University of Thi -Qar Journal for Engineering Sciences
http://www.doi.org/10.31663/tqujes.12.2.452(2022)
Vol 12.2(December 2022)

ISSN :2664- 5572 (online)
ISSN:2664-5564 (print)
Available at http://jeng.utq.edu.iq
utjeng@ utq.edu.iq

Optimal Proportional Integral Derivative Controller for Two-Wheeled Self-Balanced Mobile Robots Based on Particle Swarm Algorithm

Hasanain H. Mohsin[†], Ammar A. Aldair[‡], Walid A. Al-Hussaibi[ǂ]
[†]Department of Electrical Engineering Techniques, BETC, Southern Technical University, Basrah, Iraq
[‡]Electrical Engineering Department, University of Basrah, Basrah, Iraq
[ǂ]Department of Electrical Engineering Techniques, BETC, Southern Technical University, Basrah, Iraq
Email: h.h.mohsin@fgs.stu.edu.iq

Abstract
The two-wheeled self-balanced robot (TWSBR) is an important type of mobile *robot* for a wide range of tasks, in which stability represents a major research and technical challenge. The goal of this study is to propose an optimum control method for increasing the stability of TWSBR under perturbation effects. A proportional integral derivative (PID) controller is designed based on the particle swarm optimization (PSO) method to optimize the robot controller parameters. SimScape Multibody environment is used in this paper to examine the performance of the suggested controller without the need for the mathematical model of the considered TWSBR system. This simulation platform is used to model and visualize the robot's movements while utilizing the optimized controller. The accuracy and robustness of the designed controller are tested by changing the load placed on the mechanical structure layers during robot motion. The obtained results demonstrate the effectiveness of the suggested controller design for robot balancing during disturbance situations.

Keywords: TWSBR, PID, PSO, Simscape Multibody modeling, robustness controller test**.**

1. Introduction

As is widely known, non-holonomic wheeled mobile robot stabilization has attracted significant interest because of its wide usage in a variety of applications. A great of research works has been carried out recently on the control system design for two-wheeled self-balanced robots (TWSBR) **[1], [2]**.
TWSBR balancing issue is a hot topic for researchers. Due to its instability, nonlinearity, multi-variability, and strong coupling, it can be used a typical system for testing control methods. However, this platform is better suited for testing automobile, rocket, space, robot, and military transport control methods because of its complex control **[3], [4]**. The TWSBR is considered non-linear under-actuated system having a single control input versus two degrees of freedom ($i.e.$ robot angle and wheel position) with fundamental behavior similar to an inverted pendulum system. Thus, the TWSBR system cannot remain in the upright position without a controller **[5], [6]**.
Currently, engineers and researchers are working hard to improve control unit methods in order to keep the system more stable and robust where different techniques are designed and applied on TWSBR applications. The traditional proportional integral derivative (PID) controller is widely used in many fields of the industry with only three gains to tune **[7].** Linear Quadratic Regulator (LQR) is employed in **[8],** PID and LQR are integrated to improve the balancing in [9], optimized LQR controller design using the particle swarm optimization (PSO) algorithm **[10],** Neural Network control (NN) **[11]** and fuzzy logic controlling **[12],** and PSO based fuzzy and PID controller **[13], [14].**
For an effective controller, an accurate model is needed. The dynamic equations must be manually derived and arranged into a block diagram form. So, it is hard to see how the robot system physical components are connected. The TWSBR model can be formulated mathematically using Euler-Lagrange **[15], [16]** or Simscape in Matlab **[17].**
This study uses the Simscape Multibody library to model and displays the movements of a TWSBR that used a proposed controller. For the balancing robot control system, the PID controller has been designed. To determine the PID controller's optimum variables, the particle swarm optimization (PSO) method is implemented. Moreover, The goal of using an intelligent algorithm (PSO) to tune PID controller gains in order to solve the problem of traditional trial and error methods of parameter selection. The efficacy and robustness of the developed controller are investigated by varying the carried weight of the TWSBR. The acquired simulation results demonstrate the validity of the suggested controller for TWSBR and overall performance has been enhanced.
The remaining parts of this work are organized as detailed follows. The basics of Simscape modeling are presented in Section 2. Section 3 discusses how the control design is implemented using a PSO-based PID controller. The discussion and the results of the simulation are presented in Section 4. The conclusion of the paper is described in the final section of the paper.

2. TWSBR Simscape Multibody Modeling

The TWSBR system simulation with a controller is very crucial in order to conduct studies on the control ability of the robot when it is being moved through the impacts of road disturbances. The researchers used several simulation programs to solve this problem. In the majority of these studies, the authors used two programs to simulate the controlled system and visualize robot mobility in different environments **[19], [18].**

In this study, the Simscape Multibody environment is utilized to enhance the controller system for the TWSBR and to demonstrate the movements of the robot controller under a variety of loading disturbances. A Simscape Multibody tools has the ability to solve motion mathematical equations and evaluate the performance of control systems. The tools provides a broad range of libraries and Simulink blocks that can be used together to build any form of robot structure such as robotic manipulators, mobile robots, and walking robotic systems. **[20].** Additionally, it offers simulation and control techniques that are designed to assist researchers in achieving the appropriate simulation results and accurately visualizing of the robot system movements **[21]**.

Fig. 1 illustrates the TWSBR Simscape Multibody modeled structure while Fig. 2 demonstrates the visual representation of TWSBR in the Mechanics Explorer platform. The robot can move in a three-dimensional area in this workplace environment. Table 1 describes the functions of the Simulink blocks used in Fig 1.

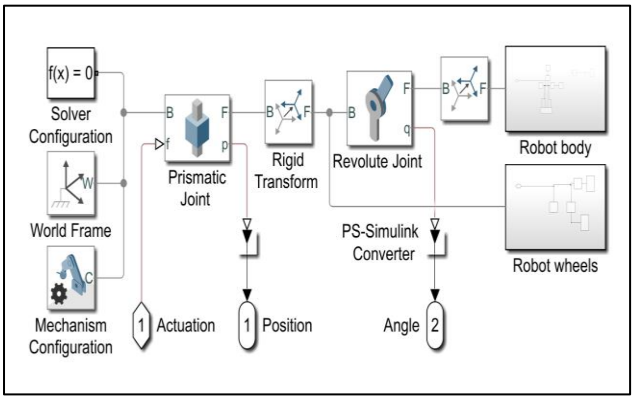


Fig.1 Simscape model for the TWSBR system

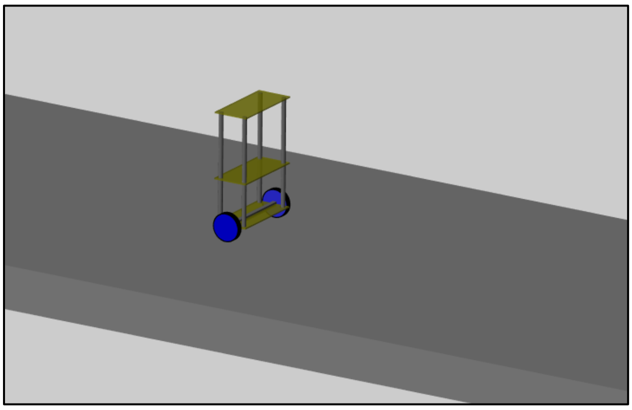
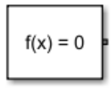
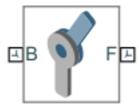
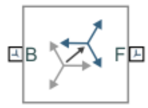
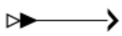


Fig.2 The 3D structure visualization of TWSBR

Table 1 describes the Simscape blocks shown within Fig 1

Blocks	Name	Descriptions
	Solver Configuration	Describes the simulation setup settings.
	World Frame	Making a frame of reference for the mechanical model in the physical design.
	Mechanism Configuration	Initialization of the simulation and mechanical parameters
	Revolute Joint	The motion at angles between the robot body and the wheels is translated by this joint
	Prismatic Joint	It translates the displacement-based motion between the entire structure and the surface.
	Rigid Transform	produce a three-dimensional rigid transition between two different frames
	Solid Block	Solid characteristics are found in these blocks.
	PS-Simulink Converter	signal connector from Simulink to physical signals

3. Controller Design of TWSBR

In recent decades, various control systems have been used to stabilize the TWSBR system and the swing up angle. The robot wheels are rotated forward and reverse movements to keep the robot in the vertical position required for the stabilization. There are various methods for implementing a control strategy on a TWSBR system. Some approaches consider the system as a single input single output (SISO) system **[22]**. Thus, the approach assumes that there is only one controller for one output. Several control schemes are presented to the self-balancing robot in order to fulfill control objectives. The present state that needs to be controlled is represented by the robot tilt angle. These are described further below.

3.1 PID controller

In a wide range of industrial and manufacturing settings, the PID controller is a common part. The proportional gain(K_p), integral gain (K_i) and derivative gain (K_d) are the three parameters that compose the design of the PID controller. Both of these parameters can be tuned to achieve the best value based on the requirements of a particular system [23]. The fundamental equation that controls the behavior of this controller is presented in the following illustration.

$$U(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de}{dt} \quad (1)$$

where K_p is the proportional value, K_i is the integral value, K_d is the derivative value and $e(t)$ is the error function. Moreover, $e(t)$ is the error signal that occurs between two signals such as the reference point value and the actual value. The actual value is the measured output response of the system while the reference value is a signal that is sent to the modeling system depending on what it needs as its input. The PID controller parameters adjusted are important in influencing the performance and time response of the system and they can have a significant impact on the functioning characteristics, whether favorably or unfavorably. The tuning of PID parameters to their optimum levels is required in order to obtain the best possible response. Furthermore, the DC motor is one of the most popular and commonly used actuators in control systems. The correction force required for a TWSBR stability issue. It is proportional to the torque produced by DC motors. It generates not only translational movement for the robot but also the rotational movement that is coupled to the wheels [24], [25]. The specifications of an actual DC motor obtained from [26] have been specified for this simulation. The PID controller design with the TWSBR is shown in Fig. 3.

3.2 Particle Swarm Optimization based on PID controller

In comparison to other optimization techniques, PSO is one of the most powerful approaches and also has a high level of effectiveness [27]. In this work, the PSO algorithm is used to identify the optimal PID controller parameters. PSO is a concept that requires the velocity of each particle to be altered at each time step in order to move it closer to both its global best and its local best positions. Moreover, the next velocity can be computed using existing velocity, present position, optimum past location and current position distance [28]. The fundamental motion equations are used to guide the operation of the particles:

$$v_i^{k+1} = w \times v_i^k + c_1 \times rand \times (x_i^b - x_i^k) + c_2 \times rand \times (x_i^g - x_i^k) \quad (2)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (3)$$

Which v_i^k is the speed of the particle, x_i^k is the position of present particle, w is the weight vector, x_i^b is the best value, x_i^g is the global best value, and rand is a randomized variable between 0 and 1. Also, the $c1$ and $c2$ are learning factors. PSO algorithm takes minimal lines of code. It's a basic, easy-to-implement and efficient algorithm [29]

The following equation is the formula of the performance index function for limited time duration:

$$F = ISE\ e^2(t)dt \quad (4)$$

In this research, we need to determine the PID controller parameters for stabilizing TWSBR using the PSO method. Three PSO variables are considered for K_i, K_p and K_d. Each of the three variables varied in terms of the number of the first population, first position and speed. The PSO algorithm specified the condition of the three variables to apply it for the physical simulation models. Fig. 4 depicts a block schematic of a PSO based PID controller for the TWSBR system.

The convergence of the approach is shown in Fig. 5 which provides the analysis of the error minimization that occurs during the iterations to reach the ISE final value. The input data for the PSO algorithm is shown in Table 2 which shows the number of iterations, populations, learning coefficients and variables details. Through the PSO online tuning process, the optimum PID controller parameters have been found as shown in Table 3.

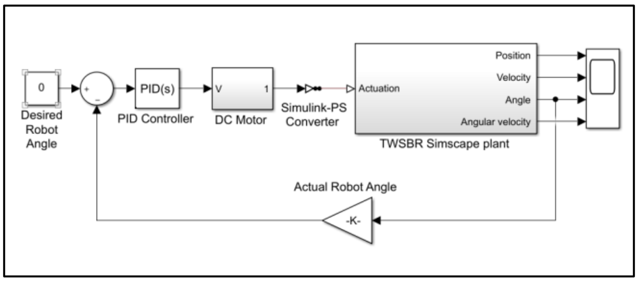


Fig.3 PID controller for TWSBR Simscape Model

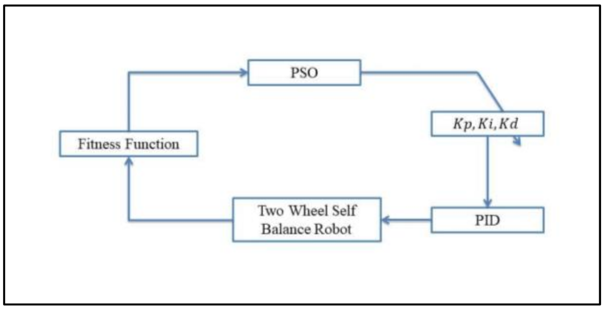


Fig.4 The block design of a PSO-based PID on a TWSBR.

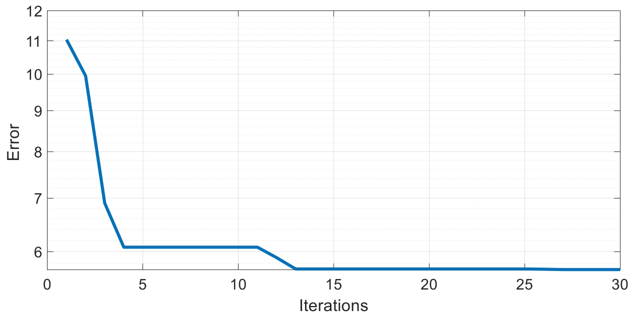


Fig.5 PSO Error minimization convergence curve

Table 2 PSO algorithms parameters

initialization Parameters	PSO
Number of iterations	30
Size of populations	30
Inertia weight	1
Personal Learning Factor	1.5
Global Learning Factor	1.5
Number of variables	3
Lower Bound of Variables	0
Upper Bound of Variables	25

Table 3 PID controller parameters using a PSO tuning

PID Parameter	Value
Kp	25
Ki	12.501
Kd	0.556

4. Simulation and Results

The results of the simulation are performed to evaluate the effectiveness of the constructed controller and confirm its capacity to maintain a robot's stability under the influence of movement road disturbances. There are two different case studies that are taken into consideration: the first PSO based on PID Controller design. Second, the robustness effectiveness responsiveness of the controller is examined by loading different masses onto the robot body structure. The simulated scenarios investigated are as follows:

4.1 PSO Algorithm Based on PID Tuning

It was found that a more efficient controller is required to stabilize the movement of the TWSBR without falling down. In response to such a situation, we are developing a PID controller. To obtain the best outcomes, the PID controller's settings are fine-tuned using a PSO approach. The control efforts for the suggested PID controller are provided to demonstrate its effectiveness.

However, the simulation result demonstrates the controller capability to reject the disturbance signal at the period shown in Fig. 6 when the initial external disturbance signal of 10 degrees is applied at the robot structure.

Further, the control efforts can reject several disturbance signals in different locations of the same movement simulation. This verified the robustness of the controller in relation to the behavior performance response system.

Fig. 7 depicts the results of the TWSBR simulation when subjected to the road disturbances influence of 10 °, -10 ° and 10 ° degrees on the straight direction.

Therefore, the controller is capable of effectively balancing the robot in the upright position and recovers its state in around 0.2 seconds.

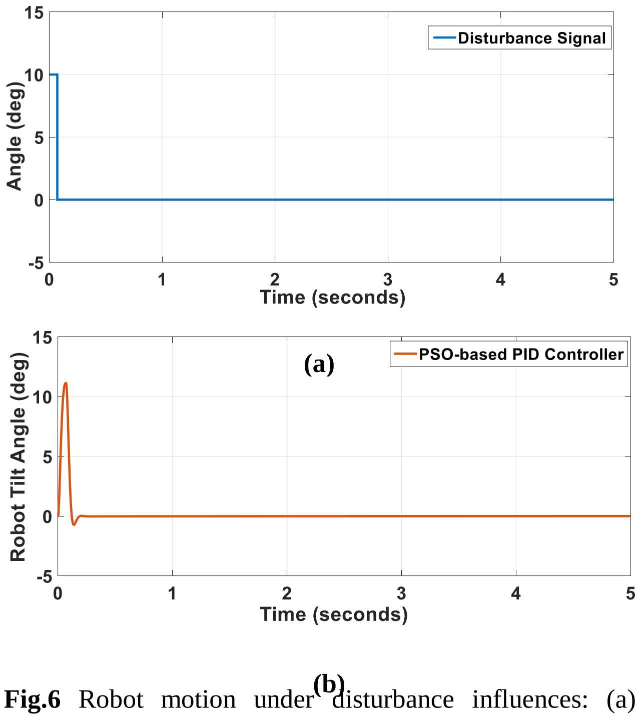


Fig.6 Robot motion under disturbance influences: (a) initial disturbance signal applied to robot; (b) TWSBR angle response

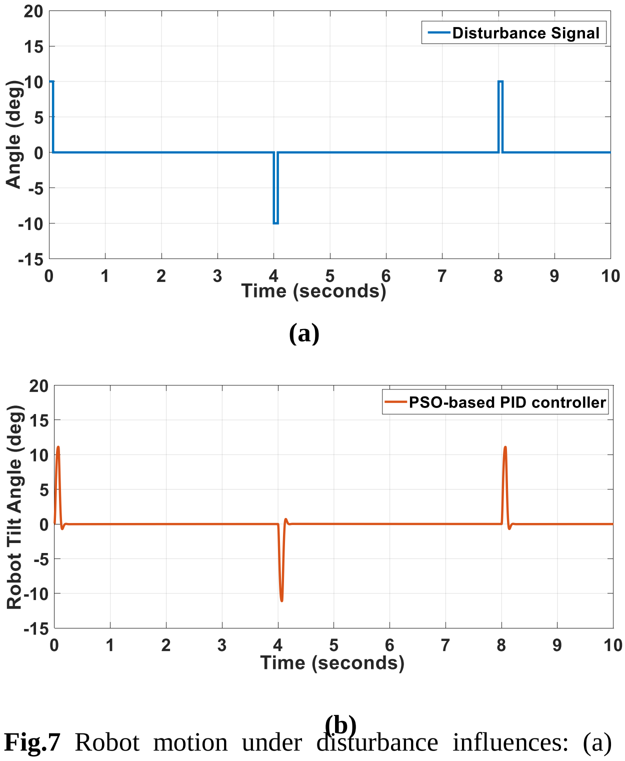


Fig.7 Robot motion under disturbance influences: (a) three disturbance signals applied to robot; (b)

4.2 Robustness Test of PSO Based on PID Controller

Loading additional weight on the TWSBR's body is a way of testing the efficiency and robustness of the proposed controllers. Two perturbation conditions are applied to the controlled system to evaluate the resilience and functioning of the developed controller. In the first case, one kilogram of additional weight is subjected to the robot's body as shown in Fig. 8. In the second case, two kilogram is loaded into the TWSBR structure to evaluate the capability of the developed controller as illustrated in Fig. 9. Therefore, Fig. 10 exhibits the PID control attempts of the TWSBR movement under the effect of a disturbance when 1 kg is added to the robot body. The TWSBR motion PID control effort under the influence of a disturbance when 2 kg is added is shown in Fig. 11. Furthermore, the transient responses comparison of the simulation results (self-balance time and overshot) for each condition are shown in Table 4, including the suggested extra weights loaded to robot layer.

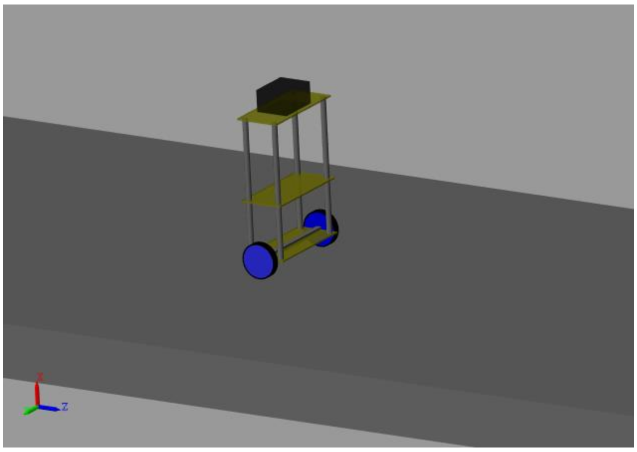


Fig.8 TWSBR Visualized in Mechanics Explorer with 1 Kg.

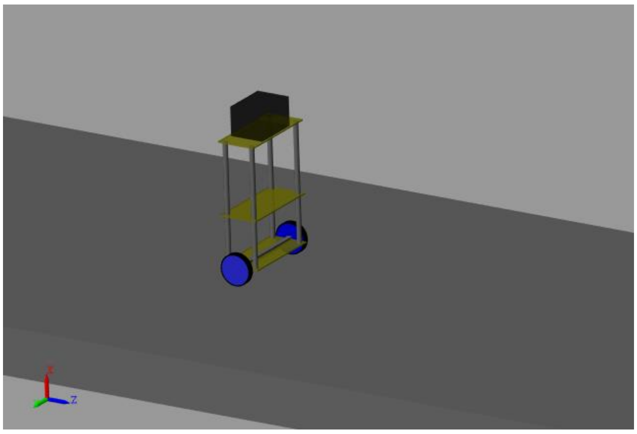


Fig.9 TWSBR Visualized in Mechanics Explorer with 2 Kg.

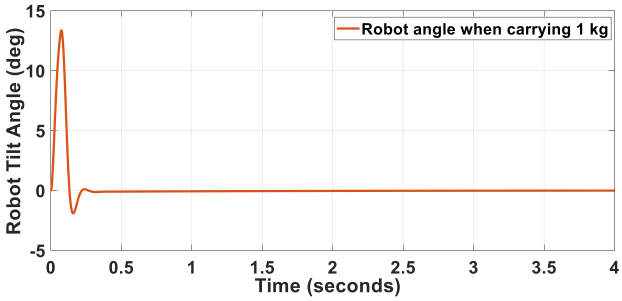


Fig.10 Robot inclination responses with 1Kg of extra weight

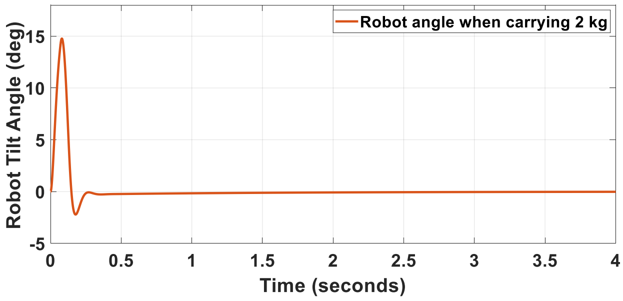


Fig.11 Robot inclination responses with 2Kg of extra weight

Table 4 The main values of simulation results comparison.

Load (kg)	Self-balance time (sec)	Overshoot (deg)
Without load	0.2	11.5
1Kg	0.4	13.6
2Kg	0.8	15

Conclusion

The PID controller is designed in this work to stabilize the TWSBR and the optimum values of the PID controller are optimized using the PSO algorithm. The modeling and simulation in 3D are displayed by analyzing the TWSBR model in a Simscape Multibody library environment without depending on mathematical equation analysis to solve the equilibrium problem. The simulation findings indicate that the proposed controllers efficiently maintained the robot's stability during the motion disturbance effect. The proposed controller robustness has been verified by putting additional weights on the robot body. The two loads are added as 1 kg and 2 kg, respectively. In the both situations, the controlled system maintained stability and provided the desired motion while rejecting disturbances. Therefore, the whole suggested system demonstrates both its application in real-life settings and its efficiency in controlling the TWSBR.

References

[1] A. Salerno and J. Angeles, “A new family of two-wheeled mobile robots: Modeling and controllability,” IEEE Trans. Robot.,. 23, 1,. 169,Feb. 2007.

[2] M. S. Mahmoud and M. T. Nasir, “Robust control design of wheeled inverted pendulum assistant robot,” IEEE/CAA J. Autom. Sin., 4,. 4,. 628, Sep.2017.

[3] M. O. Asali, F. Hadary,and B. W. Sanjaya “Modeling, simulation, and optimal control for two-wheeled self-balancing robot,” International Journal of Electrical and Computer Engineering,. 7,. 4,. 2008, Aug.2017.

[4] O. Jamil, M. Jamil, Y. Ayaz, and K. Ahmad, “Modeling, control of a two-wheeled self-balancing robot,” in Proc.2014 Int. Conf. Robot. Emerg. Allied Technol. Eng. iCREATE 2014, Islamabad, Pakistan, Apr. 2014. 191.

[5] A. Kharola, P. Patil, S. Raiwani, and D. Rajput, “A comparison study for control and stabilisation of inverted pendulum on inclined surface (IPIS) using PID and fuzzy controllers,” Perspect. Sci.,. 8,. 187, Sep.2016.

[6] P. Frankovský et al., “Modeling of two-wheeled self-balancing robot driven by DC gearmotors,” Int. J. Appl. Mech. Eng., 22, 3,. 739,Aug. 2017.

[7] A. T. Azar et al., “ Self-balancing Robot Modeling and Control Using Two Degree of Freedom PID Controller, “in proc. international conference on advanced intelligent systems and informatics AISI 2018 , Cham,Aug.2018, 845,. 64.

[8] F. R. L. Jiménez, I. A. R. Ruge, and A. F. L. Jiménez, “Modeling and control of a two wheeled auto-balancing robot: A didactic platform for control engineering education,” In Proc. LACCEI Int. Multi-conference Eng. Educ. Technol., Jul.2020,. 1.

[9] L. Sun and J. Gan, “Researching of two-wheeled self-balancing robot base on LQR combined with PID,” In Proc. - 2010 2nd Int. Work. Intell. Syst. Appl. ISA 2010, Wuhan, China ,May 2010 ,. 1.

[10] E. karam and N. Mjeed, “Modified Integral Sliding Mode Controller Design based Neural Network and Optimization Algorithms for Two Wheeled Self Balancing Robot,” Int. J. Mod. Educ. Comput. Sci.,. 10, 8, pp. 11, Aug.2018.

[11] S. Jung and S. S. Kim, “Control experiment of a wheel-driven mobile inverted pendulum using neural network,” IEEE Trans. Control Syst. Technol.,. 16,. 2, 297,Feb. 2008.

[12] T. A. Mai, D. N. Anisimov, T. S. Dang, and V. N. Dinh, “Development of a microcontroller-based adaptive fuzzy controller for a two-wheeled self-balancing robot,” Microsyst. Technol.,. 24,. 9, 3677,Sep. 2018.

[13] G. R. Yu, Y. K. Leu, and H. T. Huang, “PSO-based fuzzy control of a self-balancing two-wheeled robot,” in Proc.IFSA-SCIS 2017 - Jt. 17th World Congr. Int. Fuzzy Syst. Assoc. 9th Int. Conf. Soft Comput. Intell. Syst., Otsu, Japan, Jun.2017,. 1.

[14] E. Mendez-Flores et al., “Self-Balancing Robot Control Optimization Using PSO,” in Proc. 2020 5th Int. Conf. Control Robot. Eng. ICCRE 2020, Osaka, Japan, Apr. 2020,. 1,. 7.

[15] F. Grasser, A. D’Arrigo, S. Colombi, and A. C. Rufer, “JOE: A mobile, inverted pendulum,” IEEE Trans. Ind. Electron., 49, 1,. 107, Feb.2002.

[16] A. Ghaffari, A. Shariati, and A. H. Shamekhi, “A modified dynamical formulation for two-wheeled self-balancing robots,” Nonlinear Dyn.,. 83, 1,2,. 217, Jan.2016.

[17] S. Mohapatra, R. Srivastava, and R. Khera, “Implementation of a two wheel self-balanced robot using MATLAB simscape multibody,” in Proc. 2019 2nd Int. Conf. Adv. Comput. Commun. Paradig. ICACCP 2019, Gangtok, India, Feb.2019, 1.

[18] S. Kim and S. J. Kwon, “Dynamic modeling of a two-wheeled inverted pendulum balancing mobile robot,” Int. J. Control. Autom. Syst.,. 13,. 4, 926, Aug.2015.

[19] A. A. Aldair, A. Al-Mayyahi, and B. H. Jasim, “Control of Eight-Leg Walking Robot Using Fuzzy Technique Based on SimScape Multibody Toolbox,” in proc. IOP Conf. Ser. Mater. Sci. Eng., Feb.2020, 745,. 1,.012015.

[20] P. Noskievic and D. Walica, “Design and Realisation of the Simulation Model of the Stewart Platform using the MATLAB-Simulink and the Simscape Multibody Library,” in Proc. 2020 21st Int. Carpathian Control Conf. ICCC 2020, High Tatras, Slovakia, Oct. 2020,. 0.

[21] A. Kathpal and A. Singla, “SimMechanics™ based modeling, simulation and real-time control of Rotary Inverted Pendulum,” in Proc. 2017 11th Int. Conf. Intell. Syst. Control. ISCO 2017, Coimbatore, India, Jan. 2017, 166.

[22] M. E. Mousa, M. A. Ebrahim, and M. A. M. Hassan, “Stabilizing and Swinging-Up the Inverted Pendulum Using PI and PID Controllers Based on Reduced Linear Quadratic Regulator Tuned by PSO,” Int. J. Syst. Dyn. Appl.. 4,. 4,. 52, Oct.2015.

[23] A. A. Aldair, A. Al-Mayyahi, and W. Wang, “Design of a Stable an Intelligent Controller for a Quadruped Robot,” J. Electr. Eng. Technol.,. 15,. 2, 817, Mar.2020.

[24] W. An and Y. Lir,“Simulation and Control of a Two-wheeled Self-balancing Robot,” in Proc. 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), Shenzhen, China, Dec. 2013, 456.

[25] A. T. Ali et al., “Design and Implementation of Two-Wheeled Self-Balancing Robot Using PID Controller,” in Proc. 2020 Int. Conf. Comput. Control. Electr. Electron. Eng. ICCCEEE 2020,

Khartoum, Sudan ,Feb.2021,. 1.

[26] C. Iwendi et al., “Robust Navigational Control of a Two-Wheeled Self-Balancing Robot in a Sensed Environment,” IEEE Access, 7, 82337, Jun. 2019.

[27] S. A. Kahdum ,S. A., Abdul-Hassan, K. M., and Ali, R. S. “Modeling of Structural Human Dynamic Simulation of Speed Control for Synchronous Reluctance Motor Based on Tuning Cascaded PID Controller with PSO Algorithm,” University of Thi-Qar Journal for Engineering Sciences,. 7,. 2,.1, 2016.

[28] S. Howimanporn, S. Thanok, S. Chookaew, and W. Sootkaneung, “Design and implementation of PSO based LQR control for inverted pendulum through PLC,” in Proc.SII 2016 - 2016 IEEE/SICE Int. Symp. Syst. Integr., Sapporo, Japan, Dec. 2016,. 664.

[29] Ü. Önen, Çakan, A. and İlhan, İ. “Particle Swarm Optimization Based LQR Control of an Inverted Pendulum,” Eng. Technol. J., 2,. 5, May 2017.