JJEE
Volume X | Number X | Month 20XX | Pages XXX-XXX

Jordan Journal of Electrical Engineering

ISSN (print): 2409-9600, ISSN (online): 2409-9619

Homepage: jjee.ttu.edu.jo

A Comparative Analysis of Metaheuristic Algorithms Tuned Supper Twisting Sliding Mode Control of a Self-balancing Segway

Tefera T. Yetayew[1*], Daniel G. Tesfaye[2]

[1] Department of electrical power and control, Adama Science and Technology University, Ethiopia
E-mail: tefera06@gmail.com
[2] Department of electrical and computer engineering, Wolaita Sodo University, Ethiopia

Received: Jan 29, 2024 *Revised: Apr 05, 2024* *Accepted: Apr 11, 2024* *Available online: Sep 14, 2024*

***Abstract*—** A self-balancing two-wheel Segway is a multivariable, nonlinear, coupled and unstable personal transport that is among the benchmarks of under-actuated systems to test different control schemes. The proper operation of the system needs stable and robust controllers for the balancing and direction of the Segway. Stable and robust performance of such systems can be achieved using adaptive, optimal and non-linear control schemes. A conventional sliding mode controller is a nonlinear controller, considered a robust controller except for the chattering problem that can be solved using higher order sliding mode controllers such as a supper twisted sliding mode controller (STSMC). In this research, STSMC is proposed for balancing and PID controller for direction control of a self-balancing Segway. To solve the impacts of design technique, skill, and experience of the designer on the control schemes, controllers are tuned using metaheuristic optimization algorithms, namely Grey wolf algorithm (GWO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) technique. The overall system has been implemented in MATLAB/Simulink, and performance evaluations have been executed for the time domain specifications of settling and rise times as comparison criteria for the Segway operating conditions of variable driver mass and tilt angles. The results show relatively quick responses - for initial pitch angel of 0.3 rad, and initial yaw angle of 0.1 rad – of GWO-STSMC and GWO-PID controllers. For variable driver mass and constant initial tilt angles, the pitch response is found to be less sensitive to the increase in mass of the driver for GWO-STSMC, while the yaw angle response shows a slight sensitivity decrease for GWO-PID controller. For variable initial tilt angles and constant mass of the drive, both GWO-STSMC and GWO-PID controllers show the capability of tracking the reference pitch and yaw angles.

***Keywords*—** Self-balancing Segway; Metaheuristic optimization algorithms; Genetic algorithm; Grey wolf algorithm; Particle swarm optimization; Supper twisted sliding mode controller; PID controller.

1. INTRODUCTION

These days, fast growth of urbanization, population growth, increasing transportation demand, fast depletion of fossil fuels, ever increasing traffic and global worming issues are urging the development of personal transportations that are compact, energy efficient, environmentally friendly, low cost and portable. Among a wide variety of electric vehicles, e-scooters such as Self-balancing Segway are personal transportation technology that are portable, compact, nearly zero fuel cost and contributes a lot to global environment challenge by which they are the best competent to the ever-increasing traffic. However, the revolution of suck kinds of electric scooters also has challenges of lack of well-defined rules and regulations to reduce accidents; so development of rules and regulations all over the world that govern the use of such e-scooters is crucial [1-7].

* Corresponding author
DOI: https://doi.org/10.5455/jjee.204-1706551479

The Segway is a multivariable, nonlinear, coupled, and unstable under actuated system. The basic principle of the Segway and others like self-balancing two-wheeled mobile robots is based on the principle of inverted pendulum. The Segway can move forward, backward, turn directions left and right where all these riding activities need balancing and direction controllers. Practically, disturbances, parameter uncertainties and terrain inclinations are common causes that may result in loss of system stability and performance degradation unless robust control schemes are used. Robust controllers for multivariable, nonlinear, coupled and unstable systems can be summarized as nonlinear controllers like sliding mode controller, optimal controllers, adaptive controllers and hybrid controllers where the overall performance of these control schemes can be affected by the design approach, the skill, experience of the designer and accuracy of the mathematical model of the system [8-19]. Large number of research articles related to control of self-balancing Segway are available where majority of the researches did focus that multivariable, nonlinear, unstable, strong coupling and complex systems can show robust performance under different disturbances and uncertainty conditions for the controllers of nonlinear such as sliding mode controllers, optimal controllers, adaptive controllers and hybrid of intelligent algorithms such as neuro-fuzzy, fuzzy sliding mode, self-tuning fuzzy-PID, optimally tuned PID controllers, robust adaptive LQR, ANFIS, and etc. The conventional sliding mode controller is considered as a robust, less complex and low computational requirement to implement widely applicable to multivariable, nonlinear and coupled systems except chattering problem. There are a number of techniques to solve the chattering problem of the conventional sliding mode controller such as higher-order sliding mode controllers, hybrid fuzzy sliding mode, supper twisting sliding mode controller, adaptive supper twisting sliding mode controller, terminal sliding mode controller, and etc., [9, 11, 19-24]. An adaptive neural network based PID control of an inverted pendulum has been proposed in [21] where the capability of the algorithm is verified using simulation results. Robust controller of a Segway has been proposed in [25] and verified using the simulation result under the conditions of different uncertainties. However, the technique of achieving optimal combinations of the controller parameters is not covered. An extended Kalman filter based robust controller for a self-balancing robot has been proposed in [26]. The performance comparison of the proposed controller with a conventional sliding mode controller showed the effectiveness of the proposed controller. However, the impact of design approach is not addressed. A low-order robust controller for self-balancing control of two-wheel bicycle has been proposed in [27, 28] where the effectiveness of the algorithm is verified using simulation results under the conditions of external disturbance and model uncertainty. An extended state observer based robust nonlinear PID controller for a two-wheel self-balancing mobile robot is proposed in [29] where the robustness of the controller is verified using simulation results. A novel backstepping linear quadratic Gaussian controller for self-balancing inverted pendulum has been proposed in [30] and performance evaluation in comparison with optimal PID and other robust algorithms from literatures under external disturbance uncertainties has been verified using simulation results. An adaptive observer based sliding mode controller to control a self-balancing mobile robot with two wheels has been proposed in [31] and performance tests under the operating conditions of external disturbance and terrain inclination has been verified using simulation results. The authors in [9] proposed fuzzy logic control system for a Segway type mobile robot and performance evaluation has been done using simulation results. However, manual design approach of the

proposed controller has been considered. In [8], fuzzy-PID modified state feedback controller has been proposed to control a two-wheel inverted pendulum. The simulation results revealed the effectiveness of the proposed combination of control algorithms. However, commutation requirement for practical implementation, optimal design approach and skill requirement to design the fuzzy part issues is not considered. The authors in [32] proposed interval Type-2 fuzzy logic-based balancing and direction control of mobile wheeled inverted pendulum and simulation-based verification has been done. In summary, the robustness of the controller is highly affected by the design approach, the model, skill and experience of the designer apart from the control scheme. These challenges can be solved if the optimal combinations of the corresponding controller parameter values are tuned using metaheuristic optimization algorithms such as Particle Swarm Optimization (PSO) technique, Genetic Algorithm (GA), *Cuckoo* Search (CS), Artificial Bee Colony (ABC), Firefly Algorithm (FFA), Bat Algorithm (BA), etc., [18, 33-36]. Authors in [33] proposed adaptive controller for a self-balancing robot where the controllers PID and LQR parameters are tuned using colonial selection algorithm (CSA) and performance evaluation has been done using simulation results. PSO based LQR controller for self-balancing mobile robot is proposed in [37] where simulation-based effeteness verification has been done. The research in [38] proposed a modified integral sliding mode controller based on neural network to enhance the performance of conventional sliding mode controller, modified PSO and modified Cuckoo search algorithms are used to tune controller parameters. The effectiveness of the proposed controller is verified based on the numerical simulation results of the overall system. Classical PID, GA tuned PID and Model Predictive controllers are proposed to control a self-balancing Segway [39]. The performance evaluation based on the online control showed that model predictive controller has superior performance. GA and bacteria foraging optimization (BFO) tuned optimal LQR controller for a Self-balancing Segway has been proposed in [40]. The performance comparison showed that BFO-LQR has reduced percentage overshoot and rise time relative to GA-LQR. Fuzzy logic and GA tuned LQR controller for a two-wheeled mobile robot has been proposed in [41]. Summation based performance evaluation showed superior performance of fuzzy-GA-LQR compared to Fuzzy-LQR controller.

In this research, metaheuristic optimization algorithms such GWO, GA and PSO are formulated to tune optimal combinations of controller parameter values of STSMC for balancing and PID for direction control application of a self-balancing Segway by which the impacts of design technique, skill and experience of the designer on the effectiveness of the controller can be reduced. The complete system has been modelled and implemented in MATLAB/Simulink. The performance comparison of the corresponding tuned controllers (STSMC and PID) using GA, GWO and PSO has been done using time domain performance comparison criterions of rise time and settling time under different test conditions of drive mass and initial tilt angles (pitch and yaw) variations of the Segway. The contents of the research are organized in to six sections. The second section is all about the description of the Segway, parts, principle and the control system architecture of the self-balancing Segway. The mathematical modelling tasks are presented in section three of the research. Section four is all about the problem formulation and the corresponding controllers tuning task for the specified application using the proposed optimization algorithms. The results and the corresponding discussions are summarized in section five of the research and section six is the conclusions of the research.

2. SELF-BALANCING SEGWAY

In general, a self-balancing scooter (can be hover board, Segway, electric scooter or self-balancing board) is a two-wheel self-balancing transport system that comprises of two wheels driven by two electric motors and the body of the scooter with the steer handle bars. The wheels are connected to pads where the driver puts their feet to control different movements of the scooter such as forward/backward movements and left/right direction turnings of the system. When the passenger needs to move backward or forward, the passenger or the driver simply leans backward or forward. If the driver needs left or right turn, tuning the lean steer handlebars to the right or left direction is required. If the driver stands upright, the Segway stops. The Segway, apart from the wheels, the body with handlebars and the electric motors, comprises of a series of sensors and a microcomputer system where the control algorithms and other required features are programmed. The Segway system controller knows the situation of the driver lean backward, forward, upright position or tuning the handlebars right or left and takes corrective action where this is possible using the proper coordination of the microcomputer with the control algorithm and the sensors. Fig. 1 shows the schematic diagram of the possible riding positions of the Segway [2, 42].

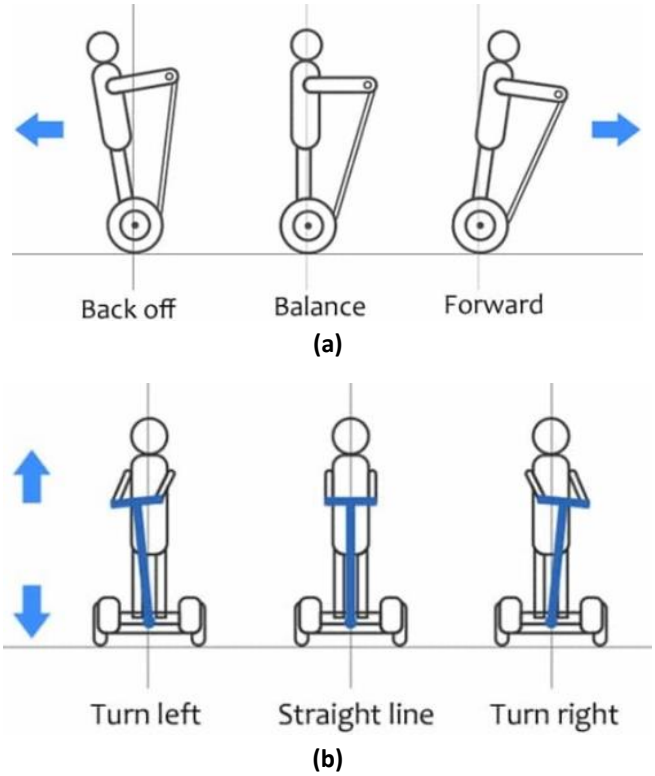

Fig. 1. Segway positions for different ridings: a) forward and backward rid positions; b) left and right turns.

In this research, STSMC is proposed for the balancing and PID control scheme is proposed for right and left tuning of the Segway where the corresponding control scheme parameter values are tuned using metaheuristic optimization algorithms of GWO, GA and PSO. In block diagram form, the complete Segway control system and the components is presented in Fig. 2. The block diagram comprises of the plant (the Segway and actuators), the balance controller, STSMC and direction controller, PID. The errors, e_1 and e_2 are the errors for the pitch and yaw angles respectively. U_1 and U_2 are control inputs to the actuators of the

Segway using STSMC and PID controllers respectively. θ_{ref} and δ_{ref} are reference pitch and yaw angles respectively.

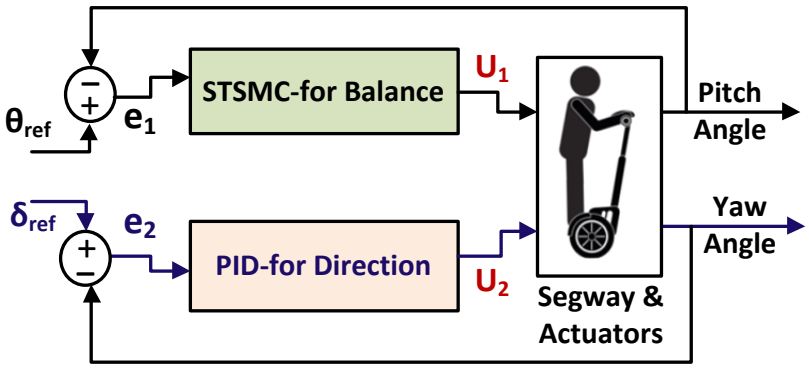

Fig. 2. Block diagram of the Segway control system.

3. MATHEMATICAL MODEL OF THE SEGWAY

Mathematical modeling of a system can simplifies the required analysis and controller design tasks where accuracy of the model affects the overall performance of the system. So, producing comprehensive mathematical model of the Segway that incorporate possible behaviors of the system can improve overall accuracy of the analysis and controllers [11-13, 15, 16, 27]. The Segway comprises of the physical body and two-wheel driven by DC motors as shown by the free body diagram in Fig.3. The free body diagram for the two wheels is given in Fig. 3(a) and for the Segway is given in Fig. 3(b) [11-13, 26]. Table 1 gives summary of parameters, values and descriptions for the Segway [26].

Table 1. Parameters of the Segway.

Symbol	Value/unit	Description
θ	Degree	Pitch angle
δ	Degree	Yaw angle
M_W	6 Kg	Mass of the wheel
M_B	90 Kg	Mass of the body
R	0.2 m	Radius of the wheel
L	1 m	Distance between z-axis and gravity centre of Segway
D	0.6 m	Distance between the contact patches of the wheel
g	9.8 m/s²	Acceleration due to Gravity
T_L, T_R	N.m	Left and right wheel input torques
H_{TL}, H_{TR}	N	Friction between the two wheels and the ground
H_L, H_R	N	Reaction forces on both wheels
J_{TL}, J_{TR}	N.m	Rotating body Inertia moments along x-axis
θ_{WL}, θ_{WR}	rad	Left and right wheels pitch angles
J_B	N.m	Chassis Inertia moment along z-axis

Mathematical modelling process of a complete Segway system needs many equations and as much as possible, unnecessary substitutions, re-arrangements, additions, multiplications, and derivative stages are reduced by incorporating the concepts in the equations considered and presented. Eq. (1) shows the expressions for summation of torques (ΣM_O) about x-axis, summation of torques ($J_\delta \ddot{\delta}$) on the chassis along y-axis, summation of

forces ($M\ddot{x}$) on the wheels, summation of forces along y-axis on the chassis (ΣF_y) and the distance travelled by the chassis (x_B).

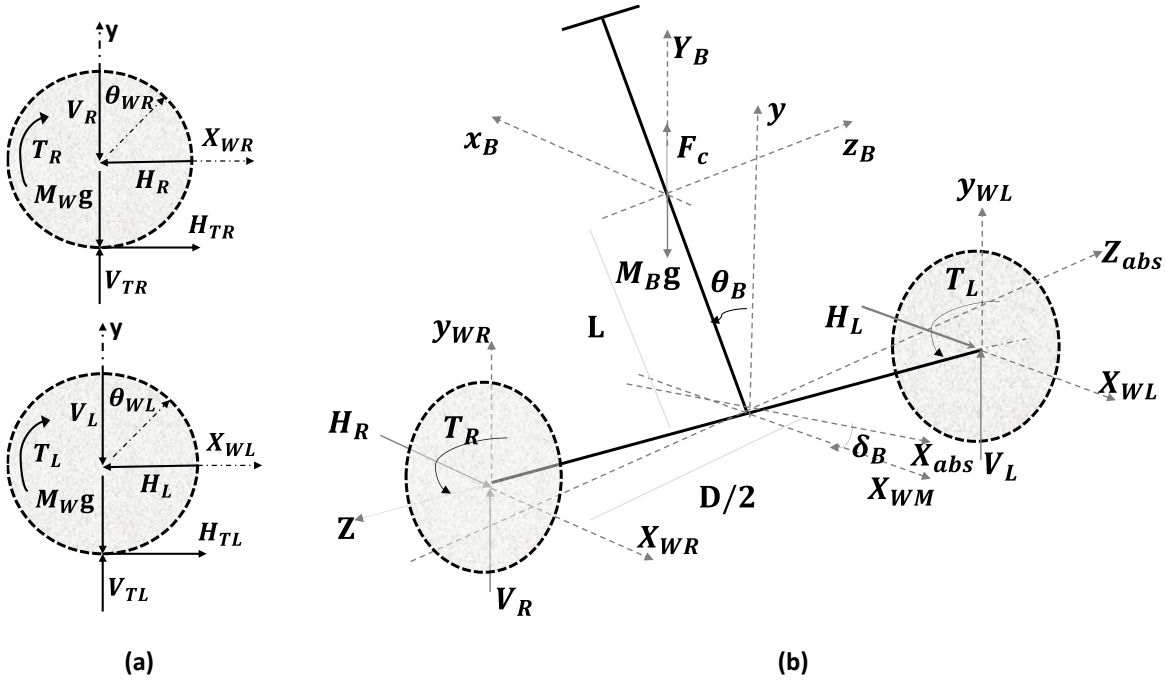

Fig. 3. Free body diagrams of the self-balancing Segway: a) right and left wheels; b) scooter body [11, 27].

$$\begin{cases} \sum M_O = I\alpha = J_\theta \ddot{\theta}_B = (V_L + V_R)L\sin(\theta_B) - (H_L + H_R)L\cos(\theta_B) - (T_L + T_R) \\ J_\delta \ddot{\delta} = \dfrac{D}{2}(H_L - H_R) \\ \sum F_x = ma = M_B \ddot{x}_B = H_L + H_R \\ \sum F_y = ma = M_B \ddot{y}_B = V_L + V_R - M_B g + \dfrac{T_L + T_R}{L}\sin(\theta_B) \\ x_B = L\sin(\theta_B) + \dfrac{x_{WL} + x_{WR}}{2} \end{cases} \quad (1)$$

Applying Newton's second law of motion for the rotating part, Eq. (2) is summary of expressions for summation of moments for both left ($J_{WL}\ddot{\theta}_{WL}$) and right ($J_{WR}\ddot{\theta}_{WR}$) wheels, and segment distance for both wheels (x_{WL}) and (x_{WR}).

$$\begin{cases} \sum M_O = I\alpha = J_{WL}\ddot{\theta}_{WL} = T_L - H_{TL}R \\ \sum M_O = I\alpha = J_{WR}\ddot{\theta}_{WR} = T_R - H_{TR}R \\ x_{WL} = \theta_{WL}R \\ x_{WR} = \theta_{WR}R \end{cases} \quad (2)$$

Following similar laws of motion, Eq. (3) gives expressions for summation of acting forces on both left and right wheels along x (ΣF_x) axis and y (ΣF_y) axis.

$$\begin{cases} \sum F_x = ma = M_W \ddot{x}_{WL} = H_{TL} - H_L \\ \sum F_x = ma = M_W \ddot{x}_{WR} = H_{TR} - H_R \\ \sum F_Y = ma = M_W y_{WL} = V_{TL} - V_L - M_W g \\ \sum F_Y = ma = M_W y_{WR} = V_{TR} - V_R - M_W g \end{cases} \quad (3)$$

Average of segment length that is circumference of both wheels characterized by linear moment of the chassis (x_{WM}), the yaw angle (δ) that is due to rotation rate of wheels change and moment of inertia of the wheels ($J_{wR} = J_{wL}$) where the total moment of inertia can be summation of all moments of inertia involved such as the chassis and riser's moment of inertia in addition to the wheel moment of inertia are summarized by expressions in Eq. (4).

$$\begin{cases} x_{WM} = \dfrac{x_{WL} + x_{WR}}{2} \\ \delta = \tan^{-1}\left[\dfrac{x_{WL} - x_{WR}}{D}\right] \\ J_{WR} = J_{WL} = \dfrac{1}{2} M_{WL} R^2 \end{cases} \quad (4)$$

The schematic diagram representation of the DC motor used as actuator in the Segway is given in Fig. 4 [26] and Table 2 shows the technical data sheet of the corresponding DC-motor. In the diagram, v_a is the applied input voltage to the motor armature, i_a is the armature current flowing and $T_{L/R}$ is the generated torques for both the left and right motors.

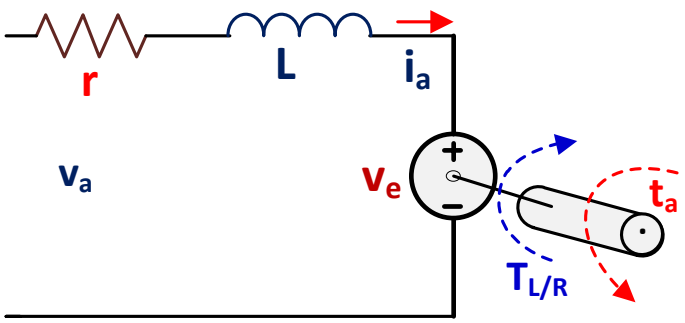

Fig. 4. Schematic diagram of the DC motor used as actuator in the Segway.

Table 2. DC motor parameters.

Symbol	Description	Value
r	Armature Resistance	2.5 Ω
L	Armature Inductance	0.5 mH
V_a	Armature Voltage	24 V
K_t	DC motor back emf. constant	0.58 Vs/rad
K_m	DC motor torque constant	0.58 Nm/A
J_R	Moment inertia of the motor	0.02 kg.m^2/s^2

Applying Kirchhoff's voltage law, ohms law, Newton's second law of motion and by doing the necessary substitutions and re-arrangements, the summary of expressions for the torque generated ($T_{L/R}$) and the back emf (*ve*) linearly related to wheel angular velocity where K_f is considered as the friction constant and t_a is the load torque are described by Eq. (5).

$$\begin{cases} T_{L/R} = k_m i_a \\ v_e = K_e \dot{\theta}_{WL/WR} \\ v_a - r i_a - L_{ia} \dfrac{di_a}{dt} - v_e = 0 \\ T_{L/R} - K_f \dot{\theta}_{WL/WR} - t_a = J_R \dot{\theta}_{WL/WR} \end{cases} \tag{5}$$

Based on the expressions in Eqs. (1)-(5), doing the required substitutions, derivatives and multiplications, Eq. (6) gives summary of expressions for horizontal reaction forces between the wheels and the ground (H_{TL} and H_{TR}) that is described in terms of the segment of both wheels (x_{WL} and x_{WR}) and summation of the left and right wheels moment ($J_{WL}\ddot{\theta}_{WL}$ and right $J_{WR}\ddot{\theta}_{WR}$), the armature current (i_a) derived using the relations between the motor voltages (left, v_L and right, v_R), the back emf (ve) and the control torques for the left (T_L) and right (T_R) wheels by assuming the inductance L is zero.

$$\begin{cases} H_{TL} = \dfrac{T_L R - J_{Wl} \ddot{x}_{WL}}{R^2} \\ H_{TR} = \dfrac{T_R R - J_{Wr} \ddot{x}_{WR}}{R^2} \\ i_a = \dfrac{v_a}{r} - \dfrac{K_e \dot{\theta}_{WL/WR}}{r} \\ T_{L/R} = k_m i_a = J_R \dot{\theta}_{WL/WR} \end{cases} \tag{6}$$

Approximating the friction constant and load torque to zero, expressions for the left and right motor torques ($T_{L/R}$) and expressions for control torques (T_θ and T_δ) for the pitch and yaw angles that is a function of control voltages (v_θ and v_δ) for the pitch and yaw angles the left and right motor control torques (T_L and T_R), the chassis linear moment position (xwm) and yaw angle (δ) are described in Eq. (7) where control voltages for the pitch and yaw angles are function of both the left and right wheel motors, $v_\theta = v_L + v_R$ and $v_\delta = v_L + v_R$ that need decoupling to reduce loop interaction and simplify controllers design.

$$\begin{cases} T_L = \dfrac{k_m v_L}{r} - \dfrac{k_m K_e \dot{x}_{WL}}{rR} \\ T_R = \dfrac{k_m v_r}{r} - \dfrac{k_m K_e \dot{x}_{WR}}{rR} \\ T_\theta = T_L + T_R = \dfrac{k_m}{r}(v_\theta) - \dfrac{2k_m K_e}{Rr}(\dot{x}_{WM}) \\ T_\delta = T_L - T_R = \dfrac{k_m}{r}(v_\delta) - \dfrac{k_m K_e}{Rr}\left(D \sec^2(\delta)\dot{\delta}\right) \end{cases} \tag{7}$$

Summarizing the intermediate tasks of multiplication, subtraction, derivatives and substitutions, the differential equations for the Segway are given in Eq. (8).

$$\begin{cases} d(\theta)\ddot{\theta} = h(\theta)\dot{\theta}^2 - k(\theta)U_1 \\ \ddot{\delta} = \dfrac{6}{(M_B + 9M_W)DR} U_2 \\ z(\theta)\ddot{x}_{WM} = w(\theta)\dot{\theta}^2 - x(\theta)U_1 \end{cases} \tag{8}$$

where,

$$d(\theta)=(2M_w+M_B)-\frac{3}{4}\frac{(M_BL\cos(\theta)+M_wR)\cos(\theta)}{L}$$

$$h(\theta)=\frac{3}{4}\left(\frac{g(2M_W+M_B)\sin(\theta)}{L}-M_B\sin(\theta)\cos(\theta)\right)$$

$$k(\theta)=\frac{3}{4}\left(\frac{(2M_W+M_B)(1+sin^2(\theta)}{M_BL^2}+\frac{\cos(\theta)}{RL}\right)$$

$$z(\theta)=-\frac{3}{4}\frac{\cos(\theta)(M_BL\cos(\theta)+M_WR)}{L}+(2M_w+M_B)$$

$$w(\theta)=-\frac{3}{4}\frac{(M_BL\cos(\theta)+M_WR)g\sin(\theta)}{L}+M_BL\sin(\theta)$$

$$x(\theta)=\frac{3}{4}\frac{(M_BL\cos(\theta)+M_WR)(1+sin^2(\theta))}{M_BL^2}+\frac{1}{R}$$

$$U_1=T_L+T_R$$

$$U_2=T_L-T_R$$

For the state variables, $x_1=\theta$, $x_2=\dot{\theta}$, $x_3=x$, $x_4=\dot{x}$, $x_5=\delta$ and $x_6=\dot{\delta}$, state equations of the Segway can be given by expressions in Eq. (9).

$$\begin{cases}\dot{x}_1=x_2\\ \dot{x}_2=f_1(x_1)+f_2(x_1,x_2)+g_1(x_1)U_1\\ \dot{x}_3=x_4\\ \dot{x}_4=f_4(x_1)+f_4(x_1,x_2)+g_2(x_1)U_1\\ \dot{x}_5=x_6\\ \dot{x}_6=g_3U_2\end{cases} \tag{9}$$

where,

$$f_1(x_1)=\left(\frac{\frac{-0.75gsin(x_1)}{L}}{\frac{0.75(M_WR+M_BL\cos(x_1))\cos(x_1)}{(2M_W+M_B)L}-1}\right)$$

$$f_2(x_1,x_2)=\left(\frac{\frac{0.75M_BLsin(x_1)\cos(x_1)}{(2M_W+M_B)L}x_2{}^2}{\frac{0.75(M_WR+M_BL\cos(x_1))\cos(x_1)}{(2M_W+M_B)L}-1}\right)$$

$$f_3(x_1)=\left(\frac{-3g(M_WR+M_BLcosx_1)\sin x_1}{(2M_W+M_B)4L-3(M_WR+M_Bcosx_1)\cos x_1}\right)$$

$$f_4(x_1,x_2)=\left(\frac{M_B\sin(x_1)x_2{}^2}{2M_W+M_B-\frac{3(M_WR+M_BLcosx_1)\cos x_1}{4L}}\right)$$

$$g_1(x_1)=\begin{pmatrix}\dfrac{0.75\left(1+\left(\sin(x_1)\right)^2\right)}{M_B L^2}+\dfrac{0.75\cos(x_1)}{(2M_W+M_B)RL}\\ \dfrac{0.75\left(M_W R+M_B L\cos(x_1)\cos(x_1)\right)}{\left(2M_{W+M_B}\right)L}-1\end{pmatrix}$$

$$g_2(x_1)=\begin{pmatrix}\dfrac{0.75\left(M_W R+M_B L\cos x_1\right)\left(1+\sin(x_1)^2\right)}{M_B L^2}+\dfrac{1}{R}\\ 2M_W+M_B-\dfrac{0.75\left(M_W R+M_B L\cos x_1\right)\cos x_1}{L}\end{pmatrix}$$

$$g_3=\frac{6}{(M_B+9M_W)DR}$$

Loop interaction in multi-input multi output (MIMO) systems is inherent feature that makes difficult to adapt the control design techniques like the case in single input single output (SISO) systems.

Thus, unwanted cross interaction effect among the loops can be removed using an appropriate decoupling technique by which the corresponding controllers design can be done based on the SISO design approaches even though interaction suppression among the loops is a challenge. Literatures summarize the decoupling techniques into dynamic, static and approximate decoupling approaches.

Dynamic decoupling can eliminate the interaction among the loops, static decoupling can eliminate interaction in the steady-state condition and approximate decoupling approach approximates the transfer function approximately equal to the bandwidth where the loop interaction is too much around that frequency.

Thus, Eq. (10) is used as decoupling transformation for v_θ and v_δ ($v_\theta = V_\theta$, $v_\delta = V_\delta$) to the left, v_L and right, v_R ($v_L = V_{aL}$, $v_R = V_{aR}$) wheel motors voltage and it is also the approximation of voltage for the pitch and yaw angles [26, 27].

$$\begin{cases}V_\theta = V_{aL}+V_{aR}\\ V_\delta = V_{aL}-V_{aR}\end{cases} \tag{10}$$

Further re-arrangement and substations we can get the expressions for the right and left wheel motors input voltage expressions as shown in Eq. (11).

$$\begin{cases}V_{aL}=0.5(V_\theta+V_\delta)\\ V_{aR}=0.5(V_\theta-V_\delta)\end{cases} \tag{11}$$

Based on the interaction level and the corresponding voltage assumption expressions in Eq. (11), the system with decoupler is represented by the block diagram in Fig. 5. It is a two input (reference pitch angle, θ_{ref} and reference yaw angle, δ_{ref}) and two output (angle, θ and yaw angle, δ) MIMO system where the pitch angle controller, STSMC, is to keep the balance and PID is used for direction control of the Segway.

The controllers output is input to the decoupling unit and the output is applied to the corresponding DC motors so that necessary torque can be produced for the balancing and direction control of the Segway. In the diagram, $D_1 = D_2 = D_3 = 0.5$ and $D_4 = -0.5$.

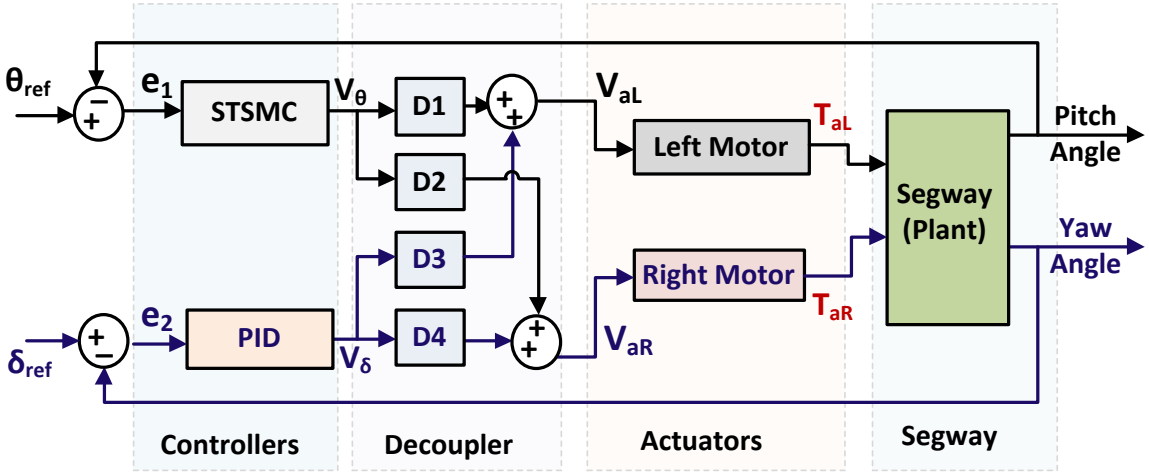

Fig. 5. Block diagram of the system with decoupler.

4. CONTROLLERS' DESIGN

Summary of related research articles about control of the self-balancing Segway done in section two of this research shows that robustness of the multivariable, unstable and nonlinear systems can be achieved using nonlinear controllers, adaptive controller, optimal and hybrid controllers where the design technique, the skill and experience of the designer can affect the performance of the controllers. In this research, optimal combination of parameter values STSMC for balancing and PID for direction control of the Segway respectively is tuned using metaheuristic optimization algorithms of GWO, PSO and GA. Subsections below are summary of the corresponding controller algorithms, the optimization algorithms, problem formulations, the tuning procedures, the required parameters values, convergence curves and tuned controller parameter values using the corresponding algorithms.

4.1. Balancing and Direction Controller for the Segway

The Segway control system controls the balance using STSMC and direction using PID controller. The STSMC has three unknown parameters and the PID controller has three unknowns to be determined using the proposed optimization algorithms.

i) STSMC for Segway balancing control: Sliding mode control is widely known as a nonlinear control algorithm having the features of robust, simple design/tuning approach and less processor computation requirement for practical implementation. There are first order and higher order sliding mode control algorithms where the higher order is advancement to the first order algorithm having the advantage of eliminating chattering problem of the first order sliding mode control algorithm. Sliding mode control has two basic advantages such as the system response will not be affected by uncertainties/disturbances and by selecting specific sliding function; the dynamic behaviour of the system can be modified. Apart from handling nonlinearity, robustness, simplicity and less computation requirement feature, sliding mode control can reduce complexity of feedback design by allowing by decoupling the system to separate, partial and small components. Eq. (12) shows the expression for the switching control for the conventional sliding mode control algorithm where k is a positive constant and sgn (s) is a symbolic function [20, 24, 30].

$$u_{sw} = -ksgn(s)$$

where

$$sgn(s)=\begin{cases}-1 & if \quad s<0\\ 0 & if \quad s=0\\ +1 & if \quad s>0\end{cases} \tag{12}$$

The expression in Eq. (12) can have chattering problem especially at high switching frequency. The chattering problem of first order sliding mode controller can solved by smooth approximation to substitute the discontinuous sign with an equivalent continuous smooth approximation. However, this approach can make the controller to loss robustness. Second order sliding mode controllers can solve the problem of chattering and keep the robustness. Supper twisting sliding mode controller is among second order sliding mode controllers that can eliminate chattering and is robust [20, 24]. In this research, the stability proof of the proposed STSMC is done using quadratic like Lyapunov functions by which explicit relation for the controller parameters can be obtained. Sliding mode controller design has two steps of sliding surface and control input design steps. Accordingly, Eq. (13) is expression for the sliding surface where, s is sliding surface, e is error and c must satisfy the Hurwitz condition

$$s=ce+\dot{e} \tag{13}$$

Eq. (14) shows the expressions for pitch angle tracking error where $x_{1ref}=\theta_{ref}$, the reference value of pitch angle.

$$\begin{cases}e=x_1-x_{1ref}\\ \dot{e}=\dot{x}_1-\dot{x}_{1ref}\\ \ddot{e}=\ddot{x}_1-\ddot{x}_{1ref}=\dot{x}_2-\ddot{x}_{1ref}\end{cases} \tag{14}$$

Thus, we have expressions in Eq. (15).

$$\begin{cases}\dot{s}=c\dot{e}+\ddot{e}=c\left(\dot{x}_1-\dot{x}_{1ref}\right)+\dot{x}_2-\ddot{x}_{1ref}\\ \dot{x}_2=f_1\left(x_1\right)+f_2\left(x_1,x_2\right)+g_1\left(x_1\right)U_1\\ \dot{s}=c\left(\dot{x}_1-\dot{x}_{1ref}\right)+f_1\left(x_1\right)+f_2\left(x_1,x_2\right)\\ \qquad +g_1\left(x_1\right)U_1-\ddot{x}_{1ref}\end{cases} \tag{15}$$

Eq. (16) shows the expression for the Lyapunov candidate function. For the balancing Segway to be on the surface, stability requirement of Lyapunov, i.e., $\dot{s}s<0$, must be satisfied.

$$\begin{cases}V=\dfrac{1}{2}s^2\\ \dot{V}=s\dot{s}=s\begin{pmatrix}c\left(\dot{x}_1-\dot{x}_{1ref}\right)+f_1\left(x_1\right)\\ +f_2\left(x_1,x_2\right)+g_1\left(x_1\right)U_1-\ddot{x}_{1ref}\end{pmatrix}\end{cases} \tag{16}$$

Thus, the conventional sliding mode controller can be expressed by Eq. (17) where c *and* η are constants.

$$U_1=\frac{c\left(\dot{x}_{1ref}-\dot{x}_1\right)-f_1\left(x_1\right)-f_2\left(x_1,x_2\right)+\ddot{x}_{1ref}}{g_1\left(x_1\right)} -\eta sgn\left(s\right) \tag{17}$$

The two parts of the conventional sliding mode control in Eq. (17) is given by Eq. (18).

$$\begin{cases} u_c = \eta sgn(s) \ \ and \\ u_{eq} = \dfrac{c\left(\dot{x}_{1ref} - \dot{x}_1\right) - f_1\left(x_1\right) - f_2\left(x_1, x_2\right) + \ddot{x}_{1ref}}{g_1\left(x_1\right)} \end{cases} \quad (18)$$

Like the conventional sliding mode control algorithm in Eq. (18), STSMC algorithm in Eq. (19) comprises of two parts where the first part is a discontinuous function of sliding variable and the second part is a continuous function of its derivative to eliminate chattering where C_1 and C_2 are constants [20, 24].

$$\begin{cases} u_c = -C_1 \left|s\right|^{\frac{1}{2}} sgn(s) + v \\ \dot{v} = -C_2 sgn(s) \end{cases} \quad (19)$$

The proposed STSMC can be described in compact from by Eq. (20).

$$U_1 = u_{eq} + U_c \quad (20)$$

By substitution, the proposed STSMC can be expressed by Eq. (21) where C_1, C_2 and c are parameters of the proposed controller for balancing.

$$U_1 = \frac{c\left(\dot{x}_{1ref} - \dot{x}_1\right) - f_1\left(x_1\right) - f_2\left(x_1, x_2\right) + \ddot{x}_{1ref}}{g_1\left(x_1\right)} - C_1 \left|s\right|^{\frac{1}{2}} sgn(s) - C_2 \int_0^t sgn(s)\, ds \quad (21)$$

(ii) PID controller for direction control of Segway: PID control algorithm is known as simple to design, less complex for implementation and widely applicable to different processes control application. Eq. (22) shows the mathematical representation of the algorithm for the direction control application of the Segway where, δ_{ref} is the reference yaw angle, δ_{ac} is measured value of yaw angle and k_P, k_i and k_d are PID controller gains.

$$\begin{cases} U_2 = k_P e(t) + k_i \int e(t)\, dt + k_d \dfrac{de(t)}{dt} \\ \quad where, \ e(t) = \delta_{ref} - \delta_{ac} \end{cases} \quad (22)$$

The robustness of the PID and STSMC controller is highly dependent on the technique, the skill and experience of the designer. The impact of these factors can be solved by using metaheuristic optimization techniques so that optimal combinations of the corresponding controller parameter values to achieve robustness can be tuned.

4.2. STSMC and PID Controllers Tuning Using GWO, PSO and GA

The balancing and direction controllers' parameter values in this research are proposed to be tuned using optimization algorithms of GA, PSO and GWO. Accordingly, GWO, PSO and GA are formulated to tune optimal controller parameters of STSMC for balancing and PID for direction control of Segway.

4.2.1. Overview of GWO, PSO and GA

Recent rapid development of metaheuristic optimization algorithms such as Particle Swarm Optimization (PSO) technique, Genetic Algorithm (GA), Cuckoo Search (CS), Artificial Bee Colony (ABC), Firefly Algorithm (FFA), Bat Algorithm (BA), etc., increased the opportunity of designing/tuning optimal variable for a certain application [34, 36].

(i) Grey-wolf algorithm: The grey wolf algorithm is among nature inspired metaheuristic optimization algorithms inspired by hierarchy, structured living style and hunting for prey behaviours of grey wolfs. The grey wolfs pack consists of four packs such as Alpha (α), Beta (β), Delta (δ), and Omega (ω) where their role in the pack is different as shown in Fig. 6 [36, 43].

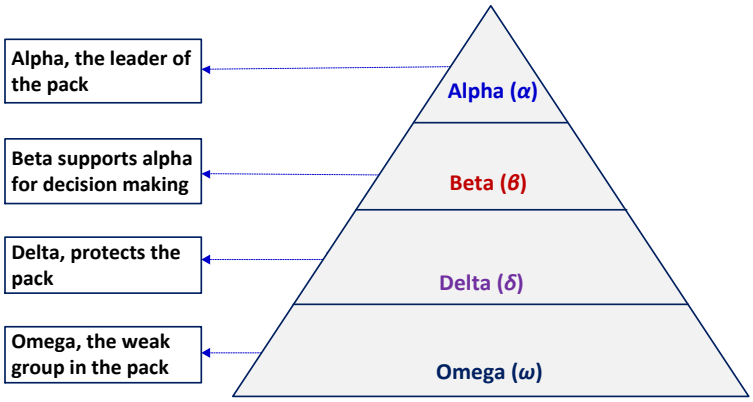

Fig. 6. Hierarchy of grey wolfs.

From the hierarchy wolfs diagram in Fig. 6, alpha (α) wolfs play the leading role, beta (β) ranking of the hierarchy represents the wolfs that are dedicated in assisting the alpha ranking of wolfs in decision making and other activities. Beta wolf accepts order from alpha wolf, confirms and feedback to the alpha wolf. The beta wolves' gender can be male or female and these wolfs will be candidates to become alpha when alpha wolves become old or die. The third rank, the delta (δ) wolfs are scouts, care takers, elders, sentinels and hunters and they are dominant over the omega wolf (the lowest ranking of the fray wolf). The social hierarchy, the tracking, the encircling, and the prey attacking activities of the gray wolves are considered in the mathematical modeling of the algorithm [36, 43-44].

Social hierarchy: In the social hierarchy of gray wolves, alpha (α) is considered as the fitness solution, beta (β) is the second-best solution, delta (δ) is the third best solution and other candidate solutions are considered as omega (ω). In modeling gray wolves as an optimization algorithm, the hunting process is guided using α, β, δ and the omega (ω) wolves follow the three wolves**.**

Encircling prey: The gray wolves encircle their target during hunting operation and this behavior can be described the expressions in Eq. (23).

$$\begin{cases} \vec{D} = \left| \vec{C}.\vec{X}_p(t) - \vec{X}(t) \right| \\ \vec{X}(t+1) = \vec{X}_p(t) - \vec{A}.\vec{D} \end{cases} \quad (23)$$

where, t *is* current iteration, $\vec{A}$ and $\vec{C}$ are vector coefficients, $\vec{X}_P$ is position vector of the prey, and $\vec{X}$ is position vector of a grey wolf. Eq. (24) shows the expressions to calculate the vectors, $\vec{A}$ and $\vec{C}$. The components of $\vec{a}$ are linearly vary from 2 to 0 in the range of iterations $\vec{r}_1$ and $\vec{r}_2$ that are random vectors range in between [0, 1].

$$\begin{cases} \vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a} \\ \vec{C} = 2.\vec{r}_2 \end{cases} \quad (24)$$

Hunting: The best solutions saved so far are used for mathematical representation of the gray wolves hunting action and based on the best search agents' status, the remaining search

agents including omega adjust their position. The expressions in Eq. (25) are suggested for these activities.

$$\begin{cases} \vec{D}_\alpha = \left|\vec{C}_1.\vec{X}_\alpha - \vec{X}\right|, \ \vec{D}_\beta = \left|\vec{C}_2.\vec{X}_\beta - \vec{X}\right|, \ \vec{D}_\delta = \left|\vec{C}_3.\vec{X}_\delta - \vec{X}\right| \\ \vec{X}_1 = \vec{X}_\alpha - \vec{A}_1.(\vec{D}_\alpha), \ \vec{X}_2 = \vec{X}_\beta - \vec{A}_2.(\vec{D}_\beta), \ \vec{X}_3 = \vec{X}_\delta - \vec{A}_3.(\vec{D}_\delta) \\ \vec{X}(t+1) = \dfrac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \end{cases} \quad (25)$$

Attacking prey (exploitation): The hunting process of the prey by wolfs will end by attacking the prey when the prey stops moving. Convergence of the candidate solution will not be guaranteed, and convergence will be assured when $|A| \geq 1$. To assure convergence, the value for $|A| < 1$ is recommended.

Search for prey (exploitation): The search process of the grey wolfs is based on the position of alpha, beta and delta. The wolfs move randomly to search for the prey and come to converge where there is prey and to attack the prey. To diverge from the prey, the control parameter A should be greater than one or less than -1.

Grey wolfs hunting process is summarized as searching for the prey, tracking, chasing, approaching the prey, pursing, encircling, harassing the prey until the prey stop movement and attacking. The procedures of the GWO algorithm can be summarized as population initialization, parameters (α, A, C, Max Generation), fitness calculation for the three best hierarchy of wolfs, position update, parameters update, update the fitness values, check for termination criterion and converge with optimal tuned values of criterion is reached otherwise the process will be restarted.

(ii) PSO: It is among nature inspired metaheuristic optimization algorithms inspired by the social interactions and movements of insects, birds, and fish. It employs a swarm of particles that navigate the search space to find the best solution. Each particle modifies its movement based on its own history and that of other particles. Application of PSO for tuning or design activities can be summarized by the procedures: population initialization, assign the best values: p_{best} and g_{best}, position and velocity update using the expressions in Eq. (26) and termination based on achievement of maximum number of generations or pre-set termination criterion is met [29, 32-34].

$$\begin{cases} v_i(t) = \theta v_i(t-1) + c_1 r_1 \left(p_{best,i} - x_i(t-1)\right) \\ \qquad\quad + c_2 r_2 \left(g_{best,i} - x_i(t-1)\right) \\ x_i(t) = x_i(t-1) + v_i(t) \end{cases} \quad (26)$$

where, v_i is i^{th} particle velocity at iteration t, x_i is i^{th} particle current position at iteration t, c_1, c_2 are individual and social cognitive constants, θ is inertia weight factor, r_1, r_2 are random numbers between 0 and 1, t is iteration pointer, $p_{best,i}$ is i^{th} particle best value and $g_{best,i}$ is global best.

(iii) GA: GA is among nature inspired metaheuristic optimization algorithms inspired by the biological principal of natural selection, where the strongest individuals will survive and reproduce. In the GA the individual parameters are encoded as strings of numbers called chromosomes. The process starts by creating a random population of potential solutions; these are then evaluated using a fitness function. A weighted roulette wheel selection method is then used to find the strongest members of that population to pass

them through to the next stage, crossover. The crossover stage selects two random parent chromosomes and combines them to form two child chromosomes. Mutation is the final stage, where single elements may be randomly swapped to create a more diverse population. The process then starts again with the new population and repeats until convergence is achieved or the specified number of iterations is reached. In GA's the value of fitness represents the performance which is used to rank 0 and the ranking is then used to determine how to allocate reproductive opportunities. This means that individual with a higher fitness value will have a higher opportunity of being selected as a parent. The fitness function is essentially the objective function for the problem [34, 40, 41].

4.2.2. Objective Function

The problem formulation process is to adapt metaheuristic optimization algorithms for the parameters tuning tasks that need defining appropriate objective function to evaluate the fitness value of the populations of the proposed approach. Availability of different performance indices to be used as objective functions is presented in literatures even though the objective function depends on the type of problem to be solved. Some of the most widely used performance indexes are Integral Absolute Error (IAE), Integral Square Error (ISE), Integral Time Absolute Error (ITAE), Integral Time Square Error (ITSE), Integral Error (IE) and Mean Square Error (MSE). Eqs. (27) and (28) give the corresponding mathematical expressions of the performance indexes described [34, 35, 36].

$$\begin{cases} IAE = \int_0^{\tau} |e(t)| \\ ISE = \int_0^{\tau} |e(t)|^2 \\ ITAE = \int_0^{\tau} t\,|e(t)| \end{cases} \tag{27}$$

$$\begin{cases} ITSE = \int_0^{\tau} t\,|e(t)|^2 \\ IE = \int_0^{\tau} e(t) \\ MSE = \dfrac{\int_0^{\tau} e(t)^2}{t} \end{cases} \tag{28}$$

In this research, the system is a MIMO system and Eq. (29) shows the expression for the objective function that can minimize the error as a sum of two error functions, $e_1(t) = \theta_{ref} - \theta$, $e_2(t) = \delta_{ref} - \delta$ where θ and δ are the pitch and yaw angles respectively.

$$J = ITAE = \int_0^{\tau} t\left(\left|e_1(t)\right| + \left|e_2(t)\right|\right) \tag{29}$$

4.2.3. Algorithms, Control Parameters and Tuned Optimal Values

Based on the objective function, the system model, the parameters that can control the speed and convergence of the corresponding algorithms and the procedure of each algorithm is depicted in Fig. 7. Fig. 7(a) shows the procedures of the corresponding algorithms for the defined application. The system diagram in Fig. 7(b) shows the model and the communications

between the MATLAB/Simulink model and the tuning algorithms. The algorithm controller parameter values are presented in Table 3.

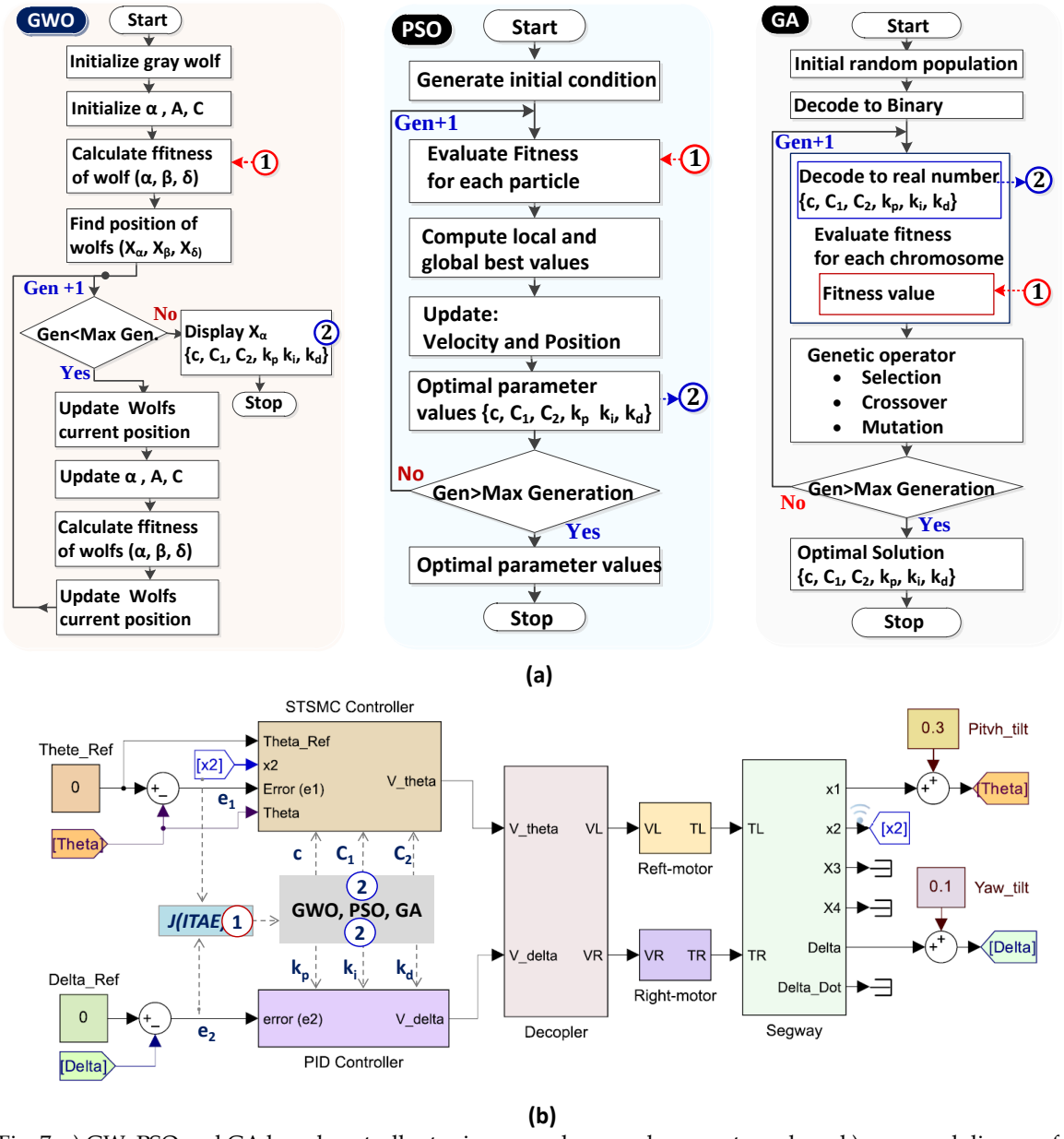

Fig. 7. a) GW, PSO and GA based controller tuning procedures and parameter values; b) proposed diagram for balancing and direction control.

In Figs. 7(a) and (b), there are numbers in circles to indicate the communication between the algorithm and system model during tuning process. The upper and lower bounds of the corresponding controller parameters are summarized in Table 4.

Accordingly, the tuning process has been done and the corresponding convergence curves for each algorithm and tuned controller parameter values of both STSMC and PID are presented in Table 4 where L_b stands for the lower boundary and U_b stands for the upper boundary of the controller parameter values.

The tabular summary shows that for equal number of iterations (30), the best value for each algorithm is 0.011, 0.012 and 0.017 for GWO, PSO and GA based tuning process respectively. From the convergence curve plots in Fig. 8, GWO converges fast compared to PSO and GA based approaches.

The probability and speed of convergence the metaheuristic optimization algorithms is highly dependent on the algorithm controller parameter values selection process where experience of the designer plays the role apart from the computer where the tuning process is to be done [34, 36].

Table 3. Algorithm's control parameters.

Algorithm	Parameter		Value
GWO	Number of search agent		20
	Maximum number of iterations		30
	Random vectors, r_1, r_2		0, 1
	Coefficient vectors, c_1, c_2		0, 2
PSO	Number of particles, P		20
	Maximum number of iterations		30
	Inertia weight (θ)	θ_{min}	0.2
		θ_{max}	0.9
	Individual cognitive (c_1)		1
	Social cognitive (c_2)		2
GA	Population size		20
	Maximum generation		30
	Cross over rate		0.8
	Selection		Roulette wheel

Table 4. Boundary and tuned controller parameters.

Controller	Parameter	Boundary values		Tuned values		
		L_b	U_b	GWO	PSO	GA
STSMC	c	0.0001	15	1.19	0.89	6.21
	C_1	0.0001	15	1.06	4.53	14.98
	C_2	0.0001	15	10.47	7.41	14.97
PID	k_p	0.0001	600	598.38	452.42	535.3
	k_i	0.0001	600	587.71	435.44	332.5
	k_d	0.0001	600	61.81	62.79	94.34
Best values				0.011	0.012	0.017

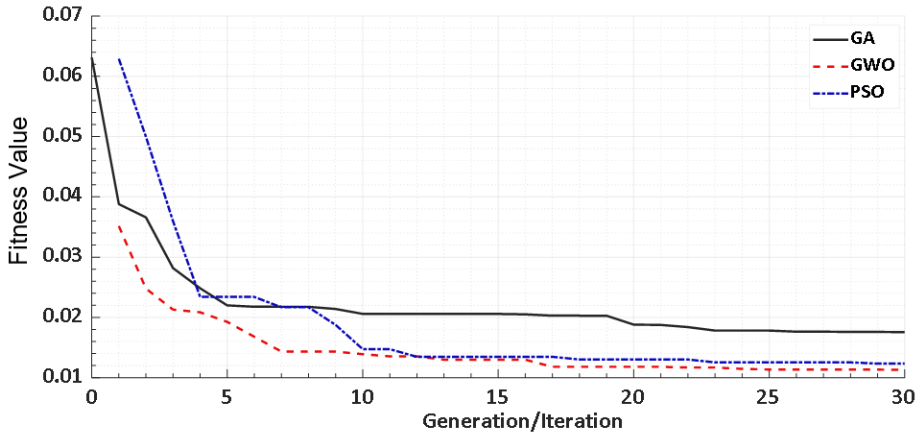

Fig. 8. Convergence curves of GW, PSO and GA for the specified application.

5. RESULTS AND DISCUSSIONS

The overall system implementation is done in MATLAB/Simulink and results analysis is done based on the simulation responses using time domain performance comparison criterions of rise time and settling time for the closed loop system for operating conditions of variable mass and variable initial tilt angles (pitch and yaw angles). The open loop and closed loop system responses are analysed.

5.1. Open-Loop Responses

In literatures, it is summarized that the self-balancing Segway is unstable. Stabilizing and robust controller for the balancing and direction are required. Figs. 9(a) and (b) are the open loop responses for balancing and direction of the Segway respectively. The open loop response plots for the pitch and yaw angle show that the system without controllers is unstable.

5.2. Closed-Loop Responses

The complete closed loop MATLAB/Simulink model of the Segway is given in Fig. 10. In the diagram, STMC is the supper twisting sliding mode controller for balancing of the Segway, the PID block is the corresponding controller for the Segway direction components.

The test conditions considered are mass variations and initial tilt angles (pitch and yaw) variations.

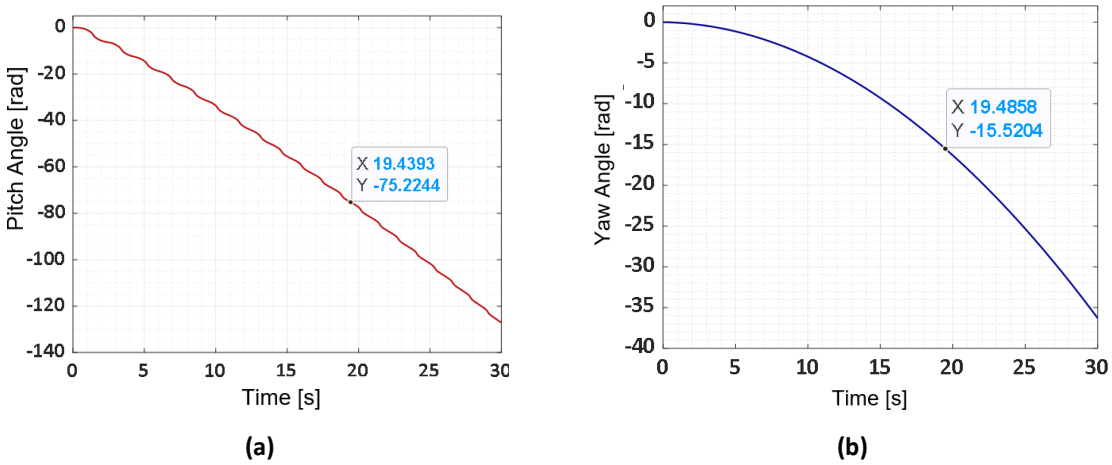

Fig. 9. Segway open-loop responses for: a) balancing; b) direction.

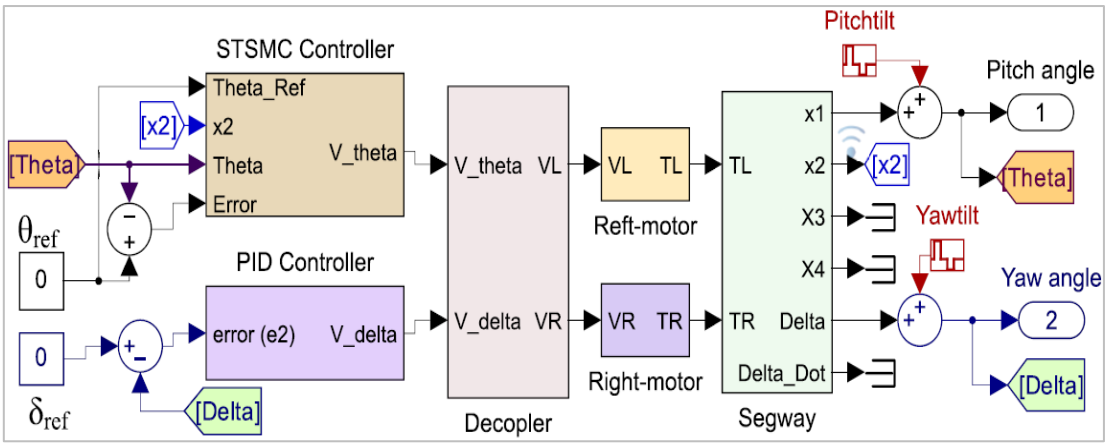

Fig. 10. MATLAB/Simulink block diagram of the system.

5.2.1. *Initial Responses*

The initial response is for the constant tilt angles and constant mass. For the reference values of 0 rad pitch and yaw angles with initial tilt angles of 0.3rad and 0.1 rad pitch and yaw angles respectively, the responses are presented in Fig. 11 where Fig. 11(a) is for the pitch angle response and Fig. 11 (b) is for the yaw angle response plots.

The time domain performance comparison summary is presented in Table 5. Accordingly, for the pitch angle response, rise time is 0.154 sec for GWO-STSMC, 0.161 sec for PSO-STSMC and 0.415 sec for GA-STSMC. Settling time for the pitch angle response is 0.243 sec for GWO-STSMC, 0.275 sec for PSO-STSMC and 0.649 sec for GA-STSMC. At the same time, for the yaw angle response, the recorded rise time is 0.189 sec for GWO-PID, 0.261 sec for PSO-PID and 0.319 sec for GA-PID controller. The recorded settling time for the yaw angle response is also 0.764 sec for GWO-PID, 0.913 sec for PSO-PID and 1.003sec for GA-PID controller.

The quantitative results summary presented in Table 5 for GWO-STSMC and GWO-PID controllers showed relatively quick response that is verified by the relatively small rise and settling times recorded. The second quick response is recorded for PSO tuned balancing and direction controllers.

Table 5. Performance comparison for initial pitch and yaw angles.

Angle	Specification	Algorithm		
		GWO	PSO	GA
Pitch angle at θ = 0.3 rad	Rise time [s]	0.154	0.161	0.415
	Settling time [s]	0.243	0.275	0.649
Yaw angle at δ = 0.1 rad	Rise time [s]	0.189	0.261	0.319
	Settling time [s]	0.764	0.913	1.003

Qualitative analysis shows that controllers tuned by the proposed optimization algorithms can perform well. The relative and small deviations recorded for the rise and settling times are not significant and the deviations are due to nature of the optimization algorithms, the skill and experience of the designer in selecting algorithm controller parameters values.

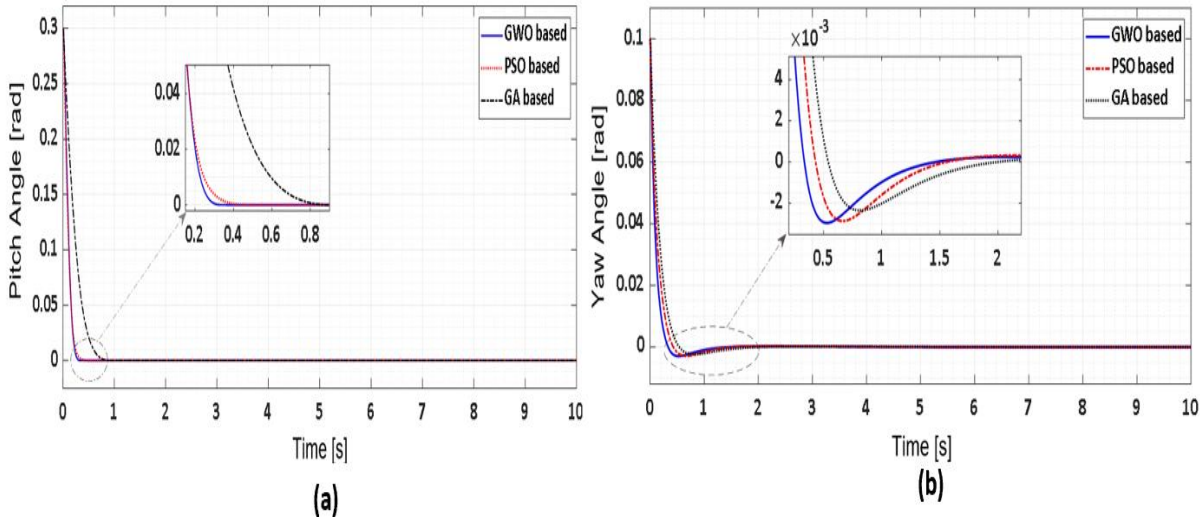

Fig. 11. Initial pitch response with initial θ = 0.3 rad; b) yaw angles response with initial δ = 0.1 rad.

5.2.2. Mass Variation

The driver load variation responses for initial tilt angles of 0.3 rad pitch and 0.1 rad yaw angles are recorded in Fig. 12.

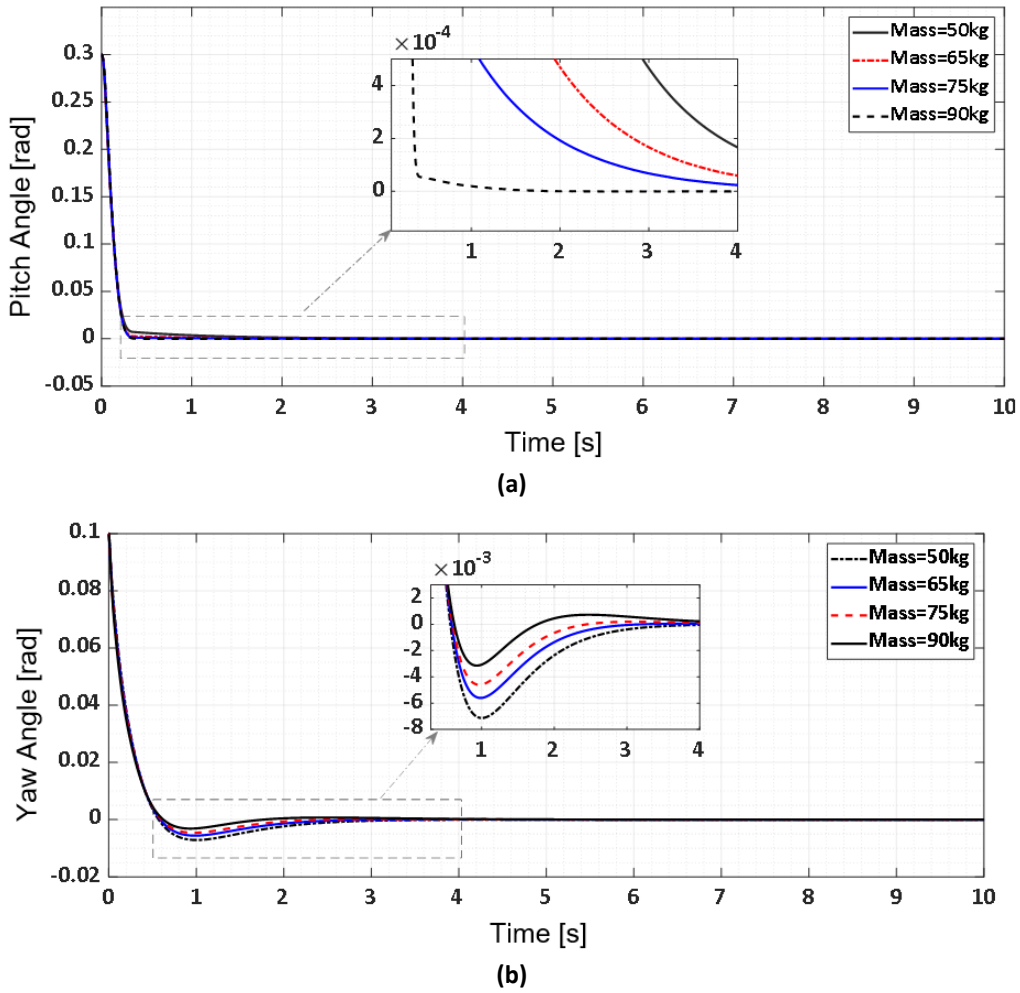

Fig. 12. Pitch and yaw angle responses during mass variation from 50kg to 90kg; a) pitch angle response with initial θ = 0.3 rad for mass variation; b) Yaw angle response with initial δ = 0.1 rad for mass variation.

The mass variations from 50 Kg to 90 Kg pitch angle responses for initial pitch angle of 0.3 rad is given in Fig. 12 (a). The yaw angle responses for initial yaw angle of 0.1 rad and for the mass variations from 50 Kg to 90 Kg are plotted in Fig. 12 (b). For the pitch angle response with the mass variation, the response is coming less sensitive to increase in mass which is due to the robust feature STSMC. For the yaw angle with the mass variations from 50Kg to 90Kg, there is slight sensitivity decrease when the mass increases. The yaw angle controller is PID controller. In summary, we can say that both controllers of STSMC and PID are robust and performed well even though STSMC is less sensitive to mass variations compared to PID controller.

5.2.3. Initial Tilt Angle Variations

For the initial pitch angle variations, $\theta = 0.3(u(t-1) - u(t-3)) - 0.2(u(t-5) - u(t-7))$ the step change in initial pitch angles and the corresponding responses are plotted in Fig. 13. The initial pitch angle as step inputs is given in Fig. 13 (a).

The response plot for the step change in initial angles given in Fig. 13(a) is plotted in Fig. 13(b). Close observation to the results shows that the STSMC for balancing of the Segway is tracking the reference pitch angle for variable initial pitch angles plotted in Fig. 13(a). This capability of tracking the reference pitch angle for variable initial pitch angles as input (disturbance) show the robustness of the proposed controller. From the response plots, GWO-STSMC and PSO-STSMC controllers showed relatively better performance compared to GA-STSMC.

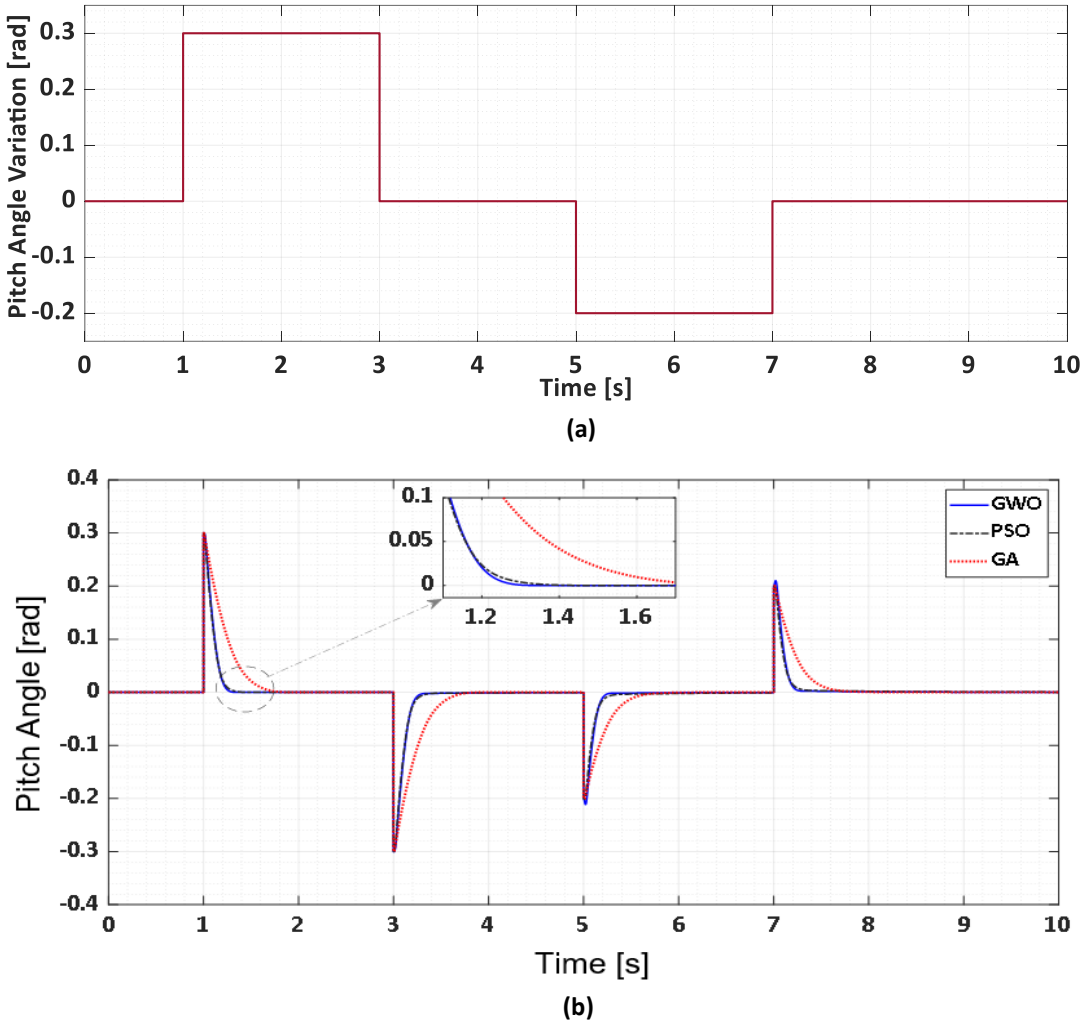

Fig. 13. Pitch angle responses for variable initial pitch angles: a) pitch anagle variation; b) pitch angle responses.

For the initial yaw angle variations, $\delta = 0.1(u(t - 1) - u(t - 3)) - 0.15(u(t - 5) - u(t - 7))$, given as step change as shown in Fig. 14 (a), the corresponding yaw angle responses are recorded in Fig. 14 (b). The response for the corresponding yaw (direction) controllers of GWO-PID, PSO-PID and GA-PID revealed the capability of tracking the reference yaw angle for the given step change in initial yaw angles as disturbances.

Close observation to the response plots shows that GWO-PID based response has relatively reduced sensitivity and quick response to variations of initial tilt angle. However, all the controllers can track the reference yaw angle for the variations in initial yaw angle well. There is slight variation in speed of the response and sensitivity and these variations are due to the nature of the proposed tuning algorithms and the skill/experience of the designer to select the algorithm control parameters values.

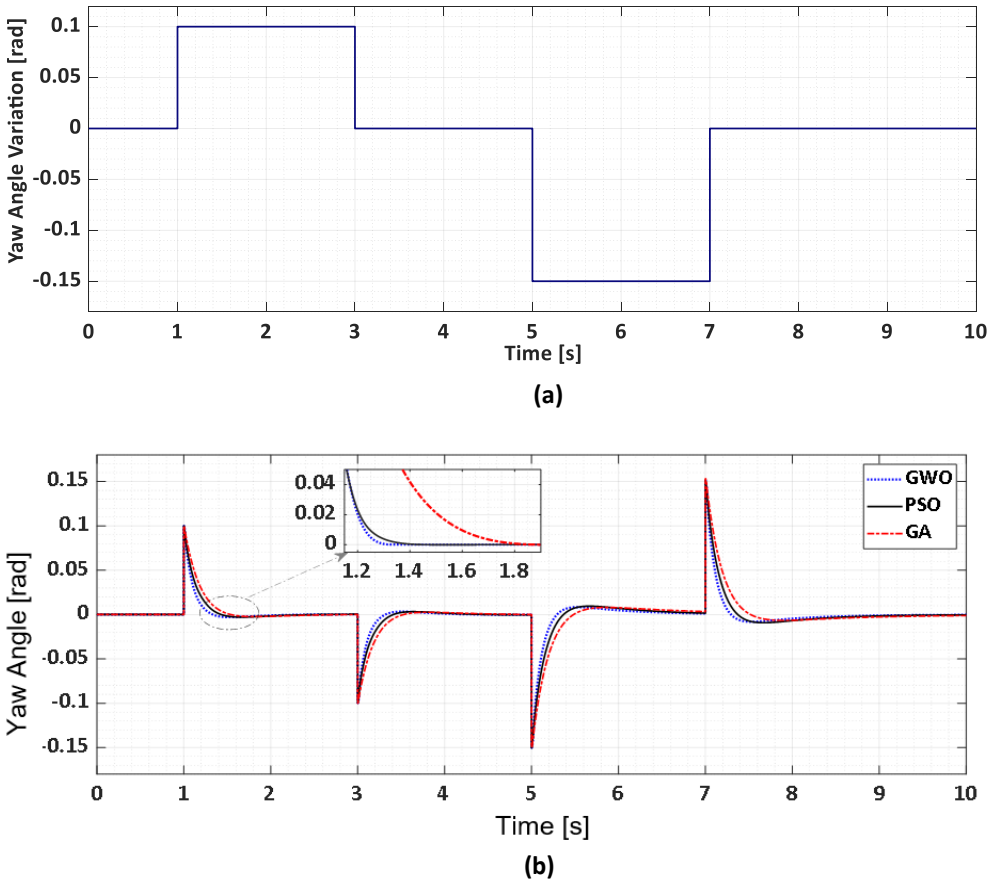

Fig. 14. Yaw angle responses for variable yaw angles: a) yaw anagle variation; b) yaw angle responses.

6. CONCLUSIONS

This research focused on formulating metaheuristic optimization algorithms such as GA, PSO, and GWO to tune the optimal combination of STSMC for balancing and PID for direction control of a self-balancing Segway. A review of related works to control the proposed system has summarized that stable and robust performance can be achieved for the control schemes of nonlinear, optimal, adaptive, and hybrid schemes where the robustness of the controllers is highly affected by the design approach, skill, and experience of the designer.

Mathematical modelling, problem formulation to tune controller parameter values, the algorithms control parameter values, the procedures of GA, PSO, and GWO algorithms for the specified application, the corresponding convergence characteristic plots, and tabulating tuned values have been done. The overall system is implemented in MATLAB/Simulink and the comparative performance evaluations of the corresponding controllers have been done using time domain performance specifications of rise and settling times as criteria for the Segway operating conditions of variable mass and tilt angles. The responses for mass variations showed that GWO-tuned STSMC has a quick response and is less sensitive to an increase in the mass of the driver compared to PSO and GA-tuned STSMC. For the GWO-tuned PID controller, there is a slight decrease in sensitivity for increasing mass. At the same time, for variable initial tilt angles of pitch and yaw, GWO-tuned STSMC and GWO-tuned PID showed superior performance compared to PSO and GA tuned controllers even though the deviations in speed of the response and sensitivity to mass variations are not significant. In summary, the proposed controllers showed the capability to track the reference pitch and yaw

angles. However, the slight variations are due to the nature of the optimization algorithms and the skill/experience of the designer in selecting control parameter values of the corresponding algorithms.

REFERENCES

[1] U. Lilhore, S. Simaiya, P. Ghosh, A. Garg, N. Trivedi, A. Anand, "Role of swarm intelligence and artificial neural network methods in intelligent traffic management," Machine Learning and Autonomous Systems, 2022, doi: 10.1007/978-981-16-7996-4_15.

[2] P. Saikia, Kartheek, P. Kumar, S. Khayyum, K. Balasubramanian, T. Ponnuvel, "Design and fabrication of Rover board," *Materials Today: Proceedings*, vol. 62, no. 4, pp. 2271-2276, 2022, doi: 10.1016/j.matpr.2022.03.612.

[3] S. Kim, S. Hwang, D. Lee, M. Myeong, "Empirical study on the development of driving environments for personal mobility vehicles," *Transportation research record*, vol. 2678, no. 3, pp. 829-845, 2024, doi: 10.1177/03611981231182962.

[4] M. Ignaccolo, G. Inturri, E. Cocuzza, N. Giuffrida, M. Pira, V. Torrisi, "Developing micromobility in urban areas: network planning criteria for e-scooters and electric micromobility devices," *Transportation Research Procedia*, vol. 60, pp. 448–455, 2022, doi: 10.1016/j.trpro.2021.12.058.

[5] C. Laverdet, P. Malola, T. Meyer, P. Delhomme, "Electric personal mobility device driver behaviors, their antecedents and consequences: A narrative review," *Journal of Safety Research*, vol. 86, pp. 274–285, 2023, doi: 10.1016/j.jsr.2023.07.006.

[6] N. Hashimoto, K. Tomita, O. Matsumoto, A. Boyali, "Effects of human factors on public use of standing-type personal mobility vehicle," *Journal of Advanced Transportation*, vol. 2020, no. 8876040, pp. 1–14, 2020, doi: 10.1155/2020/8876040.

[7] A. Olabi, T. Wilberforce, K. Obaideen, E. Sayed, N. Shehata, A. Alami, M. Abdelkareem, "Micromobility: progress, benefits, challenges, policy and regulations, energy sources and storage, and its role in achieving sustainable development goals," *International Journal of Thermofluids*, vol. 17, no. 100292, pp. 1-25, 2023, doi:10.1016/j.ijft.2023.100292.

[8] K. Nader, D. Sarsri, "Modelling and control of a two-wheel inverted pendulum using fuzzy-PID-modified state feedback," *Journal of Robotics*, vol. 2023, pp. 1–13, 2023, doi: 10.1155/2023/4178227.

[9] H. Yoo, B. Choi, "Design of simple-structured fuzzy logic systems for Segway-type mobile robot," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 15, no. 4, pp. 232–239, 2015, doi: 10.5391/IJFIS.2015.15.4.232.

[10] S. Kim, S. Kwon, "Robust transition control of underactuated two-wheeled self-balancing vehicle with semi-online dynamic trajectory planning," *Mechatronics*, vol. 68, no. 2, pp. 1-10, 2020, doi: 10.1016/j.mechatronics.2020.102366.

[11] T. Altalmas, A. Aula, S. Ahmad, M. Tokhi, R. Akmeliawati, "Integrated modeling and design for realizing a two-wheeled wheelchair for disabled," *Assistive Technology*, vol. 28, no. 3, pp. 159–174, 2016, doi: 10.1080/10400435.2016.1140688.

[12] R. Rensburg, N. Steyn, L. Trénoras, Y. Hamam, E. Monacelli, "Stability and enhancement analysis of a modelled self-balancing verticalized mobility aid using optimal control techniques," *African Journal of Science, Technology, Innovation and Development*, vol. 9, no. 1, pp. 93–109, 2017, doi: 10520/EJC-6aafb4f77.

[13] S. Haddout, "Nonlinear reduced dynamics modelling and simulation of two-wheeled self-balancing mobile robot: SEGWAY system," *Systems Science & Control Engineering*, vol. 6, no. 1, pp. 1–11, 2018, doi: 10.1080/21642583.2017.1413436.

[14] S. Jain, S. Jain, M. Makkar, "Non-linear modelling and control of self-balancing human transporter," Perspectives in Dynamical Systems II: Mathematical and Numerical Approaches, 2021, doi: 10.1007/978-3-030-77310-6_1.

[15] V. Hà, T. Thuong, N. Thanh, V. Vinh, "Research on some control algorithms to compensate for the negative effects of model uncertainty parameters, external interference, and wheeled slip for mobile robot," *Actuators,* vol. 13, no. 1, pp. 31-62, 2024, doi: 10.3390/act13010031.

[16] G. Baskota, R. Devkota, S. Paneru, S. Yadav, D. Neupane, O. Dhakal, "Analytical and experimental approach for modeling, simulation and validation of two-wheeled self-balancing robot," International Conference on Inventive Computation Technologies, 2023, doi: 10.1109/ICICT57646.2023.10133983.

[17] M. Hayajneh, "Experimental validation of integrated and robust control system for mobile robots," *International Journal of Dynamics and Control*, vol. 9, no.4, pp. 1491-1504, 2021, doi: 10.1007/s40435-020-00751-7.

[18] L. Moreno-Suarez, L. Morales-Velazquez, A. Jaen-Cuellar, R. Osornio-Rios, "Hardware-in-the-loop scheme of linear controllers tuned through genetic algorithms for BLDC motor used in electric scooter under variable operation conditions," *Machines*, vol. 11, no. 6, pp. 663-684, 2023, doi: 10.3390/machines11060663.

[19] V. Ha, V. Vinh, "Experimental research on avoidance obstacle control for mobile robots using Q-learning (QL) and deep Q-learning (DQL) algorithms in dynamic environments", In *Actuators,* vol. 13, no. 1, pp. 26-44, 2024, doi: 10.3390/act13010026.

[20] S. Ri, J. Huang, Y. Wang, M. Kim, S. An, "Terminal sliding mode control of mobile wheeled inverted pendulum system with nonlinear disturbance observer," *Mathematical Problem in Engineering*, vol. 2014, pp. 1–8, 2014, doi: 10.1155/2014/284216.

[21] H. Gao, X. Li, C. Gao, J. Wu, "Neural network supervision control strategy for inverted pendulum tracking control," *Discrete Dynamics in Nature and Society*, vol. 2021, pp. 1–14, 2021, doi: 10.1155/2021/5536573.

[22] I. Chawla, V. Chopra, A. Singla, "Robust LQR based ANFIS control of x-z inverted pendulum," Amity International Conference on Artificial Intelligence, 2019, doi: 10.1109/AICAI.2019.8701333.

[23] R. Lamba, S. Singla, S. Sondhi, "Robust stabilization of an inverted pendulum using ANFIS controllers," Lecture Notes in Electrical Engineering, 2022, doi: 10.1007/978-981-19-4300-3_9.

[24] J. Huang, M. Zhang, T. Fukuda, "Sliding mode variable structure-based chattering avoidance control for mobile wheeled inverted pendulums," Robust and Intelligent Control of a Typical Underactuated Robot, 2023, doi: 10.1007/978-981-19-7157-0_4.

[25] B. Kim, B. Park, "Robust control for the Segway with unknown control coefficient and model uncertainties," *Sensors*, vol. 16, no. 7, pp. 1000-1011, 2016, doi: 10.3390/s16071000.

[26] O. Choudhry, M. Wasim, A. Ali, M. Choudhry, J. Iqbal, "Modelling and robust controller design for an underactuated self-balancing robot with uncertain parameter estimation," *PLoS One*, vol. 18, no. 8, pp. 1-44, 2023, doi: 10.1371/journal.pone.0285495.

[27] N. Vu, H. Nguyen, "Design low-order robust controller for self-balancing two-wheel vehicle," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–22, 2021, doi: 10.1155/2021/6693807.

[28] N. Vu, H. Nguyen, "Balancing control of two-wheel bicycle problems," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–12, 2020, doi: 10.1155/2020/6724382.

[29] A. Lima-Pérez, J. Díaz-Téllez, V. Gutiérrez-Vicente, J. Estévez-Carreón, J. Pérez-Pérez, R. García-Ramirez, Chávez-Galán, "Robust control of a two-wheeled self-balancing mobile robot," International Conference on Mechatronics, Electronics and Automotive Engineering, 2021, doi: 10.1109/ICMEAE55138.2021.00038.

[30] A. Patra, S. Biswal, P. Rout, "Backstepping linear quadratic Gaussian controller design for balancing an inverted pendulum," *IETE Journal of Research*, vol. 68, no. 1, pp. 150–164, 2022, doi: 10.1080/03772063.2019.1592716.

[31] I. Jmel, H. Dimassi, S. Hadj-Said, F. M'Sahli, "Adaptive observer-based sliding mode control for a two-wheeled self-balancing robot under terrain inclination and disturbances," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–15, 2021, doi: 10.1155/2021/8853441.

[32] J. Huang, M. Zhang, T. Fukuda, "Interval type-2 fuzzy logic control of mobile wheeled inverted pendulums," Robust and Intelligent Control of a Typical Underactuated Robot, 2023, doi: 10.1007/978-981-19-7157-0_5.

[33] Y. Olmez, G. Koca, Z. Akpolat, "Clonal selection algorithm based control for two-wheeled self-balancing mobile robot," *Simulation Modelling Practice and Theory*, vol. 118, no. 102552, pp. 1-21, 2022, doi: 10.1016/j.simpat.2022.102552.

[34] X. Yang, *Engineering Optimization: An Introduction with Metaheuristic Applications*. Hoboken, NJ: Wiley-Blackwell, 2010.

[35] S. Joseph, E. Dada, A. Abidemi, D. Oyewola, B. M. Khammas, "Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems," *Heliyon*, vol. 8, no. 5, pp. 1-20, 2022, doi: 10.1016/j.heliyon.2022.e09399.

[36] K. Reddy, A. Saha, "A review of swarm-based metaheuristic optimization techniques and their application to doubly fed induction generator," *Heliyon*, vol. 8, no. 10, pp. 1-33, 2022, doi: 10.1016/j.heliyon.2022.e10956.

[37] J. Fang, "The LQR controller design of two-wheeled self-balancing robot based on the particle swarm optimization algorithm," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–6, 2014, doi: 10.1155/2014/729095.

[38] E. Karam, Al-Mustansirya, N. Mjeed, "Modified integral sliding mode controller design based neural network and optimization algorithms for two wheeled self-balancing robot," *International Journal of Modern Education and Computer Science*, vol. 10, no. 8, pp. 11–21, 2018, doi: 10.5815/ijmecs.2018.08.02.

[39] G. Prabhakar, S. Selvaperumal, P. Pugazhenthi, K. Umamaheswari, P. Elamurugan, "Online optimization based model predictive control on two wheel Segway system," *Materials Today: Proceedings*, vol. *33*, pp. 3846-3853, 2020, doi:: 10.1016/j.matpr.2020.06.227.

[40] I. Mohammed, A. Abdulla, "Balancing a Segway robot using LQR controller based on genetic and bacteria foraging optimization algorithms," *TELKOMNIKA*, vol. 18, no. 5, pp. 2642-2653, 2020, doi: 10.12928/telkomnika.v18i5.14717.

[41] Z. Jahromi, H. Shishavan, "A new approach for control of two-wheeled mobile robot," *Universal Journal of Electrical and Electronic Engineering*, vol. 6, no. 2, pp. 71–78, 2019, doi: 10.13189/ujeee.2019.060204.

[42] A. Azizi, H. Nourisola, A. Emamgholi, F. Naderisafa, "Adaptive PSO-LS-wavelet H∞ control for two-wheeled self-balancing scooter," *International Journal of Control, Automation and Systems*, vol. 15, no. 5, pp. 2126–2137, 2017, doi: 10.1007/s12555-016-0001-2.

[43] S. Nadweh, O. Khaddam, G. Hayeh, B. Atieh, H. Alhelou, "Steady state analysis of modern industrial variable speed drive systems using controllers adjusted via grey wolf algorithm & particle swarm optimization," *Heliyon*, vol. 6, no. 11, pp. 1-9, 2020, doi: 10.1016/j.heliyon.2020.e05438.

[44] T. Dogruer, "Grey wolf optimizer-based optimal controller tuning method for unstable cascade processes with time delay," *Symmetry*, vol. 15, no. 1, pp. 1-16, 2022, doi: 10.3390/sym15010054.