Journal of Intelligent & Fuzzy Systems 26 (2014) 1987–1999
DOI:10.3233/IFS-130877
IOS Press

Optimization of a self-tuning PID type fuzzy controller and a PID controller for an inverted pendulum

Ahmadreza Ahmadi, Herlina Abdul Rahim* and Ruzairi Abdul Rahim
Faculty of Electrical Engineering, UniversitiTeknologi Malaysia, Johor, Malaysia

Abstract. By relating to the conventional PID and PID type fuzzy, an optimized PID type fuzzy controller and an optimized PID controller are designed for an inverted pendulum with two outputs. The optimized PID type fuzzy tunes its scaling factors on line. The optimization is done by imperialist competitive algorithm (ICA). The simulation results show a better performance in overshoot, settling time, rise time, integral absolute error, and robustness in compared to the conventional and self-tuning designs.

Keywords: PID type fuzzy controller, imperialist competitive algorithm, inverted pendulum, PID controller

1. Introduction

The inverted pendulum problem is a challenging problem, and well established benchmark in the control area. It is unstable, non minimum phase, and highly non linear. Many efforts have been done to swing up the inverted pendulum, and it is a good example for controlling non linear systems [1–3]. Conventional PID controller is used for different type of inverted pendulums, but since the system is highly non-linear Conventional PID controller cannot meet the whole requirements [4]. PID type fuzzy controller with self-tuning scaling factors was proposed by Zhi-Wei Woo in 2000 [5]. It took advantages of both fuzzy and conventional PID controllers, but proposed method was designed for step response, and by changing the reference there were some defects. M. G*ü*zelkaya at 2003 offered another type of PID type fuzzy which was similar to PID type fuzzy with self-tuning scaling factors offered in 2000, but one of the inputs of fuzzy logic controller was different which was called normalized acceleration [6]. The normalized acceleration gave a more flexibility to the controller and it had effects on the speed of the controller. When the system was slow, fast, or normal, the gains of integral and derivative were different from each other.

Tuning of the controller have been always an important issue for control design and different methods have been implemented to overcome this issue, and it has been always challenging for control engineers to get the optimal results. It is more challenging when it comes to many variables, and trial and error cannot be used. S. Bouallegue used particle swarm optimization method to optimize PID type fuzzy controller at 2011 and compared the results with genetic algorithm optimization method [7]. He showed that particle swarm optimization algorithm worked better than genetic optimization algorithm in terms of efficiency and robustness. One of the latest optimization algorithms is imperialist competitive algorithm proposed by EsmaeilAtashpaz in 2007 [8]. Imperialist competitive algorithm (ICA) is a recent and developed optimization algorithm which

*Corresponding author. Herlina Abdul Rahim, Faculty of Electrical Engineering, UniversitiTeknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia. E-mail: herlina@fke.utm.my.

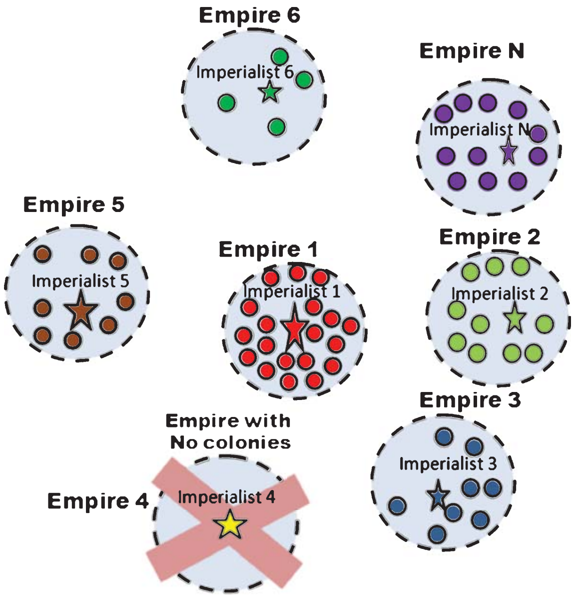

Fig. 1. Empires.

works like other evolutionary algorithms based on initial population, but unlike genetic algorithm that the initial populations are chromosomes the initial populations are countries. Countries are divided to two parts: colonies and imperialists. Empires are formed by colonies and imperialists altogether. Imperialist competition among the empires forms the basis of the imperialist competitive algorithm. There are competitions among imperialists to attract more countries as their colonies. The stronger they are the more colonies they have. During the competitions the weaker imperialist collapse and the stronger ones become stronger, and take positions of the others. The imperialist competition is depicted in Fig. 1 where the red color (Empire 1) shows the strongest empire, and yellow color (Empire 4) shows the weakest empire which is already discarded, and dark green (Empire 6) color shows the empire that is very close to collapse.

Imperialist competitive algorithm has been applied to optimize different cost functions in different systems and compared with other optimization algorithms, and it has been shown that it works better than them in terms of efficiency, accuracy, and especially computational time [9–11]. In this paper, the design of PID type fuzzy controller is based on PID type fuzzy controller with self-tuning scaling factors, but the function for derivative gain is different which can deal with different inputs, and it is not only restricted to the step input. The parameters of the controllers are optimized by the imperialist competitive algorithm, and results are compared with the conventional PID controllers (two PID controllers) and self-tuning controllers (one PID type fuzzy controller via relative rate observer and one optimized PID controller).

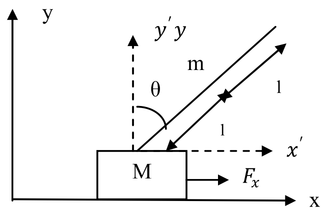

Fig. 2. The inverted pendulum.

2. Structure and model of the inverted pendulum

The inverted pendulum on a pivot driven by a horizontal force is depicted in Fig. 2. The inverted pendulum is controlled by the horizontal force based on the displacement of the pivot. Lagrange's equations have been used in order to model the inverted pendulum. The information about the inverted pendulum's equations can be obtained from [4].

3. Stability analysis of the inverted pendulum

To evaluate the stability analysis of the inverted pendulum, the nonlinear state-space model of the inverted pendulum is linearized using Taylor series. More details about the linearization process can be obtained in [12]. The linearized state-space model is given in Equations (1) and (2).

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -0.98 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 35.93 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ -3.66 \end{bmatrix} F_x \tag{1}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{2}$$

The root locus diagrams of the open-loop model of the inverted pendulum are shown in Figs. 3 and 4 for each

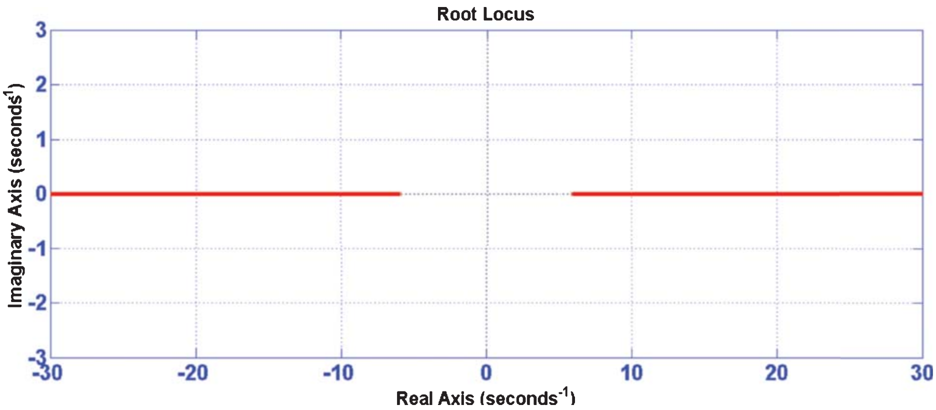

Fig. 3. The root locus diagram for θ.

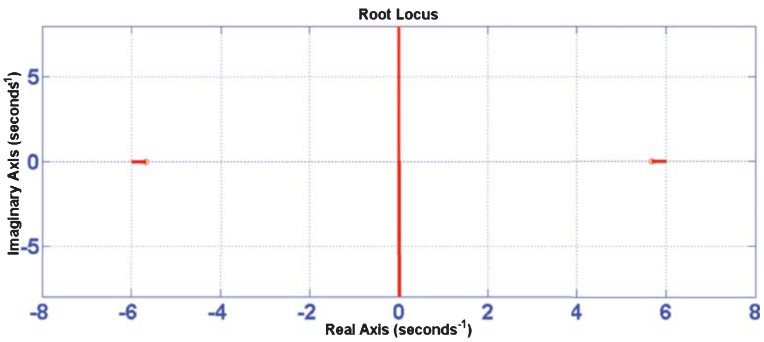

Fig. 4. The root locus diagram for x.

Table 1
A general PID type fuzzy rule base

E/ΔE	NL	NS	NM	ZR	PS	PM	PL
PL	ZR	PS	PM	PL	PL	PL	PL
PM	NS	ZR	PS	PM	PL	PL	PL
PS	NM	NS	ZR	PS	PM	PL	PL
ZR	NL	NM	NS	ZR	PS	PM	PL
NS	NL	NL	NM	NS	ZR	PS	PM
NM	NL	NL	NL	NM	NS	ZR	PS
NL	NL	NL	NL	NL	NM	NS	ZR

output (the angle of the inverted pendulum and displacement of the pivot) separately. As can be seen, the open-loop system is unstable for both outputs.

4. PID type fuzzy control

There are different types of PID type fuzzy controllers. The whole schematic of a PID type fuzzy controller is shown in Fig. 5. The both fuzzy PD and PI are combined to get better performance.

There are two inputs and one output for the fuzzy login control [5]. The inputs are error and the change of the error respectively. Triangular membership functions are used for all inputs and output shown in Fig. 6, and rule table is designed the way to get a linear response shown in Table 1.

The output of the controller is:

$$u_c = \alpha\, u + \beta \int udt = \alpha(\mathrm{A} + \mathrm{P}K_e\mathrm{e} + \mathrm{D}K_d\dot{e}) + \int (\mathrm{A} + \mathrm{P}K_e\mathrm{e} + \mathrm{D}K_d\dot{e})\mathrm{dt} = \alpha\mathrm{A} + \beta\mathrm{At} + (\alpha K_e\mathrm{P} + \beta\, K_d\mathrm{D})e + \beta\, K_e\mathrm{P}\int \mathrm{edt} + \alpha K_d D\dot{e} \tag{3}$$

Form the formula, it can be extracted that control components are:

Proportional: $\alpha K_e\mathrm{P} + \beta\, K_d\mathrm{D}$
Integral: $\beta\, K_e\mathrm{P}$
Derivative: $\alpha K_d D$

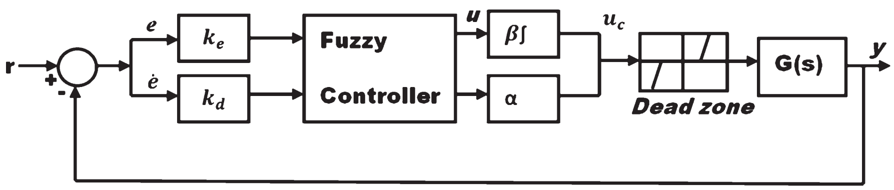

Fig. 5. The PID fuzzy type control system.

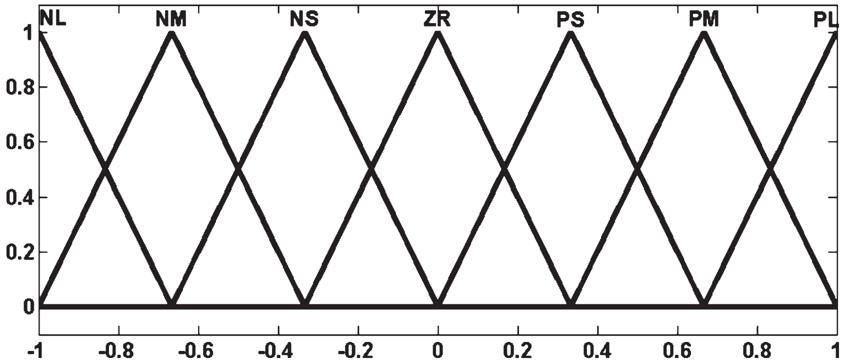

Fig. 6. MF of u, e, Δe.

5. A PID type fuzzy controller with self-tuning scaling factors

In this method, parameter adaptive method is used in order to improve the transient response [5]. Two functions are defined to make the system adaptive:

$$\mathrm{f(e(t))} = \mathrm{a_1} \times \mathrm{abs(e(t))} + \mathrm{a_2} \tag{4}$$

$$\mathrm{g(e(t))} = \mathrm{b_1} \times (1 - \mathrm{abs(e(t))}) + \mathrm{b_2} \tag{5}$$

where, a_1, a_2, b_1, and b_2 are all positive constants. Then the self-tuning scaling factors changing with time are described follows:

$$\beta_s(\mathrm{e(t)}) = \beta \times \mathrm{f(e(t))} \tag{6}$$

$$K_{ds}(\mathrm{e(t)}) = K_d \times \mathrm{g(e(t))} \tag{7}$$

β and K_d are initial scaling factor values. The whole schematic of the controller is depicted in Fig. 7.

The whole concept of the two functions can be explained by the concept of an ideal PID control design. When the error is maximum, it is ideal that the derivative gain is minimum, and the integral gain is maximum, and while the error is minimum, it is ideal that the derivative gain is maximum, and the integral gain is minimum.

6. PID type fuzzy controller via relative rate observer (PTFCRRO)

It was first proposed by M. G*ü*zelkaya in 2003 [6] and has been used in many research works [13–15].

6.1. The concept of normalized acceleration

The normalized acceleration gives "relative rate" information about fastness and slowness of the system response [6]. The normalized acceleration $r_v(k)$ is defined as follows:

$$r_v(k) = \frac{d_e(k) - de(k-1)}{de(.)} = \frac{dde(k)}{de(.)} \tag{8}$$

where, e(k) is the error value de(k) is the incremental change in error which is given by:

$$\mathrm{de(k)} = \mathrm{e(k)} - \mathrm{e(k-1)} \tag{9}$$

and *dde(k)* is called the acceleration in error and it is given by

$$dde(k) = de(k) - de(k-1) \tag{10}$$

de(.) is chosen as follows:

$$de(.) = \begin{cases} de(k) & if \ \ |de(k)| \geq |de(k-1)| \\ de(k-1) & if \ \ |de(k)| < |de(k-1)| \end{cases} \tag{11}$$

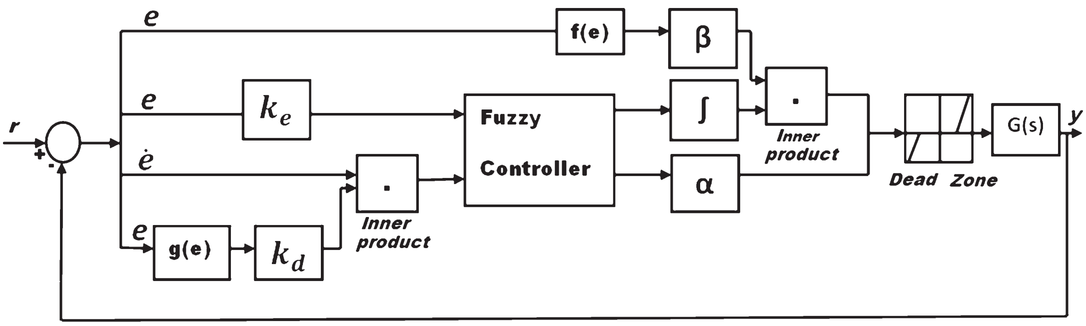

Fig. 7. The PID type fuzzy control system with function tuner.

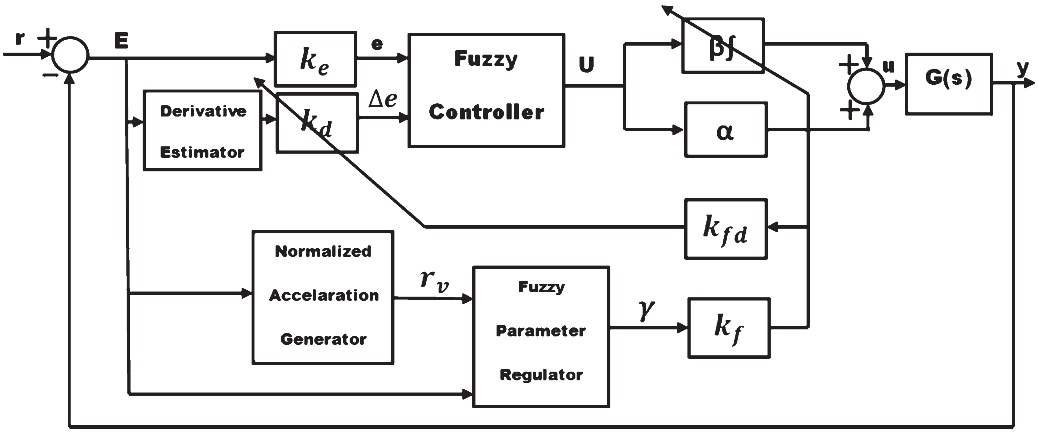

Fig. 8. The closed-loop control structure for fuzzy PID controller using relative rate method.

The normalized acceleration $r_v(k)$ gives us relative-rate information about the system response within a range of $[-1, 1]$. When the system response is very fast, $r_v(k)$ approaches to "1", and when the system response is very slow, it approaches to "-1". For the change of the system response with a constant rate, $r_v(k)$ takes the value of zero.

6.2. *Parameter adaptive method via relative rate observer*

The block diagram of this design (PTFCRRO) is shown in Fig. 8. The output of the fuzzy parameter regulator is designated as γ. The output of the scaling factor k_d is adjusted by multiplying its predetermined value by γ, where as the scaling factor β is adjusted by dividing its predetermined value by the same coefficient factor as it is given below:

$$k_d = k_{ds} k_{fd} k_f \gamma, \quad \beta = \frac{\beta_s}{k_{f\gamma}} \tag{12}$$

It is seen that k_f is the output scaling factor for the fuzzy parameter regulator block and k_{fd} is the additional parameter that affects only the derivative factor of the FLC. More information about the fuzzy controller and fuzzy parameter regulator can be obtained from [6].

7. Proposed method

The previous method is suitable for step reference since g(e(t)) is defined based on abs(1−e(t)). To solve this problem in this paper a new function for g(e(t)) is defined which is:

$$\mathrm{g(e(t))} = \mathrm{b_1} \times \exp(-5 \times \mathrm{e(t)}) + \mathrm{b_2} \tag{13}$$

Based on this design, when the error is maximum, the function is minimum, and when the error is minimum, the function is maximum. The difference between this function with the previous function is that it does not only depend on the specified reference (step function), and it gives good response to different kinds of reference functions. F(e(t)) is the same as previous one.

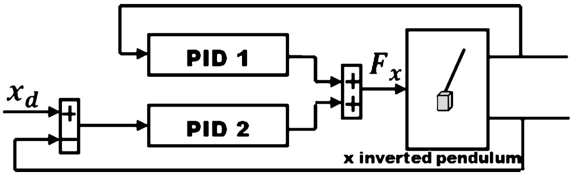

Fig. 9. Two PID controllers design.

Both two outputs of the inverted pendulum are controlled which are the angle of the inverted pendulum, and the displacement of the pivot. For controlling the angle of the pendulum, optimized PID control is used which is tuned by imperialist competitive algorithm, and PID type fuzzy is used to control the displacement of the pivot. Two PID type fuzzy controllers are not used in this paper since the robustness of the system is not as good as when one PID and one PID type fuzzy controllers are used. The results are compared with the design of Jia-Jun Wang who used two conventional PID controllers at 2010 [4]. His design is shown in Fig. 9. The results are also compared with another self-tuning method where an optimized PID controller is used to control the angle and a PID type fuzzy controller using relative rate observer (PTFCRRO) is used to control the displacement having the schematic as Fig. 10, only PTFCRRO is used instead of our proposed PID type fuzzy controller.

The gains of two PIDs are [4]:

PID1: P = 25, I = 15, D = 3; PID2 : P = −2.4, I = −1, D = −0.75,

Where:

M = 1 kg, m = 0.1 kg, l = 0.3 m, g = 9.8 $^{m}/_{s^2}$,

The whole schematic of our design is shown in Fig. 10. All break points of fuzzy logic controller and 11 variables are optimized by imperialist competitive algorithm. Two outputs are used for fuzzy logic controller in order to increase the flexibility of the optimization. Gaussian membership functions are used for all inputs and out outputs and the number of membership functions for each input, and output are 5 which is used based on trial and error. Optimized membership functions are depicted in Figs. 11 and 12, and fuzzy rule base is shown in Table 1 which is based on the whole concept of the general PID fuzzy rule base. The optimized parameters of our design are shown in Table 2. The optimized parameters of PTFCRRO and PID_1 are shown in Tables 3 and 4 respectively. To make a fair comparison between the proposed method and the PID type fuzzy controllers using relative rate method, the same gains are used for PID1 in both designs. The parameters of the PID type fuzzy controllers using relative rate method are obtained using ICA with the same condition of our proposed method. The plot of minimum costs and mean costs are depicted in Figs. 13 and 14 which are for our proposed design and PTFCRRO respectively. The cost function is defined to minimize absolute error, settling time, rise time, and maximum overshoot for both of outputs. As it can be seen optimization is done in 30 decades, and initial population (countries) is 250, and the number of imperialists are 10. The initial value for the angle of the inverted pendulum is 0.5 rad, and for the other state variables, they are zeros.

Table 2
Optimized values for the proposed controller

β	k_d	a_1	a_2	b_1	b_2	α	k_e
−1.03	0.2	5.82	4.95	−1.21	2.35	−4.11	0.26

8. Simulation results

All simulation results were based on the nonlinear model of the inverted pendulum. In Fig. 15a, b, c, the stabilization of one Optimized self-tuning PID type fuzzy controller in addition to one optimized PID controller, two PID controllers which are tuned based on the design of Jia-Jun Wang [4] and one optimized PTFCRRO in addition to one optimized PID controller in absence of the disturbances are given for θ respectively. In Fig. 16a, b, the positions on the pivot, and the control signals are given for three designs respectively. Our proposed design is better than two other designs in terms of the undershoot, overshoot, settling time, rise time for both the angle of the inverted pendulum, and the position of the pivot. However the control signal of the two PID designs is smoother than our proposed design where as the control signal of our proposed design is much smoother than the self-tuning design (one PTFCRRO controller and one optimized PID controller).

The stability analysis of the proposed, two PID, and self-tuning designs are depicted in presence of the disturbances which are $d_1 = d_2 = 20\sin(20\,\pi\, t)$ in Figs. 17–19 respectively. Again, it can be clearly seen that our proposed method is better than other methods in terms of the undershoot, overshoot, settling time, and rise time, but the control effort is less smooth. It shows that our design is good in term of robustness.

From Figs. 20–22, it can be seen that the tracking of our design is better than other controllers in absence of the disturbances, and Figs. 23–25 show the same results in presence of the disturbances.

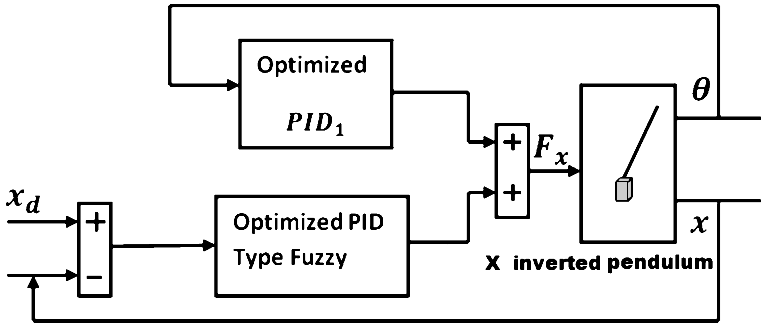

Fig. 10. One PID and one PID type Fuzzy controllers design.

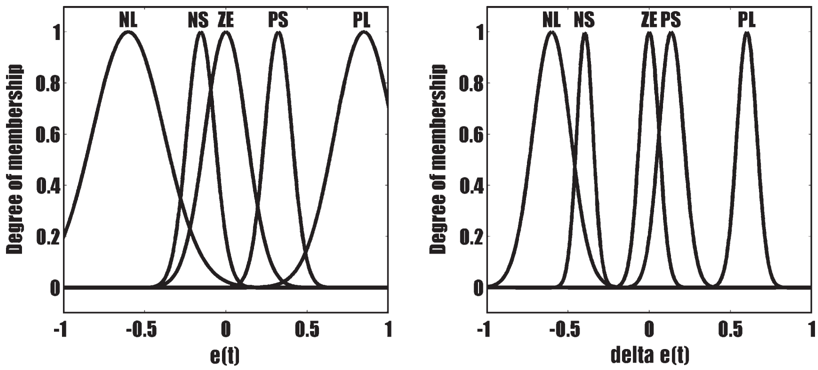

Fig. 11. Optimized input membership functions.

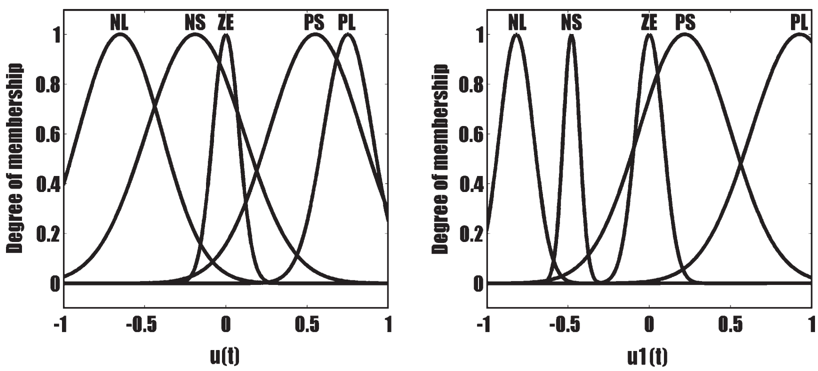

Fig. 12. Optimized output membership functions.

Table 3
Optimized values for the PTFCRRO controller

β_s	k_{ds}	α	k_e	k_f	k_{fd}
−9.85	0.48	10	−1.49	−5.61	0.72

Table 4
Optimized values for PID_1

P_1	I_1	D_1
46.6	32.83	4.28

In Table 5, the integral absolute errors of the angle of the inverted pendulum and the position of the pivot in presence and absence of the disturbances are given for all designs. In the table, "out dist" stands for without disturbances and "with dist" stands for with disturbances. The integral absolute errors of the proposed design, conventional design (Two PID controllers design), and self-tuning design are compared in the table. Results show that the integral absolute errors of our design are better than the conventional and self-tuning designs

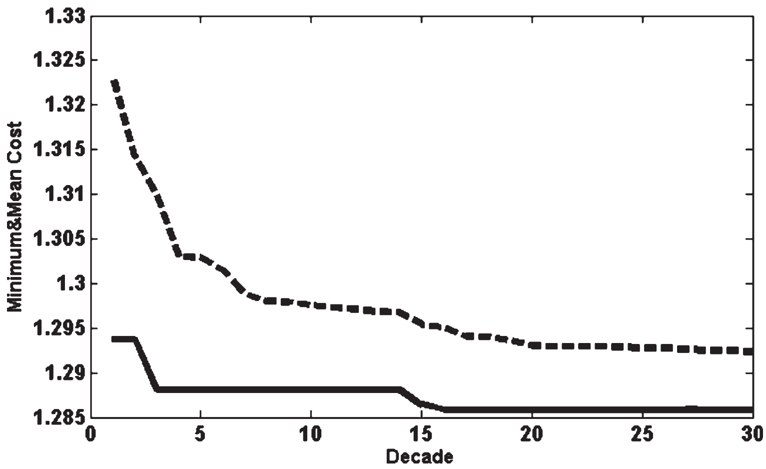

Fig. 13. The minimum cost (—-). The mean cost (—-) for the proposed design.

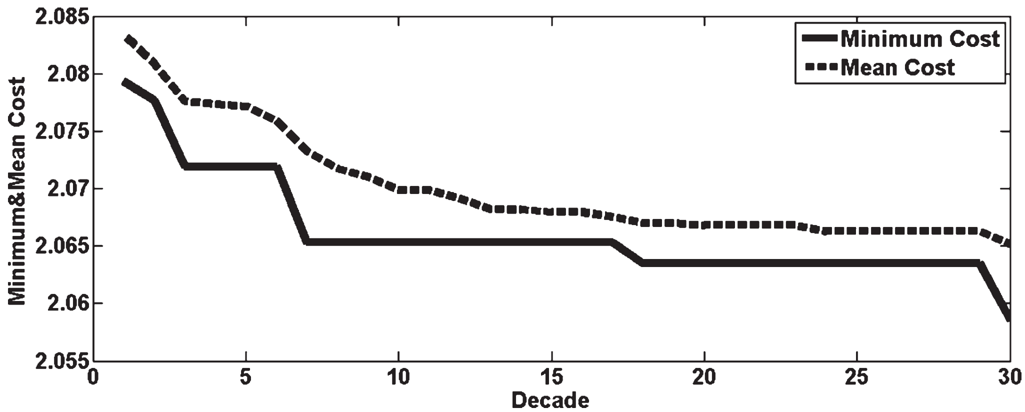

Fig. 14. The minimum cost and mean cost for the self-tuning design.

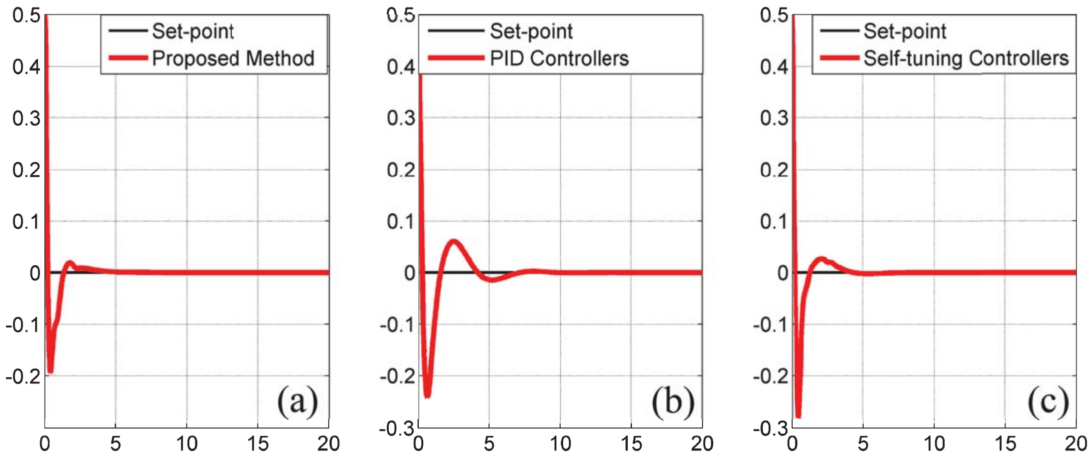

Fig. 15. Stabilization simulations of the (a) proposed design (b) two PID controllers and (c) self-tuning design without disturbances for θ.

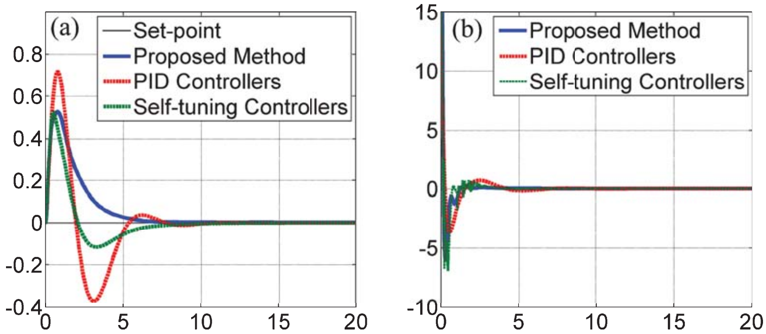

Fig. 16. Stabilization simulations of three designs (a) for x and (b) control signal without disturbances.

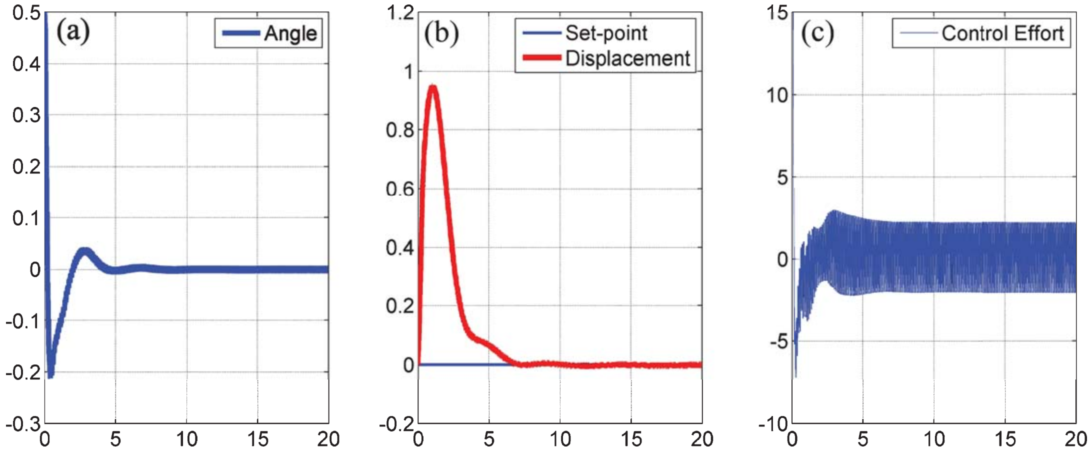

Fig. 17. Stabilization simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal with disturbances for the proposed method.

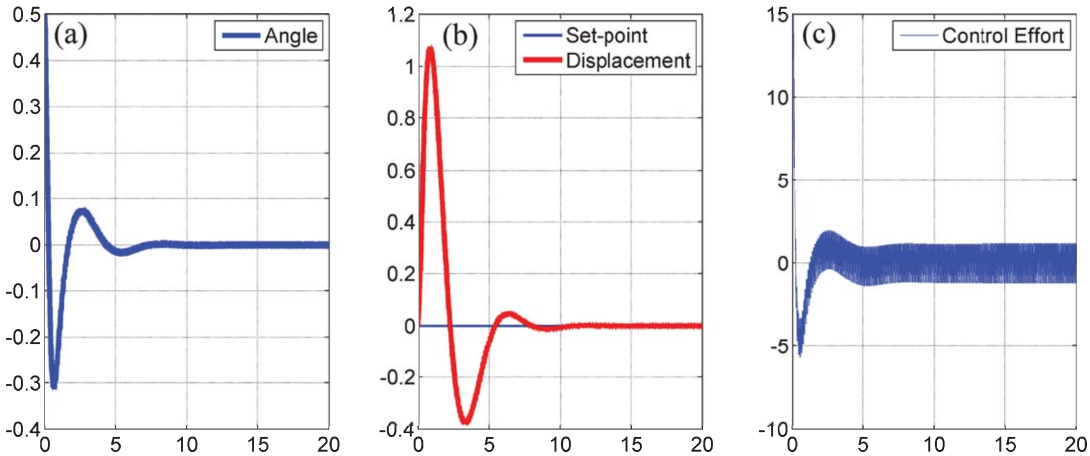

Fig. 18. Stabilization simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal with disturbances for the conventional method.

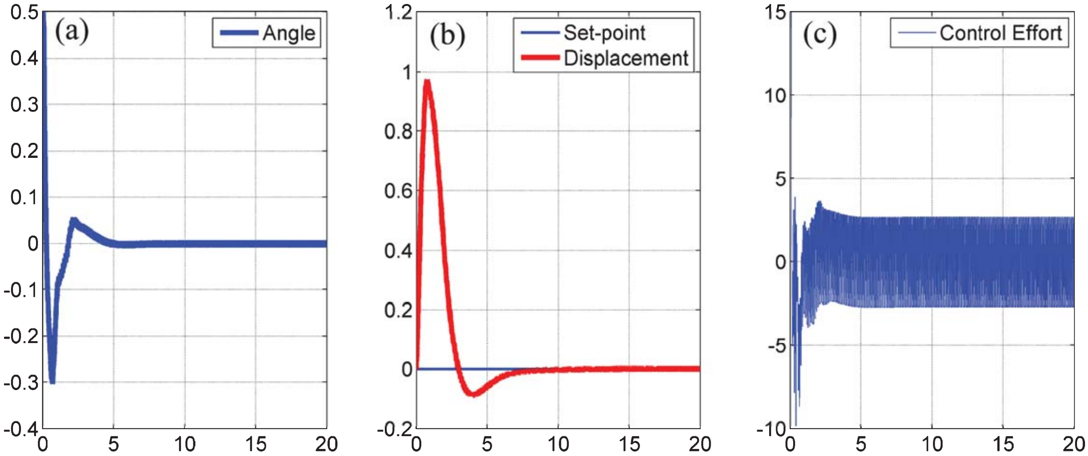

Fig. 19. Stabilization simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal with disturbances for the self-tuning method.

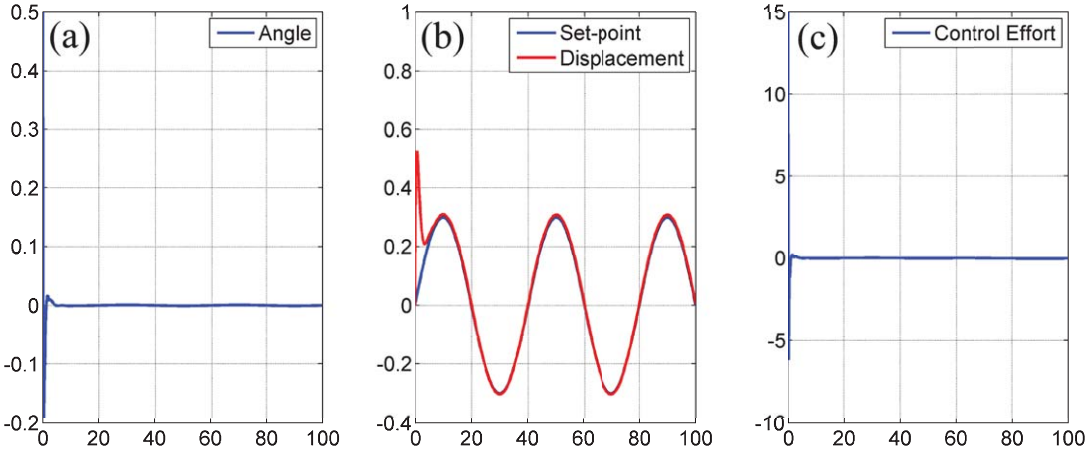

Fig. 20. Tracking simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal without disturbances for the proposed method.

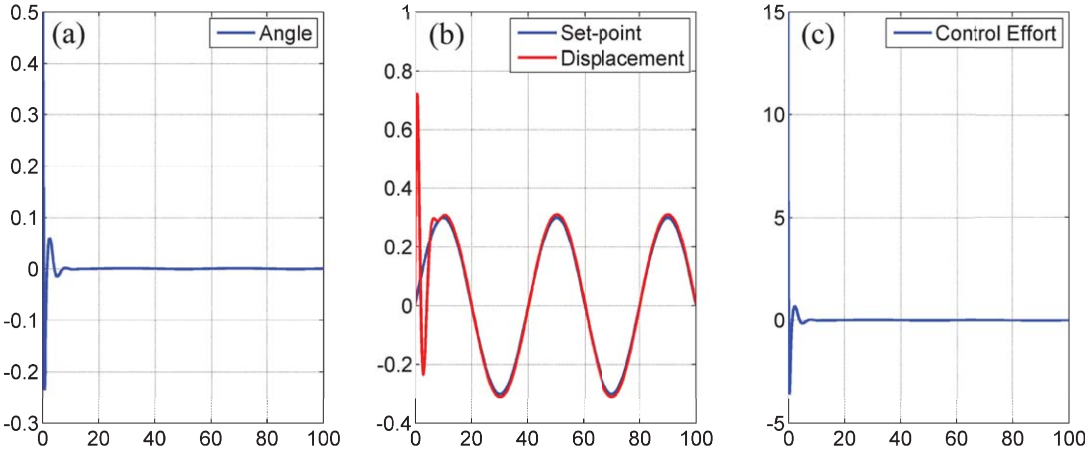

Fig. 21. Tracking simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal without disturbances for the conventional method.

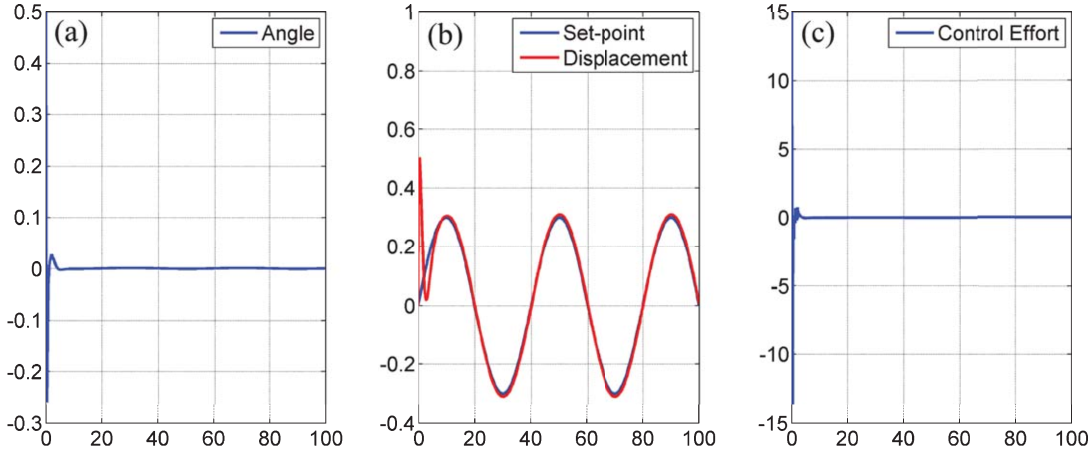

Fig. 22. Tracking simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal without disturbances for the self-tuning method.

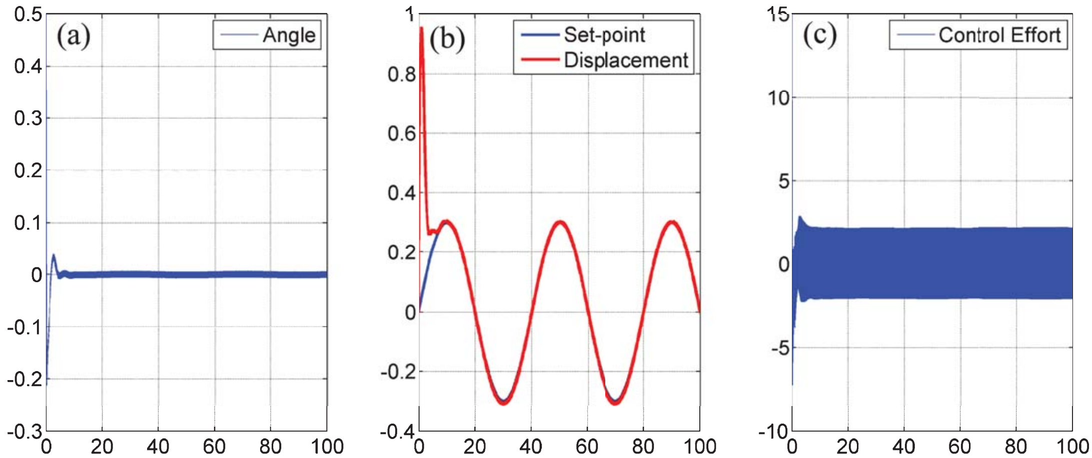

Fig. 23. Tracking simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal with disturbances for the proposed method.

in all cases. It also shows that the self-tuning design has less amount of the integral absolute error than the conventional design in all cases which proves its better performance. In other words, the proposed design is the best design compared to other designs and the self-tuning design comes next.

9. Conclusion

In this paper, an optimized self-tuning PID type fuzzy controller and one optimized PID controller were used for an inverted pendulum with two outputs which were the angle of the inverted pendulum, and the position of the pivot. The optimization process has been done by using imperialist competitive algorithm. The results were compared with two other designs:

1. Conventional PID controllers which were tuned based on Jia-Jun Wang's design [4].
2. Self-tuning controllers (one PID type fuzzy controller via relative rate observer and one optimized PID controller).

It was seen that the proposed method had better results in terms of the overshoot, undershoot, settling time, rise time, and integral absolute error in both cases of presence and absence of the disturbances. The tracking of our controllers had good performances in compared with the conventional and self-tuning controllers in both presence and absence of the disturbances.

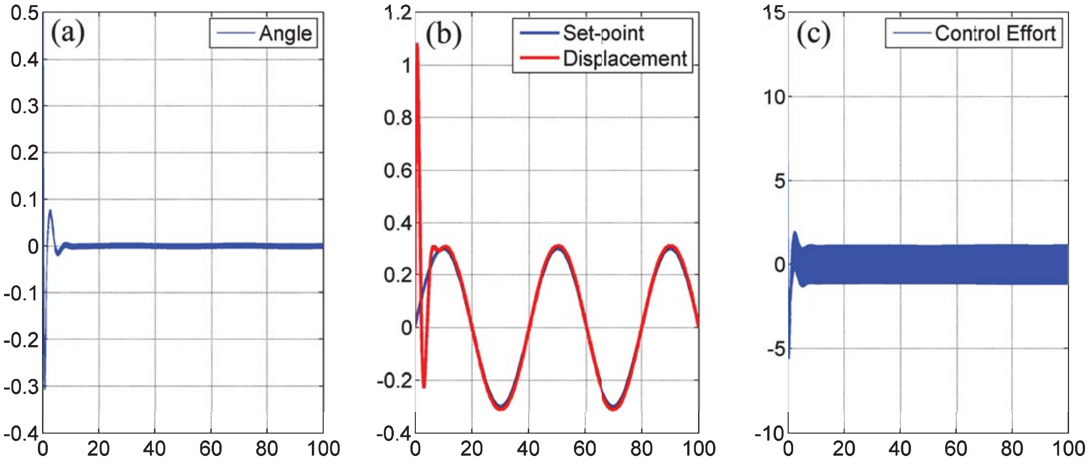

Fig. 24. Tracking simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal with disturbances for the conventional method.

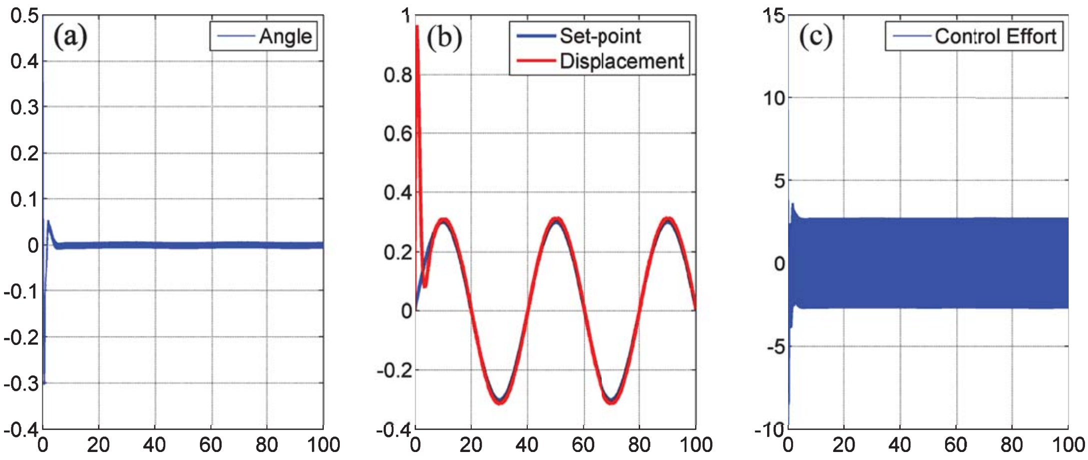

Fig. 25. Tracking simulations of (a) the angle of the inverted pendulum (b) displacement of the pivot and (c) control signal with disturbances for the self-tuning method.

Table 5
Integral absolute error

	IAE_θ				IAE_x			
	Stabilization simulation		Tracking Simulation		Stabilization simulation		Tracking Simulation	
	Out dist	With dist	Out dist	With dist	Out dist	With dist	Out dist	With dist
Proposed design	0.2	0.36	0.24	0.66	1.12	2.14	1.35	2.4
Two PID design	0.38	0.52	0.42	0.8	1.62	2.19	2.22	2.79
Self-tuning design	0.25	0.41	0.29	0.72	1.3	2.16	1.51	2.55

Acknowledgments

This work is partially supported by Ministry of Higher Education Malaysia and UniversitiTeknologi Malaysia under project number R.J130000.7823.4L007.

References

[1] K.J. Åström and K. Furuta, Swinging up a pendulum by energy control, *Automatica* **36** (2000), 287–295.

[2] P. Mason, M. Broucke and B. Piccoli, Time optimal swing-up of the planar pendulum, Automatic Control, *IEEE Transactions on* **53** (2008), 1876–1886.

[3] C. Tao, J.S. Taur, T.W. Hsieh and C. Tsai, Design of a fuzzy controller with fuzzy swing-up and parallel distributed pole assignment schemes for an inverted pendulum and cart system, Control Systems Technology, *IEEE Transactions on* **16** (2008), 1277–1288.

[4] J.-J. Wang, Simulation studies of inverted pendulum based on PID controllers, *Simulation Modelling Practice and Theory* **19** (2011), 440–449.

[5] Z.-W. Woo, H.-Y. Chung and J.-J. Lin, A PID type fuzzy controller with self-tuning scaling factors, *Fuzzy Sets and Systems* **115** (2000), 321–326.

[6] M. Güzelkaya, I. Eksin and E. Yeşil, Self-tuning of PID-type fuzzy logic controller coefficients via relative rate observer, *Engineering Applications of Artificial Intelligence* **16** (2003), 227–236.

[7] S. Bouallègue, J. Haggège, M. Ayadi and M. Benrejeb, PID-type fuzzy logic controller tuning based on particle swarm optimization, *Engineering Applications of Artificial Intelligence* **25** (2012), 484–493.

[8] E. Atashpaz-Gargari and C. Lucas, Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition, Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, IEEE, 2007, pp. 4661–4667.

[9] M. Yousefi, A. Darus and H. Mohammadi, An imperialist competitive algorithm for optimal design of plate-fin heat exchangers, *International Journal of Heat and Mass Transfer* **55** (2012), 3178–3185.

[10] A. Kaveh and S. Talatahari, Optimum design of skeletal structures using imperialist competitive algorithm, *Computers & Structures* **88** (2010), 1220–1229.

[11] C. Lucas, Z. Nasiri-Gheidari and F. Tootoonchian, Application of an imperialist competitive algorithm to the design of a linear induction motor, *Energy Conversion and Management* **51** (2010), 1407–1411.

[12] M. Bugeja, Non-linear swing-up and stabilizing control of an inverted pendulum system, EUROCON 2003. Computer as a Tool. The IEEE Region 8, IEEE, 2003, pp. 437–441.

[13] O. Karasakal, M. Guzelkaya, I. Eksin, E. Yesil and T. Kumbasar, Online tuning of fuzzy PID controllers via rule weighing based on normalized acceleration, *Engineering Applications of Artificial Intelligence* **26** (2013), 184–197.

[14] K.-Y. Chen, P.-C. Tung, M.-T. Tsai and Y.-H. Fan, A self-tuning fuzzy PID-type controller design for unbalance compensation in an active magnetic bearing, *Expert Systems with Applications* **36** (2009), 8560–8570.

[15] O. Karasakal, M. Guzelkaya, I. Eksin and E. Yesil, An error-based on-line rule weight adjustment method for fuzzy PID controllers, *Expert Systems with Applications* **38** (2011), 10124–10132.