2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Robust and Remote Center of Cyclic Motion Control for Redundant Robots with Partially Unknown Structure

Long Jin[1], Kun Liu[1], and Mei Liu[1]

***Abstract*— Remote center of motion (RCM) describes a robot with a rod-like end-effector operating through a hole in the interface separating the internal space from the external space. Considering that the control of RCM may be influenced by perturbations (noises) and that the end-effector is frequently replaced to complete different tasks, the structural information related to the robot manipulator and its rod-like end-effector may contain errors. This paper proposes an acceleration-level remote center of cyclic motion (ARC^{2}M) control scheme, which takes into account the cyclic motion index and the physical limitations of robot manipulators to achieve repetitive motion planning and RCM control at the acceleration level. Additionally, a parameter calculation method is proposed to compute unknown parameters of the end-effector under the influence of noise. Kalman filter and a neural dynamics-based method are employed to address noises effects, and related theoretical analyses are given. To validate the proposed ARC^{2}M scheme, simulations and physical experiments are carried out. The source code is available at https://github.com/LongJin-lab/ARCM.**

I. Introduction

Remote center of motion (RCM) is often utilized in space-constrained operations of robots. It refers to a fixed point devoid of any physical revolute joints around which a robot or its part can rotate. Specifically, the RCM problem involves controlling end-effector movements in the internal space by managing robot movements in the external space through a small hole in the interface between both spaces. RCM has received considerable attention across different domains, especially in the field of surgery, including robot-assisted minimally invasive laparotomy [1], brain surgery [2], eye surgery [3] and nasal sampling [4].

To improve the performance in robot-assisted surgery, an RCM constraint is used in [5] to implement the trajectory control of redundant manipulators. In [6], a neural-learning-enhanced Cartesian admittance control scheme is presented for a robot manipulator to handle dynamic environments with moving RCM constraints. These RCM control schemes are all designed at the velocity level, which may generate discontinuous joint velocities and do not consider the joint-acceleration limits. It is not suitable for direct application to robots. To remedy the weaknesses of joint-velocity level RCM control, this paper is dedicated to constructing joint-acceleration one with noises considered.

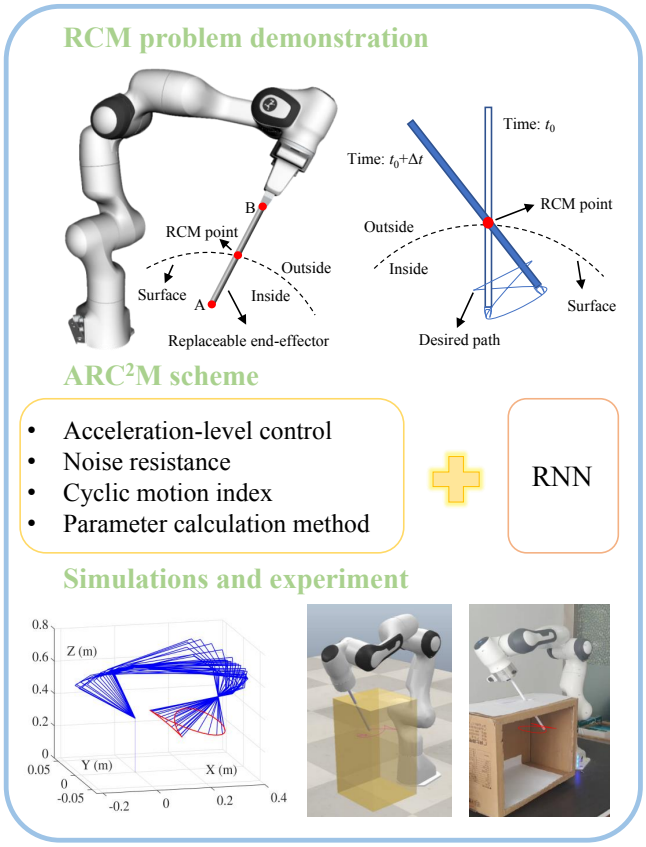

Fig. 1. RCM problem description, algorithmic framework, and presentation of results.

During a robot-assisted minimally invasive surgery, the robot end-effector needs to be changed frequently, which can lead to errors in the structural information of the robot. A control method for surgical robots based on data-driven technology is presented in [7]. A data-driven framework is presented in [8] for the optimal view control of endoscopes. Moreover, in [9], a data-driven technology is employed to achieve precise cyclic motion of robots with unknown structures. However, the data-driven technology requires the entire Jacobian matrix to be updated at every iteration, which consumes a significant amount of computing burden. In the case of medical robot manipulators, the error due to wear is negligible, and only the error caused by the change of the end-effector needs to be considered. Therefore, this paper

This work was supported in part by the National Natural Science Foundation of China under Grant 62176109 and Grant 62306130, in part by the Fundamental Research Funds for the Central Universities under Grant lzujbky-2023-eyt04, and Grant lzujbky-2022-23, in part by the Natural Science Foundation of Gansu Province under Grant 21JR7RA531 and Grant 22JR5RA427, in part by the CAAI-Huawei MindSpore Open Fund under Grant CAAIXSJLJJ-2022-020A, and in part by the Supercomputing Center of Lanzhou University. *(Corresponding authors: Long Jin and Mei Liu.)*

[1]The authors are with the School of Information Science and Engineering, Lanzhou University, Lanzhou, China. {jinlongsysu,liumei7}@foxmail.com

proposes a parameter calculation method to compute the end-effector's parameters based on its actual position, which can obtain accurate parameters of the end-effector while reducing the computational burden.

During the operation of a robot, numerous sources of perturbations (noises) can be encountered [10]. Noises have a detrimental effect on the precise control and repetitive motion planning of the robot during robot-assisted surgery, which could result in joint drift. The effect of environmental noises on the performance of robot-assisted laparoscopic surgery is studied and confirmed in [11]. Therefore, extensive research is conducted to counteract the effects of noises on robotic control. To solve the noise problem in robotic control, a noise-suppressing neural dynamics (NSND) solver is presented to eliminate lagging errors in the solution process with the presence of noises [12], [13]. Inspired by the NSND solver, this paper proposes a neural dynamics-based method for solving the acceleration-level remote center of cyclic motion (ARC^{2}M) scheme aided with a Kalman filter with noises considered.

Based on the above insights, this paper proposes the acceleration-level remote center of cyclic motion (ARC^{2}M) scheme for dealing with RCM control of a robot with unknown structural information under the influence of noises. The main contributions of this paper are summarised below.

- A novel parametric calculation method is proposed, which enables the precise control of the robot manipulator by accurately computing parameters of the end-effector of the robot manipulator while reducing the computational burden.
- An ARC^{2}M scheme is constructed, which is further complemented by the proposed neural dynamic-based solver and Kalman filter to mitigate the impact of noises.

II. Preliminaries and Scheme Construction

In this section, the forward kinematics of robots, the description of the RCM problem, the construction of the ARC^{2}M scheme, and the parametric calculation methods are given, which form the basis for subsequent work.

A. Forward Kinematics

There are two coordinate systems utilized in the kinematic control of a robot manipulator, i.e., Cartesian space $\mathbf{r}(t) \in \mathbb{R}^m$ and joint space $\boldsymbol{\theta}(t) \in \mathbb{R}^n$, which are crucial when discussing robotic control problems. The relationship between these two coordinate systems at different levels is shown as:$\mathbf{r}(t) = \Gamma(\boldsymbol{\theta}(t)), \dot{\mathbf{r}}(t) = J(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}, \ddot{\mathbf{r}}(t) = \dot{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} + J(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}}$, where $\Gamma(\cdot)$ is the transformation function from joint space to Cartesian space; $\dot{\mathbf{r}}$ and $\ddot{\mathbf{r}}$ denote the velocity and acceleration of the robot manipulator's end-effector, respectively; $\dot{\boldsymbol{\theta}}$ and $\ddot{\boldsymbol{\theta}}$ denote the angular velocity and the angular acceleration of the joint, respectively; $J(\boldsymbol{\theta}) = \partial\Gamma(\boldsymbol{\theta})/\partial\boldsymbol{\theta} \in \mathbb{R}^{m\times n}$ is the Jacobian matrix of the robot manipulator; $\dot{J}$ is the time derivative of J.

In addition, considering the physical limitations of the robot motor, restrictions are imposed to maintain joint properties within reasonable limits to protect the robot manipulator: $\boldsymbol{\theta}^- \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}^+, \dot{\boldsymbol{\theta}}^- \leq \dot{\boldsymbol{\theta}} \leq \dot{\boldsymbol{\theta}}^+, \ddot{\boldsymbol{\theta}}^- \leq \ddot{\boldsymbol{\theta}} \leq \ddot{\boldsymbol{\theta}}^+$, where $\boldsymbol{\theta}^-$, $\dot{\boldsymbol{\theta}}^-$, and $\ddot{\boldsymbol{\theta}}^-$ stand for the lower bounds of the joint angle, velocity, and acceleration, respectively; $\boldsymbol{\theta}^+$, $\dot{\boldsymbol{\theta}}^+$, and $\ddot{\boldsymbol{\theta}}^+$ denote the upper bounds of the joint angle, velocity, and acceleration, respectively.

B. RCM Problem Description and Scheme Construction

For a more precise representation, point A and point B are defined as the head and tail of the end-effector, respectively, as shown in Fig. 1. Ideally, the three points A, B, and the RCM point should fall in a straight line. Therefore the ideal relationship between point A, point B, and the RCM point can be expressed as

$$\mathbf{r}_\mathrm{c} - \mathbf{r}_\mathrm{b} = k(\mathbf{r}_\mathrm{a} - \mathbf{r}_\mathrm{b}), \tag{1}$$

where $\mathbf{r}_\mathrm{a}$, $\mathbf{r}_\mathrm{b}$, and $\mathbf{r}_\mathrm{c}$ are the positions of point A, point B and RCM point, respectively; $0 \leq k \leq 1$ is a parameter used to constrain RCM point on line AB. The kinematic relationship between joint angles and their corresponding Cartesian coordinates can be expressed as $\mathbf{r}_\mathrm{a} = \Gamma_\mathrm{a}(\boldsymbol{\theta}(t))$, $\dot{\mathbf{r}}_\mathrm{a} = J_\mathrm{a}\dot{\boldsymbol{\theta}}$, $\mathbf{r}_\mathrm{b} = \Gamma_\mathrm{b}(\boldsymbol{\theta}(t))$, and $\dot{\mathbf{r}}_\mathrm{b} = J_\mathrm{b}\dot{\boldsymbol{\theta}}$, where $J_\mathrm{a} \in \mathbb{R}^{m\times(n+1)}$ and $J_\mathrm{b} \in \mathbb{R}^{m\times n}$ are Jacobian matrices of the robot with and without replaceable end-effector, respectively.

Since point A, point B, and the RCM point are in a straight line, equation (1) should be met. To obtain the acceleration-level constraint, equation (1) should take the second time derivative:

$$\ddot{\mathbf{r}}_\mathrm{c} - \ddot{\mathbf{r}}_\mathrm{b} = \ddot{k}(\mathbf{r}_\mathrm{a} - \mathbf{r}_\mathrm{a}) + \dot{k}(\dot{\mathbf{r}}_\mathrm{a} - \dot{\mathbf{r}}_\mathrm{b}) + k(\ddot{\mathbf{r}}_\mathrm{a} - \ddot{\mathbf{r}}_\mathrm{b}), \tag{2}$$

where $\dot{\mathbf{r}}_\mathrm{a}$, $\dot{\mathbf{r}}_\mathrm{b}$, and $\dot{\mathbf{r}}_\mathrm{c}$ are velocities of point A, point B, and the RCM point, respectively; $\ddot{\mathbf{r}}_\mathrm{a}$, $\ddot{\mathbf{r}}_\mathrm{b}$, and $\ddot{\mathbf{r}}_\mathrm{c}$ are the acceleration of point A, point B, and the RCM point, respectively; $\dot{k}$ and $\ddot{k}$ are the first and the second time derivative of k, respectively.

According to the relationship between these two coordinates, equation (2) can be converted into:

$$\begin{aligned}\ddot{\mathbf{r}}_\mathrm{c} = {} & \ddot{k}(\mathbf{r}_\mathrm{a} - \mathbf{r}_\mathrm{b}) + (2\dot{k}(J_\mathrm{a} - J_\mathrm{b}) + k\dot{J}_\mathrm{a} + (1-k)\dot{J}_\mathrm{b})\dot{\boldsymbol{\theta}} \\ & + (kJ_\mathrm{a} + (1-k)J_\mathrm{b})\ddot{\boldsymbol{\theta}},\end{aligned} \tag{3}$$

and $\ddot{\mathbf{r}}_\mathrm{c}$ should be converted to a convergent expression to minimize the position change of the RCM point:

$$\ddot{\mathbf{r}}_\mathrm{c} = -\sigma(\dot{\mathbf{r}}_\mathrm{c} - \dot{\mathbf{r}}_\mathrm{c}(0)) = -\sigma^2(\mathbf{r}_\mathrm{c} - \mathbf{r}_\mathrm{c}(0)), \tag{4}$$

where σ is a positive parameter controlling the convergence rate of $\ddot{\mathbf{r}}_\mathrm{c}$. Then, the RCM constraint at the acceleration level is expressed as

$$\begin{aligned}\mathbf{0} = {} & \ddot{k}(\mathbf{r}_\mathrm{a} - \mathbf{r}_\mathrm{b}) + (2\dot{k}(J_\mathrm{a} - J_\mathrm{b}) + k\dot{J}_\mathrm{a} + (1-k)\dot{J}_\mathrm{b})\dot{\boldsymbol{\theta}} \\ & + (kJ_\mathrm{a} + (1-k)J_\mathrm{b})\ddot{\boldsymbol{\theta}} + \sigma^2(\mathbf{r}_\mathrm{c} - \mathbf{r}_\mathrm{c}(0)).\end{aligned} \tag{5}$$

The kinematic constraint can be expressed as

$$\mathbf{0} = J_\mathrm{a}\ddot{\boldsymbol{\theta}} - \ddot{\mathbf{r}}_\mathrm{d} + \dot{J}_\mathrm{a}\dot{\boldsymbol{\theta}}, \tag{6}$$

where $\ddot{\mathbf{r}}_\mathrm{d}$ is the desired acceleration of the end-effector.

To sum up, the ARC^{2}M scheme is formulated as follows:

$$\begin{aligned}
&\min_{\ddot{\boldsymbol{\theta}},\ddot{k}} \ \frac{1}{2}\ddot{\boldsymbol{\theta}}^{\mathrm{T}}\ddot{\boldsymbol{\theta}} + (\alpha\dot{\boldsymbol{\theta}} + \beta(\boldsymbol{\theta}-\boldsymbol{\theta}_0))^{\mathrm{T}}\ddot{\boldsymbol{\theta}} + \frac{m_2\ddot{k}^2}{2} + \frac{m_1\dot{k}^2}{2},\\
&\text{s.t. } \mathbf{0} = \ddot{k}(\mathbf{r}_{\mathrm{a}}-\mathbf{r}_{\mathrm{b}}) + (k\widetilde{J}_{\mathrm{a}} + (1-k)J_{\mathrm{b}})\ddot{\boldsymbol{\theta}}\\
&\qquad + \sigma^2(\mathbf{r}_{\mathrm{c}}-\mathbf{r}_{\mathrm{c}}(0))\\
&\qquad + (2\dot{k}(\widetilde{J}_{\mathrm{a}}-J_{\mathrm{b}}) + k\dot{\widetilde{J}}_{\mathrm{a}} + (1-k)\dot{J}_{\mathrm{b}})\dot{\boldsymbol{\theta}},\\
&\qquad \mathbf{0} = \widetilde{J}_{\mathrm{a}}\ddot{\boldsymbol{\theta}} - \ddot{\mathbf{r}}_{\mathrm{d}} + \dot{\widetilde{J}}_{\mathrm{a}}\dot{\boldsymbol{\theta}} - d_0(\mathbf{r}_{\mathrm{d}}-\mathbf{r}_{\mathrm{a}}) - d_1(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}_{\mathrm{a}}),
\end{aligned} \tag{7}$$

where α and β are positive parameters for the cyclic motion; $\widetilde{J}_{\mathrm{a}}$ is the estimated Jacobian matrix of point A; J_{b} is the Jacobian matrix of point B. It is worth pointing out that $\widetilde{J}_a$ is an estimated matrix, and J_{b} is a known one. Since the RCM point is expected to have as little displacement as possible, $m_2\ddot{k}^2/2 + m_1\dot{k}^2/2$ is added to avoid the unnecessary movement of the RCM point during the task, and m_1, m_2 are set to control convergence performance. Besides, $d_0(\mathbf{r}_{\mathrm{d}}-\mathbf{r}_{\mathrm{a}}) - d_1(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}_{\mathrm{a}})$ is added to improve the tracking accuracy.

C. Parameter Calculation Method

To achieve the transformation from the joint angle coordinate to the Cartesian coordinate, the Denavit-Hartenberg (D-H) method is necessary. This method fixes a coordinate system on each linkage of the robot, and then a four-dimensional matrix is given to describe the relationship between two adjacent linkages. The modified D-H matrix is depicted as

$${}^{n-1}T_n = \left[\begin{array}{cccc} \mathrm{c}\theta & -\mathrm{s}\theta & 0 & a \\ \mathrm{s}\theta\times\mathrm{c}\alpha & \mathrm{c}\theta\times\mathrm{c}\alpha & -\mathrm{s}\alpha & -d\times\mathrm{s}\alpha \\ \mathrm{s}\theta\times\mathrm{s}\alpha & \mathrm{c}\theta\times\mathrm{s}\alpha & \mathrm{c}\alpha & d\times\mathrm{c}\alpha \\ \hline 0 & 0 & 0 & 1 \end{array}\right], \tag{8}$$

where ${}^{n-1}T_n$ denotes the transformation matrix from $(n-1)$-th joint to n-th joint; a, α, d, and θ are parameters that describe the spatial relationship between two adjacent linkages [14]; $\mathrm{s}\alpha$ and $\mathrm{c}\alpha$ are abbreviations for $\sin\alpha$ and $\cos\alpha$, respectively, to simplify the expression.

This paper treats the replaceable end-effector as a new linkage, which means the end-effector also has a transformation matrix ${}^{n}T_{n+1}$. The total transformation matrix of the robot can be obtained by multiplying the transformation matrices of all linkages:

$$\begin{aligned}
T &= {}^0T_1 \times {}^1T_2 \times \ldots \times {}^{n-1}T_n,\\
T_{n+1} &= T \times {}^nT_{n+1} = \begin{bmatrix} R & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix},
\end{aligned} \tag{9}$$

where T and T_{n+1} are the total transformation matrices of the robot without and with the replaceable end-effector, respectively; $R \in \mathbb{R}^{3\times3}$ is robot's rotation matrix; $\mathbf{p} \in \mathbb{R}^{3\times1}$ is robot's rotation matrix and position vector. Combined with the end-effector position feedback, the parameters of the replaceable end-effector are computed by

$$\mathbf{r}_{\mathrm{d}}(t) = \mathbf{p}(t), \tag{10}$$

where $\mathbf{r}_{\mathrm{d}}(t)$ is the desired position of end-effectors at time t.

III. Solution and Analyses

In this section, a neural dynamics-based method is constructed to solve the proposed ARC^{2}M scheme (7) and the parameter calculation method under the effect of the linear noise. Then, a Kalman filter is employed to address the impact of Gaussian noise. In the end, related analyses and proofs of ARC^{2}M scheme's validity are given.

A. Neural Dynamics-Based Method

To solve the ARC^{2}M scheme, the Lagrangian function of (7) can be constructed as

$$\begin{aligned}
\mathcal{L} &= \frac{1}{2}\ddot{\boldsymbol{\theta}}^{\mathrm{T}}\ddot{\boldsymbol{\theta}} + (\alpha\dot{\boldsymbol{\theta}} + \beta(\boldsymbol{\theta}-\boldsymbol{\theta}_0))\ddot{\boldsymbol{\theta}} + \frac{m_2\ddot{k}^2}{2} + \frac{m_1\dot{k}^2}{2}\\
&+ \boldsymbol{\lambda}_1^{\mathrm{T}}(\widetilde{J}_{\mathrm{a}}\ddot{\boldsymbol{\theta}} - \ddot{\mathbf{r}}_{\mathrm{d}} + \dot{\widetilde{J}}_{\mathrm{a}}\dot{\boldsymbol{\theta}} - d_0(\mathbf{r}_{\mathrm{d}}-\mathbf{r}_{\mathrm{a}}) - d_1(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}_{\mathrm{a}}))\\
&+ \boldsymbol{\lambda}_2^{\mathrm{T}}(\ddot{k}(\mathbf{r}_{\mathrm{a}}-\mathbf{r}_{\mathrm{b}}) + (k\widetilde{J}_{\mathrm{a}} + (1-k)J_{\mathrm{b}})\ddot{\boldsymbol{\theta}} + \sigma^2(\mathbf{r}_{\mathrm{c}}-\mathbf{r}_{\mathrm{c}}(0))\\
&+ (2\dot{k}(\widetilde{J}_{\mathrm{a}}-J_{\mathrm{b}}) + k\dot{\widetilde{J}}_{\mathrm{a}} + (1-k)\dot{J}_{\mathrm{b}})\dot{\boldsymbol{\theta}}),
\end{aligned} \tag{11}$$

where $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ are Lagrange multipliers; superscript $^{\mathrm{T}}$ denotes the transpose of a matrix or vector. Using the Lagrange multiplier method, equation (11) is converted as

$$\begin{aligned}
\frac{\partial\mathcal{L}}{\partial\ddot{\boldsymbol{\theta}}} &= \ddot{\boldsymbol{\theta}} + \alpha\dot{\boldsymbol{\theta}} + \beta(\boldsymbol{\theta}-\boldsymbol{\theta}(0)) + \widetilde{J}_{\mathrm{a}}^{\mathrm{T}}\boldsymbol{\lambda}_1\\
&\quad + (k\widetilde{J}_{\mathrm{a}} + (1-k)J_{\mathrm{b}})^{\mathrm{T}}\boldsymbol{\lambda}_2,\\
\frac{\partial\mathcal{L}}{\partial\ddot{k}} &= m_1\dot{k} + m_2\ddot{k} + \boldsymbol{\lambda}_2^{\mathrm{T}}(\mathbf{r}_{\mathrm{a}}-\mathbf{r}_{\mathrm{b}}),\\
\frac{\partial\mathcal{L}}{\partial\ddot{\boldsymbol{\lambda}}_1} &= \widetilde{J}_{\mathrm{a}}\ddot{\boldsymbol{\theta}} - \ddot{\mathbf{r}}_{\mathrm{d}} + \dot{\widetilde{J}}_{\mathrm{a}}\dot{\boldsymbol{\theta}} - d_0(\mathbf{r}_{\mathrm{d}}-\mathbf{r}_{\mathrm{a}}) - d_1(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}_{\mathrm{a}}),\\
\frac{\partial\mathcal{L}}{\partial\ddot{\boldsymbol{\lambda}}_2} &= \ddot{k}(\mathbf{r}_{\mathrm{a}}-\mathbf{r}_{\mathrm{b}}) + (k\widetilde{J}_{\mathrm{a}} + (1-k)J_{\mathrm{b}})\ddot{\boldsymbol{\theta}} + \sigma^2(\mathbf{r}_{\mathrm{c}}-\mathbf{r}_{\mathrm{c}}(0)),\\
&\quad + (2\dot{k}(\widetilde{J}_{\mathrm{a}}-J_{\mathrm{b}}) + k\dot{\widetilde{J}}_{\mathrm{a}} + (1-k)\dot{J}_{\mathrm{b}})\dot{\boldsymbol{\theta}}.
\end{aligned} \tag{12}$$

To get the solution of scheme (7), equation (12) is written as $M(t)\mathbf{y}(t) = \mathbf{a}(t)$, with

$$\begin{aligned}
M(t) &= \begin{bmatrix} E & 0 & \widetilde{J}_{\mathrm{a}}^{\mathrm{T}} & w \\ 0 & m_2 & 0 & (\mathbf{r}_{\mathrm{a}}-\mathbf{r}_{\mathrm{b}})^{\mathrm{T}} \\ \widetilde{J}_{\mathrm{a}} & 0 & 0 & 0 \\ w & \mathbf{r}_{\mathrm{a}}-\mathbf{r}_{\mathrm{b}} & 0 & 0 \end{bmatrix}, \mathbf{y}(t) = \begin{bmatrix} \ddot{\boldsymbol{\theta}} \\ \ddot{k} \\ \boldsymbol{\lambda}_1 \\ \boldsymbol{\lambda}_2 \end{bmatrix},\\
\mathbf{a}(t) &= \begin{bmatrix} -(\alpha\dot{\boldsymbol{\theta}} + \beta(\boldsymbol{\theta}-\boldsymbol{\theta}(0))) \\ -m_1\dot{k} \\ \ddot{\mathbf{r}}_{\mathrm{d}} - \dot{\widetilde{J}}_{\mathrm{a}}\dot{\boldsymbol{\theta}} + d_0(\mathbf{r}_{\mathrm{d}}-\mathbf{r}_{\mathrm{a}}) + d_1(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}_{\mathrm{a}}) \\ v \end{bmatrix},
\end{aligned} \tag{13}$$

where $w = k\widetilde{J}_{\mathrm{a}} + (1-k)J_{\mathrm{b}}$ and $v = -(2\dot{k}(\widetilde{J}_{\mathrm{a}}-J_{\mathrm{b}}) + k\dot{\widetilde{J}}_{\mathrm{a}} + (1-k)\dot{J}_{\mathrm{b}})\dot{\boldsymbol{\theta}} - \sigma^2(\mathbf{r}_{\mathrm{c}}-\mathbf{r}_{\mathrm{c}}(0))$. By constructing the error function for equation (13), the following formula can be derived:

$$\boldsymbol{\xi}(t) = M(t)\mathbf{y}(t) - \mathbf{a}(t). \tag{14}$$

During the operation of a robot, there inevitably are some noises that impact the control performance. To solve the problem online, with the aid of the design formula depicted in $\dot{\boldsymbol{\xi}}(t) = -(\lambda+\gamma)\boldsymbol{\xi}(t) - \lambda\gamma\int_0^t\boldsymbol{\xi}(\tau)\mathrm{d}\tau$, and take into

account the effect of noises, a neural dynamics-based method is constructed as

$$\begin{aligned} M(t)\dot{\mathbf{y}}(t) = &\dot{\mathbf{a}}(t) - \dot{M}(t)\mathbf{y}(t) - \lambda(M(t)\mathbf{y}(t) - \mathbf{a}(t)) \\ &- \gamma(M(t)\mathbf{y}(t) - \mathbf{a}(t)) \\ &+ \gamma\lambda \int_0^t (M(\tau)\mathbf{y}(\tau) - \mathbf{a}(\tau))\mathrm{d}\tau + \boldsymbol{\eta}(t). \end{aligned} \tag{15}$$

where λ and γ are control parameters; $\boldsymbol{\eta}(t)$ is the noises during the motion of the robot. Considering that $M(t)$ is a time-varying matrix, there exist deviations between its theoretical and actual values, which are regarded as noises.

Similarly, parameters of the end-effector can be calculated from the actual position of the end-effector under noise conditions as shown in (10), which can be further written as $\widetilde{\mathbf{p}}(t) = [\mathbf{p}(t);\ 1]$, where $\widetilde{\mathbf{p}}(t)$ is the last column of $T \times {}^{n}T_{n+1}$ and $\mathbf{p}(t) \in \mathbb{R}^{3\times 1}$ is the position vector of end-effector. It is worth noting that $\widetilde{\mathbf{p}}(t)$ is an estimated value. By calculating $\partial\widetilde{\mathbf{p}}(t)/\partial\theta$, $\partial\widetilde{\mathbf{p}}(t)/\partial\alpha$, and $\partial\widetilde{\mathbf{p}}(t)/\partial d$, an equation like (12) can be obtained. As this formula is too lengthy, it will not be expanded upon here. At the last, the neural dynamics-based method for the parameter calculation method can be expressed as

$$\begin{aligned} T(t)\dot{\mathbf{z}}(t) = &\dot{\mathbf{b}}(t) - \dot{T}(t)\mathbf{z}(t) - \lambda(T(t)\mathbf{z}(t) - \mathbf{b}(t)) \\ &- \gamma(T(t)\mathbf{z}(t) - \mathbf{b}(t)) \\ &+ \gamma\lambda \int_0^t (T(\tau)\mathbf{z}(\tau) - \mathbf{b}(\tau))\mathrm{d}\tau + \boldsymbol{\mu}(t), \end{aligned} \tag{16}$$

where $T(t)$, $\mathbf{z}(t)$, and $\mathbf{b}(t)$ denote the parameter matrix, the parameter vector of the end-effector and the parameter vector, which correspond to $M(t)$, $\mathbf{y}(t)$, and $\mathbf{a}(t)$ in (15), respectively; $\boldsymbol{\mu}(t)$ is the noise during the motion of the robot.

B. Kalman Filter

Apart from the noise during the robot operation, there is also Gaussian noise generated during the measurement process and so on. Therefore, the Kalman filter is employed to eliminate the errors caused by Gaussian noise in the measurement process. The continuous Kalman filter model is as follows:

$$\begin{aligned} K &= PHP^{-1}, \\ \dot{\boldsymbol{X}} &= F\boldsymbol{X} + G\boldsymbol{W} + K\boldsymbol{V}, \\ \dot{P} &= FP + PF^{\mathrm{T}} + Q - KRK^{\mathrm{T}}, \end{aligned} \tag{17}$$

where $\boldsymbol{X}$ is the state information of the robot at time t; $\boldsymbol{W}$ and $\boldsymbol{V}$ are the input white noise and the observation noise, respectively, with the mean value of 0 and variance of Q and R; K is the Kalman gain; P is the covariance; F is the state transfer matrix; H is used to control the observation values.

In the above continuous Kalman filter, $\boldsymbol{X}$ as the input corresponds to the output of the neural dynamics-based method $\dot{\boldsymbol{y}}$. The step of predicting $\boldsymbol{X}$ in equation (17) is already done by the neural dynamics-based method, so X can be directly used as the input.

C. Theoretical Analyses

In this section, several essential theorems are given and proved to demonstrate the validity of the proposed ARC^{2}M scheme (7) and the effectiveness of the proposed neural dynamics-based method in eliminating the impact of noises.

Theorem 1: For the solution of ARC^{2}M scheme (7), its variable $\mathbf{y}(t) = [\ddot{\boldsymbol{\theta}};\ \ddot{k};\ \boldsymbol{\lambda}_1;\ \boldsymbol{\lambda}_2]$ converges globally to the desired value. That is, the position of the RCM point $\mathbf{r}_\mathrm{c}$ and the angular acceleration of joint $\ddot{\theta}$ converge to their desired values.

Proof: First, construct the following equation:

$$\boldsymbol{\delta}(t) = \boldsymbol{\xi}(t) + \lambda \int_0^t \boldsymbol{\xi}(\tau)\mathrm{d}\tau, \tag{18}$$

and the time derivative of $\boldsymbol{\xi}(t)$ can be expressed as

$$\dot{\boldsymbol{\delta}}(t) = \dot{\boldsymbol{\xi}}(t) + \lambda\boldsymbol{\xi}(t). \tag{19}$$

Considering that $\boldsymbol{\delta}(t)$ encompasses $\ddot{\theta}$ and $\ddot{k}$, it is imperative for both $\ddot{\theta}$ and $\ddot{k}$ to globally converge to their optimal values, provided that $\boldsymbol{\delta}(t)$ is demonstrated to globally converge to 0. Combining (19) and NSND solver (15) yields $\dot{\boldsymbol{\delta}}(t) = -\sigma\boldsymbol{\delta}(t)$. Introducing the Lyapunov function:

$$\mathcal{L}_1(t) = \boldsymbol{\delta}^{\mathrm{T}}(t)\boldsymbol{\delta}(t)/2, \tag{20}$$

Combining the Lyapunov theory, it can be deduced that $\mathcal{L}_1(t) > 0$ when $\boldsymbol{\delta}(t) \neq 0$, which means that $\mathcal{L}_1(t)$ is nonnegative. Then, the time derivative of $\mathcal{L}_1(t)$ can be expressed as

$$\dot{\mathcal{L}}_1(t) = -\sigma\boldsymbol{\delta}^{\mathrm{T}}(t)\boldsymbol{\delta}(t) \leq 0. \tag{21}$$

Therefore, $\boldsymbol{\delta}(t)$ converges globally to 0. From equation (19), it can be obtained:

$$\dot{\boldsymbol{\xi}}(t) = -\lambda\boldsymbol{\xi}(t). \tag{22}$$

The Lyapunov function can eventually be expressed as

$$\mathcal{L}_2(t) = \boldsymbol{\xi}^{\mathrm{T}}(t)\boldsymbol{\xi}(t)/2. \tag{23}$$

In this sense, the error function $\boldsymbol{\xi}(t)$ for the solution of the ARC^{2}M scheme (7) converges globally to zero. Specifically, the position of the RCM point $\mathbf{r}_\mathrm{c}$ and the angular acceleration of joint $\ddot{\theta}$ converge to their desired values, which means the successful execution of the task. The proof is thus completed.

Theorem 2: When applying the proposed ARC^{2}M scheme (7) to an arbitrary bounded noise $I(t)$ scenario, the variable $\mathbf{y}(t) = [\ddot{\boldsymbol{\theta}};\ \ddot{k};\ \boldsymbol{\lambda}_1;\ \boldsymbol{\lambda}_2]$ converges globally to the optimal value. That is, the position of the RCM point $\mathbf{r}_\mathrm{c}$ and the angular acceleration of joint $\ddot{\boldsymbol{\theta}}$ converge to their desired values.

Proof: In the NSND solver, the derivative of the error function is defined as $\dot{\boldsymbol{\xi}}(t) = -(\lambda+\gamma)\boldsymbol{\xi}(t) - \lambda\gamma\int_0^t \boldsymbol{\xi}(\tau)\mathrm{d}\tau$. Combining this with equation (14), the i-th element of the NSND solver can be expressed as

$$\dot{\boldsymbol{\xi}}_i(t) = -(\lambda+\gamma)\boldsymbol{\xi}_i(t) - \lambda\gamma\int_0^t \boldsymbol{\xi}_{i-1}(\tau)\mathrm{d}\tau + I_i(t). \tag{24}$$

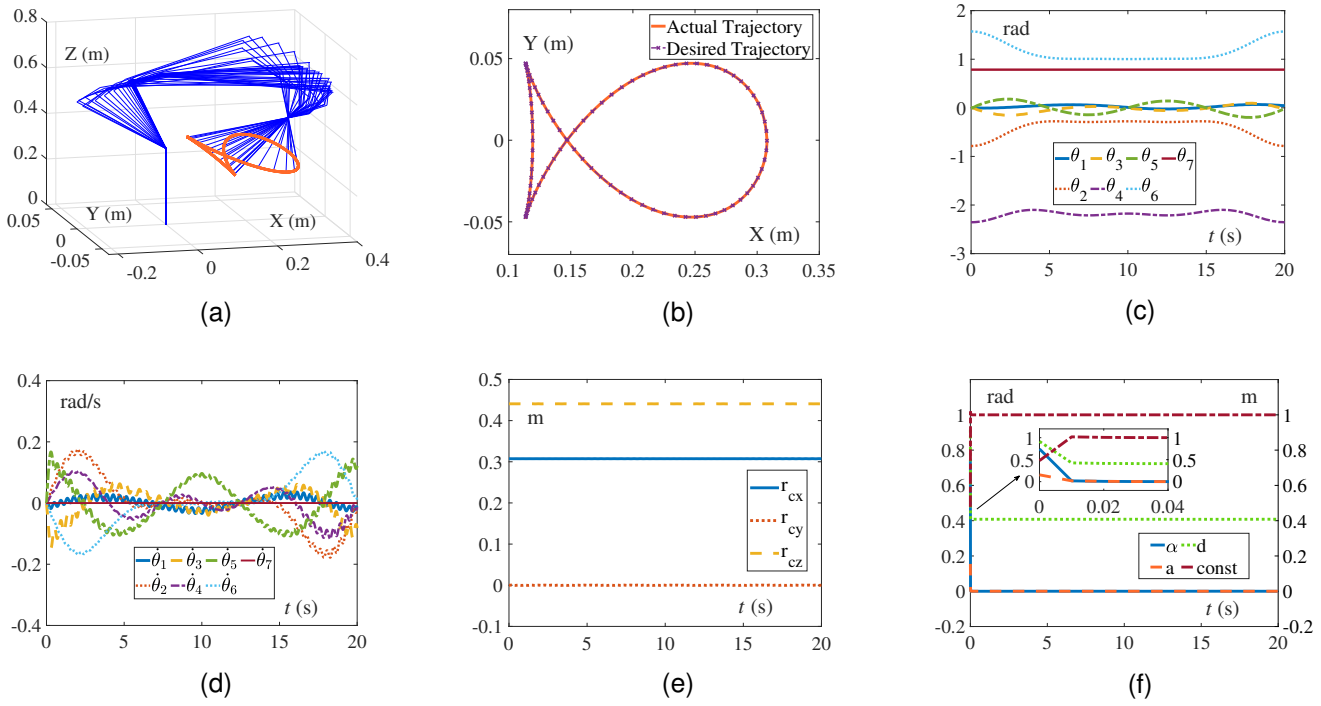

Fig. 2. Simulation results of a Franka Emika Panda robot with rod end-effector following a fish-shaped trajectory in a limited space, guided by the proposed ARC^{2}M scheme (7). (a) The trajectory of the robot. (b) Expected and actual trajectories of the end-effector. (c) Joint angles. (d) Joint velocities. (e) Position of the RCM point. (f) Parameters of the replaceable end-effector.

Using the Laplace transform, equation (24) is transformed as

$$\Gamma_i(s) = \frac{s}{s^2 + (\lambda+\gamma)s + \lambda\gamma} I_i(s), \tag{25}$$

where $\Gamma_i(s)$ and $I_i(s)$ are the Laplace transform of $\boldsymbol{\xi}_i(t)$ and $I_i(t)$, respectively. Laplace transform $L_i(s)$ can be defined as

$$L_i(s) = \frac{s}{s^2 + (\lambda+\gamma)s + \lambda\gamma}. \tag{26}$$

By defining $s = jw$, $|L_i(jw)|^2$ is written as

$$|L_i(jw)|^2 = \frac{1}{w^2 + \frac{\lambda^2\gamma^2}{w^2} + (\lambda^2+\gamma^2)}. \tag{27}$$

From the mean value inequality, it follows that when $w^2 = \lambda\gamma$, $|L_i(jw)|^2$ achieves its maximum value. Similarly, the maximum of $|L_i(jw)|$ is $1/(\lambda+\gamma)$. Combined with Parseval's theorem, it follows that the values of λ and γ are inversely proportional to the value of $L_i(s)$. The relationship between $\boldsymbol{\xi}_i(t)$ and I_i can be obtained from equation (25) as

$$\boldsymbol{\xi}_i(t) = \int I_i(t-\tau)L(\tau)\mathrm{d}\tau. \tag{28}$$

By defining the upper bound of $I_i(t)$ as M_I, the maximum value of $|\xi_i(t)|$ is expressed as

$$\begin{aligned} \max|\boldsymbol{\xi}_i(t)| &= \max\Big|\int I_i(t-\tau)L_i(\tau)\mathrm{d}\tau\Big| \\ &\le M\big|\max\int |L_i(\tau)|\mathrm{d}\tau\big|. \end{aligned} \tag{29}$$

Obviously, $L_i(s)$ imposes restrictions on $I_i(t)$. So as long as λ and γ are large enough, $|\boldsymbol{\xi}(t)|$ will be small enough. Specifically, the position of the RCM point $\mathbf{r}_c$ and the angular acceleration of joint $\ddot{\boldsymbol{\theta}}$ converge to the optimal values, which means the successful execution of the task. The proof is thus completed.

IV. Experiments and Comparisons

In this section, the proposed ARC^{2}M scheme (7) is proven to be validated through simulations conducted on MATLAB and CoppeliaSim platforms, and experiments conducted on a real robot manipulator. Additionally, comparisons are carried out with other RCM control schemes to illustrate the advantages of the proposed ARC^{2}M scheme (7).

A. Simulations and Experiment

Simulations are based on the Franka Emika Panda robot, augmented with a 0.3 m slender rod as a replaceable end-effector to simulate actual usage scenarios, and the axis of the rod coincides with the axis of rotation of the last joint of the robot. Details of the parameter settings are $k_0 = 0.5, m_1 = m_2 = 0.1, \lambda = 1e5, \gamma = 1e2, Q = 0.01, R = 0.25, \boldsymbol{\theta}^+ = -\boldsymbol{\theta}^- = 2.5$ rad, $\dot{\boldsymbol{\theta}}^+ = -\dot{\boldsymbol{\theta}}^- = 0.45$ rad/s, $\ddot{\boldsymbol{\theta}}^+ = -\ddot{\boldsymbol{\theta}}^- = 1$ rad/s^2, $\eta(t) = \mu(t) = 100 \times [50 + i \times t$ for i in range(1, 15)], $\boldsymbol{\theta}_0 = [0;\ -0.25\pi;\ 0;\ -0.75\pi;\ 0;\ 0.5\pi;\ 0.25\pi]$ rad.

The simulation is designed to carry out a path-following task through a hole in the surface of the space with unknown parameters of the end-effector. Results of the simulation are presented in Fig. 2. Figure 2(a) shows the trajectory of the

TABLE I

COMPARISONS AMONG DIFFERENT RCM CONTROL SCHEMES

Scheme	Control level	Cyclic motion	Noise tolerance	Structural information
Proposed	A[1]	✓	✓	Partially unknown
[5]	V[2]	✗	✗	Known
[15]	V[2]	✗	✗	Known
[16]	V[2]	✗	✓	Known
[17]	V[2]	✗	✗	Known
[18]	P[3]	✗	✗	Known
[19]	P[3]	✗	✓	Known
[20]	V[2]	✗	✗	Known

[1] A: Acceleration-level control
[2] V: Velocity-level control
[3] P: Position-level control

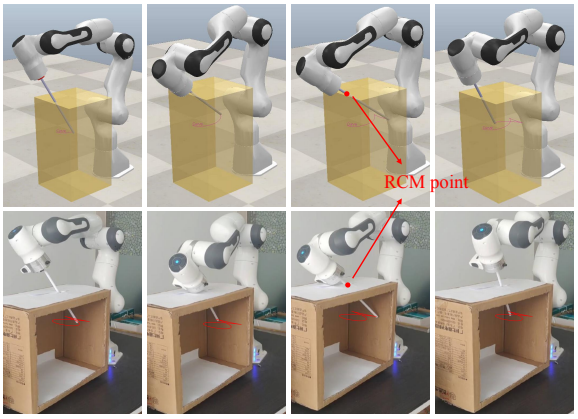

Fig. 3. Snapshots of Franka Emika Panda robot driven by the proposed ARC^{2}M scheme (7) performing a fish-shaped trajectory-tracking task in a limited space on the CoppeliaSim platform and a real robot manipulator. The video is available at https://youtu.be/0jOUTjJBFzI.

robot during the task. Desired and actual trajectories appear to be highly compatible, as shown in Fig. 2(b). Besides, the joint angle and velocity are shown in Figs. 2(c) and (d), which can be seen to be within limits. The curve of the change in position of the RCM points is shown in Fig. 2(e). The curve of parameters of the end-effector is shown in Fig. 2(f). The parameter values stabilize after 0.03 s and are very close to the true values.

The physical experiment is also carried out with the same parameters as the simulation experiment, as shown in Fig. 3. It can be seen that the tracking task is well executed within the boundary of the circular hole.

B. Comparisons

In Table II, the proposed ARC^{2}M scheme (7) is compared with other RCM control schemes to demonstrate innovations. The use of the parameter calculation method and the neural dynamics-based method enables the robot to perform RCM operations accurately under the influence of noise.

In addition to the qualitative analyses, a comparative experiment on scheme [15] and the proposed ARC^{2}M scheme (7) is carried out, where both of them perform the same trajectory tracking experiments. The experimental results

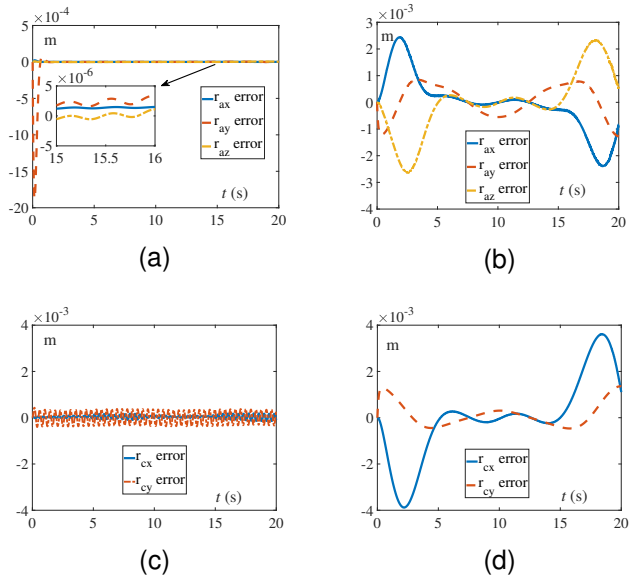

Fig. 4. End-effector position error and RCM point position error during the task guided by the proposed ARC^{2}M scheme (7) and the scheme [15]. (a) End-effector position error of the proposed ARC^{2}M scheme (7). (b) End-effector position error of the scheme [15]. (c) RCM point position error of the proposed ARC^{2}M scheme (7). (d) RCM point position error of the scheme [15].

shown in Fig. 4 indicate that the proposed ARC^{2}M scheme (7) achieves the higher precision control of the robot through acceleration-level control compared with the scheme [15]. Figures 4(a) and (b) show the position error of the end-effector, and it can be seen that the error of the proposed ARC^{2}M scheme (7) is stable at the order of 10^{-6} m and that of the scheme [15] at the order of 10^{-3} m. Figures 4(c) and (d) show the position deviation of the RCM point during the task execution, respectively. It can be seen that the position deviation of the RCM point in the proposed ARC^{2}M scheme (7) is less than that in scheme [15], which demonstrates the superiority of the proposed ARC^{2}M scheme (7).

V. CONCLUSIONS

This paper has proposed an ARC^{2}M scheme (7) to address the challenge of unknown parameters of the robot with a replaceable end-effector. The scheme includes cyclic motion control, acceleration-level RCM control, parameter calculation, and a noise reduction module, which enables acceleration-level RCM control in noisy environments for a robot with a replaceable end-effector. Physical restrictions on the joints have been considered to make the ARC^{2}M scheme (7) easier to apply to a physical robot. In summary, this paper has provided an efficient scheme for the problem of executing RCM control for replaceable end-effector robots, enabling their accurate control in complex scenarios. As a potential future research direction, it is possible to combine endoscopy and target detection technology to achieve real-time calculation of the surgical path and trajectory tracking of the robot end-effector.

REFERENCES

[1] A. B. Aldanmaz, O. Ayit, G. Kiper, and M. I. C. Dede, "Gravity compensation of a 2R1T mechanism with remote center of motion for minimally invasive transnasal surgery applications," *Robotica*, vol. 41, no. 3, pp. 807–820, Mar. 2023.

[2] H. Gao, X. Xiao, L. Qiu, M. Q.-H. Meng, N. K. K. King, and H. Ren, "Remote-center-of-motion recommendation toward brain needle intervention using deep reinforcement learning," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Xi'an, China, 2021, pp. 8295–8301.

[3] J. Xia *et al.*, "Microscope-guided autonomous clear corneal incision," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Paris, France, 2020, pp. 3867–3873.

[4] C.-Y. Maeng, J. Yoon, D.-Y. Kim, J. Lee, and Y.-J. Kim, "Development of an inherently safe nasopharyngeal swab sampling robot using a force restriction mechanism," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 11150–11157, Oct. 2022.

[5] H. Su, Y. Hu, H. R. Karimi, A. Knoll, G. Ferrigno, and E. De Momi, "Improved recurrent neural network-based manipulator control with remote center of motion constraints: Experimental results," *Neural Networks*, vol. 131, pp. 291–299, Nov. 2020.

[6] H. Su, Y. Schmirander, S. E. Valderrama-Hincapié, W. Qi, S. E. Ovur, and J. Sandoval, "Neural-learning-enhanced cartesian admittance control of robot with moving RCM constraints," *Robotica*, vol. 41, no. 4, pp. 1231–1243, Dec. 2023.

[7] M. Liu, K. Liu, P. Zhu, G. Zhang, X. Ma, and M. Shang, "Data-driven remote center of cyclic motion (RC^2M) control for redundant robots with rod-shaped end-effector," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2024.3353930.

[8] B. Li, B. Lu, Y. Lu, Q. Dou, and Y.-H. Liu, "Data-driven holistic framework for automated laparoscope optimal view control with learning-based depth perception," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Xi'an, China, 2021, pp. 12366–12372.

[9] Z. Xie, L. Jin, X. Luo, S. Li, and X. Xiao, "A data-driven cyclic-motion generation scheme for kinematic control of redundant manipulators," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 1, pp. 53–63, Jan. 2021.

[10] 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.*, to be published, doi: 10.1109/TIV.2024.3358229.

[11] K.-C. Siu, I. H. Suh, M. Mukherjee, D. Oleynikov, and N. Stergiou, "The impact of environmental noise on robot-assisted laparoscopic surgical performance," *Surgery*, vol. 147, no. 1, pp. 107–113, Jan. 2010.

[12] L. Wei, L. Jin, and X. Luo, "Noise-suppressing neural dynamics for time-dependent constrained nonlinear optimization with applications," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 52, no. 10, pp. 6139–6150, Oct. 2022.

[13] Y. Liufu, L. Jin, M. Shang, X. Wang, and F.-Y. Wang, "Acp-incorporated perturbation-resistant neural dynamics controller for autonomous vehicles," *IEEE Trans. Intell. Veh.*, to be published, doi: 10.1109/TIV.2023.3348632.

[14] H. Madhusudanan *et al.*, "Automated eye-in-hand robot-3D scanner calibration for low stitching errors," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Paris, France, 2020, pp. 8906–8912.

[15] Z. Li and S. Li, "Kinematic control of manipulator with remote center of motion constraints synthesised by a simplified recurrent neural network," *Neural Process. Lett.*, vol. 54, no. 2, pp. 1035–1054, Apr. 2022.

[16] Z. Cui, W. Li, X. Zhang, P. W. Y. Chiu, and Z. Li, "Accelerated dual neural network controller for visual servoing of flexible endoscopic robot with tracking error, joint motion, and RCM constraints," *IEEE Trans. Ind. Electron.*, vol. 69, no. 9, pp. 9246–9257, Sep. 2021.

[17] W. Li, L. Han, X. Xiao, B. Liao, and C. Peng, "A gradient-based neural network accelerated for vision-based control of an RCM-constrained surgical endoscope robot," *Neural. Comput. Appl.*, vol. 34, pp. 1329–1343, Sep. 2021.

[18] A. H. Khan, S. Li, and X. Cao, "Tracking control of redundant manipulator under active remote center-of-motion constraints: An RNN-based metaheuristic approach," *Sci. China Inf. Sci.*, vol. 64, pp. 1869–1919, Mar. 2021.

[19] H. Su *et al.*, "Internet of things (IoT)-based collaborative control of a redundant manipulator for teleoperated minimally invasive surgeries," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, Paris, France, 2020, pp. 9737–9742.

[20] T. Kastritsi and Z. Doulgeri, "A control method for time-variant RCM constraint in hands-on ramis procedures," in *Mediterr. Conf. Control Automation (MED)*, Puglia, Italy, 2021, pp. 729–734.