НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
ноябрь–декабрь 2023 Том 23 № 6 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
November–December 2023 Vol. 23 No 6 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

doi: 10.17586/2226-1494-2023-23-6-1096-1105

Dynamic surface control for omnidirectional mobile robot with full state constrains and input saturation

Chen Zhiqiang[1]✉, Aleksandr Yu. Krasnov[2], Liao Duzhesheng[3], Yang Qiusheng[4]

[1,2,3,4] ITMO University, Saint Petersburg, 197101, Russian Federation

[1] Snowchen612@outlook.com✉, https://orcid.org/0009-0007-6813-2287
[2] aykrasnov@itmo.ru, https://orcid.org/0000-0001-6026-6706
[3] ldzs2015@gmail.com, https://orcid.org/0009-0000-6389-1355
[4] 1799797481@qq.com, https://orcid.org/0009-0007-5839-0258

Abstract

In this paper, we study the trajectory tracking problem of a three-wheeled omnidirectional mobile robot with full state constraints and actuator saturation. Firstly, we analyze a three-wheeled omnidirectional mobile robot and give control model with actuator saturation. By using tan-type Barrier Lyapunov Function and backstepping method, kinematic and dynamic controllers are built, which can ensure that the system full states will not violate the given constraints when the robot is performing trajectory tracking. Then, considering the differential explosion problem which occurs when solving the derivatives of the virtual control law, we use a second-order differential sliding mode surface to calculate it, so as to reduce the complexity of the operation. In addition, due to the output saturation problem of the robot drive motor, an auxiliary compensation system is adopted to compensate for the error generated by the saturation function. Finally, an experimental simulation is performed in MATLAB and the simulation results illustrate the effectiveness of the control algorithm proposed in this paper.

Keywords

full state constrains, barrier Lyapunov function, input saturation, omnidirectional mobile robot, dynamic surface control, backstepping method

For citation: Zhiqiang C., Krasnov A.Yu., Duzhesheng L., Qiusheng Y. Dynamic surface control for omnidirectional mobile robot with full state constrains and input saturation. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2023, vol. 23, no. 6, pp. 1096–1105. doi: 10.17586/2226-1494-2023-23-6-1096-1105

УДК 62-50

Динамическое поверхностное управление всенаправленным мобильным роботом с полными ограничениями состояния и насыщением входа

Чэнь Чжицян[1]✉, Александр Юрьевич Краснов[2], Ляо Дучжэшэн[3], Ян Цюшэн[4]

[1,2,3,4] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация

[1] Snowchen612@outlook.com✉, https://orcid.org/0009-0007-6813-2287
[2] aykrasnov@itmo.ru, https://orcid.org/0000-0001-6026-6706
[3] ldzs2015@gmail.com, https://orcid.org/0009-0000-6389-1355
[4] 1799797481@qq.com, https://orcid.org/0009-0007-5839-0258

Аннотация

Введение. Исследована задача управления траекторией движения всенаправленного мобильного робота с полными ограничениями на состояние и насыщением входного сигнала. При движении робота в узком пространстве чрезмерные ошибки отслеживания траектории и скорости могут привести к столкновению. На практике явление насыщения входного сигнала двигателя может привести к тому, что контроллер не сможет достичь требуемых характеристик слежения. По этой причине при проектировании контроллера важно ограничить вектор состояния робота и компенсировать ошибку момента, которая возникает из-за насыщения привода. **Метод.** С помощью барьерной функции Ляпунова и метода backstepping проектируется виртуальный

регулятор и регулятор динамики, которые обеспечивают стабилизацию состояния системы в заданной области ограничений при движении робота по траектории. Производная виртуального закона управления рассчитывается методом динамических поверхностей, что снижает вычислительную сложность. Для устранения влияния неопределенности параметров на движение робота и оценки неизвестных частей его модели применены нейронные сети. Для компенсации ошибок, возникающих при насыщении исполнительных механизмов, создана вспомогательная система. **Основные результаты.** Имитационный эксперимент выполнен в пакете прикладных программ MATLAB. Экспериментальные исследования показали, что разработанный алгоритм управления может реализовать точное траекторное движение робота в условиях ограничения состояний системы и насыщения входного сигнала в исполнительных механизмах. **Обсуждение.** Метод может быть применен для решения задачи управления мобильным роботом в ограниченном пространстве. Аналогичным образом он может быть применен ко всем роботам с аналогичной математической моделью.

Ключевые слова

ограничения полного состояния, барьерная функция Ляпунова, насыщение входа, всенаправленный мобильный робот, динамическое управление поверхностью, метод backstepping

Ссылка для цитирования: Чжицян Ч., Краснов А.Ю., Дучжэшэн Л., Цюшэн Я. Динамическое поверхностное управление всенаправленным мобильным роботом с полными ограничениями состояния и насыщением входа // Научно-технический вестник информационных технологий, механики и оптики. 2023. Т. 23, № 6. С. 1096–1105 (на англ. яз.). doi: 10.17586/2226-1494-2023-23-6-1096-1105

Introduction

In recent years, with the rapid development of robotics technology, mobile robots have been widely used in various fields, such as logistics and warehousing, factory manufacturing, military field, and space exploration. In comparison with differential drive robots, omnidirectional mobile robots do not have the problem of incomplete constraints and are able to realize unconstrained plane motion [1]. At present, many scholars have studied omnidirectional mobile robots [2–4]. Watanabe [2] studied the omnidirectional mobile robotic robot and designed the controller using the feedback control method. Tamás [3] studied the four-wheeled omnidirectional mobile robot, established the kinematic and dynamic models, and proposed the trajectory planning method and trajectory tracking control algorithm; Liu [4] investigated the three-wheeled omni-directional mobile robot and designed the controller based on the dynamics model by using the trajectory linearization method. In actuality, however, there is uncertainty in the robot model due to manufacturing errors and disturbances in external environmental factors. The control methods proposed by the above scholars do not take into account the effects of parameter uncertainty and external disturbances, making it difficult to realize the desired control performance in practical experiments.

On the trajectory tracking problem, many scholars have proposed mature control plans (e.g., adaptive control, sliding mode control, neural network control, model predictive control, and fuzzy control). Considering the problems of wheel slippage and model parameter uncertainty, Huang [5] proposed an adaptive backstepping control method. Alakshendra [6] considered the effects of parameter uncertainty and external perturbation, proposed an adaptive sliding mode control method by combining sliding mode control with adaptive control which achieved good performance. Lu [7] designed an adaptive neural network sliding mode control scheme by using neural networks to approximate the uncertain part of the model. Zijie [8] used a fuzzy neural network to adjust parameters of the control gain for reducing the chattering phenomenon in the sliding mode control. In practice, however, it is often necessary to consider the safety of robot usage. According to the environment and the size of the space in which the robot operates, limits are set on the tracking error and the velocity of the motion. In addition, due to the limitation of hardware performance, the output torque of the controller may exceed the maximum output of the motor resulting the input saturation problem and affecting the trajectory tracking accuracy. Therefore, the input saturation problem also must be considered.

The Barrier Lyapunov Function (BLF) is a frequently used method for the tracking control problem with state constraints. Tee [9] proposed an adaptive control method by combining the backstepping method with the Barrier Lyapunov Function. Xi [10] improved the Lyapunov barrier function and used a radial basis neural network to approximate the unknown part of the robot model, ensuring changing in state time-varying constraints. Ding [11] used neural networks and Lyapunov barrier functions in the trajectory tracking problem of a two-wheeled differential mobile robot and achieved good tracking performance. Dong [12] created a finite time tracking controller to ensure that the state of the system is stabilized in a certain range for a finite period of time.

Saturation compensation is a common approach for actuator input saturation problem. Doyle [13] proposed a control scheme for input saturation by replacing the saturation function with an inverse tangent function, and proved the stability of the controller by using the backstepping method. Mofid [14] created an adaptive robust controller by considering the effects of input saturation and external perturbations on the system. Chen [15] limited the speed to a diamond-shaped range, achieving tracking control in the case of input saturation. By designing an auxiliary system to compensate for the nonlinear term generated by input saturation, Yang [16] had achieved state constrains and input saturation control of a nonlinear system.

Inspired by the above literature, this paper investigates the trajectory tracking problem of a three wheeled omnidirectional mobile robot with full state constraints and actuator saturation. The details of the study are as follows:

— For the problem of full-state constraints. By combining the tan-type Lyapunov barrier function and the

backstepping method, kinematics and dynamics controllers were designed. They ensure that all the states of a three-wheeled omnidirectional mobile robot in trajectory tracking motion are within the constraints.
— In order to avoid the differential explosion problem, the derivative of the virtual control law is calculated using the Dynamic Surface Control technique.
— According to reference [17], an auxiliary system is used to compensate for the nonlinear disturbance generated by input saturation.
— Simulation experiments were carried out using MATLAB. The experimental results show the effectiveness of the proposed algorithm in this paper.

Model of Robot

The three-wheeled omnidirectional mobile robot is shown in Fig 1. $\{x, o, y\}$ is the world coordinate system and $\{x_r, y_r, \theta_r\}$ is the robot coordinate system; o_r is the mass center of the robot; L is the distance from the robot's mass center to the driving wheel; I_V is the robot's moment of inertia; $\mathbf{q} = (x, y, \theta)^T$ is the robot's posture; (x, y) is the coordinates of the robot's mass center in the world coordinate system; and θ is the angle between the robot's coordinate system and the world coordinate system. The robot is supported by three omnidirectional wheels which allow longitudinal and lateral movements as well as rotational movements on a plane surface. (f_1, f_2, f_3) is the driving force output from the drive wheels.

The kinematic model of the robot is shown in equation

$$\dot{\mathbf{q}} = \mathbf{S}(\mathbf{q})\mathbf{V} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix}, \qquad (1)$$

where $\dot{\mathbf{q}} = (\dot{x}, \dot{y}, \dot{\theta})^T$ is the velocity and angular velocity of the robot's mass center in the world coordinate system; $\mathbf{S}(\mathbf{q}) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$; $\mathbf{V} = (v_x, v_y, \omega)^T$ is the speed and angular velocity of the robot's mass center in the robot's coordinate system.

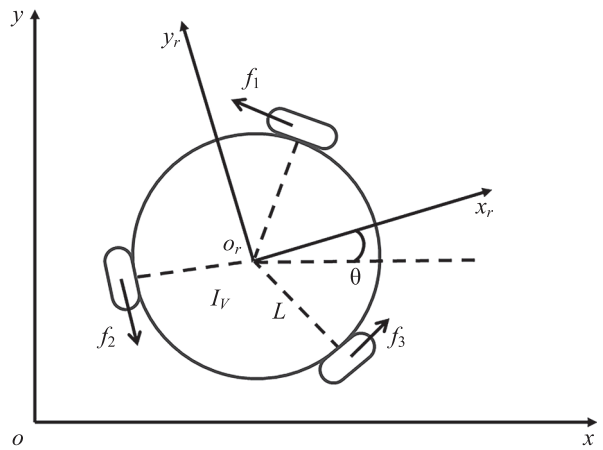

Fig. 1. Three-wheeled omnidirectional mobile robot

The relationship between the velocity of the robot's mass center and the angular velocities of the three driving wheels is shown in the equation

$$\mathbf{V} = \mathbf{A}\mathbf{w} = \begin{pmatrix} -\frac{r}{2} & -\frac{r}{2} & r \\ \frac{\sqrt{3}r}{2} & -\frac{\sqrt{3}r}{2} & 0 \\ \frac{r}{L} & \frac{r}{L} & \frac{r}{L} \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix}, \qquad (2)$$

where $\mathbf{A} = \begin{pmatrix} -\frac{r}{2} & -\frac{r}{2} & r \\ \frac{\sqrt{3}r}{2} & -\frac{\sqrt{3}r}{2} & 0 \\ \frac{r}{L} & \frac{r}{L} & \frac{r}{L} \end{pmatrix}$; $\mathbf{w} = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix}$ is the angular velocities of the three driving wheels; r is the radius of driving wheels.

According to reference [4], the force on a three-wheeled omnidirectional mobile robot can be expressed as follows

$$\begin{cases} m(\dot{v}_x - v_y\dot{\theta}) = -\frac{f_1}{2} - \frac{f_2}{2} + f_3 \\ m(\dot{v}_y - v_x\dot{\theta}) = \frac{\sqrt{3}}{2}(f_1 - f_2) \\ I_V\ddot{\theta} = L(f_1 + f_2 + f_3) \end{cases}, \qquad (3)$$

where m is the mass of the robot.

By using the direct current motor equation, the relationship between the output torque of the driving wheel motor and the driving force can be derived as

$$I_w\dot{\omega}_i + c\omega_i = n\tau_i - rf_i, \qquad (4)$$

where I_w is the moment of inertia of the driving wheel; c is the coefficient of viscous friction; n is the speed reduction ratio of the speed reducer; r is the radius of the driving wheel; τ_i $(i = 1, 2, 3)$ is output torque of the driving wheel motors.

Combining (1)–(4) yields a dynamic model of the robot as:

$$\dot{\mathbf{V}} = \mathbf{A}_1\mathbf{V} + F(\mathbf{V}) + \mathbf{B}\boldsymbol{\tau} + \mathbf{d} =$$
$$= \begin{pmatrix} a_1 v_x \\ a_1 v_y \\ a_3\omega \end{pmatrix} + \begin{pmatrix} a_2\omega v_y \\ -a_2\omega v_x \\ 0 \end{pmatrix} + \begin{pmatrix} -b_1 & -b_1 & 2b_1 \\ \sqrt{3}b_1 & \sqrt{3}b_1 & 0 \\ b_2 & b_2 & b_2 \end{pmatrix}\boldsymbol{\tau} + \mathbf{d}, \qquad (5)$$

where $\mathbf{A}_1 = diag(a_1, a_1, a_3)$, $F(\mathbf{V}) = (a_2\omega v_y, -a_2\omega v_x, 0)^T$, $\mathbf{B} = \begin{pmatrix} -b_1 & -b_1 & 2b_1 \\ \sqrt{3}b_1 & \sqrt{3}b_1 & 0 \\ b_2 & b_2 & b_2 \end{pmatrix}$, $\boldsymbol{\tau}$ is output torque; $\mathbf{d}$ is disturbance of the system; system parameters are $a_1 = 3c/(3I_w + 2mr^2)$, $a_2 = 2mr^2/(3I_w + 2mr^2)$, $a_3 = 3cL^2/(3I_wL^2 + I_Vr^2)$, $b_1 = nr/(3I_w + 2mr^2)$, $b_2 = krL/(3I_wL^2 + I_Vr^2)$; k is the gain factor of driven torque.

Description of problem

In engineering practice, the drive motor of the mobile robot can only provide limited torque output, so the input saturation problem must be considered when designing the controller. The saturation function of the output torque can be given by the following equation

$$sat(\tau_i) = \begin{cases} \tau_{i\max} & \tau_i > \tau_{i\max} \\ \tau_i & \tau_{i\min} \le \tau_i \le \tau_{i\max} \\ -\tau_{i\min} & \tau_i < \tau_{i\min} \end{cases}, \quad (6)$$

$$sat(\tau_i) = \tau_i - \Delta\tau_i, \quad (7)$$

where $\tau_{i\max}$ and $\tau_{i\min}$ are the upper and lower bounds of the motor output; $sat(\tau_i)$ is the input saturation torque; $\Delta\tau_i$ is input saturation error; τ_i is control torque needs to be designed latter.

Assumption 1. Input saturation error $\boldsymbol{\Delta\tau}$ is bounded, and there exists a positive constant $\theta_{\Delta\tau}$ such that $\|\boldsymbol{\Delta\tau}\|^2 \le \theta_{\Delta\tau}$.

Combining (1), (5), and (6), one can obtain the model of the three wheeled omnidirectional mobile robot as:

$$\begin{cases} \dot{\mathbf{q}} = \mathbf{S}(\mathbf{q})\mathbf{V} \\ \dot{\mathbf{V}} = \mathbf{A}_1\mathbf{V} + F(\mathbf{V}) + \mathbf{B}sat(\boldsymbol{\tau}) + \mathbf{d} \end{cases}. \quad (8)$$

Assumption 2. External disturbances $\mathbf{d}$ is a bounded vector, and there exists a positive constant θ_d such that $\|\mathbf{d}\|^2 \le \theta_d$.

Assumption 3. The model parameters of the three wheeled omnidirectional mobile robot are unknown. $G_i = [\mathbf{A}_1\mathbf{V} + F(\mathbf{V})]_i$ ($i = 1, 2, 3$) is the unknown continuous function.

Since RBF Neural Network (RBFNN) can approximate any continuous function, this paper uses it to estimate the model unknown term G. RBFNN can be represented as

$$G_i = (\hat{\mathbf{W}}^T\boldsymbol{\varphi})_i + \boldsymbol{\delta}_i, \quad (9)$$

where $\mathbf{W}^T$ is the unknown weight vector for the neural network; $\boldsymbol{\varphi}_i$ is activation function vector; $\boldsymbol{\delta}_i$ is the approximation error. The value of the approximation function for the unknown term G can be expressed as $\hat{G}_i = (\hat{\mathbf{W}}^T\boldsymbol{\varphi})_i$.

In this paper, the Gaussian basis function [11] is used as the activation function of the neural network

$$\varphi = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_i\|^2}{\sigma_i^2}\right), \; i = 1, \ldots, n,$$

where $\mathbf{x}$ is the input vector; $\mathbf{c}_i$ and σ_i are the center and width of the Gaussian function, respectively; n is the number of neural nodes.

Assumption 4. The neural network approximation error $\boldsymbol{\delta}$ is a bounded vector, and there exists a positive constant θ_δ such that $\|\boldsymbol{\delta}\|^2 \le \theta_\delta$.

The input saturation phenomenon presented in the drive motors generates nonlinear perturbations and reduces the trajectory tracking accuracy of the robot, thus it needs to be compensated. Inspired by [17], the compensation system can be designed as

$$\begin{cases} \dot{\boldsymbol{\beta}}_1 = -\mathbf{L}_1\boldsymbol{\beta}_1 + \mathbf{S}\boldsymbol{\beta}_2 \\ \dot{\boldsymbol{\beta}}_2 = -\mathbf{L}_2\boldsymbol{\beta}_2 + \mathbf{B}\boldsymbol{\Delta\tau} \end{cases}, \quad (10)$$

where $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are the state variables of the compensation system; $\mathbf{L}_1 = diag(l_{11}, l_{12}, l_{13}) > 0$, $\mathbf{L}_2 = diag(l_{21}, l_{22}, l_{23}) > 0$ are system parameters.

Assumption 5. The state variables of the compensation system are bounded and satisfy $|\beta_{1i}| < \eta_{1i}$, $|\beta_{2i}| < \eta_{2i}$, η_{1i}, η_{2i} are constants.

Assumption 6. Reference trajectory $\mathbf{q}_r$, $\mathbf{V}_r = \mathbf{S}^{-1}(\mathbf{q})\dot{\mathbf{q}}_r$ is bounded and satisfies $|q_{ri}| < p_{1i}$, $|V_{ri}| < p_{2i}$. Moreover, the states of robot system are also bounded and satisfy $|q_i| < y_{1i}$, $|V_i| < y_{2i}$.

The purpose of this paper is to design a controller to ensure that the robot accurately tracks a given trajectory, in the presence of unknown model parameters, restricted system states, and input saturation in the actuators.

Controller Design

First of all, the robot trajectory tracking errors are defined as

$$\mathbf{e}_1 = \mathbf{q} - \mathbf{q}_r - \boldsymbol{\beta}_1, \quad (11)$$

$$\mathbf{e}_2 = \mathbf{V} - \boldsymbol{\alpha} - \boldsymbol{\beta}_2, \quad (12)$$

where $\mathbf{q}_r = (x_r, y_r, \theta_r)^T$ is the reference trajectory of the robot; $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)^T$ is the virtual control law which will be designed later.

Step 1. Taking the derivative of (11), we have

$$\dot{\mathbf{e}}_1 = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r - \dot{\boldsymbol{\beta}}_1, \quad (13)$$

bringing (1), (10) and (12) into (13) one has

$$\dot{\mathbf{e}}_1 = \mathbf{SV} - \dot{\mathbf{q}}_r - \dot{\boldsymbol{\beta}}_1 = \mathbf{S}(\mathbf{e}_2 + \boldsymbol{\alpha}) - \dot{\mathbf{q}}_r + \mathbf{L}_1\boldsymbol{\beta}_1. \quad (14)$$

Then we select the barrier Lyapunov function as

$$V_1 = \sum_{i=1}^{3} \frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right) + \frac{1}{2}\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1, \quad (15)$$

where $\mathbf{b}_1 = (b_{11}, b_{12}, b_{13})^T$ is boundary on the tracking error of the system, to be discussed later; V_1 is a continuous function and greater than zero.

Define

$$\lambda_{ji} = \frac{e_{ji}}{\cos^2\left(\frac{\pi e_{ji}^2}{2b_{ji}^2}\right)} \quad (16)$$

taking the derivative of (15) and bringing (16), (14) into the equation gets

$$\begin{aligned} \dot{V}_1 &= \sum_{i=1}^{3} \frac{e_{1i}\dot{e}_{1i}}{\cos^2\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right)} + \boldsymbol{\beta}_1^T\dot{\boldsymbol{\beta}}_1 = \\ &= \sum_{i=1}^{3} \lambda_{1i}[\mathbf{S}(\mathbf{e}_2 + \boldsymbol{\alpha}) - \dot{\mathbf{q}}_r + \mathbf{L}_1\boldsymbol{\beta}_1]_i + \boldsymbol{\beta}_1^T(-\mathbf{L}_1\boldsymbol{\beta}_1 + \mathbf{S}\boldsymbol{\beta}_2) = \\ &= \sum_{i=1}^{3} \lambda_{1i}[\mathbf{S}(\mathbf{e}_2 + \boldsymbol{\alpha}) - \dot{\mathbf{q}}_r + \mathbf{L}_1\boldsymbol{\beta}_1]_i - \boldsymbol{\beta}_1^T\mathbf{L}_1\boldsymbol{\beta}_1 + \boldsymbol{\beta}_1^T\mathbf{S}\boldsymbol{\beta}_2. \end{aligned} \quad (17)$$

The we choose of virtual control law as

$$\boldsymbol{\alpha} = \mathbf{S}^{-1}\left[\frac{-\mathbf{k}_1\mathbf{b}_1^2\sin\left(\frac{\pi\mathbf{e}_1^2}{\mathbf{b}_1^2}\right)}{2\pi\mathbf{e}_1} + \dot{\mathbf{q}}_r - \mathbf{L}_1\boldsymbol{\beta}_1\right], \tag{18}$$

where $\mathbf{k}_1 = (k_{11}, k_{12}, k_{13})^T > 0$ is the control gain.

From Young's inequality one gets

$$\boldsymbol{\beta}_1^T\mathbf{S}\boldsymbol{\beta}_2 \le \frac{1}{2}\boldsymbol{\beta}_1^T\mathbf{S}\boldsymbol{\beta}_1 + \frac{1}{2}\boldsymbol{\beta}_2^T\mathbf{S}\boldsymbol{\beta}_2. \tag{19}$$

Carrying equations (18) and (19) into (17) we obtain

$$\begin{aligned}\dot{V}_1 \le \sum_{i=1}^{3} -k_{1i}\frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right) + \sum_{i=1}^{3}\lambda_{1i}(\mathbf{S}\mathbf{e}_2)_i - \\ - \gamma_1\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 + \frac{1}{2}\boldsymbol{\beta}_2^T\mathbf{S}\boldsymbol{\beta}_2,\end{aligned} \tag{20}$$

where $\gamma_1 = \lambda_{\min}\left(\mathbf{L}_1 - \frac{\mathbf{S}}{2}\right) > 0$.

Step 2. Finding the derivative of (11) and bringing (7), (8) and (12) into the derivative equation we obtain

$$\begin{aligned}\dot{\mathbf{e}}_2 = \dot{\mathbf{V}} - \dot{\boldsymbol{\alpha}} - \dot{\boldsymbol{\beta}}_2 = \mathbf{W}^T\boldsymbol{\varphi} + \boldsymbol{\delta} + \mathbf{B}(sat(\boldsymbol{\tau}) - \Delta\boldsymbol{\tau}) + \mathbf{d} - \dot{\boldsymbol{\alpha}} + \mathbf{L}_2\boldsymbol{\beta}_2 = \\ = \mathbf{W}^T\boldsymbol{\varphi} + \boldsymbol{\delta} + \mathbf{B}\boldsymbol{\tau} + \mathbf{d} - \dot{\boldsymbol{\alpha}} + \mathbf{L}_2\boldsymbol{\beta}_2\end{aligned} \tag{21}$$

and construct the Lyapunov function as

$$V_2 = V_1 + \sum_{i=1}^{3}\frac{b_{2i}^2}{\pi}\tan\left(\frac{\pi e_{2i}^2}{2b_{2i}^2}\right) + \frac{1}{2}\boldsymbol{\beta}_2^T\boldsymbol{\beta}_2, \tag{22}$$

where b_{2i} is a bound on the tracking error of the system whose values are discussed later; V_2 is a continuous function and it is greater than zero.

Differentiating (22), bringing (9), (21), (12) into function, and assuming (16) yields

$$\begin{aligned}\dot{V}_2 = \dot{V}_1 + \sum_{i=1}^{3}\lambda_{2i}\dot{e}_{2i} + \boldsymbol{\beta}_1^T\dot{\boldsymbol{\beta}}_1 = \\ = \dot{V}_1 + \sum_{i=1}^{3}\lambda_{2i}(\mathbf{W}^T\boldsymbol{\varphi} + \boldsymbol{\delta} + \mathbf{B}\boldsymbol{\tau} + \mathbf{d} - \dot{\boldsymbol{\alpha}} + \mathbf{L}_2\boldsymbol{\beta}_2)_i + \\ + \boldsymbol{\beta}_1^T(-\mathbf{L}_2\boldsymbol{\beta}_2 + \mathbf{B}\Delta\boldsymbol{\tau}).\end{aligned} \tag{23}$$

The derivatives of the virtual control rate can make the computation complicated and cause the differential explosion problem. In order to reduce the computational complexity, this paper uses the second-order sliding mode differentiator as a dynamic surface to calculate its derivative.

$$\begin{aligned}\dot{\boldsymbol{\xi}}_1 = -r_1|\boldsymbol{\xi}_1 - \boldsymbol{\alpha}|^{\frac{1}{2}}sign(\boldsymbol{\xi}_1 - \boldsymbol{\alpha}) + \boldsymbol{\xi}_2, \\ \dot{\boldsymbol{\xi}}_2 = -r_2 sign(\boldsymbol{\xi}_1 - \boldsymbol{\alpha}),\end{aligned} \tag{24}$$

where $\boldsymbol{\xi}_1 = (\xi_{11}, \xi_{12}, \xi_{13})^T$, $\boldsymbol{\xi}_2 = (\xi_{21}, \xi_{22}, \xi_{23})^T$ are the state variables of the second-order filter; r_1, r_2 are the system parameters; $sign()$ is the sign function. According to reference [18], $\boldsymbol{\xi}_1$ and $\dot{\boldsymbol{\xi}}_1$ will converge to the virtual control law $\boldsymbol{\alpha}$ and its derivative $\dot{\boldsymbol{\alpha}}$ respectively. When inputting signal with disturbances, the system will generate tracking errors $\boldsymbol{\varepsilon}$ and $|\dot{\boldsymbol{\xi}}_{1i} - \dot{\alpha}_i| < \varepsilon_i$.

Design control torque is as

$$\begin{aligned}\boldsymbol{\tau} = \boldsymbol{\tau}_1 + \boldsymbol{\tau}_2, \\ \boldsymbol{\tau}_1 = \mathbf{B}^{-1}\left[\frac{-\mathbf{k}_2\mathbf{b}_2^2\sin\left(\frac{\pi\mathbf{e}_2^2}{\mathbf{b}_2^2}\right)}{2\pi\mathbf{e}_2} - \frac{\boldsymbol{\lambda}_1}{\boldsymbol{\lambda}_2}\mathbf{S}\mathbf{e}_2 - \right. \\ \left. - \mathbf{L}_2\boldsymbol{\beta}_2 - \begin{pmatrix}\hat{\mathbf{W}}_1^T\boldsymbol{\varphi}_1 \\ \hat{\mathbf{W}}_2^T\boldsymbol{\varphi}_2 \\ \hat{\mathbf{W}}_3^T\boldsymbol{\varphi}_3\end{pmatrix} + \dot{\boldsymbol{\xi}}_1\right],\end{aligned} \tag{25}$$

where $\mathbf{k}_2 = (k_{21}, k_{22}, k_{23})^T > 0$ is control gain; $\boldsymbol{\tau}_2$ as disturbance compensation term to be designed later.

Define the neural network parameter estimation error as

$$\tilde{\mathbf{W}}_i = \mathbf{W}_i - \hat{\mathbf{W}}_i, \tag{26}$$

where $\tilde{\mathbf{W}}_i$ is the parameter estimation error; $\hat{\mathbf{W}}_i$ is the estimated value of the parameter.

By taking (20), (24)–(26) into (23) we obtain

$$\begin{aligned}\dot{V}_2 \le \sum_{i=1}^{3}\left\{-k_{1i}\frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right)\right\} + \sum_{i=1}^{3}\left\{-k_{2i}\frac{b_{2i}^2}{\pi}\tan\left(\frac{\pi e_{2i}^2}{2b_{2i}^2}\right)\right\} - \\ - \gamma_1\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2^T\left(\mathbf{L}_2 - \frac{\mathbf{S}}{2}\right)\boldsymbol{\beta}_2 + \\ + \sum_{i=1}^{3}\left\{\lambda_{2i}\left(\begin{pmatrix}\tilde{\mathbf{W}}_1^T\boldsymbol{\varphi}_1 \\ \tilde{\mathbf{W}}_2^T\boldsymbol{\varphi}_2 \\ \tilde{\mathbf{W}}_3^T\boldsymbol{\varphi}_3\end{pmatrix} + \boldsymbol{\delta} + \mathbf{B}\boldsymbol{\tau}_2 + \mathbf{d} + \dot{\boldsymbol{\xi}}_1 - \dot{\boldsymbol{\alpha}}\right)_i\right\} + \boldsymbol{\beta}_1^T\mathbf{B}\Delta\boldsymbol{\tau}.\end{aligned} \tag{27}$$

Step 3. Select the Lyapunov function as

$$V_3 = V_2 + \frac{1}{2\rho}\sum_{i=1}^{3}\tilde{\mathbf{W}}_i^T\tilde{\mathbf{W}}_i. \tag{28}$$

Taking the derivative of (28) and bringing (26), (27) into it yields

$$\begin{aligned}\dot{V}_3 = \dot{V}_2 - \frac{1}{\rho}\sum_{i=1}^{3}\tilde{\mathbf{W}}_i^T\dot{\hat{\mathbf{W}}}_i \le \\ \le \sum_{i=1}^{3}\left\{-k_{1i}\frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right)\right\} + \sum_{i=1}^{3}\left\{-k_{2i}\frac{b_{2i}^2}{\pi}\tan\left(\frac{\pi e_{2i}^2}{2b_{2i}^2}\right)\right\} - \\ - \gamma_1\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2^T\left(\mathbf{L}_2 - \frac{\mathbf{S}}{2}\right)\boldsymbol{\beta}_2 + \\ + \sum_{i=1}^{3}\left\{\lambda_{2i}\left(\begin{pmatrix}\tilde{\mathbf{W}}_1^T\boldsymbol{\varphi}_1 \\ \tilde{\mathbf{W}}_2^T\boldsymbol{\varphi}_2 \\ \tilde{\mathbf{W}}_3^T\boldsymbol{\varphi}_3\end{pmatrix} + \boldsymbol{\delta} + \mathbf{B}\boldsymbol{\tau}_2 + \mathbf{d} + \boldsymbol{\varepsilon}\right)_i\right\} + \boldsymbol{\beta}_1^T\mathbf{B}\Delta\boldsymbol{\tau} - \\ - \frac{1}{\rho}\sum_{i=1}^{3}\tilde{\mathbf{W}}_i^T\dot{\hat{\mathbf{W}}}_i.\end{aligned} \tag{29}$$

According to Young's inequality, we obtain

$$\sum_{i=1}^{3}\lambda_{2i}\delta_i \le \frac{1}{2}\sum_{i=1}^{3}\lambda_{2i}^2 + \frac{1}{2}\|\boldsymbol{\delta}\|^2 \le \frac{1}{2}\sum_{i=1}^{3}\lambda_{2i}^2 + \frac{1}{2}\theta_\delta^2, \quad (30)$$

$$\sum_{i=1}^{3}\lambda_{2i}d_i \le \frac{1}{2}\sum_{i=1}^{3}\lambda_{2i}^2 + \frac{1}{2}\|\mathbf{d}\|^2 \le \frac{1}{2}\sum_{i=1}^{3}\lambda_{2i}^2 + \frac{1}{2}\theta_d^2, \quad (31)$$

$$\sum_{i=1}^{3}\lambda_{2i}\varepsilon_i \le \frac{1}{2}\sum_{i=1}^{3}\lambda_{2i}^2 + \frac{1}{2}\|\boldsymbol{\varepsilon}\|^2 \le \frac{1}{2}\sum_{i=1}^{3}\lambda_{2i}^2 + \frac{1}{2}\theta_\varepsilon^2, \quad (32)$$

$$\boldsymbol{\beta}_1^T\mathbf{B}\Delta\boldsymbol{\tau} \le \frac{1}{2}\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 + \frac{1}{2}\|\mathbf{B}\|^2\|\Delta\boldsymbol{\tau}\|^2 \le \frac{1}{2}\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 + \frac{1}{2}\|\mathbf{B}\|^2\theta_{\Delta\tau}^2. \quad (33)$$

Then choosing the control law, we get adaptive control law

$$\boldsymbol{\tau}_2 = \mathbf{B}^{-1}\left(-\frac{3}{2}\boldsymbol{\lambda}_2\right), \quad (34)$$

$$\dot{\hat{\mathbf{W}}}_i = \rho(-\rho_w\hat{\mathbf{W}}_i + \lambda_{2i}\varphi_i), \quad (35)$$

where ρ and ρ_w are the control gains.

Here, we can get the control law as

$$\boldsymbol{\tau} = \mathbf{B}^{-1}\left[\frac{-\mathbf{k}_2\mathbf{b}_2^2\sin\left(\frac{\pi\mathbf{e}_2^2}{\mathbf{b}_2^2}\right)}{2\pi\mathbf{e}_2} - \frac{\boldsymbol{\lambda}_1}{\boldsymbol{\lambda}_2}\mathbf{S}\mathbf{e}_2 - \mathbf{L}_2\boldsymbol{\beta}_2 - \begin{pmatrix}\hat{\mathbf{W}}_1^T\boldsymbol{\varphi}_1\\ \hat{\mathbf{W}}_2^T\boldsymbol{\varphi}_2\\ \hat{\mathbf{W}}_3^T\boldsymbol{\varphi}_3\end{pmatrix} + \dot{\boldsymbol{\xi}}_1 - \frac{3}{2}\boldsymbol{\lambda}_2\right].$$

Bringing (30)–(35) into (29) yields

$$\dot{V}_3 \le \sum_{i=1}^{3}\left\{-k_{1i}\frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right)\right\} + \sum_{i=1}^{3}\left\{-k_{2i}\frac{b_{2i}^2}{\pi}\tan\left(\frac{\pi e_{2i}^2}{2b_{2i}^2}\right)\right\} - \gamma_1\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 - \gamma_2\boldsymbol{\beta}_2^T\boldsymbol{\beta}_2 + \sum_{i=1}^{3}\rho_w\tilde{\mathbf{W}}_i^T\hat{\mathbf{W}}_i + \frac{1}{2}\|\mathbf{B}\|^2\theta_{\Delta\tau}^2 + \frac{1}{2}\left(\theta_\delta^2 + \theta_d^2 + \theta_\varepsilon^2\right), \quad (36)$$

where $\gamma_2 = \lambda_{\min}\left(\mathbf{L}_2 - \frac{\mathbf{S}}{2}\right) - \frac{1}{2} > 0$.

For $\sum_{i=1}^{3}\tilde{\mathbf{W}}_i^T\hat{\mathbf{W}}_i$, we have

$$\tilde{\mathbf{W}}_i^T\hat{\mathbf{W}}_i \le -\frac{1}{2}\tilde{\mathbf{W}}_i^T\tilde{\mathbf{W}}_i + \frac{1}{2}\mathbf{W}_i^T\mathbf{W}_i. \quad (37)$$

Bringing (37) into (36) gives

$$\dot{V}_3 \le \sum_{i=1}^{3}\left\{-k_{1i}\frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right)\right\} + \sum_{i=1}^{3}\left\{-k_{2i}\frac{b_{2i}^2}{\pi}\tan\left(\frac{\pi e_{2i}^2}{2b_{2i}^2}\right)\right\} - \gamma_1\boldsymbol{\beta}_1^T\boldsymbol{\beta}_1 - \gamma_2\boldsymbol{\beta}_2^T\boldsymbol{\beta}_2 - \frac{1}{2}\rho_w\sum_{i=1}^{3}\tilde{\mathbf{W}}_i^T\tilde{\mathbf{W}}_i + \frac{1}{2}\|\mathbf{B}\|^2\theta_{\Delta\tau}^2 + \frac{1}{2}\left(\theta_\delta^2 + \theta_d^2 + \theta_\varepsilon^2\right) + \sum_{i=1}^{3}\mathbf{W}_i^T\mathbf{W}_i.$$

Let $a = \min\{\lambda_{\min}(\mathbf{k}_1), \lambda_{\min}(\mathbf{k}_2), 2\gamma_1, 2\gamma_2, \rho\rho_w\}$,

$$b = \frac{1}{2}\|\mathbf{B}\|^2\theta_{\Delta\tau}^2 + \frac{1}{2}\left(\theta_\delta^2 + \theta_d^2 + \theta_\varepsilon^2\right) + \sum_{i=1}^{3}\mathbf{W}_i^T\mathbf{W}_i,$$

then one has $\dot{V}_3 \le -aV_3 + b$. (38)

Integrating (38) yields

$$V_3(t) \le \left(V_3(0) - \frac{b}{a}\right)\exp(-at) + \frac{b}{a} \le V_3(0) + \frac{b}{a}. \quad (39)$$

According to equations (28) and (39), it is known that

$$\sum_{i=1}^{3}\left\{\frac{b_{1i}^2}{\pi}\tan\left(\frac{\pi e_{1i}^2}{2b_{1i}^2}\right)\right\} \le V_3(t) \le V_3(0) + \frac{b}{a},$$

$$\sum_{i=1}^{3}\left\{\frac{b_{2i}^2}{\pi}\tan\left(\frac{\pi e_{2i}^2}{2b_{2i}^2}\right)\right\} \le V_3(t) \le V_3(0) + \frac{b}{a}.$$

Solving the equations obtains

$$|e_{1i}| \le \sqrt{\frac{2b_{1i}^2}{\pi}\arctan\left[\frac{\pi}{b_{1i}^2}\left(V_3(0) + \frac{b}{a}\right)\right]} < b_{1i},$$

$$|e_{2i}| \le \sqrt{\frac{2b_{2i}^2}{\pi}\arctan\left[\frac{\pi}{b_{2i}^2}\left(V_3(0) + \frac{b}{a}\right)\right]} < b_{2i}.$$

Obviously, the control algorithm designed in this paper can guarantee that the trajectory tracking errors stay within the constraints. From robot trajectory tracking errors (11) and (12), we can conclude that $|\mathbf{e}_1| = |\mathbf{q} - \mathbf{q}_r - \boldsymbol{\beta}_1| \le \mathbf{b}_1$, $|\mathbf{e}_2| = |\mathbf{V} - \boldsymbol{\alpha} - \boldsymbol{\beta}_2| \le \mathbf{b}_2$.When the initial state of the system satisfies the limits $|\mathbf{q}(0) - \mathbf{q}_r(0) - \boldsymbol{\beta}_1(0)| \le \mathbf{b}_1$, $|\mathbf{V}(0) - \boldsymbol{\alpha}(0) - \boldsymbol{\beta}_2(0)| \le \mathbf{b}_2$, the system will operate within the specified error range, then the states of the system will satisfy the inequality $|\mathbf{q}| \le |\mathbf{b}_1| + |\mathbf{q}_r| + |\boldsymbol{\beta}_1|$, $|\mathbf{V}| \le |\mathbf{b}_2| + |\boldsymbol{\alpha}| + |\boldsymbol{\beta}_2|$.

Now we discuss the problem of system boundary selection. It can be known from assumption 5 and 6 that $|\beta_{1i}| < \eta_{1i}$, $|\beta_{2i}| < \eta_{2i}$, $|q_{ri}| < p_{1i}$, $|V_{ri}| < p_{2i}$, $|q_i| < y_{1i}$, $|V_i| < y_{2i}$, and we've proved that $|e_{1i}| < b_{1i}$, $|e_{2i}| < b_{2i}$. According to definition (11), we get $|q_i| \le |e_{1i}| + |q_{ri}| + |\beta_{1i}| < b_{1i} + p_{1i} + \eta_{1i}$, when choosing $b_{1i} = y_{1i} - p_{1i} - \eta_{1i}$, the state of the system will satisfy the boundary condition $|q_i| < y_{1i}$. Similarly, from definition (12) one has $|V_i| \le |e_{2i}| + |\alpha_i|_{\max} + |\beta_{2i}| < b_{2i} + |\alpha_i|_{\max} + \eta_{2i}$, where $|\alpha_i|_{\max}$ is the maximum of virtual control law. When choosing $b_{2i} = y_{2i} - |\alpha_i|_{\max} - \eta_{2i}$, the speed of robot will satisfy the boundary condition $|V_i| < y_{2i}$.

Numerical simulation

In this section, simulation experiment demonstrates the effectiveness of the controller proposed in this paper. The parameters of the robot are chosen as $m = 9$ kg, $L = 0.4$ m, $r = 0.05$ m, $I_V = 11$ kg·m^2, $I_w = 0.05$ kg·m^2, $c = 0.02$ kg·m^2/s. The reference trajectory is $q_{r1} = 1.5\sin(0.1t)$ m, $q_{r2} = \sin(0.2t)$ m, $q_{r3} = \sin(0.1t)$ rad. The control gains are $\mathbf{k}_1 = (0.5, 0.5, 0.5)^T$, $\mathbf{k}_2 = (0.1, 0.1, 0.1)^T$.

The dynamic surface parameters are chosen as $r_1 = 1.1$, $r_2 = 1.1$. The parameters of the auxiliary system are $\mathbf{L}_1 = = diag(20, 20, 20)$, $\mathbf{L}_2 = diag(20, 20, 20)$.

The boundaries of the tracking errors are chosen as $\mathbf{b}_1 = (0.5, 0.5, 0.5)^T$, $\mathbf{b}_2 = (0.5, 0.5, 0.5)^T$. The neural network with 11 nodes, the center is selected to be uniformly distributed in the interval [–1, 1], and the width is 2. The adaptive gains are $\rho = \rho_w = 2$. The initial values for each system are chosen as $\mathbf{q}(0) = (0.4,–0.4, 0.3)^T$, $\mathbf{V}(0) = = (0.1, 0.1, 0.1)^T$, $\mathbf{W}_i(0) = zeros(11.1)$, $\boldsymbol{\beta}_1 = (0, 0, 0)^T$, $\boldsymbol{\beta}_2 = = (0, 0, 0)^T$. The external disturbance on the robot is $d_1 = 0.01\sin(t)$, $d_2 = 0.01\sin(2t)$, $d_3 = 0.01\sin3(t)$. The saturation input is limited to $\tau_{max} = 300$ N·m, $\tau_{min} = = 300$ N·m.

The results of the experiment can be summarized as follows: Fig. 2 shows the whole motion of the robot; Fig. 3, Fig. 4 show the robot position and velocity following respectively; Fig. 5 shows the errors of robot position and velocity; Fig. 6 gives the estimated output of the dynamic surface; Fig. 7 presents the saturation control torque $sat(\boldsymbol{\tau})$.

The experimental results demonstrate that the control algorithm proposed in this paper can enable the robot to trace the reference trajectory with the given initial conditions. The robot system states $\mathbf{q}$ and $\mathbf{V}$ can gradually converge to the given states by satisfying the defined constraints; the position error and velocity error are able to be gradually stabilized in a smaller region near zero. The saturation control input guarantees that the control torque does not exceed the maximum output of the motor.

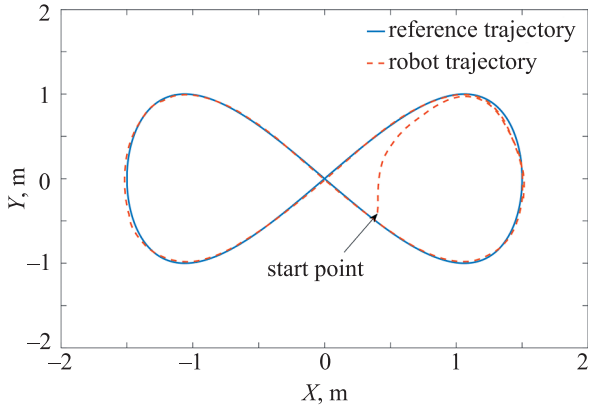

Fig. 2. The trajectory of robot

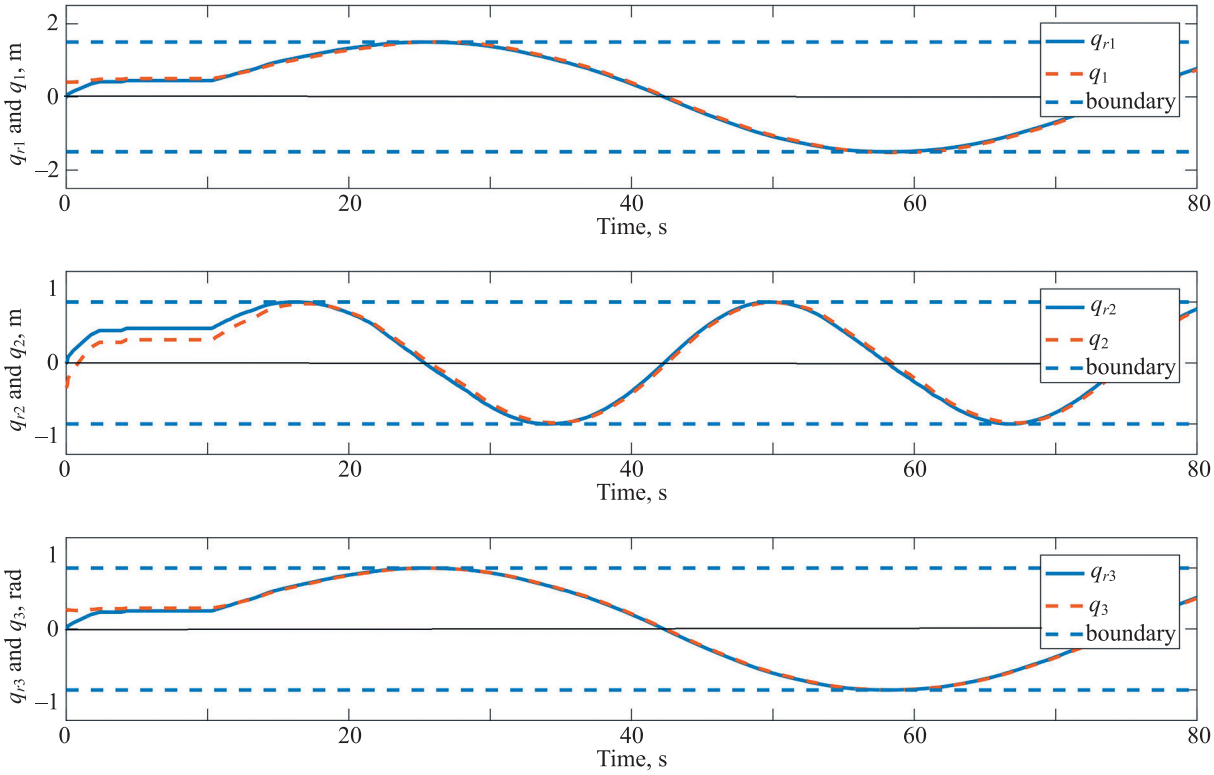

Fig. 3. Comparison between $\mathbf{q}$ and $\mathbf{q}_r$

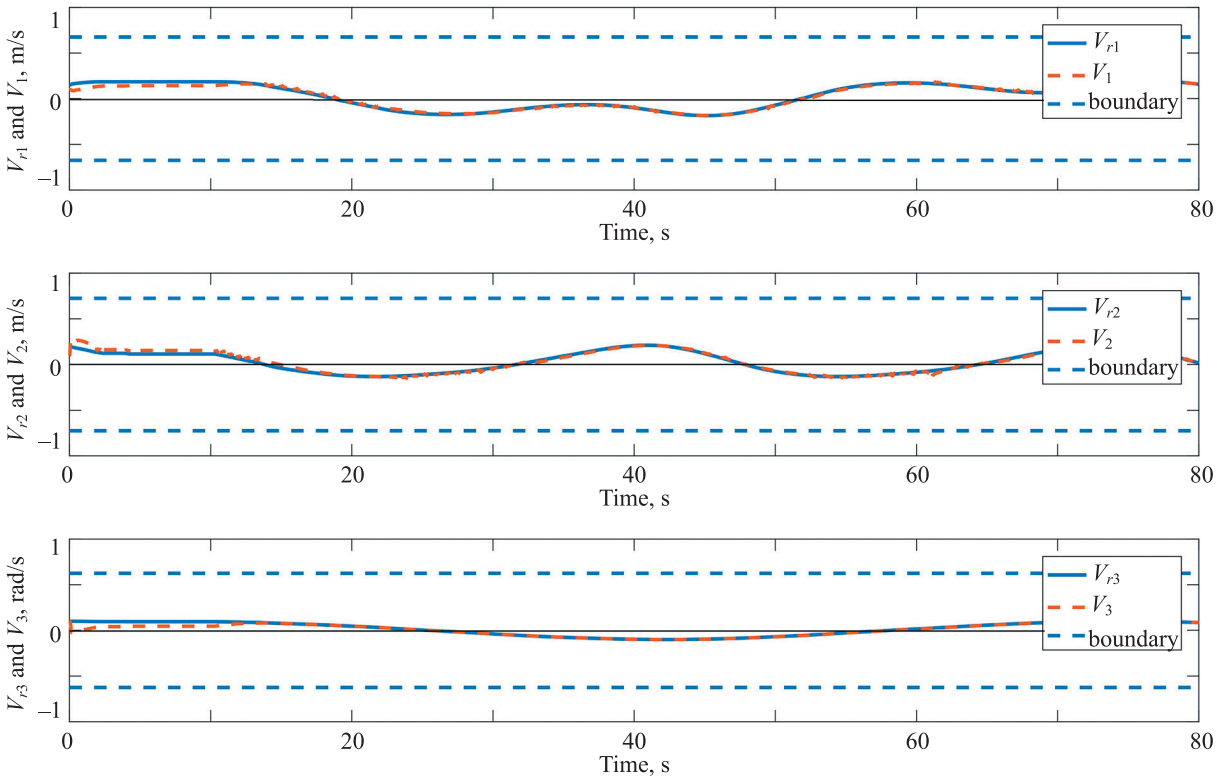

Fig. 4. Comparison between **V** and $\mathbf{V}_r$

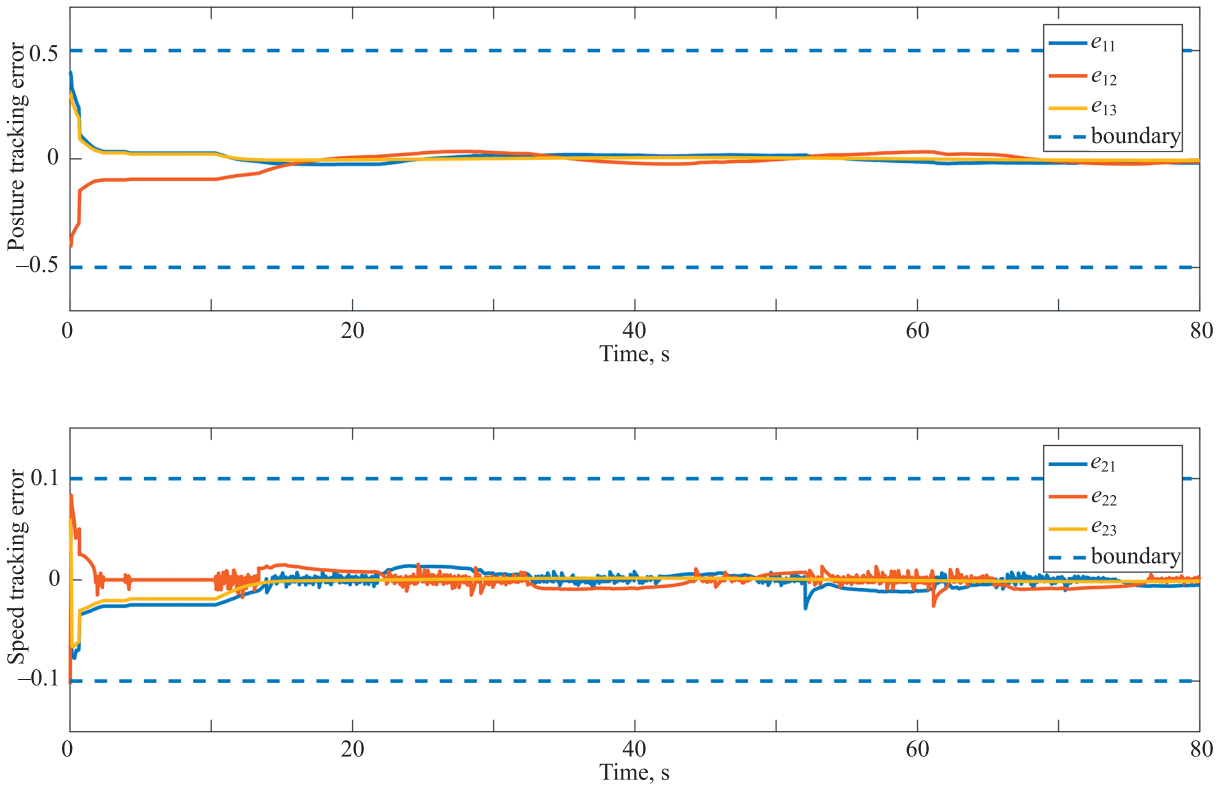

Fig. 5. Trajectory tracking error and speed tracking error

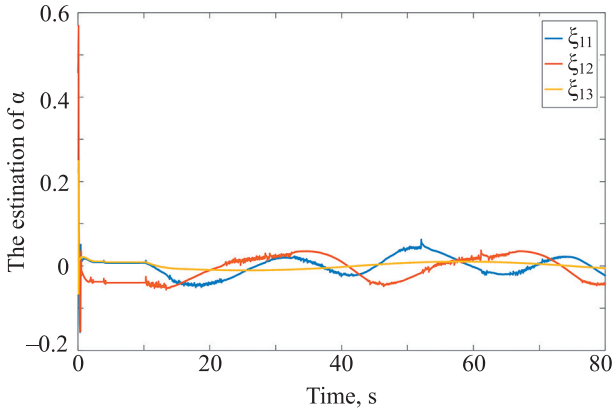

Fig. 6. The output of dynamic surface

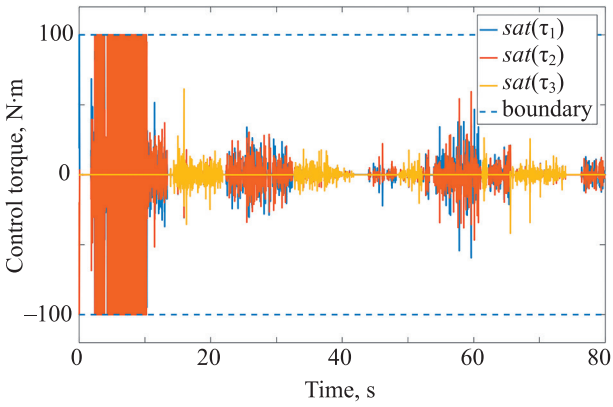

Fig. 7. Control torque $sat(\boldsymbol{\tau})$

Conclusion

In this paper, we focus on the trajectory tracking motion of a three-wheeled omnidirectional mobile robot with full state constraints and the presence of output saturation of the driving motors. Firstly, we analyze the kinematic and dynamic models of the three-wheeled omnidirectional mobile robot. The model of the robot with actuator saturation is given. When designing the controller, we adopt tan-type barrier Lyapunov function and backstepping method. This ensures that the system full states will not violate the given constraints when the robot is performing trajectory tracking. Considering the differential explosion problem, which occurs in solving the derivatives of the virtual control law, we use a second-order differential sliding mode surface to calculate it, so as to reduce the complexity of the operation. In addition, due to the output saturation problem of the robot drive motor, we build an auxiliary compensation system to compensate the error generated by the saturation function. It eliminates the influence of input saturation on the robot trajectory tracking accuracy. Finally, an experimental simulation is performed in MATLAB, and the simulation results illustrate the effectiveness of the control algorithm proposed in this paper.

1. However, there are still some places that need to be improved for the study in this paper. The algorithm proposed in this paper can only ensure that the system error converges to zero asymptotically. Robots often need to realize trajectory tracking quickly in practical application. For this reason, the controller can be further improved to converge to zero at a specified time.
2. Due to the RBFNN, the dynamic surface and the saturation function generate output errors that affect the robot tracking accuracy. This paper only partially compensates for the disturbance, and does not completely eliminate the affect generated by the error. It can be further improved by estimating the upper bound of the error by adaptive algorithms to further reduce the perturbation of the error.

References

1. Kramer J., Scheutz M. Development environments for autonomous mobile robots: A survey. *Autonomous Robots*, 2007, vol. 22, no. 1, pp. 101–132. https://doi.org/10.1007/s10514-006-9013-8
2. Watanabe K., Shiraishi Y., Tzafestas S.G., Tang J., Fukuda T. Feedback control of an omnidirectional autonomous platform for mobile service robots. *Journal of Intelligent and Robotic Systems*, 1998, vol. 22, pp. 315–330. https://doi.org/10.1023/A:1008048307352
3. Kalmár-Nagy T., D'Andrea R., Ganguly P. Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle. *Robotics and Autonomous Systems*, 2003, vol. 46, no. 1, pp. 47–64. https://doi.org/10.1016/j.robot.2003.10.003
4. Liu Y., Zhu J.J., Williams R.L. II, Wu J. Omni-directional mobile robot controller based on trajectory linearization. *Robotics and autonomous Systems*, 2008, vol. 56, no. 5, pp. 461–479. https://doi.org/10.1016/j.robot.2007.08.007
5. Huang H.C., Tsai C.C. Adaptive trajectory tracking and stabilization for omnidirectional mobile robot with dynamic effect and uncertainties. *IFAC Proceedings Volumes*, 2008, vol. 41, no. 2, pp. 5383–5388. https://doi.org/10.3182/20080706-5-KR-1001.00907
6. Alakshendra V., Chiddarwar S.S. Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties. *Nonlinear Dynamics*, 2017, vol. 87, no. 4, pp. 2147–2169. https://doi.org/10.1007/s11071-016-3179-1
7. Lu X., Zhang X., Zhang G., Fan J., Jia S. Neural network adaptive sliding mode control for omnidirectional vehicle with uncertainties.

Литература

1. Kramer J., Scheutz M. Development environments for autonomous mobile robots: A survey // Autonomous Robots. 2007. V. 22. N 1. P. 101–132. https://doi.org/10.1007/s10514-006-9013-8
2. Watanabe K., Shiraishi Y., Tzafestas S.G., Tang J., Fukuda T. Feedback control of an omnidirectional autonomous platform for mobile service robots // Journal of Intelligent and Robotic Systems. 1998. V. 22. P. 315–330. https://doi.org/10.1023/A:1008048307352
3. Kalmár-Nagy T., D'Andrea R., Ganguly P. Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle // Robotics and Autonomous Systems. 2003. V. 46. N 1. P. 47–64. https://doi.org/10.1016/j.robot.2003.10.003
4. Liu Y., Zhu J.J., Williams R.L. II, Wu J. Omni-directional mobile robot controller based on trajectory linearization // Robotics and autonomous Systems. 2008. V. 56. N 5. P. 461–479. https://doi.org/10.1016/j.robot.2007.08.007
5. Huang H.C., Tsai C.C. Adaptive trajectory tracking and stabilization for omnidirectional mobile robot with dynamic effect and uncertainties // IFAC Proceedings Volumes. 2008. V. 41. N 2. P. 5383–5388. https://doi.org/10.3182/20080706-5-KR-1001.00907
6. Alakshendra V., Chiddarwar S.S. Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties // Nonlinear Dynamics. 2017. V. 87. N 4. P. 2147–2169. https://doi.org/10.1007/s11071-016-3179-1
7. Lu X., Zhang X., Zhang G., Fan J., Jia S. Neural network adaptive sliding mode control for omnidirectional vehicle with uncertainties //

ISA Transactions, 2019, vol. 86, pp. 201–214. https://doi.org/10.1016/j.isatra.2018.10.043
8. Zijie N., Qiang L., Yonjie C., Zhijun S. Fuzzy control strategy for course correction of omnidirectional mobile robot. *International Journal of Control, Automation and Systems*, 2019, vol. 17, no. 9, pp. 2354–2364. https://doi.org/10.1007/s12555-018-0633-5
9. Tee K.P., Ge S.S., Tay E.H. Barrier Lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*, 2009, vol. 45, no. 4, pp. 918–927. https://doi.org/10.1016/j.automatica.2008.11.017
10. Xi C., Dong J. Adaptive neural network-based control of uncertain nonlinear systems with time-varying full-state constraints and input constraint. *Neurocomputing*, 2019, vol. 357, pp. 108–115. https://doi.org/10.1016/j.neucom.2019.04.060
11. Ding L., Li S., Liu Y.J., Gao H., Chen C., Deng Z. Adaptive neural network-based tracking control for full-state constrained wheeled mobile robotic system. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017, vol. 47, no. 8, pp. 2410–2419. https://doi.org/10.1109/TSMC.2017.2677472
12. Dong C., Liu Y., Wang Q. Barrier Lyapunov function based adaptive finite-time control for hypersonic flight vehicles with state constraints. *ISA Transactions*, 2020, vol. 96, pp. 163–176. https://doi.org/10.1016/j.isatra.2019.06.011
13. Doyle J.C., Smith R.S., Enns D.F. Control of plants with input saturation nonlinearities. *American Control Conference*, IEEE, 1987, pp. 1034–1039. https://doi.org/10.23919/ACC.1987.4789464
14. Mofid O., Mobayen S. Adaptive finite-time backstepping global sliding mode tracker of quad-rotor UAVs under model uncertainty, wind perturbation, and input saturation. *IEEE Transactions on Aerospace and Electronic Systems*, 2022, vol. 58, no. 1, pp. 140–151. https://doi.org/10.1109/TAES.2021.3098168
15. Chen X., Jia Y., Matsuno F. Tracking control for differential-drive mobile robots with diamond-shaped input constraints. *IEEE Transactions on Control Systems Technology*, 2014, vol. 22, no. 5, pp. 1999–2006. https://doi.org/10.1109/TCST.2013.2296900
16. Yang C., Huang D., He W., Cheng L. Neural control of robot manipulators with trajectory tracking constraints and input saturation. *IEEE Transactions on Neural Networks and Learning Systems*, 2021, vol. 32, no. 9, pp. 4231–4242. https://doi.org/10.1109/TNNLS.2020.3017202
17. Gao Y.-F., Sun X.-M., Wen C., Wang W. Adaptive tracking control for a class of stochastic uncertain nonlinear systems with input saturation. *IEEE Transactions on Automatic Control*, 2017, vol. 62, no. 5, pp. 2498–2504. https://doi.org/10.1109/TAC.2016.2600340
18. Levant A. Higher-order sliding modes, differentiation and output-feedback control. *International Journal of Control*, 2003, vol. 76, no. 9-10, pp. 924–941. https://doi.org/10.1080/0020717031000099029

ISA Transactions. 2019. V. 86. P. 201–214. https://doi.org/10.1016/j.isatra.2018.10.043
8. Zijie N., Qiang L., Yonjie C., Zhijun S. Fuzzy control strategy for course correction of omnidirectional mobile robot // International Journal of Control, Automation and Systems. 2019. V. 17. N 9. P. 2354–2364. https://doi.org/10.1007/s12555-018-0633-5
9. Tee K.P., Ge S.S., Tay E.H. Barrier Lyapunov functions for the control of output-constrained nonlinear systems // Automatica. 2009. V. 45. N 4. P. 918–927. https://doi.org/10.1016/j.automatica.2008.11.017
10. Xi C., Dong J. Adaptive neural network-based control of uncertain nonlinear systems with time-varying full-state constraints and input constraint // Neurocomputing. 2019. V. 357. P. 108–115. https://doi.org/10.1016/j.neucom.2019.04.060
11. Ding L., Li S., Liu Y.J., Gao H., Chen C., Deng Z. Adaptive neural network-based tracking control for full-state constrained wheeled mobile robotic system // IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2017. V. 47. N 8. P. 2410–2419. https://doi.org/10.1109/TSMC.2017.2677472
12. Dong C., Liu Y., Wang Q. Barrier Lyapunov function based adaptive finite-time control for hypersonic flight vehicles with state constraints // ISA Transactions. 2020. V. 96. P. 163–176. https://doi.org/10.1016/j.isatra.2019.06.011
13. Doyle J.C., Smith R.S., Enns D.F. Control of plants with input saturation nonlinearities // American Control Conference. IEEE, 1987. P. 1034–1039. https://doi.org/10.23919/ACC.1987.4789464
14. Mofid O., Mobayen S. Adaptive finite-time backstepping global sliding mode tracker of quad-rotor UAVs under model uncertainty, wind perturbation, and input saturation // IEEE Transactions on Aerospace and Electronic Systems. 2022. V. 58. N 1. P. 140–151. https://doi.org/10.1109/TAES.2021.3098168
15. Chen X., Jia Y., Matsuno F. Tracking control for differential-drive mobile robots with diamond-shaped input constraints // IEEE Transactions on Control Systems Technology. 2014. V. 22. N 5. P. 1999–2006. https://doi.org/10.1109/TCST.2013.2296900
16. Yang C., Huang D., He W., Cheng L. Neural control of robot manipulators with trajectory tracking constraints and input saturation // IEEE Transactions on Neural Networks and Learning Systems. 2021. V. 32. N 9. P. 4231–4242. https://doi.org/10.1109/TNNLS.2020.3017202
17. Gao Y.-F., Sun X.-M., Wen C., Wang W. Adaptive tracking control for a class of stochastic uncertain nonlinear systems with input saturation // IEEE Transactions on Automatic Control. 2017. V. 62. N 5. P. 2498–2504. https://doi.org/10.1109/TAC.2016.2600340
18. Levant A. Higher-order sliding modes, differentiation and output-feedback control // International Journal of Control. 2003. V. 76. N 9-10. P. 924–941. https://doi.org/10.1080/0020717031000099029

Authors

Chen Zhiqiang — PhD Student, ITMO University, Saint Petersburg, 197101, Russian Federation, SC 58181996400, https://orcid.org/0009-0007-6813-2287, Snowchen612@outlook.com
Aleksandr Yu. Krasnov — PhD, Lecturer, ITMO University, Saint Petersburg, 197101, Russian Federation, SC 55355811700, https://orcid.org/0000-0001-6026-6706, aykrasnov@itmo.ru

Liao Duzhesheng — PhD Student, ITMO University, Saint Petersburg, 197101, Russian Federation, SC 57211507575, https://orcid.org/0009-0000-6389-1355, ldzs2015@gmail.com
Yang Qiusheng — Student, ITMO University, Saint Petersburg, 197101, Russian Federation, https://orcid.org/0009-0007-5839-0258, 1799797481@qq.com

Авторы

Чжицян Чэнь — аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, SC 58181996400, https://orcid.org/0009-0007-6813-2287, Snowchen612@outlook.com
Краснов Александр Юрьевич — кандидат технических наук, преподаватель, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, SC 55355811700, https://orcid.org/0000-0001-6026-6706, aykrasnov@itmo.ru
Дучжешэн Ляо — аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, SC 57211507575, https://orcid.org/0009-0000-6389-1355, ldzs2015@gmail.com
Цюшэн Ян — студент, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, https://orcid.org/0009-0007-5839-0258, 1799797481@qq.com

Received 02.10.2023
Approved after reviewing 08.11.2023
Accepted 25.11.2023

Статья поступила в редакцию 02.10.2023
Одобрена после рецензирования 08.11.2023
Принята к печати 25.11.2023