Balancing a Segway robot using LQR controller based on genetic and bacteria foraging optimization algorithms

Ibrahim K. Mohammed, Abdulla I. Abdulla
Systems and Control Engineering Department, College of Electronics Engineering, Ninevah University, Iraq

Article Info	ABSTRACT
	A two-wheeled single seat Segway robot is a special kind of wheeled mobile robot, using it as a human transporter system needs applying a robust control system to overcome its inherent unstable problem. The mathematical model of the system dynamics is derived and then state space formulation for the system is presented to enable design state feedback controller scheme. In this research, an optimal control system based on linear quadratic regulator (LQR) technique is proposed to stabilize the mobile robot. The LQR controller is designed to control the position and yaw rotation of the two-wheeled vehicle. The proposed balancing robot system is validated by simulating the LQR using Matlab software. Two tuning methods, genetic algorithm (GA) and bacteria foraging optimization algorithm (BFOA) are used to obtain optimal values for controller parameters. A comparison between the performance of both controllers GA-LQR and BFO-LQR is achieved based on the standard control criteria which includes rise time, maximum overshoot, settling time and control input of the system. Simulation results suggest that the BFOA-LQR controller can be adopted to balance the Segway robot with minimal overshoot and oscillation frequency.

Corresponding Author:

Ibrahim K. Mohammed,
Systems and Control Engineering Department,
College of Electronics Engineering,
Ninevah University,
Mosul, Iraq.
Email: ibrahim.mohammed@uoninevah.edu.iq

1. INTRODUCTION

The Segway robot is an electric, two-wheeled self-balancing human transporter with a computer-controlled gyroscopic stabilization and control system. This new vehicle is a complex and hybrid machine that requires engineering competence in many fields; vehicle dynamics, automatic control, battery technology, power electronics, software engineering, microcomputer programming, network and communication engineering. The two-wheeled robot was revealed by Dean Kamen in 2001 to replace the car or buggies which were more polluting [1]. The personal transporter has the wide applications in the military and civilian field due to its simple construction, high energy using rate, stable movement and ability to adapt with surrounding environmental conditions. However, the single-seat robot faces many challenges related to nonlinearity, high order variables, strong coupling and inherent unstable dynamics of the system. Therefore, balance problem and development of this type of vehicle is a hot research topic of interest to many robotics laboratories around the world. Segway is a known two-wheeled single-seat mobile robot which has been utilized as a commercial personal transportation vehicle. In the Segway robotic system a collection of tiltsensors and five gyroscopes are

used to remain the human user in the straight position [2]. Moreover, [3] use two teams, human transporter platform (HT) and robot mobility platform (RMP), to created a new mobile robot, called Segway Soccer. The walking robotic system is used to examine the coordination of dynamically formed, mixed personal-mobile robot teams within the domain of a team function that requires making a decision and response.

An initial model of a digitally controlled mobile robot has been constructed in [4]. In the presented prototype, which is based on the inverted pendulum, the human rider is simulated by attaching weights to the scheme. A linear state feedback controller is used to stabilize the proposed personal mobile robot based on sensory information from motor encoders and a gyroscope. A new two-wheeled inverted pendulum mobile robot is presented by [5]. A control system based on pole placement technique is designed to stabilize the rider in an upright position. It is worth considering that, there are another models for two-wheeled self-balancing human vehicles invented by robotic engineers. In 2003, David P. A. built the Nbot balancing robot used a human transporter provided by a linear controller used to stabilize the system using information from motor encoder and the inertial sensor. Furthermore, [6] has successfully fabricated a new model of robot called Legway based on Mindstorms robotics kit. A linear controller is used to stabilize the electric vehicle based information of the tilt angle obtained from two electro-optical proximity detector (EOPD) sensors. However, fine tuning for the controller parameters is required to achieve a peter performance. Hussien et al. [7] introduced another model for two-wheeled mobile robot. Nonlinear control system based on model reference adaptive control (MRAC) using Lpayunovs stability theorem is employed to balance the proposed single-seat electric vehicle. Simulation and experimental results are presented to validate tracking performance of the proposed robot actuation system. However, the performance of the robot control system is based on the controller coefficients, which are chosen arbitrary, therefore, for peter performance an optimization algorithm should be used. All the proposed controllers used to stabilize the mobile robots are not optimized using optimization methods, the gain parameters are obtained using trial and error procedure. In order to achieve a stable actuation performance for mobile robot system a proper controller technique based on tuning algorithm should be used. The linear quadratic regulator (LQR) is among most popular linear state feedback controllers implemented in field of industrial applications. The implementation of the LQRcontroller can be noticed by research works published by [4, 8-11].

In this study, the LQR controller is adopted to implement the control system used to balance the two-round human vehicle as it seeks basically a trade off between the best control performance and minimum power input [12, 13]. However, using this controller approach includes many problems and disadvantages such as relying on the designer experience and skill and the trial and error procedure in the determine of the controller weighting matrices Q and R. Therefore, there is a enormous concern by robotic engineers in computer-aided optimization algorithms that can be considered to obtain the global optimum solution of Q and R matrices..

There are many optimization approaches that have been adopted by control engineers to obtain optimum values for elements of LQR matrices, such as a particle swarm inspired evolutionary algorithm (PS-EA), particle swarm optimization (PSO), genetic algorithm (GA) [14], combination of simulated annealing (SA) and GA [15], differential evaluation (DE) [16] and ant colony optimization (ACO) [17]. These intelligent optimization tools make the LQR controller system very robust, and insensitive to noisy and/or missing data. Bacteria foraging optimization algorithm (BFOA) is a new tuning algorithm that can be applied to optimize the cost function of several problems in different application fields. The BFOA is a combinatorial optimization algorithm which applied to achieve the best global solution for the proposed LQR controller. It is worth considering that there is a significant improvement inthe control response of the LQR controller is achieved by using these tuning algorithms, however, the research work is still open to explore for further controller improvements and developments. In this research, two tuning algorithms, GA and BFOA are utilized to improve the behavior of the LQR controller.

The remainder of the paper is organised as follows: section 2 presents modelling and dynamics of the proposed two-wheeled self-balancing human robot. In section 3, the technique of the controller system is introduced. Section 4, presents optimization methods of LQR controller. Section 5 and section 6 introduce controller design and simulation results of GA-LQR and section 7 introduce BFOA-LQR controllers for the robot system respectively. Conclusions and future work are presented in section 8.

2.   SYSTEM MODELING
The performance of a walking robot depends on the reliability of the system modeling and robustness of the controller design. The structure of the two-wheeled Segway personal robot composes mainly of an electrical sub system and mechanical subsystem. Figure 1 demonstrates graphic model of the Segway robot. In this section, the motion equation of the inverted pendulum is derived and the dynamic model of the motors is formulated. The motor model is then utilized in formulation the dynamic model of

the personal robot scheme to give a functional relationship between applied voltage to the DC motors and adjusting magnetic torque required to stabilize the human mobile robot.

2.1. Electrical subsystem modeling

The main part of the electric sub system is the DC motors, which are used to rotate the left and right wheel of the robot. The electric circuit of the DC motor is shown in Figure 2. Applying voltage V_a (V) to the motor terminals generates a current $i(t)$ (A) in the motor armature. The excited motor produces a torque (Nm) governed by the following relationship.

$$C = K_m i \tag{1}$$

where K_m is torque constant (Nm/A). The back electromotive (emf) voltage V_e (V) produced in the motor coil can be approximated as a linear function of motor angular velocity $\dot{\theta}$ (rad/s), as follows:

$$V_e = K_e \dot{\theta} \tag{2}$$

where K_e is torque constant (Vs/rad). Applying Kirchoff's voltage law to the motor circuit shown in Figure 2 yields the following expression:

$$V_a = Ri + L\frac{di}{dt} + V_e \tag{3}$$

It is worth considering that the dynamic of the mechanical system is considered slow compared to that of electrical system, therefore, the current transients of the system can be omitted. Hence, solving (3) for the current yields:

$$i = \frac{V_a - V_e}{R} \tag{4}$$

based on (2), (4) can be written as follows:

$$i = \frac{V_a}{R} - \frac{K_e}{R}\dot{\theta} \tag{5}$$

substituting (5) in (1) gives an expression for the torque produced by DC motor:

$$C = \frac{K_m}{R}V_a - \frac{K_m K_e}{R}\dot{\theta} \tag{6}$$
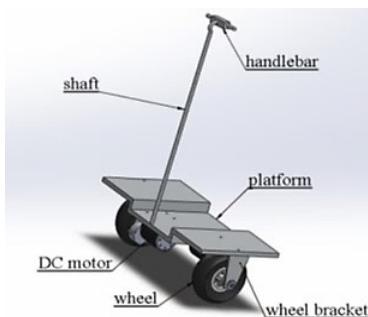
Figure 1. Model of the two-wheeled robot [18]   Figure 2. Electric model circuit of DC motor
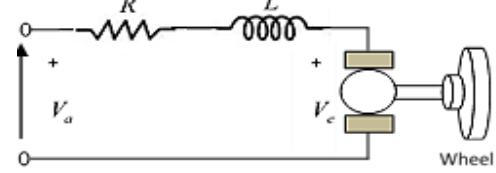
2.2. Mechanical subsystem modeling

It consists of chassis, which behaviors as inverted pendulum, and the left and right wheels. In this study, the parameter of mass, radius and moment of inertia of the two wheels are assumed the same. Based on this assumption the dynamic modeling of the right wheel is the same as that of the left wheel. In this research, formulation of the right wheel model is considered in detail. It is worth considering that the modeling strategy of the mechanical robot subsystem is based on an idea that the dynamics of

the inverted pendulum and wheels are modeled separately at the beginning, and then equations of motion which completely describe the dynamic behavior of the system are derived [19].

2.2.1. Wheel model

Figure 3 presents the free body diagram of the left and right wheels for the mobile Segway transport system. Using second Newton's law of motion, the sum of the external forces $F(N)$ exerted on the wheel, which governs its translation motion in the horizontal x-direction is given by [20].

$$\sum F_x = M_w a \tag{7}$$

$$M_w \ddot{x} = H_f - H \tag{8}$$
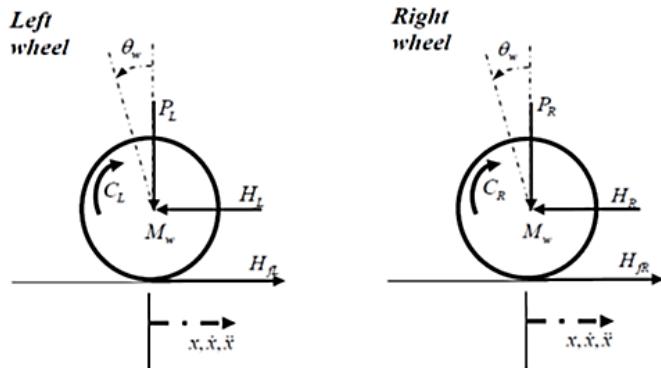
Figure 3. Free body diagram of the robot wheels

where $M_w(Kg)$ is the wheel mass of the robot, a is the gravity acceleration (m/s^2) and H_f is the friction force between ground and wheels(N). While the rotational motion M_o of the wheel is given by: go

$$\sum M_o = I_w \ddot{\theta}_w \tag{9}$$

$$I_w \ddot{\theta}_w = C - H_f r \tag{10}$$

where $I_w \ (Kgm^2)$ and $\ddot{\theta}_w \ (m/s^2)$ are moment of inertia and acceleration of the wheels respectively, and r is the radius of wheel (m). Based on (6), the above equation becomes:

$$I_w \ddot{\theta}_w = \frac{K_m}{R} V_a - \frac{K_m K_e}{R} \dot{\theta}_w - H_f r \tag{11}$$

$$H_f = \frac{K_m}{Rr} V_a - \frac{K_m K_e}{Rr} \dot{\theta}_w - \frac{I_w}{r} \ddot{\theta} \tag{12}$$

where $\dot{\theta}_w$ is the angular velocity of wheel (m/s). Substituting (12) into (8) yields (13) and (14) for the left and right wheels respectively.

$$M_w \ddot{x} = \frac{K_m}{Rr} V_a - \frac{K_m K_e}{Rr} \dot{\theta}_w - \frac{I_w}{r} \ddot{\theta}_w - H_L \tag{13}$$

$$M_w \ddot{x} = \frac{K_m}{Rr} V_a - \frac{K_m K_e}{Rr} \dot{\theta}_w - \frac{I_w}{r} \ddot{\theta}_w - H_R \tag{14}$$

Because the center of the robot wheel is acted by the linear motion, the angular rotation of the wheel can be transformed into linear motion by the following simple transformation, $\ddot{\theta}_w r = \ddot{x} \rightarrow \ddot{\theta} = \ddot{x}/r$, $\dot{\theta}_w r = \dot{x} \rightarrow \dot{\theta} = \dot{x}/r$. By the linear transformation, (13) and (14) become as follows:

$$M_w \ddot{x} = \frac{K_m}{Rr} V_a - \frac{K_m K_e}{Rr^2} \dot{x}_w - \frac{I_w}{r^2} \ddot{x} - H_L \tag{15}$$

$$M_w \ddot{x} = \frac{K_m}{Rr} V_a - \frac{K_m K_e}{Rr^2} \dot{x}_w - \frac{I_w}{r^2} \ddot{x} - H_R \tag{16}$$

Adding (15) and (16) together yields the following expression,

$$2(M_w + \frac{I_w}{r^2})\ddot{x} = 2\frac{K_m}{Rr}V_a - 2\frac{K_m K_e}{Rr^2}\dot{x}_w - (H_L + H_R) \tag{17}$$

2.2.2. Chassis model
The chassis of the mobile robot can be modeled as an inverted pendulum, Figure 4 presents the free body diagram of the chassis. Again, based on Newton's law of motion, the sum of forces acting on the chassis in the horizontal x-direction is given by [20]:

$$\sum F_x = M_p \ddot{x} \tag{18}$$

$$M_p \ddot{x} = H_L + H_R - M_p l \ddot{\theta}_{\ddot{p}} \cos\theta_p + M_p l \dot{\theta}^2_{p} \sin\theta_p \tag{19}$$

where M_p is the wheel mass of the robot (Kg) and θ_p is the rotational angle of the chassis (rad). The above equation is rearranged as follows:

$$H_L + H_R = M_p \ddot{x} + M_p l \ddot{\theta}_p \cos\theta_p - M_p l \dot{\theta}^2_{p} \sin\theta_p \tag{20}$$

The sum of perpendicular forces acting on the pendulum is:

$$\sum F_p = M_p \ddot{x} \cos\theta_p \tag{21}$$

$$(H_L + H_R)\cos\theta_p + (P_L + P_R)\sin\theta_p - M_p g \sin\theta_p - M_p l \ddot{\theta}_{\ddot{p}} = M_p \ddot{x} \cos\theta_p \tag{22}$$

where P_L and P_R are the reaction forces between left and right wheel and chassis (N) respectively and $\ddot{\theta}_p$ is chassis angular acceleration(m/s^2), sum of moments around the center of pendulum mass is given by:

$$\sum M_o = I_p \ddot{\theta}_p \tag{23}$$

$$I_p \ddot{\theta}_p = -(H_L + H_R)l\cos\theta_p - (P_L + P_R)l\sin\theta_p - (C_L + C_R) \tag{24}$$

where C_L and C_R are the motor torque applied to left and right wheels respectively (Nm) the motor torque exerted on the pendulum applied as defined in (6) and after linear transformation,

$$C_L + C_R = 2\frac{K_m}{R}V_a - 2\frac{K_m K_e}{R}\frac{\dot{x}}{r} \tag{25}$$

substituting (25) into (24) yields,

$$I_p \ddot{\theta}_p - 2\frac{K_m K_e}{R}\frac{\dot{x}}{r} + 2\frac{K_m}{R}V_a = -(H_L + H_R)l\cos\theta_p (P_L + P_R)l\sin\theta_p \tag{26}$$

substitute (26) in (22) after multiply (22) by$(-l)$ yields:

$$I_p \ddot{\theta}_p - 2\frac{K_m K_e}{R}\frac{\dot{x}}{r} + 2\frac{K_m}{R}V_a + M_p gl\sin\theta_p + M_p l^2 \ddot{\theta}_{\ddot{p}} = -M_p l\ddot{x}\cos\theta_p \tag{27}$$

Sum of moments around the center of pendulum mass is given:

$$-M_p l\ddot{x}\cos\theta_p = (I_p + M_p l^2)\ddot{\theta}_p - 2\frac{K_m K_e}{Rr^2}\dot{x} + 2\frac{K_m}{Rr}V_a + M_p gl\sin\theta_p \tag{28}$$

to eliminate the term $(H_L + H_R)$ from the motor dynamic (20) is substituted in (17).

$$2\frac{K_m}{Rr}V_a = (2M_w + \frac{I_w}{r^2} + M_p)\ddot{x} + 2\frac{K_m K_e}{Rr^2}\dot{x} + M_p l\ddot{\theta}_p \cos\theta_p - M_p l\dot{\theta}^2_{\ddot{p}} \sin\theta_p \tag{29}$$

for purpose of design a linear control system, the system dynamic (28) and (29) are linearized about an operating point based on the assumption $\theta_w = \pi + \phi$ where φ denotes an angle measured from the vertical upward direction. Therefore, $sin\,\theta_p = -\phi$, $cos\,\theta_p = -1$ and $(d\theta_p/dt)^2 = 0$ for purpose of state space representation, the dynamic (28) and (29) are rewritten as follows:

$$\ddot{\phi} = \frac{M_p I}{(I_p + M_p I^2)}\ddot{x} + \frac{2K_m K_e}{Rr(I_p + M_p I^2)}\dot{x} + \frac{M_p g I}{Rr(I_p + M_p I^2)}\phi - \frac{2K_m}{(I_p + M_p I^2)}V_a \qquad (30)$$

$$\ddot{x} = \frac{2K_m}{RrK_w}V_a - \frac{2K_m K_e}{Rr^2 K_w}\dot{x} + \frac{M_p I}{Rr^2 K_w}\ddot{\phi} \qquad (31)$$

where $K_w = 2M_w + 2I_w/r^2$. After a series of algebraic manipulation, the state equation becomes:

$$\begin{bmatrix}\dot{x}\\\ddot{x}\\\dot{\phi}\\\ddot{\phi}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\0 & \frac{K_1(M_p I r - I_p - M_p I^2)}{Rr^2\alpha} & \frac{M^2{}_p g I^2}{\alpha} & 0\\0 & 0 & 0 & 1\\0 & \frac{K_1(r\beta - M_p I)}{Rr^2\alpha} & \frac{M_p g I\beta}{\alpha} & 0\end{bmatrix}\begin{bmatrix}x\\\dot{x}\\\phi\\\dot{\phi}\end{bmatrix} + \begin{bmatrix}0\\\frac{2K_m(I_p + M_p I^2 - M_p I r)}{Rr\alpha}\\0\\\frac{2K_m(M_p I - \beta)}{Rr\alpha}\end{bmatrix}V_a \qquad (32)$$

where, $K_1 = 2K_m K_e$, $\beta = 2M_w + \frac{2I_w}{r^2} + M_p$ and $\alpha = \left[I_p\beta + 2M_p I^2(M_w + \frac{I_w}{r^2})\right]$. It is worth considering that the system modeling is based on the supposition that both wheels of the walking robot are assumed in state of contact with the ground and without sliding. Cornering forces produced by vehicle wheels during cornering are also considered negligible [19].
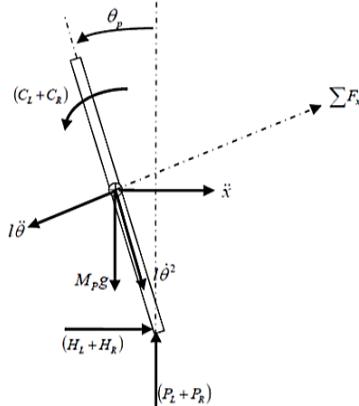

Figure 4. Free body diagram of the chassis

3. CONTROLLER TECHNIQUE

In this research, LQR technique is utilized to implement the proposed robot control system. LQR is a common controller approach used effectively in the control applications of the movement systems. Figure 5 shows block diagram of the Segway robot control system based on LQR controller. The state and output matrix equations describing the Segway robot equations of motion are given by:

$$\dot{X}(t) = AX(t) + Bu(t) \qquad (33)$$

$$y(t) = CX(t) + Bu(t) \qquad (34)$$

where $A(nxn), B(nxm), C(pxn), D(pxm)$ are the system, input, output and feed forward matrix respectively. In this approach, the input vector:

$$u(t) = -KX(t) \qquad (35)$$

where $K = [K_1 \quad K_2 \quad K_3 \quad K_4]$ is an optimal feedback gain matrix of the controller used to track the input command while the following performance index:

$$J = \int_0^\infty (X^T(t)Q(t)X(t) + u^T(t)R(t)u(t))dt \tag{36}$$

where $Q(t)$ and $R(t)$ are weighting state and input matrices respectively. The feedback gain matrix K can be determined by using the following:

$$K = R^{-1}B^T P \tag{37}$$

where P denotes the solution of the following Riccati:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{38}$$

The controller weighting matrices should be tuned properly in order to minimize the following performance index (J):

$$J = \int_0^\infty (q_{11}x_1{}^2 + q_{22}x_2{}^2 + q_{33}x_3{}^2 + q_{44}x_4{}^2 + Ru^2)dt \tag{39}$$

where q_{11}, q_{22}, q_{33} and q_{44} represent the weighting elements of position, speed, angle and angular velocity of the proposed robot system respectively. It is worth considering that by using controller weighting matrices $Q(t)$ and $R(t)$ which govern the behavior of the robot system states and control effort respectively, the optimal LQR gain matrix K is computed based on the Matlab command " lqr ".
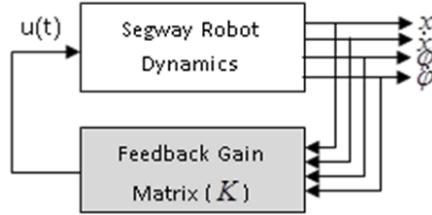

Figure 5. Segway robot control system using LQR controller

4.   LQR OPTIMIZATION METHODS
In this research, two optimization algorithms, GA and BFOA, are used to tune the $Q(t)$ and $R(t)$ matrices of the LQR controller, which are adopted to calculate the gain matrix required to balance the Segway system. Based on optimized gain matrix K a good output time response with minimal of rise time (t_r), settling time (t_s), maximum overshoot (Mo) and steady state error (e_{ss}) can be investigated.

4.1. Genetic algorithm
GA is an optimization technique used to find global solution for more control problems. Based on the mechanisms of natural selection. In this optimization approach, the solution space is selected by generating a population of candidate individuals to find optimum values for the problem. The procedure of GA optimization method includes three basic steps, namely selection, crossover and mutation. By applying these stages new individuals can be created, which, could be better than their parents. Based on the fitness function of the system, the GA steps are repeated for many generations and eventually stop at generating candidate individual elements that can represent the best solution for the application problem [14]. Figure 6 shows the graphical illustration of the GA loop.
The definition of the GA steps is as follows Abdulla [14]: Random initial population - Choose individuals for mating - Mate the population to generate progeny - Mutate progeny - New individuals inserted into population - Are criteria satisfied? - End of solution searching. Each chromosome represented by five real value cells that correspond to the LQR controller weighting matrices $Q(t)$ and $R(t)$ as shown in Figure 7. The chromosome elements $q_{11}, q_{22}, q_{33}, q_{44}$ and R should be adjusted properly by optimum positive numbers in order to achieve best control performance.

q_{11}	q_{22}	q_{33}	q_{44}	R
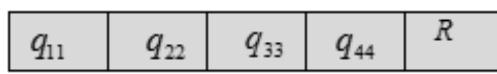
Figure 6. Genetic algorithm loop
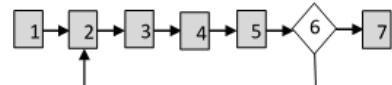
Figure 7. Chromosome definition

4.2. BFO algorithm

The bacteria foraging behavior is a computational model, which, has attracted more attention as it is a rich source of potential engineering applications. BFO is a simple and powerful population-based numerical optimization algorithm that has been introduced by Passino in 2002 [21]. The BFO algorithm has been gaining a considerable interest in researchers due to its efficiency in solving and optimizing more engineering problems in several application domains, such as optimal control [22], harmonic estimation [23] and transmission loss reduction [24]. The strategy of bacteria selection for the BFO algorithm bases on an idea that the bacteria with poor foraging is eliminated and following up those bacteria which have successful foraging to maximize energy obtained per unit time [21]. It is worth considering that, in this new optimization algorithm a social foraging approach of the E-coli bacteria can be applied successfully to solve multi-optimal function optimization problem. The E-coli bacteria can move in two ways namely, swimming and tumbling. Figure 8 shows the swim and tumble movement of a bacterium.
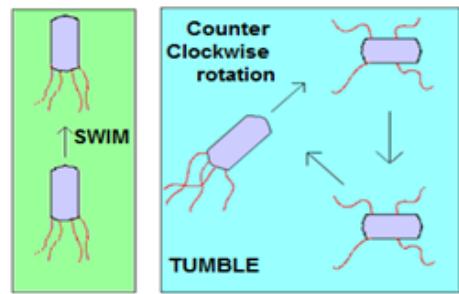

Figure 8. movement modes of a bacterium

4.3. BFO foraging strategy [25]

- Chemotaxis: This process is related to movement of bacterium during search for food. The E-coli bacteria can move in two ways namely, swimming and tumbling, and they are able to alternate between these two movement styles for the whole of their lifetime. In the swimming mode, the bacteria walk in a certain direction for gathering food, while in the tumbling mode, they move with random directions.
- Swarming: The swarming action means the bacteria with a good fitness value, try to attract others to form groups, so that they all can arrive at the desired location. These groups of E-coli cells arrange themselves, in which, they can move as concentric patterns for food searching.
- Reproduction: In this stage, all the bacteria population are classified based on health status. The healthier bacteria, which, have had sufficient nutrients will be reproduced, while the less healthy bacteria will die. The surviving bacteria will split into an identical replica of itself placed in the same locations with a that number equals to the number of the dead ones.
- Elimination and Dispersal: During this evolutionary step, gradual or sudden events or attacks in the local living environment of the bacteria, may occur due to significant rising of the temperature caused by occupancy of a high density of bacteria for a specific area. This high temperature may kill a group of bacteria and dispersal of others into some new locations.

5. ROBOT CONTROL SYSTEM DESIGN

In this research, an optimal control system for the Segway robot is designed using the state feedback LQR controller. The gain parameters of the controller are tuned effectively, using optimization algorithms, GA and BFOA. The proposed controller design is validated using Matlab programming. Based on step input, the control system is designed for the following requirements: rise time less than 10 (ms), settling time less than 30 (ms), maximum overshoot percentage less than 5%. The fitness function of the robot control system is as follows:

$$F = S.t_r, t_{rmax} + S.t_s, t_{smax} + S.O, M_o \tag{40}$$

where S is the closed-loop transfer function of the Segway robot scheme t_r is the rise time (s), t_{rmax} is the maximum rise time (s), t_s is the settling time (s), t_{smax} is the maximum settling time (s), O is the overshoot value of the output response and Mo is the maximum over shoot value. This function is considered in optimization process of the controller gain matrix using GA and BFO tuning algorithms.

6. GA-LQR CONTROLLER DESIGN AND RESULTS

The block diagram of the Segway control system using GA-LQR controller is shown in Figure 9. Based on the system parameters ($R = 1.6\Omega$, $L = 1.2mH$, $l = 0.16m$, $M_w = 0.02Kg$, $M_p = 0.52Kg$, $I_w = 0.0032Kgm^2$, $I_p = 0.0038\ Kgm^2$) the state and output equation are given in (41) and (42) respectively. The global optimal solution for the LQR controller problem is achieved using genetic algorithm programming. This solution includes determine optimum values for the weighting matrices elements, which, are given below. Based on the optimized weighting matrices, the LQR gain matrix K was computed using the Matlab command "lqr" as follows:$Q = blkdia(q_{11}, q_{22}, q_{33}, q_{44})$ where $q_{11} = 8.969.$, $q_{22} = 0.308$, $q_{33} = 0.121$, $q_{44} = 0.0085$, $R = 2.123*10^{-5}$ and $K = [\text{-6.4988 -3.8592 -5.1982 -0.7348}]$.

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.0316 & 0.3817 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0.3927 & 49.5531 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0.05746 \\ 0 \\ -0.7141 \end{bmatrix} V_a \tag{41}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \tag{42}$$

The response of the system output $\Delta x(t)$ and $\Delta \phi(t)$ using the above optimized feedback gain matrix is shown in Figure 10. Figure 11 presents the input signal of the Segway system. It is clear from Figures 10 and 11 that the tuned controller utilizing the GA tuning method can effectively perform a fast and stable response under an acceptable input effort. Based on Figure 10, the output states $\Delta x(t)$ and $\Delta \phi(t)$ tracked the demand input trajectories without overshoot, rise time and settling time of approximately 80 ms and 95 ms respectively and approximately zero steady state error. Figure 12 presents converging of the LQR weighting matrices through iterations based on GA tuning method.
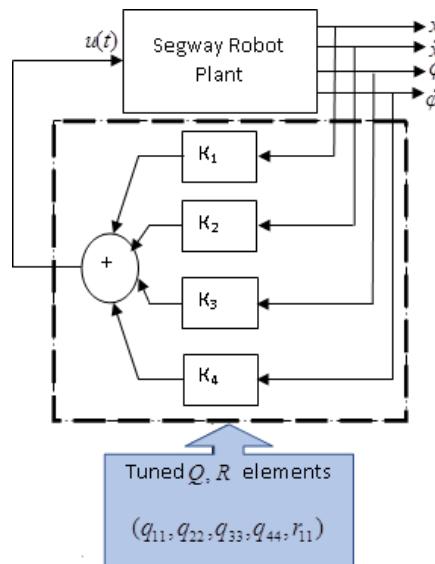

Figure 9. Block diagram of Segway controller GA-LQR
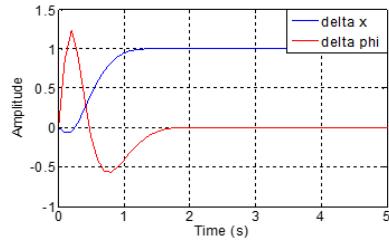
Figure 10. System response using GA-LQR controller
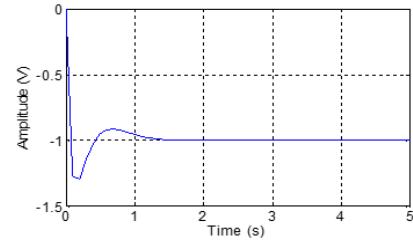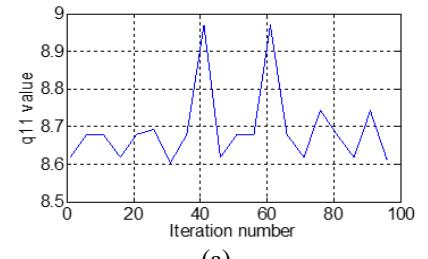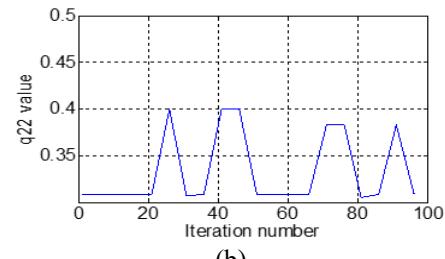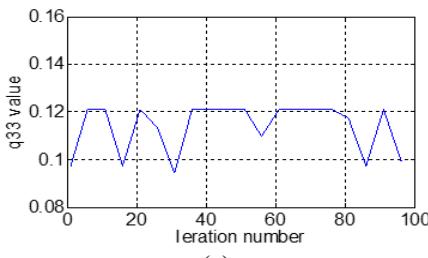

Figure 11. System input based on GA-LQR controller
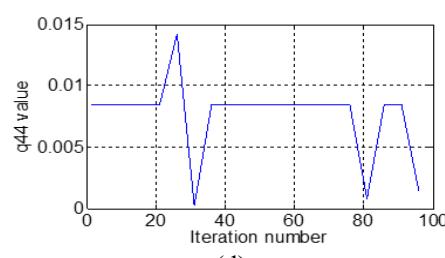
(a)
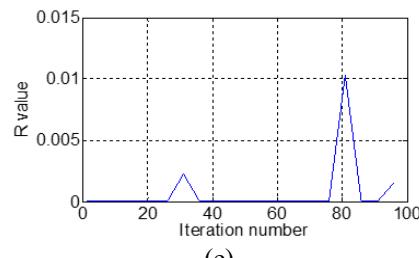
(b)

(c)

(d)

(e)

Figure 12. Generation elements of $Q(t)$ and $R(t)$ matrices for GA-LQR controller; (a) q_{11} element, (b) q_{22} element, (c) q_{33} element, (d) q_{44} element, and (e) R element

7. BFO-LQR CONTROLLER DESIGN AND RESULTS

The design of LQR controller based on BFOA method is analogous to that of the GA-LQR control system as previously presented in Figure 9. The optimized LQR matrices $Q(t)$ and $R(t)$ and controller gain matrix using BFO algorithm are as follows: $Q = blkdia(q_{11}, q_{22}, q_{33}, q_{44})$ where q_{11} =289.104, q_{22} =-5.15*10^{-5}, q_{33} =-5*10^{-5}, q_{44} = -5.1*10^{-5} R =0.00028 and $K = $ [-1022 -260.2 886.72 77.7].

The time response of the robot states $\Delta x(t), \Delta\phi(t)$ using BFOA-LQR controller is shown in Figure 13. Figure 14 shows the control effort required to stabilize the Segway robot system. Regarding the optimization of the controller, Figure 15 introduces converging elements of weighting matrices of the LQR controller through iterations based on BFOA tuning algorithm. By comparing the behavior of the robot system shown in Figures 10, 11, 13 and 14. It should be noted that the BFOA-LQR controller can achieve more stable and faster response through following the desired trajectories effectively. Table 1 shows comparison results between the GA-LQR controller and BFOA-LQR controller based on control criteria parameters. Based on Table 1, the BFOA-LQR controller enabled the walking robot to follow the demand trajectories effectively.
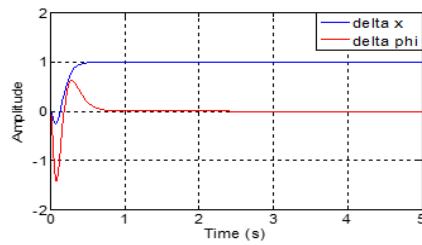
Figure 13. Robot response using BFOA-LQR
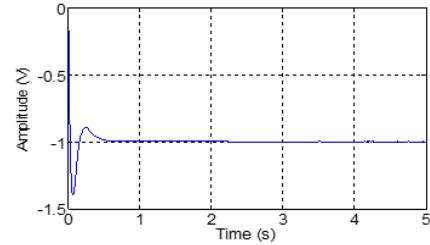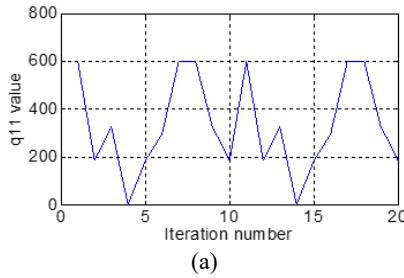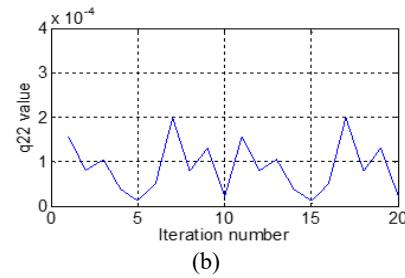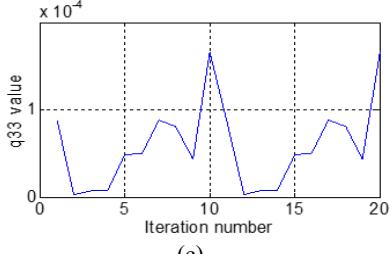
Figure 14. Control effort of robot system
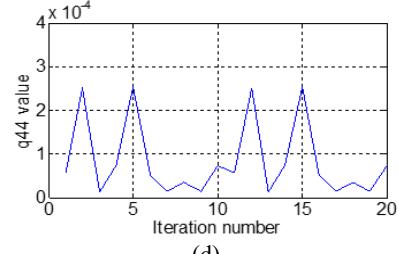
(a)
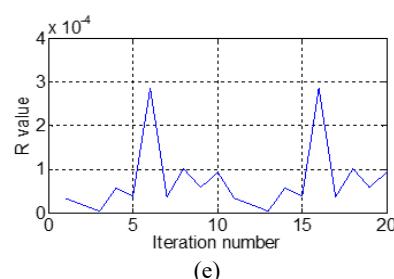
(b)

(c)

(d)

(e)

Figure 15. Generation elements of $Q(t)$ and $R(t)$ matrices for BFO-LQR controller; (a) q_{11} element,
(b) q_{22} element, (c) q_{33} element(c) q_{33} element, (c) q_{33} element,
(d) q_{44} element, and (e) R element (continue)

Table 1. Performance parameters of the Segway using GA-LQR and BFOA-LQR controller

Cont. Type	System Output	Control Criteria Parameter			
		Rise Time	Settling Time	Max. Overshoot	Control Input
GA-LQR	Position	1.25 s	1 s	5 %	1.3 V
	Yaw Angle	1.8 s	10 ms	120 %	
BFOA-LQR	Position	0.5 s	0.33 s	0 %	1.4 V
	Yaw Angle	0.85 s	4 ms	130 %	

8. CONCLUSIONS

In this research, an optimal linear control system was adopted to balance a Segway two-wheeled mobile robot. The dynamics of the robot system is modeled in state space form in order to design a state feedback stabilizing controller for the Seqway robot system. A LQR controller was proposed to stabilize the Segway robot in upright position. The controller is optimized using two tuning algorithms, GA and BFO.

An active stabilizing scheme for the Segway robot system has been implemented successfully using GA-LQR controller and BFOA-LQR controller. Simulation results of the controllers are introduced and then compared based on standard stabilizing parameters. The comparison revealed that the BFOA-LQR controller can be adopted to implement faster and a more stable balancing system for the Segway vehicle.

REFERENCES

[1] Salerno A., and Angeles J., "A new family of Two Wheeled Mobile Robot: Modeling and Controllability," *IEEE Transaction of Robotics*, vol. 23, no. 1, pp. 169-173, 2007.

[2] Khaled M., "Mathematical Modelling and Control of a Two-Wheeled PUMA-LikeVehicle," *Journal of Mechanical Engineering Research*, vol. 6, no. 2, pp. 11-22, 2016.

[3] Searock J., B. Browning, and M. Veloso, "Segway CM-RMP Robot Soccer Player," *Proceedings of the RoboCup International Symposium, Lisbon, Portugal*, 2004.

[4] Grasser F., Alonso D., Silvio C., and Alfred C. R., "JOE: A Mobile, Inverted Pendulum," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 1, pp. 107-114, 2002.

[5] Nawawi W., Ahmad N. and Osman S., "Real-Time Control of a Two-Wheeled Inverted Pendulum Mobile Robot," *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, vol. 2, no. 3, pp. 406-412, 2008.

[6] Hassenplug S., "Steve's LegWay," Retrieved September 22, 2018. [Online]. Available: Steve's LegWay: http://www.teamhassenplug.org/robots/legway.

[7] Hessein A., Daniel A., and Naseem D., "Model Reference Adaptive Control of a Two-Wheeled Mobile Robot," *Proceedings of the International Conference on Robotic and Automation,* pp. 1155-1161, 2019.

[8] Shiroma N., Matsumoto O., Kajita S. and Tani K., "Cooperative Behaviour of a Wheeled Inverted Pendulum for Object Transportation," *Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 396-401, 1996.

[9] Nakajima R., Takashi T., Shin'ichi Y. and Eiji K., "A Development of a New Mechanism of an Autonomous Unicycle," *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pp. 906-912, 1997.

[10] Takahashi, Y., Ishikawa, N. and Hagiwara, T, "Inverse pendulum controlled two wheel drive system," *Proceedings of the SICE Annual Conference, International Session Papers*, pp. 112 -115, 2001.

[11] Mohammed I., Sharif B., Neasham J., and D. Giaouris, "Novel MIMO 4-DOF Position Control for Capsule Endoscope," *Proceedings of the IEEE ISCAS, Rio de Janeiro*, pp. 909-912, Brazil 2011.

[12] Mohammed I., Abdulla I., "Design of optimised linear quadratic regulator for capsule endoscopes based on artificial bee colony tuning algorithm," *International Journal for Engineering Modeling*, vol. 31. no. 1-2, pp. 77-98, 2018.

[13] Mohammed I., Abdulla I., "Fractional Order PID Controller Design for Speed Control DC Motor based on Artificial Bee Colony Optimization," *International Journal of Computer Applications*, vol. 179, no. 24, pp. 43-49, 2018.

[14] Abdulla A., Mohammed I. and Jasim A., "Roll Control System Design Using Auto Tuning LQR Technique," *International Journal of Engineering and Innovative Technology,* vol. 6, no. 2, pp. 11-22, 2017.

[15] Lee Y. J. and Cho K. H., "Determination of the Weighting Parameters of the LQR System for Nuclear Reactor Power Control using the Stochastic Searching Methods," *Journal of the Korean Nuclear Society*. vol. 29, no. 1, pp. 68-77, 1997.

[16] Storn R. and Price K., "Differential Evolution-A Fast and Efficient Heuristic for Global Optimization over Continuous Spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341-359, 1997.

[17] Ali D., Hend L., and Hassani M., "Optimized Eigenstructure Assignment by Ant System and LQR Approaches," *International Journal of Computer Science and Applications*, vol. 5, no. 4, pp. 45-56, 2008.

[18] Wei A. and Yangmin L., "Simulation and Control of a Two-wheeled Self-balancing Robot," *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO) Shenzhen*, pp. 34-38, 2013.

[19] Ooi R., "Balancing a Two-Wheeled Autonomous Robot," PhD Thesis, The University of Western Australia School of Mechanical Engineering, Faculty of Engineering and Mathematical Sciences, University of Western Australia; 2003.

[20] Fang J., "The LQR Controller Design of Two-Wheeled Self-Balancing Robot Based on the Particle Swarm Optimization Algorithm," *Institute of Engineering and Technology*, Changchun: 130052, China 2014.

[21] Passino M. K., "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control and System Magnetic*, vol. 22, no. 3, pp. 52-67, 2002.

[22] Kim D. H. and Cho J. H., "Adaptive tuning of PID controller for multivariable system using bacterial foraging based optimization," *Proceedings of the 3rd International Atlantic Web Intelligence Conference on Advances in Web Intelligence (AWIC '05)*, vol. 3528 of Lecture Notes in Computer Science, pp. 231-235, 2005.

[23] Mishra S., "A hybrid least square-fuzzy bacterial foraging strategy for harmonic estimation," *IEEE Transaction on Evolutionary Computation*, vol. 9, no. 1, pp. 61-73, 2005.

[24] Tripathy M., Mishra S., Lai L., and Zhang, Q. P., "Transmission loss reduction based on FACTS and bacteria foraging algorithm," *Proceedings of the 9th International Conference on Parallel Problem Solving from Nature (PPSN '06)*, pp. 222-231, 2006.

[25] Swagatam D., Arijit B., Sambarta D. and Ajith A., "Bacteria Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Appications," *Foundations of Computational Intelligence*, vol. 3, SCI 203, pp. 23-55, 2009.