BALL BALANCING ON 3-RRS PARALLEL MANIPULATOR USING PD AND LQR CONTROL

Huy-Hoang Pham, Duc-Huy Doan*, Hoang-Nguyen Vo, Van-Tuong Dang, Hoang-Quoc-Bao Nguyen, Trieu-Vy Le, Thi-Thuy-Quynh Truong, Tien-Dat Tran

Ho Chi Minh city University of Technology and Education (HCMUTE)
Vo Van Ngan Street, No. 01, Ho Chi Minh city, Vietnam
***** Corresponding author. E-mail: 19151235@student.hcmute.edu.vn

Abstract: Ball and plate system (BPS) is known as a nonlinear multi-input multi-output (MIMO) system that has been usually used to investigate and develop control algorithms. In this paper the study deals with a 3-RRS parallel manipulator which has a metal ball, three DC motors with linkage mechanism to a fixed base that can be a resistive touch screen . The moving base has two rotational movements, it can be rotated around the x-axis and y-axis and translated along the z-axis. The main purpose of this study is obtaining dynamics equations of motions and designing PD and LQR controllers. Through simulation and experiment, both control algorithms are proved to work well on this model.
Keywords: 3-RRS parallel manipulator, ball and plate, LQR control, PD control.

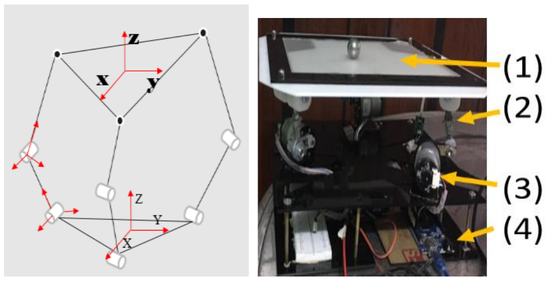
1. Introduction

Balancing in modern control engineering is always a challenge due to its complexity in control robots. There has been plenty of projects for this similar model, such as, ball-and-beam systems and kinds of pendulums. Ball-on-plate is popular nowadays and it is an improvement of the ball-and-beam. Ball is able to roll freely on BPS plate which is operated in two directions. Specifically, ball-on-plate provides two things. First, the ball will be kept stable at one position by using stabilization control. Second, trajectory tracking controller makes the ball follow a certain trajectory with small error in least time.

Ball-on-plate has been utilized and researched by many control algorithms, such as, LQR controller and nonlinear control such as sliding mode control (SMC). Study of [1] has shown accuracy of control quality is about 2.66x10^{-5} and 3.0xm^2 on x and y directions .The PID neural network controller which is optimized by genetic algorithm (GA) for BPS is engaged in which GA provides training weighting factor of multilayered forward neural network [2]. This controller has flexbility, hardiness and a fine performance. Study [3] is a report about design, improvement and control strategies of a ball on a 2-DOF plate by using phototransistor sensors which are provoked by laser beams. In study [4], there is a comparision between a integral SMC and state feedback controllers for BPS, where both of it are programmed to track to a fixed specified trajectory. The result shows that SMC is extremely fast and handle to obtain a better response at higher speed. Implementation in study [5] uses a PID controller which is used to balance ball on 2-DOF system and a single PD controller is used to detect the ball's postion. In study [6], another comparison between a SMC and state feedback controllers shows that these algorithms make system to follow well trajectories. SMC gives quick response with a significant error of about 0.11m. Another BPS model [7] uses dynamic and MTJ to control the ball following a square trajectory with a very small error. PSO Fuzzy Neural Network controller [8] and new Fuzzy Adaptive Control [9] structures are also used to applied to BPS in simulation. However, experimental results are lacked in these researches.

In this paper, we examine 3-DOF BPS – which is more challenging than usual other 2-DOF manipulators. The difference between these two models is that 3-DOF BPS has three – instead of two - angular freedoms to rotate platform. Thence, the ball is able to move along in three axes. After modeling mathematical dynamic equations of motion from Euler-Lagrangian method, suitable LQR and PD controllers will be applied to control the ball on the plate on both simulation and experiment.

2. Dynamics Equation of BPS



(a) (b)

Fig. 1. Form of BPS (a)-common form (b)-complete system

1 - Resistive touch screen
2 - Active Limbs
3 - Dc motor JGB37-520 70RPM
4 - Arduino Mega 2560

2.1. Mathematical Model for BPS

Ball will be stable at one point on the surface so it is necessary to know the ball's position and velocity. The model is built on the following assumptions:
- BPS is kept in touch and there is no slipping constraint.
- Surface must be homogeneous and symmetric.
- The motor connection to the tilt axis must be rigid.

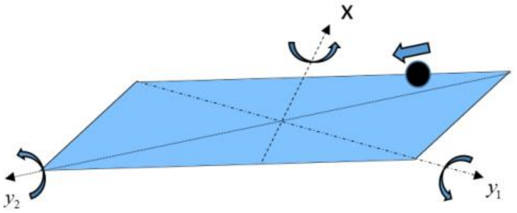


Fig. 2. The plate diagram of BPS

The ball rolls freely on the plate along x-axis and y-axis. However, y-axis is divided into 2 axes - y1 axis and y2 axis - corresponding to two diagonals which come from the center of the equilateral triangle. Diagram is shown in Fig. 2.

The Lagrange equation is written as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial q_i'}\right)-\frac{\partial L}{\partial q_i}=Q_i \qquad (1)$$

Kinetic energy of ball is calculated by rotational energy relative to the center of ball:

$$T_b=\frac{1}{2}m_b(x'^2+y_1'^2+y_2'^2)+\frac{1}{2}I_b(\omega_x{}^2+\omega_{y_1}{}^2+\omega_{y_2}{}^2) \qquad (2)$$

where T_b is kinetic energy, m_b is mass of the ball; I_b is inertia moment of the ball; (x, y_1, y_2) and ($\omega_x, \omega_{y_1}, \omega_{y_2}$) are the position and angular velocity of the rolling ball.

Based on theories above, connections between translational and rotational velocities are

$$x'=r_b\omega_x\ ,\ y_1'=r_b\omega_{y_1},\ y_2'=r_b\omega_{y_2} \qquad (3)$$

where r_b is the radius of the ball .

Replacing (3) into (2), kinetic energy of ball can be written as:

$$T_b=\frac{1}{2}\left(m_b+\frac{I_b}{r_b^2}\right)\left(x'^2+y_1'^2+y_2'^2\right) \qquad (4)$$

Kinetic energy of the plate is calculated as:

$$T_p=\frac{1}{2}\left(I_b+I_p\right)\left(\theta_1'^2+\theta_2'^2+\theta_3'^2\right)+\frac{1}{2}m_b\left(x\theta_1'+y_1\theta_2'+y_2\theta_3'\right)^2 \qquad (5)$$

where $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3$ are angle rates of inclination about x-axis, y1-axis and y2-axis. Then, total kinetic energy of system is T=T_b+T_p. Moreover, potential energy of the ball is:

$$V=m_bgh=m_bgh\left(x\sin\theta_1+y_1\sin\theta_2+y_2\sin\theta_3\right) \qquad (6)$$

Then, Lagragian operator is calculated from (6) and (5).

Applying Euler-Lagrange equation (1), non-linear equations of motion for BPS are:

$$x''\left(m_b+\frac{I_b}{r_b^2}\right)+gm_b\sin\theta_1-m_b\theta_1'\left(\theta_1'x+\theta_2'y_1+\theta_3'y_2\right)=0 \qquad (7)$$

$$y_1''\left(m_b+\frac{I_b}{r_b^2}\right)+gm_b\sin\theta_2-m_b\theta_2'\left(\theta_1'x+\theta_2'y_1+\theta_3'y_2\right)=0$$

$$y_2''\left(m_b+\frac{I_b}{r_b^2}\right)+gm_b\sin\theta_3-m_b\theta_3'\left(\theta_1'x+\theta_2'y_1+\theta_3'y_2\right)=0$$

$$\tau_1=\theta_1''(I_b+I_p)+m_bx(\theta_1''x+\theta_2''y_1+\theta_3''y_2)+gmx\cos\theta_1$$

$$\tau_2=\theta_2''(I_b+I_p)+m_by_1(\theta_1''x+\theta_2''y_1+\theta_3''y_2)+gmy_1\cos\theta_2$$

$$\tau_3=\theta_2''(I_b+I_p)+m_by_2(\theta_1''x+\theta_2''y_1+\theta_3''y_2)+gmy_2\cos\theta_3$$

Tab. 1. Parameters of BPS

Parameters	Descriptions	Values
m_b(kg)	Ball mass	0.3
m_p(kg)	Plate mass	0.6
I_b(kg.m^2)	Ball moment of inertia	1.8 x 10^{-5}
I_p(kg.m^2)	Plate moment of inertia	6.6 x 10^{-3}
r_b(m)	Ball radius	0.022

2.2. Mathematical Model for DC Motor

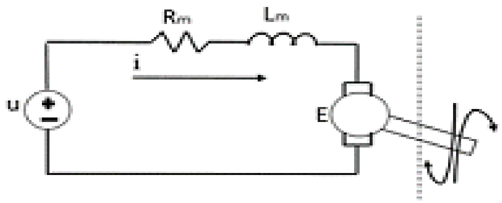


Fig. 3. DC motor circuit

DC motor structure has two parts: electricity and mechanism.

Electricity equation is

$$e=L_m\frac{di}{dt}+R_mi+E \qquad (8)$$

where

$$E=K_b\omega \qquad (9)$$

Replacing (9) into (8), we have:

$$e=L_m\frac{di}{dt}+R_mi+K_b\omega \qquad (10)$$

Mechanism:

$$J_m\frac{d\omega}{dt}=K_ti-C_m\omega-\tau_m \qquad (11)$$

Using principle conversation of energy

$$P_e = P_m;\ Ei = \tau_i \omega;\ K_b \omega i = K_t \omega i;\ K_b = K_t \tag{12}$$

Since the electrical speed is faster than the mechanism speed $e >> L_m \frac{di}{dt}$, the inductance component is so small that we can ignore it

$$\tau_i = K_t i = \frac{K_t}{K_m} e_i - \frac{R_t}{R_m} K_b \omega \tag{13}$$

Rotational angle of motor is equal to rotational angle ($\omega = \dot{\theta}_i$) of plate, τ_i can be written as

$$\tau_i = \frac{K_t}{R_m} e_i - \frac{K_t}{R_m} K_b \theta_i' \tag{14}$$

Replacing (14) into (11), it yield

$$\tau_m = -K_3 \theta_i'' - K_2 \theta_i' + K_1 e_i \tag{15}$$

where $K_1 = \frac{K_t}{R_m}; K_2 = C_m + \frac{K_t}{R_m} K_b; K_3 = J_m$

Based on (15), (7) can be written as

$$\begin{aligned} &-K_3\theta_1'' - K_2\theta_1' + K_1 e_1 = \theta_1''\left(I_b + I_p\right) + \\ &m_b x\left(\theta_1'' x + \theta_2'' y_1 + \theta_3'' y_2\right) + gmx\cos\theta_1 \\ &-K_3\theta_2'' - K_2\theta_2' + K_1 e_2 = \theta_2''\left(I_b + I_p\right) + \\ &m_b y_1\left(\theta_1'' x + \theta_2'' y_1 + \theta_3'' y_2\right) + gmx\cos\theta_2 \\ &-K_3\theta_3'' - K_2\theta_3' + K_1 e_3 = \theta_2''\left(I_b + I_p\right) + \\ &m_b y_2\left(\theta_1'' x + \theta_2'' y_1 + \theta_3'' y_2\right) + gmx\cos\theta_3 \end{aligned} \tag{16}$$

Parameters of DC motor are given in Tab. 2 below

Tab. 2. Parameters of DC motor

Parameters	Description	Values
R_m (Ω)	Resistance of motor	12
K_b (V/rad/s)	Velocity constant of motor	0.19
J_m (kg.m^2)	Motor inertia	0.00006
C_m (Nm/rad/s)	Friction coefficient of motor	0.00006

3. Control design for BPS

3.1. State-Space Equation

We linearize nonlinear equations into linear equations around its working space [x_0, y_0]. In addition, motors rotate a small angle and initial velocity almost zero.

General linear state-space equation is written as:

$$dy/dx = Ax + Bu \tag{17}$$

where x=[$x, \dot{x}, y_1, \dot{y}_1, y_2, \dot{y}_2, \theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2, \theta_3, \dot{\theta}_3$]; u=[$u_1, u_2, u_3$]; ma trix A and B are shown in Appendix of this paper

3.2. Linear Quadratic Regulator (LQR)

LQR controllers are often applied on non-linear systems with multiple inputs and outputs. From system variables, we calculate and convert it into control signal for the system. Matrix Q and R use for calculate K inside LQR controller as follow:

Q=eye(12); R=eye(3); Q(1,1)=500; Q(3,3)=500; Q(5,5)=500; K=lqr(A,B,Q,R) (18)

Control block diagram is presented in Fig. 4

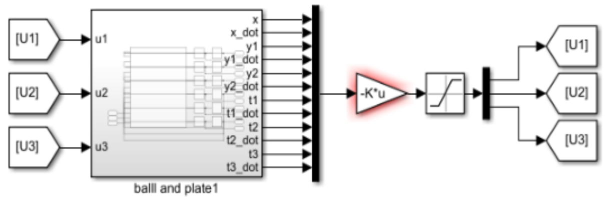


Fig. 4. BPS block diagram

3.2. Linear Quadratic Regulator (LQR)

PD based on PID when "I" equal zero. PID is a loop feedback controller which is popular in industry. For BPS, PID structure is shown in Fig. 5.

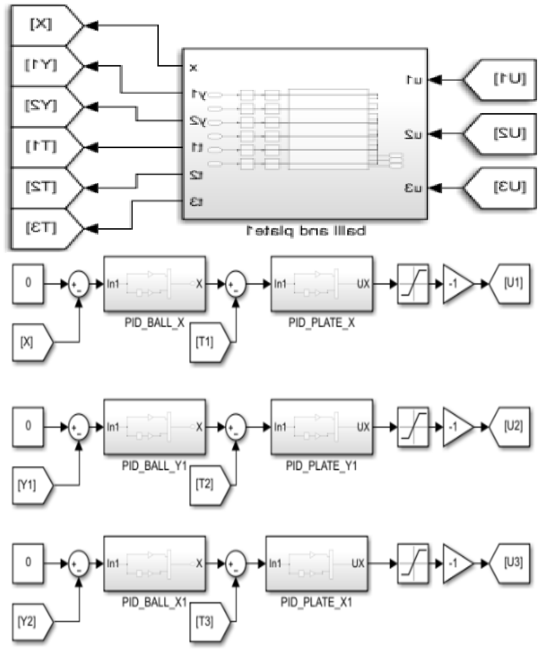


Fig. 5. Block diagrams State-Space of BPS and PD controllers

Six PD controllers are applied in addition to three positions and three rotations of DC motors. Each of PD parameters is based on trial-and-error test and they are used for the other axis. Kp and Kd are shown in Tab. 3.

Tab. 3. K_p and K_d parameters

Parameters	Values
K_p_x	95
K_d_x	95
K_p_θ	80
K_d_θ	2

4. Simulation results

4.1. LQR controller

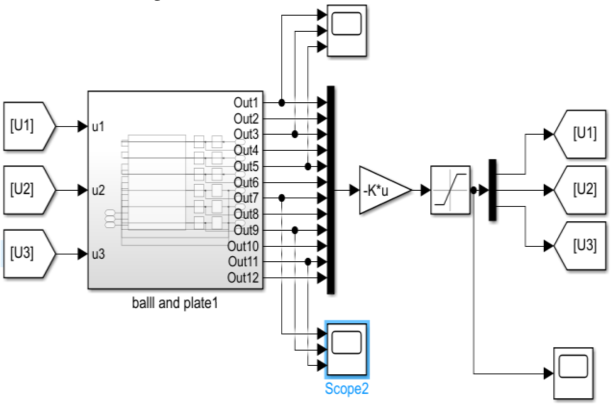


Fig. 6. Ball plate system with LQR controller

Ball is stable at desired point around 6 sec in Fig. 7. Rotational angle of the plate is small in Fig. 8. It is about between $\pm 4 \times 10^{-3}$ rad. So, LQR controller successfully works. In both Fig. 7 and Fig. 8, settling time is around 8 sec.

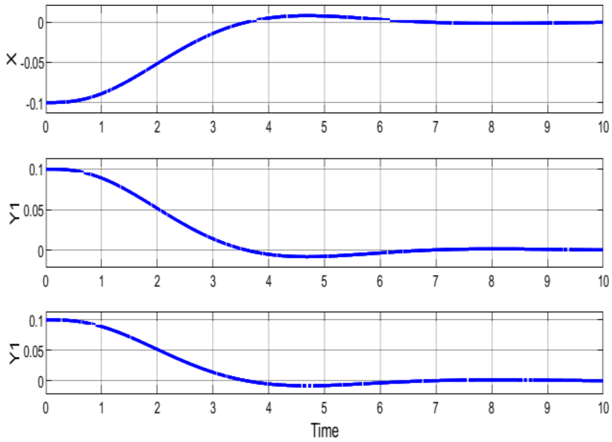


Fig. 7. Ball's position on the surface

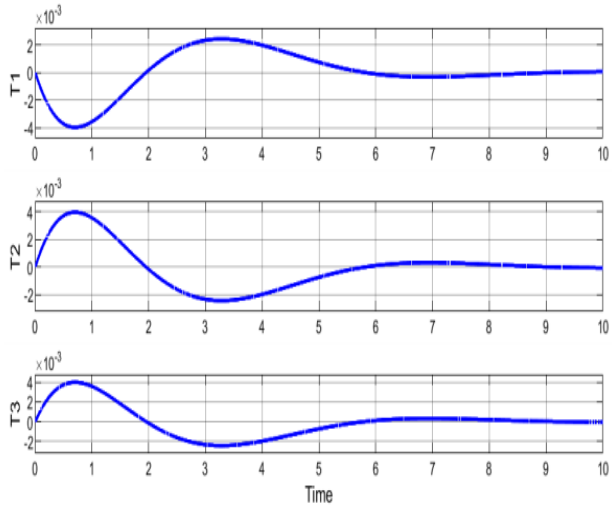


Fig. 8. Rotational angle of plate

4.2. PD controller

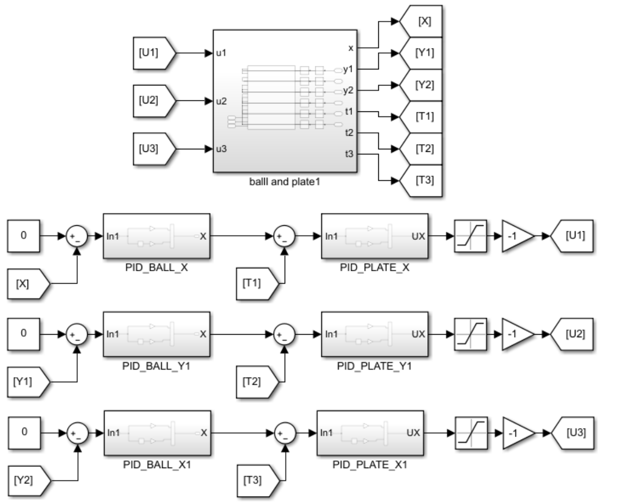


Fig. 9. BPS with PD controller

Setting time of the system is 2 sec with rotational angle is 0.03 rad. There is no overshoot in Fig. 10. In Fig. 9, the ball gets to the desired point quickly in 2 sec. Consequently, PD controller adapts for BPS well.

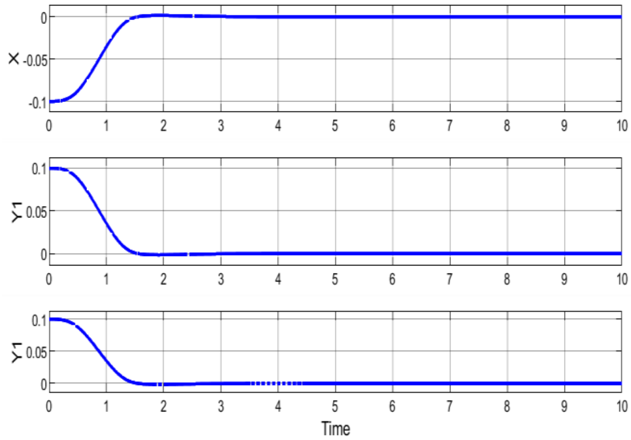


Fig. 10. Ball's position on the plate

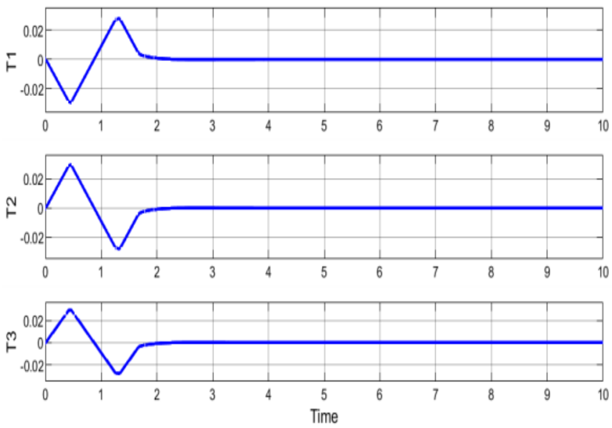


Fig. 11. Rotational angle of plate

In this section, results of two controllers are shown while ball is tracking trajectory. Using PID, BPS is tested to control ball on desired point that is near center of plate.

In Fig. 11**,** ball needs nearly 1.8 sec to get to the desired point with a least error 2mm. Then, ball was pushed by hand and it takes 6 sec to get stable again. Ball position trajectory is presented in Fig. 11.

5. Experimental results

Experimental results of real PBS under PID and LQR controllers are shown in Fig. 12 and Fig. 13.

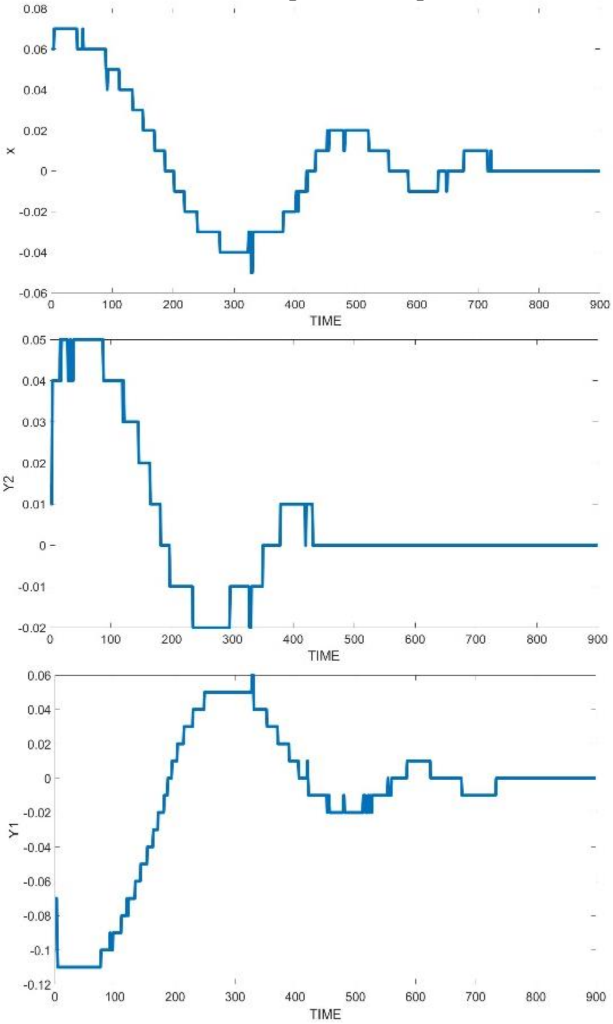


Fig. 12. x, y_1, y_2 position (m) of ball under PID controller through time (msec)

In Fig. 12, after 8 sec, PID controller makes system stable in x, y_1, y_2 position with no error. In Fig. 13, errors of positions in x and y_1 are about 10 mm. But, system under LQR controller gets shorter settling time than under PID controller: settling time is 3 sec under LQR control and 8 sec under PID control. Thence, LQR get better settling time when PID get better settling error when being applied for BPS.

In Fig. 14, we apply LQR for making the ball to follow the rectangle trajectory. LQR is proved to work well a\when the ball follow well the trajectory with error is betwwen 0 to 0.5 cm.

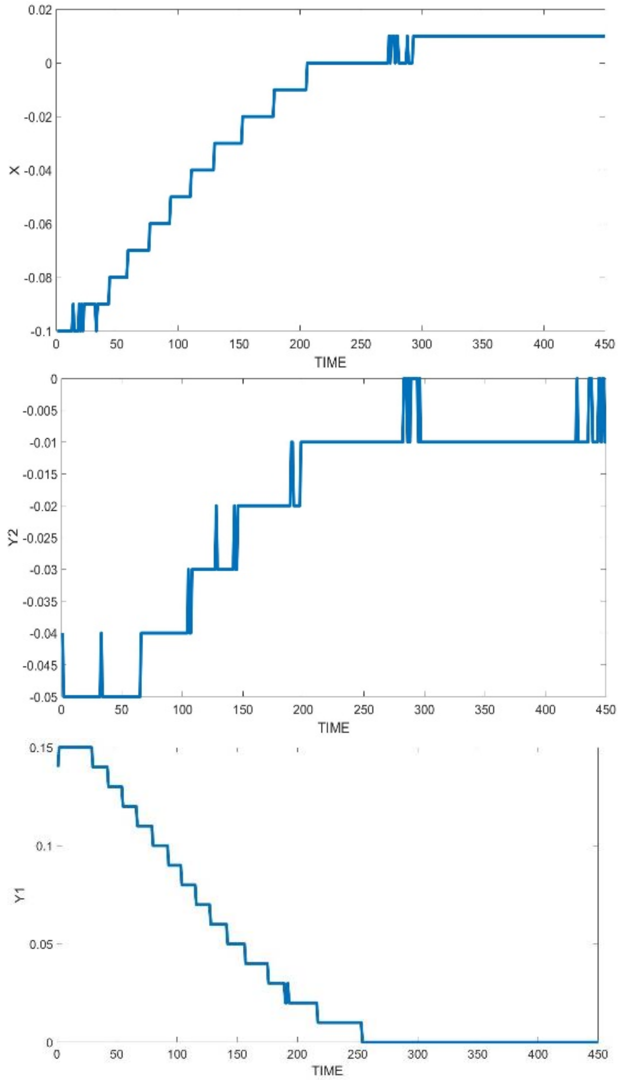


Fig. 13. x, y_1, y_2 position (m) of ball under LQR controller (msec)

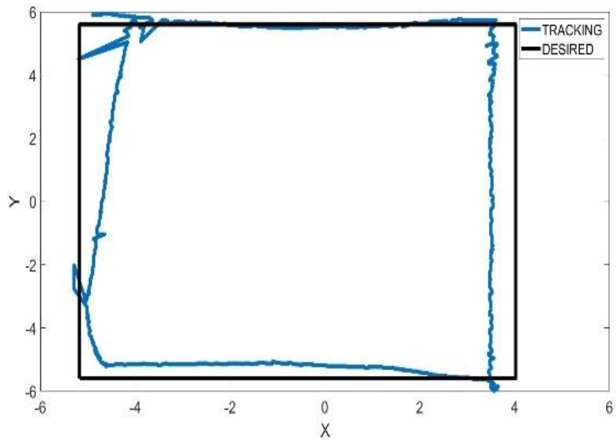


Fig. 14. Tracking trajectory ball on moving platform under LQR control (size in cm)

6. Conclusion

In this paper, we present a BPS which is developed form 2-DOF ball-and-plate. Through Euler-Lagrange method, dynamic equations are presented. An experimental model is also introduced for laboratory test. Based on this model, we propose a LQR and PD controllers for both simulation and experiment. These methods are proved to make system balanced and the methods also make system to follow trajectories well.

Acknowledgement

We want to give thanks to PhD. Van-Dong-Hai Nguyen (Faculty of Electrical and Electronics Engineering – HCMUTE) due to his supports in creating this hardware.

7. References

[1] Pour R.K. et al.: "Fuzzy Logic Trajectory Tracking Control of a 3-RRS Ball and Plate Parallel Manipulator" International Conference on Robotics and Mechatronics, pp. 343-347, IEEE, Iran, 2016.

[2] Dong X. et al.: "Applying genetic algorithm to on-line updated PID neural network controllers for ball and plate system," In Innovative Computing, Information and Control (ICICIC), Fourth International Conference on, IEEE, pp. 751-755, 2009.

[3] Zeeshan A. et al.: "Design, control and implementation of a ball on plate balancing system," In Proceedings of International Bhurban Conference on Applied Sciences & Technology (IBCAST), pp. 22-26, IEEE, 2012.

[4] Debono D., Bugeja M.: "Application of sliding mode control to the ball and plate problem", 12th International Conference on Informatícs in Control, Automation and Robotics(ICINCO), Vol. 1, pp. 412-419, IEEE, 2015.

[5] Awtar S. et al.: "Mechatronic design of a ball-on-plate balancing system," Mechatronics, Vol. 12, No. 2, pp. 217-228, 2002.

[6] Yang G. et al.: "Analysis and design of a 3-DOF flexure-based zero-torsion parallel manipulator for nano-alignment applications," In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pp. 2751-2756, May 2011.

[7] Bang H.S., Lee Y.S.: "Implementation of a ball and plate control system using sliding mode control", IEEE Access, 6:32401-32408, 2018.

[8] Dong X. et al.: "Design of PSO fuzzy neural network control for ball and plate system," International Journal of Innovative Computing, Information and Control, Vol. 7, No. 12, pp. 7091-7103, 2011.

[9] Lian-Kui Q. et al.: "Fuzzy adaptive control scheme for ball and plate system and comparison with PD and MPC controller," International Journal of Digital Content Technology and its Applications, Vol. 7, No. 9, 2013, pp. 572.

APENDIX

Matrix A and B in (17) are calculated as:

$$A = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -8.728 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -8.728 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -8.728 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1.731 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -263.857 & 0 & 0 & 0 & 0 \\
0 & 0 & -1.731 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -263.857 & 0 & 1 \\
0 & 0 & 0 & 0 & -1.731 & 0 & 0 & 0 & 0 & 0 & 0 & -263.857
\end{bmatrix}; \quad
B = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
1.564 & 0 & 0 \\
0 & 0 & 0 \\
0 & 1.564 & 0 \\
0 & 0 & 0 \\
0 & 0 & 1.564
\end{bmatrix}.$$