WILEY | Hindawi

Research Article

Acceleration Level Control of Redundant Manipulators with Physical Constraints Compliance and Disturbance Rejection under Complex Environment

Jinglun Liang,[1] Yisheng Rong,[1] Guoliang Ye,[1] Xiaoxiao Li [ID],[2] Jianwen Guo,[1] and Zhenzhen He[3]

[1]*School of Mechanical Engineering, Dongguan University of Technology, Dongguan, Guangdong 523808, China*
[2]*Guangdong Key Laboratory of Modern Control Technology, Guangdong Institute of Intelligent Manufacturing, Guangzhou, Guangdong 510070, China*
[3]*Guangdong OPT Co., Ltd., Dongguan, Guangdong, China*

Correspondence should be addressed to Xiaoxiao Li; xiaoxiaoli1993@sina.com

Investigation of joint torque constraint compliance is of significance for robot manipulators especially working in complex environments. A lot of which is attributed to that, on the one hand, it is beneficial to the improvement of both safety and reliability of the mission execution. On the other hand, the energy consumption required by the robot to complete the desired mission can be reduced. Most existing schemes do not take the joint torque limit and other inherent physical structure limits in a manipulator into account at the same time. In addition, many unavoidable uncertainties such as the external environmental disturbance and/or electromagnetism interferences in the circuit system may influence the accuracy and effectiveness of the task execution for a robot. In this study, we cast light on the acceleration level control of redundant robot manipulators considering both four physical constraint limits and interference rejection. A robust unified quadratic-programming-based hybrid control scheme is proposed, where the joint torque constraints are converted as two inequality constraints based on the robots' dynamics equation. A re-current-neural-network-based controller is designed for solving the control variable. Numerical experiments performing in PUMA 560 manipulator and planer manipulator illustrate that a rational torque distribution is obtained among the joints and the considered physical structural vectors are all restricted to the respective constraint range. In addition, even disturbed by the noise, the manipulator still successfully tracks the desired trajectory under the proposed control scheme.

1. Introduction

With the gradually mature robotic technology, the robot is being applied to all kinds of complicated or dangerous tasks such as deep-sea exploring, search, and rescue tasks in quake-hit areas [1]. It imposes a challenge on the safety and reliability of the mission execution for a robot [2, 3]. A manipulator is considered to be redundant if its degrees of freedom (DOFs) is more than the minimal ones required by the robot to complete the desired end-effector task [4–8]. Due to redundancy, except that the manipulator can complete the primary end-effector task, optimization of some performance indices and multiple additional subtasks such as physical constraints compliance [9, 10], avoiding collision with the detected obstacles [11, 12], repetitive motion planning [13, 14], etc., can be achieved simultaneously for redundant manipulators [15]. Multiobjectives-integrated hybrid tasks have been achieved in both a single manipulator [4–6] and collective ones [7–9]. As one of the important physical variables, joint torque limits compliance of a redundant manipulator is necessary to be considered. If the joint torque is suddenly enlarged, the robotic structure or the surrounding things will be possible to be damaged. Moreover, if too large torque is always imposed on the

manipulator, the service life of the motor embedded inside the robot will be reduced. More importantly, if the torque is overload, the desired end-effector task cannot be done at all owing to the deviation between the desired driving force and the actual driving force output by the motor-driven robot. Joint torque optimization is equivalent to optimize the input power of the manipulators' joint actuator [15, 16]. Therefore, to improve both the safety and reliability of mission execution for a robot manipulator and reduce energy consumption while completing the desired task, it is necessary to consider the joint torque limits of redundant manipulator among the robot control scheme.

Many efforts taking joint torque limits into account have been made. The easier schemes are solved through the null-space or pseudoinverse method [17, 18]. In addition to the found instability problem for both, the pseudoinverse method is with the higher computational cost due to the pseudoinverse of the Jacobian matrix needs to be computed. Moreover, high demand for the robot working in a complicated environment makes the robot expected to simultaneously achieve the primary end-effector task and multiple secondary subtasks. It has no ability to meet this requirement obviously owing to the pseudoinverse method that does not handle such physical constraints inherent in the robot. To solve it, the quadratic-programming- (QP-) based optimization scheme is developed and widely used, whose typical feature is that these subtasks such as obstacle avoidance and physical constraints are uniformly described as independent attachment equality or inequality constraint.

The literature [19] investigated the inverse kinematics problem of redundant manipulators subject to torque limit, where minimum torque infinity norm (MIN) was chosen as the objective function that was to be minimized, the primary task was described as an equality constraint. Therefore, a time-varying QP formulation was obtained, which was solved by the recurrent neural network (RNN) or called Lagrange neural network. Merely, the joint angle, velocity, and acceleration limits were ignored. As an extension of [19], Zhang et al. considered joint angle physical constraint in [20] by describing it as an inequality constraint. The resultant QP formulation was solved with a dual neural network, where the cost function was chosen as a minimum torque norm (MTN). In [15] and [21], physical constraints such as joint velocity and joint acceleration were considered in the QP formulation and a linear-variational-inequalities-based primal-dual neural network was then employed to solve the control variable. Five schemes were investigated in [15] by minimizing different objective functions. It was concluded that the minimum acceleration norm (MAN) was superior to the MTN scheme. However, these three papers did not take the joint torque limit into account. In [16] and [21], the bicriteria joint torque minimization was considered, where the objective functions were chosen as a weighted combination of the MTN and MIN schemes. In [14], bicriteria minimization integrating both the MTN and the repetitive motion planning schemes was studied. However, any joint limits were not considered in this paper. In [22], the motion-force control problem of redundant manipulators was investigated based on RNN. The joint

acceleration limit and the dynamics were not be considered yet. Among the above-mentioned control scheme, they did not take physical limits including joint torque, joint angle, velocity, and acceleration into account at the same time. Considering the physical structure of a robot manipulator, whose every joint is usually driven by a motor, the reachable workspace and the output torque of a robot are eventually constrained. No matter what any physical constraints are satisfied, the robot would have no ability to execute the desired task, and the output control variables are unavailable.

A consensus in [14–16, 19–21] is that neural networks are utilized to control the redundant manipulators. Recently, the neural network method is popular and has been widely used, such as in [22–30] and references therein due to its being parallel, nonlinear, and simple to be realized by hardware [31]. However, most of them are based on an assumption of disturbance-free inverse kinematic control of the manipulators among the reported products. In practice, many uncertainties such as the external environmental disturbance and/or electromagnetism interferences in circuit system inevitably exist during the control signal transmission [32–35]. Disturbed by them, the accuracy and effectiveness of the task execution may be influenced greatly by a robot system.

To sum up, although some brilliant advances have been made in joint torque optimization for redundant manipulators, there exist some limits. For one thing, some works do not take the torque limit into account or joint limits such as velocity, angle, and acceleration are not considered. For another thing, the existing torque optimization works take the disturbance into account except for reference [14]. However, [14] did not take any physical limits from robots into account. Motivated by it, this paper investigates the inverse kinematics control problem of redundant manipulator considering both the interference rejection and the above-mentioned four physical constraints compliance and proposes a robust unified QP-based hybrid optimization scheme. Among the resultant QP scheme, the MAN is chosen as the objective function, and the inverse kinematics problem is described as an equality constraint. Joint angle, velocity, and acceleration limits are uniformly described as inequality constraints and solved in acceleration level. Joint torque limits are converted to two inequality constraints based on the robot's dynamics equation. To illustrate the difference between this paper and the previous QP-based works, a comparison between them is conducted and listed in Table 1. The main contributions of this paper are summarized as follows:

(1) Based on RNN, this paper investigates the acceleration level inverse kinematics control problem of redundant manipulators with physical constraints compliance and disturbance rejection. A multi-objectives-integrated robust unified QP hybrid scheme is proposed.

(2) Different from the previous works, this paper considers joint angle, velocity, acceleration, and joint torque limits simultaneously. A simple way is given

TABLE 1: Comparison between this paper and the existing QP-based torque schemes.

Methods	Joint angle limit	Joint velocity limit	Joint acceleration limit	Joint torque limit	Disturbances	Dynamics
Our	Yes	Yes	Yes	Yes	Yes	Yes
[19]	No	No	No	No	No	Yes
[20]	Yes	No	No	No	No	Yes
[15]	Yes	Yes	Yes	No	No	Yes
[16]	No	No	No	Yes	No	Yes
[21]	Yes	Yes	Yes	No	No	Yes
[14]	No	No	No	No	Yes	Yes
[22]	Yes	Yes	No	Yes	No	No

to handle the considered four double-sides inequality constraints because they are built in different levels.

(3) Under the hybrid scheme, high joint acceleration and joint torque can be avoided during the mission execution. Constant-valued and time-varying noises are investigated, validating the robustness of the proposed scheme against external interferences.

(4) The theoretical analysis and numerical experiment combinatorially show that the effectiveness and feasibility of the designed dynamic RNN controller for the redundant manipulators' acceleration level control.

2. QP Problem Formulation

In this paper, we are aimed at achieving that in addition to successfully complete the desired end-effector tracking task, high joint angle, velocity, acceleration, and torque are all avoided, and they all should be restricted within the respective constrained range. At the same time, the control scheme is expected to be robust against external interferences. Therefore, three objectives need to be achieved simultaneously. They are as follows:

Objective 1 (inverse kinematic control): The inverse kinematic control of the manipulator is a fundamental problem in robotic control. Given the desired trajectory of the manipulators' end-effector, to find the corresponding joint variables is called the inverse kinematics problem of the robot. As for Objective 1, we have the following:

$$r \longrightarrow r_d, \tag{1}$$

where $r_d \in \mathbb{R}^m$ denotes the desired end-effector coordinate that the robot is expected to follow in Cartesian space. The actual end-effector coordinate achieved by the manipulator is denoted by the vector r and $r \in \mathbb{R}^m$. In addition, the relationship between the Cartesian coordinate r of the manipulators' end-effector and its joint space coordinate vector θ is described as follows:

$$r = f(\theta), \tag{2}$$

where $\theta \in \mathbb{R}^n$. $f(\cdot)$ denotes a nonlinear mapping from joint space to Cartesian space.

Objective 2 (physical constraint compliance): Every joint is usually driven by a motor for a manipulator. Therefore its reachable workspace and the output joint torque, velocity, and acceleration are all limited. No matter which constraints are violated, the accuracy of task execution for a manipulator will be affected. For practicability and safety, therefore, overloaded or high joint velocity, acceleration, and joint torque should be avoided when the manipulator executes the desired task. These above-mentioned robot physical limits can be described by the following two-sides inequality constraints:

$$\theta^- \leq \theta \leq \theta^+, \tag{3a}$$

$$\dot{\theta}^- \leq \dot{\theta} \leq \dot{\theta}^+, \tag{3b}$$

$$\ddot{\theta}^- \leq \ddot{\theta} \leq \ddot{\theta}^+, \tag{3c}$$

$$\tau^- \leq \tau \leq \tau^+, \tag{3d}$$

where the variables θ, $\dot{\theta}$, $\ddot{\theta}$, and $\tau \in \mathbb{R}^n$ are joint angle vector, joint velocity vector, joint acceleration vector, and joint torque vector of the manipulator, respectively. θ^-, $\dot{\theta}^-$, $\ddot{\theta}^-$, τ^- and θ^+, $\dot{\theta}^+$, $\ddot{\theta}^+$, τ^+ correspond to lower bound and upper bound of θ, $\dot{\theta}$, $\ddot{\theta}$, τ, respectively.

Due to redundancy ($m < n$), the unique solution satisfying equation (2) does not exist. In addition, because joint variables θ, $\dot{\theta}$, $\ddot{\theta}$, and τ are built in different levels, directly solving them is difficult. In this paper, the control problem is solved at the acceleration level. Specially, computing the second-derivatives of equation (2), the acceleration level kinematics is described as follows:

$$\ddot{r} = J\ddot{\theta} + \dot{J}\dot{\theta}, \tag{4}$$

where $\dot{J}$ is derivative of J. $J \in \mathbb{R}^{m \times n}$ is a Jacobian matrix, determined by the DH parameters of the manipulator. In general, the manipulators' physical structure and DH parameters are known in advance; therefore J and $\dot{J}$ are convenient to be obtained. $\ddot{r}$ is the second-derivative of r, denoting acceleration vector of the robot in Cartesian space.

As for the two-sides inequality constraints equations (3a)–(3c), based on [36, 37], they can be described as follows:

$$\xi^+ = \min\left\{\kappa_1\left(\theta^+ - \vartheta - \theta\right), \kappa_2\left(\dot{\theta}^+ - \dot{\theta}\right), \ddot{\theta}^+\right\},$$
$$\xi^- = \max\left\{\kappa_1\left(\theta^- + \vartheta - \theta\right), \kappa_2\left(\dot{\theta}^- - \dot{\theta}\right), \ddot{\theta}^-\right\}, \tag{5}$$

where constants $\vartheta, \kappa_1, \kappa_2 > 0 \in \mathbb{R}$ are adjusted by the designer based on the required experimental results. For τ, based on the manipulator's dynamics, it is obtained [15, 20] that

$$\tau = M(\theta)\ddot{\theta} + c(\theta, \dot{\theta}) + g(\theta), \tag{6}$$

where $M(\theta) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $c(\theta, \dot{\theta}) \in \mathbb{R}^n$ is the Coriolis and centrifugal force vector, and $g(\theta) \in \mathbb{R}^n$ is the gravitational force vector. Therefore, equation (3d) can be rewritten as two inequality constraints related to $\ddot{\theta}$:

$$\begin{cases} M(\theta)\ddot{\theta} \leq \tau^+ - c(\theta, \dot{\theta}) - g(\theta), \\ -M(\theta)\ddot{\theta} \leq -\tau^- + c(\theta, \dot{\theta}) + g(\theta). \end{cases} \tag{7}$$

Objective 3 (disturbance rejection): Many uncertainties originating from external and internal interferences are possible to influence both the accuracy and effectiveness of the manipulator's mission execution. Uncertainties caused by the change of external environment or interaction between the robot and environment are viewed as external disturbances. Uncertainties caused by the internal parameters deviation among the manipulator structure are then viewed as the internal disturbances, such as differential error.

In this paper, we focus on the external disturbance rejection. The rejection investigation of the internal disturbance can refer to [14]. Uncertainties that disturb the task execution precision of the robot system can be described mathematically. As described in [35], the external disturbances can be mathematically described as linear-form disturbance, sine-form, exponential-form, and random disturbances and so on, respectively. Specifically, offset errors in robot hardware implementation could be described as linear disturbances. In the process of signal processing and transmission for robots, the signal disturbances caused by electromagnetic interference are described as sine-form disturbances. The case of instantaneous decline of a power source which causes the loss of control signal is viewed as the exponential-decay-form disturbance. Interferences caused by the change of external environment is then described as a random disturbance. The Objective 1 disturbed by the external disturbance in acceleration level can be rewritten as follows:

$$\ddot{r} + \omega \longrightarrow \ddot{r}_d, \qquad J\ddot{\theta} + \dot{J}\dot{\theta} + \omega \longrightarrow \ddot{r}_d, \tag{8}$$

where $\omega \in \mathbb{R}^m$ denotes the external disturbance vector. $\ddot{r}_d$ is the second-order derivation of r_d, denoting the desired acceleration vector.

To reject these noises and achieve $\ddot{r} + \omega \longrightarrow \ddot{r}_d$, inspired by [38–40], an integration-enhanced negative feedback is introduced in equation (8), that is to say,

$$J\ddot{\theta} = \ddot{r}_d - \dot{J}\dot{\theta} - \alpha(r - r_d) - \beta\left(J\dot{\theta} - \dot{r}_d\right) - \gamma\int_0^t (r - r_d)\mathrm{d}t + \omega, \tag{9}$$

where constants $\alpha, \beta, \gamma > 0 \in \mathbb{R}$ are used to scale the tracking accuracy to the desired tracking trajectory. $\dot{r}_d$ denotes the desired velocity vector. The specific derivation of equation (9) would be shown in the appendix.

In conclusion, the above-mentioned three objectives are able to be uniformly described as a dynamic QP formulation:

$$\min_{\ddot{\theta}} \ddot{\theta}^T\left(\frac{\ddot{\theta}}{2}\right), \tag{10a}$$

$$\text{s.t.} \quad J\ddot{\theta} = b_0, \tag{10b}$$

$$M(\theta)\ddot{\theta} \leq b_1, \tag{10c}$$

$$-M(\theta)\ddot{\theta} \leq b_2, \tag{10d}$$

$$\xi^+ = \min\left\{\kappa_1\left(\theta^+ - \vartheta - \theta\right), \kappa_2\left(\dot{\theta}^+ - \dot{\theta}\right), \ddot{\theta}^+\right\}, \tag{10e}$$

$$\xi^- = \max\left\{\kappa_1\left(\theta^- + \vartheta - \theta\right), \kappa_2\left(\dot{\theta}^- - \dot{\theta}\right), \ddot{\theta}^-\right\}, \tag{10f}$$

where the cost function is chosen as the minimization of the joint acceleration norm. b_0, b_1, and b_2 are defined as $b_0 = \ddot{r}_d - \dot{J}\dot{\theta} - \alpha e - \beta\dot{e} - \gamma\int_0^t e\mathrm{d}t + \omega$, $b_1 = \tau^+ - c(\theta, \dot{\theta}) - g(\theta)$, $b_2 = -\tau^- + c(\theta, \dot{\theta}) + g(\theta)$, respectively. $e = r - r_d$, denoting the tracking error between the desired trajectory and the actual trajectory achieved by the manipulator, and $\dot{e}$ is derivative of e, $\dot{e} = J\dot{\theta} - \dot{r}_d$.

Remark 1. For comparison, we give the velocity level QP formulation without considering the joint acceleration and torque limits in [12], as follows:

$$\min_{\dot{\theta}} \dot{\theta}^T\left(\frac{\dot{\theta}}{2}\right), \tag{11a}$$

$$\text{s.t.} \quad J\dot{\theta} = \dot{r}_d - k_1(r - r_d), \tag{11b}$$

$$\max\left(k_2\left(\theta^- - \theta\right), \dot{\theta}^-\right) \leq \dot{\theta} \leq \min\left(\dot{\theta}^+, k_2\left(\theta^+ - \theta\right)\right), \tag{11c}$$

where the minimization of the joint velocity norm is chosen as the cost function because the kinematics of redundant manipulator is solved in velocity level. Parameters k_1 is similar to α, β, γ in equation (10), $k_2 > 0 \in \mathbb{R}$.

3. RNN Solver

In this section, we will design a dynamic neural solver to solve equation (10) based on RNN. Then, the theoretical proof is given that under the designed solver, the optimal solution of equation (10) can be found.

To solve equation (10), a Lagrange function is defined as follows:

$$L = \ddot{\theta}^T \frac{\ddot{\theta}}{2} + \lambda_1^T\left(b_0 - J\ddot{\theta}\right) + \lambda_2^T\left(M(\theta)\ddot{\theta} - b_1\right) + \lambda_3^T\left(-M(\theta)\ddot{\theta} - b_2\right), \tag{12}$$

where λ_1, λ_2, and λ_3 are the Lagrange multiplier. Based on the KKT conditions, the optimal solution of equation (12) can be equivalently rewritten as follows:

$$\ddot{\theta} = P_\Omega\left(\ddot{\theta} - \frac{\partial L}{\partial \ddot{\theta}}\right), \tag{13a}$$

$$J\ddot{\theta} = b_0, \tag{13b}$$

$$\begin{cases} \lambda_2 = 0, & \text{if } M(\theta)\ddot{\theta} \le b_1, \\ \lambda_2 > 0, & \text{otehwise,} \end{cases} \tag{13c}$$

$$\begin{cases} \lambda_3 = 0, & \text{if } -M(\theta)\ddot{\theta} \le b_2, \\ \lambda_3 > 0, & \text{otehwise,} \end{cases} \tag{13d}$$

where P_Ω is a projection operation to a set Ω, and $P_\Omega(x) = \arg\min_{y\in\Omega}\|y - x\|$ [41]. Equations (13c) and (13d) can be further written as follows:

$$\begin{aligned} \lambda_2 &= \max\left(\left(\lambda_2 + M(\theta)\ddot{\theta} - b_1\right), 0\right), \\ \lambda_3 &= \max\left(\left(\lambda_3 - M(\theta)\ddot{\theta} - b_2\right), 0\right). \end{aligned} \tag{14}$$

The designed RNN controller is as follows:

$$\epsilon\ddot{\theta} = -\ddot{\theta} + P_\Omega\left(J^T\lambda_1 - M(\theta)^T\lambda_2 + M(\theta)^T\lambda_3\right), \tag{15a}$$

$$\epsilon\dot{\lambda}_1 = b_0 - J\ddot{\theta}, \tag{15b}$$

$$\epsilon\dot{\lambda}_2 = \max\left(\left(M(\theta)\ddot{\theta} - b_1 + \lambda_2\right), 0\right) - \lambda_2, \tag{15c}$$

$$\epsilon\dot{\lambda}_3 = \max\left(\left(-M(\theta)\ddot{\theta} - b_2 + \lambda_3\right), 0\right) - \lambda_3, \tag{15d}$$

where $(\epsilon > 0)$ is a constant which is used to scale the convergence rate of the neural network. In general, the smaller ϵ, the faster the RNN controller converges.

Rewrite the designed RNN controller equation (15) as follows:

$$\epsilon\begin{bmatrix} \ddot{\theta} \\ \dot{\lambda}_1 \\ \dot{\lambda}_2 \\ \dot{\lambda}_3 \end{bmatrix} = \begin{bmatrix} -\ddot{\theta} + P_\Omega\left(\ddot{\theta} + \delta - \ddot{\theta}\right) \\ -\lambda_1 + \left(\lambda_1 + b_0 - J\ddot{\theta}\right) \\ -\lambda_2 + \left(\lambda_2 + M(\theta)\ddot{\theta} - b_1\right) \\ -\lambda_3 + \left(\lambda_3 - M(\theta)\ddot{\theta} - b_2\right) \end{bmatrix}, \tag{16}$$

where $\delta = J^T\lambda_1 - M(\theta)^T\lambda_2 + M(\theta)^T\lambda_3$.

Let $\eta = [\ddot{\theta}^T, \lambda_1^T, \lambda_2^T, \lambda_3^T]^T$, equation (16) is reformulated as follows:

$$\epsilon\dot{\eta} = -\eta + P_{\overline{\Omega}}(\eta - \mathscr{F}(\eta)), \tag{17}$$

in which

$$\mathscr{F}(\eta) = \begin{bmatrix} \ddot{\theta} - J^T\lambda_1 + M(\theta)^T\lambda_2 - M(\theta)^T\lambda_3 \\ J\ddot{\theta} - b_0 \\ -M(\theta)^T\ddot{\theta} - b_1 \\ M(\theta)^T\ddot{\theta} + b_2 \end{bmatrix}, \tag{18}$$

Therefore, the gradient of $\mathscr{F}$ is as follows:

$$\nabla\mathscr{F} = \frac{\partial\mathscr{F}}{\partial\eta} = \begin{bmatrix} I & -J^T & M(\theta)^T & -M(\theta)^T \\ J & 0 & 0 & 0 \\ -M(\theta)^T & 0 & 0 & 0 \\ M(\theta)^T & 0 & 0 & 0 \end{bmatrix}, \tag{19}$$

$$\nabla\mathscr{F}(\eta) + \nabla\mathscr{F}^T(\eta) = \begin{bmatrix} 2I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{ is positive semidefinite;}$$

therefore $\mathscr{F}(\eta)$ is a monotone function. Following [9], it can be said that the constructed RNN controller equation (15) will globally converge to an equilibrium that is equivalent to the optimal solution of equation (10).

4. Numerical Experiments

In this part, numerical experiments are performed based on MATLAB ROBOTIC TOOLBOX. The effectiveness and robustness of the proposed QP control scheme equation (10) are shown by a six-DOFs PUMA 560 manipulator and a four-DOFs planer manipulator to show application feasibility of the proposed scheme equation (10) in all kinds of redundant robot manipulators.

4.1. PUMA 560 Manipulator Experiments. In this experiment, two different trajectory tracking experiments, i.e., the butterfly and triangle trajectories, are conducted. Table 2 shows DH parameters of the used PUMA 560 redundant manipulator. Other parameters involved in the simulation experiment are concluded in Table 3, where $\theta(0)$, $\dot{\theta}(0)$, $\ddot{\theta}(0)$ denote the initial value of θ, $\dot{\theta}$, $\ddot{\theta}$ at the initial instant, respectively. The simulation duration T_d is 20 s.

4.1.1. Butterfly Trajectory Tracking. Assume that the manipulator is required to track a butterfly trajectory with the following definition:

$$r_d = \begin{bmatrix} 0.05\sin(0.5t)(\exp(a) - 2b) + 0.4521 \\ 0.05\cos(0.5t)(\exp(a) - 2b) - 0.3 \\ 0.6318 \end{bmatrix}, \tag{20}$$

TABLE 2: The D-H parameter of the redundant manipulator PUMA 560 used in this paper.

Link	a (m)	α (rad)	d (m)
1	0	$(\pi/2)$	0
2	0.4318	0	0
3	0.0203	$-(\pi/2)$	0.15005
4	0	$(\pi/2)$	0.4318
5	0	$-(\pi/2)$	0
6	0	0	0.2

TABLE 3: Parameters involved in the simulative experiment.

Parameters	Value	Parameters	Value
$\theta(0)$	$0 \in \mathbb{R}^6$	θ^+	$+4$
$\dot{\theta}(0)$	$0 \in \mathbb{R}^6$	θ^-	-4
$\ddot{\theta}(0)$	$0 \in \mathbb{R}^6$	$\dot{\theta}^+$	$+4$
(ε)	0.002	$\dot{\theta}^-$	-4
α	21	$\ddot{\theta}^+$	$+4$
β	7	$\ddot{\theta}^-$	-4
γ	21	ϑ	0.1
κ_1	20	τ^+	40
κ_2	20	$\tau-$	-40

where $a = \cos(0.5t)$, $b = \cos(2t)$. Firstly, a comparison between the velocity level scheme equation (11) and the acceleration level scheme equation (10) is conducted with corresponding simulative results illustrated in Figure 1. Figures 1(a)–1(f) represent results achieved by equation (11). Figures 1(g)–1(l) represent the ones achieved by equation (10). From left to right are 3D tracking results, planer tracking results, tracking error profiles at x-axis, y-axis, and z-axis, joint speed profiles, joint acceleration profiles, and joint torque profiles consecutively. Note that in the velocity level experiment, parameters k_1 and k_2 are valued as 10 and 20, respectively. We observe from Figures 1(a) and 1(b) that the actual trajectory (black color path) achieved by the manipulator successfully coincides with the desired butterfly path (red color path). It can also be observed from Figure 1(c) that the deviations between the desired trajectory and the actual trajectory at x-axis, y-axis, and z-axis quickly reduces to zero. Following Figure 1(d), the manipulator's six joint speeds are smooth and within the respective constraint ranges. For now, equation (11) without considering the joint acceleration limit seems to be effective for the kinematic control of the PUMA 560 manipulator. However, in this case that does not consider the joint acceleration limit, the manipulator's acceleration and torque profiles are extremely high (up to $-200 \, (\text{rad/s}^2)$ and $-600 \, \text{Nm}$, respectively) at the beginning, as shown in Figures 1(e) and 1(f). It is unreasonable in practice, obviously. Compared to the velocity level scheme equations (11) and (10) taking the joint acceleration and the joint torque limits into account, the kinematics problem is solved in acceleration level. Following the simulation results illustrated in Figures 1(g)–1(l), what the naked eye can see is that the manipulator does not only track the desired butterfly trajectory, its joint speed, joint acceleration, and torque are
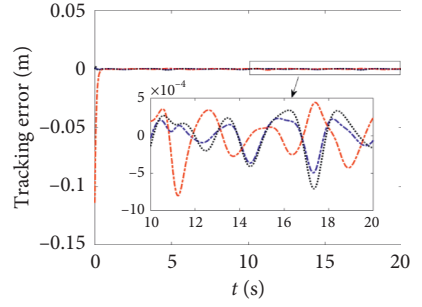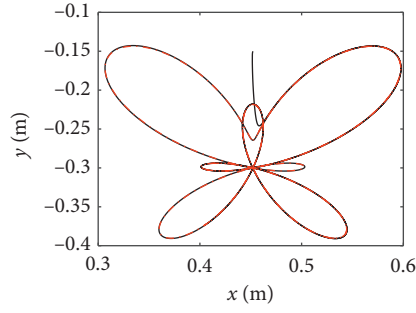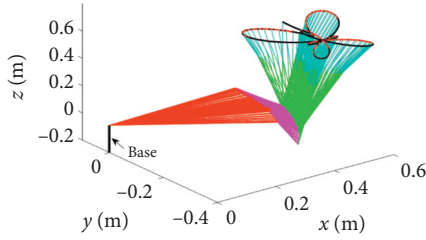
all restricted to the constrained range. As shown in Figure 1(k), when the acceleration of the joint 1 exceed its lower bound, the joint acceleration would be restricted and maintained in its lower bound, avoiding the high acceleration and joint torque. Therefore, it is concluded that the acceleration level optimization scheme considering the joint acceleration and torque limits is superior to the velocity level for the butterfly path tracking.

4.1.2. Triangle Trajectory Tracking. To further validate the effectiveness and superiority of the proposed hybrid acceleration level scheme equation (10), in this experiment, the manipulator is required to track a triangle trajectory with the following definition:

$$r_d = \begin{bmatrix} 0.075 \sin(\rho) - 0.0325 \cos(2\rho) + 0.48 \\ 0.075 \cos(\rho) - 0.0325 \sin(2\rho) - 0.10 \\ 0.6 \end{bmatrix}, \quad (21)$$

where $\rho = (2\pi t/10)$. Comparative results between the scheme equation (11) and the scheme equation (10) are illustrated in Figure 2. Figures 2(a)–2(f) represent results achieved by equation (11). Others represent ones achieved by equation (10). The simulation environment and parameters in the triangle trajectory tracking experiment are the same as the butterfly trajectory tracking experiment. Similar to the previous butterfly experiment, we observe from Figures 2(a)–2(f) that the manipulator successfully tracks the desired triangle path with a tracking error being 10^{-4} level, and the manipulator's six joint speeds are smooth and within the respective constraint ranges. However, the manipulator's acceleration and torque profiles that are shown in Figures 2(e) and 2(f) are extremely high (up to $(200 \, \text{rad/s}^2)$ and $600 \, \text{Nm}$, respectively) at the beginning. When both the inherent joint acceleration limit and torque limit are considered, the manipulator does not only track the desired triangle trajectory. Its joint speed, joint acceleration, and torque are all restricted to the respective constrained ranges. As shown in Figure 2(k), when the acceleration of joint 1 exceeds its upper bound, the joint acceleration would be restricted and maintained in the upper bound, avoiding the high acceleration and joint torque. Therefore, it is concluded that the acceleration level optimization scheme equation (10) considering the joint acceleration and torque limits is superior to the velocity level scheme equation (11). This experiment validates the effectiveness of the proposed hybrid scheme equation (10) and the RNN controller equation (15) for butterfly trajectory tracking task.

4.1.3. Robustness Comparison. In real scenes, the external disturbance is unavoidable and may influence the tracking accuracy of the manipulator in the trajectory tracking mission. To this, we propose a robust hybrid QP minimization scheme equation (10) by introducing integration-enhanced feedback to reject these interferences. In general, these disturbances can be mathematically described as constant-valued noise and time-varying noise.
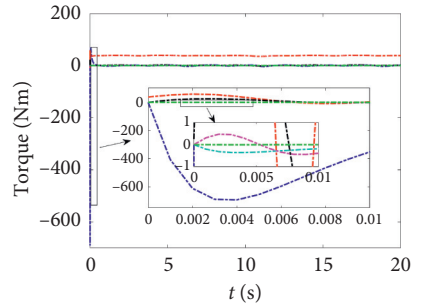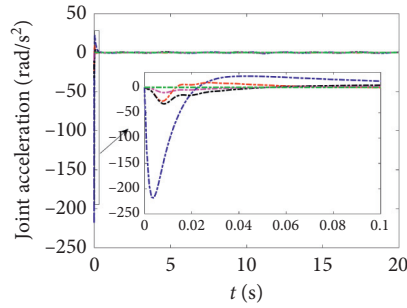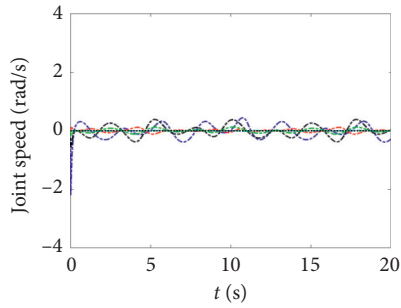
(a)

(b)

(c)

(d)

(e)

(f)
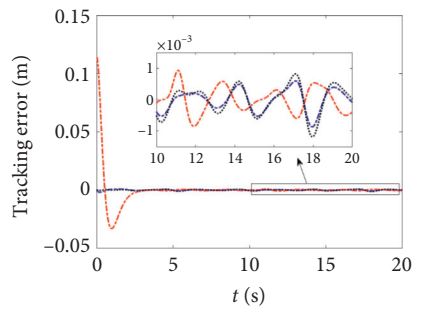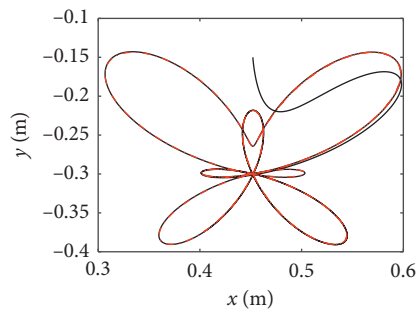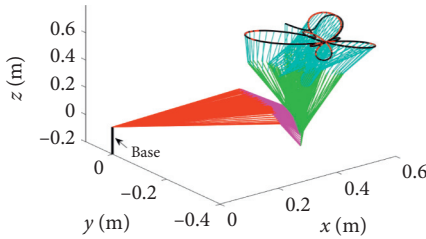
(g)

(h)
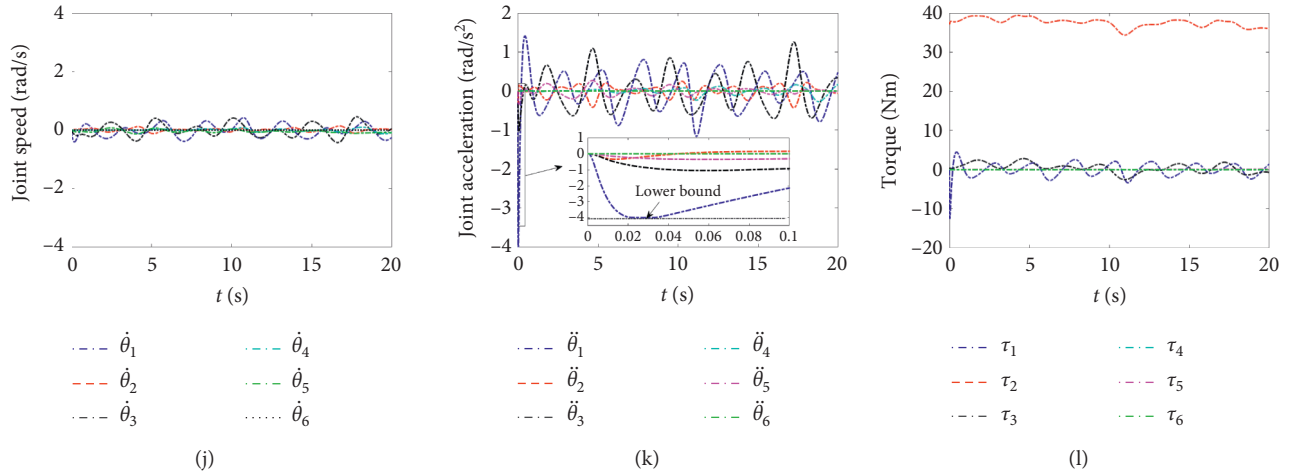
(i)

FIGURE 1: Continued.

(j)

(k)

(l)

FIGURE 1: Comparison between the velocity level scheme equation (11) (a–f) and the acceleration scheme equation (10) (g–l) when the manipulator is expected to follow a butterfly path. From left to right are 3D tracking results, planer tracking results, tracking error profiles, joint speed profiles, joint acceleration profiles, and joint torque profiles.
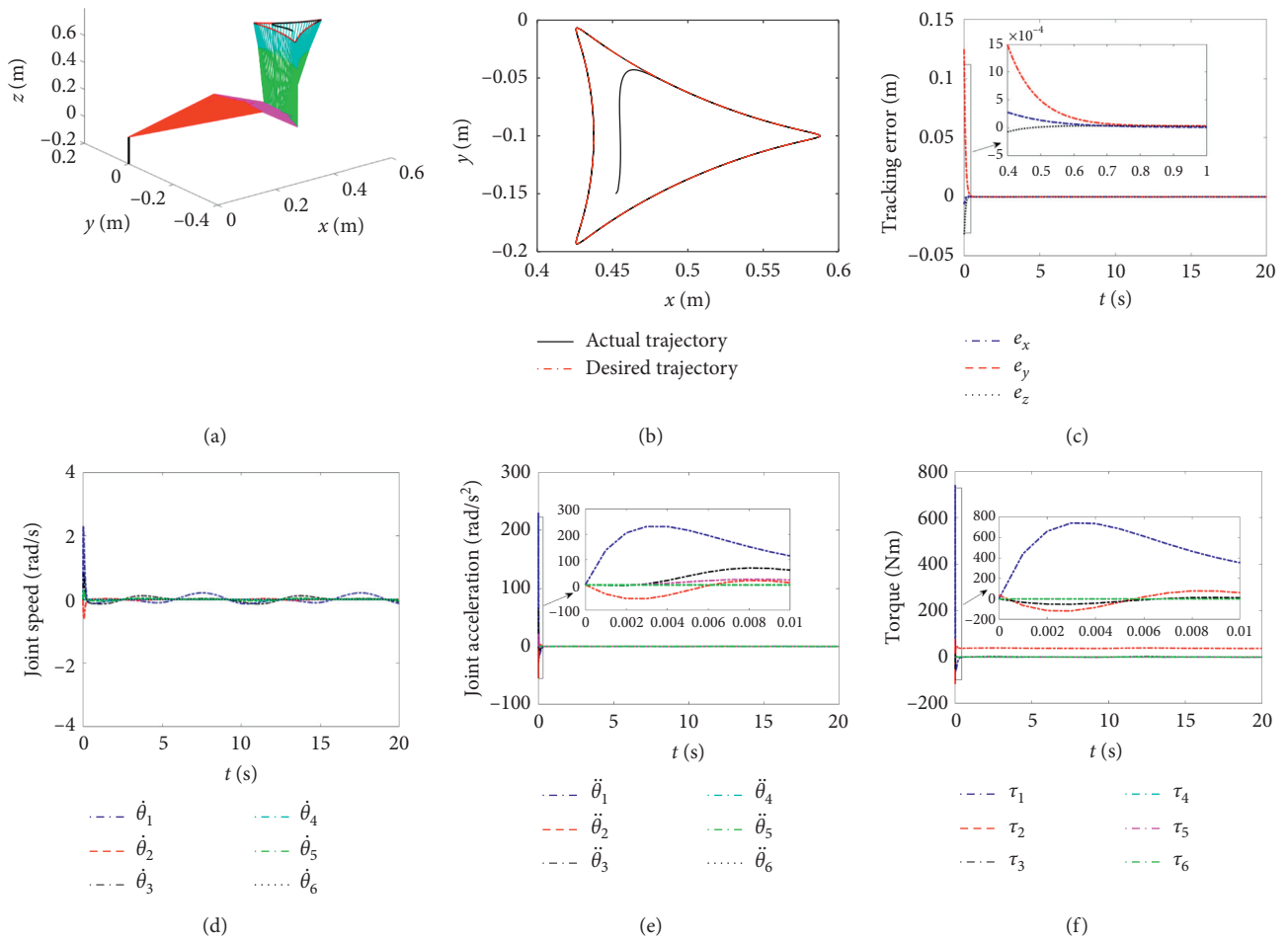

(a)

(b)

(c)
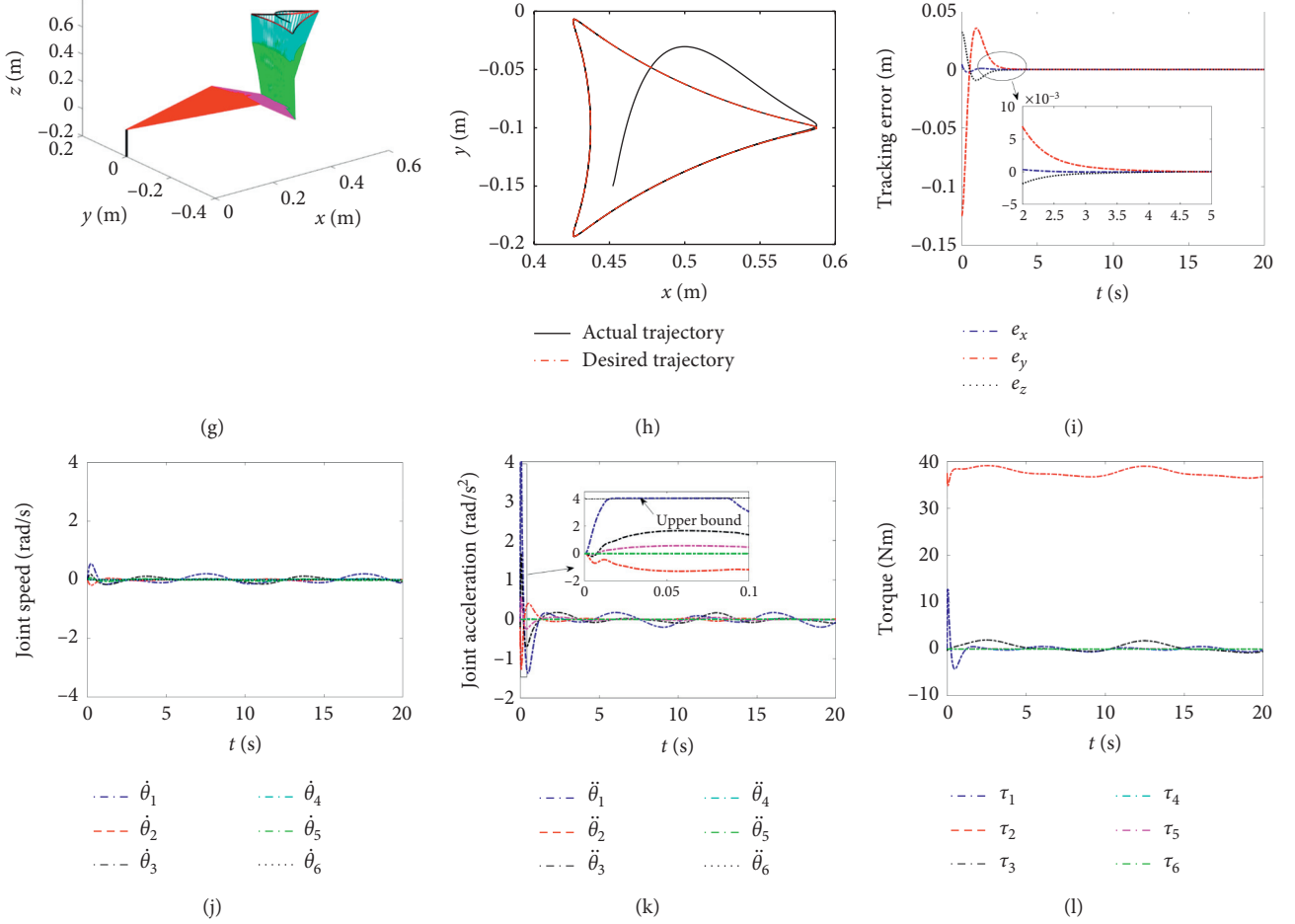
(d)

(e)

(f)

FIGURE 2: Continued.

FIGURE 2: Comparison between the velocity level scheme equation (11) (a–f) and the acceleration scheme equation (10) (g–l) when the manipulator is expected to follow a triangle path. From left to right are 3D tracking results, planer tracking results, tracking error profiles, joint speed profiles, joint acceleration profiles, and joint torque profiles.

In this part, we would show the robustness of the scheme equation (10) against external disturbances, where four common types of time-varying external disturbances are considered. They are linear disturbances denoting offset errors in robot hardware implementation, sine-form disturbances caused by signal electromagnetic interference in the process of signal processing and transmission, the exponential-decay-form disturbance, and random disturbance caused by the change of the external environment, respectively. For comparison, the conventional acceleration level scheme is given as follows:

$$\min_{\ddot{\theta}} \quad \ddot{\theta}^T \left(\frac{\ddot{\theta}}{2} \right), \tag{22a}$$

$$\text{s.t. } J\ddot{\theta} = \ddot{r}_d - \dot{J}\dot{\theta} - \delta_1 e - \delta_2 \dot{e} + \omega, \tag{22b}$$

$$M(\theta)\ddot{\theta} \le b_1, \tag{22c}$$

$$-M(\theta)\ddot{\theta} \le b_2, \tag{22d}$$

$$\xi^+ = \min\left\{ \kappa_1 \left(\theta^+ - \vartheta - \theta \right), \kappa_2 \left(\dot{\theta}^+ - \dot{\theta} \right), \ddot{\theta}^+ \right\}, \tag{22e}$$

$$\xi^- = \max\left\{ \kappa_1 \left(\theta^- + \vartheta - \theta \right), \kappa_2 \left(\dot{\theta}^- - \dot{\theta} \right), \ddot{\theta}^- \right\}, \tag{22f}$$

where $\delta_1, \delta_2 > 0 \in \mathbb{R}$ are feedback gain parameters, which are used to scale the trajectory tracking accuracy. $\delta_1 = 21, \delta_2 = 7$ in the noise-rejection experiments; other parameters are the same as the ones used in equation (10). Figure 3 shows tracking results of two desired trajectories achieved by schemes (10) and (22) disturbed by the constant-valued noise and time-varying random noise, respectively. In simulative experiments, the constant-valued disturbance and time-varying random disturbance are formulated as 1 and rand, respectively, where rand $\in \{0, 1\}$. Figure 3 shows the trajectory tracking results achieved by equation (22) and our scheme equation (10). Following it, it can be observed that the conventional scheme equation (22) successfully tracks the triangle and butterfly shapes of the desired following trajectories. However, due to the disturbance by the constant-valued and time-varying random noise, the actual
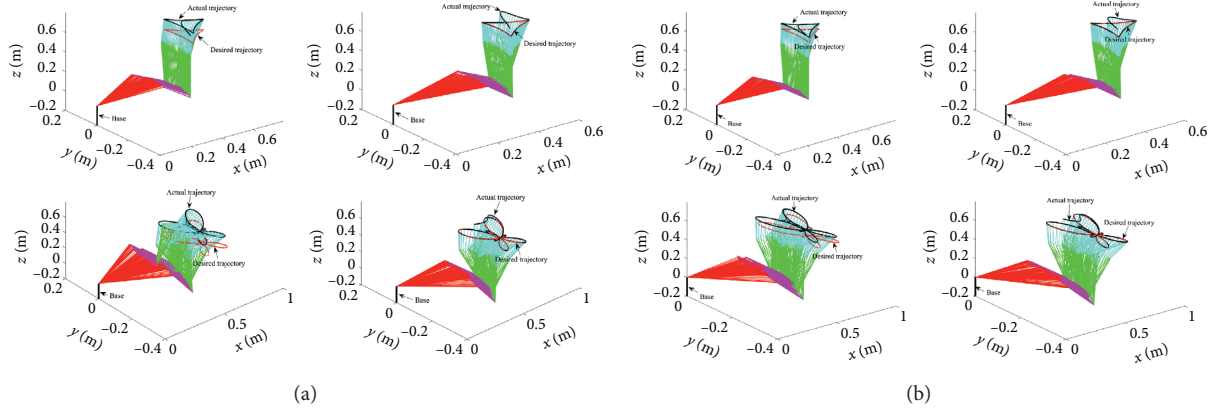
(a)

(b)

FIGURE 3: Comparison between schemes equations (22) and (10) under (a) constant-valued noise $\omega = 1$ and (b) time-varying random noise $\omega = \text{rand} \in \{0, 1\}$, respectively.
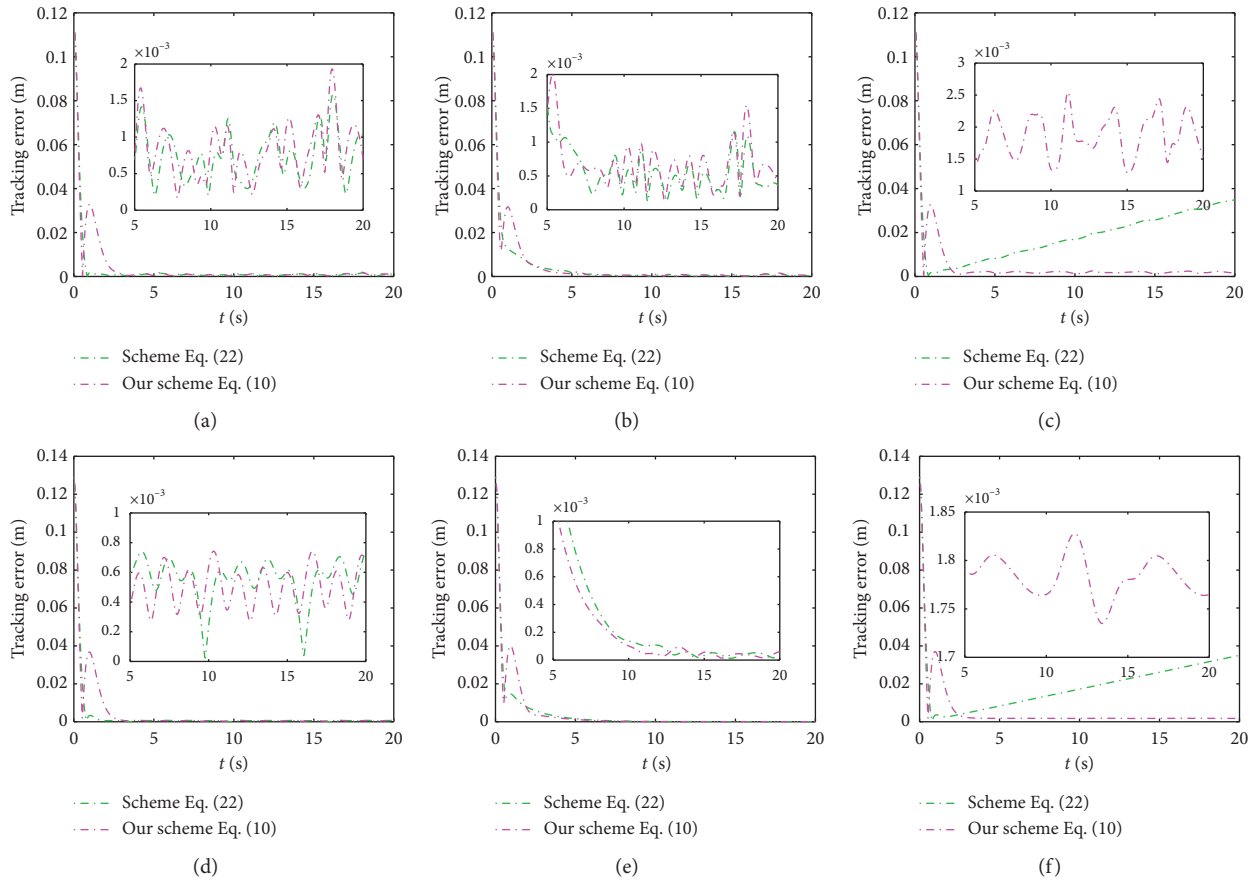

(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 4: (a–f) Tracking error comparison for butterfly and triangle trajectory tracking tasks under sine-form noise, exponential noise, and linear noise, respectively. Top: butterfly trajectory. Below: triangle trajectory.

trajectories generated by the manipulator deviate from the desired position. This is to say that equation (22) fails in rejecting these two kinds of noise. On the contrary, although disturbed by the noise, our scheme equation (10) accurately and stably generates the desired triangle and butterfly trajectories.

Figure 4 shows the tracking error profiles achieved by equations (22) and (10) disturbed by the other three time-varying external noises, where the tracking error is defined as $\sqrt{e_x^2 + e_y^2 + e_z^2}$. Linear disturbances, sine disturbances, and exponential disturbances are formulated as $[0.03t, 0.02t, 0.01t]^T$,

TABLE 4: The D-H parameter of the planer redundant manipulator with 4-DOFs.

Link	a (m)	α (rad)	d (m)
1	0.3	0	0
2	0.3	0	0
3	0.2	0	0
4	0.2	0	0

(a)

(b)

(c)

(d)

(e)

(f)
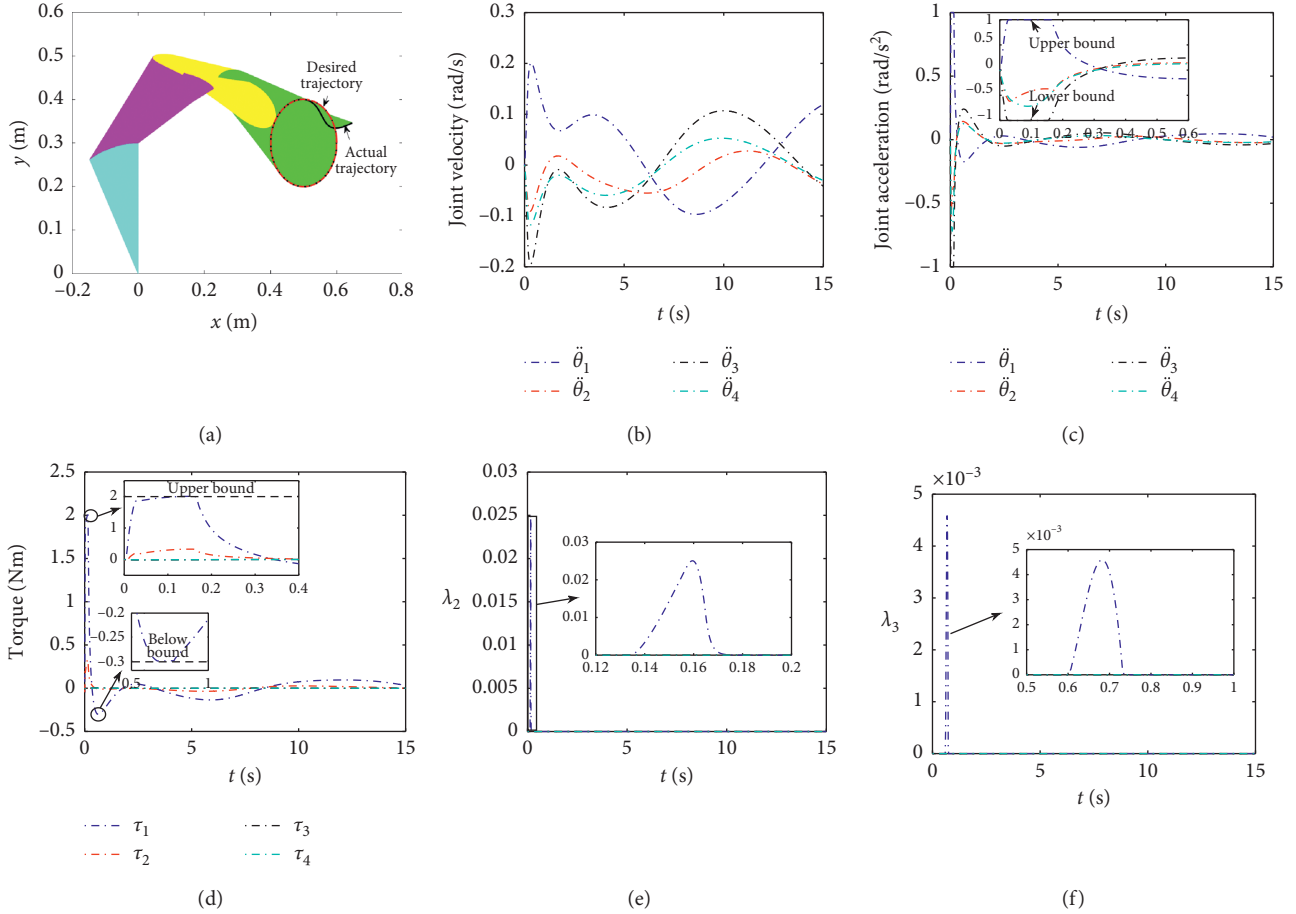
FIGURE 5: Simulative results achieved by equation (10) when the manipulator is expected to follow a circle path. (a) Tracking result. (b) Joint speed profiles. (c) Joint acceleration profiles. (d) Joint torque profiles. (e) State variables λ_2. (f) State variables λ_3.

$[0.01 \sin(2t), 0.01 \cos(0.5t), 0.01 \sin(t)]^T$, and $[0.35 \exp(-0.5t), 0.15 \exp(-0.8t); 0.1 \exp(-t)]^T$, respectively. Based on the simulative results, we conclude that under control of the schemes equations (10) and (22), the manipulator disturbed by the sine-form noise and exponential-form noise can track the desired butterfly and triangle trajectories. The tracking error can reach 10^{-3} level. For linear disturbance, the tracking errors achieved by the scheme equation (22) would not converge to zero, as shown in Figures 4(c) and 4(f). In contrast, the tracking errors corresponding to our proposed scheme equation (10) would evaluate toward zero quickly and maintain a bounded value.

4.2. Planer Manipulator Experiments. The 4-DOFs planer redundant manipulator is employed in this experiment,

which is assumed to track a circle trajectory with the following definition:

$$r_d = \begin{bmatrix} 0.5 + 0.1 \cos(0.5t) \\ 0.3 + 0.1 \sin(0.5t) \end{bmatrix}. \tag{23}$$

Table 4 gives the DH parameters corresponding to the used planer manipulator in this study. Among the simulation setup, $\theta(0) = [(\pi/2), -(\pi/3), -(\pi/4), 0]^T$ rad, $\tau^+ = 2$ Nm, $\tau^- = -0.3$ Nm, $\dot{\theta}(0) = 0 \in \mathbb{R}^4$ (rad/s), and $\ddot{\theta}(0) = 0 \in \mathbb{R}^4$ (rad/s)^2, $\ddot{\theta}^+ = -\ddot{\theta}^- = 1$. Other parameters remain the same as the previous experiment. Simulation duration is 15 s.

Simulative results achieved by the proposed hybrid scheme equation (10) are illustrated in Figure 5 when the manipulator is expected to follow a circle path. Similar to the

(a)   (b)   (c)   (d)   (e)

(f)   (g)   (h)   (i)   (j)
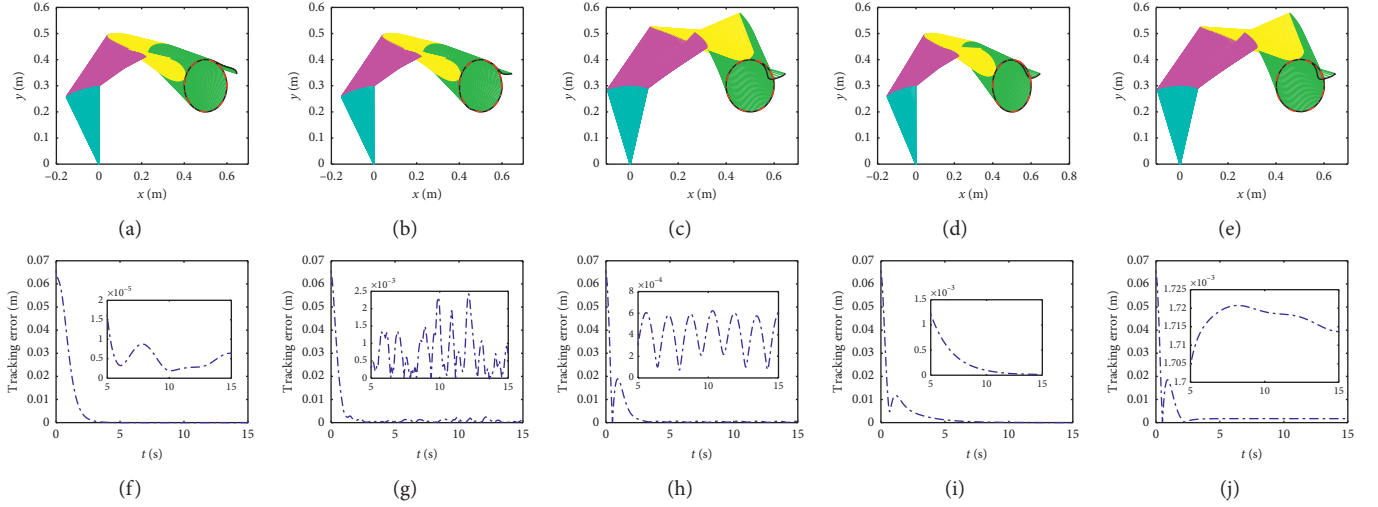
FIGURE 6: Circle trajectory tracking results and tracking errors achieved by equation (10) disturbed by the external disturbances. (a) Constant-valued noise $\omega = 1$. (b) Random noise $\omega \in \{0, 1\}$. (c) Sine-form noise $\omega = [0.01 \sin(2t), 0.01 \cos(0.5t)]^T$. (d) Exponential-form noise $\omega = [0.35 \exp(-0.5t), 0.15 \exp(-0.8t)]^T$. (e) Linear noise $\omega = [0.03t, 0.02t]^T$.

previous experiment, the actual trajectory achieved by the planer manipulator quickly coincides with the desired path, simultaneously avoiding high joint velocity, acceleration, and torque values. Following Figures 5(c) and 5(d), when joint acceleration and joint torque vectors exceed their respective bound, they would be restrained within the upper or lower bound. Figures 5(e) and 5(f) show the process profiles corresponding to state variables λ_2 and λ_3. When the inequality constraints equations (10c) and (10d) are not satisfied, i.e., joint torque bounds of the manipulator are not reached, λ_2 and λ_3 would remain on zero. On the contrary, λ_2 and λ_3 would be greater than zero. Figure 6 shows circle trajectory tracking results achieved by equation (10) disturbed by five external noises and the corresponding tracking errors. It can be seen from it that for the considered five noises, the manipulator successfully tracks the desired circle trajectory with different tracking accuracies. Compared to other types of noises, the influence of random noise on the trajectory tracking error of the manipulator is stronger.

Based on Figures 1–6, we can conclude that under the proposed hybrid scheme equation (10), the manipulator can accurately track the desired trajectory, simultaneously avoiding high joint velocity, acceleration, and joint torque. In addition, equation (10) is robust against the constant-valued external disturbance and four types of time-varying external disturbances.

5. Conclusion

In this paper, a robust multiobjectives-integrated hybrid scheme equation (10) has been proposed and used for controlling a redundant manipulator to track the desired trajectory. Under equation (10), high joint velocity, acceleration, and joint torque may exceed the manipulator actuators' ability that can be avoided. They all are restrained on respective constraint bound. In addition, integrated-enhanced feedback has been generalized into the inverse

kinematics of the manipulator to reject the influence of external disturbances on the tracking tasks. Numerical experiments have been performed on the PUMA 560 manipulator and the planer manipulator, validating the effectiveness and robustness of the hybrid scheme equation (10).

Multiple robots cooperative control is being investigated and has made success in environmental monitoring [42] and source seeking [43]. In future work, we will be devoted to the motion-force hybrid cooperative control of multiple robots at dynamics level.

Appendix

In this appendix, the authors give the derivation process of equation (9). The velocity level inverse kinematics of the manipulator is usually described as follows [6, 9, 12]:

$$\dot{r} = \dot{r}_d - \mu(r - r_d), \tag{A.1}$$

where $\mu > 0$ is a constant, which is used to scale accuracy of the trajectory tracking task.

Define a new error function $\ell = \dot{r} - \dot{r}_d + \mu(r - r_d)$. Based on [38, 39], the time-derivative evolution formulation of the error function ℓ can be constructed as follows:

$$\dot{\ell} = -\mu_1 \ell - \mu_2 \int_0^t \ell dt, \tag{A.2}$$

where the same as μ, $\mu_1, \mu_2 > 0 \in \mathbb{R}$. Differentiating ℓ, we obtain that

$$\dot{\ell} = \ddot{r} - \ddot{r}_d + \mu(\dot{r} - \dot{r}_d). \tag{A.3}$$

Combining equations (A.2) with (A.3), we have the following:

$$\begin{aligned}\ddot{r} &= \ddot{r}_d - \mu(\dot{r} - \dot{r}_d) - \mu_1(\dot{r} - \dot{r}_d) - \mu_1\mu(r - r_d) \\ &\quad - \mu_2 \int_0^t (\dot{r} - \dot{r}_d)dt - \mu_2\mu \int_0^t (r - r_d)dt.\end{aligned} \tag{A.4}$$

It can be further rewritten as follows:

$$\ddot{r} = \ddot{r}_d - (\mu + \mu_1)\left(\dot{r} - \dot{r}_d\right) - (\mu_1\mu + \mu_2)(r - r_d) - \mu_2\mu\int_0^t (r - r_d)\mathrm{d}t. \tag{A.5}$$

Because μ, μ_1, μ_2 are constants >0, therefore, equation (A.5) can be converted as follows:

$$J\ddot{\theta} = \ddot{r}_d - \dot{J}\dot{\theta} - \alpha(r - r_d) - \beta\left(J\dot{\theta} - \dot{r}_d\right) - \gamma\int_0^t (r - r_d)\mathrm{d}t, \tag{A.6}$$

by using α, β, γ to replace $\mu_1\mu + \mu_2, \mu + \mu_1, \mu_2\mu$, respectively. Li et al. have proved the inherent noise tolerance of equation (A.2) to constant-valued and time-varying disturbances in [38]. Therefore, robustness proof of equation (A.6) disturbed by external disturbances (i.e., equation (9)) is omitted here.

Data Availability

No data were used to support this study.

Conflicts of Interest

Zhenzhen He is employed at the company "Guangdong OPT Co., LTD." The remaining authors declare that there are no conflicts of interest.

Acknowledgments

References

[1] N. Wang, C. Chen, and A. Di Nuovo, "A framework of hybrid force/motion skills learning for robots," *IEEE Transactions on Cognitive and Developmental Systems*, p. 1, 2020.

[2] C. Yang, C. Zeng, Y. Cong, N. Wang, and M. Wang, "A learning framework of adaptive manipulative skills from human to robot," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1153–1161, 2018.

[3] C. Zeng and C. Yang∗, H. Cheng, Y. Li, and S.-L. Dai, Simultaneously encoding movement and sEMG-based stiffness for robotic skill learning," *IEEE Transactions on Industrial Informatics*, p. 1, 2020.

[4] Y. Zhang, S. Li, S. Kadry, and B Liao, "Recurrent neural network for kinematic control of redundant manipulators with periodic input disturbance and physical constraints," *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4194–4205, 2018.

[5] Y. Zhang, J. Wang, and Y. Xia, "A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 658–667, 2003.

[6] Z. Xu, S. Li, X. Zhou, W. Yan, T. Cheng, and D. Huang, "Dynamic neural networks based kinematic control for redundant manipulators with model uncertainties," *Neurocomputing*, vol. 329, pp. 255–266, 2019.

[7] S. Li, J. He, Y. Li, and M. U. Rafique, "Distributed recurrent neural networks for cooperative control of manipulators: a game-theoretic perspective," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 415–426, 2016.

[8] L. Jin, S. Li, X. Luo, Y. Li, and B. Qin, "Neural dynamics for cooperative control of redundant robot manipulators," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3812–3821, 2018.

[9] X. Li, Z. Xu, S. Li, H. Wu, and X. Zhou, "Cooperative kinematic control for multiple redundant manipulators under partially known information using recurrent neural network," *IEEE Access*, vol. 8, pp. 40029–40038, 2020.

[10] Z. Zhang, S. Chen, J. Xie, and S. Yang, "Two hybrid multi-objective motion planning schemes synthesized by recurrent neural networks for wheeled mobile robot manipulators," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, p. 1, 2019.

[11] D. Guo and Y. Zhang, "A new inequality-based obstacle-avoidance MVN scheme and its application to redundant robot manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1326–1340, 2012.

[12] Z. Xu, X. Zhou, and S. Li, "Deep recurrent neural networks based obstacle avoidance control for redundant manipulators," *Frontiers in Neurorobotics*, vol. 13, p. 47, 2019.

[13] L. Xiao and Y. Zhang, "Acceleration-level repetitive motion planning and its experimental verification on a six-link planar robot manipulator," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 906–914, 2013.

[14] Z. Zhang and Z. Yan, "Hybrid-level joint-drift-free scheme of redundant robot manipulators synthesized by a varying-parameter recurrent neural network," *IEEE Access*, vol. 6, pp. 34967–34975, 2018.

[15] Y. Zhang, S. S. Ge, and T. H. Lee, "A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 5, pp. 2126–2132, 2004.

[16] S. Liu and J. Wang, "Bi-criteria torque optimization of redundant manipulators based on a simplified dual neural network," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 2796–2801, Montreal, Canada, August 2005.

[17] J. Hollerbach and K. Ki Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 308–316, 1987.

[18] K. Kazerounian and A. Nedungadi, "Redundancy resolution of robotic manipulators at the acceleration level," in *Proceedings of the World Congress on Theory of Machines & Mechanisms (IFToMM)*, Sevilla, Spain, September 1987.

[19] W. S. Tang and J. Wang, "Two recurrent neural networks for local joint torque optimization of kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 30, no. 1, pp. 120–128, 2000.

[20] Y. Zhang and J. Wang, "A dual neural network for constrained joint torque optimization of kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 32, no. 5, pp. 654–662, 2002.

[21] Y. Zhang, N. Tan, and C. Lai, "Bi-criteria torque minimization of redundant robot arms with schemes, models and methods compared," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 2397–2402, Guilin, China, December 2009.

[22] Z. Xu, S. Li, X. Zhou, S. Zhou, and T. Cheng, "Dynamic neural networks for motion-force control of redundant manipulators: an optimization perspective," *IEEE Transactions on Industrial Electronics*, p. 1, 2020.

[23] X. Wu, J. Liu, C. Huang, M. Su, and T. Xu, "3-D path following of helical micro swimmers with an adaptive orientation compensation model," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 2, pp. 823–832, 2020.

[24] S. Li, S. Chen, B. Liu, Y. Li, and Y. Liang, "Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks," *Neurocomputing*, vol. 91, pp. 1–10, 2012.

[25] S. Li, H. Cui, Y. Li, B. Liu, and Y. Lou, "Decentralized control of collaborative redundant manipulators with partial command coverage via locally connected recurrent neural networks," *Neural Computing and Applications*, vol. 23, no. 3-4, pp. 1051–1060, 2013.

[26] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.

[27] H. Huang, C. Yang, and C. L. P. Chen, "Optimal robot-environment interaction under broad fuzzy neural adaptive control," *IEEE Transactions on Cybernetics*, p. 1, 2020.

[28] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4710–4720, 2017.

[29] T. Xu, J. Yu, C.-I. Vong, B. Wang, X. Wu, and L. Zhang, "Dynamic morphology and swimming properties of rotating miniature swimmers with soft tails," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 924–934, 2019.

[30] T. Xu, Y. Guan, J. Liu, and X. Wu, "Image-based visual servoing of helical microswimmers for planar path following," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 325–333, 2020.

[31] S. Li, Y. Lou, and B. Liu, "Bluetooth aided mobile phone localization: a nonlinear neural circuit approach," *ACM Transactions on Embedded Computing Systems*, vol. 13, no. 4, 2014.

[32] D. Guo, F. Xu, and L. Yan, "New pseudoinverse-based path-planning scheme with PID characteristic for redundant robot manipulators in the presence of noise," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 6, pp. 2008–2019, 2017.

[33] L. Jin, Y. Zhang, S. Li, and Y. Zhang, "Noise-tolerant ZNN models for solving time-varying zero-finding problems: a control-theoretic approach," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 992–997, 2017.

[34] D. Chen and Y. Zhang, "Robust zeroing neural-dynamics and its time-varying disturbances suppression model applied to mobile robot manipulators," *IEEE Transactions on Neural Networks & Learning Systems*, vol. 29, no. 9, pp. 1–13, 2017.

[35] D. Chen, S. Li, Q. Wu, and X. Luo, "New disturbance rejection constraint for redundant robot manipulators: an optimization perspective," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2221–2232, 2020.

[36] D. Guo and Y. Zhang, "Acceleration-level inequality-based MAN scheme for obstacle avoidance of redundant robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6903–6914, 2014.

[37] D. Guo, Q. Feng, and J. Cai, "Acceleration-level obstacle avoidance of redundant manipulators," *IEEE Access*, vol. 7, pp. 183040–183048, 2019.

[38] X. Li, J. Yu, S. Li, and L. Ni, "A nonlinear and noise-tolerant ZNN model solving for time-varying linear matrix equation," *Neurocomputing*, vol. 317, pp. 70–78, 2018.

[39] X. Li, J. Yu, S. Li, Z. Shao, and L. Ni, "A non-linear and noise-tolerant ZNN model and its application to static and time-varying matrix square root finding," *Neural Processing Letters*, vol. 50, pp. 1687–1703, 2018.

[40] L. Jin, Y. Zhang, and S. Li, "Integration-enhanced Zhang neural network for real-time-varying matrix inversion in the presence of various kinds of noises," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 12, pp. 2615–2627, 2016.

[41] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2243–2254, 2017.

[42] S. Li, Y. Guo, and B. Bingham, "Multi-robot cooperative control for monitoring and tracking dynamic plumes," in *Proceedings of the IEEE International Conference on Robotics & Automation (ICRA)*, Hong Kong, China, June 2014.

[43] S. Li and Y. Guo, "Distributed source seeking by cooperative robots: all-to-all and limited communications," in *Proceedings of the IEEE International Conference on Robotics & Automation (ICRA)*, pp. 1107–1112, Saint Paul, MN, USA, May 2012.