

ĐIỀU KHIỂN RÔBÔT THEO NGUYÊN LÝ TRƯỢT SỬ DỤNG MẠNG NƠON

NGUYỄN TRẦN HIỆP¹, PHẠM THƯỢNG CÁT²

¹Học viện Kỹ thuật Quân sự, Bộ Quốc Phòng

²Viện Công nghệ thông tin, Viện Khoa học và Công nghệ Việt nam

Abstract. This paper proposes a neural sliding mode control method for robot tracking problem to overcome the noises and large uncertainties in robot dynamics. The Lyapunov direct method has been used to prove the stability of the overall system. Simulation results are given to illustrate the applicability of the proposed method.

Tóm tắt. Điều khiển trượt là một phương pháp điều khiển được dùng phổ biến trong trong khiển rôbôt. Nhược điểm lớn nhất của phương pháp điều khiển này là xuất hiện dao động xung quanh mặt trượt. Bài báo này đưa ra một phương pháp khử được những dao động đó và tăng độ chính xác của điều khiển bằng xấp xỉ các thành phần phi tuyến bất định của rôbôt. Tiêu chuẩn ổn định Lyapunov được sử dụng để chứng minh tính ổn định toàn cục của hệ thống. Các kết quả mô phỏng cũng minh họa cho khả năng ứng dụng của phương pháp trên.

1. ĐẶT VẤN ĐỀ

Rôbôt là một lĩnh vực có sự phát triển liên tục suốt năm mươi năm qua [1, 18, 19]. Hệ phương trình động lực học của rôbôt là một hệ phi tuyến cao với nhiều yếu tố bất định. Trong nhiều ứng dụng, rôbôt phải chuyển động nhanh từ một vị trí này sang vị trí khác hoặc bám theo một quỹ đạo mong muốn trong không gian ba chiều với độ chính xác cao. Các phương pháp điều khiển cổ điển như PID, phương pháp tính moment hay điều khiển tối ưu [3, 18] lúc này không cho kết quả khả quan do các phương pháp này thường đòi hỏi phải biết chính xác các tham số của hệ động lực rôbôt. Điều khiển trượt [17] là một phương pháp điều khiển có tính bền vững cao và đã được nhiều tác giả áp dụng cho bài toán điều khiển rôbôt [10]. Tuy nhiên nhược điểm chính của phương pháp điều khiển trượt là tự dao động xung quanh mặt trượt về cả biên độ và tần số đều không xác định được. Để giảm bớt tác động xấu này, nhiều tác giả đã đề xuất nhiều phương pháp khác nhau như đưa thành phần tích phân vào mặt trượt, sử dụng hàm bão hòa ($sat(s)$) hay hàm sigmoid thay thế hàm dấu ($sign(s)$), nhận dạng hoặc xấp xỉ các thành phần bất định của rôbôt [5, 7, 8].

Mạng nơon đã được sử dụng rộng rãi trong lĩnh vực điều khiển do các tính chất nổi trội của nó như là sử lý song song, khả năng tự học và tự thích nghi cao. Mạng nơon có khả năng xấp xỉ các hàm phi tuyến từ các dữ liệu thực nghiệm mà không cần biết chính xác các tham số và cấu trúc của hàm đó [4, 6]. Nhiều công trình nghiên cứu đã sử dụng tính chất này của mạng nơon để xấp xỉ các đặc tính phi tuyến bất định của rôbôt và minh chứng qua các

kết quả mô phỏng [13, 14]. Tuy nhiên phần lớn các đề xuất không đưa ra được lý luận chặt chẽ chứng minh tính ổn định của toàn cục của hệ điều khiển rôbôt sử dụng mạng nơon.

Bài báo này đưa ra một phương pháp mới điều khiển rôbôt bám theo quỹ đạo mong muốn theo nguyên lý trượt có sử dụng mạng nơon để xấp xỉ các thành phần không xác định. Độ ổn định toàn cục của hệ thống được chứng minh bằng phương pháp ổn định Lyapunov. Bài báo được chia thành 6 phần. Phần một đặt vấn đề về điều khiển rôbôt theo nguyên lý trượt. Phần hai mô tả bài toán cần giải quyết. Phần ba đưa ra cấu trúc của hệ thống bao gồm khối tạo mặt trượt PD, khối bù phi tuyến, khối tính toán thuật điều khiển và khối mạng nơon tự học xấp xỉ các thành phần phi tuyến bất định. Tính ổn định toàn cục của hệ thống được chứng minh trong phần này. Phần bốn của bài báo đưa ra một số đánh giá về độ chính xác và tốc độ hội tụ của phương pháp đề xuất. Phần 5 là kết quả mô phỏng. Phần cuối tóm tắt một số kết luận của bài báo.

2. BÀI TOÁN ĐIỀU KHIỂN RÔBÔT BẮM THEO QUỸ ĐẠO VỚI NHIỀU THÀNH PHẦN PHI TUYẾN BẤT ĐỊNH

Phương trình động lực học của một hệ rôbôt n khớp cứng có thể được mô tả bằng phương trình vi phân phi tuyến MIMO như sau:

$$\hat{M}(\mathbf{q})\ddot{\mathbf{q}} + \hat{B}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{g}(\mathbf{q}) + \mathbf{d}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau}, \quad (1)$$

trong đó, $\mathbf{q} = [q_1, q_2, \dots, q_n]^T$, $\dot{\mathbf{q}} = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n]^T$, $\ddot{\mathbf{q}} = [\ddot{q}_1, \ddot{q}_2, \dots, \ddot{q}_n]^T$ là các vector $n \times 1$ biểu diễn vị trí, vận tốc và gia tốc góc của các khớp tương ứng.

$\boldsymbol{\tau} = [\tau_1, \tau_2, \dots, \tau_n]^T$ là vector $n \times 1$ biểu diễn momen tác động lên các khớp.

$\hat{M}(\mathbf{q}) \in \mathbf{R}^{n \times n}$ là ma trận quán tính.

$\hat{B}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbf{R}^{n \times n}$ là ma trận hệ số Coriolis và lực hướng tâm.

$\hat{g}(\mathbf{q}) \in \mathbf{R}^{n \times 1}$ là gia tốc trọng trường.

$\mathbf{d}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbf{R}^{n \times 1}$ là thành phần lực ma sát và nhiễu tác động lên các khớp của rôbôt.

Các đại lượng $\hat{M}(\mathbf{q})$, $\hat{B}(\mathbf{q}, \dot{\mathbf{q}})$, $\hat{g}(\mathbf{q})$ thường không xác định được chính xác. Ta có thể mô tả các đại lượng này như sau:

$$\hat{M}(\mathbf{q}) = \mathbf{M}(\mathbf{q}) + \Delta\mathbf{M}(\mathbf{q}), \quad (2a)$$

$$\hat{B}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}) + \Delta\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}), \quad (2b)$$

$$\hat{g}(\mathbf{q}) = \mathbf{g}(\mathbf{q}) + \Delta\mathbf{g}(\mathbf{q}), \quad (2c)$$

trong đó $\mathbf{M}(\mathbf{q})$, $\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})$, $\mathbf{g}(\mathbf{q})$, là các thành phần xác định được, $\Delta\mathbf{M}(\mathbf{q})$, $\Delta\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})$, $\Delta\mathbf{g}(\mathbf{q})$ là các thành phần không biết do tính bất định của mô hình rôbôt. Giả thiết $\|\Delta\mathbf{M}(\mathbf{q})\| < m_0$, $\|\Delta\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})\| < b_0$, $\|\Delta\mathbf{g}(\mathbf{q})\| < g_0$, m_0, b_0, g_0 là các giá trị được biết). Phương trình (1) có thể được biểu diễn dưới dạng:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau}, \quad (3a)$$

$$\boldsymbol{\tau} = \boldsymbol{\tau}_0 + \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}), \quad (3d)$$

$$\boldsymbol{\tau}_0 = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}), \quad (3e)$$

$$\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) = \Delta\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \Delta\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \Delta\mathbf{g}(\mathbf{q}) + \mathbf{d}(\mathbf{q}, \dot{\mathbf{q}}), \quad (3e)$$

$\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbf{R}^{n \times 1}$ trong (3) là tổng hợp các thành phần không xác định của hệ động lực, ma sát, nhiễu loạn tác động lên rôbot và bị chặn bởi $\|\mathbf{f}\| \leq f_0$ với f_0 có thể xác định được.

Để xây dựng thuật điều khiển thì các tính chất quan trọng sau đây của hệ động lực rôbot được sử dụng:

- Ma trận quán tính $\mathbf{M}(\mathbf{q})$ là đối xứng và xác định dương.
- Ma trận $(\dot{\mathbf{M}}(\mathbf{q}) - 2\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}))$ là đối xứng lệch hay

$$\mathbf{x}^T [\dot{\mathbf{M}}(\mathbf{q}) - 2\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})]\mathbf{x} = 0 \quad \forall \mathbf{x} \in \mathbf{R}^{n \times 1} \Rightarrow \mathbf{x}^T \dot{\mathbf{M}}(\mathbf{q})\mathbf{x} = 2\mathbf{x}^T \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{x}. \quad (4)$$

Mục đích của điều khiển là xác định tín hiệu điều khiển $\boldsymbol{\tau}$ sao cho rôbot bám được quỹ đạo \mathbf{q}_d mong muốn, nghĩa là sai lệch $\mathbf{e} = (\mathbf{q} - \mathbf{q}_d) \rightarrow 0$, $\dot{\mathbf{e}} = (\dot{\mathbf{q}} - \dot{\mathbf{q}}_d) \rightarrow 0$. Ở đây $\mathbf{e}, \dot{\mathbf{e}}$ là các giá trị sai số về vị trí, vận tốc.

3. XÁC ĐỊNH THUẬT ĐIỀU KHIỂN RÔBOT THEO NGUYÊN LÝ TRƯỢT SỬ DỤNG MẠNG NƠON

Các hệ điều khiển theo phương pháp trượt có khả năng điều khiển các hệ phi tuyến, các hệ MIMO có nhiều tham số bất định. Để sử dụng điều khiển trượt, cho hệ rôbot (1) việc đầu tiên phải xác định mặt trượt. Ở đây mặt trượt được chọn dưới dạng PD:

$$\mathbf{s}(t) = \mathbf{C}\mathbf{e} + \dot{\mathbf{e}}, \quad (5)$$

\mathbf{C} thường được chọn là ma trận đường chéo xác định dương và $\mathbf{s} = [s_1, s_2, \dots, s_n]^T$. Phương trình (5) cho thấy quan hệ nhất quán giữa $(\mathbf{e}, \dot{\mathbf{e}})$ và \mathbf{s} . Do đó phương trình (3b) có thể viết dưới dạng:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_0 + \mathbf{f}(\mathbf{s}). \quad (6)$$

Sự bất định của $\mathbf{f}(\mathbf{s})$ là nguyên nhân chính làm giảm chất lượng của quá trình điều khiển. Nếu chúng ta thực hiện bù thành phần phi tuyến bất định này thì chất lượng của điều khiển có thể được cải thiện. Theo định lý Stone-Weierstrass [2] ta có thể chọn được một mạng nơon nhân tạo (ANN) với số nút nơon hữu hạn có khả năng xấp xỉ một hàm phi tuyến với độ chính xác cho trước. Để xấp xỉ hàm $\mathbf{f}(\mathbf{s})$ ta chọn cấu trúc mạng đơn giản như sau:

$$\mathbf{f}(\mathbf{s}) = \mathbf{W}\boldsymbol{\sigma} + \boldsymbol{\varepsilon}, \quad (7)$$

hay

$$\mathbf{f}(\mathbf{s}) = \hat{\mathbf{f}} + \boldsymbol{\varepsilon}, \quad (7b)$$

trong đó $\hat{\mathbf{f}} = [\hat{f}_1, \hat{f}_2, \dots, \hat{f}_n]^T = \mathbf{W}\boldsymbol{\sigma}$ là thành phần xấp xỉ của $\mathbf{f}(\mathbf{s})$, $\boldsymbol{\varepsilon}$ là sai số của phép xấp xỉ.

Với $\|\mathbf{f}(\mathbf{s})\| < f_0$ ta có thể xác định được giới hạn ε_0 của $\boldsymbol{\varepsilon}$: $\|\boldsymbol{\varepsilon}\| \leq \varepsilon_0$.

Đặt w_i là vector cột thứ i của ma trận \mathbf{W} ta có:

$$\hat{\mathbf{f}} = \mathbf{W}\boldsymbol{\sigma} = [w_1, w_2, \dots, w_n]\boldsymbol{\sigma}. \quad (8)$$

Với rôbot có n khớp, ta xây dựng mạng nơon có cấu trúc một lớp ẩn như hình 1.

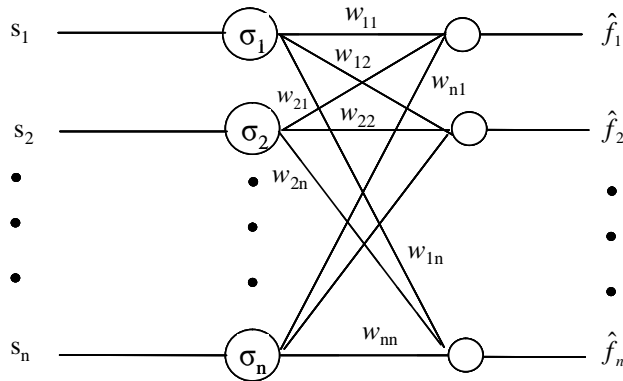
Đây là cấu trúc mạng nơon nhân tạo dạng RBF (Radial Basic Function). Cấu trúc này đã được chứng minh là thoả mãn định lý Stone-Weierstrass [2]. Chọn hàm tác động σ_i dạng phân bố Gauss ta có:

$$\sigma_i = \exp\left(-\frac{(s_i - c_i)^2}{\lambda_i^2}\right), \quad (9a)$$

trong đó c_i là trọng tâm, và λ_i là tham số chuẩn hoá có thể tuỳ chọn. Lúc này hàm xấp xỉ của $f(\mathbf{s})$ có dạng:

$$\hat{f}_i = \sum_{j=1}^n w_{ij} \sigma_j \text{ với } i, j = 1, 2, \dots, n, \quad (9b)$$

w_{ij} là các trọng số của mạng nơron xấp xỉ.



Hình 1. Mạng Nơron dùng để xấp xỉ các thành phần phi tuyến bất định

Bài toán điều khiển chính là tìm được thuật điều khiển τ với thuật học w_{ij} của mạng nơron (8) sao cho $\mathbf{s} \rightarrow 0$ kéo theo $\mathbf{C}\mathbf{e} + \dot{\mathbf{e}} = 0$ và hệ thống sẽ trượt về gốc tọa độ $\mathbf{e} = 0$ đảm bảo $\mathbf{q}(t) \rightarrow \mathbf{q}_d$.

Định lý. Hệ động lực rôbôt n bậc tự do (1) với mạng nơron (8) và mặt trượt (5) sẽ bám theo quỹ đạo mong muốn \mathbf{q}_d với sai số $\mathbf{e} = (\mathbf{q} - \mathbf{q}_d) \rightarrow 0$ nếu ta chọn thuật điều khiển momen τ và thuật học w_i của mạng nơron như sau:

$$\tau = \mathbf{M}\ddot{\mathbf{q}}_d + \mathbf{B}\dot{\mathbf{q}}_d + \mathbf{g} - \mathbf{M}\mathbf{C}\dot{\mathbf{e}} - \mathbf{B}\mathbf{C}\mathbf{e} - \mathbf{K}\mathbf{s} - \gamma \frac{\mathbf{s}}{\|\mathbf{s}\|} + (1 + \eta)\mathbf{W}\boldsymbol{\sigma}, \quad (10)$$

$$\dot{w}_i = -\eta \mathbf{s} \sigma_i, \quad (11)$$

trong đó các tham số tự chọn $\mathbf{K} = \mathbf{K}^T > 0$ là ma trận đối xứng xác định dương, $\eta, \gamma > 0$. Như vậy cấu trúc của hệ điều khiển có thể mô tả theo sơ đồ trên Hình 2.

Momen τ gồm ba thành phần chính: thành phần bù phi tuyến $\tau_{ff} = \mathbf{M}\ddot{\mathbf{q}}_d + \mathbf{B}\dot{\mathbf{q}}_d + \mathbf{g} - \mathbf{M}\mathbf{C}\dot{\mathbf{e}} - \mathbf{B}\mathbf{C}\mathbf{e}$, thành phần trượt $\tau_s = -\mathbf{K}\mathbf{s} - \gamma \frac{\mathbf{s}}{\|\mathbf{s}\|}$ và thành phần xấp xỉ $\hat{\mathbf{f}}$ là mạng nơron có thuật học online.

Định lý này được chứng minh bằng nguyên lý ổn định Lyapunov đảm bảo ổn định toàn cục như sau.

Chứng minh. Chọn hàm:

$$V = \frac{1}{2}[\mathbf{s}^T \mathbf{M} \mathbf{s} + \sum_{i=1}^n w_i^T w_i]. \quad (12)$$

Vì \mathbf{M} là ma trận quán tính của rôbôt có tính chất đối xứng xác định dương nên ta có $V > 0$ với tất cả $(\mathbf{s}^T, \mathbf{w}^T) \neq 0$ và $V = 0$ khi $(\mathbf{s}^T, \mathbf{w}^T) = 0$. Hàm V thoả mãn các điều kiện khác của nguyên lý ổn định Lyapunov như $V \rightarrow \infty$ khi $\mathbf{s} \rightarrow \infty, w_i \rightarrow \infty$. Nếu ta xác định

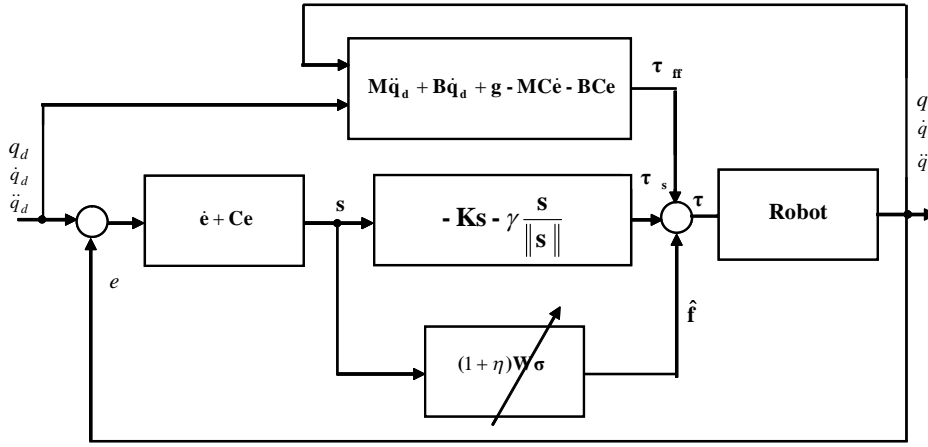
được thuật điều khiển τ sao cho $\dot{V} < 0$ thì theo nguyên lý ổn định Lyapunov $s \rightarrow \infty$ hay hệ thống sẽ tiếp cận và duy trì trên mặt trượt. Tính ổn định toàn cục của hệ thống sẽ được bảo đảm.

Lấy đạo hàm V theo thời gian ta nhận được

$$\dot{V} = \frac{1}{2}[\dot{s}^T M s + s^T \dot{M} s + s^T M \dot{s} + \sum_{i=1}^n (\dot{w}_i^T w_i + w_i^T \dot{w}_i)] = s^T M \dot{s} + \frac{1}{2} s^T \dot{M} s + \sum_{i=1}^n (w_i^T \dot{w}_i). \quad (13)$$

Vì ma trận $(\dot{M}(q) - 2B(q, \dot{q}))$ là ma trận đối xứng lệch do đó $s^T \dot{M} s = 2s^T B s$ nên

$$\dot{V} = s^T M \dot{s} + s^T B s + \sum_{i=1}^n w_i^T \dot{w}_i. \quad (14)$$



Hình 2. Sơ đồ cấu trúc hệ điều khiển trượt sử dụng mạng nơron

Mặt khác ta có

$$M\dot{s} + Bs = M(\ddot{e} + C\dot{e}) + B(\dot{e} + Ce) = M(-\ddot{q}_d + C\dot{e}) + B(-\dot{q}_d + Ce) + B\dot{q} + M\ddot{q}. \quad (15a)$$

Từ phương trình (1), (3a) ta rút ra

$$B\dot{q} + M\ddot{q} = \tau - g(q) - f(s). \quad (15b)$$

Thay (15a), (15b), (7b) vào (14) và sau khi biến đổi ta có

$$\dot{V} = s^T [-M\ddot{q}_d + MC\dot{e} - B\dot{q}_d + BCe + \tau - g - W\sigma - \varepsilon] + \sum_{i=1}^n w_i^T \dot{w}_i. \quad (16)$$

Thay τ được chọn theo (10) vào phương trình (16) ta nhận được

$$\dot{V} = s^T [Ks - \gamma \frac{s}{\|s\|} + \eta W\sigma - \varepsilon] + \sum_{i=1}^n w_i^T \dot{w}_i. \quad (17)$$

Với thuật học (11) thành phần cuối cùng của (17) có thể viết dưới dạng:

$$\sum_{i=1}^n w_i^T \dot{w}_i = -\eta \sum_{i=1}^n w_i^T s \sigma_i = -\eta s^T \sum_{i=1}^n w_i \sigma_i. \quad (18)$$

Mặt khác ta có

$$W\sigma = \sum_{i=1}^n w_i \sigma_i. \quad (19)$$

Thay (18), (19) vào (17) ta được

$$\dot{V} = -s^T K s - s^T \gamma \frac{s}{\|s\|} - s^T \varepsilon. \quad (20)$$

Với s là khả vi (5), xét thành phần $\frac{s}{\|s\|}$ trong (10), (20).

Trong không gian n chiều, xác định $\lim_{s \rightarrow \infty} \frac{s}{\|s\|}$ khi $\|s\| = [s_1, s_2, \dots, s_n]^T$ và $\|s\| = \sqrt{s_1^2 + s_2^2 + \dots + s_n^2}$ ta có $\lim_{s \rightarrow \infty} \frac{s}{\|s\|} = [\lim_{s_1 \rightarrow \infty} \frac{s_1}{\|s\|}; \lim_{s_2 \rightarrow \infty} \frac{s_2}{\|s\|}; \dots; \lim_{s_n \rightarrow \infty} \frac{s_n}{\|s\|}]^T$ tồn tại hữu hạn do từng thành phần của véc tơ có tử số luôn nhỏ hơn hoặc bằng mẫu số. Do vậy τ và \dot{V} trong (10), (20) là liên tục khi $s \rightarrow 0$.

Chọn $\gamma = \delta + \varepsilon_0$ với $\delta > 0$,

$$\dot{V} = -s^T K s - s^T (\delta + \varepsilon_0) \frac{s}{\|s\|} - s^T \varepsilon = -s^T K s - \delta \|s\| - (\varepsilon_0 \|s\| + s^T \varepsilon) \leq 0. \quad (21)$$

Vì $\|\varepsilon\| \leq \varepsilon_0$ do đó ta luôn có $\dot{V} < 0$ với mọi $s \neq 0$ và $\dot{V} = 0$ khi $s = 0$. Như vậy theo nguyên lý ổn định Lyapunov ta có $s \rightarrow 0$ từ phương trình (5) suy ra $e \rightarrow 0$, $\dot{e} \rightarrow 0$, hay nói cách khác quỹ đạo của rôbôt bám theo quỹ đạo mong muốn với sai số $e \rightarrow 0$. Định lý cũng như tính ổn định toàn cục của hệ điều khiển trượt sử dụng mạng nơon mô tả trong hình 2 đã được chứng minh.

4. VỀ ĐỘ CHÍNH XÁC VÀ TỐC ĐỘ HỘI TỤ CỦA THUẬT HỌC

Trong hệ phương trình động lực rôbôt, các giá trị sai lệch $\Delta M(q), \Delta B(q, \dot{q}), \Delta g(q)$, ma sát và nhiễu là những yếu tố bất định gây nên sai số và giảm tốc độ hội tụ của thuật điều khiển. Tuy nhiên với hệ rôbôt thì các đại lượng bất định này đều có giới hạn và có thể xác định được từ các số liệu của rôbôt. Cấu trúc mạng nơon có số nơon trong lớp ẩn bằng số bậc tự do của rôbôt. Các nơon sử dụng hàm tác động dạng phân bố Gauss và tín hiệu trên các nút ở lớp ra của mạng nơon là phép tổng hợp tuyến tính của các hàm đầu ra lớp ẩn. Độ chính xác của phép xấp xỉ phụ thuộc vào việc chọn các tham số c_i, λ_i của các hàm phân bố Gauss sao cho bao phủ được hết dải biên độ của các đại lượng bất định của rôbôt. Tốc độ hội tụ của mạng nơon phụ thuộc vào thuật học (11) trong đó có sự tham gia của mặt trượt s và hệ số η . Như vậy nếu ta chọn mặt trượt có độ dốc lớn (C, K lớn) và hệ số học η lớn thì tốc độ hội tụ sẽ cao. Tuy nhiên lúc này có thể xuất hiện quá điều chỉnh làm ảnh hưởng đến chất lượng điều khiển. Vấn đề chọn các tham số c_i, λ_i, η , tối ưu sẽ cho kết quả xấp xỉ tốt nhất và tốc độ hội tụ tối ưu. Có thể ứng dụng phương pháp mô phỏng hoặc giải thuật di truyền (GAs) để xác định các tham số tối ưu này.

5. MÔ PHỎNG TRÊN RÔBÔT PHẪNG HAI BẬC TỰ DO

Để minh chứng thuật điều khiển đề xuất chúng tôi đã mô phỏng bài toán chuyển động của rôbôt bám theo quỹ đạo một cung tròn trong không gian Đề các. rôbôt được chọn là loại

rôbot phẳng hai bậc tự do [18] được mô tả trong hình 3 với các tham số ghi trong Bảng 1. Nhiệm vụ của rôbot là điều khiển tay nắm chuyển động theo cung tròn từ điểm A đến điểm B trong không gian Đề các (Oxy), trong thời gian T và bảo đảm tốc độ chuyển động theo thời gian có dạng hình thang. Điều này có nghĩa là quỹ đạo chuyển động từ A đến B sẽ có đoạn đầu tăng tốc từ tốc độ ban đầu là 0 đến đoạn chuyển động với tốc độ đều và sau đó có đoạn giảm tốc để đầu tay nắm khi tới điểm B có tốc độ bằng 0.

Đặt $m_i = m_{mi} + m_{li}$; Động lực học của rôbot này có dạng

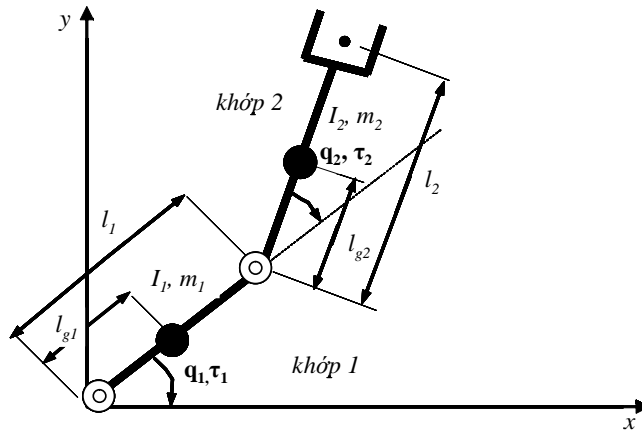
$$\hat{M}(\mathbf{q})\ddot{\mathbf{q}} + \hat{B}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{g}(\mathbf{q}) + \hat{d}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau},$$

với

$$\hat{M}(\mathbf{q}) = \begin{bmatrix} m_1 l_{g1}^2 + I_1 + I_2 + m_2(l_1^2 + l_{g2}^2 + 2l_1 l_{g2} \cos q_2) & m_2(l_{g2}^2 + l_1 l_{g2} \cos q_2) + I_2 \\ m_2(l_{g2}^2 + l_1 l_{g2} \cos q_2) + I_2 & m_2 l_{g2}^2 + I_2 \end{bmatrix}$$

$$\hat{g}(\mathbf{q}) = \begin{bmatrix} (m_1 l_{g1} + m_2 l_1)g \cos(q_1) + m_2 l_{g2} g \cos(q_1 + q_2) \\ m_2 l_{g2} g \cos(q_1 + q_2) \end{bmatrix}$$

$$\hat{B}(\mathbf{q}, \dot{\mathbf{q}}) = m_2 l_1 l_{g2} \begin{bmatrix} -2\dot{q}_2 & -\dot{q}_2 \\ \dot{q}_1 & 0 \end{bmatrix} \sin q_2$$



Hình 3. Robot phẳng hai bậc tự do

Bảng 1. Các tham số của rôbot phẳng hai bậc tự do

	Khớp thứ nhất	Khớp thứ hai
Trọng lượng thanh nối mli [kg]	50.0	50.0
Trọng lượng của động cơ mmi [kg]	5.0	5.0
Mô men quán tính của thanh nối li [kg.m ²]	10.0	10.0
Độ dài của khớp li [m]	1	1
Khoảng cách từ khớp đến trọng tâm của thanh nối lgi [m]	0.5	0.5

Ta chọn các tham số điều khiển cho moment $\boldsymbol{\tau}$ và thuật chỉnh trọng mạng neuron \dot{w}_i cho rôbot như sau

$$\boldsymbol{\tau} = \hat{M}(\mathbf{q})\ddot{\mathbf{q}} + \hat{B}\dot{\mathbf{q}} + \hat{g} - \hat{M}\dot{\mathbf{c}} - \hat{B}\mathbf{c} - \mathbf{K}\mathbf{s} - \gamma \frac{\mathbf{s}}{\|\mathbf{s}\|} + (1 + \eta)\mathbf{W}\boldsymbol{\sigma},$$

$$\dot{w}_i = -\eta \mathbf{s}\boldsymbol{\sigma}_i, \quad i = 1, 2,$$

với $K = \begin{bmatrix} 150 & 0 \\ 0 & 150 \end{bmatrix}$, $\gamma = 5, \eta = 100$.

Trạng thái ban đầu của rôbôt là $q_{10} = -0,4; q_{20} = 1,85$.

Với các tham số hàm Gauss của mạng nơon $\lambda_1 = 1, \lambda_2 = 2, c_1 = 0,01, c_2 = 0,02$.

Độ bất định của rôbôt được chọn tới 30% giá trị thật.

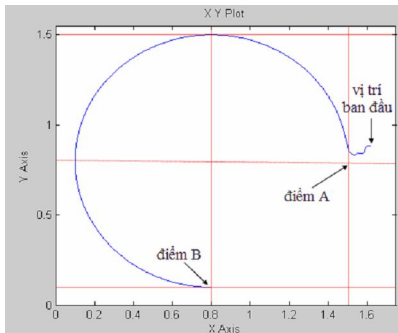
Thành phần ma sát và nhiễu loạn được giả thiết

$$d(q, \dot{q}) = d(t) = \begin{bmatrix} 3 \sin(20t) + 1 \\ \cos(20t) \end{bmatrix} + \begin{bmatrix} 5 & 3 \\ & \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}.$$

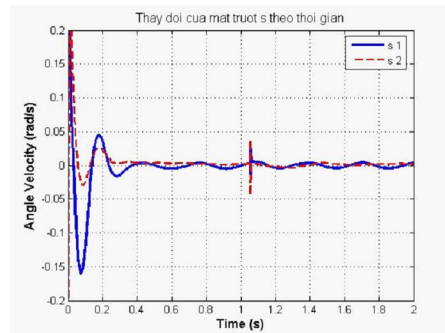
Ma trận C của mặt trượt được chọn là $C = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$ Cần xác định quỹ đạo mong muốn $q_{1d}(t)$ và $q_{2d}(t)$ khi tay nắm di chuyển từ điểm đầu $A(1,5; 0,8)$ đến điểm cuối $B(0,8; 0,1)$ theo đường tròn tâm $C(0,8; 0,8)$ theo chiều ngược chiều kim đồng hồ trong thời gian T theo quy luật tốc độ góc hình thang với gia tốc góc β cho trước. $\alpha(t)$ là góc trên cung AB. $\alpha(0) = 0, \alpha(T) = 3\pi/2$.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} 0,8 \\ 0,8 \end{bmatrix} + \begin{bmatrix} \cos \alpha \\ \sin \alpha \end{bmatrix}$$

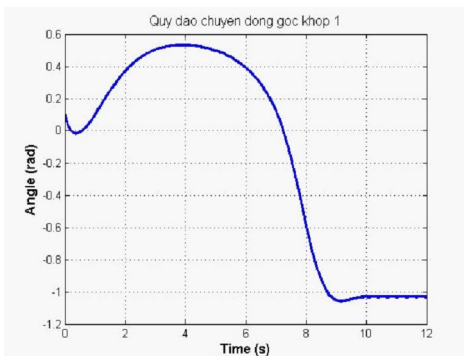
Kết quả mô phỏng: Các kết quả mô phỏng được mô tả trong các Hình 4, 5, 6, 7, 8 và 9.



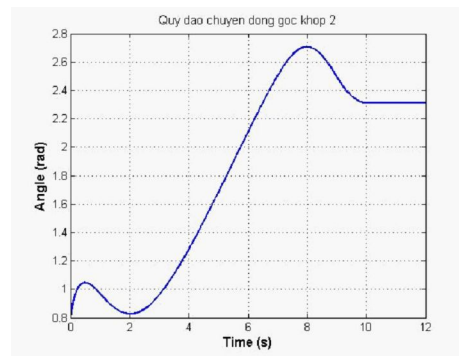
Hình 4. Quỹ đạo thực tế của tay nắm trong không gian Đề các



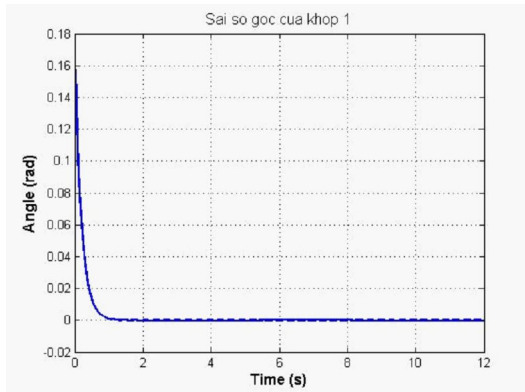
Hình 5. Thay đổi của mặt trượt theo thời gian



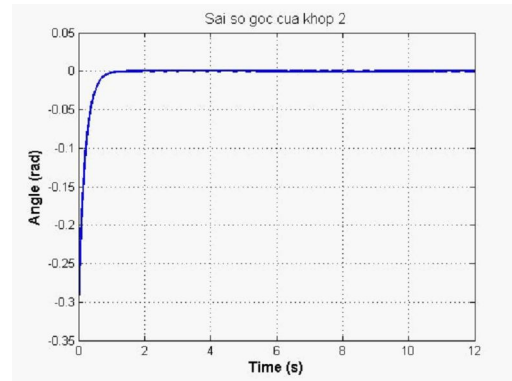
Hình 6a. Quỹ đạo chuyển động góc khớp 1



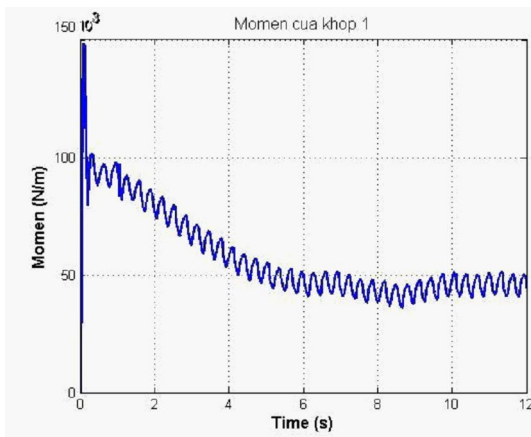
Hình 6b. Quỹ đạo chuyển động góc khớp 2



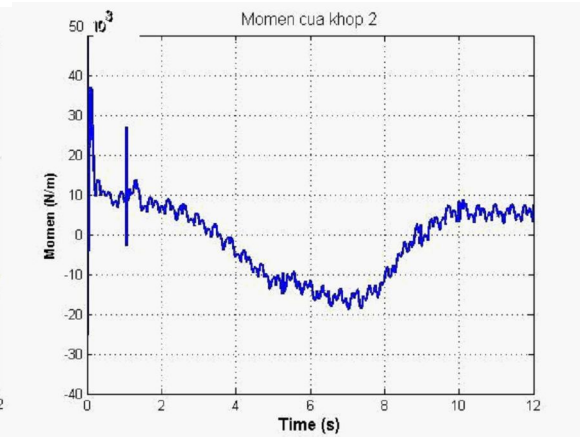
Hình 7a. Sai số của góc khớp 1



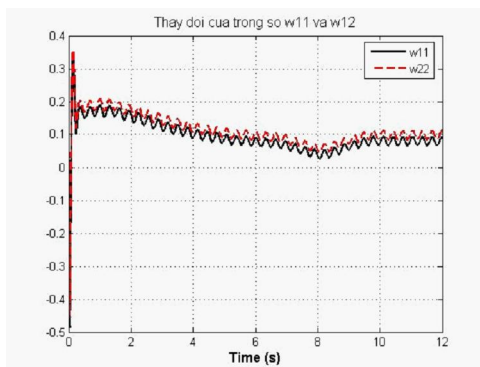
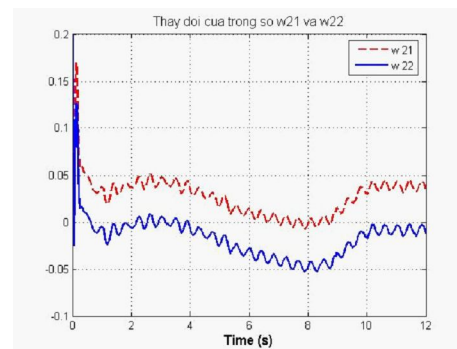
Hình 7b. Sai số của góc khớp 2



Hình 8a. Mômen của khớp 1



Hình 8b. Mômen của khớp 2

Hình 9a. Thay đổi trọng số w_{11} và w_{12} Hình 9b. Thay đổi trọng số w_{21} và w_{22}

Qua mô phỏng ta thấy rôbot được điều khiển bám rất sát quỹ đạo mong muốn trong khi sai số của mô hình lên đến 30 % và có cả thành phần không xác định là ma sát và nhiễu bên ngoài tác động lên từng khớp. Điều này minh chứng cho tính đúng đắn và khả năng ứng dụng thực tế của phương pháp đề xuất trong thực tiễn.

5. KẾT LUẬN

Báo cáo này đã đề xuất một mô hình điều khiển theo nguyên lý trượt có sử dụng mạng nơron bù những yếu tố bất định của rôbôt và chứng minh được tính ổn định toàn cục của hệ thống. Mạng nơron cho phép giảm nhỏ hơn nữa các giá trị giới hạn sai lệch do tính bất định của đối tượng. Các mô phỏng trên rôbôt phẳng hai bậc tự do bám sát quỹ đạo mong muốn trong không gian Đề các đã cho các kết quả rất phù hợp với các nguyên lý đề xuất của báo cáo. Qua quá trình mô phỏng với nhiều mức độ bất định và nhiễu tác động cũng như sự lựa chọn các tham số và thuật chỉnh trọng mạng nơron khác nhau ta luôn được các kết quả hội tụ chứng tỏ tính ổn định bền vững của phương pháp. Tùy thuộc vào cách chọn mặt trượt ta có thể xây dựng các bộ điều khiển trượt tối ưu theo mô hình trượt PD hay trượt PID. Các nghiên cứu sâu hơn về tốc độ hội tụ và tối ưu các tham số sử dụng các thuật học di truyền GA sẽ là các vấn đề được tiếp tục nghiên cứu.

TÀI LIỆU THAM KHẢO

- [1] John J. Cr. Caig, *Robotics Mechanism and Control*, Second Edition, 1955.
- [2] N. E. Cotter, The stone-weierstrass and its application to neural networks, *IEEE Tran on Neural Networks*, **1** (4) (1990) p 290-295.
- [3] J. Somlo, B. Lantos, P.T. Cat, *Advanced Robot Control*, Akademiai Kiado, Budapest 1997.
- [4] D. Psaltis, A. Sideris, Yamamura, A neural controllers, *Proc IEEE 1st Int. Conf. Neural Networks*, San Diego, Jun 1987.
- [5] Ahmet Karakasoglu, M. K. Sundareshan, A recurrent neural network-based adaptive variable structure model-following control of robotic manipulators, *Automatica* **31** (10) (1995) 1495–1507.
- [6] Chin - Teng Lin and C.S George Lee, *Neural Fuzzy Systems*, Book is to the Chiao-Tung University Centennial 1996.
- [7] H. Daniel Patino, Ricardo Carelli, and Benjamin R. Kunchen, Neural networks for advanced control of robot manipulators, *IEEE Transactions on Neural Networks* **13** (2) (2002) 343–353.
- [8] K. Jezernik, M. Rodic, R. Safaric, Neural network sliding mode robot control, *Roboticka* **15** (1997) 23–30.
- [9] Mohamad Teshnehlab, Keigo Wantanabe, *Intelligent Control Base on Flexible Neural Networks* Kluwer, Academic Publishers, 1999.
- [10] M. On der Efe, Okyay Kaynak., *Sliding mode Control of a Three Degrees of Freedom Anthropoid Robot by Driving the Controller Parameters to an Equivalaent Regime*, Bogazici University, Electrical and Electronic Eng. Dep. Bebek, 80815 Istanbul Turkey, Vol.122 December 2000 (632–639).
- [11] Miran Rodi, Karel Jezernik - Siding mode based neural network learning procedure, *V: 5th Int. Workshop on Robotics in Alpe-Adria-Danube Region, RAAD'96*, Budapest, Hungary, 10-13 June, 1996 (Proceedings, pp.547–552) ISBN 963 420 482 1.

- [12] N. Murata, S Yoshizawa, S Amari, Network information criterion-determining the number of hidden units for an artificial neural network model, *IEEE Trans, Neural Networks* **5** (6) (1994) 856–887.
- [13] A. Ishiguro, T. Fururashi, and Uchikawa, A neural network compensator for uncertainties of robotic manipulator, *IEEE on Neural Networks* **7** (2) (1996) 388–399.
- [14] F.C Sun, Z.Q Sun, and P.Y Woo, Neural network based adaptive controller design of robotic manipulator with observer, *IEEE Trans on Neural Networks* **12** (1) (2001) 54–57.
- [15] Z. Wang and C.D Massimo, A procedure for determining the topology of multilayer feedforward neural networks, *Neural Networks* **7** (2) (1994) 74–92.
- [16] T. Ozaki, T.suzuke, T. Furuhashi, S.Okuma, and Y. Uchikawa, Trajectory control of robotic manipulator using neural networks, *IEEE Tran. on Ind. Electron* **38** (3) (1991) 195–202.
- [17] V.L. Utkin, *Sliding modes in control and optimization*, Springer Verlag, New York, 1992.
- [18] Lorenzo Sciavicco, Bruno Siciliano, *Modeling and Control of Robot Manipulators*, International Edition 1996.
- [19] H. Asada, J. J. E. Slotine, “Robot analysis and control”, Massachusetts Institute of Technology, 1985.

Nhận bài ngày 19 - 2 - 2008