

Optimal Control Approach for Robot System Using LQG Technique

Ibrahim Khalaf Mohammed, Mohanad N. Noaman*

Department of Systems and Control Engineering, College of Electronics Engineering, Ninevah University, Mosul 4100, Iraq

Corresponding Author Email: mohanad.noaman@uoninevah.edu.iq



<https://doi.org/10.18280/jesa.550513>

ABSTRACT

Received: 13 July 2022

Accepted: 6 October 2022

Keywords:

robot system, linear quadratic gaussian, Kalman filter, linear quadratic regulator, particle swarm optimization, stabilization

A two-wheeled self-balancing robot system bases on the physical problem of an inverted pendulum. Stabilization of this type of mobile robot requires applying an active control approach. This paper proposes an efficient Linear Quadratic Gaussian (LQG) optimal control for the two-wheeled robot system. The LQG (a combination of a Kalman Filter (KF) and Linear Quadratic Regulator (LQR)) controller is designed to stabilize the robot while reducing the effect of the process and measurement noises on its performance. The LQG controller parameters (elements of state and control weighting matrices of the LQR and KF) are optimally tuned using the Particle Swarm Optimization (PSO) optimization method. The robot stabilization scheme is simulated utilizing MATLAB software to validate the proposed PSO-LQG controller system. The effectiveness of the proposed controller is validated based on the control criteria parameters, which are rise time, settling time, maximum overshoot, and steady-state error. The results prove that the proposed PSO-LQG controller can give very good movement performance in terms of both transient and steady-state responses.

1. INTRODUCTION

Linear Quadratic Gaussian (LQG) optimal control technique is one of the modern controls which is based on Kalman Filter (KF) in combination with a Linear Quadratic Regulator (LQR) controller. LQG controllers can be successfully applied to both Linear Time-Variant (LTV) systems and Linear Time-Invariant (LTI) systems [1]. Typically, LQG controller technique is employed effectively to control dynamic systems with disturbances. This controller approach is used to stabilize systems that are subject to process noise and measurement noise. These disturbance sources can affect in the performance of schemes. In the LQG controller design process, the designers try to tune the state and control weighting matrices of LQR controller (Q , R), and Kalman Filter (Q_e , R_e) so that the performance requirements are achieved. Many researchers attempted to find a way to tune the LQG controller parameters. They presented either trial and error or complicated procedures to set the controller parameters [2]. Trial and error adjusting approach is not an optimum tuning method for controller elements as it takes a lot of effort and consumes more time. The quality and the control of the manual trial and error method rely on the tuning skill of the control engineer and his knowledge in control processing. Output control depends in addition, there is no guarantee that the obtained controller parameters are the best. Therefore, adopting computational tuning algorithms to find best values for LQG controller parameters has attracted attention of many researchers during the last decades. Prodic, and Maksimovic utilized classic adjusting approaches to optimize the performance of the closed-loop control system. Classical frequency response techniques have been used such as Ziegler-Nichols and root locus-type methods to find best values of pole placement and PID controller gain parameters.

However, the presented tuning methods are not able to find the best solution for some control problems due to long calculating process time or early convergence problem. Therefore, more powerful intelligent optimization methods have been presented by researchers to find best global solution for many control problems in different applications fields such as Genetic Algorithm (GA) [3-5], Bacteria Foraging Optimization Algorithm (BFOA) [5], Big Bang-Big Crunch (BBBC) algorithm [6], Particle Swarm Optimization (PSO) algorithm [3, 7], Particle Swarm Inspired Evolutionary Algorithm (PS-IEA) [8], Artificial Bee Colony (ABC) [9]. GA tuning method cannot be employed to find a global solution for complex optimization schemes due to stuck problem in the calculation process, solution divergence and large number of iterations. In this study, PSO tuning method is used to optimize the proposed controller system due to its ability to find global solution for simple and complex control problems. PSO, which was first introduced by Kennedy and Eberhart in 1995, is an intelligent exploratory approach inspired by the innate behavior for individuals and birds' swarms [10]. It composes of number of particles, which represent a population of candidate solutions of control problems. Each particle has a specific position and velocity vectors, which are adopted to distinguish each from other. The particles swarms follow their trajectory towards the global optimum solution based on Newtonian mechanics.

During the last decades, different classic and optimal controllers are designed by numerous researchers to actuate and stabilize various two-wheels robot systems. LQR technique based optimal control scheme is presented to stabilize a noiseless two-wheels robot system [11]. The modeling of the system is derived based on the separation between the dynamics of wheels and pendulum. The proposed LQR controller is optimized using the PSO algorithm. Jung

and Kim [12] proposed a nonlinear adaptive controller to balance a two-wheeled robot system. They used Neural Network control and two PID controllers to control position and angle of the robot system. The forward and backward actuation performances of the mobile cart based on the Neural Network (NN) technique are compared with that of PID controllers under one-dimension desired trajectory and presented. The actuation results of the system have shown the ability of the proposed Neural Networks approach to balance and move the robot scheme in one dimension effectively.

Sun et al. [13] proposed a balance control method for a two wheeled personal robot system using LQR controller and neural networks. In the presented method, a NN is used to perform the self-tuning process of the control system. Wu and Zhang [14] designed a pole placement state feedback controller and fuzzy logic controller for dual-wheels robot systems. The simulation results have shown that the robot system based on the fuzzy logic control method can realize better dynamic performance compared with the pole placement technique. A self-balance control approach is proposed for a two-wheeled Segway robot system using LQR controller technique [5]. The proposed controller system is optimized utilizing GA and BFOA. The simulation results have shown that compared with the GA-LQR controller, the optimized BFOA-LQR controller achieved best tracking performance and succeeded in realizing the anticipated control goals of the robot system. Ali and Shareef [15] introduced design and simulation a robust control system for two-wheeled pendulum model. Two controller techniques are adopted to control the pendulum system. Full state feedback H_2 controller is used to stabilize the pendulum to upright position and H_∞ controller to realize more robust actuation performance. Nasir et al. [16] proposed a classic PID controller and an optimal LQR controller to control both of position and angle of a two-wheeled robot system. The two controllers are simulated and their actuation performance are presented and compared based on transient response parameters. However, the proposed controllers for robot actuation systems have some drawbacks, as some of the presented controllers are tuned manually or by using classic tuning approaches such as the Zeiglar-Nichols method. In addition, some of the proposed controller techniques are classic, for instance, PID and fuzzy logic, which are not robust enough to keep the balance and stability of the robot scheme. Moreover, the presented controllers are designed and implemented based on the idea that the robot system is free of working noise that is considered as an unrealistic case.

In this research, a LQG controller is proposed to stabilize a two wheeled robot system. The proposed LQG controller is optimized by utilizing the PSO tuning algorithm that is used to find best values for its weighting matrices elements. Practically, the robot system is subject to the effect of the predicted working circumstances such as noise and uncertainties problems. In this study, two working disturbances, which are process noise and measurement noise, are considered in design and simulation the controller of personal transporter system.

2. ROBOT MATHEMATICAL MODEL

The body of two wheeled self-balancing robots as shown in Figure 1 is mainly constituted of chassis, which is based on one degree of freedom inverted pendulum, and the left and

right wheel that are rotated by DC motor. The motor is followed by encoders that used to provide the control system by position readings of the robot. The handlebar is used to steer the two wheels robot system. Motion control of the robot system is fixed on the handlebar unit. The body axle of the robot system is upside down on the top of the gravity center, in order to keep the balanced state of the single person cart system. The structure of the personal robot system is composed of electrical subsystem and mechanical subsystem. Figure 2 illustrates the electric model of the robot DC motor.



Figure 1. Platform of dual wheeled robot system

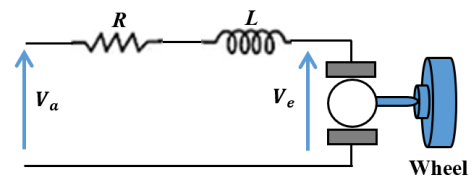


Figure 2. Electric circuit of robot system

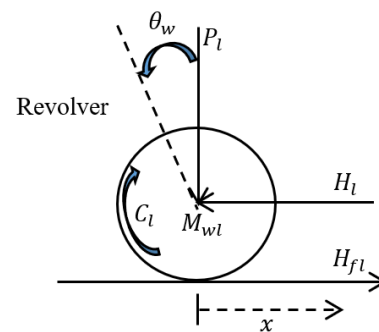


Figure 3. The robot's left wheel free body diagram

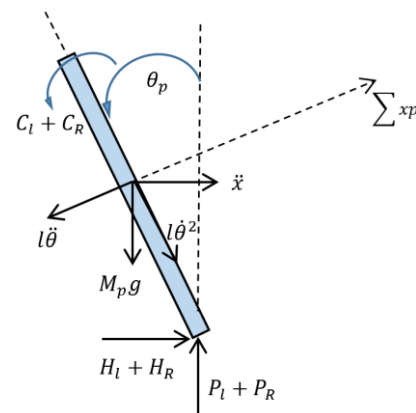


Figure 4. Force analysis robot's chart

Figures 3 and 4 present a front view of the robot system and force analysis diagram of its left wheel [5]. In this application, the two wheels are assumed to have the same physical dimensions and moment of inertia. Using Newton's law on the basis of stress analysis, and considering the voltage and torque relationships of the DC motor, the robot state space equation is given by [5, 11]:

$$\begin{bmatrix} \dot{x} \\ \dot{\dot{x}} \\ \dot{\varphi} \\ \dot{\dot{\varphi}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{2K_m K_e (M_P I_r - I_P - M_P I^2)}{R r^2 \alpha} & \frac{M_P^2 g I^2}{\alpha} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{2K_m K_e (r\beta - M_P I)}{R r^2 \alpha} & \frac{M_P g I \beta}{\alpha} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0.4891 \\ 0 \\ 2.3634 \end{bmatrix} V_a \quad (1)$$

where, $\beta = \left(2M_w + \frac{2I_w}{r^2} + M_p\right)$ and $\alpha = \left(I_p \beta + 2M_p I^2 \left(M_w + \frac{I_w}{r^2}\right)\right)$. Table 1 presents the symbols and the actual values of the robot system parameters [14]. Based on Table 1, the robot state Eq. (1) is:

$$\begin{bmatrix} \dot{x} \\ \dot{\dot{x}} \\ \dot{\varphi} \\ \dot{\dot{\varphi}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.1038 & 25.5862 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -0.5015 & 238.468 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0.4891 \\ 0 \\ 2.3634 \end{bmatrix} V_a \quad (2)$$

And the robot output equation is given by [5]:

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \varphi \\ \dot{\varphi} \end{bmatrix} \quad (3)$$

where, x and \dot{x} are the displacement and velocity of the wheel in the x -axis respectively, φ is angle measured from vertical upward direction of the wheels, and $\dot{\varphi}$ is angular velocity of the two wheels [12].

Table 1. Physical and electrical parameters of the robot system

No.	Symbol	Quantity	Value (unit)
1	I_w	Moment of inertia of the robot's wheels	0.0032 kgm ²
2	I_P	Moment of inertia of the robot's chassis	0.0038 kgm ²
3	M_P	Mass of the robot's chassis	0.52 kg
4	M_w	Mass of the wheel connected to both sides of the robot	0.02 kg
5	I	Length to the body's center of mass	0.16 m
6	K_m	Torque constant	0.0136 Nm/A
7	K_e	Back emf constant	0.01375 V/(rad/s)
8	R	Nominal terminal resistance	1.6 Ω
9	g	Gravity constant	9.8 ms ⁻²
10	r	Radius of wheel	0.025 m

3. OPTIMIZED CONTROL METHOD

In this section, the theoretical background of the proposed LQG controller employed to balance the two-wheels robot system is presented. The theory and procedure of the PSO algorithm used to find best values for the LQG parameters are also introduced.

3.1 Controller technique

In this study, the stabilization system of the mobile robot is based on LQG technique. LQG controller is a full state feedback optimal compensator system based on KF and LQR, which is also called system states observer. The state estimator serves in estimation of the unmeasurable system states. The state and output equations of the noisy Linear Time- Invariant (LIT) scheme is both defined in (3) and (4) respectively:

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t) \quad (4)$$

$$y(t) = Cx(t) + v(t) \quad (5)$$

where, $w(t)$ is the process noise of the robot system while $v(t)$ is the measurement noise of the system output sensor. The general cost function of the optimal control system is given by:

$$J = \int_0^{\infty} [x^T(t)Qx(t) + U^T(t)Ru(t)]dt \quad (6)$$

where, Q and R are the control and input optimal controller weighting matrices, which should be designed to minimize the cost function (6) through applying the following control effort ($u(t)$).

$$u(t) = -Kx(t) = -R^{-1}B^T Px(t) \quad (7)$$

where, K is the LQR gain matrix and P is symmetric and positive semi-definite covariance matrix, which is obtained from solving the following algebraic matrix Riccati Eq. (8).

$$PA + A^T P - PBR^{-1}BP + Q = 0 \quad (8)$$

After designing LQR gain matrix, the state estimation of the system based on process and measurement noises should be discussed. The second component of the LQG controller is KF, which serves as an optimum state's observer to the controller. The observer gain matrix of the LQG controller represented as [8, 17]:

$$K_e = P_o C^T R_o^{-1} \quad (9)$$

and P_o is covariant matrix obtained from the solution of the below algebraic Riccati equation:

$$AP_o + P_o A^T + BQ_o B^T - P_o C^T R_o^{-1} CP_o = 0 \quad (10)$$

3.2 PSO algorithm

PSO is an intelligent tuning algorithm that has been used successfully to find optimum solution for complex control problems. PSO is a type of the modern heuristic algorithms that has been motivated by the behavior of organisms, such as fish schooling, bird flocking, bee searching, and swarm theory [18, 19]. As a naturally inspired and global-optimized tuning algorithm, PSO is characterized by its simple concept, high quality, fast convergence, computationally efficient and easy to implement in practical optimization problems [20, 21]. Unlike the other heuristic tuning algorithms, PSO method has a well-balanced mechanism and flexibility that can enhance and adapt both the global and local search abilities for problem solution [22]. The PSO is a type of evolutionary algorithms,

and its steps are straightforward, begin initially from a random solution and seek for the best global solution of control problem through the iterative approach [23]. Supposing X particles are combined of a group in d dimensional space, in this space, the position and velocity of i^{th} particle are represented by; $x_i=(x_{i1}, x_{i2}, \dots, x_{iD})$, $v_i=(v_{i1}, v_{i2}, \dots, v_{iD})$, $i=1, 2, \dots, m$ respectively, the best position that the i^{th} particle experiences is represented by $P_{best}(i)$ while $g_{best}(i)$ is used to denote the best position that all the particles in the group experience. In each iteration, the status of position and velocity of the whole particle swarm are updated through tracking the optimal value. The updating process is expressed by the following equations [24, 25].

$$v_i(t+1) = w \cdot v_i(t) + C_1 \cdot rand(P_{best}(t) - x_i(t)) + C_2 \cdot rand(g_{best}(t) - x_i(t)) \quad (11)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (12)$$

$$W = W_{max} - \frac{(W_{max} - W_{min}) \cdot iter}{iter_{max}} \quad (13)$$

where, $x_i(t+1)$ and $v_i(t+1)$ denote the position and velocity of the i^{th} particle at $(t+1)$ iteration respectively, C_1 and C_2 represent constants of acceleration named cognitive learning rate and social learning rate respectively, W represents the weighting function, and $rand$ indicates a random real number with a value range from 0 and 1.

4. PSO BASED LQG OPTIMAL CONTROL

The PSO tuning method was mainly used to determine best elements values for LQR and KF weight matrices (Q, R, Q_e) and (R_e), such that an output of the controlled system could follow a desired input efficiently. The block diagram of the closed-loop robot actuation scheme using PSO based LQG controller is shown in Figure 5.

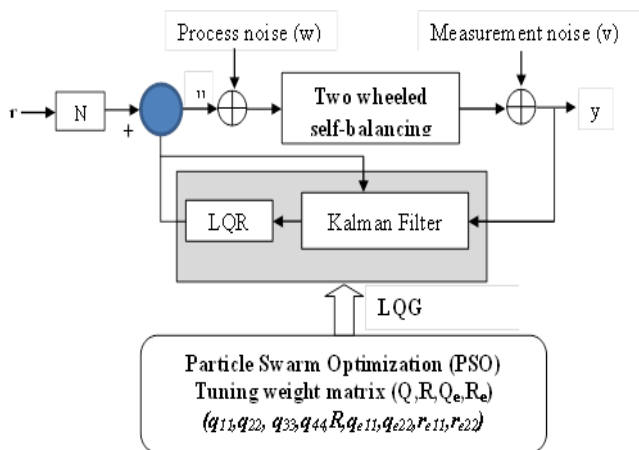


Figure 5. Schematic diagram of LQG controller based on PSO tuning method for robot system

In the procedure of the PSO algorithm, the population is used to define the group and the “individual” is utilized to replace the particle. In this application, the optimal LQG controller parameters (Q, R, Q_e) and (R_e), composed nine elements individual vector ($q_{11}, q_{22}, q_{33}, q_{44}, R, q_{e11}, q_{e22}, r_{e11}, r_{e22}$). These elements are assigned as real values. If a system

has n individuals in a population, then the population dimension becomes $(n \times 9)$. Applying optimum LQG parameters (Q, R, Q_e) and (R_e), enables the controlled system to show a best output response in time domain and minimizes the following cost function, which is formulated based on the standard performance criteria, rise time (t_r) settling time (t_s), maximum overshoot ($\%Mp$) and steady state error (e_{ss}).

$$J = 0.3t_r + 0.3t_s + 0.2M_o + 0.2e_{ss} \quad (14)$$

5. SIMULATION AND RESULTS

In the proposed system, the MATLAB script and Simulink tool are utilized as the simulation environment to validate the proposed PSO-LQG controller for the robot system according to the standard control parameters including $t_r, t_s, \%M_o, e_{ss}$ and $u(t)$. The parameters of the PSO algorithm used to optimize the proposed controller are listed in Table 2. Figures 6 and 7 present convergence elements of LQR and KF weighting matrices (Q, R, Q_e) and (R_e), respectively to best values over 100 iterations. The weighting matrices (Q, R) and (Q_e, R_e) obtained for LQR controller and KF respectively by using PSO algorithm are: $Q = blkdiag(q_{11}, q_{22}, q_{33}, q_{44}) = blkdiag(220, 1.051, 0.001, 0.0093)$, $R = 0.0019$, $Q_e = blkdiag(q_{e11}, q_{e22}) = blkdiag(0.001, 0.2696)$, $R_e = blkdiag(r_{e11}, r_{e22}) = blkdiag(10, 0.8994)$. These parameters of LQG matrices are obtained to effectively enable the output states of the robot following the desired input trajectory while the system control efforts are kept as small as possible. Based on the state equation matrices (A, B) and optimized LQG weighting matrices (Q, R, Q_e) and (R_e), the LQR gain elements were easily calculated through utilizing the MATLAB command $K=lqr(A, B, Q, R)$, as follows: $K=[-469.0416 \ -146.1671 \ 613.9371 \ 52.1916]$, while the observer gain matrix was simply determined in this study with the MATLAB command, $K_e=kalman(A, B, Q_e, R_e)$ and its value is given by: $K_e=[3.55 \ 5.84; \ 0 \ 0; \ 5.84 \ 16.33; \ 0 \ 0]$ whereas the scaling matrix (N) employed to minimize the e_{ss} of the system output, which is calculated using the following expression:

$$N = -(C(sI - (A - BK))^{-1}B)^{-1} \quad (15)$$

and its value is $N=467.267$. In the regulating case, based on the above gain matrices, the time response of the robot system with initial states: $[x \ \dot{x} \ \phi \ \dot{\phi}]^T = [1 \ 0 \ 0 \ 0]^T$ are shown in Figure 8. It can be observed from the regulation response that the robot outputs under action of the controller followed the desired inputs effectively with short rise and settling time of 0.1s and 0.3s respectively and minimal steady state error due to the robustness of the LQG controller to deal with the inherently unstable noisy systems.

Table 2. Parameters of PSO algorithm

Parameter	Value
Population size	20
Iterations number	100
Cognitive component C_1	1.2
Social component C_2	1.2
Maximum speed	10
Maximum inertia weight (W_{max})	0.9
Minimum inertia weight (W_{min})	0.4

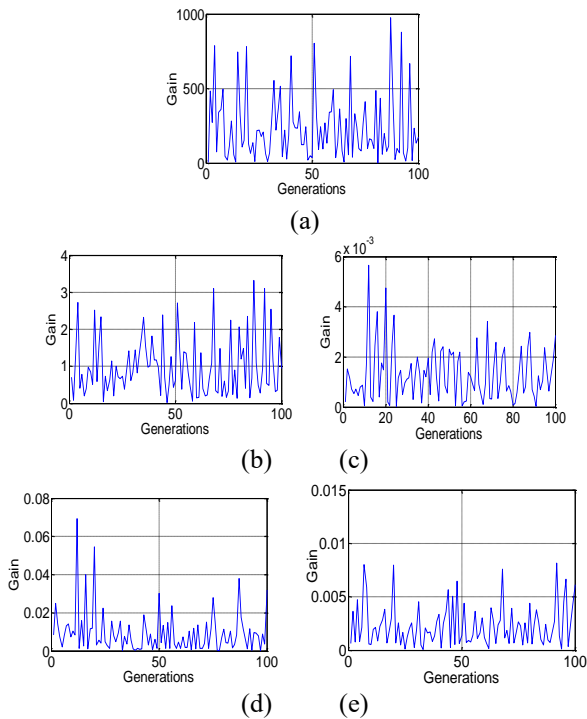


Figure 6. Generation of Q and R elements for LQR controller, (a) q_{11} (b) q_{22} (c) q_{33} (d) q_{44} (e) R

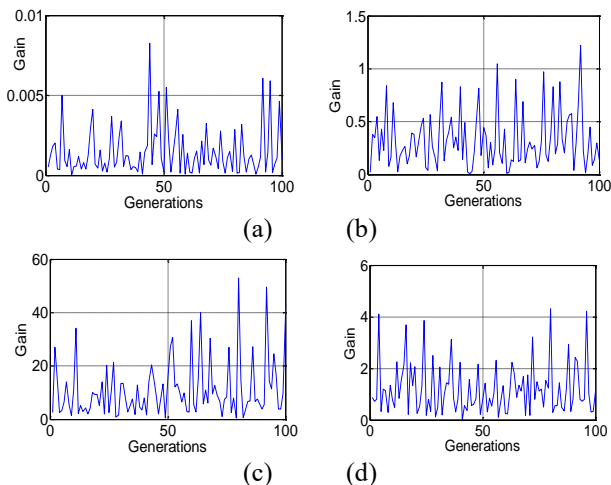


Figure 7. Generation of KF elements (a) q_{e11} (b) q_{e22} (c) r_{e11} (d) r_{e22}

Figure 9 presents the output response of the single seat robot system with zero initial states under desired step input. It is clear from Figure 9 that the PSO-LQG controller forced the robot output states to follow the trajectory of the desired inputs effectively with a transient response t_r of 0.5s, t_s of 0.8s and $M_o=0.1\%$, while steady state error e_{ss} of 0.0001. This response will definitely support the feasibility of applying the proposed LQG controller through its ability to overcome the oscillation problem during the robot navigation and enhance its stability. Figure 10 presents the output response of the robot regulation scheme using PSO-LQG controller with initial states of $[x \dot{x} \phi \dot{\phi}]^T = [1 \ 0 \ 30^\circ \ 0]^T$. It is obvious from the minifigure of Figure 10 that the output of the single seat transporter system under action of the PSO-LQG controller succeeded to follow the trajectory of the desired input effectively with transient response t_r of 0.5s, t_s of 0.8s and M_o of 5%, while steady state error e_{ss} of 0.00012. The reason behind this

response is the non-zero initial value of the robot's states, which requires fast and hard controller action. However, the response of the noisy system is still an acceptable. Consequently, it can be said that the proposed PSO-LQG controller is able to reject the effect of the noises on the performance of the robot system and guide its output states through the desired trajectory input efficiently. Figure 11 shows the control law $u(t)$ response of the PSO-LQG controller for mobile transport system. It can be noted from Figure 11 that the value of input signal required to balance the two-wheeled robot system is within an acceptable range. Based on Figure 10, the time response specifications for the transport system based on the optimized PSO-LQG controller are given in Table 3. From Figures 8-11, it can be shown that the PSO-LQG controller is efficient to guide the robots' output states through the desired input trajectory with a reasonable control effort value.

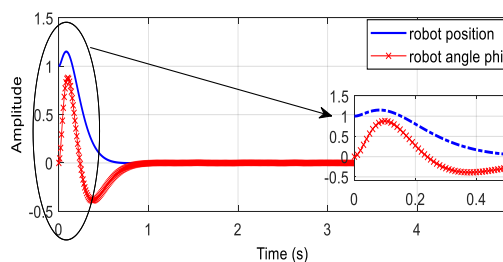


Figure 8. Robot output with initial position of 1

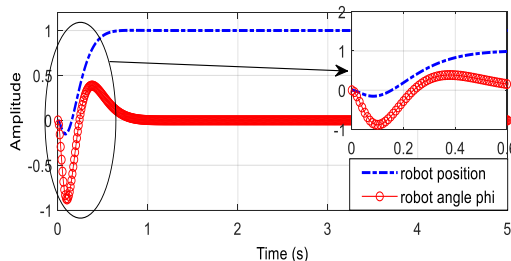


Figure 9. Step response of robot scheme

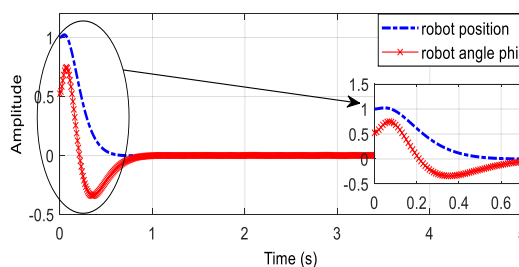


Figure 10. Robot out. with initial states $[1 \ 0 \ 30^\circ \ 0]^T$

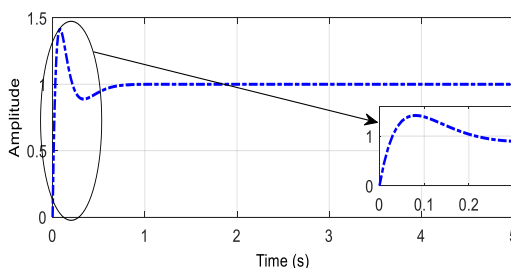


Figure 11. Control law of the robot system

Table 3. Performance characteristics for two wheeled robots

Symbol	$t_r(s)$	$t_s(s)$	$M_o\%$	e_{ss}
Robot Position	0.54	0.27	1.49	0.0025
Robot Angle	0.859	0.12*10 ⁻¹⁶	0	0

6. CONCLUSIONS

This paper has presented a reliable stabilization system to balance a mobile robot scheme with working disturbances by using LQG controller technique. The proposed LQG controller was used to reduce the effect of process noise and measurement noise on the dynamic behavior of mobile robot. The dynamic of the robot system was modelled mathematically and then LQG controller system was designed to stabilize it at desired position and angle. A PSO algorithm has been utilized to obtain best parameters values for LQR controller and KF which are optimizing the performance of the proposed LQG controller. To verify the performance of the proposed controller, the MATLAB has been used to simulate the results. The transient and steady state output response of the robot system evaluated based on the characteristics parameters including maximum overshoot, settling time, rise time and steady state error. The simulation results have proven that the PSO-LQG controller is efficient to stabilize the robot system that is subject to working disturbances efficiently.

REFERENCES

- [1] Saadatmand, M., Gharehpetian, G.B., Kamwa, I., Siano, P., Guerrero, J.M., Haes Alhelou, H. (2021). A survey on FOPID controllers for LFO damping in power systems using synchronous generators, FACTS devices and inverter-based power plants. *Energies*, 14(18): 5983. <https://doi.org/10.3390/en14185983>
- [2] Prodic, A., Maksimovic, D. (2002). Design of a digital PID regulator based on look-up tables for control of high-frequency DC-DC converters. In 2002 IEEE Workshop on Computers in Power Electronics, Proceedings, pp. 18-22. <https://doi.org/10.1109/cipe.2002.1196709>
- [3] Abdulla, A., Mohammed, I., Jasim, A. (2017). Roll control system design using auto tuning LQR technique. *International Journal of Engineering and Innovative Technology*, 6(2): 11-22.
- [4] Mohammed, I.K., Abdulla, A.I. (2020). Elevation, pitch and travel axis stabilization of 3DOF helicopter with hybrid control system by GA-LQR based PID controller. *International Journal of Electrical and Computer Engineering*, 10(2): 1868. <https://doi.org/10.11591/ijece.v10i2.pp1868-1884>
- [5] Mohammed, I.K., Abdulla, A.I. (2020). Balancing a Segway robot using LQR controller based on genetic and bacteria foraging optimization algorithms. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18(5): 2642-2653. <https://doi.org/10.12928/telkomnika.v18i5.14717>
- [6] Gupta, A.K., Kumar, D., Reddy, B.M., Samuel, P. (2017). BBBC based optimization of PI controller parameters for buck converter. In 2017 Innovations in power and advanced computing technologies (i-PACT), pp. 1-6. <https://doi.org/10.1109/ipact.2017.8244983>
- [7] Al-Mahturi, A., Wahid, H. (2017). Optimal tuning of linear quadratic regulator controller using a particle swarm optimization for two-rotor aerodynamical system. *International Journal of Electronics and Communication Engineering*, 11(2): 196-202.
- [8] Srinivasan, D., Seow, T.H. (2003). Particle swarm inspired evolutionary algorithm (PS-EA) for multiobjective optimization problems. In The 2003 Congress on Evolutionary Computation, 2003. CEC'03., 4: 2292-2297. <https://doi.org/10.1109/cec.2003.1299374>
- [9] Mohammed, I.K., Abdulla, A.I. (2018). Design of optimised linear quadratic regulator for capsule endoscopes based on artificial bee colony tuning algorithm. *International Journal for Engineering Modelling*, 31(1-2): 77-98. https://doi.org/10.31534/engmod.2018.1-2.ri.02_vjan
- [10] Mohammed, I., Abdulla, A. (2018). Fractional order PID controller design for speed control DC motor based on artificial Bee colony optimization. *International Journal of Computer Applications*, 179(24): 43-49. <https://doi.org/10.5120/ijca2018916525>
- [11] Kennedy, J., Eberhart, R. (1995). Particle swarm optimization. In Proceedings of ICNN'95-International Conference on Neural Networks, 4: 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>
- [12] Jung, S., Kim, S.S. (2008). Control experiment of a wheel-driven mobile inverted pendulum using neural network. *IEEE Transactions on Control Systems Technology*, 16(2): 297-303. <https://doi.org/10.1109/tcst.2007.903396>
- [13] Sun, C., Lu, T., Yuan, K. (2013). Balance control of two-wheeled self-balancing robot based on Linear Quadratic Regulator and Neural Network. In 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP), pp. 862-867. <https://doi.org/10.1109/icicip.2013.6568193>
- [14] Wu, J., Zhang, W. (2011). Design of fuzzy logic controller for two-wheeled self-balancing robot. In Proceedings of 2011 6th International Forum on Strategic Technology, 2: 1266-1270. <https://doi.org/10.1109/ifost.2011.6021250>
- [15] Ali, H.I., Shareef, Z.A.M. (2017). Robust controller design for two wheeled inverted pendulum system. *Al-Nahrain Journal for Engineering Sciences*, 20(3): 562-569.
- [16] Nasir, A.N.K., Ahmad, M.A., Ismail, R.R. (2010). The control of a highly nonlinear two-wheels balancing robot: A comparative assessment between LQR and PID-PID control schemes. *International Journal of Mechanical and Mechatronics Engineering*, 4(10): 942-947.
- [17] Chrif, L., Kadda, Z.M. (2014). Aircraft control system using LQG and LQR controller with optimal estimation-Kalman filter design. *Procedia Engineering*, 80: 245-257. <https://doi.org/10.1016/j.proeng.2014.09.084>
- [18] Zheng, Y., Zhong, P., Yue, Q. (2016). Double inverted pendulum based on LQG optimal control. In 2016 International Conference on Automatic Control and Information Engineering, pp. 83-87. <https://doi.org/10.2991/icacie-16.2016.20>
- [19] Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39(3): 459-471. <https://doi.org/10.1007/s10898-007-9149-x>

[20] Girdhar, A. (2015). Swarm intelligence and flocking behaviour. International Journal of Computer Applications, 975: 8887.

[21] El Dor, A., Clerc, M., Siarry, P. (2012). A multi-swarm PSO using charged particles in a partitioned search space for continuous optimization. Computational Optimization and Applications, 53(1): 271-295. <https://doi.org/10.1007/s10589-011-9449-4>

[22] Li, J., Zhang, J., Jiang, C., Zhou, M. (2015). Composite particle swarm optimizer with historical memory for function optimization. IEEE Transactions on Cybernetics, 45(10): 2350-2363. <https://doi.org/10.1109/tcyb.2015.2424836>

[23] Cartwright, H. (2002). Swarm intelligence. By James Kennedy and Russell C Eberhart with Yuhui Shi. Morgan Kaufmann Publishers: San Francisco, The Chemical Educator, 7(2): 123-124. <https://doi.org/10.1007/s00897020553a>

[24] Myrtellari, A., Marango, P., Gjonaj, M. (2016). Analysis and performance of linear quadratic regulator and PSO algorithm in optimal control of DC motor. International Journal of Latest Research in Engineering and Technology, 2(4).

[25] Al-Mahturi, A., Wahid, H. (2017). Optimal tuning of linear quadratic regulator controller using a particle swarm optimization for two-rotor aerodynamical system. International Journal of Electronics and Communication Engineering, 11(2): 196-202. <http://dx.doi.org/10.5281/zenodo.1128899>

NOMENCLATURE

I	Length to the body's centre of mass
R	Nominal terminal resistance
g	Gravity constant
Q, R	State and control weighting matrices of LQR controller

Q_e, R_e	Weighting matrices of Kalman Filter
r	Radius of wheel
x	Displacement of the wheel in the x-axis
\dot{x}	Velocity of the wheel in the x-axis
y	Robot output equation
N	Scaling matrix
K	LQR gain matrix
P	Symmetric and positive semi-definite covariance matrix

Greek symbols

α	Angle
β	Angle
φ	Angle measured from vertical upward direction of the wheels
$\dot{\varphi}$	Angular velocity of the two wheels

Subscripts

I_w	Moment of inertia of the robot's wheels
I_P	Moment of inertia of the robot's chassis
M_P	Mass of the robot's chassis
M_w	Mass of the wheel connected to both sides of the robot
K_m	Torque constant
K_e	Back emf constant
P_o	Covariant matrix
x_i	Position of i^{th} particle
v_i	Velocity of i^{th} Particle
t_r	Rise time
t_s	Settling time
M_o	Maximum overshoot
e_{ss}	Steady state error
C_1	Cognitive component
C_2	Social component
W_{max}	Maximum inertia weight
W_{min}	Minimum inertia weight