

Research Article

Stabilization and Tracking Control of Inverted Pendulum Using Fractional Order PID Controllers

Sunil Kumar Mishra and Dinesh Chandra

Department of Electrical Engineering, Motilal Nehru National Institute of Technology, Allahabad 211004, India

Correspondence should be addressed to Sunil Kumar Mishra; hariomsunil88@gmail.com

Received 7 February 2014; Revised 5 March 2014; Accepted 2 April 2014; Published 23 April 2014

Academic Editor: Haranath Kar

Copyright © 2014 S. K. Mishra and D. Chandra. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This work focuses on the use of fractional calculus to design robust fractional-order PID ($PI^\lambda D^\mu$) controller for stabilization and tracking control of inverted pendulum (IP) system. A particle swarm optimisation (PSO) based direct tuning technique is used to design two $PI^\lambda D^\mu$ controllers for IP system without linearizing the actual nonlinear model. The fitness function is minimized by running the SIMULINK model of IP system according to the PSO program in MATLAB. The performance of proposed $PI^\lambda D^\mu$ controllers is compared with two PID controllers. Simulation results are also obtained by adding disturbances to the model to show the robustness of the proposed controllers.

1. Introduction

The inverted pendulum (IP) system, nonlinear and unstable system, is widely used in laboratories to implement and validate new ideas emerging in control engineering. The control of IP system can be broadly divided into three sections, swing-up control, stabilization, and tracking control. Swing-up control is basically used to swing the pendulum rod from pending position to stabilization zone. Then a balancing or stabilization control is essential to uphold it in upright position for long interval. A switching mechanism between swinging and stabilization zone is necessary for effective control [1, 2]. For swing-up control, a technique based on energy control had been proposed by Åström and Furuta [3].

There are several different techniques accessible in literature for stabilization and tracking control of IP system, for example, linear quadratic regulator (LQR), PID control, neural network control, fuzzy logic control, neural-fuzzy control, sliding mode control, and so forth. The LQR, an optimal state feedback controller designed by minimizing a performance index, is ordinarily used controller for IP system modelled in state space form [4]. Here, the state space model of IP system is to be inevitably linearized which leads to modelling error. The PID controller, most widely used controller in several industrial control problems, is one of the favourite controllers

for IP system. The comparison of PID controller with other control techniques of IP system was carried out in many studies [5, 6]. The major task of PID controller design is the selection of control parameters for desired response. Some tuning methods of PID controller for IP system could be found in literature [7–9]. In [10], the stabilization as well as tracking control of IP system with actual nonlinear model using PID controllers was investigated but how to choose controllers parameters was not clarified. Various techniques other than PID are also existing [11–15].

In last two decades, the fractional calculus has become much popular among the researchers of different streams but its origin is as older as that of classical integer order calculus. Fractional calculus was not much popular earlier because of its highly complex mathematical expressions. But with the development of computational technologies it has become possible to deal with fractional calculus. Fractional calculus provides much accurate and generalized solution as compared to integer order calculus. The applications of fractional calculus include modelling and control of physical systems [16–18]. One such application is the modelling of two-electric pendulum [19].

In the area of control engineering, an application of fractional calculus is the fractional order PID ($PI^\lambda D^\mu$) controller which is an advanced form of PID control. In some recent

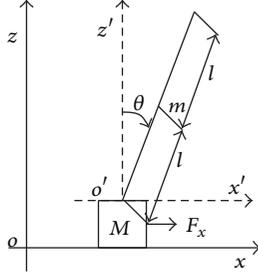


FIGURE 1: Structure of inverted pendulum.

studies [20–24], the $PI^\lambda D^\mu$ controller gives better outcomes than PID controller. Though there are some applications of $PI^\lambda D^\mu$ controller for IP system [25, 26], $PI^\lambda D^\mu$ controller has not received considerable attention for unstable systems similar to IP system.

Hence, in the present work, fractional order PID controller is designed in time domain to control pendulum angle as well as cart position. Fractional order PID controller is challenging to design because of the use of fractional calculus with very complex calculations. Therefore, a direct approach is used for calculating the parameters of both fractional order PID controllers with the help of a multiobjective fitness function (the fitness function consists of the sum of integral squares of pendulum angle, cart position, and control voltage). The fitness function is minimized by running the model according to a particle swarm optimization (PSO) [27–29] program in MATLAB. PSO is used in this work as it provides greater convergence towards optimal values as compared to other optimization techniques and it has the simple algorithm.

The rest of the paper is divided into the following sections. Section 2 gives a description of the inverted pendulum system and derivation of system equations in state space form. Section 3 describes the fractional calculus and structure of the $PI^\lambda D^\mu$ controller. Section 4 gives details about PSO. Section 5 gives a complete description of control strategy. Section 6 gives a comparison of the simulation results for $PI^\lambda D^\mu$ controller and PID controller with and without disturbances. The paper ends with the conclusions in Section 7, which is followed by the references.

2. Inverted Pendulum System

As shown in Figure 1, an IP system has a cart which can move horizontally. One end of the pendulum rod is connected to the centre of the upper surface of the cart which is called the pivot point while the other end is free to move in vertical plane (xz -plane). This pendulum rod is stable in extreme downwards position and known as normal pendulum. But when the pendulum rod remains in upright position, it is known as IP system. This is an unstable condition which needs a continuous balancing force (F_x) on cart in order to remain upright.

In Figure 1 horizontal force is used as control action to displace the pivot around x axis and the total kinetic energy

TABLE 1: Inverted pendulum parameters.

M (Kg)	m (Kg)	l (m)	g (m/s ²)
1	0.1	0.3	9.8

(K) due to mass of the pivot (M) in x direction, mass of pendulum rod (m) in x and z directions, and potential energy (P) of the IP system [10] which are

$$K = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}_p^2 + \dot{z}_p^2), \quad P = mgz_p, \quad (1)$$

where

$$x_p = x + l \sin \theta, \quad z_p = z + l \cos \theta \quad (2)$$

l = the distance from the pivot to the mass centre of the pendulum, (x, z) = the position of the pivot in the xoz coordinate, (\dot{x}, \dot{z}) = the speed in the xoz coordinate, (x_p, z_p) = the position in the $x'o'z'$ coordinate, (\dot{x}_p, \dot{z}_p) = the speed in the $x'o'z'$ coordinate, and g = the acceleration constant due to gravity. It is assumed that the inertia of the pendulum is negligible. Numerical values of all the parameters of IP system are given in Table 1.

The Lagrange's equations of the IP system are given as follows:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x} = F_x, \quad (3a)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = 0, \quad (3b)$$

where $L = K - P$. By putting the expression of L in (3a) and (3b) and after solving it, Lagrange's equations of the IP system can be expressed as

$$(M + m) \ddot{x} + ml \cos \theta \ddot{\theta} - ml \sin \theta \dot{\theta}^2 = F_x, \quad (4a)$$

$$\cos \theta \ddot{x} + l \ddot{\theta} - g \sin \theta = 0, \quad (4b)$$

where

$$-0.5 \leq x \leq 0.5. \quad (5)$$

According to (4a) and (4b), the state equations of the IP system can be expressed as

$$\dot{x}_1 = x_2, \quad (6a)$$

$$\dot{x}_2 = \frac{-mg \cos x_3 \sin x_3 + ml \sin x_3 x_4^2 + F_x}{M + m \sin^2 x_3} + d_1, \quad (6b)$$

$$\dot{x}_3 = x_4, \quad (6c)$$

$$\dot{x}_4 = \frac{-ml \cos x_3 \sin x_3 x_4^2 - \cos x_3 F_x}{Ml + ml \sin^2 x_3} + \frac{(M + m) g \sin x_3}{Ml + ml \sin^2 x_3} + d_2, \quad (6d)$$

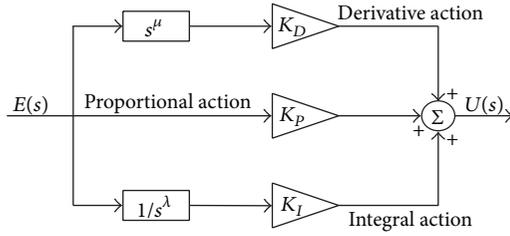


FIGURE 2: FOPID controller structure.

where $x_1 = x$, $x_2 = \dot{x}$, $x_3 = \theta$, $x_4 = \dot{\theta}$, and d_1, d_2 are the external disturbances.

As it is known that IP system is a highly nonlinear and unstable system. Therefore, accurate modelling of a system having nonlinear dynamics cannot be obtained using standard linearization techniques. Hence, in this paper (6a), (6b), (6c), and (6d) are considered as it is without using any linearization technique.

3. Fractional Calculus and Fractional Order PID Controller

3.1. Fractional Calculus. Fractional calculus [16–18] is a branch of mathematics which deals with integration and differentiation operators that have fractional number powers. Though these types of operators are complex in nature as compared to integer order operators, they provide a generalization which also includes integer order operators.

The three most frequently used definitions are Riemann-Liouville, Grünwald-Letnikov, and Caputo [22]. The most common definition is known as Riemann-Liouville:

$${}_a D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau) d\tau}{(t-\tau)^{\alpha-n+1}}. \quad (7)$$

The second one is the Grünwald-Letnikov given as

$${}_a D_t^\alpha f(t) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{j=0}^{(t-a)/h} (-1)^j \binom{\alpha}{j} f(t-jh), \quad (8)$$

where $\binom{\alpha}{j} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)}$.

Finally, Caputo expression is defined as

$${}_a D_t^\alpha f(t) = \frac{1}{\Gamma(\alpha-n)} \int_a^t \frac{f^n(\tau) d\tau}{(t-\tau)^{\alpha-n+1}}, \quad (9)$$

where, $n-1 < \alpha < n$, n is an integer number, a is initial condition.

3.2. Fractional Order PID Controller. PID controller [30, 31] is one of the most extensively used controllers, but in the last two decades the advancement in fractional calculus has introduced the fractional order PID controller in control applications. $PI^\lambda D^\mu$ controller is a generalized form of PID controller. The $PI^\lambda D^\mu$ controller structure is shown in Figure 2, in which the introduction of two extra parameters λ and μ

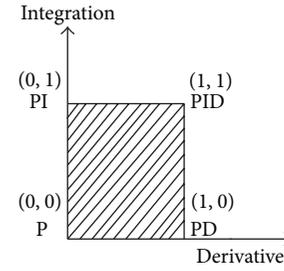


FIGURE 3: Generalization of the PID controller.

makes it complex as compared to PID controller because of introduction of fractional calculus in it.

The differential equation of the $PI^\lambda D^\mu$ controller [23] is described as

$$u(t) = K_P e(t) + K_I D^{-\lambda} e(t) + K_D D^\mu e(t). \quad (10)$$

The generalization of PID controller is shown in Figure 3, which can be obtained using different values of λ and μ in (10). PID ($\lambda = 1, \mu = 1$), PI ($\lambda = 1, \mu = 0$), PD ($\lambda = 0, \mu = 1$), or P ($\lambda = 0, \mu = 0$) are the special cases of $PI^\lambda D^\mu$ controller.

Taking Laplace transform of (10), the controller expression in s -domain is obtained as

$$C(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s^\lambda} + K_D s^\mu. \quad (11)$$

By putting $\lambda = \mu = 1$ in (11), the expression of PID controller can be written as

$$C(s) = \frac{U(s)}{E(s)} = K_P + \frac{K_I}{s} + K_D s. \quad (12)$$

Hence, in the present work PID controller is also studied along with the study of $PI^\lambda D^\mu$ controller. In addition, the performance comparisons with the PID controller based on the same design specifications to show that the $PI^\lambda D^\mu$ controller has better performance in terms of performances index are carried out.

In (11), s^λ and s^μ have fractional orders which are not directly compatible with MATLAB and it becomes difficult to realize hardware of $PI^\lambda D^\mu$ controller. Therefore, there are several integer order approximation methods available for fractional order elements [32–34]. The 5th order Oustaloup's integer order approximation [32] in the frequency range ($10^{-2}, 10^2$) rad/s is used in this work. In MATLAB fractional order PID controller is implemented using FOMCOM toolbox [35] where Oustaloup's approximation is realized.

4. Particle Swarm Optimization

While solving complex optimization problems having large search space the population based swarm intelligence method is the widely accepted alternatives to find the optimal solution. The particle swarm optimization (PSO) was first proposed by Kennedy and Eberhart [27]. The PSO method [27–29] is population based search techniques used for

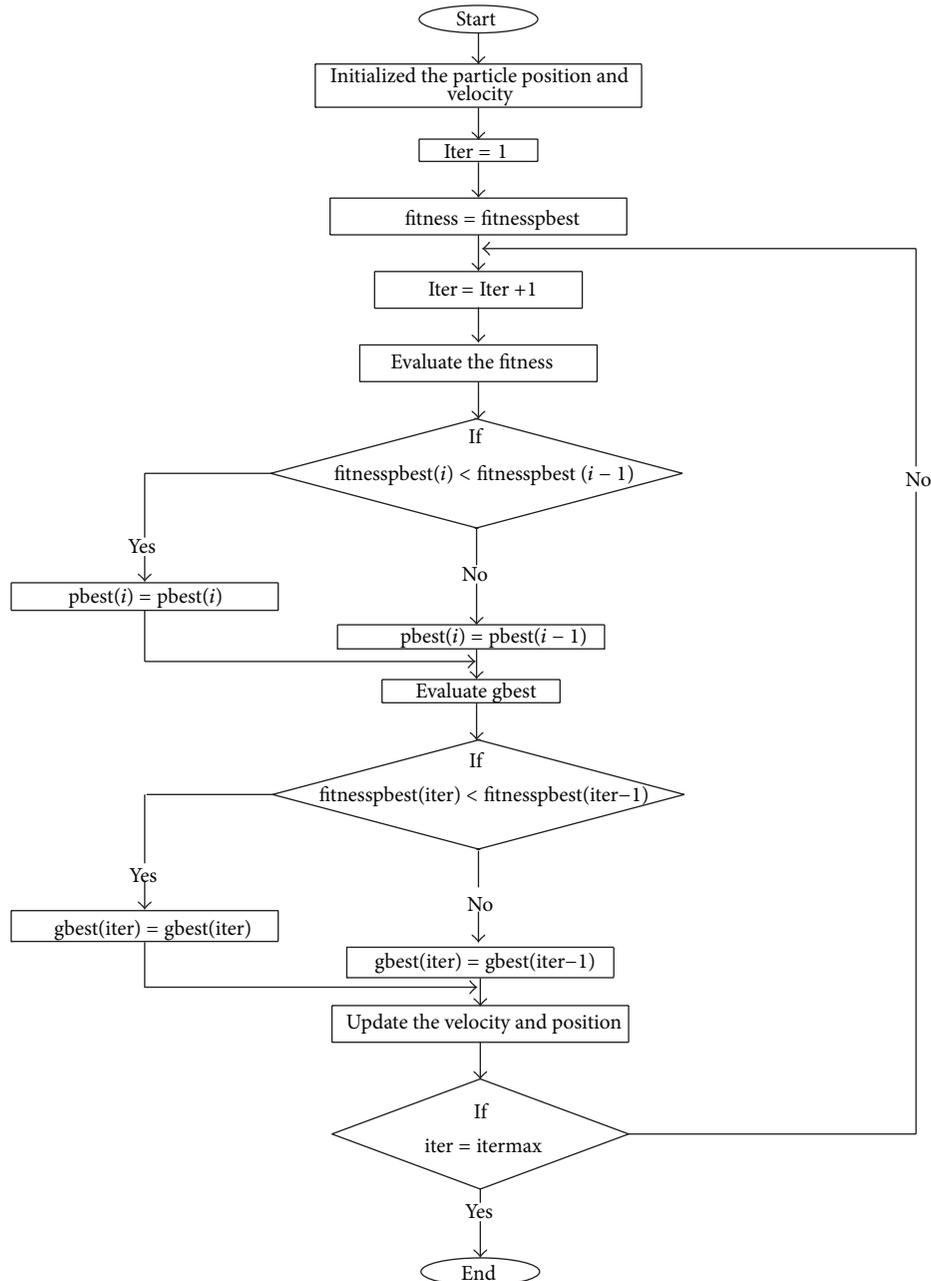


FIGURE 4: Flow diagram of basic PSO algorithm.

solving the optimization problems having a large search space. This technique mimics the behaviour of bird flocks and fish schools and their collision-free and synchronized moves. Individual bird or fish is known as particle in a PSO system and each particle has its position and velocity. Now particle moves in multidimensional search space according to its own experience and the experience of the neighbouring particles. During the movement in search space, the position and velocity of the particles are updated. There are three factors inertia, cognitive, and social upon which the velocity and position update of particles depend. The complete procedure

of PSO algorithm can be understood with the help of Figure 4 which indicates that the PSO algorithm has three steps:

- (1) to evaluate the fitness value of each particle,
- (2) to update individual best positions (pbest) and global best positions (gbest) according to best or minimum fitness values,
- (3) to update velocity and position of each particle in each iteration.

The above steps are repeated until some stopping criteria are met.

TABLE 5: Values of controller parameters.

Controller	K_P	K_I	K_D	λ	μ	J
$PI^\lambda D^\mu_1$	22.4734	1.8185	3.1337	0.3579	0.9038	1.7131
$PI^\lambda D^\mu_2$	-1.6749	-0.7636	-1.8977	0.4764	0.8908	
PID_1	23.9288	1.2548	3.2143	—	—	1.8449
PID_2	-2.6622	-1.3422	-2.7675	—	—	

to $PI^\lambda D^\mu_1$ and $PI^\lambda D^\mu_2$. The output of each $PI^\lambda D^\mu$ is added and given as control input to the IP system.

The fitness function J to be minimized using PSO is given as

$$J = \int_0^T [w_1 \times \theta^2(t) + w_2 \times x^2(t) + w_3 \times u^2(t)] dt, \quad (16)$$

where $\theta(t)$, $x(t)$, and $u(t)$ are pendulum angle, cart position, and control input, respectively, and w_1 , w_2 , and w_3 are the weights to give equal weightage to all parameters. T is the simulation time used for running model in SIMULINK. The fitness function given by (16) has three terms. The first term is the integral square of pendulum angle $\theta(t)$ which is used to stabilize pendulum angle. The second term is the integral square of cart position $x(t)$ which is used to stabilize cart position. Finally, the third term is the integral square of control input $u(t)$ which is used to minimize the required control force.

Basic block diagram of closed loop control system as shown in Figure 5 is prepared in MATLAB/SIMULINK. This model has ten unknown parameters of two $PI^\lambda D^\mu$ controllers. These parameters are supplied by PSO program. Initially, parameters $\{K_{P1}, K_{I1}, K_{D1}, \lambda_1, \mu_1\}$ and $\{K_{P2}, K_{I2}, K_{D2}, \lambda_2, \mu_2\}$ are generated randomly but later in terms of pbest and gbest by updating the velocity and position of particles in each iteration. After generating controller parameters SIMULINK model is executed according to PSO program. When this model is executed, the fitness value (as given in (16)) of the SIMULINK model is saved in MATLAB *workspace* which is further utilized by PSO program for evaluating the minimum fitness value and corresponding controller parameters. The whole process is repeated until maximum number of iteration is reached. At the end of the process the values of parameters $\{K_{P1}, K_{I1}, K_{D1}, \lambda_1, \mu_1\}$ and $\{K_{P2}, K_{I2}, K_{D2}, \lambda_2, \mu_2\}$ are obtained which provides the desired performance of the IP system. Also, by considering $\lambda = \mu = 1$ in both the $PI^\lambda D^\mu$ controllers, integer order PID controllers have been designed using the same specifications and comparative study has been carried out to show the validity of the proposed work.

6. Simulation and Results

Parameters of (16) are given in Table 2. As $\theta(t)$ and $x(t)$ lie in the ranges $\{-0.5, 0.5\}$ rad and $\{-0.5, 0.5\}$ m, respectively, and are given equal weight but $u(t)$ is given very less weightage because of its high range $\{-20, 20\}$ N. Therefore, all three terms of (16) are minimized equally.

With the help of PSO parameters and controller parameters given in Tables 3 and 4, the SIMULINK model shown

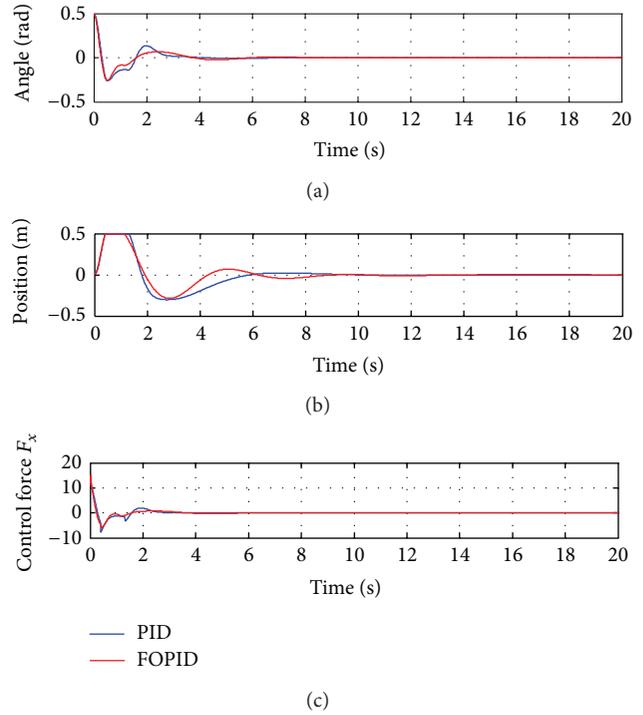


FIGURE 6: Stabilization of inverted pendulum (without disturbances $d_1 = d_2 = 0$).

in Figure 5 is executed by PSO program to obtain final parameters for both $PI^\lambda D^\mu$ and PID cases as given in Table 5.

From Table 5 it can be concluded that based on the same specifications as given in Tables 2, 3, and 4, the fitness value J using $PI^\lambda D^\mu$ controller is less as compared to PID controller. Controller parameters are calculated for stabilization control (without disturbances) but these values are also applicable in other cases. Simulation results, as shown in Figures 6–9, are obtained for stabilization and tracking control of IP system with and without disturbances. All the simulation results are particularized in next two subsections titled as stabilization and tracking control of IP system.

6.1. Stabilization of Inverted Pendulum. For stabilization, reference cart position $r(t) = 0$. In Figure 6, waveforms for $\theta(t)$, $x(t)$, and $u(t)$ settle to steady state approximately at the same time for both $PI^\lambda D^\mu$ controller and PID controller but during transient period $PI^\lambda D^\mu$ performs better than PID controller. Now to check the robustness of the designed

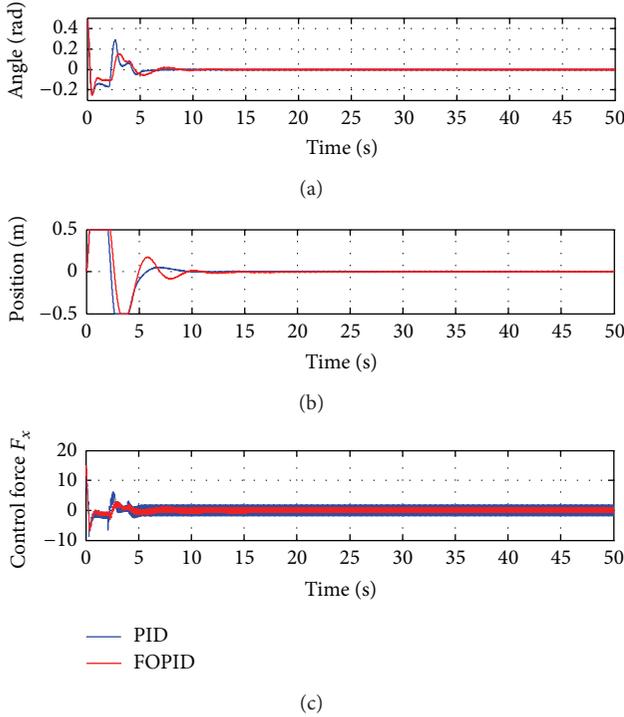


FIGURE 7: Stabilization of inverted pendulum (with disturbances $d_1 = d_2 = 20 \sin(20\pi t)$).

$PI^\lambda D^\mu$ and PID controllers, disturbances are $d_1 = d_2 = 20 \sin(20\pi t)$.

Figure 7 shows the simulation results with disturbances which are still valid and proves the effectiveness of proposed $PI^\lambda D^\mu$ and PID controllers. The waveforms for $\theta(t)$ and $u(t)$ in Figure 7 in case of $PI^\lambda D^\mu$ are better than PID controller but, for $x(t)$ PID, perform slightly better than $PI^\lambda D^\mu$ controller. Also, in the case of control input $u(t)$ $PI^\lambda D^\mu$ provides less deviation during steady state as compared to PID controller.

6.2. Tracking Control of Inverted Pendulum. For tracking control, reference cart position $r(t)$ is considered as $0.3\sin(0.05\pi t)$. Figure 8 shows the tracking control of IP system in which $PI^\lambda D^\mu$ controller for $\theta(t)$ provides less deviation in transient period and settles earlier to steady state as compared to PID controller. In Figure 8, both $PI^\lambda D^\mu$ and PID provide good tracking but in case of $PI^\lambda D^\mu$, less control effort is required. In Figure 9, in the presence of disturbances, the $PI^\lambda D^\mu$ controller still outperforms PID controller and in $PI^\lambda D^\mu$ case less control effort is required.

Simulation results shown in Figures 6–9 are furthermore important from real-time implementation viewpoint as in this simulation study practical conditions have also been taken into consideration.

7. Conclusion

The stabilization and tracking control of IP system are attained successfully using PSO based direct tuning method.

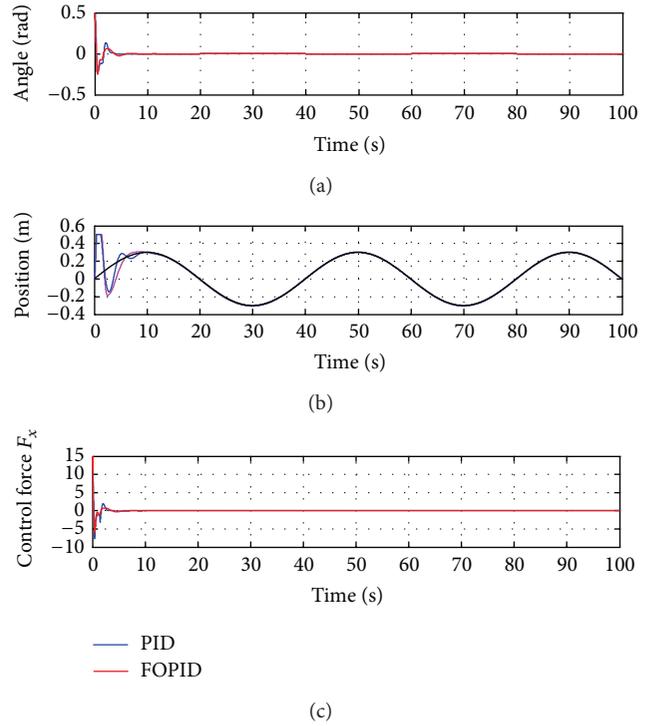


FIGURE 8: Tracking of inverted pendulum (without disturbances $d_1 = d_2 = 0$).

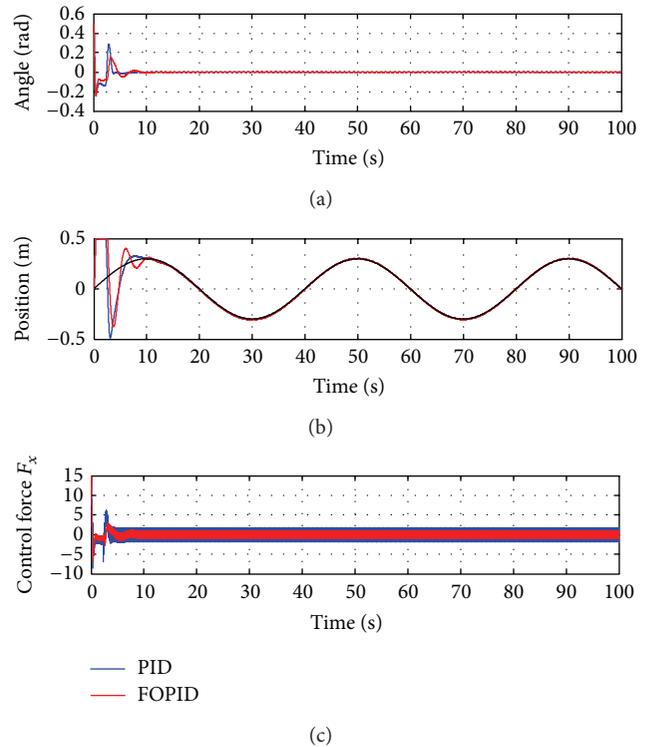


FIGURE 9: Tracking of inverted pendulum (with disturbances $d_1 = d_2 = 20 \sin(20\pi t)$).

The use of PSO technique for calculating controller parameters is very simple and provides good convergence towards optimal values. Two integer order PID controllers have also been designed by keeping the same specifications. A comparative study has been carried out and the obtained results are quite acceptable for both $PI^{\lambda}D^{\mu}$ and PID controllers but the $PI^{\lambda}D^{\mu}$ controller seems to be more robust. The $PI^{\lambda}D^{\mu}$ could be the good replacement for PID in the forthcoming years. The real time implementation of $PI^{\lambda}D^{\mu}$ controller might be the subject of further research.

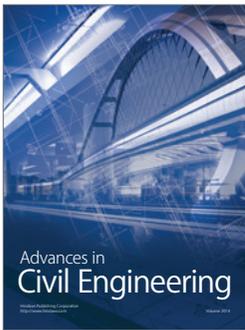
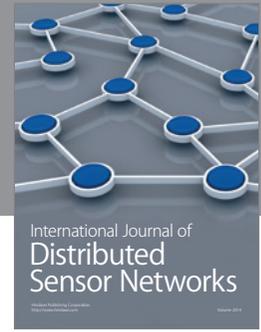
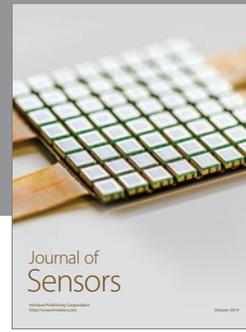
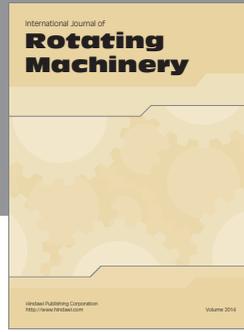
Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] N. Muškinja and B. Tovornik, "Swinging up and stabilization of a real inverted pendulum," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 2, pp. 631–639, 2006.
- [2] T. Agustinah, A. Jazidie, and M. Nuh, "Hybrid fuzzy control for swinging up and stabilizing of the pendulum-cart system," in *Proceedings of the IEEE International Conference on Computer Science and Automation Engineering (CSAE '11)*, vol. 4, pp. 109–113, June 2011.
- [3] K. J. Åström and K. Furuta, "Swinging up a pendulum by energy control," *Automatica*, vol. 36, no. 2, pp. 287–295, 2000.
- [4] A. Ghosh, T. R. Krishnan, and B. Subudhi, "Robust proportional-integral-derivative compensation of an inverted cart-pendulum system: an experimental study," *IET Control Theory & Applications*, vol. 6, no. 8, pp. 1145–1152, 2012.
- [5] C.-E. Huang, D.-H. Li, and Y. Su, "Simulation and robustness studies on an inverted pendulum," in *Proceedings of the 30th Chinese Control Conference (CCC '11)*, pp. 615–619, July 2011.
- [6] L. B. Prasad, B. Tyagi, and H. O. Gupta, "Modelling & simulation for optimal control of nonlinear inverted pendulum dynamical system using PID controller & LQR," in *Proceedings of the 6th Asia Modelling Symposium (AMS '12)*, pp. 138–143, 2012.
- [7] M. R. Dastranj, M. Moghaddas, S. S. Afghu, and M. Rouhani, "PID control of inverted pendulum using particle swarm optimization (PSO) algorithm," in *Proceedings of the 3rd IEEE International Conference on Communication Software and Networks (ICCSN '11)*, pp. 575–578, May 2011.
- [8] H. Lee, J. Lee, and J. Lee, "Hill climbing algorithm of an inverted pendulum," in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '09)*, pp. 574–579, December 2009.
- [9] S. Li, C. Huo, and Y. Liu, "Inverted pendulum system control by using modified PID neural network," in *Proceedings of the 3rd International Conference on Innovative Computing Information and Control (ICICIC '08)*, pp. 1–426, June 2008.
- [10] J.-J. Wang, "Simulation studies of inverted pendulum based on PID controllers," *Simulation Modelling Practice and Theory*, vol. 19, no. 1, pp. 440–449, 2011.
- [11] S. Jung, H.-T. Cho, and T. C. Hsia, "Neural network control for position tracking of a two-axis Inverted pendulum system: experimental studies," *IEEE Transactions on Neural Networks*, vol. 18, no. 4, pp. 1042–1048, 2007.
- [12] S. Omatu and S. Deris, "Stabilization of inverted pendulum by the genetic algorithm," in *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation (ETFA '96)*, vol. 1, pp. 282–287, November 1996.
- [13] X.-H. Yang, H.-S. Liu, G.-P. Liu, and G.-F. Xiao, "Control experiment of the inverted pendulum using adaptive neural-fuzzy controller," in *Proceedings of the International Conference on Electrical and Control Engineering (ICECE '10)*, pp. 629–632, June 2010.
- [14] M.-S. Park and D. Chwa, "Swing-up and stabilization control of inverted-pendulum systems via coupled sliding-mode control method," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 9, pp. 3541–3555, 2009.
- [15] R.-J. Wai, M.-A. Kuo, and J.-D. Lee, "Design of cascade adaptive fuzzy sliding-mode control for nonlinear two-axis inverted-pendulum servomechanism," *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 5, pp. 1232–1244, 2008.
- [16] M. D. Ortigueira, *Fractional Calculus for Scientists and Engineers*, Springer, Berlin, Germany, 2011.
- [17] R. Caponetto, G. Dongola, L. Fortuna, and I. Petráš, *Fractional Order Systems: Modeling and Control Applications*, vol. 72 of *World Scientific Series on Nonlinear Science Series A*, World Scientific Publishing, Singapore, 2010.
- [18] D. Baleanu, K. Diethelm, E. Scalas, and J. J. Trujillo, *Fractional Calculus Models and Numerical Methods*, Series on Complexity, Nonlinearity and Chaos, World Scientific Publishing, Singapore, 2012.
- [19] D. Baleanu, J. H. Asad, and I. Petras, "Fractional-order two-electric pendulum," *Romanian Reports in Physics*, vol. 64, no. 4, pp. 907–914, 2012.
- [20] I. Podlubny, "Fractional-order systems and $PI^{\lambda}D^{\mu}$ controllers," *IEEE Transactions on Automatic Control*, vol. 44, no. 1, pp. 208–214, 1999.
- [21] I. Podlubny, I. Petráš, B. M. Vinagre, P. O'Leary, and L. Dorčák, "Analogue realizations of fractional-order controllers," *Nonlinear Dynamics*, vol. 29, no. 1–4, pp. 281–296, 2002.
- [22] C. Yeroglu and N. Tan, "Note on fractional-order proportional-integral-differential controller design," *IET Control Theory and Applications*, vol. 5, no. 17, pp. 1978–1989, 2011.
- [23] D. Maiti, A. Acharya, M. Chakraborty, A. Konar, and R. Janarthanan, "Tuning pid and $PI^{\lambda}D^{\delta}$ controllers using the integral time absolute error criterion," in *Proceedings of the 4th International Conference on Information and Automation for Sustainability (ICIAFS '08)*, pp. 457–462, December 2008.
- [24] S. Das, S. Das, and A. Gupta, "Fractional order modeling of a PHWR under step-back condition and control of its global power with a robust $PI^{\lambda}D^{\mu}$ controller," *IEEE Transactions on Nuclear Science*, vol. 58, no. 5, pp. 2431–2441, 2011.
- [25] F. Ikeda and S. Toyama, "Fractional derivative control designs by inhomogeneous sampling for systems with nonlinear elements," in *Proceedings of the SICE Annual Conference (SICE '07)*, pp. 1224–1227, September 2007.
- [26] S. K. Mishra and D. Chandra, "Stabilization of inverted cart-pendulum system using $PI^{\lambda}D^{\mu}$ controller: a frequency-domain approach," *Chinese Journal of Engineering*, vol. 2013, Article ID 962401, 7 pages, 2013.
- [27] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [28] R. Eberhart, Y. Shi, and J. Kennedy, *Swarm Intelligence*, Morgan Kaufmann, San Mateo, Calif, USA, 2001.

- [29] S. Yang, M. Wang, and L. Jiao, "A quantum particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '04)*, vol. 1, pp. 320–324, 2004.
- [30] K. Ogata, *Modern Control Engineering*, Prentice-Hall, Upper Saddle River, NJ, USA, 2002.
- [31] K. A. Åström and T. Häggglund, *PID Controllers: Theory, Design and Tuning*, Instrument Society of America, Research Triangle Park, NC, USA, 1995.
- [32] A. Oustaloup, F. Levron, B. Mathieu, and F. M. Nanot, "Frequency-band complex noninteger differentiator: characterization and synthesis," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 1, pp. 25–39, 2000.
- [33] D. Valério and J. S. da Costa, "Time-domain implementation of fractional order controllers," *IEE Proceedings Control Theory and Applications*, vol. 152, no. 5, pp. 539–552, 2005.
- [34] G. E. Carlson and C. A. Halijak, "Approximation of fractional capacitors (1/s) by a regular Newton process," *IRE Transactions on Circuit Theory*, vol. 11, no. 2, pp. 210–213, 1964.
- [35] A. Tepljakov, E. Petlenkov, and J. Belikov, "FOMCON: fractional-order modeling and control toolbox for MATLAB," in *Proceedings of the 18th International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES '11)*, pp. 684–689, June 2011.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

