

Digital Fractional Order PID Controller Design And Realization

Abdelelah Kidher Mahmood

Assistant professor

Dept. of Electrical Engineering / College of Engineering / University of Mosul
Kmabdelelah@yahoo.com

Bassam Fadel Mohammed

Msc student

bassam7219@yahoo.com

Abstract

In this paper a special kind of PID controller has been designed and implemented which is called fractional order PID (FOPID) controller whose derivative and integral are fractional rather than integers. The FOPID controller has five parameters which have been tuned by using an intelligent particle swarm optimization (PSO) algorithm by minimizing the fitness function given as integral of time weighted absolute error (ITAE). The digital fractional order PID (DFOPID) controller in discrete time with suitable sampling period has been attained utilizing special approximation method called continued fraction expansion (CFE) that lead to convert the s domain transfer function into z domain using MATLAB tools. As a case of study the DFOPID controller has been designed and realized by software in C language on PIC microcontroller for DC motor as a position control. The results showed software fulfillment of DFOPID, with alike time domain performance of closed loop system for both continuous and discrete.

Key words: Continued fraction expansion (CFE), fractional order controller, fractional order toolbox for MATLAB, MATLAB Simulink, PSO algorithm.

تصميم وتحقيق المسيطر التناسبي التكاملي التفاضلي الكسري الرقمي

بسام فاضل محمود

طالب ماجستير

د. عبد الإله خضر محمود

أستاذ مساعد

قسم الهندسة الكهربائية / كلية الهندسة / جامعة الموصل

الخلاصة

في هذا البحث تم تصميم وتحقيق نوع خاص من المسيطرات وهو المسيطر التناسبي التكاملي التفاضلي الكسري الرقمي حيث ان التكامل والتفاضل ذو مرتبة كسرية بدل المرتبة الصحيحة ، ويتكون هذا المسيطر من خمسة معاملات تم ايجاد قيمها المثلى باستخدام احد الطرق الذكية وهي طريقة التحشيد الامثل للجسيمات و ذلك باستخدام القيمة الصغرى المثلى لتكامل القيمة المطلقة لزمنا اشارة الخطأ. وقد تم تحويل المسيطر الى الصيغة الرقمية باستخدام طريقة تقريب خاصة وهي التوسيع الكسري المستمر بالأستعانة ببرنامج الماتلاب ، وباستخدام لغة البرمجة سي تم برمجة المسيطر الى المعالج الدقيق للسيطرة على الموقع رقمياً و باستخدام محرك التيار المستمر ذو المغناط الدائمة. وقد تبين تشابه أداء المسيطر التناسبي التكاملي التفاضلي الكسري الرقمي بالمقارنة مع المسيطر التناسبي التكاملي التفاضلي الكسري بصيغته المستمرة للمنظومة كدارة مغلقة. الكلمات الدالة: التوسيع الكسري المستمر، خوارزمية التحشيد الامثل للجسيمات، صندوق العدة ذو المرتبة الكسرية لبرنامج الماتلاب، المسيطر التناسبي التكاملي التفاضلي الكسري، محاكاة برنامج الماتلاب.

Received: 30 - 5 - 2013

Accepted: 25 - 9 - 2013

1- Introduction

In industries and manufactures were the process control especially position control mostly requires PID controllers as they are simple and easily tuned. The few parameters like proportional, integral and derivative actions may be tuned manually or self-adaptive tuned automatically. By using this controller, the time domain performance, like small percentage overshoot and small settling time can be obtained. Much publicized conventional tuning methods were developed using time domain approach by Zeigler and Nichol's in 1942 and frequency domain approach by Hagglund and Astrom in 1955 [1]. By generalizing the derivative and integer orders, from the integer field to non-integer numbers, the fractional order PID (FOPID) control is obtained, the performance of the PID controller can be improved by using fractional order derivatives and integrals. The benefit of (FOPID) controller is flexible to design, more robust [2]. The most important advantages of the fractional order PID (FOPID) controller is the better control of dynamical systems and less sensitivity to changes in parameters of a control system [3,4]. Calculus generation to fractional, was proposed by Leibniz and Hopital for the first time. After words, the systematic studies in this field by many researchers such as Liouville (1832), Holmgren (1864) and Riemann (1953) were performed [5]. Podlubny has proposed a generalization of the PID controller as $PI^\lambda D^\mu$ controller which is known as fractional order PID controller, where λ is the non-integer order of integrator and μ is the non-integer order of the differentiator term, he also demonstrated that the $PI^\lambda D^\mu$ controller has better response than classical PID controller [4]. Many tuning techniques for obtaining the parameters of the controllers were introduced during last few decades. Tuning methods of $PI^\lambda D^\mu$ controllers are recent research subjects, the researchers oriented to the classical optimization and intelligent methods. In this paper the Particle Swarm Optimization (PSO) algorithm has been used to tune the parameter of $PI^\lambda D^\mu$ controller in order to get an optimum time domain specifications in which integral of time weighted absolute error (ITAE) has been minimized, where the five parameters (K_p , K_i , K_d , λ and μ) found directly by utilizing PSO algorithm. The key step in digital implementation of an FOC is the numerical evaluation or discretization of the fractional-order differentiator s^α [2]. The digital realization of the FOPID controller done by coding the discrete difference equation to PIC processor.

2- Fractional order controller (FOC)

A fractional order system is that system described by the following fractional order differential equation:

$$a_n D^{\alpha n} f(x) + a_{n-1} D^{\alpha(n-1)} f(x) + \dots = b_n D^{\beta n} f(x) + b_{n-1} D^{\beta(n-1)} f(x) + \dots \quad (1)$$

where $D^{\alpha n} = {}_0 D_t^{\alpha n}$, is called the fractional derivative of order αn with respect to variable t and with the starting point $t = 0$, [6]. The fractional order $PI^\lambda D^\mu$ controller can be described by the fractional order differential equation :

$$u(t) = K_p e(t) + K_i D^{-\lambda} e(t) + K_d D^\mu e(t) \quad (2)$$

Applying Laplace transform, the transfer function of the controller can be expressed by:

$$C_f(s) = \frac{U(s)}{E(s)} = K_p + K_i s^{-\lambda} + K_d s^\mu \quad (3)$$

The Figure (1) shows the control possibilities using a fractional-order PID controller, extending the four control points of the classical PID to the range of control points of the quarter-plane defined by selecting the values of λ and μ [2].

The direct discretization method using finite memory length expansion was given by Grünwald-Letnikov (GL) definition. In general, the discretization of fractional-order differentiator/integrator $s^{\pm\alpha}$, ($\alpha \in \mathbb{R}$, where \mathbb{R} is a real number), can be expressed by the generating function as $s = w(z^{-1})$. The discrete approximation of the (FOPID) controller of equation (3) can be expressed as:

$$C_f(z) = K_p + K_i(w(z^{-1}))^{-\lambda} + K_d(w(z^{-1}))^{\mu} \quad (4)$$

where K_p is a proportional constant, K_i is an integral constant, K_d is a derivative constant, λ is an integer order of integrator, μ is an integer order of the differentiator and $w(z^{-1})$ is a generating function [7].

3- Particle swarm optimization (PSO) an overview

One of the update intelligent optimization technique is the particle swarm optimization that emulate the swarms of birds, fish, etc. to find the food. The main object of PSO is to solve the computationally hard Optimization problems, where it is a robust optimization technique based on the movement and intelligence of swarms and applied successfully to a wide variety of search and optimization problems. The PSO developed in 1995 by James Kennedy and Russ Eberhart. The algorithm adopted uses a set of particles flying over a search space to locate a global optimum, where a swarm of n particles communicate either directly or indirectly with one another using search directions, in each iteration of PSO, each particle updates its position depending on three components, and determines its velocity using, previous velocity, best previous position, and the best previous position of its neighborhood. Figure (2) illustrate the flow chart of PSO algorithm. The basic concept of PSO lies in accelerating each particle toward the best position found by it so far (pbest) and the global best position (gbest) obtained so far by any particle, with a random weighted acceleration at each time step, this is done by the equations (5) and (6):

$$V_{t+1} = W * V_t + C_1 * \text{rand}(0,1) * (pbest - X_t) + C_2 * \text{rand}(0,1) * (gbest - X_t) \quad (5)$$

$$X_{t+1} = X_t + V_{t+1} \quad (6)$$

where: gbest = Global Best Position.

pbest = Self Best Position.

C_1 and C_2 = Acceleration Coefficients.

W = Inertial Weight.

Once the particle computes the new X_t it then evaluates its new location. If $\text{fitness}(X_t)$ is better than $\text{fitness}(pbest)$, then $pbest = X_t$ and $\text{fitness}(pbest) = \text{fitness}(X_t)$, in the end of iteration the $\text{fitness}(gbest) = \text{the better fitness}(pbest)$ and $gbest = pbest$ [8].

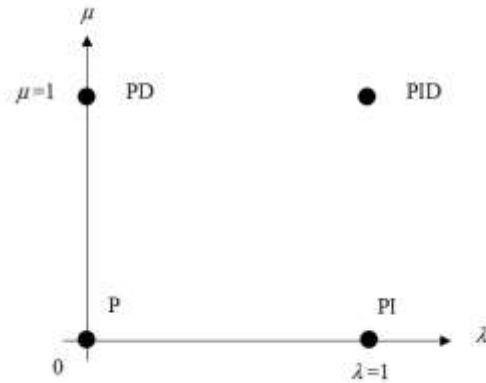


Figure (1): (Fractional order PID vs classical PID)

4- Design of PI^λD^μ controller

In order to design the (FOPID) controller the five parameter (Kp, Ki, Kd, λ and μ) of the controller must be tuned or computed by optimization. The MATLAB simulation of a negative unity feedback control system for a plant shown in Figure(3), where the fractional order PID (FOPID) controller G_c(s) implemented by using fractional control toolbox [9]. The integral of time weighted absolute error (ITAE) has been chosen as objective function were applied to the plant of a permanent magnet DC motor with gearbox given as G(s) were simulation by Simulink in MATLAB toolbox.

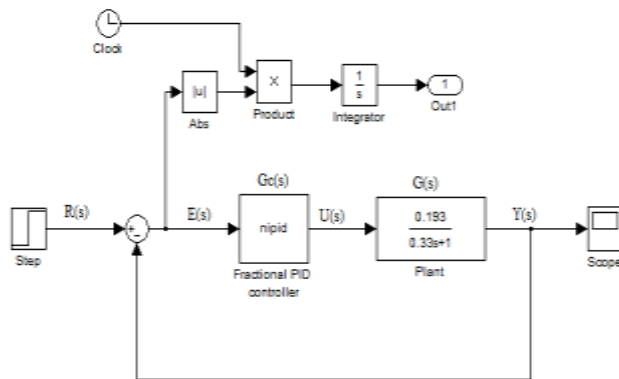


Figure (3): (Negative unity feedback FOPID control system)

The design FOPID controller given as:

$$G_c(s) = K_p + \frac{K_i}{s^\lambda} + K_d s^\mu \tag{7}$$

and the plant transfer function given as:

$$G(s) = \frac{0.193}{0.33s+1} \tag{8}$$

that must minimized the given objective function:

$$ITAE = \int_0^\infty t|e(t)|dt \tag{9}$$

5 –Particle Swarm Optimization (PSO) based tuning of FOPID controller

The PSO algorithm method has been implemented as M-file which interconnected to simulink model where the FOPID controller parameters are computed and fed to the GUI of the controller. The optimization performed with this initial parameter, number of particles (30), number of dimensions (5), maximum iteration (100), C₁=1, C₂=3, with the objective function (ITAE). The initial values of five parameters of the fractional order PID controller (Kp, Ki, Kd, λ and μ) will be generate in PSO program and submitted in simulation diagram

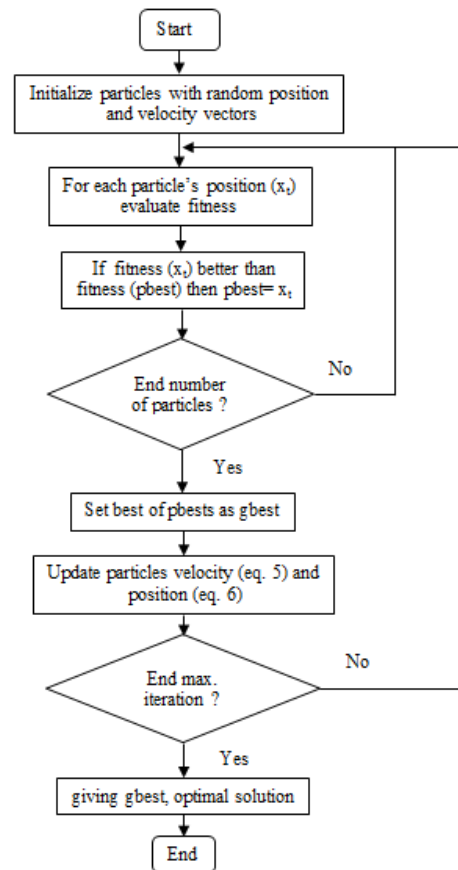


Figure (2): (Flow chart of PSO algorithm)

in figure (3) and running the simulation automatically then compute the objective function (ITAE) and go back with value of (ITAE) to PSO program to improve the value of (K_p , K_i , K_d , λ and μ) and so on, in the end of iteration the five parameters of the fractional order PID controller (K_p , K_i , K_d , λ and μ) has been obtained directly according to the minimum value of objective function (ITAE). The obtained results shown in the table (1).

Table (1): (Parameters of FOPID controller obtained by PSO algorithm)

Tuning Method and Controller	Parameters					
	K_p	K_i	K_d	λ	μ	ITAE
PSO Algorithm, (FOPID)	499.2228	1.3155	2.244	0.981	0.52	0.000064

The unit step response of the system illustrated in figure (4). The response of the system gives valuable information such as Maximum overshoot ($M_{o,s}$ %), rise time (T_r), peak time (T_p) and settling time (T_s). It can be observed from the table (2).

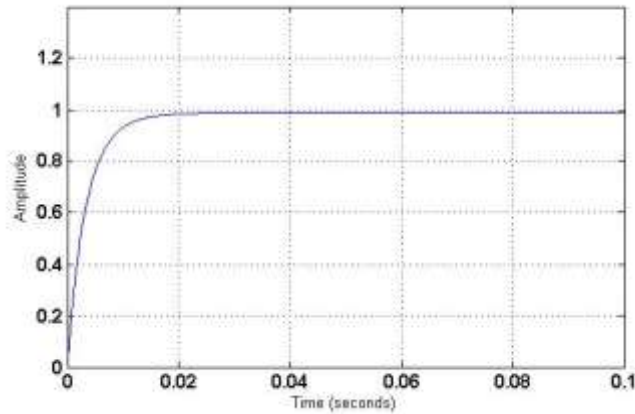


Figure (4): (Step response of negative unity feedback FOPID control system)

Table (2): (Step Response Specification of FOPID Control system)

System with Tuning Method and Controller	Step Response Specification	
	Rise time (second) (T_r)	Settling Time (second) (T_s)
PSO Algorithm, (FOPID)	0.008	0.0178

The continuous transfer function of the (FOPID) controller in equation (7) can be expressed as:

$$G_c(s) = 499.2228 + \frac{1.3155}{s^{0.981}} + 2.244s^{0.52} \quad (10)$$

6- Digital realization of (FOPID) controller

In general, there are two discretization methods: direct discretization and indirect discretization. In indirect discretization methods, two steps are required [2], i.e. frequency domain fitting in continuous time domain first and then discretizing the fit s-transfer function.

Other frequency-domain fitting methods can also be used but without guaranteeing the stable minimum-phase discretization. Existing direct discretization methods include the application of the direct power series expansion (PSE) of the Euler operator, continuous fractional expansion (CFE) of the Tustin operator, and numerical integration-based method [2]. The Tustin operator-based discretization scheme exhibits large errors in high-frequency range [2]. A new mixed scheme of Euler and Tustin operators is proposed in [10] which applies the Al-Alaoui operator [11]. By using this method the discrete transfer function approximating fractional order operators can be expressed as:

$$D^{\pm\alpha}(z) = \left(\frac{8}{7T}\right)^{\pm\alpha} CFE \left\{ \left(\frac{1-z^{-1}}{1+z^{-1}/7} \right)^{\pm\alpha} \right\}_{p,q} = \left(\frac{8}{7T}\right)^{\pm\alpha} \frac{P_p(z^{-1})}{Q_q(z^{-1})} \quad (11)$$

where T is the sampling period, P and Q are polynomials of degrees p and q, respectively, in the variable z^{-1} .

The continuous transfer function of (FOPID) controller which is given in equation (10), has been transformed to discrete transfer function implementing equation (11) and with sampling period T=0.001 second, the form in equation (12) is got:

$$G_c(z) = \frac{U(z)}{E(z)} = \frac{586.67 - 1714.79z^{-1} + 1854.451z^{-2} - 896.844z^{-3} + 179.2z^{-4} - 7.73z^{-5} - 0.772z^{-6}}{1 - 2.835z^{-1} + 2.922z^{-2} - 1.295z^{-3} + 0.205z^{-4} + 0.0042z^{-5} - 0.0023z^{-6}} \quad (12)$$

The difference equation corresponding to the controller discrete transfer function in equation (12) becomes:

$$u(k) = 2.835 u(k-1) - 2.922 u(k-2) + 1.295 u(k-3) - 0.205 u(k-4) - 0.0042 u(k-5) + 0.0023 u(k-6) + 586.67 e(k) - 1714.79 e(k-1) + 1854.451 e(k-2) - 896.844 e(k-3) + 179.2 e(k-4) - 7.73 e(k-5) - 0.772 e(k-6) \quad (13)$$

The utilization of the above FOPID controller can be realized on the microprocessor PIC16F876 in C language with flow chart of the realized program given in figure (5).

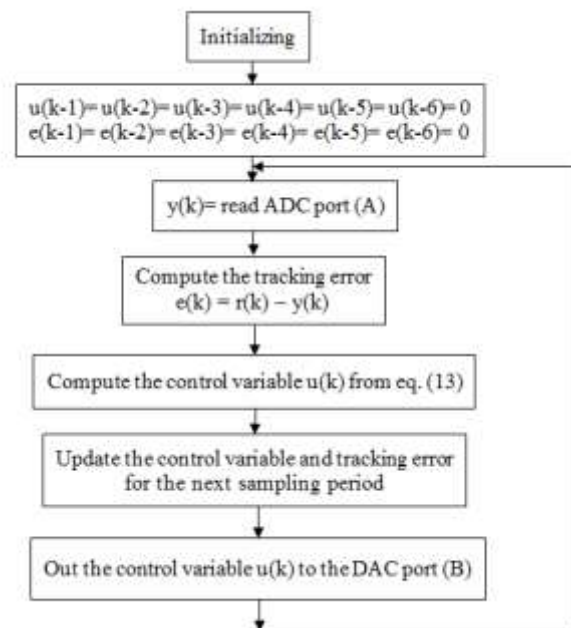


Figure (5): (Flow chart of the realized program)

The block diagram of the mentioned realization is illustrated in figure (6) [7].

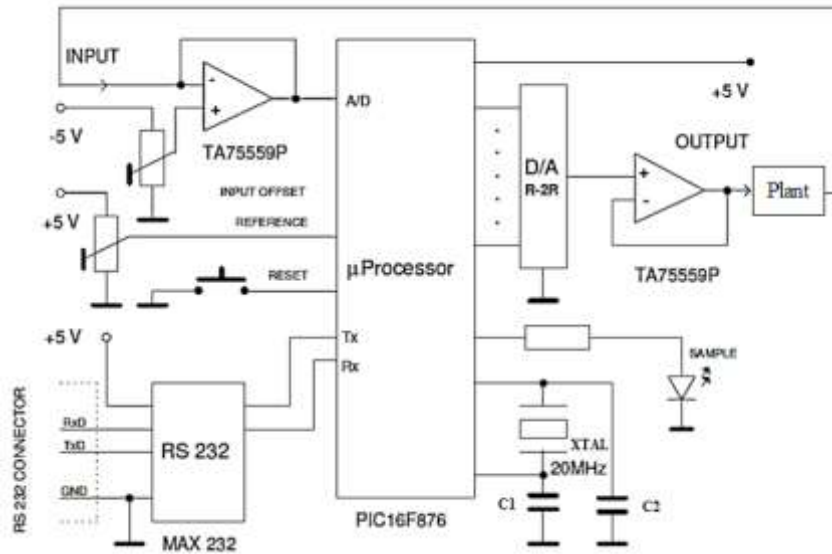


Figure (6): (Block diagram of digital FOC based on PIC)

Which content microprocessor PIC16F876, the basic features of processor PIC16F876: 256 data memory bytes, and 368 bytes of user RAM, an integrated 5-channel 10-bit A/D converter. Two timers. Precision timing interfaces are done through two CCP modules and two PWM modules. PIC16F876 has 22 I/O pins. The operating speed is 20 MHz and power supply voltage is 5 V.

The input to the microprocessor is via A/D converter which is error between measured plant output and reference signal, the plant which is permanent magnet DC motor with gearbox. Initials the built program in personal computer (PC) in C language is loaded to microprocessor by using RS232 port, then the microprocessor ready to operate as controller.

Basic features of realized digital FOC: The range of input signal (control error) is 0 – 5 V and can be adjusted for any sensor output. The output signal range (control value) is 0 – 5 V. The reference voltage range for required value setting is also 0 - 5 V. The discrete step response of the system illustrated in figure (7), the rise time T_r and settling time T_s is 0.008 second and 0.0178 second respectively, which is same as for continuous type in figure (4).

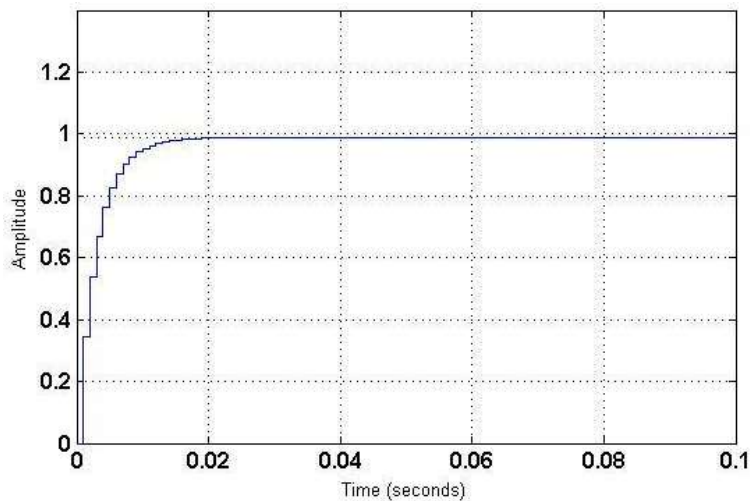


Figure (7): (Discrete step response of negative unity feedback FOPID control system)

7- Conclusion

The FOPID is a special type of conventional controller but with non-integer integration and differentiation order. The controller can be realized in both continuous and discrete time system. The tuning of the parameters have to be chosen to work as satisfactory controller for a given system. The intelligent optimization PSO method has been utilized to tune the FOPID controller for five parameters successfully. The digital FOPID controller can be realized by transform s to z , the obtained transfer function of the controller has been converted to difference equation and coded by PIC software in C language which loaded to PIC memory of microprocessor. The time response for both continuous and discrete time is identical, due to selecting suitable sampling period. The time domain performance for the system exhibits a like response for both continuous and discrete time system. The benefit of FOPID controller it is more robust and wide range tuning with wide band frequency, and improvement in time domain response of the system.

References

- [1] Astrom K. J. and Haggglund T., "*PID controllers: Theory, Design and Tuning*", 2nd Edition / Instrument society of America, 1995, p. 343.
- [2] Concepción A. M., Yang Q. C., Blas M. V., Dingyü X., and Vicente F., "*Fractional-order Systems and Controls: Fundamentals and Applications*", Springer-Verlag / London Limited / London, 2010, p. 414.
- [3] Dingyü X., and Yang Q. C., "*Fractional Order PID Control of a DC-Motor with Elastic Shaft: a Case Study*", IEEE, 1-4244-0210-7/06, 2006, pp. 3182-3187.
- [4] I. Podlubny, "*Fractional-order systems and $PI^{\lambda}D^{\mu}$ controllers*", IEEE Transactions On Automatic Control, Vol. 44, No. 1, 1999, pp. 208–214.
- [5] Vineet S., Pankaj R., and Om P., "*Tuning and Analysis of Fractional Order PID Controller*", International Journal of Electronic and Electrical Engineering. ISSN 0974-2174 Vol. 5, No. 1, 2012, pp. 11-21.
- [6] Dumitru B., José A. T., and Albert C. J., "*Fractional Dynamics and Control*", Springer Science + Business Media / USA, 2012, p. 309.
- [7] Riccardo C., Giovanni D., Luigi F., and Ivo P., "*Fractional Order Systems: Modeling and Control Applications*", Series A Vol. 72, World Scientific Series on Nonlinear Science, Singapore / 2010, p. 178.
- [8] Veysel G., and Kevin M. P., "*Swarm Stability and Optimization*", Springer Science +Business Media / German, 2011, p. 299.
- [9] Duarte P. M., "*Ninteger V. 2.3 Fractional control toolbox for MatLab*", Lisboa: Instituto Superior Técnico da Universidade Técnica de Lisboa, 2005, p. 96. (http://www.mathworks.com/matlabcentral/file_exchange).
- [10] Yang Q. C., K.L. Moore, "*Discretization schemes for fractional order Differentiators and integrators*", IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, Vol. 49, No. 3, 2002, pp. 363–367.
- [11] Mohamad A. Al-Alaoui, "*Novel digital integrator and differentiator*", Electron. Lett., Vol. 29, No. 4, 1993, pp. 376-378.

The work was carried out at the college of Engineering. University of Mosul