

XÂY DỰNG BỘ ĐIỀU KHIỂN PID-NEURON CHO HỆ CON LẮC NGƯỢC QUAY PID-NEURON CONTROLLER DESIGN FOR ROTTARY INVERTED PENDULUM SYSTEM

Nguyễn Văn Đông Hải, Ngô Văn Thuyên.

Trường Đại Học Sư Phạm Kỹ Thuật TP.HCM

TÓM TẮT

Con lắc ngược quay là hệ thống phi tuyến và là mô hình phổ biến trong các ứng dụng kiểm chứng kỹ thuật điều khiển. Khi áp dụng kỹ thuật PID cho hệ con lắc ngược quay, các thông số K_p , K_i , K_d được chọn lựa bằng phương pháp thực nghiệm. Tuy nhiên, bộ thông số đó có thể làm hệ thống ổn định nhưng còn dao động lớn. Bài báo này ứng dụng mạng thần kinh nhân tạo để chỉnh định trực tuyến thông số PID (PID-neuron). Các thông số trên được thay đổi dần để đảm bảo hệ thống ổn định và ít dao động. Mô phỏng được thực hiện trong mô trường Matlab/Simulink cho thấy thông số K_p , K_i , K_d đã thay đổi trực tuyến dần theo hướng làm hệ thống ổn định. Thuật toán điều khiển được xây dựng trên môi trường Matlab/Simulink thông qua card thu thập dữ liệu DSP-F2812. Kết quả thực nghiệm cho thấy bộ PID-neuron cho dao động của góc con lắc nhỏ hơn khi sử dụng bộ PID tĩnh.

ABSTRACT

Rotary inverted pendulum is a nonlinear system and a popular model in testing the control algorithm. When using PID control for rotary inverted pendulum system, parameters K_p , K_i , K_d are selected by experimental methods. Nevertheless, those parameters make the system stable but with large swing. This paper applies artificial neural network to calibrate online PID parameters (PID-neuron). These parameters are changed progressively to ensure the stability of the system and smaller swing. The simulation performed in Matlab/Simulink environment shows that parameters K_p , K_i , K_d which are adjusted online make the system stable. The control algorithm is tested in the real rotary inverted pendulum in Matlab/Simulink environment with data acquisition card DSP-F2812. Experimental results show that the proposed PID-neuron controller gives less swing for the pendulum angle than the static PID controller does.

I. GIỚI THIỆU

Trong các hệ thống phi tuyến, con lắc ngược quay dễ chế tạo và có các đặc trưng phi tuyến cơ bản nhất. Vì thế, hệ thống trên là đối tượng thông dụng cho các thí nghiệm nhận dạng và điều khiển.

Nhiều thuật toán điều khiển khác nhau đã được ứng dụng trên mô hình con lắc ngược quay như thuật toán tỉ tích phân tỉ lệ (PID), đặt cực (pole - placement), tối ưu (LQR), điều khiển mờ (Fuzzy), dùng mạng neuron điều khiển... và đạt thành công đáng kể.

Vũ Chấn Hưng [1] đã đề ra cách thức chọn cực phù hợp và mô phỏng thành công bộ

điều khiển đặt cực để điều khiển hệ thống con lắc ngược quay. Hăng Quanser [2] cũng đã chế tạo hệ thống con lắc ngược quay và áp dụng điều khiển LQR để phục vụ huấn luyện lý thuyết điều khiển. Những kỹ thuật điều khiển trên đòi hỏi người thiết kế phải biết trước mô hình toán học của đối tượng, đồng thời phải có kinh nghiệm chọn lựa thông số phù hợp để điều khiển ổn định.

Jia-Jun Wang [3] đã kết hợp hai bộ PID điều khiển một vào-một ra để được một bộ điều khiển thỏa hiệp, điều khiển được cho hệ một vào-nhiều ra và áp dụng thành công cho hệ thống con lắc ngược quay. Kỹ thuật trên không yêu cầu biết trước mô hình toán học hệ

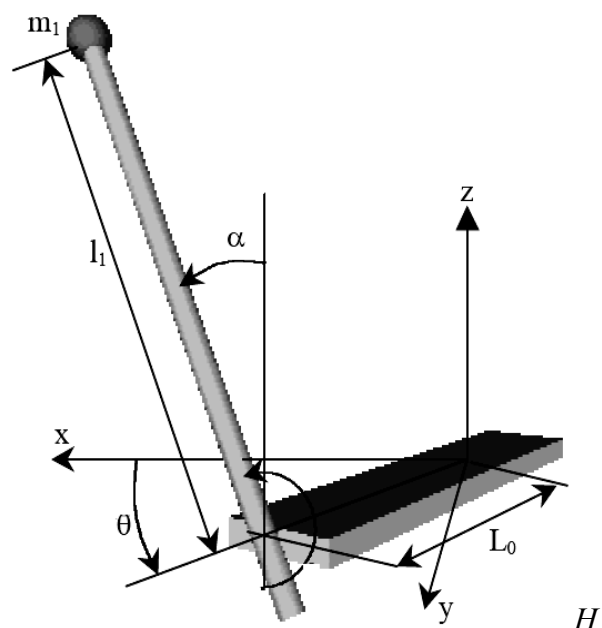
thống. Người thiết kế bộ điều khiển cần có kinh nghiệm và thời gian thử sai để có được bộ điều khiển tốt nhất. Tuy nhiên, hệ thống chỉ hoạt động tốt quanh điểm làm việc tĩnh.

Từ Diệp Công Thành [4] đã đề cập phương pháp dùng bộ điều khiển neuron-PID. Việc kết hợp bộ điều khiển PID với cấu trúc neuron cho phép thông số PID thay đổi dần để đạt giá trị tối ưu nhất. Tuy nhiên, trong bài báo, đối tượng được sử dụng là đối tượng tuyến tính, một vào – một ra và kết quả thành công chỉ dừng ở mô phỏng.

Trong bài báo này, các tác giả đề nghị sử dụng bộ PID-neuron kết hợp với PID hai biến tĩnh ở [3] để điều khiển hệ thống con lắc ngược quay. Bộ điều khiển PID-neuron được áp dụng cho hệ một vào-nhiều ra và không cần biết mô hình toán học của hệ thống. Khả năng tự thay đổi trực tuyến của mạng thần kinh chỉnh định dần thông số K_p , K_I , K_D theo hướng tối ưu hóa. Các mục tiếp theo của bài báo được trình bày theo thứ tự sau: Mục II trình bày mô hình toán học hệ con lắc ngược quay và sơ đồ điều khiển dùng hai bộ PID tĩnh. Bộ điều khiển PID-neuron cho hệ con lắc ngược quay được thiết kế trong Mục III. Mục IV và V trình bày kết quả mô phỏng và thực nghiệm của thuật toán được đề xuất. Phần kết luận được trình bày trong Mục VI.

II. MÔ HÌNH TOÁN HỌC CỦA HỆ THỐNG CON LẮC NGƯỢC QUAY

Hệ phương trình mô tả đặc tính động phi tuyến của hệ thống con lắc ngược quay (Hình 1)



ình 1. Mô hình tính toán hệ thống con lắc ngược quay

Hệ phương trình toán học được mô tả theo hệ phương trình

$$A \begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} + B \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} + C = Du \quad (1)$$

$$A = \begin{bmatrix} J_0 + m_1 L_0^2 + m_1 l_1^2 \sin^2 \alpha & -m_1 L_0 l_1 \cos \alpha \\ m_1 L_0 l_1 \cos \theta & J_1 + m_1 l_1^2 \end{bmatrix}$$

$$B = \begin{bmatrix} C_0 + \frac{K_i K_b}{R_a} + \frac{1}{2} m_1 l_1^2 \dot{\alpha} \sin 2\alpha & m_1 L_0 l_1 \dot{\alpha} \sin \alpha + \frac{1}{2} m_1 l_1^2 \dot{\theta} \sin 2\alpha \\ -\frac{1}{2} m_1 l_1^2 \dot{\theta} \sin 2\alpha & C_1 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 \\ -m_1 g l_1 \sin \alpha \end{bmatrix}$$

$$D = \begin{bmatrix} \frac{K_i K_b}{R_a} \\ 0 \end{bmatrix}$$

Trong đó:

θ : góc quay cánh tay (rad)

K_i : hằng số momen quay động cơ (Nm/A)

K_b : hằng số sức điện động cảm ứng motor (Vs/rad)

C_1 : hệ số ma sát trượt phần con lắc (kgm^2/s)

K_u : hệ số khuếch đại tín hiệu điều khiển motor (V/count)

R_a : điện trở cuộn rotor của động cơ (Ω)

u : tín hiệu điều khiển động cơ (V)

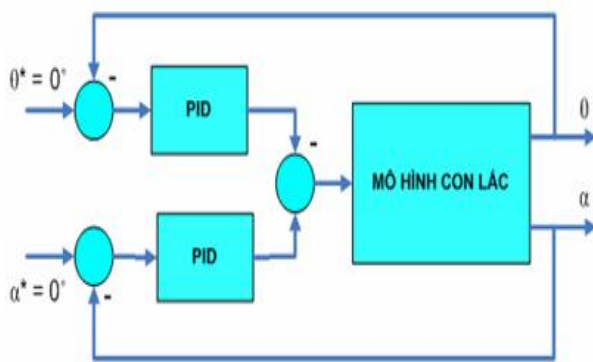
g : gia tốc trọng trường (m/s^2)

α : góc chuyển động của con lắc (rad)

J_0 : momen quán tính phần cánh tay (kgm^2)

J_1 : momen quán tính phần con lắc (kgm^2)

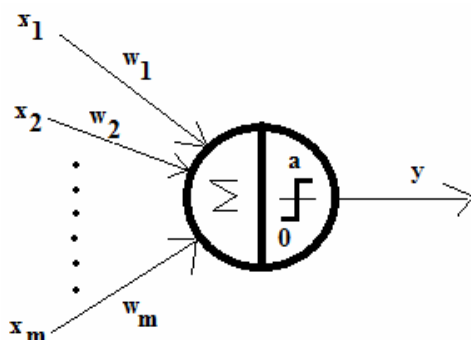
Theo [3], để điều khiển hệ thống một vào-nhiều ra như hệ con lắc ngược quay, tác giả trên đã đề xuất điều khiển PID hai biến tính với sơ đồ khối điều khiển con lắc ổn định như Hình 2. Hai bộ điều khiển PID lần lượt điều khiển góc con lắc α và góc cánh tay θ . Tín hiệu điều khiển là kết quả thỏa hiệp của hai bộ điều khiển trên.



Hình 2: Sơ đồ điều khiển PID hai biến tính điều khiển hệ con lắc ngược quay

III. XÂY DỰNG BỘ ĐIỀU KHIỂN PID-NEURON CHO HỆ CON LẮC NGƯỢC

Mạng thần kinh là mô hình toán học đơn giản của bộ não người. Mạng thần kinh gồm các tế bào thần kinh kết nối với nhau bởi các liên kết. Mỗi liên kết kèm theo một trọng số, đặc trưng cho tính kích thích hay ức chế giữa các tế bào thần kinh [5]. Hình 3 mô tả sơ đồ một nơ ron nhân tạo, trong đó x_1, x_2, \dots, x_m là các tín hiệu vào tế bào thần kinh và w_1, w_2, \dots, w_m là các trọng số của tế bào thần kinh.



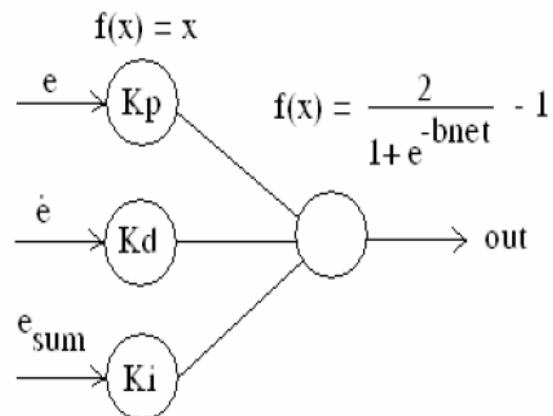
Hình 3. Sơ đồ khối tế bào thần kinh nhân tạo

Quá trình xử lý thông tin của mạng thần kinh chia làm hai phần: xử lý ngõ vào (hàm

tổng hợp) và xử lý ngõ ra (hàm tác động). Khi mạng neuron được huấn luyện, các trọng số w_1, w_2, \dots, w_m sẽ lần lượt thay đổi để đạt được giá trị tối ưu nhất. Có ba phương pháp chính huấn luyện mạng nơ ron: huấn luyện theo nhóm (batch-training), huấn luyện trực tuyến (online-training), huấn luyện ngẫu nhiên (stochastic-training) [6].

Đối tượng con lắc ngược có thời gian tác động nhanh. Phương pháp chọn ngẫu nhiên trọng số có thể làm hệ thống con lắc mất thăng bằng và không ổn định, cách thức huấn luyện ngẫu nhiên là không phù hợp. Mặt khác, trọng số mạng nơ ron phải được cập nhật liên tục với số ngõ vào trạng thái x ban đầu là xác định. Cách thức huấn luyện theo nhóm là không phù hợp. Từ đó, nhóm tác giả chọn cách thức huấn luyện trực tuyến để huấn luyện trọng số điều khiển cho hệ thống con lắc ngược quay.

Trong bài báo này, nhóm tác giả dùng thuật toán điều khiển trực tiếp. Các thông số K_p, K_I, K_D tương ứng với trọng số của mạng nơ ron gồm một nơ ron. Cấu trúc mạng nơ ron tự điều chỉnh được thể hiện trong Hình 4.



Hình 4. Bộ điều khiển PID-neuron

Hàm tổng hợp ngõ vào:

$$u = K_p e + K_I e_{sum} + K_D \dot{e} \quad (2)$$

Hàm tác động ở ngõ ra được chọn là hàm S lượng cực:

$$out = h(net) = \frac{1 - e^{-bnet}}{1 + e^{-bnet}} \quad (3)$$

Như vậy, ngõ ra của nơ ron là một hàm tangsig khả vì phù hợp cho việc thực hiện các phép tính đạo hàm riêng.

Hàm mục tiêu được chọn:

$$J = \frac{1}{2}(y_d - y)^2 \quad (4)$$

Với y_d là giá trị ngõ ra mong muốn và y là giá trị ngõ ra thực tế.

Trọng số được cập nhật theo phương pháp gradient (steepest descent), nghĩa là:

$$\theta(k+1) = \theta(k) - \eta \nabla J \quad (5)$$

Với $\eta > 0$ là hằng số học, ảnh hưởng đến tốc độ học và tính hội tụ của trọng số mạng nơ ron.

Các thông số của bộ điều khiển PID thay đổi theo quy luật:

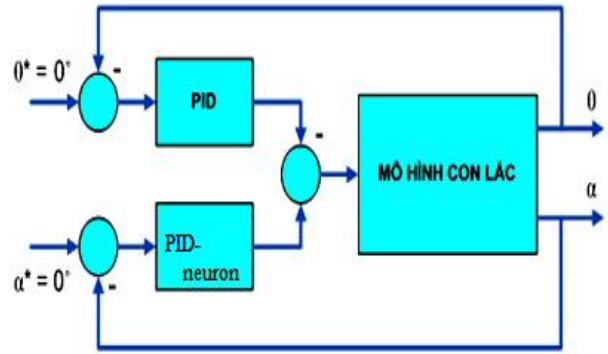
$$\begin{cases} K_p = K_p - \eta \frac{\partial E}{\partial K_p} \\ K_I = K_I - \eta \frac{\partial E}{\partial K_I} \\ K_D = K_D - \eta \frac{\partial E}{\partial K_D} \end{cases} \quad (6)$$

Tiến hành phân tích đạo hàm riêng các biểu thức (6):

$$\begin{cases} \frac{\partial E}{\partial K_p} = \frac{\partial E}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial K_p} = (-e)f'(u)e \\ \frac{\partial E}{\partial K_I} = \frac{\partial E}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial K_I} = (-e)f'(u)e_{sum} \\ \frac{\partial E}{\partial K_D} = \frac{\partial E}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial K_D} = (-e)f'(u)\dot{e} \end{cases} \quad (7)$$

Qui luật thay đổi của các trọng số của bộ điều khiển PID-neuron:

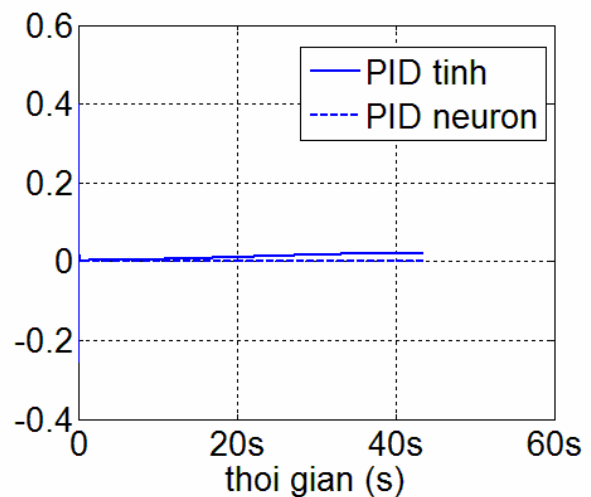
$$\begin{cases} K_p = K_p - \eta.e.f'(u).e \\ K_I = K_p - \eta.e.f'(u).e_{sum} \\ K_D = K_D - \eta.e.f'(u).\dot{e} \end{cases} \quad (8)$$



Hình 5. Sơ đồ điều khiển PID hai biến kết hợp PID-neuron điều khiển hệ con lắc ngược quay

IV. KẾT QUẢ MÔ PHỎNG

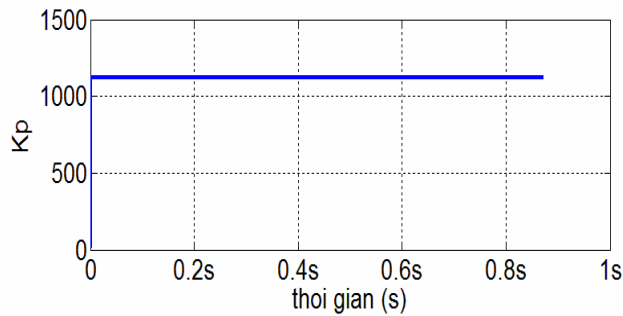
Thông số ban đầu của bộ điều khiển góc con lắc được chọn là $K_p = 10$, $K_I = 0.1$, $K_D = 1$. Thông số học η được chọn là 0.01. Hình 6 cho thấy so sánh đáp ứng ngõ ra của hệ thống con lắc ngược quay khi áp dụng bộ PID tĩnh (K_p , K_I , K_D không đổi trong quá trình hoạt động) và bộ PID có kết hợp cấu trúc neuron (K_p , K_I , K_D thay đổi dần theo hướng tối ưu). Góc lệch con lắc α lớn dần trong trường hợp áp dụng bộ điều khiển PID tĩnh, làm hệ thống mất ổn định. Trong trường hợp áp dụng bộ điều khiển PID-neuron, sau một khoảng thời gian, hệ thống vẫn ổn định với góc lệch con lắc bằng 0.



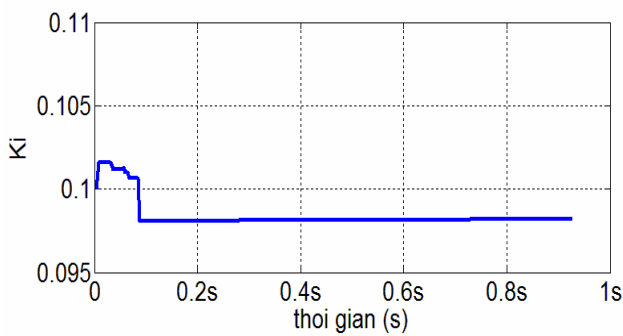
Hình 6. Đáp ứng ngõ ra hệ con lắc ngược của bộ điều khiển PID tĩnh và bộ PID-neuron

Quá trình thay đổi các thông số K_p , K_I , K_D của bộ điều khiển PID-neuron được

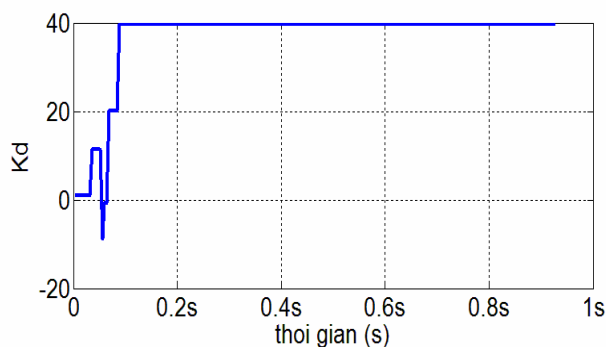
thể hiện lần lượt trong Hình 7, Hình 8 và Hình 9. Sau khoảng thời gian 0.2s, giá trị thông số PID được xác định với $K_p = 1122.5$, $K_I \approx 0.112$, $K_D = 39.6$ và không còn thay đổi đáng kể.



Hình 7. Quá trình thay đổi thông số K_p



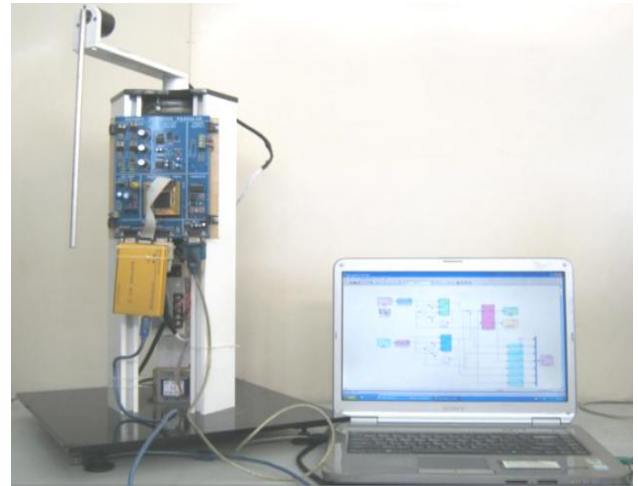
Hình 8. Quá trình thay đổi thông số K_i



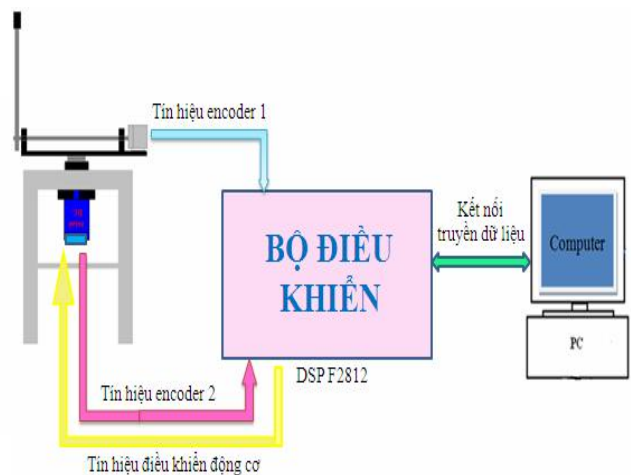
Hình 9. Quá trình thay đổi thông số K_d

V. KẾT QUẢ THỰC NGHIỆM

Mô hình hệ thống con lắc ngược quay giao tiếp với máy tính thông qua board DSP-2812 như Hình 10. Chương trình PID hai biến tĩnh và PID có kết hợp PID-neuron được thực hiện trong môi trường Matlab/Simulink. Sơ đồ khối điều khiển được thể hiện trong Hình 11.

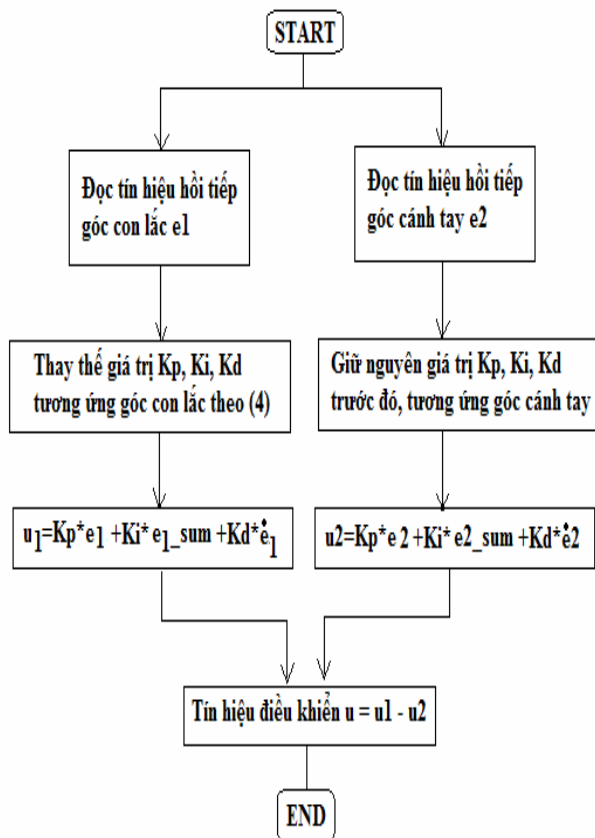


Hình 10. Hệ thống thực nghiệm



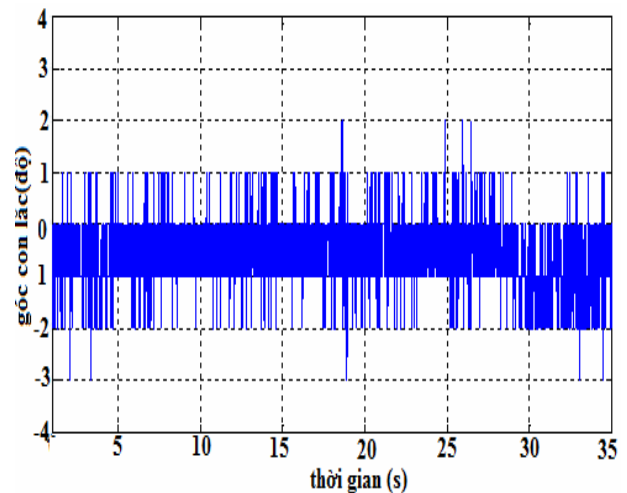
Hình 11. Sơ đồ điều khiển hệ thống con lắc ngược quay

Bộ điều khiển PID-neuron chỉ được áp dụng để điều khiển góc con lắc. Góc cánh tay được điều khiển bởi bộ bộ điều khiển PID tĩnh. Việc áp dụng PID-neuron cho cả góc cánh tay và góc con lắc ảnh hưởng sự thỏa hiệp của hai bộ PID-neuron. Từ đó, hệ thống mất ổn định. Do đó, bộ PID-neuron sẽ được áp dụng cho bộ điều khiển góc con lắc, PID tĩnh được áp dụng cho bộ điều khiển góc cánh tay, tín hiệu điều khiển cuối cùng là kết quả thỏa hiệp của hai bộ điều khiển trên. Lưu đồ giải thuật điều khiển PID hai biến kết hợp cấu trúc nơ ron thể hiện trong Hình 12.

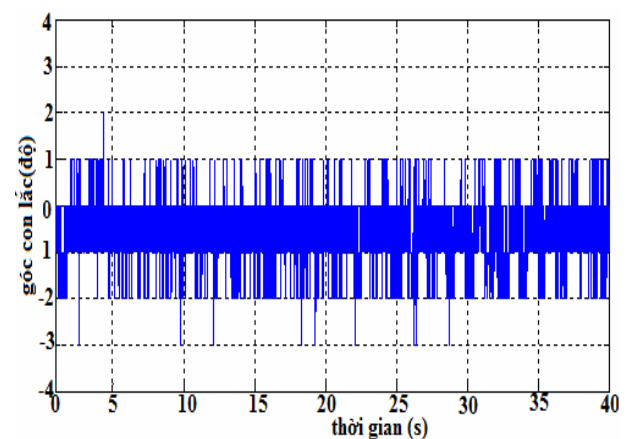


Hình 12. Lưu đồ giải thuật điều khiển PID hai biến kết hợp cấu trúc nơ ron

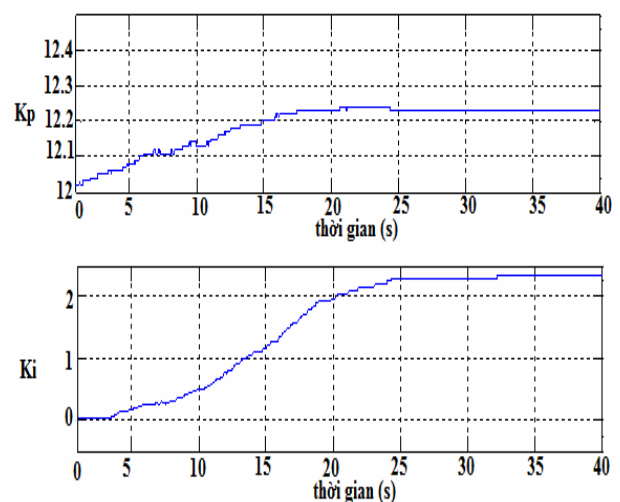
Trong Hình 13, hệ thống được điều khiển bởi bộ PID hai biến tĩnh có giá trị góc lệch α dao động trong khoảng $[-3^0, 2^0]$. Khi hệ con lắc ngược quay được điều khiển bởi bộ điều khiển PID-neuron, dao động của con lắc đã giảm từ $[-3^0, 2^0]$ sang khoảng $[-2^0, 1^0]$ như Hình 14. Quá trình thay đổi thông số K_p , K_i , K_d được thể hiện trong hình 15. Giá trị thông số bộ điều khiển ban đầu PID được chọn ban đầu lần lượt là $K_p = 12$, $K_i = 0$, $K_d = 10$. Sau khoảng thời gian 35s, các thông số PID đã đạt giá trị ổn định $K_p = 12.22$, $K_i = 2.3$, $K_d = 10.41$

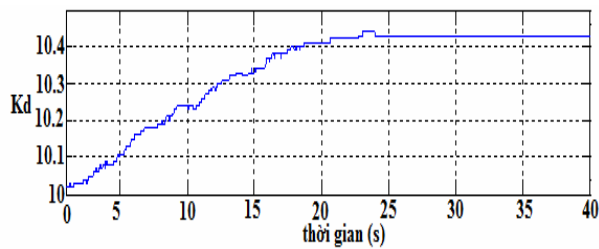


Hình 13. Đáp ứng thực của hệ thống đối với bộ điều khiển PID hai biến tĩnh



Hình 14. Đáp ứng thực của hệ thống đối với bộ điều khiển PID có kết hợp nơ ron





Hình 15. Quá trình thay đổi K_p , K_i , K_d

V. KẾT LUẬN

Bài báo đã trình bày cách thức thực hiện bộ điều khiển PID hai biến có kết hợp cấu trúc

nơ ron để chỉnh định giá trị K_p , K_i , K_d trực tuyến (PID-neuron) cho hệ thống con lắc ngược quay. Kết quả mô phỏng và thực nghiệm cho thấy việc tích hợp cấu trúc nơ ron để chỉnh định trực tuyến giá trị thông số điều khiển PID làm hệ thống dao động ít hơn, ổn định dần theo thời gian. Giá trị K_p , K_i , K_d cũng dần thay đổi theo hướng tối ưu để hệ thống đạt giá trị ổn định.

TÀI LIỆU THAM KHẢO

- [1]. Vũ Chấn Hưng, 2005, *Điều khiển hệ thống con lắc ngược quay*, tuyển tập các báo cáo khoa học: Hội nghị toàn quốc lần thứ VI về tự động hóa, số 0, trang 276-281.
- [2]. Karam, P & Levis, M & Apkarian, J, 2010, *Rotary Inverted Pendulum for Matlab/Simulink Software Users*, Quanser, Canada.
- [3]. Wang, Jia-Jun, 2011, *Simulation Studies of Inverted Pendulum based on PID Controller*, Simulation Modelling Practice and Theory, vol. 19, issue 1, pp. 440-449.
- [4]. Từ Diệp Công Thành, 2008, *Mô phỏng Bộ điều khiển neuron với luật học hệ số học thích nghi và phương pháp xung lượng*, Tạp chí phát triển KH&CN, tập 11, số 3, trang 69.
- [5]. Huỳnh Thái Hoàng (ed.), 2006, *Hệ thống điều khiển thông minh*, NXB ĐHQG TP HCM
- [6]. Yu, Hen Hu & Hwang, Jenq-Neng (eds.), 2002, *Hand Book of Neural Network Signal Proccessing*, CRC Press, USA.