

# Sử dụng giải thuật bầy đàn để chỉnh định bộ điều khiển PID trên mô hình robot PUMA 560

## Using a Particle Swarm Optimization - based PID Controller for Tuning the PUMA 560 Robot

Lê Hoàng Đăng<sup>1</sup>, Nguyễn Đình Tứ<sup>1</sup>, Trần Chí Cường<sup>1</sup>, Nguyễn Thanh Cường<sup>2</sup>, Trần Thanh Hùng<sup>1</sup>,  
Nguyễn Chí Ngôn<sup>1</sup>

<sup>1</sup>ĐH Cần Thơ, <sup>2</sup>CD nghề Đồng Tháp

Email: hoangdangtdh37@gmail.com, ncngon@ctu.edu.vn

### Abstract

PID controllers are applied widely in industry. Ziegler-Nichols algorithms are often used to adjust certain controllers. However, this method cannot achieve optimal coefficients of  $K_p$ ,  $K_i$ ,  $K_d$  due to the influence of the error of the measuring devices and interference from system disturbances. From there, the designer needs to select the appropriate parameters of the controller that is not easy to reach the optimal values and requires experience of the designer. So, this study proposes a method to determine the optimal coefficients of  $K_p$ ,  $K_i$ ,  $K_d$  around the operating point of the PID controller by using a particle swarm optimization algorithm. Simulation results show that the algorithm has performed effectively in finding optimum coefficients satisfying the control quality criteria such as small steady-state error and fast response time.

### Keywords

PID controller, particle swarm optimization, PUMA 560 robot.

### Tóm tắt<sup>1</sup>

Bộ điều khiển PID được ứng dụng rất phổ biến trong công nghiệp. Giải thuật Ziegler-Nichols thường được sử dụng để chỉnh định bộ điều khiển. Tuy nhiên, phương pháp này khó có thể tối ưu được các hệ số  $K_p$ ,  $K_i$ ,  $K_d$  do ảnh hưởng sai số của thiết bị đo và nhiễu từ hệ thống. Từ đó, người thiết kế phải lựa chọn các tham số phù hợp cho bộ điều khiển. Phương pháp này không có cơ sở để xác định giá trị tối ưu và đòi hỏi kinh nghiệm của người thiết kế. Vì thế, bài báo đề xuất một phương pháp xác định các hệ số tối ưu xung quanh điểm điều hành của bộ điều khiển PID bằng giải thuật bầy đàn. Đối tượng được sử dụng để kiểm nghiệm giải thuật này là mô hình cánh tay robot ba bậc tự do PUMA 560. Kết quả mô phỏng cho thấy giải thuật đã phát huy hiệu quả trong việc tìm kiếm hệ số tối ưu để đạt được các chỉ tiêu chất lượng điều khiển như sai số xác lập nhỏ và thời gian đáp ứng nhanh.

### Ký hiệu

Ký hiệu	Ý nghĩa
$A(q), B(q), C(q), g(q)$	Ma trận của mô hình
$G_{PID}(s)$	Hàm truyền đạt

### Chữ viết tắt

PSO	Particle Swarm Optimization
PID	Proportional Integral Derivative
D-H	Denavit-Hartenberg

### 1. Giới thiệu

Hiện nay, bộ điều khiển PID được ứng dụng rộng rãi trong các hệ thống do tính tiện lợi, dễ thiết kế và hiệu quả trong điều khiển [1-3]. Có nhiều phương pháp hiệu chỉnh hệ số của bộ điều khiển PID, nhưng phổ biến nhất là phương pháp Ziegler-Nichols [4]. Tuy nhiên, điều chỉnh hệ số này trong một số hệ thống sẽ phụ thuộc vào kinh nghiệm người thiết kế và mất nhiều thời gian. Thông thường, phương pháp Ziegler-Nichols xác định các thông số của bộ điều khiển dựa trên kết quả thực hiện của hệ thống. Tuy nhiên, do sai số của thiết bị đo và nhiễu ảnh hưởng lên tín hiệu đo, do đó làm cho bộ điều khiển PID trong quá trình hiệu chỉnh khó đạt được giá trị tối ưu. Vì vậy, trước khi áp dụng vào hệ thống cần thiết có một quá trình tinh chỉnh [5]. Vấn đề đặt ra là làm thế nào để hệ thống đạt được giá trị tối ưu dựa vào quá trình tinh chỉnh.

Việc tối ưu các thông số thiết kế đã được nhiều nhà khoa học nghiên cứu và ứng dụng [6, 7]. Trong đó, giải thuật bầy đàn (Particle Swarm Optimization, PSO) thường được sử dụng rộng rãi để tối ưu bộ điều khiển [8]. Thuật toán này để tìm kiếm lời giải tối ưu cho các thiết kế dựa trên không gian tìm kiếm, được xây dựng dựa trên khái niệm trí tuệ bầy đàn để thỏa mãn một hàm mục tiêu nào đó [9]. Tùy thuộc vào từng ứng dụng cụ thể, mà nhiều nghiên cứu áp dụng giải thuật PSO để tìm kiếm cấu trúc tối ưu cho bộ điều khiển PID đã được thực hiện [10, 11].

Bài báo đề xuất một phương pháp sử dụng giải thuật PSO để tìm kiếm hệ số tối ưu của bộ điều khiển PID. Chất lượng đáp ứng của hệ thống thiết kế được đánh giá dựa trên giá trị nhỏ nhất của hàm mục tiêu. Mô hình robot PUMA 560 được sử dụng để mô

<sup>1</sup> Ngày nhận bài: 15/12/2017; Ngày nhận bản sửa: 02/05/2018; Ngày chấp nhận: 12/05/2018; Phân biên: Nguyễn Hoàng Mai, Nguyễn Như Hiền

phòng, thiết kế và kiểm nghiệm chất lượng của giải thuật.

## 2. Phương pháp tìm kiếm hệ số bộ điều khiển PID và mô hình của đối tượng

### 2.1 Bộ điều khiển PID

Bộ điều khiển PID [12] biểu diễn trong **Error! Reference source not found.** được mô tả như sau:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (1)$$

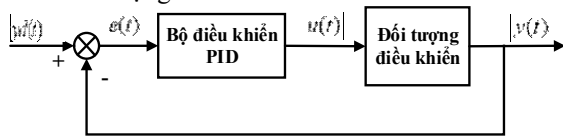
với:

$$e(t) = y_d(t) - y(t) \quad (2)$$

Hàm truyền của bộ điều khiển PID được diễn tả bởi công thức (3):

$$G_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \quad (3)$$

trong đó,  $K_p, K_i, K_d$  lần lượt là hệ số tỉ lệ, thời hằng tích phân và thời hằng vi phân,  $e(t)$  là sai số giữa ngõ ra và tín hiệu đặt,  $u(t)$  là ngõ ra của bộ điều khiển,  $y_d(t)$  và  $y(t)$  lần lượt là ngõ vào tham chiếu và ngõ ra của đối tượng.



H. 1 Cấu trúc điều khiển sử dụng bộ PID

Người thiết kế dựa trên mô hình để lựa chọn các thông số  $K_p, K_i, K_d$  để thỏa mãn các yêu cầu về chất lượng điều khiển. Cụ thể ở đây là tìm các hệ số của bộ điều khiển PID để đạt được chất lượng điều khiển mong muốn.

### 2.2 Giải thuật tối ưu PSO

PSO là một kỹ thuật tìm kiếm giá trị tối ưu dựa vào hành vi của các cá thể trong quần thể được đề xuất bởi Kennedy và Eberhart [13]. Các cá thể có khả năng hội tụ nhanh đến vị trí tối ưu cục bộ hoặc toàn cục.

Một quần thể PSO bao gồm các cá thể. Mỗi cá thể đại diện cho một giải pháp tiềm năng cho nhiệm vụ tối ưu. Tất cả các cá thể lần lượt có thể khám phá các giải pháp có thể xảy ra. Mỗi cá thể tạo ra một vị trí theo vận tốc, ghi nhớ vị trí trước đó và nó được so sánh vị trí tốt nhất được tạo ra bởi các cá thể trước đó theo hàm mục tiêu. Giải pháp tối ưu nhất sau đó được giữ lại. Mỗi cá thể tăng tốc theo vị trí tốt nhất không chỉ cục bộ mà còn tiến đến toàn cục. Nếu một cá thể có thể phát hiện ra một giải pháp có thể xảy ra, các cá thể khác sẽ di chuyển đến gần nó hơn để khám phá được khu vực tốt nhất [14].

Với  $N$  là số lượng các cá thể trong quần thể. PSO bao gồm ba thuộc tính: vị trí hiện tại tại  $a_{ij}$ , vận tốc hiện tại tại  $v_{ij}$  và vị trí tốt nhất mỗi cá thể  $Pb_{ij}$ . Mỗi cá thể trong quần thể lặp lại các thuộc tính ở trên cho đến khi theo hàm mục tiêu đạt giá trị nhỏ nhất, kích thước và vận tốc hiện tại của cá thể được cập nhật bởi phương trình (4):

$$v_{ij}(t+1) = wv_{ij}(t) + c_1 r_{1,i}(t)[Pb_{ij}(t) - a_{ij}(t)] + c_2 r_{2,i}(t)[Gb_i(t) - a_{ij}(t)] \quad (4)$$

trong đó,  $v_{ij}$  là vận tốc cá thể thứ  $i$  của quần thể  $j$ ,  $i \in 1 \dots N$ ;  $w$  là trọng số quán tính của vận tốc;  $c_1$  và  $c_2$  là hệ số gia tốc;  $r_1$  và  $r_2$  là hệ số kinh nghiệm và hệ số quan hệ xã hội, có giá trị ngẫu nhiên trong khoảng  $[0,1]$ .

Vị trí mới của cá thể được biểu diễn ở công thức (5):

$$a_{ij}(t+1) = a_{ij}(t) + v_{ij}(t+1) \quad (5)$$

Giải pháp tìm kiếm tốt nhất của cá thể được cập nhật bởi (6):

$$Pb_i(t+1) = \begin{cases} Pb_i(t) & \text{nếu } f(a_i(t+1)) \geq f(Pb_i(t)) \\ a_i(t+1) & \text{ngược lại} \end{cases} \quad (6)$$

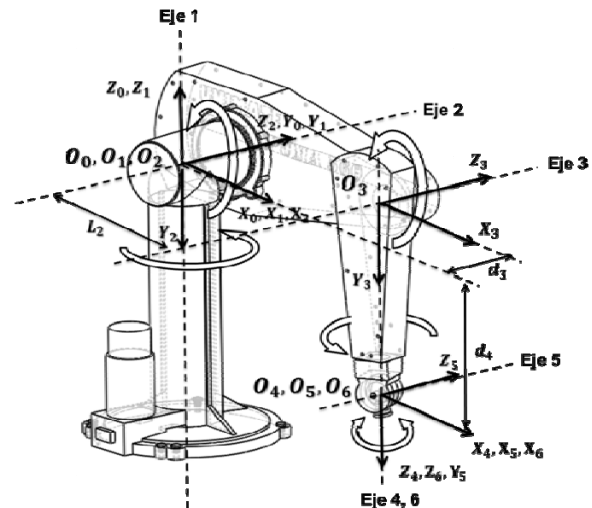
Giải pháp tối ưu toàn cục  $Gb$  được tổng hợp từ tất cả các cá thể qua ba bước trên được thể hiện như công thức (7):

$$Gb(t+1) = \arg \min_{Pb_i} f(Pb_i(t+1)), 1 \leq i \leq n \quad (7)$$

Mục tiêu của giải pháp đề xuất là giảm thiểu giá trị của  $f(e_1, e_2)$  trong (19) được cập nhật sau mỗi lần lặp. Trong trạng thái ban đầu của PSO, tất cả các vị trí của cá thể  $a_i$ , vận tốc  $v_i$ ,  $Pb_i$  và  $Gb_i$  được khởi tạo ngẫu nhiên trong khoảng  $[0,1]$ . Sau khi các cá thể di chuyển theo (4) mỗi cá thể sẽ tìm ra một giải pháp tiềm năng, vị trí mới tốt nhất trong quá khứ sẽ cập nhật bởi (5) và vị trí tốt của quần thể được cập nhật bởi (6). Các cá thể sẽ tiếp tục di chuyển để tìm một giải pháp tốt hơn cho đến khi nó đạt đến mục tiêu hoặc đáp ứng các điều kiện kết thúc, sau đó ta thu được kết quả tối ưu toàn cục theo công thức (7) [15].

### 2.3 Mô hình động lực học của robot PUMA 560

Cánh tay robot ba bậc tự do PUMA 560 [16] có cấu trúc tương tự như sáu bậc tự do, được dùng để kiểm nghiệm giải thuật điều khiển. Trong mô hình này (hình H. 2), ba khớp cuối được giữ cố định, còn ba khớp đầu chuyển động.



H. 2 Các khâu phối hợp của robot PUMA 560 [17]

Phương trình động lực học của cánh tay máy 3 khớp được miêu tả trên ma trận Lagrange-Euler hoặc Newton-Euler như sau (8):

$$\begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \end{bmatrix} = A(q) \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + B(q) \begin{bmatrix} \dot{q}_1 \dot{q}_2 \\ \dot{q}_1 \dot{q}_3 \\ \dot{q}_2 \dot{q}_3 \end{bmatrix} + C(q) \begin{bmatrix} \dot{q}_1^2 \\ \dot{q}_2^2 \\ \dot{q}_3^2 \end{bmatrix} + g(q) \quad (8)$$

Ma trận A là ma trận đối xứng 3x3 (9):

$$A(q) = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (9)$$

Khi đó, ma trận B được biểu diễn (10) là:

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} \\ 0 & 0 & b_{214} & 0 & 0 & b_{223} \\ 0 & 0 & b_{314} & 0 & 0 & 0 \\ b_{412} & b_{413} & 0 & b_{415} & 0 & 0 \\ 0 & 0 & b_{514} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (10)$$

Ma trận C được thể hiện ở (11):

$$C(q) = \begin{bmatrix} 0 & c_{12} & c_{13} \\ c_{21} & 0 & c_{23} \\ c_{31} & c_{32} & 0 \end{bmatrix} \quad (11)$$

và ma trận G là:

$$g(q) = \begin{bmatrix} 0 \\ g_2 \\ g_3 \end{bmatrix} \quad (12)$$

Gia tốc góc được thể hiện như công thức (13):

$$\ddot{q} = A^{-1}(q) \left\{ \Gamma - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)] \right\} \quad (13)$$

Khi đó:

$$I = \left\{ \Gamma - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)] \right\} \quad (14)$$

Kết hợp công thức (13) và (14) ta suy ra (15):

$$\ddot{q} = A^{-1}(q)I \quad (15)$$

Từ công thức (14) ta có:

$$I_1 = \Gamma_1 - [b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + b_{123}\dot{q}_2\dot{q}_3] - [c_{12}\dot{q}_2^2 + c_{13}\dot{q}_3^2] \quad (16)$$

$$I_2 = \Gamma_2 - [b_{223}\dot{q}_2\dot{q}_3] - [c_{21}\dot{q}_1^2 + c_{23}\dot{q}_3^2] - g_2 \quad (17)$$

$$I_3 = \Gamma_3 - [c_{31}\dot{q}_1^2 + c_{32}\dot{q}_2^2] - g_3 \quad (18)$$

Chú ý:

$$s_i = \sin(\theta_i), c_i = \cos(\theta_i), c_{ij} = \cos(\theta_i + \theta_j),$$

$$s_{ijk} = \sin(\theta_i + \theta_j + \theta_k), cc_i = \cos(\theta_i)\cos(\theta_i),$$

$$cc_i = \cos(\theta_i)\cos(\theta_i), cs_i = \cos(\theta_i)\sin(\theta_i)$$

### 3. Tối ưu bộ điều khiển PID bằng giải thuật bầy đàn

#### 3.1 Hàm mục tiêu

Để tìm các hệ số  $K_p, K_i, K_d$  của bộ điều khiển PID, công thức (19) được sử dụng. Trong quá trình tìm kiếm, giải thuật PSO tìm kiếm các cá thể tốt nhất dựa trên sai số nhỏ nhất của hàm mục tiêu.

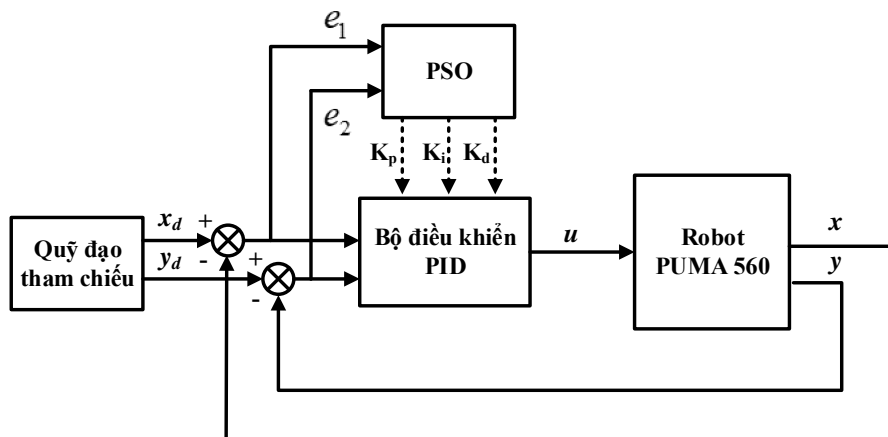
$$f(e_1, e_2) = t e_1(t) \frac{de_1}{dt} + \alpha t e_2(t) \frac{de_2}{dt} \quad (19)$$

trong đó:  $e_1, e_2$  là sai số của bộ điều khiển thứ 1 và 2,  $\alpha$  là hệ số khảo sát.

Sơ đồ bộ điều khiển PID được cấu trúc bằng giải thuật PSO trong hình H. 3. Trong đó, ngõ vào đặt cho robot là quỹ đạo  $x_d$  và  $y_d$ , sai số giữa ngõ vào đặt và ra thực tế là  $e_1$  và  $e_2$  được sử dụng để tính toán hàm mục tiêu.

#### 3.2 Tìm kiếm hệ số PID sử dụng thuật toán bầy đàn

Giải thuật PSO dựa vào giá trị của hàm mục tiêu để lựa chọn các thông số tối ưu nhất cho bộ điều khiển PID.

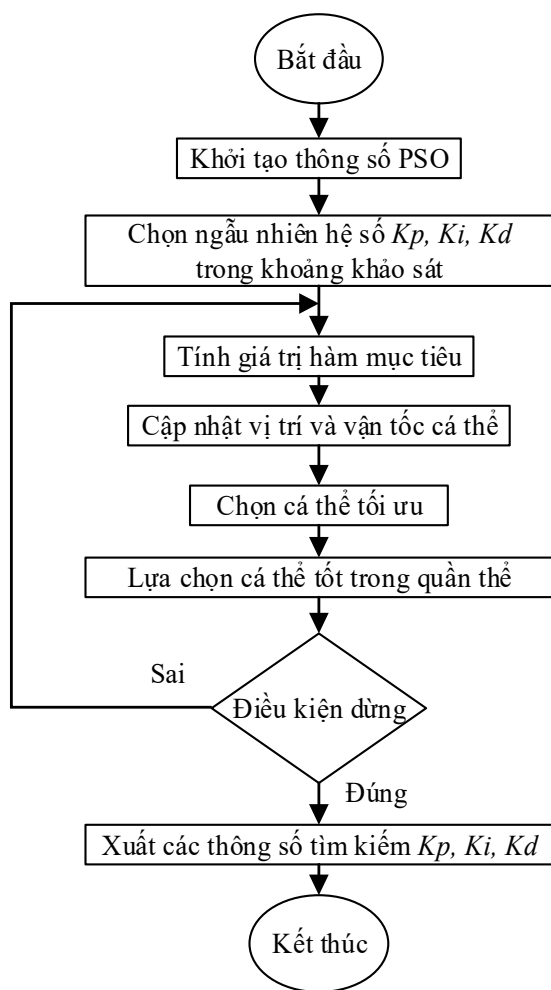


H. 3 Sơ đồ bộ điều khiển PID cấu trúc bằng PSO

Thuật toán tìm kiếm các hệ số của bộ điều khiển PID [18] được chia thành các bước như sau:

- **Bước 1:** Khởi tạo các thông số PSO.
- **Bước 2:** Chọn ngẫu nhiên các hệ số  $K_p, K_i, K_d$  trong khoảng khảo sát.
- **Bước 3:** Tính hàm mục tiêu dựa trên sai số hồi tiếp trong bộ điều khiển PID.
- **Bước 4:** Cập nhật vị trí và vận tốc của cá thể trong quần thể tìm kiếm.
- **Bước 5:** Chọn ra cá thể tối ưu trong mỗi lần lặp (tối ưu cục bộ).
- **Bước 6:** Lựa chọn cá thể tốt nhất trong quần thể sau tất cả vòng lặp (tối ưu toàn cục).
- **Bước 7:** Thoát khỏi vòng lặp nếu thỏa điều kiện.
- **Bước 8:** Dựa vào tối ưu toàn cục để lựa chọn các thông số  $K_p, K_i, K_d$  cho quá trình mô phỏng của robot.

Lưu đồ giải thuật của quá trình tìm kiếm được thể hiện trong hình H. 4. Trong đó, giải thuật bày dàn để tìm kiếm hệ số tối ưu trong bộ điều khiển PID dựa vào giá trị của hàm mục tiêu. Điều kiện dừng là số vòng lặp, được lựa chọn dựa vào kinh nghiệm trong quá trình tìm kiếm.



H. 4 Lưu đồ tìm kiếm thông số bộ điều khiển PID bằng PSO

Ban đầu sẽ khởi tạo các thông số của PSO, sau đó chương trình sẽ chọn các hệ số ngẫu nhiên của PID trong khoảng giới hạn. Chương trình sẽ bắt đầu với vòng lặp đầu tiên trong khoảng 1,7 giây để lựa chọn các cá thể tối ưu trong quần thể. Vòng lặp vẫn tiếp tục diễn ra sau đó đến khi thỏa mãn điều kiện dừng (khảo sát ở 20 vòng). Cuối cùng chương trình sẽ chọn ra các cá thể tối ưu tiệm cận rồi gán cho các hệ số PID cần tìm.

### 3.3 Các thông số mô phỏng

Các thông số D-H cơ bản [19] trong mô phỏng của robot PUMA 560 được trình bày trong B.1. Trong đó, phương trình động lực học xây dựng chưa có các thiết bị truyền động. Ngoài ra, còn nhiều chi tiết, đặc biệt là các hệ số truyền động đã được lược bỏ vì thiếu không gian.

#### B.1 Thông số D-H của robot PUMA 560 [17]

Khâu	$\alpha_{i-1}$	$\theta_i$	$a_{i-1}$	$d_i$
1	0	$q_1$	0	0
2	-90	$q_2$	0	0.2435
3	0	$q_3$	0.4318	-0.0934

Để tìm phương trình động học ba bậc tự do của robot, một ma trận chuyển đổi và hệ tọa độ D-H mới được thành lập dựa trên thông số sáu khớp tự do. Tuy nhiên, trong phương trình động lực học  $q_4, q_5$  và  $q_6$  được đặt bằng 0. Hệ số của các khớp được trình bày trong B.2 và

#### B.3.

#### B.2 Hằng số quán tính của robot (kg.m<sup>2</sup>)[17]

$I_1 = 1.43 \pm 0.05$	$I_2 = 1.75 \pm 0.07$
$I_3 = 1.38 \pm 0.05$	$I_4 = 0.69 \pm 0.02$
$I_5 = 0.372 \pm 0.031$	$I_6 = 0.333 \pm 0.016$
$I_7 = 0.298 \pm 0.029$	$I_8 = -0.134 \pm 0.014$
$I_9 = 0.0238 \pm 0.012$	$I_{10} = -0.0213 \pm 0.0022$
$I_{11} = -0.0142 \pm 0.0070$	$I_{12} = -0.011 \pm 0.0011$
$I_{13} = -0.00379 \pm 0.0009$	$I_{14} = 0.00164 \pm 0.00007$
$I_{15} = 0.00125 \pm 0.0003$	$I_{16} = 0.00124 \pm 0.0003$
$I_{17} = 0.000642 \pm 0.0003$	$I_{18} = 0.000431 \pm 0.00013$
$I_{19} = 0.0003 \pm 0.0014$	$I_{20} = -0.000202 \pm 0.0008$
$I_{21} = -0.0001 \pm 0.0006$	$I_{22} = -0.000058 \pm 0.000015$
$I_{23} = 0.00004 \pm 0.00002$	$I_{m1} = 1.14 \pm 0.27$
$I_{m2} = 4.71 \pm 0.54$	$I_{m3} = 0.827 \pm 0.093$

#### B.3 Hằng số trọng lực (N.m) [17]

$g_1 = -37.2 \pm 0.5$	$g_2 = -8.44 \pm 0.20$
$g_3 = 1.02 \pm 0.50$	$g_4 = 0.249 \pm 0.025$
$g_5 = -0.0282 \pm 0.0056$	

Các thông số khởi tạo của PSO [8] được trình bày trong bảng B.4.

**B.4 Thông số PSO tìm kiếm hệ số PID [8]**

Thông số thuật toán PSO	Giá trị
Số cá thể trong quần thể	6
Kích thước quần thể	20
Số lần lặp	20
Trọng số quán tính	0.6
Hệ số kinh nghiệm	2
Hệ số quan hệ xã hội	2

Số vòng lặp được chọn dựa vào kinh nghiệm trong quá trình khảo sát. Các cá thể là các hệ số tìm kiếm của bộ điều khiển PID, trọng số quán tính, hệ số kinh nghiệm và hệ số quan hệ xã hội là các thông số mặc định trong giải thuật bầy đàn.

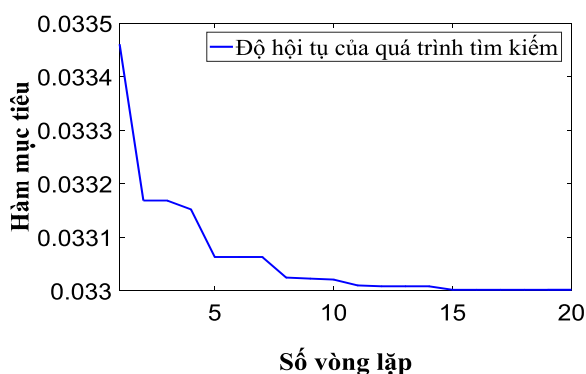
Kết quả tìm kiếm thông số  $K_p, K_i, K_d$  trong bộ điều khiển PID sử dụng giải thuật bầy đàn được trình bày trong B.5.

**B.5 Thông số bộ điều khiển PID**

Các hệ số tìm kiếm	Giá trị
$K_{p1}$	182,87
$K_{i1}$	998,56
$K_{d1}$	13,9
$K_{p2}$	119,72
$K_{i2}$	1681,8
$K_{d2}$	13,84

**3.4 Kết quả mô phỏng**

Độ hội tụ của tìm kiếm thông số bộ điều khiển PID bằng giải thuật bầy đàn được thể hiện trong hình H. 5. Cho thấy các hệ số tìm kiếm được tối ưu tiệm cận sau số lần lặp dựa vào hàm mục tiêu.

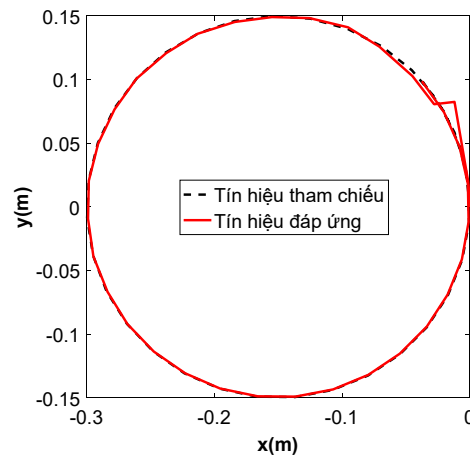


**H. 5 Độ hội tụ tìm kiếm hệ số PID bằng giải thuật bầy đàn**

Khảo sát tín hiệu đáp của bộ điều khiển PID được chỉnh định bằng giải thuật bầy đàn trên mô hình robot PUMA 560. Ngõ vào đặt là quỹ đạo tròn có hệ phương trình như (20):

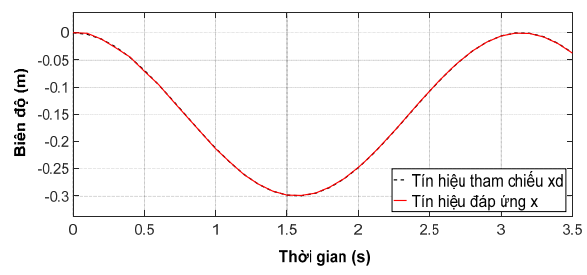
$$\begin{cases} x_d = 0.15 \cos(2t) - 0.15 \\ y_d = 0.15 \sin(2t) \end{cases} \quad (m) \quad (20)$$

Tín hiệu đáp ứng của mô phỏng được thể hiện trong hình H. 6, H. 7 và H. 8 cho thấy, robot bám quỹ đạo tốt với thời gian tăng khoảng  $0.3 \pm 0.05s$  và thời gian xác lập nhỏ khoảng  $0.5 \pm 0.05s$ .

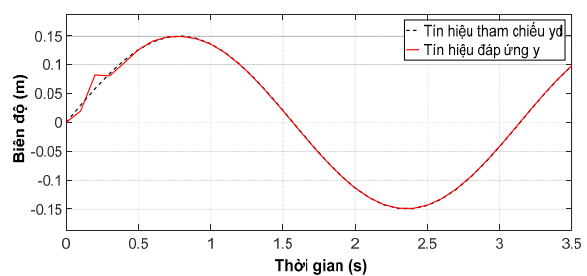


**H. 6 Tín hiệu bám quỹ đạo tròn của robot**

Sau kết quả mô phỏng robot bám quỹ đạo với tín hiệu đặt hình tròn (hình H. 6), sai số toàn phương trung bình của hệ thống là  $mse = 2,27 \times 10^{-6}$ .



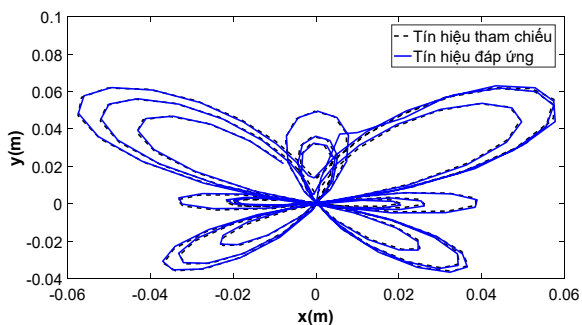
**H. 7 Tín hiệu đáp ứng của x**



**H. 8 Tín hiệu đáp ứng của y**

Đáp ứng của bộ điều khiển PID được điều chỉnh bằng giải thuật bầy đàn trên quỹ đạo hình bướm với hệ phương trình (21):

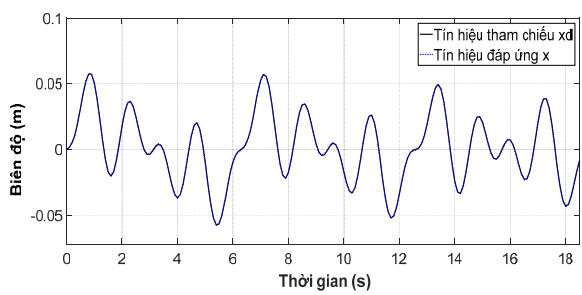
$$\begin{cases} x_d = 0.02 \cos(t) \left[ e^{\cos t} - 2 \cdot \cos(4t) - \sin^5 \left( \frac{t}{12} \right) \right] \\ y_d = 0.02 \sin(t) \left[ e^{\cos t} - 2 \cdot \cos(4t) - \sin^5 \left( \frac{t}{12} \right) \right] \end{cases} \quad (m) \quad (21)$$



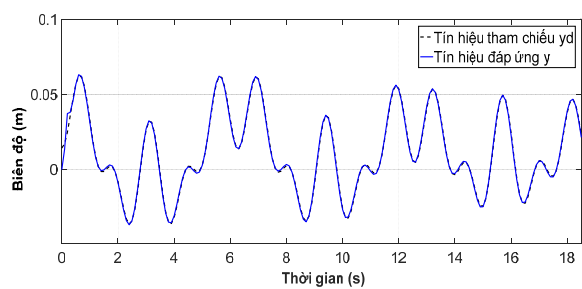
H. 9 Tín hiệu bám quỹ đạo hình bướm của robot

Kết quả mô phỏng ứng với quỹ đạo đặt hình bướm (hình H. 9), sai số toàn phương trung bình của hệ thống là  $mse = 3,09 \times 10^{-7}$ .

Hình **Error! Reference source not found.** trình bày đáp ứng bám quỹ đạo hình bướm ứng với bộ điều khiển PID được đặt bởi thông số trong Bảng 5. Từ các kết quả thể hiện trên hình H. 9, H. 10 và H. 11 ta thấy giải thuật đã tìm kiếm được các hệ số tốt cho bộ điều khiển PID. Kết quả mô phỏng cho thấy thời gian tăng nhanh khoảng  $0.4 \pm 0.05s$  và thời gian xác lập ngắn khoảng  $0.7 \pm 0.05s$ .



H. 10 Tín hiệu đáp ứng của x



H. 11 Tín hiệu đáp ứng của y

#### 4. Kết luận

Bài báo đã đề xuất một phương pháp tìm kiếm hệ số tối ưu cho bộ điều khiển PID bằng giải thuật bầy đàn, khảo sát trên đối tượng cánh tay robot ba bậc tự do. Phương pháp này có ưu điểm là thiết kế được bộ điều khiển tối ưu mà không cần quan tâm đến mô hình toán của đối tượng. Vì trong quá trình thiết kế bộ điều khiển, giải thuật tối ưu bầy đàn đã dựa vào tín hiệu vào ra để tìm kiếm các hệ số thích hợp. Do đó, nhược điểm của phương pháp này là tốn nhiều thời gian để thực hiện vòng lặp. Trong thực tiễn, điều này rất ít được áp dụng. Vì thế, nếu muốn áp dụng thực tế, hệ

thống cần có thêm thiết bị đo đặc tín hiệu. Từ đó, bộ điều khiển thời gian thực trở nên khả thi cho nhiều đối tượng bằng cách kết nối thiết bị đo đặc với máy tính.

#### Tài liệu tham khảo

- [1] Lei G (2017) *Application of improved PID control technology in hydraulic turbine governing systems*. 3rd Intern. Conf. on Control, Automation and Robotics (ICCAR), pp. 275-279
- [2] Park CY, Jung Mg, Kim HY, Lee Mc (2017) *Development of 3D printing simulator nozzle system using PID control for building construction*. 14th Intern. Conf. on Ubiquitous Robots and Ambient Intelligence (URAI), pp. 368-369
- [3] Wei L, Ma B (2017) *Application of improved gravitational search algorithm in PID control for boiler drum water level*. 29th Chinese Conf. on Control and Decision (CCCD), pp. 1852-1857
- [4] Anto EK, Asumadu LA, Okyere PY (2016) *PID control for improving P&O-MPPT performance of a grid-connected solar PV system with Ziegler-Nichols tuning method*. IEEE 11th Conf. on Industrial Electronics and Applications (ICIEA), pp. 1847-1852
- [5] Meshram P, Kanojiya RG (2012) *Tuning of PID controller using Ziegler-Nichols method for speed control of DC motor*. Intern. Conf. on Advances in Engineering, Science and Management (ICAESM), pp. 117-122
- [6] Bhambhani A, Shah P (2016) *PID parameter optimization using Cohort intelligence technique for DC motor control system*. Intern. Conf. on Automatic Control and Dynamic Optimization Techniques (ICACDOT), pp. 465-468
- [7] Dewantoro G, Yohanes BW (2016) *A comparative study of population-based optimizations for tuning PID parameters*. 2nd Intern. Conf. on Control Science and Systems Engineering (ICCSSE), pp. 133-136
- [8] Clerc M (2010) *Particle swarm optimization*. vol. 93, John Wiley & Sons
- [9] Zhang Y, Wang S, Ji G (2015) *A comprehensive survey on particle swarm optimization algorithm and its applications*. Mathematical Problems in Engineering, vol. 2015
- [10] Fan X, Cao J, Yang H, Dong X, Liu C, Gong Z, et al (2013) *Optimization of PID parameters based on improved particle-swarm-optimization*. Intern. Conf. on Information Science and Cloud Computing Companion (ISCC-C), pp. 393-397
- [11] Xin W, Ran L, Yanghua W, Yong P, Bin Q (2013) *Self-tuning PID controller with variable parameters based on particle swarm optimization*. 3rd Intern. Conf. on Intelligent System Design and Engineering Applications (ISDEA), pp. 1264-1267

- [12] Åström KJ, Hägglund T (1995) *PID controllers: theory, design, and tuning*. vol. 2, Isa Research Triangle Park, NC.
- [13] Eberhart R, Kennedy J (1995) *A new optimizer using particle swarm theory*. Proc. of the 6th Intern. Symp. on Micro Machine and Human Science MMHS'95, pp. 39-43
- [14] Gudise VG, Venayagamoorthy GK (2003) *Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks*. Proc. of the IEEE Swarm Intelligence Symposium SIS'03, pp. 110-117
- [15] Lin CL, Hsieh ST, Liu CC (2005) *PSO-based learning rate adjustment for blind source separation*. Proc. of Intern. Symp. on Intelligent Signal Processing and Communication Systems ISPACS, pp. 181-184
- [16] Armstrong B, Khatib O, Burdick J (1986) *The explicit dynamic model and inertial parameters of the PUMA 560 arm*. Proc. of IEEE Intern. Conf. on Robotics and Automation, pp. 510-518
- [17] <https://grabcad.com/library/robot-puma-560>, Truy cập ngày: 29/04/2018
- [18] Chen J, Omidvar MN, Azad M, Yao X (2017) *Knowledge-based particle swarm optimization for PID controller tuning*. Proc. of IEEE Congress on Evolutionary Computation (CEC), pp. 1819-1826
- [19] Merheb AR, (2008) *Nonlinear Control Algorithms applied to 3 DOF PUMA Robot*. METU