

## Xây dựng bộ điều khiển thông minh bằng phương pháp máy học cho hệ bóng thanh trục giữa sử dụng công cụ trên MATLAB

### Designing intelligent controller through machine leaning method for mid-axis rod ball by toolbox on matlab

Nguyễn Minh Quân\*, Trần Gia Huân, Nguyễn Tất Thành, Ngô Phùng Huy, Trần Tiến Hoài, Đặng Đức Thành, Đỗ Nguyễn Nhất Huy, Nguyễn Ngọc Gia Huy, Nguyễn Tuấn Kiệt, Dương Trọng Khiêm

Trường Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh

\* 18151230@student.hcmute.edu.vn

Ngày nhận bài:

20/12/2023

Ngày chấp nhận đăng:

03/3/2024

**Keywords:** intelligent control, LQR control, fuzzy control, Neuron control, ANFIS, ball and beam.

#### ABSTRACT

Mid-axis rod ball is a classic nonlinear system often present in research laboratories and universities. Linearized LQR control can be applied on this model. However, manually building smart controllers with classical methods (specifically fuzzy controls - Fuzzy and Neuron) for this nonlinear system is very complicated and time-consuming. Therefore, the authors researched the tools "Anfis" and "Neuron Toolbox" on Matlab, allowing automatic construction of fuzzy controllers and learning Neurons according to an available map (LQR). Through simulation, the authors summarize the design process of fuzzy and neural maps through expert experiences. Besides, comparative surveys of the above conditions help creating an academic document for subsequent intelligent control designs.

#### TÓM TẮT

Bóng thanh trục giữa (BTTG) là hệ phi tuyến kinh điển thường có mặt trong các phòng nghiên cứu và các trường đại học. Điều khiển Linear Quadratic Regulator (LQR) sau khi được tuyến tính hóa có thể được áp dụng cho mô hình này. Tuy nhiên, việc xây dựng thủ công bộ điều khiển (BĐK) thông minh (cụ thể là điều khiển mờ - Fuzzy và Neuron) cho hệ phi tuyến này theo hướng thủ công bằng các phương pháp kinh điển thì rất phức tạp và tốn thời gian. Bởi vậy, nhóm tác giả thực hiện nghiên cứu công cụ "Anfis" và "Neuron Toolbox" trên Matlab, cho phép xây dựng tự động BĐK mờ và Neuron học theo một BĐK sẵn có (LQR). Thông qua mô phỏng, nhóm tác giả tổng kết quy trình thiết kế các BĐK mờ và Neuron thông qua kinh nghiệm chuyên gia. Bên cạnh đó, các khảo sát so sánh các BĐK trên giúp tạo ra một tài liệu có tính học thuật cho các thiết kế điều khiển thông minh (ĐKTM) sau đó.

**Từ khóa:** điều khiển thông minh, điều khiển LQR, điều khiển mờ, ANFIS, hệ bóng thanh.

### 1. Giới thiệu

Hệ BTTG là một hệ thống đã xuất hiện từ lâu trong điều khiển tự động, được xây dựng nhằm kiểm chứng tính chính xác của các giải thuật điều

khiển. Sở dĩ được gọi là kinh điển là bởi hệ này sở hữu nhiều yếu tố cần thiết cho việc nghiên cứu trong lĩnh vực điều khiển học như: có tính ứng dụng thực tế cao (áp dụng cân bằng máy bay khi

hạ cánh gặp phải các luồng khí đối lưu, ...), tính phức tạp của một hệ phi tuyến (BTTG là một hệ phi tuyến dạng một vào - nhiều ra (single input-multi output – SIMO), một ngõ vào nhiều ngõ ra), đơn giản để chế tạo mô hình (thuận tiện cho việc giảng dạy ở các trường học, là một đối tượng đơn giản để học viên mới bắt đầu nghiên cứu), ...

Với nhiều ưu điểm như vậy, hệ này được rất nhiều nhà nghiên cứu trên thế giới quan tâm nghiên cứu và làm thực nghiệm. Trong số các nghiên cứu đã thành công có thể kể đến công trình của (Đặng Ngọc Trung và cộng sự, 2014) với tên gọi “Bóng cân bằng trên thanh đỡ”. Đó là 1 hệ thống áp dụng phương pháp điều khiển Proportional Integral Derivative (PID) cổ điển, thu thập tín hiệu vị trí bóng bằng cảm biến điện trở được gắn trên thanh đỡ và cho tín hiệu điều khiển bằng cách sử dụng chip Digital Signal Processor (DSP) xử lý tín hiệu đầu vào. Tuy nhiên, với việc sử dụng BDK PID với các thông số cố định, nghiên cứu này được nhiều nhà phê bình đánh giá là không đủ linh hoạt với các điều kiện biến đổi của một đối tượng phi tuyến SIMO (Nguyễn Chí Ngôn và cộng sự, 2021). Ngoài ra, nhiều nghiên cứu khác cũng có thể kể đến như mô hình “Bóng trên thanh trượt” của (Hirsch, 1999), hay mô hình với tên gọi “Ball and beam” của (Quanser, 2023), ...

Nhiều nghiên cứu thành công dẫn đến việc phát triển một BDK PID hay LQR cho đối tượng phi tuyến SIMO như BTTG trở nên khả thi. Tuy nhiên, nhiều đối tượng phi tuyến phức tạp, không thể xây dựng mô hình toán học vẫn cần áp dụng các giải thuật điều khiển thông minh (ĐKTM) (điều khiển mờ, điều khiển Neuron, ...). Hay trong trường hợp đã thực hiện được BDK PID hoặc LQR cho hệ BTTG (chẳng hạn) và nhà nghiên cứu muốn thực hiện ĐKTM cho hệ thống của mình thì việc này là khá khó khăn khi thực hiện xây dựng thủ công.

Các nghiên cứu ĐKTM trước đây (Nguyễn Ngọc Anh và cộng sự, 2023), (Huỳnh Ngọc Thảo và cộng sự, 2023) trên đối tượng SIMO đều hướng đến việc chứng minh ĐKTM điều khiển tốt hệ thống thông qua kết quả cuối cùng. Tuy vậy, một bài báo trình bày quy trình từng bước để thực hiện một BDK thông minh bên cạnh việc khảo sát kết quả điều khiển là chưa có. Do đó, trong bài báo này, nhóm tác giả hướng đến việc

xây dựng một tài liệu tham khảo (TLTK) để mọi thành phần độc giả có thể dễ dàng tiếp cận công cụ Matlab để xây dựng nhanh BDK thông minh, áp dụng cho đối tượng điều khiển của mình bên cạnh đối tượng được giới thiệu chính trong bài báo này – hệ BTTG. Nhóm tác giả đề xuất một phương pháp khác để thiết kế BDK thông minh cho hệ thống đã có trước một BDK (PID, LQR, ...). Phương pháp này sử dụng các công cụ của phần mềm Matlab để xây học theo BDK sẵn có của hệ thống. Cụ thể, nhóm tác giả sử dụng công cụ “Anfis” giúp tự động xây dựng BDK mờ và công cụ “Neuron Toolbox” giúp tự động xây dựng BDK Neuron học theo BDK LQR đã được nhóm tác giả xây dựng cho BTTG.

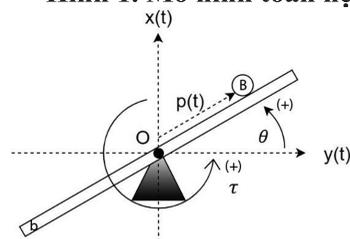
Trong bài báo, việc sử dụng BDK LQR đóng vai trò như một “chuyên gia” về điều khiển. Quá trình thu thập dữ liệu là quá trình thu thập dữ liệu hoạt động của chuyên gia. Việc xây dựng BDK Neuron hay Fuzzy thành công trong bài báo hỗ trợ tài liệu học thuật để các độc giả quan tâm có thể áp dụng các bước quy chuẩn mà tài liệu này mang lại để huấn luyện được các khối thông minh (Neuron – Fuzzy) của mình thông qua khối dữ liệu hoạt động của chuyên gia (mà trong thực tế, ta chỉ thu thập chứ không biết rõ là BDK loại gì. Điều này khác với bài báo là nhóm tác giả sử dụng LQR để giả lập “chuyên gia”). Bên cạnh đó, việc khảo sát so sánh giữa Fuzzy và Neuron trong việc học BDK – trong trường hợp này là tuyến tính LQR – là một tài liệu hỗ trợ cho các nhà thiết kế lựa chọn BDK về sau.

## 2. Nội dung

### 2.1. Điều khiển LQR cho hệ BTTG

#### 2.1.1. Phương trình toán học

Hình 1. Mô hình toán hệ BTTG



Hệ BTTG gồm:

- Một động cơ có encoder là cơ cấu động lực chính của hệ thống.
- Một thanh b đồng chất, có thể tự quay quanh tâm, ngay chính giữa thanh, nhờ vào moment  $\tau$  do động cơ sinh ra.

- Một quả bóng B đồng chất, có thể lăn tự do trên thanh b. Xét hệ thống có: O(0; 0): góc tọa độ, tâm quay, mốc thế năng tại O, p: vị trí của viên bi so với O (m),  $\theta$ : góc lệch của thanh b so với phương ngang (rad),  $\tau$ : moment do động cơ gây ra (Nm).

Thế năng của hệ được tính bằng tổng thế năng bóng B ( $P_B$ ) và thế năng thanh b ( $P_b$ ). Tuy nhiên, bởi thế năng thanh b,  $P_b = 0$  (do trọng tâm luôn trùng mốc thế năng), thế năng hệ được tính bởi công thức:

$$P = P_B = m_B g p \sin \theta \quad (1)$$

Trong đó: g = 9.81: gia tốc trọng trường,  $m_B$ : khối lượng bóng B (kg).

Động năng hệ thống được tính bằng tổng động năng  $T_B$  của bóng B và động năng  $T_b$  của thanh b:

$$T = T_B + T_b \quad (2)$$

$$T = \frac{1}{2} \dot{p}^2 \left( m_B + \frac{J_B}{R^2} \right) + \frac{1}{2} (J_b + m_B p^2) \dot{\theta}^2 \quad (3)$$

Trong đó:

$J_b = \frac{1}{12} m_b L_b^2$ : moment quán tính của bóng khi quay quanh tâm bóng,

$J_B = \frac{2}{5} m_B R^2$ : moment quán tính của bóng khi quay quanh tâm bóng,

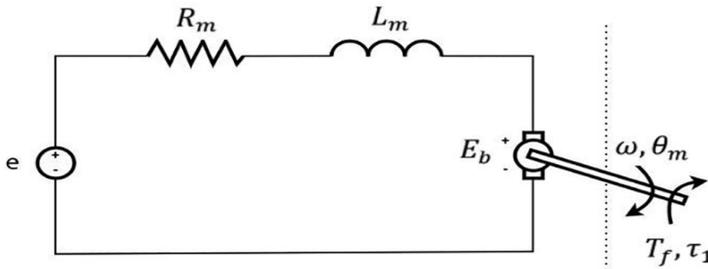
R: bán kính quả bóng B (m),

$m_b$ : khối lượng thanh b (Kg),

$L_b$ : chiều dài thanh b (m).

Bỏ qua ma sát, hệ này chỉ nhận tác động duy

**Hình 2. Cấu tạo động cơ**



Moment do động cơ sinh ra được tính bởi công thức:

$$\tau = K_t \frac{e - K_b \omega}{R_m} \quad (6)$$

Từ (5), (6), phương trình toán học hệ thống có thể viết lại về dạng theo điện áp đầu vào động cơ:

$$\ddot{p} = \frac{m_B p \dot{\theta}^2 - m_B g \sin \theta}{m_B + \frac{J_B}{R^2}} \quad (7)$$

$$\ddot{\theta} = \frac{K_t \frac{e - K_b \omega}{R_m} - 2m_B p \dot{p} \dot{\theta} - m_B g \cos \theta}{m_B p^2 + J_b} \quad (8)$$

### 2.1.2. Tuyến tính hóa hệ thống

nhất từ moment  $\tau$  của động cơ. Áp dụng phương pháp Euler – Lagrange, phương trình toán học hệ thống được tính có dạng:

$$\begin{cases} \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{p}} \right) - \frac{\partial L}{\partial p} = 0 \\ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = \tau \end{cases} \quad (4)$$

Trong đó,  $L = T - P$ : đại lượng Lagrange.

Thực hiện phép tính (4), ta được phương trình toán học của hệ thống theo moment  $\tau$

$$\begin{cases} \ddot{p} = \frac{m_B p \dot{\theta}^2 - m_B g \sin \theta}{m_B + \frac{J_B}{R^2}} \\ \ddot{\theta} = \frac{\tau - 2m_B p \dot{p} \dot{\theta} - m_B g \cos \theta}{m_B p^2 + J_b} \end{cases} \quad (5)$$

Tại (5), ta đã có được phương trình toán học của hệ BTTC theo moment  $\tau$  do động cơ sinh ra. Kết quả này cũng tương ứng đúng với kết quả được chỉ ra tại các công trình (Huỳnh Thái Hoàng, 2006), (Nguyễn Thị Phương Hà, 2002). Tuy nhiên, thực tế điều khiển lại thông qua việc điều chỉnh điện áp cấp cho động cơ, từ đó, làm thay đổi moment  $\tau$  cấp cho hệ. Do vậy, để phân tích sát với thực tế hơn, bài báo này sẽ tiếp tục phân tích phương trình toán học theo moment  $\tau$  đã có ở trên về phương trình toán học theo điện áp đầu vào e.

Xét cấu tạo động cơ cơ bản:

Trong đó:

$R_m$ : điện trở ( $\Omega$ ),

$K_t$ : hằng số moment (Nm/A),

$K_b$ : Hằng số Back EMF (V/rad/s),

e: điện áp đầu vào của động cơ (V),

$\omega$ : vận tốc góc của động cơ (rad/s).

Ta định nghĩa ma trận biến trạng thái là:

$$x = [x_1 \ x_2 \ x_3 \ x_4]^T = [p \ \dot{p} \ \theta \ \dot{\theta}]^T \quad (9)$$

Phương trình toán học hệ thống có thể viết thành:

$$x_1 = x_2 = f_1(x, e) \quad (10)$$

$$x_2 = \frac{m_B x_1 x_4^2 - m_B g \sin x_3}{m_B + \frac{J_B}{R^2}} = f_2(x, e) \quad (11)$$

$$x_3 = x_4 = f_3(x, e) \quad (12)$$

$$x_4 = \frac{K_t e - K_b K_t x_4 - 2m_B x_1 x_2 x_4 R_m}{(m_B p^2 + J_b) R_m} - \frac{R_m m_B g \cos x_3}{(m_B p^2 + J_b) R_m} = f_4(x, e) \quad (13)$$

Từ (10)-(13), phương trình hệ thống có dạng:

$$\dot{x} = f(x, e) \quad (14)$$

$$\text{Trong đó: } f = [f_1 \ f_2 \ f_3 \ f_4]^T$$

Chọn điểm làm việc tại vị trí thanh b cân bằng theo phương ngang, quả bóng cân bằng tại tâm thanh b, tương ứng với vị trí thỏa:

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \frac{\partial f_1}{\partial x_4} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \frac{\partial f_2}{\partial x_4} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \frac{\partial f_3}{\partial x_4} \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_2} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \end{bmatrix}_{x=0, e=0} \quad B = \begin{bmatrix} \frac{\partial f_1}{\partial e} \\ \frac{\partial f_2}{\partial e} \\ \frac{\partial f_3}{\partial e} \\ \frac{\partial f_4}{\partial e} \end{bmatrix}_{x=0, e=0}$$

$$p = \dot{p} = \theta = \dot{\theta} = e = 0 \quad (15)$$

Hệ phi tuyến BTTG có thể được xấp xỉ tuyến tính tại điểm làm việc về dạng:

$$\dot{x} = Ax + Be \quad (16)$$

Trong đó, A, B: các ma trận tuyến tính được tính như sau:

Các thông số của mô hình thực được tham khảo từ mô hình thực nghiệm ở tài liệu (Huỳnh Ngọc Thảo và cộng sự, 2023) như sau:

**Bảng 1. Thông số mô hình thực tế**

STT	Ký hiệu	Ý nghĩa	Giá trị thực	Đơn vị
1	$K_t$	Hằng số moment xoắn của động cơ	0.06494	N
2	$K_b$	Hằng số Back EMF của động cơ	0.06494	V/rad/s
3	$R_m$	Điện trở của động cơ	6.83572	$\Omega$
4	$L_m$	Hệ số điện kháng của động cơ	0.25086	H
5	$m_B$	Khối lượng bóng B	0.06	Kg
6	$m_b$	Khối lượng thanh b	0.38	Kg
7	$R$	Bán kính bóng B	0.01	m
8	$L_b$	Chiều dài thanh b	0.35	m

Với các giá trị thực của mô hình, tại điểm làm việc  $x = [0 \ 0 \ 0 \ 0]^T$ ,  $e = 0$ , có thể tính được kết quả của matrix A, B.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -7.01 & 0 \\ 0 & 0 & 0 & 1 \\ -151.73 & 0 & 0 & -0.16 \end{bmatrix}; B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 2.4490 \end{bmatrix} \quad (17)$$

### 2.1.3. ĐK LQR

#### Hình 3. Cấu trúc BDK LQR.



BDK tối ưu, viết tắt là điều khiển LQR có thể điều khiển hệ thống nhờ vào tín hiệu điều khiển:

$$u = -Kx \quad (18)$$

Trong đó:

$K = R^{-1}B^T P$ : ma trận tối ưu, xác định bằng cách giải phương Riccati hoặc sử dụng Matlab.

Chất lượng BDK LQR được đánh giá qua hàm mục tiêu J.

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (19)$$

Trong đó: Q: ma trận dương hoặc bán dương, R: ma trận dương.

Chọn

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, R = 0.01 \quad (20)$$

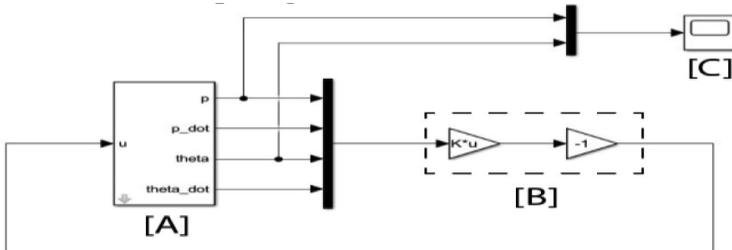
Sử dụng Matlab, ma trận K được xác định bằng cách thực hiện lệnh:

$$K = lqr(A, B, Q, R) \quad (21)$$

$$K = [-179.60 \quad -59.22 \quad 101.46 \quad 13.46] \quad (22)$$

### 2.1.4. Mô phỏng điều khiển LQR

Hình 4. Điều khiển LQR cho BTTG



Trong Hình 4:

[A]: hệ BTTG giữa với 1 tín hiệu vào  $u$  (điện áp động cơ) và 4 tín hiệu ra  $(p, \dot{p}, \theta, \dot{\theta})$ ,

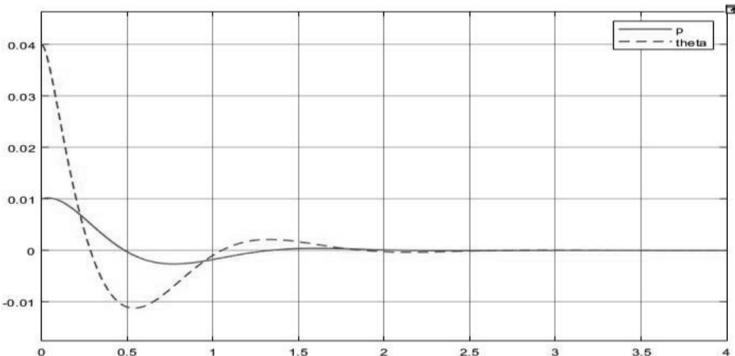
[B]: BDK LQR,

[C]: đồng hồ để đo vị trí bóng và góc lệch thanh theo thời gian.

Kết quả cân bằng bóng tại vị trí cân bằng O, với các điều kiện ban đầu.

$p = 0.01 \text{ m}$ ;  $\dot{p} = 0.01 \text{ m/s}$ ;  $\theta = 0.04 \text{ rad}$ ;  $\dot{\theta} = -0.02 \text{ rad/s}$ .

Hình 5. Kết quả ổn định vị trí bóng và góc lệch thanh



Trong Hình 5, trục tung là thời gian (s), trục hoành là giá trị đại số, ứng với từng đại lượng sẽ có đơn vị tương ứng. Ứng với vị trí bóng ( $p$ ) có đơn vị cm, ứng với góc lệch thanh ( $\theta$ ) có đơn vị là rad.

Để chi tiết, ta sẽ xem xét các tiêu chí ổn định là sai số xác lập ( $e_{xl}$ ), thời gian xác lập ( $t_{xl}$ ) và độ vọt lố (POT).

Bảng 2. Bảng đánh giá các tiêu chí ổn định của hệ BTTG được điều khiển bằng LQR

	$e_{xl}$	$t_{xl}$ (s)	POT
$p$	0 (m)	2	0.0035 (m)
$\theta$	0 (rad)	2.5	0.012 (rad)

Dựa trên kết quả mô phỏng, ta thấy LQR đã ổn định được BTTG về vị trí cân bằng O.

## 2.2. Dùng “Anfis” xây dựng BDK mờ từ BDK LQR

### 2.2.1. Giới thiệu công cụ “Anfis”

Các bước xây dựng 1 BDK mờ hay tiếng anh là “Fuzzy Controller”

- Xác định ngõ vào, ngõ ra.
- Mờ hóa, xác định các hàm liên thuộc.
- Xây dựng luật mờ.
- Giải mờ để có được tính hiệu điều khiển.

Việc 1 BDK mờ có hiệu quả hay không phụ thuộc rất nhiều vào kinh nghiệm điều khiển của chuyên gia để từ đó đưa ra luật mờ phù hợp.

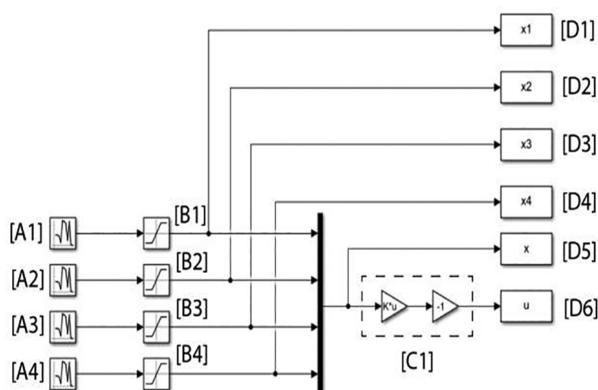
Với Matlab việc xây dựng BDK mờ trở nên dễ dàng hơn với công cụ “Fuzzy Logic Designer”. Tuy nhiên, công cụ này cần chúng ta tự xây dựng 1 luật mờ và điều này là khá khó khăn nếu không có kinh nghiệm điều khiển.

Trong phần này, nhóm tác giả xin giới thiệu 1 công cụ cho phép xây dựng 1 BDK mờ dựa trên dữ liệu kết quả điều khiển đã có. Công cụ này có tên “Anfis”, thuộc bộ công cụ “Deep Learning” của Matlab. Nhóm tác giả sẽ thực hiện xây dựng BDK mờ học theo BDK LQR vừa được xây dựng ở trên, để điều khiển cân bằng cho hệ BTTG.

## 2.2.2 Xây dựng BDK mờ dùng công cụ “Anfis”

**Bước 1:** Thu thập dữ liệu điều khiển LQR làm cơ sở xây dựng luật mờ cho BDK mờ.

**Hình 6. Thu thập dữ liệu điều khiển của LQR.**

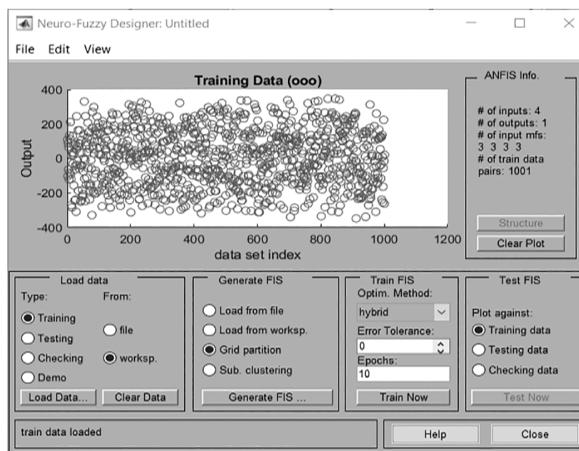


**Bước 2:** Gán các giá trị thu thập được vào ma trận và lưu vào không gian làm việc bằng lệnh.

$$u = [x_1 \ x_2 \ x_3 \ x_4 \ u]$$

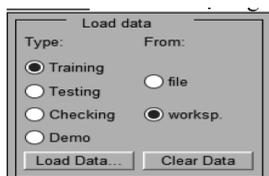
**Bước 3:** Khởi chạy “Anfis” bằng lệnh “Anfisedit”.

**Hình 7. Công cụ “Anfis”.**

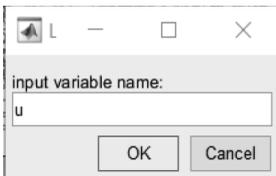


**Bước 4:** Load dữ liệu nguồn và cấu hình cho BDK mờ.

**Hình 8. Chọn nguồn cung cấp dữ liệu.**



**Hình 9. Nhập biến chứa dữ liệu thu thập.**



Để “Anfis” load dữ liệu từ dữ liệu đã thu thập, trong bảng “Load data” ta thực hiện chọn “worksp”.

Tiếp đến, ấn “Load data” và nhập “u” - tức là biến chứa giá trị thu thập vừa tạo ở bước 2.

Trong Hình 6:

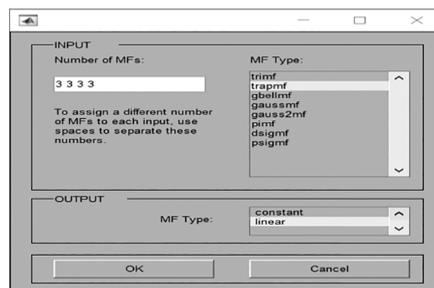
[A1], [A2], [A3], [A4]: các khối “Random”, với mục đích giả lập giá trị 4 thông số ngõ vào của khối LQR ( $p, \dot{p}, \theta, \dot{\theta}$ ),

[B1], [B2], [B3], [B4]: các khối “Saturation”, với mục đích giới hạn giá trị 4 tín hiệu ngõ vào trong khoảng [-1; 1],

[C1]: BDK LQR,

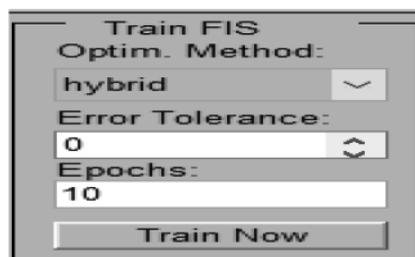
[D1], [D2], [D3], [D4], [D5], [D6]: các khối “To workspace”, với mục đích thu thập lần lượt các giá trị của  $p, \dot{p}, \theta, \dot{\theta}$ , tổng tín hiệu đầu vào của bộ LQR và tín hiệu đầu ra của bộ LQR.

**Hình 10. Xác định hàm ngõ vào, ngõ ra**



Xác định hàm liên thuộc ngõ vào, ngõ ra bằng cách ấn “Generate FIS”. Một bảng mới hiện ra, ta tiến hành định cấu hình cho bộ Fuzzy. Như Hình 10, ta đã xác định rằng có 4 ngõ vào và 1 ngõ ra. Số hàm liên thuộc của mỗi ngõ vào là 3. Hàm liên thuộc của ngõ vào có dạng trapmf, ngõ ra có dạng linear.

**Hình 11. Nhập Epochs mong muốn**



Cuối cùng, ta nhập số Epochs muốn học, số lượng Epochs càng lớn thì thời gian hoàn thành càng lâu và bộ Fuzzy càng sát bộ LQR đã có.

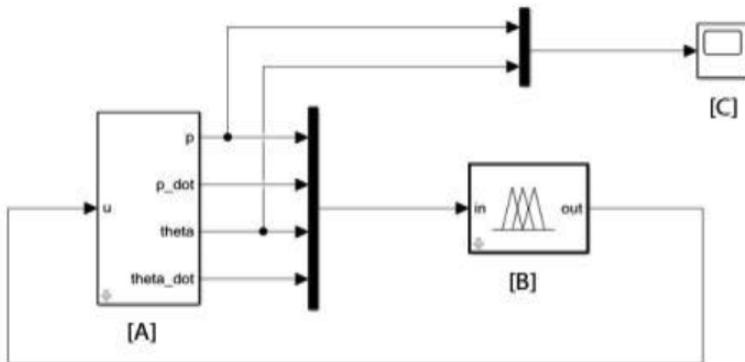
**Bước 5:** Sau khi hoàn thành, chọn “File”, chọn “Export”, chọn “To File”. Nhập tên và chọn đường dẫn cho bộ Fuzzy vừa tạo.

Tới đây, việc xây dựng BDK Fuzzy học theo LQR cho BTTG đã hoàn tất. Việc còn lại là

kiểm tra khả năng điều khiển của BDK Fuzzy vừa tạo.

### 2.2.3. Mô phỏng Fuzzy điều khiển hệ BTTG

Hình 12. Mô hình mô phỏng BDK Fuzz



Trong Hình 12:

[A]: hệ BTTG 1 tín hiệu vào  $u$  (điện áp động cơ) và 4 tín hiệu ra ( $p, \dot{p}, \theta, \dot{\theta}$ ),

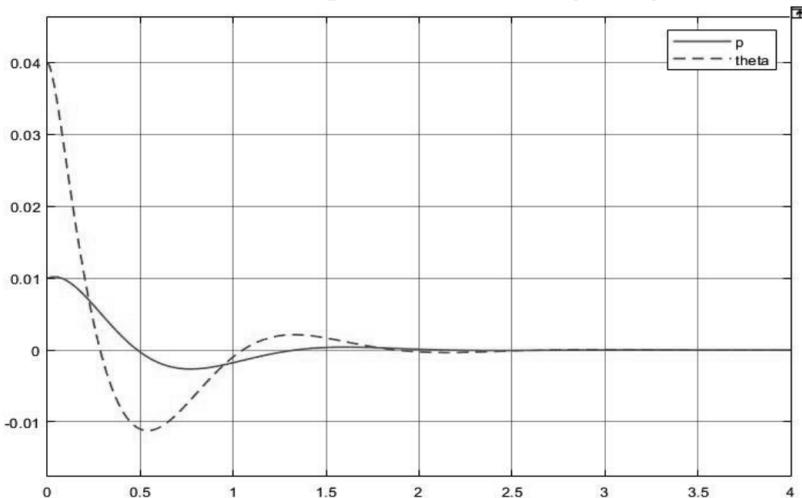
[B]: BDK Fuzzy,

[C]: đồng hồ “Scope” để đo tín hiệu vị trí bóng và góc lệch thanh theo thời gian.

Kết quả cân bằng bóng tại vị trí cân bằng 0, với các điều kiện ban đầu

$$p = 0.01 \text{ m}; \dot{p} = 0.01 \text{ m/s}; \theta = 0.04 \text{ rad}; \dot{\theta} = -0.02 \text{ rad/s}.$$

Hình 13. Kết quả ổn định vị trí bóng B và góc lệch thanh b của BDK Fuzzy



Trong Hình 13, trục tung là thời gian (s), trục hoành là giá trị đại số, ứng với từng đại lượng sẽ có đơn vị tương ứng. Ứng với vị trí bóng ( $p$ ) có đơn vị cm, ứng với góc lệch thanh ( $\theta$ ) có đơn vị là rad.

Để chi tiết, ta xem xét các tiêu chí ổn định là sai số xác lập ( $e_{xl}$ ), thời gian xác lập ( $t_{xl}$ ) và độ vọt lố (POT).

Bảng 3. Bảng đánh giá các tiêu chí ổn định của hệ BTTG được điều khiển bằng Fuzzy

	$e_{xl}$	$t_{xl}$ (s)	POT
$p$	0	2	0.0035
$\theta$	0	2.5	0.012

Kết quả mô phỏng cho thấy bộ Fuzzy học theo bộ LQR đã đáp ứng yêu cầu điều khiển, hệ thống đã ổn định tại vị trí cân bằng.

## 2.3. Dùng Neuron xây dựng BDK thông minh từ BDK LQR

### 2.3.1. Giới thiệu công cụ Neuron

Giải thuật Neuron là một phương pháp điều khiển thuộc trường phái ĐKTM. Lý thuyết về điều khiển Neuron dựa trên ý tưởng về các

tế bào thần kinh của con người, từ đó xây dựng một BDK linh hoạt như cách mà mạng thần kinh xử lý tín hiệu.

Việc thiết kế 1 BDK Neuron có thể điều khiển được hệ thống theo hướng thủ công bằng các phương pháp kinh điển thì rất phức tạp, đòi hỏi nhiều kiến thức và tốn nhiều thời gian.

Tuy nhiên, sự ra đời của công cụ “Neuron Toolbox” – một công cụ thuộc bộ công cụ “Deep Learning” của phần mềm Matlab, đã giúp cho việc thiết kế BDK Neuron học theo BDK có sẵn

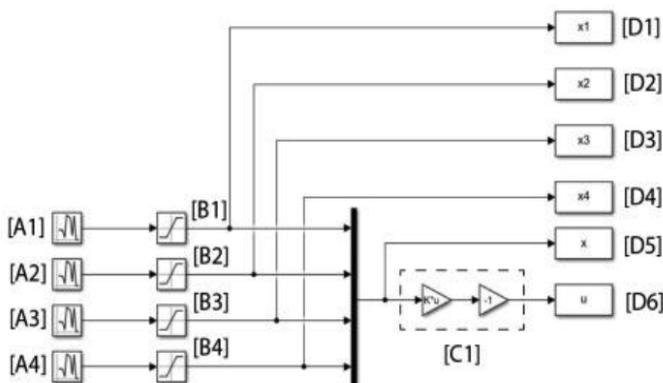
trở nên dễ dàng hơn.

Ở phần này, nhóm tác giả xin trình bày chi tiết các bước sử dụng công cụ “Neuron Toolbox” tạo một BDK Neuron học theo BDK LQR đã có, để điều khiển ổn định hệ BTTG

### 2.3.2. Xây dựng BDK thông minh dùng công cụ Neuron

Bước 1: Thu thập dữ liệu.

**Hình 14. Mô hình thu thập dữ liệu điều khiển của BDK LQR.**



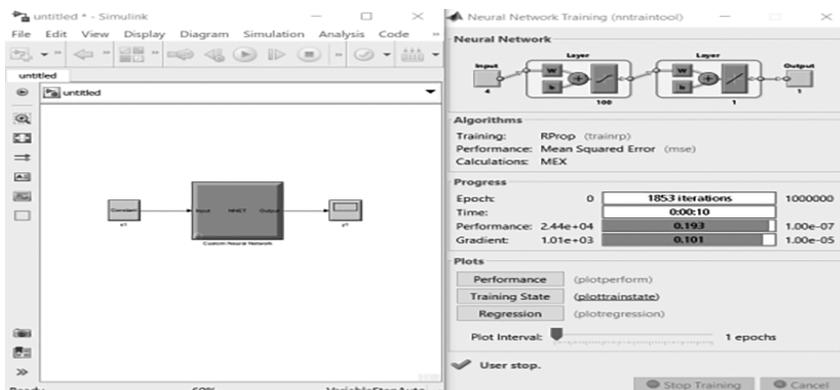
Trong Hình 14:  
 [A1], [A2], [A3], [A4]: các khối “Random”, với mục đích giả lập giá trị 4 thông số ngõ vào của khối LQR ( $p, \dot{p}, \theta, \dot{\theta}$ ),  
 [B1], [B2], [B3], [B4]: các khối “Saturation”, với mục đích giới hạn giá trị 4 tín hiệu ngõ vào trong khoảng [-1; 1],  
 [C1]: BDK LQR,  
 [D1], [D2], [D3], [D4], [D5], [D6]: các khối “To workspace”, với mục đích thu thập lần lượt các giá trị  $p, \dot{p}, \theta, \dot{\theta}$  (tín hiệu đầu vào của bộ LQR) và tín hiệu đầu ra của bộ LQR.

Bước 2: Viết chương trình Matlab sử dụng dữ liệu vừa thu thập được để Matlab tự động xây dựng mạng Neuron điều khiển.

```
ngo_vao = out.x.data';
ngo_ra = out.u.data';
net = newff([minmax(ngo_vao)], [100 1], {'tansig' 'purelin'}, 'trainrp');
net.trainParam.epochs = 1000000;
net.trainParam.lr = 0.00001;
net.trainParam.goal = 0.0000001;
net = train(net, [ngo_vao], [ngo_ra]);
gensim(net, 0.001)
```

Bước 3: Khởi chạy chương trình và chờ đến khi chương trình hoàn tất sẽ thu được BDK Neuron dưới dạng 1 khối điều khiển Simulink.

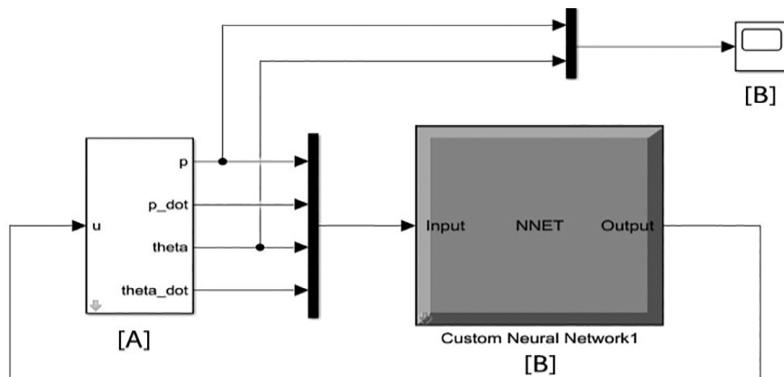
**Hình 15. Kết quả Neuron tạo BDK thông minh học theo LQR**



Đến đây, việc tạo BDK thông minh học theo BDK LQR đã hoàn tất. Việc còn lại là kiểm tra tính chính xác của BDK mới này.

### 2.3.3 Mô phỏng Neuron điều khiển hệ BTTG

Hình 16. Mô hình mô phỏng LQR đã được Neuron tối ưu.



Trong Hình 16:

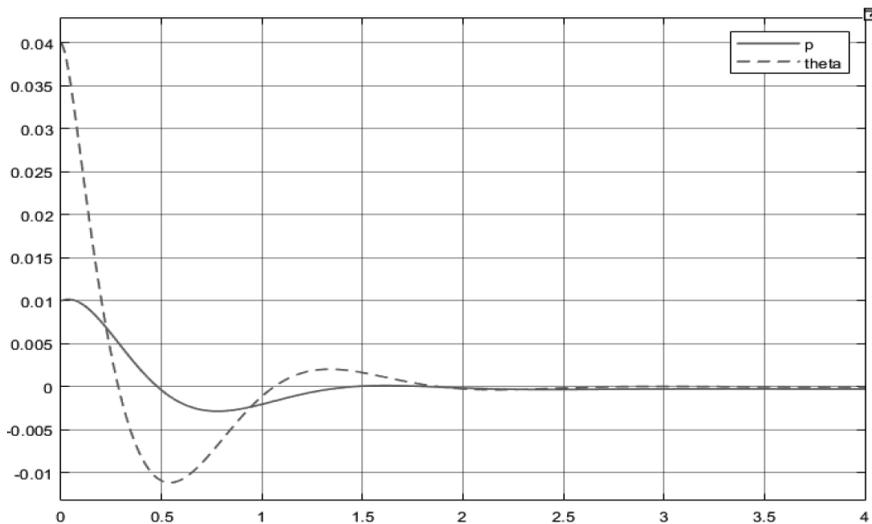
[A]: hệ BTTG với 1 tín hiệu vào  $u$  (điện áp động cơ) và 4 tín hiệu ra ( $p, \dot{p}, \theta, \dot{\theta}$ ),

[B]: BDK Neuron,

[C]: đồng hồ để đo tín hiệu vị trí bóng và góc lệch thanh theo thời gian.

Kết quả cân bằng bóng tại vị trí cân bằng O, với các điều kiện ban đầu  $p = 0.01 \text{ m}$ ;  $\dot{p} = 0.01 \text{ m/s}$ ;  $\theta = 0.04 \text{ rad}$ ;  $\dot{\theta} = -0.02 \text{ rad/s}$ .

Hình 17. Kết quả ổn định vị trí bóng B và góc lệch thanh b của BDK Neuron học LQR.



Trong Hình 17, trục tung là thời gian (s), trục hoành là giá trị đại số, ứng với từng đại lượng sẽ có đơn vị tương ứng. Ứng với vị trí bóng ( $p$ ) có đơn vị cm, ứng với góc lệch thanh ( $\theta$ ) có đơn vị là rad.

Để chi tiết, ta xét các tiêu chí ổn định là sai số xác lập ( $e_{xl}$ ), thời gian xác lập ( $t_{xl}$ ), độ vọt lố (POT).

Bảng 4. Bảng đánh giá các tiêu chí ổn định của hệ BTTG được điều khiển bằng Neuron

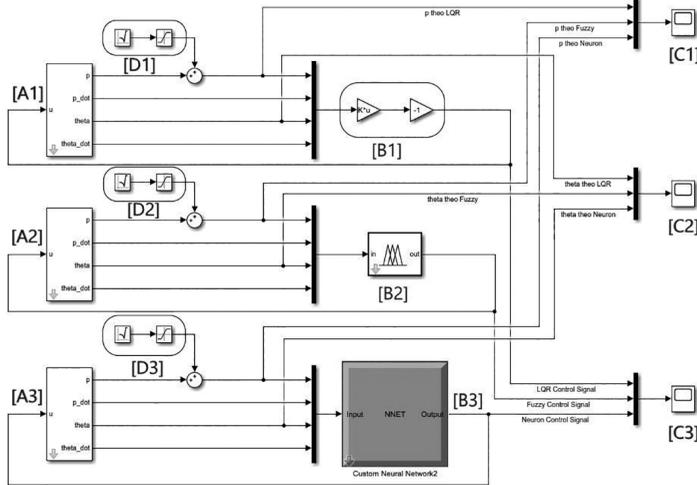
	$e_{xl}$	$t_{xl}$ (s)	POT
$p$	$0.5 * 10^{-3}$ (m)	2	0.0025 (m)
$\theta$	0 (rad)	2.5	0.018 (rad)

Kết quả mô phỏng cho thấy BDK Neuron đã có thể điều khiển hệ thống, có khả năng ổn định hệ thống về sát vị trí cân bằng. Tuy vị trí bóng còn lệch với vị trí cân bằng, nhưng sai số ( $0.5 * 10^{-3}$  m) là rất nhỏ.

### 2.4. So sánh 3 BDK LQR, Fuzzy, Neuron

Để so sánh, ta sẽ xây dựng 1 sơ đồ simulink trong trường hợp có nhiều tín hiệu  $p$  trả về và đo các tín hiệu  $p, \theta, u$  trên cùng 1 đồng hồ "Scope" để dễ đối chiếu và đưa ra nhận định.

### Hình 18. Sơ đồ Simulink mô phỏng 3 BDK LQR, Fuzzy, Neuron điều khiển hệ BTTG



Trong Hình 18:

[A1], [A2], [A3]: cùng là mô hình hệ BTTG giống nhau,

[B1], [B2], [B3]: lần lượt là các BDK theo LQR, Fuzzy, Neuron,

[C1]: đồng hồ “Scope”, với mục đích đo giá trị  $p$  theo thời gian của 3 BDK LQR, Fuzzy, Neuron,

[C2]: đồng hồ “Scope”, với mục đích đo giá trị  $\theta$  theo thời gian của 3 BDK LQR, Fuzzy, Neuron,

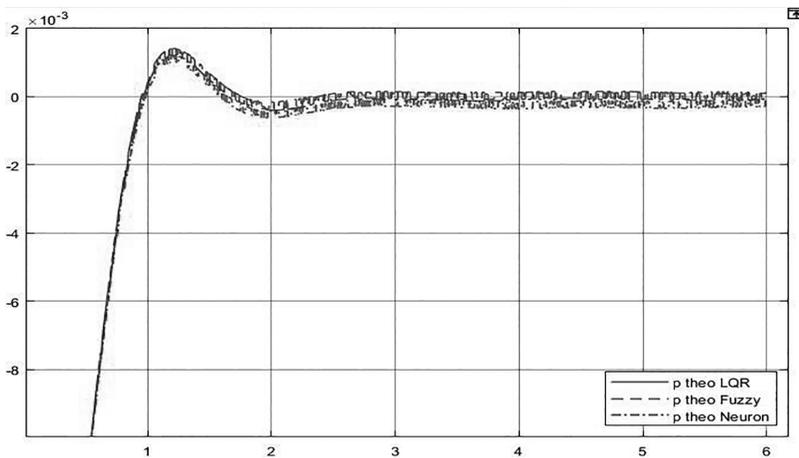
[C3]: đồng hồ “Scope”, với mục đích đo giá trị  $u$  theo thời gian của 3 BDK LQR, Fuzzy, Neuron.

[D1], [D2], [D3]: là các khối Random, để giả lập nhiễu cho tín hiệu vào

Kết quả cân bằng bóng tại vị trí cân bằng O, với cùng các điều kiện ban đầu

$$p = -0.02 \text{ m}; \dot{p} = 0.01 \text{ m/s}; \theta = 0.01 \text{ rad}; \dot{\theta} = 0.07 \text{ rad/s.}$$

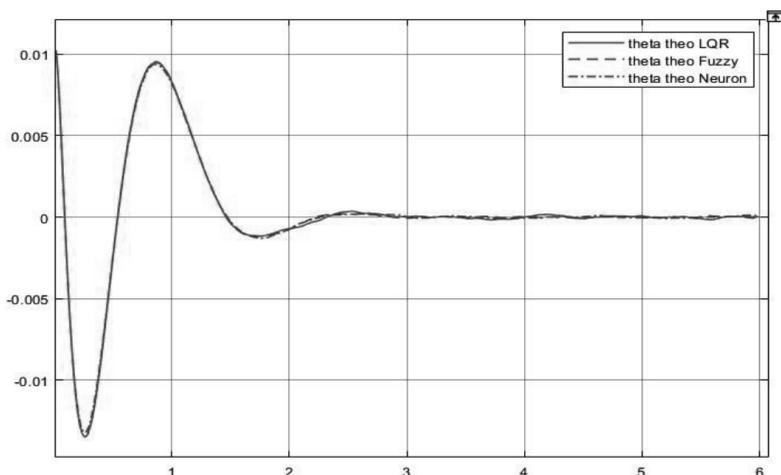
Hình 19. Vị trí bóng theo thời gian của 3 BDK



Trong Hình 19, trục tung là thời gian (s), trục hoành là giá trị đại số, ứng với từng đại lượng sẽ có đơn vị tương ứng. Ứng với vị trí bóng ( $p$ ) có đơn vị cm, ứng với góc lệch thanh ( $\theta$ ) có đơn vị là rad.

Theo đồ thị Hình 19, đồ thị vị trí bóng theo thời gian của 2 BDK Fuzzy và LQR gần như trùng nhau. Kết quả ổn định vị trí bóng của Neuron tuy có sai lệch so với LQR nhưng độ sai lệch là rất nhỏ ( $0.5 * 10^{-3}$  m).

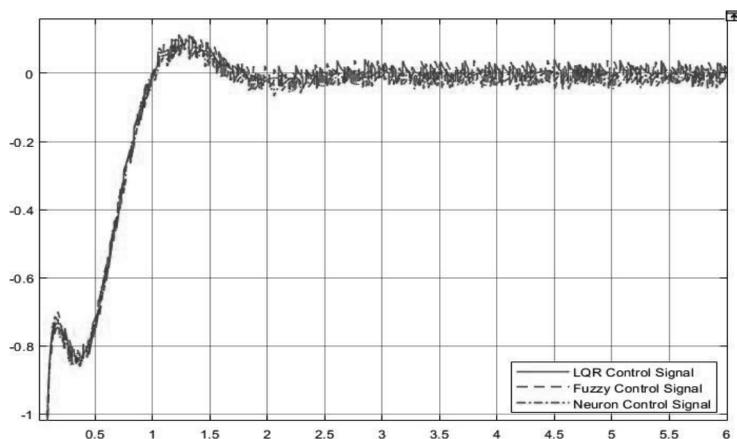
Hình 20. Góc thanh beam theo thời gian của 3 BDK



Trong Hình 20, trục tung là thời gian (s), trục hoành là giá trị đại số, ứng với từng đại lượng sẽ có đơn vị tương ứng. Ứng với vị trí bóng ( $p$ ) có đơn vị cm, ứng với góc lệch thanh ( $\theta$ ) có đơn vị là rad.

Theo đồ thị Hình 20, ta thấy kết quả ổn định góc lệch thanh  $\theta$  theo thời gian của cả 3 BDK là gần giống nhau (3 đường đồ thị gần như trùng nhau).

**Hình 21. Tín hiệu điều khiển của 3 BDK theo thời gian**



Trong Hình 21, trục tung là thời gian (s), trục hoành là giá trị đại số, ứng với từng đại lượng sẽ có đơn vị tương ứng. Ứng với vị trí bóng (p) có đơn vị cm, ứng với góc lệch thanh ( $\theta$ ) có đơn vị là rad.

Theo đồ thị Hình 21, ta thấy tín hiệu điều khiển theo thời gian của 2 BDK Fuzzy và LQR gần như trùng nhau. Tín hiệu điều khiển Neuron tuy có sai lệch so với LQR nhưng sai lệch là rất nhỏ.

Trong trường hợp có nhiễu, các BDK vẫn cho thấy khả năng ổn định hệ tốt dù vẫn còn dao động do nhiễu luôn tác động. Để chi tiết, ta so sánh 3 BDK theo các tiêu chí ổn định là sai số xác lập ( $e_{xl}$ ), thời gian xác lập ( $t_{xl}$ ) và độ vọt lố (POT).

**Bảng 5. Bảng đánh giá các tiêu chí ổn định của hệ BTTG được điều khiển bằng LQR, Fuzzy và Neuron**

		$p$	$\theta$	$u$
$e_{xl}$	LQR	0.0001 m	0.0002	0.001
	Fuzzy	0.0001m	0.0001	- 0.001
	Neuron	$0.5 * 10^{-3}$ m	0.0001	- 0.02
$t_{xl}$ (s)	LQR	2.5	2.55	2.6
	Fuzzy	2.5	2.5	2.6
	Neuron	2.65	2.6	2.7
POT	LQR	0.0014 m	0.0125 rad	0.1
	Fuzzy	0.0014 m	0.0125 rad	0.1
	Neuron	0.0012 m	0.012 rad	0.09

BDK Fuzzy học theo LQR bằng cách sử dụng “Anfis” đã có thể điều khiển được BTTG về vị trí cân bằng và bám rất tốt bộ LQR (Fuzzy có phần tốt hơn khi ổn định góc lệch thanh B trong trường hợp có nhiễu)

BDK Neuron học theo LQR bằng cách dùng công cụ “Neuron Toolbox” đã có thể điều khiển được BTTG về vị trí sát vị trí cân bằng và bám tương đối tốt BDK LQR (dù còn sai số nhưng rất nhỏ).

### 3. Kết luận

Thông qua nghiên cứu này, nhóm tác giả trình bày các bước thực hiện để thực hiện các công cụ mà Matlab hỗ trợ để xây dựng các BDK mờ và neuron tương ứng để tạo ra các BDK thông minh trên cơ sở chuyên gia. Nghiên cứu này hướng đến việc phổ quát công cụ ĐKTM để các

thành phần nghiên cứu không quá cao cấp vẫn nhanh chóng dễ dàng tiếp cận và áp dụng công cụ, được hỗ trợ sẵn, áp dụng thành công cho đối tượng mà mình quan tâm. Thông qua mô phỏng, việc sử dụng “Anfis” và “Neuron Toolbox” học theo BDK LQR là khả thi và kết quả ổn định hệ thống tốt, bám sát BDK LQR có sẵn đối với đối tượng BTTG. Chất lượng các BDK có chất lượng gần như nhau cho thấy các BDK thông minh mà toolbox của Matlab hỗ trợ là phù hợp là có chất lượng tương đương với BDK LQR có sẵn và hoàn toàn có thể thay thế BDK chuyên gia có trước. Cách thức thu thập dữ liệu và huấn luyện thực hiện trên Matlab được trình bày trong bài báo là TLTK cho việc thiết kế BDK thông minh cho đối tượng hộp đen và hộp xám cho trước thông qua kinh nghiệm chuyên gia.

## Lời cảm ơn

Nhóm tác giả xin gửi lời cảm ơn đến TS. Nguyễn Văn Đông Hải (Khoa Điện, HCMUTE) đã hướng dẫn, hỗ trợ để chúng tôi hoàn thành nghiên cứu này.

## TÀI LIỆU THAM KHẢO

- Đặng Ngọc Trung, Trần Thị Thanh Hải. (2014). *Nghiên cứu và ứng dụng phương pháp điều khiển tối ưu LQR cho hệ ball and beam*. Tạp chí khoa học trường đại học kỹ thuật công nghiệp Thái Nguyên, số 72, trang 90 – 93.
- Nguyễn Chí Ngôn và cộng sự. (2021). *Điều khiển giám sát hệ cầu cân bằng với thanh và bóng dùng mạng nơ-ron hàm cơ sở xuyên tâm*. Tạp chí khoa học trường đại học Cần Thơ, số 3A, trang 26 – 35.
- R. Hirsch (1999), "Ball on Beam Instructional System", Shandor Motion Systems, Como, Italy.
- Quanser Inc, "Ball and beam", <https://www.quanser.com/products/ball-and-beam/>, truy cập ngày 12/9/2023.
- Ngoc-Anh Nguyen et al. (2023). Application of Fuzzy-ANFIS Controller for Ball on Wheel System. *Robotica & Management*, Vol. 28, No. 2, pp. 33-37
- Huỳnh Ngọc Thảo và cộng sự. (2023). *Áp dụng giải thuật di truyền hỗ trợ tối ưu hóa cho bộ điều khiển LQR đối với hệ bóng thanh: mô phỏng và thực nghiệm*. Tạp chí Khoa học Trường Đại học Bạc Liêu, Số 12 (12), trang 52-63.
- Huỳnh Thái Hoàng (2006). *Hệ thống điều khiển thông minh*. NXB ĐHQG TPHCM.
- Nguyễn Thị Phương Hà (2002). *Lý thuyết điều khiển tự động*, NXB ĐHQG TPHCM.
- Nguyễn Ngọc Anh và cộng sự. (2023). *Xây dựng bộ điều khiển mờ cho hệ bóng trên bánh xe*. Tạp chí Khoa học Trường Đại học Bạc Liêu, ISSN: 2734-973X, Số 12 (12), trang 64-73.