APSO Based Weighting Matrices Selection of LQR Applied to Tracking Control of SIMO System

S. Karthick, Jovitha Jerome, E. Vinodh Kumar and G. Raaja

Abstract This paper employs an adaptive particle swarm optimization (APSO) algorithm to solve the weighting matrices selection problem of linear quadratic regulator (LQR). One of the important challenges in the design of LQR for real time applications is the optimal choice state and input weighting matrices (Q and R), which play a vital role in determining the performance and optimality of the controller. Commonly, trial and error approach is employed for selecting the weighting matrices, which not only burdens the design but also results in non-optimal response. Hence, to choose the elements of Q and R matrices optimally, an APSO algorithm is formulated and applied for tracking control of inverted pendulum. One of the notable changes introduced in the APSO over conventional PSO is that an adaptive inertia weight parameter (AIWP) is incorporated in the velocity update equation of PSO to increase the convergence rate of PSO. The efficacy of the APSO tuned LQR is compared with that of the PSO tuned LQR. Statistical measures computed for the optimization algorithms to assess the consistency and accuracy prove that the precision and repeatability of APSO is better than those of the conventional PSO.

Keywords APSO · LQR · Inverted pendulum · Riccati equation · Tracking control

S. Karthick (✉) · J. Jerome · G. Raaja
Department of Instrumentation and Control Systems Engineering,
PSG College of Technology, Coimbatore 641004, India
e-mail: skpattitude@gmail.com

J. Jerome
e-mail: jovithajerome@gmail.com

G. Raaja
e-mail: raaja1393@gmail.com

E. Vinodh Kumar
School of Electrical Engineering, VIT University, Vellore 632014, India
e-mail: vinothmepsg@gmail.com

A. Nagar et al. (eds.), *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics*, Smart Innovation, Systems and Technologies 43, DOI 10.1007/978-81-322-2538-6_2

1 Introduction

Linear Quadratic Regulator, a corner stone of modern optimal control, has attracted considerable attention in the recent years due to its inherent robustness and stability properties [1]. A minimum phase margin of $(-60^{\circ}, 60^{\circ})$ and a gain margin of $(-6, \infty)$ db provided by LQR enable the system to yield satisfactory response even during the small perturbations. Moreover, by minimizing a quadratic cost function which consists of two penalty matrices, namely Q and R matrices, LQR yields an optimal response between the control input and speed of response. Hence, the LQR techniques have been successfully applied to a large number of complex systems such as vibration control system [2], fuel cell systems [3] and aircraft [4]. Nevertheless, one of the major issues of LQR design for real time applications is the choice of Q and R weighting matrices. Even though, the performance of LQR is highly dependent on the elements of Q and R matrices, conventionally the matrices have been tuned either based on the designer's experience or via trial and error approach. Such approach is not only tedious but also time consuming. Hence, in this paper the conventional LQR design problem is reformulated as an optimization problem and solved using particle swarm optimization algorithm.

In literature, several results have been reported on PSO based state feedback controller design. For instance, in [5] selection of weighting matrices of LQR controller for tracking control of inverted pendulum has been solved using PSO. In [6] the performances of GA and PSO for FACTS based controller design have been assessed and reported that both the convergence and time consumption of PSO are less than those of the GA based feedback controller design. PSO based variable feedback gain control design for automatic fighter tracking problems have been investigated in [7] and it has been reported that PSO based LQR design yields better tracking response than the LMI based methods. However, the standard PSO has two important undesirable dynamical properties that degrade its exploration abilities. One of the most important problems is the premature convergence. Due to the rapid convergence and diversity loss of the swarm, the particles tend to be trapped in the local optima solution when solving multimodal tasks. The second problem is the ability of the PSO to balance between global exploration and local search exploitation. Overemphasize of the global exploration prevents the convergence speed of swarm, while too much search exploitation causes the premature convergence of swarm. These limitations have imposed constraints on the wider applications of the PSO to real world problems [8]. Hence, to better the convergence rate and speed of conventional PSO, we propose an adaptive PSO, whose inertia weight is varied adaptively according to the particle's success rate. The key aspect of the proposed APSO is that an adaptive inertia weight parameter (AIWP), whose weights are varied adaptively according to the nearness of the particles towards the optimal solution, is introduced in the velocity update equation of conventional PSO to accelerate the convergence of the algorithm. To assess the performance of the

APSO based LQR control strategy, simulation studies have been carried out on a benchmark inverted pendulum, which is a typical single input multi output system (input: motor voltage, output: cart position and pendulum angle).

2 Problem Formulation

Consider a linear time invariant (LTI) multivariable system,

$$\dot{X}(t) = AX(t) + Bu(t) \tag{1}$$

$$Y(t) = CX(t) + Du(t) \tag{2}$$

The conventional LQR design problem is to compute the optimal control input u^* by minimizing the following quadratic cost function.

$$J(u^*) = \frac{1}{2}\int_0^\infty \left[X^T(t)QX(t) + u^T(t)Ru(t)\right]dt \tag{3}$$

where $Q = Q^T$ is a positive semi definite matrix and $R = R^T$ is a positive definite matrix. By solving the following Lagrange multiplier optimization technique, the optimal state feedback gain matrix (K) can be computed.

$$K = R^{-1}B^T P \tag{4}$$

where P is the solution of following ARE.

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{5}$$

The elements of Q and R matrices play a vital role in determining the penalty on system states and control input when the system deviates from the equilibrium position. Normally, the Q and R matrices are chosen as diagonal matrices such that the quadratic performance index is a weighted integral of squared error. The sizes of Q and R matrices depend on the number of state variables and input variables respectively. As an alternate to conventional trial and error based manual tuning of these weighting matrices, in the following section, a bio-inspired evolutionary algorithm, an adaptive PSO, has been incorporated in the LQR control strategy for the optimal selection of Q and R.

3 Adaptive PSO

In the last decade, several variants of PSO have been put forward to enhance the performance of conventional PSO. All the proposed variations are mainly to improve the convergence and exploration-exploitation capabilities of PSO. One of the variations incorporated in the PSO is the use of inertia weight parameter to accelerate the convergence of particles towards optimum value. As the inertia weight not only determines the contribution rate of a particle's previous velocity to its current velocity but also yields the required momentum for the particles to move across the solution space, it is important to control the inertia weight to strike a balance between the global search and local exploitation. The larger value of inertia weight concentrates more on global search, while the smaller inertia weight focuses highly on fine tuning the current search space. A comprehensive survey on the use of inertia weight schemes in PSO algorithms is given in [9]. In this paper, we extend the idea of adaptive inertia weight strategy to solve the LQR optimization problem.

To implement an adaptive inertia weight strategy, it is important to evaluate the position of the swarm during every iteration step. Hence, the success percentage (SP) of particles is used to update the velocity adaptively. Large value of SP indicates that the particles have reached the best value and the particles are slowly progressing towards the optimum, whereas a small value of SP implies that the particles are fluctuating around the optimum value with very less improvement. Hence, the success rate can be used to modify the inertia weight adaptively. If the fitness of the current iteration is less than that of the previous iteration the success count (SC) is set to 1, else it is set to zero. Computing the ratio of the SC to the number of iterations, the SP value is computed and used to arrive at the adaptive inertia weight parameter (AIWP) as given below. Table 1 gives the pseudo code of an adaptive PSO algorithm.

$$w(t) = (w_{\max} - w_{\min})SP + w_{\min} \tag{6}$$

Table 1 APSO pseudo code

Randomly initialize Particle swarm, minimum and maximum values of inertia weight ($w_{\min}, w_{\max}$)
for i <=100
Set Success Count (SC) = 0
Evaluate the fitness of particle swarm using $f = ISE = \int e^2(t)dt$
for i = 1 to 30
if $f < f_{pbest_i}$
SC = SC + 1
$f_{pbest_i} \leftarrow f$
$x_{pbest_i} \leftarrow x_i$
end if
if $f < f_{gbest_i}$
$f_{gbest_i} \leftarrow f$
$x_{gbest_i} \leftarrow x_i$
end if
for d = 1 to dimensions
$v_i^d(t+1) = w * v_i^d(t) + c_1 * rand_1 \left(pbest_i^d(t) - x_i^d(t)\right) + c_2 * rand_2 * \left(pbest_g^d(t) - x_i^d(t)\right)$

(continued)

Table 1 (continued)

$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1)$
end for
PS = SC/100;
$w = (w_{\max} - w_{\min}) * SP + w_{\min}$
end for
end for

4 Single Inverted Pendulum

Single inverted pendulum is used as a typical benchmark system to evaluate effectiveness of various control schemes due to its highly nonlinear and inherently unstable properties. It consists of a DC motor and a pendulum, which is attached to the shaft of the motor. Two encoders are used to measure the position of the cart and the angle of the pendulum. Figure 1 shows the schematic diagram of a single inverted pendulum.

Two control schemes, namely swing up control and stabilization control, are used to meet the control objective of maintaining the pendulum angle at zero degree while the cart tracks the reference trajectory. The stabilization control is implemented using LQR due to the practical limitation on control input (motor voltage) given to the cart system. Using Euler-Lagrangian energy based approach the nonlinear equation of motion of pendulum can be written as

$$\left(M_c + M_p\right)\ddot{x}_c(t) + B_{eq}\dot{x}_c(t) - \left(M_p l_p \cos(\alpha(t))\right)\ddot{\alpha}(t) + M_p l_p \sin(\alpha(t))\dot{\alpha}^2(t) = F_c(t) \tag{7}$$

and

$$-M_p l_p \cos(\alpha(t))\,\ddot{x}_c(t) + \left(I_p + M_p l_p^2\right)\ddot{\alpha}(t) + B_p\dot{\alpha}(t) - M_p g l_p \sin(\alpha(t)) = 0 \tag{8}$$

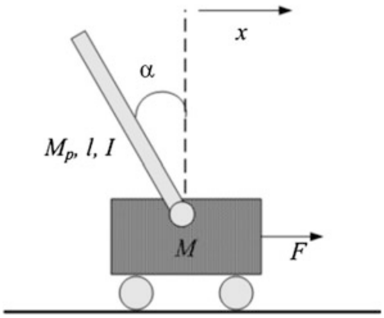

Fig. 1 Schematic diagram of single inverted pendulum

To obtain the state model, four variables namely, cart position, pendulum angle, cart velocity and pendulum velocity are taken as state variables and the state space model is obtained by linearizing the model around the equilibrium point ($\sin(\alpha) \cong \alpha$, $\cos(\alpha) \cong 1$). The following numerical state space model of inverted pendulum system is borrowed from [10] for controller design.

$$\begin{bmatrix} \dot{x}_c \\ \dot{\alpha} \\ \ddot{x}_c \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 2.2643 & -15.8866 & -0.0073 \\ 0 & 27.8203 & -36.6044 & -0.0896 \end{bmatrix} \begin{bmatrix} x_c \\ \alpha \\ \dot{x}_c \\ \dot{\alpha} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 2.2772 \\ 5.2470 \end{bmatrix} u \tag{9}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ \alpha \\ \dot{x}_c \\ \dot{\alpha} \end{bmatrix} \tag{10}$$

5 Results and Discussion

The APSO based LQR tracking control algorithm is implemented in MATLAB 2013b. Table 2 gives the parameters used for PSO and APSO algorithms. The dimension of the optimization algorithms are chosen to be 3 as the number of variables to be optimized in the LQR design is 3 (q_{11}, q_{22} and r). Moreover, the number of iterations, particle size and cognitive acceleration constants in both PSO and APSO are same except the inertia weight. In case of conventional PSO inertia weight is linearly varied, whereas in APSO the inertia weight is adaptively varied according to the particle's success rate as given in (6). According to the fitness function ISE, the optimization algorithms are executed for the specified number of iterations and the global best of the particles, the weights of LQR, are obtained. Figure 2 illustrates the fitness function of both PSO and APSO algorithms.

Table 2 Parameters of PSO and APSO algorithms

Parameters	PSO	APSO
No. of population (N)	30	30
No. of iterations (i)	100	100
Dimensions (d)	3	3
C_1	0.9	0.9
C_2	1.2	1.2
Inertia weight (w)	0.9	AIWP

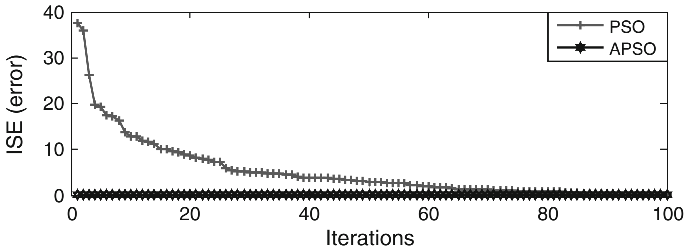

Fig. 2 Fitness function of PSO and APSO

From Table 3, it can be inferred that the minimum fitness function of APSO is less than that of the PSO, which accentuates that the accuracy of the APSO is better than that of PSO. Moreover, the convergence speed of APSO is faster than that of PSO. Figure 3 shows the surface plot of the optimization algorithms. It can be noted

Table 3 Statistical analysis of PSO and APSO

Statistical parameter	PSO	APSO
Mean (μ)	0.1011	0.0316
Standard deviation (σ)	0.2123	0.0367
Minimum (m_x)	0.00032	0.0020
Maximum (M_x)	0.6962	0.1122
Range (R)	0.6942	0.1119

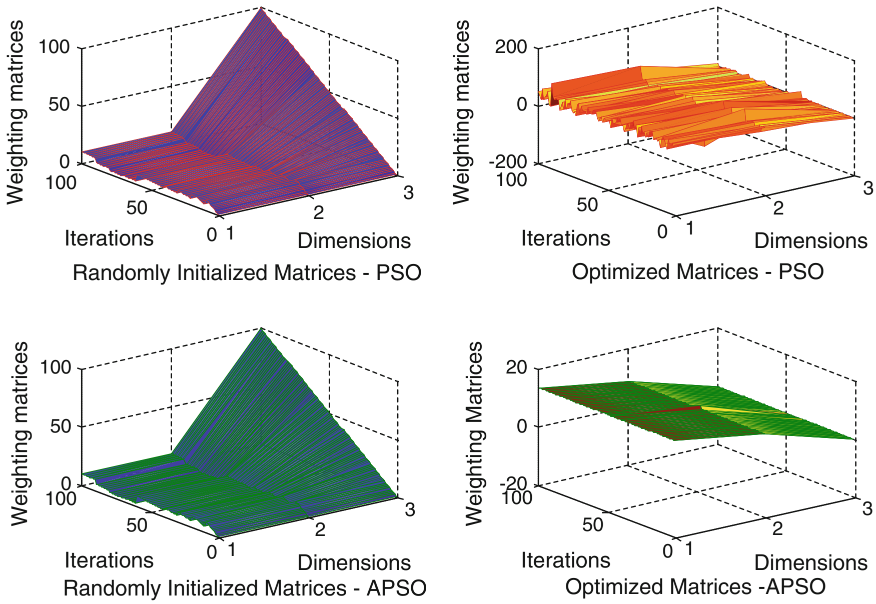

Fig. 3 Surface plots of PSO and APSO

Table 4 Weighting matrices and state feedback controller gains of PSO and APSO

Optimization algorithm	Weighting matrices	Controller gain
PSO	$Q = diag[31.88 \quad 8.97 \quad 0 \quad 0]$ $R = 0.22$	$K = [-82.61 \quad 145.47 \quad -53.16 \quad 18.85]$
APSO	$Q = diag[13.65 \quad 8.92 \quad 0 \quad 0]$ $R = 0.002$	$K = [-82.61 \quad 145.47 \quad -53.16 \quad 18.85]$

that the smoothness of the convergence is significantly better in APSO compared to PSO. Table 4 gives the corresponding Q and R matrices and controller gain of LQR obtained using the PSO and APSO algorithms.

5.1 Trajectory Tracking Response

To assess the tracking response of the APSO tuned LQR controller framework, a square test signal of 0.05 Hz with 40 cm (peak to peak) displacement amplitude is given and the response is illustrated in Fig. 4.

From Table 5, which gives the time domain specifications of the cart position response, it is worth to note that both the settling time and the dead time of the APSO tuned LQR is better than those of PSO tuned LQR. The pendulum angle response and its corresponding motor voltage are shown in Fig. 5. Table 6 gives the deviation and convergence time of pendulum angular response. The convergence time of APSO based pendulum angular response is faster than that of PSO tuned pendulum angle response.

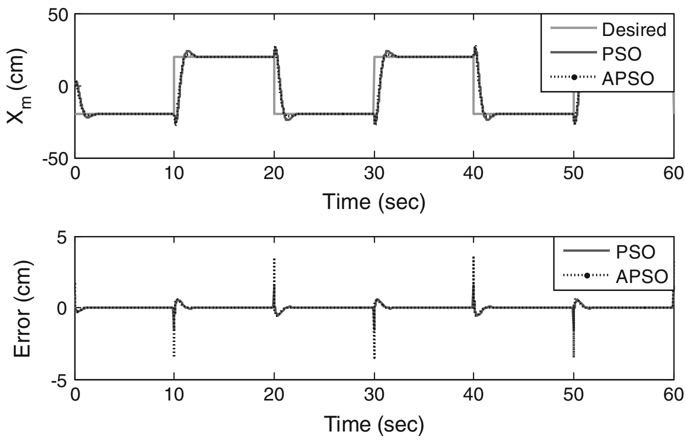

Fig. 4 Cart position and tracking error for square trajectory

Table 5 Comparison of cart position response

Optimization method	Time domain parameters			Performance index ISE
	t_d	t_s	$\%M_p$	
PSO	0.4	3.5	20	0.412
APSO	0.25	3	10	0.376

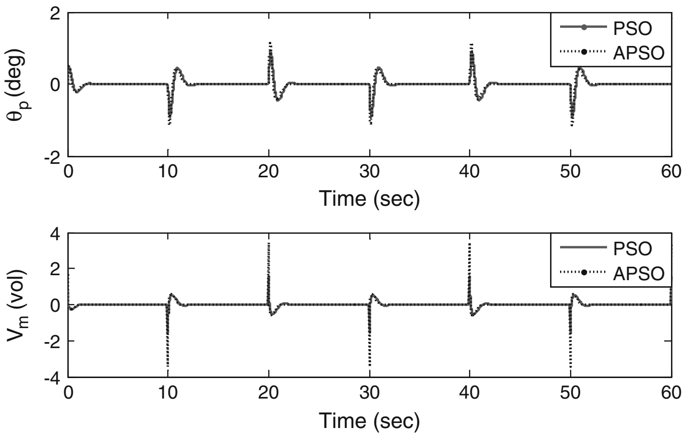

Fig. 5 Pendulum angle and motor voltage for square trajectory

Table 6 Pendulum angle response

Optimization method	Angle deviation (°)	Convergence time (s)
PSO	1.3	3.2
APSO	1.2	3.0

6 Conclusions

In this paper, the weight selection problem of LQR has been solved using the APSO algorithm and the efficacy of the controller has been tested on a benchmark inverted pendulum. To increase the convergence speed and precision of the conventional PSO, an AIWP has been introduced in the velocity update equation of PSO. Statistical measures calculated for the optimization algorithms prove that the introduction of AIWP significantly increases both the accuracy and consistency of the conventional PSO. Moreover, the trajectory tracking response of inverted pendulum accentuate that compare to PSO tuned LQR, the APSO tuned LQR controller framework can result in not only improved tracking response but also reduced tracking error.

References

1. Wang, L., Ni, H., Zhou, W., Pardalos, P.M., Fang, J., Fei, M.: MBPOA-based LQR controller and its application to the double-parallel inverted pendulum system. Eng. Appl. Artif. Intell. **36**, 262–268 (2014)
2. Ang, K.K., Wang, S.Y., Quek, S.T.: Weighted energy linear quadratic regulator vibration control of piezoelectric composite plates. J. Smart Mater. Struct. **11**(1), 98–106 (2002)
3. Niknezhadi, A., Miguel, A.F., Kunusch, C., Carlos, O.M.: Design and implementation of LQR/LQG strategies for oxygen stoichiometry control in PEM fuel cells based systems. J. Power Sources **196**(9), 4277–4282 (2011)
4. Usta, M.A., Akyazi, O., Akpinar, A.S.: Aircraft roll control system using LQR and fuzzy logic controller. In: IEEE Conference on Innovations in Intelligent Systems and Applications (INISTA), pp. 223–227. Istanbul (2011)
5. Solihin, M.I., Akmeliawati, R.: Particle swam optimization for stabilizing controller of a self-erecting linear inverted pendulum. Int. J. Electr. Electron. Syst. Res. **3**, 410–415 (2010)
6. Panda, S., Padhy, N.P.: Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design. Appl. Soft Comput. **8**(4), 1418–1427 (2008)
7. Tsai, S.J., Huo, C.L., Yang, Y.K., Sun, T.Y.: Variable feedback gain control design based on particle swarm optimizer for automatic fighter tracking problems. Appl. Soft Comput. **13**, 58–75 (2013)
8. Lim, W.H., Isa, N.A.M.: Teaching and peer-learning particle swarm optimization. Appl. Soft Comput. **18**, 39–58 (2014)
9. Nickabadi, A., Ebadzadeh, M.M., Safabakhsh, R.: A novel particle swarm optimization algorithm with adaptive inertia weight. Appl. Soft Comput. **11**, 3658–3670 (2011)
10. Vinodh Kumar, E., Jovitha, J.: Stabilizing controller design for self erecting single inverted pendulum using robust LQR. Aust. J. Basic Appl. Sci. **7**(7), 494–504 (2013)