Control System Design using Particle Swarm Optimization (PSO)

J.Hamidi

Abstract— The main purpose of this paper is to select the appropriate weighting matrices for designing of optimal controller using Particle Swarm Optimization (PSO) algorithm as an intelligent procedure. Generally speaking, it is not easy to determine the optimal weighting matrices for a high-dimension control system via analytical methods. There is no direct relation between the elements of weighting matrices and desirable control system characteristics and selecting these weights is performed using time-consuming trial and error method and based on designer experiences. Superior features of PSO method are fast tuning of the parameters, rapid convergence, less computational burden and capability to avoid from local optima. Simulation results demonstrate that our proposed method is more efficient and robust compared with other heuristic method, i.e., the Genetic Algorithm (GA) method.

Index Terms— Weighting matrices, Particle Swarm Optimization (PSO), Genetic Algorithm (GA).

I. INTRODUCTION

Linear quadratique optimal control technique is important for modern control theory, because it can be implemented easily for engineering problems and it is the elementary technique of other control methods. However, in the particular case where the evaluation function is a linear quadratique function, the optimal solution converges to the solution of the LQR [1]. Finally, the problem is to find the appropriate weighting matrices to satisfy the performance of the control system under inequality constraints.

LQR method is increasingly used in aspects like control of induction motors, vehicular drive-shaft control, etc. The problem of selecting weighting matrices has been investigated by various methods. Kalman [2] firstly presented a method for weighting matrices selection based on the given poles. Wang [3] expanded the researches in Kalman direction. The design methods only compute the weighting matrices based on the given poles, and don't guarantee the performance and constraints of the system. Recently, more researches have been performed for LQR controller design using GA method [4-7]. This method has some disadvantages in designing of LQR controller such as high computational time and low rate of successful optimization. In order to overcome these difficulties, a new approach is proposed which employing new optimization method called PSO. Our goal is to design a LQR controller that has a well control performance and achieves to desired system characteristics.

This paper is organized as follows: section II contains a description of the LQR control problem. In section III, we present an overview of PSO algorithm and present the implementation of PSO method for designing of LQR and selecting appropriate weighting matrices. Simulation results are presented in section IV followed by conclusion in sectionV.

II. LINEAR QUADRATIC OPTIMAL CONTROL

Consider a linear time-invariant (LTI) system represented by the following state equation:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ u &= -Kx \end{aligned} \tag{1}$$

where ***x*** is n×1 state vector, ***u*** is m×1 input vector, ***A*** and ***B*** indicate the constant system model parameters and ***K*** is state-feedback matrix. The pair (***A,B***) is assumed to be stabilized. The linear quadratic cost function is defined as:

$$J = \frac{1}{2}\int_{t_0}^{t_f} (x^T Qx + u^T Ru)dt \tag{2}$$

Where ***Q*** is positive semi-definite n×n matrix and ***R*** is positive definite m×m matrix. The conventional LQ problem is to find the optimal input u* such that J is minimized. For $t_f = \infty$, the state feedback matrix $K = R^{-1}B^T G$ can be obtained by solving the following algebraic Riccati equation:

$$Q - GBR^{-1}B^T G + A^T G + GA = 0 \tag{3}$$

When we determine the control objective in optimal control system and select the weighting matrices Q and R, the resultant optimal state feedback matrix is unique. However, there aren't yet suitable systematic methods for weighing matrices selection. The selection of weighting matrices is usually done by trial and error procedure and based on the designer's experience. Then, regarding to the importance of weighting matrices selection, we describe the PSO algorithm and use it for generating of weighting matrices and designing the desirable LQR controller.

III. PARTICLE SWARM OPTIMIZATION

Considering the social behavior of swarm of fish, bees and other animals, the concept of PSO is developed. PSO is a robust stochastic evolutionary computation method based on the movement of swarms looking for the most fertile feeding location [8]. In general, PSO implementation is easier than GA. Indeed,

Manuscript received July 09, 2011.

Javad Hamidi, Electrical Engineering Department, Islamic Azad University, Sarakhs Branch, Sarakhs, Iran, +985116655937/ +9151193964., (e-mail: Hamidi.Javad@gamil.com).

PSO only has one operator; velocity calculation, so the computation time is decreased significantly. The reason is PSO does not perform the selection and crossover operations in evolutionary process.

Another difference between GA and PSO is the ability to control convergence. Crossover and mutation rates can affect the convergence of GA, but nothing can compare to the level of control achieved through manipulating of the inertial weight. The more decrease of inertial weight the more increase the swarm's convergence. This type of control allows determining the rate of convergence, and the level of 'stagnation' eventually achieved. Stagnation occurs in GA when all of the individuals have the same genetic code. In that case the gene pool is uniform, crossover has little or no effect on population and each successive generation is essentially same as the first. However, in the PSO, this effect can be controlled or prevented [9-10].

All solutions in PSO 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 the most optimization techniques, PSO requires a fitness evaluation function relevant to the particle's position. 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 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 [8]:

$$V_i(new) = w \times V_i(old) + c_1 \times rand() \times (X_{PB} - X_i) + c_2 \times Rand() \times (X_{GB} - X_i) \quad (4)$$

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 [8]:

$$w = 0.5(1 + rand(0,1)) \quad (5)$$

The positions are updated based on their movement over a discrete time interval (Δt) as follows:

$$X_i = X_i + V_i \times \Delta t \quad (6)$$

Where Δt usually is set to 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 the new point. If the particle's fitness value is greater than P_{best}, then P_{best} is replaced by the current position. The flowchart of PSO algorithm is illustrated in 'Figure 1'. PSO algorithm parameters are set based on trial and error as follows:

- Number of particles for each controller = 30 ;
- Acceleration constants $c1 = c2 = 1.5$;
- Maximum generation=20.

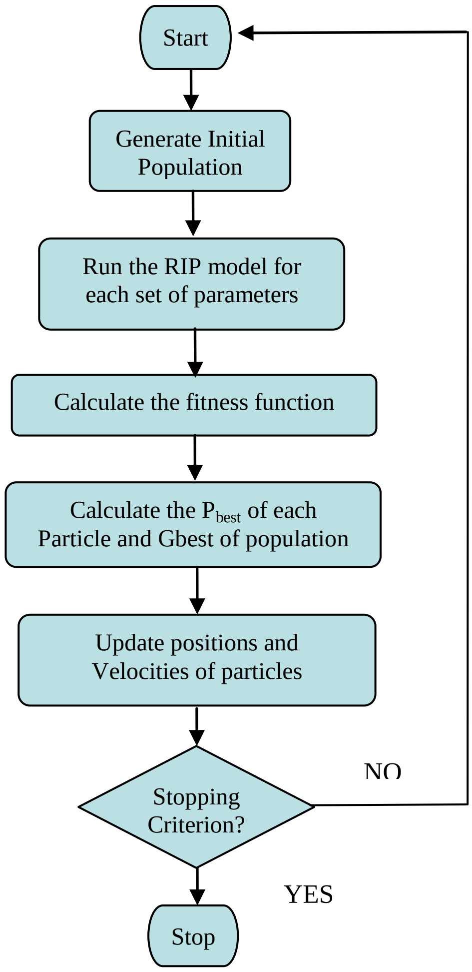

Figure 1: Flowchart of PSO controller design procedure

IV. SIMULATION RESULTS

Consider the aircraft landing flare system as a high dimensional system which is modeled by the simplified equations as:

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \dot{h} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} -0.058 & 0.065 & 0 & -0.171 & 0 & 1 \\ -0.303 & -0.685 & 1.109 & 0 & 0 & 0 \\ 0.072 & -0.685 & -0.947 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1.133 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.571 \end{bmatrix} \cdot \begin{bmatrix} u \\ w \\ q \\ \theta \\ h \\ e \end{bmatrix} + \begin{bmatrix} 0 & 0 & -0.119 \\ -0.054 & 0 & 0.074 \\ -1.117 & 0 & 0.115 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0.571 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mu \\ \gamma \\ \delta \end{bmatrix} \quad (7)$$

where u is the aircraft forward speed along its main body axis (ms^{-1}), w is the velocity downwards at right angles to the main body axis (ms^{-1}), q is the angular velocity of pitch with respect to the ground ($degrees^{-1}$), θ is the pitch with respect to the ground ($degrees^{-1}$),

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

h is the height with respect to 1 m below the ground (m), e is the forward acceleration due to throttle action (ms^{-2}), μ is the elevator angle (degrees), γ is the
throttle value (ms^{-2}), δ is the spoiler angle (degrees). The airplane system is shown in 'Figure 2'. For formulating the problem, we design the input (u γ δ)T such that the aircraft comes into land along an exponential path.

$$h^{\bullet} + 0.2h = 0 \qquad (8)$$

and the control criterion which is the integral of absolute error ($IAE = \int_0^{t_f} |e| dt$) is minimized. The system state initial conditions are:

$x(0)$=[$u(0)$ $w(0)$ $q(0)$ $\theta(0)$ $h(0)$ $e(0)$]^T = [5 -2.5 -1 -3 15 0.5]^T

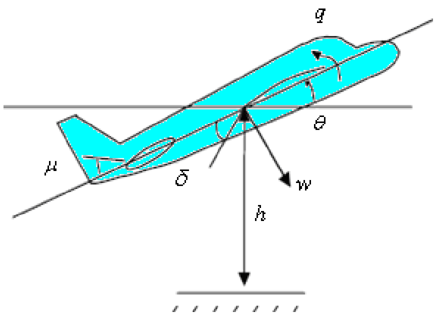

Figure 2 The aircraft landing system

From equation (7), we have $h^{\bullet} = -w + 1.133\theta$ and taking it into equation (8), we can obtain

$$\int_0^{30} |-w + 1.133\theta + 0.2h| dt = \int_0^{30} |-x_2 + 1.133x_4 + 0.2x_5| dt \qquad (9)$$

where $w=x_2$, $\theta=x_4$, $h=x_5$. The design method is first to select matrices Q and R. Then, equations (1) and (3) are solved using computer computation. Simulation results reveal whether any the system constraints exceed or not. If the constraints doesn't exceed, the weighting matrices are reselected and the procedure is repeated. Selecting the weighting matrices is very difficult in this case study using this procedure. Then, trial and error method should be spent longer time for simulation, and check the system response whether in state or input variables exceed constraints or not. One of the results using trial and error is:

$$Q = \begin{bmatrix} 0.618 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.073 & 0 & -0.129 & -0.8 & 0 \\ 0 & 0 & 0.484 & 0 & 0 & 0 \\ 0 & -0.129 & 0 & 0.055 & 0.667 & 0 \\ 0 & -0.8 & 0 & 0.667 & 0.054 & 0 \\ 0 & 0 & 0 & 0 & 0.798 & 0 \end{bmatrix}, R = \begin{bmatrix} 0.621 & 0 & 0 \\ 0 & 0.421 & 0 \\ 0 & 0 & 0.143 \end{bmatrix} \qquad (10)$$

Then, the state feedback matrix is computed as follows:

$$K = \begin{bmatrix} -0.3477 & 0.805 & -0.8403 & -1.6266 & -0.1734 & -0.1620 \\ 1.0778 & -0.2552 & 0.1345 & 0.4379 & 0.1669 & 1.5443 \\ -1.0942 & 0.4669 & -0.0395 & -0.8032 & -0.4046 & -0.679 \end{bmatrix} \qquad (11)$$

The trajectory of the aircraft landing is shown in 'Figure 3'. The integral absolute error (IAE) value is 32.448. The results of applying the GA and PSO methods are summarized as follows:

The parameters of GA method set as range at [-5 5], population size= 100, generation number= 60, crossover rate= 0.96, mutation rate= 0.01, elitism number= 3.

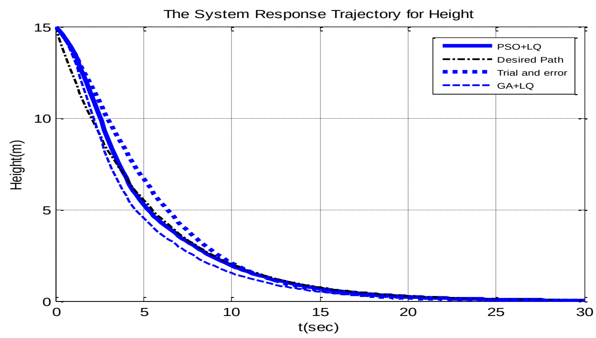

Figure 3: The aircraft landing system time response

The weighting matrix parameters and state feedback matrix using GA are obtained as below:

$$Q = \begin{bmatrix} 0.8576 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.2632 & 0 & -0.5673 & -0.9422 & 0 \\ 0 & 0 & 0.3183 & 0 & 0 & 0 \\ 0 & -0.5673 & 0 & 0.1033 & 0.248 & 0 \\ 0 & -0.9422 & 0 & 0.248 & 0.3585 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.9617 \end{bmatrix} \qquad (12)$$

$$R = \begin{bmatrix} 0.1749 & 0 & 0 \\ 0 & 0.3625 & 0 \\ 0 & 0 & 0.1477 \end{bmatrix} \qquad (13)$$

$$K = \begin{bmatrix} -0.7012 & 1.5938 & -0.9416 & -2.8291 & -0.366 & -0.3625 \\ 1.3722 & -0.5417 & 0.2795 & 1.014 & 0.4719 & 1.4732 \\ -0.6538 & 0.5989 & -0.1836 & -1.1315 & -0.5803 & -0.407 \end{bmatrix} \qquad (14)$$

Also, the weighting matrices and state feedback matrix using PSO method are obtained as follows:

$$Q = \begin{bmatrix} 0.7568 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4415 & 0 & -0.7114 & -0.9582 & 0 \\ 0 & 0 & 0.2555 & 0 & 0 & 0 \\ 0 & -0.7114 & 0 & 0.1109 & 0.3899 & 0 \\ 0 & -0.9582 & 0 & 0.3899 & 0.2494 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.2334 \end{bmatrix} \qquad (15)$$

$$R = \begin{bmatrix} 0.5213 & 0 & 0 \\ 0 & 0.3814 & 0 \\ 0 & 0 & 0.281 \end{bmatrix} \qquad (16)$$

$$K = \begin{bmatrix} -0.7012 & 1.5938 & -0.9416 & -2.8291 & -0.366 & -0.3625 \\ 1.3722 & -0.5417 & 0.2795 & 1.014 & 0.4719 & 1.4732 \\ -0.6538 & 0.5989 & -0.1836 & -1.1315 & -0.5803 & -0.407 \end{bmatrix} \qquad (17)$$

The parameters used in PSO algorithm are: population size= 100, acceleration constants values: c_1=1.5, c_2=1.5, search range= [-5, 5] and iteration number= 60.

The IAE values implementing each of the proposed methods are presented in Table 1.

The corresponding system input vector for elevator angle, spoiler angle and throttle value are shown in 'figures 4, 5 and 6'. The comparison of *IAE* values for control inputs using various methods is shown in Table 2.

Table 1: Comparison of Trajectory following IAE values

Trajectory Following	*Trial & Error*	*GA+LQ*	*PSO+LQ*
IAE	32.448	22.214	10.406

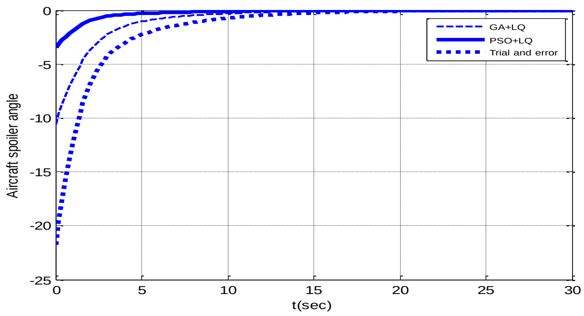

Figure 4: The aircraft landing system spoiler angle

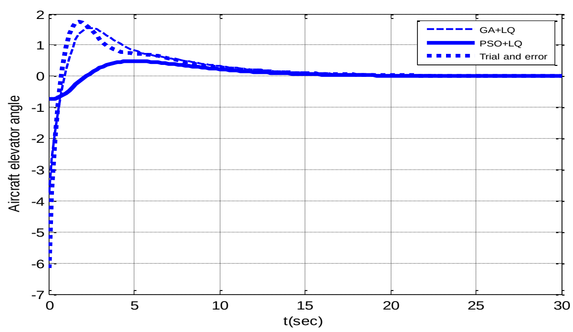

Figure 5: The aircraft landing system elevator angle

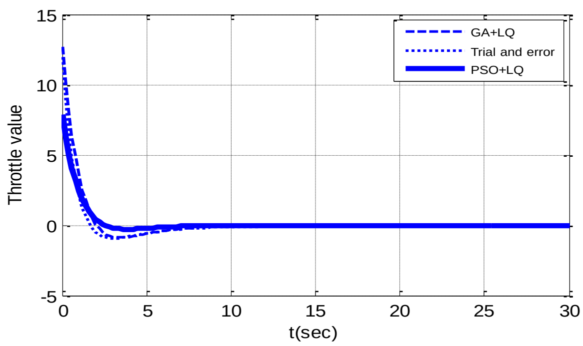

Figure 6: The aircraft landing system throttle value

Table 2: Comparison of IAE values for control inputs

IAE	*Elevator angle*	*Throttle value*	*Spoiler angle*
Trial & Error	28.305	71.236	124.598
GA+LQ	27.2	62.236	64.19
PSO+LQ	11.852	39.824	22.621

V. CONCLUSION

In this paper we present a Particle Swarm Optimization (PSO) algorithm as an intelligent procedure for designing of optimal controller. Superior features of PSO method are fast tuning of the parameters, rapid convergence, less computational burden and capability to avoid from local optima. Simulation results demonstrate that our proposed method is more efficient and robust compared with other heuristic method, i.e., the Genetic Algorithm (GA) method.

REFERENCES

1. Kirk, Donald E., Optimal control theory: an introduction, Carmel/ California, 1937.
2. Kalman, R. E., “When is a linear control system optimal?,” J. Basci Eng. Trans., ASME-86D:51-56,1964.
3. Wang Yaoqing, “The determination of weighting matrices in LQ optimal control system,” ACTA Automatic Sinica, 1992.
4. Gheng-Chung Sung, Gong Chen, “Optimal control systems design associated with genetic algorithm,” Proceedings of 2006 CACS Automatic Control Conference St. John's University, Tamsui, Taiwan, Nov. 10-11, 2006.
5. C. P. Bottura and J. V. da Fonseca Neto, “Rule based decision making unit for eigenstructure assignment via parallel genetic algorithm and LQR design,” Proceeding of the American Control Conference, Chicago, Illinois, Vol. 1, pp. 467- 471, 2000.
6. C. P. Bottura and J. V. da Fonseca Neto, “Parallel eigenstructure assignment via LQR design and genetic algorithms,” Proceeding of the American Control Conference, San Diego, Vol. 4, pp. 2295-2299, 1999.
7. Iraj Hassanzadeh, Saleh Mobayen and Abbas Harifi, “Input-output feedback linearization cascade controller using genetic algorithm for rotary inverted pendulum,” American Journal of Applied Sciences, 5 (10), pp: 1322-1328, 2008.
8. Iraj Hassanzadeh and Saleh Mobayen, “Optimum design of PID controller for 5-bar-linkage manipulator using particle swarm optimization,” Proceeding of the 4th International Symposium on Mechatronics and its Applications (ISMA07), Sharjah, U.A.E. March 26-29, 2007.
9. J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proc. IEEE Int.Conf. Neural Networks, vol. IV, Perth, Australia, 1995, pp. 1942-1948.
10. Eberhart, R. C., and Kermedy, J. “A New Optimizer Using Particles Swarm Theory,” Proc. Sixth International Symposium on Micro Machine and Human Science(Nagoya, Japan), IEEE Service Center, Pkcataway, NJ, 39-43, 1995.

AUTHORS PROFILE

Javad Hamidi, was born in Mashhad, Iran in 1983. He received the B.Sc. degree in Electronic Engineering from university of Guilan, Rasht, Iran in 2006 and M.Sc. degree in Power Electrical Engineering from University of Tabriz, Iran in 2009. His research interests include Power System Harmonics and Power Electronic Converter and Application of modern control and Intelligent Algorithm in Power Systems.