512s^5 + 98.83s^4 + 531.7s^3 + 442.3s^2 + 56.87s + 1}{s^5 + 56.87s^4 + 442.3s^3 + 531.7s^2 + 98.83s + 2.512}, \] (37)

\[ s^{0.9}_O = \frac{531.7s^5 + 1303s^4 + 3679s^3 + 1606s^2 + 108.4s + 1}{s^5 + 108.4s^4 + 1606s^3 + 3679s^2 + 1303s + 63.1}, \] (38)

and

\[ s^{0.2}_K = \frac{2.125s^2 + 5.051s + 1.325}{1.325s^2 + 5.051s + 2.125}, \] (39)

\[ s^{0.9}_K = \frac{3.71s^2 + 4.215s + 0.1095}{0.1095s^2 + 4.215s + 3.71}. \] (40)

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

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

and

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

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

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

The step response is shown in Table 3.

<table>
<thead>
<tr>
<th>Step Response Specifications</th>
<th>( C_C(s) )</th>
<th>( C_K(s) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rise Time</td>
<td>0.0102</td>
<td>0.0357</td>
</tr>
<tr>
<td>Settling Time</td>
<td>2.8249</td>
<td>0.0640</td>
</tr>
<tr>
<td>Settling Min.</td>
<td>0.0801</td>
<td>0.9034</td>
</tr>
<tr>
<td>Settling Max.</td>
<td>1.9593</td>
<td>1.000</td>
</tr>
<tr>
<td>Overshoot</td>
<td>95.9334</td>
<td>9.294e-4</td>
</tr>
<tr>
<td>Peak</td>
<td>1.9593</td>
<td>1.000</td>
</tr>
<tr>
<td>Peak Time</td>
<td>0.0304</td>
<td>0.16067</td>
</tr>
</tbody>
</table>
The optimal BFO-FoPID controller using Oustaloup’s approximation denoted by $C_O(s)$ is given by [49]:

$$C_O(s) = 82.3532 + \frac{40.888}{s^{1.5733}} + 27.1914s^{1.8797}.$$  \hspace{1cm} (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.,

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

and

$$s^{0.8797} = \frac{57.46s^5 + 1209s^4 + 3478s^3 + 1547s^2 + 106.4s + 1}{s^6 + 106.4s^5 + 1547s^4 + 3478s^3 + 1209s + 57.46}.$$  \hspace{1cm} (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 $s^\delta = s^{(6/2)} \times s^{(6/2)}$, 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 3rd-order Oustaloup’s approximation to $s^{0.9398}$ and $s^{0.5733}$ is used for $\omega \in (0.01, 100)$ rad/s, i.e.,

$$s^{0.9398} = \frac{659.8s^3 + 7655s^2 + 879.4s + 1}{s^3 + 879.4s^2 + 7655s + 659.8},$$  \hspace{1cm} (46)

and

$$s^{0.5733} = \frac{52.47s^3 + 1416s^2 + 378.2s + 1}{s^3 + 378.2s^2 + 1416s + 52.47}.$$  \hspace{1cm} (47)

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

$$C_O(s) = \frac{9.5e05s^7 + 4.02e07s^6 + 4.4e08s^5 + 1.05e09s^4 + 4.9e08s^3 + 4.76e08s^2 + 5.5e07s + 1.4e06}{52.47s^7 + 4.76e04s^6 + 1.65e06s^5 + 1.12e07s^4 + 3.83e06s^3 + 2.57e05s^2 + 659.8s}.$$  \hspace{1cm} (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.
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).

<table>
<thead>
<tr>
<th>Step Response Specifications</th>
<th>$C_G(s)$</th>
<th>$C_K(s)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rise Time</td>
<td>0.074</td>
<td>0.0326</td>
</tr>
<tr>
<td>Settling Time</td>
<td>0.371</td>
<td>0.291</td>
</tr>
<tr>
<td>Settling Min.</td>
<td>0.9227</td>
<td>0.905</td>
</tr>
<tr>
<td>Settling Max.</td>
<td>1.2423</td>
<td>1.155</td>
</tr>
<tr>
<td>Overshoot</td>
<td>24.23%</td>
<td>15.5%</td>
</tr>
<tr>
<td>Peak</td>
<td>1.2423</td>
<td>1.155</td>
</tr>
<tr>
<td>Peak Time</td>
<td>0.1847</td>
<td>0.0787</td>
</tr>
</tbody>
</table>

Clearly, El-Khazali’s approximation method provides a lower order controller and yields a better system performance than the case when using Ous-saloup’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


