Fuzzy PD Cascade Controller Design for Ball and Beam System Based on an Improved ARO Technique

Mojtaba Kazemi[a,*], Jalal Najafi[a], Mohamad Bagher Menhaj[b]

[a]*Faculty of Electrical and Computer Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran,*
[b]*Department of Electrical Engineering, Amirkabir University, Tehran, Iran*

Abstract

The ball and beam system is one of the most popular laboratory setups for control education. In this paper, we design a fuzzy PD cascade controller for a ball and beam system using Asexual Reproduction Optimization (ARO) technique. The ball & beam system consists of a servo motor, a grooved beam, and a rolling ball. This system utilizes a servo motor to control ball's position on the beam. Changing the angle of servo motor results in the movement of the beam and, subsequently, the ball rolling on it. We designed a fuzzy PD cascade and a PD cascade controller scheme which consists of the two controller loops. The first (outer) controller and the second (inner) controller are organized in a cascaded construction.

Keywords: Ball and Beam, Fuzzy PD, ARO Optimization, PD

1. Introduction

Ball and beam system is available in most control laboratories due to its simple mechanical structure and inexpensive implementation. This system is inherently nonlinear and unstable which makes it suitable for testing various control techniques. It is also used to provide balance for such systems as mobile robots and space crafts and to control position in aerospace engineering[1].

Ball and beam system utilizes a servo motor to control ball's position on the beam. Changing the angle of servo motor results in movement of the beam and, subsequently, the ball on it. When the ball reaches the desired position, the beam is stabilized in horizontal state. Ball and beam system in open loop form is an unstable system. To balance the beam and positioning the ball in the desired position, closed loop control is needed. For this reason, different control techniques have been applied that include LQR[2, 3], LQG, GPC[4], neural networks and PD cascade[5, 6].

One of the best control schemes implemented for ball and beam system is fuzzy PD control. Fuzzy PD control, thanks to the derivative term, provides a faster response to variations[7]. On the other hand, in fuzzy logic the system behaviour is characterized using human knowledge which directly leads to the design of control algorithm on the basis of fuzzy rules. These rules are in terms of the relationship of inputs to their corresponding outputs, and precisely determine the controller parameters. Any adjustment or debugging only requires modification in these fuzzy rules instead of the redesigning the controller. Hence, the control technique based on fuzzy logic not only simplifies the design but also reduces the monotonous task of solving complex mathematical equations for nonlinear systems. As a result, fuzzy logic controller delivers a better performance in cases where the conventional controller does not cope well with the non-linearity of a process under control.

One of the challenges to design controller is adjusting its parameters. In the literature, various algorithms such as genetic, PSO and ant colony are used to adjust fuzzy PD parameters[8]. In this paper, parameters are optimized using a novel technique called ARO[9, 10].

Genetic algorithm (GA) is the most popular intelligent algorithm employed in finding optimal values of various problems. But GA is a population-based method and has a slow convergence rate. Mansouri et al.[9]introduced an

Fig. 1. Setup of ball and beam system[1]

* Corresponding author. Email: m.kazemi@qiau.ac.ir

individual based algorithm (ARO) which intelligently guides the search process and can reach the global optimum in an astonishing time possessing advantages of both population and individual based algorithms.

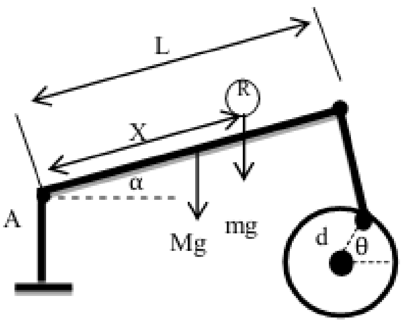

Fig. 2. Ball andBeamdiagram[7].

2. Modelling

The experimental setup of the ball and beam system is presented in Fig. 1. In this model, the aim is to place the ball in the desired position on the beam[1].

Changing the angle of motor causes the beam to move and place the ball in the desired position.

To simulate and test the proposed controller, dynamic model of the system is required which is obtained using the diagram shown in Fig. 2. If friction and other disturbances are not considered, Langrian method can be used to develop system dynamics. Point A is considered as zero reference[7].

The kinetic energy of the system is[1, 3]

$$T = T_1 + T_2 \tag{1}$$

where1 and 2 are kinetic energies of the beam and ball, respectively.

$$T_1 = \frac{1}{2} J_1 \dot{\alpha}^2 \tag{2}$$

$$T_2 = \frac{1}{2}(mx^2)\dot{\alpha}^2 + \frac{1}{2} m\dot{x}^2 + \frac{1}{2} J_2 \omega_2^2 \tag{3}$$

where J_2 is the moment of inertia of the ball, $\dot{r}$ and ω_2 are radial and rotational velocities of the ball and m is its mass.

$$J_1 = mx^2, J_2 = \frac{2}{5} mR^2, \dot{x} = R\omega_2 \tag{4}$$

$$T = \frac{1}{2}\left[\left(J_1 + mx^2\right)\dot{\alpha}^2 + \frac{7}{5} m\dot{x}^2\right] \tag{5}$$

$$P = mgx \sin\alpha + Mg\frac{L}{2}\sin\alpha \tag{6}$$

The Lagrange equation is

$$L = T - P = \frac{1}{2}\left[\left(J_1 + mx^2\right)\dot{\alpha}^2 + \frac{7}{5} m\dot{x}^2\right] - \left(mgx + Mg\frac{L}{2}\right)\sin\alpha \tag{7}$$

Because no external force is applied to the ball in radial direction, Lagrange equations become as (8)

$$\begin{aligned} &\frac{d}{dt}\left[\frac{\partial L}{\partial \dot{\alpha}}\right] - \frac{\partial L}{\partial \alpha} = \tau \\ &\frac{d}{dt}\left[\frac{\partial L}{\partial \dot{x}}\right] - \frac{\partial L}{\partial x} = 0 \end{aligned} \tag{8}$$

$$\begin{aligned} &\left(J_1 + mx^2\right)\ddot{\alpha} + 2mx\dot{x}\dot{\alpha} + \left(mgx + \frac{L}{2}Mg\right)\cos\alpha = \tau \\ &\frac{7}{5}\ddot{x} - x\dot{\alpha}^2 + g\sin\alpha = 0 \end{aligned} \tag{9}$$

When the system is near its steady state, $\dot{\alpha} \approx 0$:

$$\ddot{x} = \frac{5}{7} g \sin\alpha \tag{10}$$

For small angles, $\sin\alpha = \alpha$, and (1) becomes:

$$\frac{X(s)}{\alpha(s)} = \frac{5g}{7s^2} \tag{11}$$

The transfer function between motor voltage (v_m) and output angle (θ) is expressed as follows[7]:

$$\frac{\theta(s)}{v_m(s)} = \frac{\eta_g \eta_m K_t K_g}{J_{eq} R_m s^2 + (B_{eq} R_m + \eta_g \eta_m K_m K_t K_g^2)s} \tag{12}$$

The relation between angle (θ) of lever arm and angle (α) of the beam is:

$$\theta r = \alpha L \tag{13}$$

This finally leads to the following transfer function:

$$\frac{\alpha(s)}{\theta(s)} = \frac{r}{L} \qquad (14)$$

$$\frac{X(s)}{V_m(s)} = \frac{\theta(s)}{V_m(s)} \frac{\alpha(s)}{\theta(s)} \frac{X(s)}{\alpha(s)} \qquad (15)$$

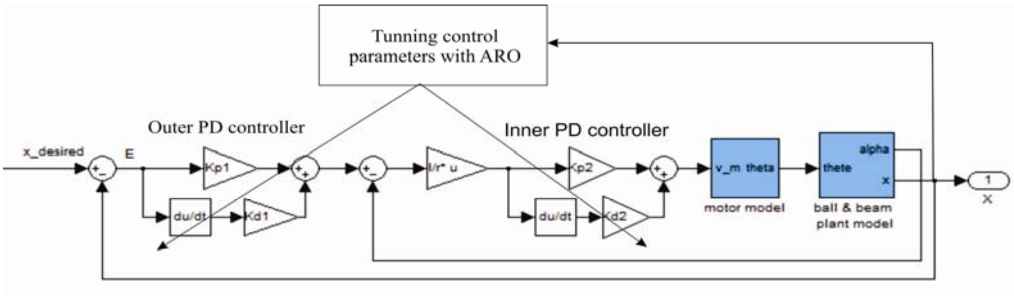

Fig. 3. An architecture of the PD cascade controller

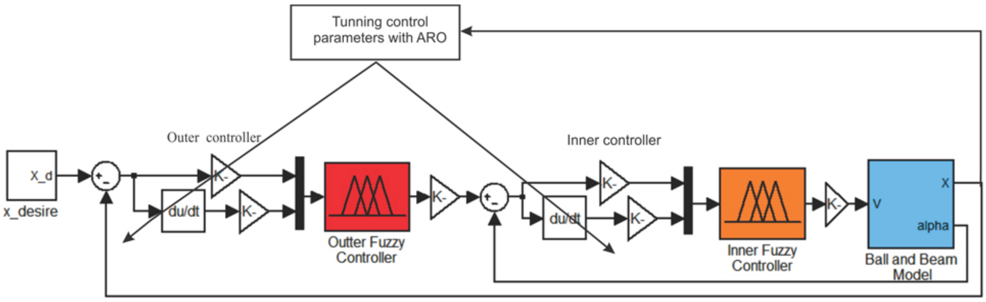

Fig. 4. An architecture of the fuzzy cascade controller

A state-feedback controller can be designed based on the open loop transfer function of the form[11].

3. Design of the Controller

The PD cascade controller is shown in Fig. 3. Outer loop controller determines the desired angle of beam based on the measured position of the ball and inner loop controller regulates the angle of motor to produce the desired beam angle. In this design, outer loop controller reduces disturbances in the system and produces inner loop signal in a manner that abrupt variations in the inner loop are avoided. Transfer functions of two PD controllers are as in (16)[11]. ARO technique tunes PD parameters.

$$G_1(s) = K_{p1} + K_{D1}s$$
$$G_2(s) = K_{p2} + K_{D2}s \qquad (16)$$

Fuzzy PD cascade controller is illustrated in Fig. 4.
In this controller, error and its rate of change are the inputs of fuzzy controller.

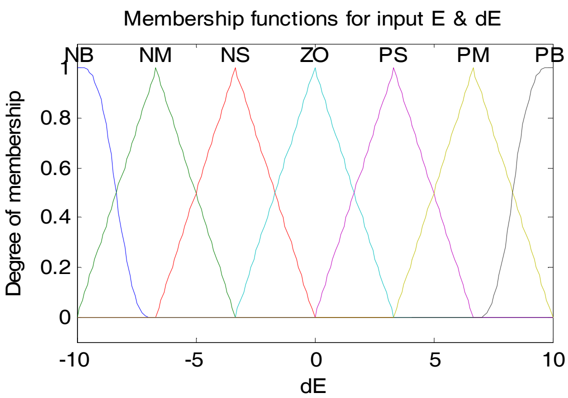

Fig. 5. Member functions for input E and ΔE

Due to using linguistic rules, fuzzy controller is fit for nonlinear systems and gives simpler design and better response. Fuzzy controller of type Mamdanifuzzifies the inputs using membership functions. Membership functions include 5 triangular and 2 Z-shaped functions for input (see Fig. 6) and 7 triangular functions for output (see Fig. 5). Linguistic variables for fuzzy controller are NB (Negative Big –m3), NM (Negative Medium –m2), NS (Negative Small –m1), ZO (zero 0), PS (Positive Small m1), PM (Positive Medium m2), PB (Positive Big m3).

the rule base of the fuzzy controller can be characterized using(17)

$$R^k : E_i \text{ is } A_1^k \text{ and } dE_i \text{ is } A_2^k \text{ then } U_i = B_k \qquad (17)$$

R^k is the kth rule $(k = 1, 2, ..., m)$, E_i and dE_i are the error and derivative error of the inner and outer loop.

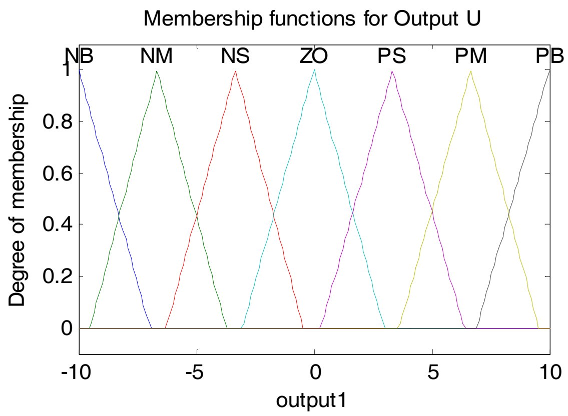

Fig. 6. Member functions for Output

A_i^k and B_k are linguistic variables as in Table 1. Defuzzification is done using center of gravity method.

Table 1
Rule base of the fuzzy controller

		E						
		NB	***NM***	***NS***	***ZO***	***PS***	***PM***	***PB***
dE	***NB***	-m3	-m3	-m3	-m3	-m2	-m1	0
	NM	-m3	-m3	-m3	-m3	-m1	0	m1
	NS	-m3	-m3	-m3	-m1	0	m1	m2
	ZO	-m3	-m3	-m1	0	m1	m2	m3
	PS	-m2	-m1	0	m1	m2	m3	m3
	PM	-m1	0	m1	m2	m3	m3	m3
	PB	0	m1	m2	m3	m3	m3	m3

Abrief introduction of ARO technique and its procedure is provided in the following.

The proposed algorithm is inspired by the budding method of asexual reproduction. Each individual is represented by a binary string like the binary representation in evolutionary algorithms. A decision variable vector $X = (x_1, x_2, ..., x_n)$; $X \in \Re^n$ is called an individual in ARO technique and each variable is considered as the chromosome made by a number of bits called genes.

We assume that each solution in the search space (*S*) is an organism in its environment. In addition, it is supposed that there are limited resources in the environment such that only the most deserving individual can survive. To start the algorithm, an individual is randomly (or specifically chosen) initiated in the distinctive domain of S, thereafter the individual reproduces an offspring labelled bud by a particular operator called reproduction mechanism completely described later. The parent and its offspring compete to survive according to a performance index or a fitness function. If the bud wins the competition, its parent will be discarded. Therefore, the bud is replaced with its parent and it becomes the new parent. If the parent triumphs, then, the bud will be thrown away. The algorithm repeats steps illustrated in 0until the stopping criteria are satisfied[9].

Table 2
Pseudo code of ARO[9].

```
Begin
        t = 1;
        P = Initialize (L,U); % Parent Initialization between lower and
   upper bound
        Fitness_P = fit(P); % Fitness of P is calculated
        While stopping conditions are not met % Stopping Criteria
             Bud(t) = Reproduce(P); % P reproduces a Bud
             Fitness_Bud(t) = fit(Bud(t)); % Fitness of Bud(t) is
   calculated
             If Fitness_Bud(t) is better than Fitness_P
                  P = Bud(t); % Bud(t) is replaced with P
             Else
                  clear Bud(t); % Bud(t) is discarded
             end
             t = t + 1;
        End
end
```

In order to reproduce, a copy of parent named larva is produced. Then a substring with g bits $g \sim Uniform[1, LI]$ (LI is the length of individual) in larva is randomly chosen. Afterward bits of the substring mutate such that in any selected gene, 1 is replaced by 0 and vice versa[9].

In fact, larva is a mutated form of its parent. After larva was produced by its parent, for each bit of substring chosen randomly from larva, a random number uniformly distributed in [0, 1] is generated. If this number is less than 0.5, the bit will be selected from the parent otherwise it will be chosen from larva till bud is completed. It means that merging is definitely performed. The number of bits going to be altered, g, is a random number. When g is large, more exploration is expected and vice versa, while the exploitation applied is done based on the aforementioned procedure; this means that the amount of exploration is merely controlled by g.; consequently, bud is generated similar to its biological model. On the other hand, during mutation, crossover is implicitly occurred. 0shows the reproduction mechanism[12].

Once produced, the bud fitness is evaluated according to the performance index. As illustrated in 0, bud fitness is compared with its parent fitness. At last, the most merited is capable of subsisting to reproduce.

In this paper, parameters are optimized using ARO to improve settling time and rise time. Fitness function used in the optimization process is

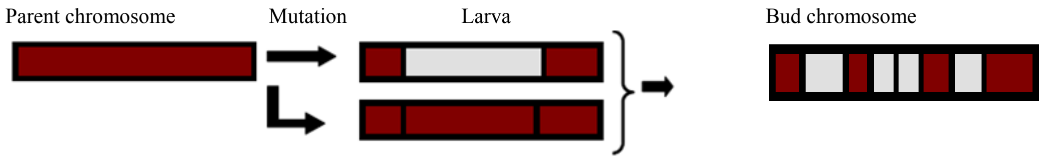

Fig. 7. Reproduction mechanism generating bud chromosome.

$$Ff = \frac{1}{(1+t_r+t_s)} \quad (18)$$

Where t_r and t_s are rise time and settling time, respectively. Decreasing values of t_r and t_s converge the fitness function to one. Fitness value at desired rise time and settling time is calculated and fed into ARO algorithm as a stopping criterion.

Parameters of controller were optimized by online method. According to desired function y_d, which is a generalized logistic function that specifies the desired path, RSS (Residual Sum of Squares) was calculated and fed into fitness function.

$$y_d = a + \frac{k-a}{(1+qe^{-b(t-m)})^{1/v}} \qquad RSS = \sum_t (y_d - y)^2 \quad (19)$$

$$Ff = \frac{1}{(1+RSS)} \quad (20)$$

where y is the position of ball.

4. Results

In this section, the results of simulation are presented. The position of ball with PD cascade and PD fuzzy cascade controllers are shown in 0fuzzy controller has lesser overshoot than PD cascade. As it is illustrated in 0PD cascade has high overshoots starting from second step and is stabilized far after fuzzy cascade. 0shows the performance of the controllers.

Fitness variations in consecutive iterations for PD cascade and fuzzy PD cascade controllers are shown in 0 and 0respectively. ARO algorithm stops when the fitness function reaches the goal fitness.

0and 0illustrate inner and outer controllers output, respectively. As seen in Fig. 11, values of α in the fuzzy controller is bound to the allowed range (-20 to 20 degrees).

5. Conclusion

In this research, PD cascade and fuzzy cascade controllers were used to control ball and beam system. As expected, fuzzy controller stabilized the system much better than PD cascade. We used ARO, an individual based optimization algorithm, for online optimization of controller parameters. Using fuzzy PD cascade controller optimized with ARO, optimum settling time and rising time were achieved. For the future works, a comparison between performance of controllers optimized with genetic and ARO algorithms is proposed.

Table 3
The performance of the optimal controllers

	Fuzzy PD cascade	*PD Cascade*
Rise Time(second)	2.32	2.35
Settling Time(second)	2.13	2.46
maximum Overshoot	1.88%	20%
Best cost of Fitness	0.1739	0.1721

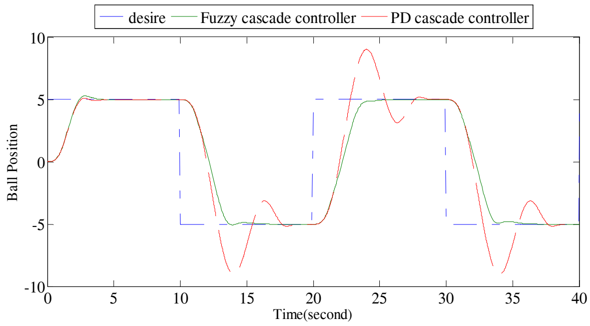

Fig. 8. The position of ball with pd and Fuzzy PD cascade controller

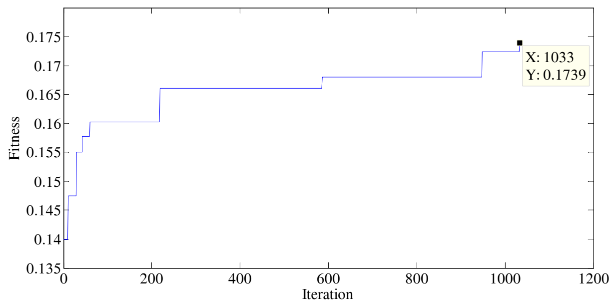

Fig. 9. Fitness cost changes in iteration for Fuzzy PD cascade controllers

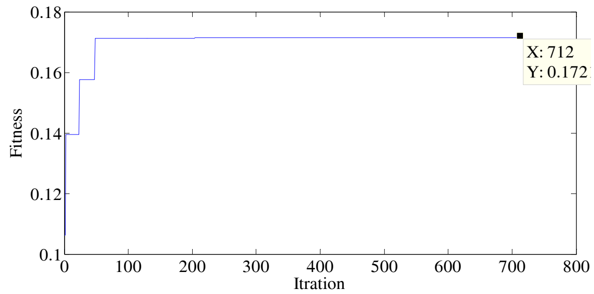

Fig. 10. Fitness cost changes in itration for PD cascade controllers

Table 4
Parameters of the system

symbol	Description	value
k_t	Motor torque constant	0.00767
k_m	Back-emf constant	0.00767
k_g	SRV02 systemgear ratio	70
R_m	Armature resistance	2.6
J_{eq}	Equivalent moment of inertia at the load	2.0e-3
B_{eq}	Equivalent viscous damping cofficient	4.0e-3
r	Lever arm offset (in.)	1
L	Beam length (in.)	16.75
g	Earth's gravitational constant(m/s^2)	9.8
η_g	Gearbox efficiency	0.9
η_g	Motor efficiency	0.36
m	Mass of the Ball (Kg)	0.064

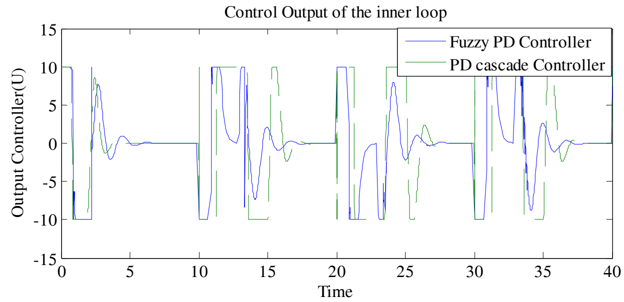

Fig. 11. Controller output of inner loop in PD and fuzzy PD cascade controller.

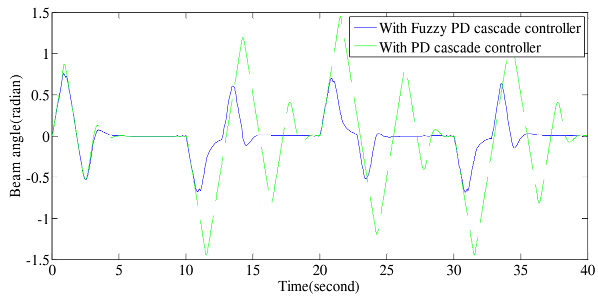

Fig. 12. Angle of beam for two controllers

References

[1] S.K. Oh, H.J. Jang, W. Pedrycz, "The design of a fuzzy cascade controller for ball and beam system: A study in optimization with the use of parallel genetic algorithms," Engineering Applications of Artificial Intelligence, vol. 22, pp. 261-271, 2009.
[2] N. Duc-Hoang , H. Thai-Hoang, "A SFLA-based fuzzy controller for balancing a ball and beam system," in Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on, 2008, pp. 1948-1953.
[3] M. Keshmiri, A. F. Jahromi, A. Mohebbi, M. H. Amoozgar, and W. F. Xie, "Modeling and control of ball and beam sys tem using model based and non-model based control approaches," Smart Sensing And Intelligent Systems, vol. 5, p. 22, 2012.
[4] A.Raees , M. B. Kadri, "Fuzzy Hammerstein model based generalized predictive control for ball and beam system," in Emerging Technologies (ICET), 2012 International Conference on, 2012, pp. 1-6.
[5] W. Wei , X. Peng, "A Research on Control Methods of Ball and Beam System Based on Adaptive Neural Network," in Computational and Information Sciences (ICCIS), 2010 International Conference on, 2010, pp. 1072-1075.
[6] Y.H. Chang, C.W. Chang, C.W. Tao, H.W. Lin, and J.S. Taur, "Fuzzy sliding-mode control for ball and beam system with fuzzy ant colony optimization," Expert Systems with Applications, vol. 39, pp. 3624-3633, 2012.
[7] Y. Wen , F. Ortiz, "Stability analysis of PD regulation for ball and beam system," in Control Applications, 2005. CCA 2005. Proceedings of 2005 IEEE Conference on, 2005, pp. 517-522.
[8] S.K. Oh, H.J. Jang, W. Pedrycz, "A comparative experimental study of type-1/type-2 fuzzy cascade controller based on genetic algorithms and particle swarm optimization," Expert Syst. Appl., vol. 38, pp. 11217-11229, 2011.
[9] A. Farasat, M. B. Menhaj, T. Mansouri, M. R. S. Moghadam, "ARO: A new model-free optimization algorithm inspired from asexual reproduction," Appl. Soft Comput., vol. 10, pp. 1284-1292, 2010.
[10] A. R. Khanteymoori, M. B. Menhaj, M. M. Homayounpour, "A new evolutionary algorithm for structure learning in Bayesian networks," presented at the Computer Conference, 2009. CSICC 2009. 14th International CSI, Tehran, 2009.
[11] 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, pp. 181-195, 2011.
[12] A. R. Khanteymoori, M. B. Menhaj, and M. M. Homayounpour, "Structure Learning in Bayesian Networks Using Asexual Reproduction Optimization," ETRI Journal, vol. 33, p. 10, 2011.