Research Article

Genetic Algorithm and Particle Swarm Optimization Based Cascade Interval Type 2 Fuzzy PD Controller for Rotary Inverted Pendulum System

Mukhtar Fatihu Hamza, Hwa Jen Yap, and Imtiaz Ahmed Choudhury

Department of Mechanical Engineering, University of Malaya, 50603 Kuala Lumpur, Malaysia

Correspondence should be addressed to Mukhtar Fatihu Hamza; mfhamza@siswa.um.edu.my

This paper presents the design of an optimized Interval Type 2 Fuzzy Proportional Derivative Controller (IT2F-PDC) in cascade form for Rotary Inverted Pendulum (RIP) system. The parameters of the IT2F-PDC are optimised by using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The goal is to balance the pendulum in upright unstable equilibrium position. The IT2F-PDC which is the extended version of conventional type 1 fuzzy logic controller, improves the control strategy by using the advantage of its footprint of uncertainty for the fuzzy membership function. The performance characteristics considered for the controller are steady state error, settling time, rise time, maximum overshoot, and control energy. Experimental and simulation results indicated that the effectiveness and robustness of the proposed GA- and PSO-based controllers on the RIP with respect to load disturbances, parameter variation, and noise effects have been improved over state-of-the-art method. However, the comparative results for simulation and experiment based on cascade IT2F-PDC indicate that GA-based IT2F-PDC has lower steady state error while PSO-based IT2F-PDC has lower overshoot, settling time, and control energy, but both have almost the same rise time. The proposed control strategy can be regarded as a promising strategy for controlling different unstable and nonlinear systems.

1. Introduction

Most real industrial systems are nonlinear in nature and exhibit some level of uncertainty [1, 2]. In the past decade, some modern controls such as nonlinear control, adaptive control, variable structure control and optimal control were used [3–5]. Although these control strategies exhibit a very good performance, they are also complex and difficult to implement [6]. The conventional proportional integral derivative (PID) controller exhibits good performance for linear system and it is widely employed in industry due to its simple structure and robustness in different operation conditions. However, the tuning of the parameters of PID accurately becomes difficult because most of industrial plants are highly complex and have some issues such as nonlinearities, time delay, and higher order [7]. Due to the complexity of most industrial plants and the limitation of PID controller, an unprecedented interest was diverted to the applications

of the fuzzy logic controller (FLC). This is because it uses the expert knowledge and its control action is described by linguistic rules. Also, the FLC does not require the complete mathematical model of the system to be controlled and it can work properly with nonlinearities and uncertainties [1, 2, 8–13]. FLC are of two types, namely, type 1 fuzzy logic controller (T1FLC) and type 2 fuzzy logic controller (T2FLC). In T1FLC, the uncertainty is represented by a precise number in a range of (0, 1) interpreted as a degree of membership functions (MF). In view of the fact that it is too difficult to know a precise value for uncertainty, working with type 1 model is more reasonable. However, some researchers argued that, in case where there is high level of uncertainty, T1FLC has limited ability to handle it because its membership degree for each input is a crisp number [14]. The T2FLC which uses type 2 fuzzy set (T2FS) was introduced to circumvent the limitations of the T1FLC. The main characteristic of T2FLC is that its MFs are fuzzy. Therefore, it has more

degree of freedom in designing varieties of systems with uncertainties [1, 2, 8]. The control performance of T1FLC can be improved by T2FLC because it has the advantage of footprint of uncertainty (FOU) that can be used to improve the corresponding MF [15].

The major disadvantage of T2FLC is that the process of tuning becomes more difficult and much time consuming for increasing number of inputs and outputs of the controller [16]. This problem of tuning can be solved by using optimization methods [15, 16].

Looking at Rotary Inverted Pendulum (RIP) from the control point of view, the RIP exhibits many interesting and challenging properties, such as nonlinearities and instability. The RIP is in the class of underactuated mechanical systems. These features make RIP become known widely as experimental setup for testing linear and nonlinear control algorithms [11]. RIP consists of a rotational servomotor system which drives the output gear, rotational arm, and a pendulum. It has many important real applications like robotics, aerospace vehicles, pointing control, and marine vehicles [15].

Four control objectives of RIP as found in the literature are categorized as follows [17, 18]: (1) swing-up control: controlling the pendulum from downward (stable position) to upward (unstable position); (2) stabilization control, that is, regulating the pendulum to remain at the unstable position; (3) switching control, that is, the switching between swing-up control and stabilization control; (4) trajectory control, that is, to control the RIP in such a way that the arm tracks a desired time varying trajectory while the pendulum remains at unstable position.

Recently, a lot of research regarding the control of RIP has been published. For example, energy based compensation, PD cascade scheme, edge trigger counter, sliding mode control, type 1 fuzzy control, and adaptive control have been used for swing-up control of RIP [18–24]. To solve the stabilization control problem on RIP, multiobjective integral sliding mode controller, microcontrollers, fuzzy logic regulator, pole placement technique, optimized PID controller, and Linear Quadratic Regulator (LQR) have been applied [7, 18, 20, 23, 25–28]. Energy was considered for switching control in [18, 29], and mode controller was used for the same purpose in [20]. For trajectory control, adaptive PID with sliding mode control, linear active disturbance rejection control, hybrid of linear fusion function based on LQR mapping and adaptive control with ANFIS tuning, feedback linearization based controller, and energy based compensation controller was used [17, 30–32]. Optimized cascade type 1 fuzzy logic controller was used in [33] for controls of pendulum angle and arm angles of RIP.

To the best of the author's knowledge, at this moment there is not any kind of GA and PSO optimised type 2 fuzzy logic control applied to RIP. A cascade control method is effective for a system with high level of disturbances and large time error such as the RIP [24]. Also, as mentioned earlier type 2 fuzzy control strategy is effective and gives robust control response for systems with high level of uncertainty and/or inaccurate model. Putting type 2 fuzzy in cascade topology will have the advantages of type 2 fuzzy, and cascade

structure which will eventually give more robust controller for system with uncertainties and large time error [33].

In this paper, the IT2F-PDC is designed in cascaded form for RIP. The parameters of the IT2F-PDC are optimised using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The GA and PSO were chosen in this research in view of the fact that these algorithms are more established in the literature than other evolutionary algorithms [34, 35]. In addition, the GA and PSO have proven to improve performance over other algorithms in solving optimization problems [36, 37]. The goal is to balance the pendulum in upright position. The servo behaviour (reference tracking) of RIP is analysed. Disturbances rejection of the proposed controller is analysed by adding the internal noise and external disturbance to the system. Also, the controller is applied on the real RIP to validate the simulation results. The performances of GA and PSO for the optimization of the parameters of IT2F-PDC were compared. Also, the designing of IT2F-PDC as an optimization dilemma that slightly altered five performance indices was formulated which includes steady state error, settling time, rise time, maximum overshoot of the response, and control energy of the system.

The organisation of the paper is as follows. Section 2 introduces the RIP, Section 3 gives brief on fuzzy logic system, IT2FL-PDC, and the design of cascade IT2-PDC, and Section 4 defines optimization method and gives some basics on GA and PSO. The formulation of the problem is discussed in Section 5. Sections 6 and 7 present the simulation and experimental results, respectively. Finally, Section 8 presents conclusions.

2. RIP

The RIP is in the class of underactuated mechanical system, consisting of a servomotor system that drives the output gear. The rotational arm is attached to the output gear and a pendulum is attached at one end of the rotating arm as shown in Figure 1(a). The arm is driven by the gear with the aim of balancing the pendulum in an upright position. The angle (ϕ) is the angle of the arm and its direction depends on the direction of the control voltage (V_m) (voltage applied to the servomotor). In this study, the counterclockwise direction is considered as positive direction for the arm and the clockwise direction is the positive direction for the pendulum. The servomotor drives the rotating arm to move in horizontal plane XY in such a way that it sets the pendulum to the inverted position in the XZ direction, perpendicular to the arm. (α) is the pendulum angle. Figure 1(b) shows the picture of the experimental setup using Quanser RIP.

2.1. Nonlinear and Linear Dynamics Model of RIP. The dynamic equations that describe the motion of rotary arm and the pendulum with respect to the servomotor voltage were obtained using Euler-Lagrange equation, which is the systematic way of obtaining the equation of motion [38]. Once the kinetic and potential energy are obtained and the Lagrangian is found. Subsequently, the task is to compute various derivatives to get the equations of motion of the system.
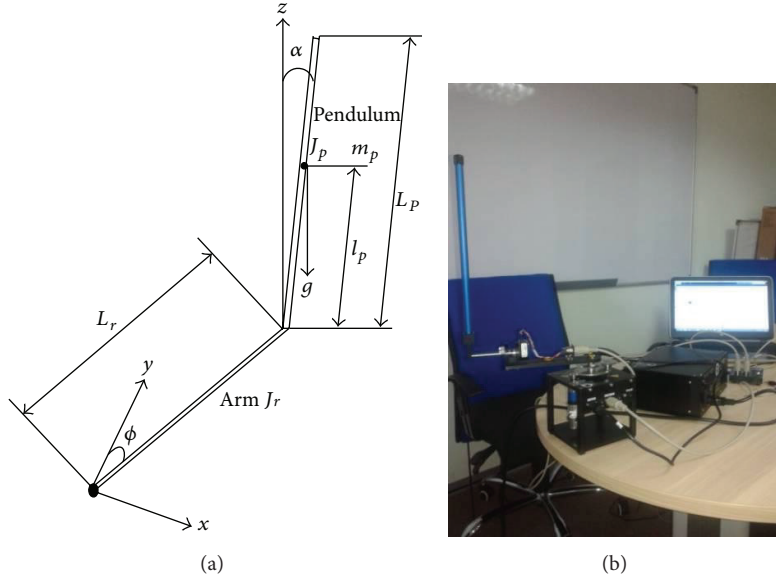
FIGURE 1: (a) Convention of RIP, (b) Quanser RIP experimental setup.

After going through the process, the nonlinear equations of motion for the RIP can be found [38]. Considering the total length of the pendulum (L_p), viscous damping coefficient is seen at the pivot axis for arm and pendulum (B_r) and (B_p), respectively, and moment of inertia about the centre of mass for arm and pendulum (J_r) and (J_p), respectively. The nonlinear equations of motions of RIP are found as follows:

$$
\left(m_p L_r^2 + \frac{1}{4}m_p L_P^2 - \frac{1}{4}m_p L_P^2 \cos(\alpha)^2 + J_r\right)\ddot{\phi}
$$

$$
- \left(\frac{1}{2}m_p L_p L_r \cos(\alpha)\right)\ddot{\alpha}
$$

$$
+ \left(\frac{1}{2}m_p L_P^2 \sin(\alpha)\cos(\alpha)\right)\dot{\alpha}\dot{\phi}
$$

$$
+ \left(\frac{1}{2}m_p L_p L_r \sin(\alpha)\right)\dot{\alpha}^2 = \tau - B_r\dot{\phi}, \tag{1}
$$

$$
\left(-\frac{1}{2}m_p L_p L_r \cos(\alpha)\right)\ddot{\phi} + \left(J_p + \frac{1}{4}m_p L_P^2\right)\ddot{\alpha}
$$

$$
- \left(\frac{1}{4}m_p L_P^2 \cos(\alpha)\sin(\alpha)\right)\dot{\phi}^2
$$

$$
- \left(\frac{1}{2}m_p L_p g \sin(\alpha)\right) = -B_p\dot{\alpha}.
$$

Let (1) be called nonlinear 1. Alternatively the nonlinear equation of RIP can be found in (2) when the length of the pendulum from its centre of mass (l_p), equivalent moment of inertia as seen at the load (J_{eq}), and equivalent viscous

damping coefficient as seen at the load (B_{eq}) are considered. Let (2) be called nonlinear 2. Consider

$$
\left(J_{eq} + m_p L_r^2\right)\ddot{\phi} - \left(m_p l_p L_r \cos(\alpha)\right)\ddot{\alpha}
$$

$$
+ \left(m_p l_p L_r \sin(\alpha)\right)\dot{\alpha}^2 = \tau - B_{eq}\dot{\phi}, \tag{2}
$$

$$
\left(\frac{4}{3}m_p l_p^2\right)\ddot{\alpha} - \left(m_p l_p L_r \cos(\alpha)\right)\ddot{\phi} - m_p l_p g \sin(\alpha) = 0.
$$

The torque (τ) at the load gear is generated by servomotor and is described by

$$
\tau = \frac{\eta_g K_g \eta_m k_t \left(V_m - K_g k_m \dot{\phi}\right)}{R_m}. \tag{3}
$$

The linearized model of the nonlinear equations of RIP can be found by substituting $\sin(\alpha) = \alpha$ and $\cos(\alpha) = 1$ in nonlinear equation [33] and presented in matrix form in (4)–(7). The summary descriptions of the parameters and their values are given in Table 1. Consider

$$
\begin{bmatrix} \dot{\phi} \\ \dot{\alpha} \\ \ddot{\phi} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{k_2 k_4}{k_5} & \dfrac{-k_3 k_7}{k_5} & 0 \\ 0 & \dfrac{k_1 k_4}{k_5} & \dfrac{-k_2 k_7}{k_5} & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \alpha \\ \dot{\phi} \\ \dot{\alpha} \end{bmatrix}
$$

$$
+ \begin{bmatrix} 0 \\ 0 \\ k_3 \dfrac{\eta_g K_g \eta_m k_t}{R_m k_5} \\ k_2 \dfrac{\eta_g K_g \eta_m k_t}{R_m k_5} \end{bmatrix} V_m, \tag{4}
$$

TABLE 1: The summary of the parameters values of RIP.

Symbol	Description	Value	Unit
m_p	Mass of pendulum	0.127	kg
L_p	Total length of pendulum	0.337	m
J_p	Pendulum moment of inertia about its centre of mass	0.0012	kg·m^2
B_p	Pendulum viscous damping coefficient as seen at the pivot axis	0.0024	N·m·s/rad
l_p	Length of pendulum centre of mass	0.156	m
L_r	Rotary arm length	0.168	m
B_r	Rotary arm viscous damping coefficient as seen at the pivot axis	0.0024	N·m·s/rad
J_r	Rotary arm moment of inertia about its centre of mass	0.000998	kg·m^2
R_m	Motor armature resistance	2.6	Ω
k_t	Motor current-torque constant	0.00768	N-m/A
k_m	Motor back-emf constant	0.00768	V/(rad/s)
K_g	High-gear total gear ratio	70	
η_m	Motor efficiency	69	%
η_g	Gearbox efficiency	90	%
J_{eq}	Equivalent moment of inertia as seen at the load	0.0036	kg·m^2
B_{eq}	Equivalent viscous damping coefficient as seen at the load	0.004	Nms/rad

where $k_1 = J_{eq} + m_p L_r^2$, $k_2 = m_p l_p L_r$, $k_3 = (4/3)m_p l_p^2$, $k_4 = l_p m_p g$, $k_5 = k_1 k_3 - k_2^2$,

$$k_7 = \frac{\eta_g K_m \eta_m k_t k_g^2 + B_{eq} R_m}{R_m}, \tag{5}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\alpha} \\ \ddot{\phi} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 34.91 & -16.21 & 0 \\ 0 & 75.44 & -13.12 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \alpha \\ \dot{\phi} \\ \dot{\alpha} \end{bmatrix}$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 28.56 \\ 23.07 \end{bmatrix} V_m, \tag{6}$$

$$\begin{bmatrix} \phi \\ \alpha \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \alpha \\ \dot{\phi} \\ \dot{\alpha} \end{bmatrix}. \tag{7}$$

The open loop poles for the proposed RIP system are $w = [0, 7.491, -18.04, -5.6596]$. The system has one pole at the right hand side of s-plane which confirms that the proposed RIP system is unstable. Therefore, before continuing with any control action, the test for controllability of the system has to be done, in order to check for the rank of the matrix W in (8). If W has full rank then the system is controllable [39]:

$$W = [B, AB, A^2 B, A^3 B, \ldots, A^{n-1} B], \quad \text{rank}(W) = 4. \tag{8}$$

The matrix W has full rank; then the system is controllable, meaning that the RIP system has complete state controllability.

2.2. Open Loop Response of Linear and Nonlinear Model of RIP. The open loop responses for nonlinear 1, nonlinear 2, and linear dynamic of RIP are shown in Figures 2(a) and 2(b). Initially the pendulum is positioned in an inverted position with very small displacement (0.005°), and then it is allowed to fall by applying a pulse signal to the model. The simulation results show that nonlinear 1 and nonlinear 2 are similar and their behaviours are the same for both pendulum angle and arm angle. This shows that nonlinear 1 and nonlinear 2 can be considered as nonlinear model of the RIP because nonlinear 1 has already been proved and used in [33]. The linear model depicts the nonlinear pendulum motion for the first 1.3 seconds, until it attains 21◦; then it began to deviate from the actual motion. The response shows that the whole system is nonlinear and unstable.

3. Fuzzy Logic Controller

A concise overview of FLC was presented in this section with the intention of providing the basic knowledge needed to understand the basic idea and formulation of interval type 2 fuzzy logic controller.

3.1. Interval Type 2 Fuzzy Logic System (IT2FLS). The idea of fuzzy logic systems and T2FS was pioneered by Zadeh in 1965 and 1975, respectively [40, 41]. The uncertain knowledge is used to build the fuzzy logic rules which leads to uncertain antecedents and consequents of the rule to be uncertain, which can not be handled by conventional type 1 fuzzy set MF. This leads to the introduction of type 2 fuzzy logic which can handle the issue of uncertainty by using the advantage of FOU [2, 12]. All the secondary grades of the IT2FS are equal to 1, and it is completely described by upper MF and lower MF (UMF and LMF). Figure 3(a) shows the triangular MF of
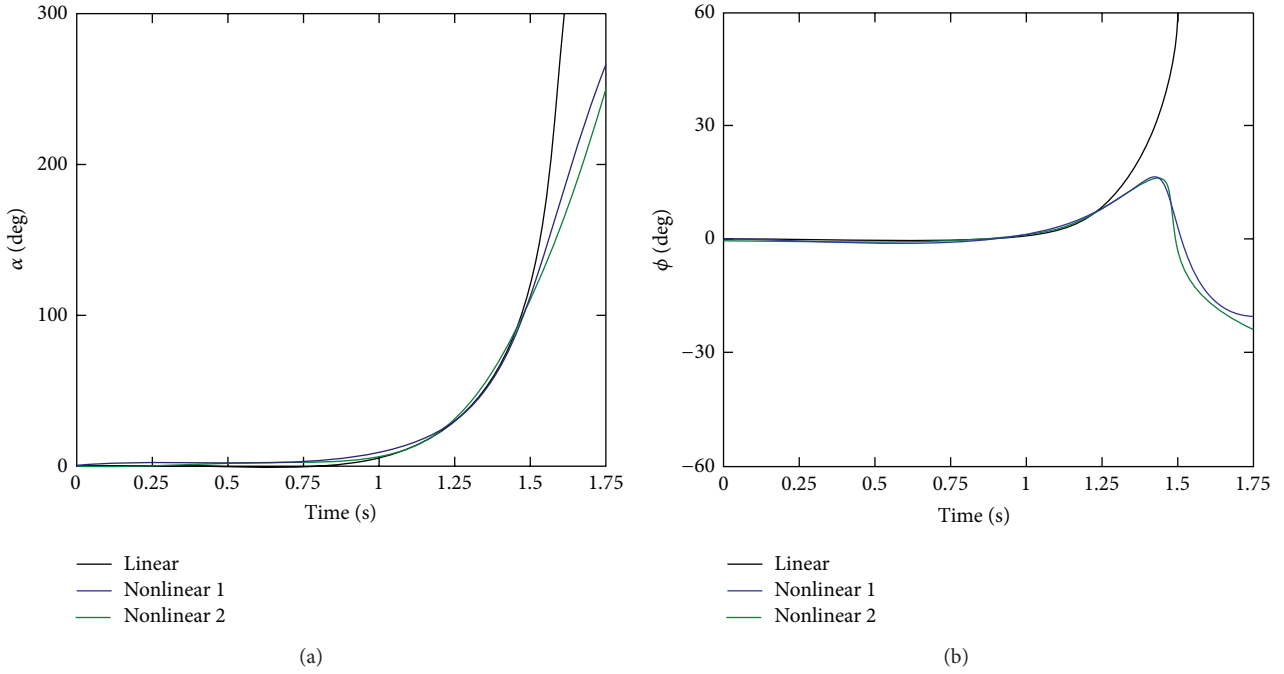
FIGURE 2: Open loop response for RIP: (a) pendulum angle, (b) pivot arm angle.

IT2FLS and its associated quantities. Also the corresponding secondary MF is shown in Figures 3(b) and 3(c).

T2FLS and T1FLS are similar in terms of their normal architecture. The main difference between them is in their structure. The defuzzifier block in type 1 fuzzy is substituted with the output processing in type 2 fuzzy comprised of type reduction and defuzzifier blocks [15]. The block diagram of T2FLS is shown in Figure 4.

3.2. Fuzzification. The fuzzifiers in T1FLS and T2FLS are doing the same work, which is transforming numeric vector entries $x = (x_1, \ldots, x_p)^T \in X_1 * X_1 * \cdots * X_p \equiv X$ into $\widetilde{A}_x$ (type 2 fuzzy set) defined in X. Giving the singleton numeric inputs, the mapping can be performed as follows [16]:

$$\mu_{\widetilde{A}_x}(x) = 1/1 \text{ with } x = x',$$

$$\mu_{\widetilde{A}_x}(x) = 1/0, \tag{9}$$

$$\text{for } \forall x \in x \text{ with } x \neq x'.$$

Equation (9) shows that $\mu_{\widetilde{x}_i}(x_i) = 1/1$ when $x_i = x_i'$ and $\mu_{\widetilde{x}_i}(x_i) = 1/0$ when $x_i \neq x_i'$ for all $i = 1, \ldots, p$.

3.3. Rules. Both T1FLS and T2FLS use IF-THEN rules. In type 2, the antecedents and consequent MF are represented by T2FS. The ith rule can be expressed as

$$R^i : \text{IF } x_1 \text{ is } \widetilde{A}_1^i, \text{ and} \cdots x_p \text{ is } \widetilde{A}_p^i,$$

$$\text{THEN } Y^i = C_0 + C_1^i x_1 + \cdots + C_r^i x_p, \tag{10}$$

where $i = 1, \ldots, m$; C_j^i $(j = 0, 1, \ldots, p)$ are the consequent type 1 fuzzy set; $\widetilde{A}_k^i$ $(k = 1, \ldots, p)$ are type 2 antecedent fuzzy set.

3.4. Inference. The inference mechanism in T2FLS is like the one in type 1 fuzzy. It is a rule combination to produce a mapping from input T2FSs to output T2FSs. It is necessary to calculate the intersection, union, and composition of type 2 relations in order to realise this mapping [15]. The T2FS $\widetilde{A}_x$ whose MF is $\mu_{\widetilde{A}_x} = \cap_{i=1}^{p} \mu_{x_i}(x_i)$, where x_i $(i = 1, \ldots, p)$ are level of fuzzy sets describing the input. The result of type 2 input and antecedent process which are in the firing set $\cap_{i=1}^{p} \mu_{F_{i_i}}(x_i' \equiv F^i(x'))$ is an interval type 1 fuzzy set as in (11)-(12) [9]:

$$F^i(x') = \left[\underline{f}^i(x'), \overline{f}^i(x')\right] \equiv \left[\underline{f}^i, \overline{f}^i\right], \tag{11}$$

where

$$\underline{f}^i(x_i') = \underline{\mu}_{F_1^i}(x_1') * \cdots * \underline{\mu}_{F_p^i}(x_p'),$$

$$\overline{f}^i(x_i') = \overline{\mu}_{F_1^i}(x_1') * \cdots * \overline{\mu}_{F_p^i}(x_p'). \tag{12}$$

3.5. Output Processing. The output processing constitutes the type reduction that generates type 1 fuzzy set and defuzzifier that converts the generated type 1 fuzzy set to the crisp output [15]. In this paper, the Wu-Mendel uncertainty bound method was used to approximate the type-reduced set, and
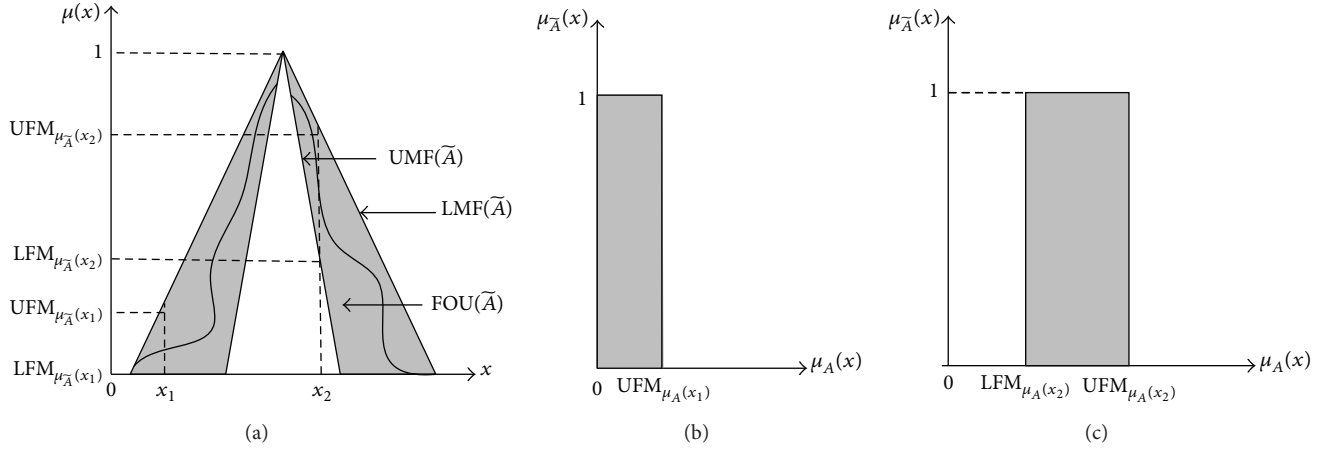
Figure 3: (a) FOU of Triangular MF of type 2 fuzzy logic and its associated quantities (b) and (c) the corresponding secondary MF.
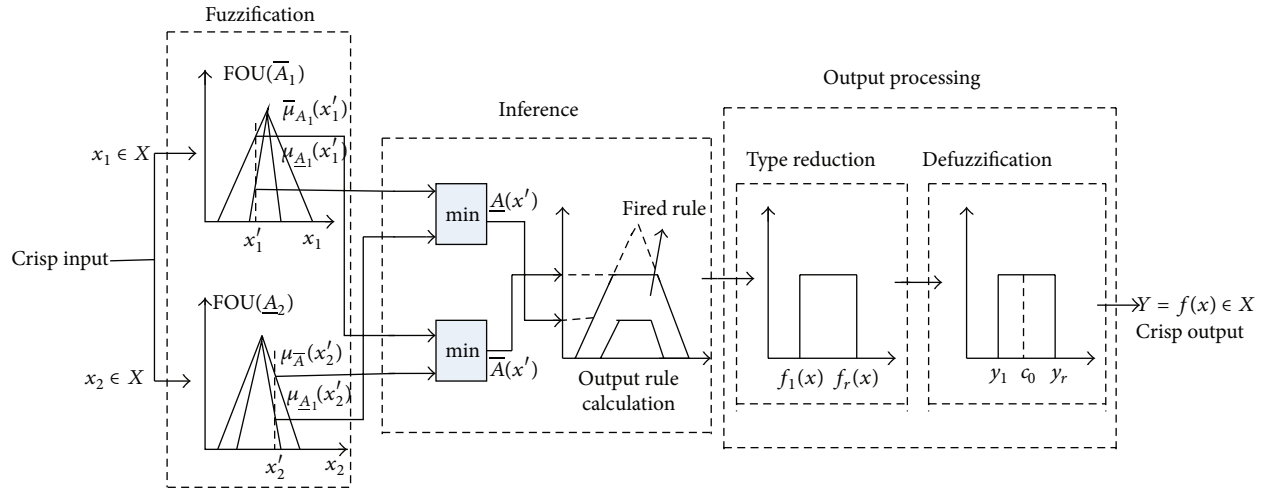

Figure 4: Type 2 fuzzy logic system.

Karnik and Mendel (KM) algorithm was used for calculating the two end points of centroids of y_l and y_r [14] presented in

$$y_1 = \frac{\sum_{i=1}^{M} f_1^i y_1^i}{\sum_{i=1}^{M} f_1^i},$$

$$y_r = \frac{\sum_{i=1}^{M} f_r^i y_r^i}{\sum_{i=1}^{M} f_r^i}. \tag{13}$$

Once y_1 and y_r are found then the average of y_1 and y_r is taken so that the output of the defuzzifier of an interval singleton type 2 fuzzy is as follows:

$$y(x) = \frac{y_1 + y_r}{2}. \tag{14}$$

3.6. Cascade Interval Type 2 Fuzzy Logic PD Controller. In this paper, the cascade controller was designed for RIP. The RIP is a single input multiple output (SIMO) system. In SIMO systems, change of one output by some disturbances affects the control of the other output [33]. Considering

nonlinearities behaviour of RIP system, it is difficult to achieve the past settling time. Also it has high level of disturbances and large time constant. For system like this the best control strategy is cascade because it has the advantage of attenuating the effect of disturbances and improving the dynamics of entire control loop [24]. The structure of the cascade interval type 2 fuzzy logic PD controller is shown in Figure 5. Both controllers have two inputs and one output. The overall architecture comprises the outer and the inner controllers. The input of the inner controller is the error $E_2(t)$ and change in error $\Delta E_2(t)$, its output is the control voltage to the servomotor $V_m(t)$, and three gains g_4, g_5, and g_6 are used to scale $e_2(t)$, $\Delta e_2(t)$, and $v_m(t)$, respectively, as shown in

$$E_2(t) = g_4 e_2(t) = g_4\left(U(t) - \phi(t)\right),$$

$$\Delta E_2(t) = g_5 \Delta e_2(t) = g_5\left(e_2(t) - e_2(t-1)\right), \tag{15}$$

$$V_m(t) = g_6 v_m(t),$$

where t is the instance sampling, $U(t)$ is the control signal from the outer loop, and $\phi(t)$ is the arm angle. The input of the
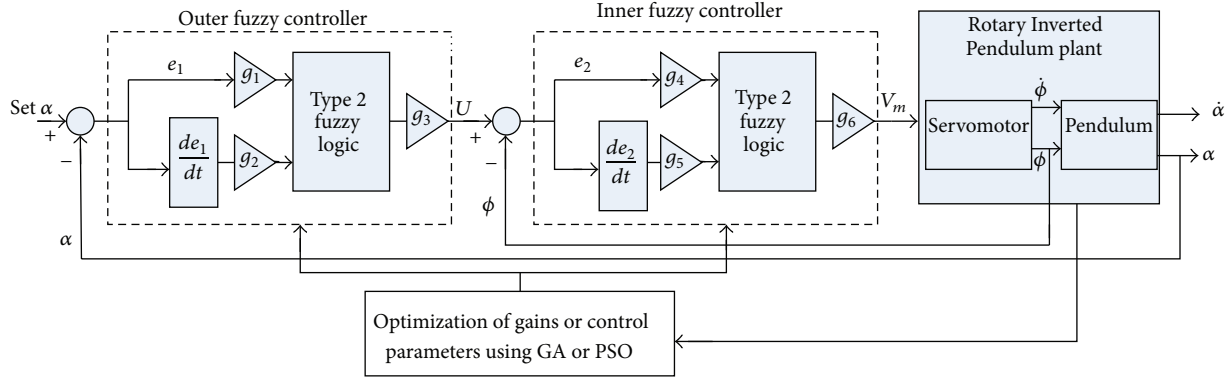
FIGURE 5: Overall architecture of IT2FL-PDC in cascade form.

TABLE 2: Fuzzy rules of IT2FL-PDC.

Δe	e				
	NL	NS	ZO	PS	PL
NL	PL	PL	PL	PS	ZO
NS	PL	PL	PS	ZO	NS
ZO	PL	PS	ZO	NS	NL
PS	PS	ZO	NS	NL	NL
PL	ZO	NS	NL	NL	NL

TABLE 3: Optimised gains of cascade IT2FL-PDC.

Gains	$\alpha_r = 0$		$\alpha_r = 180$	
	GA	PSO	GA	PSO
g_1	−9.5701	−9.6046	6.5632	5.9825
g_2	−1.3161	−1.0527	0.0119	0.0102
g_3	3.2616	2.7104	8.9646	10.0109
g_4	−1.3664	−3.2773	0.3679	0.1998
g_5	3.8597	6.2357	0.0612	0.0701
g_6	0.8597	0.6589	0.3951	0.5094

outer controller is the error $E_1(t)$ and change in error $\Delta E_1(t)$, its output is the control signal to the inner loop $u(t)$, and three gains g_1, g_2, and g_3 are used to scale $e_1(t)$, $\Delta e_1(t)$, and $u(t)$, respectively, as shown in

$$E_1(t) = g_1 e_1(t) = g_1 (\alpha_r(t) - \alpha(t)),$$
$$\Delta E_1(t) = g_2 \Delta e_1(t) = g_2 (e_1(t) - e_1(t-1)), \quad (16)$$
$$U(t) = g_3 u(t),$$

where α_r and α are reference pendulum angle and pendulum angle, respectively. These scaling gains $(g_1, g_2, \ldots, g_6)$ are adjustable parameters just like in any PID controller, and they are used to calibrate input and output [11]. The excellent performance of the controller depends on the values of these gains [16]. The trial-and-error approach can be used to find appropriate values of the gains but it is not feasible, so there is a need of using a systematic procedure which is easier for finding the optimised values of these gains. In this paper GA and PSO are used (Table 3). The triangular MF of T2FS was used and it is defined in the input and output spaces as shown in Figures 6(a) and 6(b). These MFs are the same for both two input variables (e and Δe) and they are applied to both controllers (inner and outer). The fuzzy rules of IT2F-PDC are shown in Table 2. Where NL, NS, ZO, PS, and PL are negative large, negative small, zero, positive small, and positive large respectively.

4. Optimization Method

Optimization refers to a class of soft computing techniques that relate to obtaining the optimal or satisfactory or best

solution for a particular problem, and the solution may be absolutely best out of some other possible solution [42].

GA and PSO are stochastic global optimization techniques. Over the past years GA has received a considerable attention and it is applied for searching the optimal fuzzy parameters which lead to the genetic fuzzy system [9, 10]. Although GA is successful in so many applications, sometimes it has a problem of getting trapped before it reaches the optimal region of the search space, especially for multimodal and highly dimensional problems [43, 44]. The advantages that make PSO attract more attention are as follows: the particles in PSO have memory, all the particles retained the knowledge of good solution so far, and there is a constructive cooperation among the particles, meaning the particles in the swarm share the information among themselves [45].

4.1. Genetic Algorithm. The basic foundation of GAs was proposed in 1975 by John Holland [46]. It is based on Darwin's ideas. Darwin's stated that, in a competing environment, the stronger individuals are more likely to be the winners [47]. GA is a metaheuristic search algorithm based on natural selection and genetic process [46, 47]. In GA, the potential solution to a problem is an individual which can be represented by the set of parameters. These parameters are just like a gene of a chromosome and can be represented by the string of values in binary form [46]. The fitness value is used to test the degree of goodness of the chromosome for solving a problem that is directly related to the objective value. The operators employed in a simple GA include selection, crossover, and mutation [46]. GAs are often regarded as
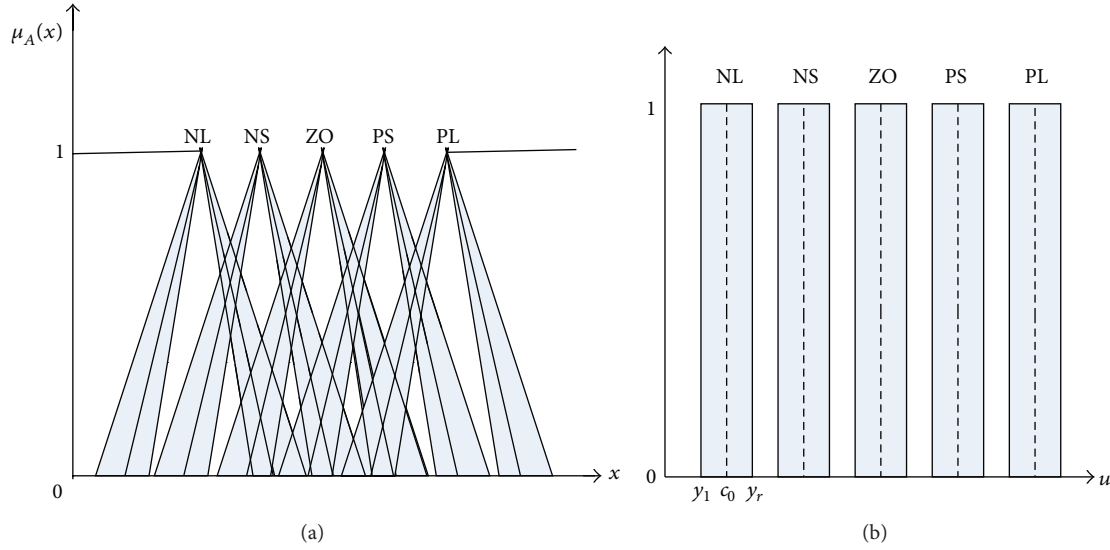
FIGURE 6: MFs of the (a) premise input variables, (b) input variables.

function optimizers and they have been applied in many optimization problems such as energy consumption [48]. In particular, the use of GAs for fuzzy systems design equips them with the adaptation and learning capabilities which bring about genetic fuzzy systems [10].

4.2. Particle Swarm Optimization. The PSO was introduced in 1995 by Eberhart and Kennedy [49]. PSO is metaheuristic search algorithm based on social and population behaviour, just like flocking of bird or fish schooling. The population in PSO is called swarm that can contain many particles. Each particle in PSO updates its velocity based on (17) [49, 50]. Consider

$$V_i(m) = X\left[V_i(m-1) + c_1 r_1 \left(P_{\text{best}_i} - Q_i(m-1)\right)\right.$$
$$\left. + c_2 r_2 \left(g_{\text{best}} - Q_i(m-1)\right)\right],$$

$$X = \frac{2}{\left|2 - \beta - \sqrt{\beta^2 - 4\beta}\right|}, \quad \beta = c_1 + c_2 > 4,$$

(17)

where P_{best} is the best position attained for the individual particle and g_{best} is the best position attained for the particle among all the population. P_i is the position of particle i, r_1 and r_2 are random numbers between 0 and 1, c_1 and c_2 are position constants learning rate, and X is the constriction factor. Each particle changes its position toward the P_{best} and g_{best} based on the updated velocity as given in (18) [51]. Consider

$$P_i = P_i(m-1) + V_i(m). \tag{18}$$

The velocity is restricted within $[-V_{\max}, +V_{\max}]$. If the velocity deviates from this range it has to be forced to be within the range [50, 51].

The following steps are used for the implementation of PSO algorithm.

Step 1. Specify the upper and lower bounds of the controller parameter and generate the particles randomly; the values of the performance criteria in time domain are calculated iteratively by sending each controller parameter (particle) to Matlab Simulink; after that, the cost function is evaluated for each particle according to these performance criteria. Then evaluate each particle in the initial population using the objective function and search for p_{best} and g_{best}.

Step 2. Calculate the velocity and the constriction factor for the particles and check for the maximum velocity; then update the velocity and position of each particle.

Step 3. For each particle, p_{best} is reset in comparison with the previous p_{best} through fitness of objective function; then g_{best} is updated in comparison with best p_{best}.

Step 4. If one of the terminating conditions is satisfied, then stop; else go to Step 2.

Step 5. The particle that has the latest g_{best} is an optimal parameter.

The trial-and-error method commonly used in the literature [52] was used for the selection of PSO and GA parameters: For PSO method, number of particles = 25; position constant learning rate $c_1 = c_2 = 2$; maximum iteration = 150; and constriction factor = 0.4. For GA method, number of individuals = 25; crossover rate = 0.5; mutation rate = 0.1; maximum generation = 150; also the type of operators used for each population in GA is linear ranking selection algorithm, simple crossover, and uniform mutation. Both algorithms are implemented in Matlab and each of the methods is tested in 60 independent runs with 60 distinct initial trial solutions. All these were conducted in Matlab R2013a, 2.4 GHz processor with 8 GB RAM.
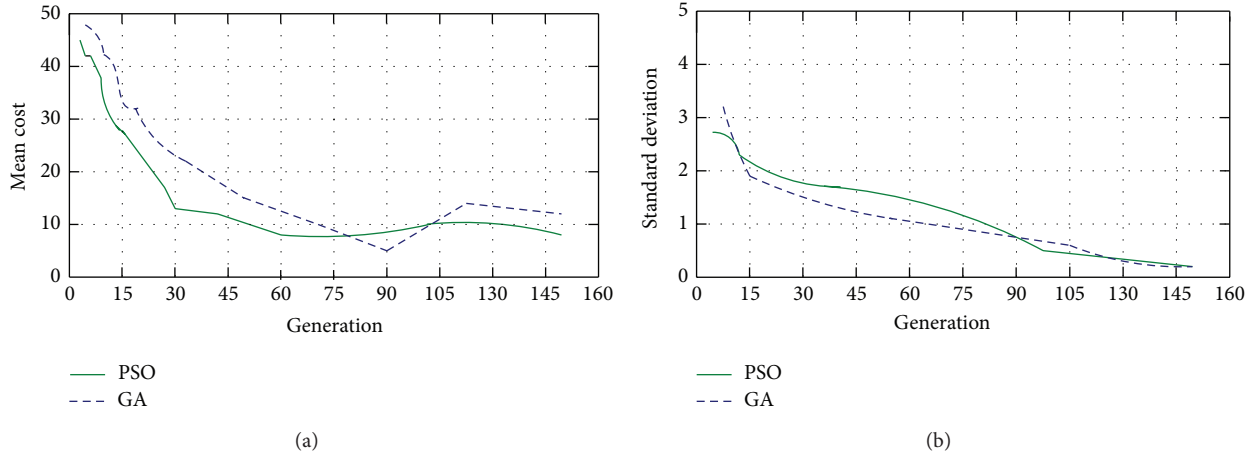
FIGURE 7: Tendency of convergence of (a) mean value of cost function, (b) standard deviation of cost function.

TABLE 4: Best results of the cascade IT2FL-PDC for RIP system obtained by GA and PSO in 60 runs.

Optimization method	α_r (deg)	Condition for disturbances	M_p (%)	E_{ss}	t_s (s)	t_r (s)	E_u (J)	Cost
GA	0	No disturbances	27.7	0.0009	1.46	0.0089	18.375	18.210
		With disturbances	30.8	0.0021	1.69	0.0086	18.973	19.531
	180	No disturbances	28.4	0.0018	1.52	0.0086	21.485	19.997
		With disturbances	32.8	0.0028	2.83	0.0089	22.032	21.903
PSO	0	No disturbances	21.4	0.0029	1.31	0.0088	17.094	15.551
		With disturbances	24.6	0.0035	1.52	0.0090	19.759	17.934
	180	No disturbances	22.1	0.0081	1.39	0.0086	18.981	16.732
		With disturbances	26.9	0.0099	2.34	0.0089	20.995	19.431

5. Problem Formulation

In this research, the performance index which consists of control energy (E_u), steady state error (E_{ss}), settling time (t_s), rise time (t_r), and overshoot (M_p) was considered. The appropriate IT2FL-PDC parameters that minimize the performance index were searched, and the cost function proposed is given in

$$\text{cost} = \frac{1}{2}\left(M_p + E_{ss} + E_u\right) - \frac{e^{-\gamma}}{2}\left(t_r - t_s + E_{ss} + M_p\right). \quad (19)$$

In this study, the weighing factor (γ) was considered to be equal to 1. The standard deviation (σ) and the mean value (λ) of cost value of each individual were examined in order to measure the dynamic and convergence characteristic of the proposed methods. The standard deviation is used to measure the convergence speed while the mean value is used to measure the accuracy of the algorithm. Equations (20) and (21) show the formula for calculating σ and λ, respectively [53]. Consider

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\text{cost}_{p_i} - \lambda\right)^2}, \quad (20)$$

$$\lambda = \frac{\sum_{i=1}^{n}\text{cost}_{p_i}}{n}, \quad (21)$$

where n is the population size and cost_{p_i} is the individual cost value.

6. Results and Discussion

For us to see the convergence characteristics of the controllers, two simulations are performed (mean and standard deviation). Figure 7(a) shows that both controllers (GA-based and PSO-based) secure stable mean cost value, using the same simulation conditions and cost function. However, PSO-based controller has best mean value and cost value, which indicate that it can achieve better accuracy than GA-based. Similarly, Figure 7(b) shows that in the tendency of convergence of standard deviation of cost values PSO-based controller is faster than GA-based controller. This indicates that PSO method has the best convergence efficiency. The run time in 150 iterations for PSO is 6,768.39 sec and for GA is 14,496.19 sec. The summary of the best simulation results in 60 runs under different operating condition is shown in Table 4. It can be observed that the percentage overshoot is lower for PSO-based method compared to each correspondent GA-based method. The improvements in overshoot are 22.75% (for $\alpha_r = 0$, no disturbances), 20.1% (for $\alpha_r = 0$, with disturbances), 22.2% (for $\alpha_r = 180$, no disturbances), and 18.0% (for $\alpha_r = 0$, with disturbances). The settling time is smaller in GA-based method in all the
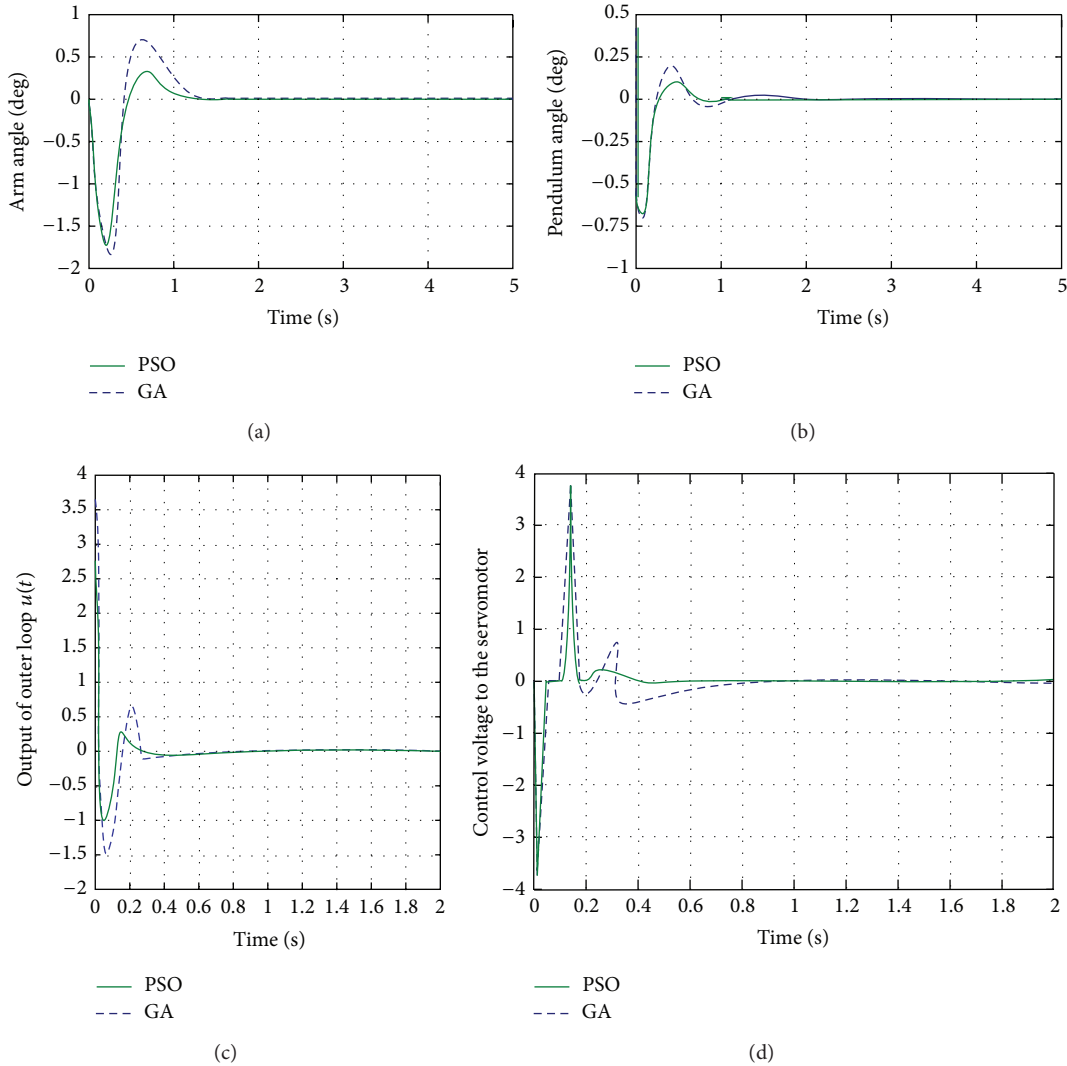
(a)

(b)

(c)

(d)

FIGURE 8: Simulated results for α_r equal to zero: (a) pendulum angle, (b) arm angle, (c) output of the outer controller, and (d) output of inner controller (control voltage to the servomotor).

simulations, but the rise time is almost the same for both PSO and GA in all simulations.

6.1. Reference Tracking. In this section, the servo behaviour (reference tracking) has been analysed by using two different values of reference pendulum angle (α_r). First α_r was set to be equal to zero; that is, the pendulum is at equilibrium upright unstable position and we want to control it to go to a stable position, Figures 8(a), 8(b), 8(c), and 8(d) show the best results of the pendulum angle, arm angle, output of outer controller, and the control voltage to the servomotor, respectively.

Second α_r was set to be equal to π (180°); that is, the pendulum is at stable position and we want to control it to go equilibrium upright unstable position. Figures 9(a), 9(b), 9(c), and 9(d) show the best results of the pendulum angle, arm angle, output of outer controller, and the control voltage to the servomotor, respectively.

6.2. Disturbances Rejection Analysis. The internal noise and external disturbance was added to the system in order to test for the robustness of the proposed controllers as follows: A load of 0.052 m height and 0.045 kg mass was added to the end of the pendulum, also, the white noise of 0.00634 power and 5% parameter value changes was added to the process output as shown in Figure 10. The simulation results shown in Figure 11 indicate the effectiveness and robustness of the proposed controllers.

7. Validation and Comparison

The experiments were performed on Quanser SRV02 RIP setup. The US Digital S1 single-ended optical shaft encoder that can offer a high resolution of 1024 lines per revolution (4096 counts per revolution in quadrature mode) was used for measuring the pendulum angle and arm angle. The power amplifier used was VoltPAQ-X1. The data acquisition device
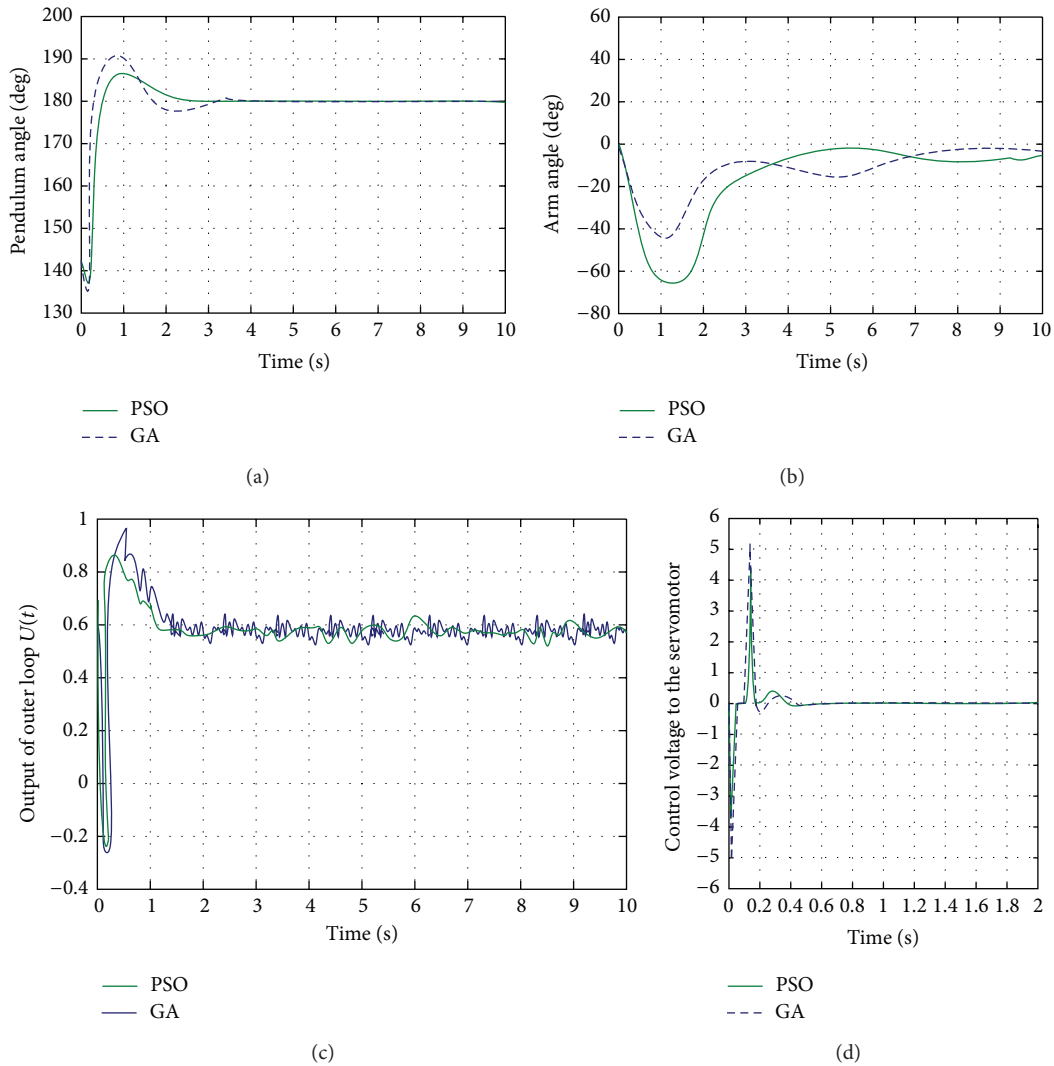
FIGURE 9: Simulated results for α_r equal to π: (a) pendulum angle, (b) arm angle, (c) output of the outer controller, and (d) output of inner controller (control voltage to the servomotor).

used was Quanser Q2-USB. The experimental results that compared the proposed controllers and the conventional energy based controller for pendulum angle, arm angle, and control voltage to the servomotor are shown in Figures 12(a), 12(b), and 12(c), respectively. The time starts from zero in order to show the performances from the swing-up to the balance mode. Looking at the real plant's results, some initial oscillations were noticed due to the swing motion needed to bring the pendulum from stable position to the vertical unstable position. Also the two controllers (PSO-based and GA-based) manifest the considerable level of robustness. On the other hand, the conventional energy based controller manifests a lot of oscillations before it becomes stable. The time taken to reach the steady state is higher for conventional energy based compared to type 2 fuzzy (for both PSO- and GA-based) controllers. The experimental results agreed with the simulation results as shown in Figures 8, 9, and 10 which justifies the availability of the system models and conforms

the performance of the proposed methods. Experimental and simulation results indicated that the effectiveness and robustness of the proposed controllers with respect to load disturbances, parameter variation, and noise effects have been improved over state-of-the-art method. In summary, it can be concluded based on the evidence emanated from the experiment results that the GA- and PSO-based controllers have advanced the performance of the conventional energy based controller on the RIP.

8. Conclusions

In this study, a cascade type 2 fuzzy logic PD controller was designed with the aid of optimization methods realised by GA and PSO. The application of the proposed controller was tested for controlling the pendulum angle and arm angle of RIP system which is highly nonlinear and unstable. The
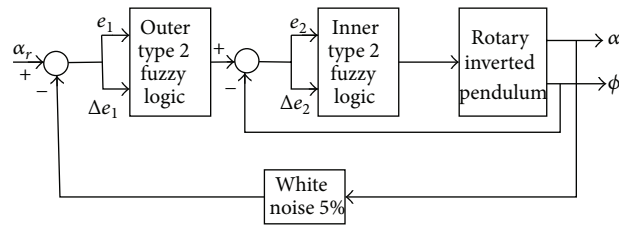
FIGURE 10: Cascaded type 2 fuzzy PD controller with source of noise.
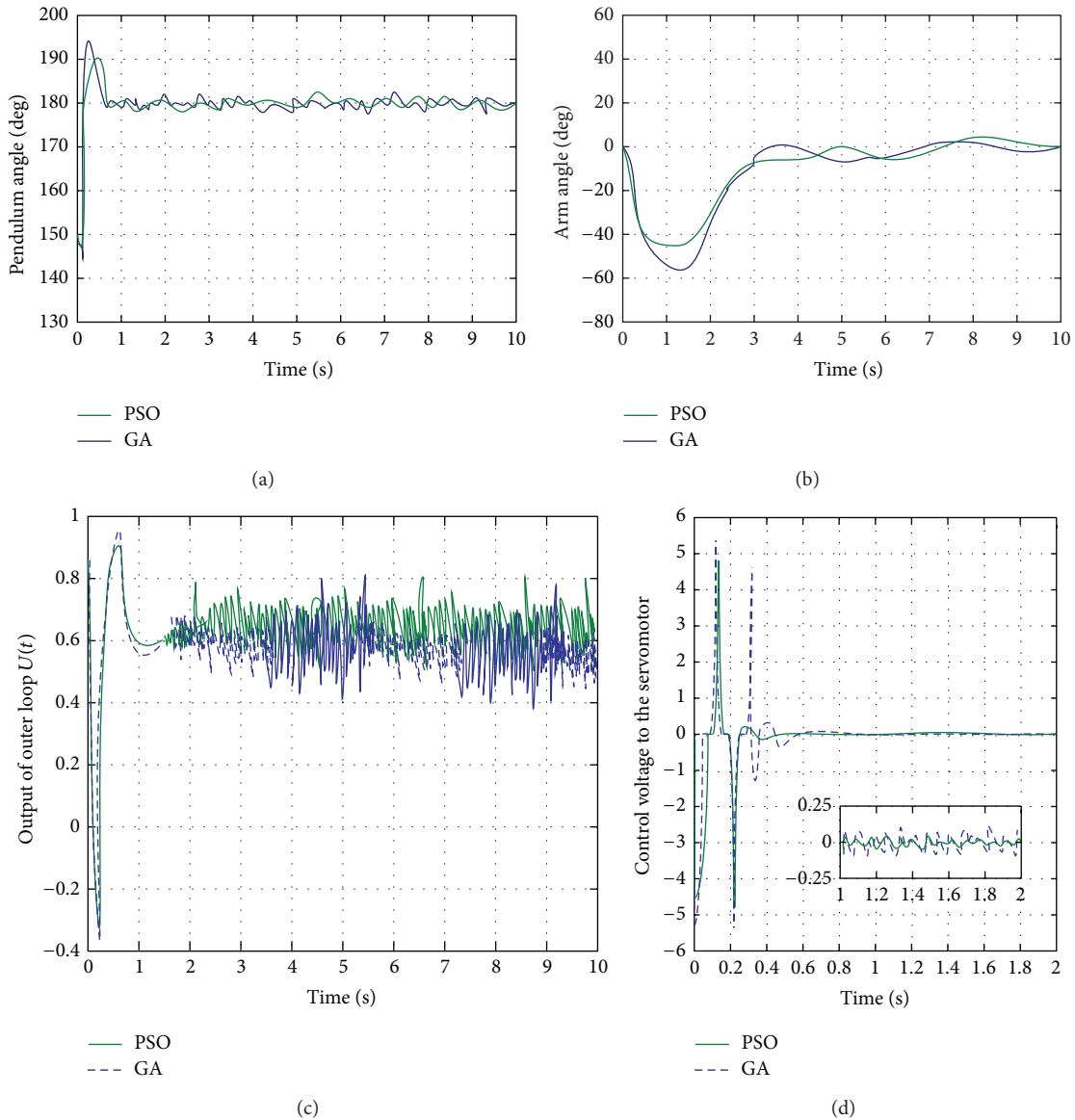

(a)

(b)

(c)

(d)

FIGURE 11: Simulated results for system with disturbances: (a) pendulum angle, (b) arm angle, (c) output of the outer controller, and (d) output of inner controller (control voltage to the servomotor).

model of the RIP was also realised in two different ways, and the architecture of type 2 fuzzy PD controller was discussed.

The comparative analysis was done in order to test the performance of the proposed controllers in terms of reference tracking and the disturbances rejection. Both the simulations

and experimental results show that both PSO and GA can be used effectively for optimization of parameters of the proposed controller. Moreover, based on this study, PSO and GA optimized cascade type 2 fuzzy PD controllers show some certain level of robustness when subjected to noise
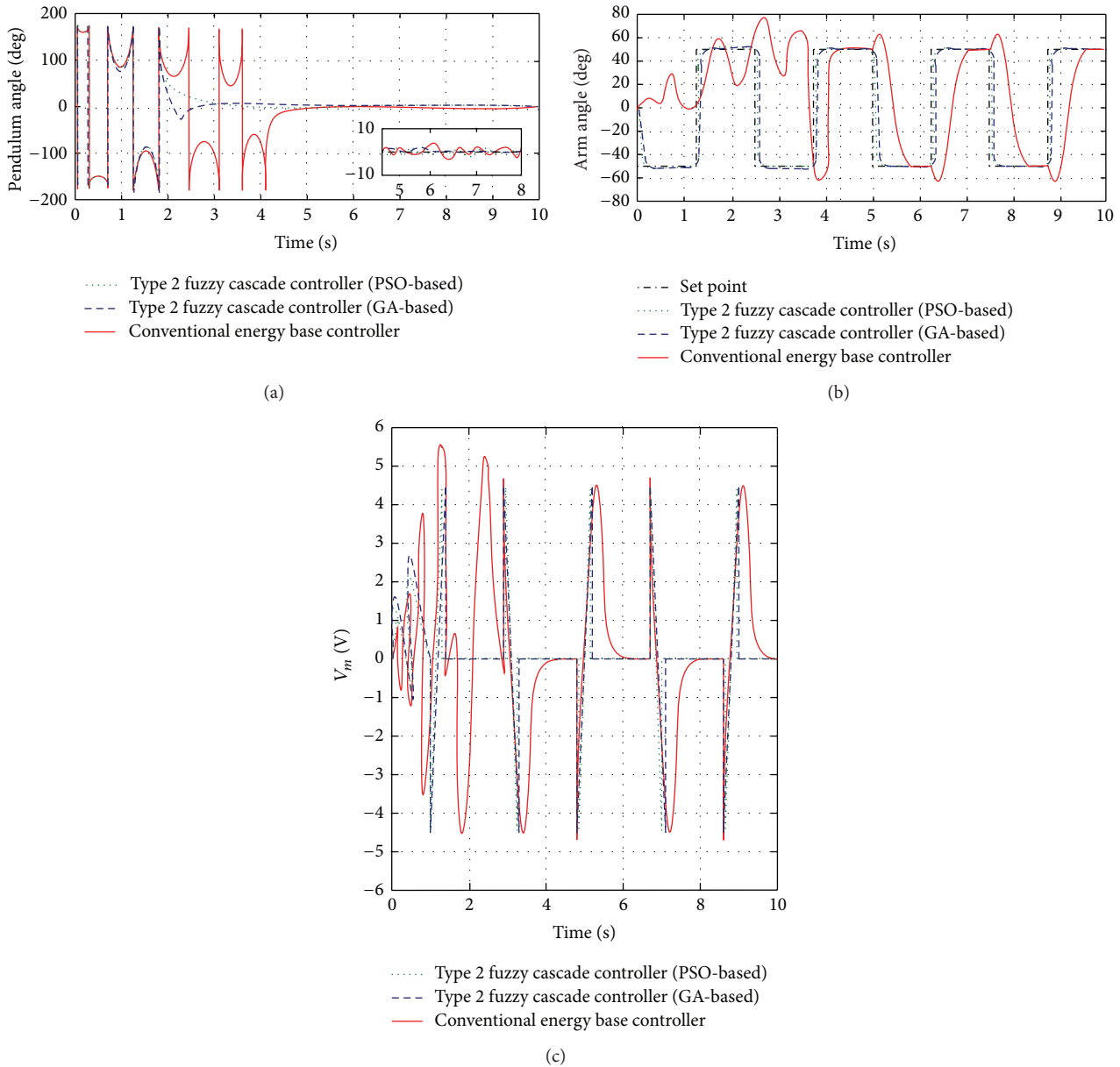
(a)

(b)

(c)

FIGURE 12: Experimental results comparing the conventional energy based controller and IT2F-PD controller (PSO-based and GA-based): (a) pendulum angle, (b) arm angle, and (c) control voltage to the servomotor.

and disturbances; also they outperformed the conventional energy based controller in terms of the performance criteria such as steady state error, settling time, rise time, and maximum overshoot. However, in case of GA and PSO in some performance criteria PSO is better than GA, for example, maximum overshoot (PSO has some improvement over GA, which is between 18.0% and 22.75%), settling time (PSO has lower settling time between 1.31 sec and 2.34 sec), and control energy (PSO requires a low control energy between 17.094 J and 20.995 J). For other performance criteria, GA is better than PSO, for example, steady state error (GA has steady state error between 0.0009 and 0.0028 while PSO has steady state error between 0.0029 and 0.0099). In case

of rise time both GA and PSO have almost the same rise time of 0.0089 sec in all the simulation. In summary, it can be concluded based on the evidence emanated from the experiment results that the GA- and PSO-based controllers have advanced the performance of the conventional energy based controller on the RIP.

The proposed control strategy can be regarded as a promising strategy for controlling different complex systems which are unstable and nonlinear.

General type 2 fuzzy logic controller which has more degree of freedom than interval type 2 fuzzy logic controller is recommended for future work on RIP control. Also, more powerful evolutionary optimization algorithms like hybrid

optimization method are recommended for adjusting the parameters associated with type 2 fuzzy logic controller for RIP.

Conflict of Interests

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

Acknowledgment

References

[1] R. Martínez-Soto, O. Castillo, and L. T. Aguilar, "Type-1 and Type-2 fuzzy logic controller design using a Hybrid PSO-GA optimization method," *Information Sciences*, vol. 285, pp. 35–49, 2014.

[2] J. M. Mendel, H. Hagras, W. Tan, W. W. Melek, and H. Ying, *Introduction to Type-2 Fuzzy Logic Control: Theory and Applications*, John Wiley & Sons, Hoboken, NJ, USA, 2014.

[3] R. A. Krohling, J. P. Rey, R. A. Krohling, and J. P. Rey, "Design of optimal disturbance rejection PID controllers using genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 5, no. 1, pp. 78–82, 2001.

[4] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*, John Wiley & Sons, 1995.

[5] J. Y. Hung, W. Gao, and J. C. Hung, "Variable structure control. A survey," *IEEE Transactions on Industrial Electronics*, vol. 40, no. 1, pp. 2–22, 1993.

[6] Z. Zhang and S. Zhang, "Type-2 fuzzy soft sets and their applications in decision making," *Journal of Applied Mathematics*, vol. 2012, Article ID 608681, 35 pages, 2012.

[7] I. Hassanzadeh and S. Mobayen, "Controller design for rotary inverted pendulum system using evolutionary algorithms," *Mathematical Problems in Engineering*, vol. 2011, Article ID 572424, 17 pages, 2011.

[8] J. M. Mendel, "General type-2 fuzzy logic systems made simple: a tutorial," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 5, pp. 1162–1182, 2014.

[9] F. Herrera, "Genetic fuzzy systems: taxonomy, current research trends and prospects," *Evolutionary Intelligence*, vol. 1, no. 1, pp. 27–46, 2008.

[10] O. Cordón, "A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: designing interpretable genetic fuzzy systems," *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 894–913, 2011.

[11] A. M. El-Nagar, M. El-Bardini, and N. M. El-Rabaie, "Intelligent control for nonlinear inverted pendulum based on interval type-2 fuzzy PD controller," *Alexandria Engineering Journal*, vol. 53, no. 1, pp. 23–32, 2014.

[12] O. Castillo, J. R. Castro, P. Melin, and A. Rodriguez-Diaz, "Universal approximation of a class of interval type-2 fuzzy neural networks in nonlinear identification," *Advances in Fuzzy Systems*, vol. 2013, Article ID 136214, 16 pages, 2013.

[13] S. Zhao and H. Li, "The construction of type-2 fuzzy reasoning relations for type-2 fuzzy logic systems," *Journal of Applied Mathematics*, vol. 2014, Article ID 459508, 13 pages, 2014.

[14] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 6, pp. 808–821, 2006.

[15] O. Castillo, *Type-2 Fuzzy Logic in Intelligent Control Applications*, vol. 272 of *Studies in Fuzziness and Soft Computing*, Springer, Berlin, Germany, 2012.

[16] S.-K. Oh, H.-J. Jang, and W. Pedrycz, "A comparative experimental study of type-1/type-2 fuzzy cascade controller based on genetic algorithms and particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11217–11229, 2011.

[17] C. Aguilar-Avelar and J. Moreno-Valenzuela, "A composite controller for trajectory tracking applied to the Furuta pendulum," *ISA Transactions*, 2015.

[18] S.-E. Oltean, "Swing-up and stabilization of the rotational inverted pendulum using PD and fuzzy-PD controllers," *Procedia Technology*, vol. 12, pp. 57–64, 2014.

[19] H. Qin, W. Shao, and L. Guo, "Research and verification on swing-up control algorithm of rotary inverted pendulum," in *Proceedings of the 26th Chinese Control And Decision Conference (CCDC '14)*, pp. 4941–4945, Changsha, China, May 2014.

[20] V. Nath and R. Mitra, "Swing-up and control of Rotary Inverted Pendulum using pole placement with integrator," in *Proceedings of the Recent Advances in Engineering and Computational Sciences (RAECS '14)*, pp. 1–5, IEEE, Chandigarh, India, March 2014.

[21] P. Faradja, G. Qi, and M. Tatchum, "Sliding mode control of a Rotary Inverted Pendulum using higher order differential observer," in *Proceedings of the 14th International Conference on Control, Automation and Systems (ICCAS '14)*, pp. 1123–1127, October 2014.

[22] K. Y. Chou and Y. P. Chen, "Energy based swing-up controller design using phase plane method for rotary inverted pendulum," in *Proceedings of the 13th International Conference on Control Automation Robotics & Vision (ICARCV '14)*, pp. 975–979, Singapore, December 2014.

[23] Y.-F. Chen and A.-C. Huang, "Adaptive control of rotary inverted pendulum system with time-varying uncertainties," *Nonlinear Dynamics*, vol. 76, no. 1, pp. 95–102, 2014.

[24] S.-K. Oh, H.-J. Jang, and W. Pedrycz, "Optimized fuzzy PD cascade controller: a comparative analysis and design," *Simulation Modelling Practice and Theory*, vol. 19, no. 1, pp. 181–195, 2011.

[25] M. A. Fairus, Z. Mohamed, M. N. Ahmad, and W. S. Loi, "LMI-based multiobjective integral sliding mode control for rotary inverted pendulum system under load variations," *Jurnal Teknologi*, vol. 73, no. 6, 2015.

[26] S.-E. Oltean and A.-V. Duka, "Balance control system using microcontrollers for a rotational inverted pendulum," *Procedia Technology*, vol. 12, pp. 11–19, 2014.

[27] Q. V. Dang, B. Allouche, L. Vermeiren, A. Dequidt, and M. Dambrine, "Design and implementation of a robust fuzzy controller for a rotary inverted pendulum using the Takagi-Sugeno descriptor representation," in *Proceedings of the IEEE Symposium on Computational Intelligence in Control and Automation (CICA '14)*, pp. 1–6, Orlando, Fla, USA, December 2014.

[28] M. U. Hassan, M. B. Kadri, and I. Amin, "Proficiency of fuzzy logic controller for stabilization of rotary inverted pendulum based on LQR mapping," in *Fuzzy Logic and Applications*, vol. 8256 of *Lecture Notes in Computer Science*, pp. 201–211, Springer, 2013.

[29] P. M. Mary and N. S. Marimuthu, "Minimum time swing up and stabilization of rotary inverted pendulum using pulse step

control," *Iranian Journal of Fuzzy Systems*, vol. 6, no. 3, pp. 1–15, 2009.

[30] M. Ramírez-Neria, H. Sira-Ramírez, R. Garrido-Moctezuma, and A. Luviano-Juárez, "Linear active disturbance rejection control of underactuated systems: the case of the Furuta pendulum," *ISA Transactions*, vol. 53, no. 4, pp. 920–928, 2014.

[31] T. C. Kuo, Y. J. Huang, and B. W. Hong, "Adaptive PID with sliding mode control for the rotary inverted pendulum system," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '09)*, pp. 1804–1809, July 2009.

[32] Y. Singh, A. Kumar, and R. Mitra, "Design of ANFIS controller based on fusion function for rotary inverted pendulum," in *Proceedings of the International Conference on Advances in Power Conversion and Energy Technologies (APCET '12)*, pp. 1–5, Mylavaram, India, August 2012.

[33] S.-K. Oh, S.-H. Jung, and W. Pedrycz, "Design of optimized fuzzy cascade controllers by means of Hierarchical Fair Competition-based Genetic Algorithms," *Expert Systems with Applications*, vol. 36, no. 9, pp. 11641–11651, 2009.

[34] P. Civicioglu and E. Besdok, "A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms," *Artificial Intelligence Review*, vol. 39, no. 4, pp. 315–346, 2013.

[35] O. Castillo and P. Melin, "Optimization of type-2 fuzzy systems based on bio-inspired methods: a concise review," *Information Sciences*, vol. 205, pp. 1–19, 2012.

[36] H. Chiroma, S. Abdulkareem, and T. Herawan, "Evolutionary Neural Network model for West Texas Intermediate crude oil price prediction," *Applied Energy*, vol. 142, pp. 266–273, 2015.

[37] N. M. Nawi, A. Khan, M. Rehman, H. Chiroma, and T. Herawan, "Weight optimization in recurrent neural networks with hybrid metaheuristic Cuckoo search techniques for data classification," *Mathematical Problems in Engineering*. In press.

[38] J. Mei, W. Ren, and G. Ma, "Distributed coordinated tracking with a dynamic leader for multiple Euler-Lagrange systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1415–1421, 2011.

[39] J. Lather, S. Dhillon, and S. Marwaha, "Modern control aspects in doubly fed induction generator based power systems: a review," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 2, pp. 2149–2161, 2013.

[40] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[41] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," in *Learning Systems and Intelligent Robots*, pp. 1–10, Springer, 1974.

[42] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*, Springer, Berlin, Germany, 2007.

[43] Y.-T. Kao and E. Zahara, "A hybrid genetic algorithm and particle swarm optimization for multimodal functions," *Applied Soft Computing*, vol. 8, no. 2, pp. 849–857, 2008.

[44] C.-T. Li, C.-H. Lee, F.-Y. Chang, and C.-M. Lin, "An interval type-2 fuzzy system with a species-based hybrid algorithm for nonlinear system control design," *Mathematical Problems in Engineering*, vol. 2014, Article ID 735310, 19 pages, 2014.

[45] C. A. C. Coello, S. Dehuri, and S. Ghosh, *Swarm Intelligence for Multi-Objective Problems in Data Mining*, vol. 242, Springer Science+Business Media, 2009.

[46] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, John Wiley & Sons, New York, NY, USA, 2004.

[47] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*, Springer, 2008.

[48] H. Chiroma, S. Abdulkareem, E. N. Sari et al., "Soft computing approach in modeling energy consumption," in *Computational Science and Its Applications—ICCSA 2014*, vol. 8584 of *Lecture Notes in Computer Science*, pp. 770–782, Springer, 2014.

[49] R. Eberhart and J. Kennedy, "New optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, October 1995.

[50] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the Congress on Congress on Evolutionary Computation*, pp. 81–86, May 2001.

[51] S. Dehuri, S. Ghosh, and C. A. C. Coello, "An introduction to swarm intelligence for multi-objective problems," in *Swarm Intelligence for Multi-objective Problems in Data Mining*, S. Dehuri, S. Ghosh, and C. A. C. Coello, Eds., vol. 242 of *Studies in Computational Intelligence*, pp. 1–17, Springer, Berlin, Germany, 2009.

[52] H. Chiroma, S. Abdulkareem, A. Abubakar, and M. J. Usman, "Computational intelligence techniques with application to crude oil price projection: a literature survey from 2001–2012," *Neural Network World*, vol. 23, no. 6, pp. 523–551, 2013.

[53] P. Melin, D. Sanchez, and L. Cervantes, "Hierarchical genetic algorithms for optimal type-2 fuzzy system design," in *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS '11)*, pp. 1–6, IEEE, El Paso, Tex, USA, March 2011.