

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Cooperative Kinematic Control For Multiple Redundant Manipulators Under Partially Known Information Using Recurrent Neural Network

XIAOXIAO LI¹, ZHIHAO XU¹, SHUAI LI², HONGMIN WU¹, and XUEFENG ZHOU¹

¹ Guangdong Key Laboratory of Modern Control Technology, Guangdong Institute of Intelligent Manufacturing, Guangzhou, Guangdong, 510070, P. R. China (e-mail: xiaoxiaoli1993@sina.com)

² School of Engineering, Swansea University, Swansea, United Kingdom (e-mail: shuaili@ieee.org)

Corresponding author: Xuefeng Zhou (e-mail: xf.zhou@giim.ac.cn).

This work is supported by the Guangdong Province Key Areas R&D Program (Grant NO. 2019B090919002), GDAS' Foundation of Introduction of Full-time Ph.D. Talent (2018GDASCX-0957), Guangzhou Science Research Plan-Major Project (Grant No. 201804020095), Guangdong Innovative Talent Project of Young College (Grant No. 2016TQ03X463), Science and Technology Planning Project of Guangzhou (Grant NO.201803010106).

ABSTRACT In this study, we investigate the problem of cooperative kinematic control for multiple redundant manipulators under partially known information using recurrent neural network (RNN). The communication among manipulators is modeled as a graph topology network with the information exchange that only occurs at the neighbouring robot nodes. Under partially known information, four objectives are simultaneously achieved, i.e, global cooperation and synchronization among manipulators, joint physical limits compliance, neighbor-to-neighbor communication among robots, and optimality of cost function. We develop a velocity observer for each individual manipulator to help them to obtain the desired motion velocity information. Moreover, a negative feedback term is introduced with a higher tracking precision. Minimizing the joint velocity norm as cost function, the considered cooperative kinematic control is built as a quadratic programming (QP) optimization problem integrating with both joint angle and joint speed limitations, and is solved online by constructing a dynamic RNN. Moreover, global convergence of the developed velocity observer, RNN controller and cooperative tracking error are theoretically derived. Finally, under a fixed and variable communication topology, respectively, application in using a group of iiwa R800 redundant manipulators to transport a payload and comparison with the existing method are conducted, illustrating effectiveness and superiority of the designed controller.

INDEX TERMS Velocity observer, Multiple redundant manipulators, Recurrent neural network, Motion planning, Zeroing neural network.

I. INTRODUCTION

The investigation on kinematic control of the redundant robot manipulator has continued for decades. Redundant manipulators refer to a class of serial manipulators whose degree of freedom (DOF) is more than ones needed to accomplish the desired task. Compared to the non-redundant manipulators, the redundant one is possible to perform both primary and secondary tasks simultaneously [1] because of the existence of redundancy, with extra system flexibility, reliability and versatility. Redundant manipulators are usually designed as a series of links connected by motor-driven

joints which extend from a fixed base to an end-effector [2]. Until now, researches on redundant manipulators have made great progress, including [3]–[15], and have achieved the extension from single manipulator to a collection of redundant manipulators such as [8]–[15], just name a few.

Benefit from the information interaction between multiple robots, multiple robot manipulators systems show a powerful performance in complicated or dangerous tasks such as the disaster relief and recovery task [16], [17], welding automation [18], [19], etc, with performance evaluation metrics being the system reliability, flexibility, wider

application, and the task total accomplish time or energy consumption, etc., and having lower cost [20]. However, extending single manipulator to multiple scene, on one hand, the increased structure complexity and heavy computation load impose more demands on real-time processing of robot system. On the other hand, how to design an efficient cooperative strategy for manipulators is also one of the major challenges that have to be thought over.

With the further research of the researchers using neural network to control the redundant manipulator, recurrent neural network (RNN) has shown a compelling advantage in real-time processing, and has been successfully implemented to the cooperative control of multiple robots. In [21], a RNN-based neural controller was developed for avoiding the joint drift phenomenon, achieving synchronous control of dual robot manipulators. Note that, the problem was considered at joint-acceleration level. However, in this paper, communication topology type between manipulators was not explicitly mentioned. In [12], a special RNN called the zeroing neural network was used to the cooperative control of multiple robot arms only communicated with own neighbors. In [8], a dual RNN with independent modules was employed to the cooperative control of a group of manipulators. The designed neural network consisted of modules controlling a single manipulator separately. However, in this paper, centralized topology was considered, this is to say, all robots need to access the command signal. In [9], the control law proposed in [8] was redesigned, hierarchical topology structure was considered. In [11], the problem was investigated again from the perspective of game-theory based on the RNN, alleviating the weakness of these methods proposed in [8], [9], [21].

In this study, we consider the problem of cooperative kinematic control of multiple redundant robot manipulators under partially known information. A RNN-based neural dynamic method is proposed. The communication among manipulators is modeled as a graph topology network with the information exchange that only occurs at the neighbouring robot manipulator nodes. Only partial manipulators have opportunity to access the command signal about the desired motion velocity, therefor global topology information is not known to all robots. We develop a velocity observer for each individual manipulator to help it to obtain the desired motion velocity relying on local information obtained from itself and own neighboring manipulators. Minimizing the joint velocity norm as the cost function, from perspective of optimization, the considered cooperative problem is built as a time-varying quadratic programming (QP) problem integrating with both joint angle and joint speed limitations. In pursuit of higher tracking precision, a extra negative feedback term is introduced. To solve it, then a RNN is designed with global stability. Based on the designed controller, four objectives are simultaneously achieved, i.e, global cooperation and synchronization among manipulators, joint physical limits compliance, neighbor-to-neighbor communication among robots, and optimality of

cost function.

The ensuing part of this paper is arranged around following aspects: Model description of multi-robot system and kinematic description of redundant manipulator are introduced in Section II to lay a basis. The cooperative payload transport task using multiple manipulators and control objective are also described in this part. In next chapter, a velocity observer is developed, the considered cooperative task of multiple redundant manipulators is built as a constrained QP problem, then a RNN is designed to solve it. We theoretically derives the global convergence of the velocity observer, the RNN controller in Section IV. It is guaranteed that the tracking error can convergent to zero. Simulative experiments are conducted in Section V, under a fixed and variable communication topology, respectively, application in using several manipulators to transport a payload illustrates effectiveness of the designed controller and correctness of theoretical analyses. Moreover, the comparison with the method mentioned in [11] reveals the superiority of our method. Finally, Section VI summarize the research.

The main contributions of this paper are summarized as follows:

- 1) Under partially known information, a RNN-based neural dynamic method is proposed for the cooperative kinematic control problem of multiple redundant robot manipulators. From the perspective of optimization, this problem is built as a time-varying QP problem integrating both joint angles and joint speed limitations. Then, a dynamic RNN controller with global stability is designed to solve it.
- 2) Different from the existing works, due to only partial manipulator nodes can access the command signal, a velocity observer is developed for each individual manipulator to help robots to obtain the desired motion velocity information.
- 3) The global convergence of the developed velocity observer, RNN controller are theoretically derived. It is guaranteed that the tracking error can convergent to zero. Under a fixed and variable communication topology, respectively, application in using several manipulators to transport a payload illustrates feasibility of the designed controller and correctness of theoretical analyses. Moreover, the comparison with the method mentioned in [11] is also conducted, revealing the superiority of our method.

II. PRELIMINARY

A. MODEL DESCRIPTION OF MULTI-ROBOT SYSTEM

Following [22]–[27], communications between N robots manipulators is modeled as a graph topology $\mathcal{G} = (V, E, A)$ where the nodes set $V = \{1, 2, \dots, i, \dots, N\}$ denotes a group of manipulators, and node i are the i -th robot manipulator. Communication link between two manipulators is denoted by an edge in the graph, $E \subseteq V \times V$. An edge in \mathcal{G} is denoted by an unordered pair (i, j) , $i, j \in \{1, \dots, N\}$.

If manipulator i can communicate with manipulator j , then $(i, j) \in E$. $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ is an adjacency matrix whose elements a_{ij} are nonnegative, with

$$a_{ij} = \begin{cases} 1, & (i, j) \in E, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

The neighbor set of manipulator i , i.e., the manipulators that can directly communicate with i , is defined as $\mathcal{N}(i) = \{j \in V | (i, j) \in E\}$. The Laplacian matrix of a graph is defined as $\mathcal{L} \in \mathbb{R}^{N \times N} = D - A$, with

$$D_{ij} = \begin{cases} \deg(i), & i = j, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where $\deg(i)$ denotes the degree of node i , or say the neighbor number of i .

Inspired by [22]–[25], the command signal is viewed as a virtual manipulator node. To show the communication state between the i -th manipulator with the command signal, a diagonal matrix $\mathcal{B} = \text{diag}\{b_1, b_2, \dots, b_n\} \in \mathbb{R}^{N \times N}$ is introduced, where $b_i = 1$, $i = 1, \dots, N$ if and only if the manipulator i can receive the command signal information, otherwise, $b_i = 0$. For simplifier the subsequent controller design and theoretical analyses, a assistant matrix $\mathcal{H} \in \mathbb{R}^{N \times N}$, $\mathcal{H} = \mathcal{L} + \mathcal{B}$ is defined.

B. KINEMATICS DESCRIPTION OF REDUNDANT MANIPULATOR

As described in [7], [10], [12], [14], etc, the cartesian coordinate of end-effector of a manipulator is only related to its configuration in joint space. That is to say,

$$r(t) = f(\theta(t)), \quad (3)$$

where $r(t) \in \mathbb{R}^n$ and $\theta(t) \in \mathbb{R}^m$ denote the coordinate of the end-effector of a manipulator in Cartesian space and joint space at time t , respectively, and $m > n$. $f(\cdot)$ is a nonlinear function denoting the mapping relation between cartesian space and joint space. Calculating the time-derivative of both sides of Eq. (3), we can obtain the kinematic mapping between $\dot{r}(t)$ and $\dot{\theta}(t)$ in velocity level:

$$\dot{r}(t) = J(\theta(t))\dot{\theta}(t), \quad (4)$$

where $J(\theta(t)) = \partial f(\theta(t))/\partial \theta(t) \in \mathbb{R}^{m \times n}$ is Jacobian matrix. $\dot{r}(t)$ and $\dot{\theta}(t)$ are velocity of the end-effector of a manipulator in cartesian and joint space, respectively.

Owing to nonlinear and redundancy of $f(\cdot)$, it is not easy to obtain $\theta(t)$ for the known $r(t) \rightarrow r_d(t)$. In contrast, the mapping from joint to Cartesian space at velocity level (4) significantly simplifier the kinematic problem.

C. COOPERATIVE PAYLOAD TRANSPORT FOR MULTIPLE ROBOTS

Consider such a problem same as [11] that a collection of redundant manipulators are assigned to cooperatively transport a payload along a desired trajectory. Assume that

the end effector of each manipulator uniformly holds a different place on the payload. For solving, this task should be considered from two aspects:

- 1) Center point of the payload is chosen as the reference point, which is required to track the desired trajectory.
- 2) All end-effectors of manipulators should maintain the original formulation in workspace. Due to the relative movement between manipulators, the resulted stretch and squeeze of the payload is required to avoid.

Therefore, to satisfy the above requirement, the movement velocity of all manipulators should be set to be same as that of the reference point. By assigning a same tracking velocity, $v_d(t)$, along the desired trajectory, to the reference point and all manipulators, we obtain the following equation

$$J_i(\theta_i(t))\dot{\theta}_i(t) = v_d(t), \quad i = 1, 2, \dots, N \quad (5)$$

where $v_d(t) \in \mathbb{R}^n$ denotes the desired velocity of the reference point at time t . $\theta_i(t) \in \mathbb{R}^m$ and $\dot{\theta}_i(t) \in \mathbb{R}^m$ are the coordinate and velocity of the i -th manipulator in joint space, respectively. $J_i(\theta_i(t))$ are the Jacobian matrix corresponding to the i -th manipulator. To avoid confusion and easy to read, Eq. (5) is abbreviated as:

$$J_i\dot{\theta}_i = v_d, \quad i = 1, 2, \dots, N \quad (6)$$

in which J_i , θ_i , $\dot{\theta}_i$, and v_d are short for $J_i(\theta_i(t))$, $\theta_i(t)$, $\dot{\theta}_i(t)$, $v_d(t)$, respectively.

D. CONTROL OBJECTIVE

In this study, we investigate the cooperative control for multiple redundant manipulators, where the desired movement is only known to part of robots. The control objective can be summarised as below:

Define the lower and upper bounds the joint angles θ_i and velocities $\dot{\theta}_i$ of the i -th manipulator as $\theta_i^-, \theta_i^+, \dot{\theta}_i^-, \dot{\theta}_i^+$, respectively, and the displacement of each robot to its initial time as $r_i(t) - r_i(0)$, $i = 1, \dots, N$. The control objective is to design joint velocity commands for robots under partially known information, to accomplish the same task together without relative displacement, that is to say, to ensure $r_1(t) - r_1(0) = \dots = r_N(t) - r_N(0) = r_d(t) - r_d(0)$, and optimize specific performance to make full use of the redundant DOFs.

III. MAIN STEPS

In this part, we will show the main steps of the proposed control scheme. Firstly, in order to handle the high precision tracking under limited information, a outer-loop controller in cartesian space is built, in which an observer is introduced based on local information. Secondly, a dynamic neural network based solver is designed to obtain control command in joint space, where the physical constraints as well as performance optimization is ensured.

A. VELOCITY OBSERVER DESIGN

As above mentioned, a distributed architecture among multiple manipulators is formed owing to the information exchange only occurs at the neighboring manipulators and partial manipulators have opportunity to have accessible to the desired reference velocity. This means that the communication range is limited for each manipulator, the state up-dation of a manipulator relies on itself state and information obtained from own neighbors. Therefore, a velocity observer \hat{v}_i is designed for i -th manipulator to help it to obtain a observation of the desired velocity v_d by decoupling the unified multiple manipulators cooperative kinematic control problem as ones on individual robot manipulator in velocity level, as follows:

$$\begin{aligned} \dot{\hat{v}}_i(t) = & -(k_1 + 1)\hat{v}_i(t) - \int_0^t (k_2 \sum_{j=0}^N a_{ij}(\hat{v}_i(t) - \hat{v}_j(t)) \\ & + k_3 \text{sgn}(\sum_{j=0}^N a_{ij}(\hat{v}_i(\tau) - \hat{v}_j(\tau)))d\tau, \end{aligned} \quad (7)$$

where $k_1, k_2, k_3 > 0$ are a tunable parameter, $\hat{v}_i, \hat{v}_j, i, j = 1, \dots, N$, stand for the observed value of the i -th manipulator and the j -th manipulator to the desired motion velocity v_d . Assume that the desired motion velocity and its first-order, second-order derivation are derivable and the derivations are bounded. Theoretically, the designed observer is with global convergence, with the corresponding proof that will be given in ensuing section.

Using the proposed observer, all robot could get access to the desired movement. This is very important to build a high-precision position controller. One feasible method is to use the output of Eq. (7) as the desired instruction directly [7], [11], in the sense that

$$J_i \dot{\theta}_i = \hat{v}_i, \quad i = 1, 2, \dots, N. \quad (8)$$

Nevertheless, if the output of the velocity observer Eq. (7) is directly assigned as the velocity of the end effector of the i -th robot manipulator, the drift to tracking error will occur, which eventually results in the relative movement between the end effectors of all manipulators. To this, a distributed error e_i corresponding to the manipulator i is defined

$$e_i = \sum_{j=0}^N a_{ij}((r_i - r_i(0)) - (r_j - r_j(0))), \quad (9)$$

where $r_i(0)$ and $r_j(0)$ denote the Cartesian coordinate of the i -th and the j -th manipulator at initial time, respectively. To achieve the kinematic cooperative control between manipulators, the velocity of the end effector is improved as $\dot{r}_i = \hat{v}_i - ke_i$ by introducing a negative feedback parameter $k > 0$. In this sense, based on the robot kinematic model, we have

$$J_i \dot{\theta}_i = \hat{v}_i - ke_i. \quad (10)$$

We will show in ensuing section that it can achieve the cooperative control of multiple robot manipulators system if the velocity of end-effector of a manipulator is designed as the \dot{r}_i .

B. QP TYPE PROBLEM DESCRIPTION AND RNN DESIGN

Obviously, Eq. (10) describes the condition that the robot joint angle velocity $\dot{\theta}_i$ needs to satisfy if the end effector is expected to move at velocity \dot{r}_i , which is also the sufficient and necessary condition for the system tracking error to converge to zero. Given the \dot{r}_i and J_i , solving $\dot{\theta}_i$ is called as the inverse kinematic problem of a manipulator in robotics. Pseudo-inverse method that for example was investigated in [28], [29], etc, is relatively simple for such a problem. The corresponding joint angle velocity $\dot{\theta}_i = J_i^\dagger(\hat{v}_i - ke_i) + (I - J_i^\dagger J_i)\alpha$ can be obtained by solving the pseudo-inverse of J_i^\dagger , where α is joint velocity component in null-space of Jacobian. However, such a scheme usually not take the physical constraint such as joint angle or joint velocity into account. For a practical robot manipulator, the joint rotatable angle is usually limited. Moreover, all joints of a manipulator are driven by a servo motor, resulting that the limited joint velocity. Once the joint limitation is violated, it will lead to larger tracking error and physical damages to the manipulator [1]. Not only that, considering the redundant property of robot system, rank of the matrix J_i satisfies $\text{rank}(J_i) \leq m < n$, $\dot{\theta}_i$ satisfying Eq. (10) is not unique. This requires us to choose a best one from numerous solutions based on some certain criterion such as minimizing the joint velocity norm or infinity norm. In this study, with aid of the constraint-optimization idea, the kinematic motion control problem of multiple redundant manipulators is described as a constraint QP problem by abstracting the physics constraints on manipulators as a set of inequality equation, where minimizing the joint velocity norm is chosen as the cost function. Specifically, the kinematic control problem of multiple redundant manipulators can be formulated as

$$\min \quad \sum_{i=1}^N \dot{\theta}_i^T \dot{\theta}_i / 2, \quad (11a)$$

$$\text{s.t.} \quad J_i \dot{\theta}_i = \hat{v}_i - ke_i, \quad (11b)$$

$$\theta_i^- \leq \theta_i \leq \theta_i^+, \quad (11c)$$

$$\dot{\theta}_i^- \leq \dot{\theta}_i \leq \dot{\theta}_i^+. \quad (11d)$$

So far, the control objective design with constraints is completed.

Obviously, the direct solving of the equation (11) is with remarkable difficulty resulting from the reason that the inequality constraint Eq. (11c) and other equations are described in different level. Therefore, based on the escape velocity method [30], Eq. (11c) and Eq. (11d) are incorporated into velocity level by introducing a negative feedback

parameter $\alpha > 0$. Consequently, Eq. (11) is reformulated as

$$\min \quad \sum_{i=1}^N \dot{\theta}_i^T \dot{\theta}_i / 2, \quad (12a)$$

$$\text{s.t.} \quad J_i \dot{\theta}_i = \hat{v}_i - k e_i, \quad (12b)$$

$$\dot{\theta}_i^{*-} \leq \dot{\theta}_i \leq \dot{\theta}_i^{*+}, \quad (12c)$$

where $\dot{\theta}_i^{*-} = \max\{\alpha(\theta^- - \theta), \dot{\theta}^-\}$, $\dot{\theta}_i^{*+} = \min\{\alpha(\theta^+ - \theta), \dot{\theta}^+\}$. For Eq. (12), define a Lagrange function as

$$L(\dot{\theta} \in \Omega_i, \lambda) = \sum_{i=1}^N \dot{\theta}_i^T \dot{\theta}_i / 2 + \lambda^T (\hat{v}_i - k e_i - J_i \dot{\theta}_i), \quad (13)$$

where $\lambda \in \mathbb{R}^m$ is the Lagrange multiplier corresponding to Eq. (12b). Based on the KKT conditions, the optimal solution of Eq. (12) satisfies

$$-\frac{\partial L}{\partial \dot{\theta}_i} \in N_{\Omega_i}(\dot{\theta}_i), \quad \frac{\partial L}{\partial \lambda} = 0. \quad (14)$$

Eq. (14) can be further rewritten as

$$\dot{\theta}_i = P_{\Omega_i}(\dot{\theta}_i - \frac{\partial L}{\partial \dot{\theta}_i}), \quad (15a)$$

$$J_i \dot{\theta}_i = \hat{v}_i - k e_i, \quad (15b)$$

where P_{Ω_i} is a projection operation to a set Ω_i , with a definition of $P_{\Omega_i}(x) = \operatorname{argmin}_{y \in \Omega_i} \|y - x\|$ [10].

Due to the nonlinear property of Eq. (15), the direct solving of the equation (15) is very difficult. On one hand, there is no general method that can directly solve Eq. (15). On the other hand, the efficiency achieved by the numerical algorithm is limited and lower, especially when it is employed in a multiple robot system. Therefore, in this part, a RNN controller is constructed to iteratively solve the Eq. (15), as follows:

$$\epsilon \ddot{\theta}_i = -\dot{\theta}_i + P_{\Omega_i}(-J_i^T \lambda_i), \quad (16a)$$

$$\epsilon \dot{\lambda}_i = \hat{v}_i - k e_i - J_i \dot{\theta}_i, \quad (16b)$$

where $\epsilon > 0$ is a constant.

IV. THEORETICAL ANALYSES

In this part, convergence of the established dynamic neural network, velocity observer and tracking error are given via Lyapunov analysis.

A. GLOBAL CONVERGENCE OF VELOCITY OBSERVER

Before giving the corresponding proof, it is necessary to give the following assumption and important properties:

Property 1 All of the non-zero eigenvalues of Laplace matrix \mathcal{L} are positive [31]. If the undirect graph \mathcal{G} is connected, then zero is a simple eigenvalue of \mathcal{L} and its eigenvector is 1_n , where $1_n = [1, \dots, 1] \in \mathbb{R}^N$.

Property 2 If \mathcal{G} is an undirect connected graph, then H is a positive definite and symmetric matrix, i.e., $\lambda_{\min}(\mathcal{H}) > 0$,

$\lambda_{\max}(\mathcal{H}) < N + 1$, where $\lambda_{\min}(\bullet)$ and $\lambda_{\max}(\bullet)$ is the minimum and maximum eigenvalue of the matrix \bullet , respectively [27], [32].

Assumption 1 The desired motion velocity v_d and its first-order, second-order derivation \dot{v}_d , \ddot{v}_d are derivable and the derivations are bounded.

Theorem 1: For an undirected connected graph \mathcal{G} , the velocity observer (7) can achieve the exact observation about the desired motion velocity v_d , i.e., $\hat{v}_i \rightarrow v_d$ when $t \rightarrow \infty$.

proof: Define an observation error:

$$\bar{v}_i = \sum_{j=0}^N a_{ij}(\hat{v}_i - \hat{v}_j). \quad (17)$$

Let $\hat{v} = [\hat{v}_1^T, \hat{v}_2^T, \dots, \hat{v}_N^T]^T$, $\bar{v} = [\bar{v}_1^T, \bar{v}_2^T, \dots, \bar{v}_N^T]^T$, Rewrite Eq. (7) and (17) as:

$$\dot{\hat{v}} = -(k_1 + 1)\hat{v} - \int_0^t (k_2 \bar{v}(\tau) + k_3 \operatorname{sgn}(\bar{v}(\tau))) d\tau, \quad (18)$$

$$\bar{v} = \mathcal{H}\hat{v} - \mathcal{B}v_d. \quad (19)$$

Combining Eq. (18) and (19), the time derivatives of \bar{v} is reformulated as

$$\dot{\bar{v}} = -(k_1 + 1)\mathcal{H}\hat{v} - \mathcal{H} \int_0^t (k_2 \bar{v}(\tau) + k_3 \operatorname{sgn}(\bar{v}(\tau))) d\tau - \mathcal{B}\dot{v}_d. \quad (20)$$

Let $\mathcal{H}s$ be an error function about \bar{v} :

$$\mathcal{H}s = \dot{\bar{v}} + \bar{v}. \quad (21)$$

In terms with Eq. (19) and (20), then the time derivatives of $\mathcal{H}s$ is

$$\begin{aligned} \mathcal{H}\dot{s} &= -(k_1 + 1)\mathcal{H}\dot{\hat{v}} - k_2 \mathcal{H}\bar{v} - k_3 \mathcal{H}\operatorname{sgn}(\bar{v}) \\ &\quad + \mathcal{H}\dot{\hat{v}} - \mathcal{B}\dot{v}_d - \mathcal{B}\ddot{v}_d \\ &= -k_1 \mathcal{H}s - (k_2 \mathcal{H} - k_1)\bar{v} - k_3 \mathcal{H}\operatorname{sgn}(\bar{v}) + \mathcal{H}\phi_d, \end{aligned}$$

where $\phi_d = -\mathcal{H}^{-1}(\mathcal{B}\ddot{v}_d + (k_1 + 1)\mathcal{B}\dot{v}_d)$. Based on assumption 1 and property 2, both ϕ_d and $\dot{\phi}_d$ are bounded. Let $c_1, c_2 > 0$ be the maximum value of ϕ_d and $\dot{\phi}_d$, we have $\|\phi_d\| < c_1$, $\|\dot{\phi}_d\| < c_2$.

Lemma 1: [23]. Given the parameter $k_2 > c_1 + c_2$, the polynomial $p(t)$ defined below is non-negative:

$$\begin{aligned} p(t) &= \bar{v}^T(0)k_3 \operatorname{sgn}(\bar{v}(0)) - \bar{v}^T(0)\phi_d(0) \\ &\quad - \int_0^t s^T \mathcal{H}(\phi_d - k_3 \operatorname{sgn}(\bar{v})) d\tau, \end{aligned} \quad (22)$$

Select a Lyapunov function candidate as

$$V = s^T \mathcal{H}s / 2 + k_2 \bar{v}^T \bar{v} / 2 + p(t), \quad (23)$$

Calculating the time derivatives of V yields

$$\begin{aligned} \dot{V} &= s^T \mathcal{H}\dot{s} + k_2 \bar{v}^T \dot{\bar{v}} + \dot{p}(t) \\ &= s^T [-k_1 \mathcal{H}s - (k_2 \mathcal{H} - k_1)\bar{v} - k_3 \mathcal{H}\operatorname{sgn}(\bar{v}) + \mathcal{H}\phi_d] \\ &\quad + k_2 \bar{v}^T (\mathcal{H}s - \bar{v}) - s^T \mathcal{H}(\phi_d - k_3 \operatorname{sgn}(\bar{v})) \\ &= -k_1 s^T \mathcal{H}s + k_1 s^T \bar{v} - k_2 \bar{v}^T \bar{v} \end{aligned} \quad (24)$$

Let $\zeta = [s^T, \bar{v}^T]^T$, Eq. (25) can be reformulated as

$$\dot{V} = -\zeta^T A \zeta, \quad (25)$$

where $A = \begin{bmatrix} k_1 \mathcal{H} & -k_1/2 \\ -k_1/2 & k_2 \end{bmatrix}$. Therefore, when $k_1 k_2 \mathcal{H} - k_1/4 > 0$, i.e., $k_1 < 4k_2 \lambda_{\min}(\mathcal{H})$, it is guaranteed that $\dot{V} \leq 0$ such that $s \rightarrow 0$ for $t \rightarrow \infty$. Based on Property 2 and Eq. (21), we have $\bar{v} \rightarrow 0$ when $t \rightarrow \infty$. In addition,

$$\begin{aligned} \bar{v} &= \mathcal{H}\hat{v} - \mathcal{B}v_d \\ &= \mathcal{H}(\hat{v} - v_d) + \mathcal{L}v_d. \end{aligned} \quad (26)$$

According to Property 1, $\mathcal{L}v_d = 0$, therefore, $\bar{v} = \mathcal{H}(\hat{v} - v_d) \rightarrow 0$ as $t \rightarrow \infty$. Due to \mathcal{H} is positive-definite, therefore, $\hat{v} - v_d \rightarrow 0$. This means, $\hat{v}_i \rightarrow v_d, i = 1, \dots, N$, as $t \rightarrow \infty$.

B. GLOBAL CONVERGENCE OF THE RNN BASED CONTROLLER

Before giving the corresponding proof, it is necessary to give the following definition and lemma:

Definition 1 For a continuously differentiable function $\mathcal{F}(\bullet)$, if $\nabla \mathcal{F} + \nabla \mathcal{F}^T$ is positive semi-definite, then $\mathcal{F}(\bullet)$ is monotone, where $\nabla \mathcal{F}$ stands for the gradient of $\mathcal{F}(\bullet)$.

Lemma 2 [5], [10], [33] A dynamic neural network is said to converge to an equilibrium that is equivalent to the optimal solution of the considered problem if and only if it satisfies

$$\kappa \dot{x} = -x + P_S(x - \varrho \mathcal{F}(x)), \quad (27)$$

where $\kappa > 0$ and $\varrho > 0$ are constant parameters, and $P_S = \text{argmin}_{y \in S} \|y - x\|$ is a projection operator to closed set S .

Theorem 2: The constructed RNN controller Eq.(16) will global converge to an equilibrium that is equivalent to the optimal solution of the problem Eq.(12) with time.

proof: Rewrite the dynamics of the designed RNN Eq.(16) as

$$\epsilon \begin{bmatrix} \ddot{\theta} \\ \dot{\lambda}_i \end{bmatrix} = \begin{bmatrix} -\dot{\theta} + P_\Omega(\dot{\theta} - J_i^T \lambda_i - \dot{\theta}) \\ -\lambda_i + (\lambda_i + \hat{v}_i - k e_i - J_i \dot{\theta}_i) \end{bmatrix}, \quad (28)$$

Let $\xi = [\dot{\theta}^T, \lambda_1^T]^T$, Eq. (28) is reformulated as

$$\epsilon \dot{\xi} = -\xi + P_\Omega(\xi - F(\xi)), \quad (29)$$

where $F(\xi) = [-J_i^T \lambda_i + \dot{\theta}; -\hat{v}_i + k e_i + J_i \dot{\theta}_i]$. Then,

$$\nabla \mathcal{F} = \partial \mathcal{F} / \partial \xi = \begin{bmatrix} I & -J_i^T \\ J_i & 0 \end{bmatrix}. \quad (30)$$

Following Definition 1, $\nabla \mathcal{F}(\xi) + \nabla \mathcal{F}^T(\xi) = \begin{bmatrix} 2I & 0 \\ 0 & 0 \end{bmatrix}$ is positive semi-definite, then $\mathcal{F}(\xi)$ is a monotone function. It is remarkable that Eq. (30) satisfies the dynamics mentioned in Lemma 2, where $\kappa = \epsilon$ and $\varrho = 1$. and $P_{\Omega_i} = [P_{\Omega_i}; P_R]$, in which P_R is a special projection operator of λ_i to \mathbb{R}^m , with its bounds being $\pm\infty$, respectively. By using Lemma 2, one conclusion can be obtained that the established neural networks would converge to the equilibriums globally.

C. GLOBAL CONVERGENCE OF TRACKING ERROR

Theorem 3: Based on the velocity observer Eq.(7) and RNN controller Eq.(16), the tracking error will global converge to zero, i.e., $r_i(t) - r_i(0) \rightarrow r_d(t) - r_d(0)$ with time $t, i = 1, \dots, N$.

proof: For the sake of convenience in writing, let $d_i = r_i(t) - r_i(0)$, $d_0 = r_d(t) - r_d(0)$, and define $e = [e_1; \dots; e_N]$, $d = [d_1, \dots, d_N]$ and $\underline{d}_0 = [d_0; \dots; d_0] \in \mathbb{R}^{n \times N}$.

Similarly, $\mathcal{L}\underline{d}_0 = 0$. Then Eq. (9) can be rewritten as

$$e = \mathcal{H}d - \mathcal{B}\underline{d}_0 = \mathcal{H}(d - \underline{d}_0). \quad (31)$$

In the last part, we have proved the convergence of the RNN controller, i.e., the output $\hat{\theta}_i$ converges to the optimal solution of Eq. (12), then the equality constraint Eq. (12b) will hold. It is noteworthy that $\dot{d}_0 = \dot{r}_d$ and $\dot{d}_i = \dot{r}_i = J_i \dot{\theta}_i$. Besides, we have already shown the convergence of both the observer in outer-loop and the dynamic neural networks in the inner-loop. We have obtained the conclusion that when the desired movement is partially known to only part of robots, \hat{v}_i can replace \dot{r}_d equivalently as $t \rightarrow \infty$. Therefore, rewrite Eq. (12b) as

$$J\dot{\theta} = \dot{r}_d - k\mathcal{H}(d - \underline{d}_0). \quad (32)$$

where J is a block diagonal matrix composed of J_1, \dots, J_N , $\theta = [\theta_1; \dots; \theta_N]$.

Define Lyapunov function

$$V_2 = e^T e / 2. \quad (33)$$

Calculate its time derivative of V_2 :

$$\begin{aligned} \dot{V}_2 &= e^T (\dot{e}) \\ &= e^T \mathcal{H}(\dot{d} - \dot{\underline{d}}_0) \\ &= e^T \mathcal{H}(J\dot{\theta} - \dot{r}_d) \\ &= -k\mathcal{H}e^T e \leq 0. \end{aligned} \quad (34)$$

Then we have, $e \rightarrow 0$ as $t \rightarrow \infty$. Based on the property that $\mathcal{L}\dot{r}_d = 0$, we have $\mathcal{H}d - \mathcal{B}\underline{d}_0 = \mathcal{H}(d - \underline{d}_0) \rightarrow 0$. Pre-multiplying \mathcal{H}^{-1} yields $d - \underline{d}_0 \rightarrow 0$, which means $r_i(t) - r_i(0) \rightarrow r_d(t) - r_d(0)$, $i = 1, \dots, N$.

TABLE 1: The D-H parameter of the redundant manipulator iiwa R800 used in the circle trajectory tracking experiment.

Link	a(m)	α (rad)	d(m)
1	0	$\pi/2$	0.34
2	0	$\pi/2$	0
3	0	$\pi/2$	0.4
4	0	$\pi/2$	0
5	0	$\pi/2$	0.4
6	0	$\pi/2$	0
7	0	0	0.126

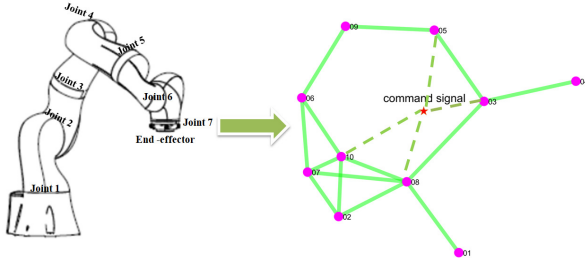


FIGURE 1: Robot system consisting of ten 7-DOF iiwa R800 manipulators which are randomly located in the workspace. Left: schematic of an iiwa R800 manipulator; Right: the communication topology corresponding to the robot system. The red dots, five-pointer star, green solid lines, and green dashed lines stand for manipulators, command signal about the desired motion trajectory, network topology, the connection to the command signal, respectively.

V. SIMULATION EXPERIMENTS

In this section, we will consider using a robot system consisting of a group of iiwa R800 manipulators with 7 DOF to verify the feasibility of the proposed coordinated control scheme. The left side of Fig. 1 shows the schematic of an 7-DOF iiwa R800 robot manipulator considered in the paper. Assume that the end-effector of every manipulator has ability to reach any position at a given orientation within its workspace. The robot system is assigned to cooperatively transport a payload, where every manipulator is held in a different position on the load so that the load can keep balance. Note that in the following simulative experiments, we only consider the position control in 3D space, therefore the used 7-DOF iiwa R800 manipulator is redundant for such a particular problem. Two simulative experiments tracking a circle desired trajectory and rhodonea trajectory are conducted, respectively.

A. CIRCLE TRAJECTORY

In this experiment, the number of robot is set to be ten, and the corresponding communication topology is as shown in the right side of Fig. 1. Among Fig. 1, only the nodes 3, 5, 8, and 10 has opportunity to access to the command signal about the desired motion trajectory. All manipulators exchange information based on the communication topology as shown in Fig. 1. Table 1 shows the D-H parameter of the redundant manipulator iiwa R800 used in this simulative experiment. The desired motion trajectory is to track a circular path with the radius being 0.1. As for the neural network parameters, we choose that $\epsilon_1 = \epsilon_2 = \dots = \epsilon_{10} = 5 \times 10^{-5}$, the upper and low bound of both the joint angles and the joint speed are set to be $\theta_i^- = -\theta_i^+ = -1.5$, $\dot{\theta}_i^- = -\dot{\theta}_i^+ = -0.7$, respectively, $i = 1, 2, \dots, 10$. The parameters k_1, k_2, k_3 are chosen as 5, 20 and 20, respectively.

Simulative results achieved by the proposed RNN controller when the multi-robot manipulators system is assigned

to track a circle trajectory are illustrated in Fig. 2. The simulation duration time is 10s. We only give the motion trajectory corresponding to the 1-th robot manipulator, as shown in Fig. 2(a), due to orientation of all manipulators are almost identical originating from the initial joint angle is set to be $0.1 * \text{randn}(70, 1) + 1$, where $\text{randn}(\cdot)$ is a MATLAB function. The corresponding profile for the joint angle θ , joint speed $\dot{\theta}$, costate variable λ captured by the RNN controller are shown in Fig. 2(b)-(d), respectively. From Fig. 2(b)-(c), we can observe that the given θ and $\dot{\theta}$ are compliant with constraints (11b) and (11c), respectively. When some subelements of θ and $\dot{\theta}$ exceed the upper or lower bound, they will be saturated, ensuring the joint physical limits compliance. Fig. 2(e) shows the observer error achieved by the developed velocity observer (7) by subtracting the desired motion velocity. We can easily observed that the error value of every dimension of all manipulators gradually converges to zero after several seconds, showing the effectiveness of the developed velocity observer. Fig. 2(f) shows the position error profile of every manipulator in 3-D workspace, and the error accuracy reaches to 10^{-4} level. The error profile corresponding to the equality constraint $J_i \dot{\theta}_i - \hat{v}_i + k e_i$ is shown in Fig. 2(g). Fig. 2(h)-(j) shows the position error profile of $r_i(t) - r_i(0)$ in x-, y- and z-axis, respectively. The parameter $r_i(0)$ denotes the initial position of the i -th robot manipulator at 0 instant, $i = 1, 2, \dots, 10$. It is obvious that $r_i(t) - r_i(0)$ of every manipulator quickly tend to be coincident, further showing the effectiveness of the proposed RNN controller.

B. COMPARISON

To further evaluate the control accuracy, by normalizing the position error of all robot in x -, y - and z -axis, respectively, comparisons on position error between our method and the method proposed in [11] are conducted, the corresponding simulative results are shown in Fig. 3. We can observe that our method is better than [11] in terms of convergence quality.

C. RHODONEA TRAJECTORY

In this subsection, we use a robot system consisting of six iiwa R800 manipulators to track a rhodonea trajectory. We conduct the simulative experiment in virtual robot experimentation platform (Vrep) [34]. The upper and low bound of the joint angles are set to be $\theta_i^- = -\theta_i^+ = -2$, $i = 1, 2, \dots, 6$. Parameters used in the designed velocity observer k_1, k_2, k_3 are chosen as 7, 20 and 10, respectively.

Different from the first experiment, we consider this problem in a variable topology environment. Specifically, in the first five seconds of the motion-task duration, all robots exchange information with the communication topology shown in Fig. 4(a), and in the ensuing time, use the communication topology shown in Fig. 4(b). The motion-task duration is set to be 10s. We can easily observe that for Fig. 4(a), robots 01 and 03 have accessible to the command

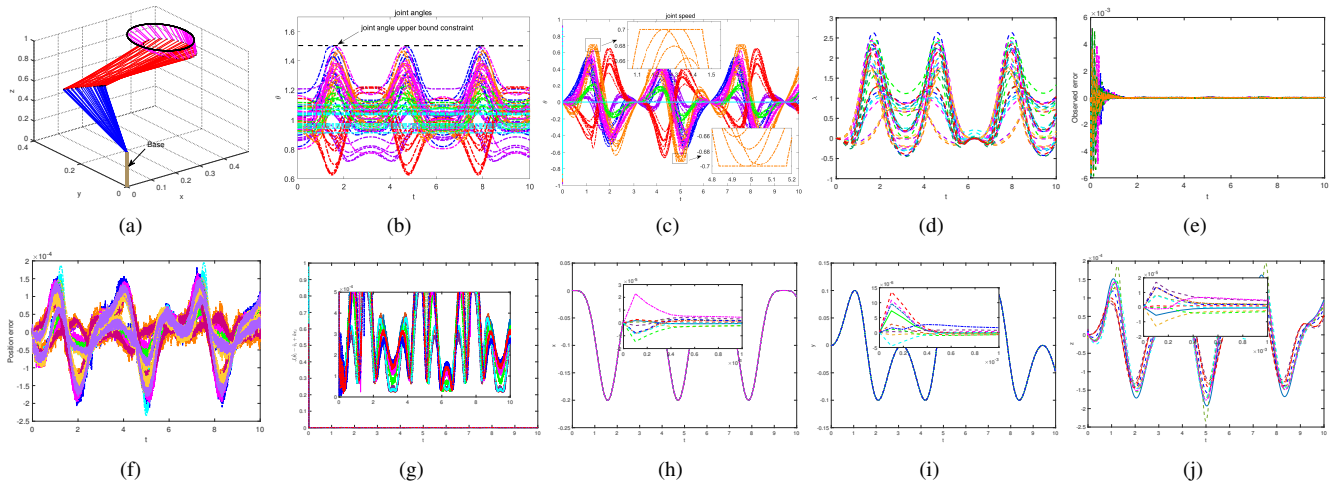


FIGURE 2: Simulative results achieved by the proposed RNN controller when the multi-robot manipulator system is assigned to track a circle trajectory. (a) Motion trajectory for the 1-th manipulator. Time history profile for (b) the joint angle $\theta(t)$, (c) joint speed $\dot{\theta}(t)$ and (d) costate variable λ , respectively. (e) Observer error. (f) Position error. (g) Error profile for the equality constraint $J_i \theta_i - \hat{v}_i + k e_i$. (h)-(j): Position error profile of $r_i(t) - r_i(0)$ in x-, y- and z-axis, respectively, $r_i(0)$ denotes the initial position of the i -th robot manipulator at 0 instant, $i = 1, 2, \dots, 10$.

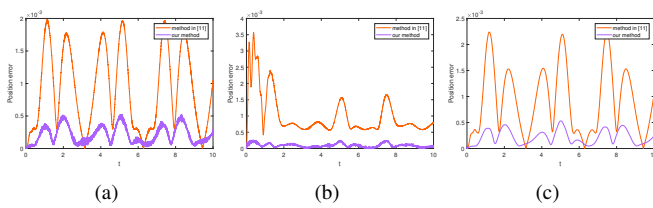


FIGURE 3: Position error comparison between our method and the method proposed in [11] in x-, y- and z-axis, respectively.

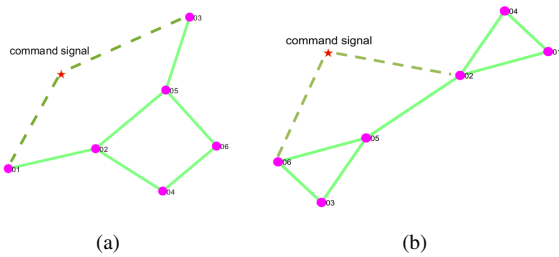


FIGURE 4: Communication topology considered in rhodonea trajectory tracking experiment. Variable communication environment is considered, in first five seconds of the motion-task duration, the communication topology as shown in Fig. 4(a) is considered. The ensuing time employs the communication topology as shown in Fig. 4(b).

signal on the desired motion velocity. For Fig. 4(b), robots 02 and 06 can access to the command signal.

The obtained simulative results are illustrated in Fig. 5- Fig. 8. Following the snapshots illustrated in Fig. 5, all robots achieve synchronous control, and achieve an accu-

rate desired rhodonea trajectory tracking despite of setting different initial joint angles for every robot. Besides this, one important point is that both the payload and the robot system are also synchronized for the rhodonea trajectory tracking by assigning a same movement velocity to them, which reveals the correctness of above mentioned in Section II-C. The corresponding time history profile of joint angles θ_i , joint velocities $\dot{\theta}_i$ and costate variable λ_i for every manipulator i are showed in Fig. 6. As demonstrated in Fig. 7(a), the velocity observer error for every manipulator in 3D workspace approach to zero with time, again showing the effectiveness of the designed observer (7). Fig. 7(b) and Fig. 7(c) show position error profile and equality constraint error profile, respectively. They all reach a better convergence accuracy. In addition, it is obvious that when topology switches, i.e., $t = 5s$, their convergence are not affected, further revealing effectiveness and robustness of our RNN controller. Position error profile of $r_i(t) - r_i(0)$ in x-, y- and z-axis are illustrated in Fig. 8, respectively, $i = 1, 2, \dots, 6$. Following them, we can say that the constructed RNN controller and velocity observer (7) is effective.

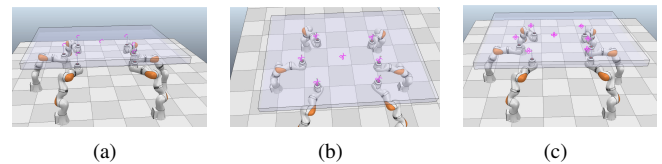


FIGURE 5: Snapshots when the robot group is required to track a rhodonea trajectory.

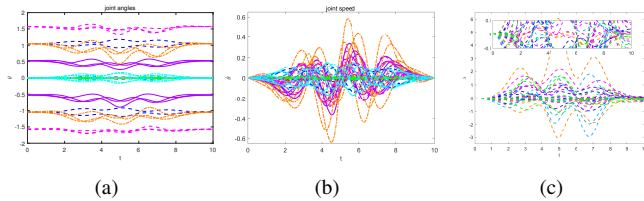


FIGURE 6: Time history profile of joint angle $\theta(t)$, joint speed $\dot{\theta}(t)$ and costate variable λ for the rhodonea trajectory tracking: (a) $\theta(t)$, (b) $\dot{\theta}(t)$ and (c) λ .

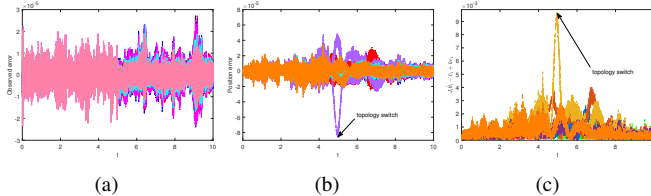


FIGURE 7: Error profile for the rhodonea trajectory tracking: (a) Observer error. (b) Position error. (c) Equality constraint error $J_i \theta_i - \hat{\theta}_i + k e_i$.

VI. CONCLUSION

A RNN-based neural dynamic scheme have been have been put forward for the problem of cooperative control of multiple redundant manipulators under partially known information in this paper. Four objectives have been simultaneously achieved, i.e, global cooperation and synchronization among manipulators, joint physical limits compliance, neighbor-to-neighbor communication among robots, and optimality of the cost function. A velocity observer have been developed for each individual manipulator to help them to obtain the desired motion velocity information. Minimizing the joint velocity norm as the cost function, the cooperative kinematic control problem has been built as a constrained QP problem, then we have designed a RNN to solve it. Global convergence of the developed velocity observer, RNN controller and cooperative tracking error have been theoretically derived. Finally, under a fixed and variable communication topology, respectively, application in using a group of iiwa R800 manipulators to transport a payload illustrated effectiveness of the proposed scheme and theoretical analyses. Moreover, comparison between our method with the existing method have also been conducted, our

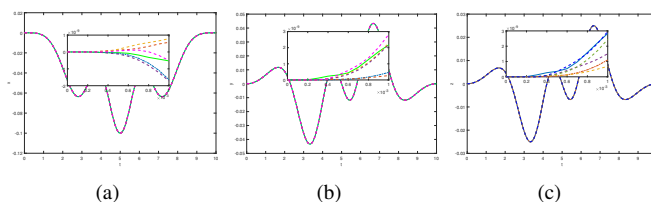


FIGURE 8: Position error profile of $r_i(t) - r_i(0)$ in x-, y- and z-axis, respectively, $i = 1, 2, \dots, 6$.

method achieved the better convergence quality.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for their valuable comments and suggestions.

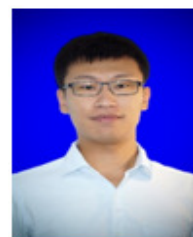
REFERENCES

- [1] 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, 2018.
- [2] L. Jin, S. Li, J. Yu, and J. He, "Robot manipulator control using neural networks: A survey," Neurocomputing, vol. 285, pp. 23–34, 2018.
- [3] K. Glass, R. Colbaugh, D. Lim, and H. Seraji, "Real-time collision avoidance for redundant manipulators," IEEE transactions on robotics and automation, vol. 11, no. 3, pp. 448–457, 1995.
- [4] Y. Xia and J. Wang, "A dual neural network for kinematic control of redundant robot manipulators," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 31, no. 1, pp. 147–154, 2001.
- [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] Y. Zhang and J. Wang, "Obstacle avoidance for kinematically redundant manipulators using a dual neural network," IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 34, no. 1, pp. 752–759, 2004.
- [7] Y. Zhang, D. Guo, B. Cai, and K. Chen, "Remedy scheme and theoretical analysis of joint-angle drift phenomenon for redundant robot manipulators," Robotics and Computer-Integrated Manufacturing, vol. 27, no. 4, pp. 860–869, 2011.
- [8] 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.
- [9] 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.
- [10] 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, 2016.
- [11] 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.
- [12] 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.
- [13] Y. Zhang, S. Li, and X. Zhou, "Adaptive kinematic control of redundant manipulators," in Deep Reinforcement Learning with Guaranteed Performance, pp. 167–197, Springer, 2020.
- [14] 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.
- [15] Y. Zhang, S. Chen, S. Li, and Z. Zhang, "Adaptive projection neural network for kinematic control of redundant manipulators with unknown physical parameters," IEEE Transactions on Industrial Electronics, vol. 65, no. 6, pp. 4909–4920, 2017.
- [16] J. Gregory, J. Fink, E. Stump, J. Twigg, J. Rogers, D. Baran, N. Fung, and S. Young, "Application of multi-robot systems to disaster-relief scenarios with limited communication," in Field and Service Robotics, pp. 639–653, Springer, 2016.
- [17] M. Roberts, T. Apker, B. Johnson, B. Auslander, B. Wellman, and D. W. Aha, "Coordinating robot teams for disaster relief," in The Twenty-Eighth International Flairs Conference, 2015.
- [18] L. Wu, K. Cui, and S.-B. Chen, "Redundancy coordination of multiple robotic devices for welding through genetic algorithm," Robotica, vol. 18, no. 6, pp. 669–676, 2000.
- [19] J. Ogbemhe and K. Mpofu, "Towards achieving a fully intelligent robotic arc welding: a review," Industrial Robot: An International Journal, vol. 42, no. 5, pp. 475–484, 2015.

- [20] Z. Yan, N. Jouandeau, and A. A. Cherif, "A survey and analysis of multi-robot coordination," *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, 2013.
- [21] L. Jin and Y. Zhang, "G2-type srmpc scheme for synchronous manipulation of two redundant robot arms," *IEEE transactions on cybernetics*, vol. 45, no. 2, pp. 153–164, 2014.
- [22] L. Cheng, Z. G. Hou, M. Tan, D. Liu, and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," in *2008 IEEE International Symposium on Intelligent Control (ISIC)*, 2008.
- [23] G. Hu, "Robust consensus tracking of a class of second-order multi-agent dynamic systems," *Systems & Control Letters*, vol. 61, no. 1, pp. 134–142.
- [24] J. Cook and G. Hu, "Experimental verification and algorithm of a multi-robot cooperative control method," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2010.
- [25] F. Zhi and G. Hu, "Distributed coordinated tracking control for multiple unknown nonlinear euler-lagrange systems," in *2013 10th IEEE International Conference on Control and Automation (ICCA)*, 2013.
- [26] R. Diestel, *Graduate texts in mathematics*. Springer-Verlag New York, Incorporated, 2000.
- [27] R. Merris, "Laplacian matrices of graphs: a survey," *Linear algebra and its applications*, vol. 197, pp. 143–176, 1994.
- [28] B. Liao and W. Liu, "Pseudoinverse-type bi-criteria minimization scheme for redundancy resolution of robot manipulators," *Robotica*, vol. 33, no. 10, pp. 2100–2113, 2015.
- [29] 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.
- [30] Y. Zhang, S. S. Ge, and H. L. Tong, "A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators," *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society*, vol. 34, no. 5, pp. 2126–2132, 2004.
- [31] P. Ogren, M. Egerstedt, and X. Hu, "A control lyapunov function approach to multi-agent coordination," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, vol. 2, pp. 1150–1155, IEEE, 2001.
- [32] Y. Hong, G. Chen, and L. Bushnell, "Distributed observers design for leader-following control of multi-agent networks," *Automatica*, vol. 44, no. 3, pp. 846–850, 2008.
- [33] Y. Zhang, S. Li, J. Gui, and X. Luo, "Velocity-level control with compliance to acceleration-level constraints: A novel scheme for manipulator redundancy resolution," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp. 921–930, 2017.
- [34] M. Freese, S. P. N. Singh, F. Ozaki, and N. Matsuhira, "Virtual robot experimentation platform v-rep: A versatile 3d robot simulator," in *Simulation, Modeling, and Programming for Autonomous Robots - Second International Conference, SIMPAR 2010, Darmstadt, Germany, November 15-18, 2010. Proceedings*, 2010.



XIAOXIAO LI received the B.E. degree from Qufu Normal University, Shandong, China in 2016, the M.S. degree in Computer Science and Technology from Qufu Normal University in 2019. She currently works at the Robotics team, Guangdong Institute of Intelligent Manufacturing, Guangzhou, China. Her current research interests include intelligent optimization neural networks and robotic control.



ZHIHAO XU received the B.E. degree and the Ph.D. degree both in School of Automation, Nanjing University of Science and Technology, Nanjing, China, in 2010 and 2016, respectively. He is currently a Postdoctoral Fellow at the Robotics team, Guangdong Institute of Intelligent Manufacturing, Guangzhou, China. His main research interests include neural networks, force control, and intelligent information processing.



SHUAI LI Shuai Li received the B.E. degree in Precision Mechanical Engineering from Hefei University of Technology, China in 2005, the M.E. degree in Automatic Control Engineering from University of Science and Technology of China, China in 2008, and the Ph.D. degree in Electrical and Computer Engineering from Stevens Institute of Technology, USA in 2014. He is currently an associate professor (reader) with School of Engineering, Swansea University, Swansea, United Kingdom. His current research interests include dynamic neural networks, robotic networks, and other dynamic problems defined on a graph.



HONGMIN WU received the Ph.D. degree in mechanical engineering, Guangdong University of Technology, Guangzhou, China, in 2019. He is currently a Postdoctoral Fellow at the Robotics team, Guangdong Academy of Sciences, Guangzhou. His research interests lie in the domain of multimodal time series modeling using Bayesian nonparametric methods, robot introspection, movement representation as well as robot learning.



XUEFENG ZHOU received the M.E. degree and the Ph.D. degree both in Mechanical Engineering, South China University of Technology, Guangzhou, China, in 2006 and 2011, respectively. He is currently the director of Robotics team in Guangdong Institute of Intelligent Manufacturing, Guangzhou, China. He won the Best Student Paper Award at the 2010 IEEE International Conference on Robotics and Biomimetics. His research interests include neural networks, force control, and intelligent information processing.

...