CrossMark

Efficient Design of Discrete Fractional-Order Differentiators Using Nelder–Mead Simplex Algorithm

K. P. S. Rana[1] · **Vineet Kumar[1]** · **Yashika Garg[1]** ·
Sreejith S. Nair[1]

Abstract Applications of fractional-order operators are growing rapidly in various branches of science and engineering as fractional-order calculus realistically represents the complex real-world phenomena in contrast to the integer-order calculus. This paper presents a novel method to design fractional-order differentiator (FOD) operators through optimization using Nelder–Mead simplex algorithm (NMSA). For direct discretization, Al-Alaoui operator has been used. The numerator and the denominator terms of the resulting transfer function are further expanded using binomial expansion to a required order. The coefficients of z-terms in the binomial expansions are used as the starting solutions for the NMSA, and optimization is performed for a minimum magnitude root-mean-square error between the ideal and the proposed operator magnitude responses. To demonstrate the performance of the proposed technique, six simulation examples for fractional orders of half, one-third, and one-fourth, each approximated to third and fifth orders, have been presented. Significantly improved magnitude responses have been obtained as compared to the published literature, thereby making the proposed method a promising candidate for the design of discrete FOD operators.

✉ K. P. S. Rana
  kpsrana1@gmail.com

  Vineet Kumar
  vineetkumar27@gmail.com

  Yashika Garg
  yashika93@nsitonline.in

  Sreejith S. Nair
  sreejith336@gmail.com

[1] Instrumentation and Control Engineering, Netaji Subhas Institute of Technology,
  Dwarka, New Delhi 110078, India

Birkhäuser

1 Introduction

The concept of fractional calculus is being used ever since its mention in the letters
exchanged between L'Hospital and Leibniz in the year 1695. Fractional calculus, in a
sense, is generalization of the integer-order calculus. As fractional calculus describes
the characteristics of most of the natural systems more realistically, a renewed interest
among the researchers has grown in the last three decades and the fractional calculus
has found applications in numerous science and engineering disciplines [8]. Appli-
cations of fractional calculus have been explored in large number of domains such
as control, signal processing, chaos theory, mechanics, physics, chemistry, electri-
cal circuit with fractance, generalized voltage divider, viscoelasticity, fractional-order
multipoles in electromagnetism, electrochemistry [4,9,10,13,17,23].

Signal processing and control engineering are the two major thrust areas where
fractional-order system applications have shown major developments. For example,
fractional-order signal processing has been used at times to estimate water level eleva-
tions of north part of the Great Salt Lake, since the data distribution was heavy-tailed,
i.e., needed large memory [20,21]. Also, in image processing, fractional-order differ-
entiator helped to improve the criterion of thin detection and immunity to noise by
extending CRONE detector to fractional order. It may be noted that earlier design used
integer-order operator [16]. Further, a comparison for modeling of speech signal using
linear predictive coding approach based on integer-order and fractional-order models
has also been reported, and it was demonstrated that fractional-order approach models
speech signal more accurately [1].

In control engineering, the first choice of a control engineer is always the con-
ventional proportional–integral–derivative (PID) controller. Making the integration
and derivative terms fractional offers two more degrees of freedom to the con-
trol design engineer. Such controllers are termed as fractional PID controllers. The
fractional PID controllers can be efficiently designed and tuned for a desired perfor-
mance even for a complex system as compared to their conventional integer-order
counterparts. Fractional-order PID controllers applied to path tracking problems in
autonomous vehicles have demonstrated better results [22]. In another recent research,
a fractional-order fuzzy PID controller for a two-link rigid robotic manipulator
has been investigated for path tracking problem. The performance of the proposed
fractional-order fuzzy PID controller was found to be superior to the fuzzy PID con-
troller for path tracking, disturbance rejection, and noise suppression applications
[19].

Despite the fact that fractional-order calculus finds its applications in numerous
domains, even today, its implementation poses challenges. In modeling and analysis
of systems using fractional calculus, fractional-order derivative (FOD) operator plays
a key role and hence an efficient implementation of FOD operator is essential. Fur-
ther, as this operator is not a standard of-the-shelf operator supplied in the various

commercially available modeling and analysis tools, its effect is implemented using an equivalent standard integer-order transfer function.

One of the well-known, time-domain definitions of FOD operator is a Grünwald–Letnikov's as defined below [6,12].

$$\text{Grünwald−Letnikov's: } D^n f(t) = \frac{d^n f(t)}{dt^n}$$

$$= \lim_{h \to 0} \frac{1}{h^n} \sum_{k=0}^{n} (-1)^n \binom{n}{k} f(t-kh); \quad n \epsilon R^+$$

(1)

where

$$\binom{n}{k} = \frac{n(n-1)(n-2)\cdots(n-k+1)}{k!}$$

FOD operators can also be used to describe the fractional-order transfer function in frequency domain as in [6].

$$G(s) = \frac{Y(s)}{X(s)} = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + b_{m-2} s^{\beta_{m-2}} \cdots + b_0 s^{\beta_0}}{a_m s^{\alpha_m} + a_{m-1} s^{\alpha_{m-1}} + a_{m-2} s^{\alpha_{m-2}} \cdots + a_0 s^{\alpha_0}}; \quad \beta, \alpha \epsilon R \quad (2)$$

Equivalently, the transfer function in z-domain can be given by

$$G(z) = \frac{b_m (\psi(z^{-1}))^{\beta_m} + b_{m-1} (\psi(z^{-1}))^{\beta_{m-1}} + b_{m-2} (\psi(z^{-1}))^{\beta_{m-2}} \cdots + b_0 (\psi(z^{-1}))^{\beta_0}}{a_m (\psi(z^{-1}))^{\alpha_m} + a_{m-1} (\psi(z^{-1}))^{\alpha_{m-1}} + a_{m-2} (\psi(z^{-1}))^{\alpha_{m-2}} \cdots + a_0 (\psi(z^{-1}))^{\alpha_0}};$$
$$\beta, \alpha \epsilon R \quad (3)$$

where ψ is a mapping function from s-domain to z-domain.

In order to represent an equivalent fractional-order operator using the integral powers, various power series such as Taylor's, Maclaurin, continued fraction expansion (CFE), binomial are normally used. Essentially, these implementations, in an ideal sense, require infinite memory. Therefore, in order to implement FOD operators, theoretically, powers of z-terms range from 0 to ∞. Since it is impractical to implement infinite number of terms on any computing hardware, as it suffers from space and time complexity, truncation of the series becomes an inherent important consideration. Usually, the higher the number of terms or the order considered, the better the resulting transfer function approximates an ideal analog transfer function. At the same time, the higher the order of transfer function, the higher is the implementation complexity, thus requiring more resources. Therefore, researchers always search for an alternative efficient way to approximate the ideal FOD operator with an equivalent lower-order discrete transfer functions having higher accuracy, i.e., frequency response be as close as possible to the ideal operator and the order of the transfer function be as low as possible.

In designing and implementation of discrete FODs, the key step is the discretization of the fractional-order differentiator (s^α). Discretization refers to a mapping

from s-domain to z-domain. It can be of two types: direct discretization and indirect discretization [28]. Indirect discretization methods involve two steps, i.e., frequency domain fitting in continuous time domain followed by discretization of the so obtained continuous time transfer function using a mapping function, whereas direct discretization methods include the application of power series expansion of Euler operator, CFE of the Tustin operator, and numerical integration-based methods. The major drawback of using these expansion techniques is that they approximate transfer function in a limited frequency range only. In [3,5,11,24], s^α has been discretized using direct discretization schemes, whereas in [15], indirect discretization schemes have been implemented. In [25], modification of the Schneider and Al-Alaoui–Schneider–Kaneshige–Groutage rule has been presented for the improvement of the fractional-order differentiator in the low frequency range. In the design of discrete FODs, a recent trend has been to explore the applications of optimization techniques such as particle swarm optimization (PSO) and genetic algorithm (GA) [7]. Das et al. have expanded various discretization operators using CFE up to the desired order followed by optimization of resulting coefficients so as to minimize a predefined error function.

In this work, the proposed technique makes use of direct discretization method followed by optimization of the resulting transfer function. For discretization of the fractional-order operator, Al-Alaoui operator has been used. The numerator and the denominator components of the resulting discrete time transfer function have been further expanded using binomial expansion for the fractional power. The coefficients of transfer function thus obtained are further re-tuned using Nelder–Mead simplex algorithm (NMSA) optimization method so as to minimize root-mean-square (RMS) error in magnitude response. It may be noted that the NMSA requires a starting solution. In the present work, the expanded binomial expansion coefficients of the z-terms, for numerator as well as the denominator, are used as the starting solutions. The obtained operators, optimized using NMSA, have offered better approximations of fractional-order differentiators in all the investigated cases. Significantly enhanced performance has been demonstrated as compared to the published literature, thereby making the proposed method a promising candidate for design of discrete FODs.

This paper is organized as follows. Following a brief literature survey in Sect. 1, various discretization schemes including Al-Alaoui operator are summarized in Sect. 2. Generic discrete transfer functions for fractional-order operators of various powers have also been derived using Al-Alaoui operator and have been presented in this section. The used optimization technique, NMSA, is then described in Sect. 3 with the help of a flowchart and a supporting numerical example. Using Al-Alaoui operator for discretization, transfer functions of various orders for different powers are then obtained for binomially expanded limited terms FODs. NMSA optimization is further utilized to tune the coefficients of the resulting differentiators for minimum magnitude response RMS error. Designs of optimized differentiators are presented in Sect. 4 along with frequency responses and magnitude and phase error v/s frequency plots. Variations of fitness function v/s iteration plots are also presented to show the convergence of NMSA optimization process. Section 5 presents performance comparisons of designed differentiators with existing differentiators of same power and approximated to same order as reported in the recent literature. Frequency responses and

the magnitude RMS errors have been compared, and the results are presented in this section. Further, a comparison of the optimization performance of NMSA with GA is also carried out for a half differentiator of third order. Comparison results in terms of timing aspect and achievable fitness function values along with statistical t test analysis of 50 independent runs have been presented in Sect. 6. The paper is finally concluded in Sect. 7.

2 Discretization Using Al-Alaoui Operator

Frequency response of an ideal fractional-order differentiator (s^α), of order α, is given by

$$H(j\omega) = (j\omega)^\alpha \tag{4}$$

where $j = \sqrt{-1}$, ω is the frequency in radians/s, and α is the fractional power of the differentiator under consideration. Frequency response can also be written as,

$$H(j\omega) = e^{j\alpha\frac{\pi}{2}}\omega^\alpha$$

Magnitude and phase responses, from the above function, can be given as,

$$|H(j\omega)| = \omega^\alpha \tag{5}$$

$$\Phi = \alpha 90° \tag{6}$$

Figure 1 illustrates frequency response up to 500 Hz of an ideal half-order ($\alpha = 0.5$) differentiator. A constant phase of 45° can be clearly observed for this differentiator.

Mapping functions are the transformations used to convert a transfer function from s-domain to z-domain. There are many mapping functions available in the literature, and some important ones are given below:

$$\text{Euler:} \ \frac{1}{s} = \frac{Tz}{z-1} \tag{7}$$

$$\text{Tustin:} \ \frac{1}{s} = \frac{T}{2}\frac{z+1}{z-1} \tag{8}$$

$$\text{Schneider:} \ \frac{1}{s} = \frac{T}{12}\frac{5z^2 + 8z - 1}{z(z-1)} \tag{9}$$

$$\text{Simpson:} \ \frac{1}{s} = \frac{T}{3}\frac{z^2 + 4z + 1}{z^2 - 1} \tag{10}$$

$$\text{F012:} \ \frac{1}{s} = \frac{2T}{5}\frac{z(z+2)}{5z^2 - 4z - 1} \tag{11}$$

$$\text{Rectangular:} \ \frac{1}{s} = \frac{T}{z-1} \tag{12}$$

Mapping function used in this work, developed by Al-Alaoui, is described by (13). According to this mapping function, fractional-order integrator is defined as follows.

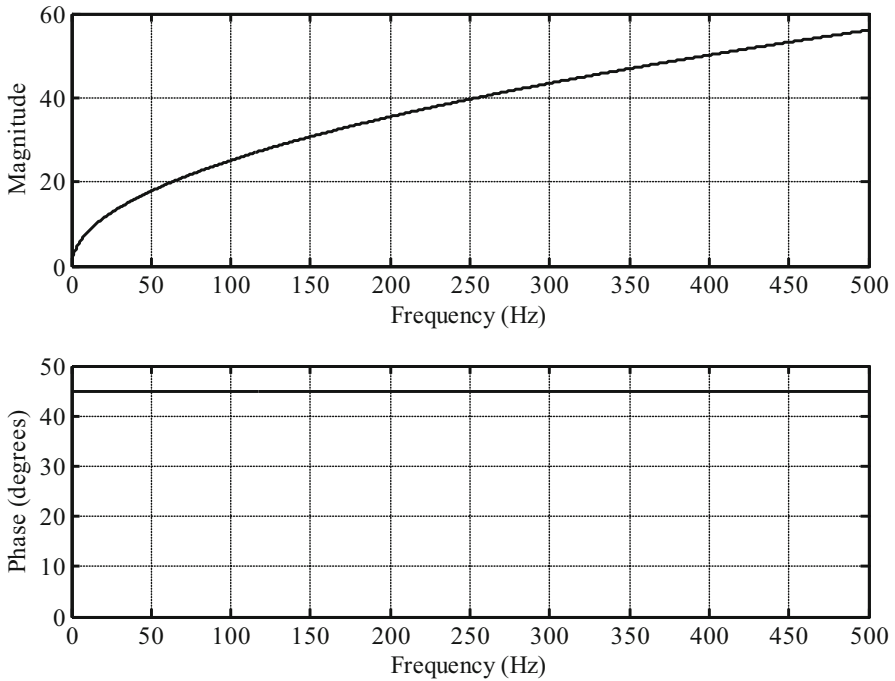$$H_{\text{Al-Alaoui}}(z) = \frac{3}{4}H_{\text{Rectangular}}(z) + \frac{1}{4}H_{\text{Tustin}}(z) \tag{13}$$

Fig. 1 Frequency response of an ideal half-order differentiator

Therefore, $H_{\text{Al-Alaoui}}(z) = \frac{1}{s} = \left(\frac{7T}{8} \frac{1 + \frac{1}{7}z^{-1}}{1 - z^{-1}} \right)$.

It gives a fractional-order differentiator as,

$$s^{\alpha} = \left(\frac{8}{7T} \frac{1 - z^{-1}}{1 + \frac{1}{7}z^{-1}} \right)^{\alpha} \tag{14}$$

On expanding using binomial series,

$$\left(\frac{8}{7T} \frac{1 - z^{-1}}{1 + \frac{1}{7}z^{-1}} \right)^{\alpha} = \left(\frac{8}{7T} \right)^{\alpha} \left(\frac{\sum_{k=0}^{\infty} \binom{\alpha}{k}(-z)^{-k}}{\sum_{k=0}^{\infty} \binom{\alpha}{k}\left(\frac{z}{7}\right)^{-k}} \right) \tag{15}$$

where $\binom{\alpha}{k} = \frac{\gamma(\alpha+1)}{\gamma(k+1)*\gamma(\alpha-k+1)}$, where the symbol γ is the gamma function.

A half differentiator, expressed up to fifth order, can now be described as:

$$s^{\frac{1}{2}} = \sqrt{\left(\frac{8}{7T} \right)}$$

$$\frac{1 + \frac{\gamma(1.5)}{\gamma(2)*\gamma(0.5)}(-z)^{-1} + \frac{\gamma(1.5)}{\gamma(3)*\gamma(-0.5)}(-z)^{-2} + \frac{\gamma(1.5)}{\gamma(4)*\gamma(-1.5)}(-z)^{-3} + \frac{\gamma(1.5)}{\gamma(5)*\gamma(-2.5)}(-z)^{-4} + \frac{\gamma(1.5)}{\gamma(6)*\gamma(-3.5)}(-z)^{-5}}{1 + \frac{\gamma(1.5)}{\gamma(2)*\gamma(0.5)}\left(\frac{z}{7}\right)^{-1} + \frac{\gamma(1.5)}{\gamma(3)*\gamma(-0.5)}\left(\frac{z}{7}\right)^{-2} + \frac{\gamma(1.5)}{\gamma(4)*\gamma(-1.5)}\left(\frac{z}{7}\right)^{-3} + \frac{\gamma(1.5)}{\gamma(5)*\gamma(-2.5)}\left(\frac{z}{7}\right)^{-4} + \frac{\gamma(1.5)}{\gamma(6)*\gamma(-3.5)}\left(\frac{z}{7}\right)^{-5}}$$

$$s^{\frac{1}{2}} = \sqrt{\left(\frac{8}{7T}\right)}$$

$$\frac{1 - \frac{0.8862}{1*(1.7725)}(z)^{-1} + \frac{0.8862}{2*(-3.5449)}(z)^{-2} - \frac{0.8862}{6*2.3633}(z)^{-3} + \frac{0.8862}{24*(0.9453)}z^{-4} - \frac{0.8862}{120*0.2701}z^{-5}}{1 + \frac{0.8862}{1*1.7725}\left(\frac{z}{7}\right)^{-1} + \frac{0.8862}{2*(-3.5449)}\left(\frac{z}{7}\right)^{-2} + \frac{0.8862}{6*2.3633}\left(\frac{z}{7}\right)^{-3} + \frac{0.8862}{24*(-0.9453)}\left(\frac{z}{7}\right)^{-4} + \frac{0.8862}{120*0.2701}\left(\frac{z}{7}\right)^{-5}}$$

$$s^{\frac{1}{2}} = \sqrt{\left(\frac{8}{7T}\right)}$$

$$\frac{1 - 0.5z^{-1} - 0.1250z^{-2} - 0.0625z^{-3} - 0.0390625z^{-4} - 0.02734375z^{-5}}{1 + 0.07142857z^{-1} - 0.00255102z^{-2} + 0.0001822z^{-3} - 0.00001626926z^{-4} + 0.000001626926z^{-5}}$$

For $T = 0.001$, $s^{\frac{1}{2}}$ can be expressed as

$$s^{\frac{1}{2}} = 33.80617$$

$$\frac{1 - 0.5z^{-1} - 0.1250z^{-2} - 0.0625z^{-3} - 0.0390625z^{-4} - 0.02734375z^{-5}}{1 + 0.07142857z^{-1} - 0.00255102z^{-2} + 0.0001822z^{-3} - 0.00001626926z^{-4} + 0.000001626926z^{-5}} \tag{16}$$

Similarly,

$$s^{\frac{1}{3}} = 10.45516$$

$$\frac{1 - 0.333333z^{-1} - 0.111111z^{-2} - 0.0617284z^{-3} - 0.04115226z^{-4} - 0.03017833z^{-5}}{1 + 0.04761905z^{-1} - 0.002267574z^{-2} + 0.000179966z^{-3} - 0.0000171396z^{-4} + 0.00000179558z^{-5}} \tag{17}$$

$$s^{\frac{1}{4}} = 5.814307$$

$$\frac{1 - 0.25z^{-1} - 0.09375z^{-2} - 0.0546875z^{-3} - 0.03759766z^{-4} - 0.02819824z^{-5}}{1 + 0.03571429z^{-1} - 0.001913265z^{-2} + 0.000159438z^{-3} - 0.0000156591z^{-4} + 0.000001677768z^{-5}} \tag{18}$$

Corresponding third-order designs can be obtained by truncating the above expressions up to z^{-3}. From (16) to (18), we can find the following coefficients (b and a) to design the resulting fractional operators.

$$s^{\alpha} \cong \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} \cdots + b_n z^{-m}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} \cdots + a_n z^{-n}}; m \leq n \tag{19}$$

The coefficients of z-terms are then fed to NMSA, as the initial solutions, to minimize the real-valued RMS error of the magnitude response of the system $H(j\omega)$ defined as:

$$\text{RMS Magnitude Error} = \sqrt{\frac{\sum \left||H(\omega)|_{\text{Proposed}} - |H(\omega)|_{\text{Ideal}}\right|^2}{n}} \tag{20}$$

where n is the number of sampled frequency points.

3 Optimization Using Nelder–Mead Simplex Algorithm

The NMSA is a popular direct search method for unconstrained minimization [14,27]. It aims at minimizing a real-valued function $f(x)$ for $x \epsilon R^n$. It is a direct search method;

i.e., it does not use derivative information unlike gradient methods of optimization. In cases where it is difficult to obtain a derivative of fitness functions, direct search methods are more preferred over gradient-based methods. It may also be noted that the NMSA is a local optimization method and global optimum search is not guaranteed. Local optimization methods locate a minima, i.e., minimum value, in a region nearby the proposed starting point, whereas global optimization techniques aim at locating the global minima, i.e., lowest possible value of the fitness function. As will be seen later, the popular global optimization method GA also tends to take more time than the aforementioned method. As mentioned earlier, in the NMSA, used in this work, an initial estimation of the solution is already calculated using discretization of Al-Alaoui operator followed by binomial expansion of the numerator and the denominator. The only task left is to search for the best possible coefficient values in the nearby region without any explicit boundaries. Therefore, NMSA has worked pretty well. This method is termed as "simplex" method, but it should not be confused with the well-known simplex method for linear programming.

At the beginning of every iteration, a new simplex is given, along with its (n+1) vertices such that

$$f(x_1) \leq f(x_2) \leq \cdots \leq f(x_{n+1})$$

where $x_1, x_2, x_3, \ldots x_{n+1}$ are the arrays of n points each, n is the number of variables for which the fitness function is to be optimized, x_1 is referred to as the best point, and x_{n+1} is referred to as the worst point. Similarly, $f(x_1)$ is referred to as the best fitness function value, whereas $f(x_{n+1})$ as the worst fitness function value. The next iteration defines a new simplex and generates a new set of $(n+1)$ vertices by moving the worst vertex around a centroid which is an average value of the remaining vertices. The process continues in the same manner while minimizing the worst fitness function value. To provide a big picture, NMSA "shrinks" a polygon of $(n+1)$ vertex, where n is the number of variables to be optimized. The flowchart in Fig. 2 describes the algorithm completely. Various computations for the implementation can be described as:

$$\text{Reflection point: } x_{\mathrm{r}} = \bar{x} + \rho(\bar{x} - x_{n+1}) \tag{21}$$

$$\text{Expansion point: } x_{\mathrm{e}} = \bar{x} + \chi(x_{\mathrm{r}} - \bar{x}) \tag{22}$$

$$\text{Outside contraction: } x_{\mathrm{c}} = \bar{x} + \beta(x_{\mathrm{r}} - \bar{x}) \tag{23}$$

$$\text{Inside contraction: } x_{\mathrm{cc}} = \bar{x} - \beta(\bar{x} - x_{n+1}) \tag{24}$$

Coefficient of reflection: $\rho = 1$; coefficient of expansion: $\chi = 2$; coefficient of contraction: $\beta = 0.5$; coefficient of shrinkage: $\sigma = 0.5$; and the centroid: $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.

Fitness function to be minimized, in this particular problem, is defined as follows:
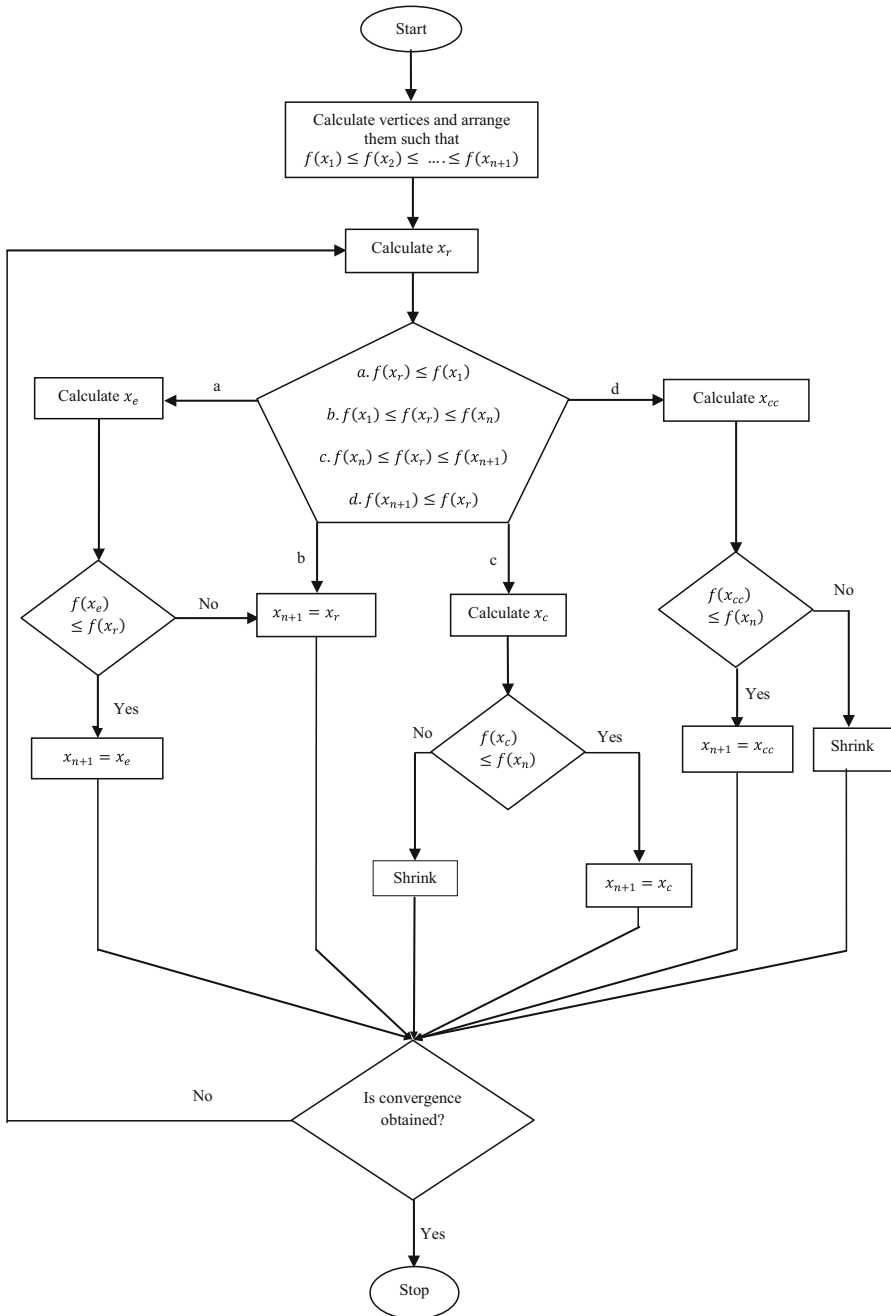
$$f(x) = \text{RMS}_{\text{Magnitude}}$$

Fig. 2 Nelder–Mead simplex algorithm

Table 1 Definitions of variables

j	$\Delta_{1,j}$	$\Delta_{2,j}$
2	$p = \frac{a}{N\sqrt{2}}\sqrt{N+1} + N - 1$	$q = \frac{a}{N\sqrt{2}}\sqrt{N+1} - 1$
3	$q = \frac{a}{N\sqrt{2}}\sqrt{N+1} - 1$	$p = \frac{a}{N\sqrt{2}}\sqrt{N+1} + N - 1$

To demonstrate the implementation of the algorithm, a numerical example is given below. Let us assume we need to find the minima of a function defined by $f(x, y) = x^2 + y^2 + 2x + 2y + 2$. It may be noted that in this case the function itself becomes the fitness function. Let the starting initial vertex be (0, 1). Since it is a function in two variables, the simplex here would be a triangle. Therefore, $x = [0, 1]$ and $x_j = x_1 + \Delta_j$, where $j = 2, 3$ and Δ_j is determined using the following Table 1 [2]. Length of the sides of the simplex $a = 3$, $N = 2$. Therefore, $p = 2.84, q = 0.84$. The stepwise iterative computations are shown as follows:

S. no.	Iteration no. 1	Iteration no. 2	Iteration no. 3	Iteration no. 20
1	$x_1 = [0, 1]$	$x_1 = [0, 1]$	$x_1 = [-0.84, -1.84]$	$x_1 = [-0.9988, -1.0094]$
2	$x_2 = [2.84, 1.84]$	$x_2 = [2, -1]$	$x_2 = [0, 1]$ ...	$x_2 = [-1.0105, -1.0018]$
3	$x_3 = [0.84, 3.84]$	$x_3 = [2.84, 1.84]$	$x_3 = [2, -1]$	$x_3 = [-0.9846, -0.9802]$
4	$f(x_1) = 5$	$f(x_1) = 5$	$f(x_1) = 0.7312$	$f(x_1) = 0.000088923$
5	$f(x_2) = 11.4512$	$f(x_2) = 9$	$f(x_2) = 5$ ...	$f(x_2) = 0.00011445$
6	$f(x_3) = 15.4512$	$f(x_3) = 11.4512$	$f(x_3) = 9$	$f(x_3) = 0.0006317$
7	$\bar{x} = [1.42, 1.42]$	$\bar{x} = [10]$	$\bar{x} = [-0.42, -0.42]$	$\bar{x} = [-1.0047, -1.0056]$
8	$x_r = [2, -1]$	$x_r = [-0.84, -1.84]$	$x_r = [-2.84, 0.16]$ ...	$x_r = [-1.0247 - 1.0310]$
9	$f(x_r) = 9$	$f(x_r) = 0.7312$	$f(x_r) = 4.6712$	$f(x_r) = 0.0016$
10		$x_e = [-2.68, -3.68]$		$x_{cc} = [-0.9946, -0.9929]$
11		$f(x_e) = 10.0048$		$f(x_{cc}) = 0.000079301$

As can be seen, a decreasing error is observed for successive iterations. The process continues till a convergence is obtained, which would depend upon the tolerance set by the user. In the case under consideration, at 20th iteration, the vertices of the triangle are obtained as $x_1 = [-0.9988, -1.0094]$, $x_2 = [-1.0105, -1.0018]$, $x_3 = [-0.9846, -0.9802]$ and an error of $f(x_{cc}) = 0.000079301$ against the actual result of $[-1, -1]$.

The NMSA has both been acclaimed and criticized: acclaimed, mainly because it is one of the first derivative-free optimization techniques and is very simple to understand, robust, and reliable, and criticized mainly because it has been developed heuristically and no proofs of convergence have been derived.

4 Fractional-Order Differentiator Designs Using NMSA

To demonstrate the capability of the proposed optimization technique, NMSA, six case studies have been investigated and the obtained results are presented in this section.

These investigations include a half differentiator of third order, a half differentiator of fifth order, a one-third differentiator of third order, a one-third differentiator of fifth order, a one-fourth differentiator of third order, and lastly, a one-fourth differentiator of fifth order. For each of the cases under consideration, the initial starting solution, as required by NMSA, is first obtained as demonstrated in Eqs. (16)–(18) for respective values of α. The coefficients of the respective transfer functions for differentiators of various orders are then fine-tuned for minimum RMS magnitude error with the help of the NMSA. Tables 2 and 3 present the final optimized differentiator designs with their RMS magnitude errors for various fractional-order differentiators. The magnitude responses of various differentiators are computed for a sampling time $T = 0.001$ s. It may be noted that the choice of T is purely arbitrary and it is applicable to other sampling rates as well by suitably scaling the designs. As shown in Tables 2 and 3, RMS magnitude errors of 1.193117 and 0.773236 have been achieved for half differentiators of third and fifth orders, respectively. Similarly, RMS magnitude errors of 0.441906 and 0.307687 have been achieved for one-third differentiators of third and fifth orders, respectively, and RMS magnitude errors of 0.243824 and 0.175219 have been achieved for one-fourth differentiators of third and fifth orders, respectively. For these optimizations, the numbers of iterations have been kept as 200, 450, 120, 300, 100, and 250, respectively, for third- and fifth-order half differentiators, third- and fifth-order one-third differentiators, and third- and fifth-order one-fourth differentiators. These numbers have been decided after investigating the pole–zero behavior of respective differentiators and ensuring a small pole radius (<0.15, in all cases). It was also observed that higher number of iterations yielded better RMS magnitude error, but the design possessed larger pole radii, thereby lengthening the transient response. Due to this reason, the numbers of iterations were kept as mentioned above.

Table 2 Designed third-order fractional differentiators

α	Designed differentiator coefficients				Magnitude RMS error
$\frac{1}{2}$	b_0	b_1	b_2	b_3	1.193117
	3.434456E+1	−1.795178E+1	−2.757224	−2.575568	
	a_0	a_1	a_2	a_3	
	1	5.048566E−2	−5.266890E−3	6.843964E−5	
$\frac{1}{3}$	b_0	b_1	b_2	b_3	0.441906
	1.052659E+1	−3.934956	−7.753260E−1	−7.132925E−1	
	a_0	a_1	a_2	a_3	
	1	1.560748E−2	−3.464132E−3	2.157619E−4	
$\frac{1}{4}$	b_0	b_1	b_2	b_3	0.243824
	5.845709	−1.547169	−4.325582E−1	−3.163191E−1	
	a_0	a_1	a_2	a_3	
	1	2.282034E−2	−2.507615E−3	1.593232E−4	

Table 3 Designed fifth-order fractional differentiators

| α	Designed differentiator coefficients						Magnitude RMS error
$\frac{1}{2}$	b_0	b_1	b_2	b_3	b_4	b_5	0.773236
	3.414086E+1	−1.704155E+1	−3.346160	−2.660430	−9.102775E−1	−1.055785	
	a_0	a_1	a_2	a_3	a_4	a_5	
	1	8.168769E−2	−2.921810E−3	1.889661E−4	−1.450192E−5	1.573012E−6	
$\frac{1}{3}$	b_0	b_1	b_2	b_3	b_4	b_5	0.307687
	1.052335E+1	−3.560287	−9.617495E−1	−7.250713E−1	−3.446479E−1	−3.297198E−1	
	a_0	a_1	a_2	a_3	a_4	a_5	
	1	4.931106E−2	−2.254036E−3	1.903312E−4	−2.012126E−5	1.770592E−6	
$\frac{1}{4}$	b_0	b_1	b_2	b_3	b_4	b_5	0.175219
	5.841132	−1.490550	−4.645373E−1	−3.474446E−1	−1.840672E−1	−1.657178E−1	
	a_0	a_1	a_2	a_3	a_4	a_5	
	1	3.594986E−2	−2.011865E−3	1.529481E−4	−1.590108E−5	1.774913E−6	
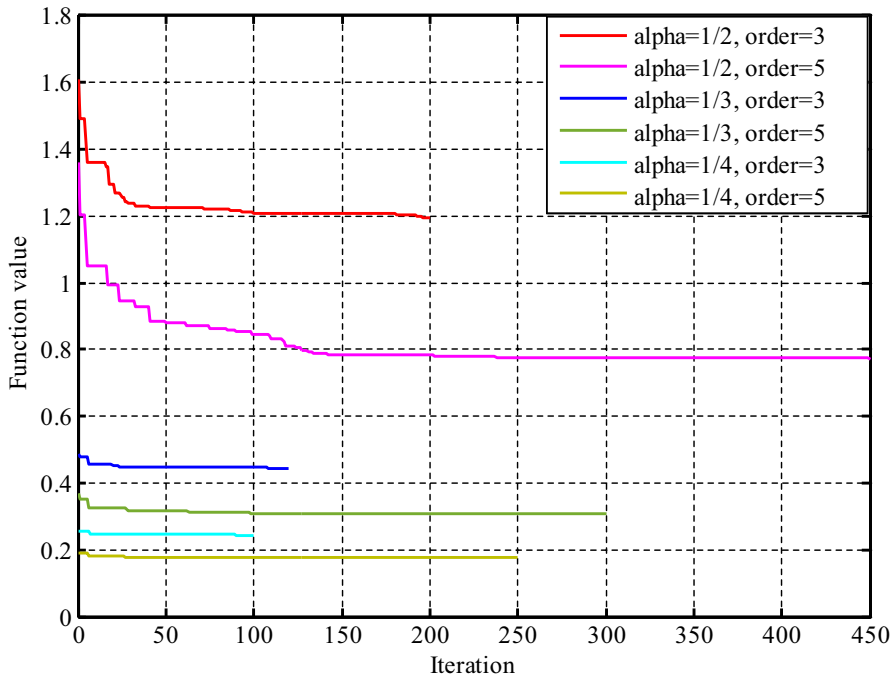
Fig. 3 Fitness function v/s generations (NMSA)

Figure 3 presents variations of fitness function with generations for all the considered differentiators. As seen from figure, all the obtained fitness curves have flattened; therefore, it can be concluded that the best possible designs have been achieved and there is not much scope for further improvement in the NMSA optimization process. Figure 4 compares the frequency response with the ideal one for half differentiator of third and fifth orders. Figure 5 presents the corresponding absolute error v/s frequency for both designs of half differentiators. Similarly, Figs. 6, 7, 8, and 9 present the same for one-third and one-fourth differentiators. All of the proposed designs can be seen to be tracking the ideal responses very closely in these figures.

5 Comparison with the Existing Differentiators

Several fractional-order differentiators of various orders and fractional powers, designed by various other methods, have been reported in the literature. This section presents a comparative study with the respective class of differentiators. Table 4 presents designs of half differentiators of third order recently reported in the literature along with the proposed design. Corresponding RMS magnitude and phase errors are also listed in Table 5. It can be clearly observed that the RMS magnitude error, 1.193117, of proposed half differentiator is the lowest among the compared ones. Figure 10 plots the frequency magnitude responses along with the phase responses of these designs. Figure 11 depicts the corresponding RMS magnitude errors and the
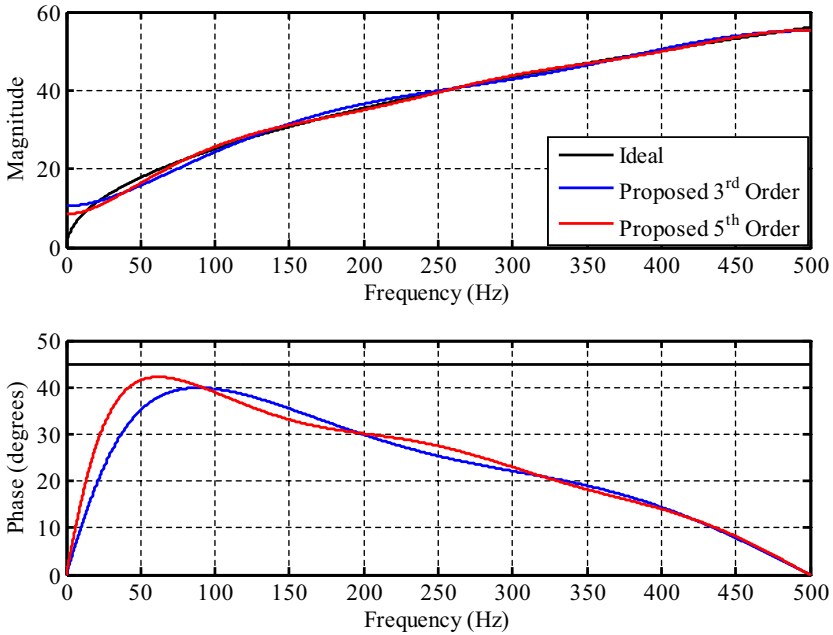
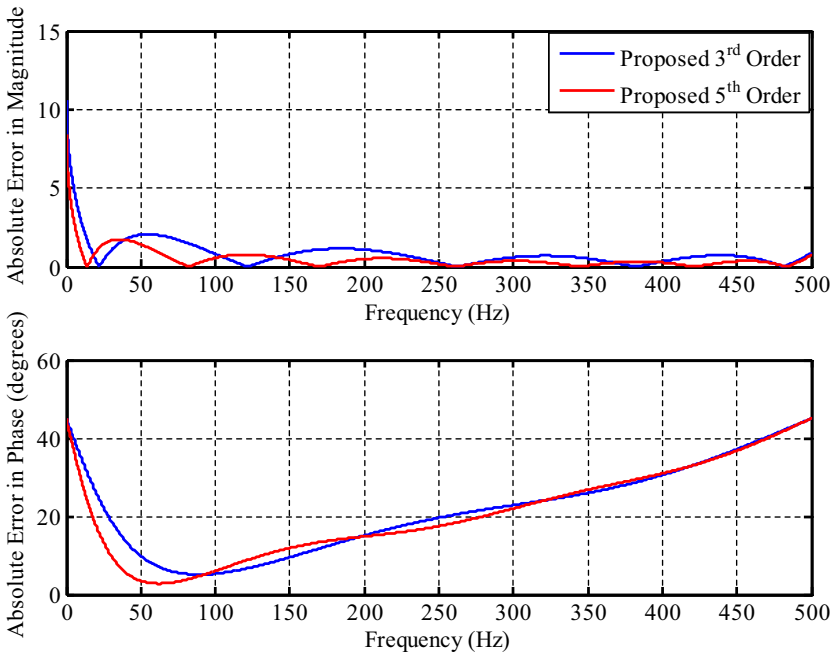Fig. 4 Frequency responses of half differentiators of third and fifth orders

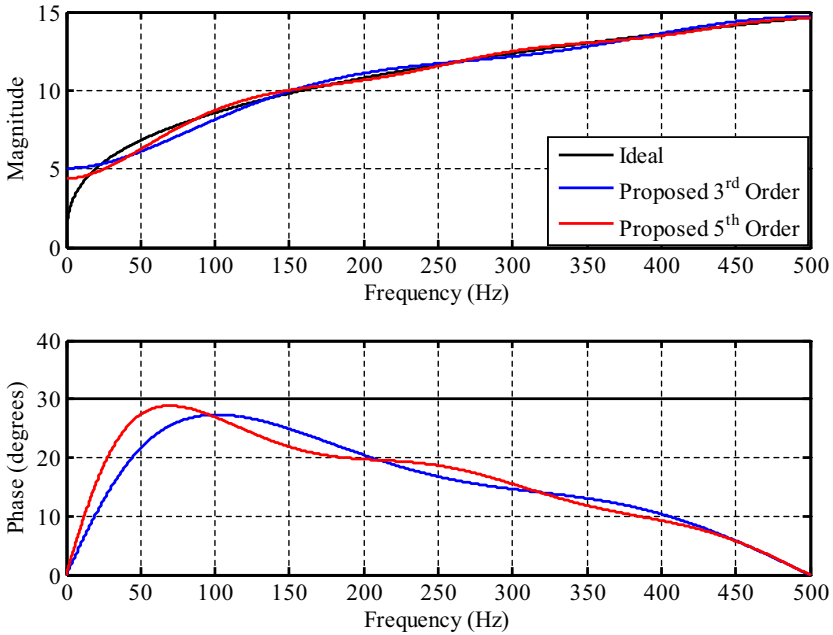Fig. 5 Absolute error v/s frequency plots of half differentiators of third and fifth orders

Fig. 6 Frequency responses of one-third differentiators of third and fifth orders
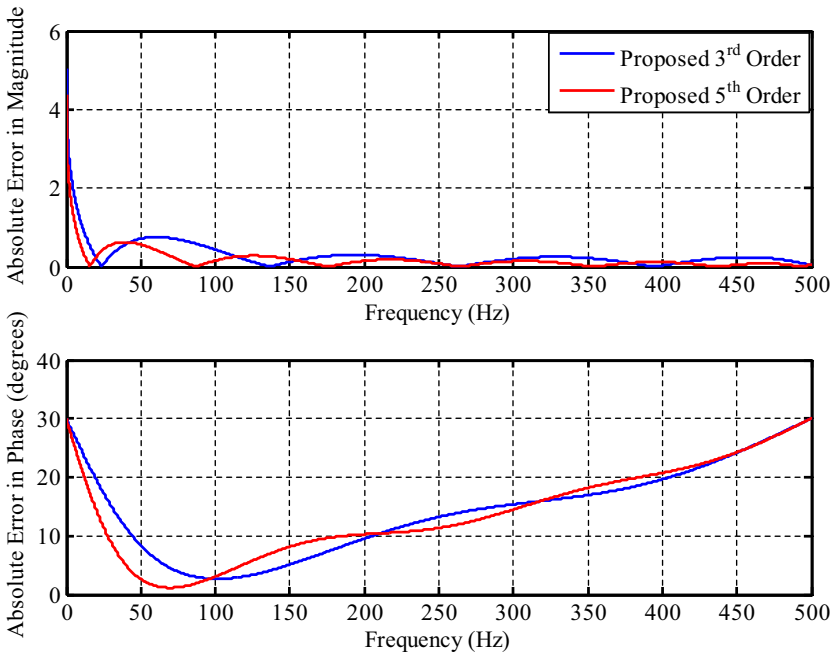

Fig. 7 Absolute error v/s frequency plots of one-third differentiators of third and fifth orders
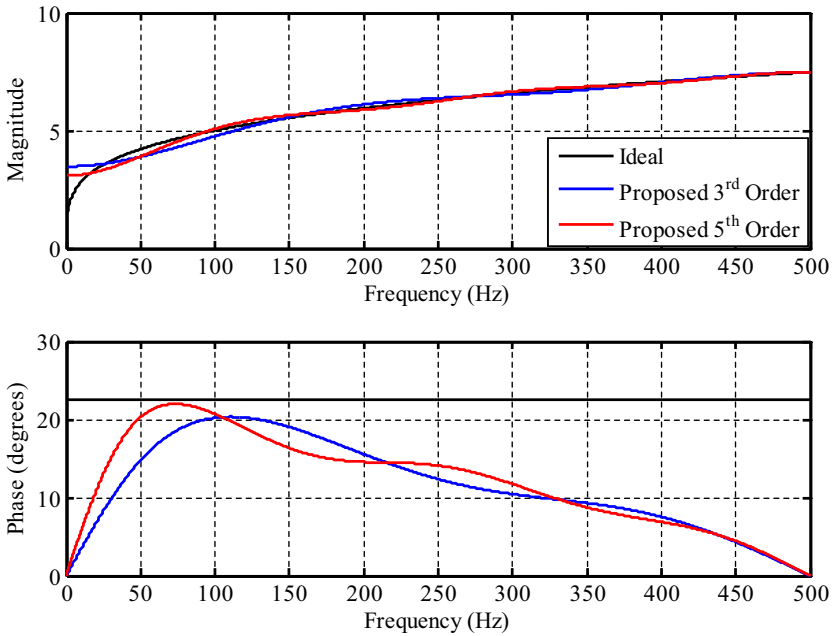
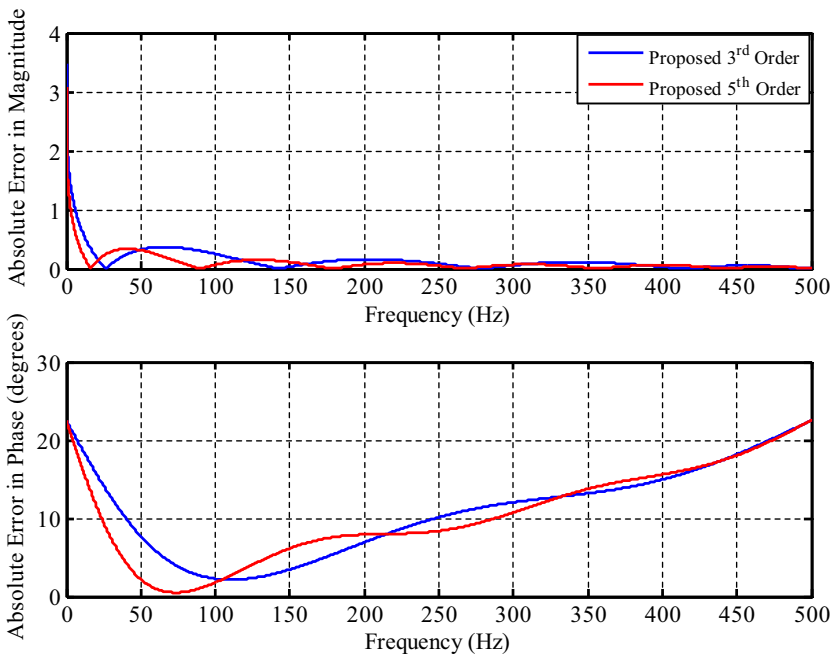**Fig. 8** Frequency responses of one-fourth differentiators of third and fifth orders

Fig. 9 Absolute error v/s frequency plots of one-fourth differentiators of third and fifth orders

Table 4 Designs of third-order half differentiators

Differentiator	Differentiator coefficients			
Proposed	b_0	b_1	b_2	b_3
	3.434456E+1	−1.795178E+01	−2.757224	−2.575568
	a_0	a_1	a_2	a_3
	1	5.048566E−2	−5.266890E−3	6.843964E−5
[3] 2002	b_0	b_1	b_2	b_3
	4.472136E+1	−2.236068E+1	3.726780	−7.453560
	a_0	a_1	a_2	a_3
	1	5.000000E−1	8.333333E−2	1.666667E−1
[5] 2003	b_0	b_1	b_2	b_3
	3.176047E+1	−1.065595E+1	−2.727465E+1	9.447736
	a_0	a_1	a_2	a_3
	1	8.463817E−2	−5.000000E−1	−2.115954E−2
[7] 2011	b_0	b_1	b_2	b_3
	3.288333E+1	−5.420833E+1	2.418333E+1	−2.209167
	a_0	a_1	a_2	a_3
	1	−1.108333	2.388333E−1	7.442500E−3
[11] 2011 (1)	b_0	b_1	b_2	b_3
	3.728000E+1	−1.864000E+1	−4.660457	−2.330078
	a_0	a_1	a_2	a_3
	1	2.468000E−1	−1.356062E−1	1.945877E−1
[11] 2011 (2)	b_0	b_1	b_2	b_3
	3.566510E+1	−2.987237E+1	5.623533	−4.127119
	a_0	a_1	a_2	a_3
	1	−2.544000E−1	5.220900E−2	−4.872150E−3
[24] 2003	b_0	b_1	b_2	b_3
	4.472136E+1	−2.236068E+1	−2.236068E+1	5.590170
	a_0	a_1	a_2	a_3
	1	5.000000E−1	−5.000000E−1	−1.250000E−1

Table 5 RMS magnitude and phase errors of third-order half differentiators

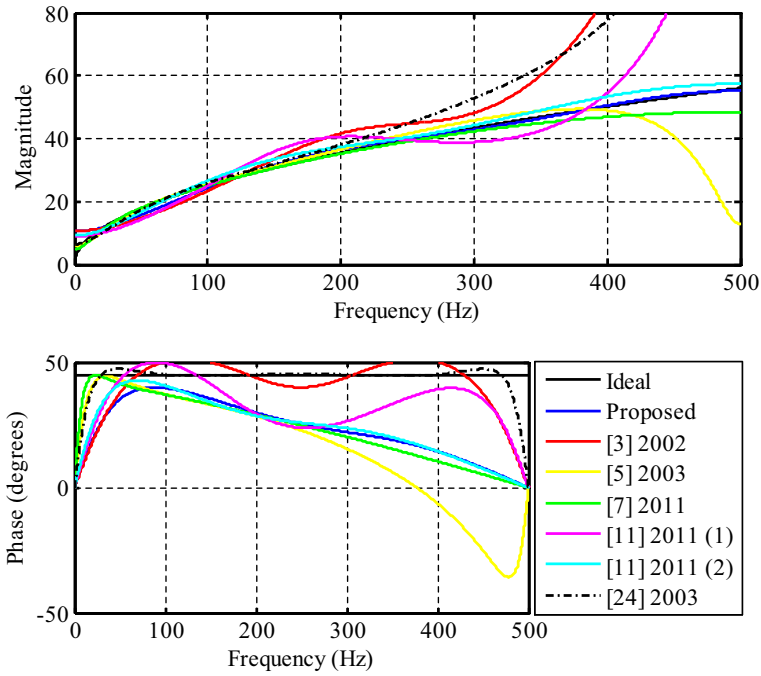| Differentiator	Magnitude RMS error	Phase RMS error
Proposed	1.193117	24.290711
[3] 2002	41.382609	12.346795
[5] 2003	9.504301	37.220378
[7] 2011	2.599890	24.965005
[11] 2011 (1)	18.735659	15.774734
[11] 2011 (2)	2.144570	23.713771
[24] 2003	51.777445	7.014340

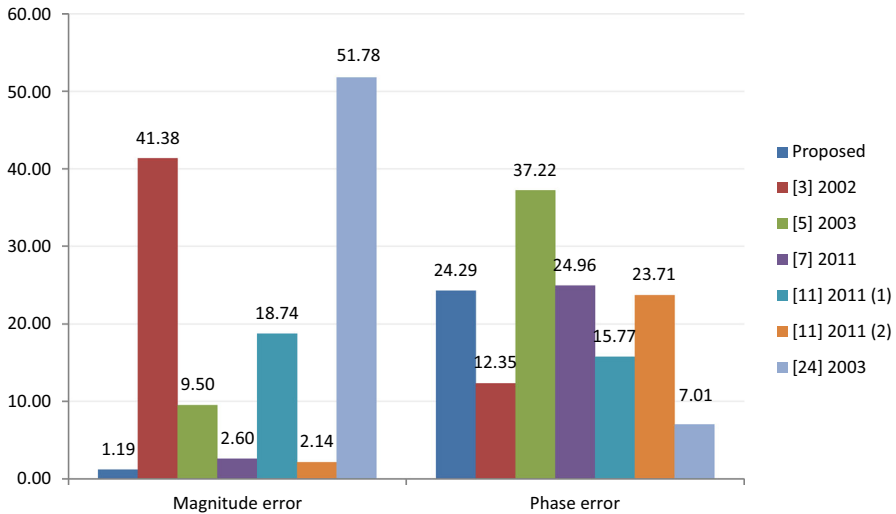**Fig. 10** Frequency responses of third-order half differentiators

Fig. 11 Comparison of magnitude and phase RMS errors of third-order half differentiators

RMS phase errors. Similarly, Table 6 presents the designs and Table 7 presents the corresponding performances of the half differentiators of fifth order. Again, the RMS magnitude error, 0.773236, of proposed half differentiator is the lowest among the

compared ones. Figure 12 plots the frequency magnitude responses along with the phase responses of these designs. Figure 13 plots the corresponding RMS magnitude errors and the RMS phase errors. For all the comparisons made in this paper, the references have been designated by the respective serial numbers as these have appeared in the references followed by the year. In case of multiple designs from a given reference, the above syntax has been further appended with numbers, 1, 2, and so on. For example, [7] 2011 (2) refers to second design of reference number 7, published in 2011.

Furthermore, Table 8 presents the designs and Table 9, the corresponding performances of the one-third differentiators of third order. The RMS magnitude error, 0.441906, of proposed one-third differentiator is again the lowest among the compared ones. Figure 14 plots the frequency magnitude responses along with the phase responses of these designs. Figure 15 plots the corresponding RMS magnitude errors and the RMS phase errors. Similarly, Table 10 presents the designs and Table 11, the corresponding performances of the one-third differentiators of fifth order. Again, the RMS magnitude error, 0.307687, of the proposed one-third differentiator is the lowest among the compared ones. Figure 16 plots the corresponding frequency magnitude responses along with the phase responses. Figure 17 plots the corresponding RMS magnitude errors and the RMS phase errors of these designs.

Table 12 presents the designs and Table 13, the corresponding performances of the one-fourth differentiators of third order. Here, also it can be clearly observed that the RMS magnitude error, 0.243824, of proposed half differentiator is the lowest among the compared ones. Figure 18 plots the frequency magnitude responses along with the phase responses. Figure 19 plots the corresponding RMS magnitude errors and the RMS phase errors. Finally, Table 14 presents the designs and Table 15, the corresponding performances of the one-fourth differentiators of fifth order. Again, the RMS magnitude error, 0.175219, of proposed one-fourth differentiator is the lowest among the compared ones. Figure 20 plots the frequency magnitude responses along with the phase responses. Figure 21 plots the corresponding RMS magnitude and phase errors.

6 Comparative Study of NMSA with Genetic Algorithms

The performance of NMSA, in terms of the achievable fitness function values and the timing aspect, has been compared with the GA optimization technique using their respective MATLAB packages. Applications of NMSA for designing FODs have already been demonstrated. A sample case study considering only the half differentiator of third-order design by GA has been conducted, and the results are presented in this section. Various parameters used for GA optimization technique are listed in Table 16. To evaluate the performance comparison of these optimization methods, 50 independent trial runs of both methods were conducted and the obtained values of fitness function were recorded. NMSA was observed to produce zero standard deviation, whereas GA produced a standard deviation of 6.039116 in RMS magnitude error. The zero standard deviation in the designs by NMSA is a unique feature indicating the same result for all the 50 trial runs. Figure 22 presents variation of fitness

Table 6 Designs of fifth-order half differentiators

Differentiator	Differentiator coefficients					
Proposed	b_0	b_1	b_2	b_3	b_4	b_5
	3.414086E+1	−1.704155E+1	−3.346160	−2.660430	−9.102775E−1	−1.055785
	a_0	a_1	a_2	a_3	a_4	a_5
	1	8.168769E−2	−2.921810E−3	1.889661E−4	−1.450192E−5	1.573012E−6
[3] 2002	b_0	b_1	b_2	b_3	b_4	b_5
	4.472136E+1	−2.236068E+1	4.472136	−7.826238	2.236068	−4.472136
	a_0	a_1	a_2	a_3	a_4	a_5
	1	5.000000E−1	1.000000E−1	1.750000E−1	5.000000E−2	1.000000E−1
[7] 2011	b_0	b_1	b_2	b_3	b_4	b_5
	3.392900E+1	−8.171300E+1	6.630300E+1	−1.963624E+1	1.023479	1.506614E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	−1.831349	9.765212E−1	−1.007606E−1	−2.379299E−2	1.238095E−3
[11] 2011 (1)	b_0	b_1	b_2	b_3	b_4	b_5
	3.362000E+1	−6.007558E+1	4.526714E+1	−2.166275E+1	6.133797	−9.173436E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	2.960000E−2	8.624479E−2	−7.758709E−2	−1.326591E−2	2.010063E−1
[11] 2011 (2)	b_0	b_1	b_2	b_3	b_4	b_5
	3.566510E+1	−1.783612E+1	−4.458953	−2.228168	−1.393147	−9.749691E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	8.392000E−2	−3.387584E−2	1.279152E−2	−1.649062E−3	5.719951E−4

Table 6 continued

Differentiator	Differentiator coefficients					
[15] 2006	b_0	b_1	b_2	b_3	b_4	b_5
	4.472100	−9.193300	3.925600	2.351900	−1.741400	1.887000E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	−1.055700	−6.776000E−1	8.743000E−1	−7.250000E−2	−5.260000E−2
[24] 2003	b_0	b_1	b_2	b_3	b_4	b_5
	4.472136E+1	−2.236068E+1	−4.472136E+1	1.677051E+1	8.385255	−1.397542
	a_0	a_1	a_2	a_3	a_4	a_5
	1	5.000000E−1	−1.000000	−3.750000E−1	1.875000E−1	3.125000E−2

Table 7 RMS magnitude and phase errors of fifth-order half differentiators

Differentiator	Magnitude RMS error	Phase RMS error
Proposed	0.773236	23.664123
[3] 2002	46.063501	10.118620
[7] 2011	0.981034	23.636441
[11] 2011 (1)	52.362907	39.419624
[11] 2011 (2)	2.052938	23.351066
[15] 2006	32.922017	5.875392
[24] 2003	60.760537	4.438816
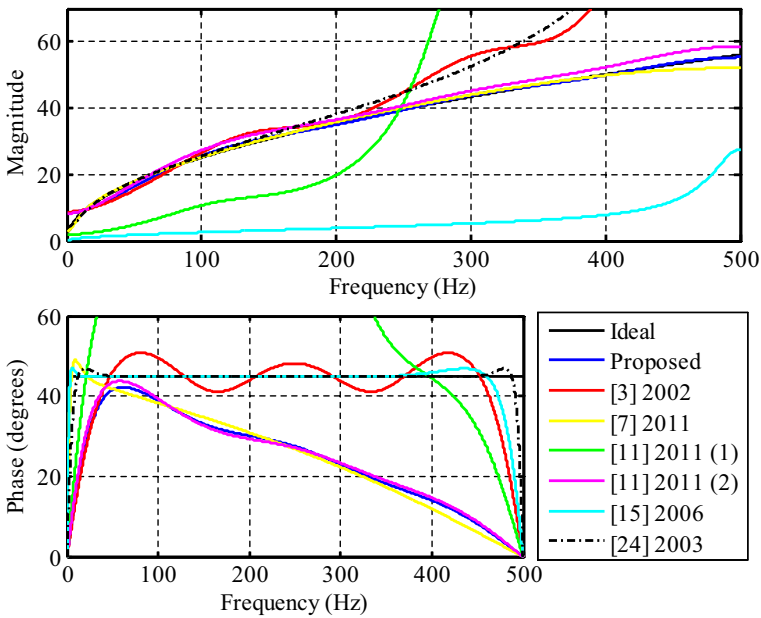
Fig. 12 Frequency responses of fifth-order half differentiators

function with number of iterations as optimized using GA as a sample design study. Comparing Figs. 3 and 22, it can be clearly seen that NMSA converges much faster and better. In this case, GA was able to achieve a best fitness value of 5.80047 in 200 iterations against 1.193116 of NMSA in 200 iterations. It should be noted that the same fitness function has been used for both the optimization methods. Also, on an average, NMSA was able to execute an iteration in 0.6795 s as compared to 26.0796 s taken by GA. Based on this input, it can be concluded that NMSA is 38 times faster than GA. The above computations were performed on an Intel Core i5 @ 3.2 GHz CPU with 4 GB RAM. A sample design obtained by GA optimization is given in Table 17 along with its performance in terms of the RMS magnitude error. The corresponding frequency response is shown in Fig. 23. Table 18 summarizes the comparison of GA-
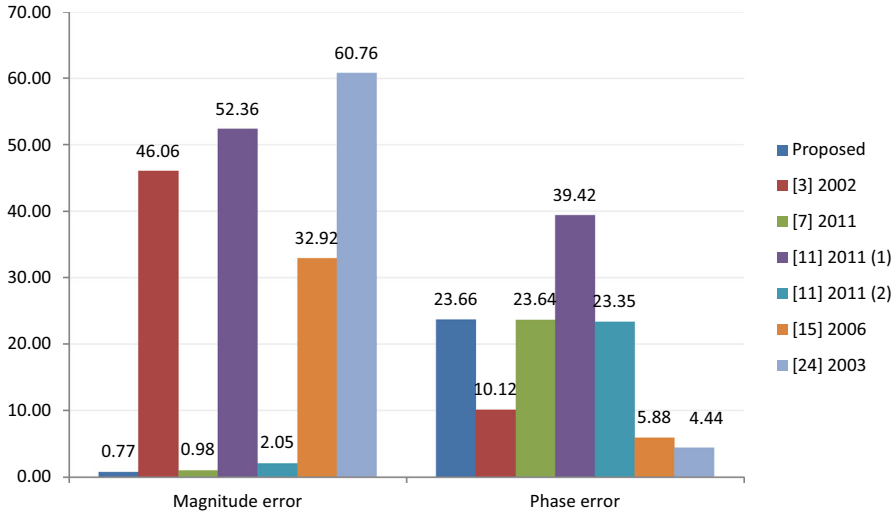
Fig. 13 Comparison of magnitude and phase RMS errors of fifth-order half differentiators

Table 8 Designs of third-order one-third differentiators

Differentiator	Differentiator coefficients			
Proposed	b_0	b_1	b_2	b_3
	1.052659E+1	−3.934956	−7.753260E−1	−7.132925E−1
	a_0	a_1	a_2	a_3
	1	1.560748E−2	−3.464132E−3	2.157619E−4
[11] 2011 (1)	b_0	b_1	b_2	b_3
	1.205100E+1	−4.016598	−1.339088	−7.438297E−1
	a_0	a_1	a_2	a_3
	1	3.220000E−1	−2.288478E−1	2.623792E−1
[11] 2011 (2)	b_0	b_1	b_2	b_3
	1.339000E+1	−4.462887	−1.487875	−8.264775E−1
	a_0	a_1	a_2	a_3
	1	−4.828000E−1	9.160996E−2	−9.191928E−3

Table 9 RMS magnitude and phase errors of third-order one-third differentiators

Differentiator	Magnitude RMS error	Phase RMS error
Proposed	0.441906	16.096583
[11] 2011 (1)	12.984587	13.156384
[11] 2011 (2)	4.943222	35.045273

based designs with the NMSA-based design. Based on this study, it can be concluded that NMSA is a better option as compared to GA for the fractional-order differentiator design. Inferior results obtained by GA can be attributed to its tendency to get trapped
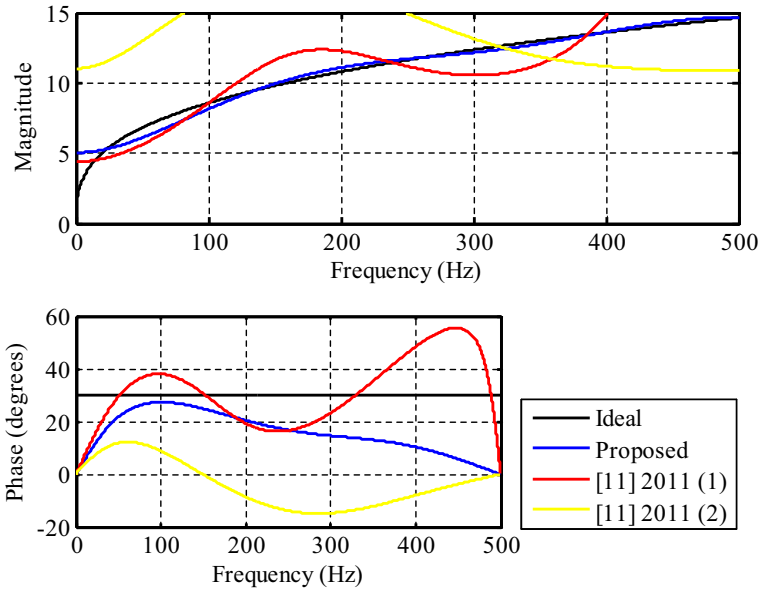
Fig. 14 Frequency responses of third-order one-third differentiators
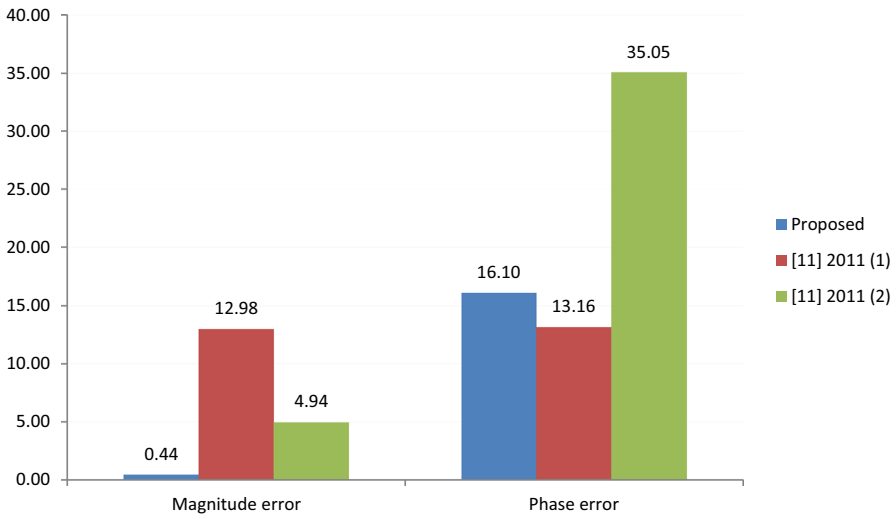

Fig. 15 Comparison of magnitude and phase RMS errors of third-order one-third differentiators

in local minima. This is in line with the other works where GA has been claimed to yield poor convergence and solution quality [18].

To further compare the effectiveness of NMSA and GA, a hypothetical t test was performed on the fitness function values obtained in the above-mentioned 50 independent trial runs. t test, t Student statistic or distribution t is a method for determining the statistical significance of the difference between two independent samples of an

Table 10 Designs of fifth-order one-third differentiators

Differentiator	Differentiator coefficients					
Proposed	b_0	b_1	b_2	b_3	b_4	b_5
	1.052335E+1	−3.560287	−9.617495E−1	−7.250713E−1	−3.446479E−1	−3.297198E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	4.931106E−2	−2.254036E−3	1.903312E−4	−2.012126E−5	1.770592E−6
[11] 2011 (1)	b_0	b_1	b_2	b_3	b_4	b_5
	1.021260E+1	−3.403860	−1.135208	−6.297690E−1	−4.206274E−1	−3.081880E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	−1.410000E−2	7.528558E−2	−4.136636E−2	−6.682273E−2	2.695440E−1
[11] 2011 (2)	b_0	b_1	b_2	b_3	b_4	b_5
	1.339000E+1	−4.464226	−1.488366	−8.261712E−1	−5.508015E−1	−4.038895E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	5.595000E−2	−2.338737E−2	9.187388E−3	−1.502804E−3	5.384931E−4

Table 11 RMS magnitude and phase errors of fifth-order one-third differentiators

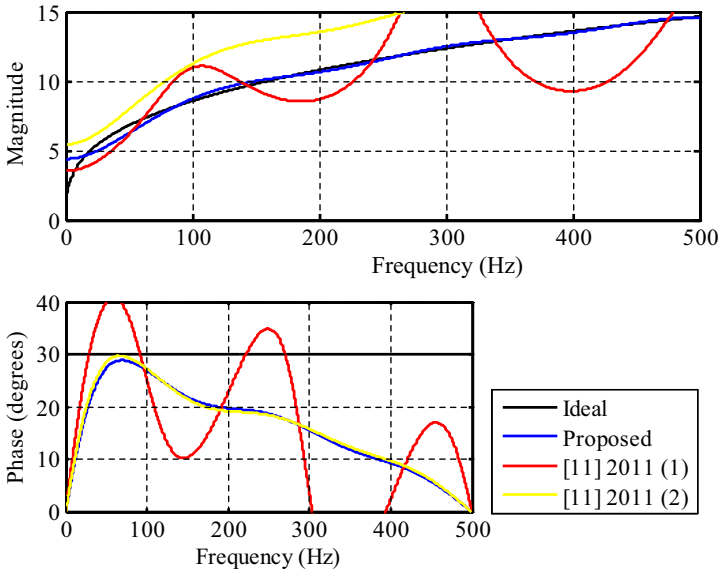| Differentiator	Magnitude RMS error	Phase RMS error
Proposed	0.307687	15.883499
[11] 2011 (1)	2.836125	21.446461
[11] 2011 (2)	3.121272	15.696433

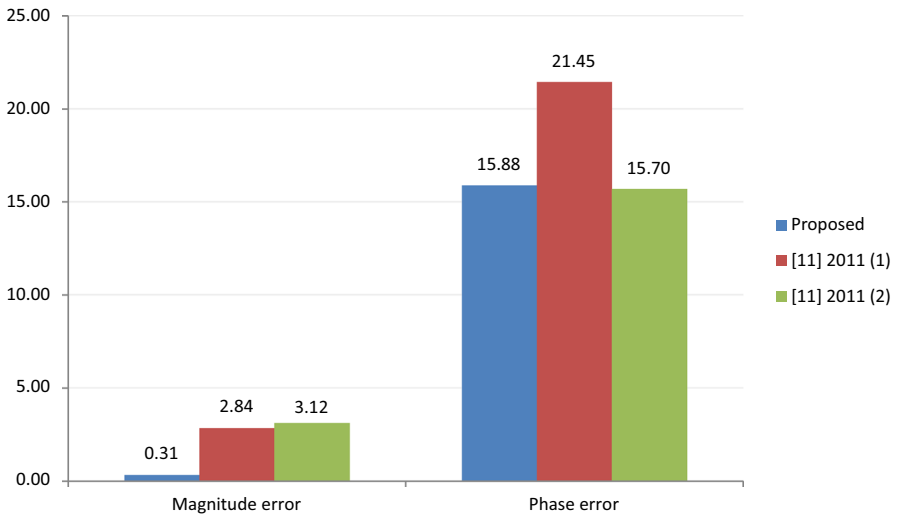**Fig. 16** Frequency responses of fifth-order one-third differentiators

Fig. 17 Comparison of magnitude and phase RMS errors of fifth-order one-third differentiators

Table 12 Designs of third-order one-fourth differentiators

Differentiator	Differentiator coefficients			
Proposed	b_0	b_1	b_2	b_3
	5.845709	-1.547169	$-4.325582E-1$	$-3.163191E-1$
	a_0	a_1	a_2	a_3
	1	$2.282034E-2$	$-2.507615E-3$	$1.593232E-4$
[11] 2011 (1)	b_0	b_1	b_2	b_3
	6.993000	-1.748250	$-6.558637E-1$	$-3.824480E-1$
	a_0	a_1	a_2	a_3
	1	$4.000000E-1$	$-2.899532E-1$	$2.839968E-1$
[11] 2011 (2)	b_0	b_1	b_2	b_3
	5.794000	-1.448500	$-5.434112E-1$	$-3.168746E-1$
	a_0	a_1	a_2	a_3
	1	$4.200000E-2$	$-1.780381E-2$	$7.145376E-3$

Table 13 RMS magnitude and phase errors of third-order one-fourth differentiators

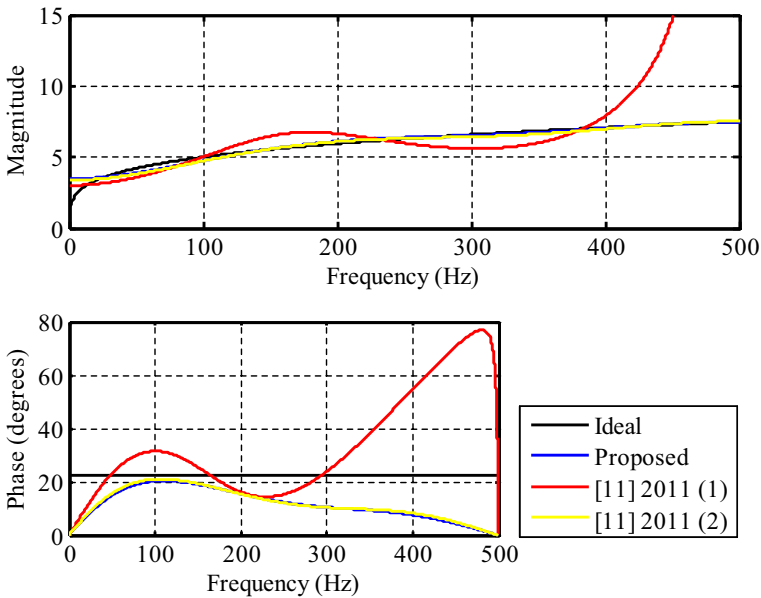| Differentiator	Magnitude RMS error	Phase RMS error
Proposed	0.243824	12.333105
[11] 2011 (1)	25.403590	22.665038
[11] 2011 (2)	0.255035	12.063852

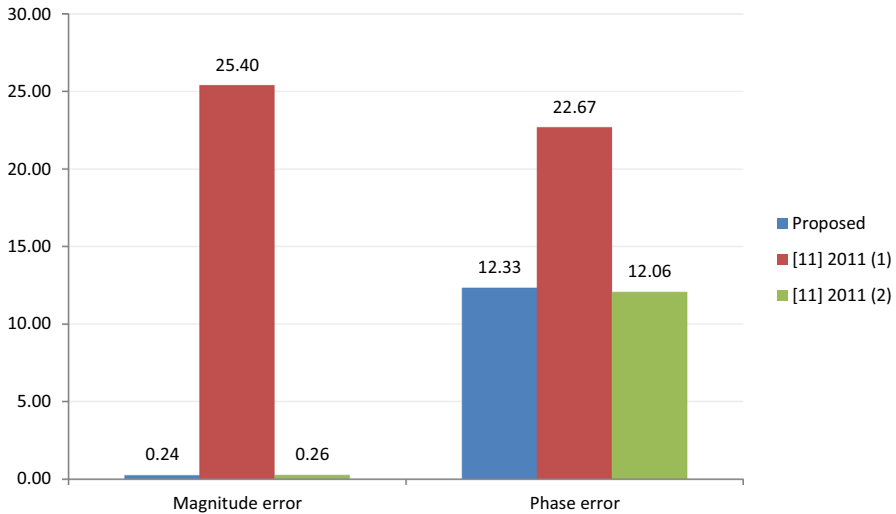**Fig. 18** Frequency responses of third-order one-fourth differentiators

Fig. 19 Comparison of magnitude and phase RMS errors of third-order one-fourth differentiators

equal sample size [26]. It may be noted that the *t* test method is developed without mathematical proof. Higher positive t value indicates the goodness of reference result with respect to the comparing result and vice versa. On the other hand, negative t value indicates the badness of reference result with respect to the comparing result. Furthermore, for a mean minimization problem, *t* test value will be positive if algorithm 1's mean is less than algorithm 2's mean; thus, algorithm 1 is better than algorithm 2. On the other hand, *t* test value being negative means algorithm 1 is poorer than the algorithm 2 with respect to their means [18,26].

The *t* Student statistic is described as:

$$t = \frac{\overline{X_2} - \overline{X_1}}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}} \tag{25}$$

where $\overline{X_1}$ and $\overline{X_2}$ are the means of the samples X_1 and X_2, respectively, S_1^2 and S_2^2 are the variances of sample 1 and sample 2, respectively, and finally, n_1 is the size of the sample 1 and n_2 is the size of the sample 2. In the present work, the NMSA is considered as the algorithm 1 (reference) and GA is considered as the algorithm 2, whose performance is to be compared. Table 19 presents the results of *t* test. The value $t = 14.1710$ gives sufficient statistical information to say that NMSA outperforms GA along the realized simulations.

Table 14 Designs of fifth-order one-fourth differentiators

Differentiator	Differentiator coefficients					
Proposed	b_0	b_1	b_2	b_3	b_4	b_5
	5.841132	−1.490550	−4.645373E−1	−3.474446E−1	−1.840672E−1	−1.657178E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	3.594986E−2	−2.011865E−3	1.529481E−4	−1.590108E−5	1.774913E−6
[11] 2011 (1)	b_0	b_1	b_2	b_3	b_4	b_5
	5.761000	−1.440250	−5.407366E−1	−3.148332E−1	−2.167277E−1	−1.624209E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	8.600000E−3	2.964140E−2	1.021899E−2	−1.122010E−1	2.908863E−1
[11] 2011 (2)	b_0	b_1	b_2	b_3	b_4	b_5
	5.794000	−1.449079	−5.432833E−1	−3.168626E−1	−2.178793E−1	−1.633512E−1
	a_0	a_1	a_2	a_3	a_4	a_5
	1	4.205000E−2	−1.781582E−2	7.147625E−3	−1.283045E−3	4.671773E−4

Table 15 RMS magnitude and phase errors of fifth-order one-fourth differentiators

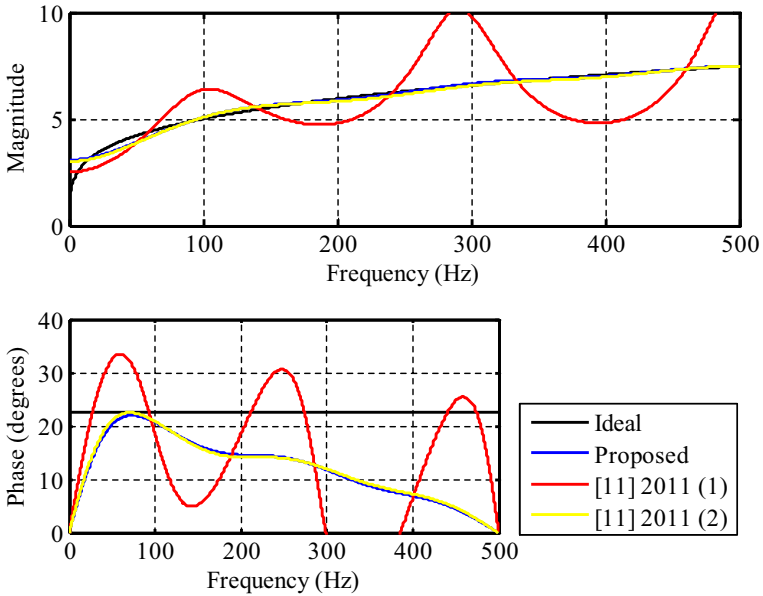| Differentiator	Magnitude RMS error	Phase RMS error
Proposed	0.175219	11.943589
[11] 2011 (1)	1.594565	16.536123
[11] 2011 (2)	0.184170	11.813988

Fig. 20 Frequency responses of fifth-order one-fourth differentiators
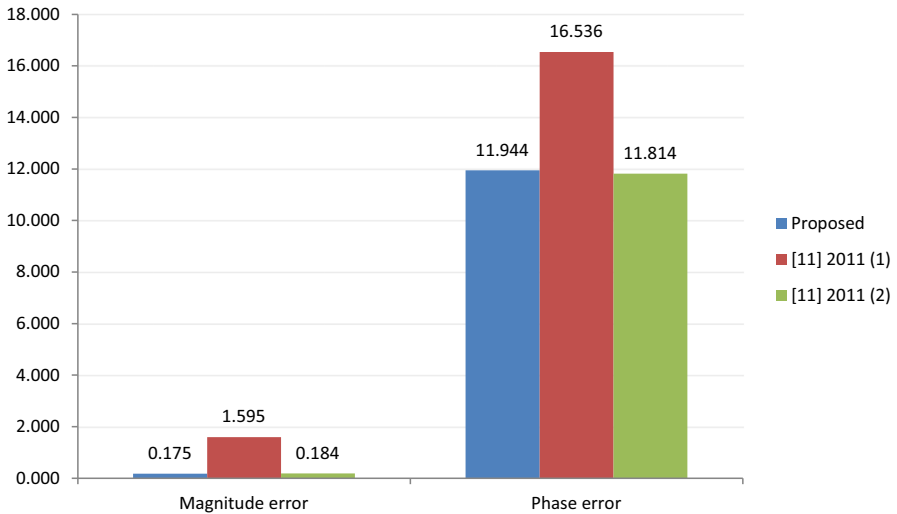

Fig. 21 Comparison of magnitude and phase RMS errors of fifth-order one-fourth differentiators

Table 16 Used GA parameters

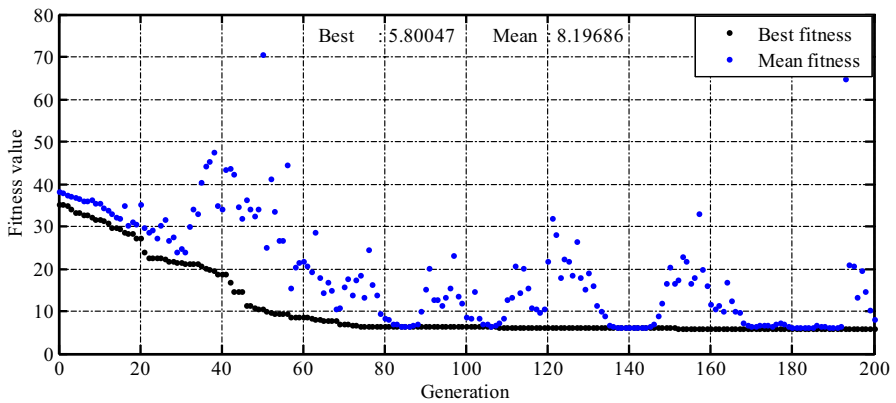| Parameters		Values
Coefficients $(b_0, b_1, b_2, b_3, a_0, a_1, a_2, a_3)$	Upper bounds	[35, 35, 35, 35, 35, 35, 35, 35]
	Lower bounds	[−18, −18, −18, −18, −18, −18, −18, −18]
Population size		50
Selection method		Stochastic uniform
Crossover function		Scattered
Mutation function		Constrain dependent
Generations		200

Fig. 22 Fitness function v/s generations (GA)

Table 17 Half differentiator of third-order designed using GA

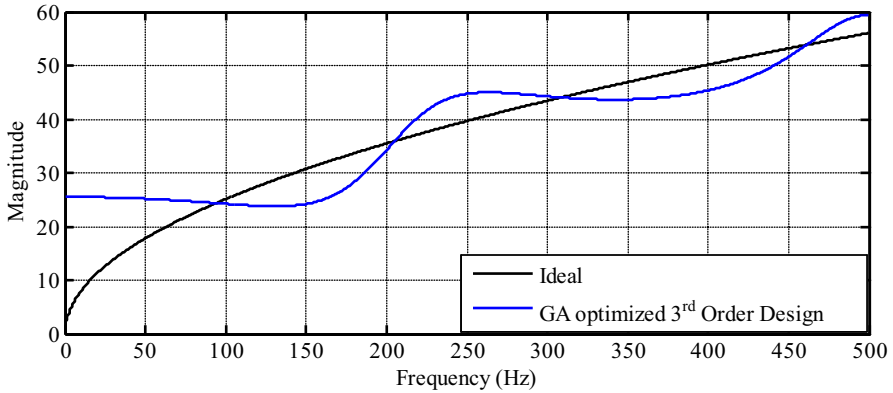| Differentiator	Differentiator coefficients				Magnitude RMS error
GA-optimized half differentiator of third order	b_0	b_1	b_2	b_3	5.80047
	25.2851	17.2285	33.4429	29.9089	
	a_0	a_1	a_2	a_3	
	1.0	1.0982	0.0117	0.6887	

Fig. 23 Frequency response of half differentiator of third order optimized using GA

Table 18 Comparison of NMSA and GA for third-order half differentiator designs

Optimization technique	Average computation time per iteration (s)	Mean RMS error in magnitude	Standard deviation in RMS error in magnitude
NMSA	0.6795	1.193116	0
GA	26.0796	13.29598	6.039116

Table 19 Comparison of NMSA and GA using t test

$\overline{X_1} =$ NMSA	$\overline{X_2} =$ GA
Null hypothesis	$\overline{X_1} \neq \overline{X_2}$
Test statistical, t	14.1710
Critical values t	± 2.0096
Value P	0.00000
Grades of freedom	49
95 % confidence interval	$10.38657 < \mu1 - \mu2 < 13.81916$

7 Conclusions

This paper presented a novel application of Nelder–Mead simplex optimization technique for designing efficient fractional-order differentiators. For discretization, the well-known Al-Alaoui operator has been used. To prove the efficacy of proposed method, six case studies have been presented, namely a half differentiator of third order, a half differentiator of fifth order, a one-third differentiator of third order, a one-third differentiator of fifth order, a one-fourth differentiator of third order, and a one-fourth differentiator of fifth order. The obtained simulation results have been compared with the recently published works, and significantly reduced RMS magnitude errors have been obtained in all of the investigated cases, thereby making the proposed technique a potential optimization method for designing fractional dif-

ferentiators. Further, a comparative study with genetic algorithm-based optimization technique has also been presented and the proposed method has been found to outperform GA-based designs. As a future research scope, various mapping functions, other than Al-Alaoui operator, can be chosen and corresponding performances can be investigated.

References

1. K. Assaleh, W. M. Ahmad, Modeling of speech signals using fractional calculus, in *Proceedings of 9th International Symposium on Signal Processing and Its Applications, (ISSPA'07)*, Sharjah, UAE (2007)
2. Z.J. Bortolot, An adaptive computer vision technique for estimating the biomass and density of loblolly pine plantations using orthophotography and LiDAR imagery., Doctorate thesis. Virginia Polytechnic Institute and State University, Virginia (2004)
3. Y.Q. Chen, K.L. Moore, Discretization schemes for fractional-order differentiators and integrators. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **49**(3), 363–367 (2002)
4. Y.Q. Chen, I. Petras, D. Xue, Fractional order control—a tutorial, in *Proceedings of the American Control Conference (ACC '09)*, Logan, UT, USA, pp. 1397–1411 (2009)
5. Y.Q. Chen, B.M. Vinagre, A new IIR-type digital fractional order differentiator. Signal Process. **83**(11), 2359–2365 (2003)
6. Y.Q. Chen, B.M. Vinagre, D. Xue, V. Feliu, *Fractional order Systems and Controls: Fundamentals and Applications* (Springer, London, 2010)
7. S. Das, B. Majumder, A. Pakhira, I. Pan, S. Das, A. Gupta, Optimizing continued fraction expansion based IIR realization of fractional order differ-integrators with genetic algorithm, in *Proceedings of Process Automation, Control and Computing (PACC)International Conference*, Coimbatore 20–22 July, 2011, pp. 1–6 (2011)
8. L. Debnath, Recent applications of fractional calculus to science and engineering. J. Math. Math. Sci. **54**, 3413–3442 (2003)
9. Z.E.A. Fellah, C. Depollier, Application of fractional calculus to the sound waves propagation in rigid porous materials: validation via ultrasonic measurement. Acta Acust. **88**(1), 34–39 (2002)
10. N.M. Fonseca Ferreira, F.B. Duarte, M.F.M. Lima, M.G. Marcos, J.A. Tenreiro Machado, Application of fractional calculus in the dynamical analysis and control of mechanical manipulators. Fract. Calc. Appl. Anal. **11**(1), 91–113 (2008)
11. M. Gupta, P. Varshney, G.S. Visweswaran, Digital fractional-order differentiator and integrator models based on first-order and higher order operators. Int. J. Circuit Theory Appl. **39**(5), 461–474 (2011)
12. B.T. Krishna, Studies on fractional order differentiators and integrators: a survey. Signal Process. **91**(3), 386–426 (2011)
13. V.V. Kulish, J.L. Lage, Application of fractional calculus to fluid mechanics. J. Fluids Eng. **124**(3), 803–806 (2002)
14. J.C. Lagarias, J.A. Reeds, M.H. Wright, P.E. Wright, Convergence properties of the Nelder–Mead simplex method in low dimensions. SIAM J. Optim. Soc. Ind. Appl. Math. **9**(1), 112–147 (1998)
15. G. Maione, A rational discrete approximation to the operator $s^{0.5}$. IEEE Signal Process. Lett. **13**(3), 141–144 (2006)
16. B. Mathieu, P. Melchior, A. Oustaloup, C. Ceyral, Fractional differentiation for edge detection. Signal Process. **83**(11), 2421–2432 (2003)
17. M.D. Ortigueira, An introduction to the fractional continuous-time linear systems: the 21st century systems. IEEE Circuits Syst. Mag. **147**(1), 19–26 (2000)
18. S.K. Saha, S.P. Ghoshal, R. Kar, D. Mandal, Cat swarm optimization algorithm for optimum linear phase FIR filter design. ISA Trans. **52**, 781–794 (2013)
19. R. Sharma, K.P.S. Rana, V. Kumar, Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator. Expert Syst. Appl. **41**(9), 4274–4289 (2014)
20. H. Sheng, Y.Q. Chen, FARIMA with stable innovations model of Great Salt Lake elevation time series. Signal Process. **91**(3), 553–561 (2011)
21. H. Sheng, Y.Q. Chen, T.S. Qiu, *Fractional Processes and Fractional-Order Signal Processing* (Springer, London, 2012)

22. J.I. Suarez, B.M. Vinagre, A.J. Calderon, C.A. Monje, Y.Q. Chen, Using fractional calculus for lateral and longitudinal control of autonomous vehicles, in *Computer Aided System Theory (EUROCAST 2003)*, pp. 337–348, (2003). doi:10.1007/978-3-540-45210-2_31
23. A. Tofighi, H.N. Pour, Expansion and the fractional oscillator. Phys. A **374**(1), 41–45 (2007)
24. B.M. Vinagre, Y.Q. Chen, I. Petras, Two direct Tustin discretization methods for fractional-order differentiator/integrator. J. Frankl. Inst. **340**(5), 349–362 (2003)
25. G.S. Viswesaran, P. Varshney, M. Gupta, New approach to realize fractional power in z-domain at low frequency. IEEE Trans. Circuits. Syst. II Express Briefs **58**(3), 179–183 (2011)
26. R.E. Walpole, R.H. Myer, S.L. Myer, K. Ye, *Probability and Statistics for Engineers and Scientists* (Macmillan, New York, 1978)
27. M.H. Wright, Nelder, *Mead, and the Other Simplex Method, Documenta Mathematica, Extra Volume: Optimization Stories* (ISMP, 2012), pp. 271–276
28. R. Yadav, M. Gupta, Design of fractional order differentiators and integrators using indirect discretization approach, in *Proceedings of International Conference on Advances in Recent Technologies in Communication and Computing, (ARTCom' 10)*, Kottayam, Kerala (2010)