# APPLICATION OF PID-FUZZY CONTROL FOR PENDUBOT

ỨNG DỤNG ĐIỀU KHIỂN PID-MỜ CHO HỆ PENDUBOT

**Tran Hoang Chinh, Nguyen Minh Tam, Nguyen Van Dong Hai**
*Ho Chi Minh city University of Technology and Education, Vietnam*

Received 03/10/2017, Peer reviewed 11/10/2017, Accepted for publication 25/10/2017

## ABSTRACT

*PID is a classical control method. Control parameters of PID can be chosen due to experts' experiences, error test or searching algorithm, such as, genetic algorithm (GA) or swarm algorithm (SA). However, the fact that parameters* $K_p$, $K_i$, $K_d$ *are fixed in operating time of system describes a disadvantage which makes system non-flexible enough to adapt highly with system's condition. This paper presents a method of using fuzzy algorithm to change flexibly control parameters of PID when system operates. Thence, control quality is improved. Object which is used for testing in this paper is pendubot- a popular model in laboratories of control engineering, balance control for pendubot at Top position as a requirement. Using PID-Fuzzy algorithm for this model is proved to be more optimal than PID through simulation (Matlab/Simulink) and experiment.*

***Keywords:*** *pendubot; PID control; fuzzy control; PID-fuzzy algorithm; balance control; Top-position; inverted pendulum.*

## TÓM TẮT

*Vi tích phân tỉ lệ (PID) là một giải thuật điều khiển kinh điển. Việc lựa chọn thông số PID có thể dựa vào kinh nghiệm chuyên gia, phép thử sai hoặc giải thuật tìm kiếm như giải thuật di truyền (genetic algorithm - GA) hoặc bầy đàn (swarm algorithm - SA). Tuy nhiên, việc thông số* $K_p$, $K_i$, $K_d$ *không thay đổi trong suốt quá trình hoạt động của hệ thống cũng là một nhược điểm làm bộ điều khiển không linh động thay đổi để thích ứng cao với trạng thái hệ thống. Bài báo giới thiệu một cách thức sử dụng bộ điều khiển mờ để linh động thay đổi các thông số của bộ điều khiển PID. Từ đó, chất lượng điều khiển của hệ thống được cải thiện. Đối tượng được sử dụng trong bài báo là mô hình Pendubot- một mô hình thông dụng trong các phòng thí nghiệm về điều khiển tự động, yêu cầu đặt ra là điều khiển cân bằng cho pendubot ở vị trí Top. Việc sử dụng giải thuật PID-mờ cho hệ thống trên được chứng minh tối ưu hơn PID trên cả mô phỏng (Matlab/Simulink) và trên mô hình thực tế.*

***Từ khóa:*** *pendubot; điều khiển PID; điều khiển mờ; giải thuật PID-mờ; điều khiển cân bằng; vị trí Top; con lắc ngược.*

## 1. INTRODUCTION

Research control under-actuated systems more and more necessary for society. Besides, "Industry 4.0" is coming soon, robotics will work instead of worker in producing at factory. Designing robotics, upgrading control algorithm require more challenges for engineering.

Under-actuated system is a system which has less number of actuators than degrees of freedom. Research control under-actuated systems is a way to apply controllers for robotics, industry,…

Under-actuated systems include: inverted pendulum, ball and beam system, cart and pole,... These systems are non-stabilized and difficult to be controlled at balancing position.

The pendubot is an interesting example of inverted pendulum; it is an under-actuated system since the number of its control inputs is less than the number of its degrees of freedom, which makes it difficult to control. However, controlling such systems is challenging due to nonlinear dynamics, non-holonomic behavior, and lack of linearizability exhibited by these systems. The pendubot itself is a two-link planar robot with a single actuator at the base (shoulder) of the first link, and the (elbow) joint between the two links is unactuated and allowed to swing freely [5].

Most industrial processes nowadays are still controlled by PID controllers [3]. It's easy to meet PID controllers in industrial applications: control speed motor, auto system in electric companies, hydroelectric power plant factories…

Fuzzy Logic controller (FLC) is controller which is designed by experiences. FLC is one of the intelligent algorithms that bring high productivity.

Pendubot is a non-stabilized and complicated single input-multi output (SIMO) system which is highly nonlinear. This model is popularly used in testing control algorithm in laboratories. With this system, PID controller (PC) - a classical controller - was proved to work well both in simulation and experiment [1-2]. However, in PC, parameters $K_p$, $K_i$, $K_d$ are unchanged in total operating period. This characteristic makes controller non-flexible enough to adapt condition of system to increase control quality. Fuzzy algorithm is a suggested solution in this case. Parameters of PC are calibrated when system works.

More before research, sliding mode [4], hybrid control [6], LQR, has used for control pendubot. Now, authors would like to design a controller by PID and Fuzzy logic controller together. We use theory, next simulate, then record through experiment by UART board connected to STM32F407VG board. Data is shown on laptop to verify results.

Paper consists of five sections. Section 1 presents topic. Then, section 2 describes dynamic equation of pendubot. Section 3 introduces method of PID-FUZZY for pendubot. Section 4 listed the simulation results and experimental results. Lastly, conclusion in Section 5 ends the paper.

## 2. SYSTEM MODEL OF PENDUBOT

Mathematical structure of pendubot is shown in Fig. 1. Table 1 listed all variables and parameters of pendubot if we consider input signal control of system as moment created by motor.

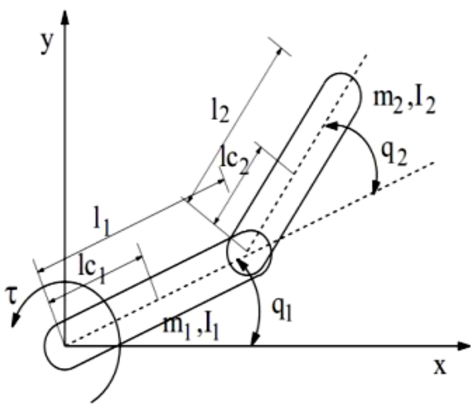


***Fig. 1.*** *Pendubot structure*

***Table 1.*** *Variables/ parameters of system*

| **Variables and parameters** | **Description** |
|---|---|
| $q_1$ | Angle of link 1 |
| $q_2$ | Angle of link 2 |
| $l_1$ | Length of link 1 |
| $l_{c1}$ | Distance from the axis of motor to center of link 1 |
| $l_2$ | Length of link 2 |
| $l_{c2}$ | Distance from the axis of encoder of link 1 to center of link 2 |
| $m_1$ | Mass of link 1 |
| $m_2$ | Mass of link 2 |
| $I_1$ | Inertia moment of link 1 |
| $I_2$ | Inertia moment of link 2 |
| $g$ | Gravitational acceleration |
| $\tau_1$ | Moment on link 1 |
| $\tau_2$ | Moment on link 2 |

From (8.5) and (8.6) of [2], dynamic equations of pendubot is described as

$$\ddot{q}_1 = \frac{\begin{bmatrix} \beta_2\tau_1 + \beta_2\beta_3\left(x_2 + x_4\right)^2 \sin x_3 + \\ +\beta_3^2 x_2^2 \sin x_3 \cos x_3 - \beta_2\beta_4 g \cos x_1 + \\ +\beta_3\beta_5 g \cos x_3 \cos\left(x_1 + x_3\right) \end{bmatrix}}{\beta_1\beta_2 - \beta_3^2 \cos^2 x_3} \quad (1)$$

$$\ddot{q}_2 = \frac{\begin{bmatrix} \left(-\beta_2 - \beta_3 \cos x_3\right)\tau_1 + \beta_4 g\left(\beta_2 + \beta_3 \cos x_3\right)\cos x_1 \\ -\beta_3\left(\beta_2 + \beta_3 \cos x_3\right)\left(x_2 + x_4\right)^2 \sin x_3 + \\ -\beta_5 g\left(\beta_1 + \beta_3 \cos x_3\right)\cos\left(x_1 + x_3\right) + \\ -\beta_3 x_2^2 \sin x_3\left(\beta_1 + \beta_3 \cos x_3\right) \end{bmatrix}}{\beta_1\beta_2 - \beta_3^2 \cos^2 x_3} \quad (2)$$

Where:

$$\beta_1 = m_1 l_{c1}^2 + m_2 l_1^2 + I_1 \quad (3)$$

$$\beta_2 = m_2 l_{c2}^2 + I_2 \quad (4)$$

$$\beta_3 = m_2 l_1 l_{c2} \quad (5)$$

$$\beta_4 = m_1 l_{c1} + m_2 \ l_1 \quad (6)$$

$$\beta_5 = m_2 l_{c2} \quad (7)$$

In this paper, authors control Pendubot system at Top Position.

$$x1 \ = \ q1 - \frac{\pi}{2} \quad (8)$$

$$x2 = \dot{q}_1 \quad (9)$$

$$x3 \ = \ q2 \quad (10)$$

$$x4 \ = \dot{q}_2 \quad (11)$$

In real experiment, the input control signal is voltage applying for motor to generate moment. Therefore, it is necessary to equivalent system into a new form that the signal control input is voltage.

From [1] and Fig. 2, output moment of motor can be consider

$$\tau_m = \frac{K_t}{R_m} e - \left(\frac{K_t K_b}{R_m}\right)\omega \quad (12)$$

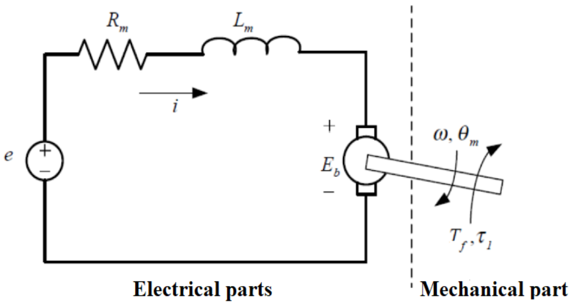


***Fig. 2.*** *The model of DC motor*

Where

$\tau_m$ : moment of motor (Nm)

e : voltage on motor (V)

$\omega$: angle velocity of motor (rad/s)

Combine (1-12), after some calculations, we obtain

$$\dot{x}_1 = x_2 \quad (13)$$

$$\dot{x}_2 = f_1(x) + b_1(x)u \quad (14)$$

$$\dot{x}_3 = x_4 \quad (15)$$

$$\dot{x}_4 = f_2(x) + b_2(x)u \quad (16)$$

Where

$$f_1(x) = \frac{\begin{bmatrix} \beta_2\beta_3 \sin(x_3)(x_2 + x_4)^2 + \\ +\beta_3{}^2 \sin(x_3)\cos(x_3)x_2^2 + \\ -\beta_2\beta_4 g\cos(x_1) - K_2\beta_2 x_2 + \\ +\beta_3\beta_5 g\cos(x_3)\cos(x_1 + x_3) \end{bmatrix}}{K_3\beta_2 + \beta_1\beta_2 - \beta_3{}^2 \cos^2(x_3)} \quad (17)$$

$$b_1(x) = \frac{\beta_2 K_1}{K_3\beta_2 + \beta_1\beta_2 - \beta_3{}^2 \cos^2(x_3)} \quad (18)$$

$$f_2(x) = \frac{\begin{bmatrix} (-\beta_3 \sin(x_3)(\beta_2 + \beta_3\cos(x_3))(x_2 + x_4))^2 - \\ \beta_3 \sin(x_3)x_2^2(\beta_1 + \beta_3\cos(x_3)) - K_3\beta_3 x_2^2 \sin(x_3) + \\ -\beta_5 g\cos(x_1 + x_3)(K_3 + \beta_1 + \beta_3\cos(x_3)) \\ +\beta_4 g\cos(x_1)(\beta_2 + \beta_3\cos(x_3)) + K_2 x_2(\beta_2 + \beta_3\cos(x_3)) \end{bmatrix}}{K_3\beta_2 + \beta_1\beta_2 - \beta_3{}^2 \cos^2(x_3)} \quad (19)$$

$$b_2(x) = \frac{-\beta_2 - \beta_3\cos(x_3)K_1}{K_3\beta_2 + \beta_1\beta_2 - \beta_3{}^2 \cos^2(q_2)} \quad (20)$$

$$K_1 = \frac{K_t}{R_m} \quad (21)$$

$$K_2 = \frac{K_t K_b}{R_m} + C_m \quad (22)$$

$$K_3 = J_m \quad (23)$$

## 3. PID-FUZZY METHOD

With parameters $K_{p1}$, $K_{d1}$, $K_{p2}$, $K_{d2}$, PC can control pendubot around balancing position but operating time is limited. So, in order to work quickly, authors use Fuzzy Controller and PC together. Then, a "PID-Fuzzy Controller" (PFC) is suggested. This controller not only balances pendubot but also stabilizes system with a short settling time.

PC with input variables are error signals and error derivative, and Fuzzy controller with output variables are $u_1$, $u_2$, $u_3$, $u_4$.

The control signal is determined as:

$$\begin{aligned} u = (K_{p1} + U_1)e_1 + (K_{d1} + U_2)\dot{e}_1 + \\ + (K_{p2} + U_3)e_2 + (K_{d2} + U_4)\dot{e}_2 \end{aligned} \tag{24}$$

$K_{p1}$, $K_{d1}$, $K_{p2}$, $K_{d2}$ are parameters of PC.

$$U_1 = K_3 u_1 \tag{25}$$

$$U_2 = K_4 u_2 \tag{26}$$

$$U_3 = K_7 u_3 \tag{27}$$

$$U_4 = K_8 u_4 \tag{28}$$

u1, u2, u3, u4 are outputs from Fuzzy1, Fuzzy2, Fuzzy3, Fuzzy4 controllers.

$$e1(t) = \text{set-point } (q1) - q1(t) \tag{29}$$

$$e2(t) = \text{set-point } (q2) - q2(t) \tag{30}$$

$K_1$, $K_2$, $K_3$, $K_4$, $K_5$, $K_6$, $K_7$, $K_8$ are chosen by experiments.

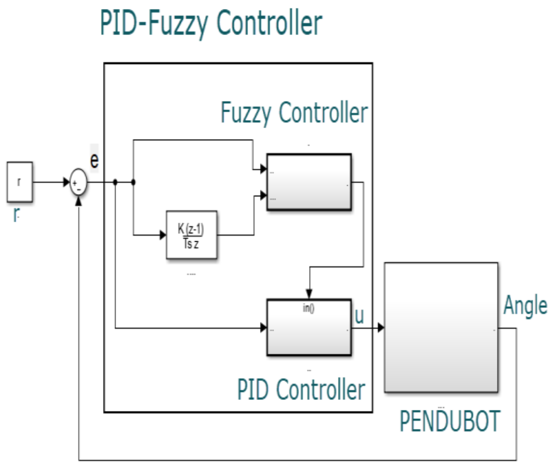


***Fig. 3.*** *The scheme of PFC for Pendubot*

***PC*** in this case includes two smaller PCs: PID1 - controls $q_1$ - and PID2 - controls $q_2$. The scheme of PC is shown in Fig. 4.

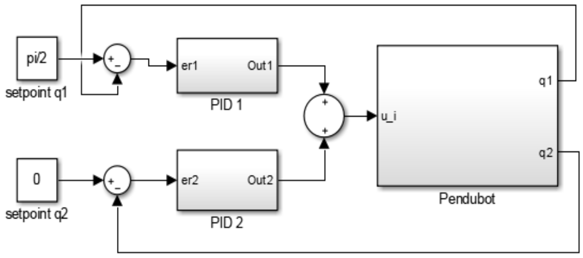


***Fig. 4.*** *The scheme of PC for Pendubot*

In this paper, authors neglect integrative element $I_{out} = K_i \int_0^t e(\tau)d\tau$ (31)

because this element is created by plus of many discrete- errors before system works. So, in real experiment, this element becomes very big just before model operates. Then, PC is regarded useless. PID1 and PID2 control parameters are $K_{p1}$, $K_{d1}$ and $K_{p2}$, $K_{d2}$, correspondingly.

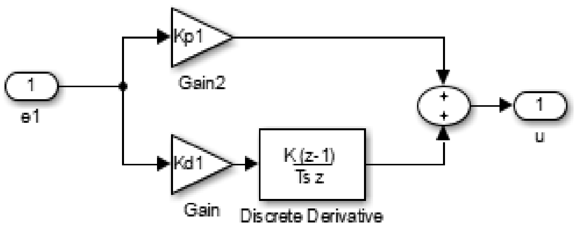


***Fig. 5.*** *The scheme of each PID1 controller*

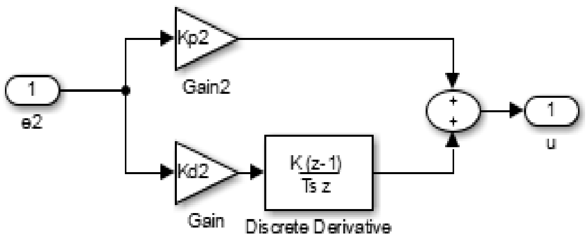


***Fig. 6.*** *The scheme of PID2 controller*

***Fuzzy Controller*** includes 4 controllers: Fuzzy1, Fuzzy2, Fuzzy3, Fuzzy4.

### Designing 4 Fuzzy controllers

According to testing in simulations, determine memberships of $e_1$, $\dot{e}_1$, $e_2$, $\dot{e}_2$ as in Fig. 7-10. Also, memberships of output signal are determined as in Fig. 11-14.

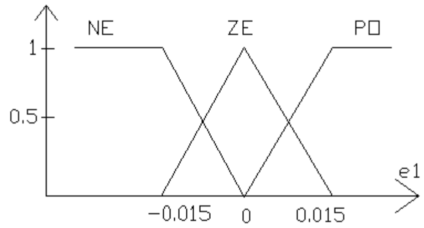


***Fig. 7.*** *Membership functions of standardized* $e_1$ *input Fuzzy 1 controller*

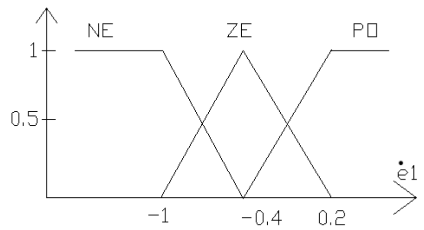


***Fig. 8.*** *Membership functions of standardized* $\dot{e}_1$ *input Fuzzy 2 controller*

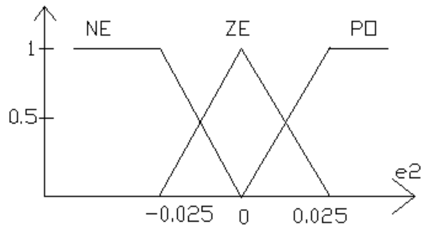


***Fig. 9.*** *Membership functions of standardized* $e_2$ *input Fuzzy 3 controller*

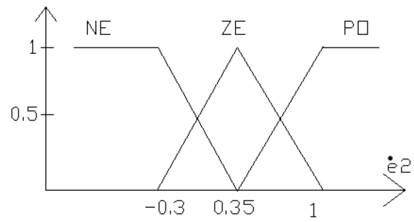


***Fig. 10.*** *Membership functions of standardized* $\dot{e}_2$ *input Fuzzy-4 controller*

According to testing in simulations, determined $\dot{e}_2$ input from -0.3 to 1.

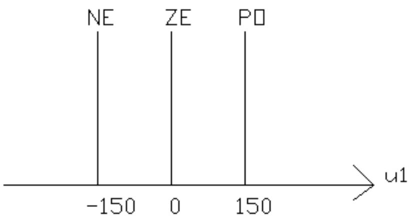


***Fig. 11.*** *Membership functions of standardized* $u_1$ *output Fuzzy-1 controller*

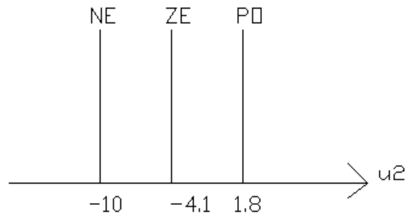


***Fig. 12.*** *Membership functions of standardized* $u_2$ *output Fuzzy-2 controller*

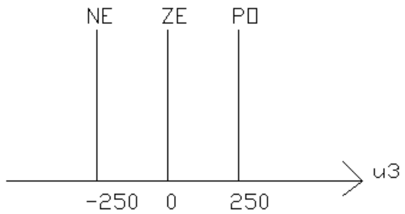


***Fig. 13.*** *Membership functions of standardized* $u_3$ *output Fuzzy-3 controller*


***Fig. 14.*** *Membership functions of standardized* $u_4$ *output Fuzzy-4 controller*

**Fuzzy Controllers Rules**

The number of rules are used in fuzzy controller is 3 for each Fuzzy controller. Some rules are listed below:

If input is NE then output is NE

If input is ZE then output is ZE

If input is PO then output is PO

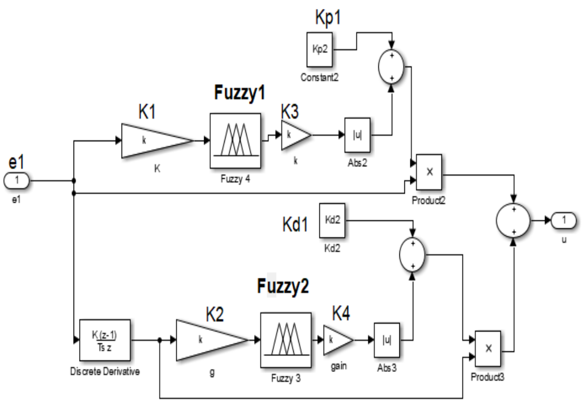


***Fig. 15.*** *The scheme of PID 1-Fuzzy controller for pendubot*

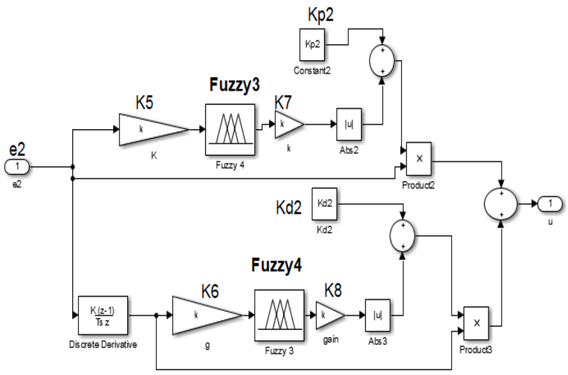


***Fig. 16.*** *The scheme of PID 2-Fuzzy controller for pendubot*

## 4. SIMULATION AND EXPERIMENTAL RESULTS

From Fig. 17, parameters in simulation and experiment are determined the same as:

$m_1$=1.068($Kg$); $m_2$=0.07($Kg$); $l_1$=0.146($m$); $l_2$=0.253($m$); $l_{c1}$=0.055($m$); $l_{c2}$=0.149($m$); $K_b$=0.0864(V/$(rad/s)$); $K_t$=0.0864($Nm/A$);

$R_m$=3.7($\Omega$); $J_m$=0.000427($Kg.m^2$);

$C_m$=0.000415($Nm/(rad/s)$);

g=9.80665($m/s^2$); $I_1$=0.0189($Kg.m^2$); $I_2$=0.000409($Kg.m^2$);

Explication of Fig. 17:

(1): Link 2

(2): Link 1

(3): Encoder calculating angle of link 2

(4): Motor DC and Encoder calculating angle of link 1

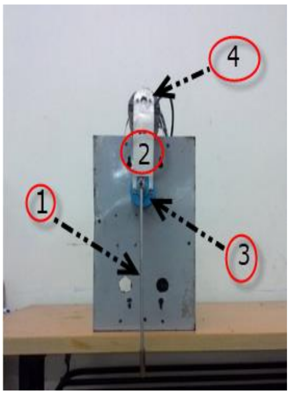


***Fig. 17.*** *Experiment model*

***Fig. 18.*** *Pendubot at Top Position*

The PC parameters are selected as follow:

$K_{p1}$= -236.7253; $K_{d1}$= -24.7728;

$K_{p2}$= -239.1236; $K_{d2}$= -18.1740;

and Fuzzy controller parameters are:

$K_1$=20; $K_2$=40; $K_3$=0.1; $K_4$=0.4;

$K_5$=0.9; $K_6$=0.125; $K_7$=0.02; $K_8$=0.5.

Parameters of PID controllers are selected by genetic algorithm. GA programs were written by file.m in Matlab software (following sample programs [7]), then run to search parameters. This parameters was simulated on computer (Matlab/Simulink).

In order to select the most suitable parameters for model. Authors continue testing these parameters with real model. And authors also practice a work called “error test”. In this case, “error test” means increase or decrease parameters that we searched by GA programs until finishing searching most suitable parameters. Finally, parameters of PC, FLC were selected by “error test” for simulation results and experiment results are well.

### 4.1 Control results of link 1

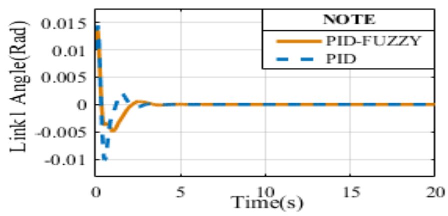


***Fig. 19.*** *Simulation result of $q_1$ (rad)*

Following Fig. 19, considering $q_1$, settling time of system under PFC is faster than PC at about 2.5s. When system starts, while $q_1$ of PC drops down to -0.01(rad), $q_1$ of PFC just drops down about -0.003(rad).

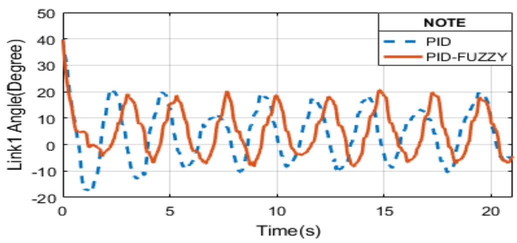


***Fig. 20.*** *Experimental result of q1 (degree)*

Following Fig. 20, it is obvious that we have from simulation as the same as experiment results. When system starts, $q_1$ of PC drops down to $-18^0$, q1 of PFC just drops down about $-5^0$.

## 4.2 Control results of link 2

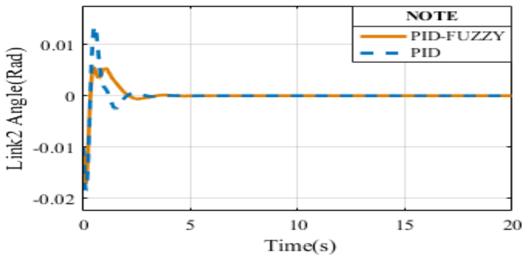


***Fig. 21.*** *Simulation result of $q_2$ (rad)*

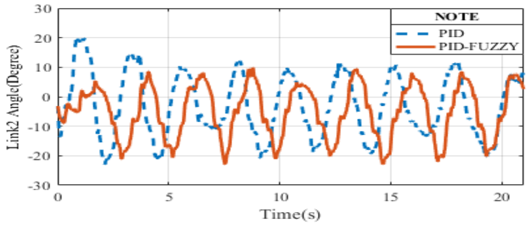


***Fig. 22.*** *Experimental result of $q_2$ (degree)*

Following Fig. 21, considering $q_2$, settling time of system under PFC is faster than PC about 2s. When system starts, $q_2$ of PC increases up to about 0.014(rad), $q_2$ of PFC just increases up about 0.005 (rad).

Following Fig. 22, result from simulation is as the same as experiment result. When system begins, $q_2$ of PC increases up to $20^0$, $q_2$ of PFC just increases up about $2^0$.

## 5. CONCLUSION

In the paper, pendubot system works stably around balancing top-position. Results were proved through simulation and experiment. It is obvious that PFC gives better responses than PC. However, there is still vibration under real model, comparing to simulation, due to the uncertainty of real model.

## REFERENCES

[1] Nguyen Van Dong Hai, *Input-output linearization for Cart and Pole system*, Master Thesis of Ho Chi Minh city of Technology, 2011.

[2] Xiao Qing Ma, *Fuzzy Control for an under-actuated Robotic Manipirlator: Pendubot*, A Thesis in The Department of Mechanical Engineering, August 2001.

[3] Li Chun Lai, Yu Yi Fu, Chia-Nan Ko, *MQPSO Algorithm Based Fuzzy PID Control for a Pendubot System*, International Conference on Artificial Life and Robotics (ICAROB 2017), pp. 19-22, Seagaia Convention Center, Miyazaki, Japan, 2017.

[4] Cao Van Kien, Ngoc Son Nguyen, Ho Pham Huy Anh, *Swing Up and Balancing Implementation for the Pendubot Using Advanced Sliding Mode Control*, Proceeding of International Conference on Recent Advances in Electrical Engineering and Related Sciences pp 411-425, Springer, January,2015.

[5] Martin Gulan, Michal Salaj, Boris Rohal, Ilkiv, *Achieving an Equilibrium Position of Pendubot via Swing-up and Stabilizing Model Predictive Control*, Journal of Electrical Engineering, Vol. 65, No. 6, pp. 356–363, 2014

[6] Cao Van Kien, Ho Pham Huy Anh, Pendubot *Balancing System Using Hybrid Fuzzy PID Control Optimized by Differential Evolution Algorithm*, Vietnam conference on Mechatronics (VCM), November, 2016.

[7] Huỳnh Thái Hoàng, "*Hệ Thống Điều Khiển Thông Minh*", Ho Chi Minh City University of Technology, 2014.

**Corresponding author:**
Tran Hoang Chinh
Ho Chi Minh city University of Technology and Education, Vietnam
Email: 13142021@student.hcmute.edu.vn