Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2011, Article ID 572424, 17 pages
doi:10.1155/2011/572424

Research Article

Controller Design for Rotary Inverted Pendulum System Using Evolutionary Algorithms

Iraj Hassanzadeh[1, 2] and Saleh Mobayen[1, 2]

[1] *Robotics Research Laboratory, Control Engineering Department, Faculty of Electrical & Computer Engineering, University of Tabriz, P.O. Box 5166616471, Tabriz, Iran*
[2] *Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada T6G 2V4*

Correspondence should be addressed to Iraj Hassanzadeh, ihassanz@ualberta.ca

Received 25 December 2010; Revised 4 May 2011; Accepted 5 August 2011

Academic Editor: Peter Wolenski

This paper presents evolutionary approaches for designing rotational inverted pendulum (RIP) controller including genetic algorithms (GA), particle swarm optimization (PSO), and ant colony optimization (ACO) methods. The goal is to balance the pendulum in the inverted position. Simulation and experimental results demonstrate the robustness and effectiveness of the proposed controllers with regard to parameter variations, noise effects, and load disturbances. The proposed methods can be considered as promising ways for control of various similar nonlinear systems.

1. Introduction

During the past decades, many modern control methodologies such as nonlinear control, optimal control, adaptive control, and variable structure control have been widely proposed for control approaches [1–3]. However, these methods are theoretically complex and difficult to implement. Also, proportional-integral-derivative (PID) controller is a well-known method for industrial control processes. The later approach has been broadly employed in industries because of its simple structure and robust performance in a wide range of operating conditions [3]. Unfortunately, it has been difficult to tune up PID controller gains accurately because many industrial plants are often very complex consisting of issues such as higher order, time delays, and nonlinearities [4, 5]. Ziegler and Nichols proposed the first method utilizing the classical tuning rules. Though, it is hard to determine optimal PID controller parameters with Ziegler-Nichols formula in general [1, 3]. To overcome these difficulties, various methods have been proposed. The ability of using numerical methods for efficiently and accurately characterizing the quality of a particular design has excited control engineers to apply stochastic global optimizers.

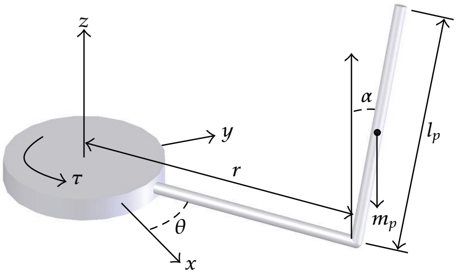


Figure 1: Schematic view of RIP system.

Over the past years, several evolutionary algorithms have been proposed to search for optimal PID controllers. Among them, GA has received great attention and PSO has been successfully applied to various fields [6–10]. Also, ACO is a relatively recent approach to solve optimization problems by simulating the behavior of ant colonies and modeling the behavior of ants, which are known to be able to find the shortest path from their nest to a food source [11]. The ACO method proposed in our paper has the following advantages: applicability to any kind of optimization problems, combinatorial or continuous, easy implementation, high rate of successful optimizations, and low run time.

In this paper, we compare the efficiency of three intelligent algorithms, that is, GA, PSO, and ACO methods. These evolutionary algorithms are used to adjust the PID controller parameters in order to ensure adequate servo and regulatory behavior of the closed-loop system. Also, we formulate the problem of designing PID controllers as an optimization dilemma which adjusts five performance indexes, that is, maximum overshoot, rise time, settling time, and steady-state error of the response and system control energy.

2. Rotational Inverted Pendulum System

The rotational inverted pendulum system is a well-known test platform for evaluating various control algorithms. Also, it has some significant real-life applications such as position control, aerospace vehicles control, and robotics [12, 13]. The system consists of a rotational arm and a pendulum where the rotational arm is actuated by a motor with the objective of balancing the pendulum in the inverted position. A schematic diagram of the RIP system is represented in Figure 1, where u, l_p, m_p, α, r, θ, and J_b are the motor input, the pendulum length, the pendulum mass, the pendulum angle, the arm length, the arm angle and effective mass moment of inertia, respectively.

The plane of the pendulum is orthogonal to the radial arm. Figure 2 indicates the RIP system built in robotics research lab in our department. Also, the block diagram of the whole system is shown in Figure 3.

In this section, the dynamic equations of the RIP system considering backlash and friction effects are presented. The RIP dynamics are governed by [12, 14]

$$\left(a + b\sin^2\alpha\right)\ddot{\theta} + (c\cos\alpha)\ddot{\alpha} - (c\sin\alpha)\dot{\alpha}^2 + (b\sin 2\alpha)\dot{\alpha}\dot{\beta} + f\dot{\theta} + g\,\mathrm{sgn}\left(\dot{\theta}\right) + h\theta = iu, \tag{2.1}$$

$$b\ddot{\alpha} + (c\cos\alpha)\ddot{\theta} - (b\sin\alpha\cos\alpha)\dot{\theta}^2 - d\sin\alpha + e\dot{\alpha} = 0. \tag{2.2}$$

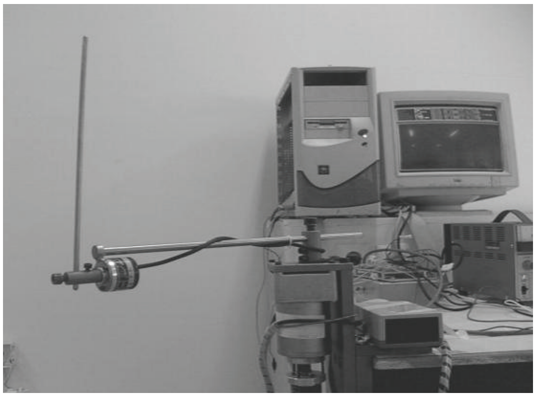


Figure 2: Built in-RIP system (advanced robotics research lab).

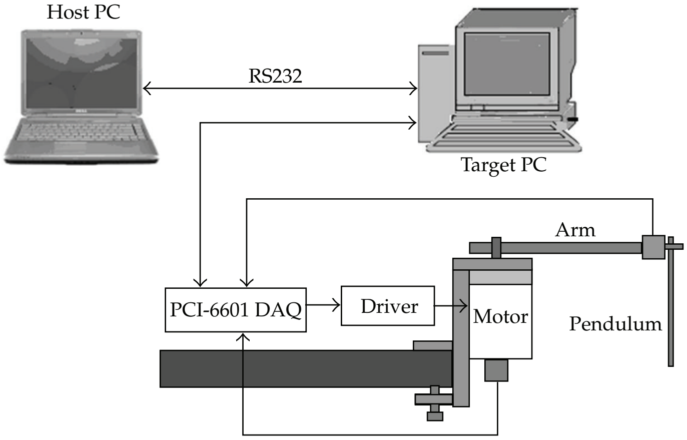


Figure 3: Block diagram of the whole system.

The above nonlinear model can be found in the following equations:

$$\begin{bmatrix} a+b\sin^2\alpha & c\cos\alpha \\ c\cos\alpha & b \end{bmatrix}\begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix}+\begin{bmatrix} f+b(\sin 2\alpha)\dot{\alpha} & -c(\sin\alpha)\dot{\alpha} \\ -b(\sin\alpha\cos\alpha)\dot{\theta} & e \end{bmatrix}\begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix}+\begin{bmatrix} g\,\mathrm{sgn}(\dot{\theta})+h\theta \\ -d\sin\alpha \end{bmatrix}=\begin{bmatrix} iu \\ 0 \end{bmatrix}. \tag{2.3}$$

The parameters of nonlinear model of the system are represented in Table 1.

Using (2.1)–(2.3), the RIP system is easily simulated using Simulink and Matlab. The Simulink block diagram of RIP system shown in Figures 4 and 5 illustrates the step response without controllers indicating the whole system is unstable. The controller parameters generated by heuristic algorithms are employed iteratively in relevant simulation blocks, and the cost function is calculated in the manner presented in the next section.

Table 1: Parameters of the RIP system.

Parameters	Values	Parameters	Values
a	3.29	f	14.283
b	0.1252	g	1.4286
c	0.2369	h	1.72
d	6.052	i	141.32
e	0.0132	w	0.0012

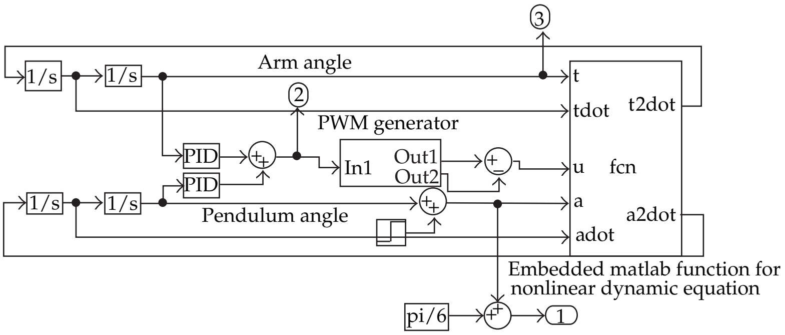


Figure 4: Block diagram of RIP system.

3. GA, PSO, and ACO: An Overview

In the following, brief reviews of GA, PSO and ACO principle are illustrated.

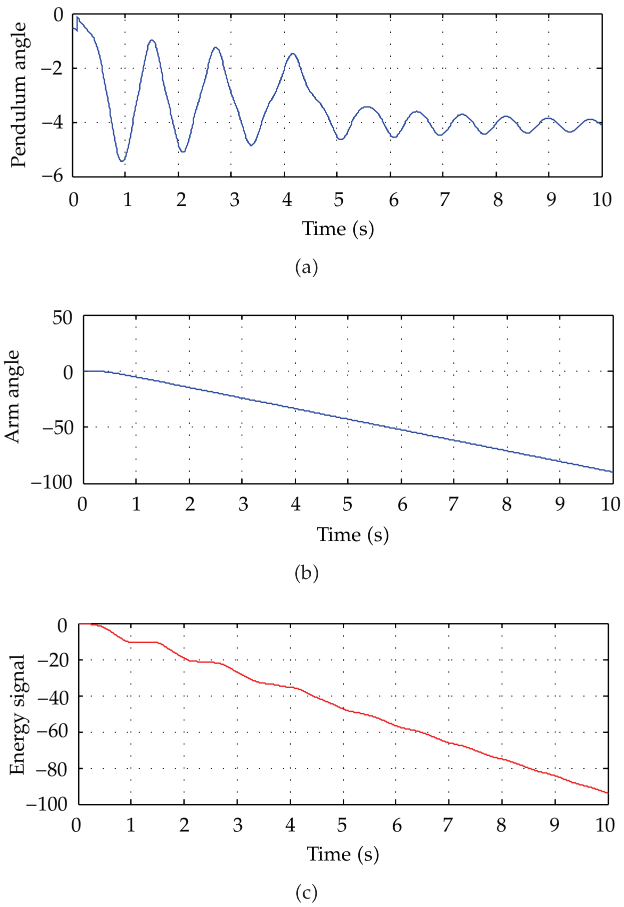
3.1. GA

Considering Darwin's original ideas, life in all its diverse forms is evolved by natural selection and adaptation processes controlled by the survivability of the fittest species. GA is an evolutionary optimizer that takes a sample of possible individuals and employs selection, crossover, and mutation as the primary operators for optimization [15, 16].

3.2. PSO

Considering the social behavior of swarm of fish, bees, and other animals, the concept of the particle swarm optimization (PSO) is developed. The PSO is a robust stochastic evolutionary computation method based on the movement of swarms looking for the most fertile feeding location [16].

From the above statements, it is obvious that the theoretical bases of the two optimization methods rest upon two completely different structures. The GA is based on genetic encoding and natural selection, and the PSO method is based on social swarm behavior. PSO is based on the principle that all solutions can be represented as particles in a swarm. Each particle has a position and velocity vector, and each position coordinate represents a parameter value. Similar to GA, PSO requires a fitness evaluation function that takes the particle's position and assigns a fitness value to it. X_{PB} and X_{GB} are the personal best (P_{best}) position and global best (G_{best}) position of the ith particle. Each particle is initialized



(a)

(b)

(c)

Figure 5: System response without PID controller.

with a random position and velocity. The velocity of each particle is accelerated toward the global best and its own personal best based on the following equation:

$$V_i(\text{new}) = w \times V_i(\text{old}) + c_1 \times \text{rand}() \times (X_{\text{PB}} - X_i) + c_2 \times \text{Rand}() \times (X_{\text{GB}} - X_i). \tag{3.1}$$

Here rand() and Rand() are two random numbers in the range $[0, 1]$, c_1 and c_2 are the acceleration constants, and w is the inertia weight factor. The parameter w helps the particles converge to G_{best}, rather than oscillating around it. Suitable selection of w provides a balance between global and local explorations. In general, w is set according to the following equation [16, 17]:

$$w = 0.5(1 + \text{rand}(0, 1)). \tag{3.2}$$

The positions are updated based on their movement over a discrete time interval (Δt) as follows, with Δt usually set to 1:

$$X_i = X_i + V_i \times \Delta t. \tag{3.3}$$

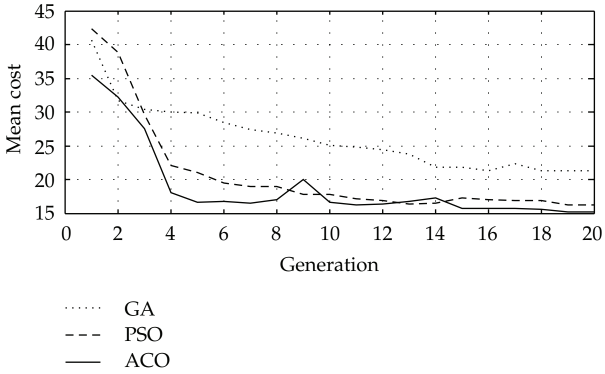


Figure 6: Convergence tendency of mean values of cost function with $\beta = 1$.

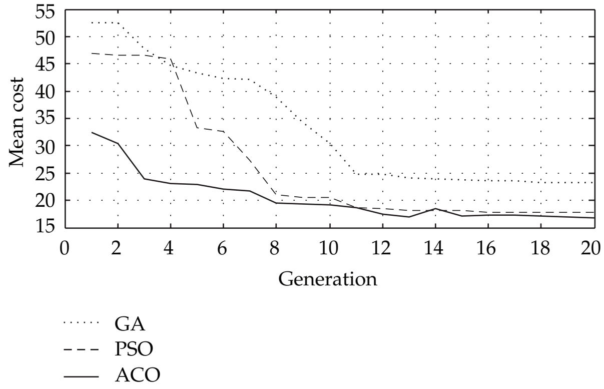


Figure 7: Convergence tendency of mean values of cost function with $\beta = 1.5$.

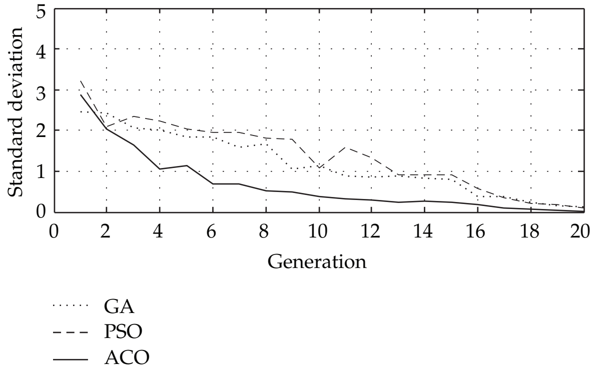


Figure 8: Convergence tendency of standard deviation values of cost function with $\beta = 1$.

Then the fitness at each position is reevaluated. If any fitness is greater than G_{best}, then the new position becomes G_{best} and the particles are accelerated toward that point. If the particle's fitness value is greater than P_{best}, then P_{best} is replaced by the current position.

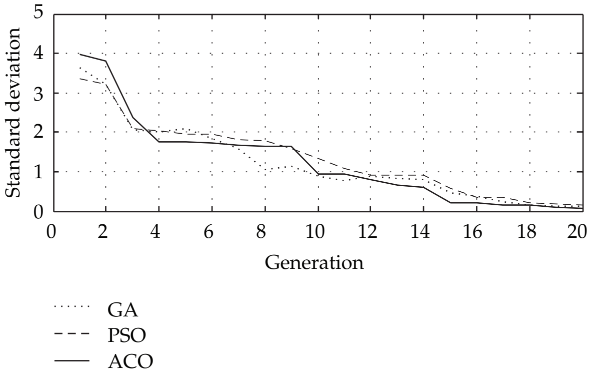


Figure 9: Convergence tendency of standard deviation values of cost function with $\beta = 1.5$.

3.3. ACO

ACO is a relatively recent approach to solve optimization problems by simulating the behavior of ant colonies and modeling the behavior of ants, which are known to be able to find the shortest path from their nest to a food source [18]. ACO is an optimization technique that has been recently developed and recognized as effective for combinatorial optimization problems.

All ants start their tours from source node and end up their tours in destination node. In each node, an ant chooses its path probabilistically, and the probability of choosing an edge is proportional to the pheromone on the edge, that is, roulette wheel selection.

All edges have an initial amount of pheromone, τ_0. After completion of all tours, first pheromone values on all edges are lowered, reflecting evaporation:

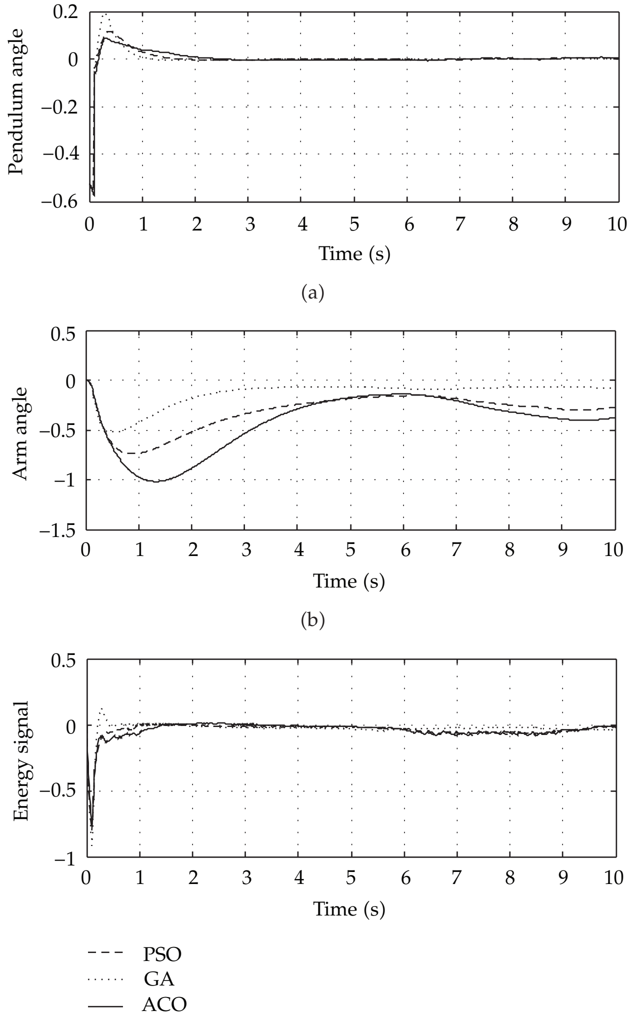
$$\tau_{ij} \longleftarrow (1-\rho)\tau_{ij}, \tag{3.4}$$

where $\rho \in (0,1]$ is the pheromone evaporation rate. Then all ants deposit pheromone on all edges they have crossed in their tours:

$$\begin{gathered} \tau_{ij} \longleftarrow \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}{}^{k}, \\ \Delta\tau_{ij}^{K} = \begin{cases} \dfrac{1}{\text{cost }(k) + c_0} & \text{if edge } (i,j) \text{ belongs to } T^K \\ 0 & \text{otherwise,} \end{cases} \end{gathered} \tag{3.5}$$

where T^K is the tour of the kth ant, cost (k) is its cost, and c_0 is a positive constant which allows to adjust maximum pheromone deposit.

The algorithms parameters have been chosen based on trial and error as follows. For GA method, population size = 50, crossover rate = 0.5, mutation rate = 0.01, and maximum generations = 20; for PSO algorithm, number of particles = 50; acceleration constants $c_1 = c_2 = 1.5$, and maximum iteration = 20; for ACO algorithm, $\rho = 0.4$, $\tau_0 = \tau_{\max} = 10$, $\tau_{\min} = 1$ and $c_0 = 0.5$. Each algorithm is implemented in Matlab. All of the programs are run on a 2.1 GHz Core 2 Duo processor with 2 GB of memory. Each of the optimization methods is tested in 50 independent runs involving 50 different initial trial solutions.



(a)

(b)

(c)

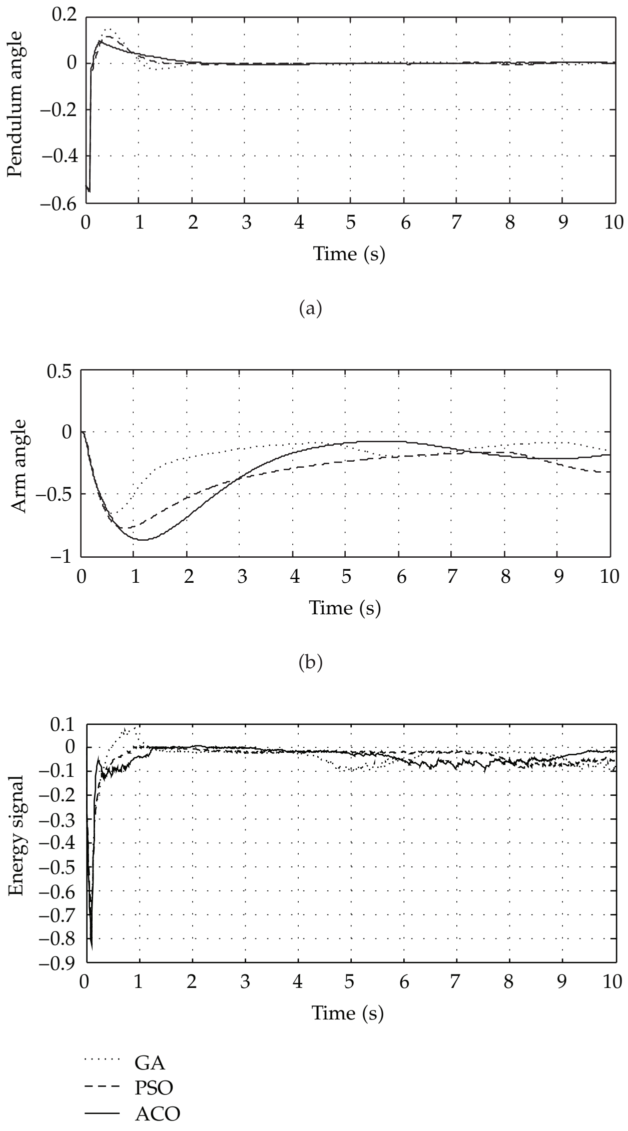
Figure 10: System response: (a) pendulum angle, (b) arm angle, and (c) energy signal, with $\beta = 1$ for $y_r = 0$.

4. Problem Formulation and Controller Design

Considered performance index includes the overshoot M_P, rise time T_r, settling time T_s, steady-state error E_{ss}, and control energy E_u. We find the appropriate parameters for the controllers minimizing the performance indexes. The proposed cost function is considered as follows [3, 19]:

$$\text{cost} = \frac{1 - e^{-\beta}}{2}\left(M_P + E_{ss}\right) + \frac{e^{-\beta}}{2}\left(t_s - t_r\right) + \frac{1}{2}E_u, \tag{4.1}$$

where β is the weighing factor. In this paper, β is set to 1 and 1.5 to investigate different possible solutions. The algorithms stages for searching proper parameters of PID controller are as follows.



(c)

Figure 11: System response: (a) pendulum angle, (b) arm angle, and (c) energy signal, with $\beta = 1.5$ for $y_r = 0$.

First, specify the lower and upper bounds of controller parameters and initialize the particles of the population randomly. Each particle, that is, K (controller parameters) is sent to Matlab Simulink. Then, the values of the performance criteria in the time domain, namely, M_P, T_r, T_s, E_{ss}, and E_u are calculated iteratively. After that, cost function is evaluated for each particle according to these performance criteria. If the cost for local best solution is less than the cost of the current global best solution, the global solution is replaced with local solution. At the end of each iteration, the program checks the stop criterion. If the number of iterations reaches the maximum designated by the user, the latest global best solution is recorded and the algorithm is brought to an end.

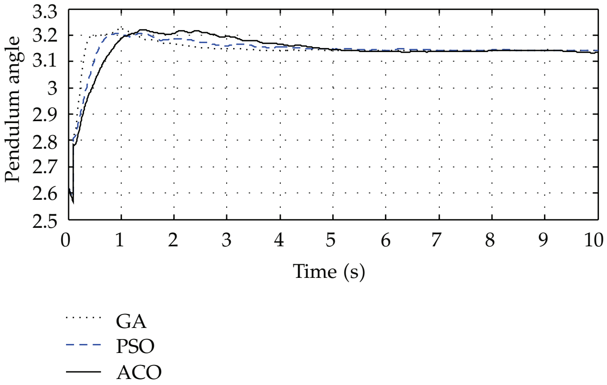


Figure 12: Pendulum angle (servo response) of the RIP control system with $\beta = 1$ for $y_r = \pi$.

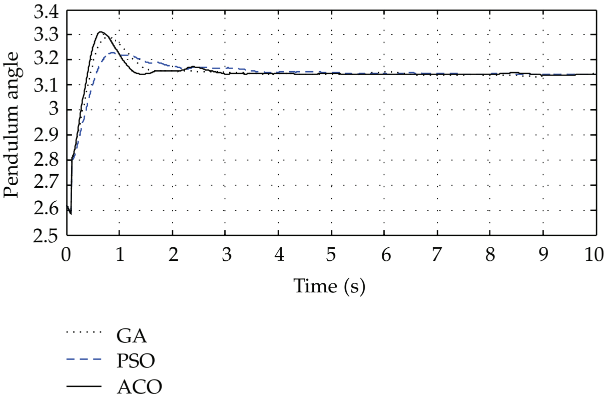


Figure 13: Pendulum angle (servo response) of the RIP control system with $\beta = 1.5$ for $y_r = \pi$.

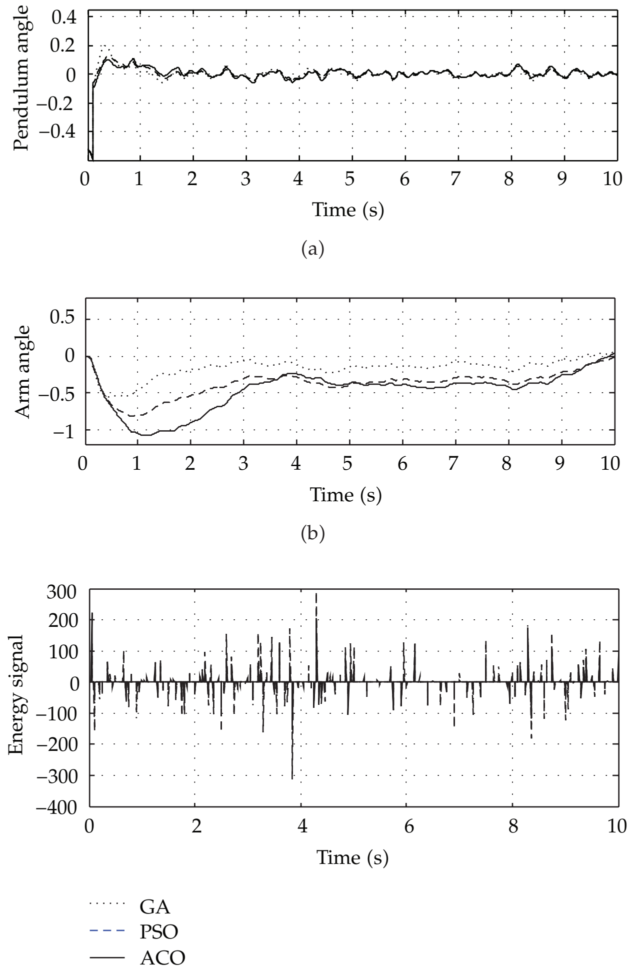
5. Simulation Results

The lower and upper bounds of the three controller parameters are shown in Table 2.

In order to examine the dynamic behaviors and convergence characteristics of the proposed methods, two statistical indexes, namely, the mean value (μ) and the standard deviation (σ) of cost values of all individuals during the computation processes, are used. The mean value displays the accuracy of the algorithm, and the standard deviation measures the convergence speed of the algorithm. The formulas for calculating these values are as follows, respectively [19, 20]:

$$\mu = \frac{\sum_{i=1}^{n} \mathrm{cost}_{P_i}}{n},$$
$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{cost}_{P_i} - \mu\right)^2}, \tag{5.1}$$

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



(a)

(b)

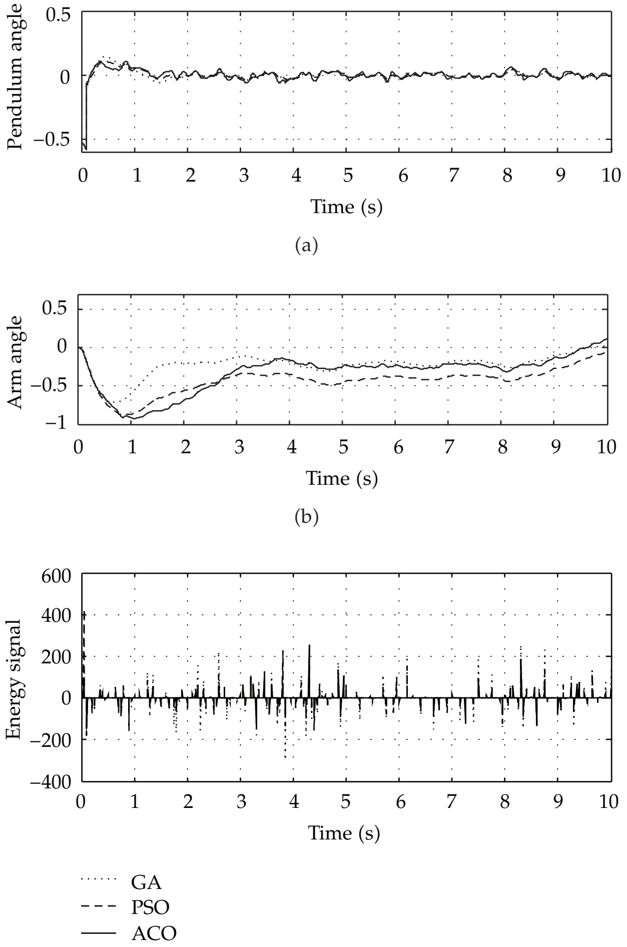
(c)

Figure 14: System response: (a) pendulum angle, (b) arm angle, and (c) energy signal, with $\beta = 1$ subjected to the band-limited white noise (noise power = 0,000523 and sampling time = 0.1 sec) and 10% disturbance.

Table 2: range of three controller parameters.

Controller parameters	Lower bounds	Upper bounds
K_{pi}	0	4
K_{di}	0	4
K_{po}	0	4
K_{do}	0	4

Several simulations are performed to investigate and compare controllers' convergence characteristics. As it can be seen in simulations in Figures 6 and 7, though all controllers can obtain stable mean cost value using the same cost function and simulation conditions, the ACO-PID controller has better cost value and mean value, showing that it can achieve better accuracy. Simultaneously, we can also find from Figures 8 and 9 that the convergence



(c)

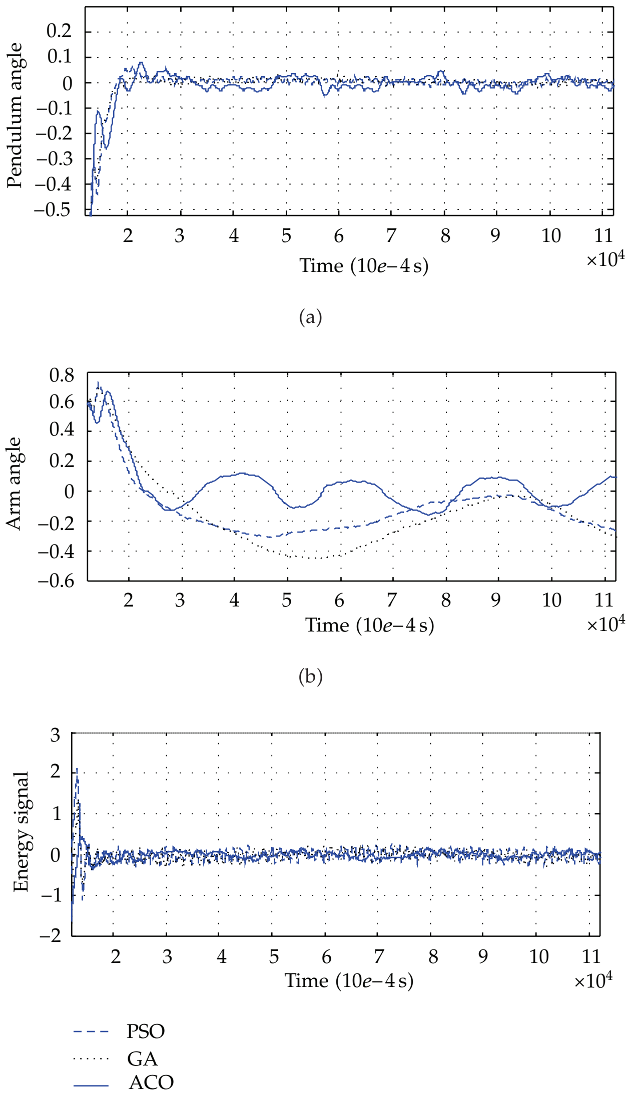
Figure 15: System response: (a) pendulum angle, (b) arm angle, and (c) energy signal, with $\beta = 1.5$ subjected to the band limited white noise (noise power = 0,000523 and sampling time = 0.1 sec) and 10% disturbance.

tendency of the standard deviation of cost values in the ACO-PID controller is much faster than other ones. This can prove that the ACO method has better convergence efficiency.

Also, our simulation results demonstrate that ACO method is faster than GA and slower than PSO and the run time in 20 iterations for ACO is 2408.17 sec in comparison with 1805.42 sec for PSO and 3865.903 sec for GA.

5.1. Servo Behavior

In the following, the optimization procedure has been applied to the RIP system in reference tracking (servo behavior). For analysis of this behavior, the reference signal (y_r) for pendulum angle is first given equal to zero, and then it is determined equal to π. Figures 10 and 11 show the best results of the arm and pendulum angles and control energy value for different values of β. As it can be seen, ACO-based controller makes fine responses, indicating the superiority over other controllers.



(a)

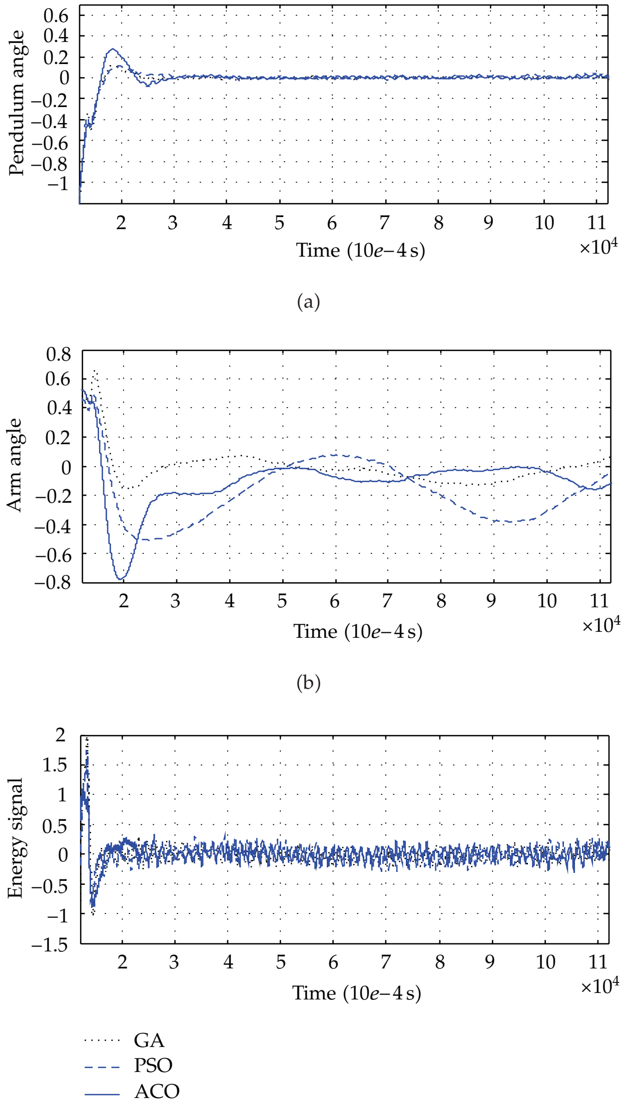
(b)

(c)

Figure 16: Experimental results of system responses with $\beta = 1$: (a) pendulum angles, (b) arm angles, and (c) energy signals.

The simulation results of the best solution for various values of β in 50 runs are summarized in Table 3. As it can be observed from Table 3, the overshoot is decreased using ACO-PID in comparison with GA-PID and PSO-PID. This improvement is 25.693% ($\beta = 1$) and 19.335% ($\beta = 1.5$) in comparison with PSO-PID and 55.071% ($\beta = 1$) and 37.936% ($\beta = 1.5$) in comparison with GA-PID. Also, it is inferred that rise time of the step response in all simulations is quite similar and settling time of step response using GA is better than other controllers. According to total cost value, ACO-PID controlled systems have less cost values.

Now, the servo performance is considered for the reference signal (y_r) of pendulum angle equal to π. Figures 12 and 13 show the pendulum angles (servo response) for $y_r = \pi$ and different values of β.



(a)

(b)

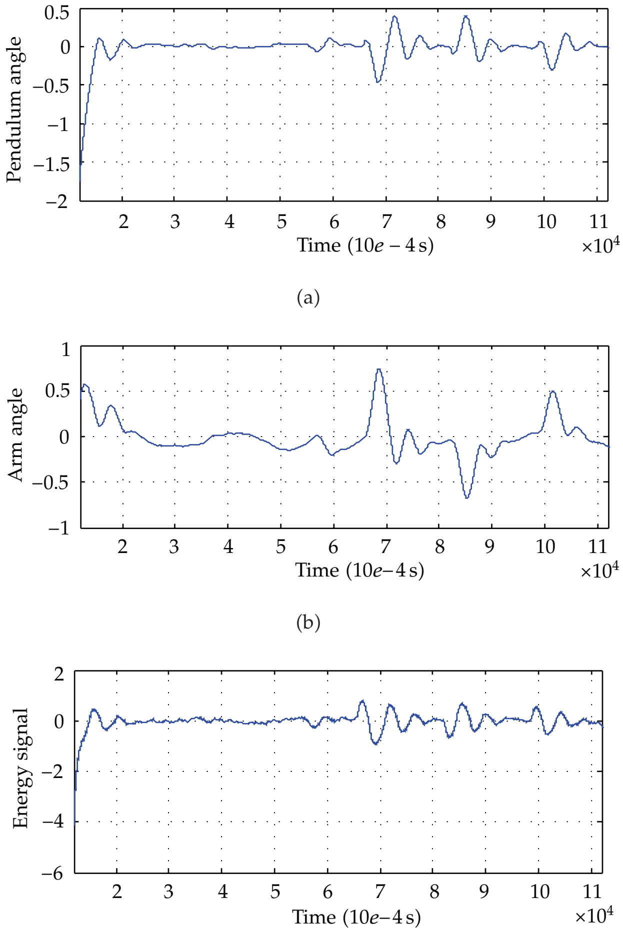
(c)

Figure 17: Experimental results of system responses with $\beta = 1.5$: (a) pendulum angles, (b) arm angles, and (c) energy signals.

5.2. Regulatory Behavior

The control systems are always subject to external disturbances and internal noise which affect the system dynamics. If the nature of the disturbance is identified, it can be modeled mathematically. However, in practice, the nature of the disturbances is not clear and we may not be able to simulate them easily. In RIP system, the disturbance and noise effects can be applied by adding an additional load to the end of pendulum and adding noise to the position sensor, respectively.

The simulations are done subject to a band-limited white noise (noise power = 0,000523 and sampling time = 0.1 sec) and 10% parameter value changes. Simulation results



(a)

(b)

(c)

Figure 18: Experimental results of system responses using ACO-PID by adding a body mass to the end of pendulum and in the presence of disturbances: (a) pendulum angle, (b) arm angle, and (c) energy signal.

shown in Figures 14 and 15 illustrate the robustness and effectiveness of the proposed controllers subject to the noise and disturbance.

Servo and regulatory results motivate to consider the proposed procedure as a suitable tool for controller parameters design and also stimulate investigating the possibility of further research on design and development of other practical control systems.

6. Experimental Results

We have performed experiments on the RIP system set at the University of Tabriz in the robotics research lab. The applied card in this project is PCI-6602 which creates the connection between computer and system and has A/D and D/A converters. Also the arm and pendulum links angles are measured using two E40S Autonics company encoders. The experimental results of the proposed methods on RIP system are shown in Figures 16 and 17. In these figures, the time interval $[0, 1.2]$ which belongs to the swing-up period is deleted

Table 3: Best results of PID controllers with different β values gained by PSO, GA, and ACO algorithms in 50 runs.

	β	K_{dp}	K_{pp}	K_{da}	K_{pa}	$M_p\%$	t_r(sec)	t_s(sec)	t_p(sec)	E_{ss}	E_u	cost
GA	1	0.473	4.118	0.51	0.356	36.787	0.008	0.958	0.29	0.0008	18.213	20.908
	1.5	0.666	3.276	0.444	0.248	28.34	0.008	1.776	0.44	0.0046	22.624	22.519
PSO	1	0.551	3.989	0.398	0.153	22.243	0.008	1.301	0.38	0.001	17.839	16.189
	1.5	0.529	3.206	0.35	0.119	21.805	0.006	1.24	0.41	0.0008	18.242	17.728
ACO	1	0.35	4.091	0.281	0.125	16.528	0.035	1.864	0.32	0.0022	19.904	15.513
	1.5	0.539	2.821	0.441	0.23	17.57	0.008	1.721	0.28	0.0022	19.095	16.571

to focus on controllers' performance in the balance mode. The experimental results are very consistent with the simulation results shown in Figures 10 and 11 which not only prove the performance of proposed methods but also verify the availability of the system model. The fact that the simulated and real controlled responses are practically identical validates the identified system model.

In order to study the stability of designed control system using ACO algorithm subject to parameters variations, we perform the following experiment. In this experiment, the adding effect of a 25 g body mass with the length of 13 cm and in the presence of disturbances is validated. Figure 18 shows the results of this practical test. Then, the proposed method is a very powerful technique in completely eliminating the effects of the disturbances and providing satisfactory servo behavior.

7. Conclusion

In this paper, we present three evolutionary algorithms for designing of intelligent controllers of the RIP system. Each of the algorithms is tested in 50 independent runs involving 50 different initial solutions. The rotational inverted pendulum system is considered as a case study. Through the simulation results, the proposed controllers perform efficient search for proper PID parameters. To evaluate the controller performance, we tested the ability of the closed-loop system to follow set point changes (servo behavior) and the ability of the closed-loop system to reject disturbances (regulatory behavior). The work demonstrates that all methods can solve searching and tuning the controller parameters efficiently. The proposed methods could be considered as promising ways for nonlinear control systems in general. One of the important features of the system is using of xPC-Target toolbox and input-output card in Simulink environment which utilizes hardware in the loop (HIL), telelab implementation and fast-prototyping properties. The topic of our future researches is to employ other cognitive methods in order to achieve better results for designing controller and improving the performance in real time. Also, implementation, of heuristic algorithms for designing adaptive controllers will be our future challenging task. Furthermore, teleoperation control of RIP system using haptic device would be another challenge.

Acknowledgments

This paper was financially supported by the Research Affairs at University of Tabriz. The authors would like to appreciate Mr. Abbas Harifi for his guidance and assistance.

References

[1] N. B. Nichols and J. G. Ziegler, “Optimum settings for automatic controllers,” *ASME Transactions*, vol. 64, pp. 759–768, 1942.

[2] 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.

[3] Z. L. Gaing, “A particle swarm optimization approach for optimum design of PID controller in AVR system,” *IEEE Transactions on Energy Conversion*, vol. 19, no. 2, pp. 384–391, 2004.

[4] A. Visioli, “Tuning of PID controllers with fuzzy logic,” *IEE Proceedings: Control Theory and Applications*, vol. 148, no. 1, p. 18, 2001.

[5] G. Zhou and J. D. Birdwell, “Fuzzy logic-based PID autotuner design using simulated annealing,” in *Proceedings of the IEEE/IFAC Joint Symposium on Computer-Aided Control System Design*, pp. 67–72, Tucson, Ariz, USA, March 1994.

[6] D. H. Kim, “Hybrid GA-BF based intelligent PID controller tuning for AVR system,” *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 11–22, 2011.

[7] M. W. Iruthayarajan and S. Baskar, “Evolutionary algorithms based design of multivariable PID controller,” *Expert Systems with Applications*, vol. 36, no. 5, pp. 9159–9167, 2009.

[8] Y. Y. Fu, C. J. Wu, T. L. Chien, and C. N. Ko, “Integration of PSO and GA for optimum design of fuzzy PID controllers in a pendubot system,” *Artificial Life and Robotics*, vol. 13, no. 1, pp. 223–227, 2008.

[9] V. Mukherjee and S. P. Ghoshal, “Intelligent particle swarm optimized fuzzy PID controller for AVR system,” *Electric Power Systems Research*, vol. 77, no. 12, pp. 1689–1698, 2007.

[10] W. D. Chang, “PID control for chaotic synchronization using particle swarm optimization,” *Chaos, Solitons and Fractals*, vol. 39, no. 2, pp. 910–917, 2009.

[11] L. D. S. Coelho and D. L. D. A. Bernert, “A modified ant colony optimization algorithm based on differential evolution for chaotic synchronization,” *Expert Systems with Applications*, vol. 37, no. 6, pp. 4198–4203, 2010.

[12] 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.

[13] H. Mei and Z. He, “Study on stability control for single link rotary inverted pendulum,” in *Proceedings of the International Conference on Mechanic Automation and Control Engineering (MACE '10)*, pp. 6127–6130, Wuhan, China, June 2010.

[14] Q. Yan, “Output tracking of under-graduated rotary inverted pendulum by nonlinear controller,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 3, pp. 2395–2400, Maui, Hawaii, USA, 2003.

[15] Y. Rahmat-Samii, “Genetic algorithm and particle swarm optimization in engineering electromagnetics,” in *Proceedings of the 17th International Conference on Applied Electromagnetics and Communications*, Dubrovnik, Croatia, 2003.

[16] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of the International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, WA , Australia, November 1995.

[17] I. Hassanzadeh and S. Mobayen, “PSO-based controller design for rotary inverted pendulum system,” *Journal of Applied Sciences*, vol. 8, no. 16, pp. 2907–2912, 2008.

[18] M. Dorigo and T. Stützle, *Ant Colony Optimization*, MIT Press, Cambridge, 2004.

[19] S. Mobayen, *Identification and control of rotational inverted pendulum using intelligent methods*, M.S. thesis, University of Tabriz, Tabriz, Iran, 2008.

[20] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1998.