Obstacle Avoidance and Trajectory Tracking of Redundant Manipulators with Unknown Physical Parameters Under Multiple Constraints

Lei Jia*, **Hui Deng**
College of Computer Science (College of Software), College of Artificial Intelligence, Inner Mongolia University, Hohhot 010021, China
ljia@imu.edu.cn, 32409297@mail.imu.edu.cn

Abstract

Redundant manipulators are broadly used in safety tasks that require precise execution. However, some practical factors, such as assembly defects and mechanical wear, inevitably introduce uncertainties in their physical parameters. Due to insufficient excitation, existing methods fail to achieve high-accuracy physical parameter identification, which seriously affects the accurate operation of multiple safety tasks. Considering that quadratic programming (QP) can integrate desired behaviors with constraints into a unified optimization framework and can be solved online in real time, this paper formulates the collision-free trajectory tracking of redundant manipulators with uncertain structure as a QP problem, which describes the trajectory tracking, obstacle avoidance and joint motion limits simultaneously. To tackle the above issues, we first develop an online physical parameters identification method called regularization and zeroing neural dynamics (RZND) by using the state information of the manipulator, which realizes highly accurate physical parameters identification and real-time Jacobian matrix updates. On the basis of this, a multi-task quadruple projection neural network (MT-QPNN) solver is proposed by applying projection operators to handle the joint constraints, and further the trajectory tracking and obstacle avoidance tasks of the redundant manipulator are well achieved. Relevant experiments verify that the presented RZND method and MT-QPNN solver can accurately identify physical parameters and enable collision-free tracking with superior performance.

1 Introduction

Redundant manipulators have been applied in various fields such as industry and physical therapy due to their high flexibility [Yan and Di, 2022]. Therefore, they have to execute precise motion tasks while interacting with humans and surrounding objects [Roveda *et al.*, 2023; Gombolay, 2024]. In these scenarios, obstacles are inevitable and may cause dangerous collisions, task failure, or even damage to humans or robots [Okumura and Défago, 2023]. Accordingly, practical manipulator motion control not only needs to accurately complete the tasks, but also should maintain safety from the obstacles in the scene [Koutras *et al.*, 2023; Li *et al.*, 2023].

Learning-based obstacle avoidance methods often depend on offline data collection and specific scene training [Shu *et al.*, 2024], which can increase deployment and adaptation cost when the environment changes. However, in actual scenarios, manipulators are expected to make real-time decisions when encountering obstacles. Therefore, a more practical strategy is to continuously monitor the distance between the manipulators and the obstacles, and then adjust the motion of the manipulators in real time, so that avoidance actions are activated when the manipulators approach obstacles [Song and Lin, 2024]. Regrettably, existing methods [Chen *et al.*, 2024; Zhang *et al.*, 2020] rely on accurate physical structure parameters, the motion adjustment may fail to produce the expected distance increment, thereby weakening the obstacle avoidance effect and disrupting the trajectory tracking behavior. Moreover, in many obstacle-avoidance designs [Huang *et al.*, 2025; Li *et al.*, 2025], joint acceleration limits of manipulators are often neglected, which may result in overly aggressive transients and reduce the stability of the movement. Optimization problems are well suited for real-time motion control because they can be solved online within each control cycle [Sun *et al.*, 2025]. Accordingly, this paper formulates obstacle avoidance and joint state limits in a unified constrained optimization problem, so that the manipulators can generate feasible motions.

In practical applications, the uncertainties in operating environment are often considered, yet the physical structure of the manipulators is also subject to unknown. This uncertainty is often caused by assembly tolerances, long-term wear, payload changes, and maintenance or reconfiguration, which leads to deviations in physically calibrated parameters of the factory and the related unreliable [Yu *et al.*, 2025; Zhang *et al.*, 2018]. To address this issue, many optimal learning algorithms combined with control strategies have been developed. For example, [Liu *et al.*, 2022] proposed a simultaneous learning and control scheme for redundant manipulators under physical constraints, addressing the tra-

*Corresponding author: Lei Jia.
Main Code: https://github.com/LeiJia223/Code-IJCAI2026

jectory tracking problem for manipulators under model-free settings while enforcing joint velocity and acceleration constraints. [Xie *et al.*, 2025b] proposed a data-driven obstacle avoidance scheme for redundant robots with unknown structure, which expands the application scenarios under structure uncertainty. However, such methods cannot accurately estimate the structural parameters of manipulators when avoiding obstacles and performing other tasks. Thus, it is necessary to further develop high-precision physical parameter identification methods to guarantee the accurate task execution.

In this paper, we transform the collision-free trajectory tracking for redundant manipulators with unknown physical parameters under joint velocity and acceleration limits into a quadratic programming (QP) problem with multiple constraints. Then, the regularized zeroing neural dynamics (RZND) method to precisely identify the physical parameters and the multi-task quadruple projection neural network (MT-QPNN) solver to deal with the corresponding QP problem and finally realize the control of manipulators are proposed. The whole research framework of this paper is shown in Figure 1. The main contributions are summarized as follows.

- We formulate the collision-free trajectory tracking for redundant manipulators with uncertain structure as a real-time QP problem that enforces tracking and obstacle-avoidance tasks, joint velocity and acceleration limits simultaneously.
- We develop a RZND physical parameter identification method based on regularization and zeroing neural dynamics. This presented method identifies the physical parameters with high precision and updates the Jacobian matrix online for the control of manipulators.
- We propose an MT-QPNN solver built on projection operators, which effectively realizes trajectory tracking and obstacle avoidance of manipulators under multiple joint constraints by leveraging the updated Jacobian matrix.

2 Methodologies and Problem

In this section, kinematic equations of redundant manipulators are first provided. Further, RZND identification method for unknown physical parameters and the obstacle avoidance method are proposed in detail, and then the corresponding QP problem is eventually formulated.

2.1 Forward Kinematics

It is well known that forward kinematics defines the mapping relationship between the joint space and the task space of manipulators. The forward kinematics of a redundant manipulator is commonly expressed as follows [Chen *et al.*, 2024]:

$$\boldsymbol{r}_e(t) = f(\boldsymbol{q}(t)), \tag{1}$$

where $f(\cdot)$ is a smooth nonlinear mapping that describes the forward kinematic relationship, $\boldsymbol{q} = [q_1, q_2, ..., q_n]^{\mathrm{T}} \in \mathbb{R}^n$ represents the joint angles and $\boldsymbol{r}_e(t) \in \mathbb{R}^m$ denotes the end-effector position in the task space, where $n > m$. Then, differentiating (1) with respect to time t yields an expression regarding the joint velocity level:

$$\dot{\boldsymbol{r}}_e(t) = J(\boldsymbol{q}(t))\dot{\boldsymbol{q}}(t), \tag{2}$$

where $J(\boldsymbol{q}(t)) = \partial \boldsymbol{f}(\boldsymbol{q})/\partial \boldsymbol{q} \in \mathbb{R}^{m\times n}$ is the Jacobian matrix, which is used to describe the mapping relationship between the joint velocity and the end-effector velocity $\dot{\boldsymbol{r}}_e(t)$.

Based on (1), the error $\boldsymbol{\varepsilon}(t) \in \mathbb{R}^m$ between the actual trajectory of the end-effector and the desired trajectory $\boldsymbol{r}_d(t) \in \mathbb{R}^m$ is defined as follows

$$\boldsymbol{\varepsilon}(t) = \boldsymbol{r}_e(t) - \boldsymbol{r}_d(t). \tag{3}$$

By differentiating (3) with respect to time t and combining the velocity-level kinematic relation in (2), the error $\dot{\boldsymbol{\varepsilon}}(t)$ between the actual velocity of the end-effector and the corresponding desired velocity can be written as

$$\dot{\boldsymbol{\varepsilon}}(t) = J(\boldsymbol{q}(t))\dot{\boldsymbol{q}}(t) - \dot{\boldsymbol{r}}_d(t). \tag{4}$$

2.2 RZND Identification Method

In some uncertain practical scenarios, the physical parameters of the redundant manipulator are often difficult to obtain, with the corresponding Jacobian matrix unknown, which seriously affects the control and movement of the manipulator. To solve this problem, an RZND method is proposed in this paper that can accurately identify the unknown physical parameters of the manipulator based on the regularization and zeroing neural dynamics.

Assuming that the Jacobian matrix to be obtained is $\hat{J}_e(t) \in \mathbb{R}^{m\times n}$, and then based on the end-effector velocity error (4), it can be obtained that:

$$\boldsymbol{e}(t) = \dot{\boldsymbol{r}}_e^*(t) - \hat{J}_e(t)\dot{\boldsymbol{q}}(t), \tag{5}$$

where $\dot{\boldsymbol{r}}_e^*(t) \in \mathbb{R}^m$ represents the precise end-effector velocity obtained by sensors of the manipulator. In order to better identify the physical parameters, we first represent the end-effector position in a linear regression form:

$$\boldsymbol{r}_e(t) = R(t)\hat{\boldsymbol{p}}(t), \tag{6}$$

where $R(t) \in \mathbb{R}^{m\times k}$ is the regressor matrix determined by the joint angle $\boldsymbol{q}(t)$, k is the number of physical parameters for the manipulator and $\hat{\boldsymbol{p}}(t) \in \mathbb{R}^k$ is the vector of unknown physical parameters. Then, the Jacobian matrix is decomposed by vectorization as:

$$\mathrm{vec}(\hat{J}_e(t)) = S(t)\hat{\boldsymbol{p}}(t), \tag{7}$$

where $S(t) \in \mathbb{R}^{(m\times n)\times k}$ is the vectorized regression matrix that depends on the joints of the manipulator.

In addition, due to the vectorization operation of Jacobian matrix, the form of the joint velocity $\dot{\boldsymbol{q}}(t)$ in (5) also needs to be rewritten:

$$Q(t) = I \otimes \dot{\boldsymbol{q}}^{\mathrm{T}}(t), \tag{8}$$

where $Q(t) \in \mathbb{R}^{m\times(m\times n)}$ represents the mapping matrix constructed based on Kronecker product, which is used to expand the end-effector velocity equation (5) into a vectorized form, $I \in \mathbb{R}^{m\times m}$ is the identity matrix.

Based on (5) and (6), it follows that:

$$\boldsymbol{\phi}_p(t) = \beta\big(\boldsymbol{r}_e^*(t) - R(t)\hat{\boldsymbol{p}}(t)\big), \tag{9a}$$

$$\boldsymbol{\phi}_v(t) = \alpha\big(\dot{\boldsymbol{r}}_e^*(t) - Q(t)S(t)\hat{\boldsymbol{p}}(t)\big), \tag{9b}$$

where $\boldsymbol{\phi}_p(t), \boldsymbol{\phi}_v(t) \in \mathbb{R}^m$ are two errors that combine the position and velocity of end-effector, $\alpha > 0$, $\beta > 0$. $\boldsymbol{r}_e^*(t) \in$

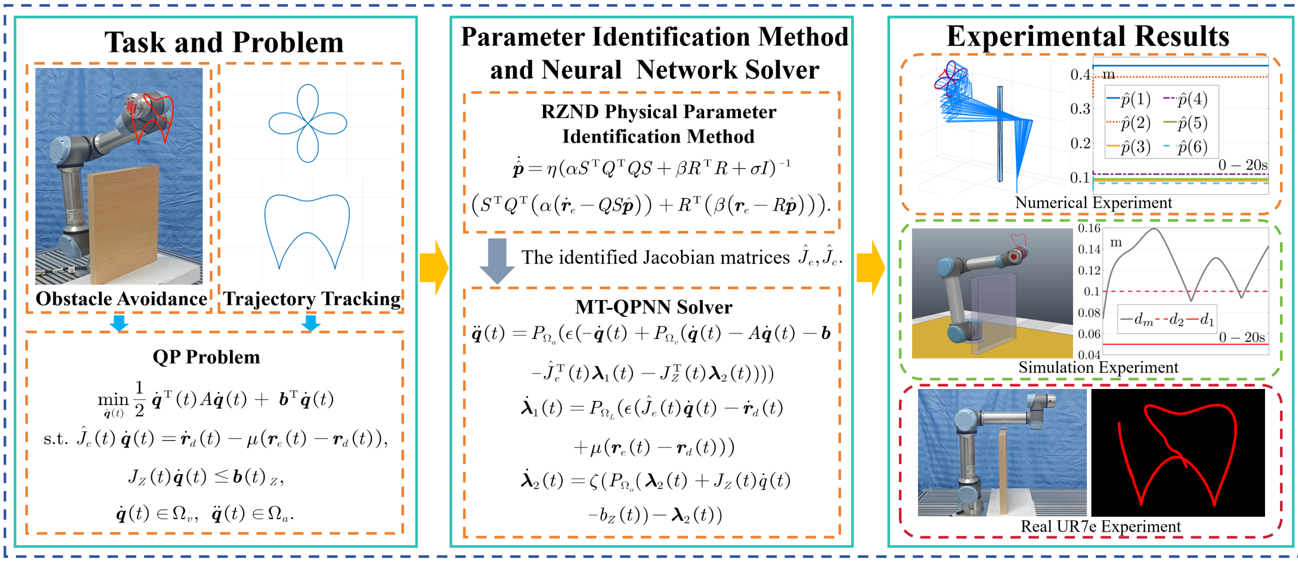

Figure 1: Research framework of this paper.

$\mathbb{R}^m$ is the end-effector position obtained from the sensors. Next, according to zeroing neural dynamics [He *et al.*, 2025], the evolution formula of error $\boldsymbol{\phi}(t)$ is written as:

$$\dot{\boldsymbol{\phi}}(t) = -\gamma \boldsymbol{\phi}(t), \tag{10}$$

where $\gamma > 0$. According to (10), taking the derivatives of (9a) and (9b) with respect to $\hat{\boldsymbol{p}}$ respectively, it has

$$\dot{\hat{\boldsymbol{p}}}(t) = \gamma R^{\mathrm{T}}(t)(\beta(\boldsymbol{r}_e^*(t) - R(t)\hat{\boldsymbol{p}}(t))), \tag{11a}$$

$$\dot{\hat{\boldsymbol{p}}}(t) = \gamma S^{\mathrm{T}}(t)Q^{\mathrm{T}}(t)(\alpha(\dot{\boldsymbol{r}}_e^*(t) - Q(t)S(t)\hat{\boldsymbol{p}}(t))). \tag{11b}$$

Combining (11a) and (11b), it follows

$$\dot{\hat{\boldsymbol{p}}} = \frac{\gamma}{2}(S^{\mathrm{T}}Q^{\mathrm{T}}(\alpha(\dot{\boldsymbol{r}}_e^* - QS\hat{\boldsymbol{p}})) + R^{\mathrm{T}}(\beta(\boldsymbol{r}_e^* - R\hat{\boldsymbol{p}}))). \tag{12}$$

Finally, in order to prevent the matrix from becoming non-reversible due to weak excitation, a Tikhonov regularization term [Liu and Cao, 2025] is introduced. Thus, result (12) can ultimately be written as:

$$\begin{aligned}\dot{\hat{\boldsymbol{p}}} =& \eta(\alpha S^{\mathrm{T}}Q^{\mathrm{T}}QS + \beta R^{\mathrm{T}}R + \sigma I)^{-1}(S^{\mathrm{T}}Q^{\mathrm{T}}(\alpha(\dot{\boldsymbol{r}}_e^* \\ &- QS\hat{\boldsymbol{p}})) + R^{\mathrm{T}}(\beta(\boldsymbol{r}_e^* - R\hat{\boldsymbol{p}}))),\end{aligned} \tag{13}$$

where $\eta = \gamma/2$; $\sigma > 0$ is a small positive constant; $I \in \mathbb{R}^{k\times k}$ is the identity matrix.

2.3 Obstacle Avoidance Method

To prevent potential collisions between the redundant manipulator and surrounding obstacles, an obstacle avoidance constraint is introduced at the speed level. In general, each obstacle can be represented by one or several points that are closest to the manipulator, and the parts of the manipulator that are prone to collision are described by a set of key points. By monitoring the relative motion between each critical point and the corresponding obstacle point, an inequality constraint can be formulated to guarantee that the manipulator moves away from the obstacle, which is expressed as:

$$J_Z(t)\dot{\boldsymbol{q}}(t) \leq \boldsymbol{0}, \tag{14}$$

where $J_Z(t) = -\operatorname{sgn}(\vec{\boldsymbol{r}}_C) \diamond \hat{J}_c(t) \in \mathbb{R}^{m\times n}$ and $\operatorname{sgn}(\cdot)$ denotes a scalar signum function. Besides, the symbol $\diamond$ calculates that $a \diamond B = [a_1B_1, a_2B_2, ..., a_\ell B_\ell]^{\mathrm{T}}$, where a is a column vector and B is a matrix. The definition of $\vec{\boldsymbol{r}}_C$ is

$$\vec{\boldsymbol{r}}_C = [x_C - x_O,\ y_C - y_O,\ z_C - z_O]^{\mathrm{T}}, \tag{15}$$

where (x_O, y_O, z_O) and (x_C, y_C, z_C) respectively represent the three-dimensional coordinates of the obstacle point O and the key point C. Specifically, when the manipulator's structure is unknown, the key point C is not derived from the analytical kinematics calculation, but is determined through geometric detection based on the minimum distance between the manipulator and the obstacle. In addition, the corresponding Jacobian matrix $\hat{J}_c$ is calculated based on the identified physical parameters:

$$\hat{J}_c(t) = f_J(\hat{\boldsymbol{p}}(t), \boldsymbol{q}(t)), \tag{16}$$

where $\hat{\boldsymbol{p}}(t)$ denotes the identified parameters, and $f_J(\cdot)$ represents the mapping from the estimated physical parameters and joint angles to the local Jacobian matrix of the critical point C.

To avoid the discontinuity of $\dot{\boldsymbol{q}}(t)$ in (14), further smoothing processing of the constraints is designed as follows:

$$J_Z(t)\dot{\boldsymbol{q}}(t) \leq \boldsymbol{b}_Z(t), \tag{17}$$

where $\boldsymbol{b}_Z(t) = s(d_m)\max\{J_Z(t)\dot{\boldsymbol{q}}(t)|_{d_m=d_1}, \boldsymbol{0}\}$, and the definition of $s(d_m)$ is

$$s(d_m) = \begin{cases} 1, & \text{if } d_m \geq d_2, \\ \sin^2\left(\dfrac{\pi(d_m - d_1)}{2(d_2 - d_1)}\right), & \text{if } d_1 < d_m < d_2, \\ 0, & \text{if } d_m \leq d_1, \end{cases} \tag{18}$$

where d_m denotes the distance between the critical point C and the obstacle point O. The parameter d_1 denotes the safe distance, which depends on the link thickness of the manipulator. When the link of the manipulator collides with an obstacle, at this point d_m is less than d_1. The parameter d_2 represents the effective distance of the obstacle avoidance. As the value of d_2 increases, the movement of the redundant manipulator becomes safer. On the other hand, a larger operating space will reduce the flexibility of the manipulator and increase the computational cost. Therefore, the parameter d_2 should be set to an appropriate value based on safety requirements and hardware performance.

2.4 QP Problem with Multiple Constraints

In this subsection, the trajectory tracking and obstacle avoidance tasks of redundant manipulators with uncertain structure are formulated as a QP problem in the joint velocity space. It imposes an equality constraint on the end-effector velocity and incorporates several inequality constraints, including joint velocity limits, joint acceleration limits, and obstacle avoidance conditions. By solving this QP problem at each control step, the computed joint velocity can simultaneously achieve precise closed-loop trajectory tracking, obstacle avoidance safety, and strict compliance with the physical limitations of the manipulator. The details are as follows:

$$\min_{\dot{\boldsymbol{q}}(t)} \ \frac{1}{2}\dot{\boldsymbol{q}}^{\mathrm{T}}(t)A\dot{\boldsymbol{q}}(t)+\ \boldsymbol{b}^{\mathrm{T}}\dot{\boldsymbol{q}}(t), \tag{19a}$$

$$\text{s.t.}\ \ \hat{J}_e(t)\,\dot{\boldsymbol{q}}(t)=\dot{\boldsymbol{r}}_d(t)-\mu(\boldsymbol{r}_e(t)-\boldsymbol{r}_d(t)), \tag{19b}$$

$$J_Z(t)\dot{\boldsymbol{q}}(t)\le \boldsymbol{b}_Z(t), \tag{19c}$$

$$\dot{\boldsymbol{q}}(t)\in\Omega_v, \tag{19d}$$

$$\ddot{\boldsymbol{q}}(t)\in\Omega_a, \tag{19e}$$

where $A=I\in\mathbb{R}^{n\times n}$ and $\boldsymbol{b}\in\mathbb{R}^n$. The joint velocity constraints are collected in the set $\Omega_v=\{\dot{\boldsymbol{q}}(t)\in\mathbb{R}^n \mid \dot{\boldsymbol{q}}^-\le\dot{\boldsymbol{q}}(t)\le\dot{\boldsymbol{q}}^+\}$, while the set $\Omega_a=\{\ddot{\boldsymbol{q}}(t)\in\mathbb{R}^n \mid \ddot{\boldsymbol{q}}^-\le\ddot{\boldsymbol{q}}(t)\le\ddot{\boldsymbol{q}}^+\}$ specifies the admissible joint acceleration. Together, they reflect the physical constraints of the redundant manipulator. The parameter $\mu>0$ is used to adjust the position error. Clearly, $\boldsymbol{0}\in\Omega_v,\Omega_a$.

3 MT-QPNN Solver Design

For QP problem (19), Lagrange multipliers are introduced and the Karush–Kuhn–Tucker (KKT) optimality conditions [Boyd and Vandenberghe, 2004] are applied, thereby yielding the results that satisfy optimality, feasibility, and boundary constraints. The Lagrangian function associated with QP problem (19) is constructed in the following form:

$$\begin{aligned}\mathcal{L}(\dot{\boldsymbol{q}}(t),\boldsymbol{\lambda}_1(t),\boldsymbol{\lambda}_2(t),t)&=\frac{1}{2}\dot{\boldsymbol{q}}^{\mathrm{T}}(t)A\dot{\boldsymbol{q}}(t)+\boldsymbol{b}^{\mathrm{T}}\dot{\boldsymbol{q}}(t)\\&+\boldsymbol{\lambda}_1^{\mathrm{T}}(t)\big(\hat{J}_e(t)\dot{\boldsymbol{q}}(t)-\dot{\boldsymbol{r}}_d(t)+\mu(\boldsymbol{r}_e(t)-\boldsymbol{r}_d(t))\big)\\&+\boldsymbol{\lambda}_2^{\mathrm{T}}(t)(J_Z(t)\dot{\boldsymbol{q}}(t)-\boldsymbol{b}_Z(t)).\end{aligned} \tag{20}$$

Based on the KKT conditions, the steady state of QP problem (19) can be expressed as:

$$\begin{cases}\dot{\boldsymbol{q}}(t)=P_{\Omega_v}\Big(\dot{\boldsymbol{q}}(t)-\dfrac{\partial\mathcal{L}}{\partial\dot{\boldsymbol{q}}}\Big),\ \dfrac{\partial\mathcal{L}}{\partial\boldsymbol{\lambda}_1}=\boldsymbol{0},\\ J_Z(t)\dot{\boldsymbol{q}}(t)-\boldsymbol{b}_Z(t)\le\boldsymbol{0},\\ (\boldsymbol{\lambda}_2)_i(t)\ge 0,\\ (\boldsymbol{\lambda}_2)_i(t)\cdot(J_Z(t)\dot{\boldsymbol{q}}(t)-\boldsymbol{b}_Z(t))_i=0,\end{cases} \tag{21}$$

where $i=1,2,...,m$, $P_{\Omega_v}(\boldsymbol{y})=\arg\min_{\boldsymbol{z}\in\Omega_v}\|\boldsymbol{y}-\boldsymbol{z}\|_2^2$ is a projection operator to handle the joint velocity limits. Then, equation (21) is further rewritten as follows:

$$\begin{aligned}\dot{\boldsymbol{q}}(t)=P_{\Omega_v}(&\dot{\boldsymbol{q}}(t)-A\dot{\boldsymbol{q}}(t)-\boldsymbol{b}\\&-\hat{J}_e^{\mathrm{T}}(t)\boldsymbol{\lambda}_1(t)-J_Z^{\mathrm{T}}(t)\boldsymbol{\lambda}_2(t)),\end{aligned} \tag{22a}$$

$$\boldsymbol{0}=\hat{J}_e(t)\dot{\boldsymbol{q}}(t)-\dot{\boldsymbol{r}}_d(t)+\mu(\boldsymbol{r}_e(t)-\boldsymbol{r}_d(t)), \tag{22b}$$

$$\boldsymbol{\lambda}_2(t)=P_{\Omega_o}(\boldsymbol{\lambda}_2(t)+J_Z(t)\dot{\boldsymbol{q}}(t)-\boldsymbol{b}_Z(t)), \tag{22c}$$

where Ω_o is a closed convex set that defines the allowable obstacle avoidance multiplier and it is nonnegative, i.e., $\Omega_o=\{\boldsymbol{\lambda}_2(t)\in\mathbb{R}^m \mid \boldsymbol{\lambda}_2(t)\ge\boldsymbol{0}\}$.

Accordingly, inspired by the multiple projection method in [Zhang, 2022], the MT-QPNN solver for addressing QP problem (19) is proposed:

$$\begin{aligned}\ddot{\boldsymbol{q}}(t)=&P_{\Omega_a}\big(\epsilon\big(-\dot{\boldsymbol{q}}(t)+P_{\Omega_v}\big(\dot{\boldsymbol{q}}(t)-A\dot{\boldsymbol{q}}(t)-\boldsymbol{b}\\&-\hat{J}_e^{\mathrm{T}}(t)\boldsymbol{\lambda}_1(t)-J_Z^{\mathrm{T}}(t)\boldsymbol{\lambda}_2(t)\big)\big)\big),\end{aligned} \tag{23a}$$

$$\dot{\boldsymbol{\lambda}}_1(t)=P_{\Omega_L}\big(\epsilon\big(\hat{J}_e(t)\dot{\boldsymbol{q}}(t)-\dot{\boldsymbol{r}}_d(t)+\mu(\boldsymbol{r}_e(t)-\boldsymbol{r}_d(t))\big)\big), \tag{23b}$$

$$\dot{\boldsymbol{\lambda}}_2(t)=\zeta(P_{\Omega_o}(\boldsymbol{\lambda}_2(t)+J_Z(t)\dot{\boldsymbol{q}}(t)-\boldsymbol{b}_Z(t))-\boldsymbol{\lambda}_2(t)), \tag{23c}$$

where $\epsilon>0$ and $\zeta>0$ represent adjustable gain parameters; $\Omega_L=\{\dot{\boldsymbol{\lambda}}_1(t)\in\mathbb{R}^m \mid \dot{\boldsymbol{\lambda}}_1^-\le\dot{\boldsymbol{\lambda}}_1(t)\le\dot{\boldsymbol{\lambda}}_1^+\}$ with $\dot{\boldsymbol{\lambda}}_1^-$ and $\dot{\boldsymbol{\lambda}}_1^+$ specifying the desired lower and upper limits of $\dot{\boldsymbol{\lambda}}_1$. Four sets Ω_v, Ω_a, Ω_L, Ω_o respectively bound joint velocity, joint acceleration, the derivative of the Lagrange multiplier and the obstacle avoidance multiplier.

Finally, with the combined effect of four projection operators $P_{\Omega_a}(\cdot)$, $P_{\Omega_v}(\cdot)$, $P_{\Omega_L}(\cdot)$, $P_{\Omega_o}(\cdot)$ and the RZND physical parameter identification method, the proposed MT-QPNN solver (23) not only enables the redundant manipulator to meet various physical limitations, but also effectively solves the obstacle avoidance and trajectory tracking problems in the case of unknown structure of the manipulator.

4 Experiments

This section is divided into three subsections. The first part describes the numerical experiment of the UR5 manipulator on MATLAB, and later the proposed method is compared with methods in [Xie *et al.*, 2025a; Zhang and Zhang, 2025] and [Xie *et al.*, 2025b]. The second part is the simulation experiment of the UR5 manipulator using CoppeliaSim. The last part is the real UR7e physical experiment. Specifically, all experiments are conducted on UR-series manipulators, for which the end-effector task dimension is $m=3$ and the number of joints is $n=6$.

4.1 Numerical Experiments

This subsection presents the numerical experiments on the MATLAB R2022a software, with the execution time $T=20\,\mathrm{s}$. The experiment requires the UR5 manipulator to avoid the vertically placed triangular prism obstacle in real time, complete the trajectory tracking of the four-leaf clover trajectory, and accurately identify the parameters of each physical link of this redundant manipulator. The upper and lower

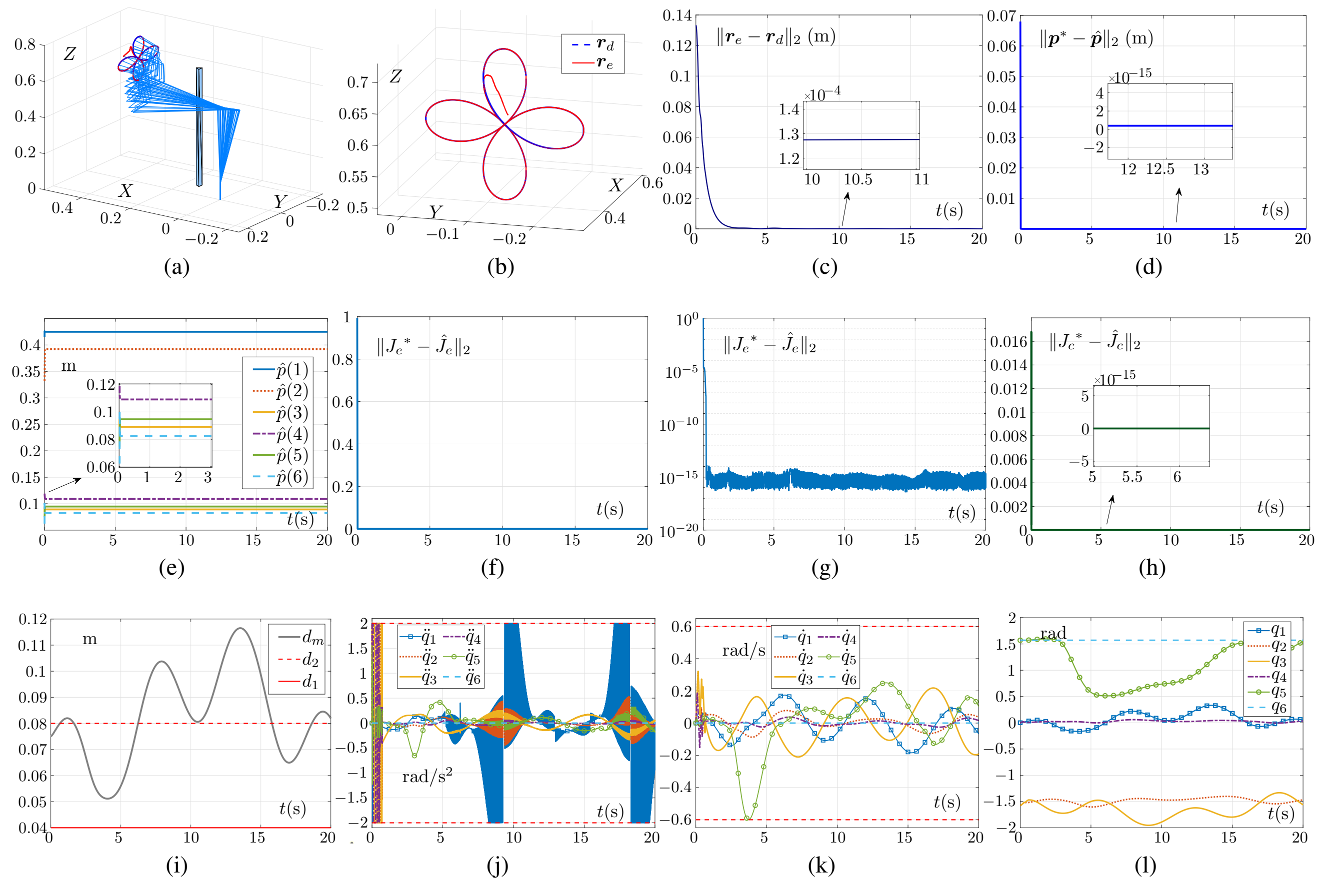

Figure 2: Experimental results by MT-QPNN solver (23) to estimate parameters, avoid obstacles and track trajectory (a) Motion process of the manipulator. (b) Desired and actual trajectories. (c) Norm of tracking error. (d) Norm of physical parameter error. (e) Estimated physical parameters. (f) Norm of the estimated Jacobian matrix J_e. (g) Error accuracy of the estimated Jacobian matrix. (h) Norm of estimated Jacobian matrix J_c. (i) Distances between the obstacle and manipulator links. (j) Joint acceleration. (k) Joint velocity. (l) Joint angle.

bounds for the joint velocity constraints in this experiment are set as $\mathbf{0.6}\,\mathrm{rad/s}$ and $\mathbf{-0.6}\,\mathrm{rad/s}$ respectively, and the upper and lower bounds for the joint acceleration constraints are set as $\mathbf{2}\,\mathrm{rad/s^2}$ and $\mathbf{-2}\,\mathrm{rad/s^2}$, and the limits for the derivative of the Lagrange multiplier in MT-QPNN solver (23) are set to $\mathbf{15}$ and $\mathbf{-15}$.

The related parameters of RZND method (13) for identifying the physical parameters are set as $\eta = 4000$, $\sigma = 5 \times 10^{-5}$. The parameters of MT-QPNN solver (23) are set as $\epsilon = 6666$, $\mu = 2$ and $\zeta = 10000$. In addition, since the physical parameters of the redundant manipulator are unknown, we set the initial state of the estimated parameters to $\hat{\boldsymbol{p}}(0) = [0.415, 0.332, 0.0841, 0.119, 0.075, 0.062]^{\mathrm{T}}$ m. The obstacle is modeled as a regular equilateral triangular prism, whose three vertices are $(0.14, -0.095, 0)$, $(0.14, -0.075, 0)$ and $(0.1573, -0.085, 0)$, and its height is 0.65 m. Finally, the initial joint angles are set to $[0, -\pi/2, -\pi/2, 0, \pi/2, \pi/2]^{\mathrm{T}}$ rad, the initial joint velocity and joint acceleration are set to $\mathbf{0}$.

First, the experimental results are provided in Figure 2. In Figure 2a, the 20 s experimental process is depicted as a sampling graph that sampled every 0.5 s. Figures 2b and 2c show that the actual trajectory closely coincides with the desired trajectory, and the error norm of two trajectories converges rapidly to the order of 10^{-4}. Figures 2d and 2e present the results of the physical parameter identification of the UR5 manipulator, and it is obvious that each physical parameter is quickly and accurately identified. In Figures 2f and 2g, because of the high-precision physical parameters identification, the error norm between the estimated Jacobian matrix and true Jacobian matrix remains very small, with the accuracy reaching the order of 10^{-15}. Similarly, the Jacobian matrix of the critical point C derived from the identified physical parameters converges to the true value very quickly with high accuracy in Figure 2h. Figure 2i presents the curve of the minimum distance d_m between the manipulator and the obstacle. The minimum distance is always higher than d_1, which indicates that obstacle avoidance is effective and the safe distance is maintained. The changes in the joint acceleration, joint velocity, and joint angle of the manipulator during the experiment are listed in Figures 2j to 2l. Although the joint acceleration curves have many abrupt changes, they always comply with the pre-defined physical constraints.

Next, the comparison experiments of the Jacobian matrix obtained by the proposed RZND method (13) and other meth-

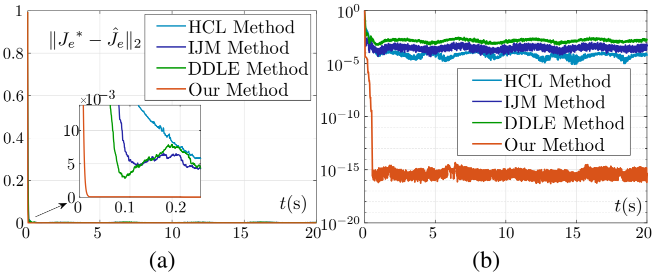

Figure 3: Identification results of the Jacobian matrix J_e obtained by four methods. (a) Norm of the estimated Jacobian matrix error. (b) Accuracy of the Jacobian matrix error norm.

Different studies	Parameter estimation	Obstacle avoidance	Physical limits
Our study	yes	yes	acceleration
[Xie *et al.*, 2025a]	yes	no	velocity
[Zhang and Zhang, 2025]	no	no	velocity
[Xie *et al.*, 2025b]	no	yes	velocity

Table 1: Comparisons of the comprehensive capabilities on the manipulator with different studies.

ods are displayed in Figure 3, where the compared methods include the hierarchical control and learning network (HCL) in [Xie *et al.*, 2025a], the improved Jacobian matrix estimation law (IJM) in [Zhang and Zhang, 2025], and the data-driven learning equation (DDLE) in [Xie *et al.*, 2025b]. Figure 3a presents the error norm curves between the Jacobian matrices identified by four methods and the actual Jacobian matrices. Obviously, the Jacobian matrix identified by the proposed RZND method can converge to the true value more quickly and with higher accuracy. In Figure 3b, the corresponding accuracy of error norms is demonstrated, and it can be observed that our method drives the Jacobian identification error norm down to around 10^{-15}, which is about 10 orders of magnitude lower than the other three methods. Table 1 finally summarizes the comprehensive performance of the four studies. Our study is the only one that simultaneously enables parameter identification and obstacle avoidance while explicitly considering acceleration constraints, whereas the other three methods consider only joint velocity limits and each lacks at least one capability.

4.2 Simulation Experiments

In this subsection, the simulation experiments are conducted on the MATLAB R2022a and CoppeliaSim 4.1.0 co-simulation platform. In the experiments, the desired trajectory to be tracked is a tooth pattern and the relevant physical constraints are $\ddot{\boldsymbol{q}}^+ = [2.1, 2.2, 2.3, 2.4, 2.5, 2.6]^{\mathrm{T}}\,\mathrm{rad/s^2}$, $\ddot{\boldsymbol{q}}^- = [-2.1, -2.2, -2.3, -2.4, -2.5, -2.6]^{\mathrm{T}}\,\mathrm{rad/s^2}$, $\dot{\boldsymbol{q}}^+ = \mathbf{0.7}\,\mathrm{rad/s}$, $\dot{\boldsymbol{q}}^- = -\mathbf{0.7}\,\mathrm{rad/s}$ and $\dot{\boldsymbol{\lambda}}_1^+ = \mathbf{15}$, $\dot{\boldsymbol{\lambda}}_1^- = -\mathbf{15}$. Besides, the related parameters of our method are set to $\eta = 5000$, $\alpha = 2$, $\beta = 4$, $\epsilon = 6666$, $\mu = 2$ and $\zeta = 1000$, respectively. Similarly, the initial joint angles are set to $[0, -\pi/2, -\pi/2, 0, \pi/2, \pi/2]^{\mathrm{T}}\,\mathrm{rad}$ and the initial

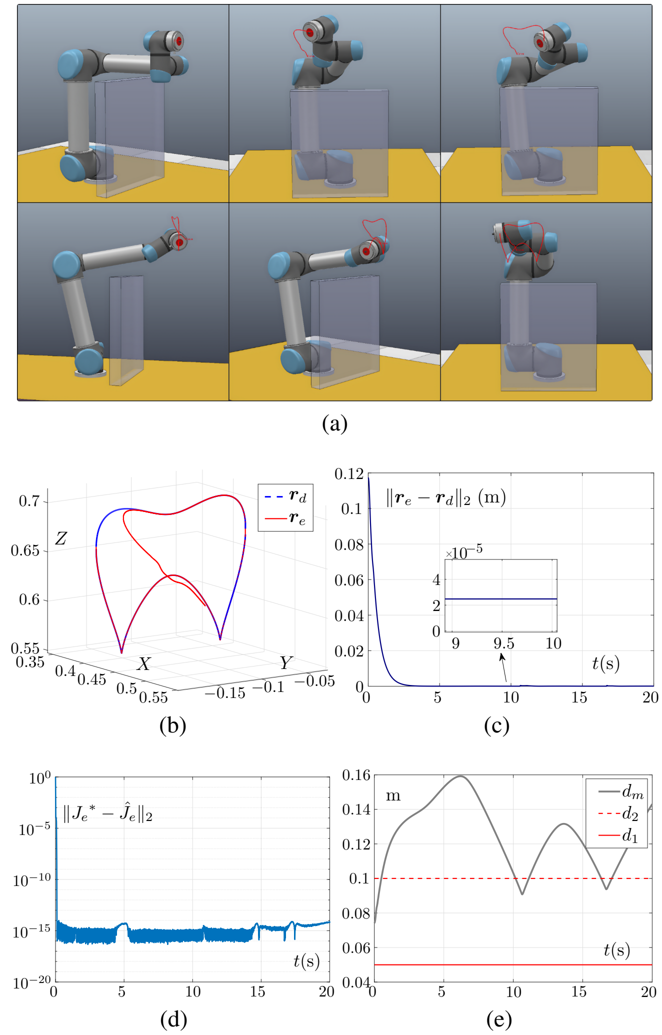

Figure 4: Experimental results on the UR5 manipulator obtained by MT-QPNN solver (23). (a) Snapshots of the motion process of manipulator. (b) Desired and actual trajectories. (c) Norm of tracking error. (d) Accuracy of the Jacobian matrix error norm. (e) Distances between the obstacle and manipulator links.

joint velocity and joint acceleration of the manipulator remain zero. Specifically, the obstacle in this simulation is a thick wooden board, which is $0.4\,\mathrm{m}$ long, $0.05\,\mathrm{m}$ wide and $0.45\,\mathrm{m}$ high. Finally, the base position of the UR5 manipulator is $(0, 0, 0)\,\mathrm{m}$ and the centroid position of the obstacle is $(0.175, 0, 0.225)\,\mathrm{m}$.

Figure 4 displays the results of this simulation, where the UR5 manipulator successfully avoids obstacles and tracks the tooth trajectory. In Figure 4a, six representative experimental snapshots are presented, and the manipulator obviously avoids the obstacle. In Figure 4b, the actual trajectory coincides with the desired trajectory, indicating excellent trajectory tracking performance. Figure 4c presents the convergence of the trajectory error norm, which rapidly and stably converges to $10^{-5}\,\mathrm{m}$. In Figure 4d, the error of the Jacobian matrix still converges to 10^{-15}, which proves that the proposed RZND method can accurately identify the parameters

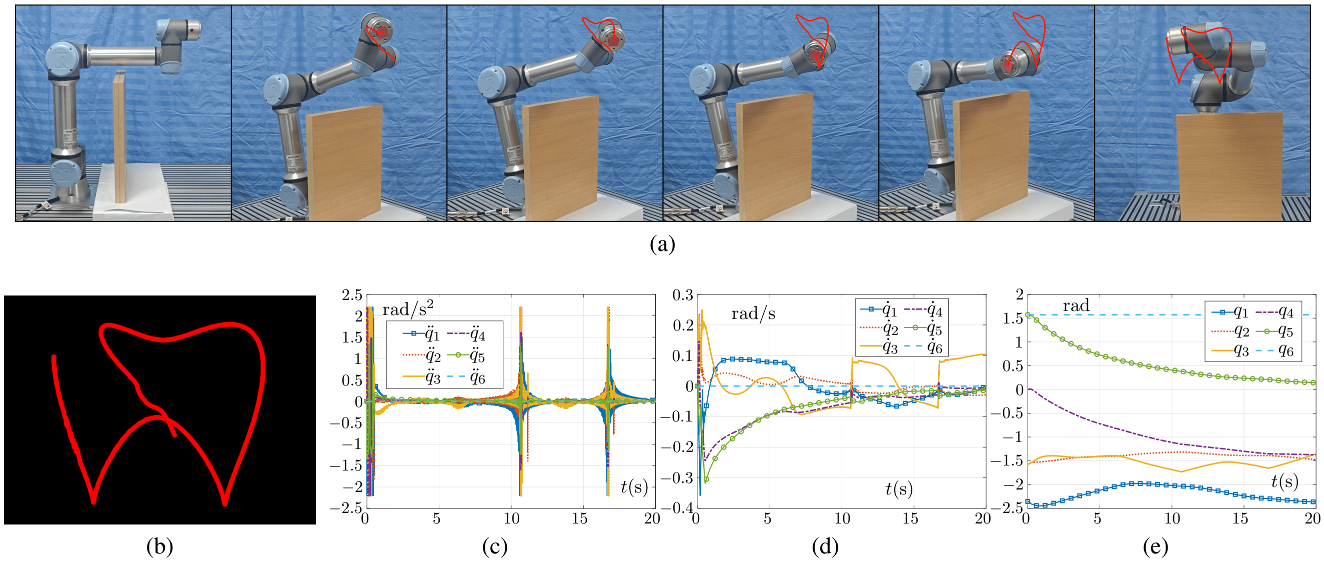

Figure 5: Experimental results on the real UR7e manipulator executing trajectory tracking and obstacle avoidance tasks by MT-QPNN solver (23). (a) Snapshots of the motion process. (b) Actual trajectory captured by OpenCV. (c) Joint acceleration. (d) Joint velocity. (e) Joint angle.

and obtain a very precise Jacobian matrix. Finally, Figure 4e exhibits the distance between the manipulator and the obstacle, and the motion of UR5 manipulator is always within the safe range in the simulation.

4.3 Real Experiments

In order to further validate the proposed MT-QPNN solver (23) in practical applications, this experiment is conducted on the real UR7e manipulator. The desired trajectory remains a tooth pattern, whose maximum radius is $0.1\,\mathrm{m}$. In addition, the thick wooden board with a length of $0.365\,\mathrm{m}$, a width of $0.035\,\mathrm{m}$ and a height of $0.45\,\mathrm{m}$ is used as the obstacle. The upper and lower constraints of the joint acceleration are $\ddot{\boldsymbol{q}}^{+} = \mathbf{2.2}\,\mathrm{rad/s^2}$ and $\ddot{\boldsymbol{q}}^{-} = \mathbf{-2.2}\,\mathrm{rad/s^2}$, the parameter ϵ in MT-QPNN solver (23) is set to 2000 and the remaining parameters are consistent with the settings in Subsection 4.2.

Figure 5 presents the results of trajectory tracking and obstacle avoidance on the UR7e manipulator. Figure 5a shows six snapshots taken every four seconds from $0\,\mathrm{s}$ to $20\,\mathrm{s}$ in the experiment. Under the MT-QPNN solver proposed in this paper, the UR7e manipulator accurately completes the trajectory tracking and obstacle avoidance tasks. Besides, the motion process of the end tool of the UR7e manipulator is captured in PyCharm 2023.3.4 using OpenCV and the actual trajectory is shown in Figure 5b. Figures 5c to 5e illustrate the joint variations of the UR7e manipulator during the experiment. The joint velocity and acceleration remain within their prescribed constraints and quickly become stable after short transient peaks, while the joint angle evolves smoothly and continuously over the whole movement, which indicates that the manipulator still moves stably under the imposed physical constraints. In summary, all these experiments in the above subsections well verify the effectiveness of the proposed RZND method (13) and MT-QPNN solver (23) in dealing with the parameter identification, obstacle avoidance and trajectory tracking of the redundant manipulator.

5 Conclusion

This paper addresses collision-free trajectory tracking for redundant manipulators with uncertain physical parameters under joint velocity and acceleration constraints, which is formulated as a QP problem. First, we propose a RZND method to identify the physical parameters of the redundant manipulator with high precision and concurrently to update the Jacobian matrix in real time. Based on this, an MT-QPNN solver is presented to solve the QP problem and thereby control the redundant manipulator to perform trajectory tracking and obstacle avoidance motions under multiple constraints. It is worth pointing out that compared with previous parameter identification or Jacobian matrix adaptive methods, the identified physical parameters and the updated Jacobian matrix obtained by the proposed RZND method have errors of only 10^{-15} compared with the true values. Finally, the superior performance of the proposed scheme is adequately verified through numerical, simulation and real experiments.

Acknowledgments

This work was supported by the Natural Science Foundation of Inner Mongolia Autonomous Region under Grant 2024QN06001.

References

[Boyd and Vandenberghe, 2004] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[Chen *et al.*, 2024] Shukang Chen, Xiyuan Zhang, Naimeng Cang, Dongsheng Guo, Weidong Zhang, and Shan Xue. Design and validation of new obstacle avoidance scheme with noise suppression capability for redundant robot manipulators. In *2024 14th Asian Control Conference (ASCC)*, pages 2301–2306, 2024.

[Gombolay, 2024] Matthew Gombolay. Human-robot alignment through interactivity and interpretability: Don't assume a "spherical human". In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 8523–8528, 2024.

[He *et al.*, 2025] Yongjun He, Lin Xiao, Qiuyue Zuo, Hang Cai, and Yaonan Wang. A predefined-time robust sliding mode control based on zeroing neural dynamics for position and attitude tracking of quadrotor. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 55(4):2438–2450, 2025.

[Huang *et al.*, 2025] Yisen Huang, Jian Li, Weibing Li, Xue Zhang, Yichong Sun, Ke Xie, Yingbai Hu, Philip Wai Yan Chiu, and Zheng Li. An accelerated anti-noise adaptive neural network for robotic flexible endoscope with multitype surgical objectives and constraints. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 55(2):990–1003, 2025.

[Koutras *et al.*, 2023] Leonidas Koutras, Konstantinos Vlachos, George S. Kanakis, Fotios Dimeas, Zoe Doulgeri, and George A. Rovithakis. Enforcing constraints for dynamic obstacle avoidance by compliant robots. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5221–5227, 2023.

[Li *et al.*, 2023] Weibing Li, Zilian Yi, Yanying Zou, Haimei Wu, Yang Yang, and Yongping Pan. A novel obstacle-avoidance solution with non-iterative neural controller for joint-constrained redundant manipulators. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 237–243, 2023.

[Li *et al.*, 2025] Weibing Li, Zilian Yi, Ruiqi Rao, Dongsheng Guo, Yongping Pan, and Kai Huang. A quadratic programming solution to position-level repetitive motion planning and obstacle avoidance of joint-constrained robots. *IEEE Transactions on Industrial Electronics*, 72(2):1860–1870, 2025.

[Liu and Cao, 2025] Xinghua Liu and Ming Cao. High-order regularization dealing with ill-conditioned robot localization problems. *IEEE Transactions on Robotics*, 41:3539–3555, 2025.

[Liu *et al.*, 2022] Mei Liu, Jialiang Fan, Yu Zheng, Shuai Li, and Long Jin. A simultaneous learning and control scheme for redundant manipulators with physical constraints on decision variable and its derivative. *IEEE Transactions on Industrial Electronics*, 69(10):10301–10310, 2022.

[Okumura and Défago, 2023] Keisuke Okumura and Xavier Défago. Quick multi-robot motion planning by combining sampling and search. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI)*, pages 252–261, 2023.

[Roveda *et al.*, 2023] Loris Roveda, Andrea Testa, Asad Ali Shahid, Francesco Braghin, and Dario Piga. Q-learning-based model predictive variable impedance control for physical human-robot collaboration (extended abstract). In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI)*, pages 6959–6963, 2023.

[Shu *et al.*, 2024] Tianye Shu, Ke Shang, Cheng Gong, Yang Nan, and Hisao Ishibuchi. Learning pareto set for multi-objective continuous robot control. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4920–4928, 2024.

[Song and Lin, 2024] Kai-Tai Song and Chih-Hsuan Lin. MPC-based optimization design for 3D collision avoidance of a mobile manipulator based-on obstacle velocity estimation. In *2024 International Automatic Control Conference (CACS)*, pages 1–6, 2024.

[Sun *et al.*, 2025] Zhongbo Sun, Shijun Tang, Yuzhe Fei, Xingtian Xiao, Yunfeng Hu, and Junzhi Yu. An orthogonal repetitive motion and obstacle avoidance scheme for omnidirectional mobile robotic arm. *IEEE Transactions on Industrial Electronics*, 72(5):4978–4989, 2025.

[Xie *et al.*, 2025a] Zhengtai Xie, Long Jin, and Xin Lv. A hierarchical control and learning network for redundant manipulators with unknown physical parameters. *IEEE Transactions on Industrial Electronics*, 72(7):7106–7116, 2025.

[Xie *et al.*, 2025b] Zhengtai Xie, Mei Liu, Zhenming Su, Zhongbo Sun, and Long Jin. A data-driven obstacle avoidance scheme for redundant robots with unknown structures. *IEEE Transactions on Industrial Informatics*, 21(2):1793–1802, 2025.

[Yan and Di, 2022] Fuhan Yan and Kai Di. Multi-robot task allocation in the environment with functional tasks. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4710–4716, 2022.

[Yu *et al.*, 2025] Peng Yu, Ning Tan, and Xiaoyi Gu. A hybrid neurodynamic scheme for bimanual synchronized tracking control of robotic manipulators with uncertain kinematics. *IEEE Transactions on Industrial Informatics*, 21(4):2967–2976, 2025.

[Zhang and Zhang, 2025] Mingyang Zhang and Zhijun Zhang. A data-driven distributed recurrent neural network for a collaborative system of multiple redundant manipulators with unknown structure. *IEEE Transactions on Cybernetics*, 55(12):5661–5672, 2025.

[Zhang *et al.*, 2018] Yinyan Zhang, Siyuan Chen, Shuai Li, and Zhijun Zhang. Adaptive projection neural network for kinematic control of redundant manipulators with unknown physical parameters. *IEEE Transactions on Industrial Electronics*, 65(6):4909–4920, 2018.

[Zhang *et al.*, 2020] Zhijun Zhang, Siyuan Chen, Xupeng Zhu, and Ziyi Yan. Two hybrid end-effector posture-maintaining and obstacle-limits avoidance schemes for redundant robot manipulators. *IEEE Transactions on Industrial Informatics*, 16(2):754–763, 2020.

[Zhang, 2022] Yinyan Zhang. Tri-projection neural network for redundant manipulators. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 69(12):4879–4883, 2022.