Design of Optimal Fractional Order PID Controller Using PSO Algorithm

Mohammad Reza Dastranj, Mojtdaba Rouhani, and Ahmad Hajipoor

***Abstract*—An intelligent optimization method for designing Fractional Order PID (FOPID) controllers based on Particle Swarm Optimization (PSO) ear presented in this paper. Fractional calculus can provide novel and higher performance extension for FOPID controllers. However, the difficulties of designing FOPID controllers increase, because FOPID controllers append derivative order and integral order in comparison with traditional PID controllers. To design the parameters of FOPID controllers, the enhanced PSO algorithms is adopted, which guarantee the particle position inside the defined search spaces with momentum factor. Experimental results show the proposed design method can design effectively the parameters of FOPID controllers**

***Index Terms*—Fractional order PID, fractional calculus, PID control, PSO Algorithm.**

I. Introduction

Fractional order control systems are described by fractional order differential equations. Fractional calculus allows the derivatives and integrals to be any real number. The FOPID controller is the expansion of the conventional PID controller based on fractional calculus. FOPID controllers' parameters designed have five, and the derivative and integral orders improve the design flexibility.

A. Fractional calculus

There are several definitions of fractional derivatives [1]. Grunwald-Letnikov definition is perhaps the best known one due to its most suitable for the realization of discrete control algorithms. The m order fractional derivative of continues function f(t) is given by:

$$D^m f(t) = \lim_{h\to 0} h^{-m} \sum_{j=0}^{[x]} (-1)^{-j} \binom{m}{j} f(t-jh)$$

$$= \frac{d^m f(t)}{dt^m} \qquad (1)$$

Manuscript received March 10, 2012; revised May 5, 2012.

The authors are with a Young Researchers Club, Gonabad Branch, Islamic Azad University,Gonabad, Iran (e-mail: mohamadrezadastranj@gmail.com).

where$[x]$ is a truncation and $x = \frac{t-m}{h}$; $\binom{m}{j}$ is binomial coefficients,

$$\binom{m}{j} = \frac{m(m-1)\ldots(m-j+1)}{j!} \qquad (2)$$

$\binom{m}{j} = 1, (j = 0)$, it can be replaced by Gamma function,$\binom{m}{j} = \frac{\Gamma(m+1)}{j!\Gamma(m-j+1)}$

The general calculus operator, including fractional order and integer, is defined as:

$$D_t^{\alpha} = \begin{cases} \frac{d^{\alpha}}{dt} & R(\alpha) > 0 \\ 1 & R(\alpha) > 0 \\ \int^{t} (d\tau)^{-\alpha} & R(\alpha) > 0 \end{cases} \qquad (3)$$

0

$$\mathcal{L}\{D^{\alpha} f(t)\} = s^{\alpha} F(s) - [D^{\alpha-1} f(t=0)] \qquad (4)$$

Where $f(s)$ is the Laplace transform of $f(t)$ The Laplace transform of the fractional integral of $f(t)$ is given as follows:

$$\mathcal{L}\{D^{-\alpha} f(t)\} = s^{-\alpha} F(s) \qquad (5)$$

II. Fractional Order Controllers

The differential equation of fractional order controller $PI^{\alpha} D^{\beta}$ is described by [2]:

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^{\delta} e(t). \qquad (6)$$

The continuous transfer function of FOPID is obtained through Laplace transform, which is given by:

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^{\delta} e \qquad (7)$$

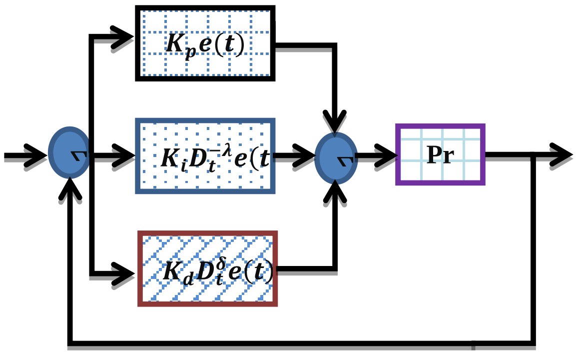


Fig. 1. Generic closed loop control system with a FOPID controller

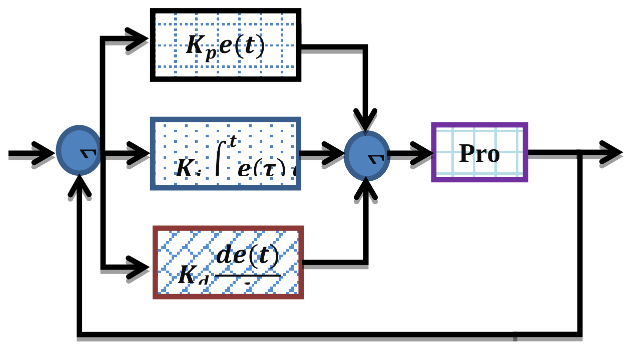


Fig. 2. Generic closed loop control system with a PID controller

$$G_c(s) = K_p + K_i s_t^{-\lambda} + K_d s_t^{\delta} \qquad (7)$$

It is obvious that the FOPID controller not only need design three parameters,Kp Ki and, Kd but also des, λ, δ of integral and derivative controllers. The orders , λ δ are not necessarily integer, but any real numbers. As shown in Fig. 3 the FOPID controller generalizes the conventional integer order PID controller and expands it from point to plane. This expansion could provide much more flexibility in PID control design. [3]

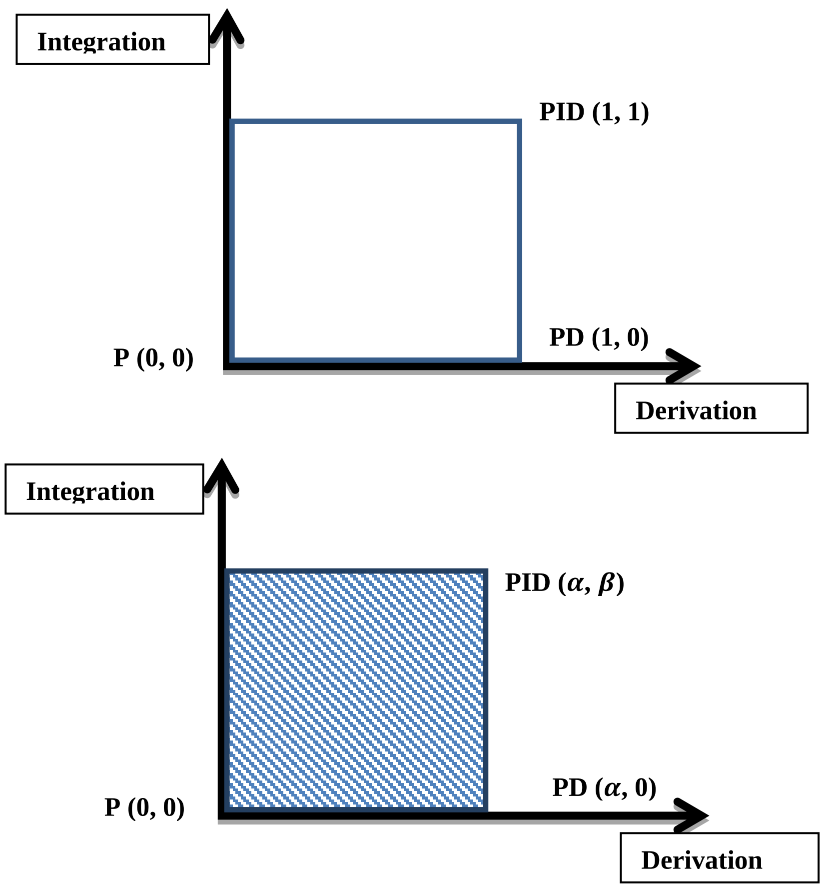


Fig. 3. PID controllers with fractional orders

Particle swarm optimization, first developed by Kennedy and Eberhart [4], is one of the modern heuristic algorithms. It was inspired by the social behavior of bird and fish schooling, and has been found to be robust in solving continuous nonlinear optimization problems.

This algorithm is based on the following scenario: a group of birds are randomly searching food in an area and there is only one piece of food. All birds are unaware where the food is, but they do know how far the food is at each time instant. The best and most effective strategy to find the food would be to follow the bird which is nearest to the food. Based on such scenario, the PSO algorithm is used to solve the optimization problem.

In PSO, each single solution is a "bird" in the search space; this is referred to as a "particle". The swarm is modelled as particles in a multidimensional space, which have positions and velocities. These particles have two essential capabilities: their memory of their own best position and knowledge of the global best. Members of a swarm communicate good positions to each other and adjust their own position and velocity based on good positions according to (8).

$$v(k+1)_{i,j} = w.v(k)_{i,j} + c_1 r_1 \left(gbest - x(k)_{i,j}\right) + c_2 r_2 \left(pbest_j - x(k)_{i,j}\right) \qquad (8)$$

$$x(k+1)_{i,j} = x(k)_{i,j} + v(k)_{i,j}$$

where

$V_{i,j}$ Velocity of particle i and dimension j

$X_{i,j}$ Position of particle i and dimension j

c_1, c_2 Acceleration constants

w Inertia weight factor

r_1, r_2 Random numbers between 0 and 1

pbest Best position of a specific particle

gbest Best particle of the group

1. Initialize a group of particles including the random positions, velocities and accelerations of particles.
2. Evaluate the fitness of each particle.
3. Compare the individual fitness of each particle to its previous pbest. If the fitness is better, update the fitness as

pbest.

4. Compare the individual fitness of each particle to its previous gbest. If the fitness is better, update the fitness as gbest.

5. Update velocity and position of each particle according to (8).

6. Go back to step 2 of the process and keep repeating until some stopping condition is met.

Although PSO has been relatively recently developed, there already exist several applications based on the PSO algorithm. One application of PSO algorithms is the optimized particle filter. A particle filter aims to estimate a sequence of hidden parameters based only on the observed data. This can help for example in removing noise and by this improving the measured data. The PSO can be merged into the particle filter for optimization.

The basic particle filter is suboptimal in the sampling step, by applying PSO to particle filter, particle impoverishment problem is avoided and estimation accuracy is improved.

The PSO algorithms can also be applied in the job shop scheduling [5]. Job shop scheduling is an optimization problem in which ideal jobs are assigned to resources at particular times. In this problem, several jobs of varying size are given, which need to be scheduled on different identical machines, while trying to minimize the total length of the schedule. By converting the job shop scheduling problem to PSO we can search the optimal solution for job shop scheduling. Through using PSO for the job shop problem we can find a scheduling solution at least as good as the best known solution. Moreover PSO generally results in a better solution than GA while its implementation and manipulation is easier.

Fig 4 illustrates the block structure of the FOPID controller optimizing process with PSO. All parameters of the FOPID controllers are updated at every final time (t f). It should be noted that the values of an element in the particle may exceed its reasonable range. In this case, inspired from practical requirements and from the papers focusing on tuning the parameters of FOPID in application of the different systems, the lower bound of the FOPID parameters are zero and their upper bounds are set.

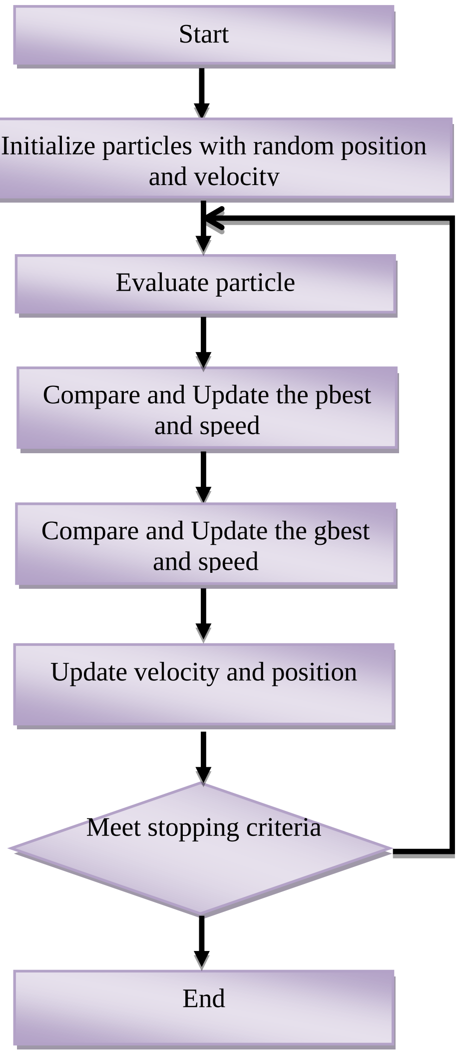


Fig. 4. The PSO algorithm procedure

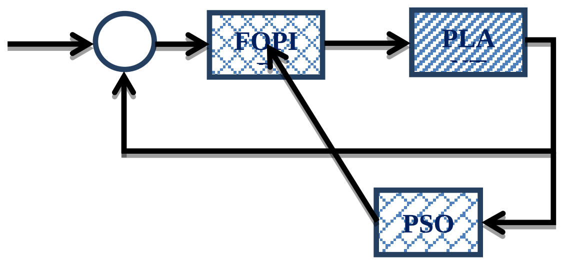
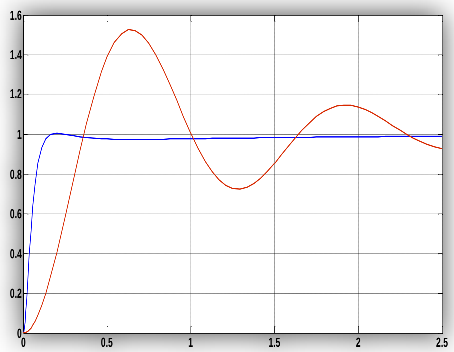


Fig. 5. Tuning process of the FOPID Controller parameters with PSO

III. Simulation Researches

The control objective has the transfer function $\mathbf{G(s)} = \frac{\mathbf{25}}{\mathbf{s^2+2s}}$



IV. Conclusion

It has been demonstrated that the parameters optimization of fractional order controller based on modified PSO is highly effective. According to optimization target, the proposed method can search the best global solution for FOPID controllers' parameters and guarantee the objective solution space in defined search space. Based on improved PSO, the design and application of FOPID will be appeared in various fields.

References

[1] A. Carpinteri and F. Mainardi, *Fractals and Fractional Calculus in Continuum Mechanics,Springer-Verlag*, Wien and New York, pp. 223-276, 1997.

[2] I. Podlubny, “Fractional-order systems and PIλ Dδ controllers,” *IEEE Trans. on Automatic Control*, vol. 44, no. 1, pp. 208-213, January1999.
[3] J.-Y. Cao and B.-G. Cao, “Design of Fractional Order Controller Based on Particle Swarm optimization,” *International Journal of Control, Automation, and Systems*, vol. 4, no. 6, pp. 775-781, December 2006
[4] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proc. IEEE int. conf. Neural Networks*, vol. IV, Perth, Austrailia, 1995, pp.1942-1948.
[5] Z. Liu, “Investigation of Particle Swarm Optimization for Job Shop Scheduling Problem,” in *Proc. IEEE 3rd Int. Conf. on Natural Computation*, Haikou, China, Dec. 2007, pp.799-803

Mohamad Reza Dastranj received the MS degree in control engineering from the control department, Islamic Azad University of Gonabad,Iran, in 2009 and 2011, respectively. His research interests focus on fuzzy systems and Genetic Algorithm.