International Journal of Control, Energy and Electrical Engineering (CEEE)
Vol.7 pp.30-34

Tuning of Fractional $PI^{\lambda}D^{\mu}A$ Controllers by Using PSO

Khalfa Bettou, Abdelfatah Charef

Département d'Electronique, Université des Frères Mentouri de Constantine, route Ain El-bey, Constantine 25000, Alegria

Bettou_kh,afcharef@yahoo.com

***Abstract*— The paper presents the development of a new tuning method for fractional order $PI^{\lambda}D^{\mu}A$ controller. The basic ideas of this new tuning method are based, in the first place, on the classical tuning methods for setting the parameters of the fractional order $PI^{\lambda}D^{\mu}A$ controller for $\lambda=\mu=1$, which means setting the parameters of the classical PIDA controller, and on the minimum of integral absolute error (IAE) criterion by using particle swarm optimisation (PSO) algorithm for setting the fractional integration action order λ and the fractional differentiation action order μ. It is clearly shown that the fractional order $PI^{\lambda}D^{\mu}A$ controller, which the parameters obtained by the proposed tuning method, gives better response than the classical one for the same system.**

***Keywords*— Integer order PIDA controller, Fractional order $PI^{\lambda}D^{\mu}A$ controller, PSO algorithm, IAE criterion, Induction motor.**

I. INTRODUCTION

The most commonly and practically controller used in all industrial feedback control applications is the proportional-integral-derivative (PID) controller. Many techniques have been suggested for their parameters tuning [1]-[3]. In many control application, the systems are modelled as a third order. PID controllers are unsuitable, especially for third-order systems. This is the reason that a new structure of the controller becomes the necessity of such systems.

In 1996, Jung and Dorf have proposed a new structure of controller and termed as proportional-integral-derivative and acceleration (PIDA) controller [4]. A new analytical approach of the PIDA controller parameters design was proposed by Kitt's [5]. A comparative design and analysis of PIDA controller was presented in [6].

Fractional calculus is a mathematical topic with more than 300 years old history but its application in physics and engineering has been reported only in recent years. In the last decades, besides the theoretical research in the field of fractional integrals and derivatives [7],[8], there are growing numbers of applications of fractional calculus in different areas of control engineering [9]-[11]. The idea of using fractional calculus in feedback control systems dates back to the early sixties. Oustaloup was the one who really introduced a fractional order controller [9]. More recently, Podlubny proposed a generalisation of the PID controller, namely the fractional order $PI^{\lambda}D^{\mu}$ controller [10]. Many researchers have been interested in the use and tuning of this type of controller and more effort is being taken in order to define new effective tuning techniques for fractional order $PI^{\lambda}D^{\mu}$ controllers using classical control theory [12]-[14].

Therefore, a possible way to enhance the performances of a feedback control system with the classical PIDA controllers is to extend the orders of integration and differentiation actions of the classical PIDA controller to real numbers instead of both limited to one. In this paper, we propose the design of the fractional order $PI^{\lambda}D^{\mu}A$ controller of a classical unity feedback control system whose plant's transfer function is considered to be a third order system. The controller is the fractional order $PI^{\lambda}D^{\mu}A$ controller whose transfer function is given as:

$$C_F(s) = k_p + k_i \frac{1}{s^{\lambda}} + k_d s^{\mu} + k_a s^2 \qquad (1)$$

With k_p is the proportional constant, k_i is the integration constant, k_d is the differentiation constant, k_a is the acceleration constant, λ is the fractional integration action order such that $0<\lambda<1$ and μ is the fractional differentiation action order such that $0<\mu<1$.

The proposed tuning method is based, in the first place, on any existed tuning methods for setting the parameters k_p, k_i, k_d and k_a of the fractional order $PI^{\lambda}D^{\mu}A$ controller for $\lambda=\mu=1$ which means setting the parameters of the classical PIDA controller. In this work, we have used Kitti's and Jung-Dorf tuning methods [6]. Then using the parameters k_p, k_i, k_d and k_a obtained in the first step, the error function $e(t)=r(t)-y(t)$, where the input $r(t)$ is the unit step, is minimised through the IAE optimisation criterion to determine the optimum settings of the fractional integration action order λ and the fractional differentiation action order μ of the fractional $PI^{\lambda}D^{\mu}A$ controller. The IAE minimisation criterion is obtained by using PSO algorithm [15]. To use this PSO algorithm, the irrational transfer functions of the fractional order $PI^{\lambda}D^{\mu}A$ controller must be approximated by a rational function, in a given frequency band of practical interest using the singularity function method [16].

The proposed tuning method can also use any already tuned classical PIDA controller by any method for the four parameters k_p, k_i, k_d and k_a and then determine the optimum settings of the fractional order integration λ and the fractional differentiation μ. The optimum tuning of the parameters of the fractional order $PI^{\lambda}D^{\mu}A$ controller is not the main objective of our proposed design method. Instead,

ISSN 2356-5608

our objective is to enhance the control performances of the feedback control system already using a classical PIDA controller by just adding the fractional order integration λ and the fractional order differentiation μ. The paper is organised as follows: Section 2 introduces the approximation of the fractional integrator, differentiator and the fractional order $PI^{\lambda}D^{\mu}A$ controller by a rational function in a limited frequency band of interest. In section 3, we introduce the proposed tuning method for the fractional order $PI^{\lambda}D^{\mu}A$ controller. In section 4, an illustrative example is presented to demonstrate the advantages of the proposed tuning method. Finally, section 5 draws the main conclusions.

II. Approximation of Fractional Order PIDA Controller

When fractional order controllers have to be implemented or simulations using them have to be performed, fractional order transfer functions are usually replaced by integer order transfer functions whose behaviour is close enough to the desired ones. There are many different ways to get such approximations, in our work; the singularity function method of approximation of the fractional order operators by rational transfer function has been used [16].

A. Fractional Order Integrator

In the frequency domain, the fractional order integrator, which is the integration action of the fractional order $PI^{\lambda}D^{\mu}A$ controller, is represented by the following irrational function:

$$C_I(s) = \frac{1}{s^{\lambda}} \tag{2}$$

Where λ is a positive real number such that 0<λ<1.

In a given frequency band of practical interest, the fractional order integrator of (2) is approximated by a rational function as [16]:

$$C_I(s) = \left(\frac{1}{K_{II}} \frac{\prod_{i=0}^{N_I-1}\left(1+\frac{s}{z_{Ii}}\right)}{\prod_{i=0}^{N_I}\left(1+\frac{s}{p_{Ii}}\right)} \right) \tag{3}$$

The gain K_{II}, the poles p_I's and the zeros z_I's are given as in [16].

B. Fractional Order Differentiator

In the frequency domain, the fractional order differentiator, which is the differentiation action of the fractional order PIDA controller, is represented by the following irrational function:

$$C_D(s) = s^{\mu} \tag{4}$$

Where μ is a positive real number such that 0<μ<1.

In a given frequency band of practical interest, the fractional order differentiator of (4) is approximated by a rational function as [16]:

$$C_D(s) = \left(K_{DD} \frac{\prod_{i=0}^{N_D}\left(1+\frac{s}{z_{Di}}\right)}{\prod_{i=0}^{N_D}\left(1+\frac{s}{p_{Di}}\right)} \right) \tag{5}$$

The gain K_{DD}, the poles p_d's and the zeros z_d's are given as in [16].

C. Fractional Order $PI^{\lambda}D^{\mu}A$ Controller

In sections II.A and II.B, we showed how we can approximate the fractional order integrator and differentiator by rational functions, in a given frequency band of practical interest; so (1) becomes:

$$C_F(s) = k_p + k_i.C_I(s) + k_d.C_D(s) + k_a s^2 \tag{6}$$

III. Fractional order PIDA Controller Tuning

In this paper we propose the design of the fractional order $PI^{\lambda}D^{\mu}A$ controller of a classical unity feedback control system shown in Fig.1.

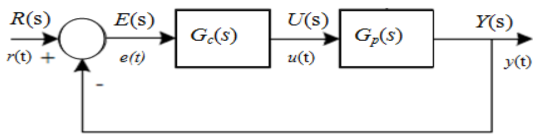

Fig. 1 Classical unity feedback control system

The proposed tuning method is based, in the first place, on any existed classical PIDA controller. In this work, we have used classical PIDA controllers tuned by using Kitti's and Jung-Dorf tuning methods [6]. The error function $e(t)=r(t)-y(t)$, where the input $r(t)$ is the unit step, is minimised through the IAE optimisation criterion to determine the optimum settings of the fractional integration action order λ and the fractional differentiation action order μ of the fractional $PI^{\lambda}D^{\mu}A$ controller. The IAE minimisation criterion is obtained by using PSO algorithm [15].

A. Particle Swarm Optimization (PSO)

Our Particle Swarm Optimization algorithm is an intelligent optimization algorithm intimating the bird swarm behavior which was proposed by psychologist Kennedy and Dr. Eberhart in 1995 [15]. Compared to other optimization algorithms, the Particle Swarm Optimization is more objective, easy and performs well. It is applied in many fields such as the function optimization, the neural network training, the fuzzy system control, etc. In Particle Swarm Optimization algorithm, each individual is called "particle" which represents a potential solution. The algorithm achieves the best solution by the variability of some particles in the tracing space. The particles search in the solution space following the best particle by changing their positions and the fitness frequently; the flying direction and velocity are determined by the objective function.

ISSN 2356-5608

Assuming $X_i = (x_{i1}, x_{i2}, ..., x_{iD})$ is the position of i-th particle in D-dimension, $V_i = (v_{i1}, v_{i2}, ..., v_{iD})$ is its velocity which represents its direction of searching. In iteration process, each particle keeps the best position *pbest* found by itself, besides, it also knows the best position *gbest* searched by the group particles, and changes its velocity according two best positions. The standard formula of Particle Swarm Optimization is as follow:

$$v_{id}^{k+1} = wv_{id}^{k} + c_1 r_1 (p_{id} - x_{id}^{k}) + c_2 r_2 (p_{gd} - x_{id}^{k}) \tag{7}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{8}$$

In which: i=1,2,…N; N the population of the group particles; d=1,2,…,D; k the maximum number of iteration; r_1 and r_2 the random values in [0,1] used to keep the diversity of the group particles; c_1 and c_2 the learning coefficients, also they are called acceleration coefficients; v_{id}^{k} the number d component of the velocity of particle i in k-th iteration; x_{id}^{k} the number d component of the position of particle i in k-th iteration; p_{id} the number d component of the best position particle i has ever found; p_{gd} the number d component of the best position the group particles have ever found; w denotes the inertia weight factor.

The procedure of standard Particle Swarm Optimization is given as following:

Step1: Initialize the original position and velocity of particle swarm;

Step 2: Calculate the fitness value of each particle;

Step 3: For each particle, compare the fitness value with the fitness value of *pbest*, if current value is better, then renew the position with current position, and update the fitness value simultaneously;

Step 4: Determine the best particle of group with the best fitness value, if the fitness value is better than the fitness value of *gbest*, then update the *gbest* and its fitness value with the position;

Step 5: Check the finalizing criterion, if it is satisfied, quit the iteration; otherwise, return to step 2.

B. Tuning of the Parameters K_p, K_i, K_d and K_a

Our tuning strategy is based, in the first place, on Kitti's or Jung-Dorf tuning methods for setting the parameters K_p, K_i, K_d and K_a of the fractional $PI^{\lambda}D^{\mu}A$ controller for λ=μ=1 which means setting the parameters fo a simple classical PIDA controller.

C. Tuning of the parameters λ and μ

With the parameters k_p, k_i, k_d and k_a obtained in the first step, we use the PSO algorithm [15] to determine the optimum settings of the fractional integration action order λ and the fractional differentiation action order μ of the fractional order $PI^{\lambda}D^{\mu}A$ controller. The PSO algorithm consists of finding, for a linear system, a controller minimising the IAE of a classical unity feedback control system for a unit step input. The IAE is given as:

$$Min\ J = Min\left[\int_0^{\infty} |e(t)|\,dt\right] \tag{9}$$

Where $e(t)$=[$r(t)$-$y(t)$] is the error signal.

IV. Illustrative Example

In this section, we will present two simulation examples for the same system by using two tuning methods of classical PIDA controller; this is to show the effectiveness of the proposed design method of the fractional PIDA controller in the performance enhancement of the feedback control system.

A simplified induction motor position control proposed in [6] is used. The transfer function of the induction motor is given as:

$$G(s) = \frac{168.0436}{s(s^2 + 25.921s + 168.0436)} \tag{10}$$

A. Case 1:

In this case, first the parameters λ and μ are set to be λ=μ=1, which means the fractional order $PI^{\lambda}D^{\mu}A$ controller becomes a classical PIDA controller. Then using Jung-Dorf tuning method [6], the parameters k_p, k_i, k_d and k_a are found to be k_p=12.2383, k_i=21.8548, k_d=2.4601 and k_a=0.1268. For this case, the fractional order $PI^{\lambda}D^{\mu}A$ controller's transfer function becomes:

$$C_F(s) = 12.2383 + 21.8548\frac{1}{s^{\lambda}} + 2.4601 s^{\mu} + 0.1268 s^2 \tag{11}$$

To set the parameters λ and μ using our proposed method, $C_F(s)$ is approximated by a rational function using the method proposed in section II.

From the simulation results, the minimum IAE index J(λ,μ) obtained, using PSO algorithm, correspond to the couple (λ,μ)=(0.0750,0.2553). Then the fractional order $PI^{\lambda}D^{\mu}A$ controller's transfer function $C_F(s)$ required is given as:

$$C_F(s) = 12.2383 + 21.8548\frac{1}{s^{0.0750}} + 2.4601 s^{0.2553} + 0.1268 s^2 \tag{12}$$

Fig. 2 shows the step responses of the closed-loop control system with both the classical PIDA and the fractional order $PI^{\lambda}D^{\mu}A$ controllers in its rational form. Apparently, the fractional order $PI^{\lambda}D^{\mu}A$ controller shows a superior performance than the conventional PIDA controller, for the set-point response.

For feedback control performance enhancement comparison, we have summarised some performance characteristics in Table I for the feedback control system with both controllers. It can be noticed that the fractional order $PI^{\lambda}D^{\mu}A$ controller obtained from the proposed tuning method can provide very satisfactory response better than

ISSN 2356-5608

the classical PIDA controller in terms of overshoot (O%), settling time (St) and rise time (Rt).

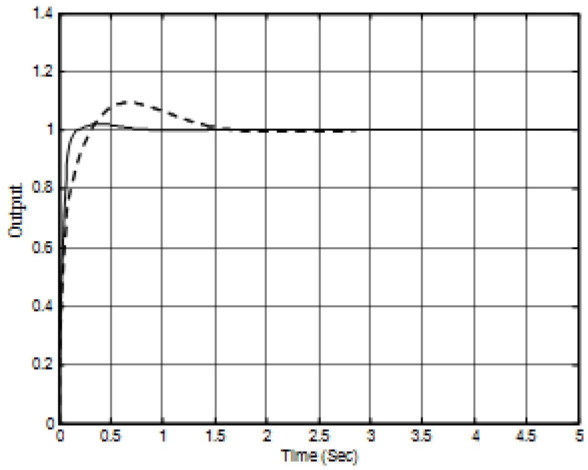

Fig. 2 Step responses of the closed-loop system with a classical PIDA controller (dashed line), and fractional $PI^{\lambda}D^{\mu}A$ controller (solid line)

TABLE I
TEMPORAL CHARACTERISTICS

Controller	**Rt(0.1:0.9)**	**St (2%)**	**O %**	**J**
PIDA	0.181	1.38	9.48	0.0714
$PI^{0.075}D^{0.2553}A$	0.0791	0.486	2.29	0.0114

B. Cas2:

In this case, first the parameters λ and μ are set to be λ=μ=1, which means the fractional order $PI^{\lambda}D^{\mu}A$ controller becomes a classical PIDA controller. Then using Kitt's tuning method [6], the parameters k_p, k_i, k_d and k_a are found to be k_p=5.6672, k_i=9.3764, k_d=0.7027 and k_a=0.0248. For this case, the fractional order PIDA controller's transfer function becomes:

$$C_F(s) = 5.6672 + 9.3764\frac{1}{s^{\lambda}} + 0.7027s^{\mu} + 0.0248s^2 \qquad (13)$$

To set the parameters λ and μ using our proposed method, $C_F(s)$ is approximated by a rational function using the method proposed in section II.

From the simulation results, the minimum IAE index J(λ,μ) obtained, using PSO algorithm, correspond to the couple (λ,μ)=(0.0370,0.5663). Then the fractional order $PI^{\lambda}D^{\mu}A$ controller's transfer function $C_F(s)$ required is given as:

$$C_F(s) = 5.6672 + 9.3764\,\frac{1}{s^{0.037}} + 0.7027\,s^{0.5663} + 0.0248\,s^2 \qquad (14)$$

Fig. 3 shows the step responses of the closed-loop control system with both the classical PIDA controller and the fractional order $PI^{\lambda}D^{\mu}A$ controller in its rational form. The performance improvement of the proposed fractional order $PI^{\lambda}D^{\mu}A$ control structure for set point change is clear.

For feedback control performance enhancement comparison, we have summarised some performance characteristics in Table II for the feedback control system with both controllers. It can be noticed that the fractional order $PI^{\lambda}D^{\mu}A$ controller obtained from the proposed tuning method can provide very satisfactory response better than the classical PIDA controller in terms of smoothest and fastest response.

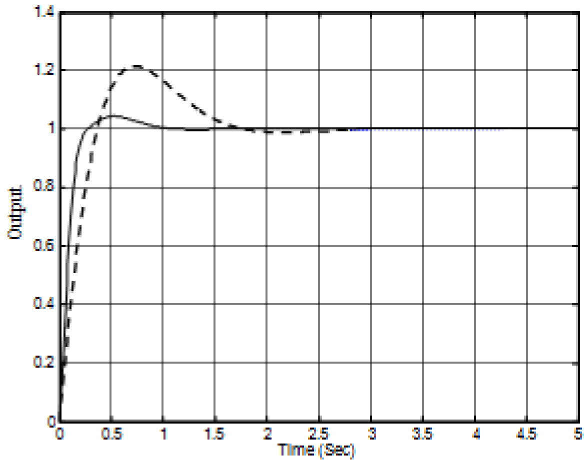

Fig. 3 Step responses of the closed-loop system with a classical PIDA controller (dashed line), and fractional $PI^{\lambda}D^{\mu}A$ controller (solid line)

TABLE III
TEMPORAL CHARACTERISTICS

Controller	**Rt(0.1:0.9)**	**St (2%)**	**O %**	**J**
PIDA	0.278	1.59	21	0.3234
$PI^{0.037}D^{0.5663}A$	0.151	0.805	4.4	0.1107

V. CONCLUSION

In this paper fractional order $PI^{\lambda}D^{\mu}A$ controller have been introduced. The novelty of the proposed controllers consists in the extension of integration and derivation order from integer to fractional numbers. This fact opens the way in the designing of more flexible class of controllers and therefore towards the solution of wider variety of control problems, such as, for example, the control of processes with resonances, integrators and unstable transfer functions.

Our proposed fractional order $PI^{\lambda}D^{\mu}A$ controller is a generalization of the classical PIDA controller. The presented tuning method of the proposed fractional order $PI^{\lambda}D^{\mu}A$ controller is based on the idea of using Jung-Dorf or Kitt's tuning methods and PSO algorithm. The parameters k_p, k_i, k_d and k_a of the fractional order $PI^{\lambda}D^{\mu}A$ controller for λ=μ=1, which means setting the parameters of the classical PIDA controller, have been computed from Jung-Dorf or Kitt's tuning methods and the remaining parameters λ and μ

ISSN 2356-5608

have been found from an optimization problem using PSO algorithm.

Values of the fractional order $PI^{\lambda}D^{\mu}A$ controller parameters are tuned to achieve better step response. The simulation results demonstrated that the fractional order $PI^{\lambda}D^{\mu}A$ controller has better response than the classical PIDA controller.

Our further research efforts include: testing on more type's criterions, experiment on real plants.

REFERENCES

[1] J.G. Ziegler and N.B. Nichols, 'Optimum Settings for Automatic Controllers', Transactions of the ASME, 64,759–768. 1942.

[2] K.J. Astrom and T. Hagglund, *PID Controllers: theory, design and tuning*. Instrument Society of America, Research Triangle Park, North Carolina, U.S.A. 1995.

[3] A. Dwyer, *Handbook of PI and PID controller tuning rules*. Imperial College Press, London. 2003.

[4] S. Jung, and R.C. Dorf, Analytic PIDA controller design technique for a third order system. *Proceedings of the* 35*th IEEE Conference on Decision and Control*, Kobe, pp. 2513-2518. 1996.

[5] P. Ukakimaparn, P. Pannil, P. Boonchuay and T. Trisuwannawat, PIDA controller designed by kitti's method. In: ICCAS-SICE.pp.1547-1550. 2009.

[6] D.K. Sambariya and D. Paliwal, Comparative Design and Analysis of PIDA Controller Using Kitti's and Jung-Dorf Approach for Third Order Practical Systems. *British Journal of Mathematics & Computer Science* Vol.16, Nº 5, pp.1-16. 2016.

[7] K.S. Miller and B. Ross, An Introduction to the Fractional Calculus and Fractional Differential Equations, New York: Wiley. 1993.

[8] K.B. Oldham, and J.Spanier, The Fractional Calculus, New York: Academic Press. 1 974.

[9] A. Oustaloup, La Commande CRONE, Paris: Herme`s (In French). 1991.

[10] I. Podlubny, 'Fractional Order Systems and $PI^{\lambda}D^{\mu}$ Controllers', IEEE Transactions on Automatic Control, 44(1), 208–214. 1999.

[11] I. Petras, and B.M. Vinagre, 'Practical Application of Digital Fractional-order Controller to Temperature Control', Acta Montanistica Slovaca, 7(2), 131–137. 2002.

[12] C.A. Monje, B.M. Vinagre, V. Feliu and Y.Q. Chen, Tuning and Autotuning of Fractional Order Controllers for Industry Applications. *Control Engineering Practice*, 16, pp. 798–812. 2008.

[13] F. Padula, and A. Visioli, Tuning rules for optimal PID and fractional order PID controllers. *Journal of Process Control*. Vol. 21, No.1, pp.69-81. 2011.

[14] A. Neçaibia, and S. Ladaci, Self-tuning fractional order $PI^{\lambda}D^{\mu}$ controller based on extremum seeking approach. *International Journal of Automation and Control* , Vol.8, Nº 2 , pp. 99–121. 2014.

[15] J. Kennedy and R. C. Eberhart, Particle swarm optimization. *Proc. Of IEEE Int. Conf. on Neural Networks*, pp.1942-1948. 1995.

[16] A. Charef, H.H. Sun, Y.Y. Tsao and B.Onaral, Fractal system as represented by singularity function. *IEEE Tran. On Auto. Cont*. Vol. 37, N°9, pp. 1465-1470. 1992.

ISSN 2356-5608