A Collaboration Scheme for Controlling Multi-Manipulator System: A Game-Theoretic Perspective

Jiazheng Zhang, *Student Member, IEEE*, Long Jin, *Senior Member, IEEE*, Yang Wang, *Senior Member, IEEE*, Chenguang Yang, *Senior Member, IEEE*

***Abstract*—In some task-oriented multi-manipulator applications, the system not only needs to complete the main assigned tasks, but also should optimize some sub-objectives. In order to tap the redundancy potential of individual manipulators and improve the performance of the system, a hybrid multi-objective optimization solution with robustness is proposed in accordance with the realistic execution requirements of the tasks. The entire control scheme is designed from the perspective of the Nash game and further refined into a problem to determine the Nash equilibrium point. Furthermore, a neural-network-assisted model is established to seek the best response of each manipulator to others. Theoretical analysis provides support for proving the convergence and robustness of the model. Finally, the feasibility of the control design is illustrated by simulation studies of the multi-manipulator system.**

***Index Terms*—Distributed control, redundancy resolution, hybrid multi-objective optimization, neural networks.**

I. Introduction

BENEFITING from the observation of group behavior in nature, it is found that collaborative behavior has an excellent economy. With the deepening of industrial intelligence, a single robot in operation is no longer advantageous [1]–[3]. In contrast, multi-robot systems have abilities to perform tasks more efficiently through communication and coordination among individuals [4] and have potential in such applications as environmental surveillance [5], underwater tasks [6], and robot machining [7]. Redundant manipulators are widely favored in intelligent production, given their well executable ability [8]–[10]. Although the structure with multiple degrees of freedom brings much convenience for them to perform tasks, it also substantially increases their control complexity [11], [12]. In recent years, a great deal of effort has been put into promoting the development of manipulator control technology [13]–[15]. Different from driving a single manipulator, controlling multiple manipulators to perform tasks in collaboration needs to consider the information interaction between each individual. The way of broadcast communication depends heavily on the performance of the central processing unit, which is not conducive to the development and evolution of the manipulator system, so it is not suitable for large-scale systems [16]–[18]. In this regard, a distributed design concept can be incorporated into the system construction. By allocating the communication and calculation burden to multiple microprocessors, the control benefits will be largely increased.

Performance metrics are important considerations for guiding the motion synthesis of manipulators, which can improve the operational performance of manipulators to a certain extent. Such as the dexterity index [19] reflecting the execution potential of a manipulator, and the kinematic conditioning index [20] showing the worst possible performance of a manipulator. The velocity and acceleration of the joint are typical indexes reflecting the motion characteristics of the manipulator in engineering [21]. In the process of task execution, manipulators are limited by the driving power and need to run slowly and steadily [22]. On the other hand, the manipulators need to reduce the acceleration to avoid position overshoot or oscillation and lower the output torque. In fact, excessive velocity or acceleration may aggravate the wear and tear of the hardware mechanism, which is not conducive to the production activities [23], [24]. Moreover, for tasks that need to move along a closed path in the workspace, the joint angle of the manipulator may not be able to return to the initial value after long-term operations. To this end, an effective approach is to minimize the Euclidean norm of joint velocity (i.e., minimum velocity norm, MVN) and joint acceleration (i.e., minimum acceleration norm, MAN) to optimize the output performance of the manipulators while considering a joint-drift-free (JDF) index to meet the demands of repetitive task execution.

One of the factors that make many models not applicable for online control is that they do not have the ability to solve problems in real time [25]–[28]. Although some control methods have been applied to the real-time tracking control of robots, they are essentially designed to solve static problems. Given that time-varying parameters are not static throughout the process, there are certain limitations when using these

This work was supported in part by the National Natural Science Foundation of China under Grant 62176109; in part by the CIE-Tencent Robotics X Rhino-Bird Focused Research Program under Grant 2021-01; in part by the Natural Science Foundation of Gansu Province under Grant 21JR7RA531; in part by the Chongqing Key Laboratory of Mobile Communications Technology under Grant cqupt-mct-202004; in part by the Science and Technology Project of Chengguan District of Lanzhou under Grant 2021JSCX0014; in part by Sichuan Science and Technology Program No.2022nsfsc0916; in part by the Supercomputing Center of Lanzhou University; and in part by the Fundamental Research Funds for the Central Universities under Grant lzujbky-2021-it35 and Grant lzujbky-2021-65. *(Corresponding author: Long Jin.)*

J. Zhang and L. Jin are with the School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China (e-mails: zhangjzh15@foxmail.com; jinlongsysu@foxmail.com).

Y. Wang is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400000, China (e-mail: wangyang@cqupt.edu.cn).

C. Yang is with Bristol Robotics Laboratory, University of the West of England, Bristol, BS16 1QY, UK (e-mail: cyang@ieee.org).

methods to deal with time-varying problems, such as delay errors, slow response, etc [29]. Once these parameters change rapidly and drastically, the error of the entire system may exceed expectations. Therefore, it is urgent to build a real-time and efficient solution algorithm [30]. In [31], a controller that integrates neural networks and fuzzy systems is developed to regulate the transient performance of hypersonic vehicles. The excellent approximation performance of the neural network provides the basis for the realization of the above functions. In [32], a computationally efficient neural network is designed and verified to have higher efficiency than traditional numerical methods, and also performs well in dealing with uncertainties and disturbances in control [33]. Given that the states and behaviors involved in the robot control are complex and it is difficult to filter each state and behavior, the neural-network-assisted solution algorithms are highly recommended [34]. For example, a random neural network control scheme is presented in [35], which effectively overcomes the error accumulation in the control of robots by introducing error feedback to the optimization target. In [36], a deep neural network-based method is adopted to accurately realize the pose tracking of the fish-like robot. Admittedly, there are more challenges in real-time control of multi-manipulator systems. Some attempts on multi-manipulator systems using neural networks have also achieved preliminary results. For instance, a dual neural network-based controller is developed to drive multiple robots to complete the assigned tasks in [37]. To optimize the joint velocity of the robot in operation, a neural network-guided control scheme is implemented to adjust the real-time state of each joint [38]. However, most of these schemes take the joint velocity as a decision variable, which is likely to produce a greater joint acceleration. Besides, noise suppression is also a factor worth considering.

This research attempts to incorporate the Nash game theory into the collaborative control design of a multi-manipulator system. The Nash game defines a scenario, in which each participant has limited access to information; Each participant's decision has impact on the other participant's payoffs; Each participant needs to make a favorable decision by guessing the intentions of others. Nash equilibrium is an important term in the game, which emphasizes that each participant cannot unilaterally tamper with its personal strategy to earn higher payoffs [39]. Based on this pioneering work of John Nash, many game-based strategies for distributed systems have been developed, such as providing fair channel resource allocation for vehicle networks [40], seeking Nash equilibrium for games with partial information [41], etc. These ideas motivate the present work. Compared with the previous work, the main contributions of this paper are as follows.

1) A collaborative control scheme for a multi-manipulator system is formulated from the game-theoretic perspective, which naturally restricts each manipulator to its own domain for strategy selection, thus facilitating the development of efficient distributed algorithms.
2) A hybrid multi-objective optimization index is designed for the motion synthesis of the manipulator, which effectively optimizes the joint motion of the manipulator and suppresses the occurrence of joint drift.
3) A computationally effective neural network model is established to ensure that each manipulator can make the best strategic response to other manipulators in time.

The rest of this paper is organized as follows. Section II develops the control laws of each manipulator inspired by game theory. A neural-network-assisted model is proposed in Section III for handling the proposed scheme. Theoretical analysis is provided in Section IV. Section V demonstrates the effectiveness of the proposed scheme by means of simulations and experiments. Section VI summarizes the paper.

II. Preliminaries and Problem Statement

This section introduces some basic principles of robot kinematics. On this basis, it further explores the collaborative control of multiple manipulators.

A. Robot Kinematics

Robot kinematics studies the conversion relation between the joints and the end-effector of the robots. Typically, a general model of the robot kinematics is expressed as

$$\dot{\mathbf{r}}(t) = J(\mathbf{q}(t))\dot{\mathbf{q}}(t), \tag{1}$$

of which $\dot{\mathbf{r}}(t) \in \mathbb{R}^m$ contains the velocity information of the end-effector; $\dot{\mathbf{q}}(t) \in \mathbb{R}^n$ contains the joint-velocity information of the manipulator; $J(\mathbf{q}(t)) \in \mathbb{R}^{m\times n}$ signifies the transmission ratio of the robot's kinematic velocity from joint space to operating space. For a well-defined manipulator, $J(\mathbf{q}(t))$ is usually known. However, when $m < n$, the solution satisfying (1) is not unique, so poor performance solutions need to be eliminated according to the requirements of the tasks.

Notice that the minimum norm solution is usually an appropriate choice among many feasible solutions for manipulator control, which represents a scheme option with minimal energy expenses. Typical indexes include MVN $\|\dot{\mathbf{q}}^\top(t)\dot{\mathbf{q}}(t)\|_2^2$, MAN $\|\ddot{\mathbf{q}}^\top(t)\ddot{\mathbf{q}}(t)\|_2^2$, etc. In addition, joint drift is a negative factor affecting the execution of the manipulator. For some repetitive motion tasks, the accumulation of joint drift errors will directly affect the operating accuracy. For this reason, a JDF index $\|\mathbf{q}(t)-\mathbf{q}_\mathrm{d}\|_2^2$ with $\mathbf{q}_\mathrm{d}$ representing the predetermined angle state is designed to overcome this phenomenon. By forcing the start and end states of the robot to tend to the predetermined configuration, the level of joint drift can be effectively alleviated. Besides, it is noted that (1) only provides the variable information of the velocity level, but the study of the pure velocity-level research is not enough to support the stability control of a manipulator. In this regard, by calculating the time derivative of (1), the kinematic relationship of the manipulator at the acceleration level is given as

$$\ddot{\mathbf{r}}(t) = \dot{J}(\mathbf{q}(t))\dot{\mathbf{q}}(t) + J(\mathbf{q}(t))\ddot{\mathbf{q}}(t), \tag{2}$$

with $\ddot{\mathbf{r}}(t) = \mathrm{d}\dot{\mathbf{r}}(t)/\mathrm{d}t \in \mathbb{R}^m$, $\ddot{\mathbf{q}}(t) = \mathrm{d}\dot{\mathbf{q}}(t)/\mathrm{d}t \in \mathbb{R}^n$, and $\dot{J}(\mathbf{q}(t)) = \partial J(\mathbf{q}(t))/\partial t \in \mathbb{R}^{m\times n}$. In the following context, $\mathbf{r}(t)$ is replaced by $\mathbf{r}$ for convenience. The detailed preparatory knowledge on robot kinematics can be referred to [42], [43].

Assumption 1: Manipulators do not easily fall into the singularity, i.e., the Jacobian matrix J is assumed to be full rank in this work. Research on avoiding the singularity of the multi-manipulator system can be referred to [17].

B. Collaborative Control and Trajectory Planning

The collaborative control of multiple manipulators focuses on how to enable each manipulator to reach a certain coordination goal according to the requirements of the tasks. Firstly, a translation transformation is performed for the motion trajectory of the manipulator, so as to facilitate the collaborative constraint design. Concretely, the end-effector trajectory of the ith manipulator after the translation operation is set to

$$\mathcal{A}_i(t)=\mathbf{r}_i-\hbar_{\mathsf{c}i}\in\mathbb{R}^m, \tag{3}$$

where $\hbar_{\mathsf{c}i}$ is the distance vector pointing from the end-effector i to the pre-determined trajectory $\mathbf{r}_\mathsf{d}(t)$ at the time instant t. The purpose of this operation is to transform the trajectories of manipulators adjacent to manipulator i into its reference coordinate system for easy comparison.

Trajectory planning of a manipulator is a practical application of the inverse kinematics solution. To put it simply, the vibration of a manipulator in operation can be effectively reduced, and its working life can be prolonged through reasonable planning of the changes of each joint in the case of preset end-effector trajectory. However, in the process of real-time inverse kinematics solution, there may be a small deviation between the actual generated trajectory and the preset trajectory. For that reason, a trajectory deviation feedback $\mathbf{r}-\mathbf{r}_\mathsf{d}$ of the end-effector is introduced to compensate for this deviation. Considering that, in a multi-manipulator system with the limited communication, a manipulator without connecting the command center takes the end-effector of the adjacent manipulator k as the tracking object, the trajectory deviation feedback of the ith manipulator is defined as

$$\epsilon_i(t)=\mathcal{A}_i(t)-\mathcal{A}_k(t). \tag{4}$$

For simplicity, $\mathcal{A}(t)$ is abbreviated to $\mathcal{A}$. To force $\mathcal{A}_i$ to be consistent with $\mathcal{A}_k$, (4) is further processed as

$$\dot{\epsilon}_i(t)=-\mu\epsilon_i(t)=-\mu(\mathcal{A}_i-\mathcal{A}_k), \tag{5}$$

of which $\mu\in\mathbb{R}^+$. Additionally, in order to further obtain the trajectory deviation feedback at the acceleration level, conducting a similar process to (5) derives

$$\dot{\varepsilon}_i(t)=-\rho\varepsilon_i(t)=-\rho(\dot{\epsilon}_i(t)+\mu(\mathcal{A}_i-\mathcal{A}_k)), \tag{6}$$

of which $\varepsilon_i(t)=\dot{\epsilon}_i(t)+\mu(\mathcal{A}_i-\mathcal{A}_k)$ with $\dot{\epsilon}_i(t)=\dot{\mathbf{r}}_i-\dot{\mathbf{r}}_k$, $\dot{\varepsilon}_i(t)=\mathrm{d}\dot{\epsilon}_i(t)/\mathrm{d}t+\mu(\dot{\mathbf{r}}_i-\dot{\mathbf{r}}_k)$ with $\mathrm{d}\dot{\epsilon}_i(t)/\mathrm{d}t=\ddot{\mathbf{r}}_i-\ddot{\mathbf{r}}_k$, and $\rho\in\mathbb{R}^+$. Based on (6), it is evident that

$$\ddot{\mathbf{r}}_i-\ddot{\mathbf{r}}_k=-(\rho+\mu)(\dot{\mathbf{r}}_i-\dot{\mathbf{r}}_k)-\rho\mu(\mathcal{A}_i-\mathcal{A}_k)=\mathcal{D}_\mathsf{e}, \tag{7}$$

which is the trajectory deviation feedback of the ith manipulator at the acceleration level.

Remark 1: In Eqs. (5) and (6), the design formula $\dot{\epsilon}(t)=-\mu\epsilon(t)$ is adopted to promote $\epsilon(t)$ and $\dot{\epsilon}(t)$ close to zero. Then the solution $\epsilon(t)=\epsilon(0)\exp(-\mu t)$ is obtained by solving $\dot{\epsilon}(t)=-\mu\epsilon(t)$, which implies that the error $\epsilon(t)$ would converge to zero exponentially. Referring to this guideline and assuming that $\epsilon(0)=0.1$ m and $\mu=70$, $0.1\exp(-70t)=0.05$ m is obtained within $\Delta t=0.01$ s. This indicates that the error converges over time to an acceptable value. The above statement provides a theoretical basis for the selection of design parameters.

C. Game-Theoretic Formulation of Collaboration Control

In a distributed network, each terminal node plays the role of both a receiver and a sender. For manipulators in the network, they strictly follow the established communication rules, so that the information is transferred reliably. Restricted by the power, bandwidth, and interaction distance of the communication equipment, any manipulator with disjoint communication areas cannot directly establish the connection on the information. Therefore, each manipulator can only guess or estimate the state of remaining manipulators based on the information obtained from neighboring manipulators, thus forming an information acquisition approach imitating game behavior. Here, each manipulator relies on local information exchange to form a private decision domain. Each decision domain exists independently but affects each other. It is worth pointing out that the means used to drive the manipulator to complete tasks efficiently is regarded as a strategy. Each manipulator discreetly adjusts its own strategy to increase performance gains, which is conducive to the execution of special required subtasks for the manipulators.

The velocity and acceleration of the joint are typical indexes reflecting the motion characteristics of the manipulator in engineering. On the one hand, due to the power limitations of the drive, the manipulator is expected to operate at a slow and steady velocity to ensure the smooth execution of the task. On the other hand, a large acceleration is likely to cause positional overshoot or oscillations in engineering, so minimizing acceleration is a reliable choice. Besides, in order to restrain the joint drift of the manipulator, a JDF index acting on the beginning and end of the task is introduced. Remarkably, these performance indexes can be used to reflect the gains and losses of the performance of manipulators in a game. Aiming at the collaboration problem of s manipulators in the framework of game theory, a control law of the ith manipulator is formulated as

payoff function:

$$-\frac{1}{2}\big(\gamma\,\|\dot{\mathbf{q}}_i\|_2^2+(1-\gamma)\,\|\ddot{\mathbf{q}}_i\|_2^2\big)+\Psi\big(\|\mathbf{q}-\mathbf{q}_\mathsf{d}\|_2^2\big) \tag{8a}$$

$$\text{default rules: } \mathcal{A}_i^{\mathsf{p/v}}=\frac{1}{\sum_{k\in N(i)}\mathcal{W}_{ik}}\sum_{k\in N(i)}\mathcal{W}_{ik}\mathcal{A}_k^{\mathsf{p/v}} \tag{8b}$$

$$\text{with } \mathcal{A}_k^{\mathsf{p}}=\begin{cases}\mathcal{A}_k, & \text{for } k=1,\ldots,s\\ \mathbf{r}_\mathsf{d}, & \text{for } k=0\end{cases} \tag{8c}$$

$$\mathcal{A}_k^{\mathsf{v}}=\begin{cases}J_k\dot{\mathbf{q}}_k, & \text{for } k=1,\ldots,s\\ \dot{\mathbf{r}}_\mathsf{d}, & \text{for } k=0\end{cases} \tag{8d}$$

$$\Psi(\mathtt{y})=\begin{cases}\mathtt{y}\cdot\big(1-2\mathtt{sin}(\pi t/\mathrm{T})\big), & t<\frac{\mathrm{T}}{6} \text{ or } t>\frac{5\mathrm{T}}{6}\\ 0, & \frac{\mathrm{T}}{6}\le t\le\frac{5\mathrm{T}}{6}\end{cases}$$

where operator $\{\cdot\}^{\mathsf{p/v}}$ distinguishes the choice between the displacement level $\mathtt{p}$ and velocity $\mathtt{v}$; $N(i)$ represents the neighborhood of the manipulator i; $\mathcal{W}_{ik} = 1$ indicates the connection between manipulator i and k is activated or 0 inactivated; T represents the execution period; $\gamma \in \mathbb{R}^+$ represents the weight coefficient of the objective index. Concretely, Formula (8a) provides a hybrid index that minimizes joint velocity and acceleration, and enables JDF index to function at the beginning and end of the task. It reflects the gains and losses of the manipulator in a game. Formula (8b) formulates a default rule to estimate the global information of the manipulator. Formula (8c) and Formula (8d) respectively store the displacement and velocity information of the manipulators connected with manipulator i, wherein, in the case of $k = 0$, the information obtained by manipulator i is consistent with the information released by the command center.

It is worth pointing out that the control law (8) is an effective action plan to drive the operation of manipulators, considering the performance gains and the information acquisition rule of the manipulator in task execution comprehensively. Every manipulator can be imagined as a player seeking profit maximization within the default rules, who is rational and does not act in a way that is detrimental to their own interests. The resulting strategy of each manipulator can be regarded as the best response to the strategies of other manipulators. In fact, game theory has proved that, in this limited information exchange situation, the information obtained by manipulators through estimation will eventually tend to be consistent, which is the most favorable choice for them.

III. Nash Equilibrium of The Game

This section aims to obtain the optimal strategy for guiding the collaborative operation of manipulators.

A. Redefinition of Performance Index and Constraints

To minimize $\|\dot{\mathbf{q}}_i\|_2^2$ at the acceleration level, a bold goal is to make $\dot{\mathbf{q}}_i$ as close to zero as possible. Noticeably, forcing $\ddot{\mathbf{q}}_i = -\varrho(\dot{\mathbf{q}}_i - 0)$ to be true has the effect of minimizing $\dot{\mathbf{q}}_i$. Therefore, the goal of minimizing $\|\dot{\mathbf{q}}_i\|_2^2$ can be redefined as

$$\text{minimizing}\, \|\dot{\mathbf{q}}_i\|_2^2 \triangleq \|\ddot{\mathbf{q}}_i + \varrho\dot{\mathbf{q}}_i\|_2^2 = \ddot{\mathbf{q}}_i^\top\ddot{\mathbf{q}}_i + 2\varrho\dot{\mathbf{q}}_i^\top\ddot{\mathbf{q}}_i + \varrho^2\dot{\mathbf{q}}_i^\top\dot{\mathbf{q}}_i,$$

with $\varrho \in \mathbb{R}^+$. Since the proposed scheme focuses on harvesting the feasible solution at the acceleration level, the decision variable is set to $\ddot{\mathbf{q}}_i$. Then, bring the above formula into the Formula (8a), we get

$$\frac{1}{2}\left(\gamma\|\dot{\mathbf{q}}_i\|_2^2 + (1-\gamma)\|\ddot{\mathbf{q}}_i\|_2^2\right) = \gamma\ddot{\mathbf{q}}_i^\top\ddot{\mathbf{q}}_i + \gamma\varrho\dot{\mathbf{q}}_i^\top\ddot{\mathbf{q}}_i + \varrho^2\dot{\mathbf{q}}_i^\top\dot{\mathbf{q}}_i/2,$$

with γ representing the weights of $\|\dot{\mathbf{q}}_i\|_2^2$. Additionally, the index $\Psi\big(\|\mathbf{q} - \mathbf{q}_\mathsf{d}\|_2^2\big)$ can be processed in a similar method, and then its index at the acceleration level is obtained as $\Psi\big(\mathbf{h}^\top\ddot{\mathbf{q}}_i\big)$ with $\mathbf{h} = (\rho+\mu)\dot{\mathbf{q}}_i + \rho\mu(\mathbf{q}_i - \mathbf{q}_\mathsf{d})$.

Beyond that, in order to yield the constraint equation at the acceleration level, a second-order system described by Laplace matrix is obtained by combining Formula (8b), Formula (8c), and Formula (8d), which is expressed as

$$(L+\mathcal{V})\otimes E_m(\mathcal{A}^\mathsf{p} + \mathcal{A}^\mathsf{v}) - (\mathcal{V}\otimes E_m)\big(\mathbf{1}_s \otimes (\mathbf{r}_\mathsf{d} + \dot{\mathbf{r}}_\mathsf{d})\big) = 0, \tag{9}$$

where $L = \operatorname{diag}(\mathcal{V}\mathbf{1}_s) - \mathcal{W} \in \mathbb{R}^{s\times s}$ is a Laplacian matrix; $E_m = \operatorname{diag}(1,\ldots,1) \in \mathbb{R}^{m\times m}$; $\mathcal{W} = [\mathcal{W}_{ik}] \in \mathbb{R}^{s\times s}$ is an adjacency matrix; $\mathcal{A}^\mathsf{p} = [\mathcal{A}_1;\ldots;\mathcal{A}_s] \in \mathbb{R}^{ms}$; $\mathcal{A}^\mathsf{v} = \check{J}\check{\dot{\mathbf{q}}} \in \mathbb{R}^{ms}$ with $\check{J} = \operatorname{diag}(J_1;\ldots;J_s)$, $\check{\dot{\mathbf{q}}} = [\dot{\mathbf{q}}_1,\ldots,\dot{\mathbf{q}}_s]$; $\mathcal{V} = \operatorname{diag}(v) \in \mathbb{R}^s$ with $v = [v_1;\ldots;v_s]$, where $v_i = 1$ if the manipulator i is connected to the virtual manipulator, otherwise $v_i = 0$. However, a material fact to consider is that attempting to directly control an open-loop system like (9) is an inefficient choice. On this account, Equation (9) is transformed into a closed-loop system described from the perspective of error, so as to obtain

$$\begin{aligned}&(L+\mathcal{V})\otimes E_m(\check{J}\check{\dot{\mathbf{q}}} + \check{\dot{J}}\check{\dot{\mathbf{q}}} + \check{J}\check{\ddot{\mathbf{q}}}) - (\mathcal{V}\otimes E_m)\big(\mathbf{1}_s\otimes(\dot{\mathbf{r}}_\mathsf{d} + \ddot{\mathbf{r}}_\mathsf{d})\big)\\ &= \varsigma\big((\mathcal{V}\otimes E_m)\big(\mathbf{1}_s\otimes(\mathbf{r}_\mathsf{d} + \dot{\mathbf{r}}_\mathsf{d})\big) - (L+\mathcal{V})\otimes E_m(\mathcal{A}^\mathsf{p} + \check{J}\check{\dot{\mathbf{q}}})\big),\end{aligned}$$

where $\check{\dot{J}} = \operatorname{diag}(\dot{J}_1;\ldots;\dot{J}_s) \in \mathbb{R}^{ms\times ns}$; $\check{\ddot{\mathbf{q}}} = [\ddot{\mathbf{q}}_1;\ldots;\ddot{\mathbf{q}}_s] \in \mathbb{R}^{ns}$; $\ddot{\mathbf{r}}_\mathsf{d}$ represents the acceleration instruction issued by the command center.

B. Real-Time Redundancy Resolution

Based on the redefined hybrid index, constraints and deviation feedback $\mathcal{D}_\mathrm{e}$, the control law (8) is conceived as the following optimization procedure:

$$\begin{aligned}\min\quad & \check{\ddot{\mathbf{q}}}^\top\check{\ddot{\mathbf{q}}}/2 + \gamma\varrho\check{\dot{\mathbf{q}}}^\top\check{\ddot{\mathbf{q}}} + \varrho^2\check{\dot{\mathbf{q}}}^\top\check{\dot{\mathbf{q}}}/2 + \omega\Psi\big(\check{\mathbf{h}}^\top\check{\ddot{\mathbf{q}}}\big)\\ \text{s.t.}\quad & (L+\mathcal{V})\otimes E_m(\check{J}\check{\dot{\mathbf{q}}} + \check{\dot{J}}\check{\dot{\mathbf{q}}} + \check{J}\check{\ddot{\mathbf{q}}} + \mathcal{D}_\mathrm{e}) - \o_\mathrm{m} = 0\\ \text{with}\quad & \o_\mathrm{m} = (\mathcal{V}\otimes E_m)\big(\mathbf{1}_s\otimes(\dot{\mathbf{r}}_\mathsf{d} + \ddot{\mathbf{r}}_\mathsf{d})\big)\\ &\qquad + \varsigma(L+\mathcal{V})\otimes E_m(\mathcal{A}^\mathsf{p} + \check{J}\check{\dot{\mathbf{q}}})\\ &\qquad - \varsigma(\mathcal{V}\otimes E_m)\big(\mathbf{1}_s\otimes(\mathbf{r}_\mathsf{d} + \dot{\mathbf{r}}_\mathsf{d})\big),\end{aligned}\tag{10}$$

of which $\check{\mathbf{h}} = (\rho+\mu)\check{\dot{\mathbf{q}}} + \rho\mu(\check{\mathbf{q}} - \mathbf{1}_s\otimes\mathbf{q}_\mathsf{d})$, $\check{\mathbf{q}} = [\mathbf{q}_1;\ldots;\mathbf{q}_s] \in \mathbb{R}^{ns}$, $\omega \in \mathbb{R}^+$. Remarkably, the solution satisfying (10) is the optimal strategy for driving the collaborative motion of manipulators. Then, Lagrange multiplier method [44] is adopted to further process (10), thus yielding

$$\begin{aligned}\mathcal{L}(\check{\ddot{\mathbf{q}}},\vartheta) =& \check{\ddot{\mathbf{q}}}^\top\check{\ddot{\mathbf{q}}}/2 + \gamma\varrho\check{\dot{\mathbf{q}}}^\top\check{\ddot{\mathbf{q}}} + \varrho^2\check{\dot{\mathbf{q}}}^\top\check{\dot{\mathbf{q}}}/2 + \omega\Psi\big(\check{\mathbf{h}}^\top\check{\ddot{\mathbf{q}}}\big)\\ &+ \vartheta^\top\big((L+\mathcal{V})\otimes E_m(\check{J}\check{\dot{\mathbf{q}}} + \check{\dot{J}}\check{\dot{\mathbf{q}}} + \check{J}\check{\ddot{\mathbf{q}}} + \mathcal{D}_\mathrm{e}) - \o_\mathrm{m}\big),\end{aligned}$$

where $\vartheta \in \mathbb{R}^{ms}$ is the Lagrange multiplier. Then, by setting $\partial\mathcal{L}(\check{\ddot{\mathbf{q}}},\vartheta)/\partial\check{\ddot{\mathbf{q}}} = 0$ and $\partial\mathcal{L}(\check{\ddot{\mathbf{q}}},\vartheta)/\partial\vartheta = 0$, one has

$$\begin{cases}\check{\ddot{\mathbf{q}}} + \gamma\varrho\check{\dot{\mathbf{q}}} + \omega\Psi\big(\check{\mathbf{h}}\big) + ((L+\mathcal{V})\otimes E_m\check{J})^\top\vartheta = 0\\ (L+\mathcal{V})\otimes E_m(\check{J}\check{\dot{\mathbf{q}}} + \check{\dot{J}}\check{\dot{\mathbf{q}}} + \check{J}\check{\ddot{\mathbf{q}}} + \mathcal{D}_\mathrm{e}) - \o_\mathrm{m} = 0.\end{cases}\tag{11}$$

Remarkably, the solution of (11) constitutes the optimal set of strategies for the manipulators, which is the equilibrium point of the Nash equilibrium. For the sake of description, (11) is rewritten as an equilibrium equation:

$$\Gamma(t)\zeta(t) = \Upsilon(t), \tag{12}$$

where

$$\Gamma(t) = \begin{bmatrix} E_{ns\times ns} & ((L+\mathcal{V})\otimes E_m\check{J})^\top\\ (L+\mathcal{V})\otimes E_m\check{J} & 0_{ms\times ms}\end{bmatrix},$$

$$\zeta(t) = \begin{bmatrix}\check{\ddot{\mathbf{q}}}\\ \vartheta\end{bmatrix}, \Upsilon(t) = \begin{bmatrix} -\gamma\varrho\check{\dot{\mathbf{q}}} - \omega\Psi\big(\check{\mathbf{h}}\big)\\ \o_\mathrm{m} - (L+\mathcal{V})\otimes E_m(\check{J}\check{\dot{\mathbf{q}}} + \check{\dot{J}}\check{\dot{\mathbf{q}}} + \mathcal{D}_\mathrm{e})\end{bmatrix}.$$

Then, the error function of (12) is set to $\mathbf{e}(t) = \Gamma(t)\zeta(t) - \Upsilon(t) \in \mathbb{R}^{(m+n)s}$. By pushing $\mathbf{e}(t)$ closer to zero, the entire system will move towards one of its Nash equilibriums to get the best response for each manipulator. To speed up this process, a neural-network-assisted formula is exploited for approximating the optimal strategies in (11), which is expressed as

$$\dot{\mathbf{e}}(t) = -\alpha\mathcal{F}_\Omega(\mathbf{e}(t)) - \beta\mathcal{F}_\Omega\big(\mathbf{e}(t) + \alpha\int_0^t \mathcal{F}_\Omega(\mathbf{e}(\nu))\mathrm{d}\nu\big), \quad (13)$$

where $\mathcal{F}_\Omega$ denotes the activation function, and $\alpha,\ \beta \in \mathbb{R}^+$. Furthermore, substituting the definition of error into (13) yields

$$\begin{aligned}\dot{\zeta}(t) =& \Gamma(t)^{-1}\bigg\{ - \dot{\Gamma}(t)\zeta(t) - \alpha\mathcal{F}_\Omega\big(\Gamma(t)\zeta(t) - \Upsilon(t)\big) \\ &+ \dot{\Upsilon}(t) - \beta\mathcal{F}_\Omega\bigg(\Gamma(t)\zeta(t) - \Upsilon(t) \\ &+ \alpha\int_0^t \mathcal{F}_\Omega\big(\Gamma(\nu)\zeta(\nu) - \Upsilon(\nu)\big)\mathrm{d}\nu\bigg)\bigg\},\end{aligned} \quad (14)$$

which is the proposed neural-network-assisted model for the manipulator system.

Remark 2: The activation function contributes to the convergence of the model to a certain extent. In general, the activation function is defined as $\mathcal{F}_\Omega(\theta) = \{\arg\ \min_{\Theta\in\Omega}\|\Theta - \theta\|, 0 \in \Omega_\imath\}$ where $\|\cdot\|$ represents the Frobenius norm, which is essentially a mapping of θ on set Ω.

IV. Theoretical Analysis

This section aims to theoretically verify that the strategy obtained through solution model (14) is the best response of each manipulator to the strategies of other manipulators.

A. Convergence Analysis

As far as model (14) is concerned, it is a typical interconnected system. In addition, we know that if the residual error $\mathbf{e}(t)$ approaches zero, the state variable $\zeta(t)$ will approach the equilibrium point, and the strategy $\check{\ddot{\mathbf{q}}}^*$ obtained at this point is the best response of each manipulator to others.

Theorem 1: The solution synthesized by scheme (10) and model (14) is the best response of each manipulator to others.

Proof: First, based on the existence of (13), an auxiliary variable $\chi(t)$ is designed as

$$\chi(t) = \mathbf{e}(t) + \alpha\int_0^t \mathcal{F}_\Omega(\mathbf{e}(\nu))\mathrm{d}\nu. \quad (15)$$

Then, take the time derivative of (15), and $\dot{\chi}(t) = \dot{\mathbf{e}}(t) + \alpha\mathcal{F}_\Omega(\mathbf{e}(t))$ can be found. Here, re-substitute (13) into $\dot{\chi}(t)$, and an equation for $\chi(t)$ is expressed as

$$\dot{\chi}(t) = -\beta\mathcal{F}_\Omega(\chi(t)). \quad (16)$$

Furthermore, a Lyapunov function candidate is selected as $\mathcal{L}(t) = \chi^\top(t)\chi(t)/2$. By calculating the time derivative of $\mathcal{L}(t)$ and combining it with (16), we have

$$\dot{\mathcal{L}}(t) = -\beta\chi^\top(t)\mathcal{F}_\Omega(\chi(t)). \quad (17)$$

Referring to the definition of $\mathcal{F}_\Omega$, one deduces

$$\|\chi(t) - \mathcal{F}_\Omega(\chi(t))\|_2^2 \leq \|\chi(t) - \Theta\|_2^2, \Theta \in \Omega.$$

Notice that $\|\chi(t) - \mathcal{F}_\Omega(\chi(t))\|_2^2 \leq \|\chi(t)\|_2^2$ at $\Theta = 0$, so $0 \leq \mathcal{F}_\Omega^\top(\chi(t))\mathcal{F}_\Omega(\chi(t)) \leq 2\chi^\top(t)\mathcal{F}_\Omega(\chi(t))$ is concluded. Then, by substituting the aforementioned inequality into (17), we can obtain

$$\dot{\mathcal{L}}(t) = -\beta\chi^\top(t)\mathcal{F}_\Omega(\chi(t)) \leq 0.$$

According to LaSalle's invariance principle [45], it can be determined that $\chi(t)$ converges to zero. Thus, (13) degenerates into $\dot{\mathbf{e}}(t) = -\alpha\mathcal{F}_\Omega(\mathbf{e}(t))$ over time. Since the discussions on $\dot{\mathbf{e}}(t) = -\alpha\mathcal{F}_\Omega(\mathbf{e}(t))$ are similar to the analysis on (16), it is omitted here. As a result, $\mathbf{e}(t)$ eventually converges to zero, which implies that the model (14) is able to assist (10) in finding the optimal state variable $\zeta^*(t)$ for the system to reach the equilibrium point. Meanwhile, the searched strategy $\check{\ddot{\mathbf{q}}}^*$ is the best response of each manipulator to others. ■

B. Robustness Analysis

In industrial production, noise is one of the disadvantageous factors affecting system stability. The disturbance attached to the solution model will cause the internal response of the system, and then induce the system to fluctuate. In this section, the effects of constant, linear and white noises on model (14) are considered.

Theorem 2: The model (14) subject to unknown additive constant noise $p \in \mathbb{R}^{ms+ns}$ is able to find the best response for each manipulator to others.

Proof: First, the $\imath$th subsystem ($i \in 1, 2, \ldots, ms + ns$) of the model (14) disturbed by additive constant noise p can be abbreviated as

$$\begin{aligned}\dot{\mathbf{e}}_i(t) =& -\alpha\mathcal{F}_\Omega(\mathbf{e}_i(t)) \\ &- \beta\mathcal{F}_\Omega\big(\mathbf{e}_i(t) + \alpha\int_0^t \mathcal{F}_\Omega(\mathbf{e}_i(\nu))\mathrm{d}\nu\big) + p_i.\end{aligned} \quad (18)$$

Referring to (15), an intermediate variable is defined as $\chi_i(t) = \mathbf{e}_i(t) + \alpha\int_0^t \mathcal{F}_\Omega(\mathbf{e}_i(\nu))\mathrm{d}\nu$, and its time derivative is calculated as $\dot{\chi}_i(t) = \dot{\mathbf{e}}_i(t) + \alpha\mathcal{F}_\Omega(\mathbf{e}_i(t))$. Then, using $\chi_i(t)$ and $\dot{\chi}_i(t)$ to simplify (18), the following equation is acquired:

$$\dot{\chi}_i(t) = -\beta\mathcal{F}_\Omega(\chi_i(t)) + p_i. \quad (19)$$

Here, to test the stability of the $\imath$th subsystem, a Lyapunov function candidate is selected as

$$\ell_i(t) = \big(\beta\mathcal{F}_\Omega(\chi_i(t)) - p_i\big)^2/2. \quad (20)$$

The time derivative of $\ell_i(t)$ is calculated as

$$\begin{aligned}\dot{\ell}_i(t) &= \beta\big(\beta\mathcal{F}_\Omega(\chi_i(t)) - p_i\big)\frac{\partial\mathcal{F}_\Omega(\chi_i(t))}{\partial\chi_i}\dot{\chi}_i(t) \\ &= -\beta\frac{\partial\mathcal{F}_\Omega(\chi_i(t))}{\partial\chi_i}\big(\beta\mathcal{F}_\Omega(\chi_i(t)) - p_i\big)^2.\end{aligned} \quad (21)$$

By the definition of $\mathcal{F}_\Omega(\cdot)$, we know that $\mathrm{d}\mathcal{F}_\Omega(\chi_i(t))/\mathrm{d}\chi_i$ is always a non-negative value, so $\dot{\ell}_i(t) \leq 0$. Assuming that time progresses to a node t_{final} representing the large enough period, it is clear that $\ell_i(t)$ will eventually converge to zero, i.e., $\lim_{t\to t_{\text{final}}}\beta\mathcal{F}_\Omega(\chi_i(t)) - p_i = 0$. According to

(19), $\lim_{t\to t_{\text{final}}}\dot{\chi}_i(t)=0$ can be obtained. Thus, (18) will eventually degenerate to $\dot{\mathbf{e}}_i(t)=-\alpha\digamma_{\Omega}(\mathbf{e}_i(t))$. At this point, the proof of the noise resistance of the subsystem is further transformed into the proof of the convergence of $-\alpha\digamma_{\Omega}(\mathbf{e}_i(t))$ with time. Note that the proof of $\dot{\mathbf{e}}_i(t)=-\alpha\digamma_{\Omega}(\mathbf{e}_i(t))$ is essentially similar to the analysis of (16), so it is omitted here. According to the aforementioned analysis, it can be deduced that $\mathbf{e}(t)$ finally converges to zero, implying that the model (14) is able to find the optimal state variable $\zeta^*(t)$ that enables the system to reach the equilibrium point under the additive constant noise. Meanwhile, the searched strategy $\check{\ddot{\mathbf{q}}}^*$ is the best response of each manipulator to others. ■

Theorem 3: The linearly activated model (14) subject to bounded linear time-varying noise $\upsilon(t)=(\mathbf{1}_{ms+ns}\otimes\iota)t\in\mathbb{R}^{ms+ns}$ with $t\in[0,t_{\text{final}}]$ is able to find the best response of each manipulator to others that makes the residual error converge to $\|\iota\|_2/(\alpha\beta)$.

Proof: First, the ith subsystem ($i\in 1,2,\ldots,ms+ns$) of the linearly activated model (14) affected by the bounded linear time-varying noise $\upsilon(t)=(\mathbf{1}_{ms+ns}\otimes\iota)t\in\mathbb{R}^{ms+ns}$ with $t\in[0,t_{\text{final}}]$ can be expressed as

$$\dot{\mathbf{e}}_i(t)=-(\alpha+\beta)\mathbf{e}_i(t)-\alpha\beta\int_0^t\mathbf{e}_i(\nu)\mathrm{d}\nu+\upsilon_i(t). \tag{22}$$

To further analyze the regulation process of the subsystem, the Laplace transform is performed on it:

$$s\mathbf{e}_i(s)-\mathbf{e}_i(0)=-(\alpha+\beta)\mathbf{e}_i(s)-\frac{\alpha\beta}{s}\mathbf{e}_i(s)+\frac{\upsilon_i(s)}{s^2}. \tag{23}$$

Then, utilizing the final value theorem gets

$$\begin{aligned}\lim_{t\to\infty}\mathbf{e}_i(t)&=\lim_{s\to 0}\mathbf{e}_i(s)\\&=\lim_{s\to 0}\frac{s^2\mathbf{e}_i(0)+\iota}{s^2+(\alpha+\beta)s+\alpha\beta}=\frac{\iota}{\alpha\beta}.\end{aligned} \tag{24}$$

From the aforementioned derivations we know that $\mathbf{e}_i(t)$ eventually converges to $\|\iota\|_2/(\alpha\beta)$. Note that by adjusting the parameters α and β, the error can be reduced to an acceptable range for performing the task. Under this execution accuracy, model (14) is able to find the optimal state variable $\zeta^*(t)$ that enables the system to reach the equilibrium point. Meanwhile, the searched strategy $\check{\ddot{\mathbf{q}}}^*$ is the best response of each manipulator to others. ■

Theorem 4: The linearly activated model (14) subject to bounded white noise $w(t)\in\mathbb{R}^{ms+ns}$ is able to find the best response of each manipulator to others so that the residual error converges to the upper bounds $2\sigma\sqrt{ms+ns}/|\alpha-\beta|$ for $\alpha\neq\beta$ and $\sigma\sqrt{ms+ns}(\kappa/\varkappa+1/\alpha)$ for $\alpha=\beta$.

Proof: The ith subsystem ($i\in 1,2,\ldots,ms+ns$) of the linearly activated model (14) disturbed by bounded white noise $w(t)$ can be abbreviated as

$$\dot{\mathbf{e}}_i(t)=-(\alpha+\beta)\mathbf{e}_i(t)-\alpha\beta\int_0^t\mathbf{e}_i(\nu)\mathrm{d}\nu+w_i(t). \tag{25}$$

Depending on the value of α and β, the solution of the subsystem (25) is discussed in the following possible cases:

1) For $\alpha\neq\beta$, the solution of (25) is

$$\begin{aligned}\mathbf{e}_i(t)=&\frac{\mathbf{e}_i(0)(\psi_1\exp(\psi_1 t)-\psi_2\exp(\psi_2 t))}{\psi_1-\psi_2}\\&+\frac{1}{\psi_1-\psi_2}\bigg(\int_0^t(\psi_1\exp(\psi_1(t-\nu))\\&-\psi_2\exp(\psi_2(t-\nu)))w_i(\nu)\mathrm{d}\nu\bigg),\end{aligned}$$

where $\psi_{1,2}=\big(-(\alpha+\beta)\pm\sqrt{(\alpha+\beta)^2-4\alpha\beta}\big)/2$. Next, based on the triangle inequality, there is

$$\begin{aligned}|\mathbf{e}_i(t)|\leq&\frac{|\mathbf{e}_i(0)(\psi_1\exp(\psi_1 t)-\psi_2\exp(\psi_2 t))|}{\psi_1-\psi_2}\\&+\frac{\int_0^t|\psi_1\exp(\psi_1(t-\nu))|\cdot|w_i(\nu)|\mathrm{d}\nu}{\psi_1-\psi_2}\\&+\frac{\int_0^t|\psi_2\exp(\psi_2(t-\nu))|\cdot|w_i(\nu)|\mathrm{d}\nu}{\psi_1-\psi_2}\\\leq&\frac{|\mathbf{e}_i(0)(\psi_1\exp(\psi_1 t)-\psi_2\exp(\psi_2 t))|}{\psi_1-\psi_2}\\&+\frac{2}{\psi_1-\psi_2}\max_{0\leq\nu\leq t}|w_i(\nu)|.\end{aligned}$$

Thus for $\alpha\neq\beta$, i.e., $\alpha>\beta$ or $\alpha<\beta$, there is $\psi_1-\psi_2=|\alpha-\beta|$. Therefore, there is

$$\lim_{t\to\infty}\sup\|\mathbf{e}_i(t)\|_2\leq\frac{2\sigma\sqrt{ms+ns}}{|\alpha-\beta|},$$

where $\sigma=\max_{1\leq i\leq ms+ns}\{\max_{0\leq\nu\leq t}|w_i(\nu)|\}$.

2) For $\alpha=\beta$, the solution of (25) is

$$\begin{aligned}\mathbf{e}_i(t)=&\mathbf{e}_i(0)\exp(\psi_1 t)(1+\psi_1 t)\\&+\int_0^t(t-\nu)\psi_1\exp(\psi_1(t-\nu))w_i(\nu)\mathrm{d}\nu\\&+\int_0^t\exp(\psi_1(t-\nu))w_i(\nu)\mathrm{d}\nu.\end{aligned}$$

Referring to Theorem 1 in [46], there exist $\kappa\in\mathbb{R}^+$ and $\varkappa\in\mathbb{R}^+$ such that $|\psi_1|t\exp(\psi_1 t)\leq\kappa\exp(-\varkappa t)$. Then, based on the triangle inequality, one can derive

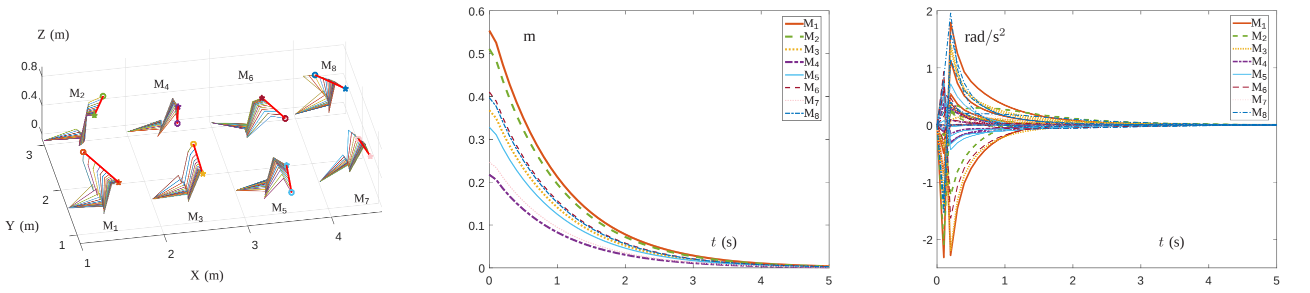
$$\begin{aligned}|\mathbf{e}_i(t)|\leq&|\mathbf{e}_i(0)\exp(\psi_1 t)(1+\psi_1 t)|\\&+\int_0^t|\kappa\exp(-\varkappa(t-\nu))|\cdot|w_i(\nu)|\mathrm{d}\nu\\&+\int_0^t|\exp(\psi_1(t-\nu))|\cdot|w_i(\nu)|\mathrm{d}\nu\\=&|\mathbf{e}_i(0)\exp(\psi_1 t)(1+\psi_1 t)|\\&+(\frac{\kappa}{\varkappa}-\frac{1}{\psi_1})\max_{0\leq\nu\leq t}|w_i(\nu)|.\end{aligned}$$

Thus for $\alpha=\beta$, $\psi_1=-\alpha$. Accordingly, there exists

$$\lim_{t\to\infty}\sup\|\mathbf{e}_i(t)\|_2\leq\sigma\sqrt{ms+ns}(\frac{\kappa}{\varkappa}+\frac{1}{\alpha}),$$

where $\sigma=\max_{1\leq i\leq ms+ns}\{\max_{0\leq\nu\leq t}|w_i(\nu)|\}$.

From the aforementioned derivations we know that the upper bound of $|\mathbf{e}(t)|$ is $2\sigma\sqrt{ms+ns}/|\alpha-\beta|$ for $\alpha\neq\beta$ or $\sigma\sqrt{ms+ns}(\kappa/\varkappa+1/\alpha)$ for $\alpha=\beta$ in the presence of white noise w. By selecting the appropriate parameters α and



(a)

(b)

(c)

Fig. 1. Simulation results of eight manipulators moving to preset points directed by scheme (10) assisted by model (14). (a) The movement process of manipulators. (b) Real-time distance between end-effector and preset point. (c) Joint acceleration.

β, the error is reduced to an acceptable range for performing the task. The strategy $\check{\ddot{\mathbf{q}}}^*$ searched with this error precision is the best response of each manipulator to others. ■

Remark 3: The collaborative control problem of the multi-manipulator system is reformulated using quadratic programming techniques in this paper. In this way, the problem is evolved to solve a convex quadratic programming problem so that the analytic solution can be obtained relatively easily, and the solution is the global optimal solution.

Remark 4: In this paper, the proposed scheme relies on the accurate manipulator model to obtain the good operation effect. However, there are some limitations in systems with unknown structures. Inspired by [47], the fuzzy logic system can be introduced to further improve the robustness of the system, which will be a beneficial attempt.

V. Simulations and Experiments

In this section, a group of robots following control law (8) is driven to execute the trajectory planning task. According to the operation characteristics of the manipulator, the task is further subdivided into point-to-point motion and continuous-path motion. Then, simulations are performed to verify the operation effect of manipulators in these two task scenarios.

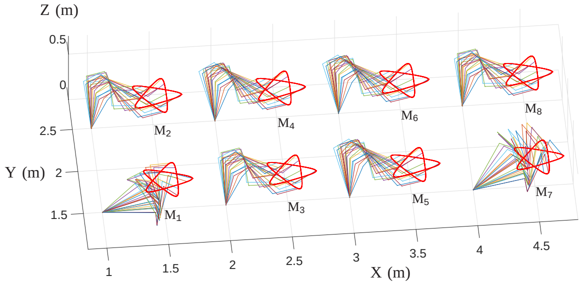


Fig. 2. The entire process of eight manipulators performing the pentagram curve motion task directed by scheme (10) assisted by model (14).

A. Point-to-Point Motion task

Point-to-point motion only requires the manipulator to move toward a preset point without considering what path it is running on. One typical application scenario is that manipulators should actively adjust their end-effectors to a specified initial position before executing the task, which can be treated as a preprocessing operation. Besides, imagine an extreme case: when the manipulator appears illegal configuration in the execution of the task, it needs to go back to a specific configuration for further overhaul. In that sense, initializing the manipulators to a fixed configuration is of great significance for their control. With these factors in mind, eight randomly initialized manipulators are assigned to perform a point-to-point motion task to test the validity of the scheme (10) assisted by solution model (14).

In the simulation of the task, the initial conditions are set as $\gamma = 0.5$, $\varrho = 100$, $\alpha = \beta = \varsigma = 25$, $\omega = \rho = \mu = 0$, and the operation time is allocated to 5 s. Manipulators M$_1$ and M$_6$ are set to connect to the virtual manipulator $\mathtt{0}$, thus $v_1 = v_6 = 1$. Besides, the manipulator obeys the rule that information can be exchanged between adjacent nodes, i.e., $\mathcal{W}_{ik} = 1$ only if $|i-k| \leq 1$. Based on the above settings, the simulation results of the point-to-point motion task performed by the manipulators are shown in Fig. 1. Specifically, Fig. 1(a) is a record of the point-to-point motion task performed by the randomly initialized manipulators during the period, in which the pentagonal star marks the position of the preset points and the initial position of the end-effectors is marked with a circle. Subsequently, the real-time distance $\|(x_i, y_i, z_i) - (x_{\mathtt{d}}, y_{\mathtt{d}}, z_{\mathtt{d}})\|_2$ of each end-effector is revealed in Fig. 1(b). Clearly, the distance between the end-effector and the preset point is constantly decreasing, which implies that the point-to-point motion task is performed smoothly. Furthermore, Fig. 1(c) records the variation of the joint-acceleration variables of each manipulator. The above simulation results indicate that the manipulators are able to be driven to the desired position points under the action of scheme (10) assisted by solution model (14).

B. Continuous-Path Motion Task in Noisy Environment

In this part, a continuous-path motion task is assigned to the manipulator system and executed in a noisy environment. The information exchange rules between manipulators are the same as those set in Sec. V-A, i.e., when $|i-k| \leq 1$, $\mathcal{W}_{ik} = 1$; the amplitude of additive noise is set as t; the motion of the end-effector is prescribed by a pentagram curve $\mathbf{r}_{\mathtt{d}} = [3\cos(\pi t/5)/20 + 3\cos(3\pi t/10)/50 +$

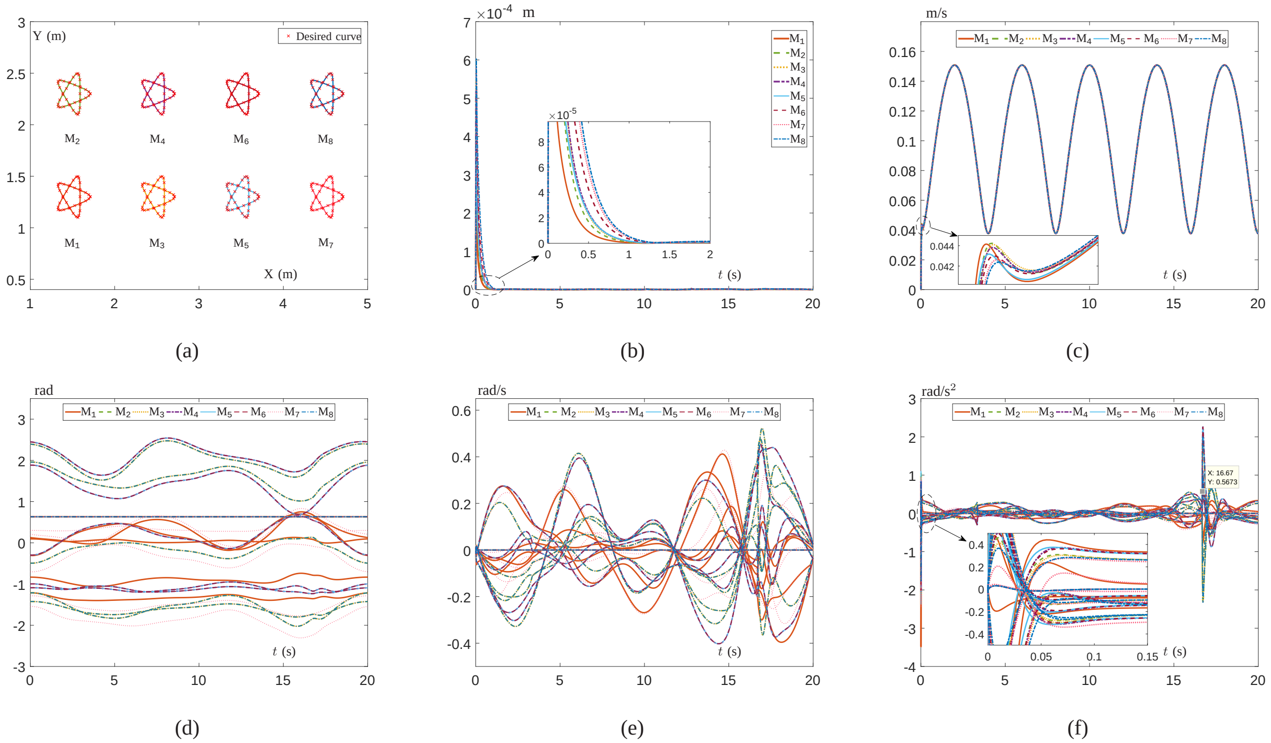


Fig. 3. Simulation results of eight manipulators tracking the pentagram curve directed by scheme (10) assisted by model (14). (a) The actual trajectories plotted in X-Y plane. (b) Tracking errors. (c) Speed of the end-effector. (d) Joint angle. (e) Joint velocity. (f) Joint acceleration.

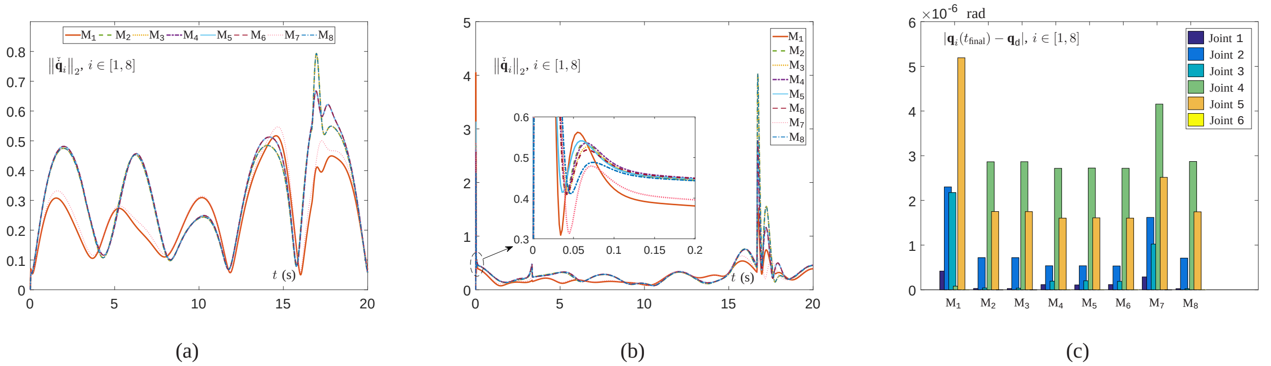
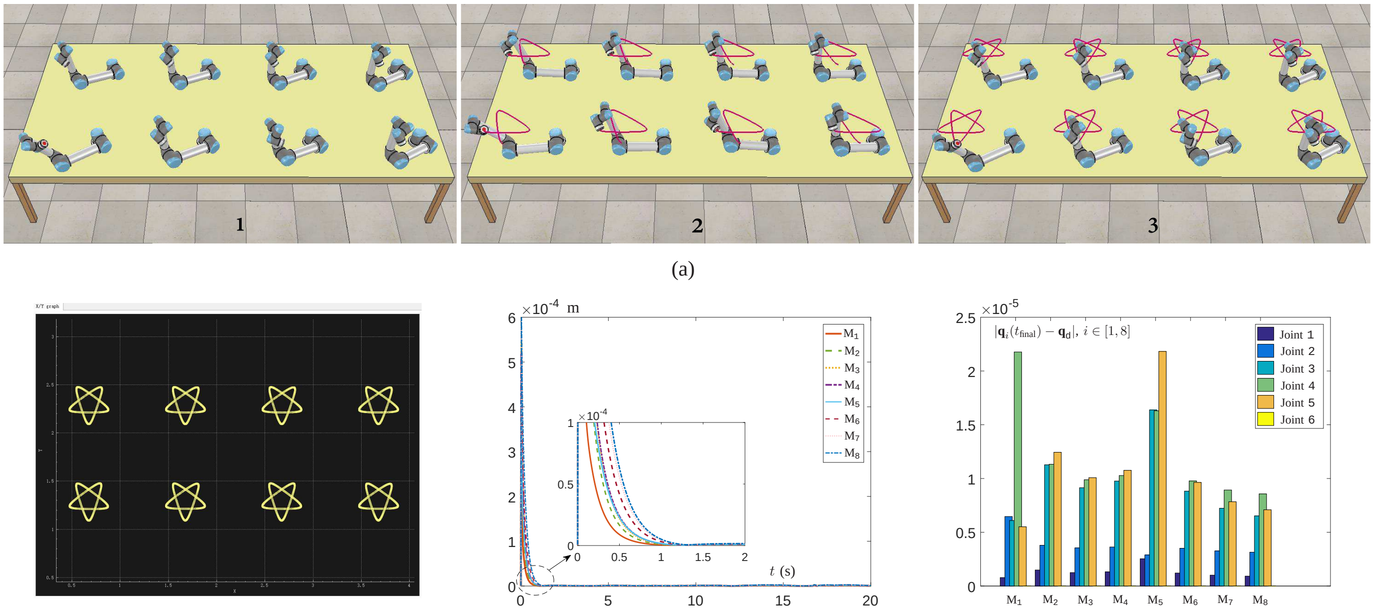


Fig. 4. Statistics on the performance index of eight manipulators directed by scheme (10) assisted by model (14) during the task execution. (a) Profiles of $\|\check{\dot{\mathbf{q}}}_i\|_2$. (b) Profiles of $\|\check{\ddot{\mathbf{q}}}_i\|_2$. (c) Joint drift errors $|\mathbf{q}_i(t_{\text{final}}) - \mathbf{q}_\text{d}|$.

$1/2;\ \ 3\sin(3\pi t/10)/50 - 3\sin(\pi t/5)/20 - 1/5;\ \ 1/2]$; the task period is set to 20 s. Additionally, the initial conditions used for this simulation are set to $\gamma = 0.5$, $\varrho = 100$, $\alpha = \beta = \varsigma = 70$, $\omega = 0.2$, and $\rho = \mu = 70$, and the simulation results obtained are shown as below.

Figure 2 records the entire process of the manipulators performing the continuous-path motion task. Judging from the execution results, the task is well executed. Furthermore, Fig. 3 records the joint changes of each manipulator and the tracking states of each end-effector. Specifically, the motion trajectories of each end-effector on the X-Y plane are drawn in Fig. 3(a), where the desired curves are marked with a cross-lines cursor. As the results show, the end-effectors perform the task smoothly along the desired curve. Next, Fig. 3(b) shows the execution error $\xi_i = \sqrt{(\mathbf{r}_i - \mathbf{r}_\text{d})^\top(\mathbf{r}_i - \mathbf{r}_\text{d})}$ of each end-effector performing the continuous-path motion task. The results reveal that $\xi = [\xi_1; \ldots; \xi_s] \in \mathbb{R}^{ms}$ is always kept within a tiny range throughout the task duration, which is completely in line with the actual task requirements. Subsequently, the changes in the velocity of each end-effector are recorded in Fig. 3(c), which shows that all end-effectors are running at a consistent speed. The changes of joint-angle, joint-velocity, and joint-acceleration of the manipulator are recorded in Fig. 3(d)-(f). However, it is noted from Fig. 3(f) that there is a small spike in the joint-acceleration of the manipulator at 0 s and 16.67 s of the task, which is caused by the action of the JDF index at these two instants. But soon, this peak is eliminated under the combined action of the minimum velocity and acceleration indexes. To intuitively reflect the performance of the manipulators during operation, the values of $\|\check{\dot{\mathbf{q}}}_i\|_2$ and $\|\check{\ddot{\mathbf{q}}}_i\|_2$ are measured and shown in Fig. 4(a)-(b). Besides, Fig. 4(c) reveals the joint drift



(a)

(b)

(c)

(d)

Fig. 5. Experiment of UR5 manipulators tracking the pentagram curve directed by scheme (10) assisted by model (14). (a) Experimental snapshots. (b) Tracking trajectories in the X-Y plane. (c) Profiles of tracking error. (d) Joint drift errors $|\mathbf{q}_i(t_{\text{final}}) - \mathbf{q}_\text{d}|$.

TABLE I

Comparisons Among Various Control Schemes for Manipulators

	Number of manipulators	Network topology	Position error compensation	Disturbances rejection	Performance index	Experimental verification	Mathematical formulation
This paper	Multiple	Distributed	Y[a]	Y[b]	Hybrid index[c]	Y[d]	Game theory
Paper [1]	Two	Centralized	N/A	N	N/A	N	Dynamic programming
Paper [8]	Single	N/A	N	Y[b]	JDF index	Y[d]	Optimization
Paper [14]	Single	N/A	Y[a]	N	JDF index	N	Optimization
Paper [16]	Multiple	Distributed	N/A	Y[b]	Manipulability index	Y[d]	Game theory
Paper [17]	Multiple	Distributed	N	Y[b]	Manipulability index	Y[d]	Optimization
Paper [24]	Single	Distributed	Y[a]	N	Hybrid index[c]	Y[d]	Optimization
Paper [30]	Two	Centralized	N	N	MVN index	Y[d]	Optimization
Paper [37]	Multiple	Distributed	N	N	MVN index	N	Game theory
Paper [38]	Multiple	N/A	N	Y[b]	MVN index	N	Optimization

[a] The scheme introduces a trajectory deviation feedback to improve the tracking accuracy.
[b] The model is capable of resisting disturbance.
[c] The scheme takes into account a hybrid performance index.
[d] The model is validated on a virtual robot experimental platform.

errors $|\mathbf{q}_i(t_{\text{final}}) - \mathbf{q}_\text{d}|$ of each manipulator after completing the task. The aforementioned statistical results show that the manipulator system operated according to scheme (10) runs smoothly, and the state of the manipulators can be restored to the original configuration after the completion of the task. Furthermore, the comparisons between the scheme (10) solved by (14) and various typical redundancy resolution schemes are listed in Table I. It can be seen that the proposed scheme (10) adopts a hybrid performance index, so it has certain advantages in terms of control effect. Additionally, the proposed model (14) performs well on fault tolerance in the case of disturbance, so it has bright prospects in industrial applications.

C. *Validation on UR5 Robots in CoppeliaSim*

In order to make the neural-network-assisted model (14) running in the simulation environment also guide the manipulators to work in the real environment, firstly, the CoppeliaSim experimentation platform is used to bridge the gap between simulation and reality. The UR5 manipulator in the model library is selected as the experimental object, and the parameters and tasks are leveraged identical to those in Sec. V-B. On this basis, the operating states of the manipulators implanted with scheme (10) assisted by model (14) are shown in Fig. 5(a). Here, snapshot 1 records the initial postures of the manipulators prior to the task; Snapshot 2 captures an instant of the manipulators during the task execution; Snapshot 3 shows the completion of the entire task. Visibly, the manipulators with the aforementioned algorithm can accomplish the task well. Furthermore, the performance parameters of UR5 manipulators performing tasks during this period are recorded in Fig. 5(b)-(d). Specifically, Fig. 5(b) records the task completion of UR5 manipulators in the X-Y plane. Subsequently, the tracking errors of end-effectors are reported in Fig. 5(c). The results show that the tracking status of each end-effector is well

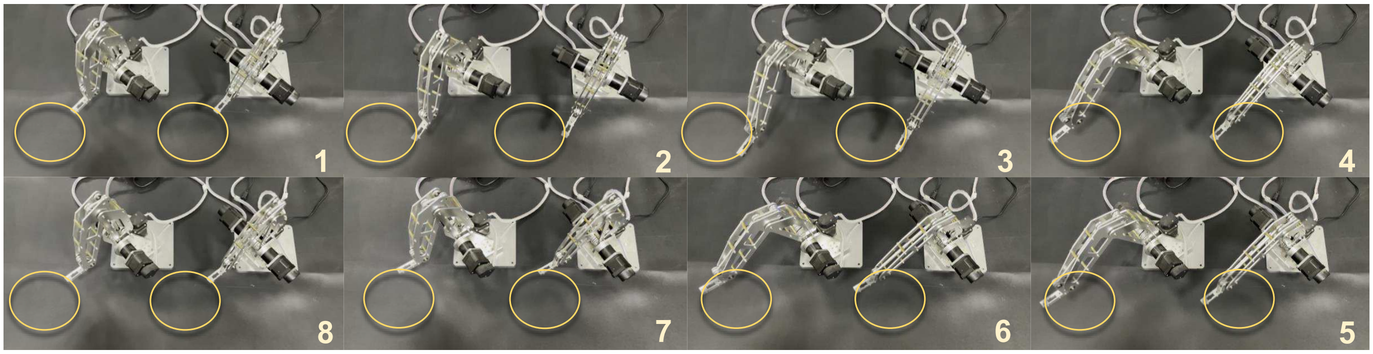


Fig. 6. Experimental snapshots of a three-axis robotic system tracks the circular curve.

during the task execution. Finally, the joint drift errors of each end-effector are counted in Fig. 5(d). The data reveal that the joint drift of the manipulators is significantly curbed at the end of the task. Through the aforementioned simulative experiments on CoppeliaSim, the feasibility of the scheme (10) solved by model (14) on the manipulators is verified. Furthermore, an experiment on a three-axis robotic system with D-H parameters listed in Table II is conducted to further demonstrate the effectiveness of the proposed model (14). The motion of the end-effector is prescribed by a circular curve $\mathbf{r}_{\mathrm{d}} = [\sin(\pi t/5)/10 + 1/10;\ \cos(\pi t/5)/10 - 1/10;\ 2/25]$, and the task period is set to 10 s. Serial port communication is adopted between manipulators, and other parameters are the same as in Section V-B. The corresponding experimental results are shown in Fig. 6.

TABLE II
D-H Parameters of a Three-Axis Robot

Link ι	$\alpha_{\iota-1}$ (rad)	$a_{\iota-1}$ (m)	d_ι (m)	Joint-Angle q_ι
1	0	0	0.097	q_1
2	$-\pi/2$	0	0	q_2
3	0	0.140	0	q_3
4	0	0.160	0	q_4

VI. Conclusions

In this paper, a hybrid multi-objective optimization solution with robustness has been developed. This solution has not only completed the main tasks assigned, but also optimized the joint velocity and acceleration of the manipulators, and effectively curbed the occurrence of joint drift. The entire control scheme has been designed from the perspective of the Nash game, and further refined into the problem of determining the Nash equilibrium point. On this basis, a neural-network-assisted model has been established to seek the best response of each manipulator to others. Theoretical analysis has provided the basis for the convergence and robustness of the model. Finally, simulations on the manipulator system have revealed the feasibility of the proposed control scheme, and the tests on the CoppeliaSim experimentation platform have implied that the scheme has the desirable engineering application potential. In the next research, an event-triggered mechanism attempts to be developed to reduce the frequent computational cost. Secondly, an attempt will be made to enhance the robustness of the system by combining the fuzzy logic system in the designed model.

References

[1] D. Kaserer, H. Gattringer, and A. Müller, “Time optimal motion planning and admittance control for cooperative grasping,” *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2216–2223, Apr. 2020.

[2] L. Jin, X. Zheng, and X. Luo, “Neural dynamics for distributed collaborative control of manipulators with time delays,” *IEEE/CAA J. Automatica Sinica*, to be published, doi: 10.1109/JAS.2022.105527.

[3] M. Liu, X. Zhang, M. Shang, and L. Jin, “Gradient-based differential $k$wta network with application to competitive coordination of multiple robots,” *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 8, pp. 1452–1463, Aug. 2022.

[4] V. P. Tran, F. Santoso, M. A. Garratt, and I. R. Petersen, “Distributed formation control using fuzzy self-tuning of strictly negative imaginary consensus controllers in aerial robotics,” *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 5, pp. 2306–2315, Oct. 2021.

[5] L. Huang, M. Zhou, K. Hao, and E. Hou, “A survey of multi-robot regular and adversarial patrolling,” *IEEE/CAA J. Automatica Sinica*, vol. 6, no. 4, pp. 894–903, Jul. 2019.

[6] Z. Zhou, J. Liu, and J. Yu, “A survey of underwater multi-robot systems,” *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 1, pp. 1–18, Jan. 2022.

[7] X. Zhao, B. Tao, and H. Ding, “Multimobile robot cluster system for robot machining of large-scale workpieces,” *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 1, pp. 561–571, Feb. 2022.

[8] W. Li, “Predefined-time convergent neural solution to cyclical motion planning of redundant robots under physical constraints,” *IEEE Trans. Ind. Electron.*, vol. 67, no. 12, pp. 10732–10743, Dec. 2020.

[9] J. Kim, W. Jie, H. Kim, and M. C. Lee, “Modified configuration control with potential field for inverse kinematic solution of redundant manipulator,” *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 4, pp. 1782–1790, Aug. 2021.

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

[11] M. Faroni, M. Beschi, N. Pedrocchi, and A. Visioli, “Predictive inverse kinematics for redundant manipulators with task scaling and kinematic constraints,” *IEEE Trans. Robot.*, vol. 35, no. 1, pp. 278–285, Feb. 2019.

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

[13] P. Hsu, “Coordinated control of multiple manipulator systems,” *IEEE Trans. Robotic. Autom.*, vol. 9, no. 4, pp. 400–410, Aug. 1993.

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

[15] T. Liu, W. Xu, T. Yang, and Y. Li, “A hybrid active and passive cable-driven segmented redundant manipulator: Design, kinematics, and planning,” *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 2, pp. 930–942, Apr. 2021.

[16] J. Zhang, L. Jin, and L. Cheng, “RNN for perturbed manipulability optimization of manipulators based on a distributed scheme: A game-theoretic perspective,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5116–5126, Dec. 2020.

[17] L. Jin, J. Zhang, X. Luo, M. Liu, S. Li, L. Xiao, and Z. Yang, "Perturbed manipulability optimization in a distributed network of redundant robots," *IEEE Trans. Ind. Electron.*, vol. 68, no. 8, pp. 7209–7220, Aug. 2021.

[18] Y. Zhang, S. Li, and J. Weng, "Distributed estimation of algebraic connectivity," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 3047–3056, May 2022.

[19] A. Y. Kumar and K. J. Waldron, "The workspaces of a mechanical manipulator," *J. Mech. Des.*, vol. 103, no. 3, pp. 665–672, Jul. 1981.

[20] J. Angeles and C. S. López-Cajún, "Kinematic isotropy and the conditioning index of serial robotic manipulators," *Int. J. Robot. Res.*, vol. 11, pp. 560–571, Nov. 1992.

[21] Z. Fu, J. Pan, E. Spyrakos-Papastavridis, Y. -H. Lin, X. Zhou, X. Chen, and J. S. Dai, "A Lie-Theory-Based dynamic parameter identification methodology for serial manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 5, pp. 2688–2699, Oct. 2021.

[22] Y. Zhang and Z. Zhang, *Repetitive Motion Planning and Control of Redundant Robot Manipulators*. New York, NY, USA: Springer-Verlag, 2013.

[23] D. Chen and Y. Zhang, "A hybrid multi-objective scheme applied to redundant robot manipulators," *IEEE Trans. Control Syst. Tech.*, vol. 14, no. 3, pp. 1337–1350, Jul. 2017.

[24] D. Chen, S. Li, W. Li, and Q. Wu, "A multi-level simultaneous minimization scheme applied to jerk-bounded redundant robot manipulators," *IEEE Trans. Control Syst. Tech.*, vol. 17, no. 1, pp. 463–474, Jan. 2020.

[25] J. Zhang, L. Jin, and C. Yang, "Distributed cooperative kinematic control of multiple robotic manipulators with improved communication efficiency," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 1, pp. 149–158, Feb. 2022.

[26] Y. Liu, P. Huang, F. Zhang, and Y. Zhao, "Distributed formation control using artificial potentials and neural network for constrained multiagent systems," *IEEE Trans. Control Syst. Tech.*, vol. 28, no. 2, pp. 697–704, Mar. 2020.

[27] M. Liu, S. B. Li, and L. Jin, "Modeling and analysis of matthew effect under switching social networks via distributed competition," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 0, pp. 1–5, Feb. 2022.

[28] L. Huang, Y. Ding, M. Zhou, Y. Jin, and K. Hao, "Multiple-solution optimization strategy for multirobot task allocation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 11, pp. 4283–4294, Nov. 2020.

[29] J. Li, Y. Zhang, S. Li, and M. Mao, "New discretization-formula-based zeroing dynamics for real-time tracking control of serial and parallel manipulators," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3416–3425, Aug. 2018.

[30] Z. Zhang, L. Zheng, Z. Chen, L. Kong, and H. R. Karimi, "Mutual-collision-avoidance scheme synthesized by neural networks for dual redundant robot manipulators executing cooperative tasks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 3, pp. 1052–1066, Mar. 2021.

[31] X. Bu, Y. Xiao, and H. Lei, "An adaptive critic design-based fuzzy neural controller for hypersonic vehicles: Predefined behavioral nonaffine control," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 4, pp. 1871–1881, Aug. 2019.

[32] Z. Zhang, S. Chen, X. Deng, and J. Liang, "A circadian rhythms neural network for solving the redundant robot manipulators tracking problem perturbed by periodic noise," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 6, pp. 3232–3242, Dec. 2021.

[33] Y. Liu, Z. Li, H. Su, and C. Y. Su, "Whole-body control of an autonomous mobile manipulator using series elastic actuators," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 2, pp. 657–667, Apr. 2021.

[34] J. Na, S. Wang, Y.-J. Liu, Y. Huang, and X. Ren, "Finite-time convergence adaptive neural network control for nonlinear servo systems," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2568–2579, Jun. 2020.

[35] Y. Li, S. Li, and B. Hannaford, "A model-based recurrent neural network with randomness for efficient control with applications," *IEEE Trans. Ind. Inf.*, vol. 15, no. 4, pp. 2054–2063, Apr. 2019.

[36] T. Zhang, J. Xiao, L. Li, C. Wang, and G. Xie, "Toward coordination control of multiple fish-like robots: Real-time vision-based pose estimation and tracking via deep neural networks," *IEEE/CAA J. Automatica Sinica*, vol. 8, no. 12, pp. 1964–1976, Dec. 2021.

[37] S. Li, J. He, Y. Li, and M. U. Rafique, "Distributed recurrent neural networks for cooperative control of manipulators: A game-theoretic perspective," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 415–426, Feb. 2017.

[38] L. Jin, S. Li, L. Xiao, R. Lu, and B. Liao, "Cooperative motion generation in a distributed network of redundant robot manipulators with noises," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 325, no. 10, pp. 1715–1724, Oct. 2018.

[39] J. Nash, "Equilibrium points in N-person games," *Proc. Nat. Acad. Sci. USA*, vol. 36, no. 1, pp. 48–49, 1950.

[40] F. Zhang, M. Zhou, L. Qi, Y. Du, and H. Sun, "A game theoretic approach for distributed and coordinated channel access control in cooperative vehicle safety systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2297–2309, Jun. 2020.

[41] M. Ye, D. Li, Q.-L. Han, and L. Ding, "Distributed Nash equilibrium seeking for general networked games with bounded disturbances," *IEEE/CAA J. Automatica Sinica*, to be published, doi: 10.1109/JAS.2022.105428.

[42] J. Baillieul, "Avoiding obstacles and resolving kinematic redundancy," in *Proc. IEEE Int. Conf. Robot. Automat.*, San Francisco, CA, 1986, pp. 1698–1704.

[43] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *J. Intell. Robot. Syst.*, vol. 3, pp. 201–212, 1990.

[44] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[45] D. Cheng, J. Wang, and X. Hu, "An extension of LaSalle's invariance principle and its application to multi-agent consensus," *IEEE Trans. Autom. Control*, vol. 53, no. 7, pp. 1765–1770, Aug. 2008.

[46] Z. Zhang and Y. Zhang, "Design and experimentation of accelerationlevel drift-free scheme aided by two recurrent neural networks," *IET Control Theory Appl.*, vol. 7, no. 1, pp. 25–42, Jan. 2013.

[47] X. Bu and Q. Qi, "Fuzzy optimal tracking control of hypersonic flight vehicles via single-network adaptive critic design," *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 1, pp. 270–278, Jan. 2022.

Jiazheng Zhang (Student Member, IEEE) received the B.E. degree in communication engineering from Lanzhou University, Lanzhou, China, in 2019, where he is currently pursuing the Ph.D. degree in computer application technology.

His main research interests include neural networks and multi-robot coordination.

Long Jin (Senior Member, IEEE) received the B.E. degree in automation and the Ph.D. degree in information and communication engineering from Sun Yat-sen University, Guangzhou, China, in 2011 and 2016, respectively. He received postdoctoral training at the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, from 2016 to 2017. He received the excellent doctoral dissertation award of Chinese Association for Artificial Intelligence (CAAI).

His current research interests include neural networks, robotics, optimization, and intelligent computing.

Yang Wang (Senior Member, IEEE) received the M.S. and Ph.D. degrees from The University of Sheffield, U.K., in 2011 and 2015, respectively. He joined the School of Communications and Information Engineering, Chongqing University of Posts and Telecommunications (CQUPT), in 2015. He is currently an Associate Professor with the Communication Department, CQUPT. His research interests include antennas and propagation, radar signature management, phase-modulating microwave structures, and wireless communications.

Chenguang Yang (M’10-SM’16) received the Ph.D. degree in control engineering from the National University of Singapore, Singapore, in 2010, and postdoctoral training in human robotics from the Imperial College London, London, U.K. He was awarded UK EPSRC UKRI Innovation Fellowship and individual EU Marie Curie International Incoming Fellowship. As the lead author, he won the IEEE Transactions on Robotics Best Paper Award (2012) and IEEE Transactions on Neural Networks and Learning Systems Outstanding Paper Award (2022). He is the Corresponding Co-Chair of IEEE Technical Committee on Collaborative Automation for Flexible Manufacturing (CAFM), and a Fellow of British Computer Society. His research interest lies in human robot interaction and intelligent system design.