

SO SÁNH BỘ ĐIỀU KHIỂN PID VỚI BỘ ĐIỀU KHIỂN LQG CHO ROBOT CÂN BẰNG TRÊN QUẢ CẦU

COMPARISON OF A PID CONTROLLER VERSUS LQG CONTROLLER FOR A BALL BALANCING ROBOT

Cao Thanh Bộ¹, Phạm Trường Hưng¹, Lê Hoài Nam², Nguyễn Danh Ngọc²

¹Sinh viên ngành Kỹ thuật Cơ điện tử, Trường Đại học Bách khoa – Đại học Đà Nẵng;

²Trường Đại học Bách khoa – Đại học Đà Nẵng; lehoainam@dut.udn.vn; ndngoc@dut.udn.vn

Tóm tắt - Với diện tích tiếp xúc giữa robot và mặt đất nhỏ, khả năng chuyển động đa hướng và đổi hướng nhanh mà không cần xoay thân, robot cân bằng trên quả cầu phù hợp với các không gian làm việc chật hẹp. Robot này vốn dĩ không ổn định nên phải luôn được giữ cân bằng một cách chủ động. Trong bài báo này, hai bộ điều khiển (bộ điều khiển PID và bộ điều khiển LQG) được so sánh, đánh giá để lựa chọn áp dụng cho nguyên mẫu robot cân bằng trên quả cầu được thiết kế và chế tạo tại Khoa Cơ khí, Trường Đại học Bách khoa, Đại học Đà Nẵng. Chất lượng của các bộ điều khiển được đánh giá thông qua mô phỏng với các điều kiện hoạt động khác nhau. Các kết quả cho thấy bộ điều khiển LQG có chất lượng tốt hơn (thời gian đáp ứng, thời gian xác lập nhanh hơn) bộ điều khiển PID.

Từ khóa - robot cân bằng trên quả cầu; thiết kế điều khiển; PID; LQG; bộ lọc Kalman.

1. Đặt vấn đề

Robot cân bằng trên quả cầu (*Ball Balancing Robot* – BBR), còn gọi là *Ballbot* theo tên nguyên mẫu đầu tiên được nghiên cứu chế tạo [1]. Đây là loại robot di động gồm một quả cầu (bóng) và một robot cân bằng trên đầu quả cầu đó bằng cách lái các bánh của nó theo hướng thích hợp để tránh ngã đổ. Trái ngược với các robot di động truyền thống vốn dựa vào trọng tâm thấp và chiều dài cơ sở (*wheel-base*) lớn để giữ robot đứng thẳng, robot cân bằng trên quả cầu vốn dĩ không ổn định nên phải luôn được giữ cân bằng một cách chủ động [2] (tức phải luôn điều khiển các bánh). Ngoài ra, các robot di động bằng bánh truyền thống thường có hai bánh lái độc lập (ví dụ *Segway*) có thể xoay quanh điểm bất kỳ, nhưng không thể đổi hướng di chuyển ngay lập tức. Hạn chế này được khắc phục bởi robot cân bằng trên quả cầu, chúng có thể thực hiện chuyển động đa hướng nhanh mà không cần thay đổi hướng của các bánh xe (không cần xoay thân) [3]. Ưu điểm này giúp robot tránh va chạm khi làm việc trong các không gian chật hẹp.

Nhiều nguyên mẫu BBR đã được thiết kế và chế tạo. Nguyên mẫu đầu tiên được phát triển vào năm 2006 tại Đại học (ĐH) Carnegie Mellon (CMU) có kích thước tương đương người thật với mục đích tương tác với con người. Robot này có một cơ chế lái khá phức tạp gồm 4 con lăn (*roller*) chủ động và một hệ thống lái nhằm thực hiện chuyển động xoay theo trục dọc. Năm 2008, ĐH Tohoku Gakuin (TGU) đã phát triển một nguyên mẫu BBR khác nhỏ hơn so với nguyên mẫu của CMU, có thể thực hiện các chuyển động tương tự (bao gồm chuyển động xoay quanh trục dọc) chỉ với 3 động cơ nối với 3 bánh xe đa hướng (*omni-wheel*) để lái quả bóng [4]. ĐH Adelaide xây dựng một BBR có hai bánh xe để lái bóng sử dụng bộ kit LEGO

Abstract - With small footprints, omnidirectional motion, and ability to swiftly reverse without rotating the body, ball balancing robots are suitable for operation in narrow areas. This robot is inherently unstable, so it must always be actively balanced. In this paper, two controllers (a PID controller and a LQG controller) are compared to apply to the prototype designed and built at the Faculty of Mechanical Engineering, University of Science and Technology, the University of Danang. The performance of the controllers is assessed through simulation under different operating conditions. The results show that LQG controllers are of better performance (small rise time, settling time) than PID controllers.

Key words - ball balancing robot; control design; PID; LQG; Kalman filter.

Mindstorms NXT và các khối LEGO vào năm 2009 [2]. ETH Zurich vào năm 2010 cũng phát triển nguyên mẫu BBR tương tự của TGU. Nguyên mẫu Rezero của ETH Zurich có độ bền vững động (*dynamic robustness*) cao, có thể đạt được tốc độ đến 2 m/s và góc nghiêng 20° [5]. Gần đây, có thêm nhiều nguyên mẫu BBR được thiết kế và chế tạo [6, 7, 8, 3].

Để BBR có thể thực hiện các chuyển động phức tạp ở tốc độ cao, nhiều chiến lược điều khiển khác nhau được đề xuất: điều khiển PD [4], PID tự chỉnh mờ [9], PID tăng [5], PID tăng tích hợp cơ cấu bù feedforward [10], cũng như điều khiển mờ [11], điều khiển toàn phương tuyến tính (*Linear Quadratic Regulator* - LQR) kết hợp với một khâu tích phân [2] được đề xuất để điều khiển BBR.

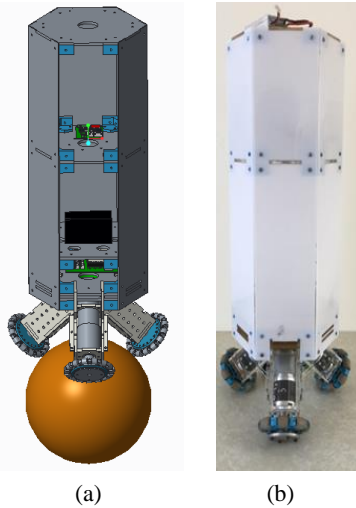
Tuy nhiên, theo hiểu biết của nhóm tác giả thì chưa nhiều nghiên cứu đề cập đến việc ứng dụng bộ điều khiển Gaussian toàn phương tuyến tính (*Linear Quadratic Gaussian* - LQG) vào điều khiển BBR. Bộ điều khiển LQG là một trường hợp cụ thể của LQR, trong đó bộ lọc Kalman được sử dụng để ước lượng trạng thái của hệ thống. Trong bài báo, bộ điều khiển LQG được thiết kế và đánh giá hiệu năng, thông qua mô phỏng, với một bộ điều khiển cổ điển là PID.

2. Mô hình hoá hệ thống

2.1. Nguyên mẫu robot

Lấy cảm hứng từ Rezero của ETH Zurich, nguyên mẫu BBR được thiết kế và chế tạo tại Khoa Cơ khí, Trường Đại học Bách khoa – Đại học Đà Nẵng gồm ba bánh xe đa hướng được truyền động bởi ba động cơ DC được bố trí lệch nhau 120°. Điểm đặc biệt là phần thân nguyên mẫu có hình lăng trụ lục giác đều, phù hợp cho việc chế tạo lớp vỏ bảo vệ. Như thể hiện trên Hình 1,

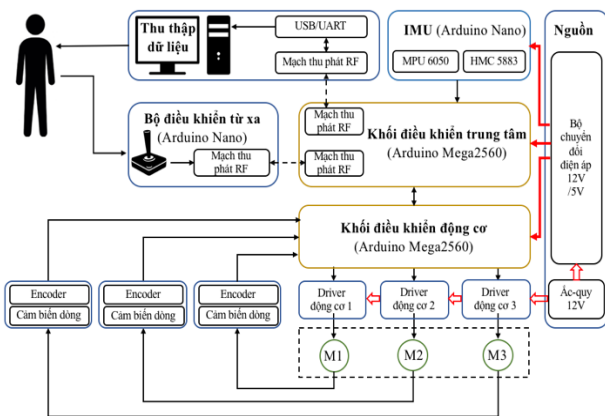
nguyên mẫu BBR bao gồm 3 phần chính: phần thân, hệ thống động lực và quả cầu.



Hình 1. Nguyên mẫu robot: (a) Mô hình 3D và (b) Mô hình thật

Thân robot bao gồm 4 tấm ngang hình lục giác và 6 tấm phẳng thẳng đứng được làm bằng vật liệu ABS và plexiglass để giảm khối lượng của BBR. Các tấm này được lắp ráp với nhau bởi các chi tiết nối được chế tạo bằng công nghệ in 3D. Hệ thống động lực được gắn chặt vào tấm đáy. Tấm thứ hai đặt nguồn cung cấp cho robot. Khối đo lường quán tính (IMU - *Inertia Measurement Unit*) và mạch điều khiển trung tâm được đặt trên tấm thứ ba.

Hình 2 minh họa cấu trúc hệ thống điều khiển của robot. Dữ liệu trạng thái của robot được thu thập bởi khối IMU bao gồm một cảm biến MPU6050 xác định vận tốc góc bằng gia tốc kế và con quay hồi chuyển (*gyroscope*) và một cảm biến HMC5883L đo lường góc thông qua hướng và độ lớn của từ trường Trái đất. Dữ liệu từ hai cảm biến được tổng hợp bởi mạch Arduino Nano trước khi gửi đến khối điều khiển trung tâm.



Hình 2. Cấu trúc hệ thống điều khiển

Khối điều khiển trung tâm là một mạch Arduino Mega2560, nhận dữ liệu trạng thái từ IMU và tính toán, truyền đến khối điều khiển động cơ, các giá trị đặt cho mỗi động cơ. Mạch này cũng đảm bảo các kết nối không dây để điều khiển và thu thập dữ liệu BBR từ xa.

Hệ thống động lực robot bao gồm ba động cơ DC dẫn động ba bánh xe đa hướng và các mạch driver tương ứng. Vị trí trục và dòng điện của các động cơ được đo bằng

encoder và cảm biến dòng. Các thông tin này được khối điều khiển động cơ sử dụng để điều chỉnh tốc độ và mô-men xoắn của mỗi động cơ theo phương pháp điều chế độ rộng xung (PWM - *Pulse Width Modulation*). Các thông số cơ bản của nguyên mẫu robot được tóm tắt bởi Bảng 1.

Bảng 1. Các thông số cơ bản của nguyên mẫu

Thông số	Ký hiệu	Thứ nguyên	Giá trị
Chiều cao phần thân	h	m	0,67
Chiều dài cạnh lục giác đều	a	m	0,115
Khối lượng phần thân	m_B	kg	8,6
Khối lượng động cơ	m_M	kg	1,3
Khối lượng bánh xe đa hướng	m_{OW}	kg	0,12
Khối lượng bóng	m_S	kg	0,8
Bán kính bóng	r_S	m	0,125
Bán kính bánh xe đa hướng	r_{OW}	m	0,05
Tỉ số truyền hộp số	k	-	99,5

2.2. Mô hình toán học

Mục này giới thiệu mô hình tuyến tính của BBR sẽ được sử dụng để thiết kế các bộ điều khiển ở mục tiếp theo. Trong bài báo này, phương pháp Lagrangian được sử dụng để xác định các phương trình chuyển động. Bằng cách giải phương trình Euler-Lagrange:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \tau_{ext,i} \quad (1)$$

Trong đó:

- $i = 1 \dots 5$ ứng với các thành phần của véc-tơ tọa độ tối thiểu q được xác định bởi:

$$q = [x_S \ y_S \ \psi_x \ \psi_y \ \psi_z]^T$$

với x_S và y_S tương ứng thể hiện dịch chuyển của quả bóng dọc theo trục x và trục y, và ψ_x , ψ_y và ψ_z xác định hướng của thân robot và các bánh xe đa hướng.

- Lagrangian $L(q, \dot{q}) = T - V$ thể hiện động năng T và thế năng V của tất cả các khối cứng của robot (bóng, thân robot và các bánh đa hướng).

- τ_{ext} thể hiện các mô-men xoắn bên ngoài. BBR được kích hoạt bởi ba bánh xe đa hướng và động cơ. Do đó, mô-men xoắn bên ngoài là mô-men xoắn của các bánh xe đa hướng, được truyền bởi mô-men xoắn của các động cơ.

Ta nhận được các phương trình mô tả chuyển động của BBR có dạng như sau:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau_{ext} \quad (2)$$

Trong đó: thành phần $M(q)$ đại diện cho các lực quán tính do gia tốc, $C(q, \dot{q})$ đại diện cho lực Coriolis và lực ly tâm, $G(q)$ đại diện cho lực hấp dẫn và về bên phải τ_{ext} đại diện cho các mô-men xoắn bên ngoài [10]. Trình tự tính toán chi tiết có thể tham khảo trong [2] và [6].

Để cân bằng BBR, cần xem xét đến đặc tính động học của robot xung quanh vị trí thẳng đứng của nó (điểm cân bằng không ổn định).

Định nghĩa véc-tơ trạng thái:

$$x = [q \ \dot{q}]^T = [x_S \ y_S \ \psi_x \ \psi_y \ \psi_z \ \dot{x}_S \ \dot{y}_S \ \dot{\psi}_x \ \dot{\psi}_y \ \dot{\psi}_z]^T \quad (3)$$

và các đầu vào u:

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \quad (4)$$

Trong đó, τ_i là mô-men xoắn của bánh xe đa hướng i .

Tại điểm cân bằng, tất cả các biến trạng thái và các đầu vào bằng 0. Mô hình tuyến tính hóa có thể được biểu diễn bằng một dạng không gian trạng thái tuyến tính:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (5)$$

Trong đó, các ma trận A, B được định nghĩa như sau:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial \ddot{x}_S}{\partial x_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{x}_S}{\partial y_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\partial \ddot{y}_S}{\partial x_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{y}_S}{\partial y_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\partial \ddot{\psi}_x}{\partial x_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_x}{\partial y_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\partial \ddot{\psi}_y}{\partial x_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_y}{\partial y_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\partial \ddot{\psi}_z}{\partial x_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_z}{\partial y_S} \Big|_{(x,u)=(\bar{x},\bar{u})} & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{\partial \ddot{x}_S}{\partial u_1} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{x}_S}{\partial u_2} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{x}_S}{\partial u_3} \Big|_{(x,u)=(\bar{x},\bar{u})} \\ \frac{\partial \ddot{y}_S}{\partial u_1} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{y}_S}{\partial u_2} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{y}_S}{\partial u_3} \Big|_{(x,u)=(\bar{x},\bar{u})} \\ \frac{\partial \ddot{\psi}_x}{\partial u_1} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_x}{\partial u_2} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_x}{\partial u_3} \Big|_{(x,u)=(\bar{x},\bar{u})} \\ \frac{\partial \ddot{\psi}_y}{\partial u_1} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_y}{\partial u_2} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_y}{\partial u_3} \Big|_{(x,u)=(\bar{x},\bar{u})} \\ \frac{\partial \ddot{\psi}_z}{\partial u_1} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_z}{\partial u_2} \Big|_{(x,u)=(\bar{x},\bar{u})} & \frac{\partial \ddot{\psi}_z}{\partial u_3} \Big|_{(x,u)=(\bar{x},\bar{u})} \end{bmatrix} \quad (6)$$

Các thành phần $\frac{\partial \ddot{q}}{\partial x_i}$ và $\frac{\partial \ddot{q}}{\partial u_i}$ có thể được xác định bằng cách giải các phương trình nhận được bằng đạo hàm các phương trình chuyển động (2) theo các biến trạng thái và biến đầu vào:

$$M(\bar{q}) \frac{\partial \ddot{q}}{\partial x_i} + \frac{\partial G(q)}{\partial x_i} \Big|_{(x,u)=(\bar{x},\bar{u})} = \frac{\partial \tau_{ext}}{\partial x_i} \Big|_{(x,u)=(\bar{x},\bar{u})} \quad (7)$$

$$M(\bar{q}) \frac{\partial \ddot{q}}{\partial u_i} = \frac{\partial \tau_{ext}}{\partial u_i} \Big|_{(x,u)=(\bar{x},\bar{u})}$$

với $\bar{q} = 0$ và $(\bar{x}, \bar{u}) = (0,0)$ (tại vị trí cân bằng).

Ma trận quan sát C thể hiện mối quan hệ giữa các giá trị đo lường và các biến trạng thái của hệ thống. Khối IMU đo trực tiếp các biến trạng thái ψ_x, ψ_y, ψ_z . Các biến trạng thái x_S, y_S có thể được xác định bằng các giá trị góc của bánh xe đa hướng (đo bằng encoder).

Sau khi hoàn thành các tính toán động lực học và thay thế các giá trị thông số mô hình của nguyên mẫu BBR (Bảng 1), ta nhận được các ma trận sau:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -2.51 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.51 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10.3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10.3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1,1741 & 1,1741 \\ 1,3556 & -0,6778 & -0,6778 \\ 2,1616 & -1,0808 & -1,0808 \\ 0 & 1,872 & -1,872 \\ -6,4787 & -6,4787 & -6,4787 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (8)$$

Trong thực tế, nhiễu xuất hiện trong các cảm biến trong quá trình đo. Do đó, để tính đến ảnh hưởng của nhiễu đo lường và nhiễu quá trình, một mô hình không gian trạng thái với nhiễu loạn được sử dụng:

$$\dot{x} = Ax + Bu + w \quad (9)$$

$$y = Cx + v$$

Trong đó, w và v lần lượt là nhiễu quá trình và nhiễu đo lường. Cả hai được giả định là nhiễu trắng và không tương quan với nhau:

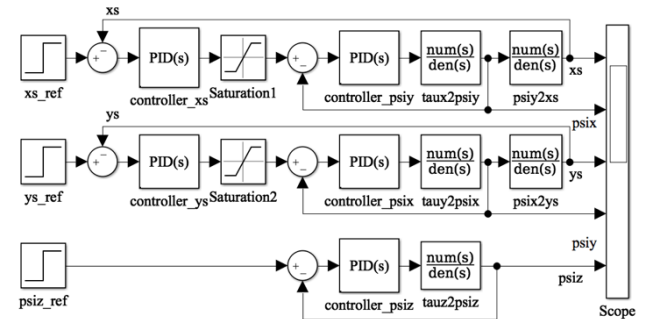
$$w \sim \mathcal{N}(0, Q), \quad v \sim \mathcal{N}(0, R) \quad (10)$$

Trong đó, Q và R là ma trận hiệp phương sai của nhiễu quá trình và nhiễu đo lường.

3. Thiết kế bộ điều khiển

Trong mục này, hai bộ điều khiển PID và LQG được đề xuất để cân bằng và điều khiển vị trí BBR.

3.1. Bộ điều khiển PID



Hình 3. Hệ thống điều khiển BBR với bộ điều khiển PID

Bộ điều khiển PID (Proportional – Integral – Derivative) là bộ điều khiển phân hồi dựa trên sai lệch giữa điểm đặt mong muốn và giá trị đo lường của quá trình. Sai lệch này được sử dụng để điều chỉnh đầu vào quá trình để giá trị đầu ra gần với giá trị đặt. Hàm truyền của bộ điều khiển PID như sau:

$$H_{PID}(s) = K_p \left[1 + \frac{1}{sT_i} + T_d s \right] \quad (11)$$

với K_p là hệ số tỉ lệ, T_i là hằng số thời gian tích phân và T_d là hằng số thời gian vi phân. Các thông số này phải được lựa chọn thích hợp để đảm bảo sự ổn định và hiệu suất của hệ thống [12].

Hệ thống điều khiển PID cho BBR bao gồm 5 bộ điều khiển tương ứng với 5 thành phần của véc-tơ tọa độ tối thiểu (Hình 3).

Đầu vào của hệ thống là 3 giá trị mô-men τ_x, τ_y, τ_z tương ứng là mô-men giả lập của bánh xe đa hướng lần lượt theo mặt phẳng yz, xz, xy [2]. Các hàm truyền hệ thống được xác định bằng cách chuyển đổi từ mô hình không gian trạng thái (5) theo [2]:

$$P = C \cdot (sI - A)^{-1} \cdot B \cdot M \quad (12)$$

Trong đó, I là ma trận đơn vị 10×10 và M là ma trận chuyển đổi giữa mô-men giả lập τ_x, τ_y, τ_z và mô-men xoắn của ba bánh xe đa hướng τ_1, τ_2, τ_3 .

$$M = \begin{bmatrix} \frac{2\sqrt{2}}{3} & 0 & \frac{-\sqrt{2}}{3} \\ -\frac{\sqrt{2}}{3} & \frac{\sqrt{6}}{3} & \frac{-\sqrt{2}}{3} \\ -\frac{\sqrt{2}}{3} & \frac{-\sqrt{6}}{3} & \frac{-\sqrt{2}}{3} \end{bmatrix} \quad (13)$$

Như vậy, đặc tính động lực học của BBR có thể được mô tả thông qua các hàm truyền sau:

$$H_{\tau_y \rightarrow \psi_y}(s) = \frac{3,0570}{s^2 - 10,3}$$

$$H_{\tau_x \rightarrow \psi_x}(s) = \frac{3,0570}{s^2 - 10,3}$$

$$H_{\tau_z \rightarrow \psi_z}(s) = \frac{9,1623}{s^2} \quad (14)$$

$$H_{\psi_y \rightarrow x_s}(s) = \frac{-0,6272s^2 + 3,95}{s^2}$$

$$H_{\psi_x \rightarrow y_s}(s) = \frac{0,6272s^2 - 3,95}{s^2}$$

Với các hàm truyền hệ thống được xác định ở trên các bộ điều khiển PID tương ứng được thiết kế với sự hỗ trợ của công cụ Tuner của MATLAB.

3.2. Bộ điều khiển LQG

Bộ điều khiển toàn phương Gauss (*Linear Quadratic Gaussian - LQG*) xem xét các hệ thống tuyến tính với các yếu tố ngẫu nhiên tuân theo phân bố Gaussian. Một cách cơ bản, LQG là sự kết hợp giữa bộ lọc Kalman và bộ điều khiển toàn phương tuyến tính (*Linear Quadratic Regulator - LQR*). Trong trường hợp này, bộ điều khiển được thiết kế với véc-tơ không gian trạng thái được ước lượng bởi bộ lọc Kalman.

Từ phương trình (9), bộ điều khiển LQG được xác định bởi các phương trình sau:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + K(y - C\hat{x}) \\ u &= -L\hat{x} \end{aligned} \quad (15)$$

Trong đó, K là ma trận độ lợi của bộ lọc Kalman, L là ma trận phản hồi của bộ điều khiển LQR.

Tại mỗi thời điểm, bộ lọc sẽ ước lượng \hat{x} của trạng thái x sử dụng các giá trị đo lường và đầu vào trước đó. Ma trận

độ lợi Kalman được tính bởi [13]:

$$K = PC^T R^{-1} \quad (16)$$

Trong đó, P đạt được bằng cách giải phương trình Riccati:

$$AP + PA^T - PC^T R^{-1} CP + Q = 0 \quad (17)$$

Ma trận phản hồi L của bộ điều khiển LQR với luật điều khiển $u = -L\hat{x}$ được xác định bằng phương pháp tối thiểu hóa phương trình chỉ tiêu chất lượng toàn phương:

$$J = \int_0^{\infty} (\hat{x}^T Q_L \hat{x} + u^T R_L u) dt \quad (18)$$

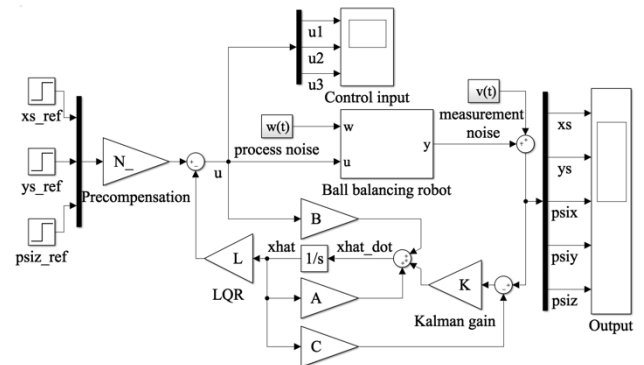
với Q_L và R_L lần lượt là ma trận trọng số trạng thái và điều khiển. Ma trận phản hồi L được tính như sau [13]:

$$L = R_L^{-1} B^T S \quad (19)$$

Trong đó, S có được bằng phương pháp giải phương trình Riccati:

$$A^T S + SA - SBR_L^{-1} B^T S + Q_L = 0 \quad (20)$$

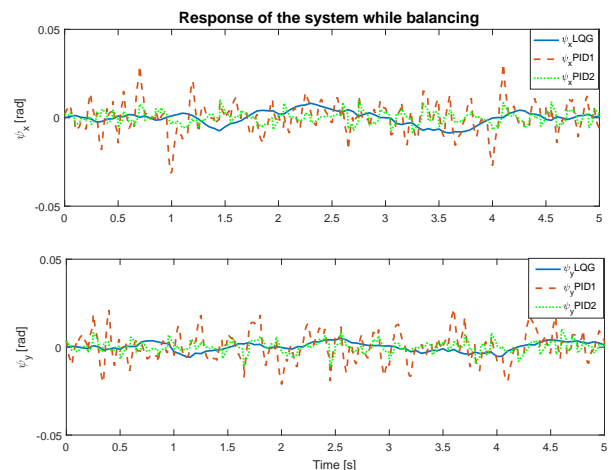
Hình 4 mô phỏng hệ thống điều khiển LQG của BBR trên Matlab/Simulink.



Hình 4. Hệ thống điều khiển BBR với bộ điều khiển LQG

4. Đánh giá chất lượng điều khiển

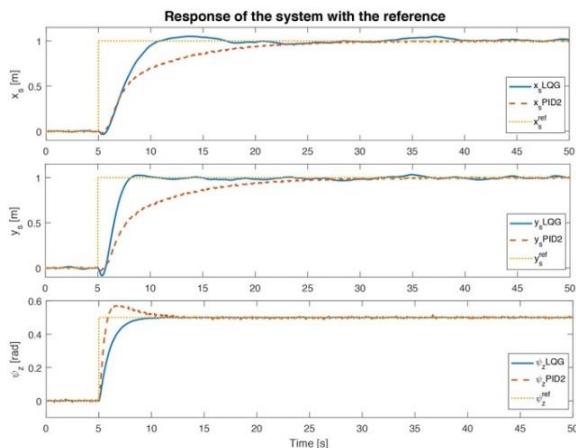
Hình 5 và 6 thể hiện các đáp ứng của BBR theo 2 bộ điều khiển PID và LQG trong trường hợp robot cân bằng và trường hợp tín hiệu đặt thay đổi. Như thể hiện trên Hình 5, cả hai loại bộ điều khiển PID và LQG đều hoạt động tốt trong trường hợp muốn cân bằng robot (giá trị đặt của góc nghiêng thân ψ_x, ψ_y bằng 0).



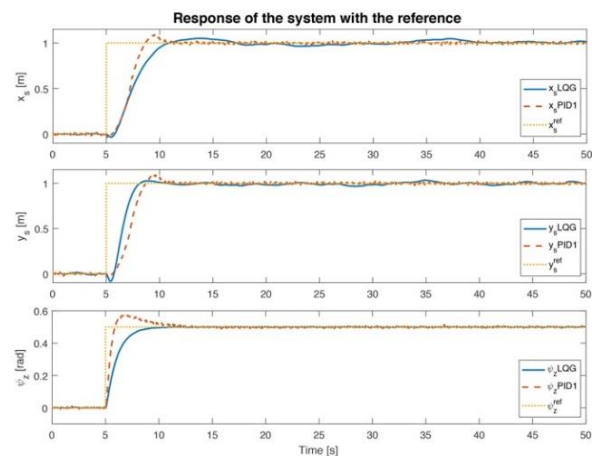
Hình 5. Đáp ứng của hệ thống khi cân bằng

Trong trường hợp cần điều khiển vị trí của robot, tức có sự thay đổi của tín hiệu đặt, bộ điều khiển LQG được so

sánh với 2 bộ điều khiển PID với các hệ số được điều chỉnh theo các yêu cầu chất lượng khác nhau. Cụ thể, một bộ điều khiển PID được điều chỉnh theo tiêu chí vọt lố nhỏ (Hình 6) và một bộ điều khiển PID theo tiêu chí đáp ứng nhanh (Hình 7).



Hình 6. Đáp ứng của hệ thống với bộ điều khiển LQG và PID vọt lố nhỏ khi tín hiệu đặt thay đổi: (a) thay đổi x_s ; (b) thay đổi y_s và (c) thay đổi ψ_z



Hình 7. Đáp ứng của hệ thống với bộ điều khiển LQG và PID đáp ứng nhanh khi tín hiệu đặt thay đổi: (a) thay đổi x_s ; (b) thay đổi y_s và (c) thay đổi ψ_z

Bảng 2. So sánh chỉ tiêu chất lượng PID và LQG

Chỉ tiêu chất lượng	LQG	PID ít vọt lố	PID đáp ứng nhanh
Thời gian quá độ (s)	3,025	16,384	2,105
Thời gian xác lập (s)	4,49	18,77	4,97
$ISE = \int_0^{\infty} e^2(t)dt$	2,0336	2,4749	1,7961
$IAE = \int_0^{\infty} e(t) dt$	3,4238	5,0018	2,5495
$IATE = \int_0^{\infty} te(t)dt$	10,0141	52,3034	12,7006

Bảng 2 so sánh các chỉ tiêu chất lượng của hai bộ điều

khiển trong trường hợp giá trị đặt $x_s = 1$ m tại thời điểm 5 s. Có thể thấy bộ điều khiển LQG có các chỉ tiêu về chất lượng tốt hơn so với hai bộ điều khiển PID.

5. Kết luận

Trong bài báo, hai kỹ thuật điều khiển tuyến tính (PID và LQG) được so sánh và đánh giá thông qua mô phỏng cho vấn đề điều khiển robot cân bằng trên quả cầu. Đầu tiên, hệ thống robot được điều khiển bởi năm bộ điều khiển PID, trong đó mỗi bộ PID điều khiển một thành phần của véc-tơ tọa độ tối thiểu q của robot. Sau đó, một bộ điều khiển LQG tích hợp một bộ lọc Kalman được xây dựng cho robot này. Kết quả mô phỏng cho thấy bộ điều khiển LQG cho đáp ứng nhanh và chất lượng tốt hơn so với PID. Do đó, bộ điều khiển LQG sẽ được xem xét áp dụng trên nguyên mẫu thực tế.

Lời cảm ơn: Bài báo này được tài trợ bởi Trường Đại học Bách khoa – Đại học Đà Nẵng với đề tài có mã số T2018-02-09.

TÀI LIỆU THAM KHẢO

- [1] Nagarajan Umashankar, *Dynamic Constraint-Based Optimal Shape Trajectory Planner for Shape Accelerated Underactuated Balancing Systems*, Proceedings of 2010 Robotics: Science and Systems, 2010, pp. 27-31.
- [2] Blonk, J. W., *Modeling and Control of a Ball-Balancing Robot*, Master's Thesis, University of Twente, 2014.
- [3] Nagarajan, U., Mampetta, A., Kantor, G. A., & Hollis, R. L., *State Transition, Balancing, Station Keeping, and Yaw Control for a Dynamically Stable Single Spherical Wheel Mobile Robot*, IEEE International Conference on Robotics and Automation, May 2009, pp. 998-1003.
- [4] Kumagai, M., & Ochiai, T., *Development of A Robot Balancing on A Ball*, International Conference on Control, Automation and Systems, Oct. 2008, pp. 433-438.
- [5] Fong, J., Uppill, S., & Cazzolato, B., *Design and Build A Ballbot*, In Report. Adelaide, Australia: The University of Adelaide, 2009.
- [6] Fankhauser, P., & Gwerder, C., *Modeling and Control of A Ballbot*, Bachelor's Thesis, Eidgenössische Technische Hochschule Zürich, 2010.
- [7] Su, X., Wang, C., Su, W., & Ding, Y., *Control of Balancing Mobile Robot on A Ball with Fuzzy Self-Adjusting PID*, Chinese Control and Decision Conference (CCDC), May 2016, pp. 5258-5262.
- [8] Vaidya, B., Shomin, M., Hollis, R., & Kantor, G., *Operation of the Ballbot on Slopes and with Center-of-Mass Offsets*, IEEE International Conference on Robotics and Automation (ICRA), May 2015, pp. 2383-2388.
- [9] Yunong, Y., Ha, H. M., Kim, Y. K., & Lee, J. M., *Balancing and Driving Control of A Ball Robot Using Fuzzy Control*, International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Oct. 2015, pp. 492-494.
- [10] André Alstrin, Emil Sundell, *Development of A Mechatronical Platform for AUTOSAR - The Ball-Balancing Robot*, Master's Thesis, Chalmers University of Technology.
- [11] Carlucho, Ignacio, et al., *Comparison of A PID Controller versus A LQG Controller for An Autonomous Underwater Vehicle*, IEEE/OES South American International Symposium on Oceanic Engineering (SAISOE), 2016.
- [12] Karl J Astrom, *PID Controllers: Theory, Design and Tuning*, Instrument Society of America, 1995.
- [13] Lavretsky, E., & Wise, K. A., *Robust and Adaptive Control*, Springer, London, 2013.