

DANH SÁCH CÁC THÀNH VIÊN THAM GIA ĐỀ TÀI

TT	Họ và tên	Đơn vị công tác
1	Lê Tiến Dũng	Trường Đại học Bách khoa – Đại học Đà Nẵng
2	Trương Thị Bích Thanh	Trường Đại học Bách khoa – Đại học Đà Nẵng
3	Giáp Quang Huy	Trường Đại học Bách khoa – Đại học Đà Nẵng
4	Đỗ Thị Thu Thảo	Trường Đại học Bách khoa – Đại học Đà Nẵng

MỤC LỤC

Danh mục các hình vẽ	v
Danh mục các bảng biểu	ix
Danh mục các chữ viết tắt	x
Thông tin về kết quả nghiên cứu bằng tiếng Việt	xi
Thông tin kết quả nghiên cứu bằng tiếng Anh	xvi
PHẦN MỞ ĐẦU	1
TÍNH CẤP THIẾT CỦA ĐỀ TÀI	1
MỤC TIÊU NGHIÊN CỨU	4
ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU	4
CÁCH TIẾP CẬN, PHƯƠNG PHÁP NGHIÊN CỨU	5
CHƯƠNG 1 – TỔNG QUAN VỀ ROBOT CÔNG NGHIỆP VÀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN TAY MÁY ROBOT CÔNG NGHIỆP	7
1.1. Tổng quan về robot công nghiệp	7
1.2. Tổng quan về các phương pháp điều khiển tay máy robot công nghiệp	9
CHƯƠNG 2 – MÔ HÌNH ĐỘNG LỰC HỌC VÀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN TRƯỢT BẬC CAO CHO TAY MÁY ROBOT CÔNG NGHIỆP	14
2.1. Giới thiệu	14
2.2. Xây dựng mô hình động lực học cho một tay máy robot công nghiệp dạng tổng quát	16
2.3. Phương pháp điều khiển trượt truyền thống cho tay máy robot công nghiệp	20
2.4. Phương pháp điều khiển trượt bậc cao cho tay máy robot công nghiệp	33
CHƯƠNG 3 – NGHIÊN CỨU ĐỀ XUẤT THUẬT TOÁN ĐIỀU KHIỂN TRƯỢT BẬC CAO THÍCH NGHI ỨNG DỤNG TRÍ TUỆ NHÂN TẠO	39
3.1. Giới thiệu chung	39

3.2. Đề xuất thuật toán điều khiển trượt bậc cao kết hợp bộ quan sát trượt bậc cao thời gian cố định	43
3.3. Mạng nơ-ron nhân tạo	44
3.4. Bù thích nghi thành phần bất định của tay máy robot bằng mạng nơ-ron	47
CHƯƠNG 4 – MÔ PHỎNG KIỂM NGHIỆM	50
4.1. Xây dựng mô hình cơ khí của robot trên Solidworks và tích hợp vào Matlab-Simulink	50
4.2. Mô phỏng so sánh các thuật toán điều khiển	53
4.3. Kết quả mô phỏng thuật toán điều khiển trượt bậc cao thích nghi ứng dụng mạng nơ-ron	55
4.4. Kết quả mô phỏng thuật toán điều khiển trượt bậc cao với bộ quan sát thời gian hữu hạn	
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	90
TÀI LIỆU THAM KHẢO	91
PHỤ LỤC	96
Phụ lục 1 – Bản sao thuyết minh và hợp đồng của đề tài	
Phụ lục 2 - Các sản phẩm khoa học của đề tài	
Phụ lục 2 – Các sản phẩm đào tạo	
Phụ lục 3 – Các sản phẩm ứng dụng	

DANH MỤC CÁC HÌNH VẼ

Tên hình vẽ	Trang
Hình 1.1 – Tay máy robot SCARA (Selective Compliance Assembly Robot Arm)	7
Hình 1.2 – Tay máy robot đầu tiên truyền động bằng động cơ điện servo của hãng Nachi, Nhật Bản	8
Hình 1.3 - Robot Opportunity được đưa lên sao hỏa năm 2004	9
Hình 1.4 - Robot tầm xa và hạng nặng đầu tiên với trọng tải 1.000 kg vào năm 2007 của hãng KUKA, Đức	10
Hình 1.5 - Robot tầm xa và hạng nặng M-2000iA với trọng tải 1.200 kg của hãng FANUC, Nhật Bản	10
Hình 1.6 - Robot đa năng IRB120 của hãng ABB	11
Hình 1.7 - Robot giống người đầu tiên được đưa lên vũ trụ năm 2011 của NASA	11
Hình 1.8 - Số lượng robot công nghiệp được lắp đặt hàng năm trên toàn thế giới giai đoạn: 2013 -2018 và dự báo giai đoạn 2019-2022	12
Hình 1.9 - Số lượng robot công nghiệp được lắp đặt hàng năm theo khu vực	13
Hình 1.10 – Thống kê số lượng robot công nghiệp tại Hàn Quốc theo năm	14
Hình 1.11 – Thống kê mật độ số lượng robot công nghiệp tại một số quốc gia nổi bật vào năm 2018	14
Hình 1.12 - Sơ đồ khối bộ điều khiển thích nghi cho tay máy robot song song truyền động bằng cơ nhân tạo	18
Hình 2.1 – Mô tả động học của thanh thứ i cho việc thiết lập công thức Lagrange	23
Hình 2.2 – Mô tả động học của motor thứ i	26

<i>cho việc thiết lập công thức Lagrange</i>	
Hình 2.3 - Điểm trạng thái trượt trên mặt trượt s và tiến về gốc tọa độ, không có hiện tượng chattering	32
Hình 2.4 - Điểm trạng thái trượt trên mặt trượt s và tiến về gốc tọa độ, xảy ra hiện tượng chattering	32
Hình 2.5 – Đồ thị hàm mặt trượt s theo thời gian t	33
Hình 2.6 – Phương pháp mặt trượt có lớp biên	33
Hình 2.7 - Đồ thị hàm mặt trượt s theo thời gian t khi có hiện tượng chattering	33
Hình 2.8 - Điểm trạng thái trượt trên mặt trượt s và tiến về gốc tọa độ	34
Hình 3.1 - Mô hình nơ-ron nhân tạo	53
Hình 3.2 - Một số dạng liên kết mạng	56
Hình 3.3 - Cấu trúc của mạng nơ-ron truyền thẳng ba lớp	59
Hình 4.1 – Mô phỏng tay máy robot trên phần mềm Solidworks	63
Hình 4.2 - Mô hình cơ khí tay máy robot sau biên dịch và tích hợp qua Multibody, Simscape của Matlab-Simulink	64
Hình 4.3 - Sơ đồ khối mô hình toàn hệ thống trên Matlab-Simulink	64
Hình 4.4 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán PD bù trọng lực	65
Hình 4.5 - Đồ thị sai số quỹ đạo của thuật toán PD theo trục X	65
Hình 4.6 - Đồ thị sai số quỹ đạo của thuật toán PD theo trục Y	66
Hình 4.7 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán BLM SMC	67

Hình 4.8 - Đồ thị sai số quỹ đạo của thuật toán BLMSMC theo trục X	67
Hình 4.9 - Đồ thị sai số quỹ đạo của thuật toán BLSMC theo trục Y	68
Hình 4.10 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán TSOSMC	69
Hình 4.11 - Đồ thị sai số quỹ đạo của thuật toán TSOSMC theo trục X	69
Hình 4.12 - Đồ thị sai số quỹ đạo của thuật toán TSOSMC theo trục Y	70
Hình 4.13 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán STSOSMC	71
Hình 4.14 - Đồ thị sai số quỹ đạo của thuật toán STSOSMC theo trục X	71
Hình 4.15 - Đồ thị sai số quỹ đạo của thuật toán STSOSMC theo trục Y	72
Hình 4.16 - So sánh sai số góc quay khớp 1 của tay máy robot	73
Hình 4.17 - Phóng to đồ thị so sánh sai số góc quay khớp 1 của tay máy robot giai đoạn đầu	73
Hình 4.18 - Phóng to đồ thị so sánh sai số góc quay khớp 1 của tay máy robot giai đoạn xác lập	74
Hình 4.19 - So sánh sai số góc quay khớp 2 của tay máy robot	74
Hình 4.20 - Phóng to đồ thị so sánh sai số góc quay khớp 2 của tay máy robot giai đoạn đầu	75
Hình 4.21 - Phóng to đồ thị so sánh sai số góc quay khớp 2 của tay máy robot giai đoạn xác lập	75
Hình 4.22 - So sánh tín hiệu điều khiển khớp 1 của tay máy robot	76

Hình 4.23 - Phóng to tín hiệu điều khiển khớp 1 của tay máy robot	76
Hình 4.24 - So sánh tín hiệu điều khiển khớp 2 của tay máy robot	77
Hình 4.25 - Phóng to so sánh tín hiệu điều khiển khớp 2 của tay máy robot	77
Hình 4.26 - Mô phỏng của tay máy robot 3 bậc tự do trên phần mềm SOLIDWORKS	78
Hình 4.27 - Giá trị giả định của những thành phần bất định và kết quả xấp xỉ bởi bộ quan sát tại mỗi khớp	81
Hình 4.28 - So sánh kết quả bám quỹ đạo của khâu chấp hành cuối khi trong không gian 3 chiều (XYZ)	81
Hình 4.29 - Sai số điều khiển vị trí của khâu chấp hành cuối theo trục X	82
Hình 4.30 - Sai số điều khiển vị trí của khâu chấp hành cuối theo trục Y	83
Hình 4.31 - Sai số điều khiển vị trí của khâu chấp hành cuối theo trục Z	84
Hình 4.32 - Những tín hiệu điều khiển tại mỗi khớp	84

DANH MỤC CÁC BẢNG BIỂU

Tên bảng	Trang
<i>Bảng 4.1. Tham số của các thuật toán điều khiển</i>	80

DANH MỤC CÁC CHỮ VIẾT TẮT

RRR	Revolute – Revolute – Revolute
DOF	Degree of freedom
P	Proportional
I	Integral
D	Derivative
RPDR	Rice Planar Delta Robot
NPD	Nonlinear PD
VSC	Variable Structure Control
SMC	Sliding Mode Control
FSMC	Fuzzy Sliding Mode Control
MLP	Multilayer perceptron
RBF	Radial Basis Function neural network
PNN	Polynomial neural networks
ANFIS	Adaptive-network-based fuzzy inference system
GMDH	Group method of data handling
MRA	Multiresolution analysis
CTC	Computed torque control
NCT	Nonlinear Computed torque
DNTSMC	Decoupled nonsingular terminal sliding mode controller
NN	Neural Network
BLM	Boundary layer method
AFC	Acceleration feedback control
SOSMC	Second Order Sliding Mode Control
TSOSMC	Twisting Second Order Sliding Mode Control
STSOSMC	Super Twisting Second Order Sliding Mode Control
HOSMC	Higher Order Sliding Mode Control

THÔNG TIN KẾT QUẢ NGHIÊN CỨU

1. Thông tin chung:

- Tên đề tài: **Điều khiển trượt bậc cao cho tay máy robot công nghiệp ứng dụng trí tuệ nhân tạo**

- Mã số: **B2019-DN02-52**

- Chủ nhiệm: PGS. TS. Lê Tiến Dũng

- Cơ quan chủ trì: Trường Đại học Bách khoa - Đại học Đà Nẵng

- Thời gian thực hiện: Từ tháng 08 năm 2019 đến tháng 08 năm 2021

2. Mục tiêu:

Mục tiêu chính của đề tài nghiên cứu là đề xuất được một thuật toán điều khiển trượt bậc cao mới để điều khiển bám quỹ đạo cho tay máy robot công nghiệp đạt được các lợi ích sau: Thừa hưởng các lợi thế của điều khiển trượt thông thường và điều khiển trượt bậc cao thông thường, có tốc độ hội tụ nhanh và độ chính xác cao.

Kết quả nghiên cứu có tính mới ở tầm quốc tế để có thể công bố 01 bài báo thuộc danh mục SCIE, 01 bài báo trong nước và đào tạo 01 Thạc sĩ.

3. Tính mới và sáng tạo:

Điều khiển trượt (Sliding Mode Control - SMC) là một trong những phương pháp điều khiển phi tuyến hiệu quả được sử dụng để điều khiển các hệ thống phi tuyến có sự tồn tại của các thành phần bất định, đặc biệt là tay máy robot. Tuy nhiên, nhược điểm của nó là chỉ đảm bảo sự ổn định tiệm cận trong khi có nhiễu dao động tần số cao trong tín hiệu điều khiển đầu vào. Có nhiều công trình nghiên cứu để phát huy phương pháp điều khiển trượt cho tay máy robot công nghiệp. Gần đây nhất là các phương pháp sử dụng thuật toán điều khiển trượt bậc cao dựa

trên mô hình động lực học của tay máy robot. Tuy nhiên, vẫn có những khó khăn tồn tại khi áp dụng đó là không dễ dàng để có được một mô hình toán học chính xác do sự hiện diện của nhiều thành phần bất định của hệ thống. Hơn nữa, không thể loại bỏ hoàn toàn hiện tượng dao động trong tín hiệu điều khiển của thuật toán

Tính mới và sáng tạo của trong đề tài này là đề xuất một phương pháp hiệu quả có thể giải quyết cùng lúc các tồn tại nêu trên đó là sử dụng một bộ quan sát tích hợp vào thuật toán điều khiển để tính gần đúng tổng các thành phần bất định ảnh hưởng đến hệ thống. Bằng cách này, những dao động tần số cao không chỉ được giảm đáng kể mà còn tăng tính bền vững của hệ thống điều khiển. Ngoài ra, đề tài còn đề xuất sử dụng mạng nơ-ron nhân tạo để ước lượng và bù sai số cho các thành phần bất định, nâng cao tính bền vững và ổn định của hệ thống.

4. Kết quả nghiên cứu:

Kết quả nghiên cứu của đề tài đã đề xuất một phương pháp điều khiển bền vững mới sử dụng sự kết hợp giữa một bộ quan sát trượt bậc cao thời gian cố định (FxTHOSMO) để ước tính toàn bộ các thành phần bất định mà không cần biết trước các thông tin về chúng, và một thuật toán điều khiển trượt đầu cuối thời gian hữu hạn (FnTTSMC). Thuật toán FnTTSMC được thiết kế dựa trên phương trình mặt trượt đầu cuối nhanh toàn cục thời gian hữu hạn và luật điều khiển tiếp cận nhanh để thiết lập sự hội tụ trong thời gian hữu hạn toàn cục của hệ thống. Hệ thống điều khiển tích hợp có các đặc tính ưu việt như độ chính xác bám quỹ đạo cao, hội tụ trong thời gian hữu hạn và bền vững với những ảnh hưởng của các thành phần bất định. Các phân tích với lý thuyết Lyapunov đã chứng minh tính ổn định trong thời gian hữu hạn và sự hội tụ của hệ thống điều khiển. Qua các kết quả mô phỏng ứng dụng cho trường hợp cụ thể với tay máy robot 3 bậc tự do đã cho thấy thuật toán điều khiển mà bài đề xuất mang lại hiệu quả tốt và chất lượng điều khiển cao hơn so với các trường hợp sử dụng các thuật toán điều khiển của các công trình nghiên cứu đi trước ở cả ba khía cạnh bao gồm độ chính

xác bám quỹ đạo, hội tụ nhanh trong thời gian hữu hạn và giảm hiện tượng chattering.

5. Sản phẩm:

- 01 bài báo quốc tế thuộc danh mục SCIE:

[1] Doan Quang Vinh, Vo Anh Tuan, Le Tien Dung*, Hee-Jun Kang and Nguyen Ngoc Hoai An, "A novel fast terminal sliding mode tracking control methodology for robot manipulators." *Applied Sciences* 10.9 (2020): 3010.

- 02 bài báo trên tạp chí trong nước:

[1] Lê Tiến Dũng, Võ Anh Tuấn, và Nguyễn Ngọc Hoài Ân, "Phương pháp điều khiển bền vững mới cho tay máy robot công nghiệp sử dụng kết hợp bộ quan sát trượt bậc cao thời gian cố định với thuật toán điều khiển trượt." *Tạp chí Khoa học và Công nghệ - Đại học Đà Nẵng*, Vol. 19, No. 5.1, Tr. 22-28, 2021.

[2] Lê Tiến Dũng, Lê Phước Lộc, Đồng Hoàng Bin, Hà Ngọc Chính, Nguyễn Quốc Huy, và Huỳnh Ngọc Khoa, "Ứng dụng mạng nơ-ron nhân tạo trong chẩn đoán lỗi cho máy biến áp lực tại truyền tải điện kon tum – công ty truyền tải điện 2", *Tạp chí Khoa học và Công nghệ - Đại học Đà Nẵng*, Vol. 18, No. 7, Tr. 84-89, 2020.

- Đào tạo 01 Thạc sỹ bảo vệ thành công luận văn tốt nghiệp:

Phan Quốc Việt, lớp K37.TĐH, chuyên ngành Kỹ thuật điều khiển và Tự động hóa. Tên đề tài luận văn: “*Nghiên cứu thiết kế thuật toán điều khiển trượt bậc 2 thích nghi cho tay máy robot ứng dụng mạng nơ-ron nhân tạo*” dưới sự hướng dẫn của PGS. TS. Lê Tiến Dũng (Quyết định giao đề tài số 333/QĐ-ĐHBK-ĐT ngày 02/03/2020, quyết định bảo vệ số /QĐ-ĐHBK-ĐT, Bằng tốt nghiệp Thạc sĩ số hiệu DND.7.0015411 cấp ngày 03/02/2021).

6. Hiệu quả, phương thức chuyển giao kết quả nghiên cứu và khả năng áp dụng:

Thông qua việc thực hiện đề tài nghiên cứu đã góp phần nâng cao năng lực nghiên cứu khoa học tại Việt Nam trong lĩnh vực điều khiển tay máy robot. Kết quả nghiên cứu của đề tài là cơ sở để có thể tiếp tục mở rộng nghiên cứu và áp dụng thử nghiệm vào thực tế.

Báo cáo tổng kết các kết quả nghiên cứu là tài liệu tham khảo tốt cho các sinh viên, học viên cao học và nghiên cứu sinh chuyên ngành Kỹ thuật điều khiển và Tự động hóa, Cơ điện tử, Kỹ thuật điện. Kết quả nghiên cứu góp phần nâng cao chất lượng đào tạo, mở ra một hướng nghiên cứu chuyên sâu về lĩnh vực điều khiển tay máy robot công nghiệp ứng dụng thuật toán điều khiển trượt bậc cao. Đây là một hướng nghiên cứu mới đang có tính thời sự, cấp thiết không chỉ trong nước mà cả trên thế giới.

Ngày 09 tháng 08 năm 2021

Cơ quan chủ trì

(ký, họ và tên, đóng dấu)

Chủ nhiệm đề tài

(ký, họ và tên)

INFORMATION ON RESEARCH RESULTS

1. General information:

Project title: **High order sliding mode control for robot manipulators using artificial intelligent**

Code number: **B2019-DN02-52**

Coordinator: Le Tien Dung

Implementing institution: University of Science and Technology

Duration: from 30th August 2019 to 30th August 2021

2. Objective(s):

The main objective of the research project is propose a novel high order sliding mode control algorithm for trajectory tracking control of robot manipulator which has the following advantages: Adopted the advantages of traditional sliding mode control and high order sliding mode control algorithms, has the fast converge and high accuracy.

The research results are new at international level in order to publish 01 paper in SCIE journal list, 01 paper in Vietnamese journal and advice 01 Master thesis.

3. Creativeness and innovativeness:

Sliding Mode Control (SMC) is one of effective nonlinear control algorithms using to control nonlinear systems with existence of uncertainties, such as robot manipulators. However, the disadvantage of SMC is just guarantee of asymptotic stability while there are many high frequency chattering in input control signals. There are many researchs to enhance the advantages of SMC for robot manipulators. In recent years, the high order SMC algorithm is proposed based on dynamic model of robot manipulators. However, there are difficulties when

applying: not easy to have the exact dynamic model of robot, and can not completely remove chattering phenomenon in control signal.

The novel creation of this research project is propose an effective method to solve above difficulties at the same time, that is using an observer integrated into control algorithm in order to estimate the sum of uncertainties which affect to the control system. By this way, high frequency chattering is not only elimenated but also increase the robustness of the system. In addition, in this project we propose a method using artificial neural network to estimate and compensate the error of estimation of uncertainties, increase the robustness of the system.

4. Research results:

The authors of this research project proposed a new robust control strategy using a combination between a fixed-time high-order sliding mode observer (FxTHOSMO) and finite-time terminal sliding mode control (FnTTSMC) for robotic manipulators. First, a FxTHOSMO is proposed to approximate the lumped uncertainties without their prior knowledge requirement. Then, according to achieved approximation information, FnTSMC is designed based on a finite-time global fast terminal sliding mode surface and fast reaching control law to establish a global finite-time convergence of the system. Consequently, the hybrid control system provides superior characteristics such as high tracking accuracy, less chattering occurrence, finite-time convergence, and robustness against the lumped uncertainties. The finite-time stability and convergence of the control system are validated by using the Lyapunov criteria. Computer simulations applied to a 3-DOF robotic manipulator are presented to verify the effectiveness and outstanding properties of the proposed control method.

5. Products:

- 01 SCIE paper:

[1] Doan, Q. V., Vo, A. T., Le, T. D., Kang, H. J., & Nguyen, N. H. A. (2020). A novel fast terminal sliding mode tracking control methodology for robot manipulators. *Applied Sciences*, 10(9), 3010.

- 02 paper in national journal:

[1] Le, Tien Dung, Vo Anh Tuan, and Nguyen Ngoc Hoai An. "A new robust control method for robot manipulators using high order sliding mode control combined with sliding mode control algorithm" *The University of Danang-Journal of Science and Technology* (2021): 22-28.

[2] Le Tien Dung, Le Phuoc Loc, et al. "Application of artificial neural network in fault diagnosis for power transformer at Kontum power transmission-Power Transmission Company No. 2 (PTC2)." *The University of Danang-Journal of Science and Technology* (2020): 84-89.

- 01 master student who successfully defend his thesis:

Phan Quốc Việt, K37.TĐH, major in Control Engineering and Automation. Name of thesis: "Study to design an adaptive second order " under the supervise of Assoc. Dr. Lê Tiến Dũng (Decision number 333/QĐ-ĐHKB-ĐT 02/03/2020, Thesis Degree number DND.7.0015411 03/02/2021).

6. Effects, transfer alternatives of research results and applicability:

The research project has the contribution in improving the capacity of scientific research in Vietnam in the field of robotic manipulator control. Research results are the basis for further expansion of the research content and application into practice.

The summary report of the research results is a useful for undergraduate students, graduate students in the field of Control Engineering and Automation, Mechatronics, Electrical Engineering. The research results have the contribution in improving the quality of training and education. This is a new research trend which is urgent not only in the Vietnam but also in the world.

PHẦN MỞ ĐẦU

TÍNH CẤP THIẾT CỦA ĐỀ TÀI

Tay máy robot công nghiệp đóng một vai trò quan trọng trong rất nhiều các hệ thống sản xuất tự động hóa. Đặc biệt, chúng rất phù hợp để làm việc trong các môi trường nguy hiểm, độc hại hoặc trong môi trường chân không nơi mà con người không thể có mặt. Tại Việt Nam, tay máy robot đã được triển khai trong các ngành sản xuất vật liệu xây dựng, luyện kim, chế tạo cơ khí, công nghiệp đóng tàu và một vài lĩnh vực khác. Trong chiến lược phát triển công nghiệp Việt Nam đến năm 2025, tầm nhìn 2035, chính phủ đã đặt mục tiêu giá trị sản phẩm công nghiệp công nghệ cao và sản phẩm ứng dụng công nghệ cao đến năm 2025 đạt khoảng 45% tổng GDP, sau năm 2025 đạt trên 50%. Trong đó, định hướng đến năm 2020 Việt Nam có thể nghiên cứu, thiết kế và sản xuất robot công nghiệp. Để làm được điều này, Việt Nam cần tập trung phát huy nghiên cứu phát triển, làm chủ công nghệ về robot - lĩnh vực trung tâm của cuộc cách mạng công nghệ lớn.

Trong các phương pháp điều khiển tay máy robot công nghiệp thì phương pháp điều khiển trượt (Sliding Mode Control) có nhiều ưu điểm như mang lại độ chính xác cao, bền vững với các nhiễu loạn và sự thay đổi của tham số trong quá trình hoạt động. Tuy nhiên, phương pháp điều khiển trượt có những hạn chế cần khắc phục như hiện tượng rung động của tín hiệu điều khiển (chattering) và khó khăn trong xây dựng mô hình động lực học của tay máy robot, đặc biệt là loại tay máy robot song song. Thời gian gần đây, phương pháp điều khiển trượt bậc cao (High Order Sliding Mode Control) đang được các nhà khoa học trên thế giới quan tâm nghiên cứu vì những ưu điểm của nó so với phương pháp điều khiển trượt thông thường.

Trong phương pháp điều khiển trượt bậc cao, điều khiển chế độ trượt bậc cao tác động nhanh không có nghiệm kỳ dị (NFTSMC – Nonsingularity Fast Terminal Sliding Mode Control) là một trong những cách tiếp cận mạnh mẽ nhất

để giải quyết bài toán bám quỹ đạo cho các hệ thống phi tuyến như tay máy robot công nghiệp. NFTSMC xử lý tốt các nhược điểm của cả Điều khiển trượt thông thường (SMC) và Điều khiển trượt bậc cao thông thường (TSMC), chẳng hạn như điểm kỳ dị, hội tụ thời gian không xác định, tán gẫu và hội tụ chậm. Chính vì vậy, đề tài này có tính cấp thiết cần nghiên cứu phương pháp điều khiển trượt bậc cao áp dụng cho điều khiển bám quỹ đạo của tay máy robot công nghiệp.

MỤC TIÊU NGHIÊN CỨU

Mục tiêu chính của đề tài nghiên cứu là đề xuất được một thuật toán điều khiển bậc cao mới cho để điều khiển bám quỹ đạo cho tay máy robot công nghiệp đạt được các lợi ích sau: Thừa hưởng các lợi thế của điều khiển trượt thông thường và điều khiển trượt bậc cao thông thường, có tốc độ hội tụ nhanh và độ chính xác cao.

Kết quả nghiên cứu có tính mới ở tầm quốc tế để có thể công bố 01 bài báo thuộc danh mục SCIE, 01 bài báo trong nước và đào tạo 01 Thạc sĩ.

ĐỐI TƯỢNG, PHẠM VI NGHIÊN CỨU

a) Đối tượng nghiên cứu

- Mô hình động lực học dạng tổng quát cho một tay máy robot công nghiệp có n bậc tự do.
- Phương pháp điều khiển trượt bậc cao cho tay máy robot công nghiệp

b) Phạm vi nghiên cứu

- Đề tài giới hạn phạm vi nghiên cứu cho tay máy robot công nghiệp là loại tay máy có các thanh nối được xem là loại cứng (rigid body).
- Trong đề tài, giả thiết các biến khớp và tốc độ của biến khớp là đo được và đưa phản hồi về bộ điều khiển chuyển động của robot. Giả thiết các thành phần bất định như sai số mô hình, các lực ma sát và nhiễu loạn từ bên ngoài là hữu hạn và bị chặn trên. Các giả thiết này đều phù hợp và sát với thực tế.

CÁCH TIẾP CẬN, PHƯƠNG PHÁP NGHIÊN CỨU

a) Cách tiếp cận

- Khảo sát các kết quả nghiên cứu đi trước trong lĩnh vực liên quan.
- Kế thừa các kết quả của các đề tài, công trình khoa học đã thực hiện của chủ nhiệm đề tài và các thành viên đề tài.
- Tổng hợp, phát triển cơ sở lý thuyết, phân tích các mô hình toán học để từ đó đề xuất giải pháp thiết kế thuật toán điều khiển trượt bậc cao mới cho tay máy robot công nghiệp.
- Mô phỏng các thành phần cơ khí của tay máy robot trên các phần mềm chuyên dụng, sát với thực tiễn, thay thế cho mô hình thật và có thể sử dụng để kiểm nghiệm giải pháp đề xuất.
- Mô phỏng kiểm nghiệm.
- Phân tích ổn định, đánh giá các kết quả.

b) Phương pháp nghiên cứu

- Xem xét các vấn đề nghiên cứu liên quan, nghiên cứu tổng quan, so sánh và đánh giá các ưu điểm, khuyết điểm của các phương pháp điều khiển bám quỹ đạo khác nhau cho tay máy robot công nghiệp để từ đó đề xuất ý tưởng cải tiến nhằm nâng cao chất lượng.
- Sử dụng các công cụ toán học phù hợp để xây dựng các mô hình toán học phục vụ cho mục đích phân tích, thiết kế tối ưu động học và xây dựng mô hình động lực học làm cơ sở thiết kế thuật toán điều khiển mới.
- Sử dụng các công cụ phần mềm: Phần mềm Matlab-Simulink để xây dựng chính xác mô hình của tay máy robot, và lập trình, mô phỏng kiểm nghiệm thuật toán.
- Thiết kế thuật toán điều khiển và thực hiện các thử nghiệm trên mô hình mô phỏng.
- Áp dụng các lý thuyết ổn định để phân tích và xác định miền ổn định của hệ thống điều khiển.
- Kiểm tra độ chính xác của kết quả điều khiển bám quỹ đạo và đánh giá chất lượng điều khiển dựa trên việc phân tích các dữ liệu kết quả.

CHƯƠNG 1

TỔNG QUAN VỀ ROBOT CÔNG NGHIỆP VÀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN TAY MÁY ROBOT CÔNG NGHIỆP

1.1. TỔNG QUAN VỀ ROBOT CÔNG NGHIỆP

1.1.1. Khái niệm tay máy robot công nghiệp

Tay máy robot công nghiệp đóng một vai trò quan trọng trong rất nhiều các hệ thống sản xuất tự động hóa. Đặc biệt, chúng rất phù hợp để làm việc trong các môi trường nguy hiểm, độc hại hoặc trong môi trường chân không nơi mà con người không thể có mặt.

Hiện nay, có nhiều định nghĩa về tay máy robot công nghiệp, có thể điểm qua một số định nghĩa như sau:

+ ***Định nghĩa theo chuẩn AFNOR (Pháp):***

Robot công nghiệp là một cơ cấu chuyển động tự động có thể lập trình, lặp lại các chương trình, tổng hợp các chương trình đặt ra trên các trục tọa độ, có khả năng định vị, định hướng, di chuyển các đối tượng vật chất: chi tiết, dao cụ, giá lắp... theo những hành trình thay đổi đã chương trình hóa nhằm thực hiện các nhiệm vụ khác nhau.

+ ***Định nghĩa theo RIA (Robot Institute of America):***

Robot là một tay máy vạn năng có thể lập lại các chương trình được thiết kế để di chuyển vật liệu, chi tiết, dụng cụ hoặc các thiết bị chuyên dùng. thông qua các chương trình chuyển động có thể thay đổi để hoàn thành các nhiệm vụ khác nhau.

+ ***Định nghĩa theo TOCT 25686-85 (Nga):***

Robot công nghiệp là một máy tự động, được đặt cố định hoặc di động được, liên kết giữa một tay máy và một hệ thống điều khiển theo chương trình, có thể lập trình lại để hoàn thành các chức năng vận động và điều khiển trong quá trình sản xuất.

Robot công nghiệp đều có các đặc trưng quan trọng sau đây:

- Là thiết bị vạn năng, được tự động hóa theo chương trình và có thể lập trình lại để đáp ứng một cách linh hoạt, khéo léo các nhiệm vụ khác nhau.
- Được ứng dụng trong những trường hợp mang tính công nghiệp đặc trưng như vận chuyển và xếp dỡ nguyên vật liệu, lắp ráp, đo lường,...
- Được sử dụng thay cho con người để thực hiện các nhiệm vụ nguy hiểm hoặc lặp đi lặp lại với độ chính xác cao.

1.1.2. Lịch sử phát triển của tay máy robot công nghiệp

Vào năm 1954, tay máy robot công nghiệp đầu tiên là Unimate được thiết kế bởi George Devol và được cấp bằng sáng chế vào năm 1961. Robot này có khả năng chuyển các vật thể từ điểm này sang điểm khác trong khoảng cách khoảng một chục feet. Devol thành lập một công ty có tên Unimation vào năm 1956 để sản xuất hàng loạt robot Unimate cung cấp cho thị trường.

Kể từ đó đến nay, tay máy robot công nghiệp ngày càng được các công ty, các tổ chức, cá nhân đầu tư nghiên cứu cải tiến và áp dụng các tiến bộ kỹ thuật về điều khiển & tự động hóa, công nghệ vật liệu, vi xử lý và công nghệ thông tin,... Số lượng robot công nghiệp đã gia tăng nhanh chóng, giá thành giảm đi rõ rệt, chất lượng hoạt động cũng như các tính năng đã có nhiều bước tiến vượt bậc. Nhờ vậy robot công nghiệp đã có vị trí quan trọng trong các dây chuyền sản xuất hiện đại. Có thể kể đến một số cột mốc quan trọng trong lịch sử phát triển của tay máy robot công nghiệp như dưới đây.

a) Giai đoạn đầu tiên của robot công nghiệp ứng dụng trong sản xuất tự động (1961 – 1979)

Trong giai đoạn đầu tiên của robot công nghiệp ứng dụng vào trong sản xuất tự động, đã có những cột mốc thành tựu đáng lưu ý. Các nước đầu tư nhiều vào nghiên cứu phát triển robot và ứng dụng trong sản xuất tự động là Mỹ, Nhật Bản, Đức, các nước ở Châu Âu như Na Uy, Thụy Điển... Vào năm **1968**, Marvin Minsky phát triển một cánh tay robot giống như bạch tuộc. GM đã lắp đặt robot hàn điểm đầu tiên tại nhà máy lắp ráp Lordstown năm **1969**. Robot thị giác (Robot vision), dùng camera để hướng dẫn robot di động, được trình diễn tại Viện nghiên

cứu Stanford năm **1969**. Các robot sơn đã được phát triển để sử dụng trong nhà vào năm **1967** để phun sơn xe cút kít trong thời gian thiếu lao động ở Na Uy. Sau đó hãng Trallfa, Na Uy, cung cấp ra thị trường robot sơn đầu tiên năm **1969**.

Cũng vào năm **1969**, hãng **Hitachi** (Nhật Bản) đã phát triển robot thông minh tự động hoàn toàn dựa trên cảm biến tầm nhìn đầu tiên trên thế giới, có khả năng hoạt động lắp ráp các chi tiết từ một bản vẽ kế hoạch cho trước. Robot có thể xây dựng các khối dựa trên thông tin được tạo ra từ hình ảnh trực tiếp của bản vẽ kế hoạch lắp ráp.

Hãng KUKA của Đức chuyển từ sử dụng robot Unimate sang phát triển robot của riêng họ. Năm 1973 KUKA đã phát triển robot đầu tiên của họ là Famulus, là robot đầu tiên trên thế giới có sáu trục được điều khiển bằng cơ điện.

Năm **1973**, giáo sư Scheinmann bắt đầu sản xuất cánh tay robot **Stanford** tại hãng Vicarm, Mỹ. Cánh tay robot **Stanford** là một cánh tay robot thực hiện lắp ráp các chi tiết nhỏ được điều khiển bằng cách sử dụng tín hiệu phản hồi từ các cảm biến áp suất và cảm biến tiếp xúc. Giáo sư **Scheinman**, nhà phát triển của Stanford Arm, đã thành lập hãng Vicarm để đưa ra thị trường cánh tay máy này cho các ứng dụng công nghiệp. Phiên bản mới của tay máy này (khi đưa ra thị trường) được điều khiển bởi một máy tính mini.

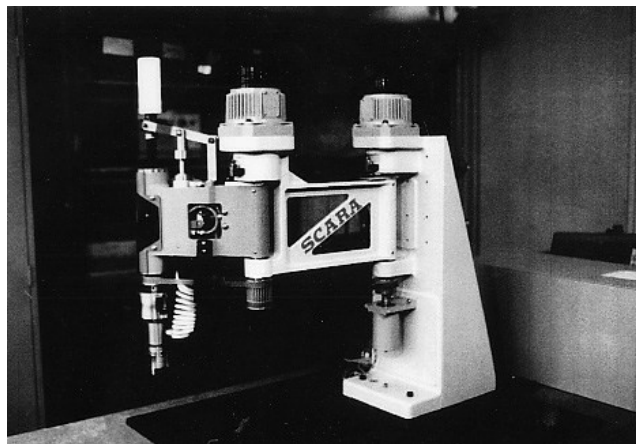
Cũng vào năm **1973**, Hitachi (Nhật Bản) đã phát triển robot bắt bu-lông tự động cho ngành công nghiệp cọc và cọc bê tông. Đây là robot công nghiệp đầu tiên có cảm biến tầm nhìn động để phát hiện các vật thể chuyển động. Nó nhận ra các bu-lông trên khuôn trong khi đang di chuyển và gắn chặt hoặc nới lỏng các bu-lông đồng bộ với chuyển động của khuôn.

Năm **1974**, Robot công nghiệp điều khiển bằng máy tính mini đầu tiên xuất hiện trên thị trường. Robot này được phát triển bởi Richard Hohn cho Tập đoàn Cincinnati Milacron. Robot được gọi là **T3**, viết tắt của **The Tomorrow Tool**.

Năm **1974**, đánh dấu sự ra đời của robot công nghiệp IRB6 của hãng ASEA (tiền thân của ABB). Đây là robot đầu tiên chạy hoàn toàn bằng điện trên thế giới. Nó cũng là robot đầu tiên được điều khiển bằng vi xử lý và sử dụng chipset đầu

tiên của hãng Intel. Với thiết kế phỏng theo hình dáng cánh tay người, chuyển động cánh tay của nó giống với cánh tay của con người, với trọng tải 6kg và 5 trục. Bộ điều khiển S1 sử dụng bộ vi xử lý 8 bit intel. Dung lượng bộ nhớ là 16KB. Bộ điều khiển có 16 I/O kỹ thuật số và được lập trình bởi 16 phím và màn hình LED bốn chữ số. Mẫu đầu tiên, IRB 6, được phát triển vào năm 1972-1973 theo sự phân công của Giám đốc điều hành ASEA, Curt Nicolin và được trưng bày lần đầu tiên vào cuối tháng 8 năm 1973. Nó được Magnussons mua lại ở Genarp để sử dụng đánh bóng ống thép không gỉ.

Năm **1975**, đánh dấu mốc giới thiệu Robot Olivetti “SIGMA”, một robot hoạt động trên hệ tọa độ Đề-các (cartesian-coordinate robot), đây là một trong những robot công nghiệp đầu tiên được sử dụng trong các dây chuyền lắp ráp. Cũng vào năm **1975**, tập đoàn ABB phát triển một robot công nghiệp tên **IRB60** có khả năng mang tải đến 60kg. Năm **1976**, tay máy robot đầu tiên ứng dụng trong không gian vũ trụ: Viking 1 và 2. **Motoman L10**, robot công nghiệp đầu tiên của hãng Yaskawa America Inc. được phát triển vào năm 1977, có năm trục và có thể di chuyển một tải trọng lượng 10kg.



Hình 1.1 – Tay máy robot SCARA (*Selective Compliance Assembly Robot Arm*)

PUMA, một cánh tay robot được thiết kế bởi Giáo sư Victor Scheinman và được phát triển bởi Vicarm, Unimation với sự hỗ trợ của General Motors vào

năm **1978**, đã được sử dụng trong các dây chuyền lắp ráp và ngày nay vẫn đang được sử dụng làm đối tượng trong các công trình nghiên cứu, giảng dạy.

Cũng vào năm **1978**, giáo sư Hiroshi Makino, ở trường Đại học Yamanashi của Nhật Bản, đã phát triển tay máy SCARA-Robot (Selective Compliance Assembly Robot Arm).

Năm **1979**, hãng Nachi, Nhật Bản, phát triển tay máy robot đầu tiên truyền động bằng động cơ điện servo. Robot này được ứng dụng trong hàn điểm, thay thế cho các robot truyền động thủy lực truyền thống.



***Hình 1.2** – Tay máy robot đầu tiên truyền động bằng động cơ điện servo của hãng Nachi, Nhật Bản*

b) Giai đoạn robot công nghiệp hiện đại (từ 1980 đến nay)

Từ năm 1980, robot công nghiệp bắt đầu được sản xuất với số lượng lớn, và ngày càng có mức độ thông minh cao hơn, độ linh hoạt, chất lượng hoạt động,.. tốt hơn. Trong giai đoạn này có thể kể đến các cột mốc phát triển đáng chú ý như sau.

Cánh tay robot công nghiệp đầu tiên với các khớp của cánh tay được truyền động trực tiếp bởi các động cơ điện được chế tạo bởi **Takeo Kanade** (một nhà khoa học người Nhật Bản) vào năm **1981**. Thiết kế này làm cho nó nhanh hơn và chính xác hơn so với các cánh tay robot công nghiệp trước đây.

Hãng Yaskawa America đã giới thiệu hệ thống điều khiển **Motoman ERC** vào năm **1988**. Nó có khả năng điều khiển tới 12 trục, số lượng trục cao nhất vào thời điểm đó. Sau đó, hệ thống điều khiển **Motoman ERC** được nâng cấp vào năm **1994** để cung cấp khả năng điều khiển tới 21 trục. Nó có thể đồng bộ hóa chuyển động của hai robot. Vào năm **1998**, bộ điều khiển **Motoman XRC** được giới thiệu có khả năng điều khiển tới 27 trục. Điều này mang lại cho nó khả năng đồng bộ hóa chuyển động của ba đến bốn robot. Vào năm **1998**, dòng robot công nghiệp **Motoman UP** được giới thiệu có ưu điểm là cánh tay robot đơn giản hơn, dễ dàng tiếp cận hơn để sửa chữa và bảo trì.

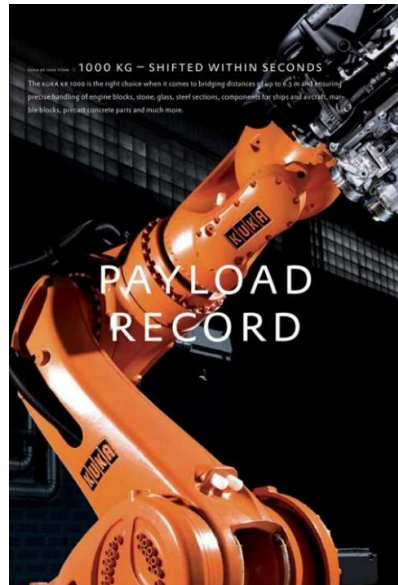
Năm **2004**, robot **Opportunity** được đưa lên sao hỏa, hoạt động 14 năm. Ra mắt vào ngày 7 tháng 7 năm 2003 như một phần của chương trình Mars Exploration Rover của NASA, nó đã hạ cánh tại Meridiani Planum vào ngày 25 tháng 1 năm 2004, ba tuần sau khi xe tự hành song sinh của nó, Spirit (MER-A) hạ cánh xuống phía bên kia hành tinh. NASA đã không liên lạc được với Opportunity kể từ ngày 10-6-2018, khi một trận bão cát kéo dài hàng tháng trời che mất ánh sáng mặt trời đến các tấm pin năng lượng của nó. Hậu quả là Opportunity cạn kiệt năng lượng do không nhận đủ quang năng. NASA đã nỗ lực liên lạc với Opportunity suốt nhiều tháng trời. Hôm 12-2, họ thực hiện nỗ lực cuối cùng sau hơn 1.000 lần thất bại nhưng vẫn không thành công.



Hình 1.3 - Robot Opportunity được đưa lên sao hỏa năm 2004

Năm 2006, hãng Comau, Italy, giới thiệu thiết bị điều khiển và lập trình cầm tay không dây đầu tiên cho tay máy robot - **The first Wireless Teach Pendant (WiTP)**

Năm 2007, KUKA, Đức, đã cho ra mắt robot tầm xa và hạng nặng đầu tiên với trọng tải 1.000 kg



***Hình 1.4** - Robot tầm xa và hạng nặng đầu tiên với trọng tải 1.000 kg vào năm 2007 của hãng KUKA, Đức*



***Hình 1.5** - Robot tầm xa và hạng nặng M-2000iA với trọng tải 1.200 kg của hãng FANUC, Nhật Bản*

Năm **2008**, FANUC, Nhật Bản, đã ra mắt robot M-2000iA hạng nặng mới với trọng tải gần 1.200kg.

Năm 2009, Yaskawa Motoman, Nhật Bản, giới thiệu hệ thống điều khiển DX100 để điều khiển đồng bộ tới 8 robot, tổng số 72 trục chuyển động.

Năm 2009, hãng ABB, Thụy Điển, giới thiệu robot công nghiệp đa năng nhỏ nhất, **IRB120**



*Hình 1.6 - Robot đa năng **IRB120** của hãng ABB*

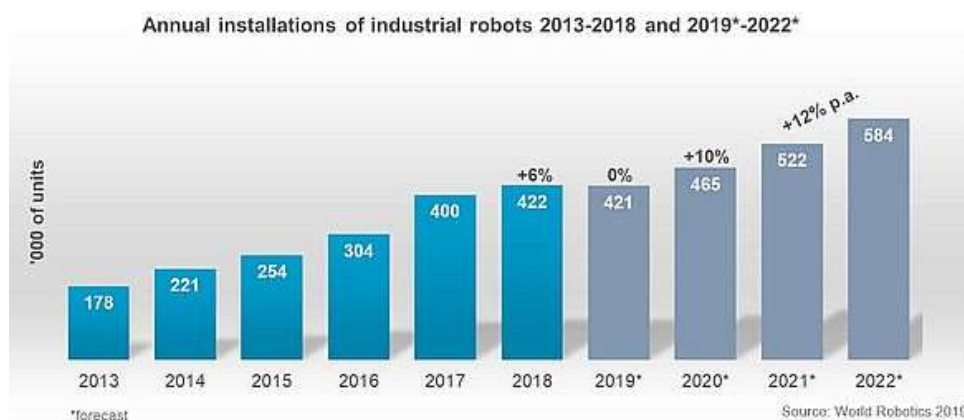


Hình 1.7 - Robot giống người đầu tiên được đưa lên vũ trụ năm 2011 của NASA

Năm 2011, Robot giống người đầu tiên được đưa lên vũ trụ - First Humanoid Robot in Space.

1.1.3. Ứng dụng của robot công nghiệp trên thế giới và tại Việt Nam

Theo số liệu thống kê của IFR năm 2019: Trung bình mỗi năm có hơn 400,000 robot được lắp đặt trong công nghiệp trên toàn thế giới. Riêng trong năm 2018, tổng số lượng robot được lắp đặt đã tăng 6% lên 422,271 robot, trị giá 16,5 tỷ USD (chưa kể phần mềm và thiết bị ngoại vi). Kết quả này gây ngạc nhiên vì các khách hàng công nghiệp chính trong lĩnh vực ô tô và điện/điện tử đã có một năm khó khăn và hai điểm đến chính, Trung Quốc và Bắc Mỹ, đã tham gia vào một cuộc chiến tranh thương mại, gây bất ổn cho nền kinh tế toàn cầu.



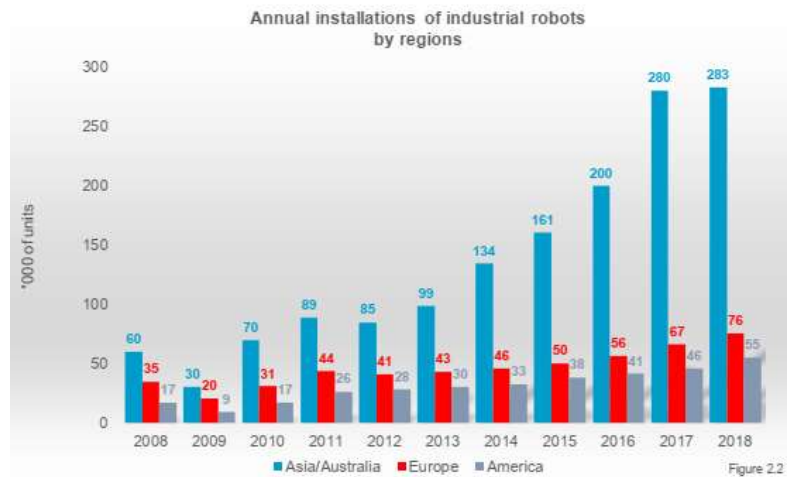
Hình 1.8 - Số lượng robot công nghiệp được lắp đặt hàng năm trên toàn thế giới giai đoạn: 2013 -2018 và dự báo giai đoạn 2019-2022

Ngành công nghiệp ô tô vẫn là khách hàng lớn nhất với 30% tổng số lắp đặt. Một số ngành công nghiệp khác: Điện/điện tử (25%), kim loại và máy móc (10%), nhựa và các sản phẩm hóa học (5%) và thực phẩm và đồ uống (3%). Ngoài ra có 19% số robot được lắp đặt nhưng không có thông tin về khách hàng.

Có 5 thị trường chính của Robot công nghiệp: **Trung Quốc, Nhật Bản, Hoa Kỳ, Hàn Quốc và Đức**. Những quốc gia này chiếm 74% số robot được lắp đặt trên toàn thế giới.

Trung Quốc là thị trường robot công nghiệp lớn nhất thế giới kể từ năm 2013 và chiếm 36% tổng số lắp đặt trong năm 2017 và 2018. Năm 2018, 154.032 robot đã được lắp đặt. Con số này ít hơn 1% so với năm 2017 (156.176 robot) nhưng

vẫn nhiều hơn số lượng robot được lắp đặt ở châu Âu và châu Mỹ cộng lại (130.772 robot).

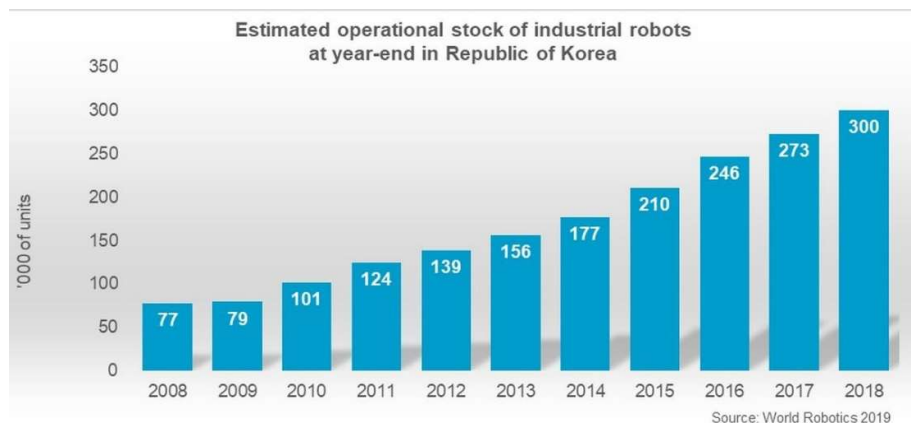


Hình 1.9 - Số lượng robot công nghiệp được lắp đặt hàng năm theo khu vực

Năm 2018, cài đặt robot tại **Nhật Bản** tăng 21% lên 55.240 robot (một đỉnh cao mới). Kể từ năm 2013, tốc độ tăng trưởng trung bình hàng năm là 17%, đáng chú ý đối với một quốc gia đã có mức độ tự động hóa cao trong sản xuất công nghiệp.

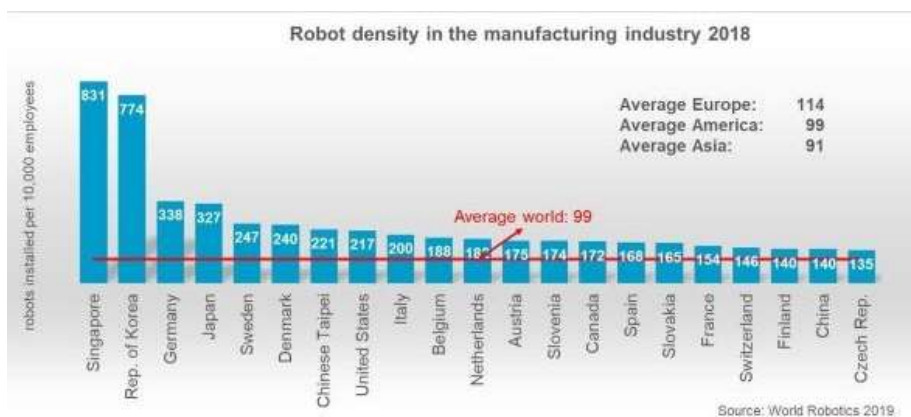
Trong năm thứ tám liên tiếp, số lượng robot được lắp đặt ở **Mỹ** đạt mức kỷ lục mới (40.373 robot; tăng 22%). Kể từ năm 2010, việc tự động hóa các quy trình sản xuất trong tất cả các ngành công nghiệp sản xuất của đất nước là xu hướng đang diễn ra. Về số lượng robot được lắp đặt hàng năm, Mỹ đã vươn lên chiếm vị trí thứ ba của Hàn Quốc vào năm 2018.

Tại **Hàn Quốc**, báo cáo Robot thế giới (World Robotics Report) cho thấy có khoảng 300.000 robot công nghiệp đang hoạt động tại Hàn Quốc vào năm 2018. Việc lắp đặt robot hàng năm đã giảm dần kể từ khi chúng đạt mức cao nhất là 41.373 robot vào năm 2016. Trong năm 2018, 37.807 robot (giảm 5%) đã được lắp đặt mới. Số robot được lắp đặt cho quốc gia này phụ thuộc mạnh mẽ vào ngành công nghiệp điện tử, vốn đã có một năm khó khăn trong năm 2018. Tuy nhiên, nhìn một cách tổng thể thì số lượng lắp đặt đã tăng trung bình 12% mỗi năm kể từ năm 2013.



Hình 1.10 – Thống kê số lượng robot công nghiệp tại Hàn Quốc theo năm

Đức là thị trường robot lớn thứ năm trên thế giới. Trong năm 2018, số lượng robot được lắp đặt tăng 26% lên mức cao nhất mới là 26.723 robot. Số lượng robot được lắp đặt tại quốc gia này chủ yếu được thúc đẩy bởi ngành công nghiệp ô tô.



Hình 1.11 – Thống kê mật độ số lượng robot công nghiệp tại một số quốc gia nổi bật vào năm 2018

Tại Việt Nam, tay máy robot đã được triển khai trong các ngành sản xuất ô tô (VINFAST, ThaCo,..), công nghệ thực phẩm, vật liệu xây dựng, luyện kim, chế tạo cơ khí, công nghiệp đóng tàu và một vài lĩnh vực khác. Trong chiến lược phát triển công nghiệp Việt Nam đến năm 2025, tầm nhìn 2035, chính phủ đã đặt mục tiêu giá trị sản phẩm công nghiệp công nghệ cao và sản phẩm ứng dụng công nghệ cao đến năm 2025 đạt khoảng 45% tổng GDP, sau năm 2025 đạt trên 50%. Trong đó, định hướng đến năm 2020 Việt Nam có thể nghiên cứu, thiết kế và sản xuất robot công nghiệp. Để làm được điều này, Việt Nam cần tập trung phát huy

nghiên cứu phát triển, làm chủ công nghệ về robot - lĩnh vực trung tâm của cuộc cách mạng công nghệ lớn.

Nhờ làn sóng đầu tư từ nước ngoài vào Việt Nam tăng cao, trong đó đáng chú ý là đầu tư của doanh nghiệp Hàn Quốc trong lĩnh vực điện tử. Các công ty trong lĩnh vực ô tô, thực phẩm, đồ uống, hàng tiêu dùng, sản xuất kim loại nói chung, lắp ráp, kiểm tra trong ngành điện tử,... có nhu cầu sử dụng robot với mức tăng trưởng cao.

1.2. TỔNG QUAN VỀ CÁC PHƯƠNG PHÁP ĐIỀU KHIỂN TAY MÁY ROBOT CÔNG NGHIỆP

Vấn đề điều khiển tay máy robot công nghiệp đã thu hút sự quan tâm của nhiều nhà nghiên cứu trên thế giới để nghiên cứu nâng cao thể mạnh của chúng. Có thể chia các phương pháp điều khiển truyền thống cho tay máy robot công nghiệp đã công bố thành 2 nhóm chính: *Phương pháp điều khiển dựa theo sai số* và *Phương pháp điều khiển dựa theo mô hình*. Phương pháp điều khiển dựa theo sai số quỹ đạo được các nhà khoa học đưa ra sớm, có ưu điểm đơn giản, dễ thực hiện vì vậy trước đây được sử dụng rộng rãi trong công nghiệp. Tuy nhiên phương pháp này không đạt được chất lượng điều khiển cao do chỉ dựa vào sai số quỹ đạo mà không tính đến các yếu tố động lực học của robot. Để đạt được chất lượng điều khiển robot tốt nhất, các phương pháp điều khiển phải được thiết kế với sự phân tích và xét đến các yếu tố động lực học của robot. Hay nói cách khác, các thành phần động lực học của robot phải được bù đầy đủ trong bộ điều khiển. Bộ điều khiển có bù đầy đủ các thành phần động lực học của robot gọi là bộ điều khiển dựa theo mô hình.

1.2.1. Điều khiển PD có bù trọng lực [1]

Phương pháp này thuộc nhóm phương pháp điều khiển dựa theo sai số. Phương pháp điều khiển này đảm bảo hệ thống ổn định tuyệt đối xung quanh điểm cân bằng ($\mathbf{q} = \mathbf{q}_d$) theo tiêu chuẩn ổn định Lyapunov. Trong đó \mathbf{q}_d là vector các biến khớp mong muốn.

Xét một tay máy robot n bậc tự do. Đặt vector biến trạng thái của hệ là $[\tilde{\mathbf{q}}^T \quad \dot{\mathbf{q}}^T]^T$, trong đó $\tilde{\mathbf{q}}^T$ là vector sai số giữa vector biến khớp mong muốn và vector biến khớp thực của robot:

$$\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q} \quad (1.1)$$

Chọn hàm Lyapunov dạng toàn phương xác định dương như sau:

$$V(\dot{\mathbf{q}}, \tilde{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{B}(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \tilde{\mathbf{q}}^T \mathbf{K}_P \tilde{\mathbf{q}} > 0, \quad \forall \dot{\mathbf{q}}, \tilde{\mathbf{q}} \neq 0 \quad (1.2)$$

trong đó \mathbf{K}_P là một ma trận dương đối xứng kích thước $(n \times n)$, $\mathbf{B}(\mathbf{q})$ là ma trận của các thành phần lực hướng tâm và Coriolis trong mô hình động lực học của robot.

Thực hiện đạo hàm hai vế của (1.2) theo thời gian, trong đó xem \mathbf{q}_d là hằng số, chúng ta có:

$$\dot{V} = \dot{\mathbf{q}}^T \mathbf{B}(\mathbf{q}) \ddot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^T \dot{\mathbf{B}}(\mathbf{q}) \dot{\mathbf{q}} - \dot{\mathbf{q}}^T \mathbf{K}_P \tilde{\mathbf{q}} \quad (1.3)$$

Tiếp theo, thay mô hình động lực học của tay máy robot vào phương trình (1.3) chúng ta có:

$$\dot{V} = \frac{1}{2} \dot{\mathbf{q}}^T (\dot{\mathbf{B}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})) \dot{\mathbf{q}} - \dot{\mathbf{q}}^T \mathbf{F} \dot{\mathbf{q}} + \dot{\mathbf{q}}^T (\mathbf{u} - \mathbf{g}(\mathbf{q}) - \mathbf{K}_P \tilde{\mathbf{q}}) \quad (1.4)$$

Trong phương trình (1.4), phần tử đầu tiên ở bên vế phải sẽ bằng 0 bởi vì tính chất của mô hình động lực học của tay máy robot. Còn phần tử thứ 2 $(-\dot{\mathbf{q}}^T \mathbf{F} \dot{\mathbf{q}})$ sẽ luôn âm do ở dạng toàn phương. Do đó, để hệ thống ổn định ($\dot{V} \leq 0$) thì luật điều khiển sẽ được chọn như sau:

$$\mathbf{u} = \mathbf{g}(\mathbf{q}) + \mathbf{K}_P \tilde{\mathbf{q}} - \mathbf{K}_D \dot{\mathbf{q}} \quad (1.5)$$

trong đó thành phần $\mathbf{g}(\mathbf{q})$ là thành phần bù trọng lực của tay máy robot.

1.2.2. Điều khiển tính mô-men (Computed torque control) [2]

Gọi $\mathbf{q}_{da}(t)$ là quỹ đạo góc quay mong muốn của các khớp chủ động của tay máy robot. Các vector sai số quỹ đạo và sai số vận tốc được định nghĩa như sau:

$$\mathbf{e} = \mathbf{q}_{da}(t) - \mathbf{q}_a(t) \quad (1.6)$$

$$\dot{\mathbf{e}} = \dot{\mathbf{q}}_{da}(t) - \dot{\mathbf{q}}_a(t) \quad (1.7)$$

Điều khiển tính mô-men là một thuật toán nổi tiếng và rất hiệu quả cho việc điều khiển chuyển động bám quỹ đạo của tay máy robot công nghiệp. Bên cạnh

đó, thuật toán này còn đảm bảo được tính ổn định toàn cục của hệ thống. Thuật toán điều khiển tính mô-men truyền thống cho tay máy robot được viết như sau:

$$\boldsymbol{\tau}_a = \mathbf{M}_a(\ddot{\mathbf{q}}_{da} + \mathbf{K}_p \mathbf{e} + \mathbf{K}_v \dot{\mathbf{e}}) + \mathbf{C}_a \dot{\mathbf{q}}_a + \mathbf{G}_a \quad (1.8)$$

Trong đó, $\mathbf{K}_p \in \mathbb{R}^{n \times n}$ và $\mathbf{K}_v \in \mathbb{R}^{n \times n}$ là các ma trận tham số xác định dương.

Phương trình thuật toán điều khiển (1.8) có thể được chia thành 2 thành phần. Thành phần thứ nhất có tác dụng bù các thành phần động lực học của robot theo quỹ đạo mong muốn:

$$\boldsymbol{\tau}_{a1} = \mathbf{M}_a \ddot{\mathbf{q}}_{da} + \mathbf{C}_a \dot{\mathbf{q}}_a + \mathbf{G}_a \quad (1.9)$$

Thành phần thứ 2 có tác dụng làm triệt tiêu sai số, đưa quỹ đạo thực của tay máy robot bám theo quỹ đạo mong muốn:

$$\boldsymbol{\tau}_{a2} = \mathbf{M}_a(\mathbf{K}_p \mathbf{e} + \mathbf{K}_v \dot{\mathbf{e}}) \quad (1.10)$$

Trước đây, thuật toán điều khiển tính mô-men truyền thống được áp dụng nhiều cho việc điều khiển bám quỹ đạo của tay máy robot nối tiếp. Gần đây thuật toán này cũng đã được nhiều nhà nghiên cứu, nhiều công trình áp dụng cho tay máy robot song song, kết quả đã có hiệu quả nhất định trong việc điều khiển tay máy robot bám theo quỹ đạo mong muốn. Tuy nhiên thuật toán điều khiển này mới chỉ xét sai số của từng khớp chủ động riêng lẻ mà chưa xét sự ảnh hưởng qua lại của các khớp chủ động với kết quả chuyển động của khâu chấp hành cuối. Đối với tay máy robot song song, do các chuỗi động học được nối kín với nhau bằng khâu chấp hành cuối nên cần thiết phải điều khiển đồng bộ giữa các khớp chủ động.

1.2.3. Điều khiển thích nghi (Adaptive control)

Trong điều khiển tay máy song song, không chỉ các lực ma sát và sai số mô hình trong các mô hình toán học biến đổi theo thời gian, mà còn có sự liên kết chặt chẽ, tính phi tuyến phức tạp vốn có và các rối loạn không xác định trong các mô hình toán học của bộ điều khiển song song. Để điều khiển bám chính xác theo các quỹ đạo mong muốn của tay máy robot, một bộ điều khiển thích nghi bền vững đã được Guoliang Tao, Xiaocong Zhu, Bin Yao và Jian Cao [3] đề xuất để

xử lý hiệu quả sai số của các tham số và các phi tuyến không chắc chắn được xem xét trong các mô hình toán học. Trong nghiên cứu này, hệ thống động lực học cho các góc quay dọc theo hai trục khác của một tay máy điều khiển song song được viết như sau:

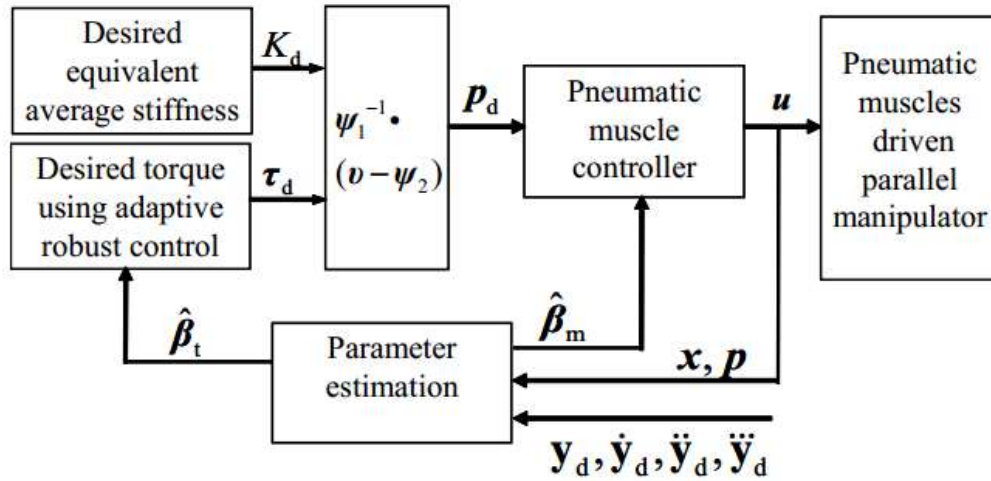
$$\tau = M(q)\ddot{q} + V(q, \dot{q})\dot{q} + G(q) + F_f(q, \dot{q}) + \tau_r(t) \quad (1.11)$$

Với M là ma trận quán tính, G là vector của lực hấp dẫn, F_f là vecto lực, τ_r là nhiễu bên ngoài

Đầu vào của bộ điều khiển thích nghi τ_d bao gồm hai phần được cho bởi:

$$\tau_d = \tau_{da} + \tau_{ds}, \tau_{da} = \varphi_2^T \hat{\beta}_t + f_{c2} \quad (1.12)$$

trong đó τ_{da} được sử dụng để bù mô hình thích nghi thông qua sự thích ứng tham số trực tuyến $\hat{\beta}_t$, τ_{ds} là một luật kiểm soát được tổng hợp



Hình 1.12 - Sơ đồ khối bộ điều khiển thích nghi

cho tay máy robot song song truyền động bằng cơ nhân tạo

Bộ điều khiển thích nghi của tay máy song song trước tiên được kiểm tra để theo dõi các vị trí mong muốn dưới cùng điều kiện ngoại trừ độ cứng vững trung bình. Kết quả thực nghiệm thấy, độ cứng vững trung bình tương đương của tay máy lớn đòi hỏi lực bên trong cao của cơ cấu và nỗ lực kiểm soát lớn hơn trong khi điều khiển nhỏ hơn. Do độ trễ lớn và các lực ma sát thay đổi theo thời gian gây ra bởi áp suất lớn hơn, các sai số theo dõi tăng nhẹ. Ngược lại, độ cứng trung bình tương đương nhỏ hơn dẫn đến nội lực bên trong thấp và các nỗ lực kiểm

soát nhỏ hơn trong khi mang lại nhiều sự thuận tiện trong điều khiển hơn do sự nhạy cảm của hệ thống đối với nhiễu loạn và tiếng ồn đo lường dưới áp suất thấp hơn. Các kết quả chứng minh rằng khả năng điều khiển với sự hạn chế độ cứng trung bình đối với bộ điều khiển song song không chỉ đảm bảo các sai số theo dõi nhỏ mà còn giảm thiểu độ lớn của tín hiệu điều khiển và tiết kiệm năng lượng.

Bên cạnh đó, hai thuật toán điều khiển không gian Cartesian thích nghi được Denis Garagić và Krishnaswamy Srinivasan [4] đề xuất để bù thành phần ma sát cho các tay máy robot có cấu trúc Stewart. Bộ điều khiển đầu tiên sử dụng một sơ đồ bù ma sát thích nghi dựa trên mô hình ma sát tuyến tính trong thông số mô phỏng. Thuật toán bù ma sát được đề xuất một cách rõ ràng giải thích cho các lực bình thường thay đổi theo thời gian cũng như sự phụ thuộc của hệ số ma sát trên vận tốc. Đặc tính ma sát Stribeck và ma sát tĩnh khác nhau được coi là nhiễu loạn biên, và được bù bằng bộ điều khiển độ trượt.

$$u = Y_I(q, \dot{q}_r, \ddot{q}_r) \hat{\Theta}_I + Y_F(q, \dot{q}_r, \ddot{q}_r) \hat{\Theta}_F + K_D \sigma + u_{sl} \quad (1.13)$$

Với Y_F là ma trận hồi quy của mô hình ma sát tuyến tính, $\hat{\Theta}_I$ và $\hat{\Theta}_F$ là thông số momen và ma sát ước tính được cập nhật nhờ bộ điều khiển thích nghi, u_{sl} là mặt trượt được thêm vào để tăng độ mịn của bộ điều khiển.

Trong bộ điều khiển thứ hai, một dạng mờ của hệ thống mờ nhiều đầu vào Takagi-Sugeno Multi-Input được phát triển để thích nghi tìm ma sát không xác định được và bù đắp cho nó.

$$u = Y_I \Theta_I + K_D \sigma + u_{sl} + Z^* \hat{A}_{FR} \Xi^* \quad (1.14)$$

Với \hat{A}_{FR} là các thông số của bộ điều khiển mờ

Qua kết quả thực nghiệm trong [4], hiệu suất của hai bộ điều khiển thích nghi được đề xuất với bù ma sát được áp dụng trên mô phỏng cho một số quỹ đạo không gian Descartes đại diện, và so sánh với đáp ứng của bộ điều khiển tích hợp PID và mômen trong không gian Descartes (CCT-PID). Kết quả thực nghiệm cho thấy bộ điều khiển được đề xuất (thứ nhất) hoạt động tốt hơn bộ điều khiển CCT-

PID. Các lỗi theo dõi lớn gây ra bởi ma sát lúc đảo chiều vận tốc bị giảm rất nhiều khi sử dụng các bộ điều khiển thích nghi.

CHƯƠNG 2

MÔ HÌNH ĐỘNG LỰC HỌC VÀ PHƯƠNG PHÁP ĐIỀU KHIỂN TRƯỢT BẬC CAO CHO TAY MÁY ROBOT CÔNG NGHIỆP

2.1. GIỚI THIỆU

Mô hình động lực học của tay máy robot công nghiệp đóng vai trò quan trọng trong việc mô phỏng chuyển động của robot, phân tích cấu trúc, cũng như để làm cơ sở thiết kế các bộ điều khiển robot. Có 2 phương pháp thông dụng để xây dựng mô hình động lực học của tay máy robot: Phương pháp sử dụng công thức Lagrange và phương pháp Newton-Euler. Trong đề tài sử dụng phương pháp dùng công thức Lagrange bởi sự hiệu quả và phù hợp của nó đối với mục tiêu nghiên cứu của đề tài.

2.2. XÂY DỰNG MÔ HÌNH ĐỘNG LỰC HỌC CHO MỘT TAY MÁY ROBOT CÔNG NGHIỆP DẠNG TỔNG QUÁT

2.2.1. Công thức Lagrange

Mô hình động lực học của một tay máy robot mô tả mối quan hệ giữa các mô-men truyền động cho các khớp và chuyển động của robot.

Với phương pháp sử dụng công thức Lagrange, các phương trình chuyển động có thể được thiết lập một cách hệ thống độc lập với các hệ tọa độ tham chiếu. Trước hết, một tập hợp các biến q_i , $i = 1, \dots, n$, gọi là các tọa độ tổng quát, được chọn để mô tả một cách có hiệu quả vị trí của các thanh nối của một tay máy robot công nghiệp n bậc tự do. Sau đó, hàm Lagrange được định nghĩa là một hàm của các hệ tọa độ cơ sở có dạng như sau:

$$L = T - U \quad (2.1)$$

trong đó T và U lần lượt là tổng động năng và tổng thế năng của cả hệ thống.

Các phương trình Lagrange được biểu diễn như sau:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = f_i, \quad i = 1, \dots, n \quad (2.2)$$

trong đó f_i ($i = 1, \dots, n$) là lực cơ sở tương ứng với tọa độ q_i .

Đối với tay máy robot có cấu trúc động học hở, hệ tọa độ cơ sở được chọn là một vector biến khớp:

$$[q_1, \dots, q_n] = \mathbf{q} \quad (2.3)$$

Các lực cơ sở gồm tác dụng vào robot gồm có các mô-men truyền động tại các khớp, các lực ma sát tại các khớp cũng như các mô-men được tạo ra bởi sự tương tác giữa khâu chấp hành cuối với môi trường.

Các phương trình (2.2) thiết lập mối quan hệ giữa các lực cơ sở tác dụng vào robot và các biến vị trí, tốc độ và gia tốc của các khớp. Vì vậy chúng cho phép xây dựng mô hình động lực học của tay máy robot xuất phát từ việc xác định tổng động năng và thế năng của hệ thống cơ khí.

2.2.2. Tính tổng động năng của hệ thống

Xét một tay máy robot công nghiệp có n thanh cứng liên kết với nhau. Tổng động năng được tính bởi tổng của động năng chuyển động các thanh và động năng của các khớp có cơ cấu truyền động:

$$T = \sum_{i=1}^n (T_{l_i} + T_{m_i}) \quad (2.4)$$

trong đó T_{l_i} là động năng của thanh thứ i và T_{m_i} là động năng của động cơ truyền động cho khớp thứ i .

a) Động năng của thanh

Động năng của thanh thứ i được tính như sau:

$$T_{l_i} = \frac{1}{2} \int_{V_{l_i}} \dot{\mathbf{p}}_i^{*T} \dot{\mathbf{p}}_i^* \rho dV \quad (2.5)$$

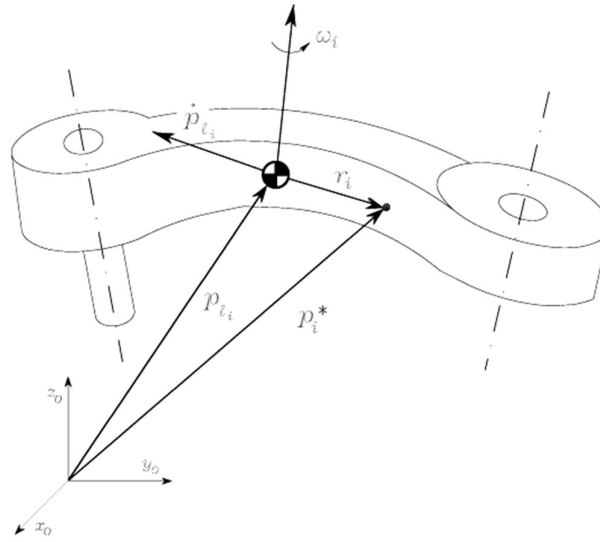
trong đó $\dot{\mathbf{p}}_i^*$ là vector vận tốc tuyến tính và ρ là khối lượng riêng của một phần tử có thể tích dV ; V_{l_i} là thể tích của link thứ i .

Vận tốc tuyến tính của một điểm trên thanh được tính bởi công thức sau:

$$\begin{aligned}\dot{\mathbf{p}}_i^* &= \dot{\mathbf{p}}_{l_i} + \boldsymbol{\omega}_i \times \mathbf{r}_i \\ &= \dot{\mathbf{p}}_{l_i} + \mathbf{S}(\boldsymbol{\omega}_i) \mathbf{r}_i\end{aligned}\quad (2.6)$$

trong đó $\dot{\mathbf{p}}_{l_i}$ là vector vận tốc tuyến tính của tâm phần tử, $\boldsymbol{\omega}_i$ là vận tốc góc của thanh thứ i và \mathbf{r}_i là vector vị trí của phần tử so với tâm của thanh xét trong hệ tọa độ cơ sở:

$$\mathbf{r}_i = [r_{ix} \ r_{iy} \ r_{iz}]^T = \mathbf{p}_i^* - \mathbf{p}_{l_i} \quad (2.7)$$



Hình 2.1 – Mô tả động học của thanh thứ i
cho việc thiết lập công thức Lagrange

Từ các phương trình (2.6) và (2.7) thay vào phương trình (2.5) chúng ta có được phương trình động năng cho các chuyển động tịnh tiến và chuyển động quay như sau:

+ **Chuyển động tịnh tiến:**

$$\frac{1}{2} \int_{V_{l_i}} \dot{\mathbf{p}}_{l_i}^T \dot{\mathbf{p}}_{l_i} \rho dV = \frac{1}{2} m_{l_i} \dot{\mathbf{p}}_{l_i}^T \dot{\mathbf{p}}_{l_i} \quad (2.8)$$

+ **Chuyển động quay:**

$$\frac{1}{2} \int_{V_i} \mathbf{r}_i^T \mathbf{S}^T(\boldsymbol{\omega}_i) \mathbf{S}(\boldsymbol{\omega}_i) \mathbf{r}_i \rho dV = \frac{1}{2} \boldsymbol{\omega}_i^T \mathbf{I}_{l_i} \boldsymbol{\omega}_i \quad (2.9)$$

trong đó \mathbf{I}_{l_i} là mô-men quán tính quay đối với tâm của thanh thứ i biểu diễn trong hệ tọa độ cơ sở. Chú ý rằng vị trí của thanh thứ i phụ thuộc vào hình dạng của tay máy robot và do đó mô-men quán tính khi biểu diễn trong hệ tọa độ cơ sở cũng phụ thuộc vào hình dạng tay máy robot.

Vận tốc góc của thanh thứ i biểu diễn trong hệ tọa độ gắn với thanh được viết bởi công thức:

$$\boldsymbol{\omega}_i^i = \mathbf{R}_i^T \boldsymbol{\omega}_i \quad (2.10)$$

trong đó \mathbf{R}_i là ma trận quay của hệ tọa độ gắn với thanh thứ i so với hệ tọa độ cơ sở.

Khi xét trong hệ tọa độ gắn với thanh, thì mô-men quán tính $\mathbf{I}_{l_i}^i$ là một hằng số. Phương trình biểu diễn mối quan hệ của $\mathbf{I}_{l_i}^i$ và \mathbf{I}_{l_i} như sau:

$$\mathbf{I}_{l_i} = \mathbf{R}_i \mathbf{I}_{l_i}^i \mathbf{R}_i^T \quad (2.11)$$

Cuối cùng, chúng ta có phương trình biểu diễn tổng động năng của thanh thứ i như sau:

$$T_{l_i} = \frac{1}{2} m_{l_i} \dot{\mathbf{p}}_{l_i}^T \dot{\mathbf{p}}_{l_i} + \frac{1}{2} \boldsymbol{\omega}_i^T \mathbf{R}_i \mathbf{I}_{l_i}^i \mathbf{R}_i^T \boldsymbol{\omega}_i \quad (2.12)$$

Đến đây, chúng ta thấy sự cần thiết phải biểu diễn tổng động năng này thành một hàm số của hệ tọa độ cơ sở \mathbf{q} của hệ thống. Để thực hiện được điều này, phương pháp tính toán hình học với ma trận Jacobian được áp dụng:

$$\dot{\mathbf{p}}_{l_i} = \mathbf{J}_{p1}^{(l_i)} \dot{\mathbf{q}}_1 + \dots + \mathbf{J}_{pi}^{(l_i)} \dot{\mathbf{q}}_i = \mathbf{J}_p^{(l_i)} \dot{\mathbf{q}} \quad (2.13)$$

$$\boldsymbol{\omega}_i = \mathbf{J}_{o1}^{(l_i)} \dot{\mathbf{q}}_1 + \dots + \mathbf{J}_{oi}^{(l_i)} \dot{\mathbf{q}}_i = \mathbf{J}_o^{(l_i)} \dot{\mathbf{q}} \quad (2.14)$$

Trong đó các vector cột của các ma trận Jacobian tương ứng với các vận tốc của khớp đã được tính cộng dồn cho đến thanh nối hiện tại thứ i . Các ma trận Jacobian được biểu diễn thành các vector cột cụ thể như sau:

$$\mathbf{J}_P^{(l_i)} = [\mathbf{J}_{P1}^{(l_i)} \dots \mathbf{J}_{Pi}^{(l_i)} \mathbf{0} \dots \mathbf{0}] \quad (2.15)$$

$$\mathbf{J}_O^{(l_i)} = [\mathbf{J}_{O1}^{(l_i)} \dots \mathbf{J}_{Oi}^{(l_i)} \mathbf{0} \dots \mathbf{0}] \quad (2.16)$$

Cuối cùng, chúng ta có được phương trình động năng của thanh thứ i viết trong mối quan hệ với hệ tọa độ cơ sở như sau:

$$T_i = \frac{1}{2} m_i \dot{\mathbf{q}}^T \mathbf{J}_P^{(l_i)T} \mathbf{J}_P^{(l_i)} \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{J}_O^{(l_i)T} \mathbf{R}_i \mathbf{I}_i^T \mathbf{R}_i \mathbf{J}_O^{(l_i)} \dot{\mathbf{q}} \quad (2.17)$$

b) Động năng của động cơ truyền động cho khớp

Động năng của động cơ truyền động tại khớp thứ i được tính theo cách tương tự như tính động năng của thanh. Chúng ta xét trường hợp điển hình là sử dụng các động cơ điện để truyền động. Giả thiết rằng khi gắn động cơ lên thanh thứ $i-1$ để truyền động cho khớp thứ i như ở Hình 2.2 thì phần cố định của động cơ (stator) đã được tính trong thanh đó. Do đó chúng ta chỉ tính đến phần động năng của rotor cho khớp thứ i .

Phương trình động năng của rotor i được viết như sau:

$$T_{m_i} = \frac{1}{2} m_{m_i} \dot{\mathbf{p}}_{m_i}^T \dot{\mathbf{p}}_{m_i} + \frac{1}{2} \omega_{m_i}^T \mathbf{I}_{m_i} \omega_{m_i} \quad (2.18)$$

trong đó m_{m_i} là khối lượng của rotor; $\dot{\mathbf{p}}_{m_i}$ là vận tốc tuyến tính của tâm của rotor; \mathbf{I}_{m_i} là mô-men quán tính của rotor trong chuyển động tương đối với tâm của nó; và ω_{m_i} là vận tốc góc của rotor.

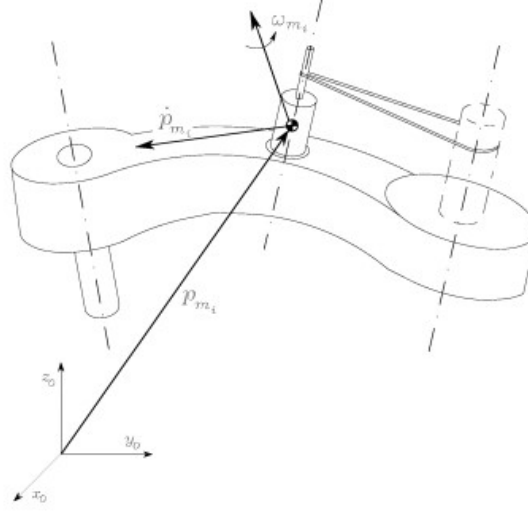
Ký hiệu ϑ_{m_i} là vị trí góc của rotor, chúng ta có:

$$k_i \dot{q}_i = \dot{\vartheta}_{m_i} \quad (2.19)$$

trong đó k_i là tỷ số truyền của hộp số.

Để biểu diễn động năng của rotor là một hàm của các biến khớp, chúng ta cần biểu diễn chuyển đổi tốc độ tuyến tính sang biến khớp bằng phương trình sau:

$$\dot{\mathbf{p}}_{m_i} = \mathbf{J}_P^{(m_i)} \dot{\mathbf{q}} \quad (2.20)$$



Hình 2.2 – Mô tả động học của motor thứ i
cho việc thiết lập công thức Lagrange

Trong đó ma trận Jacobian của chuyển động tuyến tính được tính bởi công thức:

$$\mathbf{J}_P^{(m_i)} = \begin{bmatrix} J_{P_1}^{(m_i)} & \dots & J_{P_{i-1}}^{(m_i)} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \quad (2.21)$$

Chú ý rằng $J_{P_i}^{(m_i)} = \mathbf{0}$ bởi vì tâm của rotor trùng với trục quay đã chọn.

Tương tự, phương trình biểu diễn mối quan hệ giữa vận tốc quay và biến khớp được biểu diễn như sau:

$$\omega_{m_i} = \mathbf{J}_O^{(m_i)} \dot{\mathbf{q}} \quad (2.22)$$

Trong đó:

$$\mathbf{J}_O^{(m_i)} = \begin{bmatrix} J_{O_1}^{(m_i)} & \dots & J_{O_{i-1}}^{(m_i)} & J_{O_i}^{(m_i)} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \quad (2.23)$$

Thay các công thức trên vào (2.18) chúng ta có:

$$T_{m_i} = \frac{1}{2} m_{m_i} \dot{\mathbf{q}}^T \mathbf{J}_P^{(m_i)T} \mathbf{J}_P^{(m_i)} \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{J}_O^{(m_i)T} \mathbf{R}_{m_i} \mathbf{I}_{m_i}^T \mathbf{R}_{m_i}^T \mathbf{J}_O^{(m_i)} \dot{\mathbf{q}} \quad (2.24)$$

Cuối cùng, bằng cách cộng tổng các thành phần động năng của tất cả các thanh và khớp của robot chúng ta có phương trình tổng động năng của robot ở dạng toàn phương như sau:

$$T = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n M_{ij}(\mathbf{q}) \dot{q}_i \dot{q}_j = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \quad (2.25)$$

trong đó \mathbf{M} là ma trận quán tính, được tính bởi công thức:

$$\begin{aligned} \mathbf{M}(\mathbf{q}) = \sum_{i=1}^n & \left(m_{l_i} \mathbf{J}_P^{(l_i)T} \mathbf{J}_P^{(l_i)} + \mathbf{J}_O^{(l_i)T} \mathbf{R}_{l_i} \mathbf{I}_{l_i}^T \mathbf{R}_{l_i}^T \mathbf{J}_O^{(l_i)} + \right. \\ & \left. + m_{m_i} \mathbf{J}_P^{(m_i)T} \mathbf{J}_P^{(m_i)} + \mathbf{J}_O^{(m_i)T} \mathbf{R}_{m_i} \mathbf{I}_{m_i}^T \mathbf{R}_{m_i}^T \mathbf{J}_O^{(m_i)} \right) \end{aligned} \quad (2.26)$$

2.2.3. Tính tổng thế năng của hệ thống

Tương tự như cách tính động năng, tổng thế năng của tay máy robot được tính bằng tổng thế năng của mỗi thành phần bao gồm thế năng của các thanh và của các rotor:

$$U = \sum_{i=1}^n (U_{l_i} + U_{m_i}) \quad (2.27)$$

Giả thiết các thanh nối của robot là tuyệt đối cứng, thế năng tạo ra bởi trọng lượng của các thanh này được tính bởi công thức:

$$U_{l_i} = - \int_{V_{l_i}} \mathbf{g}_0^T \mathbf{p}_l^* \rho dV = -m_{l_i} \mathbf{g}_0^T \mathbf{p}_{l_i} \quad (2.28)$$

trong đó \mathbf{g}_0 là vector gia tốc biểu diễn trong hệ tọa độ cơ sở (Ví dụ $\mathbf{g}_0 = [0, 0, -g]^T$ nếu như chọn trục z là trục thẳng đứng).

Tương tự, thế năng của rotor được tính bởi công thức:

$$U_{m_i} = -m_{m_i} \mathbf{g}_0^T \mathbf{p}_{m_i} \quad (2.29)$$

Thay các công thức (2.28) và (2.29) vào (2.27) chúng ta có:

$$U = -\sum_{i=1}^n \left(m_{l_i} \mathbf{g}_0^T \mathbf{p}_{l_i} + m_{m_i} \mathbf{g}_0^T \mathbf{p}_{m_i} \right) \quad (2.30)$$

trong đó, các vector vị trí \mathbf{p}_{l_i} và \mathbf{p}_{m_i} chỉ phụ thuộc vào các biến khớp \mathbf{q} và không có chứa $\dot{\mathbf{q}}$.

2.2.4. Các phương trình mô tả chuyển động

Sau khi tính toán tổng động năng và thế năng của hệ, ta có hàm Lagrange như sau:

$$\begin{aligned} L(\mathbf{q}, \dot{\mathbf{q}}) &= T(\mathbf{q}, \dot{\mathbf{q}}) - U(\mathbf{q}) \\ &= \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n M_{ij}(\mathbf{q}) \dot{q}_i \dot{q}_j + \sum_{i=1}^n \left(m_{l_i} \mathbf{g}_0^T \mathbf{p}_{l_i}(\mathbf{q}) + m_{m_i} \mathbf{g}_0^T \mathbf{p}_{m_i}(\mathbf{q}) \right) \end{aligned} \quad (2.31)$$

Bước tiếp theo là tính các thành phần đạo hàm trong phương trình Lagrange (2.2), lưu ý rằng thành phần thế năng U không phụ thuộc $\dot{\mathbf{q}}$:

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) &= \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i} \right) = \sum_{j=1}^n M_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^n \frac{dM_{ij}(\mathbf{q})}{dt} \dot{q}_j \\ &= \sum_{j=1}^n M_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n \frac{dM_{ij}(\mathbf{q})}{dq_k} \dot{q}_k \dot{q}_j \end{aligned} \quad (2.32)$$

$$\frac{\partial T}{\partial q_i} = \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \frac{dM_{ij}(\mathbf{q})}{dq_i} \dot{q}_k \dot{q}_j \quad (2.33)$$

$$\begin{aligned} \frac{\partial U}{\partial q_i} &= -\sum_{j=1}^n \left(m_{l_i} \mathbf{g}_0^T \frac{\partial \mathbf{p}_{l_j}}{\partial q_i} + m_{m_i} \mathbf{g}_0^T \frac{\partial \mathbf{p}_{m_j}}{\partial q_i} \right) \\ &= -\sum_{j=1}^n \left(m_{l_i} \mathbf{g}_0^T \mathbf{J}_{P_i}^{(l_j)}(\mathbf{q}) + m_{m_i} \mathbf{g}_0^T \mathbf{J}_{P_i}^{(m_j)}(\mathbf{q}) \right) = g_i(\mathbf{q}) \end{aligned} \quad (2.34)$$

Cuối cùng, chúng ta có các phương trình chuyển động được viết như sau:

$$\sum_{i=1}^n M_{ij}(\mathbf{q})\dot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk}(\mathbf{q})\dot{q}_k\dot{q}_j + g_i(\mathbf{q}) = f_i \quad i=1,\dots,n \quad (2.35)$$

Trong đó:

$$h_{ijk} = \frac{\partial M_{ij}}{\partial q_k} - \frac{1}{2} \frac{\partial M_{jk}}{\partial q_i} \quad (2.36)$$

Xét đến trường hợp tồn tại các lực ma sát, chúng ta có thêm các thành phần ma sát trượt và ma sát tĩnh. Thành phần ma sát trượt cho bởi công thức $\mathbf{F}_v \dot{\mathbf{q}}$, trong đó \mathbf{F}_v là ma trận đường chéo ($n \times n$) của các hệ số ma sát trượt. Một mô hình đơn giản của các thành phần ma sát tĩnh được biểu diễn bởi công thức $\mathbf{F}_s \text{sign}(\dot{\mathbf{q}})$, trong đó \mathbf{F}_s là ma trận đường chéo ($n \times n$) của các hệ số ma sát tĩnh, và $\text{sign}(\dot{\mathbf{q}})$ là một vector ($n \times 1$) của các hàm dấu của các vận tốc khớp.

Ngoài ra, các thành phần nhiễu loạn tác động từ bên ngoài lên robot cũng được tính đến, ký hiệu là $\boldsymbol{\tau}_d$.

Sau khi xét các thành phần bất định nêu trên, mô hình động lực học của tay máy robot (2.35) được viết lại ở dạng ma trận như sau:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{F}_v \dot{\mathbf{q}} + \mathbf{F}_s \text{sign}(\dot{\mathbf{q}}) + \boldsymbol{\tau}_d = \boldsymbol{\tau} \quad (2.37)$$

Trong đó \mathbf{C} là ma trận của các thành phần lực hướng tâm và Coriolis thỏa mãn phương trình sau:

$$\sum_{i=1}^n C_{ij}\dot{q}_j = \sum_{j=1}^n \sum_{k=1}^n h_{ijk}\dot{q}_k\dot{q}_j \quad (2.38)$$

Mô hình động lực học (2.37) của tay máy robot công nghiệp được sử dụng để làm cơ sở cho việc phân tích động lực học và thiết kế các bộ điều khiển chuyển động của tay máy robot ở các phần tiếp theo.

2.3. PHƯƠNG PHÁP ĐIỀU KHIỂN TRƯỢT TRUYỀN THỐNG CHO TAY MÁY ROBOT CÔNG NGHIỆP

Phương pháp điều khiển trượt là một trong những kỹ thuật điều khiển hiệu quả nhất trong điều khiển tay máy robot công nghiệp. Trong phương pháp điều

khíến trượt, vector các biến trạng thái được đưa đến mặt trượt và trượt trên mặt trượt s cho trước bằng các luật điều khiển không liên tục. Phương pháp này cung cấp độ chính xác cao và bền vững đối với các nhiễu loạn bên ngoài, sự biến thiên của tham số, ma sát trên các khớp, nhiễu của cảm biến và các thành phần bất định xảy ra trong hệ thống như tính dẻo của các khớp, các thành phần động lực học chưa mô hình hóa được,...

Xét đối tượng tay máy robot n bậc tự do có mô hình động lực học như phương trình (2.37). Sau đó, phân tích chi tiết của các ma trận tham số thành các thành phần có thể xác định chính xác và các thành phần sai số mô hình: $\mathbf{M} = \hat{\mathbf{M}} + \Delta\mathbf{M}$; $\mathbf{C} = \hat{\mathbf{C}} + \Delta\mathbf{C}$; $\mathbf{G} = \hat{\mathbf{G}} + \Delta\mathbf{G}$ là các thông số động lực học chính xác của đối tượng tay máy robot với $\Delta\mathbf{M}$, $\Delta\mathbf{C}$, $\Delta\mathbf{G}$ là các sai số mô hình hóa.

Gọi $\Delta\tau$ là vector tổng các thành phần bất định do sai số mô hình hóa và tín hiệu nhiễu ngoài, ta có:

$$\Delta\tau = \Delta\mathbf{M}\ddot{\mathbf{q}} + \Delta\mathbf{C}\dot{\mathbf{q}} + \Delta\mathbf{G} + \mathbf{d}(t) \quad (2.39)$$

Ta giả thiết rằng $\Delta\tau$ thỏa điều kiện: $|\Delta\tau_i| \leq |\Delta\tau_i|_{bound}$

Như vậy, phương trình động lực học của tay máy robot n bậc tự do có thể viết lại như sau:

$$\hat{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \hat{\mathbf{G}}(\mathbf{q}) + \Delta\tau = \tau \quad (2.30)$$

Gọi $\mathbf{q}_d \in \mathbf{R}^n$ là vector quỹ đạo mong muốn của tay máy robot. Vector sai lệch được định nghĩa như sau:

$$\begin{aligned} \mathbf{e} &= \mathbf{q} - \mathbf{q}_d \\ \dot{\mathbf{e}} &= \dot{\mathbf{q}} - \dot{\mathbf{q}}_d \\ \ddot{\mathbf{e}} &= \ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d \end{aligned} \quad (2.31)$$

Định nghĩa mặt trượt như sau:

$$\mathbf{s} = \dot{\mathbf{e}} + \wedge \mathbf{e} = \dot{\mathbf{q}} - (\dot{\mathbf{q}}_d - \wedge \mathbf{e}) = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r \quad (2.32)$$

Trong đó:

$\wedge = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ là ma trận chéo thể hiện đặc tính chuyển động của mặt trượt;

$\lambda_1, \lambda_2, \dots, \lambda_n$ là các hằng số dương;

$\dot{\mathbf{q}}_r = \dot{\mathbf{q}}_d - \wedge \mathbf{e}$ được định nghĩa là vector vận tốc tham chiếu.

Ta cần thiết kế luật điều khiển $\boldsymbol{\tau}$ sao cho quỹ đạo trạng thái của hệ thống hướng về mặt trượt và giữ lại trên đó. Về mặt tổng quát, thuật toán điều khiển trượt truyền thống (τ) bao gồm 2 thành phần:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{eq} + \boldsymbol{\tau}_{sw} \quad (2.33)$$

Trong đó, thành phần $\boldsymbol{\tau}_{eq} \in \mathbf{R}^n$ là tín hiệu điều khiển cân bằng, giúp giữ quỹ đạo trạng thái của hệ thống ở lại trên mặt trượt.

Thành phần $\boldsymbol{\tau}_{sw} \in \mathbf{R}^n$ là tín hiệu điều khiển chuyển mạch (không liên tục) với nhiệm vụ là kéo trạng thái hệ thống quay về mặt trượt mỗi khi hệ thống bị đẩy ra ngoài mặt trượt do tác động của các thành phần bất định (nhiều, ma sát, v.v.). Thành phần điều khiển cân bằng được xác định cho hệ thống ở trạng thái bình thường, không có tác động của thành phần bất định và nhiễu bên ngoài như sau:

$$\boldsymbol{\tau}_{eq} = \widehat{\mathbf{M}}(\mathbf{q})\mathbf{q} + \widehat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \widehat{\mathbf{G}}(\mathbf{q}) \quad (2.34)$$

Tín hiệu điều khiển chuyển mạch được thiết kế như sau:

$$\boldsymbol{\tau}_{sw} = -\mathbf{K} \text{sign}(\mathbf{s}) \quad (2.35)$$

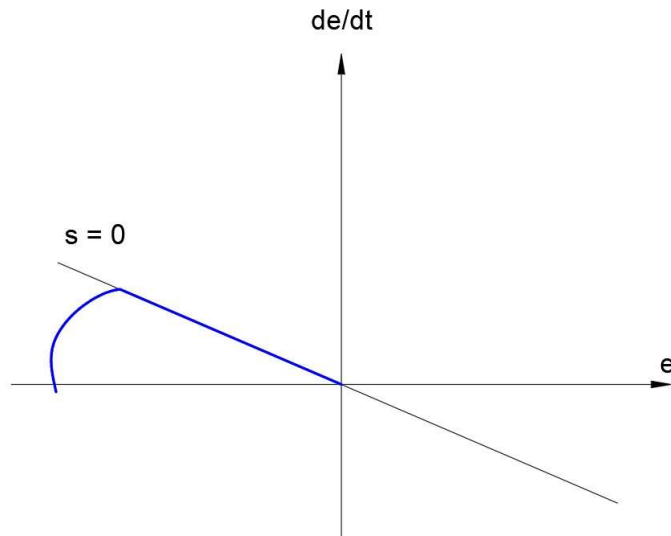
Với $\mathbf{K} = \text{diag}(k_1, k_2 \dots k_n)$ là ma trận đường chéo với các hệ số hằng dương, được chọn sao cho $k_i \geq |\Delta \tau_i|_{bound}$.

Từ (2.33), (2.34) và (2.35) chúng ta có luật điều khiển trượt truyền thống cho tay máy robot như sau:

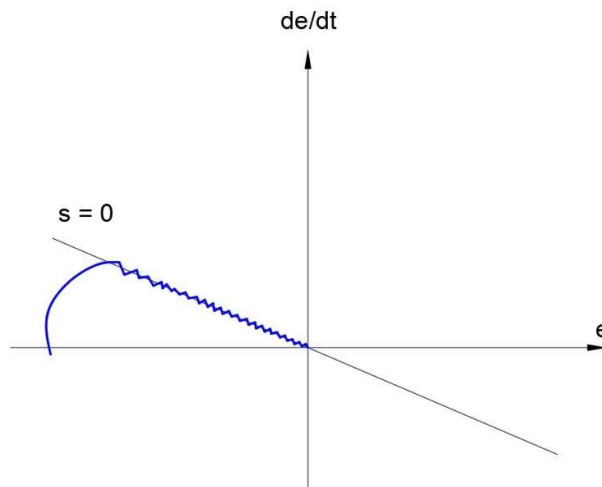
$$\boldsymbol{\tau} = \widehat{\mathbf{M}}(\mathbf{q})\mathbf{q} + \widehat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \widehat{\mathbf{G}}(\mathbf{q}) - \mathbf{K} \text{sign}(\mathbf{s}) \quad (2.36)$$

Sự ổn định của hệ thống với luật điều khiển trượt như trên có thể được chứng minh khi chọn hàm Lyapunov (được đề xuất trong tài liệu [5]):

$$V = \frac{1}{2} \mathbf{s}^T \widehat{\mathbf{M}} \mathbf{s} \quad (2.37)$$

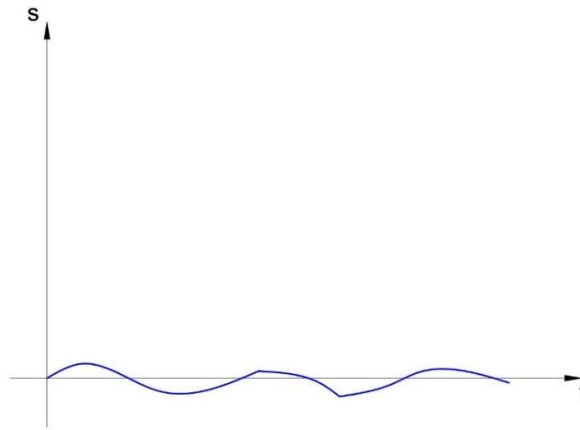


Hình 2.3 - Điểm trạng thái trượt trên mặt trượt s và tiến về gốc tọa độ, không có hiện tượng chattering

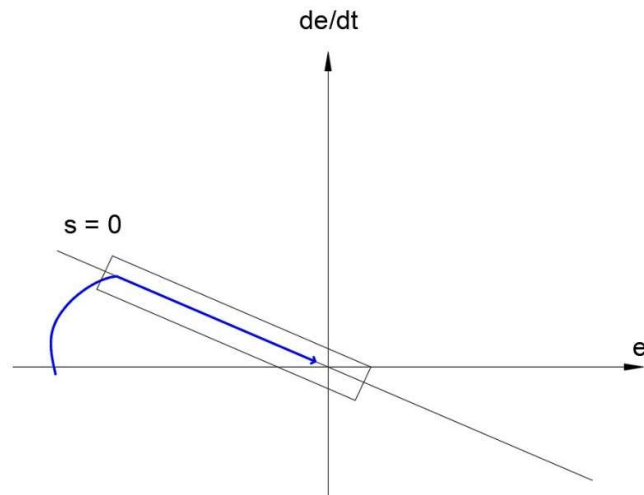


Hình 2.4 - Điểm trạng thái trượt trên mặt trượt s và tiến về gốc tọa độ, xảy ra hiện tượng chattering

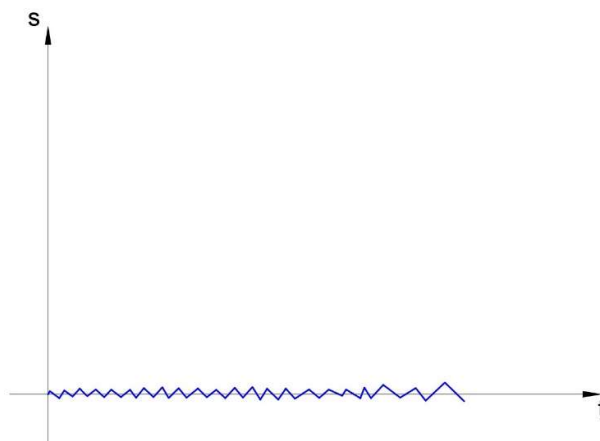
Trong thành phần τ_{sw} (thành phần chuyển mạch của thuật toán điều khiển trượt), hàm dấu $sign$ gây ra hiện tượng chuyển mạch ở tần số cao, từ đó gây ra hiện tượng rung động trên mặt trượt s , gọi là hiện tượng chattering. Hiện tượng này gây ra sự rung động nội tại cho cơ cấu chấp hành của cánh tay robot (thường là động cơ bị đảo chiều đột ngột và liên tục), tăng hao mòn và giảm tuổi thọ của cơ cấu chấp hành.



Hình 2.5 – Đồ thị hàm mặt trượt s theo thời gian t



Hình 2.6 – Phương pháp mặt trượt có lớp biên



Hình 2.7- Đồ thị hàm mặt trượt s theo thời gian t khi có hiện tượng *chattering*

Để giải quyết vấn đề này, có nhiều phương pháp tiếp cận khác nhau đã được đề xuất. Ý tưởng của các phương pháp này là thay đổi một vùng lân cận nhỏ của mặt trượt để tránh sự gián đoạn khi chuyển mạch và bảo toàn các tính chất chính của hệ thống. Cụ thể là thay hàm dấu $sign$ bằng một hàm bão hòa. Phương pháp này gọi là phương pháp lớp biên (*“boundary layer method – BLM”*) được đề xuất trong tài liệu [6]:

$$sat\left(\frac{s}{\phi}\right) = \begin{cases} sign(s) & \text{if } |s| > \phi \\ 1 & \text{if } |s| \leq \phi \end{cases} \quad (2.38)$$

với ϕ là độ dày của lớp biên.

Tuy nhiên, phương pháp lớp biên cũng không giải quyết được triệt để hiện tượng chattering cũng như tồn tại nhược điểm. Đó là nếu tăng độ dày lớp biên thì sẽ giảm hiện tượng chattering nhưng cũng đồng thời làm tăng sai số bám quỹ đạo. Ngược lại, nếu giảm độ dày lớp biên thì sẽ giảm sai số bám quỹ đạo nhưng lại tăng hiện tượng chattering.

Thuật toán điều khiển trượt truyền thống cùng với các cải tiến như phương pháp lớp biên, v.v. đã được các nhà khoa học nghiên cứu và ứng dụng trong thực tế. Tuy nhiên, với yêu cầu đòi hỏi độ chính xác ngày càng cao cần phải tiếp tục nghiên cứu đề xuất phương pháp cũng như lý thuyết điều khiển mới. Mục tiêu là giảm hoặc triệt tiêu được hiện tượng chattering và hệ thống có khả năng thích nghi cao trong điều kiện làm việc thực tế.

2.4. PHƯƠNG PHÁP ĐIỀU KHIỂN TRƯỢT BẬC CAO CHO TAY MÁY ROBOT CÔNG NGHIỆP

2.4.1. Phương pháp điều khiển trượt bậc 2 [7]

Xét hệ không dừng có tín hiệu vào u chứa thành phần bất định $d(x, u, t)$:

$$\dot{x} = f(x, u, d, t) \quad (2.39)$$

Xét mặt trượt được mô tả như sau:

$$s(s, t) = (s_1(x, t), s_2(x, t), \dots, s_m(x, t))^T = 0 \quad (2.40)$$

Tín hiệu điều khiển u sẽ là:

$$u = \begin{cases} u_{eq} & \text{khi } s(x, t) = 0 \\ u_{sw} & \text{khi } s(x, t) \neq 0 \end{cases} \quad (2.41)$$

Trong đó:

- u_{eq} là thành phần tín hiệu giữ $x(t)$ ở lại trên mặt trượt, nghĩa là:

Nếu $s(x_0, t_0) = 0$ với $x_0 = x(t_0)$ thì u_{eq} sẽ phải tạo được:

$$\dot{s}(x, t) = 0 \text{ khi } t \geq t_0 \quad (2.42)$$

- u_{sw} là thành phần tín hiệu làm cho $x(t)$ tiến về mặt trượt. Nếu sử dụng hàm xác định dương $V(s) = \frac{1}{2} s^T s$ thì đủ để $x(t)$ tiến về mặt trượt và tín hiệu điều khiển u_{sw} phải tạo ra được điều kiện trượt như sau:

$$\dot{V}(s) = s^T \dot{s} < 0 \text{ khi } s(x) \neq 0 \quad (2.43)$$

Giả sử hệ (2.39) là hệ có cấu trúc affine chứa thành phần bất định $d(x, u, t)$, tương đương:

$$\dot{s} = f(x, t) + H(x, t)[u + d(x, u, t)] \quad (2.44)$$

trong đó $H(x, t) = (h_1(x, t), \dots, h_m(x, t))$ là ma trận $n \times m$

Từ điều kiện (2.43) ta có:

$$\dot{s} = \frac{\partial s}{\partial x} [f(x, t) + H(x, t)[u + d(x, u, t)]] = 0 \quad (2.45)$$

thỏa mãn tính bị chặn: $|d(x, u, t)| \leq \rho(x, t), \forall u$

Giả sử $d(x, u, t) = 0$ thì từ (2.45) ta có:

$$\dot{s} = \frac{\partial s}{\partial x} [f(x, t) + H(x, t)u_{sw}] = 0 \quad (2.46)$$

Nếu ma trận $\frac{\partial s}{\partial x} H(x, t) \in R^{m \times m}$ không suy biến thì:

$$u_{eq} = \left(\frac{\partial s}{\partial x} H(x, t) \right)^{-1} \frac{\partial s}{\partial x} f(x, t) \quad (2.47)$$

Sai lệch giá trị tín hiệu Δ được mô tả là $u_{sw} = u_{eq} + \Delta$. Với Δ được mô tả như sau:

$$\Delta = (\Delta_1, \Delta_2, \dots, \Delta_m)^T \quad (2.48)$$

Với bộ điều khiển relay ta có:

$$\Delta_k = -a_k(x) \text{sign}(s_k(x)) \quad (2.49)$$

Với $k = 1, 2, \dots, m$

Trong đó $a_k(x) > 0, \forall x$ nên:

$$\text{sign}(s) = \begin{cases} 1 & \text{khi } s > 0 \\ -1 & \text{khi } s < 0 \\ 0 & \text{khi } s = 0 \end{cases} \quad (2.50)$$

Từ điều kiện (2.43) và sai lệch tín hiệu u_{sw} ta có:

$$\begin{aligned} \dot{s}(x) &= \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} \dot{x} \\ &= \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} [f(x, t) + H(x, t)u_{sw}] \\ &= \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} [f(x, t) + H(x, t)(u_{eq} + \Delta)] \\ &= \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} [f(x, t) + H(x, t)u_{eq}] + \frac{\partial s}{\partial x} H(x, t)\Delta \end{aligned} \quad (2.51)$$

Với (2.46) ta có $\frac{\partial s}{\partial x} [f(x, t) + H(x, t)u_{eq}] = 0$, nên thay vào (2.51) suy ra:

$$\dot{s}(x) = \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} H(x, t)\Delta \quad (2.52)$$

Giả sử chọn một hàm Lyapunov $V(s)$ xác định dương, ta có:

$$\begin{aligned} \dot{V} &= \frac{\partial V}{\partial s} \frac{\partial s}{\partial x} [f(x, t) + H(x, t)(u_{eq} + \Delta + d)] \\ &= \frac{\partial V}{\partial s} \frac{\partial s}{\partial x} H(x, t)(\Delta + d) < 0 \end{aligned}$$

Với điều kiện Δ thỏa mãn:

$$\min_{\Delta} \max_d \frac{\partial V}{\partial s} \frac{\partial s}{\partial x} H(x, t)[\Delta + d] \leq 0, \forall x \quad (2.53)$$

Từ các ký hiệu của ma trận $H(x, t)$ tại (2.53), sai lệch tín hiệu Δ tại (2.48) và điều kiện xác định của Δ tại (2.49), ta có:

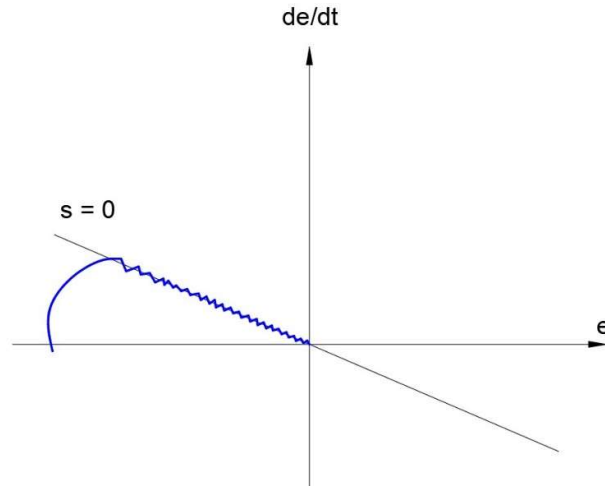
$$\Delta_k = -\rho(x, t) \text{sign} \left(\frac{\partial V}{\partial s} \frac{\partial s}{\partial x} h_k(x, t) \right), k = 1, 2, \dots, m \quad (2.54)$$

Trong thực tế, để chế tạo được thiết bị có thể tái tạo được hàm $\text{sign}(s)$ như đã mô tả tại (2.54) là không khả thi, thay vào đó sẽ là:

$$\text{sign}(x) = \begin{cases} 1 & \text{khi } x > \varepsilon \\ -1 & \text{khi } x < -\varepsilon \\ \text{giữ nguyên giá trị khi } |x| \leq \varepsilon \end{cases} \quad (2.55)$$

nên sẽ không có thành phần u_{eq} trong tín hiệu điều khiển, nghĩa là: $u = u_{sw}$.

Chính điều này gây ra hiện tượng rung (chattering) trong hệ khi mà tín hiệu điều khiển u phải chuyển đổi dấu giá trị với tần số vô cùng lớn để giữ được $x(t)$ trên mặt trượt $s(x,t)=0$.



Hình 2.8 - Điểm trạng thái trượt trên mặt trượt s và tiến về gốc tọa độ, xảy ra hiện tượng chattering

Xét hệ bất định có mô hình trạng thái như sau:

$$\dot{x} = f(x, t) + h(x, t)u \quad (2.56)$$

trong đó $f(x, t)$ và $h(x, t)$ là hai vector hàm bất định.

Điều kiện của điều khiển trượt bậc 2 là:

$$s(x, t) = \dot{s}(x, t) = 0 \quad (2.57)$$

Từ (2.56) và (2.57) ta có:

$$\begin{cases} \dot{x} = f(x, t) + h(x, t)u \\ y = s(x, t) \end{cases} \quad (2.58)$$

Điều khiển trượt bậc 2 tương đương với việc điều khiển hệ (2.56) đạt chất lượng:

$$y = \dot{y} = 0$$

Biến đổi (2.58) ta có:

$$\begin{aligned}\dot{s}(x, t, u) &= \frac{\partial s}{\partial t} + \frac{\partial s}{\partial x} [f(x, t) + h(x, t)u] \\ &= \frac{\partial s}{\partial t} + L_f s(x, t) + L_h s(x, t)u\end{aligned}\quad (2.59)$$

$$\Leftrightarrow \ddot{s}(x, t, u, \dot{u}) = \frac{\partial \dot{s}}{\partial t} + \frac{\partial \dot{s}}{\partial x} [f(x, t)h(x, t)u] + \frac{\partial \dot{s}}{\partial x} h(x, t)\dot{u} \quad (2.60)$$

Tiếp theo, đặt các biến phụ sau đây:

$$\begin{aligned}z_1(t) &= s(x, t) \\ z_2(t) &= \dot{s}(x, t)\end{aligned}\quad (2.62)$$

Ta có:

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = \varphi(x, t, u) + \gamma(x, t)\dot{u} \end{cases} \quad (2.63)$$

Với:

$$\begin{aligned}\varphi(x, t, u) &= \frac{\partial \dot{s}}{\partial t} + \frac{\partial \dot{s}}{\partial x} [f(x, t) + h(x, t)u] \\ \gamma(x, t) &= \frac{\partial \dot{s}}{\partial x} h(x, t)\end{aligned}\quad (2.64)$$

Từ (2.64) chúng ta nhận thấy rằng, u giữ vai trò tham số mô hình, còn $v = \dot{u}$ mới chính là tín hiệu điều khiển của thuật toán điều khiển trượt bậc 2.

Với các phương pháp điều khiển trượt bậc hai, với việc khái quát hơn phương pháp điều khiển trượt truyền thống bằng việc tác động trực tiếp lên các đạo hàm thời gian bậc cao hơn của mặt trượt thay vì thay đổi đạo hàm thời gian của mặt trượt tiêu chuẩn. Với phương pháp điều khiển trượt bậc hai, trong khi vẫn giữ được những ưu điểm của phương pháp điều khiển trượt truyền thống đồng thời giảm thiểu và tiến đến triệt tiêu hiện tượng chattering. Từ đó hiệu quả làm việc của hệ thống được chính xác hơn.

*** Phương pháp điều khiển trượt bậc 2 siêu xoắn:**

Luật điều khiển trượt bậc 2 tổng thể có thể viết lại như sau:

$$\begin{aligned}u(t) &= u_{ed} + u_{aux}(t) \\ &= -B_0(q)(f_0(q, \dot{q}) - \ddot{q}_d + \lambda \dot{e}(t) + \beta r(0) \exp(-\beta t) + ks(t)G^T + \\ &\quad \hat{\Gamma} sgn(s(t)G^T))\end{aligned}\quad (2.65)$$

Giả sử rằng nhiễu loạn hệ thống là không xác định nhưng bị giới hạn, tức là $\Gamma > \|D(t)\|$, trong đó Γ là hằng số dương chưa biết. Giả sử rằng $\hat{\Gamma}$ là ước lượng của Γ . Các quỹ đạo của hệ thống được hội tụ đến bề mặt trượt $s = 0$ trong thời gian hữu hạn và sau đó vẫn còn trên đó.

Phương trình đạo hàm mặt trượt:

$$\dot{s}(t) = G(-ks(t)G^T - \hat{\Gamma} \operatorname{sgn}(s(t)G^T) + D(t)) \quad (2.66)$$

Xét hàm Lyapunov sau:

$$V(t) = \frac{1}{2}s(t)^2 + \frac{1}{2}\gamma\tilde{\Gamma}^2 \quad (2.67)$$

Trong đó: $\tilde{\Gamma} = \hat{\Gamma} - \Gamma$.

Lấy đạo hàm $V(t)$ theo thời gian ta có:

$$\begin{aligned} \dot{V}(t) &= s(t)\dot{s}(t) + \gamma\tilde{\Gamma}\dot{\tilde{\Gamma}} \\ &= s(t)G \left(D(t) - ks(t)G^T - \hat{\Gamma} \operatorname{sgn}(s(t)G^T) \right) + \gamma(\hat{\Gamma} - \Gamma)\dot{\hat{\Gamma}} \\ &= s(t)G \left(D(t) - ks(t)G^T - \hat{\Gamma} \operatorname{sgn}(s(t)G^T) \right) + \gamma\kappa(\hat{\Gamma} - \Gamma)\|s(t)G^T\| \\ &\leq \|s(t)G\|\|D(t)\| - \hat{\Gamma}\|s(t)G\| + \gamma\kappa\|s(t)G\|(\hat{\Gamma} - \Gamma) + \|s(t)G\|\Gamma - \\ &\quad \|s(t)G\|\Gamma \\ &= -\|s(t)G\|(\Gamma - \|D(t)\|) - (1 - \gamma\kappa)\|s(t)G\|(\hat{\Gamma} - \Gamma) \end{aligned} \quad (2.68)$$

Vì $\Gamma > \|D(t)\|$ và $\gamma\kappa < 1$, do đó:

$$\begin{aligned} \dot{V}(t) &\leq -\sqrt{2}\|G\|(\Gamma - \|D(t)\|)\frac{|s(t)|}{\sqrt{2}} - \sqrt{\frac{2}{\gamma}}(1 - \gamma\kappa)\frac{\tilde{\Gamma}}{\sqrt{\frac{2}{\gamma}}}\|s(t)G\| \\ &\leq -\min \left\{ \sqrt{2}\|G\|(\Gamma - \|D(t)\|), \sqrt{\frac{2}{\gamma}}(1 - \gamma\kappa)\|s(t)G\| \right\} \times \left(\frac{|s(t)|}{\sqrt{2}} + \frac{\tilde{\Gamma}}{\sqrt{\frac{2}{\gamma}}} \right) \\ &= -\Omega V(t)^{\frac{1}{2}} \end{aligned} \quad (2.69)$$

$$\text{Với } \Omega = \min \left\{ \sqrt{2}\|G\|(\Gamma - \|D(t)\|), \sqrt{\frac{2}{\gamma}}(1 - \gamma\kappa)\|s(t)^T G\| \right\} > 0$$

Cuối cùng, bằng cách sử dụng bộ điều khiển điều chỉnh thích nghi, sự hội tụ thời gian hữu hạn đối với bề mặt trượt phi tuyến toàn cục $s(t) = 0$ được đảm bảo.

Để tránh hiện tượng rung động chattering, luật điều khiển có thể được biến đổi như sau để thay thế hàm dấu $sign$:

$$u(t) = -B_0(q) \left(f_0(q, \dot{q}) - \ddot{q}_d + \lambda \dot{e}(t) + \beta r(0) \exp(-\beta t) + ks(t)G^T + \hat{\Gamma} sat\left(\frac{s(t)G^T}{\delta}\right) \right) \quad (2.70)$$

Trong đó $sat(\cdot)$ là hàm bão hòa, hằng số δ là độ dày lớp biên.

Do sử dụng hàm bão hòa trong công thức (2.70), mặt trượt $s(t)$ sẽ không hoàn toàn bằng 0 trong quá trình robot hoạt động bám quỹ đạo. Do đó, các thông số thích nghi sẽ thay đổi chậm và tăng vô hạn. Để giải quyết vấn đề này, luật thích nghi có thể được sửa đổi thành:

$$\dot{\hat{\Gamma}} = \begin{cases} 0 & \text{if } \|s(t)G^T\| \leq \delta \\ \kappa \|s(t)G^T\| & \text{if } \|s(t)G^T\| > \delta \end{cases} \quad (2.71)$$

Phương pháp phổ biến để giảm hiện tượng chattering là thay thế hàm dấu không liên tục bằng hàm bão hòa liên tục. Tuy nhiên phương pháp này cũng bộc lộ nhiều nhược điểm. Nếu tăng độ lớn của lớp biên để giảm chattering thì sẽ làm giảm độ chính xác, và ngược lại giảm độ lớn của lớp biên thì sẽ tăng được độ chính xác nhưng lại làm tăng chattering.

Để giải quyết vấn đề này trong điều khiển trượt bậc cao, thuật toán siêu xoắn (super twisting) là một giải pháp phù hợp. Thuật toán điều khiển trượt bậc 2 siêu xoắn có thể được biểu diễn như sau:

$$\begin{aligned} u(t) &= -B_0(q)(f_0(q, \dot{q}) + \lambda \dot{e}(t) + \beta r(0) \exp(-\beta t) + ks(t)G^T + \\ &\quad + \mu_0 |s(t)G^T|^{0.5} sgn(s(t)G^T + v_a) \\ \dot{v}_a &= \hat{\Gamma} sgn(s(t)G^T) \end{aligned} \quad (2.72)$$

Trong đó μ_0 là hằng số dương.

2.4.2. Phương pháp điều khiển trượt bậc 3

Xét lại bài toán điều khiển trượt chống hiện tượng chattering cho hệ affine bất định được mô tả tại (2.56) với mặt trượt $s(x, t)$. Yêu cầu đặt ra là mọi quỹ đạo của điểm trạng thái $x(t)$ phải tiến tiệm cận về được mặt trượt:

$$s(x, t) = \dot{s}(x, t) = \dots = s^{(r-1)}(x, t) = 0 \quad (2.73)$$

Nếu $r = 3$, từ (2.73) ta có:

$$\begin{cases} \dot{x} = f(x, t) + h(x, t)u \\ y = s(x, t) \end{cases} \quad (2.74)$$

có bậc tương đối là 3, nghĩa là:

$$L_h L_f^k s(x, t) = \begin{cases} 0 & \text{khi } 0 \leq k \leq 1 \\ \neq 0 & \text{khi } k = 2 \end{cases} \quad (2.75)$$

Sử dụng phép biến đổi vi phân ta có:

$$z_{i+1} = L_f^i s(x, t), \quad 0 \leq i \leq 2 \quad (2.76)$$

Hệ (2.74) sẽ trở thành hệ truyền ngược chặt (được mô tả trong tài liệu [8, 9]):

$$\begin{cases} \dot{z}_i = z_{i+1} & \text{khi } 1 \leq i \leq n-1 \\ \dot{z}_r = \varphi(z, t) + \gamma(z, t)u \end{cases} \quad (2.77)$$

trong đó $\varphi(\cdot), \gamma(\cdot)$ được xác định từ các hàm gốc ban đầu $f(\cdot), h(\cdot)$ của (2.75) như sau:

$$\varphi(x, t) = L_f^3 s(x, t) \quad \text{và} \quad \gamma(x, t) = L_h L_f^2 s(x, t) \quad (2.69)$$

Nếu $f(\cdot), h(\cdot)$ là các hàm bất định thì (2.69) cũng là các hàm bất định.

Như vậy, điều khiển trượt bậc 3 (2.65) và (2.66) là tương đương với điều khiển hệ (2.56) ổn định.

Vấn đề chính khi áp dụng phương pháp điều khiển trượt bậc 3 là yêu cầu cần có các thông số của mặt trượt s phải liên hệ với phương pháp đang áp dụng. Nghĩa là nếu áp dụng phương pháp điều khiển trượt bậc 3 thì phải cần các dữ liệu của mặt trượt s bậc 2. Việc có quá nhiều dữ liệu cần tính toán và chỉnh định khi áp

dụng cho các phương pháp điều khiển trượt bậc 3 khiến việc triển khai trong thực tế gặp khó khăn. Chính vì lý do này, phương pháp điều khiển trượt bậc 2 thường được sử dụng rộng rãi nhất.

CHƯƠNG 3

NGHIÊN CỨU ĐỀ XUẤT THUẬT TOÁN ĐIỀU KHIỂN TRƯỢT BẬC CAO THÍCH NGHI ỨNG DỤNG TRÍ TUỆ NHÂN TẠO

3.1. GIỚI THIỆU

Điều khiển trượt (Sliding Mode Control - SMC) là một trong những phương pháp điều khiển phi tuyến hiệu quả được sử dụng để điều khiển các hệ thống phi tuyến có sự tồn tại của các thành phần bất định, đặc biệt là tay máy robot [10]. Tuy nhiên, nhược điểm của nó là chỉ đảm bảo sự ổn định tiệm cận trong khi có nhiễu dao động tần số cao trong tín hiệu điều khiển đầu vào. Để đảm bảo sự hội tụ trong thời gian hữu hạn của các quỹ đạo trạng thái hệ thống, rất nhiều công trình nghiên cứu đã được thực hiện và đề xuất các giải pháp, trong đó có những công trình nghiên cứu về điều khiển trượt bậc cao (HSMC). Trong những nghiên cứu [11]–[13], phương pháp điều khiển trượt đầu cuối (Terminal SMC) đã được giới thiệu để đảm bảo độ chính xác bám quỹ đạo của các trạng thái hệ thống với sự ổn định trong thời gian hữu hạn. Với phương pháp TSMC, hiệu suất hội tụ được cải thiện đáng kể. Tuy nhiên, vấn đề kỳ dị có thể xuất hiện với một số thuật toán TSMC kinh điển. Có một số thuật toán TSMC được đề xuất dành cho các tay máy robot trong đó vấn đề kỳ dị đã được giải quyết hoàn toàn như thuật toán TSMC không kỳ dị (NTSMC) [14]. Bên cạnh đó, các thuật toán điều khiển trượt đầu cuối thời gian hữu hạn (FnTTSMC) đã được đề xuất [15], [16] để có thể điều khiển các quỹ đạo trạng thái của hệ thống hội tụ đến điểm cân bằng trong một khoảng thời gian hữu hạn. Để tăng tốc độ ổn định và hiệu suất động học của TSMC và FnTTSMC, các thuật toán điều khiển TSMC tác động nhanh (FTSMC) [17]–[19] và thuật toán điều khiển TSMC tác động nhanh toàn cục (GFTSMC) [20], [21] đã được phát triển. Ý tưởng chủ đạo của GFTSMC là thêm một thành phần tuyến tính để nhanh chóng buộc các quỹ đạo trạng thái hệ thống hội tụ đến mặt trượt trong thời gian ngắn. Thuật toán GFTSMC đã được sử dụng rộng rãi trong các ứng dụng thực tế như điều khiển các tay máy robot [22], những tay máy robot được truyền động bằng dây đai kéo [23], hệ thống kính hiển vi [24], các hệ

thống phi tuyến [20], v.v. Tuy nhiên, FTSMC hoặc GFTSMC là các thuật toán điều khiển dựa trên một mô hình động lực học của hệ thống, và không dễ dàng để có được một mô hình toán học chính xác do sự hiện diện của nhiều thành phần bất định của hệ thống. Hơn nữa, không thể loại bỏ hoàn toàn hiện tượng dao động trong tín hiệu điều khiển của thuật toán FTSMC hoặc GFTSMC.

Có một phương pháp hiệu quả có thể giải quyết cùng lúc hai tồn tại nêu trên đó là sử dụng một bộ quan sát tích hợp vào thuật toán điều khiển để tính gần đúng tổng các thành phần bất định ảnh hưởng đến hệ thống. Bằng cách này, những dao động tần số cao không chỉ được giảm đáng kể mà còn tăng tính bền vững của hệ thống điều khiển. Có nhiều bộ quan sát khác nhau đã được đề xuất để tính gần đúng các thành phần bất định như bộ quan sát trượt (SMO) [25], bộ quan sát dùng logic mờ (FLO) [26], bộ quan sát trượt bậc cao (HOSMO) [27], [28], bộ quan sát dùng mạng nơ-ron (NNO) [29], bộ quan sát khuếch đại (HGO) [30], bộ quan sát mở rộng (ETO) [31], v.v. Xét về tính bền vững, thiết kế đơn giản và khả năng ứng dụng thực tế, phương pháp sử dụng các bộ quan sát trượt bậc cao (HOSMO) không chỉ cung cấp khả năng chống lại các yếu tố bất định và nhiễu động bên ngoài mà còn giảm thiểu dao động tần số cao. Tuy nhiên các bộ HOSMO thông thường không đảm bảo sự ổn định và hội tụ của hệ thống trong thời gian xác định.

Dựa trên những phân tích nêu trên, trong đề tài nghiên cứu này trước hết sẽ đề xuất một bộ quan sát trượt bậc cao thời gian cố định (FxTHOSMO) mới để ước tính toàn bộ các thành phần bất định mà không cần biết trước các thông tin về chúng. Một thuật toán điều khiển trượt đầu cuối thời gian hữu hạn (FnTTSMC) sau đó được thiết kế dựa trên phương trình mặt trượt đầu cuối nhanh toàn cục thời gian hữu hạn và luật điều khiển tiếp cận nhanh để thiết lập sự hội tụ trong thời gian hữu hạn toàn cục của hệ thống. Hệ thống điều khiển tích hợp có các đặc tính ưu việt như độ chính xác bám quỹ đạo cao, hội tụ trong thời gian hữu hạn và bền vững với những ảnh hưởng của các thành phần bất định. Tính ổn định trong thời gian hữu hạn và sự hội tụ của hệ thống điều khiển được chứng minh bằng cách sử dụng lý thuyết Lyapunov.

Sau đó, để bù thích nghi các thành phần bất định chưa thể ước lượng hết bằng bộ quan sát, đề tài sử dụng một mạng nơ-ron cùng với thuật toán học online để tăng thêm độ chính xác và tính bền vững của hệ thống.

3.2. ĐỀ XUẤT THUẬT TOÁN ĐIỀU KHIỂN TRƯỢT BẬC CAO KẾT HỢP BỘ QUAN SÁT TRƯỢT BẬC CAO THỜI GIAN CỐ ĐỊNH

3.2.1. Các bổ đề cần thiết

Xem xét hệ thống sau đây:

$$\dot{x}(t) = f(t, x), x(0) = x_0 \quad (3.1)$$

trong đó $x \in R^n$, $f(x): \Delta$ là một hàm phi tuyến trên vùng lân cận mở $\Delta \subseteq R^n$ của điểm gốc, và $f(0) = 0$. Điểm gốc được giả định là điểm cân bằng của hệ (3.1).

Định nghĩa 1: Điểm gốc của hệ thống (3.1) được gọi là điểm ổn định thời gian hữu hạn toàn cục nếu nó ổn định tiệm cận toàn cục với hàm thời gian giới hạn $T(x_0)$, tức là có tồn tại $T_{\max} > 0$ sao cho $T(x_0)$ thỏa mãn điều kiện $T(x_0) < T_{\max}$.

Bổ đề 1 [20]: Xem xét phương trình vi phân sau đây:

$$\dot{q} = -\frac{2\alpha_0}{1+e^{-\eta_0(|q|-\varepsilon_0)}} q - \frac{2\beta_0}{1+e^{\mu_0(|q|-\varepsilon_0)}} \text{sig}(q)^{\omega_0} \quad (3.2)$$

trong đó $\alpha_0, \beta_0, \eta_0, \mu_0$ là những hằng số thiết kế có giá trị dương, $0 < \omega_0 < 1$, $\varepsilon_0 = (\beta_0/\alpha_0)^{1/(1-\omega_0)}$, và $\text{sig}(q)^{\omega_0} = |q|^{\omega_0} \text{sgn}(q)$. Do đó, hệ thống (2) được thừa nhận là ổn định thời gian hữu hạn liên quan đến giá trị ban đầu $q(0)$ và thời gian hội tụ T_0 được mang lại như sau:

$$T_0 < \frac{\ln(|q(0)|) - \ln(\varepsilon_0)}{\alpha_0} + \frac{1}{\beta_0(1-\delta_0)} |\varepsilon_0|^{1-\omega_0} \quad (3.3)$$

Bổ đề 2 [32]: Xem xét phương trình vi phân sau đây:

$$\dot{q} = -\alpha_0 \text{sig}(q)^{\eta_0} - \beta_0 \text{sig}(q)^{\mu_0} \quad (3.4)$$

trong đó α_0, β_0 là các hằng số dương được chọn, $\text{sig}(q)^{\gamma_i} = |q|^{\gamma_i} \text{sgn}(q)$, $\varphi_0 > 1$, $0 < \lambda_0 < 1$. $\eta_0 = 0.5(\varphi_0 + 1) + 0.5(\varphi_0 - 1)\text{sgn}(|q| - 1)$, and $\mu_0 = 0.5(\varphi_0 + \lambda_0) + 0.5(\varphi_0 - \lambda_0)\text{sgn}(|q| - 1)$. Khi đó, hệ thống (3.4) được thừa nhận là ổn định thời gian hữu liên quan đến giá trị ban đầu $q(0)$ và thời gian hội tụ T_0 được mang lại như sau:

$$T_0 < \frac{1}{(\alpha_0 + \beta_0)(\varphi_0 - 1)} + \frac{1}{\beta_0(1 - \lambda_0)} \ln \left(1 + \frac{\alpha_0}{\beta_0} \right) \quad (3.5)$$

3.2.2. Biểu diễn mô hình động lực học của tay máy robot dạng không gian trạng thái

Xem xét mô hình động lực học của tay máy robot được thể hiện trong không gian khớp (Joint space) như sau:

$$M(p)\ddot{p} + C(p, \dot{p})\dot{p} + G(p) + F_r(\dot{p}) = \tau - \tau_d(t) \quad (3.6)$$

trong đó $p \in R^{3 \times 1}$, $\dot{p} \in R^{3 \times 1}$, và $\ddot{p} \in R^{3 \times 1}$ tương ứng với các vector vị trí, vận tốc và gia tốc trong không gian khớp nối. $M(p) \in R^{n \times n}$, $C(p, \dot{p}) \in R^{n \times n}$, and $G(p) \in R^{n \times 1}$ tương ứng với ma trận khối lượng, Coriolis và lực ly tâm, và lực hấp dẫn. $\tau \in R^{n \times 1}$, $F_r(\dot{p}) \in R^{n \times 1}$, và $\tau_d(t) \in R^{n \times 1}$ tương ứng với các vector của mô-men điều khiển, lực ma sát và nhiễu loạn.

Trên thực tế, để đạt được một mô hình động lực học chính xác của tay máy robot không phải là điều dễ dàng. Do đó, trong bài báo này giả định rằng:

$$\begin{cases} M(p) = \hat{M}(p) + dM(p) \\ C(p, \dot{p}) = \hat{C}(p, \dot{p}) + dC(p, \dot{p}) \\ G(p) = \hat{G}(p) + dG(p) \end{cases} \quad (3.7)$$

trong đó $\hat{M}(p) \in R^{n \times n}$, $\hat{C}(p, \dot{p}) \in R^{n \times n}$, and $\hat{G}(p) \in R^{n \times 1}$ tương ứng với các giá trị ước tính của các giá trị thực của $M(p)$, $C(p, \dot{p})$, và $G(p)$. $dM(p) \in R^{n \times 3}$, $dC(p, \dot{p}) \in R^{n \times n}$, và $dG(p) \in R^{n \times 1}$ là những thành phần động lực học bất định.

Chúng ta đặt $x = [x_1^T, x_2^T]^T = [p^T, \dot{p}^T]^T$ và $u = \tau$; khi đó, mô hình động lực học của tay máy robot được mô tả dưới dạng mô hình không gian trạng thái như sau:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = a(x)u + b(x) + \Delta(x, d, t), \end{cases} \quad (3.8)$$

trong đó $b(x) = -\hat{M}^{-1}(p)(\hat{C}(p, \dot{p})\dot{p} + \hat{G}(p))$ đại diện cho phần động lực học tính toán được; $a(x) = \hat{M}^{-1}(p)$ đại diện cho ma trận nghịch đảo của ma trận khối lượng; và $\Delta(x, d, t) = -\hat{M}^{-1}(p) \begin{pmatrix} F_r(\dot{p}) + dM(p)\ddot{p} \\ +dC(p, \dot{p})\dot{p} + dG(p) + \tau_d(t) \end{pmatrix}$ đại diện cho toàn bộ những thành phần bất định.

Giả định 1: Toàn bộ những thành phần bất định là hữu hạn và bị chặn trên bởi:

$$|\Delta(x, \Sigma, t)| < \bar{\Delta}, \quad (3.9)$$

trong đó $\bar{\Delta}$ là một hằng số dương.

Giả định 2: Đạo hàm bậc nhất của toàn bộ những thành phần bất định cũng là hữu hạn và bị chặn trên bởi:

$$|\dot{\Delta}(x, \Sigma, t)| \leq \Delta^*, \quad (3.10)$$

trong đó Δ^* là một hằng số dương.

3.2.3. Thiết kế bộ quan sát trượt bậc cao thời gian cố định

Bộ quan sát được thiết kế để xấp xỉ những thành phần bất định được mô tả như sau:

$$\begin{cases} \tilde{v} = v - x_2 \\ \dot{\tilde{v}} = a(x)u_d + b_n(x) + \hat{\Delta} \\ - \frac{2\alpha_3}{1 + e^{-\eta_3(|\tilde{v}| - \varepsilon_3)}} \tilde{v} - \frac{2\beta_3}{1 + e^{\mu_3(|\tilde{v}| - \varepsilon_3)}} \text{sig}(\tilde{v})^{\omega_3} \end{cases} \quad (3.11)$$

trong đó ν là giá trị xấp xỉ của x_2 ; $\alpha_3, \beta_3, \eta_3, \mu_3$ là những hằng số dương; $0 < \omega_3 < 1$, và $\varepsilon_3 = (\beta_3/\alpha_3)^{1/(1-\omega_3)}$; $\hat{\Delta}$ là giá trị xấp xỉ của những thành phần bất định.

Luật cập nhật của $\hat{\Delta}$ được thiết kế như sau:

$$\begin{cases} \dot{\varpi} = \dot{\tilde{\nu}} + \frac{2\alpha_3}{1+e^{-\eta_3(|\tilde{\nu}|-\varepsilon_3)}} \tilde{\nu} + \frac{2\beta_3}{1+e^{\mu_3(|\tilde{\nu}|-\varepsilon_3)}} \text{sig}(\tilde{\nu})^{\omega_3} \\ \dot{\hat{\Delta}} = -\frac{2\alpha_4}{1+e^{-\eta_4(|\varpi|-\varepsilon_4)}} \varpi - \frac{2\beta_4}{1+e^{\mu_4(|\varpi|-\varepsilon_4)}} \text{sig}(\varpi)^{\omega_4} \\ -\Delta^*(t) \text{sgn}(\varpi) \end{cases} \quad (3.12)$$

trong đó $\alpha_4, \beta_4, \eta_4, \mu_4$ là những hằng số dương; $0 < \omega_4 < 1$, $\varepsilon_4 = (\beta_4/\alpha_4)^{1/(1-\omega_4)}$; và $\Delta^*(t)$ là một giá trị thích nghi.

Chứng minh tính ổn định:

Tính đạo hàm bậc 1 của $\tilde{\nu}$ dựa vào những phương trình (3.11) – (3.12), chúng ta có:

$$\begin{aligned} \dot{\tilde{\nu}} &= \dot{\nu} - \dot{x}_2 \\ &= \hat{\Delta} - \Delta - \frac{2\alpha_3}{1+e^{-\eta_3(|\tilde{\nu}|-\varepsilon_3)}} \tilde{\nu} - \frac{2\beta_3}{1+e^{\mu_3(|\tilde{\nu}|-\varepsilon_3)}} \text{sig}(\tilde{\nu})^{\omega_3} \end{aligned} \quad (3.13)$$

Thay phương trình (3.13) vào trong phương trình (3.12), sau đó tính đạo hàm theo thời gian của kết quả đạt được, chúng ta có:

$$\begin{cases} \dot{\varpi} = \hat{\Delta} - \Delta \\ \dot{\varpi} = -\dot{\hat{\Delta}} - \Delta^*(t) \text{sgn}(\varpi) \\ -\frac{2\alpha_4}{1+e^{-\eta_4(|\varpi|-\varepsilon_4)}} \varpi - \frac{2\beta_4}{1+e^{\mu_4(|\varpi|-\varepsilon_4)}} \text{sig}(\varpi)^{\omega_4} \end{cases} \quad (3.14)$$

Lựa chọn hàm Lyapunov $V_1 = 0.5\varpi^2$ và tính toán đạo hàm theo thời gian của nó dựa vào kết quả trong phương trình (3.14), chúng ta đạt được:

$$\begin{aligned}
\dot{V}_1 &= \varpi \dot{\varpi} \\
&= \varpi \left(-\dot{\Delta} - \frac{2\alpha_4}{1+e^{-\eta_4(|\varpi|-\varepsilon_4)}} \varpi - \frac{2\beta_4}{1+e^{\mu_4(|\varpi|-\varepsilon_4)}} |\varpi|^{\omega_4} \operatorname{sgn}(\varpi) \right. \\
&\quad \left. - \Delta^*(t) \operatorname{sgn}(\varpi) \right) \\
&= -\dot{\Delta} \varpi - \Delta^*(t) |\varpi| - \frac{2\alpha_4}{1+e^{-\eta_4(|\varpi|-\varepsilon_4)}} \varpi^2 - \frac{2\beta_4}{1+e^{\mu_4(|\varpi|-\varepsilon_4)}} |\varpi|^{\omega_4+1} \\
&\leq -\left(\Delta^*(t) - |\dot{\Delta}| \right) |\varpi| - \frac{2\alpha_4}{1+e^{-\eta_4(|\varpi|-\varepsilon_4)}} \varpi^2 - \frac{2\beta_4}{1+e^{\mu_4(|\varpi|-\varepsilon_4)}} |\varpi|^{\omega_4+1} \\
&\leq -\frac{2\alpha_4}{1+e^{-\eta_4(|\varpi|-\varepsilon_4)}} \varpi^2 - \frac{2\beta_4}{1+e^{\mu_4(|\varpi|-\varepsilon_4)}} |\varpi|^{\omega_4+1} \\
&\leq 0
\end{aligned} \tag{3.15}$$

Như vậy chúng ta thấy điều kiện $V_1 > 0$ và $\dot{V}_1 \leq 0$ được đảm bảo. Kết quả theo lý thuyết ổn định Lyapunov, mặt trượt ϖ đã chọn cho bộ quan sát sẽ hội tụ về 0 trong khoảng thời gian hữu hạn, nghĩa là $\varpi = 0$.

Chúng ta xác định $\tilde{\Delta} = \hat{\Delta} - \Delta$ như là sai số xấp xỉ của bộ quan sát. Từ phương trình (3.14), chúng ta có:

$$\tilde{\Delta} = \hat{\Delta} - \Delta = \varpi \tag{3.16}$$

Như vậy chúng ta thấy, do $\varpi = 0$ nên $\tilde{\Delta} = \varpi = 0$. Như vậy đã chứng minh được bộ quan sát mà bài báo đề xuất có thể xấp xỉ những thành phần bất định trong khoảng thời gian hữu hạn.

Giá trị $\Delta^*(t)$ trong bài báo này có thể xác định được bằng cách áp dụng kỹ thuật thích nghi hai lớp trong các công bố [33], [34].

3.2.4. Thiết kế thuật toán điều khiển trượt bậc cao

a) Thiết kế mặt trượt với sự hội tụ thời gian hữu hạn

Đặt $x_r = [x_{1r}, \dots, x_{nr}]^T \in R^n$ là vector quỹ đạo mong muốn. Khi đó, $x_e = x_1 - x_r$ là sai số điều khiển vận tốc và $\dot{x}_e = \dot{x}_1 - \dot{x}_r$ là sai số điều khiển vị trí.

Chúng ta thiết kế một mặt trượt với sự hội tụ trong thời gian hữu hạn dựa vào

bỏ đề 1 như sau:

$$\dot{s} = \dot{x}_e + \frac{2\alpha_1}{1 + e^{-\eta_1(|x_e| - \varepsilon_1)}} x_e + \frac{2\beta_1}{1 + e^{\mu_1(|x_e| - \varepsilon_1)}} \text{sig}(x_e)^{\omega_1} \quad (3.17)$$

trong đó $s \in R^{n \times 1}$ là hàm trượt; $\alpha_1, \beta_1, \eta_1, \mu_1$ là những hằng số dương; $0 < \omega_1 < 1$,

$\varepsilon_1 = (\beta_1/\alpha_1)^{1/(1-\omega_1)}$, và $\text{sig}(x_e)^{\omega_1} = |x_e|^{\omega_1} \text{sgn}(x_e)$.

Một khi các của trạng thái hệ thống được vận hành trong chế độ trượt, chúng phải thỏa mãn điều kiện $s = 0$ [35]. Do đó, chúng ta xem xét phương trình (3.17) trong trường hợp $s = 0$ như sau:

$$\dot{x}_e = -\frac{2\alpha_1}{1 + e^{-\eta_1(|x_e| - \varepsilon_1)}} x_e - \frac{2\beta_1}{1 + e^{\mu_1(|x_e| - \varepsilon_1)}} \text{sig}(x_e)^{\omega_1} \quad (3.18)$$

Thời gian giới hạn để thực hiện chuyển động trên mặt trượt được xác định như sau:

$$\begin{aligned} T_s &< \int_{\varepsilon_1}^{x_e(0)} \frac{1}{\alpha_1 |x_e|} d(|x_e|) + \int_0^{\varepsilon_1} \frac{1}{\beta_1 |x_e|^{\delta_1}} d(|x_e|) \\ &< \frac{\ln(|x_e(0)|) - \ln(\varepsilon_1)}{\alpha_1} + \frac{1}{\beta_1 (1 - \delta_1)} |\varepsilon_1|^{1-\omega_1} \end{aligned} \quad (3.19)$$

b) Đề xuất phương trình thuật toán điều khiển trượt bậc cao

Để xác định được phương trình của tín hiệu điều khiển đầu vào, đầu tiên chúng ta đạo hàm phương trình (3.17) theo thời gian như sau:

$$\begin{aligned} \dot{s} = \ddot{x}_e + \frac{2\alpha_1}{1 + e^{-\eta_1(|x_e| - \varepsilon_1)}} \dot{x}_e + \frac{2\alpha_1 \eta_1 \dot{x}_e \text{sgn}(x_e) e^{-\eta_1(|x_e| - \varepsilon_1)}}{\left(1 + e^{-\eta_1(|x_e| - \varepsilon_1)}\right)^2} x_e \\ + \frac{2\beta_1 \omega_1}{1 + e^{\mu_1(|x_e| - \varepsilon_1)}} |x_e|^{\omega_1 - 1} \dot{x}_e - \frac{2\beta_1 \mu_1 \dot{x}_e e^{\mu_1(|x_e| - \varepsilon_1)}}{\left(1 + e^{\mu_1(|x_e| - \varepsilon_1)}\right)^2} |x_e|^{\omega_1} \end{aligned} \quad (3.20)$$

trong đó $\ddot{x}_e = \ddot{x}_2 - \ddot{x}_r$. Sau đó thay mô hình (3.8) vào phương trình (3.20) chúng ta có được:

$$\dot{s} = a(x)u_d + b_n(x) + \Delta(x, \Sigma, t) - \ddot{x}_r + \Pi \quad (3.21)$$

trong đó:

$$\begin{aligned} \Pi = & \frac{2\alpha_1}{1 + e^{-\eta_1(|x_e| - \varepsilon_1)}} \dot{x}_e + \frac{2\alpha_1\eta_1\dot{x}_e \operatorname{sgn}(x_e) e^{-\eta_1(|x_e| - \varepsilon_1)}}{\left(1 + e^{-\eta_1(|x_e| - \varepsilon_1)}\right)^2} x_e \\ & + \frac{2\beta_1\omega_1}{1 + e^{\mu_1(|x_e| - \varepsilon_1)}} |x_e|^{\omega_1 - 1} \dot{x}_e - \frac{2\beta_1\mu_1\dot{x}_e e^{\mu_1(|x_e| - \varepsilon_1)}}{\left(1 + e^{\mu_1(|x_e| - \varepsilon_1)}\right)^2} |x_e|^{\omega_1}. \end{aligned} \quad (3.22)$$

Phương trình thuật toán điều khiển trượt được thiết kế như sau:

$$u_d = -a^{-1}(x)(u_{eq} + u_r) \quad (3.23)$$

trong đó u_{eq} được xây dựng dựa trên bộ quan sát (3.13) và phương trình (3.21):

$$u_{eq} = b_n(x) + \hat{\Delta} - \ddot{x}_r + \Pi \quad (3.24)$$

và u_r là luật tiếp cận nhanh được đề xuất bởi phương trình:

$$u_r = \alpha_2 \operatorname{sig}(s)^{\eta_2} + \beta_2 \operatorname{sig}(s)^{\mu_2}. \quad (3.25)$$

với α_2 , β_2 là những hằng số dương; và: $\operatorname{sig}(s)^{\eta_i} = |s|^{\eta_i} \operatorname{sgn}(s)$, $\varphi_2 > 1$, $0 < \lambda_2 < 1$, $\eta_2 = 0.5(\varphi_2 + 1) + 0.5(\varphi_2 - 1)\operatorname{sgn}(|s| - 1)$; và $\mu_2 = 0.5(\varphi_2 + \lambda_2) + 0.5(\varphi_2 - \lambda_2)\operatorname{sgn}(|s| - 1)$.

c) Chứng minh sự ổn định

Thay các phương trình của thuật toán điều khiển (3.23), (3.24) và (3.25) vào phương trình (3.21) và sau một số biến đổi rút gọn chúng ta có được:

$$\dot{s} = -u_r - \tilde{\Delta} \quad (3.26)$$

Hàm Lyapunov $V_2 = s^2$ được chọn để phân tích sự ổn định của phương pháp điều khiển đề xuất. Đạo hàm theo thời gian của V_2 được tính toán như sau:

$$\begin{aligned}
\dot{V}_2 &= 2s\dot{s} \\
&= 2s(-u_r - \tilde{\Delta}) \\
&= 2s(-\alpha_2 |s|^{\eta_2} \operatorname{sgn}(s) - \beta_2 |s|^{\mu_2} \operatorname{sgn}(s) - \tilde{\Delta}) \\
&= -2\alpha_2 |s|^{\eta_2+1} - 2\beta_2 |s|^{\mu_2+1} - 2\tilde{\Delta}s
\end{aligned} \tag{3.27}$$

Như đã phân tích trong phương trình (3.16), chúng ta có $\tilde{\Delta} = 0$. Do đó:

$$\begin{aligned}
\dot{V}_2 &= -2\alpha_2 |s|^{\eta_2+1} - 2\beta_2 |s|^{\mu_2+1} \\
&= -2\alpha_2 V_2^{\frac{\eta_2+1}{2}} - 2\beta_2 V_2^{\frac{\mu_2+1}{2}}
\end{aligned} \tag{3.28}$$

Dựa vào bổ đề 2, mặt trượt đề xuất sẽ hội tụ đến điểm 0 trong khoảng thời gian hữu hạn T_r ; và thời gian hội tụ T_r bị chặn bởi:

$$T_r < \frac{1}{(\alpha_2 + \beta_2)(\varphi_2 - 1)} + \frac{1}{\beta_2(1 - \lambda_2)} \ln \left(1 + \frac{\alpha_2}{\beta_2} \right) \tag{3.29}$$

Do đó, tổng thời gian hội tụ của những quỹ đạo trạng thái trong hệ thống là hữu hạn và được tính toán như sau:

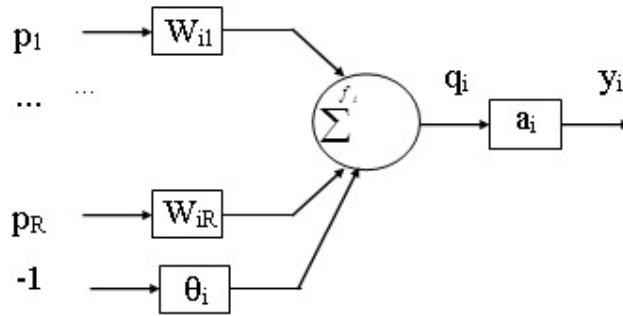
$$\begin{aligned}
T &= T_r + T_s \\
&< \frac{1}{(\alpha_2 + \beta_2)(\varphi_2 - 1)} + \frac{1}{\beta_2(1 - \lambda_2)} \ln \left(1 + \frac{\alpha_2}{\beta_2} \right) \\
&\quad + \frac{\ln(|x_e(0)|) - \ln(\varepsilon_1)}{\alpha_1} + \frac{1}{\beta_1(1 - \delta_1)} |\varepsilon_1|^{1-\alpha_1}
\end{aligned} \tag{3.30}$$

3.3. MẠNG NƠ-RON NHÂN TẠO

Mạng nơ-ron nhân tạo là dùng kỹ thuật tái tạo lại một vài chức năng tương tự bộ não con người. Việc nghiên cứu ứng dụng mạng nơ-ron mở ra một hướng mới trong việc giải quyết các bài toán kỹ thuật và kinh tế... Trong bài toán kỹ thuật, mạng nơ-ron có thể nhận dạng, điều khiển, nhận mẫu, giải quyết các bài toán tối ưu... và tỏ ra rất có hiệu quả.

Mô hình nơ-ron nhân tạo cơ bản thứ i trong mạng xây dựng dựa trên cấu trúc của nơ-ron sinh học do McCulloch và Pitts đề xuất và được Rosenblatt cải

tiến, gọi là perceptron. Nó có thể có nhiều đầu vào (R đầu vào) và chỉ có một đầu ra (Hình 3.1).



Hình 3.1 - Mô hình nơ-ron nhân tạo

Quan hệ giữa đầu ra và các đầu vào của nơ-ron thứ i:

$$y_i = a_i(q_i) = a_i(f_i(p)) \quad (3.31)$$

Trong đó:

p : vector biến đầu vào;

f_i : hàm tổng hợp;

a_i : hàm chuyển đổi;

y_i : biến đầu ra của nơ-ron thứ i;

$q_i = f_i(p) = \sum_{j=1}^R w_{ij} \cdot p_j - \theta_i$: tổng trọng số;

w_{ij} : trọng số liên kết giữa đầu vào thứ j với nơ-ron thứ i;

θ_i : ngưỡng của nơ-ron thứ i (hằng số).

Có nhiều dạng hàm tổng hợp $f_i(\cdot)$ được dùng sau đây:

- Hàm tổng hợp tuyến tính:

$$f_i(p) = \sum_{j=1}^R w_{ij} \cdot p_j - \theta_i \quad (3.32)$$

- Hàm bậc hai (Quadratic function):

$$f_i(p) = \sum_{j=1}^R w_{ij} \cdot p_j^2 - \theta_i \quad (3.33)$$

- Ngoài ra còn có các hàm hình bán cầu, hàm đa thức...

Với mục đích đơn giản, thực tế thường chọn hàm tổng hợp tuyến tính.

Hàm chuyển đổi $a(.)$ cũng có rất nhiều dạng được dùng, ví dụ một vài dạng hàm cơ bản như sau:

- Hàm chuyển đổi tuyến tính (linear function):

$$a(q) = q \quad (3.34)$$

- Hàm dấu (hàm ngưỡng - threshold function):

$$a(q) = \text{sgn}(q) = \begin{cases} 1 & \text{if } q \geq 0 \\ -1 & \text{if } q < 0 \end{cases} \quad (3.35)$$

- Hàm sigmoid một cực (unipolar sigmoid function):

$$a(q) = \frac{1}{1+e^{-\lambda q}} \quad (3.36)$$

- Hàm sigmoid hai cực (bipolar sigmoid function):

$$a(q) = \frac{1-e^{-\lambda q}}{1+e^{-\lambda q}} \quad (3.37)$$

Mô hình mạng nơ-ron được hình thành từ việc liên kết các nơ-ron với nhau theo một nguyên tắc nào đó. Có rất nhiều loại mạng và việc phân loại mạng cũng có nhiều cách:

- Theo số lớp: có mạng nơ-ron một lớp, mạng nơ-ron nhiều lớp.
- Theo cấu trúc liên kết giữa đầu vào và đầu ra: có mạng nơ-ron truyền thẳng, mạng nơ-ron hồi quy.
- Theo tính chất làm việc: có mạng tĩnh (static network) và mạng động (dynamic network)...

Phần tử gây trễ (TDL – Tapped Delay Line): là phần tử có tín hiệu ra của nó bị trễ một khoảng thời gian so với tín hiệu vào, có hai tham số trễ là thời gian trễ (bước) và bậc trễ. Phần tử này được sử dụng để lấy tín hiệu quá khứ và nó là yếu tố để tạo ra các đơn vị nơ-ron động lực học (Dynamic Neural Units) và mạng động lực học trong hệ thống rời rạc (mạng động).

Quan hệ vào - ra phần tử trễ như sau:

$$q(t) = w.p(t-i.\tau) \quad (3.38)$$

Trong đó, w là trọng số của TDL; i là bậc trễ, $i = 0, 1...n$; τ là thời gian trễ một bậc.

a) Một số dạng mạng nơ-ron thường gặp

❖ Mạng một lớp truyền thẳng (single-layer feedforward network):

Các nơ-ron cùng nhận đồng thời tín hiệu vào như nhau và có chức năng giống nhau tạo thành một lớp của mạng nơ-ron, số nơ-ron trong một lớp chính là số đầu ra của lớp đó. Với mạng nơ-ron một lớp truyền thẳng (ví dụ có R đầu vào, S đầu ra). Ma trận trọng số W sẽ có S hàng, R cột. Trong đó phần tử w_{ij} là trọng số liên kết giữa nơ-ron thứ i và đầu vào thứ j , có các hàm tổng hợp f_i , hàm chuyển đổi a_i (các hàm này thường chọn giống nhau cho mỗi lớp: $a_i = a$, $f_i = f$).

Quan hệ vào - ra có dạng sau:

$$y = a(f(x)) \quad (3.39)$$

Trong đó,

$y = [y_1, y_2 \dots y_S]^T$ là vector biến tín hiệu ra;

$x = [x_1, x_2 \dots x_R]^T$ là vector tín hiệu vào.

❖ Mạng nhiều lớp truyền thẳng (multi-layers feedforward network):

Gồm nhiều lớp (N lớp) ghép liên tiếp với nhau, đầu ra của lớp này được nối với đầu vào của lớp ngay sau nó. Lớp đầu tiên là lớp vào (input layer) có R đầu vào, S_1 đầu ra. Lớp cuối cùng là lớp ra (output layer) có S_{N-1} đầu vào, S_N (gọi tắt là S) đầu ra. Giữa chúng có thể cũng có một số lớp nối liên tiếp nhau gọi là các lớp ẩn (hidden layers), chúng đóng vai trò trung gian trong mạng, không tiếp xúc trực tiếp với bên ngoài. Mỗi lớp ẩn (ví dụ lớp thứ k) có S_{k-1} đầu vào và S_k đầu ra.

Các nơ-ron trong một lớp được nối theo cấu trúc ghép nối hoàn toàn, nghĩa là mỗi nơ-ron sẽ được nối với tất cả các tín hiệu vào của lớp đó và các nơ-ron trong cùng lớp có cấu trúc liên kết hoàn toàn giống nhau.

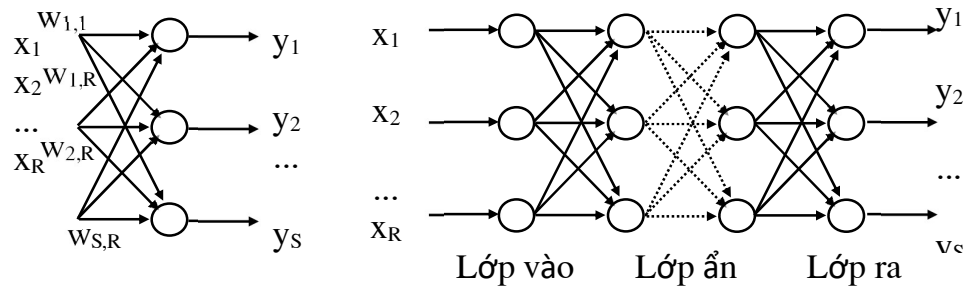
Quan hệ vào - ra của mạng có dạng sau:

$$y_S = a^N(f^N(a^{N-1}(\dots(a^1(f^1(x)))))) \quad (3.40)$$

Trong đó,

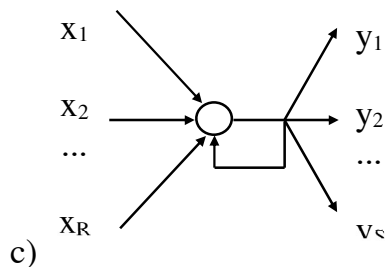
- $y_S = [y_1, y_2 \dots y_S]^T$ là vector biến tín hiệu ra;

- a^k, f^k ($k = 1 \dots N$) là hàm kích hoạt và hàm tổng hợp của lớp thứ k .

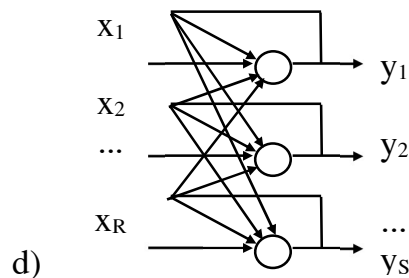


a)

b)



c)



d)

Hình 3.2 - Một số dạng liên kết mạng

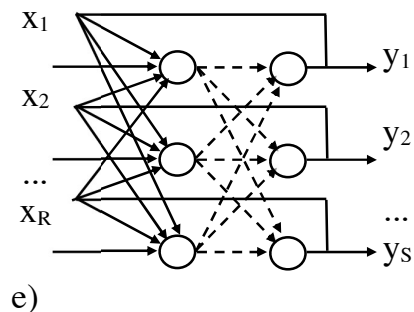
a) Mạng một lớp truyền thẳng.

b) Mạng nhiều lớp truyền thẳng.

c) Nút tự hồi quy.

d) Mạng một lớp hồi quy.

e) Mạng nhiều lớp hồi quy.



e)

❖ **Mạng nơ-ron hồi quy (recurrent network):**

Mạng nơ-ron hồi quy còn gọi là mạng phản hồi (feedback network), là loại mạng mà tín hiệu ra của nơ-ron được phản hồi ngược về làm tín hiệu vào cho các nơ-ron lớp trước nó hoặc cùng lớp đó tạo thành các vòng kín (hình 4.3d, e). Tín hiệu phản hồi có thể có phần tử TDL.

b) Các luật học

Việc “học” của mạng nơ-ron là để tìm được chính xác giá trị ma trận trọng số liên kết W của các nơ-ron và xác định được cấu trúc cụ thể của mạng để giải quyết được bài toán cụ thể. Có hai kiểu học là học cấu trúc và học thông số:

- Học cấu trúc (Structure learning): Xác định cấu trúc của mạng bao gồm số lượng nút (nơ-ron) trong mỗi lớp và giá trị của các ma trận trọng số W của mạng (số lớp không thay đổi).

- Học thông số (Parameter learning): Tìm được chính xác giá trị của các ma trận trọng số W ứng với cấu trúc (cố định) của mạng nơ-ron đã cho.

Có ba phương pháp học cơ bản sau:

- Học có giám sát (Supervised learning): Tín hiệu giám sát là những thông tin mong muốn d được cung cấp từ bên ngoài mà đầu ra y của mạng nơ-ron cần phải đạt được.

- Học củng cố (Reinforcement learning): Thông tin cung cấp từ bên ngoài d (tín hiệu củng cố) mang tính định hướng cho quá trình học (cho biết tín hiệu ra của mạng đúng hay sai) .

- Học không giám sát (Unsupervised learning): Quá trình học không có bất kỳ thông tin nào từ bên ngoài để giám sát, đó là quá trình tự cấu trúc của mạng. Mạng phải tự xác định các cặp dữ liệu mẫu (dữ liệu vào, ra), các tính chất, các mối quan hệ để tạo được ma trận W mong muốn.

Dạng chung của luật học thông số là xác định giá trị cần điều chỉnh ΔW_i cho vector trọng số W_i :

Với mạng rời rạc:

$$\Delta W_i(t) = \eta \cdot r \cdot x(t) \quad (3.41)$$

Trong đó,

- η là số dương xác định tốc độ học (hằng số học),

- r là tín hiệu học, $r = f_r(W_i, x, d_i)$.

Lúc đó giá trị vector trọng số tại thời điểm $(t+1)$ là:

$$W_i(t+1) = W_i(t) + \eta \cdot f_r(W_i, x(t), d_i) \cdot x(t) \quad (3.42)$$

Với mạng liên tục:

$$\frac{dW_i(t)}{dt} = \eta \cdot r \cdot x(t) \quad (3.43)$$

Có nhiều thuật toán học để xác định $\Delta W_i(t)$. Một trong những thuật toán học có giám sát cho mạng nơ-ron được Widrow-Hoff đề nghị là thuật toán “Gradient descent”, sử dụng nguyên lý cực tiểu sai lệch trung bình bình phương bằng việc thêm vào các trọng số liên kết mạng một giá trị theo hướng ngược với hướng tức thời của gradient. Thuật toán sử dụng phương pháp sai lệch trung bình bình phương nhỏ nhất (LMS: Least Mean Squares) hoặc gọi là phương pháp delta.

Hàm mục tiêu chuẩn sử dụng trong mạng nơ-ron, sai lệch trung bình bình phương, của thuật toán này là định nghĩa một hàm đạt cực tiểu E như sau:

$$E = \frac{1}{2} \sum_{n=1}^{N_i} (d_n(k) - y_n^L(k))^2 \rightarrow \min \quad (3.44)$$

Trong đó, $d_n(k)$ là đầu ra mong muốn thứ n tại thời điểm t; L là số lớp.

3.4. BÙ THÍCH NGHIỆ THÀNH PHẦN BẤT ĐỊNH BẰNG MẠNG NƠ-RON

Để cải tiến bộ điều khiển trượt bậc 2 siêu xoắn ứng dụng mạng nơ-ron, trong đề tài này đề xuất ứng dụng mạng nơ-ron (NN) truyền thẳng nhiều lớp để bù cho các thành phần bất định của hệ thống.

Đề tài đề xuất cho đối tượng tay máy robot 2 bậc tự do, mạng nơ-ron truyền thẳng đề xuất có cấu trúc như Hình 3.3: bao gồm lớp vào (input layer), lớp ẩn (hidden layer) và lớp ra (output layer). Lớp ra của mạng có 2 đầu ra tương ứng với tín hiệu điều khiển 2 khớp nối của tay máy robot 2 bậc tự do.

Lớp ngõ vào có vector tín hiệu vào xác định như sau:

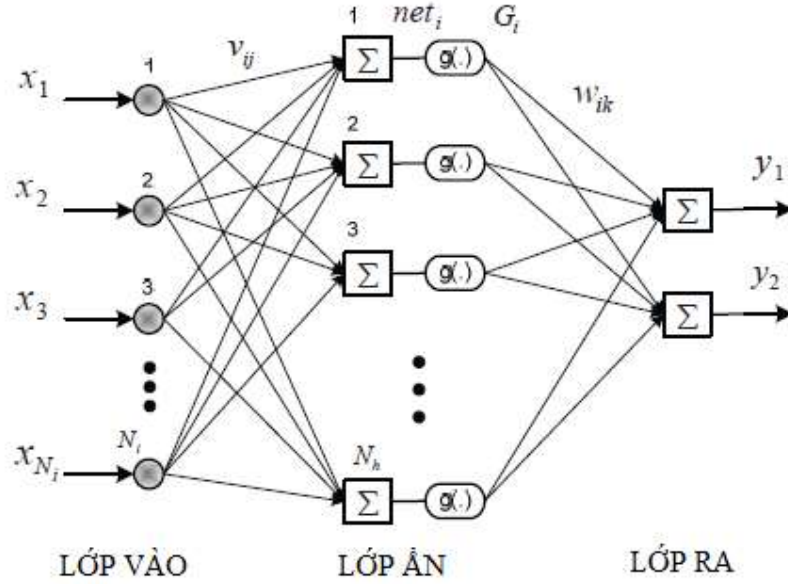
$$x = [x_1 \quad x_2 \quad \dots \quad x_{N_i}]^T \quad (3.45)$$

Trong đó, N_i là số lượng ngõ vào.

Lớp ẩn có N_h nơ-ron, ma trận trọng số liên kết giữa ngõ vào và lớp ẩn được xác định như sau:

$$V = [v_1, v_2, \dots, v_{N_h}]^T \in R^{N_h \times N_i},$$

$$v_i = [v_{i1}, v_{i2}, \dots, v_{iN_i}] \in R^{N_i}, i = \overline{1, N_h} \quad (3.46)$$



Hình 3.3 - Cấu trúc của mạng nơ-ron truyền thẳng ba lớp

Quan hệ vào-ra của lớp ẩn được xác định như sau:

$$net_i = \sum_{j=1}^{N_i} v_{ij}x_j \quad (3.47)$$

$$G_i = g(net_i), i = \overline{1, N_h} \quad (3.48)$$

Trong đó $g(.)$ là hàm truyền (chuyển đổi) của lớp ẩn, được xác định bằng hàm sigmoid đơn cực như sau:

$$g(z) = \frac{1}{1+e^{-z}} \quad (3.49)$$

Lớp ngõ ra có ma trận trọng số liên kết với lớp ẩn xác định như sau:

$$W = [w_1, w_2, \dots, w_{N_h}]^T \in R^{N_h \times 2},$$

$$w_i = [w_{i1}, w_{i2}] \in R^2, i = \overline{1, N_h} \quad (3.50)$$

Các ngõ ra của mạng nơ-ron được xác định như sau:

$$y_k = \sum_{i=1}^{N_h} w_{ik}G_i, k = 1,2 \quad (3.51)$$

Một cách tổng quát, các ngõ ra của mạng nơ-ron trên có thể được viết lại dưới dạng vector như sau:

$$y = W^T G(x, V) \in R^2 \quad (3.52)$$

trong đó: $G = [G_1, G_2 \dots G_{N_h}]^T \in R^{N_h \times 1}$

Ở chương 2 của thuyết minh đề tài đã trình thuật toán điều khiển trượt bậc 2 siêu xoắn cho tay máy robot công nghiệp:

$$\begin{aligned} u(t) = & -B_0(q)(f_0(q, \dot{q}) + \lambda \dot{e}(t) + \beta r(0) \exp(-\beta t) + ks(t)G^T \\ & + \mu_0 |s(t)G^T|^{0.5} \text{sgn}(s(t)G^T + v_a) \\ \dot{v}_a = & \hat{\Gamma} \text{sgn}(s(t)G^T) \end{aligned} \quad (3.53)$$

Thuật toán này có thể được cải tiến hơn nữa bằng cách sử dụng mạng neural truyền thẳng theo cấu trúc ở Hình 3.3 ở trên để bổ sung vào phương trình của thuật toán nhằm bù thích nghi các thành phần bất định. Nhờ vậy sẽ tăng được độ chính xác điều khiển quỹ đạo và giảm bớt sự hoạt động của các thành phần hàm dấu $\text{sgn}(s)$ do các thành phần bất định, nhiễu loạn gây ra.

Phương trình đề xuất của thuật toán điều khiển trượt bậc 2 siêu xoắn thích nghi sử dụng mạng neural nhân tạo như sau:

$$u(t) = u_{ed} + u_{aux}(t) + u_{NN} + f_{est} \quad (3.54)$$

Gọi $\Delta U_r \in R^2$ là tổng các thành phần bất định của hệ thống. Và $u_{NN} \in R^2$ là ngõ ra của mạng nơ-ron có cấu trúc như đã trình bày ở phần trước. Tín hiệu u_{NN} sẽ thực hiện việc xấp xỉ và bù online cho thành phần bất định ΔU_r . Tuy nhiên, trong thực tế ngõ ra của mạng nơ-ron không thể xấp xỉ chính xác giá trị ΔU_r , dẫn đến sai số xác lập của hệ thống không thể hội tụ về 0, do đó ta cần phải dùng thêm bộ ước lượng f_{est} nhằm làm suy giảm các sai lệch.

Trong đó, $\tilde{s} = [\tilde{s}_1, \tilde{s}_2]^T$ là mặt trượt bậc nhất được định nghĩa như sau:

$$\tilde{s} = \dot{q}_e + \lambda q_e \quad (3.55)$$

Ta chọn ngõ vào của mạng nơ-ron là một vector bao gồm các tín hiệu sai lệch và đạo hàm của sai lệch:

$$x = [q_{e1}, \dot{q}_{e1}, q_{e2}, \dot{q}_{e2}]^T \in R^4 \quad (3.56)$$

Mục tiêu của mạng nơ-ron là xấp xỉ giá trị thành phần bất định ΔU_r , do đó sẽ tồn tại một sai số nhất định giữa giá trị chính xác (tối ưu) và giá trị ước lượng như sau:

$$\Delta U_r - U_{NN} = W^{*T} G(x, V^*) - \hat{W}^T G(x, \hat{V}) + \varepsilon_N \quad (3.57)$$

Trong đó:

- $W^* \in R^{N_h \times 2}$ và $V^* \in R^{4 \times N_h}$ là các giá trị tối ưu của các ma trận trọng số W và V trong mạng nơ-ron;

- $\hat{W} \in R^{N_h \times 2}$ và $\hat{V} \in R^{4 \times N_h}$ là giá trị ước lượng của các ma trận trọng số tối ưu (W^* và V^*);

- $\varepsilon \in R^2$: là sai số tái cấu trúc do thiếu số lượng nơ-ron trong lớp ẩn.

Để đơn giản, phương trình (3.57) được viết lại như sau:

$$\Delta U_r - U_{NN} = W^{*T} \tilde{G} + \tilde{W}^T \hat{G} + \varepsilon_N \quad (3.58)$$

Trong đó, các ký hiệu G^* , \hat{G} , \tilde{G} và \tilde{W} được định nghĩa như sau:

$$G^* \equiv G(x, V^*) \in R^{N_h};$$

$$\hat{G} \equiv G(x, \hat{V}) \in R^{N_h};$$

$$\tilde{G} = G^* - \hat{G} \in R^{N_h};$$

$$\tilde{W} = W^* - \hat{W} \in R^{N_h \times 2}.$$

Khai triển Taylor mở rộng cho \tilde{G} với mỗi giá trị x cho trước như sau:

$$\tilde{G} = \begin{bmatrix} \tilde{G}_1 \\ \tilde{G}_2 \\ \vdots \\ \tilde{G}_{N_h} \end{bmatrix}_{|V=\hat{V}} + \begin{bmatrix} \frac{\partial \tilde{G}_1}{\partial V^T} \\ \frac{\partial \tilde{G}_2}{\partial V^T} \\ \vdots \\ \frac{\partial \tilde{G}_{N_h}}{\partial V^T} \end{bmatrix}_{|V=\hat{V}} (V^* - \hat{V}) + O(\hat{V}^T x) = G_V \tilde{V}^T x + O(\hat{V}^T x) \quad (3.59)$$

Trong đó:

$$G_V = \begin{bmatrix} \frac{\partial G_1}{\partial V^T x} & \frac{\partial G_2}{\partial V^T x} & \dots & \frac{\partial G_{N_h}}{\partial V^T x} \end{bmatrix}_{|V=\hat{V}}^T \in R^{N_h \times N_h}$$

$$\tilde{V} = V^* - \hat{V} \in R^{4 \times N_h}$$

$O(\hat{V}^T x) \in R^{N_h}$ là vector tổng các thành phần bậc cao, được giả định là bị chặn.

Thay (3.59) vào phương trình sai lệch xấp xỉ (3.58), ta được:

$$\begin{aligned} \Delta U_r - U_{NN} &= W^{*T} G_V \tilde{V}^T x + \tilde{W}^T \hat{G} + W^{*T} O(\hat{V}^T x) + \varepsilon_N \\ &= \hat{W}^T G_V \tilde{V}^T x + \tilde{W}^T \hat{G} + \delta \end{aligned} \quad (3.60)$$

Trong đó:

$$\delta = \tilde{W}^T G_V \tilde{V}^T x + W^{*T} O(\hat{V}^T x) + \varepsilon_N \quad (3.61)$$

Từ đây, ta sẽ thiết kế thêm bộ ước lượng sai số f_{est} để ước lượng và bù cho vector sai lệch $\delta \in R^2$ của mạng nơ-ron.

Giải thuật học của mạng nơ-ron và cấu trúc của bộ ước lượng sai số được chọn như sau:

$$\dot{\hat{W}} = -\eta \hat{G} \tilde{s}^T \quad (3.62)$$

$$\dot{\hat{V}} = -\mu x (G_V \hat{W} \tilde{s})^T \quad (3.63)$$

$$\dot{f}_{est} = -\xi \tilde{s} \quad (3.64)$$

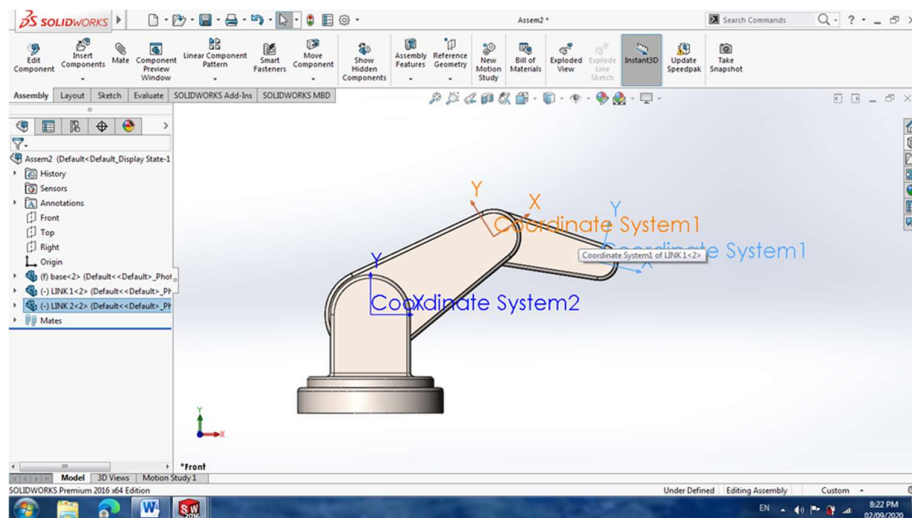
$$f_{est} = -\xi \int \tilde{s} dt \quad (3.65)$$

CHƯƠNG 4

MÔ PHỎNG KIỂM NGHIỆM

4.1. XÂY DỰNG MÔ HÌNH CƠ KHÍ CỦA ROBOT TRÊN SOLIDWORKS VÀ TÍCH HỢP VÀO SIMULINK CỦA MATLAB

Để mô phỏng chính xác phần cơ khí của tay máy robot, đề tài sử dụng phần mềm Solidworks để thiết kế chi tiết các thành phần cơ khí của robot. Sau đó gắn kết các thành phần lại với nhau trên Solidworks, kiểm chứng các chuyển động, và biên dịch mô hình để tích hợp vào trong Multibody của Simscape, Matlab-Simulink. Sơ đồ mô phỏng cơ khí của tay máy robot như trên Hình 4.1.



Hình 4.1 – Mô phỏng tay máy robot trên phần mềm Solidworks

Ban đầu, để kiểm nghiệm các thuật toán điều khiển trượt bậc cao so sánh với các thuật toán điều khiển truyền thống, mô phỏng sử dụng 2 bậc tự do của tay máy robot. Thông số như sau:

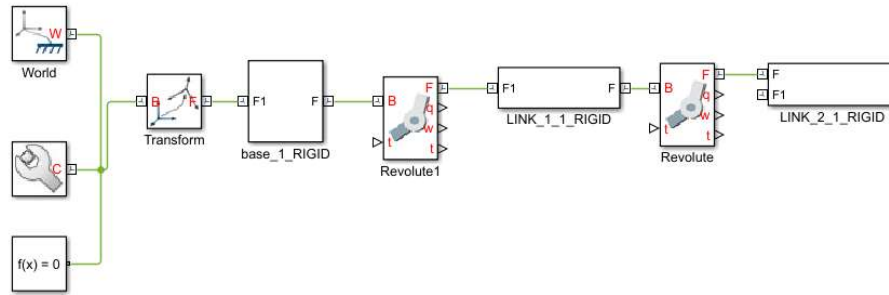
Chiều dài thanh: $l_1 = 0.8m$; $l_2 = 0.5m$

Trọng tâm: $r_1 = 0.2729m$; $r_2 = 0.2047m$

Khối lượng: $m_0 = 507kg$; $m_1 = 202kg$; $m_2 = 47.6kg$

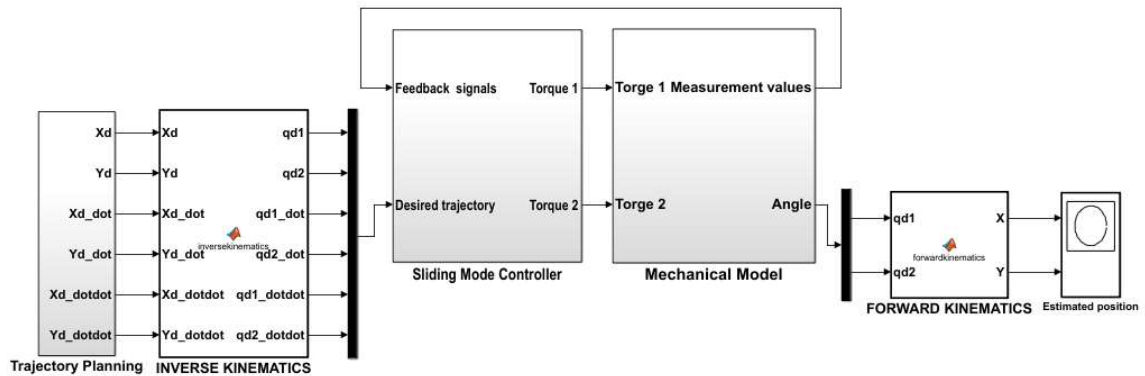
Moment quán tính: $I_1 = 1.59 \text{ kg} / m^2$; $I_2 = 0.12 \text{ kg} / m^2$

Tay máy robot được thiết kế bên Solidworks sau tích hợp qua Matlab Simulink sẽ có mô hình như trên Hình 4.2.



Hình 4.2 - Mô hình cơ khí tay máy robot sau biên dịch và tích hợp qua Multibody, Simscape của Matlab-Simulink

Điều khiển tay máy robot bám theo quỹ đạo cho trước, với quỹ đạo được thiết kế là một đường tròn tâm I(0.9;0.3) bán kính $r=0.3m$, mô hình mô phỏng của toàn bộ hệ thống trên Matlab-Simulink như Hình 4.3.

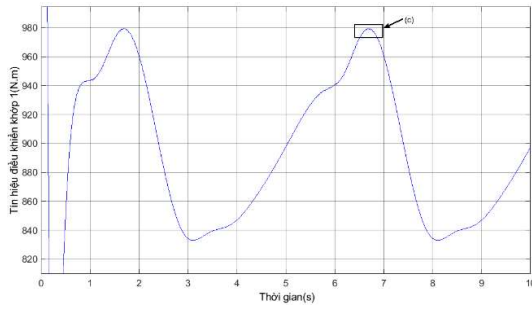


Hình 4.3 - Sơ đồ khối mô hình toàn hệ thống trên Matlab-Simulink

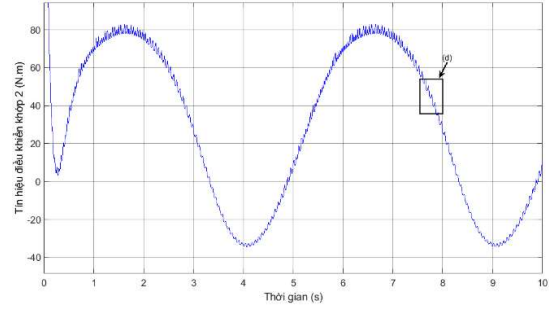
4.2. MÔ PHỎNG SO SÁNH CÁC THUẬT TOÁN ĐIỀU KHIỂN

4.2.1 Thuật toán PD bù trọng lực

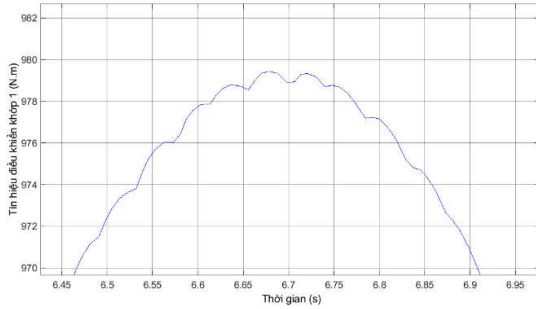
Mô phỏng trường hợp tay máy robot được điều khiển bởi thuật toán PD có bù trọng lực với thông số bộ điều khiển như sau: $k_p = 5000$; $k_d = 1000$



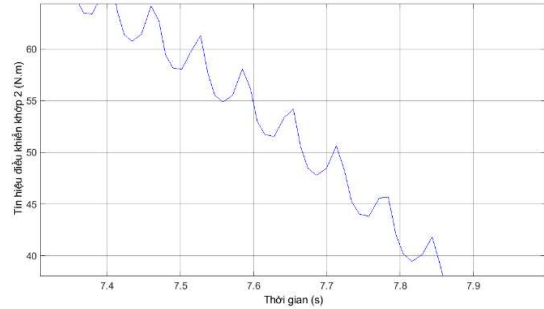
(a)



(b)

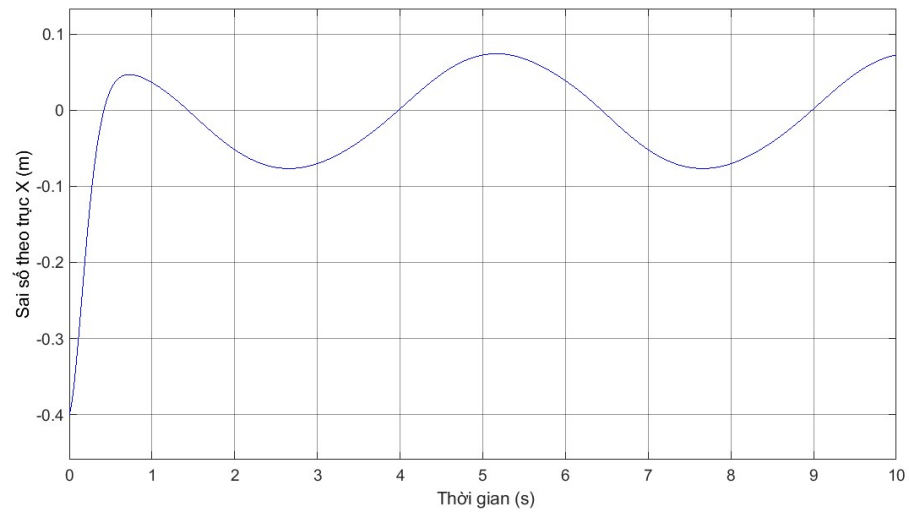


(c)

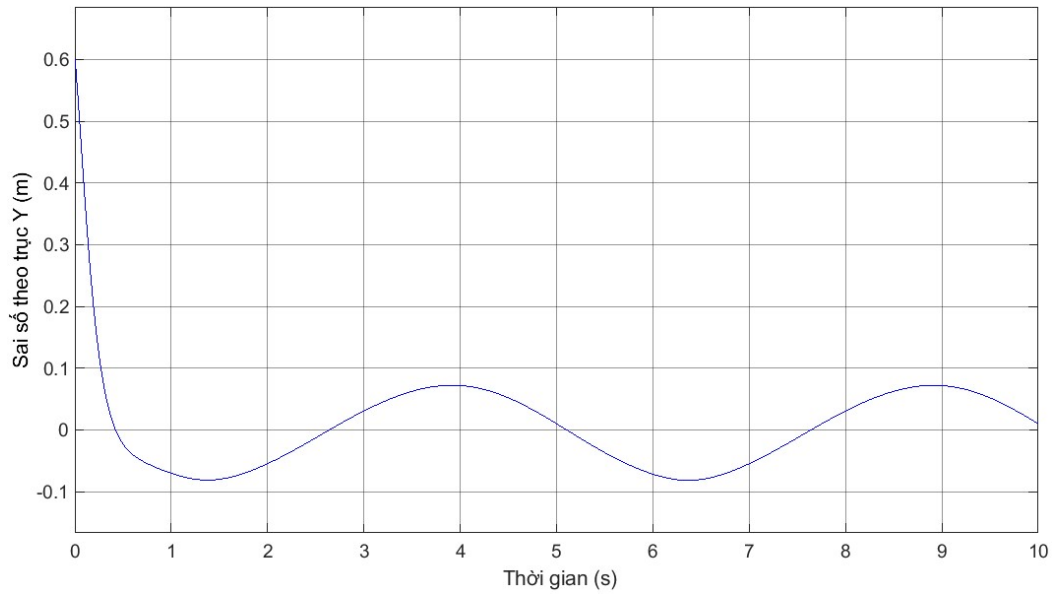


(d)

Hình 4.4 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán PD bù trọng lực
(a) Tín hiệu điều khiển khớp 1; (b) Tín hiệu điều khiển khớp 2; (c) Phóng to của tín hiệu điều khiển khớp 1; (d) Phóng to của tín hiệu điều khiển khớp 2



Hình 4.5 - Đồ thị sai số quỹ đạo của thuật toán PD theo trục X

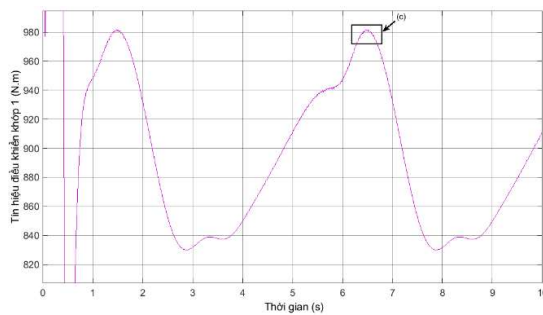


Hình 4.6 - Đồ thị sai số quỹ đạo của thuật toán PD theo trục Y

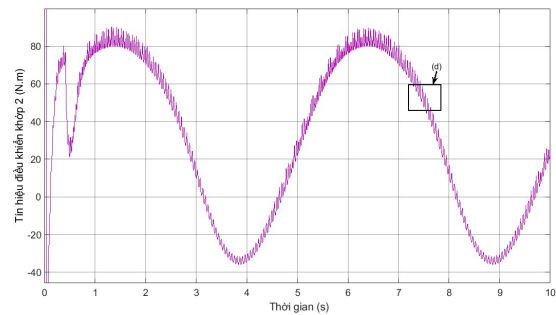
Nhận xét: Kết quả mô phỏng cho thấy, với bộ điều khiển PD bù trọng lực, tính hiệu điều khiển moment của cả 2 khớp vẫn còn có sự đập mạch lớn. Và kết quả điều khiển bám quỹ đạo của các khớp theo phương trục X và phương trục Y có sai số khá lớn.

4.2.2. Thuật toán điều khiển trượt (SMC) sử dụng phương pháp lớp biên (Boundary Layer Method)

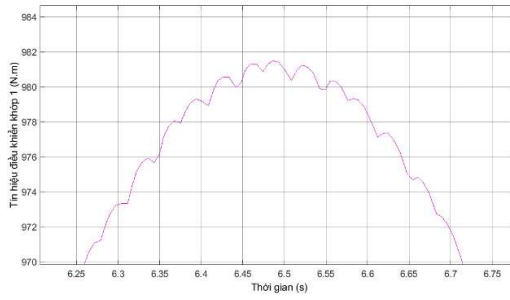
Mô phỏng điều khiển tay máy robot sử dụng thuật toán điều khiển SMC với phương pháp lớp biên để giảm chattering được thực hiện với các thông số như sau: $\lambda_1 = \lambda_2 = 10$; $k_1 = k_2 = 500$; $\varepsilon_1 = \varepsilon_2 = 0.4$



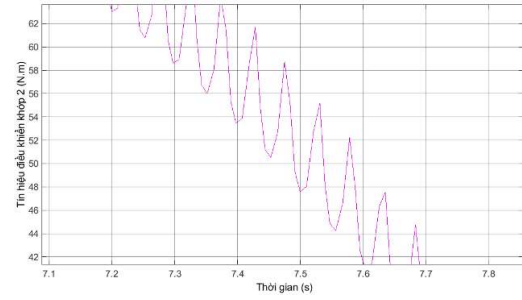
(a)



(b)

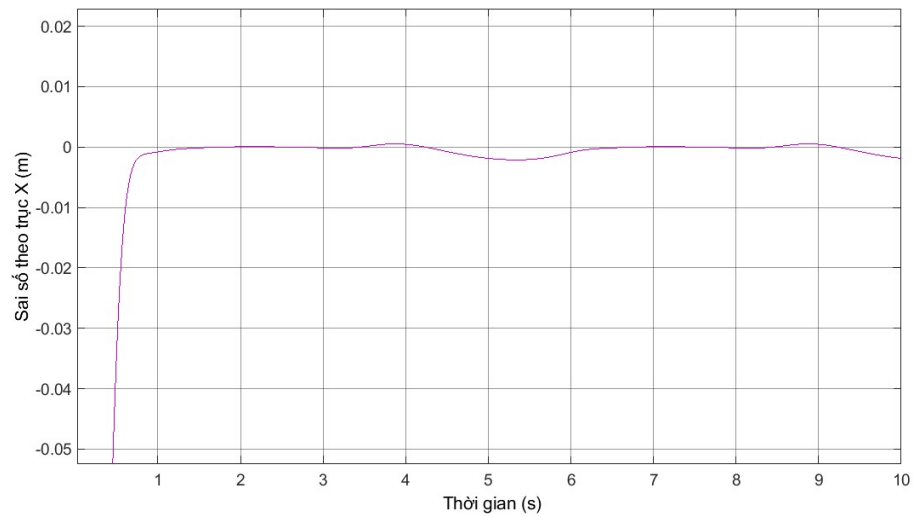


(c)

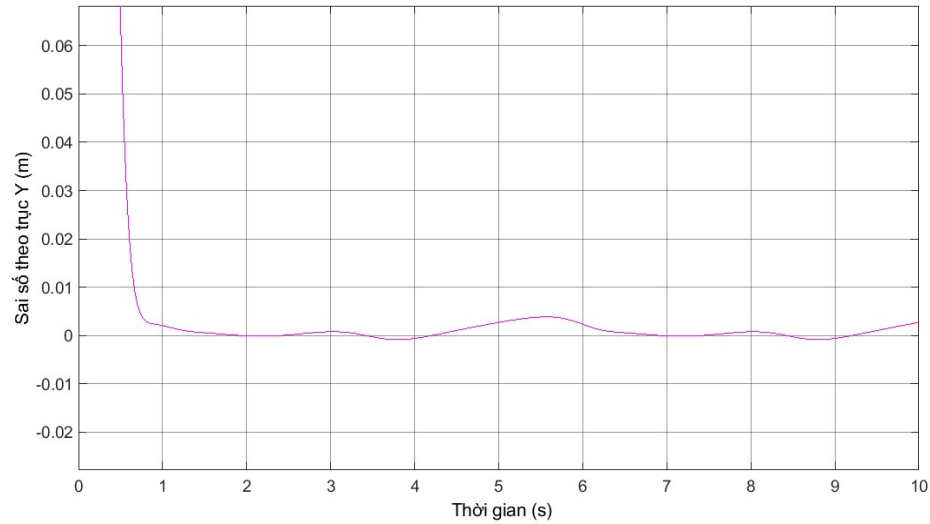


(d)

Hình 4.7 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán BLM SMC
(a) Tín hiệu điều khiển khớp 1; (b) Tín hiệu điều khiển khớp 2; (c) Phóng to của tín hiệu điều khiển khớp 1; (d) Phóng to của tín hiệu điều khiển khớp 2



Hình 4.8 - Đồ thị sai số quỹ đạo của thuật toán BLMSMC theo trục X



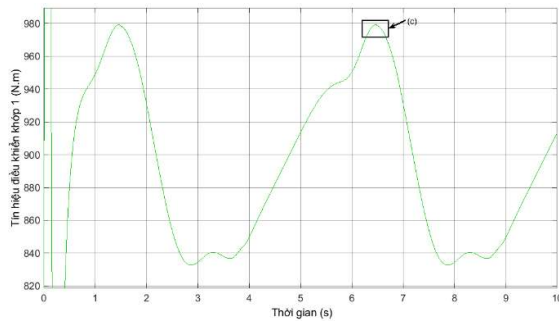
Hình 4.9 - Đồ thị sai số quỹ đạo của thuật toán BLSMC theo trục Y

Nhận xét: Kết quả mô phỏng cho thấy trong trường hợp điều khiển tay máy robot sử dụng thuật toán điều khiển trượt với phương pháp lớp biên thì kết quả điều khiển bám theo quỹ đạo chính xác hơn, sai số nhỏ hơn so với trường hợp thuật toán PD bù trọng lực. Tuy nhiên, đồ thị tín hiệu điều khiển cho thấy ở cả 2 khớp vẫn còn hiện tượng đập mạch liên tục (hiện tượng chattering) điều này làm hao mòn cơ khí, giảm tuổi thọ thiết bị nên phương pháp điều khiển này khó áp dụng vào thực tế.

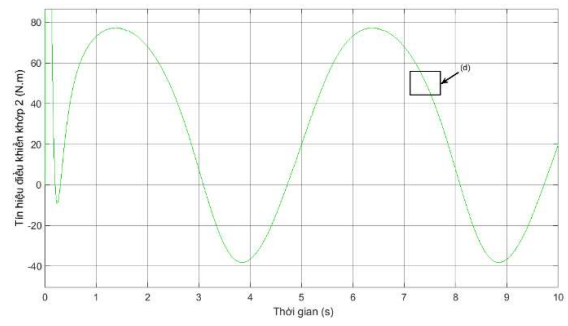
4.2.3 Thuật toán điều khiển trượt bậc 2 xoắn (Twisting Second Order Sliding Mode Control - TSOSMC)

Mô phỏng điều khiển tay máy robot sử dụng thuật toán điều khiển trượt bậc 2 xoắn TSOSMC được thực hiện với các thông số như sau:

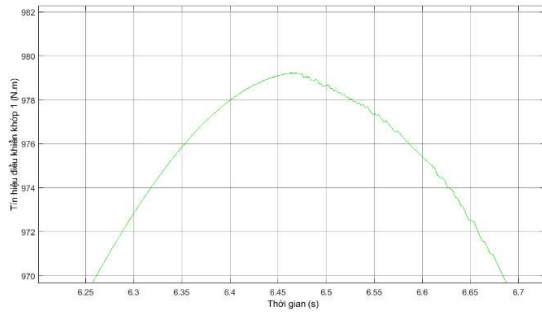
$$\lambda_1 = \lambda_2 = 10; \quad \eta_{i1} = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}; \quad \eta_{i2} = \begin{bmatrix} 2500 & 0 \\ 0 & 2500 \end{bmatrix}$$



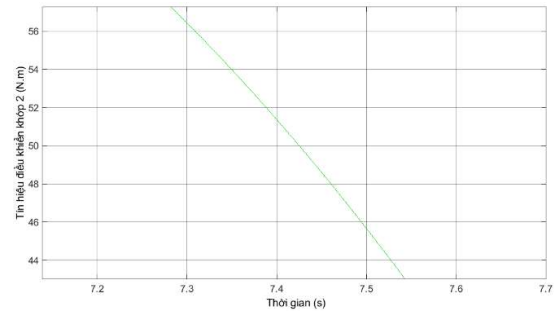
(a)



(b)

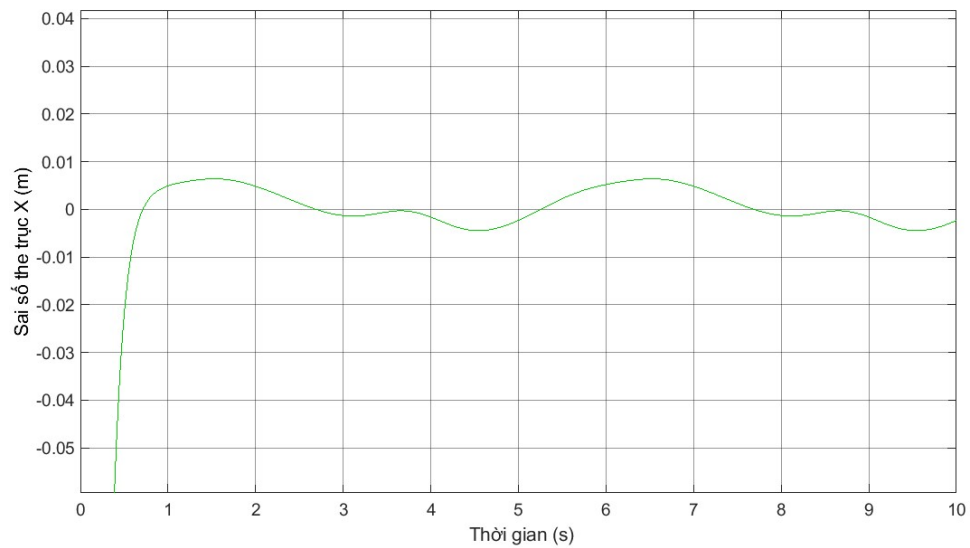


(c)

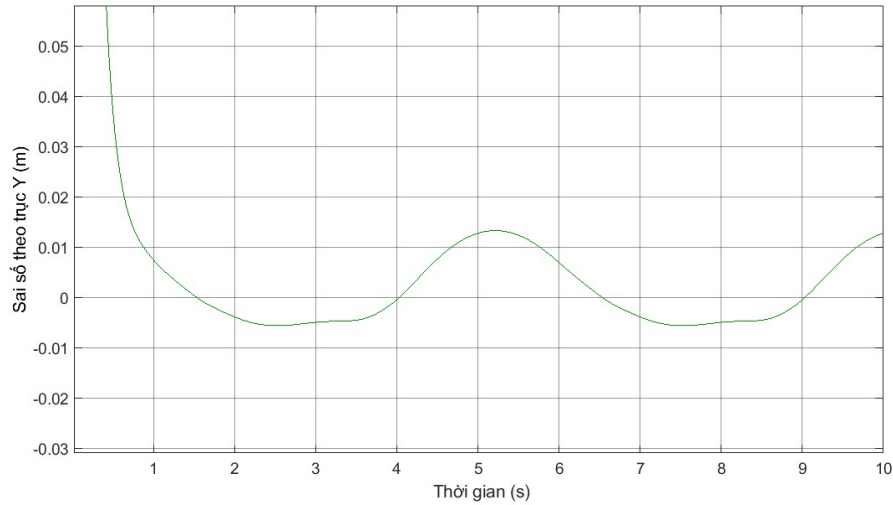


(d)

Hình 4.10 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán TSOSMC
(a) Tín hiệu điều khiển khớp 1; (b) Tín hiệu điều khiển khớp 2; (c) Phóng to của tín hiệu điều khiển khớp 1; (d) Phóng to của tín hiệu điều khiển khớp 2



Hình 4.11 - Đồ thị sai số quỹ đạo của thuật toán TSOSMC theo trục X



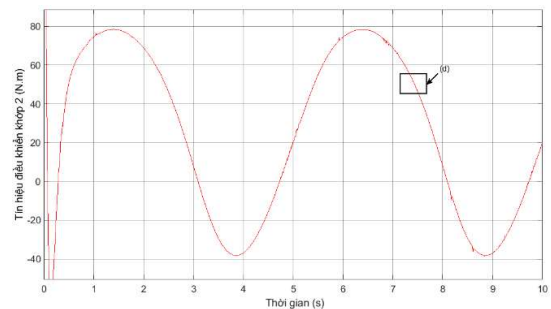
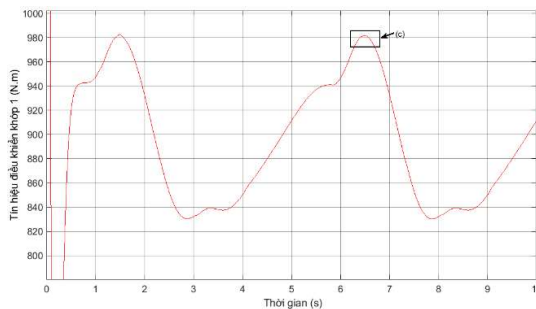
Hình 4.12 - Đồ thị sai số quỹ đạo của thuật toán TSOSMC theo trục Y

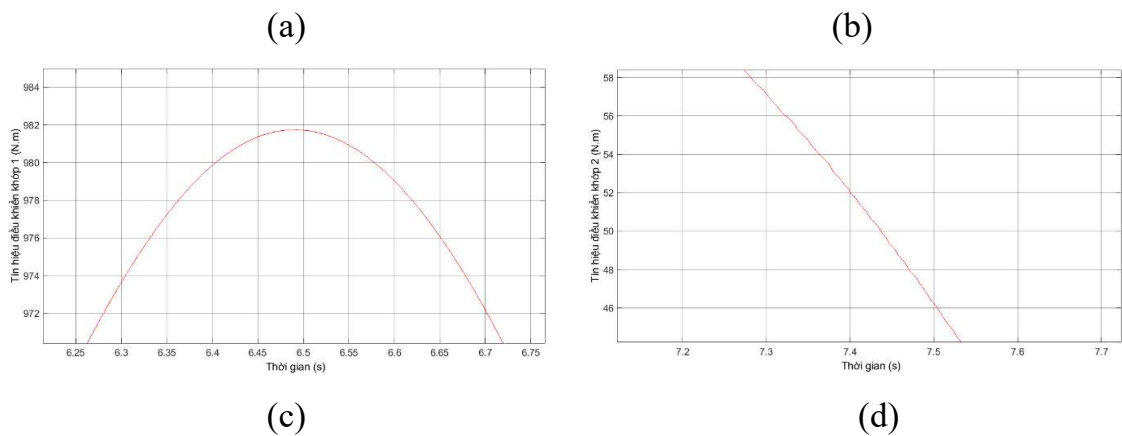
Nhận xét: Kết quả mô phỏng cho thấy trong trường hợp điều khiển tay máy robot sử dụng thuật toán điều khiển trượt bậc 2 xoắn thì hiện tượng đập mạch (chattering) của tín hiệu điều khiển của 2 khớp đã giảm rất đáng kể so với trường hợp sử dụng điều khiển trượt truyền thống. Tuy nhiên, một số thời điểm vẫn còn hiện tượng đập mạch ở tín hiệu điều khiển khớp 1 và độ chính xác đã giảm đi một chút nên thuật toán này vẫn chưa tối ưu.

4.2.4 Thuật toán điều khiển trượt bậc 2 siêu xoắn (Super Twisting Second Order Sliding Mode Control - STSOSMC)

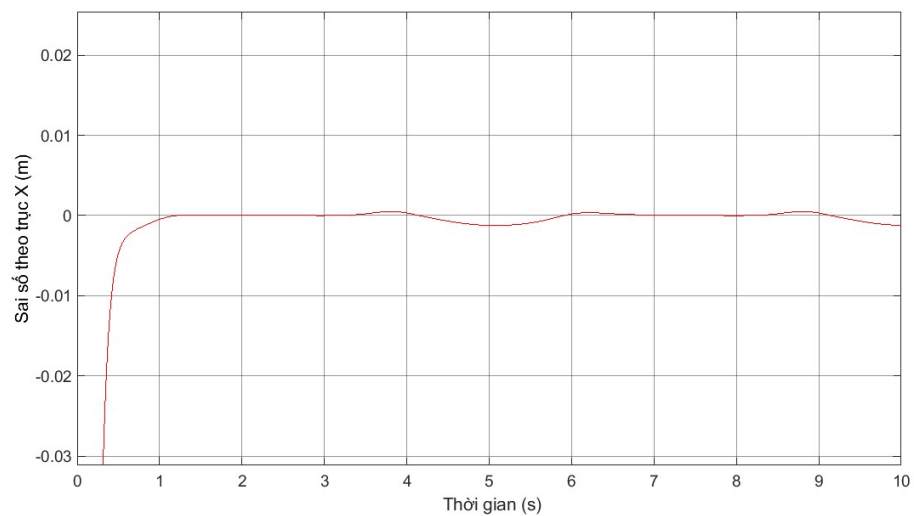
Mô phỏng điều khiển tay máy robot sử dụng thuật toán điều khiển trượt bậc 2 siêu xoắn STSOSMC được thực hiện với các thông số như sau:

$$\lambda_1 = \lambda_2 = 10; \quad \eta_{i1} = \begin{bmatrix} 2500 & 0 \\ 0 & 2000 \end{bmatrix}; \quad \eta_{i2} = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}; \quad \eta_{i3} = \begin{bmatrix} 40 & 0 \\ 0 & 40 \end{bmatrix}$$

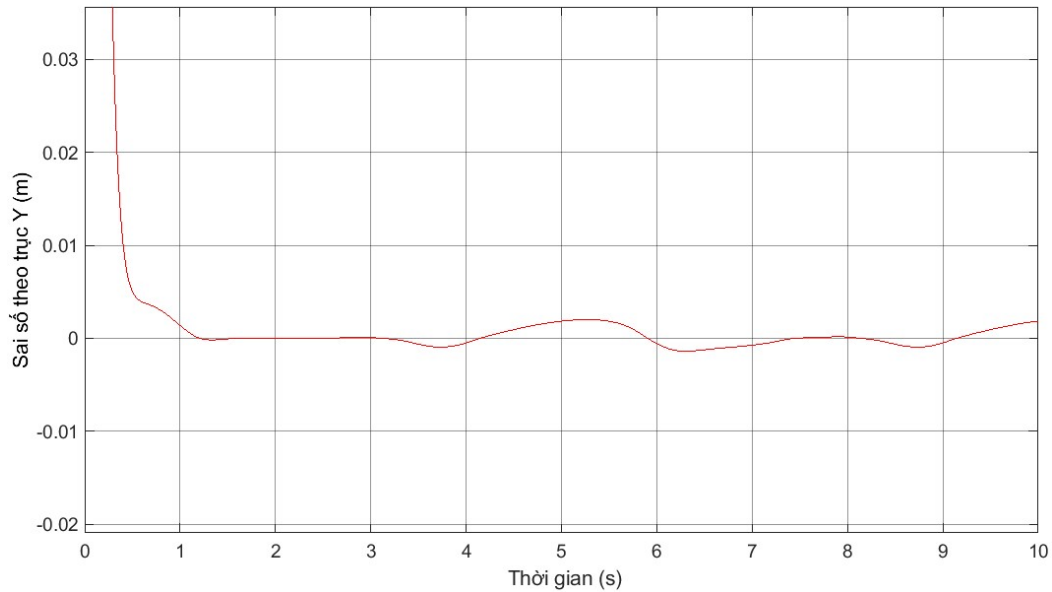




Hình 4.13 - Đồ thị tín hiệu điều khiển các khớp theo thuật toán STSOSMC
 (a) Tín hiệu điều khiển khớp 1; (b) Tín hiệu điều khiển khớp 2; (c) Phóng to của tín hiệu điều khiển khớp 1; (d) Phóng to của tín hiệu điều khiển khớp 2



Hình 4.14 - Đồ thị sai số quỹ đạo của thuật toán STSOSMC theo trục X



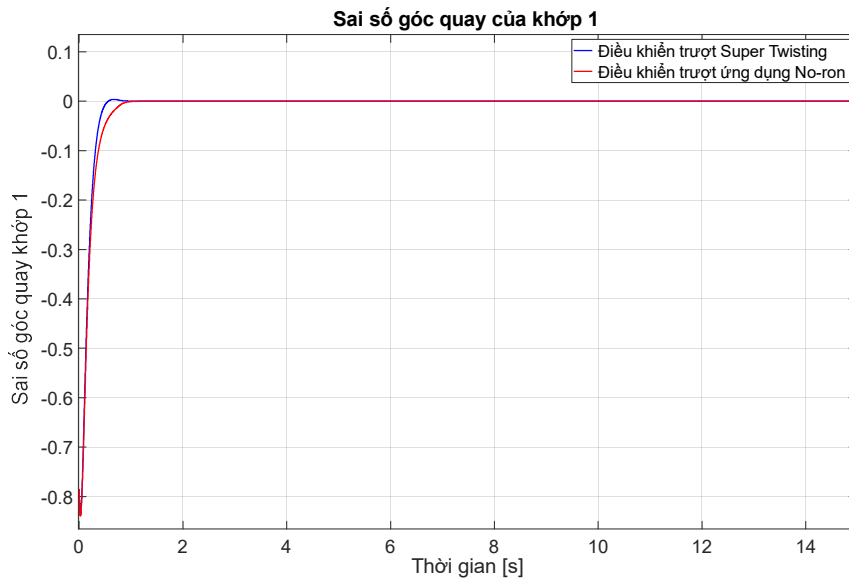
Hình 4.15 - Đồ thị sai số quỹ đạo của thuật toán STSOSMC theo trục Y

Nhận xét: Kết quả mô phỏng cho thấy trong trường hợp điều khiển tay máy robot sử dụng thuật toán điều khiển trượt bậc 2 siêu xoắn thì robot bám theo quỹ đạo rất chính xác, sai số ở cả 2 trục rất rất nhỏ. Hiện tượng đập mạch gần như đã được loại bỏ hoàn toàn.

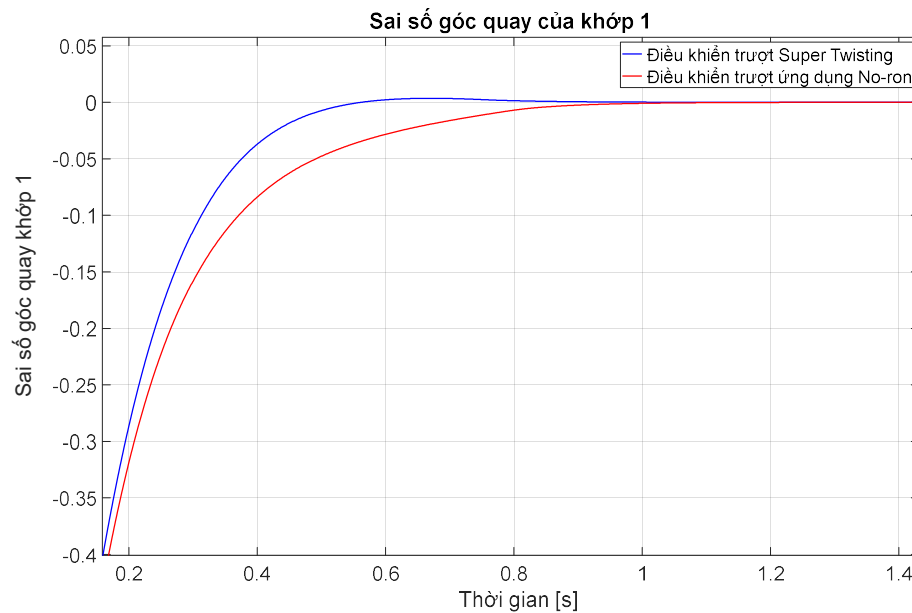
4.3. KẾT QUẢ MÔ PHỎNG THUẬT TOÁN ĐIỀU KHIỂN TRƯỢT BẬC CAO THÍCH NGHI ỨNG DỤNG MẠNG NƠ-RON

Trong mục 4.2 ở trên, đề tài đã tiến hành so sánh kết quả đạt được khi sử dụng thuật toán điều khiển trượt bậc 2 xoắn và điều khiển trượt bậc 2 siêu xoắn (super-twisting). Qua đó có thể thấy được những ưu điểm của thuật toán điều khiển trượt bậc 2 siêu xoắn (super-twisting) khi thể hiện các đặc tính bám quỹ đạo tốt hơn. Tuy nhiên, các thành phần bất định như sai số mô hình, nhiễu loạn, biến thiên tham số trong quá trình hoạt động v.v. vẫn có thể tác động đến hệ thống và làm giảm độ chính xác điều khiển bám quỹ đạo. Chính vì vậy, trong đề tài đã đề xuất thuật toán trượt bậc 2 siêu xoắn (super-twisting) thích nghi ứng dụng thêm mạng nơ-ron nhân tạo để bù các thành phần bất định. Các kết quả mô phỏng kiểm chứng trên tay máy robot 2 bậc tự do đã được thực hiện và kết quả bởi các

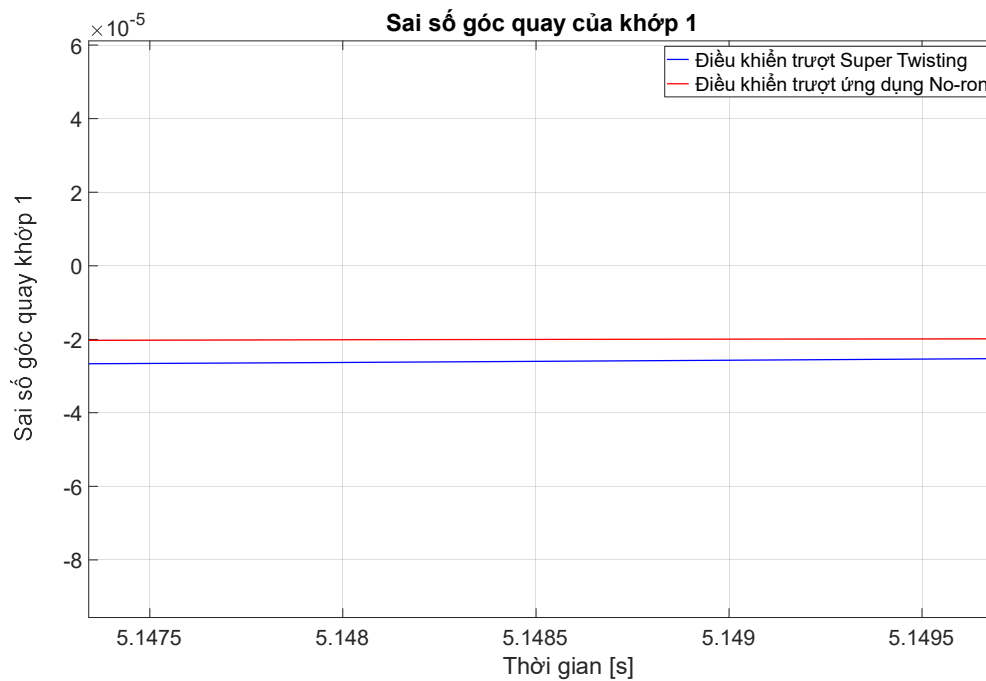
đồ thị phân tích như sau.



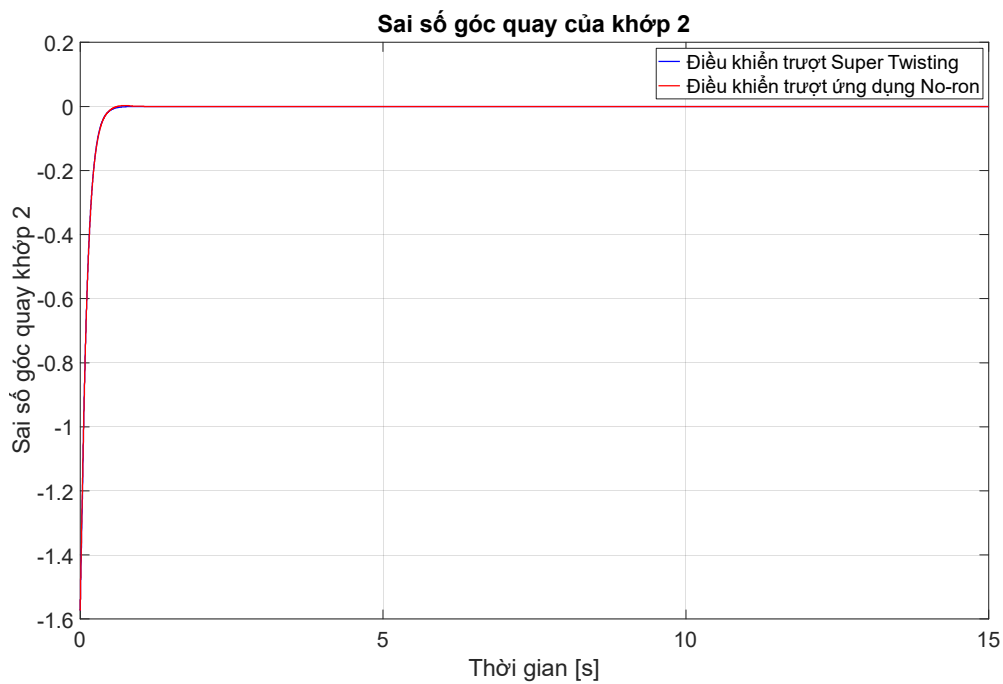
Hình 4.16 - So sánh sai số góc quay khớp 1 của tay máy robot



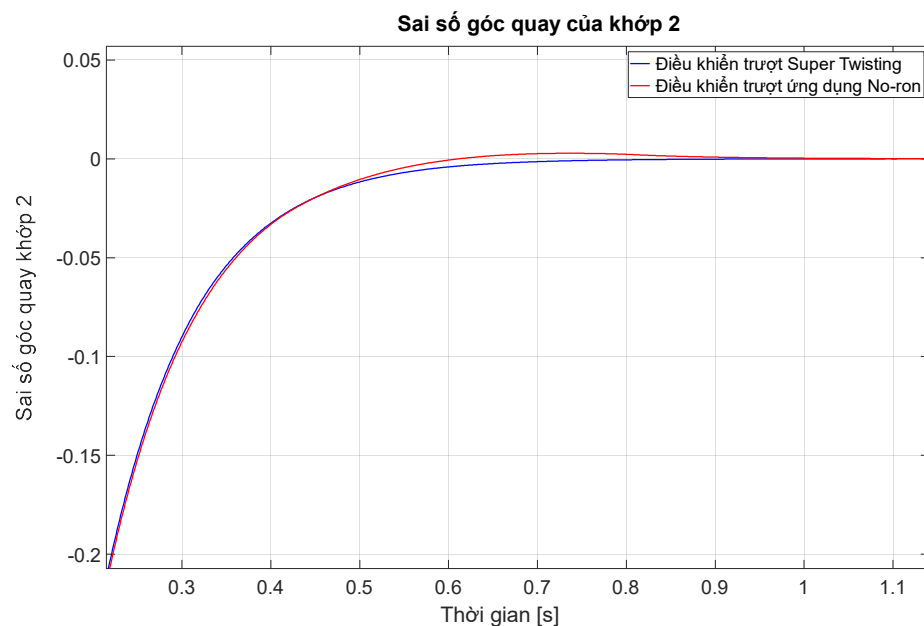
Hình 4.17 - Phóng to đồ thị so sánh sai số góc quay khớp 1 của tay máy robot giai đoạn đầu



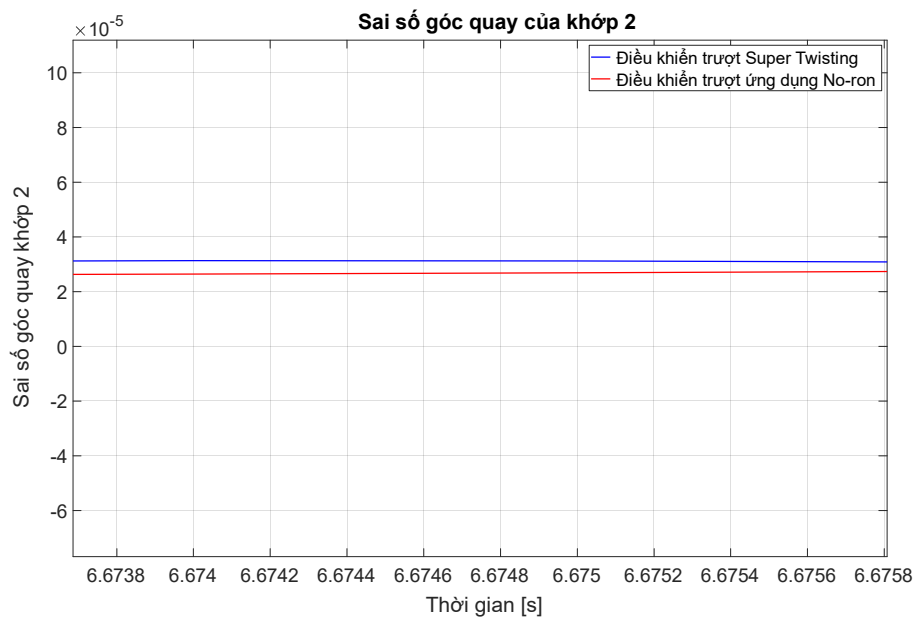
Hình 4.18 - Phóng to đồ thị so sánh sai số góc quay khớp 1 của tay máy robot giai đoạn xác lập



Hình 4.19 - So sánh sai số góc quay khớp 2 của tay máy robot



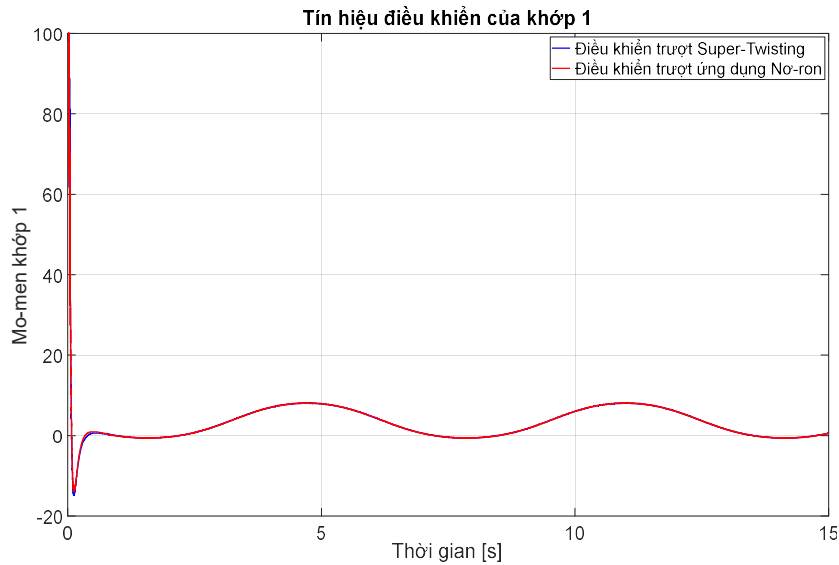
Hình 4.20 - Phóng to đồ thị so sánh sai số góc quay khớp 2 của tay máy robot giai đoạn đầu



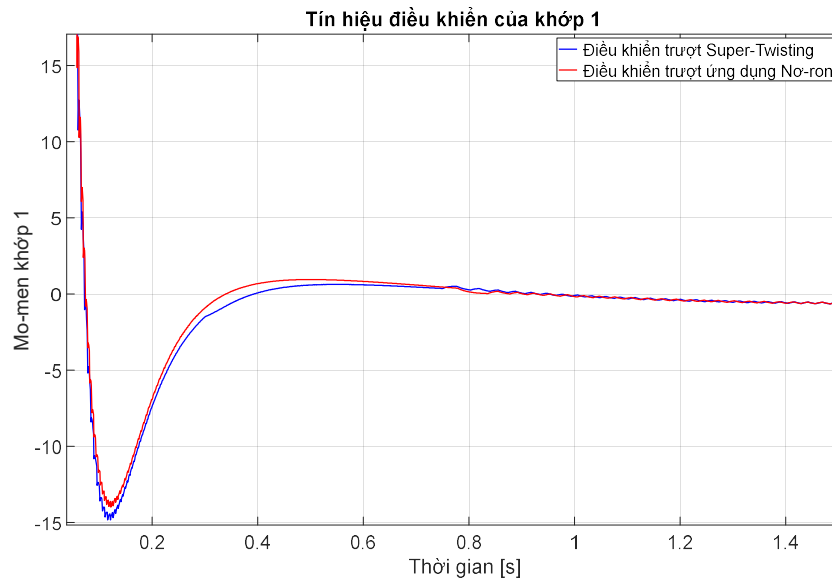
Hình 4.21 - Phóng to đồ thị so sánh sai số góc quay khớp 2 của tay máy robot giai đoạn xác lập

Từ các kết quả đồ thị thể hiện trên các hình từ Hình 4.16 đến Hình 4.21, so sánh kết quả đạt được giữa thuật toán điều khiển trượt bậc 2 siêu xoắn (super-

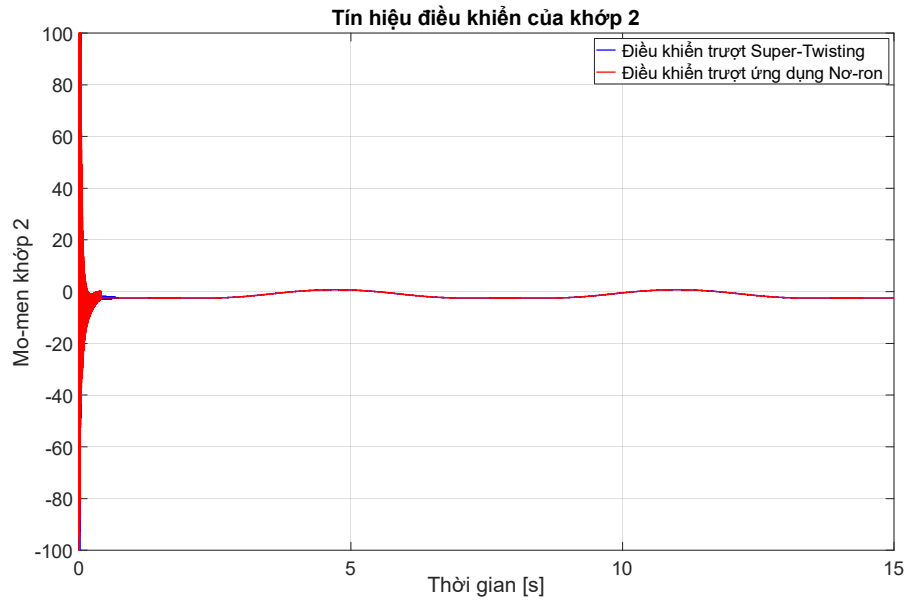
twisting) với thuật toán điều khiển trượt bậc 2 siêu xoắn thích nghi ứng dụng mạng nơ ron nhân tạo, có thể nhận thấy rằng thời gian quá độ của cả hai thuật toán điều khiển tương đương nhau. Nhưng sai số điều khiển bám quỹ đạo khi áp dụng thuật toán điều khiển trượt bậc 2 siêu xoắn thích nghi ứng dụng mạng nơ ron nhỏ hơn, độ chính xác cao hơn.



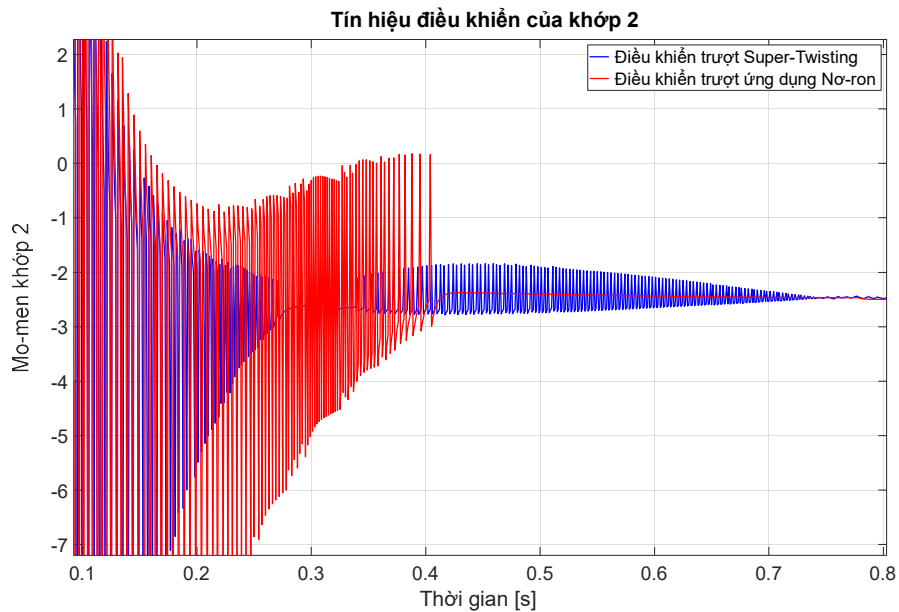
Hình 4.22- So sánh tín hiệu điều khiển khớp 1 của tay máy robot



Hình 4.23 - Phóng to tín hiệu điều khiển khớp 1 của tay máy robot



Hình 4.24 - So sánh tín hiệu điều khiển khớp 2 của tay máy robot



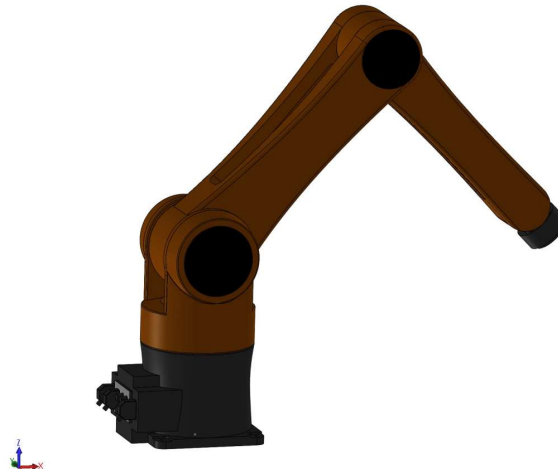
Hình 4.25 - Phóng to so sánh tín hiệu điều khiển khớp 2 của tay máy robot

Từ các đồ thị Hình 4.22, 4.23, 4.24 và 4.25 có thể thấy tín hiệu điều khiển của cả 2 thuật toán là gần như tương đương nhau. Tuy nhiên trường hợp sử dụng thuật toán điều khiển trượt bậc 2 siêu xoắn thích nghi ứng dụng mạng nơ ron nhân tạo có thời gian đạt trạng thái bão hòa nhanh hơn. Điều này thể hiện rõ tại tín hiệu điều khiển khớp 2 của tay máy robot. Lúc khởi động, thuật toán điều khiển trượt

bậc 2 siêu xoắn thích nghi dao động mạnh nhưng khoảng 0,4 giây là đã đạt trạng thái ổn định. Trong khi đó, thuật toán điều khiển trượt bậc 2 siêu xoắn (super-twisting) phải mất đến 0,7 giây thì mới đạt được trạng thái này.

4.4. KẾT QUẢ MÔ PHỎNG THUẬT TOÁN ĐIỀU KHIỂN TRƯỢT BẬC CAO VỚI BỘ QUAN SÁT THỜI GIAN HỮU HẠN

Để kiểm chứng sự hiệu quả của thuật toán điều khiển trượt bậc cao kết hợp bộ quan sát trượt bậc cao mà đề tài đã đề xuất ở chương 3, các mô phỏng kiểm nghiệm được thực hiện áp dụng lên một robot 3 bậc tự do có cấu tạo như Hình 4.26. Thiết kế chi tiết phần cơ khí của tay máy robot được thực hiện trên phần mềm SOLIDWORKS, sau đó mô hình cơ khí được tích hợp vào môi trường Simscape của Matlab và các thuật toán điều khiển được mô phỏng bằng Simulink.



Hình 4.26 - Mô phỏng của tay máy robot 3 bậc tự do trên phần mềm SOLIDWORKS.

Thông số của robot được mô phỏng như sau:

- + Khối lượng tương ứng của mỗi thanh: $m_1 = 33.429(kg)$, $m_2 = 34.129(kg)$, và $m_3 = 15.612(kg)$;
- + Chiều dài mỗi thanh là $l_1 = 0.25(m)$, $l_2 = 0.7(m)$, và $l_3 = 0.6(m)$;
- + Tọa độ trọng tâm tương ứng của mỗi thanh:

$$\begin{bmatrix} l_{c1x}, l_{c1y}, l_{c1z} \end{bmatrix}^T = \begin{bmatrix} 0, 0, -74.610 \times 10^{-3} \end{bmatrix}^T (m)$$

$$\begin{bmatrix} l_{c2x}, l_{c2y}, l_{c2z} \end{bmatrix}^T = \begin{bmatrix} 0.3477, 0, 0 \end{bmatrix}^T (m), \text{ và}$$

$$\begin{bmatrix} l_{c3x}, l_{c3y}, l_{c3z} \end{bmatrix}^T = \begin{bmatrix} 0.3142, 0, 0 \end{bmatrix}^T (m);$$

+ Mô-men quán tính:

$$\begin{bmatrix} I_{1xx}, I_{1yy}, I_{1zz} \end{bmatrix}^T = \begin{bmatrix} 0.7486, 0.5518, 0.5570 \end{bmatrix}^T (kg.m^2),$$

$$\begin{bmatrix} I_{2xx}, I_{2yy}, I_{2zz} \end{bmatrix}^T = \begin{bmatrix} 0.3080, 2.4655, 2.3938 \end{bmatrix}^T (kg.m^2), \text{ và}$$

$$\begin{bmatrix} I_{3xx}, I_{3yy}, I_{3zz} \end{bmatrix}^T = \begin{bmatrix} 0.0446, 0.7092, 0.7207 \end{bmatrix}^T (kg.m^2).$$

Khâu chấp hành cuối của robot được điều khiển theo quỹ đạo mong muốn như sau:

$$\begin{cases} x = 0.85 - 0.01t \\ y = 0.2 + 0.2 \sin(0.5t) \\ z = 0.7 + 0.2 \cos(0.5t) \end{cases} (m) \quad (4.1)$$

Ma sát và nhiễu bên ngoài được giả định như sau:

$$F_r(\dot{p}) + \tau_d(t) = \begin{bmatrix} 0.1 \operatorname{sgn}(\dot{p}_1) + 2\dot{p}_1 \\ 0.1 \operatorname{sgn}(\dot{p}_2) + 2\dot{p}_2 \\ 0.1 \operatorname{sgn}(\dot{p}_3) + 2\dot{p}_3 \end{bmatrix} + \begin{bmatrix} 4 \sin(t) \\ 5 \sin(t) \\ 6 \sin(t) \end{bmatrix} \quad (4.2)$$

Để so sánh và kiểm chứng hiệu quả, mô phỏng thực hiện thuật toán điều khiển mà đề tài đề xuất so sánh với các thuật toán điều khiển NFTSMC [17] và NFTSMC [20] có các phương trình cụ thể như sau:

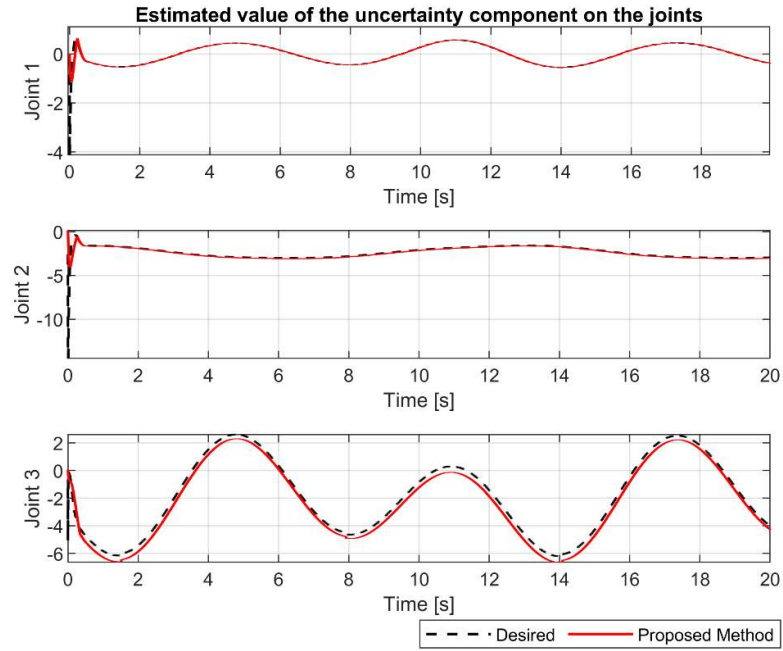
$$\begin{cases} s = \dot{x}_e + \alpha_5 x_e + \beta_5 \operatorname{sig}(x_e)^{\omega_5} \\ u_d = -\frac{1}{a(x)} \left(b_n(x) + \left(\alpha_5 + \beta_5 \omega_5 |x_e|^{\omega_5-1} \right) \dot{x}_e \right. \\ \left. + \Lambda_5 s + \left(\Delta_5^* + v_5 \right) \operatorname{sgn}(s) - \ddot{x}_r \right) \end{cases} \quad (4.3)$$

$$\begin{cases} s = \dot{x}_e + \frac{2\alpha_6}{1+e^{-\eta_6(|x_e|-\varepsilon_6)}} x_e + \frac{2\beta_6}{1+e^{\mu_6(|x_e|-\varepsilon_6)}} \text{sig}(x_e)^{\omega_6} \\ u_d = -\frac{1}{a(x)} \left(\begin{aligned} & b_n(x) + \frac{2\alpha_6}{1+e^{-\eta_6(|x_e|-\varepsilon_6)}} \dot{x}_e \\ & + \frac{2\alpha_6\eta_6\dot{x}_e \text{sgn}(x_e) e^{-\eta_6(|x_e|-\varepsilon_6)}}{\left(1+e^{-\eta_6(|x_e|-\varepsilon_6)}\right)^2} x_e \\ & + \frac{2\beta_6\omega_6}{1+e^{\mu_6(|x_e|-\varepsilon_6)}} |x_e|^{\omega_6-1} \dot{x}_e \\ & - \frac{2\beta_6\mu_6\dot{x}_e e^{\mu_6(|x_e|-\varepsilon_6)}}{\left(1+e^{\mu_6(|x_e|-\varepsilon_6)}\right)^2} |x_e|^{\omega_6} \\ & + \Lambda_6 s + (\Delta_6^* + v_6) \text{sgn}(s) - \ddot{x}_r \end{aligned} \right) \end{cases} \quad (4.4)$$

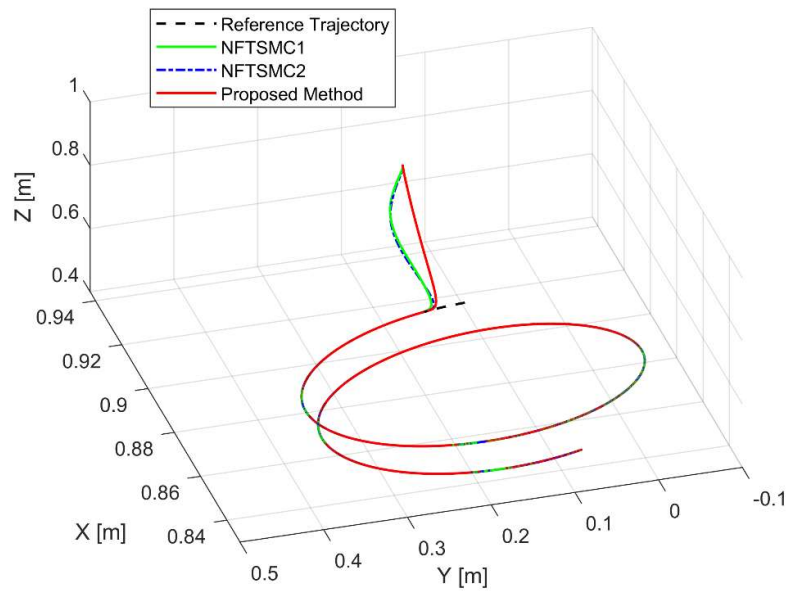
trong đó $x_e = x - x_r$, $\alpha_5, \alpha_6, \beta_5, \beta_6, \eta_6, \mu_6$ là những hằng số, $0 < \omega_5, \omega_6 < 1$, $\varepsilon_6 = (\beta_6/\alpha_6)^{1/(1-\omega_6)}$; $(\Delta_5^* + v_5)$ và $(\Delta_6^* + v_6)$ là những hằng số dương.

Bảng 4.1. Tham số của các thuật toán điều khiển

Thuật toán điều khiển	Tham số điều khiển
NFTSMC1	$\alpha_5 = 5, \beta_5 = 5, \omega_5 = 0.8, \Lambda_5 = 5, \Delta_5^* = 13, v_5 = 0.1$
NFTSMC2	$\alpha_6 = 5, \beta_6 = 5, \eta_6 = 0.9, \mu_6 = 1.2, \omega_6 = 0.8, \Lambda_6 = 5, \Delta_6^* = 13, v_6 = 0.1$
Thuật toán điều khiển trượt bậc cao kết hợp bộ quan sát trượt bậc cao mà đề tài đã đề xuất	$\alpha_1 = 5, \beta_1 = 5, \eta_1 = 0.9, \mu_1 = 1.2, \omega_1 = 0.8$
	$\alpha_2 = 5, \beta_2 = 5, \varphi_2 = 1.7, \lambda_2 = 0.6,$
	$\alpha_3 = 6, \beta_3 = 6, \eta_3 = 0.9, \mu_3 = 1.2, \omega_3 = 0.7$ $\alpha_4 = 6, \beta_4 = 6, \eta_4 = 0.9, \mu_4 = 1.2, \omega_4 = 0.7$



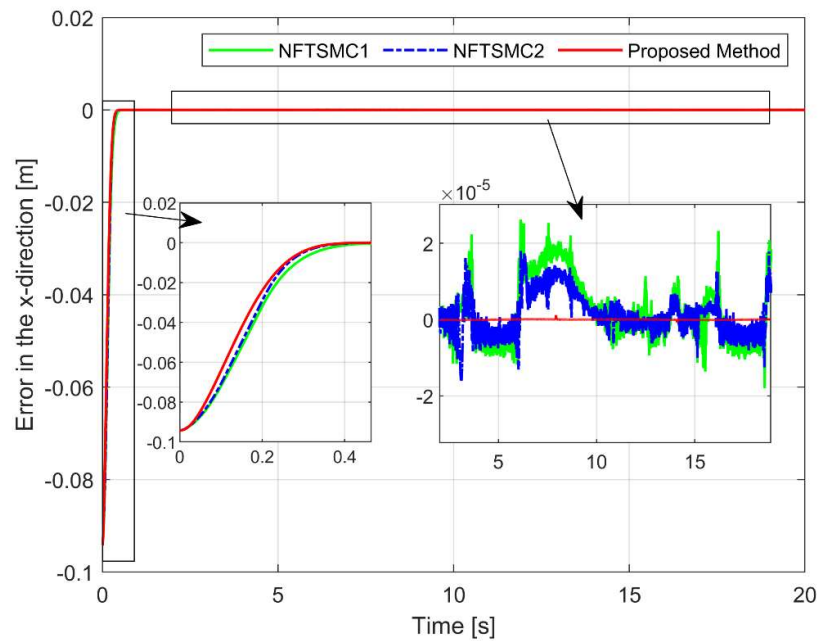
Hình 4.27 - Giá trị giả định của những thành phần bất định và kết quả xấp xỉ bởi bộ quan sát tại mỗi khớp.



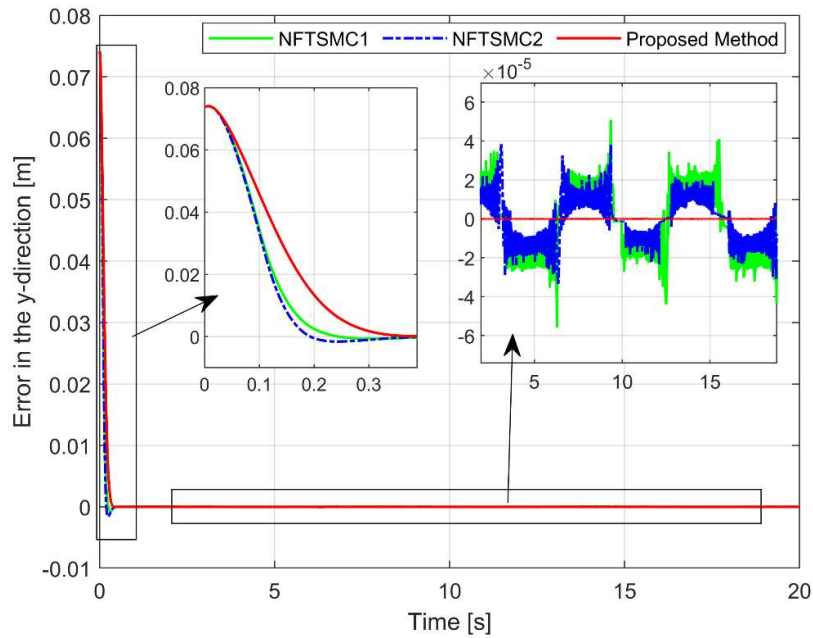
Hình 4.28 - So sánh kết quả bám quỹ đạo của khâu chấp hành cuối khi trong không gian 3 chiều (XYZ)

Đầu tiên, chúng ta phân tích sự hiệu quả của bộ quan sát được đề xuất. Với kết quả đồ thị mô phỏng được thể hiện trong Hình 2, chúng ta có thể thấy bộ quan sát đề xuất đã xấp xỉ chính xác các thành phần bất định.

Hình 4.28 biểu diễn kết quả mô phỏng của thuật toán điều khiển trượt đề xuất trong việc điều khiển bám quỹ của robot trên hệ trục tọa độ Đề-các XYZ. Kết quả đồ thị cho thấy quỹ đạo thực bám theo quỹ đạo mong muốn.

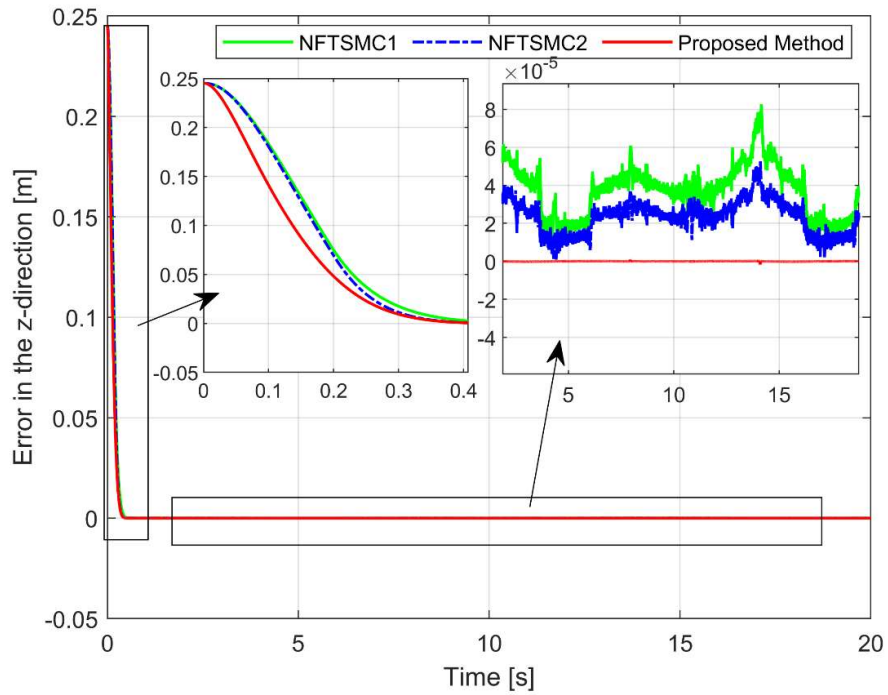


Hình 4.29 - Sai số điều khiển vị trí của khâu chấp hành cuối theo trục X

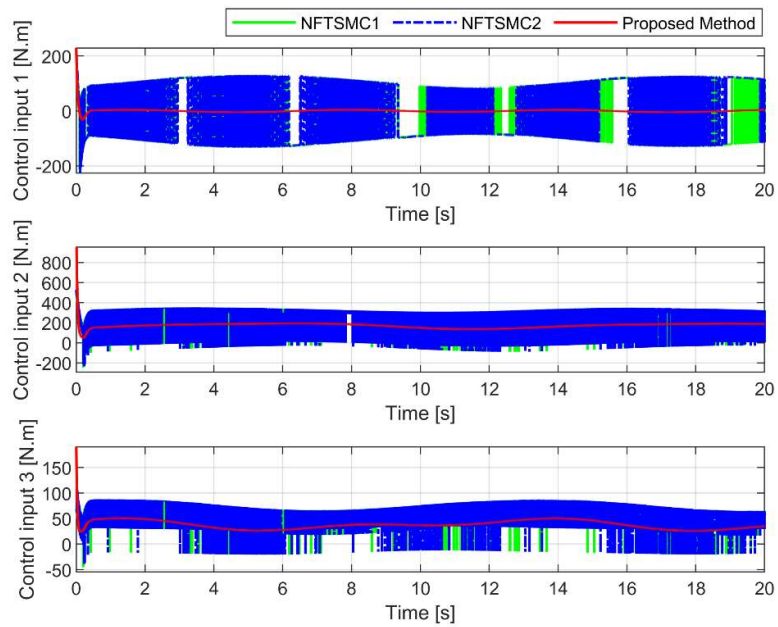


Hình 4.30 - Sai số điều khiển vị trí của khâu chấp hành cuối theo trục Y

Để đánh giá rõ hơn về độ chính xác, so sánh sai số bám quỹ đạo theo các trục X , Y , và Z được thể hiện trong các kết quả đồ thị Hình 4.29, Hình 4.30 và Hình 4.31. Có thể thấy rằng cả 3 trường hợp sử dụng các thuật toán điều khiển NFTSMC1, NFTSMC2 và thuật toán đề xuất đều cung cấp độ chính xác bám quỹ đạo cao với tốc độ hội tụ nhanh trong thời gian hữu hạn. Tuy nhiên, thuật toán NFTSMC2 có tốc độ hội tụ nhanh hơn và độ chính xác cao hơn một chút so với NFTSMC1. Đặc biệt, thuật toán điều khiển mà đề tài đề xuất có độ chính xác bám quỹ đạo cao nhất và tốc độ hội tụ nhanh nhất trong 3 thuật toán điều khiển.



Hình 4.31 - Sai số điều khiển vị trí của khâu chấp hành cuối theo trục Z



Hình 4.32 - Những tín hiệu điều khiển tại mỗi khớp

Các kết quả mô phỏng so sánh hiện tượng dao động trong tín hiệu điều khiển (chattering) giữa 3 thuật toán điều khiển được thể hiện trong Hình 4.32. Kết quả

cho thấy hiện tượng chattering xuất hiện trong cả 2 thuật toán điều khiển NFTSMC1 và NFTSMC2 là khá lớn do trong mô phỏng giả định nhiều thành phần bất định. Tuy nhiên, đối với trường hợp sử dụng thuật toán điều khiển của đề tài đề xuất thì hiện tượng chattering xuất hiện không đáng kể. Kết quả trường hợp sử dụng thuật toán điều khiển do đề tài đề xuất cho các tín hiệu điều khiển mượt mà, ít sự dao động.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN ĐỀ TÀI

KẾT LUẬN

Đề tài đã hoàn thành nhiệm vụ đặt ra. Đề tài đã nghiên cứu, đánh giá và so sánh ưu nhược điểm của các phương pháp điều khiển trượt bậc cao cho tay máy robot công nghiệp. Trong đề tài đã đề xuất sử dụng mạng nơ-ron nhân tạo để bù thích nghi cho các thành phần bất định trong thuật toán điều khiển trượt bậc 2 siêu xoắn cho tay máy robot.

Đồng thời, đóng góp chính của đề tài là đã nghiên cứu, đề xuất một phương pháp điều khiển bền vững mới sử dụng sự kết hợp giữa một bộ quan sát trượt bậc cao thời gian cố định (FxTHOSMO) để ước tính toàn bộ các thành phần bất định mà không cần biết trước các thông tin về chúng, và một thuật toán điều khiển trượt đầu cuối thời gian hữu hạn (FnTTSMC). Thuật toán FnTTSMC được thiết kế dựa trên phương trình mặt trượt đầu cuối nhanh toàn cục thời gian hữu hạn và luật điều khiển tiếp cận nhanh để thiết lập sự hội tụ trong thời gian hữu hạn toàn cục của hệ thống. Hệ thống điều khiển tích hợp có các đặc tính ưu việt như độ chính xác bám quỹ đạo cao, hội tụ trong thời gian hữu hạn và bền vững với những ảnh hưởng của các thành phần bất định. Các phân tích với lý thuyết Lyapunov đã chứng minh tính ổn định trong thời gian hữu hạn và sự hội tụ của hệ thống điều khiển. Qua các kết quả mô phỏng ứng dụng cho trường hợp cụ thể với tay máy robot 3 bậc tự do đã cho thấy thuật toán điều khiển mà bài đề xuất mang lại hiệu quả tốt và chất lượng điều khiển cao hơn so với các trường hợp sử dụng các thuật toán điều khiển của các công trình nghiên cứu đi trước ở cả ba khía cạnh bao gồm độ chính xác bám quỹ đạo, hội tụ nhanh trong thời gian hữu hạn và giảm hiện tượng chattering.

HƯỚNG PHÁT TRIỂN ĐỀ TÀI

Với xu hướng phát triển mới về sử dụng tay máy robot trong nhiều lĩnh vực sản xuất, gia công cơ khí và cả dân dụng như hiện nay thì các tay máy robot đã và đang được ứng dụng rất phổ biến trong công nghiệp cũng như trong dân dụng. Bên cạnh đó, các ứng dụng trí tuệ nhân tạo, máy học vào trong điều khiển tay

máy robot cũng đang được triển khai mạnh mẽ trong cuộc cách mạng công nghiệp 4.0. Do đó, hướng nghiên cứu tiếp theo của đề tài sẽ tập trung vào việc áp dụng các phương pháp điều khiển thông minh, mạng nơ-ron nhân tạo với các thuật toán chỉnh định online để nâng cao chất lượng hoạt động và độ chính xác của tay máy robot. Nhóm tác giả cũng sẽ tìm kiếm các nguồn kinh phí để chế tạo hệ thống thật, thử nghiệm các bộ điều khiển đề xuất trên hệ thống thật.

TÀI LIỆU THAM KHẢO

- [1] Le Tien Dung, Kang Hee-Jun, and Young-Shick Ro. "Robot manipulator modeling in Matlab-SimMechanics with PD control and online gravity compensation." *International Forum on Strategic Technology 2010*. IEEE, 2010.
- [2] Le, Tien Dung, et al. "An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-dof parallel manipulator." *Neurocomputing* 116 (2013): 53-61.
- [3] Zhu, Xiaocong, et al. "Adaptive robust posture control of a parallel manipulator driven by pneumatic muscles." *Automatica* 44.9 (2008): 2248-2257.
- [4] Garagić, Denis, and Krishnaswamy Srinivasan. "Adaptive Friction Compensation for Stewart Platform Machine Tool Structures." *IFAC Proceedings Volumes* 35.2 (2002): 555-562.
- [5] Islam, Shafiqul, and Xiaoping P. Liu. "Robust sliding mode control for robot manipulators." *IEEE Transactions on industrial electronics* 58.6 (2010): 2444-2453.
- [6] Su, C-Y., and T-P. Leung. "A sliding mode controller with bound estimation for robot manipulators." *IEEE Transactions on Robotics and Automation* 9.2 (1993): 208-214.
- [7] Capisani, Luca Massimiliano, Antonella Ferrara, and Lorenza Magnani. "Design and experimental validation of a second-order sliding-mode motion controller for robot manipulators." *International Journal of Control* 82.2 (2009): 365-377.
- [8] Nguyen, Van-Cuong, Anh-Tuan Vo, and Hee-Jun Kang. "Continuous PID sliding mode control based on neural third order sliding mode observer for robotic manipulators." *International conference on intelligent computing*. Springer, Cham, 2019.

- [9] Romdhane, Neila Mezghani Ben, and Tarak Damak. "Higher order sliding mode control of uncertain robot manipulators." *Advances and applications in sliding mode control systems*. Springer, Cham, 2015. 327-352.
- [10] N. Derbel, J. Ghommam, and Q. Zhu, *Applications of sliding mode control*, vol. 79. Springer, 2017.
- [11] S. Kamal, J. A. Moreno, A. Chalanga, B. Bandyopadhyay, and L. M. Fridman, "Continuous terminal sliding-mode controller," *Automatica*, 2016.
- [12] L. Hui and J. Li, "Terminal sliding mode control for spacecraft formation flying," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 3, pp. 835–846, 2009.
- [13] M. Chen, Q.-X. Wu, and R.-X. Cui, "Terminal sliding mode tracking control for a class of SISO uncertain nonlinear systems," *ISA Trans.*, vol. 52, no. 2, pp. 198–206, 2013.
- [14] Y. Feng, X. Yu, and Z. Man, "Non-singular terminal sliding mode control of rigid manipulators," *Automatica*, vol. 38, no. 12, pp. 2159–2167, 2002.
- [15] V. A. Tuan and H.-J. Kang, "A New Finite-time Control Solution to The Robotic Manipulators Based on The Nonsingular Fast Terminal Sliding Variables and Adaptive Super-Twisting Scheme," *J. Comput. Nonlinear Dyn.*, 2018.
- [16] S. Yu, X. Yu, B. Shirinzadeh, and Z. Man, "Continuous finite-time control for robotic manipulators with terminal sliding mode," *Automatica*, vol. 41, no. 11, pp. 1957–1964, 2005.
- [17] X. Yu and M. Zhihong, "Fast terminal sliding-mode control design for nonlinear dynamical systems," *Circuits Syst. I Fundam. Theory ...*, vol. 49, no. 2, pp. 261–264, 2002.
- [18] C. U. Solis, J. B. Clempner, and A. S. Poznyak, "Fast terminal sliding-mode control with an integral filter applied to a Van Der Pol oscillator," *IEEE Trans. Ind. Electron.*, vol. 64, no. 7, pp. 5622–5628, 2017.
- [19] S. Amirkhani, S. Mobayen, N. Iliaee, O. Boubaker, and S. H. Hosseinnia, "Fast terminal sliding mode tracking control of nonlinear uncertain mass--spring system with experimental verifications," *Int. J. Adv. Robot. Syst.*, vol. 16, no. 1, p. 1729881419828176, 2019.
- [20] H. Pan, G. Zhang, H. Ouyang, and L. Mei, "A Novel Global Fast Terminal Sliding Mode Control Scheme For Second-Order Systems," *IEEE Access*, 2020.

- [21] S. Yu, G. Guo, Z. Ma, and J. Du, "Global fast terminal sliding mode control for robotic manipulators," *Int. J. Model. Identif. Control*, vol. 1, no. 1, pp. 72–79, 2006.
- [22] S. Mobayen, F. Tchier, and L. Ragoub, "Design of an adaptive tracker for n-link rigid robotic manipulators based on super-twisting global nonlinear sliding mode control," *Int. J. Syst. Sci.*, vol. 48, no. 9, pp. 1990–2002, 2017.
- [23] Y. Wang, K. Zhu, F. Yan, and B. Chen, "Adaptive super-twisting nonsingular fast terminal sliding mode control for cable-driven manipulators using time-delay estimation," *Adv. Eng. Softw.*, vol. 128, pp. 113–124, 2019.
- [24] J. Fei and W. Yan, "Adaptive control of MEMS gyroscope using global fast terminal sliding mode control and fuzzy-neural-network," *Nonlinear Dyn.*, vol. 78, no. 1, pp. 103–116, 2014.
- [25] X.-G. Yan and C. Edwards, "Nonlinear robust fault reconstruction and estimation using a sliding mode observer," *Automatica*, vol. 43, no. 9, pp. 1605–1614, 2007.
- [26] Q. Shen, B. Jiang, and V. Cocquempot, "Adaptive fuzzy observer-based active fault-tolerant dynamic surface control for a class of nonlinear systems with actuator faults," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 2, pp. 338–349, 2014.
- [27] J.-H. Park, S.-H. Kim, and T.-S. Park, "Approximation-free output-feedback control of uncertain nonlinear systems using higher-order sliding mode observer," *J. Dyn. Syst. Meas. Control*, vol. 140, no. 12, 2018.
- [28] J.-H. Park, S.-H. Kim, and T.-S. Park, "Output-feedback adaptive neural controller for uncertain pure-feedback nonlinear systems using a high-order sliding mode observer," *IEEE Trans. neural networks Learn. Syst.*, vol. 30, no. 5, pp. 1596–1601, 2018.
- [29] Y. Yin *et al.*, "Observer-based adaptive sliding mode control of NPC converters: An RBF neural network approach," *IEEE Trans. Power Electron.*, vol. 34, no. 4, pp. 3831–3841, 2018.
- [30] H. K. Khalil, "Cascade high-gain observers in output feedback control," *Automatica*, vol. 80, pp. 110–118, 2017.
- [31] S. Hwang and H. S. Kim, "Extended disturbance observer-based integral sliding mode control for nonlinear system via T--S fuzzy model," *IEEE Access*, vol. 8, pp. 116090–116105, 2020.
- [32] Z. Zuo, "Non-singular fixed-time terminal sliding mode control of nonlinear systems," *IET Control theory Appl.*, vol. 9, no. 4, pp. 545–552, 2014.

- [33] C. Edwards and Y. Shtessel, “Adaptive dual-layer super-twisting control and observation,” *Int. J. Control*, vol. 89, no. 9, pp. 1759–1766, 2016.
- [34] H. Pan, G. Zhang, H. Ouyang, and L. Mei, “Novel Fixed-Time Nonsingular Fast Terminal Sliding Mode Control for Second-Order Uncertain Systems Based on Adaptive Disturbance Observer,” *IEEE Access*, vol. 8, pp. 126615–126627, 2020.
- [35] V. Utkin and J. Y. Gulder, “Shi j.,“,” *Sliding Mode Control Electromechanical Syst. Teylor Fr.*, 1999.
- [36] S. B. Niku, *Introduction to robotics: analysis, control, applications*. John Wiley & Sons, 2020.

PHỤ LỤC

Phụ lục 1 – Bản photo thuyết minh của đề tài

Phụ lục 2 - Các sản phẩm khoa học của đề tài

Phụ lục 3 – Sản phẩm đào tạo

Phụ lục 4 – Sản phẩm ứng dụng

PHỤ LỤC 1

BẢN PHOTO THUYẾT MINH CỦA ĐỀ TÀI

PHỤ LỤC 2

CÁC SẢN PHẨM KHOA HỌC CỦA ĐỀ TÀI

1. 01 bài báo quốc tế thuộc danh mục SCIE:

[1] Doan, Quang Vinh, Vo Anh Tuan, Hee-Jun Kang, Le Tien Dung, and Nguyen Ngoc Hoai An, "A novel fast terminal sliding mode tracking control methodology for robot manipulators." *Applied Sciences* 10.9 (2020): 3010.

2. 02 bài báo trên tạp chí trong nước:

[1] Lê Tiến Dũng, Võ Anh Tuấn, và Nguyễn Ngọc Hoài Ân, "Phương pháp điều khiển bền vững mới cho tay máy robot công nghiệp sử dụng kết hợp bộ quan sát trượt bậc cao thời gian cố định với thuật toán điều khiển trượt." *Tạp chí Khoa học và Công nghệ - Đại học Đà Nẵng*, Vol. 19, No. 5.1, Tr. 22-28, 2021.

[2] Lê Tiến Dũng, Lê Phước Lộc, Đồng Hoàng Bin, Hà Ngọc Chính, Nguyễn Quốc Huy, và Huỳnh Ngọc Khoa, “Ứng dụng mạng nơ-ron nhân tạo trong chẩn đoán lỗi cho máy biến áp lực tại truyền tải điện kon tum – công ty truyền tải điện 2”, *Tạp chí Khoa học và Công nghệ - Đại học Đà Nẵng*, Vol. 18, No. 7, Tr. 84-89, 2020.

PHỤ LỤC 3

SẢN PHẨM ĐÀO TẠO

Đào tạo 01 Thạc sỹ bảo vệ thành công luận văn tốt nghiệp:

Phan Quốc Việt, lớp K37.TĐH, chuyên ngành Kỹ thuật điều khiển và Tự động hóa. Tên đề tài luận văn: “*Nghiên cứu thiết kế thuật toán điều khiển trượt bậc 2 thích nghi cho tay máy robot ứng dụng mạng nơ-ron nhân tạo*” dưới sự hướng dẫn của PGS. TS. Lê Tiên Dũng (Quyết định giao đề tài số 333/QĐ-ĐHBK-ĐT ngày 02/03/2020, quyết định bảo vệ số /QĐ-ĐHBK-ĐT, Bằng tốt nghiệp Thạc sĩ số hiệu DND.7.0015411 cấp ngày 03/02/2021).

PHỤ LỤC 4

SẢN PHẨM ỨNG DỤNG

File mềm chương trình máy tính mô phỏng hệ thống với thuật toán điều khiển tay máy robot công nghiệp mà đề tài đề xuất.