International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:5, 2010

# Frequency-Domain Design of Fractional-Order FIR Differentiators

Wei-Der Chang, Dai-Ming Chang, Eri-Wei Chiang, Chia-Hung Lin, and Jian-Liung Chen

***Abstract*****—**In this paper, a fractional-order FIR differentiator design method using the differential evolution (DE) algorithm is presented. In the proposed method, the FIR digital filter is designed to meet the frequency response of a desired fractal-order differentiator, which is evaluated in the frequency domain. To verify the design performance, another design method considered in the time-domain is also provided. Simulation results reveal the efficiency of the proposed method.

***Keywords*****—**Fractional-order differentiator, FIR digital filter, Differential evolution algorithm.

## I. Introduction

THE research topic regarding the fractional calculus has been proposed since about 17th century. The fractional calculus is concerned with mathematical analyses and operations for noninteger-order integration and differentiation [1]. In fact, many systems in the real world are known to show the fractional-order dynamics, i.e., their dynamical equations involving noninteger-order derivatives, such as electrode-electrolyte polarization, electromagnetic waves, viscoelastic systems, mass diffusion, heat conduction, fractal porous media, and so forth [2][3]. On the other hand, the actual implementation of fractional-order derivatives on electrical circuit devices could also be found in the literatures as [4] and [5]. Such a device is called the fractance and is with the intermediate property between resistance and capacitance. Due to its successive developments in the mathematical theory and real implementation, the concept and application about the fractional-order derivative have drawn the attention of some researchers. In [6], for example, the author proposed an effective and easy-to-use means to the time-domain analysis of fractional-order systems. A new type of PID controller with fractional-order integrator and differentiator was first presented in the study. Furthermore, a generalized van der Pol chaotic system with fractional orders was investigated in [3]. The chaotic dynamics of the proposed fractional-order systems could be observed and verified via numerical analyses in the phase portraits, bifurcation diagrams, and Poincare maps, etc.

Wei-Der Chang*, Dai-Ming Chang, and Eri-Wei Chiang are with the Department of Computer and Communication, Shu-Te University, Kaohsiung 824, Taiwan. (Tel: +886-7-6158000 ext. 4808; fax: +886-7-6158000 ext. 4899; e-mail: wdchang@stu.edu.tw).

Chia-Hung Lin and Jian-Liung Chen are with the Department of Electrical Engineering, Kao-Yuan University, Lu-Chu Hsiang, Kaohsiung County 821, Taiwan. (e-mail: eechl53@educities.edu.tw).

As for the fractional-order digital differentiator, it is an extended version of general digital differentiator, which modifies the integer order to the fractional case. To deal with the digital differentiating system, it plays an important role and provides a more flexibility to design. The main function of a fractional-order digital differentiator is to provide a fractional-order differentiation on a given digital input signal. Compared with the general integer-order filter design, a fractional-order design is more difficult and complicated. In recent years, many researchers have paid attention to this study and also proposed some new design algorithms as demonstrated in [7]-[9]. In [7], the author presented a time-domain method for designing the fractional-order FIR digital differentiator. An input signal is first expanded using a Taylor series, and further based on this series its impulse response may be computed from the linear equations of Vandermonde form. The resulting FIR filter is then an approximate model of a desired fractional-order digital differentiator. In addition, for a uniformly sampled polynomial input signal a discrete-time signal processing system was designed as fractional-order derivatives of Riemann-Liouville [8]. The proposed scheme is discussed and analyzed in the time domain. Another time-domain design method for fractional differentiators may be also found in [9]. They used the least-squares method to design a digital rational approximation; that is, infinite impulse response (IIR) filter, to be a fractional-order integrator or differentiator. Some approximation approaches, such as Pade approximation, Prony's method, and Shanks' method are suggested. Different from these studies, this paper provides alternative frequency-domain design method for such a fractional-order digital differentiator. An optimal algorithm called the differential evolution (DE) is proposed to solve this issue. Our main purpose is that FIR filter coefficients are efficiently determined using the proposed DE algorithm so that its corresponding magnitude response may satisfy a desired fractional-order digital differentiator behavior. The DE algorithm possesses many numerical features such as using real-valued manipulations, global searching optimization, and quicker convergence [10]-[13]. The detailed descriptions for DE algorithms will further be demonstrated in the following section.

## II. DESCRIPTION OF FRACTIONAL-ORDER DIFFERENTIATOR

In the beginning, some mathematical expressions related to the integer-order and fractional-order derivatives described in the time domain is introduced. For a power function of time, $t^{\lambda}$, its $q$th order derivative where $q$ is a positive integer can be expressed by

$$D_t^q t^{\lambda} = \frac{d^q}{dt^q} t^{\lambda} = \lambda(\lambda-1)(\lambda-2)\cdots(\lambda-q+1)t^{\lambda-q} = \frac{\lambda!}{(\lambda-q)!} t^{\lambda-q} \quad (1)$$

According to Eq. (1), the positive integer $q$ can further be generalized to an arbitrary order $\alpha$ when $\lambda!$ and $(\lambda-q)!$ in Eq. (1) are replaced by the gamma functions $\Gamma(\lambda+1)$ and $\Gamma(\lambda-\alpha+1)$, respectively [7], and it becomes

$$D_t^{\alpha} t^{\lambda} = \frac{d^{\alpha}}{dt^{\alpha}} t^{\lambda} = \frac{\Gamma(\lambda+1)}{\Gamma(\lambda-\alpha+1)} t^{\lambda-\alpha} . \quad (2)$$

Furthermore, let us consider a time function $f(t)$ and suppose that its Taylor series expansion exists. It is interesting to explore its fractional-order derivative with respect to time $t$. First, taking Taylor series expansion for $f(t)$ at $t=0$ yields

$$f(t) = f(0) + f'(0)t + \frac{1}{2!} f''(0)t^2 + \cdots + \frac{1}{\lambda!} f^{(\lambda)}(0)t^{\lambda} + \cdots = \sum_{\lambda=0}^{\infty} \frac{1}{\lambda!} f^{(\lambda)}(0)\, t^{\lambda} . \quad (3)$$

Using (2), the $\alpha$th fractional-order derivative of $f(t)$ can be obtained by

$$D_t^{\alpha} f(t) = \frac{d^{\alpha}}{dt^{\alpha}} f(t) = \sum_{\lambda=0}^{\infty} \frac{1}{\lambda!} f^{(\lambda)}(0) \frac{\Gamma(\lambda+1)}{\Gamma(\lambda-\alpha+1)} t^{\lambda-\alpha} . \quad (4)$$

Eq. (4) represents an analytically mathematical formula for solving the fractional derivative of a time function $f(t)$.

Alternatively, the aim of this paper is to propose a fractional-order differentiator design method considered in the frequency domain. The architecture of a digital FIR filter is employed. The difference equation of FIR filter is expressed by

$$y[n] = h[0]x[n] + h[1]x[n-1] + h[2]x[n-2] + \cdots + h[N]x[n-N], = \sum_{k=0}^{N} h[k]x[n-k], \quad (5)$$

where $x$ is the input signal, $y$ is the output signal of FIR filter, $N$ denotes the filter order, and $h[k]$, $k = 0,1,\cdots,N$, is the impulse response sequence (also called the filter coefficient) which dominates the filter characteristic. In order for analysis in the frequency domain, it further needs to take $z$ transform on both sides of Eq. (5) getting

$$Y[z] = \sum_{k=0}^{N} h[k]X[z]z^{-k} . \quad (6)$$

Consequently, the transfer function of FIR digital filter, $H[z]$, is of the form

$$H[z] = \frac{Y[z]}{X[z]} = \sum_{k=0}^{N} h[k]z^{-k} . \quad (7)$$

Moreover, if let $z = e^{j\Omega}$ where $\Omega$ represents the digital frequency, the frequency response of the filter is derived as

$$H(\Omega) = \sum_{k=0}^{N} h[k]e^{-jk\Omega} . \quad (8)$$

For convenience, let $\Theta = [\theta_1, \theta_2 \cdots, \theta_{N+1}] = [h[0], h[1], \cdots h[N]]$ be a parameter vector. This vector will be used in the DE algorithm.

On the other hand, a prototype of an ideal fractional-order differentiator may be characterized by

$$D(\Omega) = (j\Omega)^{\alpha}, \; 0 \le \Omega \le \pi , \quad (9)$$

where $\alpha$ is a fractional-order value. Our main aim is that the impulse response $h[k]$ of FIR filter can be determined well so that its corresponding frequency response has a characteristic of fractional-order derivative as described by Eq. (9).

## III. DE-BASED DESIGN METHOD

The differential evolution is a simple and effective means for solving engineering optimization problem, which was proposed by Storn and Price in 1997 [10]. There are three fundamental variables in the DE algorithm, including population size $S$, mutation constant factor $F$, and crossover rate $C$. In the beginning, the algorithm randomly generates an initial population of $S$ parameter vectors. These parameter vectors will be repeatedly updated to produce new better offspring. To achieve that, some important evolutionary operations, including mutation, crossover and selection, are performed, which are kind of similar to genetic algorithms. The convergence of this algorithm toward the optimal solution is guided only by the size of the cost function. Thus it is considerably reasonable to define the cost function as

$$CF = \int_0^{\pi} \left( |D(\Omega)| - |H(\Omega)| \right)^2 d\Omega , \quad (10)$$

for the fractional-order FIR differentiator design. Minimizing the defined cost function by using the proposed DE algorithm is the main purpose of this paper. The following will clearly explain the evolutionary operations of the DE algorithm [13].

To perform the mutation operation, three different parameter vectors $\Theta_{\alpha}$, $\Theta_{\beta}$, and $\Theta_{\gamma}$ are randomly selected from the population. The vector $\Theta_{\alpha}$ is mutated to be the donor vector $V = [v_1, v_2, \cdots, v_{N+1}]$ by adding the weighted difference between another two parameter vectors $\Theta_{\beta}$ and $\Theta_{\gamma}$, i.e.,

$$V = \Theta_{\alpha} + F \cdot (\Theta_{\beta} - \Theta_{\gamma}), \quad (11)$$

where $F$ is a positive constant and called the mutation constant factor. Then this donor vector further goes through a crossover operation with a target vector $\Theta = [\theta_1, \theta_2 \cdots, \theta_{N+1}]$ in order for producing a new vector, called the trial vector. It needs to replace some elements of target vector $\Theta$ by the elements of donor vector $V$ correspondingly. To achieve that, let

$W = [w_1, w_2 \cdots, w_{N+1}]$ be a trial vector and generate a set of $N+1$ random numbers $\{r_1, r_2, \cdots, r_{N+1}\}$ uniformly chosen from the interval $(0, 1)$. According to these random numbers, another set of binary sequences $\{b_1, b_2, \cdots, b_{N+1}\}$ is constructed by setting

$$b_k = \begin{cases} 1 & if\ r_k < C \\ 0 & otherwise \end{cases}, \text{ for } k = 1, 2, \cdots, N+1, \quad (12)$$

where $C \in (0, 1)$ is called the crossover rate. Finally, the trial vector $W$ is obtained according to the following formula

$$w_k = \begin{cases} \theta_k & if\ b_k = 1 \\ v_k & if\ b_k = 0 \end{cases}, \text{ for } k = 1, 2, \cdots, N+1. \quad (13)$$

This completes the crossover operation. Having determined the trial vector *W*, a selection operation is further performed. The cost function of the resulting trial vector is evaluated and then compared with that of the target vector, and if the cost function of the trial vector is less than that of the target vector (i.e., the trial vector is better than the target vector), then the trial vector replaces the target vector; otherwise, the trail vector is rejected and the target vector still survives in the next generation. In the DE algorithm, the complete execution of the mutation, crossover and selection operations on each parameter vector is referred to as "one generation". The iterations of the DE algorithm are terminated when the pre-specified number of generations *G* has been achieved. The following shows the whole steps of the DE-based optimization process for designing the fractional-order FIR digital differentiator:

1. Randomly generate an initial population of *S* parameter vectors chosen from the interval $[-1, 1]$.
2. If the pre-specified number of generations *G* is completed, then stop the algorithm.
3. For $i = 1$ to *S*
   - Evaluate the cost function $CF(\Theta_i)$ of each target vector $\Theta_i$ in the population using Eq. (10).
   - Generate a donor vector *V* using Eq. (11).
   - Obtain a set of binary sequences $\{b_1, b_2, \cdots, b_{N+1}\}$ using Eq. (12).
   - Apply the crossover formula of Eq. (13) to obtain a trial vector *W*.
   - Evaluate the cost function $CF(W)$ of the trial vector *W* using Eq. (10) as well.
   - Perform the selection operation for both *W* and $\Theta_i$ to obtain new offspring $\Theta_i^{new}$. If $CF(W) < CF(\Theta_i)$, then $\Theta_i^{new} = W$, otherwise $\Theta_i^{new} = \Theta_i$.

   End

   For $i = 1$ to *S*;

   $\Theta_i = \Theta_i^{new}$.

   End
4. Go back to Step 2.

## IV. Simulation Results

In order to verify the proposed design method, some simulations are required. In the present search, the values assigned to the variables of the DE algorithm for designing the fractional-order digital FIR differentiator are given by population size $S = 50$, number of generations $G = 3000$, mutation constant factor $F = 0.2$, and crossover rate $C = 0.5$, respectively, for simulations. The FIR filter with order $N = 10$ is designed to match the digital fractional-order differentiator with $\alpha = 1.5$ as in Eq. (9). The DE-based design algorithms are numerically programmed by using Matlab software under the PC environments. After executing the DE algorithm, Fig. 1 shows the convergence of the cost function with respect to number of generations and Fig. 2 depicts the magnitude responses of the designed FIR filter and the desired fractional-order digital differentiator with $\alpha = 1.5$, respectively. A time-domain design method, on the other hand, proposed by [7] is also compared. The simulation result is further shown in Fig. 3. It can easily be seen from these two figures that better approximation, especially for higher frequency response, is achieved by our proposed method than another time-domain design method.

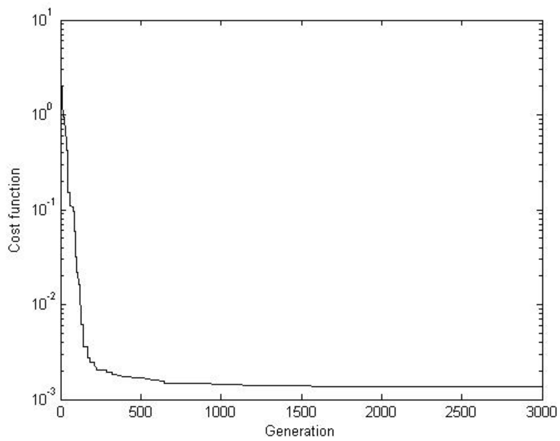


Fig. 1 Convergence of cost function

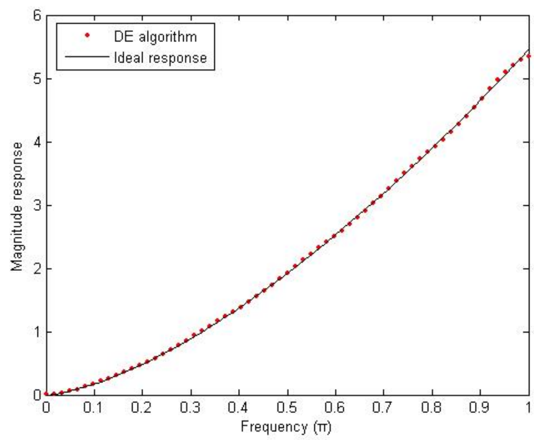


Fig. 2 Magnitude responses of the ideal and obtained fractional-order differentiators by the DE algorithm

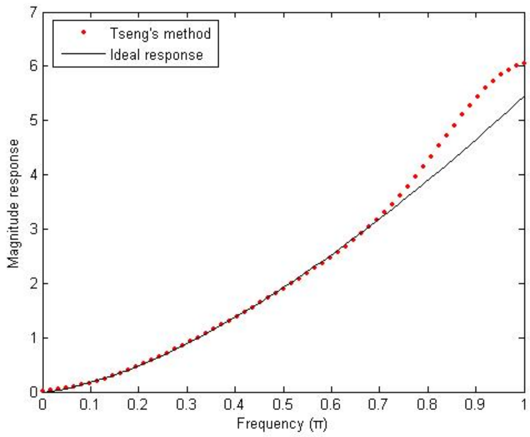


Fig. 3 Magnitude responses of the ideal and obtained fractional-order differentiators by Tseng's method [7]

## V. Conclusion

This paper has successfully applied the DE algorithm to the design of the FIR digital fractional-order differentiator. Filter coefficients are iteratively evolved by the DE algorithm so that its corresponding magnitude response matches the desired fractional-order digital differentiator. From simulation results, it can be concluded that the DE-based design method is better than another design method which works in the time domain.

## Acknowledgment

This work was supported in part by the National Science Council of Taiwan under Grant NSC 98-2221-E-366-012.

## References

[1] K. B. Oldham and J. Spanier, The fractional Calculus. New York: Academic, 1974.
[2] C. Hwang, J. F. Leu, and S. Y. Tsay, "A note on time-domain simulation of feedback fractional-order systems," IEEE Trans. on Automatic Control, vol. 47, pp. 625-631, 2002.
[3] Z. M. Ge and M. Y. Hsu, "Chaos in a generalized van der Pol system and in its fractional order system," Chaos, Solitons and Fractals, vol. 33, pp. 1711-1745, 2007.
[4] M. Nakagava and K. Sorimachi, "Basic characteristics of a fractance device," IEICE Trans. Fundamentals, vol. E75-A, pp. 1814-1818, 1992.
[5] H. Samavati, A. Hajimiri, A. Shahani, G. Nasserbakht, and T. Lee, "Fractal capacitors," IEEE J Solid-State Circ., vol. 33, pp. 2035-2041, 1998.
[6] I. Podlubny, "Fractiona-order systems and $PI^{\lambda}D^{\mu}$ -controllers," IEEE Trans. on Automatic Control, vol. 44, pp. 208-214, 1999.
[7] C. C. Tseng, "Design of fractional order digital FIR differentiators," IEEE Signal Processing Letters, vol. 8, pp. 77-79, 2001.
[8] S. Samadi, M. Omair Ahmad, and M. N. S. Swamy, "Exact fractional-order differentiators for polynomial signals," IEEE Signal Processing Letters, vol. 11, pp. 529-523, 2004.
[9] R. S. Barbosa, J. A. Tenreiro Machado, and M. F. Silva, "Time domain design of fractional differintegrators using least-squares," Signal Processing, vol. 86, pp. 2567-2581, 2006.
[10] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization, vol. 11, pp. 341-359, 1997.
[11] R. Storn, "System design by constraint adaptation and differential evolution," IEEE Transactions on Evolutionary Computation, vol. 3, pp. 95-99, 1999.
[12] S. L. Cheng and C. Hwang, "Optimal approximation of linear systems by a differential algorithm," IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, vol. 31, pp. 698-707, 2001.
[13] W. D. Chang, "Parameter identification of Chen and Lü systems: a differential evolution approach," Chaos, Solitons & Fractals, vol. 32, pp. 1464-1471, 2007.