НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
сентябрь–октябрь 2023 Том 23 № 5 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
September–October 2023 Vol. 23 No 5 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

doi: 10.17586/2226-1494-2023-23-5-904-910

Trajectory tracking control for mobile robots with adaptive gain

Chen Zhiqiang[1]✉, Liao Duzhesheng[2], Alexander Yu. Krasnov[3], Li Yanyu[4]

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

[1] Snowchen612@outlook.com✉, https://orcid.org/0009-0007-6813-2287
[2] ldzs2015@gmail.com, https://orcid.org/0009-0000-6389-1355
[3] aykrasnov@itmo.ru, https://orcid.org/0000-0001-6026-6706
[4] 888liyanyu@gmail.com, https://orcid.org/0009-0008-1776-3895

Abstract
This paper studies the trajectory tracking problem and the controller gain adjustment problem for Wheeled Mobile Robots. The controller gain has a great influence on the robot's trajectory tracking: it can influence both the tracking accuracy and the tracking speed. Therefore, it is very important to choose a suitable control gain during the controller design process. Current neural network gain controllers have a complex structure and require a lot of calculations to find the optimal value. To solve this problem, we design a trajectory tracking controller with a simple structure with adaptive gain by combining the controller with a neural network. The input to this controller is the robot's attitude error. The controller has no hidden layer and directly outputs the trajectory tracking control law. Firstly, the kinematic controller is designed based on Lyapunov function method to ensure that the robot moves according to the reference trajectory. Then, the online gain adjustment algorithm is designed by using neural network to realize the fast adjustment of the controller gain and ensure the reliability of the controller. Finally, the backstepping method is utilized to design the velocity tracking controller based on the error between the virtual velocity and the actual velocity. Considering the influence of the external environment, we also design a nonlinear disturbance observer to estimate the total disturbance on the robot. We perform simulation experiment in MATLAB. The result of the experiment shows that the control algorithm proposed in this paper can realize the accurate tracking of the robot on the specified trajectory. The gain adjustment algorithm we designed can find the optimal gain value quickly and efficiently, thus improving the stability and efficiency of the controller. The method can be applied to most mobile robot trajectory tracking problems and solves the problem of control gain adjustment.

Keywords
wheeled mobile robot, trajectory tracking control, online gain estimation, backstepping method, non-linear disturbance observer

For citation: Zhiqiang C., Duzhesheng L., Krasnov A.Yu., Yanyu L. Trajectory tracking control for mobile robots with adaptive gain. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2023, vol. 23, no. 5, pp. 904–910. doi: 10.17586/2226-1494-2023-23-5-904-910

УДК 62-50

Управление отслеживанием траектории для мобильных роботов с адаптивным коэффициентом усиления

Чэнь Чжицян[1]✉, Ляо Дучжэшэн[2], Александр Юрьевич Краснов[3], Ли Яньюй[4]

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

[1] Snowchen612@outlook.com✉, https://orcid.org/0009-0007-6813-2287
[2] ldzs2015@gmail.com, https://orcid.org/0009-0000-6389-1355
[3] aykrasnov@itmo.ru, https://orcid.org/0000-0001-6026-6706
[4] 888liyanyu@gmail.com, https://orcid.org/0009-0008-1776-3895

Аннотация

Введение. Исследованы задачи слежения за траекторией и настройки коэффициента усиления регулятора для колесных мобильных роботов. Коэффициент усиления регулятора оказывает большое влияние на процесс отслеживания траектории движения робота. Выбор требуемого коэффициента усиления в процессе проектирования регулятора очень важен, поскольку его значение может влиять на точность и скорость отслеживания. Существующие в настоящее время нейросетевые регуляторы коэффициента усиления имеют сложную структуру и требуют для поиска оптимального значения значительных вычислительных ресурсов. Для решения этой проблемы предложен контроллер слежения за траекторией с простой структурой и адаптивным коэффициентом усиления, реализованный путем объединения контроллера с нейронной сетью. Входным сигналом для контроллера служит ошибка ориентации робота. Контроллер не имеет скрытого слоя и напрямую выдает закон управления отслеживанием траектории. **Метод.** На основе метода функций Ляпунова разработан кинематический регулятор, обеспечивающий движение робота по опорной траектории. С помощью нейронной сети предложен алгоритм онлайн-регулировки коэффициента усиления, который позволил ускорить изменение коэффициента усиления регулятора и обеспечил надежность его работы. Для разработки регулятора слежения за скоростью на основе ошибки между виртуальной и реальной скоростью применен метод бэкстеппинга. Для учета влияния внешней среды и оценки суммарных возмущений предложен нелинейный наблюдатель возмущений. **Основные результаты.** Выполнен имитационный эксперимент в среде MATLAB, который показал, что предложенный алгоритм управления позволяет реализовать точное слежение за роботом по заданной траектории. Алгоритм регулировки коэффициента усиления дает возможность быстро и эффективно найти оптимальное значение коэффициента усиления, что повышает устойчивость и эффективность работы регулятора. **Обсуждение.** Метод может найти применение для решения большинства задач слежения за траекторией движения мобильного робота и решает проблему настройки коэффициента усиления управления.

Ключевые слова

колесный мобильный робот, управление отслеживанием траектории, онлайн-оценка параметров усиления, метод backstepping, наблюдатель нелинейных возмущений

Ссылка для цитирования: Чжицян Ч., Дучжэшэн Л., Краснов А.Ю., Яньюй Л. Управление отслеживанием траектории для мобильных роботов с адаптивным коэффициентом усиления // Научно-технический вестник информационных технологий, механики и оптики. 2023. Т. 23, № 5. С. 904–910 (на англ. яз.). doi: 10.17586/2226-1494-2023-23-5-904-910

Introduction

In recent years, along with the improvement of automation and artificial intelligence technology, mobile robots are playing an increasingly important role in our daily life. For example, storage robots and logistics robots can improve the efficiency of goods transport and reduce the consumption of labour. Special robots can replace human to work in extreme environments. Nowadays, scholars have designed many effective controllers to solve the mobile robot tracking control problem.

Mobile robot control problem can be divided into two types: trajectory tracking, path tracking [1–3]. Based on robot model, mobile robot trajectory tracking problem has been studied by many scholars. Initially, Kanayama proposed a non-linear controller based on kinematic model and Lyapunov function [4]. But the performance of the controller depends on the value of the control gain, the designer must make many experiments to find the best gain value. Many subsequent controllers [5–9] were developed based on Kanayama's theory. In practice, it is not enough to consider kinematic model alone. Many mechanical factors (e.g., mass, moment of inertia, kinetic energy) and environmental factors (e.g., wind, ground friction) can influence the accuracy of robot trajectory tracking. This matter has been improved by Jiang and Nijmeijer [10, 11]. They proposed a state feedback controller based on the backstepping technique which is taken into account both the kinematic and dynamic models when designing the controller. In addition, many articles made the following assumptions: (1) The mass centre of the robot coincides with the geometric centre. (2) The mass of the robot is known. In practice, however, these factors will produce errors and the parameters of the model will become uncertain. For dealing with uncertain problem, Fukao proposed an adaptive tracking controller [12] which can estimate the unknown parameters of robot model.

Although many controllers have successfully solved robot trajectory tracking problem, they all face the problem of adjusting the control gain parameters. In other areas of automation, gain adjustment already has an answer. In PID control, many scholars have already proposed methods for control gain adjustment [13–17]. Currently, intelligent algorithms, neural network and fuzzy logic approaches are the main methods for adjusting the gains. For example, a particle swarm optimisation method was proposed in the literature [16]. The work [15] used a fuzzy logic approach to adjust the gain of a PID controller. The paper [17] used neural networks for the adjustment of robot gain parameters.

In this paper, inspired by the neural network framework, we propose a trajectory tracking controller with control gain adjustment. Firstly, in reference to Kanayama's method, we designed a kinematic controller with an on-line gain adjuster based on neural network. Secondly, a dynamic controller is designed by using the backstepping method. We also created a non-linear disturbance observer for estimating total disturbance. Finally, numerical experiment we simulated in MATLAB. The simulation results illustrate the effectiveness of the control algorithm.

Robot model

The Wheeled Mobile Robot (WMR) is shown in Fig. 1, $\{X, O, Y\}$ is the global coordinate system and $\{x, o, y\}$ is the local coordinate system of the robot. R is the robot wheel radius and $2B$ is the distance between the left and right drive wheels of the robot. θ is the heading angle of the robot, ν is the velocity of motion of the robot's centre of mass and ω is the angular velocity of the robot's mass centre. The posture of the robot can be expressed as $\mathbf{q}^T = (X, Y, \theta)$, where (X, Y) is the position of the robot mass center in the global coordinate system, V_L and V_R are the linear velocities of the left and right drive wheels of the robot.

The robot constrained function is

$$\dot{X}\sin\theta - \dot{Y}\cos\theta = 0.$$

The kinematic model $\mathbf{q}$ of the robot can be expressed as

$$\dot{\mathbf{q}} = \mathbf{S}(\theta)\mathbf{V} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \nu \\ \omega \end{pmatrix}. \quad (1)$$

In formula (1) $\mathbf{S}(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \\ 0 & 1 \end{pmatrix}$, $\mathbf{V} = \begin{pmatrix} \nu \\ \omega \end{pmatrix}$ is speed vector of robot.

The dynamic model of the robot is given by

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q, \dot{q})\dot{q} + \mathbf{F}(q) + \mathbf{G}(q) + \boldsymbol{\tau}_d = \\ = \mathbf{B}(q)\boldsymbol{\tau} - \mathbf{A}^T(q)\boldsymbol{\lambda}, \quad (2)$$

where $\mathbf{M}(q)$ is the mass matrix of the robot; $\mathbf{C}(q, \dot{q})$ is Coriolis matrix; $\mathbf{F}(q)$ is friction vector; $\mathbf{G}(q)$ is gravity vector; $\boldsymbol{\tau}_d$ is a vector which represents the external environmental disturbance; $\mathbf{B}(q)$ is an input matrix; $\boldsymbol{\tau}$ is the input torque; $\mathbf{A}^T(q)$ is a full-rank matrix; $\boldsymbol{\lambda}$ is the Lagrange multiplier. The constrained equation is

$$\mathbf{A}(q)\dot{q} = 0.$$

From equations (1) and (2) we can obtain

$$\mathbf{A}(q)\mathbf{S}(q) = 0.$$

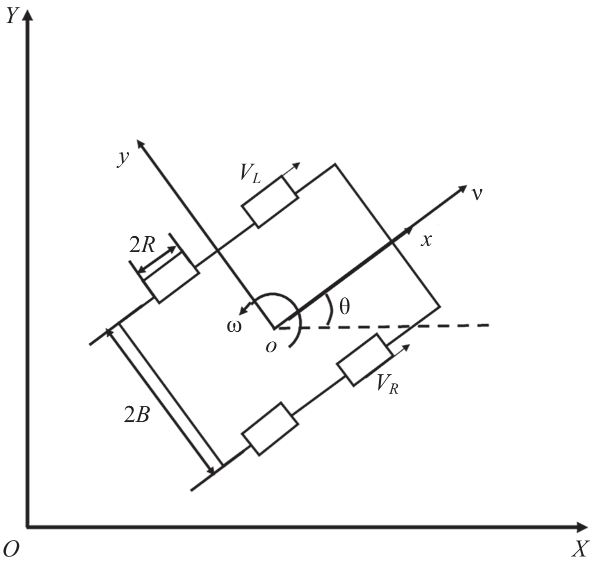

Fig. 1. The wheeled mobile robot

Taking the derivative of equation (1) and bringing the result into equation (2), multiplying both sides of the equation by $\mathbf{S}^T(q)$ get

$$\mathbf{S}^T\mathbf{M}\mathbf{S}\dot{\mathbf{V}} + \mathbf{S}^T(\mathbf{M}\dot{\mathbf{S}} + \mathbf{C}\mathbf{S})\mathbf{V} + \mathbf{S}^T(\mathbf{F} + \mathbf{G} + \boldsymbol{\tau}_d) = \mathbf{S}^T\mathbf{B}\boldsymbol{\tau}.$$

Let $\mathbf{S}^T\mathbf{M}\mathbf{S} = \overline{\mathbf{M}}$, $\mathbf{S}^T(\mathbf{M}\dot{\mathbf{S}} + \mathbf{C}\mathbf{S}) = \overline{\mathbf{C}}$, $\boldsymbol{\delta} = \mathbf{S}^T(\mathbf{F} + \mathbf{G} + \boldsymbol{\tau}_d)$ is the disturbance of robot, $\mathbf{S}^T\mathbf{B} = \overline{\mathbf{B}}$. Simplifying the equation, we get

$$\overline{\mathbf{M}}\dot{\mathbf{V}} + \overline{\mathbf{C}}\mathbf{V} + \boldsymbol{\delta} = \overline{\mathbf{B}}\boldsymbol{\tau}. \quad (3)$$

From the paper [18] the matrices take the form:

$$\mathbf{M}(q) = \begin{pmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{pmatrix}, \mathbf{C}(q, \dot{q}) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{B}(q) = \frac{1}{R}\begin{pmatrix} \cos\theta & \cos\theta \\ \sin\theta & \sin\theta \\ B & -B \end{pmatrix}, \mathbf{G}(q) = 0.$$

So that we can obtain

$$\overline{\mathbf{M}}(q) = \begin{pmatrix} m & 0 \\ 0 & I \end{pmatrix}, \overline{\mathbf{B}}(q) = \frac{1}{R}\begin{pmatrix} 1 & 1 \\ B & -B \end{pmatrix}, \overline{\mathbf{C}}(q, \dot{q}) = 0.$$

Kinematic controller design

The reference trajectory of the robot can be expressed in the form of a matrix as

$$\dot{\mathbf{q}}_r = \begin{pmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\theta}_r \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ -\sin\theta & \cos\theta \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \nu_r \\ \omega_r \end{pmatrix}.$$

According to [10], the posture error of a mobile robot is

$$\begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_r & -X \\ Y_r & -Y \\ \theta_r & -\theta \end{pmatrix}. \quad (4)$$

Taking the derivative of (4) yields

$$\begin{cases} \dot{e}_1 = -\nu + \omega e_2 + \nu_r \cos e_3 \\ \dot{e}_2 = -\omega e_1 + \nu_r \sin e_3 \\ \dot{e}_3 = \omega_r - \omega \end{cases}. \quad (5)$$

The goal of the kinematic controller is to find suitable ν and ω, ensuring that posture error converges to zero. According to Kanayama's method [4], the velocity and angular velocity of the robot were chosen as

$$\begin{cases} \nu = k_1 e_1 + \nu_r \cos e_3 \\ \omega = \omega_r + \nu_r(k_2 e_2 + k_3 \sin e_3) \end{cases}, \quad (6)$$

where $k_1, k_2, k_3 > 0$ are control gains.

Selecting the Lyapunov function as

$$J_1 = \frac{1}{2}(e_1^2 + e_2^2) + \frac{1 - \cos e_3}{k_2} \geq 0. \quad (7)$$

Taking the derivative of (7) and bringing in (5) and (6), we obtain

$$\dot{J}_1 = e_1\dot{e}_1 + e_2\dot{e}_2 + \frac{\dot{e}_3 \mathrm{sin}e_3}{k_2} =$$
$$= e_1(-\nu + \omega e_2 + \nu_r \mathrm{cos}e_3) + e_2(-\omega e_1 + \nu_r \mathrm{sin}e_3) +$$
$$+ \mathrm{sin}e_3 \frac{\omega_r - \omega}{k_2} = -k_1 e_1^2 - \nu_r \frac{k_3}{k_2}\mathrm{sin}^2 e_3 \le 0.$$

According to Lyapunov's stability theorem, the system is stable. The robot posture error will almost globally converge to zero following LaSalle invariance principle.

Online control gain estimation

In this section, we use neural network for control gain calculation. The paper [17] shows that neural network can find the optimal value of controller gain by minimizing the cost function. In this paper, a simple neural network structure is designed. It is divided into two parts, each consisting of one neuron, and it is trained by a back-propagation method. Compared to traditional multilayer neural networks, our method has a simple structure, it can reduce computational complexity and ensure the reliability of the controller.

Define the output of the neural network as

$$\mathbf{V}_c = \begin{pmatrix} \nu_c \\ \omega_c \end{pmatrix} = \begin{pmatrix} k_1(t)e_1 + \nu_r \mathrm{cos}e_3 \\ \omega_r + \nu_r(k_2(t)\, e_2 + k_3(t)\mathrm{sin}e_3) \end{pmatrix}, \qquad (8)$$

where $k_1(t)$, $k_2(t)$, $k_3(t) > 0$ are time varying gains, $\mathbf{V}_c$ is the output control law. Select the sigmoid function for activation of the neural network

$$g(x) = \frac{\mathrm{exp}^{-2xa} - 1}{a(\mathrm{exp}^{-2xa} + 1)},$$

where a is a parameter that affects the shape of the sigmoid function, x is the input value of function. Cost function can be chosen as

$$J = \frac{1}{2}(e_1^2 + e_2^2).$$

Based on the gradient descent method, the controller gain can be updated by the following equations:

$$\begin{cases} k_1' = k_1 + \dot{k}_1 T \\ k_2' = k_2 + \dot{k}_2 T \\ k_3' = k_3 + \dot{k}_3 T \end{cases},$$

where T is the iteration interval time, $\dot{k}_{i=1,2,3}$ is gain update rate, which can be calculated by the following equations:

$$\dot{k}_1 = -\lambda_1 \frac{\partial J}{\partial k_1} = -\lambda_1 \frac{\partial J}{\partial e_1}\frac{\partial e_1}{\partial \beta_1}\frac{\partial \beta_1}{\partial \nu_c}\frac{\partial \nu_c}{\partial k_1} = -\lambda_1 e_1^2 \dot{g}(\nu),$$
$$\dot{k}_2 = -\lambda_2 \frac{\partial J}{\partial k_2} = -\lambda_2 \frac{\partial J}{\partial e_2}\frac{\partial e_2}{\partial \beta_2}\frac{\partial \beta_2}{\partial \omega_c}\frac{\partial \omega_c}{\partial k_2} = -\lambda_2 e_2^2 \dot{g}(\omega),$$
$$\dot{k}_3 = -\lambda_3 \frac{\partial J}{\partial k_3} = -\lambda_3 \frac{\partial J}{\partial e_2}\frac{\partial e_2}{\partial \beta_2}\frac{\partial \beta_2}{\partial \omega_c}\frac{\partial \omega_c}{\partial k_3} = \lambda_3 e_2 \mathrm{sin}e_3 \nu_r \dot{g}(\omega),$$

where $\lambda_{i=1,2,3} > 0$ are learning rates, $\beta_{i=1,2}$ are the inputs of the sigmoid function. The derivative of the sigmoid function is

$$\dot{g}(x) = \frac{4\mathrm{exp}^{-2xa}}{(\mathrm{exp}^{-2xa} + 1)^2}.$$

The structure of control is shown in Fig. 2.

Dynamics controller design

Assumption 1. The mass and rotational inertia of the robot are known.

Assumption 2. The total disturbance of robot is a bounded continuous function.

Defining $\boldsymbol{\eta} = (\eta_1, \eta_2)^T$ as the error between the virtual speed of the robot and the actual speed. It can be calculated by the following equation:

$$\boldsymbol{\eta} = \mathbf{V} - \mathbf{V}_c = \begin{pmatrix} \nu - \nu_c \\ \omega - \omega_c \end{pmatrix}, \qquad (9)$$

where $\mathbf{V}_c = (\nu_c, \omega_c)^T$ is the virtual speed, which is output by the neural network. $\mathbf{V} = (\nu, \omega)^T$ is the real speed of robot.

Bringing the speed error (9) into dynamic model (3) gives

$$\overline{\mathbf{M}}\dot{\boldsymbol{\eta}} = \overline{\mathbf{B}}\boldsymbol{\tau} - \overline{\mathbf{M}}\dot{\mathbf{V}}_c - \boldsymbol{\delta} = \overline{\mathbf{B}}\boldsymbol{\tau} - \mathbf{F}, \qquad (10)$$

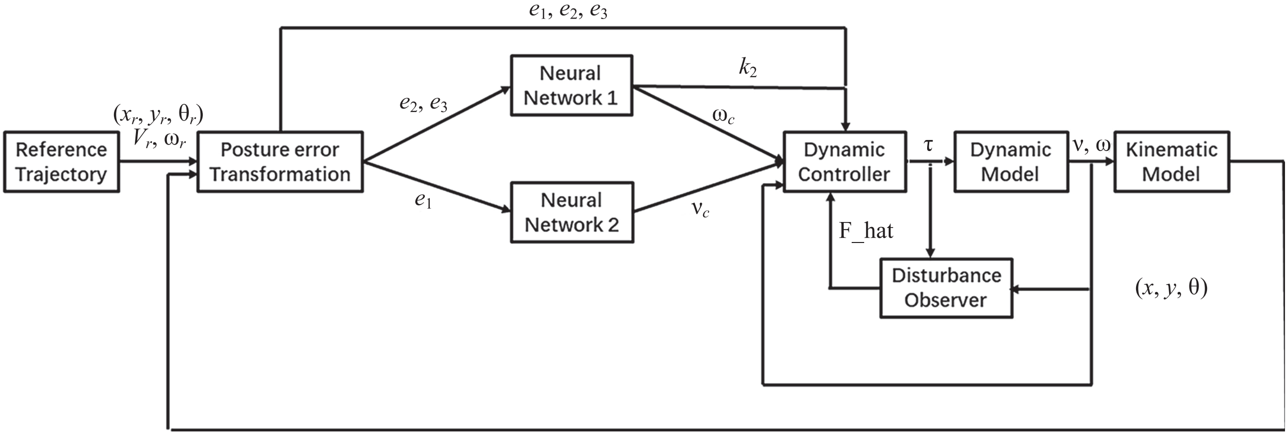

Fig. 2. The structure of control

where $\mathbf{F} = (F_1, F_2)^T = \overline{\mathbf{M}}\dot{\mathbf{V}}_c + \boldsymbol{\delta}$ is all the factors that cause the trajectory tracking error, should be estimated by observer.

Inspired by [19], the non-linear disturbance observer can be designed as follows:

$$\begin{cases} \dot{\mathbf{z}} = -\mathbf{L}\overline{\mathbf{M}}^{-1}(\mathbf{z} - \mathbf{L}\mathbf{V} - \overline{\mathbf{B}}\boldsymbol{\tau}) \\ \hat{\mathbf{F}} = \mathbf{z} - \mathbf{L}\mathbf{V} \end{cases}, \quad (11)$$

where matrix $\mathbf{L}$ is observer gain, $\mathbf{z}$ is a vector stands for observer state variable, $\hat{\mathbf{F}}$ is the estimation of total disturbance.

After the total disturbance compensation, we can get the control torque

$$\boldsymbol{\tau} = \overline{\mathbf{B}}^{-1}\left(-k_4\boldsymbol{\eta} - k_5\mathrm{sign}(\boldsymbol{\eta}) + \begin{pmatrix} e_1 \\ \frac{\sin e_3}{k_2} \end{pmatrix} + \hat{\mathbf{F}}\right), \quad (12)$$

where k_4, $k_5 > 0$ are control gains, sign(•) is the symbolic function.

Defining disturbance observation error as

$$\tilde{\mathbf{F}} = \hat{\mathbf{F}} - \mathbf{F}. \quad (13)$$

From equations (10), (11), (13) we can get

$$\dot{\tilde{\mathbf{F}}} = \dot{\hat{\mathbf{F}}} - \dot{\mathbf{F}} = \dot{\mathbf{z}} - \mathbf{L}\dot{\mathbf{V}} - \dot{\mathbf{F}} =$$
$$= -\mathbf{L}\overline{\mathbf{M}}^{-1}(\mathbf{z} - \mathbf{L}\mathbf{V} - \overline{\mathbf{B}}\boldsymbol{\tau}) - \mathbf{L}\dot{\mathbf{V}} - \dot{\mathbf{F}} =$$
$$= -\mathbf{L}\overline{\mathbf{M}}^{-1}(\mathbf{z} - \mathbf{L}\mathbf{V} - \overline{\mathbf{M}}\dot{\mathbf{V}} - \mathbf{F}) - \mathbf{L}\dot{\mathbf{V}} - \dot{\mathbf{F}} = -\mathbf{L}\overline{\mathbf{M}}^{-1}\tilde{\mathbf{F}} - \dot{\mathbf{F}}.$$

Obviously, choosing suitable observer gain $\mathbf{L}$ can make $-\mathbf{L}\overline{\mathbf{M}}^{-1}$ is a Hurwitz matrix, so that the observation error will asymptotically converge to zero.

Choosing the Lyapunov function as

$$V = J_1 + J_2 = \frac{1}{2}(e_1^2 + e_2^2) + \frac{1 - \cos e_3}{k_2} + \frac{1}{2}\boldsymbol{\eta}^T\overline{\mathbf{M}}\boldsymbol{\eta} \geq 0. \quad (14)$$

Finding derivative of (14) and combining equation (8), (9), (10), (12) we can obtain

$$\dot{V} = \dot{J}_1 + \dot{J}_2 = e_1\dot{e}_1 + e_2\dot{e}_2 + \frac{\dot{e}_3\sin e_3}{k_2} + \boldsymbol{\eta}^T\overline{\mathbf{M}}\dot{\boldsymbol{\eta}} =$$
$$= e_1(-v + \omega e_2 + v_r\cos e_3) + e_2(-\omega e_1 + v_r\sin e_3) +$$
$$+ \sin e_3\frac{\omega_r - \omega}{k_2} + \boldsymbol{\eta}^T\overline{\mathbf{M}}\dot{\boldsymbol{\eta}} = e_1(-(v_c + \eta_1) + \omega e_2 + v_r\cos e_3) +$$
$$+ e_2(-\omega e_1 + v_r\sin e_3) + \sin e_3\frac{\omega_r - (\omega_c + \eta_2)}{k_2} + \boldsymbol{\eta}^T\overline{\mathbf{M}}\dot{\boldsymbol{\eta}} =$$
$$= e_1(-v_c + v_r\cos e_3) + e_2 v_r\sin e_3 + \sin e_3\frac{\omega_r - (\omega_c + \eta_2)}{k_2} -$$
$$- \boldsymbol{\eta}^T\begin{pmatrix} e_1 \\ \frac{\sin e_3}{k_2} \end{pmatrix} + \boldsymbol{\eta}^T\overline{\mathbf{M}}\dot{\boldsymbol{\eta}} = -k_1e_1^2 - v_r\frac{k_3}{k_2}\sin^2 e_3 +$$
$$+ \boldsymbol{\eta}^T\left(\overline{\mathbf{B}}\boldsymbol{\tau} - \mathbf{F} - \begin{pmatrix} e_1 \\ \frac{\sin e_3}{k_2} \end{pmatrix}\right) = -k_1e_1^2 - v_r\frac{k_3}{k_2}\sin^2 e_3 +$$
$$+ \boldsymbol{\eta}^T(-k_4\boldsymbol{\eta} - k_5\mathrm{sign}(\boldsymbol{\eta}) + \tilde{\mathbf{F}}) \leq -k_1e_1^2 - v_r\frac{k_3}{k_2}\sin^2 e_3 -$$
$$- k_4\boldsymbol{\eta}^T\boldsymbol{\eta} - \|\boldsymbol{\eta}\|(k_5 - \|\tilde{\mathbf{F}}\|).$$

We can choose $k_5 > \|\tilde{\mathbf{F}}\|$ to guarantee $\dot{V} \leq 0$. According to the backstepping method and Lyapunov's stability principle, the system is stable and the robot posture error will asymptotically converge to zero.

Simulation

In this section, we give an example of simulation. The parameters of robot are $m = 1$ kg, $I = 2$ kg·m^2, $R = 0.1$ m, $B = 0.3$ m. Selecting reference speed as $v_r = 1$m/s, $\omega = 1$ rad/s. Initial position of robot $(X(0)\ \ Y(0)\ \ \theta(0)) = (1.2\ \ -1\ \ 0)$. The neural network parameters are set as follows $a = 1$, $\lambda_1 = 20$, $\lambda_2 = \lambda_3 = 1$. The initial value of the controller gain is $k_1 = k_2 = k_3 = k_4 = k_5 = 5$. The observer gain is $\mathbf{L} = \begin{pmatrix} 50 & 0 \\ 0 & 50 \end{pmatrix}$, the disturbance be expressed as

$$\boldsymbol{\delta} = \begin{pmatrix} \sqrt{1 + \sin^2(t)} \\ \cos(3t) + 2\sin(4t) \end{pmatrix}.$$

The simulation results are shown on Fig. 3–6.

As can be seen in Fig. 3 and 4, the robot can follow the specified trajectory and posture error converges asymptotically to zero. Fig. 5 shows the process where the gain regulator adjusts the control gains. When the robot motion is stable, the values of the three gains converge to a constant value. Fig. 6 shows the process of estimating the total disturbance by observer. It can be seen that the observer can accurately estimate the perturbations. Using the results of the observations, precise disturbance compensation of the robot can be achieved.

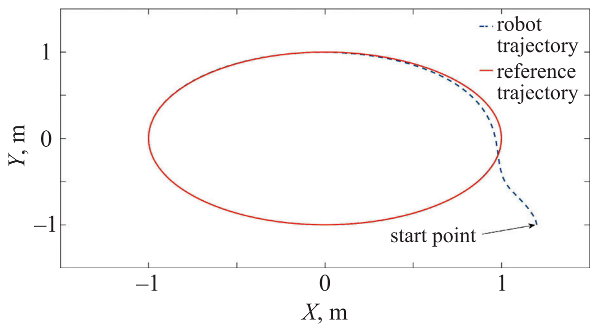

Fig. 3. The trajectory of robot

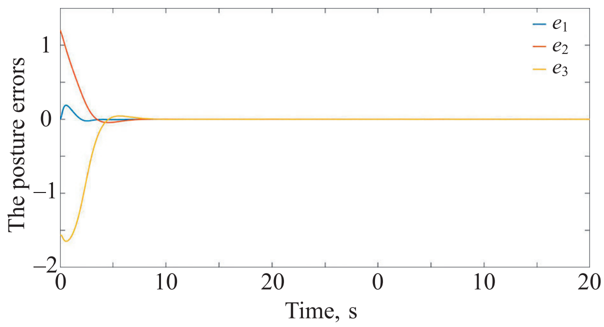

Fig. 4. The posture error of robot

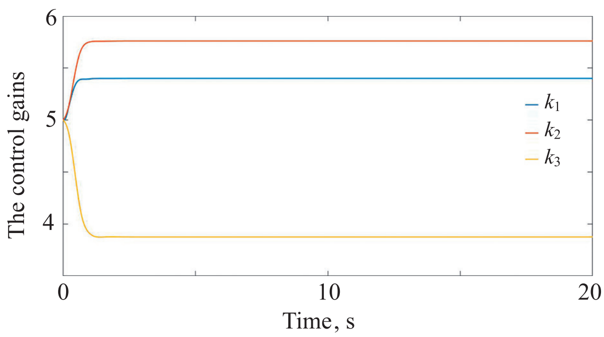

Fig. 5. The estimation of control gain

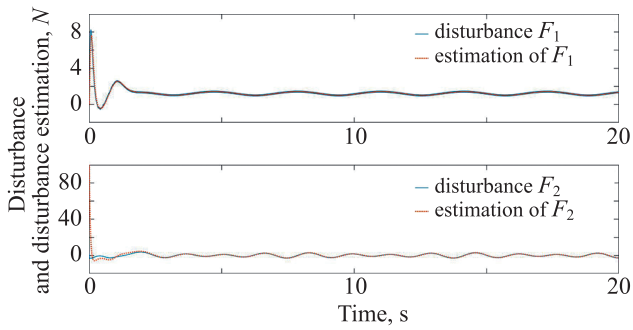

Fig. 6. The disturbance estimated by observer

Conclusion

In this paper, we propose a novel robot trajectory tracking controller with an online gain regulator. Firstly, in the kinematic part, a kinematic controller is designed based on Kanayama's method. After that, we combine the controller with a neural network to design a simple trajectory tracking controller with adaptive gain. Next, in the dynamics part, a dynamic controller was designed based on the error between the actual velocity and the virtual velocity. And a nonlinear disturbance observer was created to estimate the total disturbance. Also, the disturbance compensation is applied to the robot. Finally, simulation experiment is carried out in MATLAB. The experiments show that the controller we designed can effectively and quickly adjust the controller gain and improve its efficiency and stability.

Traditional parameter tuning algorithms treat the controller and the parameter tuner as two individual parts of the design. Neural networks have a complex structure with one or more hidden layers and require a certain amount of computation to find the optimal gain. Our controller structure is much simpler. The input of the controller is the robot pose error, there is no hidden layer, and the trajectory tracking control law of the robot is directly output. This algorithm reduces the computational complexity and is able to find the optimal gain more quickly.

However, the algorithm proposed in this paper still has some limitations. In actual, the center of mass and the center of form of the robot do not coincide, and there will be some errors in the mass. All these factors will bring uncertainty to the robot model. In addition, wheel slippage can have a significant impact on trajectory tracking. How to control the trajectory tracking of the robot under the uncertainty of the model parameters and considering the robot wheel slippage is the topic of our further research.

Reference

1. Xiao X., Liu B., Warnell G., Stone P. Motion planning and control for mobile robot navigation using machine learning: a survey. *Autonomous Robots*, 2022, vol. 46, no. 5, pp. 569–597. https://doi.org/10.1007/s10514-022-10039-8
2. Tzafestas S.G. Mobile robot control and navigation: A global overview. *Journal of Intelligent & Robotic Systems*, 2018, vol. 91, no. 1, pp. 35–58. https://doi.org/10.1007/s10846-018-0805-9
3. Pandey A., Pandey S., Parhi D.R. Mobile robot navigation and obstacle avoidance techniques: A review. *International Robotics & Automation Journal*, 2017, vol. 2, no. 3, pp. 22. https://doi.org/10.15406/iratj.2017.02.00023
4. Kanayama Y., Kimura Y., Miyazaki F., Noguchi T. A stable tracking control method for an autonomous mobile robot. *Proc. of the IEEE International Conference on Robotics and Automation*, 1990, pp. 384–389. https://doi.org/10.1109/ROBOT.1990.126006
5. Sun S. Designing approach on trajectory-tracking control of mobile robot. *Robotics and Computer-Integrated Manufacturing*, 2005, vol. 21, no. 1, pp. 81–85. https://doi.org/10.1016/j.rcim.2004.04.002
6. Xin L., Wang Q., She J., Li Y. Robust adaptive tracking control of wheeled mobile robot. *Robotics and Autonomous Systems*, 2016, vol. 78, pp. 36–48. https://doi.org/10.1016/j.robot.2016.01.002
7. Huang J., Wen C., Wang W., Jiang Z.-P. Adaptive output feedback tracking control of a nonholonomic mobile robot. *Automatica*, 2014, vol. 50, no. 3, pp. 821–831. https://doi.org/10.1016/j.automatica.2013.12.036
8. Shojaei K., Shahri A.M., Tarakameh A., Tabibian B. Adaptive trajectory tracking control of a differential drive wheeled mobile robot. *Robotica*, 2011, vol. 29, no. 3, pp. 391–402. https://doi.org/10.1017/S0263574710000202
9. Huang J., Wen C., Wang W., Jiang Z.-P. Adaptive stabilization and tracking control of a nonholonomic mobile robot with input saturation

Литература

1. Xiao X., Liu B., Warnell G., Stone P. Motion planning and control for mobile robot navigation using machine learning: a survey // Autonomous Robots. 2022. V. 46. N 5. P. 569–597. https://doi.org/10.1007/s10514-022-10039-8
2. Tzafestas S.G. Mobile robot control and navigation: A global overview // Journal of Intelligent & Robotic Systems. 2018. V. 91. N 1. P. 35–58. https://doi.org/10.1007/s10846-018-0805-9
3. Pandey A., Pandey S., Parhi D.R. Mobile robot navigation and obstacle avoidance techniques: A review // International Robotics & Automation Journal. 2017. V. 2. N 3. P. 22. https://doi.org/10.15406/iratj.2017.02.00023
4. Kanayama Y., Kimura Y., Miyazaki F., Noguchi T. A stable tracking control method for an autonomous mobile robot // Proc. of the IEEE International Conference on Robotics and Automation. 1990. P. 384–389. https://doi.org/10.1109/ROBOT.1990.126006
5. Sun S. Designing approach on trajectory-tracking control of mobile robot // Robotics and Computer-Integrated Manufacturing. 2005. V. 21. N 1. P. 81–85. https://doi.org/10.1016/j.rcim.2004.04.002
6. Xin L., Wang Q., She J., Li Y. Robust adaptive tracking control of wheeled mobile robot // Robotics and Autonomous Systems. 2016. V. 78. P. 36–48. https://doi.org/10.1016/j.robot.2016.01.002
7. Huang J., Wen C., Wang W., Jiang Z.-P. Adaptive output feedback tracking control of a nonholonomic mobile robot // Automatica. 2014. V. 50. N 3. P. 821–831. https://doi.org/10.1016/j.automatica.2013.12.036
8. Shojaei K., Shahri A.M., Tarakameh A., Tabibian B. Adaptive trajectory tracking control of a differential drive wheeled mobile robot // Robotica. 2011. V. 29. N 3. P. 391–402. https://doi.org/10.1017/S0263574710000202
9. Huang J., Wen C., Wang W., Jiang Z.-P. Adaptive stabilization and tracking control of a nonholonomic mobile robot with input saturation

and disturbance. *Systems & Control Letters*, 2013, vol. 62, no. 3, pp. 234–241. https://doi.org/10.1016/j.sysconle.2012.11.020

10. Jiang Z.P., Nijmeijer H. Tracking control of mobile robots: A case study in backstepping. *Automatica*, 1997, vol. 33, no. 7, pp. 1393–1399. https://doi.org/10.1016/S0005-1098(97)00055-1
11. Fierro R., Lewis F.L. Control of a nonholomic mobile robot: Backstepping kinematics into dynamics. *Journal of Robotic Systems*, 1997, vol. 14, no. 3, pp. 149–163. https://doi.org/10.1002/(SICI)1097-4563(199703)14:3<149::AID-ROB1>3.0.CO;2-R
12. Fukao T., Nakagawa H., Adachi N. Adaptive tracking control of a nonholonomic mobile robot. *IEEE Transactions on Robotics and Automation*, 2000, vol. 16, no. 5, pp. 609–615. https://doi.org/10.1109/70.880812
13. Mudi R.K., Pal N.R. A self-tuning fuzzy PI controller. *Fuzzy Sets and Systems*, 2000, vol. 115, no. 2, pp. 327–338. https://doi.org/10.1016/S0165-0114(98)00147-X
14. Le T.D., Kang H.J., Suh Y.S., Ro Y.-S. An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-dof parallel manipulator. *Neurocomputing*, 2013, vol. 116, pp. 53–61. https://doi.org/10.1016/j.neucom.2012.01.047
15. Zong Q., Zhao Z.S., Zhang J. Higher order sliding mode control with self-tuning law based on integral sliding mode. *IET Control Theory & Applications*, 2010, vol. 4, no. 7, pp. 1282–1289. https://doi.org/10.1049/iet-cta.2008.0610
16. Sungthong A., Assawinchaichote W. Particle swam optimization based optimal PID parameters for air heater temperature control system. *Procedia Computer Science*, 2016, vol. 86, pp. 108–111. https://doi.org/10.1016/j.procs.2016.05.027
17. Yamada T., Yabuta T. Neural network controller using autotuning method for nonlinear functions. *IEEE Transactions on Neural Networks*, 1992, vol. 3, no. 4, pp. 595–601. https://doi.org/10.1109/72.143373
18. Zeng W., Wang Q., Liu F., Wang Y. Learning from adaptive neural network output feedback control of a unicycle-type mobile robot. *ISA Transactions*, 2016, vol. 61, pp. 337–347. https://doi.org/10.1016/j.isatra.2016.01.005
19. Chen W.H., Ballance D.J., Gawthrop P.J., O'Reilly J. A nonlinear disturbance observer for robotic manipulators. *IEEE Transactions on Industrial Electronics*, 2000, vol. 47, no. 4, pp. 932–938. https://doi.org/10.1109/41.857974

Authors

Chen Zhiqiang — PhD Student, ITMO University, Saint Petersburg, 197101, Russian Federation, sc 58181996400, https://orcid.org/0009-0007-6813-2287, Snowchen612@outlook.com

Liao Duzhesheng — PhD Student, ITMO University, Saint Petersburg, 197101, Russian Federation, sc 57211507575, https://orcid.org/0009-0000-6389-1355, ldzs2015@gmail.com

Alexander Yu. Krasnov — PhD, Lecturer, ITMO University, Saint Petersburg, 197101, Russian Federation, sc 55355811700, https://orcid.org/0000-0001-6026-6706, aykrasnov@itmo.ru

Li Yanyu — Student, ITMO University, Saint Petersburg, 197101, Russian Federation, https://orcid.org/0009-0008-1776-3895, 888liyanyu@gmail.com

Received 29.05.2023
Approved after reviewing 01.08.2023
Accepted 17.09.2023

and disturbance // Systems & Control Letters. 2013. V. 62. N 3. P. 234–241. https://doi.org/10.1016/j.sysconle.2012.11.020

10. Jiang Z.P., Nijmeijer H. Tracking control of mobile robots: A case study in backstepping // Automatica. 1997. V. 33. N 7. P. 1393–1399. https://doi.org/10.1016/S0005-1098(97)00055-1
11. Fierro R., Lewis F.L. Control of a nonholomic mobile robot: Backstepping kinematics into dynamics // Journal of Robotic Systems. 1997. V. 14. N 3. P. 149–163. https://doi.org/10.1002/(SICI)1097-4563(199703)14:3<149::AID-ROB1>3.0.CO;2-R
12. Fukao T., Nakagawa H., Adachi N. Adaptive tracking control of a nonholonomic mobile robot // IEEE Transactions on Robotics and Automation. 2000. V. 16. N 5. P. 609–615. https://doi.org/10.1109/70.880812
13. Mudi R.K., Pal N.R. A self-tuning fuzzy PI controller // Fuzzy Sets and Systems. 2000. V. 115. N 2. P. 327–338. https://doi.org/10.1016/S0165-0114(98)00147-X
14. Le T.D., Kang H.J., Suh Y.S., Ro Y.-S. An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-dof parallel manipulator // Neurocomputing. 2013. V. 116. P. 53–61. https://doi.org/10.1016/j.neucom.2012.01.047
15. Zong Q., Zhao Z.S., Zhang J. Higher order sliding mode control with self-tuning law based on integral sliding mode // IET Control Theory & Applications. 2010. V. 4. N 7. P. 1282–1289. https://doi.org/10.1049/iet-cta.2008.0610
16. Sungthong A., Assawinchaichote W. Particle swam optimization based optimal PID parameters for air heater temperature control system // Procedia Computer Science. 2016. V. 86. P. 108–111. https://doi.org/10.1016/j.procs.2016.05.027
17. Yamada T., Yabuta T. Neural network controller using autotuning method for nonlinear functions // IEEE Transactions on Neural Networks. 1992. V. 3. N 4. P. 595–601. https://doi.org/10.1109/72.143373
18. Zeng W., Wang Q., Liu F., Wang Y. Learning from adaptive neural network output feedback control of a unicycle-type mobile robot // ISA Transactions. 2016. V. 61. P. 337–347. https://doi.org/10.1016/j.isatra.2016.01.005
19. Chen W.H., Ballance D.J., Gawthrop P.J., O'Reilly J. A nonlinear disturbance observer for robotic manipulators // IEEE Transactions on Industrial Electronics. 2000. V. 47. N 4. P. 932–938. https://doi.org/10.1109/41.857974

Авторы

Чэнь Чжицян — аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, sc 58181996400, https://orcid.org/0009-0007-6813-2287, Snowchen612@outlook.com

Ляо Дучжэшэн — аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, sc 57211507575, https://orcid.org/0009-0000-6389-1355, ldzs2015@gmail.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

Ли Яньюй — студент, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, https://orcid.org/0009-0008-1776-3895, 888liyanyu@gmail.com

Статья поступила в редакцию 29.05.2023
Одобрена после рецензирования 01.08.2023
Принята к печати 17.09.2023