International Journal of
Intelligent Systems and Applications in Engineering

ISSN:2147-6799 www.ijisae.org Original Research Paper

# Artificial Bee Colony Algorithm Based Linear Quadratic Optimal Controller Design for a Nonlinear Inverted Pendulum

**Baris Ata[1*], Ramazan Coban[2]**

*Received 13th September 2014, Accepted 08th December 2014*

***Abstract:*** This paper presents a linear quadratic optimal controller design for a nonlinear inverted pendulum. Linear Quadratic Regulator (LQR), an optimal control method, is usually used for control of the dynamical systems. Main design parameters in LQR are the weighting matrices; however there is no relevant systematic techniques presented to choose these matrices. Generally, selecting weighting matrices is performed by trial and error method since there is no direct relation between weighting matrices and time domain specifications like overshoot percentage, settling time, and steady state error. Also it is time consuming and highly depends on designer's experience. In this paper LQR is used to control an inverted pendulum as a nonlinear dynamical system and the Artificial Bee Colony (ABC) algorithm is used for selecting weighting matrices to overcome LQR design difficulties. The ABC algorithm is a swarm intelligence based optimization algorithm and it can be used for multivariable function optimization efficiently. The simulation results justify that the ABC algorithm is a very efficient way to determine LQR weighting matrices in comparison with trial and error method.

***Keywords:*** *ABC, LQR, Inverted Pendulum, Optimal Control, Weighting Matrices*

## 1. Introduction

The inverted pendulum is an unstable and under-actuated system with highly nonlinear dynamics [1]. The control of inverted pendulum is a classic example for design, testing, and comparing of different control techniques as a consequence the control of inverted pendulum has been a research interest in the field of control engineering and the inverted pendulum has been a standard tool in control laboratories for years. Another reason behind the extensive studies of the inverted pendulum is that several important control systems can be modelled with the help of inverted pendulum [2]. Inverted pendulum reveals many interesting system-theoretic properties and its dynamics are fundamental to maintenance balance, such as walking and two-wheeled robots [3], [4].

Optimal control theory is a mathematical optimization method as an extension of the calculus variation and it has numerous applications in control engineering. Determination of the control signals that will cause a process to meet the physical constraints and also maximization or minimization of some performance criteria are the main objectives of optimal control theory [5]. A special case of the general nonlinear optimal control problem where the cost function is a quadratic function and the system dynamics are described by a set of linear differential equations is a linear quadratic optimal control problem. Linear quadratic optimal control can be implemented in numerous control engineering problems, also it provides a basis for many other control techniques and hence it is very important for modern control theory [6]. Linear Quadratic Regulator (LQR) is one of the main solutions for linear quadratic optimal control problem. LQR has a simple process that can achieve the closed loop linear quadratic optimal control with linear state or output feedback [7], [8].

The most challenging part of LQR is selection of suitable weighting matrices which affects the control input [9]. In general, selecting weighting matrices is performed by trial and error method, however, there does not exist a direct connection between weighting matrices and time domain specifications such as overshoot percentage, settling time, and steady state error. There are no relevant systematic techniques for selecting weighting matrices, however; recently a few researchers have proposed artificial intelligence algorithms such as genetic algorithms and particle swarm optimization algorithm for this goal [10], [11]. In addition, the Artificial Bee Colony algorithm, another swarm intelligence optimization algorithm based on the intelligent behaviour of honey bee swarm, can be used to determine LQR weighting matrices.

The control problem is defined as selecting appropriate LQR weighting matrices to stabilize cart position and pole angle of nonlinear inverted pendulum while minimizing settling time, steady state error, and overshoot percentage. In this case study, the ABC algorithm is proposed to determine LQR weighting matrices and simulation results illustrate that proposed method achieves desired control system characteristics and also has a satisfactory control performance.

The rest of this paper is organized as follows. Section II contains the nonlinear mathematical model of the inverted pendulum that is used in this study. In section III, the linear quadratic optimal control problem based on linearized pendulum model is described. Section IV contains an overview of the ABC algorithm. Determination of LQR weighting matrices and simulation results are illustrated in section V followed by conclusion in section VI.

## 2. The Inverted Pendulum

[1] *Department of Computer Engg. Cukurova University, Adana, Turkey*
[2] *Department of Computer Engg. Cukurova University, Adana, Turkey*
* *Email: bata@cu.edu.tr*
*Note: This paper has been presented at the International Conference on Advanced Technology&Sciences (ICAT'14) held in Antalya (Turkey), August 12-15, 2014.*

The inverted pendulum system used in this study consists of a motor driven cart and a pendulum hinged to it, as shown in (Figure 1). The inverted pendulum system model used in this paper has been suggested by Ogata [12]. The main aim of the controller is to stabilize the pendulum as to keep pendulum upright position in response to a change in cart position. Designed block diagram of control system is shown in (Figure 2). Linear quadratic optimal control where weighting matrices are selected by the ABC algorithm can be used to determine design variables; integral gain constant $K_I$ and feedback gain matrix $K$.

Assume that the rod is massless and the pendulum mass is concentrated at the end of the rod. $\theta$ is the angel of the rod from vertical line and the control force $u$ is applied to the cart. Also assume that the sampling period $T$ is 0.1 $s$, $g$ is 9.81 $m/s^2$ and the following numerical values for $M, m$ and $l$:

$$M = 2kg,\ m = 0.1kg,\ l = 0.5m$$

In solving this design problem, we shall define state variables $x_1,\ x_2,\ x_3$ and $x_4$ as follows:

$$x_1 = \theta,\ x_2 = \dot{\theta},\ x_3 = x,\ x_4 = \dot{x}$$

for nonlinear inverted pendulum model these variables can be written as differential equations as follows: $\dot{x}_1 = x_2$ $\dot{x}_1 = x_2$

$$\dot{x}_1 = x_2 \tag{1}$$

$$\dot{x}_2 = \frac{u + \cos x_1 - (M+m) g \sin x_1 + ml \cos x_1 \sin x_1 x_2^2}{ml \cos^2 x_1 - (M+m) l} \tag{2}$$

$$\dot{x}_3 = x_4 \tag{3}$$

$$\dot{x}_4 = \frac{u + ml \sin x_1 x_2^2 - mg \cos x_1 \sin x_1}{M + m - m \cos^2 x_1} \tag{4}$$

which are solved using fourth-order Runge-Kutta method in this case study.

If displacement of the car is considered as the output of the system then the output equation $y$ becomes

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{5}$$

After linearizing the nonlinear differential (Equation 1) through (Equation 4) by taking $sin\theta \doteqdot \theta$, $cos\theta \doteqdot 1$, and $\theta\dot{\theta}^2 \doteqdot 0$, the discretized state and output equations of the system for linear quadratic optimal control can be derived as follows:

$$x(k+1) = Gx(k) + Hu(k) \tag{6}$$

$$y(k) = Cx(k) + Du(k) \tag{7}$$

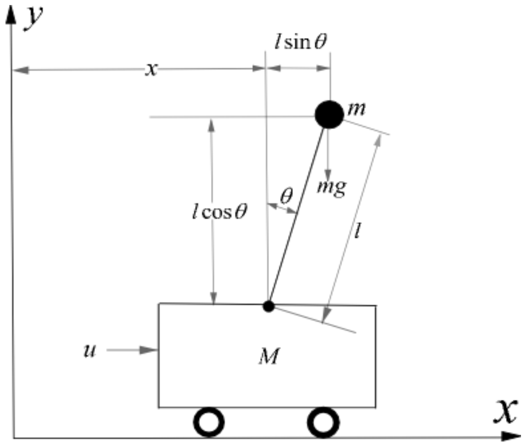


**Figure 1.** Inverted pendulum system [12]

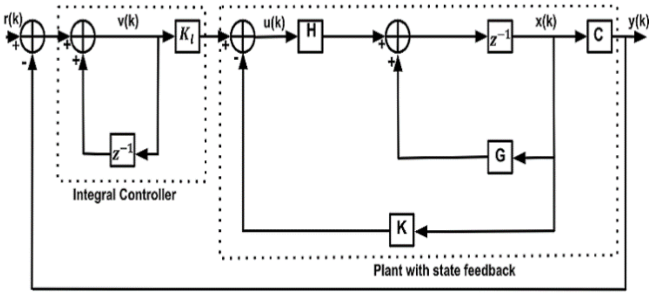


**Figure 2.** Block diagram of control system

where

$$G = \begin{bmatrix} 1.1048 & 0.1035 & 0 & 0 \\ 2.1316 & 1.1048 & 0 & 0 \\ -0.0025 & -0.0001 & 1 & 0.1 \\ -0.0508 & -0.0025 & 0 & 0 \end{bmatrix}$$

$$H = \begin{bmatrix} -0.0051 \\ -0.1035 \\ 0.0025 \\ 0.0501 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \end{bmatrix}$$

## 3. Linear Quadratic Optimal Control

The state space representation of a linear time-invariant (LTI) control system can be written as follows [12]:

$$x(k+1) = Gx(k) + Hu(k) \tag{8}$$

$$y(k) = Cx(k) \tag{9}$$

$$v(k) = v(k-1) + r(k) - y(k) \tag{10}$$

$$u = -K(k)x(k) + K_I v(k) = -\begin{bmatrix} K & -K_I \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix} \tag{11}$$

where $x(k)$ is state vector ($n$ vector) and $u(k)$ is control vector ($r$

vector), respectively. $G$ and $H$ are $n \times n$ and $n \times r$ matrices, indicate the constant system. $K$ is state feedback matrix.

At steady-state, the overall system dynamics with constant gain and integral feedback is described by the state equation which is a combination of (Equation 8) through ( Equation 11):

$$\begin{bmatrix} x(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} G & 0 \\ -CG & 1 \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} H \\ -CH \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(k) \qquad (12)$$

In (Equation 12), it is assumed that the reference is a step change hence $r(k) = r(k+1)$.

Let us define

$$\hat{G} = \begin{bmatrix} G & 0 \\ -CG & 1 \end{bmatrix}$$

$$\hat{H} = \begin{bmatrix} H \\ -CH \end{bmatrix}$$

$$\hat{K} = \begin{bmatrix} K & -K_I \end{bmatrix}$$

$$\hat{x} = \begin{bmatrix} x \\ v \end{bmatrix}$$

$$\hat{D} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Therefore, the last equation can be written as follows:

$$\hat{x}(k+1) = \left(\hat{G} - \hat{H}\hat{K}\right)\hat{x}(k) + \hat{D}r(k) \qquad (13)$$

The (Equation 13) is the state equation of the closed-loop control system. Its output equation is

$$y(k) = \hat{C}\hat{x}(k) + [0]r(k) \qquad (14)$$

where $\hat{C} = [C \quad 0]$.

Linear quadratic optimal control problem may be stated to find the optimal input $u$ sequence that minimizes the quadratic performance index which is defined as:

$$J = \frac{1}{2}\sum_{k=o}^{\infty}\left[x^*(k)Qx(k) + u^*(k)Ru(k)\right] \qquad (15)$$

where Q is positive definite $n \times n$ matrix and $R$ is positive definite $r \times r$ matrix. The notation ( $^*$) indicates complex-conjugate transpose of a matrix. Matrices $Q$ and $R$ are selected to weight the relative importance of the performance measures caused by the state vector and control vector, respectively.

The state feedback gain matrix is defined as follows:

$$\hat{K} = (R + \hat{H}^*P\hat{H})^{-1}\hat{H}^*P\hat{G} \qquad (16)$$

which is obtained by solving the following Ricatti equation:

$$P = Q + \hat{G}^*P\hat{G} - \hat{G}^*P\hat{H}(R + \hat{H}^*P\hat{H})^{-1}\hat{H}^*P\hat{G} \qquad (17)$$

By the sense of Liapunov, for a stable matrix $(\hat{G} - \hat{H}\hat{K})$, the matrix $P$ must be a positive definite, or for asymptotic stability a positive semi-definite.

According to main design parameters of the linear quadratic optimal control problem which are weighting matrices $Q$ and $R$, the quality of the controller design depends on the choice of these matrices. However, there are no relevant systematic techniques to select weighting matrices. This goal is performed by trial and error method in general. Although it depends on the designer's experience, it is highly time consuming and selected values for weighting matrices cannot establish a direct effect on the desired particular control system specifications.

According to the importance of selecting weighting matrices $Q$ and $R$, selection of these matrices is performed by the ABC algorithm in this paper.

## 4. The Artificial Bee Colony Algorithm

The Artificial Bee Colony (ABC) algorithm was introduced by Karaboga in 2005 as a new method which is based on the intelligent behavior of honey bee swarms finding nectar and sharing the information of food resources with each other in the field Swarm Intelligence to solve to optimize numeric benchmark functions [13]. Then it was extended by Karaboga and Basturk and presented to exceed other recognized heuristic methods like Genetic Algorithm as well as Differential Evolution algorithm and Particle Swarm Optimization [14], [15]. The ABC algorithm has the advantages of strong robustness, fast convergence and high flexibility, fewer control parameters and also it can be used for solving multidimensional and multimodal optimization problems [16], [17].

In the ABC algorithm, the colony of artificial bees contains three groups of bees: employed bees, onlooker bees and scout bees. An employed bee memorizes the quality of the food source and finds a food source by modifying this information. Employed bees share the food source information with other bees on the dance area. Onlooker bees watch the dance of employed bees within the hive and find the food sources using the information provided by employed bees. Scout bees search new food sources around the hive randomly. Both onlookers and scouts are also called unemployed bees. The number of employed bees is equal to the number of food sources since each employed bee is associated with one and only one food source. The position of a food source means a possible solution to the problem and the nectar amount of a food source corresponds to the fitness of the associated solution.

The general scheme of the ABC algorithm contains four phase; initialization phase, employed bees phase, onlooker bees phase, and scout bees phase. Detailed pseudo code of the ABC algorithm is as follows [18]:

1. Initialize the population of solutions
2. Evaluate the population
3. cycle=1
4. repeat
5. Produce new solutions (food source positions) $v_{ij}$ in the neighborhood of $x_{ij}$ for the employed bees using the formula $v_{ij} = x_{ij} + \Phi_{ij}(x_{ij} - x_{kj})$(k is a solution in the neighborhood of $i$ ,$\Phi$ is a random number in the range [-1,1] ) and evaluate them
6. Apply the greedy selection process between $x_i$ and $v_i$
7. Calculate the probability values $P_i$ for the solutions $x_i$ by means of their fitness values using the following equation

$$P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i} \qquad (18)$$

In order to calculate the fitness values of solutions we employed the following equation

$$fit_i = \begin{cases} \dfrac{1}{1+f_i} & if\ f_i \geq 0 \\ 1+abs(f_i) & if\ f_i < 0 \end{cases} \quad (19)$$

Normalize $P_i$ values into [0, 1]

8. Produce the new solutions (new positions) $v_i$ for the onlookers from the solutions $x_i$, selected depending on $P_i$, and evaluate them
9. Apply the greedy selection process for the onlookers between $x_i$ and $v_i$
10. Determine the abandoned solution (source), if exists, and replace it with a new randomly produced solution $x_i$ for the scout using the following equation

$$x_{ij} = min_j + rand(0,1) \times (\max_j - min_j) \quad (20)$$

11. Memorize the best food source position (solution) achieved so far
12. cycle=cycle+1
13. until cycle= Maximum Cycle Number (MCN)

## 5. Simulation Results

The importance and difficulty of selecting weighting matrices was mentioned above. The matrices $Q$ and $R$ were chosen by trial and error method as follows in [12]:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, R = [1]$$

Based on these matrices, feedback gain matrix $K$ and integral gain constant $K_I$ are determined as follows:

$$K = [-64.9346 \ \ -14.4819 \ \ -10.8475 \ \ -9.2871]$$

$$K_I = \text{-}0.5189$$

The goal of this simulation is to reduce the settling time $(t_s)$ of unit-step response $y(k)$ (the cart position) without an overshoot $(os)$ or with a minimum overshoot also minimize steady-state error $(e_{ss})$. The objective weighting method where multiple objective functions are combined into one objective function $f_{sum}$ can be used for multi-objective optimization [19]. The objective function defined as:

$$f_{sum} = K_1 t_s + K_2 os + K_3 e_{ss} \quad (21)$$

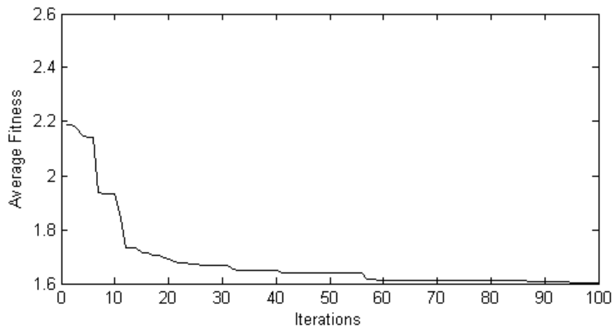


**Figure 3.** Average fitness during the ABC convergence

where $K_1$, $K_2$ and $K_3$ are weight coefficients of the fitness functions and their values were chosen as 1.0.
The ABC algorithm was employed to select best $Q$ and $R$ matrices that minimize $f_{sum}$. The results of applying the ABC algorithm to the problem are summarized as follows:
The parameters of the ABC algorithm are set in the range [0.1 100], colony size=20 and max cycle=100. The average fitness values during the ABC algorithm running is shown in (Figure 3). The weighting matrices obtained by ABC algorithm are:

$$Q = \begin{bmatrix} 82.4186 & 34.8695 & 50.4278 & 17.4539 & 66.1159 \\ 34.8695 & 24.7816 & 13.6630 & 3.5719 & 25.2813 \\ 50.4278 & 13.6630 & 51.4173 & 10.8109 & 39.9392 \\ 17.4539 & 3.5719 & 10.8109 & 11.4281 & 18.6369 \\ 66.1159 & 25.2813 & 39.9392 & 18.6369 & 60.7356 \end{bmatrix}$$

$$R = [0.1000]$$

Using the $Q$ and $R$ matrices obtained by the ABC algorithm the matrix $P$ is calculated by (Equation 17) as follows:

$$P = \begin{bmatrix} 17385 & 3737.4 & 15220 & 6395.5 & -2403.4 \\ 3737.4 & 823.83 & 3315 & 1381.7 & -507.19 \\ 15220 & 3315 & 15439 & 5924.9 & -2660.2 \\ 6395.5 & 1381.7 & 5924.9 & 2436.7 & -952.22 \\ -2403.4 & -507.19 & -2660.2 & -952.22 & 644.42 \end{bmatrix}$$

Moreover, based on these matrices, feedback gain matrix $K$ and integral gain constant $K_I$ are determined as follows:

$$K = [\text{-}137.7804 \ \ \text{-}31.1832 \ \ \text{-}68.8161 \ \ \text{-}35.4986]$$

$$K_I = \text{-}6.9144$$

**Table 1.** Performance Results

| | ***Trial and Error [12]*** | ***The Proposed*** |
|---|---|---|
| $t_s(s)$ | 4.8987 | 1.4963 |
| $os(\%)$ | 0 | 0.1038 |
| $e_{ss}$ | $2.3938\times10^{-004}$ | $5.3479\times10^{-009}$ |
| $f_{sum}$ | 4.8989 | 1.6002 |

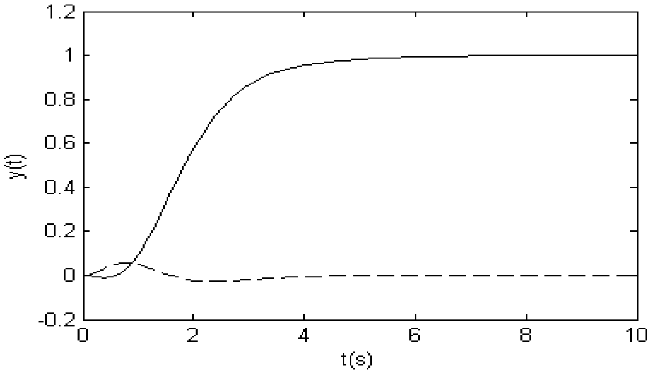


**Figure 4.** Cart Position and Pendulum Angle for Trial and Error Method

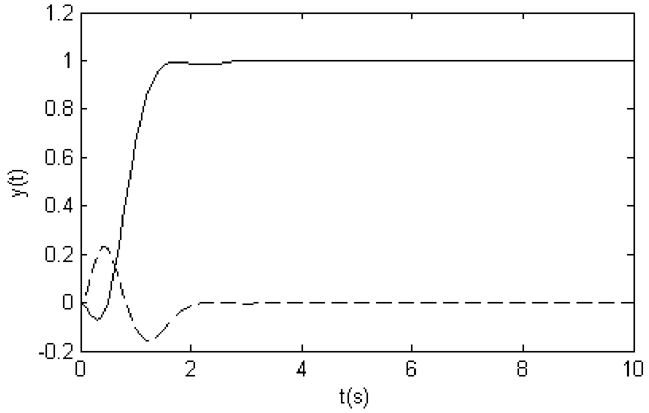


**Figure 5.** Cart Position and Pendulum Angle for Proposed Method

Since the matrix $P$ is positive definite, the closed-loop control system is stable. That is, all eigenvalues of $\left(\hat{G}-\hat{H}\hat{K}\right)$ lie inside of the unit circle in the following:

$$z_1 = 0.2077$$

$$z_2 = 0.77628 + 0.28782i$$

$$z_3 = 0.77628 - 0.28782i$$

$$z_4 = 0.75904$$

$$z_5 = 0.72792$$

The performance results are presented in (Table 1). Also plots of position of the cart and pendulum angels as unit-step response of designed system has been is in (Figure 4) and (Figure 5) for both control systems. Cart position is indicated by solid line and pendulum angle is indicated by dotted line in these figures. Both simulations are performed on the nonlinear pendulum model given by (Equation 1) through (Equation 5).

## 6. Conclusion

In this paper, the ABC algorithm based linear optimal controller design for nonlinear inverted pendulum has been presented. The ABC algorithm has been employed to determine linear quadratic optimal controller weighting matrices. Using the ABC algorithm has been proved to be effective and feasible to select weighting matrices for nonlinear pendulum controller design more than trial and error method. Also it has been shown that it can optimize multiple time domain control system specifications such as settling time, overshot and steady state error.

## References

[1] K. J. Aström and K. Furuta, "Swinging up a pendulum by energy control," Automatica, vol. 36, no. 2, pp. 287-295, 2000.

[2] J. Anderson, "Learning to control of an inverted pendulum using neural networks," IEEE Control Systems Magazine, vol. 9, no. 3, pp. 31-37, 1989.

[3] A. Kuo, "The six determinants of gaint and the inverted pendulum analogy: Adynamic walking perspective," Human Movement, vol. 26, pp. 617-656, 2007.

[4] S. Jeong and T. Takahashi, "Wheeled inverted pendulum type asistant robot: inverted mobile, standing and sitting motions," in IEEE/RSJ International Conferance on Intelligent Robots and Systems, 2007, 2007.

[5] D. E. Kirk, Optimal control theory: an introduction, New Jersey: Prentice-Hall, Inc., 1970.

[6] M. Athans, "The status of optimal control theory and applications for for deterministic systems," IEEE Transactions on Automatic Control, vol. 11, no. 3, pp. 580-596, 1966.

[7] R. E. Kalman, "When is a linear control system optimal?," Journal of Basic Engineering, vol. 86, pp. 51-56, 1964.

[8] H. Kwakernaak and R. Sivan, Linear optimal control systems, New York: Wiley-Interscience, 1972.

[9] J. V. D. F. Neto, I. S. Abreu and F. N. Da Silva, "Neural-genetic synthesis for state-space controllers based on linear quadratic regulator design for eigenstructure assignment," Trans. Sys. Man Cyber. Part B, vol. 40, no. 2, pp. 266-285, 2010.

[10] C. P. Bottura, J. V. da Fonseca Neto, "Parallel eigenstructure assignment via LQR design and genetic algorithms," in Proceedings of the American Control Conference 1999. Vol. 4. , San Diago, 1999.

[11] S. Mobayen, A. Rabiei, M. Moradi and B. Mohammady, "Linear quadratic optimal control system design using particle swarm optimization algorithm," International Journal of the Physical Sciences, vol. 6(30), pp. 6958-6966, 2011.

[12] K. Ogata, "Quadratic Optimal Control Systems," in Discrete-Time Control Systems 2nd Edition, Englewood Cliffs, NJ, Prentice-Hall, 1995, pp. 566-633.

[13] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Erciyes University, Kayseri, Turkey, 2005.

[14] D. Karaboga and B. Basturk, "An artificial bee colony (ABC) algorithm for numeric function optimization.," in IEEE Symp. Swarm Intelligence, Indianapolis, 2006.

[15] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," Applied Soft Computing, vol. 8, no. 1, pp. 687-697, 2008.

[16] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," Journal of Global Optimization, vol. 39, no. 3, pp. 459-471, 2007.

[17] D. Karaboga and B. Akay, "A comparative study of Artificial Bee Colony algorithm," Applied Mathematics and Computation, vol. 214, no. 1, p. 108–132, 2009

[18] "Detailed Pseudocode of the ABC Algorithm," 14 October 2008. [Online]. Available: http://mf.erciyes.edu.tr/abc/pub/PsuedoCode.pdf. [Accessed 1 March 2014].

[19] R. Coban , "A fuzzy controller design for nuclear research reactors using the particle swarm optimization algorithm," Nuclear Engineering and Design, vol. 241, no. 5, pp. 1899-1908, 2011.