Computers and Mathematics with Applications 66 (2013) 746–754

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

# Optimization of zero–pole interlacing for indirect discrete approximations of noninteger order operators

G. Maione [a,*], R. Caponetto [b], A. Pisano [c]

[a] Department of Electrical and Electronic Engineering, Technical University of Bari, Bari, Italy
[b] Department of Electrical, Electronic and Systems Engineering, University of Catania, Catania, Italy
[c] Department of Electrical and Electronic Engineering, University of Cagliari, Cagliari, Italy

ARTICLE INFO

*Keywords:*
Noninteger order operators
Fractional order controllers
Rational approximation
Zero–pole interlacing
Discretization
Particle swarm optimization

ABSTRACT

To realize the basic units of fractional order controllers, it is necessary to approximate irrational differintegrators. Many methods deal with analog approximation. But not many results exist for discrete approximation. In this case, approximations are most times poor in some frequency range, or the quality of the approximation is good in a narrow frequency interval. This paper proposes an indirect discretization method that takes advantage from the particle swarm optimization (PSO) to approximate fractional order operators. The method employs an heuristic procedure to optimize the interlacing of zero–pole pairs on the real axis. Then, a discretization rule is applied to obtain the discrete approximation. Simulation results show that the frequency response obtained by PSO improves the approximation offered by other efficient and recent indirect discretization techniques.



## 1. Introduction

Noninteger (fractional) order differential or integral operators $s^{\nu}$, with $\nu \in \mathbb{R}$ and often $0 < |\nu| < 1$, are irrational Laplace domain operators that derive from fractional calculus and are the basic units in the so-called fractional order controllers (FOC) or fractional order PID controllers (FOPID) or $PI^{\lambda}D^{\mu}$ controllers [1]. These controllers became quite popular because they enhance design freedom by $\lambda$ and $\mu$ and may achieve more robust behavior to gain and load variations and higher closed-loop performance than standard PID controllers, especially if the plant is modeled as a noninteger order system but even for integer order systems [2,3]. If one considers the extensive use of PID in industrial control loops [4], then it is easy to imagine the benefit and impact of FOC [1,5,6]. Then, many contributions exist in the literature to design and tune FOC, some inspired by popular approaches like the symmetrical optimum method for controlling servo systems [7]. But to realize the designed FOC, it is necessary to approximate the infinite dimensional operators $s^{\nu}$, also called *differintegrators*. In particular, efficient approximations are necessary for digital implementation [8].

The literature reports some contributions in the field of discrete approximations for digital implementation of FOC. Often good accuracy is achieved in narrow frequency ranges only. Some methods are based on finite impulse response (FIR) filters [9,10], some other on infinite impulse response (IIR) filters [11,12]. Here we consider IIR filters because, in general, rational approximations provide faster convergence and wider domain of convergence than polynomial approximations [13].

We may distinguish between direct and indirect discretization approaches [14]. The direct approach is based on using a generating function, say $s = \psi(z^{-1})$, that depends on the sampling period $T$, and on truncating power series expansions

* Corresponding author. Tel.: +39 080 5963 247.
*E-mail addresses:* gmaione@poliba.it (G. Maione), riccardo.caponetto@dieei.unict.it (R. Caponetto), pisano@diee.unica.it (A. Pisano).

0898-1221/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2013.01.007

(PSE) or continued fraction expansions (CFE). For example, the PSE of the Euler's operator $s = \frac{1-z^{-1}}{T}$ leads to a discretization formula [15]. Using Tustin's operator $s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$ provides other approximations [14,16,17]. The indirect approach follows two steps [18]: in the first one, a frequency-domain fitting is applied to obtain an analog $s$-domain approximation; in the second step, the fit $s$-transfer function is discretized by using a proper operator (e.g. Tustin, Euler, Al-Alaoui). See, for example, the method developed in [19] or in [8] for a digital realization robust to truncation errors due to finite word length.

For a useful control application, the developed approximations must be characterized by stable poles, $\lambda_1, \lambda_2, \ldots, \lambda_N$, and minimum-phase zeros, $\mu_1, \mu_2, \ldots, \mu_N$. In addition, many approximations verify an *interlacing* property, i.e. simple poles and zeros alternate along the real axis. This property holds true for many approximations both in the continuous $s$-domain and in the discrete $z$-domain [8]. In continuous approximations, interlacing occurs either because it is an "a priori choice" leading to a fractional slope of the Bode magnitude diagram of the open-loop gain [20], or because the peculiar CFE leads to interlaced zeros and poles. In particular, some classes of CFE approximations are equivalent to the Stieltjes's expansion [21], expressed in the Pringsheim notation as follows [22]:

$$f^S(s) = 1 + \frac{1|}{|\alpha_1 s} + \frac{1|}{|\beta_2} + \cdots + \frac{1|}{|\alpha_{2N-1} s} + \frac{1|}{|\beta_{2N}} \tag{1}$$

that enjoys the zero–pole interlacing and received a formal proof for this property [23].

Therefore, we may think that the quality of the approximation depends on the mutual position of the interlaced zeros and poles. Hence, our idea is to apply an indirect discretization and optimize the location of zero–pole pairs on the real axis so that the frequency response of the approximation has a small deviation from the fractional differintegrator $s^\nu$, in a frequency range of interest and for a limited amount $N$ of zero–pole pairs. In this way the magnitude plot is very close to a linear plot with a slope of 20 $\nu$ dB/decade determined by the fractional order $\nu$, and the phase plot is nearly flat and close to $\frac{\pi}{2}\ \nu$ in the considered frequency range. The value of $N$ is determined by a trade-off between accuracy of the approximation in the frequency range of interest and simplicity of realization. Namely, the higher the desired reduction in the approximation error, the higher the required value of $N$ and the higher the implementation cost.

To improve the approximation, we here use Particle Swarm Optimization (PSO) that shows good convergence properties, reduced computational cost, and the ability to solve non-linear problems with multidimensional search spaces [24–27]. PSO is a stochastic evolutionary process imitating the behavior of a swarm of insects. Each particle is randomly defined in a search interval and represents a different solution for the set of zeros and poles that characterize the approximation. However, interlacing is enforced. The solutions stem from updating a population of particles in successive generations. In particular, PSO minimizes an objective function $J$ that is a measure of the approximation error. It starts with a random population which is then updated by considering the best values of $J$ obtained by each particle and the best values of $J$ obtained by the swarm. A constriction factor improves the performance of the algorithm [28]. To test the accuracy of the optimized discrete approximation, its frequency response is compared with the ones corresponding to other effective indirect discretization methods.

## 2. Two indirect approximation methods

This section revisits two indirect approximation methods that were proposed, analyzed, and discussed in [29,30,19]. These methods are used as term of comparison with the PSO-based method. The first indirect method comes from the classical Thiele's continued fraction (CF), and originates an approximation that is well-known in the control engineering field as Matsuda's approach [31]. The second method is based on another form of the Thiele's CF and was introduced by Maione [29] in approximating $s^\nu$. Since both methods guarantee a good continuous approximation, it is reasonable to discretize the obtained approximation by one of the available discretization operators (Tustin, backward Euler, Al-Alaoui, etc.) to obtain a good approximation for digital implementation of $s^\nu$. Then, we briefly recall the two methods in the following subsections.

### 2.1. The first Thiele's CF: Matsuda's method

Matsuda's method approximates the differintegrator $s^{\pm\nu}$, with $0 < \nu < 1$, by fitting a rational fraction to a set of magnitude values of $\omega^{\pm\nu}$. These values are obtained in assigned distinct and equally logarithmic spaced support points $\omega_k$, $k = 0, 1, \ldots, 2N$, that are chosen in the approximation interval. Then, the *inverse divided differences* method produces a CFE of the differintegrator. In particular, for a differentiator, putting $\varphi_0[\omega] = \omega^\nu$ and starting with $\varphi_1[\omega_0, \omega] = \frac{\omega-\omega_0}{\varphi_0[\omega]-\varphi_0[\omega_0]}$, an iterative procedure generates the successive inverse divided differences [13]:

$$\varphi_k[\omega_0, \ldots, \omega_{k-1}, \omega] = \frac{\omega - \omega_{k-1}}{\varphi_{k-1}[\omega_0, \ldots, \omega_{k-2}, \omega] - \varphi_{k-1}[\omega_0, \ldots, \omega_{k-1}]} \tag{2}$$

for $k = 2, \ldots, 2N$. If we start with $\varphi_0[\omega] = \varphi_0[\omega_0] + \frac{\omega-\omega_0}{\varphi_1[\omega_0,\omega]}$ and we rearrange (2), we obtain:

$$\varphi_{k-1}[\omega_0, \ldots, \omega_{k-2}, \omega] = \varphi_{k-1}[\omega_0, \ldots, \omega_{k-1}] + \frac{\omega - \omega_{k-1}}{\varphi_k[\omega_0, \ldots, \omega_{k-1}, \omega]} \tag{3}$$

**Table 1**
Zeros and poles of the first Thiele's CF-based continuous approximation of $s^{0.5}$.

| | | | | |
|---|---|---|---|---|
| Zeros of $G_{T1}(s)$ | −0.0191 | −0.2593 | −1.5483 | −11.1689 |
| Poles of $G_{T1}(s)$ | −0.0895 | −0.6459 | −3.8566 | −52.4416 |

hence the classical Thiele's CF:

$$F_{T1}(\omega) = \omega^{\nu} = d_0 + \frac{\omega - \omega_0|}{|d_1} + \frac{\omega - \omega_1|}{|d_2} + \cdots + \frac{\omega - \omega_k|}{|d_{k+1}} + \cdots \tag{4}$$

where $d_0 = \varphi_0[\omega_0] = \omega_0^{\nu}$, $d_1 = \varphi_1[\omega_0, \omega_1], \ldots, d_{2N} = \varphi_{2N}[\omega_0, \omega_1, \ldots, \omega_{2N-1}, \omega_{2N}]$. If $s$ replaces $\omega$ in (4) and truncation is made to $d_{2N}$, the convergent of $F_{T1}(s)$ leads to a rational transfer function $G_{T1}(s)$ that approximates $s^{\nu}$ so that, for $s = i\omega$ with $i = \sqrt{-1}$ imaginary unit, $|G_{T1}(i\,\omega)| = (\omega_k)^{\nu}$ at the sample points $\omega_k$, $k = 0, 1, \ldots, 2N$ (i.e. the magnitude approximation error is zero at these points).

Computational problems of inverse divided differences are determined by the ordering of sample points [32] and suggest us to use the so-called *reciprocal differences*. If $\Omega$ is the approximation interval, the reciprocal differences for $\omega^{\nu}$, for every $\omega \in \Omega$, are defined by $\rho_0[\omega_0] = \varphi_0[\omega_0] = \omega_0^{\nu}$, $\rho_1[\omega_0, \omega_1] = \varphi_1[\omega_0, \omega_1]$, and by:

$$\rho_k[\omega_0, \ldots, \omega_k] = \rho_{k-2}[\omega_0, \ldots, \omega_{k-2}] + \frac{\omega_k - \omega_{k-1}}{\rho_{k-1}[\omega_0, \ldots, \omega_{k-2}, \omega_k] - \rho_{k-1}[\omega_0, \ldots, \omega_{k-2}, \omega_{k-1}]} \tag{5}$$

for $k \geq 2$ and for $\omega_0, \ldots, \omega_k \in \Omega$. The reciprocal differences can be then used to compute the partial denominators in (4):

$$d_k = \varphi_k[\omega_0, \ldots, \omega_k] = \rho_k[\omega_0, \ldots, \omega_{k-1}, \omega_k] - \rho_{k-2}[\omega_0, \ldots, \omega_{k-2}]. \tag{6}$$

Then, considering $F_{T1}(s)$ and truncating (4) to $d_{2N}$ yields the convergent:

$$G_{T1}(s) = \frac{P_{T1}(s)}{Q_{T1}(s)} = \frac{p_{T1_{N,0}}\, s^N + \cdots + p_{T1_{N,N-1}}\, s + p_{T1_{N,N}}}{q_{T1_{N,0}}\, s^N + \cdots + q_{T1_{N,N-1}}\, s + q_{T1_{N,N}}} \tag{7}$$

where the numerator and denominator polynomials $P_{T1}(s)$ and $Q_{T1}(s)$ have the same degree $N$, then the same number of zeros and poles, and can be recurrently computed as first determined and shown in [29,21,19].

**Example 1.** Consider $\nu = 0.5$, then $s^{0.5}$, $\Omega = [0.1, 10]$ rad/s and $N = 4$. We use the following logarithmic distributed sample angular frequencies (in rad/s): {0.1000, 0.1778, 0.3162, 0.5623, 1.0000, 1.7783, 3.1623, 5.6234, 10.0000}. $F_{T1}(s)$ has following partial denominators: $d_0 = 0.3162$, $d_1 = 0.7379$, $d_2 = 0.9840$, $d_3 = 1.3122$, $d_4 = 1.7499$, $d_5 = 2.3335$, $d_6 = 3.1118$, $d_7 = 4.1497$, $d_8 = 5.5336$. Then, the fourth-order approximant of $F_{T1}(s)$ is:

$$G_{T1}(s) = \frac{s^4 + 12.9956\, s^3 + 20.8376\, s^2 + 4.8765\, s + 0.0855}{0.0855\, s^4 + 4.8764\, s^3 + 20.8373\, s^2 + 12.9955\, s + 1}.$$

The convergent has the zero–pole map of Table 1. Note the interlacing between minimum-phase zeros and stable poles. Moreover, the convergent yields the frequency response in Figs. 1 and 2 (see dashed curves). The zoom in Fig. 3 shows that a very good accuracy is obtained by the magnitude approximation in the frequency range $\Omega$, whereas deviations are more significant in the phase behavior, as shown in Fig. 4.

### 2.2. The second Thiele's CF

A second form of Thiele's CF was first used for the approximation of $s^{\nu}$ by Maione [29]. It is based on reciprocal differences and on a simple idea that slightly modifies the approach of the classical Thiele's CF. Namely, the previous approach is based on making the difference between the magnitude of the approximation, i.e. $|G_{T1}(i\,\omega)|$, and the magnitude of the fractional operator, i.e. $\omega^{\nu}$, vanish at the $2N + 1$ sample points. Now, we instead require that this difference and its first $2N$ derivatives vanish at a unique sample point $\omega_0$, to which all the sample points tend to coincide. Then, we obtain the following CFE:

$$F_{T2}(s) = c_0 + \frac{s - \omega_0|}{|c_1} + \frac{s - \omega_0|}{|c_2} + \cdots + \frac{s - \omega_0|}{|c_{k+1}} + \cdots \tag{8}$$

where $c_k = \varrho_k(\omega_0) - \varrho_{k-2}(\omega_0)$. Coefficients $c_k$ can be successively calculated by starting with the initial values $\varrho_{-2}(\omega) = \varrho_{-1}(\omega) = 0$, $\varrho_0(\omega) = \omega^{\nu}$, and $\varrho_1(\omega) = (\omega^{1-\nu})/\nu$, and by using, for $m \geq 1$, the relations [29]:

$$\varrho_{2m}(\omega) = \varrho_0(\omega) \prod_{i=1}^{m} \frac{i + \nu}{i - \nu} \tag{9}$$

$$\varrho_{2m+1}(\omega) = (m + 1)\, \varrho_1(\omega) \prod_{i=1}^{m} \frac{i + 1 - \nu}{i + \nu}. \tag{10}$$

**Table 2**
Zeros and poles of the second Thiele's CF-based continuous approximation of $s^{0.5}$.

| | | | | |
|---|---|---|---|---|
| Zeros of $G_{T2}(s)$ | −0.0311 | −0.3333 | −1.4203 | −7.5486 |
| Poles of $G_{T2}(s)$ | −0.1325 | −0.7041 | −3.0000 | −32.1634 |

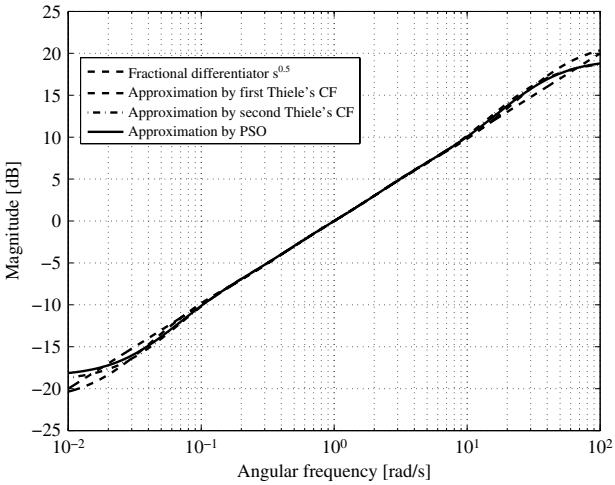


**Fig. 1.** Bode magnitude diagrams of the continuous approximations $G_{T1}(i\omega)$, $G_{T2}(i\omega)$, and $G_P(i\omega)$.

Truncation of $F_{T2}(s)$ to $c_{2N}$ gives the convergent that can be expressed as a rational transfer function:

$$G_{T2}(s) = \frac{P_{T2}(s)}{Q_{T2}(s)} = \frac{p_{T2_{N,0}}\, s^N + \cdots + p_{T2_{N,N-1}}\, s + p_{T2_{N,N}}}{q_{T2_{N,0}}\, s^N + \cdots + q_{T2_{N,N-1}}\, s + q_{T2_{N,N}}} \tag{11}$$

where the numerator and denominator polynomials $P_{T2}(s)$ and $Q_{T2}(s)$ have the same degree $N$ and can be recurrently computed [29,30,19].

**Example 2.** Consider again the approximation of $s^{0.5}$ in $\Omega = [0.1, 10]$ rad/s, with $N = 4$. If in (8) it is $\omega_0 = 1$ rad/s (i.e., if $\omega_0$ is placed at the center of the frequency interval $\Omega$), then: $c_0 = 1$, $c_1 = c_2 = c_3 = c_4 = c_5 = c_6 = c_7 = c_8 = 2$. The convergent $G_{T2}(s)$ giving the approximation as a ratio of fourth-order polynomials is:

$$G_{T2}(s) = \frac{s^4 + 9.3333\, s^3 + 14.0000\, s^2 + 4.0000\, s + 0.1111}{0.1111\, s^4 + 4.0000\, s^3 + 14.0000\, s^2 + 9.3333\, s + 1}.$$

The convergent has the zero–pole map of Table 2. Note again the interlacing between minimum-phase zeros and stable poles. The frequency response is depicted in Figs. 1 and 2 (see dash-dotted curves). The magnitude approximation has no significant deviation from the fractional operator (see zoom in Fig. 3). Instead, if we look at the phase diagram in Fig. 4, we may observe that the second Thiele's approximation performs better than the first around $\omega_0$, even if the first gives good results in a wider frequency range. This is obvious since the first convergent gives a null error in $(2N+1)$ sample frequencies, whereas the second convergent makes the error and its first $2N$ derivatives zero in $\omega_0$.

## 3. The particle swarm optimization method for approximation

The idea of PSO is to mimic the behavior of swarms of insects, ant colonies, schools of fishes, and similar biological organizations in reaching a solution minimizing a pre-defined objective function $J$. Each particle alone is not capable to find the solution, but the group as a whole may solve even complex problems. The solution is determined by the population of particles that agree on a common objective that each particle alone cannot achieve. Each particle behaves by applying simple rules derived from its own intelligence and, above all, from the swarm intelligence, based on the local information that is exchanged between particles. PSO minimizes the error index $J$ by updating randomly generated populations on the basis of the best results obtained by the particles and the entire swarm.

The optimization algorithm is iterative and each iteration evolves through successive phases, called *generations*. A stochastic evolutionary mechanism updates the results of each generation on the basis of the "position" and "velocity" variables, characterizing all particles in the swarm. Namely, given the (fixed) size of the population $N_{pop}$, i.e. the number of particles in the swarm, the transition from one generation to the next requires the variation of position and velocity of each particle that takes into account the best position of the particle through past generations (i.e. the *personal best* position) and the best position of the entire swarm (i.e. the *global best* position). The initial population of the first generation

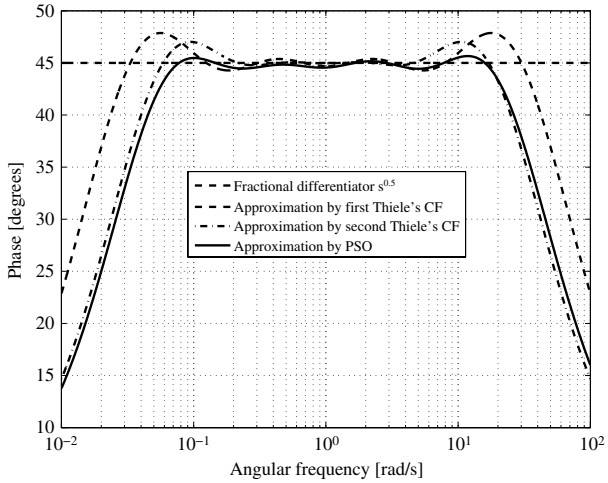


**Fig. 2.** Bode phase diagrams of the continuous approximations $G_{T1}(i\omega)$, $G_{T2}(i\omega)$, and $G_P(i\omega)$.

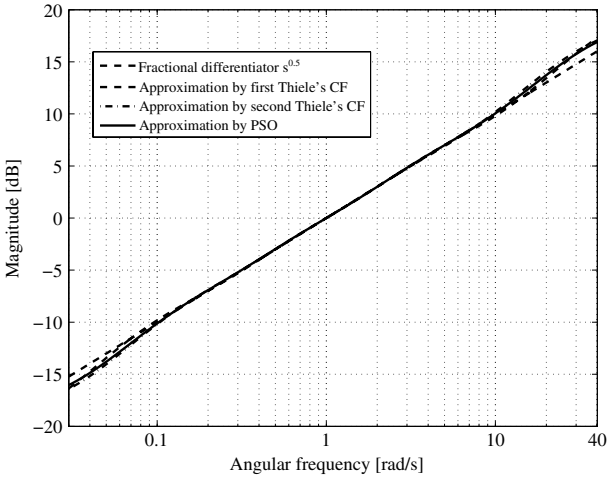


**Fig. 3.** Zoom of the magnitude diagrams of the continuous approximations.

is randomly generated. Moreover, in all generations of the iterated algorithm, each particle is randomly defined within a search interval that depends on the specific problem. In this paper, we approximate the fractional order operator by exploiting previous results of other methods. Namely, we find better values for the interlaced zeros and poles of the rational approximation. Therefore, other methods define the search intervals for the unknown better zeros and poles. In particular, let $N$ be the order of the approximation, and $\mu_{k,\min}$, $\mu_{k,\max}$, $\lambda_{k,\min}$, and $\lambda_{k,\max}$ be the minimum and maximum values for the $k$-th zero and $k$-th pole, for $k = 1, \ldots, N$, of other approximations. Then, the associated search intervals can be defined as $[\mu_{k,\min}, \mu_{k,\max}]$ and $[\lambda_{k,\min}, \lambda_{k,\max}]$. With reference to Tables 1 and 2, we may use $[-0.0311, -0.0191]$ for $\mu_1$, $[-0.3333, -0.2593]$ for $\mu_2$, $[-1.5483, -1.4203]$ for $\mu_3$, $[-11.1689, -7.5486]$ for $\mu_4$, $[-0.1325, -0.0895]$ for $\lambda_1$, $[-0.7041, -0.6459]$ for $\lambda_2$, $[-3.8566, -3.0000]$ for $\lambda_3$, $[-52.4416, -32.1634]$ for $\lambda_4$. The search intervals do not overlap, so the interlacing is guaranteed.

Based on this mechanism, the PSO algorithm stops the iterations when it achieves the optimum solution or when it reaches the maximum number of generations in the last iteration. The algorithm shows fast convergence and stability, at a reduced computational cost, and it is sensitive to the values of few parameters to be adjusted but not to the population size [26,33]. In particular, the population size $N_{pop}$ results from a trade-off between the required computation time and the desired speed of convergence (with a reduced approximation error). Clearly, a big swarm implies a high computational effort, a small swarm may not or may slowly converge. Moreover, the number $G_{\max}$ of generations and the number $N_{runs}$ of iterations of the algorithm determine the computation time and the accuracy of the optimization. Obviously, the three parameters must be properly tuned.

Among the existing versions of PSO algorithms, two are popular and commonly used; the former is the standard PSO based on a weighting factor (PSOWF); the latter is an improved version based on a constriction factor (PSOCF) [24,28]. Both

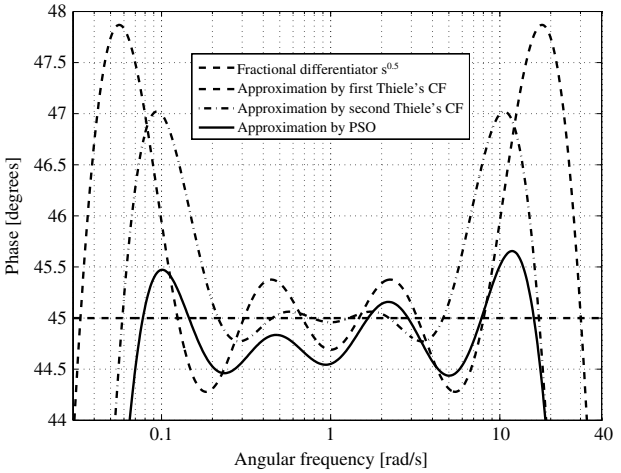


**Fig. 4.** Zoom of the phase diagrams of the continuous approximations.

PSOWF and PSOCF update the position of each particle by using the past position and current velocity:

$$x_j(g) = x_j(g-1) + v_j(g) \tag{12}$$

for $j = 1, \ldots, N_{pop}$ and $g = 1, \ldots, G_{\max}$. The initial velocity is zero and the initial position is randomly generated from a uniform distribution in the search intervals: $x_j(1) \sim U(\mu_{k,\min}, \mu_{k,\max})$ for the $k$-th zero and $x_j(1) \sim U(\lambda_{k,\min}, \lambda_{k,\max})$ for the $k$-th pole.

The PSOWF updates velocity as follows:

$$v_j(g) = \theta\, v_j(g-1) + \gamma_1\, r_1\, [x_{pbest}(g) - x_j(g-1)] + \gamma_2\, r_2\, [x_{gbest}(g) - x_j(g-1)] \tag{13}$$

where $0 < \theta < 1$ is the inertial weighting factor applied to the previous velocity; $0 < \gamma_1 < 2$ is the individual learning rate; $0 < \gamma_2 < 2$ is the social learning rate; $r_1 \sim U(0, 1)$ and $r_2 \sim U(0, 1)$ are random numbers from a uniform distribution; $x_{pbest}(g)$ is the personal best remembered individual particle position so far, and $x_{gbest}(g)$ is the global best remembered swarm position of all particles until the current generation. The $x_{pbest}(g)$ is associated to the minimum value of the objective function $J$ to optimize, while the $x_{gbest}(g)$ results from $x_{gbest}(g) = \min\{x_{gbest}(g-1), x_{pbest}(g)\}$. Then, it is important to define $J$. Note that $\theta$ reduces the variation of velocity, to avoid missing of optimal solutions due to fast increments. A high value of $\theta$ helps the global exploration, a low value the local search.

In a slightly different way, the PSOCF works as follows:

$$v_j(g) = \xi(\gamma_1, \gamma_2)\, \{v_j(g-1) + \gamma_1\, r_1\, [x_{pbest}(g) - x_j(g-1)] + \gamma_2\, r_2\, [x_{gbest}(g) - x_j(g-1)]\} \tag{14}$$

where $\xi(\gamma_1, \gamma_2)$ is the constriction factor that is determined by $\xi(\gamma_1, \gamma_2) = \frac{2}{2-\gamma-\sqrt{\gamma^2-4\gamma}}$, with $\gamma = \gamma_1 + \gamma_2 > 4$. Clearly, the parameter values are required to minimize the cost $J$ ensuring the best rational approximant of $s^{\nu}$.

To synthesize, to efficiently apply PSOCF, we have to choose the search interval of the solutions and fix the values of parameters that ensure good performance of the algorithm. However, without restrictions, the algorithm could lead to high approximation errors. Hence, we restrict the search interval by enforcing the interlacing. To this aim, we take, as reference for each zero and each pole of the approximating transfer function $G_P(s)$, the minimum and maximum values obtained by two forms of the Thiele's CF. These approximation techniques gave very good results in recent papers [29,30,19]. The gain of $G_P(s)$

$$G_P(s) = k_{PSO} \prod_{k=1}^{N} \frac{s - \mu_k}{s - \lambda_k} \tag{15}$$

results from imposing unitary gain crossover angular frequency, i.e. the same as with the fractional order operator:

$$|G_P(i\,1)| = 1 \Rightarrow k_{PSO} = \sqrt{\prod_{k=1}^{N} \frac{(1+\lambda_k^2)}{(1+\mu_k^2)}}. \tag{16}$$

Moreover, we choose $\gamma_1 = \gamma_2 = 2.05$ (then $\xi = 0.7298$) that are commonly accepted values for good performance of PSOCF. Finally, $N_{runs}$, $G_{\max}$, and $N_{pop}$ are fixed by trial-and-error empirical tuning.

Finally, we specify the objective function $J$ as a function of the amplitude and phase approximation errors between $G_P(i\,\omega)$ and the fractional operator $(i\,\omega)^{0.5}$. In particular, if $E_{m,l}$ (in dB) and $E_{p,l}$ (in degrees) indicate the magnitude and phase errors

computed in a set of angular frequencies $\omega_l$, $l = 1, \ldots, \Omega$, inside the range of interest, we define:

$$J = \min\{\max\{|E_{m,l}| \cdot P(\omega_l)\} + \max\{|E_{p,l}| \cdot P(\omega_l)\}\} \tag{17}$$

where $P(\omega)$ penalizes errors inside the range of interest, whereas it lowly weights errors outside this range. We may assume $P(\omega) = 1$ for $0.1 \leq \omega \leq 10$ rad/s and $P(\omega) = e^{-|\log_{10}(\omega)|^2}$ for $\omega \leq 0.1$ rad/s or $\omega \geq 10$ rad/s.

To sum up, each particle in a population of the PSOCF (i.e. each set of poles and zeros of an $N$-order approximating rational transfer function) is characterized by a different frequency response, different errors, then a different value of $J$. The minimum value of the error index $J$ in all iterations of the PSOCF algorithm will identify the best solution found by the algorithm, i.e. the best set of zeros and poles in the approximation.

**Example 3.** Consider the PSOCF-based approximation of $s^{0.5}$ of order $N = 4$. We run the algorithm with $N_{runs} = 20$, $G_{\max} = 30$, and $N_{pop} = 20$. These values are obtained after several tests to minimize $J$, then to optimize the frequency response. The search intervals for zeros and poles are specified as previously indicated, by using zeros and poles of $G_{T1}(s)$ and $G_{T2}(s)$. To have an indication of the PSOCF algorithm convergence, we compute the average value of $J$ over all generations in each iteration and consider the lowest average among all iterations. The iteration that yields the lowest average may be considered, to a certain extent, as the one giving the best trend of $J$ towards the minimum in all simulation runs. Fig. 5 plots the behavior of $J$ in this iteration (the reported statistics are associated to this particular iteration). The minimum value of $J$ in all generations of all iterations is $J_{\min} = 0.6336$. The associated set of zeros and poles gives the best solution as:

$$G_P(s) = 9.2165\, \frac{(s + 0.0325)(s + 0.3063)(s + 1.5201)(s + 9.1827)}{(s + 0.1244)(s + 0.6770)(s + 3.5556)(s + 36.0483)}.$$

Stability, minimum-phaseness, and interlacing are again obtained. The frequency response is shown in Figs. 1 and 2 (see solid curves). As expected, the second Thiele's approximation is better than the first around the sample point $\omega_0$, whereas the first performs better in a wider range because it guarantees a null magnitude approximation error in $2N + 1$ sample points. Note that the PSO approximation gives a flatter phased diagram, with reduced peak values with respect to the two Thiele's approximations. The absolute error is lower than 0.5° between 0.07 and 10 rad/s, in a bit wider frequency range than those achieved by the other two approximations (see Fig. 4). This result is because the PSOCF reduces the error index $J$.

## 4. Discretization

To obtain a discrete approximation, the final step of indirect discretization methods is to apply a transformation rule from the $s$-domain to the $z$-domain. Here, we adopt Tustin's rule for sake of brevity, but other mapping rules can be also considered. Applying the discretization rule $s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$ to the convergents $G_{T1}(s)$, $G_{T2}(s)$, and $G_P(s)$, expressed in the form:

$$G(s) = \frac{P(s)}{Q(s)} = \frac{p_{N,0}\, s^N + p_{N,1}\, s^{N-1} + \cdots + p_{N,N-1}\, s + p_{N,N}}{q_{N,0}\, s^N + q_{N,1}\, s^{N-1} + \cdots + q_{N,N-1}\, s + q_{N,N}} \tag{18}$$

provides us with the corresponding discretized convergents:

$$\overline{G}(z) = \frac{P(z)}{Q(z)} = \frac{\overline{p}_{N,0}\, z^N + \overline{p}_{N,1}\, z^{N-1} + \cdots + \overline{p}_{N,N-1}\, z + \overline{p}_{N,N}}{\overline{q}_{N,0}\, z^N + \overline{q}_{N,1}\, z^{N-1} + \cdots + \overline{q}_{N,N-1}\, z + \overline{q}_{N,N}} \tag{19}$$

where the coefficients $\overline{p}_{N,j}$ and $\overline{q}_{N,j}$, with $j = 0, \ldots, N$, are related to the corresponding $p_{N,j}$ and $q_{N,j}$. Namely, if we put $\alpha_0 = 2/T$ and $\alpha_1 = -2/T$, it is easy to find closed formulas. For example, for $N = 4$:

$$\begin{aligned}
\overline{p}_{4,0} &= \alpha_0^4\, p_{4,0} + \alpha_0^3\, p_{4,1} + \alpha_0^2\, p_{4,2} + \alpha_0\, p_{4,3} + p_{4,4} \\
\overline{p}_{4,1} &= 4\,\alpha_0^3\,\alpha_1\, p_{4,0} + (\alpha_0^3 + 3\,\alpha_0^2\,\alpha_1)\, p_{4,1} + (2\,\alpha_0^2 + 2\,\alpha_0\,\alpha_1)\, p_{4,2} + (3\,\alpha_0 + \alpha_1)\, p_{4,3} + 4\, p_{4,4} \\
\overline{p}_{4,2} &= 6\,\alpha_0^2\,\alpha_1^2\, p_{4,0} + (3\,\alpha_0^2\,\alpha_1 + 3\,\alpha_0\,\alpha_1^2)\, p_{4,1} + (\alpha_0^2 + 4\,\alpha_0\,\alpha_1 + \alpha_1^2)\, p_{4,2} + (3\,\alpha_0 + 3\,\alpha_1)\, p_{4,3} + 6\, p_{4,4} \\
\overline{p}_{4,3} &= 4\,\alpha_0\,\alpha_1^3\, p_{4,0} + (3\,\alpha_0\,\alpha_1^2 + \alpha_1^3)\, p_{4,1} + (2\,\alpha_0\,\alpha_1 + 2\,\alpha_1^2)\, p_{4,2} + (\alpha_0 + 3\,\alpha_1)\, p_{4,3} + 4\, p_{4,4} \\
\overline{p}_{4,4} &= \alpha_1^4\, p_{4,0} + \alpha_1^3\, p_{4,1} + \alpha_1^2\, p_{4,2} + \alpha_1\, p_{4,3} + p_{4,4}.
\end{aligned}$$

Analogous relations hold for $\overline{q}_{4,j}$, for $j = 0, 1, 2, 3, 4$.

Now, we compare the frequency responses of the discretized approximations that are obtained from the three continuous approximations. We use a sampling period $T = 0.5$ s, which is a common value for practical applications in the field of digital controlled industrial plants. Then, putting $z = e^{i\omega T}$ in $\overline{G}_{T1}(z)$, $\overline{G}_{T2}(z)$, and $\overline{G}_P(z)$ gives the frequency responses of the discretized approximations.

Fig. 6 shows the Bode phase diagram, the magnitude diagram is not shown because differences between discrete and continuous approximations are negligible. All the discretized approximations provide good results until frequency $\pi/T$. Responses of discretized approximations depart from those of the continuous approximations if the frequency increases, roughly from 2 rad/s (see Fig. 7). All the remarks made for the continuous approximations still hold true, then we may conclude that PSO improves the approximation, in particular the flatness of the phase diagram.

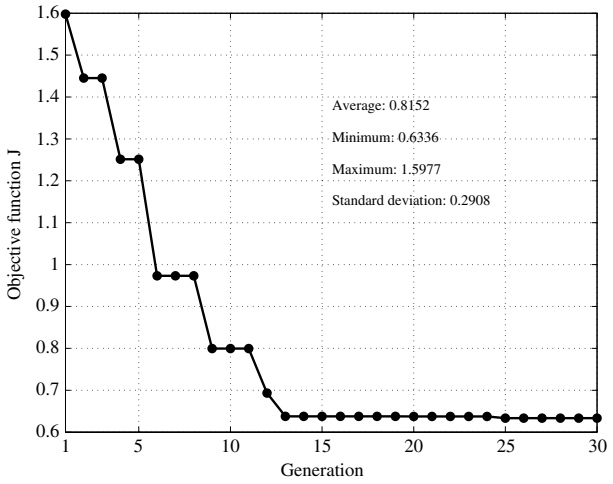


**Fig. 5.** Behavior of $J$ in the iteration with lowest average.

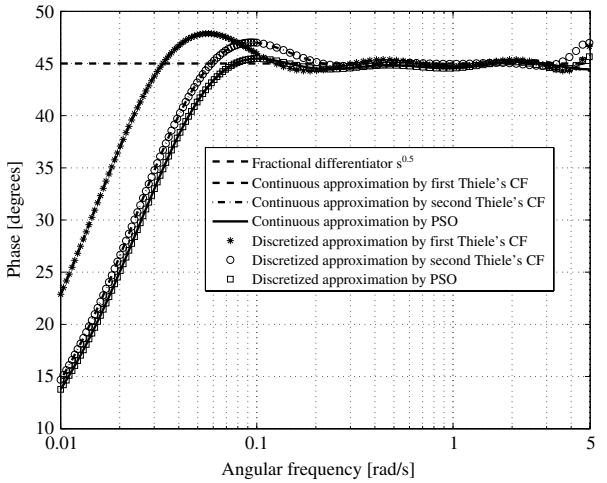


**Fig. 6.** Bode phase diagrams of continuous and discretized approximations.

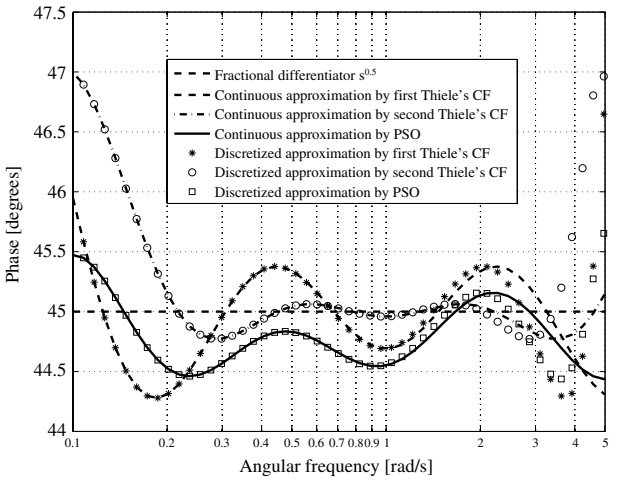


**Fig. 7.** Zoom of phase diagrams of continuous and discretized approximations.

## 5. Concluding remarks

In this paper, we apply PSO to optimize two analog efficient approximations of the fractional order operator $s^{\nu}$, the first based on the classical Thiele's CF, the second based on a new form of Thiele's CF. The aim is to obtain an even better approximation, especially in the phase behavior of the frequency response. After that, the approximation is discretized to obtain a better performance than those guaranteed by the discretizations of the two Thiele's CF-based approximations. Results show that the phase behavior is improved, which is very important for fractional order control applications requiring fractal robustness to load and gain variations. Moreover, the second Thiele's approximation gives the best results in a narrow range around the central frequency $\omega_0$, which is expected since it is based on the vanishing of the approximation error and of its derivatives in $\omega_0$. But the PSO guarantees flatness in a wider range.

## Acknowledgment

This work is supported by the Italian Ministry of University and Research (MIUR) under project *"Non integer order systems in modeling and control"*, grant no. 2009F4NZJP.

## References

[1] I. Podlubny, Fractional-order systems and $PI^{\lambda}D^{\mu}$-controllers, IEEE Trans. Autom. Cont. 44 (1) (1999) 208–214.
[2] I. Podlubny, L. Dorcák, I. Kostial, On fractional derivatives, fractional-order dynamic systems and $PI^{\lambda}D^{\mu}$ controllers, in: Proc. of the 36th Conf. on Decision and Control, San Diego (CA), USA, 10–12 Dec. 1997, pp. 4985–4990.
[3] Y.-Q. Chen, Ubiquitous fractional order controls? in: Proc. of the Second IFAC Symposium on Fractional Derivatives and its Applications (IFAC FDA'06), Vol. 2, Porto, Portugal, 2006, pp. 168–173.
[4] K.-J. Åström, T. Hägglund, in: Instrument Society of America (Ed.), PID Controllers: Theory, Design, and Tuning, second ed., Research Triangle Park, NC, 1995.
[5] R. Caponetto, L. Fortuna, D. Porto, A new tuning strategy for non-integer order PID controller, in: Proc. 1st IFAC Worksh. Fractional Differentiation and its Application, Bordeaux, France, 2004, pp. 168–173.
[6] A. Pisano, M. Rapaić, Z. Jeličìč, E. Usai, Nonlinear fractional PI control of a class of fractional-order systems, in: IFAC Conference on Advances in PID Control (PID'12), Brescia, Italy, 2012.
[7] P. Lino, G. Maione, Loop-shaping and easy tuning of fractional-order proportional integral controllers for position servo systems, Asian J. Control (2012) http://dx.doi.org/10.1002/asjc.556.
[8] G. Maione, High-speed digital realizations of fractional operators in the delta domain, IEEE Trans. Autom. Cont. 56 (3) (2011) 697–702.
[9] S. Samadi, M.O. Ahmad, M.N.S. Swamy, Exact fractional-order differentiators for polynomial signals, IEEE Signal Process. Lett. 11 (6) (2004) 529–532.
[10] C.-C. Tseng, Design of fractional order digital FIR differentiators, IEEE Signal Process. Lett. 8 (3) (2001) 77–79.
[11] Y.Q. Chen, B.M. Vinagre, A new IIR-type digital fractional order differentiator, Signal Process. 83 (2003) 2359–2365.
[12] B.M. Vinagre, Y.Q. Chen, I. Petráš, Two direct Tustin discretization methods for fractional-order differentiator/integrator, J. Frankl. Inst. 340 (5) (2003) 349–362.
[13] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, third ed., Springer-Verlag, Berlin, New York, 2002.
[14] Y.Q. Chen, K.L. Moore, Discretization schemes for fractional-order differentiators and integrators, IEEE Trans. Circuits Syst. I, Fundam. Theory Appl. 49 (3) (2002) 363–367.
[15] J.A. Tenreiro Machado, Analysis and design of fractional-order digital control systems, J. Systems Anal. Model. Simulation 27 (1997) 107–122.
[16] Y.Q. Chen, B.M. Vinagre, I. Podlubny, Continued fraction expansions approaches to discretizing fractional order derivatives. An expository review, Nonlinear Dyn. 38 (1–2) (2004) 155–170.
[17] B.M. Vinagre, I. Podlubny, A. Hernandez, V. Feliu, Some approximations of fractional order operators used in control theory and applications, Fractional Calculus Appl. Anal. 3 (3) (2000) 231–248.
[18] I. Podlubny, I. Petráš, B.M. Vinagre, P. O'Leary, L. Dorcák, Analogue realizations of fractional-order controllers, Nonlinear Dyn. 29 (1–4) (2002) 281–296.
[19] G. Maione, Thiele's continued fractions in digital implementation of noninteger differintegrators, Signal, Image Video Process. 6 (3) (2012) 401–410.
[20] A. Oustaloup, La commande CRONE. Commande Robuste d'Ordre Non Entièr, Editions Hermès, Paris, 1991.
[21] G. Maione, Conditions for a class of rational approximants of fractional differentiators/integrators to enjoy the interlacing property, in: Proc. 18th IFAC World Congr., Milan, Italy, 2011, pp. 13984–13989.
[22] A.N. Khovanskii, Continued fractions, in: L.A. Lyusternik, A.R. Yanpol'skii (Eds.), Mathematical Analysis—Functions, Limits, Series, Continued Fractions (Transl. by D.E. Brown), Pergamon Press, Oxford, 1965, p. 241.
[23] F.-R. Gantmacher, M.-G. Krein, Oscillation Matrices and Kernels and Small Vibrations of Mechanical Systems, revised ed., AMS Chelsea Publishing, American Mathematical Society, Providence, RI, 2002.
[24] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Y. Attikiouzel (Ed.), Proc. 4th IEEE Int. Conf. Neural Networks, Vol. 4, IEEE Press, Perth, Australia, 1995, pp. 1942–1948.
[25] Y. Shi, R.C. Eberhart, A modified particle swarm optimizer, in: Proc. IEEE World Congr. on Computational Intellig., The 1998 IEEE Int. Conf. Evolut. Comput., Anchorage (AK), USA, 1998, pp. 69–73.
[26] Y. Shi, R.C. Eberhart, Empirical study of particle swarm optimization, in: Proc. IEEE 1999 Congr. Evolut. Comput., vol. 3, Washington (DC), USA, 6–9 July 1999, pp. 1945–1950.
[27] M. Clerc, J. Kennedy, The particle swarm—explosion, stability, and convergence in a multidimensional complex space, IEEE Trans. Evolutionary Computation 6 (1) (2002) 58–73.
[28] R.C. Eberhart, Y. Shi, Comparing inertia weights and constriction factors in particle swarm optimization, in: Proc. IEEE Int. Congr. Evolut. Comput., vol. 1, San Diego (CA), USA, 16–19 July 2000, pp. 84–88.
[29] G. Maione, Approximation of the fractional operator $s^{\nu}$ using Thiele's continued fractions, in: Proc. 4th IFAC Worksh. on Fractional Differentiation and Its Applications, Badajoz, Spain, 2010, pp. 428–433.
[30] G. Maione, Discretizing fractional order operators approximated by Thiele's continued fractions, in: Proc. 7th Eur. Nonlin. Dyn. Conf, Rome, Italy, 24–29 July 2011.
[31] K. Matsuda, H. Fujii, $H_{\infty}$ optimized wave-absorbing control: analytical and experimental results, J. Guid. Control Dyn. 16 (6) (1993) 1146–1153.
[32] F.B. Hildebrand, Introduction to Numerical Analysis, second ed., Dover Publication Inc., New York, 1974.
[33] F. van den Bergh, A.P. Engelbrecht, A study of particle swarm optimization particle trajectories, Inform. Sci. 176 (2006) 937–971.