TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 19/38 pp1635–1651
DOI: 10.26599/TST.2024.9010194
Volume 31, Number 3, June 2026

Research Article

A Neural-Dynamics-Based Control Scheme for Double-Arm Mobile Robots with Dynamic Neural Network

Diwen Xiong, Liuyi Wen*, and Mingsheng Shang

Abstract: A double-arm mobile robot (DMR) is an advanced robotic system that integrates a mobile platform and double robotic arms. To address the control issue of the DMR, this paper provides its mathematical model and deduces kinematic equations as preliminaries. On this basis, a physically constrained velocity collaborative control (PCVCC) scheme is proposed for kinematic control of the DMR, which adjusts the mobile platform and robotic arms cooperatively using a designed optimization criterion. To explore the optimal solution of the PCVCC scheme, a gradient descent method assisted by velocity compensation is exploited to design a dynamic neural network (DNN) solver. Subsequently, theoretical analyses confirm the global convergence of the DNN solver. Finally, simulations, experiments, and comparisons demonstrate the feasibility and superiority of the proposed method.

Key words: double-arm mobile robot (DMR); kinematic control; physically constrained velocity collaborative control (PCVCC); dynamic neural network (DNN)

1 Introduction

Mobile robotic arms combine a mobile platform with robotic arms, enhancing system redundancy and eliminating location constraints[1, 2]. Additional degrees of freedom (DOFs) improve the flexibility, but simultaneously coordinating the mobile platform and robotic arm presents challenges for redundancy analyses[3]. A series of related kinematic investigations on mobile robotic arms have been carried out and achieved notable outcomes[4–7]. Specifically, a repetitive motion criterion is elaborated for mobile robotic arms to regulate the finished state to the initial one, which realizes the coordinated operation between the mobile platform and the robotic arm[4]. A vector-based constrained obstacle avoidance (VOA) scheme and a primal-dual neural network solver are developed for mobile robotic arms that can avoid obstacles while implementing a given trajectory[5]. Besides, a collaborative control approach of multiple mobile robotic arms is completed in Ref. [6] with estimation and control abilities. However, most existing methods[3–7] on the kinematic control of mobile robotic arms mainly focus on the task execution of the end-effector and neglect the precise control of the mobile platform. This leads to a defect that the movement of the mobile platform is unpredictable and cannot be accurately controlled during task execution.

As a unique type of mobile robotic arm, the double-arm mobile robot (DMR) showcases the capability of double-arm collaborations. Compared to a single-arm robotic system, double-arm collaborations enable the DMR to perform more complex and diverse tasks, such as carrying, assembling[8], and manipulating multiple objects[9]. One practical application example of DMR is the mobile wheelchair with double robotic arms (MWDRA), as illustrated in Fig. 1. The installed robotic arms can be analogized to two human arms.

• Diwen Xiong, Liuyi Wen, and Mingsheng Shang are with Chongqing Institute of Green and Intelligent Technology and Chongqing School, University of Chinese Academy of Sciences (UCAS Chongqing), Chongqing 400714, China. E-mail: xdw@cigit.ac.cn; 283727229@qq.com; msshang@cigit.ac.cn.
* To whom correspondence should be addressed.
Manuscript received: 2024-07-24; revised: 2024-09-10; accepted: 2024-10-14

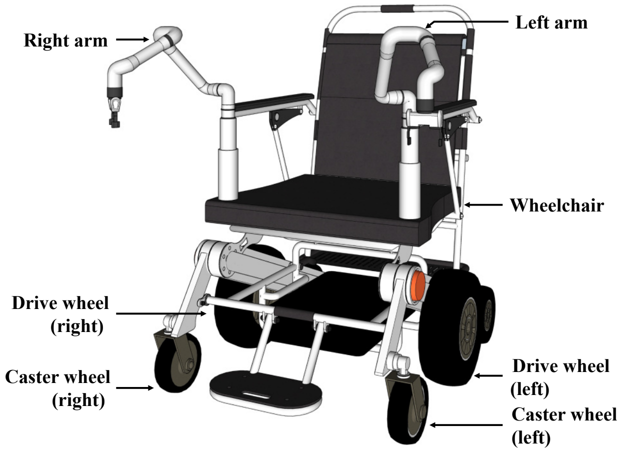


Fig. 1 Schematic of a mobile wheelchair equipped with double robotic arms.

This innovative design enables the MWDRA to substitute the operator for performing simple actions, e.g., capture[10] and placement[11]. Thus, the kinematic control of DMRs is crucial for practical applications. In this regard, recent years have witnessed extensive attention and rapid development in this field[12–18]. A ground adaptability problem of DMRs (legged humanoid robots) is investigated to improve the balance of the robot during walking[12]. Besides, a DMR constituted by a humanoid upper body and a tracked mobile platform is presented in Ref. [13]. It contributes to robotic hardware development for complex rescue scenes. Besides, an effective control method for DMRs is introduced in Ref. [14], realizing the motion control of the mobile platform and the cooperative control of double arms. However, this research implements planning and control processes of the mobile platform and robotic arms separately rather than synchronously. In Ref. [15], a redundancy resolution method for DMRs is developed to fulfill the grasp-and-place task by cooperatively manipulating the robotic system. Subsequent studies on the impedance control and haptic development of DMRs are carried out with important research results[16, 17]. Nevertheless, the aforementioned studies on DMRs are difficult to apply directly to MWDRA, since their robotic models do not reserve enough space for an operator.

Kinematics research of robotic systems is primarily concerned with solving the problems of forward kinematics or inverse kinematics[18, 19]. When handling inverse kinematics, there are infinite possible solutions for redundant robotic systems conducting a pre-design task[20, 21]. It is important to obtain the optimal solution efficiently according to diversified intelligent algorithms that further consider obstacle avoidance[22], human-robot collaboration[23], and noise suppression[24]. The core thought of these intelligent algorithms (such as model predictive control algorithms[25] and impedance controllers[26]) lies in setting the corresponding equilibrium state and guaranteeing the convergence performance of the system under various conditions[7, 27]. While the pseudoinverse method provides a feasible foundation for redundancy analyses of robotic systems, it has limitations[28]. Firstly, it cannot handle inequality constraints, including joint constraints, that are crucial for ensuring the safe operation of robotic systems. Additionally, the pseudoinverse operation may encounter singularity conditions, leading to loss of control. Zeroing neural networks (ZNNs) based on the neural dynamics method solve these shortcomings[29, 30]. However, due to depending on the derivative term, ZNNs are easy to introduce unavoidable noise in hardware implementation[31–33]. In addition, gradient neural networks (GNNs) focus on reducing errors but ignore the time-derivative information, leading to poor performance[34]. As a further improvement, dynamic neural networks (DNNs), a special type of recurrent neural network (RNN)[35, 36], incorporate velocity compensation compensation and obtain satisfactory results[37]. Then, DNNs are applied to kinematic control of robotic systems combined with optimization indexes[38, 39].

Inspired by the deficiencies of the existing research, this paper proposes a physically constrained velocity collaborative control (PCVCC) scheme for MWDRA. With the aid of a designed optimization criterion, this scheme is capable of fulfilling the coordinated control and state adjustment of the MWDRA. The remaining contents of this research can be summarized as follows. Section 2 presents a mathematical model and kinematics analyses for the MWDRA. Then, a PCVCC scheme is proposed as quadratic programming with an optimization criterion developed in Section 3. Next, Section 4 derives a corresponding DNN with its stability proved. Subsequently, in Section 5, we conduct simulations and experiments on the MWDRA and provide performance comparisons against existing methods to demonstrate the feasibility and superiority of our proposed method. Finally, Section 6 concludes the whole paper. Before concluding this section, the contributions of this paper are emphasized below.

- From a kinematic perspective, a specialized DMR is modeled and researched, where robotic arms are installed at two corners of the mobile platform. This mathematical modeling aligns with the structure of MWDRA.
- A new optimization criterion is proposed for the state adjustment of the DMR, utilizing neural dynamics to coordinately adjust the position of the mobile platform, the rotation mode of the drive wheels, and the joints of the robotic arms.
- To achieve the coordinated control of DMR, a PCVCC scheme combined with the designed optimization criterion is proposed with joint constraints considered.
- A DNN solver is constructed using the gradient descent method with velocity compensation. This solver incorporates the physical information of the PCVCC scheme. Furthermore, its convergence is proved theoretically.

2 DRA Model and Kinematic Analysis

This section provides a mathematical model and kinematics analyses for the DRA.

Similar to the MWDRA depicted in Fig. 1, we consider a specialized DMR consisting of a wheeled mobile platform equipped with two drive wheels and two 6-DOF robotic arms. These robotic arms are mounted at two corners of the mobile platform. Furthermore, a top view of the mobile platform is depicted in Fig. 2, and the corresponding parameters are provided in Table 1. Assume that driving wheels only roll without slipping. The motion relationship of the mobile platform in the platform coordinate system is described as[5]

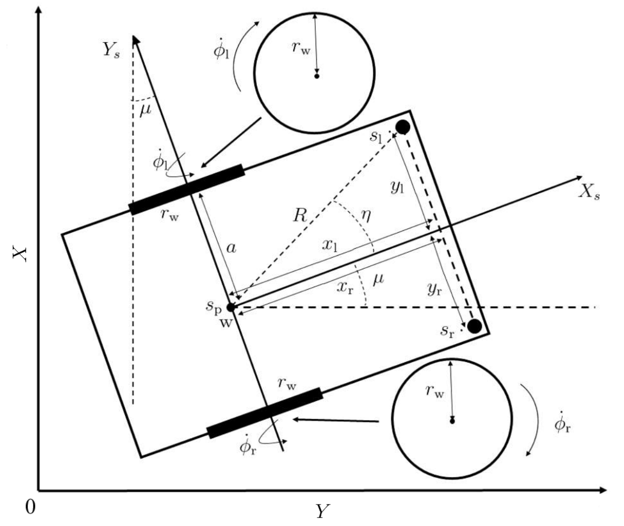


Fig. 2 Geometric construction of the mobile wheelchair with necessary parameters annotated.

Table 1 Parameter settings of mobile wheelchair in Fig. 2.

Parameter	Definition
XYZ	World coordinate system with the origin 0 and X-axis, Y-axis, and Z-axis
$X_sY_sZ_s$	Mobile wheelchair coordinate system with the origin w and coordinate axis X_s-axis, Y_s-axis, Z_s-axis
r_w	Radius of the left and right drive wheels
μ	Heading angle of the mobile wheelchair with heading angular velocity $\dot{\mu}$
$\dot{\phi}_l$, $\dot{\phi}_r$	Rotation velocity of the left and right driving wheels with rotation angles ϕ_l and ϕ_r
s_p	Midpoint of the connection between two drive wheels
a	Distance between s_p and drive wheels
s_l	Connection point between left robotic arm and mobile wheelchair with coordinate $(x^w_{s_l}, y^w_{s_l}, z^w_{s_l})$ in mobile wheelchair coordinate system and $(x^o_{s_l}, y^o_{s_l}, z^o_{s_l})$ in world system
s_r	Connection point between right robotic arm and mobile wheelchair with coordinate $(x^w_{s_r}, y^w_{s_r}, z^w_{s_r})$ in mobile wheelchair coordinate system and $(x^o_{s_r}, y^o_{s_r}, z^o_{s_r})$ in world system
x_l, y_l	Distances from the connection point s_l to the transverse symmetry axis and the longitudinal symmetry axis of the mobile wheelchair
x_r, y_r	Distances from the connection point s_r to the transverse symmetry axis and the longitudinal symmetry axis of the mobile wheelchair
R	Distance between the connection point s_l and point s_p
η	Angle between the connection point s_l and the X_s-axis

$$\dot{x}^w_{s_p} = \frac{r_w}{2}\dot{\phi}_l + \frac{r_w}{2}\dot{\phi}_r \tag{1}$$

$$\dot{y}^w_{s_p} = \dot{z}^w_{s_p} = 0 \tag{2}$$

$$\dot{\mu} = -\frac{r_w}{2a}\dot{\phi}_l + \frac{r_w}{2a}\dot{\phi}_r \tag{3}$$

where $\dot{x}^w_{s_p}$, $\dot{y}^w_{s_p}$, and $\dot{z}^w_{s_p}$ represent the linear velocities of s_p in X_s-axis, Y_s-axis, and Z_s-axis in the wheelchair coordinate system. It is worth pointing out that for simplicity, two connection points s_l and s_r are set as symmetric points for the coordinate X_s axis, i.e., $x_l = x_r$, $y_l = y_r$. Then, their velocities in the wheelchair coordinate system can be expressed as

$$\begin{bmatrix} \dot{s}^w_l \\ \dot{s}^w_r \end{bmatrix} = \begin{bmatrix} \dot{x}^w_{s_l} \\ \dot{y}^w_{s_l} \\ \dot{z}^w_{s_l} \\ \dot{x}^w_{s_r} \\ \dot{y}^w_{s_r} \\ \dot{z}^w_{s_r} \end{bmatrix} = \begin{bmatrix} \dot{x}_{s_p} - \dot{\mu}\cdot R\cdot \sin\eta \\ \dot{y}_{s_p} + \dot{\mu}\cdot R\cdot \cos\eta \\ \dot{z}_{s_p} \\ \dot{x}_{s_p} + \dot{\mu}\cdot R\cdot \sin\eta \\ \dot{y}_{s_p} + \dot{\mu}\cdot R\cdot \cos\eta \\ \dot{z}_{s_p} \end{bmatrix} = \mathcal{A}\begin{bmatrix} \dot{\phi}_l \\ \dot{\phi}_r \end{bmatrix} \tag{4}$$

where $\dot{s}_l^{\mathrm{w}} = [\dot{x}_{s_l}^{\mathrm{w}}, \dot{y}_{s_l}^{\mathrm{w}}, \dot{z}_{s_l}^{\mathrm{w}}]^{\mathrm{T}}$ and $\dot{s}_r^{\mathrm{w}} = [\dot{x}_{s_r}^{\mathrm{w}}, \dot{y}_{s_r}^{\mathrm{w}}, \dot{z}_{s_r}^{\mathrm{w}}]^{\mathrm{T}}$ denote linear velocities of connection point s_l and s_r in the wheelchair coordinate system, respectively. With $R\cdot\cos\eta = y_l = y_r$, $R\cdot\sin\eta = x_l = x_r$, we obtain

$$\mathcal{A} = \begin{bmatrix} \frac{r_w}{2}+\frac{r_w}{2a}y_l & \frac{r_w}{2}-\frac{r_w}{2a}y_l \\ -\frac{r_w}{2a}x_l & \frac{r_w}{2a}x_l \\ 0 & 0 \\ \frac{r_w}{2}-\frac{r_w}{2a}y_r & \frac{r_w}{2}+\frac{r_w}{2a}y_r \\ -\frac{r_w}{2a}x_r & \frac{r_w}{2a}x_r \\ 0 & 0 \end{bmatrix} \in \mathbf{R}^{2n\times 2} \tag{5}$$

where n is the number of positional dimensions of the end-effector of a single robotic arm in space, and it takes a value of 3 in this paper.

For the transformation of the wheelchair coordinate system and the world coordinate system, a rotation matrix is provided as follows:

$$\mathcal{B} = \begin{bmatrix} \cos\mu & -\sin\mu & 0 \\ \sin\mu & \cos\mu & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbf{R}^{3\times 3} \tag{6}$$

As a result, two connection points s_l and s_r in world coordinates can be reformulated as

$$\begin{bmatrix} \dot{s}_l^{\mathrm{o}} \\ \dot{s}_r^{\mathrm{o}} \end{bmatrix} = \begin{bmatrix} \dot{x}_{s_l}^{\mathrm{o}} \\ \dot{y}_{s_l}^{\mathrm{o}} \\ \dot{z}_{s_l}^{\mathrm{o}} \\ \dot{x}_{s_r}^{\mathrm{o}} \\ \dot{y}_{s_r}^{\mathrm{o}} \\ \dot{z}_{s_r}^{\mathrm{o}} \end{bmatrix} = \begin{bmatrix} \mathcal{B} & 0_{3\times 3} \\ 0_{3\times 3} & \mathcal{B} \end{bmatrix} \begin{bmatrix} \dot{s}_l^{\mathrm{w}} \\ \dot{s}_r^{\mathrm{w}} \end{bmatrix} \tag{7}$$

where $\dot{s}_l^{\mathrm{o}} = [\dot{x}_{s_l}^{\mathrm{o}}, \dot{y}_{s_l}^{\mathrm{o}}, \dot{z}_{s_l}^{\mathrm{w}}]^{\mathrm{T}}$ and $\dot{s}_r^{\mathrm{o}} = [\dot{x}_{s_r}^{\mathrm{o}}, \dot{y}_{s_r}^{\mathrm{o}}, \dot{z}_{s_r}^{\mathrm{o}}]^{\mathrm{T}}$ denote the linear velocities of connection point s_l and s_r in the world coordinate system, respectively. Furthermore, a matrix form of Eq. (7) can be reorganized as

$$\begin{bmatrix} \dot{s}_l^{\mathrm{o}} \\ \dot{s}_r^{\mathrm{o}} \end{bmatrix} = \mathcal{C} \begin{bmatrix} \dot{\phi}_l \\ \dot{\phi}_r \end{bmatrix} \tag{8}$$

with matrix $\mathcal{C} \in \mathbf{R}^{6\times 2}$ being

$$\begin{bmatrix} \left(\frac{r_w}{2}+\frac{r_w}{2a}y_l\right)\cos\mu+\frac{r_w}{2a}\sin\mu x_l & \left(\frac{r_w}{2}-\frac{r_w}{2a}y_l\right)\cos\mu-\frac{r_w}{2a}\sin\mu x_l \\ \left(\frac{r_w}{2}+\frac{r_w}{2a}y_l\right)\sin\mu-\frac{r_w}{2a}\cos\mu x_l & \left(\frac{r_w}{2}-\frac{r_w}{2a}y_l\right)\sin\mu+\frac{r_w}{2a}\cos\mu x_l \\ 0 & 0 \\ \left(\frac{r_w}{2}-\frac{r_w}{2a}y_r\right)\cos\mu+\frac{r_w}{2a}\sin\mu x_r & \left(\frac{r_w}{2}+\frac{r_w}{2a}y_r\right)\cos\mu-\frac{r_w}{2a}\sin\mu x_r \\ \left(\frac{r_w}{2}-\frac{r_w}{2a}y_r\right)\sin\mu-\frac{r_w}{2a}\cos\mu x_r & \left(\frac{r_w}{2}+\frac{r_w}{2a}y_r\right)\sin\mu+\frac{r_w}{2a}\cos\mu x_r \\ 0 & 0 \end{bmatrix}.$$

Equation (8) describes the velocity relationship between drive wheels and connection points of the wheelchair. For a 6-DOF robotic arm with a fixed base, calculating its forward kinematics by the homogeneous transformation matrix of each joint yields

$$\begin{bmatrix} \Gamma^{\mathrm{w}}(\theta) \\ 1 \end{bmatrix} = {}^0_1\mathcal{T}\,{}^1_2\mathcal{T}\,{}^2_3\mathcal{T}\,{}^3_4\mathcal{T}\,{}^4_5\mathcal{T}\,{}^5_6\mathcal{T}\,{}^6_{\mathrm{e}}\mathcal{T} \tag{9}$$

where ${}^{i-1}_{i}\mathcal{T} \in \mathbf{R}^{4\times 4}$ records the spatial relationships of the different adjacent joints[40]; ${}^6_{\mathrm{e}}\mathcal{T} \in \mathbf{R}^4$ represents the vector form of end-effector; $\Gamma^{\mathrm{w}}(\theta) \in \mathbf{R}^n$ denotes the nonlinear function related to joint angle $\theta \in \mathbf{R}^m$ and describes the position of the end-effector. When it comes to the MWDRA, $\Gamma_{s_l}^{\mathrm{o}}(\theta_l)$ and $\Gamma_{s_r}^{\mathrm{o}}(\theta_r)$ are designed to determine the forward kinematics of the left and right arms in the world coordinate system, respectively, with joint angles of the left and right arms as θ_l and θ_r, respectively. Analogously, via the homogeneous transformation matrix, the heading angle μ is utilized to integrate the wheelchair and two robotic arms, which produces

$$\begin{bmatrix} \Gamma_{s_l}^{\mathrm{o}}(\theta_l) \\ \Gamma_{s_r}^{\mathrm{o}}(\theta_r) \end{bmatrix} = \begin{bmatrix} \mathcal{B} & s_l^{\mathrm{o}} & 0_{3\times 3} & 0_{3\times 1} \\ 0_{3\times 3} & 0_{3\times 1} & \mathcal{B} & s_r^{\mathrm{o}} \end{bmatrix} \begin{bmatrix} \Gamma_{s_l}^{\mathrm{w}}(\theta_l) \\ 1 \\ \Gamma_{s_r}^{\mathrm{w}}(\theta_r) \\ 1 \end{bmatrix} =$$

$$\begin{bmatrix} \mathcal{B}\Gamma_{s_l}^{\mathrm{w}}(\theta_l) \\ \mathcal{B}\Gamma_{s_r}^{\mathrm{w}}(\theta_r) \end{bmatrix} + \begin{bmatrix} s_l^{\mathrm{o}} \\ s_r^{\mathrm{o}} \end{bmatrix} = r \tag{10}$$

where $r \in \mathbf{R}^{2n}$ stands for the end-effector position vector for double robotic arms; $0_{3\times 3}$ denotes a zero matrix in size 3×3; $0_{3\times 1}$ denotes a zero vector in size 3×1; $\Gamma_{s_l}^{\mathrm{w}}(\theta_l)$ and $\Gamma_{s_r}^{\mathrm{w}}(\theta_r)$ represent forward kinematics of left and right arms and are calculated by Eq. (9), respectively. Note that the distance between s_l^{o} and s_r^{o} equals to $y_l + y_r$ with oblique angle μ. According to Eqs. (3) and (8), taking the time derivative of Eq. (10) results in

$$\dot{r} = \begin{bmatrix} \mathcal{G}_l(\theta_l, \mu) \\ \mathcal{G}_r(\theta_r, \mu) \end{bmatrix} \begin{bmatrix} \dot{\mu} \\ \dot{\theta}_l \\ \dot{\theta}_r \end{bmatrix} + \mathcal{C} \begin{bmatrix} \dot{\phi}_l \\ \dot{\phi}_r \end{bmatrix} =$$

$$\underbrace{\left(\begin{bmatrix} \mathcal{G}_l(\theta_l, \mu) \\ \mathcal{G}_r(\theta_r, \mu) \end{bmatrix} \begin{bmatrix} \mathcal{D} & 0_{1\times 2m} \\ 0_{2m\times 1} & I_{2m} \end{bmatrix} + \begin{bmatrix} \mathcal{C} & 0_{2n\times 2m} \end{bmatrix} \right)}_{\mathcal{K}(\theta_l, \theta_r, \mu)} \begin{bmatrix} \dot{\phi}_l \\ \dot{\phi}_r \\ \dot{\theta}_l \\ \dot{\theta}_r \end{bmatrix},$$

where

$$\mathcal{D} = \left[-\frac{r_w}{2}, \frac{r_w}{2}\right],$$

$$\mathcal{G}_l(\theta_l, \mu) = \frac{\partial \mathcal{B}\Gamma_{s_l}^{\mathrm{w}}(\theta_l)}{\partial[\mu, \theta_l, \theta_r]^{\mathrm{T}}} \in \mathbf{R}^{n\times(2m+1)},$$

$$\mathcal{G}_r(\theta_\mathrm{r},\mu)=\frac{\partial\mathcal{B}\Gamma^{\mathrm{w}}_{s_\mathrm{r}}(\theta_\mathrm{r})}{\partial[\mu,\ \theta_\mathrm{l},\ \theta_\mathrm{r}]^\mathrm{T}}\in\mathbf{R}^{n\times(2m+1)},$$

$\dot{\theta}_\mathrm{l}$ and $\dot{\theta}_\mathrm{r}$ represent joint velocities of left and right arms, respectively; $I_{2m}\in\mathbf{R}^{2m\times 2m}$ is an identity matrix; $0_{2n\times 2m}$, $0_{1\times 2m}$, and $0_{2m\times 1}$ denote zero matrices or zero vectors in appropriate dimensions, and m refers to the number of joint degrees of freedom of a single robotic arm, which in this papaer $m=6$. By defining $\omega=[\dot{\phi}_\mathrm{l},\ \dot{\phi}_\mathrm{r},\ \dot{\theta}_\mathrm{l},\ \dot{\theta}_\mathrm{r}]^\mathrm{T}\in\mathbf{R}^{2m+2}$, the above equation can be simplified as

$$\dot{\mathrm{r}}=\mathcal{K}(\theta_\mathrm{l},\ \theta_\mathrm{r},\ \mu)\omega \tag{11}$$

where $\mathcal{K}\in\mathbf{R}^{2n\times(2m+2)}$ denotes the Jacobian matrix of the DMR. In consequence, the mathematical model of the DMR is derived completely. Notably, the above derivations are inspired by the existing work, which primarily focuses on a wheeled mobile redundant manipulator. However, the main modeling distinction lies in the consideration of a double-arm model in this work. Furthermore, different from the existing research on the double-arm model[8, 9, 12–17], this paper considers a special DMR, where robot arms are installed at the two corners of the mobile platform. This modeling form regards MWDRAs as the potential application, providing sufficient space for the operator.

3 Scheme Proposal and Optimization Criterion

A neural-dynamics-based PCVCC scheme for redundancy resolution of MWDRA is arranged in the form of quadratic programming[41] as follows:

$$\begin{aligned}\text{minimize}\quad&\frac{1}{2}\omega^\mathrm{T}\mathcal{H}\omega+\kappa^\mathrm{T}\omega,\\ \text{s.t.,}\quad&\mathcal{K}\omega=\dot{\mathrm{r}}_\mathrm{p},\\ &\underline{\omega}\leqslant\omega\leqslant\overline{\omega}\end{aligned} \tag{12}$$

where $\mathcal{H}\in\mathbf{R}^{(2m+2)\times(2m+2)}$ denotes a positive definite symmetric matrix; $\kappa\in\mathbf{R}^{2m+2}$ stands for an optimization criterion; $\underline{\omega}$ and $\overline{\omega}$ represent the lower bound and upper bound of ω, respectively; $\mathrm{r}_\mathrm{p}\in\mathbf{R}^{2n}$ and $\dot{\mathrm{r}}_\mathrm{p}\in\mathbf{R}^{2n}$ signify the pre-designed end-effector position and velocity, respectively. Notably, the advantage of double arms is their capability to perform cooperative actions reflected in the designed task in this scheme. For example, double arms can simultaneously move two workpieces to specific positions for combination operations or cooperatively carry an object, which cannot be achieved by a single arm. Therefore, double arms not only enhance task efficiency by performing the task synchronously but also compensate for the limitations of a single robot arm, thereby improving overall performance.

Enlightened by the previous derivations, involving the connection points s_l and s_r, the velocities of left and right drive wheels, and the joint velocities of each robotic arm, it is crucial to design the optimization criterion κ based on the neural dynamics method that satisfies the requirements of realistic scenarios.

Primarily, the mobile platform should possess the capability to reach a specific position with precision to ensure subsequent movements. To this aim, an error function is formulated in the following manner:

$$\mathcal{E}_1=\begin{bmatrix}x^\mathrm{o}_{s_\mathrm{l}}-x_\mathrm{p}\\ y^\mathrm{o}_{s_\mathrm{l}}-y_\mathrm{p}\end{bmatrix} \tag{13}$$

where $(x_\mathrm{p},\ y_\mathrm{p})$ stands for the pre-designed desired position of connection point $x^\mathrm{o}_{s_\mathrm{l}}$. To drive the error $\mathcal{E}_1$ to converge towards zero, the neural dynamics method[4, 9] is employed with the expression $\dot{\mathcal{E}}=-\lambda\mathcal{E}$, where $\lambda>0$ determines the convergence rate of the error. Furthermore, inspired by, we design an exponential function and derive

$$\begin{bmatrix}\dot{x}^\mathrm{o}_{s_\mathrm{l}}\\ \dot{y}^\mathrm{o}_{s_\mathrm{l}}\end{bmatrix}=-\lambda_1\exp(t-\varrho)\begin{bmatrix}x^\mathrm{o}_{s_\mathrm{l}}-x_\mathrm{p}\\ y^\mathrm{o}_{s_\mathrm{l}}-y_\mathrm{p}\end{bmatrix} \tag{14}$$

where $\lambda_1>0$ stands for a design parameter, t denotes the time, and ϱ is a design parameter to guarantee increased convergence and the mobility of the mobile platform. Secondly, in various scenarios, it is essential to adapt the rotation mode of the drive wheels accordingly. For instance, in narrow corridors, the mobile platform is expected to move in a straight line, i.e., the heading angle μ remains zero on the whole. Similarly, prolonged movement in tight spaces (such as an office desk) necessitates limited spatial displacement and in-situ rotation of the mobile platform. Aiming at the above two conditions, the subsequent design process is provided. Specifically, two error functions $\mathcal{E}_2$ and $\mathcal{E}_3$ are constructed as

$$\begin{aligned}\mathcal{E}_2&=\mu=-\frac{r_\mathrm{w}}{2a}\dot{\phi}_\mathrm{l}+\frac{r_\mathrm{w}}{2a}\dot{\phi}_\mathrm{r}=0,\\ \mathcal{E}_3&=x^\mathrm{w}_{s_\mathrm{p}}=\frac{r_\mathrm{w}}{2a}\dot{\phi}_\mathrm{l}+\frac{r_\mathrm{w}}{2a}\dot{\phi}_\mathrm{r}=0.\end{aligned}$$

To achieve exponentially convergent systems, we have

$$\begin{aligned}-\dot{\phi}_\mathrm{l}+\dot{\phi}_\mathrm{r}&=-\lambda_2(-\phi_\mathrm{l}+\phi_\mathrm{r}),\\ \dot{\phi}_\mathrm{l}+\dot{\phi}_\mathrm{r}&=-\lambda_2(\phi_\mathrm{l}+\phi_\mathrm{r}),\end{aligned}$$

where $-\lambda_2>0$ is a convergence parameter. The above equations can be integrated into

$$\mathrm{sign}(b)\dot{\phi}_{\mathrm{l}}+\dot{\phi}_{\mathrm{r}}=-\lambda_2(\mathrm{sign}(b)\phi_{\mathrm{l}}+\phi_{\mathrm{r}}) \tag{15}$$

where b denotes a design parameter; $\mathrm{sign}(b)=1$ for $b>0$ and $\mathrm{sign}(b)=-1$ for $b<0$. Thirdly, when considering the manipulation of double robotic arms, a crucial function of these arms is to adjust their status, including manipulability, joint angles, and end-effector positions. Similar to Eq. (14), an equality constraint for the joint angle adjustment of robotic arms is designed as

$$\dot{\theta}_{\mathrm{l}}=-\lambda_3\exp(t-\varrho)(\theta_{\mathrm{l}}-\theta_{\mathrm{lp}}) \tag{16}$$

$$\dot{\theta}_{\mathrm{r}}=-\lambda_3\exp(t-\varrho)(\theta_{\mathrm{r}}-\theta_{\mathrm{rp}}) \tag{17}$$

where $\lambda_3>0$ denotes a design parameter; θ_{lp} and θ_{rp} correspond to the pre-designed joint angles of left and right robotic arms. Combining the results obtained from Eqs. (14)–(17) yields

$$\begin{bmatrix} \dot{x}^{\mathrm{o}}_{s_1} \\ \dot{y}^{\mathrm{o}}_{s_1} \\ \mathrm{sign}(b)\dot{\phi}_{\mathrm{l}}+\dot{\phi}_{\mathrm{r}} \\ \dot{\theta}_{\mathrm{l}} \\ \dot{\theta}_{\mathrm{r}} \end{bmatrix} = -\underbrace{\begin{bmatrix} \lambda_1\exp(t-\varrho)(x^{\mathrm{o}}_{s_1}-x_{\mathrm{p}}) \\ \lambda_1\exp(t-\varrho)(y^{\mathrm{o}}_{s_1}-y_{\mathrm{p}}) \\ \lambda_2(\mathrm{sign}(b)\phi_{\mathrm{l}}+\phi_{\mathrm{r}}) \\ \lambda_3\exp(t-\varrho)(\theta_{\mathrm{l}}-\theta_{\mathrm{lp}}) \\ \lambda_3\exp(t-\varrho)(\theta_{\mathrm{r}}-\theta_{\mathrm{rp}}) \end{bmatrix}}_{\mathcal{I}\in\mathbf{R}^{2m+3}} \tag{18}$$

which can be further overwritten as

$$\underbrace{\begin{bmatrix} \mathcal{C}_{11} & \mathcal{C}_{12} & 0 \\ \mathcal{C}_{21} & \mathcal{C}_{22} & 0 \\ \mathrm{sign}(b) & 1 & 0 \\ 0 & 0 & I_{2m} \end{bmatrix}}_{\mathcal{W}\in\mathbf{R}^{(2m+3)\times(2m+2)}}\omega=\mathcal{I} \tag{19}$$

where $\mathcal{C}_{11}$, $\mathcal{C}_{12}$, $\mathcal{C}_{21}$, and $\mathcal{C}_{22}$ denote the corresponding elements in matrix $\mathcal{C}$ in Eq. (8). Therefore, utilizing the pseudoinverse method, the optimization objective for the state adjustment is obtained as

$$\mathcal{L}=\mathcal{W}^{\dagger}\mathcal{I}\in\mathbf{R}^{2m+2} \tag{20}$$

where superscript "†" is the pseudoinverse operation of a matrix with $\mathcal{W}^{\dagger}=(\mathcal{W}^{\mathrm{T}}\mathcal{W})^{-1}\mathcal{W}^{\mathrm{T}}$. Further, the optimization criterion κ can be obtained,

$$\kappa=\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L} \tag{21}$$

where $\alpha>0$ is a parameter to adjust the convergence rate; $I-\mathcal{K}^{\dagger}\mathcal{K}$ is the orthogonal projection matrix as reported in Ref. [38]. This orthogonal projection matrix can project the optimization objective $\mathcal{L}$ into the null space of the Jacobian matrix $\mathcal{K}$. Consequently, it can be deduced that $\alpha\mathcal{K}(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}=0$. By incorporating this orthogonal projection matrix, the optimization objective $\mathcal{L}$ is decoupled from the tracking task at the algorithmic level. As a result, both the optimization objective and tracking task can be guaranteed without mutual interference[38].

The novelty of the PCVCC scheme of Eq. (12) lies in optimization criterion κ derived above, which can coordinately adjust the position of the mobile platform, the rotation mode of drive wheels, and joints of double robotic arms. In this way, one can simultaneously achieve precise collaborative control of the mobile platform and robotic arms. In contrast, most existing methods[3–6, 15, 22] on the kinematic control of mobile robots mainly focus on the task execution of the end-effector and ignore the precise control of the mobile platform. This leads to the defect that the movement of the mobile platform is unpredictable and cannot be precisely controlled during the task execution. For an MWDRA system, these methods may compromise the operator's safety and increase the risk of accidents.

4 DNN Solver and Stability Analysis

This section proposes a DNN to solve the PCVCC scheme of Eq. (12). We begin by formulating the position error function $\varepsilon=\mathfrak{r}-\mathfrak{r}_{\mathrm{p}}$ and constructing a two-norm form $\|\varepsilon\|^2/2$, where $\|\cdot\|$ denotes the two-norm of a vector. To minimize the error, the gradient descent method[42] is employed to determine the evolution direction, which yields

$$\omega=-\zeta\frac{\partial\|\varepsilon\|^2/2}{\partial q}=\zeta\mathcal{K}^{\mathrm{T}}(\mathfrak{r}_{\mathrm{p}}-\mathfrak{r}) \tag{22}$$

where $\zeta>0$ is a design parameter; the vector $q=[\phi_{\mathrm{l}};\ \phi_{\mathrm{r}};\ \theta_{\mathrm{l}};\ \theta_{\mathrm{r}}]$. Nevertheless, Eq. (22) can be regarded as an error feedback with lagging errors and cannot fully reflect the PCVCC scheme of Eq. (12). To address this limitation, a velocity compensation item $\varpi\in\mathbf{R}^{2m+2}$ is considered as follows:

$$\omega=\zeta\mathcal{K}^{\mathrm{T}}(\mathfrak{r}_{\mathrm{p}}-\mathfrak{r})+\varpi \tag{23}$$

Multiplying both sides of Eq. (23) by $\mathcal{K}$ matrix yields the following expression:

$$\mathcal{K}\omega=\zeta\mathcal{K}\mathcal{K}^{\mathrm{T}}(\mathfrak{r}_{\mathrm{p}}-\mathfrak{r})+\mathcal{K}\varpi \tag{24}$$

Considering the desired steady state $\mathfrak{r}_{\mathrm{p}}-\mathfrak{r}=0$ of the dynamic system of Eq. (24), we can derive $\dot{\mathfrak{r}}_{\mathrm{p}}=\mathcal{K}\omega=\mathcal{K}\varpi$. The solution of ϖ to this affine system is deduced as follows:

$$\varpi = \mathcal{K}^{\dagger}\dot{\mathfrak{r}}_{\rm p} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L} \tag{25}$$

where $\mathcal{K}^{\dagger}\dot{\mathfrak{r}}_{\rm p}$ represents the general solution with $\mathcal{K}^{\dagger} = \mathcal{K}^{\rm T}(\mathcal{K}\mathcal{K}^{\rm T})^{-1}$, and $\alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$ represents the particular solution obtained based on the optimization criterion. Subsequently, an auxiliary variable $\boldsymbol{s} \in \mathbf{R}^{2n}$ is introduced to avoid the inversion operation. Specifically, this variable is developed to estimate $(\mathcal{K}\mathcal{K}^{\rm T})^{-1}\dot{\mathfrak{r}}_{\rm p}$ with storage and delivery capacity. In this sense, the steady state of auxiliary variable $\boldsymbol{s}$ should satisfy $\mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s} - \dot{\mathfrak{r}}_{\rm p} = 0$. Given the positive definite property of $\mathcal{K}\mathcal{K}^{\rm T}$ in a nonsingular state, a dynamic system is designed as $\dot{\boldsymbol{s}} = \sigma(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s})$, where $\dot{\boldsymbol{s}}$ is the time derivative of $\boldsymbol{s}$, and $\sigma > 0$ denotes a convergence coefficient. On this basis, the following dynamic system is built to process dynamic information recursively:

$$\begin{aligned} \varpi &= \mathcal{K}^{\rm T}\boldsymbol{s} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}, \\ \dot{\boldsymbol{s}} &= \sigma(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s}) \end{aligned} \tag{26}$$

By combining Eqs. (26) and (23), a DNN is established as follows:

$$\begin{aligned} \omega &= \mathcal{P}_{\underline{\varpi}}(\zeta\mathcal{K}^{\rm T}(\mathfrak{r}_{\rm p} - \mathfrak{r}) + \mathcal{K}^{\rm T}\boldsymbol{s} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}), \\ \dot{\boldsymbol{s}} &= \sigma(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s}) \end{aligned} \tag{27}$$

where projection function $\mathcal{P}_{\underline{\varpi}}(\boldsymbol{x}) = \mathrm{argmin}_{\boldsymbol{y}\in\underline{\varpi}}\|\boldsymbol{x} - \boldsymbol{y}\|$ with $\underline{\varpi} = \{\boldsymbol{y} \in \mathbf{R}^{2m+2}, \underline{\omega} \leqslant \boldsymbol{y} \leqslant \overline{\varpi}\}$ is used to ensure the inequality constraint in Eq. (12). Notably, DNN of Eq. (27) belongs to a vital component of robot development and cognitive systems. Specifically, it enhances system performance within the computational intelligence domain by incorporating the velocity compensation and orthogonal projection matrix[43]. Besides, a control block diagram of DNN of Eq. (27) is presented in Fig. 3. The inputs of DNN of Eq. (27) consist of the desired states of the DMR. Simultaneously, DNN of Eq. (27) receives the current states of the DMR. Utilizing these state data, it outputs the control signal ω in real time, including the robotic arms' joint velocities and the driving wheels' rotation velocities. This control signal regulates the DMR, enabling the current state to satisfy the desired task. To further analyze DNN of Eq. (27), the following theorem and remarks are provided.

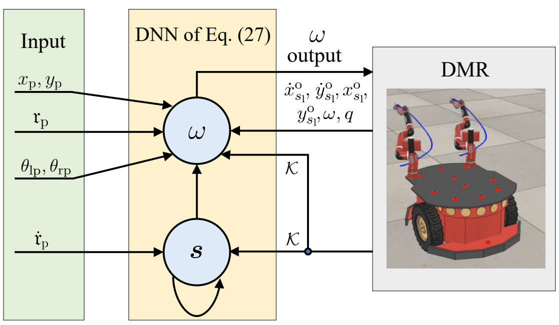


Fig. 3 Control block diagram for DNN of Eq. (27).

Theorem 1 The position error $\varepsilon = \mathfrak{r} - \mathfrak{r}_{\rm p}$ generated by the PCVCC scheme Eq. (12) aided with DNN of Eq. (27) globally converges to zero, assuming that $\mathcal{K}^{\dagger}\dot{\mathfrak{r}}_{\rm p} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L} \in \underline{\varpi}$.

Proof Since DNN of Eq. (27) is divided into two subsystems, the corresponding proof process is also segmented into two steps: the proof of convergence of $\boldsymbol{s}$ and the proof of convergence of ε.

Step 1: the proof of convergence of $\boldsymbol{s}$. Design a Lyapunov candidate,

$$\mathcal{V}_1 = \frac{(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s})^{\rm T}(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s})}{2} \geqslant 0 \tag{28}$$

and calculate its time derivative as

$$\begin{aligned} \dot{\mathcal{V}}_1 = {} & (\mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s} - \dot{\mathfrak{r}}_{\rm p})^{\rm T}\mathcal{K}\mathcal{K}^{\rm T}\dot{\boldsymbol{s}} = \\ & -(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s})^{\rm T}\mathcal{K}\mathcal{K}^{\rm T}\sigma(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s}) \leqslant \\ & -\sigma\Lambda(\mathcal{K}\mathcal{K}^{\rm T})(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s})^{\rm T}(\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s}) = \\ & -2\sigma\Lambda(\mathcal{K}\mathcal{K}^{\rm T})\mathcal{V}_1 \leqslant 0 \end{aligned} \tag{29}$$

where $\Lambda(\mathcal{K}\mathcal{K}^{\rm T}) > 0$ is the minimum eigenvalue of $\mathcal{K}\mathcal{K}^{\rm T}$. Owing to $\mathcal{V}_1 \geqslant 0$ and $\dot{\mathcal{V}}_1 \leqslant 0$, the convergence of $\dot{\mathcal{V}}_1$ is determinated. Additionally, it can be summarized from Eq. (29) that

$$0 \leqslant \lim_{t\to+\infty}\mathcal{V}_1 \leqslant \lim_{t\to+\infty}\delta\cdot\exp(-2\sigma\Lambda(\mathcal{K}\mathcal{K}^{\rm T})) = 0,$$

with $\delta > 0$ denoting a constant. Consequently, $\dot{\mathfrak{r}}_{\rm p} - \mathcal{K}\mathcal{K}^{\rm T}\boldsymbol{s}$ is able to converge exponentially to zero. Put another way, the steady state of $\boldsymbol{s}$ is $(\mathcal{K}\mathcal{K}^{\rm T})^{-1}\dot{\mathfrak{r}}_{\rm p}$.

Step 2: the proof of convergence of ε. Based on the derivations of Step 1 and Lasalle's invariant set principle[44], the steady state of $\boldsymbol{s}$ is substituted into dynamic system of Eq. (27), which results in

$$\omega = \mathcal{P}_{\underline{\varpi}}(\zeta\mathcal{K}^{\rm T}(\mathfrak{r}_{\rm p} - \mathfrak{r}) + \mathcal{K}^{\rm T}(\mathcal{K}\mathcal{K}^{\rm T})^{-1}\dot{\mathfrak{r}}_{\rm p} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}) \tag{30}$$

By following the equations $\mathcal{K}\omega = \dot{\mathfrak{r}}$ and $\mathcal{K}(I - \mathcal{K}^{\dagger}\mathcal{K}) = 0$, we extrapolate

$$\begin{aligned} \dot{\mathfrak{r}} = \mathcal{K}\omega = {} & \mathcal{K}(\mathcal{P}_{\underline{\varpi}}(\zeta\mathcal{K}^{\rm T}(\mathfrak{r}_{\rm p} - \mathfrak{r}) + \mathcal{K}^{\rm T}(\mathcal{K}\mathcal{K}^{\rm T})^{-1}\dot{\mathfrak{r}}_{\rm p} + \\ & \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}) - \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}) \end{aligned} \tag{31}$$

Then, considering the relationship $\mathcal{K}\mathcal{K}^{\dagger} = I$, the velocity error $\dot{\varepsilon}$ can be expressed as

$$\dot{\varepsilon}=\mathcal{K}(\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}).$$

Subsequently, define a Lyapunov candidate $\mathcal{V}_2=\varepsilon^{\mathrm{T}}\varepsilon/2$, and its time derivative can be derived as

$$\dot{\mathcal{V}}_2=\varepsilon^{\mathrm{T}}\dot{\varepsilon}=\varepsilon^{\mathrm{T}}\mathcal{K}(\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}})=-\frac{1}{\zeta}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon)^{\mathrm{T}}(\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}) \tag{32}$$

Recall projection function $\mathcal{P}_{\overline{\varpi}}(\boldsymbol{x})=\mathrm{argmin}_{y\in\overline{\varpi}}\|\boldsymbol{x}-\boldsymbol{y}\|$, and the inequality $\|\boldsymbol{x}-\mathcal{P}_{\overline{\varpi}}(\boldsymbol{x})\|\leqslant\|\boldsymbol{x}-\boldsymbol{y}\|$, $\forall\boldsymbol{y}\in\overline{\varpi}$ always holds. Given $\boldsymbol{x}=-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$ and $\boldsymbol{y}=\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$, it follows that

$$\|-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}-\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})\|^2\leqslant\|-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon\|^2 \tag{33}$$

which can be reformulated as

$$\|\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}\|^2\leqslant 2(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon)^{\mathrm{T}}(\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}) \tag{34}$$

Integrating Eqs. (34) and (32) results in

$$\dot{\mathcal{V}}_2\leqslant-\frac{1}{2\zeta}\|\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}\|^2\leqslant 0 \tag{35}$$

The conclusion that $\dot{\mathcal{V}}_2\leqslant 0$ and $\mathcal{V}_2\geqslant 0$ suggests the global convergence of ε. By applying LaSalle's invariance set principle[44], setting $\dot{\mathcal{V}}_2=0$ to investigate the steady state of ε produces the two situations: (1) $-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon=0$ and (2) $\mathcal{P}_{\overline{\varpi}}(-\zeta\mathcal{K}^{\mathrm{T}}\varepsilon+\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})-\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}-\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}=0$. For Case (1), it can be directly acquired that $\varepsilon=0$. In Case (2), the precondition $\mathcal{K}^{\dagger}\dot{\mathrm{r}}_{\mathrm{p}}+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}\in\overline{\varpi}$ implies a stable state retaining within the scope of the inequality constraint, leading to the same result $\varepsilon=0$. In summary, the position error ε synthesized by PCVCC scheme of Eq. (12) aided with the DNN of Eq. (27) for the MWDRA globally converges to zero, which completes the proof. ■

Remark 1 DNN of Eq. (27) belongs to a special kind of RNN with known physical information as prior knowledge. Specifically, a continuous-time RNN is presented as

$$\begin{aligned}\dot{z}(t)&=g_1(W_1z(t)+W_2x(t)),\\ y(t)&=g_2(W_3z(t)+W_4x(t))\end{aligned} \tag{36}$$

where $z(t)$ is the state of the hidden layer with its time derivative $\dot{z}(t)$; $x(t)$ is the input; W_1, W_2, W_3, and W_4 denote the weight matrices; $g_1(\cdot)$ and $g_2(\cdot)$ are activation functions; $y(t)$ denotes the output. Besides, a schematic of RNNs is depicted in Fig. 4. The corresponding relationship between Eqs. (27) and (36) is arranged as $y(t)=\omega$, $z(t)=s$, $g_1(\cdot)=\sigma(\cdot)$, $g_2(\cdot)=\mathcal{P}_{\overline{\varpi}}(\cdot)$, $W_1=-\mathcal{K}\mathcal{K}^{\mathrm{T}}$, $W_2=[I\ \ 0]$, $W_3=\mathcal{K}^{\mathrm{T}}$, $x(t)=[\dot{\mathrm{r}}_{\mathrm{p}};\zeta\mathcal{K}^{\mathrm{T}}(\mathrm{r}_{\mathrm{p}}-\mathrm{r})+\alpha(I-\mathcal{K}^{\dagger}\mathcal{K})\mathcal{L})]$, and $W_4=[0\ \ I]$. As shown, the recurrent structure of DNN corresponds to the auxiliary variable $\boldsymbol{s}$, while the output is ω. Besides, the projection function $\mathcal{P}_{\overline{\varpi}}(\cdot)$ is the activation function utilized in DNN of Eq. (27). The weight matrices W_1, W_2, W_3, and W_4 incorporate the physical information of the PCVCC scheme of Eq. (12) as prior knowledge without training.

Remark 2 Given a task for the end-effector, the PCVCC scheme of Eq. (12) for redundancy resolution of the DMR that needs to be solved is essentially an inverse kinematic problem. However, state adjustment is also critical during the daily operation of the DMR, which may not involve end-effector tasks and can be treated as a forward kinematics problem. Therefore, in the context of state adjustment, DNN of Eq. (27) could be simplified into a new form by setting $\dot{\mathrm{r}}_{\mathrm{p}}=\zeta=0$, resulting in

$$\begin{aligned}\omega&=\mathcal{P}_{\overline{\varpi}}(\mathcal{K}^{\mathrm{T}}\boldsymbol{s}+\alpha\mathcal{L}),\\ \dot{\boldsymbol{s}}&=\sigma(\mathcal{K}\mathcal{K}^{\mathrm{T}}\boldsymbol{s})\end{aligned} \tag{37}$$

The stability of DNN of Eq. (37) can be inferred similarly based on the theorem, and thus the proof is

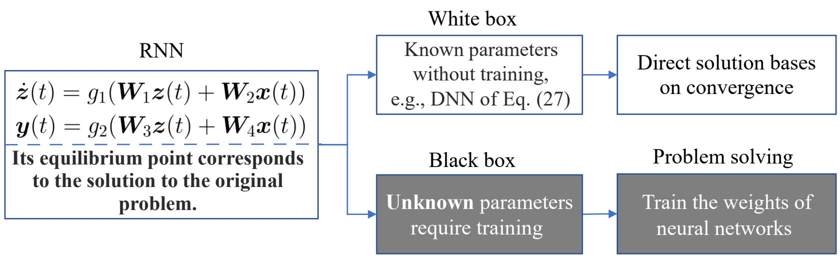


Fig. 4 Schematic of RNNs.

omitted.

Remark 3 We design DNN of Eq. (27) as a special case of RNN, which incorporates the physical information of the PCVCC scheme of Eq. (12). Therefore, the equilibrium point of DNN of Eq. (27) corresponds to the optimal solution of the PCVCC scheme of Eq. (12). As the theorem proves, DNN of Eq. (27) converges from an initial state to the equilibrium state with $\varepsilon = 0$, which is equivalent to solving the PCVCC scheme of Eq. (12). Besides, the corresponding relationship between DNN of Eq. (27) and PCVCC scheme of Eq. (12) is described as follows. As shown in Fig. 5, we can divide DNN of Eq. (27) into three parts. The projection equation $\mathcal{P}_{\overline{\underline{\omega}}}(\cdot)$ can restrict the joint velocity within limits, satisfying inequality constraint $\underline{\omega} \leqslant \omega \leqslant \overline{\omega}$. Besides, the feedback term of the position error $\zeta \mathcal{K}^{\mathrm{T}}(\mathfrak{r}_{\mathrm{p}} - \mathfrak{r})$ guarantees the trajectory task, which conforms to equality constraint $\mathcal{K}\omega = \dot{\mathfrak{r}}_{\mathrm{p}}$. According to the theorem, velocity compensation term $\mathcal{K}^{\mathrm{T}}\boldsymbol{s} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$ can converge to $\mathcal{K}^{\dagger}\dot{\mathfrak{r}}_{\mathrm{p}} + \alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$, which includes the general solution $\mathcal{K}^{\dagger}\dot{r}_{\mathrm{p}}$ and particular solution $\alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$. The general solution $\mathcal{K}^{\dagger}\dot{\mathfrak{r}}_{\mathrm{p}}$ is designed to minimize $\omega^{\mathrm{T}}\mathcal{H}\omega/2$, which is the least-squares solution to equality constraint $\mathcal{K}\omega = \dot{\mathfrak{r}}_{\mathrm{p}}$. Besides, particular solution $\alpha(I - \mathcal{K}^{\dagger}\mathcal{K})\mathcal{L}$ corresponds to optimization criterion κ.

Remark 4 In real-world scenarios, the real-time performance of the system is influenced by positive convergence parameters in DNN of Eq. (27), i.e., ζ, λ_1, λ_2, and λ_3. When these convergence parameters are set to excessive values, it can lead to an overshooting phenomenon. In this case, the robot system may exhibit shaking or oscillating behavior, which is not conducive to precise control of the robot. If the convergence parameters are set too small, the convergence rate of DNN of Eq. (27) will decrease, making it difficult for the robot to accurately complete the assigned task. To mitigate this issue, it is important to set convergence parameters to appropriate values, avoiding being too large or too small. The specific range for parameter settings often requires further determination through a trial and error method in real-world scenarios.

Remark 5 There exists abundant literature using neural networks as quadratic programming solvers in robot control, such as ZNNs[31], GNNs[34], and RNNs[37]. Compared with these neural networks, the advantages of the proposed DNN of Eq. (27) are explained as follows. ZNNs necessitate the computation of the pseudoinverse operation and time derivative of the Jacobian matrix[31]. The pseudoinverse operation may encounter singularity issues, resulting in a loss of control for the robot. In contrast, DNN of Eq. (27) avoids these computational operations and is relatively concise and stable. Furthermore, gradient neural networks employ the gradient descent method to converge errors, which inevitably has a lagging error[34]. Besides, RNNs are proved to have an error accumulation problem in Ref. [37]. Compared with recurrent neural networks and gradient neural networks, DNN of Eq. (27) designs a velocity compensation term and orthogonal projection matrix to eliminate the lagging error, and it has been theoretically proved that the position error is reduced to zero.

5 Simulation, Experiment, and Comparison

This section presents computer simulations, simulative experiments, and performance comparisons on the MWDRA to verify the feasibility of the PCVCC scheme of Eq. (12) and evaluate the efficiency of DNN

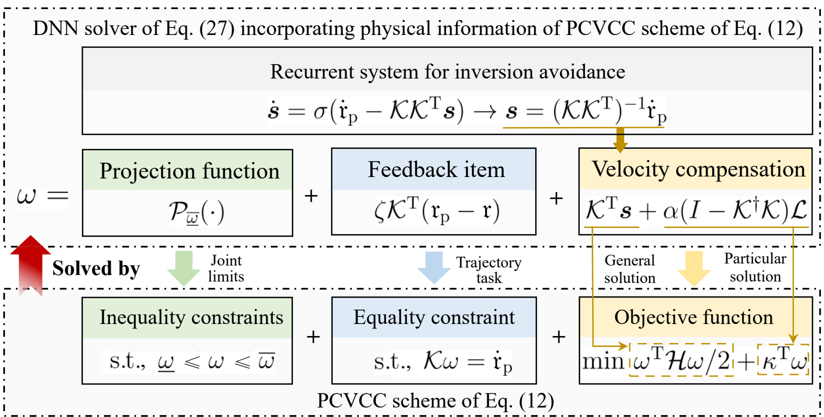


Fig. 5 Corresponding relationship between DNN of Eq. (27) and PCVCC scheme of Eq. (12).

of Eq. (27). It is worth noteworthy that in computer simulations, the structural parameters of the mobile wheelchair are designed as $a = x_l = x_r = y_l = y_r = 0.5$ m and $r_w = 0.25$ m, and the robotic arm used is 6-DOF Sawyer robotic arm[45]. In simulative experiments, the MWDRA consists of a pioneer 3-DX differential-drive mobile platform[46] and two 6-DOF Sawyer robotic arms[45]. Note that the mobile platform is scaled up to mount two robotic arms with structural parameters as $a = x_l = x_r = y_l = y_r = 0.53$ m and $r_w = 0.33$ m. Additionally, the other fixed parameters are devised as follows: $\varrho = 8$, $\sigma = 1000$, $\overline{\omega} = -\underline{\omega} = [1.5]_{14\times 1}$ rad/s, $\alpha = 10$, $\theta_l(0) = \theta_r(0) = [0, -0.67, 0, 1.3, 0, 0.5]^{\mathrm{T}}$ rad, $\mu(0) = 0$, and $x^{\mathrm{o}}_{s_1}(0) = y^{\mathrm{o}}_{s_1}(0) = 0$. Moreover, the simulation parameters include the convergence parameters of DNN of Eq. (27) and structural parameters. The convergence parameters are determined by trial and error, while structural parameters are set according to the operational conditions of the MWDRA.

5.1 Trajectory tracking task

The simulation results of trajectory tracking on the MWDRA driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (27) are presented in Fig. 6. The pre-designed trajectory is a closed “OK” curve and the design parameters are $\zeta = 10^5$, $\lambda_1 = \lambda_3 = 2$, and $\lambda_2 = 100$, $b = -1$, $x_p = y_p = 0$, and $\theta_{lp} = \theta_{rp} = \theta_l(0)$. The mobile wheelchair moves in a straight line along with the robotic arm to perform the given task, as depicted in Figs. 6a and 6b, which demonstrates the successful completion of the task. Specifically, Figs. 6c and 6d embody that the actual trajectory is consistent with the pre-designed path with the position error of the order 10^{-5} m, where ε_{lX}, ε_{lY}, ε_{lZ} and ε_{rX}, ε_{rY}, ε_{rZ} represent the errors of the left and rights robotic arm end affectors in the X, Y, and Z axes, respectively. Moreover, the motion variation properties of the mobile wheelchair are plotted in Figs. 6e and 6f, which indicate the straight movement of the mobile

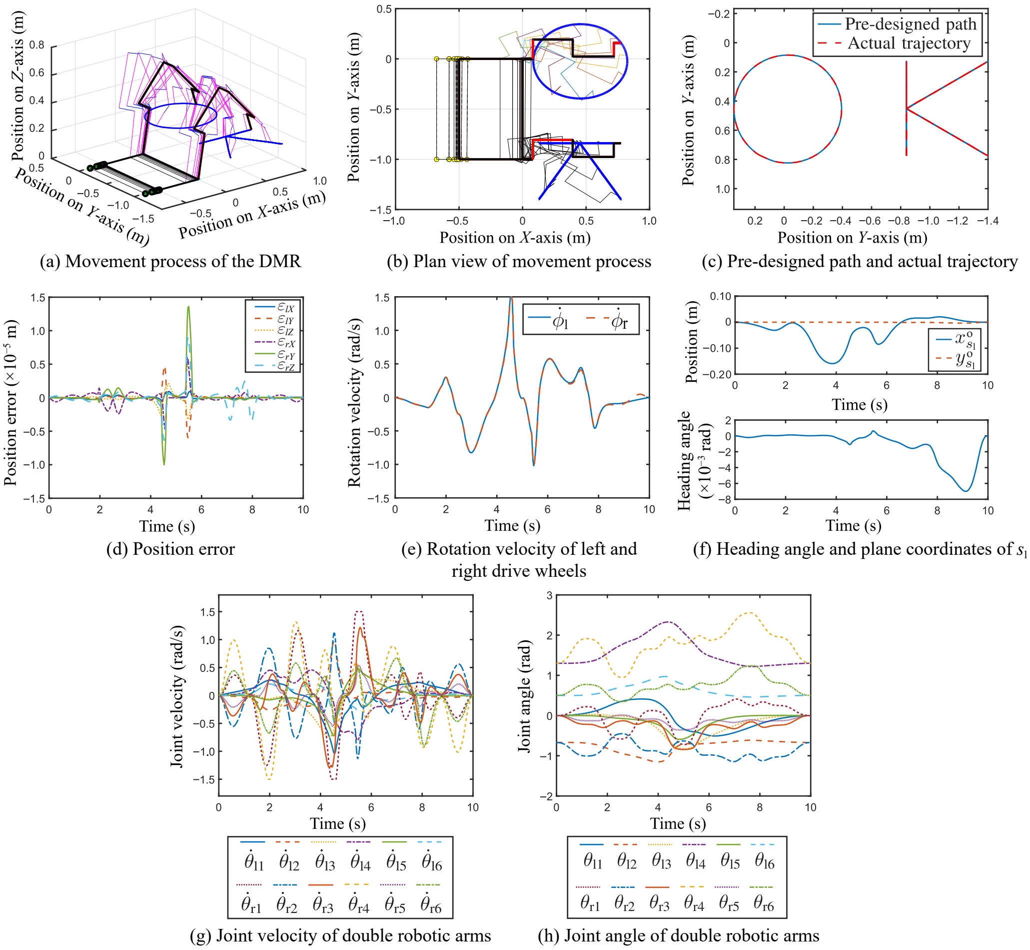


Fig. 6 Simulation results on the DMR tracking “OK” trajectory driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (27) and $b = -1$.

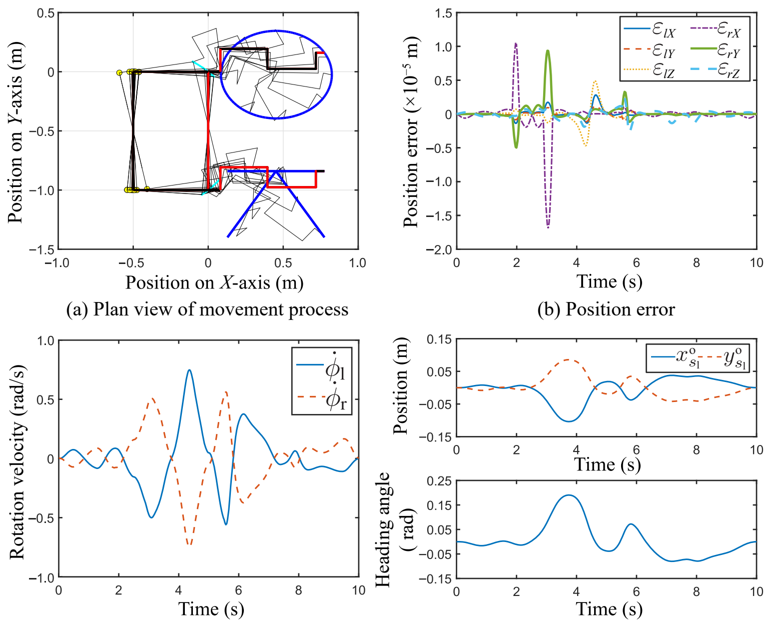
wheelchair with the left and right drive wheels retaining nearly the same velocity and almost zero heading angle. Additionally, plane coordinates of s_1 return the pre-designed values. In Fig. 6g, the joint velocity maintains within joint limits, while in Fig. 6h, the joint angle returns the pre-designed state, i.e., $\theta_1(0)$. These results illustrate the feasibility and efficiency of the proposed PCVCC scheme of Eq. (12) and DNN of Eq. (27).

Excessive displacement is often unsatisfactory when the MWDRA works in narrow areas. To address this issue, simulations are conducted with $b = 1$ and other parameters remaining unchanged. The results are illustrated in Fig. 7. It can be inferred from Figs. 7a and 7b that the pre-designed task is implemented accurately with the position errors being the order of 10^{-5} m. Furthermore, in Figs. 7c and 7d, the opposite rotation speeds of the left and right drive wheels effectively limit the mobile wheelchair's displacement, and the plane coordinates of s_1 converge to the pre-designed positions. Based on these task and simulation settings, the presented problem can be deemed as a repetitive motion scheme[4]. Figures 6 and 7 indicate that the end states of the robotic system are the same as the initial ones.

Moreover, simulative experiments are performed in CoppeliaSim software to visualize the movement of the MWDRA corresponding to Fig. 6. Except for the mobile platform, the other parameters are the same as those in the simulations. Driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (27), the MWDRA is programmed to perform the trajectory tracking task, and snapshots are captured in Fig. 8. Evidently, the "OK" trajectory task is executed accurately, with the final states identical to the initial ones. These results verify the reliability and efficiency of the PCVCC scheme of Eq. (12) and DNN of Eq. (27). Furthermore, related comparative experiments are presented in Section 4.3.

5.2 State adjustment task

To realize the state adjustment of the MWDRA, we utilize the PCVCC scheme of Eq. (12) and DNN of Eq. (37) to manipulate the MWDRA for adjusting to a required state. Note that we choose $x_p = 1$ m, $y_p = 0.5$ m, $b = -1$, $\theta_{lp} = \theta_{rp} = [-0.5]_{6\times1}$ rad, $\lambda_1 = 200$, $\lambda_2 = 0.01$, and $\lambda_3 = 1$. Related simulation results are illustrated in Figs. 9 and 10. Specifically, the joint velocity and drive wheel velocity are momentarily constrained at the joint limits and eventually reduced to zero in Fig. 9a. In Fig. 9b, joint angles of double robotic arms converge to -0.5 rad in 2 s. Similarly, Fig. 9c shows the convergence of plane coordinates of s_1 to the desired position. Covering the above characteristics, Fig. 9b



(a) Plan view of movement process
(b) Position error
(c) Rotation velocity of left and right drive wheels
(d) Heading angle and plane coordinates of s_1

Fig. 7 Simulation results on the DMR tracking "OK" trajectory driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (27) and $b = 1$.

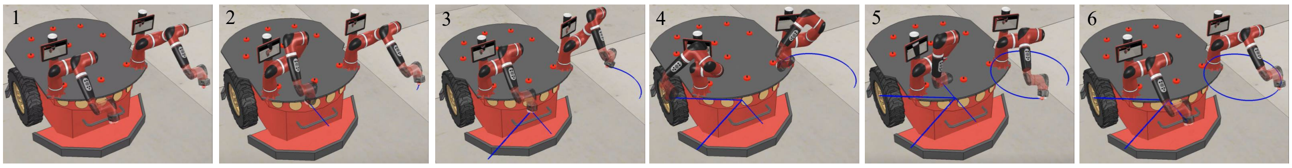
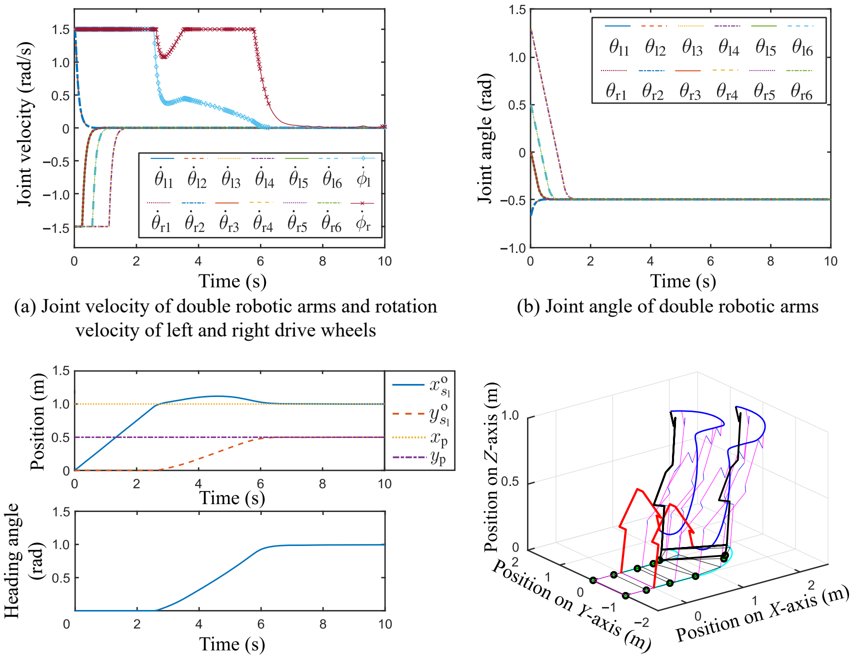


Fig. 8 Snapshots of CoppeliaSim-based simulative experiments on the DMR for tracking “OK” trajectory driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (27). The experimental video is available at https://youtu.be/E2nn9UWHPG0.



(a) Joint velocity of double robotic arms and rotation velocity of left and right drive wheels

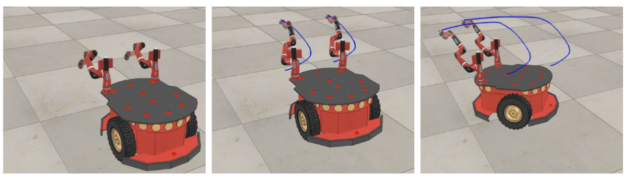
(b) Joint angle of double robotic arms

(c) Heading angle and plane coordinates of s_1

(d) Movement process of the DMR

Fig. 9 Simulation results of state adjustment on the DMR driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (37).

shows the whole process of the state adjustment of the MWDRA. Further, simulative experiments in Fig. 10 indicate the initial state, the in-motion state, and the final state, which reflects the reliable performance of the proposed method in state adjustment. Additionally, except for $a = x_\mathrm{l} = x_\mathrm{r} = y_\mathrm{l} = y_\mathrm{r} = 0.53$ m and $r_\mathrm{w} = 0.33$ m, the other parameters in the experiments are the same as those in the simulations.

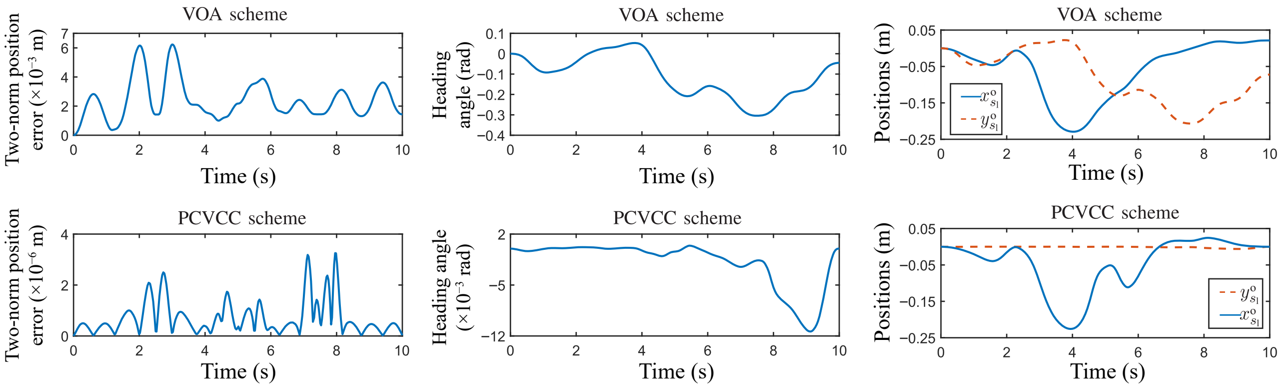


(a) (b) (c)

Fig. 10 Snapshots of CoppeliaSim-based simulative experiments on the DMR for state adjustment task driven by the PCVCC scheme of Eq. (12) aided with DNN of Eq. (37). The experimental video is available at https://youtu.be/W9Ze5VkhA8k.

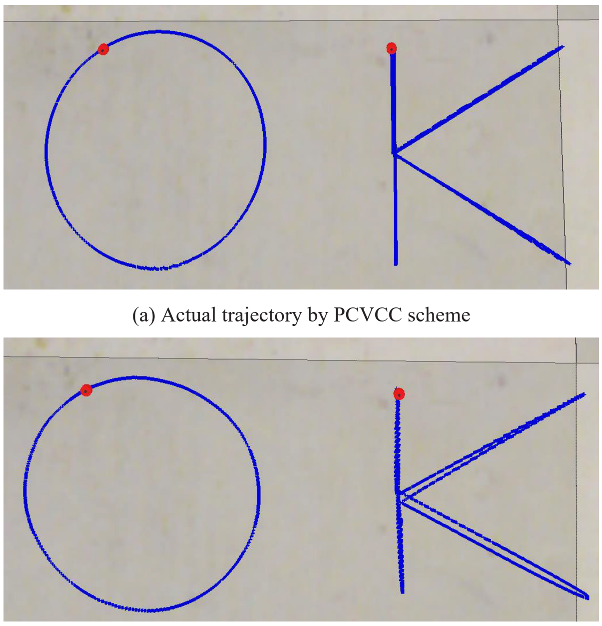
5.3 Comparative experiments

For further comparison, experiments synthesized by the PCVCC scheme of Eq. (12) and VOA scheme[5] are conducted on the MWDRA. Experimental results for tracking “OK” trajectory driven by the two schemes are shown in Figs. 11 and 12, and Table 2, and the experimental video is provided at https://youtu.be/E2nn9UWHPG0. Note that experiments synthesized by the PCVCC scheme of Eq. (12) are those in Section 4.1, and the solver for the VOA scheme is the linear variational inequality-based primal-dual neural network in Ref. [5]. From the video, one can discover that the PCVCC scheme of Eq. (12) and VOA scheme[5] perform trajectory tasks in different kinetic manners. However, the actual trajectory generated by



(a) Two-norm position error (b) Heading angle (c) Plane coordinates of s_1

Fig. 11 Simulative results on the DMR for tracking "OK" trajectory driven by different schemes.



(a) Actual trajectory by PCVCC scheme

(b) Actual trajectory by VOA scheme

Fig. 12 Simulative experiments on the MWDRA for tracking "OK" trajectory driven by different schemes. The related video is provided at https://youtu.be/E2nn9UWHPG0.

the PCVCC scheme of Eq. (12) is more precise than that generated by the VOA scheme[5], as Figs. 12a and 12b indicate. Accordingly, in Fig. 11a, the two-norm position error by PCVCC scheme of Eq. (12) maintains the order of 10^{-6} m, and that by VOA scheme[5] is relatively large. From Fig. 11b, it is clear that the heading angle by the PCVCC scheme of Eq. (12) is almost zero, embodying the straight-line motion of the mobile platform. In contrast, the heading angle by the VOA scheme[5] varies over time in an unpredictable state. As reflected in Fig. 11c, the PCVCC scheme of Eq. (12) can drive coordinates of s_1 to return to their initial states, but the VOA scheme[5] cannot achieve. Besides, detailed numerical results of comparative experiments are recorded in Table 2. It is noteworthy from Table 2 that the PCVCC scheme of Eq. (12) simultaneously manipulates the mobile platform and robotic arms to the pre-designed desired states, while the VOA scheme[5] does not. Overall, the advantage of the PCVCC scheme of Eq. (12) is that it can determine the position of the wheelchair, the rotation mode of drive wheels, and joints of robotic arms while completing the given task.

5.4 Cooperative task and comparisons

In this subsection, we assign the two robotic arms of the DMR to collaborate in completing a more meaningful trajectory tracking task. Specifically, we define a curved hexagonal trajectory as the preset trajectory. The left arm executes the left half of the trajectory from the bottom to the top, while the right arm executes the right half of the trajectory from the top to the bottom. The specific parameters are set as

Table 2 Performance comparison between PCVCC scheme of Eq. (12) and VOA scheme[5] for controlling MWDRA.

Scheme	$\lvert x^{\mathrm{o}}_{s_1}(10)-x_{\mathrm{p}}\rvert^{\diamond}$ (m)	$\lvert y^{\mathrm{o}}_{s_1}(10)-y_{\mathrm{p}}\rvert^{\diamond}$ (m)	Max$^{\star}\lvert\mu\rvert^{\diamond}$ (rad)	$\lVert\theta_{\mathrm{l}}(10)-\theta_{\mathrm{lp}}\rVert$ (rad)	$\lVert\theta_{\mathrm{r}}(10)-\theta_{\mathrm{rp}}\rVert$ (rad)	Max$^{\star}\lVert\mathrm{r}_{\mathrm{p}}-\mathrm{r}\rVert$ (m)
PCVCC scheme of Eq. (12)	6.08×10^{-6}	9.35×10^{-5}	1.14×10^{-2}	1.11×10^{-4}	1.00×10^{-4}	3.25×10^{-6}
VOA scheme[5]	2.14×10^{-2}	7.24×10^{-2}	3.04×10^{-1}	1.59×10^{-1}	1.63×10^{-1}	6.23×10^{-3}

* Max denotes finding the maximum value during the task execution.
$\lvert\cdot\rvert^{\diamond}$ represents the absolute value of operation.

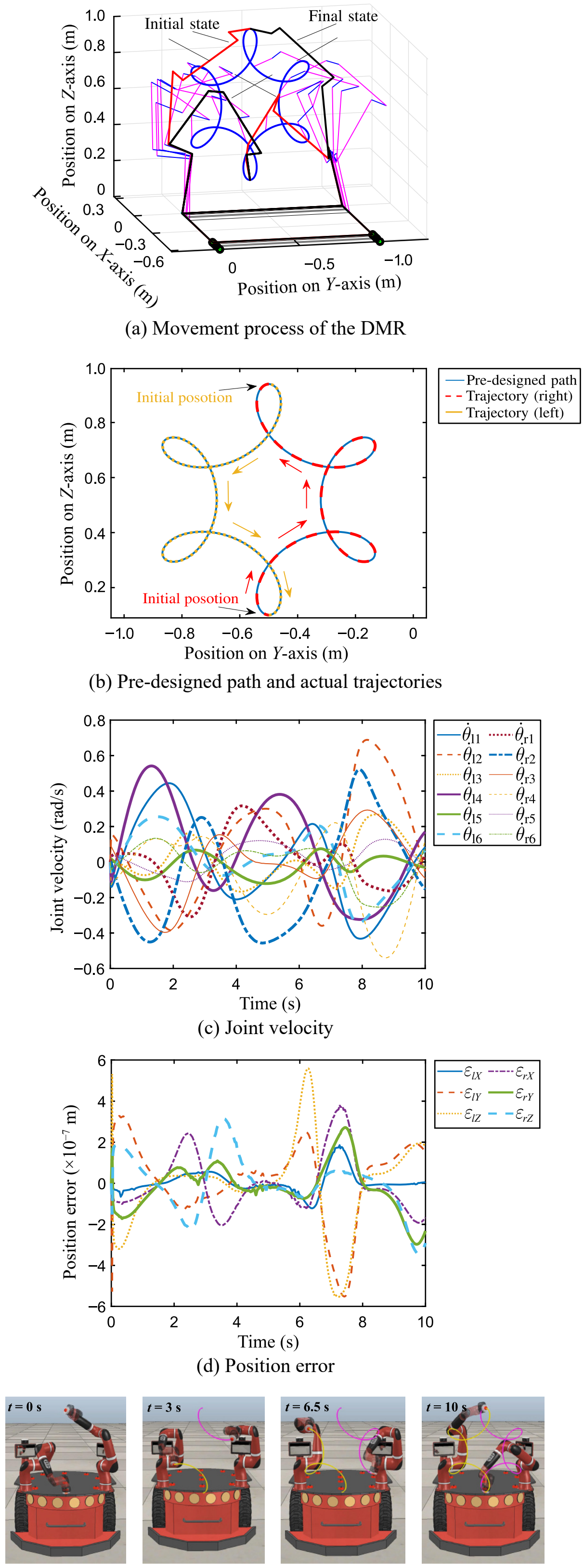
follows: $\lambda_1 = 1$, $\lambda_2 = 10$, $\lambda_3 = 0$, $\rho = 8$, $\theta_l(0) = [-1.14, -1.47, -0.44, 0.95, 0.06, 0.45]^T$ rad, and $\theta_r(0) = [1.20, -0.65, -0.33, 1.71, -0.13, 0.65]^T$ rad. The remaining parameters are the same as those in Section 4.1. The specific results are plotted in Fig. 13. In Fig. 13a, the collaboration between the two robotic arms and the mobile platform enables the successful execution of the trajectory task. Moreover, Fig. 13b demonstrates the coincidence between the actual trajectories of the two robotic arms and the pre-designed path. Additionally, the joint velocity exhibits a suitable and smooth change, as illustrated in Fig. 13c. The position error in Fig. 13d remains the order of 10^{-7} m. Besides, snapshots of the movement process in Fig. 13e verify the effective performance of the DMR in accomplishing the cooperative task. However, it is evident that if a single robotic arm executes this task, it would be constrained by its limited workspace, potentially making it difficult to reach the position on the other side. Therefore, the MRA enhances task efficiency by performing the task simultaneously and compensates for the limitations of a single robot arm, thereby improving overall performance.

5.5 Comparison of prior art

To demonstrate the novelty of this work, various robot control techniques are summarized and compared in Table 3. As displayed in Table 3, this work differs from the existing techniques in three aspects: modeling form, scheme proposal, and solver design. Firstly, this work investigates an MWDRA, where robot arms are installed at the two corners of the mobile wheelchair, providing sufficient space for the operator. However, previous studies on double-arm mobile robots[13, 15, 16] are mainly oriented toward humanoid robots, which do not reserve enough space for an operator in a wheelchair. Secondly, the proposed scheme considers the position of the wheelchair, the rotation mode of drive wheels, and joints of robotic arms to realize the arbitrary status adjustment. In contrast, previous studies[4, 5, 13, 15, 16, 31] mainly focus on the task execution of the end-effector and ignore the precise control of the mobile platform. Finally, we develop DNN of Eq. (27) incorporating physical information and prove its ability to eliminate position errors.

6 Conclusion

This paper has presented original research on a uniquely designed DMR, covering various aspects,



(a) Movement process of the DMR

(b) Pre-designed path and actual trajectories

(c) Joint velocity

(d) Position error

(e) Snapshots of movement process

Fig. 13 Simulative experiments on DMR tracking a curved hexagon trajectory driven by DNN of Eq. (27).

Table 3 Comparisons among prior art for redundancy resolution of robotic system.

Approach	Joint limit avoidance	Type of robots	Status adjustment	Number of neurons	Theoretical position error	Kinematics analysis※
PCVCC scheme of Eq. (12)	Yes	MWDRA	Yes (arbitrary)	$2m+2n+2$	Zero	Yes
Approach in Ref. [4]	Yes	Single mobile arm	Yes (restricted)☆	$m+n+2$	Nonzero	No
Approach in Ref. [5]	Yes	Single mobile arm	No	$2n+m+2$	Nonzero	No
Approach in Ref. [9]	Yes	Double fixed-based arms	Yes (restricted)☆	$2m+2n$	Zero	No
Approach in Ref. [8]	Yes	Double fixed-based arms	Yes (restricted)☆	$n^2+2m+4n$	Zero	No
Approach in Ref. [13]	No	Double-arm mobile robot	No	$2m+2n+2$	Nonzero	No
Approach in Ref. [15]	Yes	Double-arm mobile robot	No	$2m+2n+2$	Nonzero	No
Approach in Ref. [16]	No	Double-arm mobile robot	No	$2m+2n+2$	Zero	No
Approach in Ref. [21]	No	Multiple fixed-based arms§	No	$k(n+m)$	Nonzero	No
Approach in Ref. [31]	No	Single mobile arm	No	$m+2$	Zero	No

☆ This approache is able to impel the eventual states to the initial ones rather than arbitrary ones.
§ It is supposed that this approach is applied to kinematic control of k robotic arms with $k > 2$.
※ This column depends on whether the approach carries out the kinematics analyses of the MWDRA or not.

including model construction, scheme formulation, and solver design. Specifically, the DMR has been designed to incorporate two robotic arms installed at two corners of the wheeled mobile platform. On this basis, a PCVCC scheme for kinematic control of the DMR with a designed optimization criterion has been proposed. Furthermore, the gradient descent method with velocity compensation has been exploited to develop a DNN solver with zero theoretical position error. The proposed method enables precise control of the mobile platform, allowing it to move in a straight line or rotate in place, ultimately reaching a pre-designed position. Furthermore, the final joint state of the double robotic arms can be precisely determined by considering joint constraints and pre-designed tasks. Afterward, computer simulations and simulative experiments on the DMR have demonstrated the reliability and effectiveness of the proposed method. Besides, comparisons among the prior art have highlighted the novelty of the proposed method. In the future, we plan to research some important issues of DMRs, such as obstacle avoidance and constraints between two robotic arms, to guarantee the safety of DMRs.

Acknowledgment

This work was supported by the Project of "Fertilizer Robot" of Chongqing Commitee on Agriculture and Rural Afars.

References

[1] S. Xu, X. Liu, D. Li, and J. Zhao, IQABC-based hybrid deployment algorithm for mobile robotic agents providing network coverage, *Tsinghua Science and Technology*, vol. 29, no. 2, pp. 589–604, 2023.

[2] A. H. B. Boutchouang, A. Melingui, J. J. B. M. Ahanda, O. Lakhal, F. B. Motto, and R. Merzouki, Learning-based approach to inverse kinematics of wheeled mobile continuum manipulators, *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 5, pp. 3451–3462, 2022.

[3] A. H. Khan, S. Li, D. Chen, and L. Liao, Tracking control of redundant mobile manipulator: An RNN based metaheuristic approach, *Neurocomputing*, vol. 400, pp. 272–284, 2020.

[4] L. Xiao and Y. Zhang. A new performance index for the repetitive motion of mobile manipulators, *IEEE Trans. Cybern.*, vol. 44, no. 2, pp. 280–292, 2014.

[5] Z. Zhang, S. Yang, S. Chen, Y. Luo, H. Yang, and Y. Liu, A vector-based constrained obstacle avoidance scheme for wheeled mobile redundant robot manipulator, *IEEE Trans. Cogn. Dev. Syst.*, vol. 13, no. 3, pp. 465–474, 2021.

[6] A. Marino, Distributed adaptive control of networked cooperative mobile manipulators, *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 5, pp. 1646–1660, 2018.

[7] L. Jin, J. Zhao, L. Chen, and S. Li, Collective neural dynamics for sparse motion planning of redundant manipulators without Hessian matrix inversion, *IEEE Trans. Neural Network. Learn. Syst.*, vol. 36, no. 3, pp. 4326–4335, 2025.

[8] Y. Tong, J. Liu, X. Zhang, and Z. Ju, Four-criterion-optimization-based coordination motion control of dual-arm robots, *IEEE Trans. Cogn. Dev. Syst.*, vol. 15, no. 2, pp. 794–807, 2023.

[9] L. Jin and Y. Zhang, G2-type SRMPC scheme for synchronous manipulation of two redundant robot arms, *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 153–164, 2015.

[10] R. K. Al-Halimi and M. Moussa, Performing complex tasks by users with upper-extremity disabilities using a 6-DOF robotic arm: A study, *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 686–693, 2017.

[11] M. Iskandar, G. Quere, A. Hagengruber, A. Dietrich, and J. Vogel, Employing whole-body control in assistive

robotics, in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, Macau, China, 2019, pp. 5643−5650.

[12] C. Dong, Z. Yu, X. Chen, H. Chen, Y. Huang, and Q. Huang, Adaptability control towards complex ground based on fuzzy logic for humanoid robots, *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 6, pp. 1574–1584, 2022.

[13] Z. Sun, H. Yang, Y. Ma, X. Wang, Y. Mo, H. Li, and Z. Jiang, BIT-DMR: A humanoid dual-arm mobile robot for complex rescue operations, *IEEE Robot. Autom. Lett*., vol. 7, no. 2, pp. 802−809, 2022.

[14] F. Sun, Y. Chen, Y. Wu, L. Li, and X. Ren, Motion planning and cooperative manipulation for mobile robots with dual arms, *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 6, no. 6, pp. 1345–1356, 2022.

[15] T. Zhao, M. Deng, Z. Li, and Y. Hu, Cooperative manipulation for a mobile dual-arm robot using sequences of dynamic movement primitives, *IEEE Trans. Cogn. Dev. Syst.*, vol. 12, no. 1, pp. 18–29, 2020.

[16] Y. Yang, X. Wu, B. Song, and Z. Li, Whole-body fuzzy based impedance control of a humanoid wheeled robot, *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4909–4916, 2022.

[17] Q. Leboutet, E. Dean-Leon, F. Bergner, and G. Cheng, Tactile-based whole-body compliance with force propagation for mobile manipulators, *IEEE Trans. Robot.*, vol. 35, no. 2, pp. 330–342, 2019.

[18] S. Wu, Z. Li, W. Chen, and F. Sun, Dynamic modeling of robotic manipulator via an augmented deep lagrangian network, *Tsinghua Science and Technology*, vol. 29, no. 5, pp. 1604–1614, 2024.

[19] N. Kang, Z. Miao, Q. Pan, W. Li, and M. F. Tasgetiren, Multi-objective teaching-learning-based optimizer for a multi-weeding robot task assignment problem, *Tsinghua Science and Technology*, vol. 29, no. 5, pp. 1249–1265, 2024.

[20] K. Zhu and T. Zhang, Deep reinforcement learning based mobile robot navigation: A review, *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 674–691, 2021.

[21] J. Zhang, L. Jin, and Y. Wang, Collaborative control for multimanipulator systems with fuzzy neural networks, *IEEE Trans. Fuzzy Syst.*, vol. 31, no. 4, pp. 1305–1314, 2023.

[22] A. H. Khan, S. Li, and X. Luo, Obstacle avoidance and tracking control of redundant robotic manipulator: An RNN-based metaheuristic approach, *IEEE Trans. Ind. Inf.*, vol. 16, no. 7, pp. 4670–4680, 2020.

[23] A. M. Zanchettin, A. Casalino, L. Piroddi, and P. Rocco, Prediction of human activity patterns for human-robot collaborative assembly tasks, *IEEE Trans. Ind. Inf.*, vol. 15, no. 7, pp. 3934–3942, 2019.

[24] S. Li, M. C. Zhou, and X. Luo, Modified primal-dual neural networks for motion control of redundant manipulators with dynamic rejection of harmonic noises, *IEEE Trans. Neural Network. Learn. Syst.*, vol. 29, no. 10, pp. 4791–4801, 2018.

[25] L. Cheng, W. Liu, C. Yang, T. Huang, Z. G. Hou, and M. Tan, A neural-network-based controller for piezoelectric-actuated stick-slip devices, *IEEE Trans. Ind. Electron.*, vol. 65, no. 3, pp. 2598–2607, 2018.

[26] T. Sun, L. Peng, L. Cheng, Z. G. Hou, and Y. Pan, Composite learning enhanced robot impedance control, *IEEE Trans. Neural Network. Learn. Syst.*, vol. 31, no. 3, pp. 1052–1059, 2020.

[27] W. Geng, Z. Cao, P. Guan, F. Jing, M. Tan, and J. Yu, Grasp detection with hierarchical multi-scale feature fusion and inverted shuffle residual, *Tsinghua Science and Technology*, vol. 29, no. 1, pp. 244–256, 2024.

[28] D. Chen, Y. Zhang, and S. Li, Tracking control of robot manipulators with unknown models: A Jacobian-matrix-adaption method, *IEEE Trans. Ind. Inf.*, vol. 14, no. 7, pp. 3044–3053, 2018.

[29] M. Liu, F. Zhang, L. He, and M. Shang, Dynamic neural network for motion/force control of manipulators with polynomial noises, *IEEE Trans. Ind. Electron.*, vol. 71, no. 10, pp. 12559–12569, 2024.

[30] Z. Zeng, J. Wang, and X. Liao, Stability analysis of delayed cellular neural networks described using cloning templates, *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 51, no. 11, pp. 2313–2324, 2004.

[31] L. Xiao, B. Liao, S. Li, Z. Zhang, L. Ding, and L. Jin, Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator, *IEEE Trans. Ind. Inf.*, vol. 14, no. 1, pp. 98–105, 2018.

[32] L. Jin, L. Liu, X. Wang, M. Shang, and F. Y. Wang, Physical-informed neural network for MPC-based trajectory tracking of vehicles with noise considered, *IEEE Trans. Intell. Veh.*, vol. 9, no. 3, pp. 4493–4503, 2024.

[33] L. Wei and L. Jin, Collaborative neural solution for time-varying nonconvex optimization with noise rejection, *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 8, no. 4, pp. 2935–2948, 2024.

[34] B. Cai and Y. Zhang, Different-level redundancy-resolution and its equivalent relationship analysis for robot manipulators using gradient-descent and Zhang's neural-dynamic methods, *IEEE Trans. Ind. Electron.*, vol. 59, no. 8, pp. 3146–3155, 2012.

[35] H. Zhang, Z. Wang, and D. Liu, A comprehensive review of stability analysis of continuous-time recurrent neural networks, *IEEE Trans. Neural Network. Learn. Syst.*, vol. 25, no. 7, pp. 1229–1262, 2014.

[36] Z. Zeng, J. Wang, X. Liao, and X. Liao, Global exponential stability of a general class of recurrent neural networks with time-varying delays, *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.*, vol. 50, no. 10, pp. 1353–1358, 2003.

[37] S. Li, Y. Zhang, and L. Jin, Kinematic control of redundant manipulators using neural networks, *IEEE Trans. Neural Network. Learn. Syst.*, vol. 28, no. 10, pp. 2243–2254, 2017.

[38] Z. Xie, L. Jin, X. Luo, Z. Sun, and M. Liu, RNN for repetitive motion generation of redundant robot manipulators: An orthogonal projection-based scheme, *IEEE Trans. Neural Network. Learn. Syst*., vol. 33, no. 2, pp. 615–628, 2022.

[39] M. Liu, Y. Li, Y. Chen, Y. Qi, and L. Jin, A distributed

competitive and collaborative coordination for multirobot systems, *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 11436–11448, 2024.

[40] W. Li, P. W. Y. Chiu, and Z. Li, An accelerated finite-time convergent neural network for visual servoing of a flexible surgical endoscope with physical and RCM constraints, *IEEE Trans. Neural Network. Learn. Syst.*, vol. 31, no. 12, pp. 5272–5284, 2020.

[41] W. Li, Z. Su, and Z. Tan, A variable-gain finite-time convergent recurrent neural network for time-variant quadratic programming with unknown noises endured, *IEEE Trans. Ind. Inf.*, vol. 15, no. 9, pp. 5330–5340, 2019.

[42] P. S. Stanimirović and M. D. Petković, Gradient neural dynamics for solving matrix equations and their applications, *Neurocomputing*, vol. 306, pp. 200–212, 2018.

[43] L. Jin, J. Zhao, and S. Li, A data-driven sparse motion planning scheme for redundant manipulators, *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 70, no. 7, pp. 2600–2604, 2023.

[44] H. K. Khalil, *Nonlinear Systems*. 3rd ed. Englewood Cliffs, NJ, USA: Prentice Hall, 2002.

[45] J. Cornejo, E. Denegri, K. Vasquez, and O. E. Ramos, Real-time joystick teleoperation of the sawyer robot using a numerical approach, in *Proc. 2018 IEEE ANDESCON*, Santiago de Cali, Colombia, 2018, pp. 1−3.

[46] A. Nandanwar, N. K. Dhar, D. Malyshev, L. Rybak, and L. Behera, Stochastic event-based super-twisting formation control for multiagent system under network uncertainties, *IEEE Trans. Control Network. Syst.*, vol. 9, no. 2, pp. 966–978, 2022.

Diwen Xiong received the MEng degree from George Washington University, DC, USA in 2018. He is currently an engineer at Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China. His research interests are in robotics, neural networks, and artificial intelligence.

Liuyi Wen received the BEng degree from Jiangnan University, China in 2018. She is currently as master student at Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China. Her main research interests include robotics, neural networks, artificial intelligence, and intersection of computers and design.

Mingsheng Shang received the PhD degree in computer science from University of Electronic Science and Technology of China, Chengdu, China in 2007. He is currently a professor at Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, China. His research interests are in complex network analysis and big data applications.