OPTICAL SENSOR-BASED BALL TRACKING AND CENTRALIZATION USING A PID-CONTROLLED STEWART PLATFORM

Jival, A.[1*] and Nwaobasi, C.C.[2]

[1]Department of Electrical and Electronics Engineering, Modibbo Adama University, Yola

[2]Federal Polytechnic Mubi, Adamawa State

[*]Correspondent email: jivalangeti@gmail.com

ABSTRACT

This research work entails optimizing a model of a Stewart Platform system. The proposed system is designed to autonomously centralize a rolling ball on its top plate using a multiplexed array of optical sensors and a PID control algorithm. The Stewart Platform, a parallel manipulator, comprises a fixed base and a movable top plate connected by six servo-actuated extensible legs. By varying the lengths of these legs, the platform achieves roll (rotation about the x-axis) and pitch (rotation about the y-axis) motions to influence the movement of the ball. Conventional ball-balancing systems rely on resistive touch surfaces, inertial sensors (gyroscopes and accelerometers), or vision-based setups involving external computers. The problem that is inherent with these existing designs is that they require enormous computational power for floating point calculations that come with image processing and reading from resistive panels. Our proposed approach deviates from these existing designs by introducing a simple sensing technique. This technique uses a two-dimensional grid of optical sensors beneath a transparent top plate. As the ball moves, it obstructs light received by specific sensors, allowing the controller to determine its position through binary (digital) readings—eliminating the need for computationally expensive. This significantly reduces processing load and allows the system to operate on low-cost microcontrollers. A PID controller, tuned via the Ziegler-Nichols method, regulates the platform's motion to minimize positional error. To achieve this work the complete system is modeled and simulated using MATLAB Simulink's Multibody environment. Simulation results show that the platform effectively stabilizes the ball for disturbances like base tilts up to 12 degrees. Furthermore, performance comparison with previous work indicates a 5-second improvement in settling time, highlighting the efficiency and responsiveness of the proposed design.

Keywords: Stewart Platform, Agrochemical, optical sensors, Ball, PID control algorithm, pitch

1 INTRODUCTION

The Stewart Platform (SP), initially conceived by Eric Gough in 1954 and later refined for motion simulation applications, is a class of parallel manipulators designed for precise positioning and control [1]. As a parallel mechanism, the SP comprises a fixed base and a movable top platform connected via six adjustable-length legs. These legs form closed kinematic chains and can be actuated either linearly through prismatic joints or rotationally through servo or stepper motors [2]. Each leg's length is varied to achieve specific spatial orientations and positions of the top platform. In systems employing prismatic actuation**,** each leg consists of a static segment and a sliding component, producing linear motion when voltage is applied to the actuator—typically a hydraulic or electric linear actuator [3, 4]. Conversely, rotary actuation uses electric motors where leg length variation is achieved by rotating joints attached to motor horns or shafts. These methods allow the platform to move with six degrees of freedom (DoF): three translational (along x, y, and z axes) and three rotational (yaw, pitch, and roll). This research focuses on the use of the Stewart Platform for dynamic ball balancing on the top movable platform. Specifically, the objective is to maintain the ball at the platform's center in response to disturbances such as tilting of the base. Among the six DoF, only pitch and roll motions are utilized, as they are sufficient to control ball motion along the x and y directions. To stabilize the ball, the platform must continuously adjust its orientation to counter the ball's deviation from the center. This requires real-time tracking of the ball's position and velocity in both x and y directions. Conventional methods for achieving this include the use of resistive panels **or** vision-based systems. The resistive panel method requires floating-point calculations to estimate coordinates, while the camera-based approach demands significant image processing power, typically necessitating dedicated computing hardware such as single-board computers or PCs. While effective, these methods present two major drawbacks: computational overhead and hardware cost**.** They rely heavily on floating-point arithmetic or intensive image processing, which is not ideal for low-power embedded systems. Controllers with limited processing capacity may be too slow to perform these computations in real-time, resulting in instability or failure to maintain the ball at the center. Floating-point operations, particularly those involving 64-bit double-precision variables, can be computationally expensive and memory-intensive [5, 6]. Although modern personal computers can handle such tasks efficiently, they are not cost-effective for embedded control applications. In resource-constrained environments, reliance on such computation can hinder system responsiveness, especially when multiple real-time calculations are involved. To address these limitations, this study introduces an alternative approach for real-time ball position detection that minimizes computational load. The proposed method employs a 9×9 array of optical photodiodes placed beneath a transparent top platform**.** These photodiodes serve as light-sensitive switches that change state when the ball occludes incident light. By scanning the array in software using a multiplexing technique, the system can determine the ball's location using digital logic rather than floating-point arithmetic. The entire Stewart Platform, including rotary actuation, optical sensor array, and feedback control system, is modeled and simulated using MATLAB Simulink and Simscape Multibody. The platform's inverse kinematics [7] is employed to determine actuator configurations for desired pitch and roll. **A** Proportional–Integral–Derivative (PID) controller is developed to adjust platform orientation based on sensor feedback, effectively stabilizing the ball at the center. This work offers a low-cost, computation-efficient alternative to existing ball-balancing solutions by eliminating the need for complex image processing or high-precision floating-point calculations. By exploiting simple digital signals from the optical sensor array, the platform can react more swiftly to disturbances using low-power microcontrollers. This contribution provides a viable approach for embedded real-time control systems where computational resources are limited.

2 METHODOLOGY

The method employed in order to achieve this research work is summarized in the block of diagram of Figure 1. Figure 1 illustrates the Simulink block diagram of the PID-based ball-balancing Stewart Platform system. The model is primarily divided into two core components: the plant block, representing the Stewart Platform dynamics, and the controller block, implementing the PID control strategy. Within the plant block, multiple subcomponents work together to simulate the three-dimensional dynamic behavior of the Stewart Platform, including its mechanical structure and motion response. The controller block comprises standard Simulink PID controller blocks, which generate corrective control signals based on real-time feedback from the plant [8]. The control system operates using position and velocity feedback in both the x and y directions. These feedback signals are used by the controller to continuously adjust the platform's pitch and roll angles to stabilize the ball at the desired central position. Additional blocks in the model include: Desired x-position and y-position blocks, representing the reference position (typically the center); Summation blocks, depicted with circular symbols, used to compute error signals by comparing desired and actual positions; Scope blocks, used for visualizing the system's output and response in real time; A disturbance block**,** which introduces external perturbations to test the platform's response and controller effectiveness.

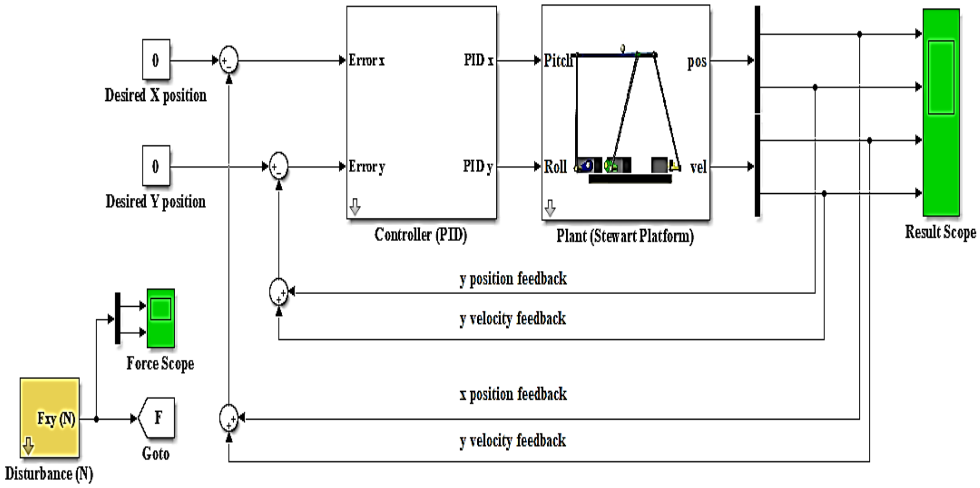

Figure 1. Simulink model of the PID based ball balancing Stewart platform

2.1 Plant model (Stewart Platform)

The plant block, as modelled in this research work, is composed of sub-blocks or sub-models depicted in Figure 2. Several sub-blocks are seen, some of which are further composed of sub-blocks.

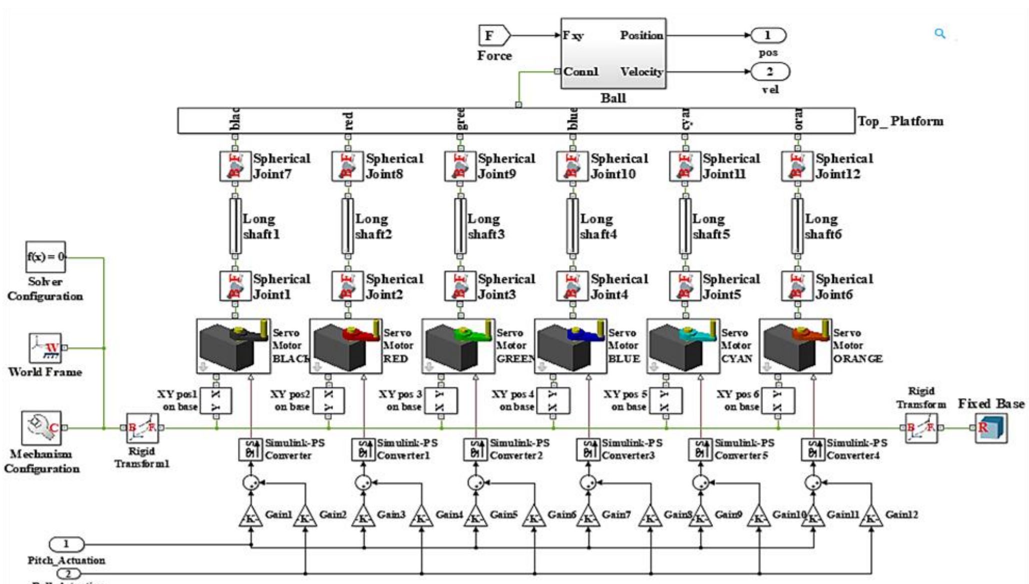

Figure 2. Plant model

The system comprises of several interconnected Simulink and Simscape Multibody blocks that collectively represent the physical and control components of the platform. The input/output blocks serve as the interface between the controller and the plant model. The input blocks, specifically the Pitch Actuation and Roll Actuation blocks, receive control signals from the PID controller. These signals guide the motion of the top platform along the *x* and *y* axes, respectively. Conversely, the output blocks transmit the position and velocity data of the ball—specifically in the *x* and *y* directions—to a scope for real-time monitoring and feedback. These outputs are also Routed back into the PID controller, enabling closed-loop feedback to correct any deviation in the ball's trajectory using the PID algorithm. To accurately model the spatial arrangement of system components, Rigid Transform blocks are utilized [9]. These blocks define the 3D positions and orientations of bodies within the simulation environment. They are also instrumental in assembling multiple solid components into unified structures. For example, the servo motor, composed of a base, shaft, and horn, is assembled using Rigid Transform blocks to position each part appropriately within the servo assembly. The Stewart Platform's dynamics are governed by joint blocks in Simscape Multibody. Two types of joints are employed: revolute joints**,** which allow one degree of rotational freedom, and spherical joints**,** which provide up to six degrees of freedom. Revolute joints are placed at the interface between the servo motor shaft and horn, enabling motor actuation. Spherical joints are used where the SP legs attach to the top movable platform, mimicking real-world Stewart Platform behavior [10]. The complete model consists of twelve spherical and six revolute joints, giving the platform both flexibility and realism. The servo motors used in this model act as rotary actuators and are constructed in Simscape Multibody. Each motor consists of a base modeled as a brick, a shaft modeled as a cylinder, and a horn designed programmatically due to its non-standard shape. These parts are combined and spatially arranged using Rigid Transform blocks and connected via revolute joints to allow rotational motion.

3 DESIGN

3.1 Designing the multiplexed 9x9 array of optical sensor

A 9x9 array of photodiodes is designed using the foundations library block in Simulink. Photodiodes behave like a one directional switch [9]. As the ball rolls on any of the photodiode sensor, it blocks light from falling on that sensor and so the sensor will not conduct, thereby registering a digital low on the interface to the column decoder. Since the 9x9 array of optical sensors will be addressed in software, the system will be able to map the interface that has gone low to the address of the sensor in the 9x9 grid and hence know the exact position of the ball when it blocks any of the sensors. Without the technique of multiplexing, it will require 81 interfaces or connectors to interface with the system. The technique of multiplexing is used to reduce the number of interfaces to the system, in this case, to 18 interfaces (sum of rows and columns) [11]. In multiplexing, a digital high signal (5V) is sent, one at a time, to all nine rows (a process called row scanning) while each column is read, also, one at a time (column decoding) for every row that receives a digital high signal [12]. If, for instance, the second row is scanned and the column decoder reads a digital low signal at the fourth column, then it means that the ball is at the (2, 4) position of the sensor array. This is how the system is able to determine the position of the ball. Figure 3 shows the first step in powering the photodiode and preparing it for row scanning and column decoding.

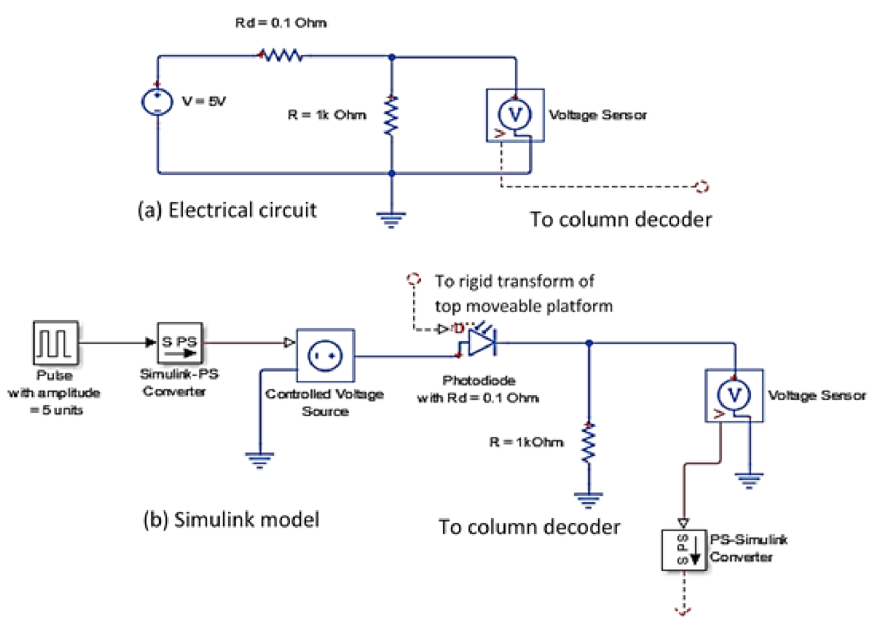

Figure 3. Photodiode circuit

Photodiodes usually have internal resistance (dark current resistance R_d) of about 0.1Ω when absorbing light and about 100, 000 Ω when not absorbing light [13]. If it is intended to power the circuit with a 5V DC supply and also have the column decoder to decode 4.9995V (approximately 5V because of the voltage drop caused by the 0.1 Ω diode) as a logic high when the diode is absorbing light, then applying ohms law equations 1 and 2.

$$V = IR = I(R_d + R) \quad (1)$$

$$I = \frac{V}{R_d + R} \quad (2)$$

Where: I is the current flowing through the circuit, V is the supply voltage = 5V, R_d is the dark resistance of the photodiode when absorbing light = 0.1 Ω, R is the resistance to be calculated such that the voltage that will be decoded will be $\approx 4.9995V$, The voltage across the voltage sensor block is calculated as current through the circuit multiplied by the resistance. This implies that: Therefore, a resistance of R = 1 kΩ is selected. When the diode is not absorbing light (that is the light is blocked by the ball), the dark resistance, R_d becomes 100, 000 Ω. Applying equation (3.11) reveals the voltage that will be decoded by the column decoder. The voltage is approximately 0V. Therefore, the column decoder reads a voltage of approximately 5V (4.9995) when the diode is conducting (absorbing light) and a voltage of approximately 0V (0.0495) when the diode is not conducting. The PID controller is designed by invoking the PID block of the Simulink software and connecting the block as shown in Figure 4.

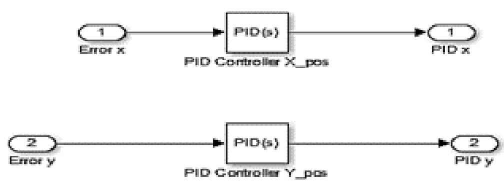

Figure 4. PID design

Double clicking both blocks opens up a dialog box where the P, I, and D parameters are set. The equation implemented for the PID algorithm is as thus:

$$PID = P + I\frac{1}{s} + D\frac{N}{1 + N\frac{1}{s}} \quad (3)$$

The Ziegler-Nichols method is adopted for tuning the PID parameters. This method involves setting all parameters to zero and increasing the value of the P gain until the system exhibits sustained oscillations. If the frequency of this oscillation is f_0 and the amplitude (critical gain) is k_c then the PID parameters are tuned based on the values shown in Table 1.

Table 1. Ziegler-Nichols Method of PID Tuning

P	I	D
$0.6k_c$	$2.0f_0$	$0.125/f_0$

For the x-axis, the tuned parameters using Ziegler-Nichols methods is:
P = 4.5, I = 5 x 10^{-7}, D = 2.8 and N = 1.35 x 10^2 and the y-axis
P = 5.2, I = 5 x 10^{-7}, D = 3.6 and N = 1.35 x 10^2

4 SIMULATION OF RESULTS

4.1 Open loop control system result

Open loop control essentially means running the system without any form of feedback from the output of the plant. Without feedback, the PID algorithm will not be able to provide any form of control that will balance and centralize the ball. Table 2 describes system parameters or simulation settings used in open loop control system scenario and figure 5 presents the result.

Table 2. Simulation Settings for Open Loop Control System

S/N	Simulation parameter / Simulation settings	Value / Action
1	X axis ball position and velocity feedback	Removed
2	Y axis ball position and velocity feedback	Removed
3	Rotate platform base about x axis (roll)	4 degrees
4	Rotate platform base about y axis (pitch)	4 degrees

Figure 5 is the simulation result obtained in an open loop system using parameters described in Table 2.

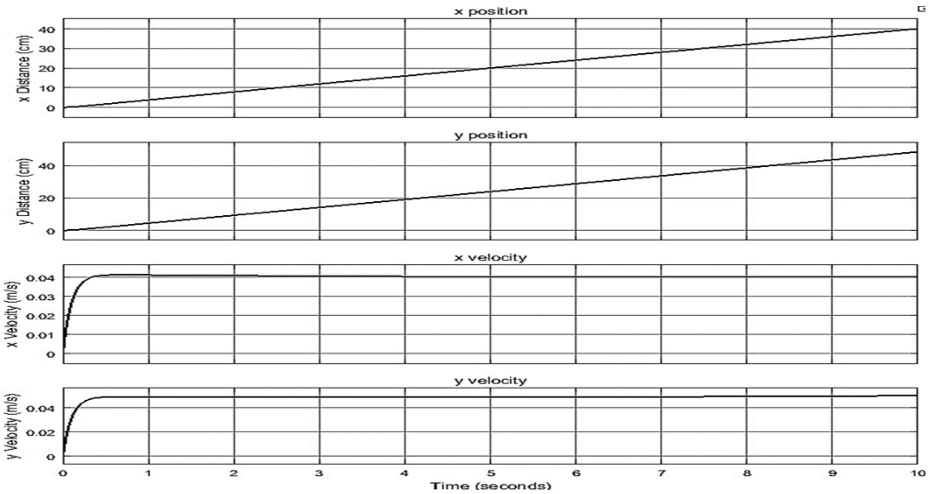

Figure 5. Open loop result

4.2 Closed loop control system results with disturbance

In closed loop control system, the output of the plant is fed back to the system. This makes the PID algorithm to be able to balance and centralize the ball even in the presence of small disturbances. Results obtained from the closed loop control system with small disturbances applied is hereby presented.

4.3 Four degrees pitch and roll of the platform base

Figure 6 is the result obtained when the platform base is subjected to a four (4) degrees pitch and roll.

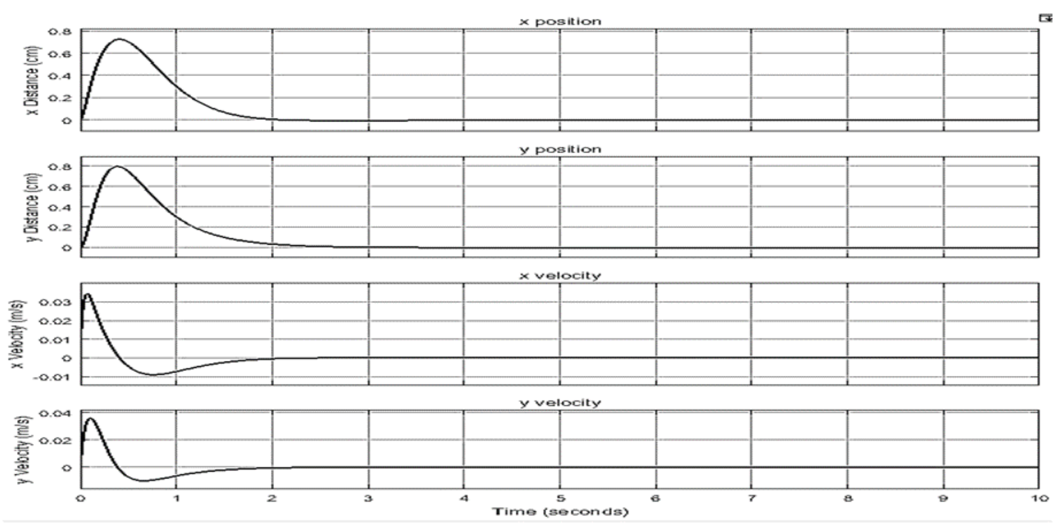

Figure 6. Four degrees roll and pitch

4.4 Twelve degrees pitch and roll of the platform base

Figure 7 is the result obtained when the platform base is subjected to a twelve (12) degrees pitch and roll.

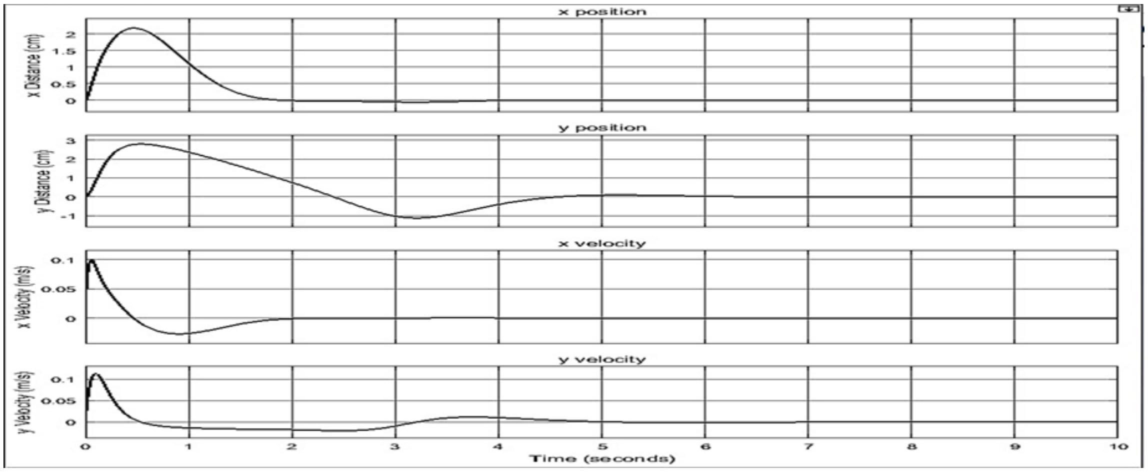

Figure 7. Twelve degrees roll and pitch

4.5 Thirteen degrees pitch and roll of the platform base

Figure 8 is the result obtained when the platform base is subjected to a Thirteen (13) degrees pitch and roll.

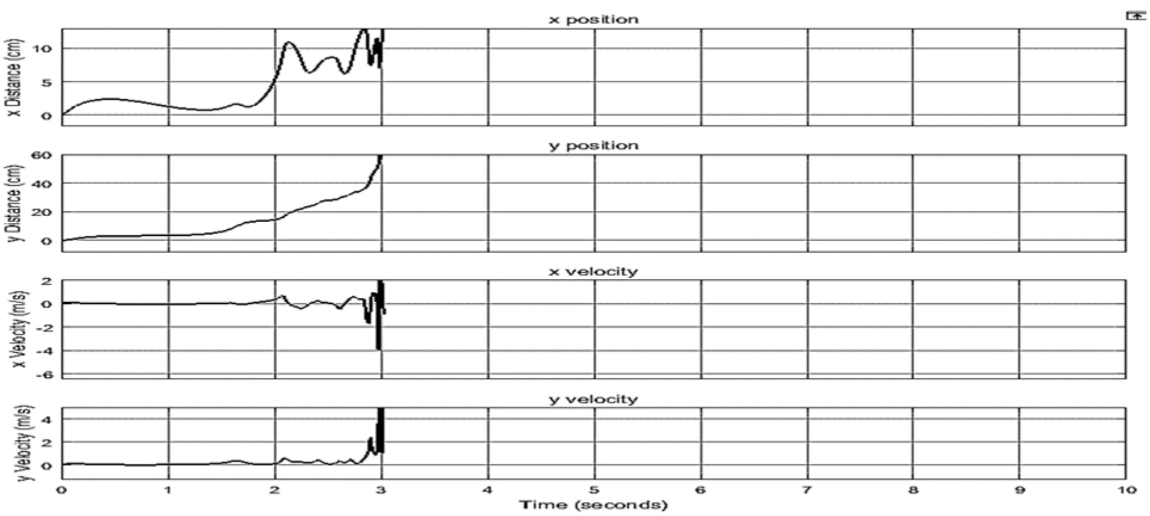

Figure 8. Thirteen degrees roll and pitch

4.6 Discussion of Results

Discussions about the results obtained is presented in the same fashion as the scenarios. Figure 5 is the result obtained in an open loop system with a roll and pitch angle of four degrees applied on the platform base. Generally, four graphs are presented in the results. Each graph indicates the x position, y position, x velocity and y velocity of the ball respectively. These parameters are the process variables that the control system seek to control. However, the absence of a feedback makes it impossible for the PID algorithm to control these process variables thereby subjecting the scenario presented in Figure 5 to that of a ball on an inclined plane. The ball therefore rolls and translates off the top platform. This is seen in the first two graphs of Figure 5 showing how the distance of the ball continue to increase. The last two graphs show the x and y velocity of the ball increasing (acceleration) before becoming constant.

4.7 Comparison with existing work

Figure 9 is a graph showing how this work is compared with an existing work [14]. It is seen that it took approximately 11 seconds to reach settling time in Zeeshan's work while in this research, the longest settling time is approximately 6 seconds when the base was rotated 12^0. The superior result is not unconnected to the use of optical sensors to track the position of the ball.

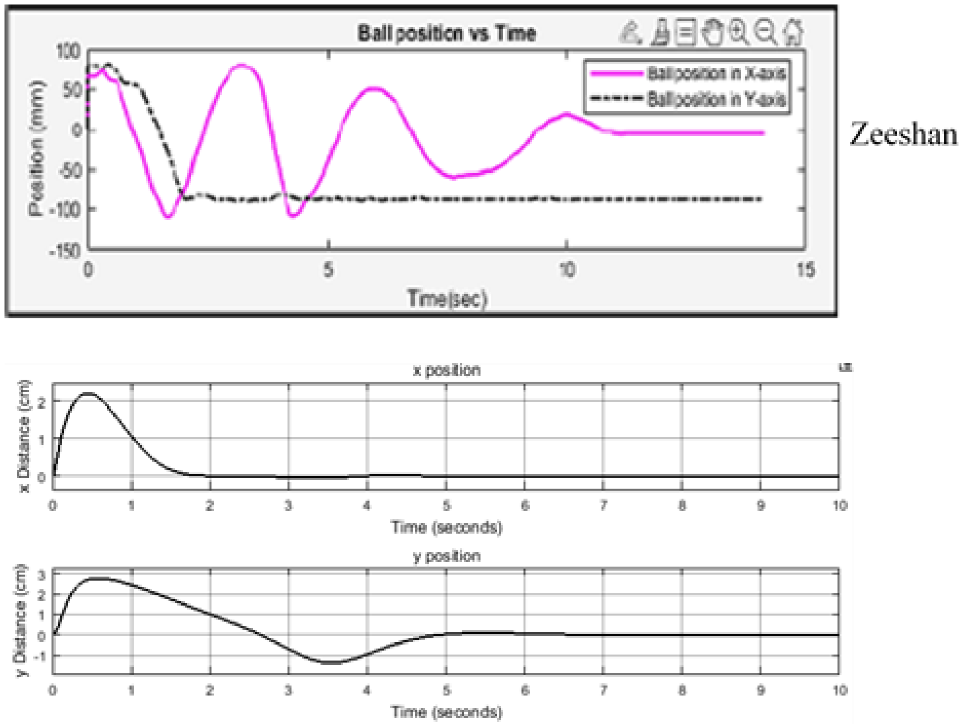

Figure 9. Comparison with existing work

5 CONCLUSIONS

This research successfully developed a dynamic model of a Stewart Platform (SP) using Simulink's Simscape Multibody software. After developing the SP model, a 9x9 array of optical sensors is designed using photodiodes with the central sensor being a single photodiode at position (5, 5). The sensor array successfully tracked the position and velocity of the ball that the system intends to control. Also, a PID controller is developed in both x and y axis to balance and centralize the ball on the SP in the presence of disturbances. The disturbances were in the form of tilting the base of the SP. After carefully developing all components of this research work, simulation was carried and results revealed that a system response of six seconds is obtained. This work proved to be five times faster than Zeeshan work.

REFERENCES

1. Shariatee, M. and Akbarzadeh, A. (2021). Optimum Dynamic Design of a Stewart Platform with Symmetric Weight Compensation System. *Journal of Intelligent & Robotic Systems*, vol. 12, no. 2, pp. 51-63.
2. Petr, N. and Dominik, W. (2020). Investigation of the Stewart Platform Workspace Using MATLAB-Simulink and Simscape Multibody Library. *presented at the 21st International Conference on Research and Education in Mechatronics (REM).*
3. Datta, S., Dash, A. and Gayen, R.K. (2021). Kinematic Analysis of Stewart Platform using MATLAB. *Presented at the 5th International Conference on Electronics, Materials Engineering & Nano-Technology*.

4. Chevalier, A., Copot, C., Ionescu, C.M. and De Keyser, R. (2016). Automatic calibration with robust control of a six DoF mechatronic system. *Journal of Mechatronics*, vol. 35, pp. 102-108.
5. Bang, H. and Lee, Y.S. (2018). Implementation of a ball and plate control system using sliding mode control. *IEEE Access*, vol. 6, pp. 401-408.
6. Kassem, A., Haddad, H. and Albitar, C. (2015). Commparison between different methods of control of ball and plate system with 6dof stewart platform. *IFAC-PapersOnLine*, vol. 8, no. 11, pp. 47-52.
7. Mohammadi, A. and Ryu, C. (2020). Neural network-based pid compensation for nonlinear systems: ball-on-plate example. *International Journal of Dynamics and Control*, vol. 8, no. 1, pp. 178-188.
8. Kempers, B. (2023). MPC control for motion systems. *Masters, Mechanical Engineering, Eindhoven University of Technology, Eindhoven*, 2023.
9. Howimanporn, S. and Chookaew, S. (2022). Monitoring and Controlling of a Real-Time Ball Beam Fuzzy Predicting Based on PLC Network and Information Technologies. *Journal of Advances in Information Technology*, vol. 13, pp. 156-169.
10. Samad, B.A., Anayi, F., Melikhov, Y. and Mohammad, M. (2022). Modelling of LQR and Fuzzy-LQR Controllers for Stabilisation of Multi-link Robotic System (Robogymnast). *Czech Republic*.
11. Singh, A., Shukla, A.K. and Ravinder, P. (2015). HgCdTe e-avalanche photodiode detector arrays. *AIP Advances*, vol. 5, no. 8, pp. 41-57.
12. Aull, B.F., Erik, K.D., Jonathan, P.F., Alexander, K.M., Daniel, R.S. and Richard, D.Y. (2018). Large-Format Geiger-Mode Avalanche Photodiode Arrays and Readout Circuits. *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 24, no. 2, pp. 1-10.
13. Abhishek, K., Singh, S.N., Shweta, T., Sanjay, K.S., Mrinal, D. and Prathap, P. (2023). A novel analytical method for determination of diode parameters from the darkforward IV characteristics of a silicon solar cell. *Physica Scripta*, vol. 98, no. 9, pp. 541-558.
14. Zeeshan, A., Mudasser, W., Muzzamil, W., Usama, A. and Ali, R. (2022). Design and development of Stewart Platform to balance the ball using fuzzy logic control. *Presented at the 2022 International Conference on IT and Industrial Technologies (ICIT), Chiniot, Pakistan*, 2022.