SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

November 2018, Vol. 61 112201:1–112201:19
https://doi.org/10.1007/s11432-017-9363-y

Cooperative deterministic learning control for a group of homogeneous nonlinear uncertain robot manipulators

Marwan ABDELATTI[1], Chengzhi YUAN[2,1*], Wei ZENG[3] & Cong WANG[4]

[1]*Department of Mechanical, Industrial and Systems Engineering, University of Rhode Island, Kingston* 02881*, USA;*
[2]*School of Mathematics and Big Data, Foshan University, Foshan* 528100*, China;*
[3]*School of Mechanical and Electrical Engineering, Longyan University, Longyan* 364012*, China;*
[4]*College of Automation Science and Engineering, South China University of Technology, Guangzhou* 510640*, China*

Received 28 October 2017/Revised 3 January 2018/Accepted 31 January 2018/Published online 23 May 2018

Abstract This paper addresses the learning control problem for a group of robot manipulators with homogeneous nonlinear uncertain dynamics, where all the robots have an identical system structure but the reference signals to be tracked differ. The control objective is twofold: to track on reference trajectories and to learn/identify uncertain dynamics. For this purpose, deterministic learning theory is combined with consensus theory to find a common neural network (NN) approximation of the nonlinear uncertain dynamics for a multi-robot system. Specifically, we first present a control scheme called cooperative deterministic learning using adaptive NNs to enable the robotic agents to track their respective reference trajectories on one hand and to exchange their estimated NN weights online through networked communication on the other. As a result, a consensus about one common NN approximation for the nonlinear uncertain dynamics is achieved for all the agents. Thus, the trained distributed NNs have a better generalization capability than those obtained by existing techniques. By virtue of the convergence of partial NN weights to their ideal values under the proposed scheme, the cooperatively learned knowledge can be stored/represented by NNs with constant/converged weights, so that it can be used to improve the tracking control performance without re-adaptation. Numerical simulations of a team of two-degree-of-freedom robot manipulators were conducted to demonstrate the effectiveness of the proposed approach.

Keywords cooperative deterministic learning, multi-robot manipulators, neural networks, adaptive control, multi-agent systems

Citation Abdelatti M, Yuan C Z, Zeng W, et al. Cooperative deterministic learning control for a group of homogeneous nonlinear uncertain robot manipulators. Sci China Inf Sci, 2018, 61(11): 112201, https://doi.org/10.1007/s11432-017-9363-y

1 Introduction

Over the past decades, applications of robots have been dramatically expanding and the complexity of their tasks has increased with more stringent performance requirements. This has inspired researchers to focus on using multiple general-purpose robots operating in a collaborative fashion to execute the assigned tasks rather than a single complex customized robot. The main goal of employing multiple robots is to divide complex tasks into smaller and simpler ones in order to save the time, energy, and cost expended, while increasing the accuracy and efficiency of their performance. An additional merit

* Corresponding author (email: cyuan@uri.edu)

of multi-robot systems is that they provide redundancy to resolve cases of failure [1, 2]. These merits have drawn the attention of many researchers and motivated them to develop various cooperative control approaches for multi-robot coordination; see [3–5].

Considerable effort has been devoted to addressing the problem of multi-robot tracking control [3–9]. In particular, in the studies described in [6–8], barrier Lyapunov functions were incorporated in an adaptive algorithm to control state constrained nonlinear systems. The authors of [9] employed critic neural networks (CNNs) and action neural networks (ANNs) to reinforce learning-based adaptive tracking control for multiple-input multiple-output nonlinear discrete-time systems. Despite the rich literature, many challenges remain. One of these challenges is how to address the nonlinear modeling uncertainties that can exert strong adverse effects on nonlinear distributed control systems [10]. Although some studies [2,11,12] considered these nonlinear model uncertainties in the robot manipulator control problem, their authors used a single robot manipulator and did not employ multiple robots. The leader-follower approach discussed in [3] assumed full knowledge about the system model and did not consider the nonlinear model uncertainties in it. Although the studies presented in [1, 5, 13, 14] considered the nonlinear uncertain dynamics for multi-robot manipulator systems, decentralized learning to only locally approximate the nonlinearities for each individual robot was employed. In the study in [15], a primal-dual neural network (PDNN) was combined with neural-dynamic optimization-based nonlinear model predictive control (NMPC) techniques for leader-follower mobile robot formation control. A control scheme for a teleoperated single robotic manipulator with dual masters that is constrained by an unknown geometrical environment was developed in [16], where radial basis function neural networks (RBF NNs) were used to deal with system uncertainties. In [17], a decentralized adaptive fuzzy control for two cooperating robotic manipulators moving an object with impedance interaction was presented. These approaches assume full communication between the follower and the leader/masters or between all the robots involved in the network. This was not our assumption in this study; however, a consensus between all the robots could still be obtained. In the research presented in [13], the graph theory was employed to control a team of robot manipulators in the presence of uncertainties and disturbances using velocity observers. However, the use of a high-gain observer in the derived controller to estimate the manipulators' velocity may excite unmodeled high-frequency dynamics and amplify measurement noise. This could negatively affect the transient performance and generate high frequency control torques that would damage the system actuators.

The deterministic learning theory using RBF NNs has been discussed in multiple papers, such as [1, 18–23]. However, in these studies the deterministic learning theory was used for each single agent independently; i.e., the agents did not share the NN learning knowledge with each other. In particular, our previous study reported in [14] considered multi-robot manipulator systems with heterogeneous nonlinear uncertain dynamics. Each robot in the team was considered a nonlinear uncertain system that was not necessarily identical to the other agents (e.g., the masses, lengths, and materials of the links differed). The NN learning/identification was performed in a fully distributed manner. One of the main challenges in this approach is that the convergent NN is different for each single agent according to the reference trajectory applied to it. This results in a convergent NN approximated for this specific reference trajectory, limiting the generality of the NNs. To explore more advanced learning capabilities of NNs, motivated by [24], in the study described in this paper we leveraged the deterministic learning theory to allow all the agents to share their adaptive NN weights with their neighbors in the network. This unifies the convergent NN weights of the agents to obtain a common approximation among all the robots and to broaden the scope of the unknown functions that each agent can approximate.

Specifically, we aimed to address the problem of trajectory tracking control and uncertain dynamics learning/identification for multi-robot manipulator systems with homogeneous nonlinear uncertain dynamics. We combine the deterministic learning theory from [18] with the consensus control theory from [25, 26]. More specifically, all the robots in the team are considered identical nonlinear uncertain systems and various reference trajectories are assigned to them. The proposed scheme, called cooperative deterministic learning (CDL), utilizes RBF NNs to approximate the robots' nonlinear uncertain dynamics. The robots communicate with each other over an undirected topology to exchange their estimated

NN weights. The weights of the RBF NNs obtained by our scheme are optimal over a domain covered by the union of all system orbits. This implies that the learned RBF NN models have a better generalization capability than those obtained by conventional deterministic learning mechanisms [1, 14]. Moreover, the accurate identification of the uncertainties introduced by our approach leads to the important property that the learned experience can be stored/represented using constant/converged NNs. These constant NNs can then be re-utilized to improve the system's performance without re-adapting the NN weights and learning knowledge exchange among the robotic agents. Extensive simulation studies were conducted to demonstrate the effectiveness of the proposed results.

Our contributions in this paper are in the following aspects. (i) Generalization capabilities are introduced into the learning scheme of a multi-robot system to approximate/identify the nonlinearities. The NN weights learned by the proposed scheme are optimal in a larger approximation domain consisting of the union of the state orbits of all robots. This scheme is more advanced than traditional decentralized learning methods, where the weights are optimal only in local approximation domains along each agent's own state orbits. (ii) In the proposed scheme, cooperative identification of the robots' nonlinear uncertain dynamics and tracking control performance can be achieved simultaneously. (iii) The proposed scheme affords a learning control law with distinctive capabilities of knowledge representation/storing and experience re-utilization. The results presented in this paper can be used to effectively improve multi-robot manipulators learning control design.

The rest of the paper is organized as follows. Some preliminary reviews on graph theory, RBF NNs, and the problem statement are given in Section 2. The main results of this study, including the CDL control design and the learning control scheme that uses experience, are presented in Sections 3 and 4, respectively. Our simulation results are provided in Section 5. Finally, in Section 6 the conclusion of the study is presented.

2 Preliminaries and problem statement

2.1 Notation and graph theory

The following notations are used in the paper. $\mathbb{R}$ denotes the set of real numbers. $\mathbb{R}_+$ represents the set of positive real numbers, $\mathbb{R}^{m\times n}$ the set of real $m \times n$ matrices, and $\mathbb{R}^n$ the set of real $n \times 1$ vectors. $\mathbb{S}^n$, $\mathbb{S}^n_+$, and $\mathbb{S}^n_-$ denote the sets of real symmetric $n \times n$ matrices and the positive definite and negative definite matrices, respectively. The identity matrix of an arbitrary dimension is denoted by I. $\mathbf{1}_n$ denotes an n-dimensional column vector, where all elements are 1. A block diagonal matrix with matrices $X_1, X_2, \ldots, X_p$ on its main diagonal is denoted by $\mathrm{diag}\{X_1, X_2, \ldots, X_p\}$. For a matrix A, $\bar{A}$ is the vectorization of A, obtained by stacking the columns of A. For a series of column vectors $x_1, x_2, \ldots, x_n$, $\mathrm{col}\{x_1, x_2, \ldots, x_n\}$ represents a column vector obtained by stacking them. For two integers $k_1 < k_2$, we denote $\mathbf{I}[k_1, k_2] = \{k_1, k_1 + 1, \ldots, k_2\}$. For a matrix M, M^{T} denotes its transpose. For $x \in \mathbb{R}^n$, the norm is defined as $\|x\| := \left(x^{\mathrm{T}}x\right)^{1/2}$. For a square matrix A, $\lambda_i(A)$ denotes its ith eigenvalue with $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ representing its maximum and minimum eigenvalues, respectively, and $\mathrm{Re}(\lambda_i(A))$ represents the real part of the ith eigenvalue of A. The notation $A \otimes B$ represents the Kronecker product of matrices A and B. We denote by B_r the open ball of radius $r > 0$ such that $B_r := \{x \in \mathbb{R}^n : \|x\| < r\}$.

In the context of multi-robot manipulator systems with interconnected communication graphs, an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a finite set of nodes $\mathcal{V} = \{1, 2, \ldots, N\}$ and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. An edge of $\mathcal{E}$, whether from node i to node j or vice versa, is denoted by (i, j), where node i is called a neighbor of node j; i.e., $(i, j) = \mathcal{E}(j, i) \in \mathcal{E}$. Note that an undirected graph is said to be connected if there is an undirected path between every pair of distinct nodes. The weighted adjacency matrix of the undirected graph $\mathcal{G}$ is a non-negative matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$, where $a_{ii} = 0$ and $a_{ij} > 0 \Rightarrow (j, i) \in \mathcal{E}$. The Laplacian of the graph $\mathcal{G}$ is denoted by $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{N\times N}$, where $l_{ii} = \sum_{j=1}^{N} a_{ij}$ and $l_{ij} = -a_{ij}$ if $i \neq j$. Therefore, given a matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$ satisfying $a_{ii} = 0$, $i \in \mathbf{I}[1, N]$ and $a_{ij} \geqslant 0$, $i, j \in \mathbf{I}[1, N]$, we can always define an undirected graph $\mathcal{G}$ such that $\mathcal{A}$ is the weighted adjacency matrix of the graph $\mathcal{G}$; we call $\mathcal{G}$ a graph of $\mathcal{A}$. It is known that at least one eigenvalue of $\mathcal{L}$ is at the origin and all nonzero eigenvalues of $\mathcal{L}$ have positive real parts [25, 26]. Moreover, according to [24, Lemma 1], $\mathcal{L}$ has one eigenvalue at the origin and all other $(N-1)$ eigenvalues have positive real parts if and only if the undirected graph $\mathcal{G}$ is connected.

2.2 Radial basis function neural networks

A standard RBF NN can be described as

$$H(X) = \sum_{i=1}^{N} w_i s_i(X) = W^{\mathrm{T}} S(X), \tag{1}$$

where $W \in \mathbb{R}^N$ is the weight vector, $X \in \Omega_X \subset \mathbb{R}^p$ is the input vector, N is the number of neurons (nodes) in the NN, $S(X) \in \mathbb{R}^N$ is the regressor vector of radial basis functions, and s_i is defined by the following common Gaussian function [27]:

$$s_i(\|X - \mu_i\|) = \mathrm{e}^{[\frac{(-(X-\mu_i)^{\mathrm{T}}(X-\mu_i))}{\varsigma_i^2}]}, \tag{2}$$

where μ_i is the mean value of the function s_i and ς_i is its width (standard deviation). The Gauss function belongs to the class of localized RBF type, where $s_i(\|X-\mu_i\|) \to 0$ as $X \to \infty$ [27]. As shown in [27], for any continuous function $f(X) : \Omega_X \to \mathbb{R}$ and for an NN function approximator with a sufficient number of neurons N, there exists an optimum constant weight vector W^* such that

$$H(X) = W^{*\mathrm{T}} S(X) + \epsilon(X), \qquad \forall X \in \Omega_X, \tag{3}$$

where $|\epsilon(X)| < \epsilon^*$ is the approximation error and ϵ^* is the upper bound of this error. In this study, we used an important class of RBF NNs called the localized RBF NN, where each basis function can only locally affect the network output [27]. This type of approximation is called spatially localized approximation [18, 27].

For any bounded trajectory $X_l(t) \subset \Omega_X$, the function $H(X)$ can be approximated using a limited number of neurons located in a local region along the trajectory,

$$H(X) = W_l^{*\mathrm{T}} S_l(X) + \epsilon_l(X), \tag{4}$$

where $S_l(X) = [s_{j1}(X), s_{j2}(X), \ldots, s_{jl}(X)]^{\mathrm{T}} \in \mathbb{R}^{N_l}$ with $N_l < N$ and $|s_{jl}| > \iota$, where $\iota > 0$ is a small positive constant. The weight vector $W_l^* = [w_{j1}^*, \ldots, w_{jl}^*]^{\mathrm{T}} \in \mathbb{R}^{N_l}$ and ϵ_l is the approximation error, where the difference $|\epsilon_l(X)| - |\epsilon(X)|$ is small [27]. Based on the previous results on the persistent excitation (PE) property of RBF networks [18, 27], it is shown that for a localized RBF network defined by $W^{\mathrm{T}} S(X)$, the centers of which are placed on a regular lattice, almost any recurrent trajectory[1)] $X(t)$ can lead to the satisfaction of the PE condition of the regressor sub-vector $S_l(X)$ [27]. The following important lemma regarding the PE condition of the RBF NNs is recalled from [18].

Lemma 1. Consider any continuous recurrent trajectory $X(t) : [0, \infty) \to \mathbb{R}^q$. $X(t)$ remains in a bounded compact set $\Omega_X \subset \mathbb{R}^q$. Then, for the RBF NN defined by $W^{\mathrm{T}} S(X)$ with centers placed on a regular lattice (sufficiently large to cover the compact set Ω_X), the regressor sub-vector $S_l(X)$ consisting of the RBFs with centers located in a small neighborhood of $X(t)$ is persistently exciting.

The following definitions and lemmas, which are important for the subsequent developments, are recalled from [24].

Consider the system

$$\dot{x} = f(t, x),\ x(t_0) = x_0, \quad t \geqslant t_0, \tag{5}$$

where $f : [t_0, \infty) \times \mathbb{R}^n \to \mathbb{R}^n$ is piecewise continuous in t and locally Lipschitz in x on $[t_0, \infty) \times \mathbb{R}^n$ and $f(t, 0) = 0$. The solution of this system is simply denoted by $x(t)$.

1) A recurrent trajectory represents a large set of periodic and periodic-like trajectories generated from linear/nonlinear dynamical systems [28]. A detailed characterization of recurrent trajectories can be found in [18].

Definition 1 ([29])**.** The origin $x = 0$ of (5) is said to be uniformly locally exponential stable (ULES) if there exist constants $\gamma_1,\ \gamma_2$, and $r > 0$ such that, for all $t > t_0$ and $(t_0, x_0) \in \mathbb{R}_+ \times B_r$, the solution satisfies

$$\|x(t, t_0, x_0)\| \leqslant \gamma_1 \|x_0\| \mathrm{e}^{-\gamma_2(t-t_0)}, \quad \forall t \geqslant t_0. \tag{6}$$

Given all the above, let us consider the system [24]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A(t,x) & B(t,x)^{\mathrm{T}} \\ -C(t,x) & -D(t,x) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{7}$$

where the states $x_1 \in \mathbb{R}^n$ and $x_2 \in \mathbb{R}^m$, $x = \left[x_1^{\mathrm{T}}, x_2^{\mathrm{T}}\right]^{\mathrm{T}}$, and $\forall t \geqslant 0$, $A \in \mathbb{R}^{n\times m},\ B \in \mathbb{R}^{m\times n},\ C \in \mathbb{R}^{m\times n}$, and $D \in \mathbb{R}^{m\times m}$ are the system matrices. Matrix D is assumed to be positive semi-definite. Let us assume the following [24].

Assumption 1. There exist $r > 0$ and ϕ_M such that $\max\{\|B(t,x)\|, \|D(t,x)\|, \|\frac{\mathrm{d}B(t,x(t))}{\mathrm{d}t}\|\} \leqslant \phi_M$ for all $t \geqslant t_0$ and $(t_0, x_0) \in \mathbb{R}_+ \times B_r$.

Assumption 2. There exist $r > 0$ and symmetric matrices $P(t,x)$ and $Q(t,x)$ such that for all $t \geqslant t_0$ and $(t_0, x_0) \in \mathbb{R}_+ \times B_r$, $A(t,x)^{\mathrm{T}}P(t,x) + P(t,x)A(t,x) + \dot{P}(t,x) = -Q(t,x)$, and $P(t,x)B(t,x)^{\mathrm{T}} = C(t,x)^{\mathrm{T}}$. Furthermore, there exist p_m, q_m, p_M, and q_M such that $p_m I_n \leqslant P(t,x) \leqslant P_M I_n$ and $q_m I_n \leqslant Q(t,x) \leqslant q_M I_n$.

Under the above two assumptions, we have the following lemma [24].

Lemma 2. Considering Assumptions 1 and 2, the system (7) is ULES where r is any fixed constant, if there exist two positive constants T_0 and α such that for all $(t_0, x_0) \in \mathbb{R}_+ \times B_r$,

$$\int_t^{t+T_0} \left[B(\tau, x(\tau, t_0, x_0))B(\tau, x(\tau, t_0, x_0))^{\mathrm{T}} + D(\tau, x(\tau, t_0, x_0))\right] \mathrm{d}\tau \geqslant \alpha I_m, \quad \forall t \geqslant t_0. \tag{8}$$

2.3 Problem statement

In this study, we aimed to address the problem of tracking control for a group of homogeneous robot manipulator systems in the presence of nonlinear uncertain dynamics. To be more specific, each robot is assigned to a different reference trajectory whereas all the robots have an identical system structure and hence the same nonlinear uncertain dynamics. These uncertain nonlinearities are approximated by utilizing the learning capability of neural networks (NNs) and the consensus theory, where the estimated weights of RBF NNs for each robot are shared over the communication topology so that a consensus about the optimum weight estimation can be reached among all agents; i.e., $\hat{W}_i \to W$ for all $i \in \mathbf{I}[1, N]$, where $\hat{W}_i$ is the estimated weight vector for the ith agent and W is the commonly convergent weight vector. The communication topology considered in our problem is an undirected connected graph. Motivated by [24], we developed a cooperative deterministic learning scheme in which the agents exchange their estimated NN weights so that all the estimated weights of the NNs can converge to small neighborhoods around their optimal values over a domain consisting of the union of all state orbits. Thus, the generalization capability of the learned controllers for accurate function approximation/identification via inter-agent collaboration [1,25,26] is better than those obtained by regular decentralized learning methods [1,5,11,14,30].

To this end, we consider a multi-robot manipulator system consisting of N robotic manipulators with homogeneous nonlinear uncertain dynamics, each of which can be modeled as [31]

$$M(q_i)\ddot{q}_i + C(q_i, \dot{q}_i)\dot{q}_i + F(\dot{q}_i) + G(q_i) = \tau_i, \quad i \in \mathbf{I}[1, N], \tag{9}$$

where the subscript i denotes the ith robotic agent in the group. For each $i \in \mathbf{I}[1, N]$, $q_i = [\, q_{i1}\ q_{i2}\ \cdots\ q_{in}\,]^{\mathrm{T}} \in \mathbb{R}^n$ represents the angular position of the joints, and $\dot{q}_i$ and $\ddot{q}_i \in \mathbb{R}^n$ represent the velocity and acceleration vectors of the joints, respectively. $\tau_i \in \mathbb{R}^n$ is the input torque. For any q_i, the inertia matrix $M(q_i)$ is positive definite (i.e., $M(q_i) \in \mathbb{S}_+^{n\times n}$), $F(\dot{q}_i) \in \mathbb{R}^n$ is the friction coefficient, and $G(q_i) \in \mathbb{R}^n$ represents the gravitational force. The centripetal torque matrix $C(q_i, \dot{q}_i) \in \mathbb{R}^{n\times n}$ is assumed to be unknown but is upper bounded by a constant matrix Y. This assumption is reasonable and made without losing any generality, and was typically adopted in many previous studies. This upper bound is local information, which is assumed to be available for the corresponding local robot. Equivalently, the above system dynamics can be rewritten as

$$\begin{aligned} \dot{x}_{i,1} &= x_{i,2}, \\ \dot{x}_{i,2} &= M^{-1}(x_{i,1})[\tau_i - C(\chi_i)x_{i,2} - G(x_{i,1}) - F(x_{i,2})], \quad \forall i \in \mathbf{I}[1, N], \end{aligned} \tag{10}$$

where $\chi_i = \mathrm{col}\{x_{i,1}, x_{i,2}\}$ with $x_{i,1}$, $x_{i,2} \in \mathbb{R}^n$, $x_{i,1} = q_i$, and $x_{i,2} = \dot{q}_i$. The dynamics terms $C(\chi_i) + G(x_{i,1}) + F(x_{i,2})$ are assumed to be uncertain.

Let us consider the following reference dynamics for each robot i to generate the position tracking reference signals:

$$\begin{aligned} \dot{x}_{di,1} &= x_{di,2}, \\ \dot{x}_{di,2} &= f_{di}(\chi_{di}, t), \quad \forall i \in \mathbf{I}[1, N], \end{aligned} \tag{11}$$

where $\chi_{di} = \mathrm{col}\{x_{di,1}, x_{di,2}\}$, where $x_{di,1}$ and $x_{di,2} \in \mathbb{R}^n$ represent the desired position and velocity, respectively, and $f_{di}(\chi_{di}, t)$ is a known continuous nonlinear function.

Remark 1. The diversity in reference trajectories assigned to each robotic agent is useful, because this may excite different unmodeled uncertain dynamics and thus broaden the search space for the optimum RBF NNs weights.

Given the multiple robotic system consisting of N number of robot manipulators in (10) and the reference trajectory in (11), we can find a non-negative matrix called the adjacency matrix $\mathcal{A} = [a_{ij}]$, $i, j \in \mathbf{I}[1, N]$, such that all the elements of $\mathcal{A}$ representing the interconnection between the agents are arbitrary non-negative numbers satisfying $a_{ii} = 0$, $\forall i \in \mathbf{I}[1, N]$. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with respect to $\mathcal{A}$. Then, $\mathcal{V} = \{1, \ldots, N\}$ corresponds to all the nodes representing the N robotic agents, and $(i, j) \in \mathcal{E}$ if and only if $a_{ij} > 0$. We consider the following assumptions regarding the reference trajectory (11) and the communication graph $\mathcal{G}$.

Assumption 3. All the states of the reference model (11) remain uniformly bounded; i.e., $\forall i \in \mathbf{I}[1, N]$, $\chi_{di} = \mathrm{col}\{x_{di,1}, x_{di,2}\} \in \Omega_i$, $\forall t \geqslant 0$, where $\Omega_i \subset \mathbb{R}^{2n}$ is a compact set. Moreover, the associated reference trajectory denoted by $\phi(\chi_{di}(0))$, starting from the initial condition $\chi_{di}(0)$, is a periodic signal.

Assumption 4. The undirected graph $\mathcal{G}$ is connected.

The above assumptions are made without losing any generality. Assumption 3 helps us prove the partial PE condition, the system stability, and estimated parameter convergence in the proposed distributed adaptive control system. However, with Assumption 4, we can prove the generalization capability of the NNs, as shown in the following.

The multi-robot manipulator control problem considered in this paper can be described as follows.

Problem 1. Given a system composed of a team of N identical robot manipulators (10) operating in an undirected connected and weighted network topology $\mathcal{G}$, our objective is to design a cooperative deterministic learning scheme such that

(1) All N robots collaboratively estimate the nonlinear uncertain dynamics $(C(\chi_i)+G(x_{i,1})+F(x_{i,2}))$, as well as accurately tracking their respective reference trajectories;

(2) The learned knowledge can be re-utilized to achieve a better control performance for all the agents without re-adapting to the nonlinear uncertain dynamics.

3 Cooperative deterministic learning using radial basis function neural networks

In this study, we assumed that each individual robotic agent can exchange its estimated knowledge with its neighboring robots. This motivated us to design a CDL scheme to enable each robotic agent to estimate the nonlinear uncertainties and exchange estimated/learned information with the other robots. To this end, let us consider the ith robotic agent and let a filtered output signal r_i be defined as

$$r_i = \dot{e}_i + \lambda_i e_i, \quad \forall i \in \mathbf{I}[1, N], \tag{12}$$

where λ_i is a positive constant and $e_i \in \mathbb{R}^n$ is the tracking error defined by

$$e_i = x_{i1} - x_{di,1}, \quad \forall i \in \mathbf{I}[1, N]. \tag{13}$$

From (10), (12), and (13), the derivative $\dot{r}_i$ is equal to

$$\begin{aligned} \dot{r}_i &= \ddot{e}_i + \lambda_i \dot{e}_i \\ &= M^{-1}(x_{i,1})(\tau_i - C(x_{i,1}, x_{i,2})x_{i,2} - F(x_{i,2}) - G(x_{i,1})) - \ddot{x}_{di,1} + \lambda_i \dot{e}_i, \quad \forall i \in \mathbf{I}[1, N]. \end{aligned} \tag{14}$$

Let the function $H(\chi_i) = [\, h_1(\chi_i) \ h_2(\chi_i) \ \cdots \ h_n(\chi_i) \,]^{\mathrm{T}}$ include all the unknown parts of the model, so that

$$H(\chi_i) = C(\chi_i)x_{i,2} + G(x_{i,1}) + F(x_{i,2}), \quad \forall i \in \mathbf{I}[1, N], \tag{15}$$

where $\chi_i = \mathrm{col}\{x_{i,1}, x_{i,2}\} \in \Omega_i \subset \mathbb{R}^{2n}$. We then employ the following RBF NNs to approximate this unknown nonlinear function:

$$H(\chi_i) = W^{\mathrm{T}} S_i(\chi_i) + \epsilon_i(\chi_i), \quad \forall i \in \mathbf{I}[1, N], \tag{16}$$

where W denotes the ideal constant weight vector and $|\epsilon_i(\chi_i)| \leqslant \epsilon_i^*$ is the approximation errors with an arbitrarily small constant $\epsilon_i^* > 0$. Let $\hat{W}_i$ be the estimate of W for individual robotic agent i; then, the feedback control law is constructed as

$$\tau_i = \hat{W}_i^{\mathrm{T}} S_i(\chi_i) + M(x_{i,1})(\ddot{x}_{di,1} - \lambda_i \dot{e}_i) - K_i r_i, \quad \forall i \in \mathbf{I}[1, N], \tag{17}$$

where $K_i \in \mathbb{S}_+^n$, $\hat{W}_i^{\mathrm{T}} S_i(\chi_i) = [\, \hat{W}_{i,1}^{\mathrm{T}} S_{i,1}(\chi_i) \ \hat{W}_{i,2}^{\mathrm{T}} S_{i,2}(\chi_i) \ \cdots \ \hat{W}_{i,n}^{\mathrm{T}} S_{i,n}(\chi_i) \,]^{\mathrm{T}}$ is used to approximate the unknown nonlinear function vector $H(\chi_i)$ in (15) along the trajectory χ_i within the compact set Ω_i. A robust self-adaptation law for online updating $\hat{W}_i$ is constructed using the σ-modification technique [27] and the consensus theory [24, 25, 29, 32, 33] through a communication topology among the agents:

$$\dot{\hat{W}}_i = \dot{\tilde{W}}_i = -\Gamma_i \left[S_i(\chi_i) r_i + \sigma_i \hat{W}_i \right] - \beta \sum_{j=1}^{N} a_{ij} \left(\hat{W}_i - \hat{W}_j \right), \quad \forall i \in \mathbf{I}[1, N], \tag{18}$$

where $\tilde{W}_i = \hat{W}_i - W$ and $\Gamma_i > 0$ is the adaptation gain, σ_i is a modification scalar constant, and $\beta > 0$ is a design parameter. Substituting (15)–(17) into (14) yields

$$\begin{aligned} \dot{r}_i &= M_i^{-1}(x_{i,1})(\tau_i - H_i) - \ddot{x}_{di,1} + \lambda_i \dot{e}_i, \\ &= M_i^{-1}(x_{i,1})(\hat{W}_i^{\mathrm{T}} S_i(\chi_i) + M_i(x_{i,1})\ddot{x}_{di,1} - M_i(x_{i,1})\lambda_i \dot{e}_i - K_i r_i - W^{\mathrm{T}} S_i(\chi_i) - \epsilon_i) - \ddot{x}_{di,1} + \lambda_i \dot{e}_i, \\ &= M_i^{-1}(x_{i,1})(\tilde{W}_i^{\mathrm{T}} S_i(\chi_i) - \epsilon_i - K_i r_i), \quad \forall i \in \mathbf{I}[1, N]. \end{aligned} \tag{19}$$

On the basis of the closed-loop dynamics (18) and (19), we first summarize the results of the overall system stability and tracking control performance in the following theorem.

Theorem 1. Given the closed-loop system consisting of the agents system (10), the reference trajectories (11), the control law (17), and the NN weight update law (18) under Assumptions 1–4, suppose the communication topology is undirected and connected. If recurrent orbits ϕ_i for the states χ_i exist in a sufficiently large compact set Ω_i such that $\chi_i \in \Omega_i$ for all $i \in \mathbf{I}[1, N]$, then, starting from any initial conditions $\chi_i(0)$ and $\hat{W}_i(0)$, we have that (i) all the signals in the closed-loop system remain uniformly bounded; (ii) the output tracking error $x_{i,1} - x_{di,1}$ converges exponentially to a small neighborhood around the origin, by appropriately choosing the design parameters K_i and σ_i for all $i \in \mathbf{I}[1, N]$; and (iii) the estimation of the NN weights $\hat{W}_i$ partially converges to a small neighborhood of their common optimal value along a trajectory $\phi_l(\chi_i(t))\,|_{t\geqslant T_i}$, and cooperative approximation of the nonlinear uncertain dynamics $H(\chi_i)$ defined in (15) can be obtained by $\hat{W}_i^{\mathrm{T}}S_i(\chi_i)$, as well as $\bar{W}_i^{\mathrm{T}}S_i(\chi_i)$, where

$$\bar{W}_i^{\mathrm{T}} = \mathrm{mean}_{t\in[t_{ia},t_{ib}]}\hat{W}_i(t), \quad \forall i \in \mathbf{I}\,[1,N]\,, \tag{20}$$

where $[t_{ia}, t_{ib}]$ $(t_{ib} > t_{ia} > T_i)$ represents a time segment after a transient period.

Proof. To prove the first point in the theorem, i.e., the boundedness of all the signals in the closed-loop system, we use the Lyapunov stability method.

(i) Let a Lyapunov function candidate for the closed-loop system (18) and (19) be

$$V = \frac{M(x_{i,1})}{2}\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i + \frac{1}{2\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\tilde{W}_i. \tag{21}$$

Then, its derivative along the trajectory of (18) and (19) is

$$\dot{V} = \frac{\dot{M}(x_{i,1})}{2}\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i + M(x_{i,1})\sum_{i=1}^{N} r_i^{\mathrm{T}}\dot{r}_i + \frac{1}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\dot{\tilde{W}}_i. \tag{22}$$

Given an appropriate definition of $C(\chi_i)$, the matrix $\frac{1}{2}\dot{M}(x_{i1}) - C(\chi_i)$ is a skew-symmetric matrix; i.e., $\frac{1}{2}\dot{M}(x_{i1}) - C(\chi_i) = 0$. This implies $\frac{1}{2}\dot{M}(x_{i1}) = C(\chi_i)$. Therefore, from (18) and (19),

$$\begin{aligned}\dot{V} \leqslant{}& Y\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i + M(x_{i,1})\sum_{i=1}^{N} r_i^{\mathrm{T}}\left[M_i^{-1}(x_{i,1})(\tilde{W}_i^{\mathrm{T}}S_i(\chi_i) - \epsilon_i - K_i r_i)\right]\\ &+ \frac{1}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\left[-\Gamma_i\left[S_i(\chi_i)r_i + \sigma_i\hat{W}_i\right] - \beta\sum_{j=1}^{N} a_{ij}\left(\hat{W}_i - \hat{W}_j\right)\right]\\ \leqslant{}& Y\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i + \sum_{i=1}^{N} r_i^{\mathrm{T}}(\tilde{W}_i^{\mathrm{T}}S_i(\chi_i) - \epsilon_i - K_i r_i)\\ &+ \frac{1}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\left[-\Gamma_i\left[S_i(\chi_i)r_i + \sigma_i\hat{W}_i\right] - \beta\sum_{j=1}^{N} a_{ij}\left(\hat{W}_i - \hat{W}_j\right)\right],\end{aligned} \tag{23}$$

where Y is the upper bound of the matrix $C(\chi_i)$. Our objective is to select the constant $K_i = K_{i1}+K_{i2}+Y$ such that K_{i1} and K_{i2} are positive definite, yielding

$$\begin{aligned}\dot{V} \leqslant{}& Y\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i - \sum_{i=1}^{N} r_i^{\mathrm{T}}\epsilon_i - \sum_{i=1}^{N} r_i^{\mathrm{T}}K_{i1}r_i - \sum_{i=1}^{N} r_i^{\mathrm{T}}K_{i2}r_i - Y\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i\\ &- \sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\sigma_i\hat{W}_i - \frac{\beta}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\left[\sum_{j=1}^{N} a_{ij}\left(\hat{W}_i - \hat{W}_j\right)\right]\\ \leqslant{}& -\sum_{i=1}^{N}\left(r_i^{\mathrm{T}}\epsilon_i + r_i^{\mathrm{T}}K_{i1}r_i + r_i^{\mathrm{T}}K_{i2}r_i\right) - \sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\sigma_i\hat{W}_i\\ &- \frac{\beta}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\left[\sum_{j=1}^{N} a_{ij}\left(\hat{W}_i - \hat{W}_j\right)\right].\end{aligned} \tag{24}$$

However, the term $\frac{\beta}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}[\sum_{j=1}^{N} a_{ij}(\hat{W}_i - \hat{W}_j)] = \frac{\beta}{\Gamma_i}\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}[\sum_{j=1}^{N} a_{ij}(\hat{W}_i - \hat{W}_j + W - W)]$. This implies

$$\dot{V} \leqslant -\sum_{i=1}^{N}\left(r_i\epsilon_i + r_i^{\mathrm{T}}K_{i1}r_i + r_i^{\mathrm{T}}K_{i2}r_i\right) - \sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\sigma_i\hat{W}_i - \frac{\beta}{\Gamma_i}\tilde{W}^{\mathrm{T}}\left(\mathcal{L}\otimes I\right)\tilde{W}, \tag{25}$$

where $\tilde{W} = [\,\tilde{W}_1^{\mathrm{T}}\ \tilde{W}_2^{\mathrm{T}}\ \cdots\ \tilde{W}_N^{\mathrm{T}}\,]^{\mathrm{T}}$ and $\mathcal{L}$ is the Laplacian matrix associated with the communication graph $\mathcal{G}$, of which all nonzero eigenvalues have positive real parts [25, 26]. Because β and Γ_i are designed to be greater than zero,

$$\dot{V} \leqslant -\sum_{i=1}^{N}\left(r_i\epsilon_i + r_i^{\mathrm{T}}K_{i1}r_i + r_i^{\mathrm{T}}K_{i2}r_i\right) - \sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\sigma_i\hat{W}_i. \tag{26}$$

From the completion of squares, we can show that

$$-\sum_{i=1}^{N}\tilde{W}_i^{\mathrm{T}}\sigma_i\hat{W}_i \leqslant -\frac{\sigma}{2}\tilde{W}^{\mathrm{T}}\tilde{W} + \sum_{i=1}^{N}\frac{\sigma_i}{2}W^{\mathrm{T}}W, \tag{27}$$

where $\sigma = \min\{\sigma_1, \ldots, \sigma_N\}$. Following the same methodology, we can show that

$$-\frac{r_i^{\mathrm{T}}K_{i2}r_i}{2} - r_i\epsilon_i \leqslant \frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})}. \tag{28}$$

Substituting (27) and (28) into (30) yields

$$\begin{aligned}
\dot{V} &\leqslant -\sum_{i=1}^{N}\frac{r_i^{\mathrm{T}}K_{i2}r_i}{2} + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})} - \sum_{i=1}^{N}r_i^{\mathrm{T}}K_{i1}r_i - \frac{\sigma}{2}\tilde{W}^{\mathrm{T}}\tilde{W} + \sum_{i=1}^{N}\frac{\sigma_i}{2}W^{\mathrm{T}}W \\
&\leqslant -\frac{1}{2}\sum_{i=1}^{N}r_i^{\mathrm{T}}\left(2K_{i1} + K_{i2}\right)r_i - \frac{1}{2}\sigma\tilde{W}^{\mathrm{T}}\tilde{W} + \frac{1}{2}\sum_{i=1}^{N}\sigma_iW^{\mathrm{T}}W + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})} \\
&\leqslant -\frac{1}{2}\rho\sum_{i=1}^{N}\left(r_i^{\mathrm{T}}r_i + \frac{1}{\Gamma_i}\tilde{W}_i^{\ \mathrm{T}}\tilde{W}_i\right) + \frac{1}{2}\sum_{i=1}^{N}\sigma_iW^{\mathrm{T}}W + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})} \\
&\leqslant -\rho V + \delta,
\end{aligned} \tag{29}$$

where $\rho = \min\{2K_{i1}, K_{i2}, \sigma\Gamma\}$ and $\delta = \frac{1}{2}\sum_{i=1}^{N}\sigma_iW^{\mathrm{T}}W + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})}$. Then, Eq. (21) satisfies

$$0 \leqslant V(t) \leqslant \frac{\delta}{\rho} + V(0)\mathrm{e}^{-\rho t}, \tag{30}$$

which implies the boundedness of $V(t)$. Thus, r_i and $\tilde{W}_i$ are uniformly bounded. In addition, from (12) and (13) we obtain the boundedness of $x_{i,1}$ and $x_{i,2}$. This implies the boundedness of the control command τ_i from (17). Therefore, all the signals from the closed-loop system are bounded. This proves the first part.

(ii) According to (21) and (30), we have

$$0 \leqslant \sum_{i=1}^{N}\left(r_i^{\mathrm{T}}r_i\right) \leqslant 2V(t) \leqslant \frac{2\delta}{\rho} + V(0)\mathrm{e}^{-\rho t}, \tag{31}$$

which implies that there exists a finite time $T_i > 0$ determined by δ and ρ such that $\forall t \geqslant T_i$, r_i converges exponentially to a small vicinity close to zero. Hence, the tracking errors e_i converge to a neighborhood close to zero according to (12). This neighborhood can be made arbitrarily small, since δ/ρ can be made arbitrarily small by appropriately selecting the design parameters K_{i1} and K_{i2} with sufficiently large $\lambda_{\max}(K_{i1}) > 0$ and $\lambda_{\max}(K_{i2}) > 0$ and small σ_i. This proves the second part.

(iii) From the definition of the localized NN Subsection 2.2 and (4), the system dynamics (14) can be written as

$$\begin{aligned}
\dot{r}_i &= M^{-1}(x_{i,1})(\tau_i - H(\chi_i) - \ddot{x}_{di,1} + \lambda_i\dot{e}_i) \\
&= M^{-1}(x_{i,1})(\hat{W}_{il}^{\mathrm{T}}S_{il}(\chi_i) - K_ir_i + \hat{W}_{i\bar{l}}^{\mathrm{T}}S_{i\bar{l}}(\chi_i) - W_lS_{il}(\chi_i) - \epsilon_{il}) \\
&= M^{-1}(x_{i,1})(\tilde{W}_{il}^{\mathrm{T}}S_{il}(\chi_i) - K_ir_i - \epsilon'_{il}), \quad \forall i \in \mathbf{I}[1, N],
\end{aligned} \tag{32}$$

where along the union of the tracking orbits $\phi_l = \phi_{l1} \cup \cdots \cup \phi_{lN}$ after time T_i; the subscript l represents parts related to regions close to the tracking orbits $\phi, \phi_1, \phi_2, \ldots, \phi_N$. $\hat{W}_{il}$ and $\tilde{W}_{il}$ are the local estimated neural weights and the local neural weights estimation error of each agent, respectively, the subscript $\bar{l}$ represents parts related to regions far from the tracking orbits, and $\epsilon'_{il} = \epsilon_{il} - \hat{W}_{i\bar{l}}^{\mathrm{T}} S_{i\bar{l}}(\chi_i) = \mathcal{O}(\epsilon_{il})$ is the NN approximation error along the tracking orbit trajectory [18], $\epsilon_{il} = [\, \epsilon_{il,1} \;\, \epsilon_{il,2} \;\, \cdots \;\, \epsilon_{il,n} \,]^{\mathrm{T}}$ and

$$W_l^{\mathrm{T}} S_{il}(\chi_i) = \left[\, W_{l,1}^{\mathrm{T}} S_{il,1}(\chi_i) \;\, W_{l,2}^{\mathrm{T}} S_{il,2}(\chi_i) \;\, \cdots \;\, W_{l,n}^{\mathrm{T}} S_{il,n}(\chi_i) \,\right]^{\mathrm{T}}, \quad \forall i \in \mathbf{I}\,[1, N].$$

In addition, the NN weight update law (18) can be rewritten as

$$\dot{\hat{W}}_i = \dot{\tilde{W}}_i = -\Gamma_i \left[S_{il}(\chi_i) r_i + \sigma_i \hat{W}_{il}\right] - \beta \sum_{j=1}^{N} a_{ij} \left(\tilde{W}_{il} - \tilde{W}_{jl}\right), \quad \forall i \in \mathbf{I}\,[1, N]. \tag{33}$$

Since

$$\begin{bmatrix} \beta \sum_{j=1}^{N} a_{1j} \left(\tilde{W}_{1l} - \tilde{W}_{jl}\right) \\ \vdots \\ \beta \sum_{j=1}^{N} a_{Nj} \left(\tilde{W}_{Nl} - \tilde{W}_{jl}\right) \end{bmatrix} = \beta \left(\mathcal{L} \otimes I\right) \tilde{W}_l,$$

where $\tilde{W}_l = [\tilde{W}_{1l}^{\mathrm{T}}, \ldots, \tilde{W}_{Nl}^{\mathrm{T}}]^{\mathrm{T}}$, the overall closed-loop adaptive learning system can be described by

$$\begin{bmatrix} \dot{r} \\ \dot{\tilde{W}}_l \end{bmatrix} = \begin{bmatrix} -\bar{M}\bar{K} & \bar{M}\Phi^{\mathrm{T}}(r_i) \\ -\Gamma\Phi(r_i) & -\beta\left(\mathcal{L} \otimes I\right) \end{bmatrix} \begin{bmatrix} r \\ \tilde{W}_l \end{bmatrix} + \begin{bmatrix} -\bar{M}\epsilon'_l \\ -\Gamma\Lambda\hat{W}_l \end{bmatrix}, \tag{34}$$

where $\bar{M} = I \otimes M^{-1}(x_{i,1})$, $\bar{K} = \mathrm{diag}\left\{K_1, K_2, \ldots, K_N\right\}$, $\Phi(r_i) = \mathrm{diag}\left\{S_{1l}(\chi_1), \ldots, S_{Nl}(\chi_N)\right\}$, $\Lambda = \mathrm{diag}\left\{-\sigma_1 I, \ldots, -\sigma_N I\right\}$, $\epsilon'_l = [\epsilon'_{1l}, \cdots, \epsilon'_{Nl}]^{\mathrm{T}}$ and $\hat{W}_l = [\hat{W}_{1l}^{\mathrm{T}}, \ldots, \hat{W}_{Nl}^{\mathrm{T}}]^{\mathrm{T}}$. Since ϵ and σ_i can be made arbitrarily small, and given the boundedness of $\hat{W}_{il}$, we conclude that $-M^{-1}(x_{i,1})\epsilon'_l$ and $\Gamma_i\Lambda\hat{W}_l$ are also arbitrarily small. Based on [34, Lemma 9.2], if the nominal part of (34), that is,

$$\begin{bmatrix} \dot{r} \\ \dot{\tilde{W}}_l \end{bmatrix} = \begin{bmatrix} -\bar{M}\bar{K} & \bar{M}\Phi^{\mathrm{T}}(r_i) \\ -\Gamma_i\Phi(r_i) & -\beta\left(\mathcal{L} \otimes I\right) \end{bmatrix} \begin{bmatrix} r \\ \tilde{W}_l \end{bmatrix} \tag{35}$$

is ULES, we conclude that $(r_i, \tilde{W}_l)$ converges to a neighborhood of the origin. Subsequently, Assumption 1 is verified based on the boundedness of V and Assumption 2 can also be verified by taking $P = \Gamma_i I$ and $Q = \Gamma_i(\bar{M}\bar{K} + \bar{K}^{\mathrm{T}}\bar{M}^{\mathrm{T}})$. Then, according to Lemma 2, to prove (35) is ULES, we need only to prove that

$$\int_t^{t+T_0} \left[\Phi(r(\tau))\Phi(r(\tau))^{\mathrm{T}} + \beta\left(\mathcal{L} \otimes I\right)\right] \mathrm{d}\tau \geqslant \eta I_N, \quad \forall t \geqslant t_0, \tag{36}$$

where $\eta \in \mathbb{R}_+$. From the proof of the boundedness of r_i, we have shown that, for all $i \in \mathbf{I}\,[1, N]$, there exists a finite time $T_i > 0$ such that $\forall t \geqslant T_i$, the tracking error e_i tends to a neighborhood close to zero. Moreover, since $x_{di,1}$ is a periodic signal according to Assumption 3, $x_{i,1}$ is also a periodic signal after a finite time T_i. Further, we can show from (12) that $x_{i,2}$ converges to the periodic signal $\dot{x}_{di,1}$ and thus $x_{i,2}$ is periodic. Consequently, since the RBF NN input $\chi_i = \mathrm{col}\{x_{i,1},\, x_{i,2}\}$ constitutes a periodic signal for all $t \geqslant T_i$, by referring to Lemma 1, we conclude that $\Phi(r(t))$ is PE, i.e., $\int_t^{t+T_0} \left[\Phi(r(\tau))\Phi(r(\tau))^{\mathrm{T}}\right] \mathrm{d}\tau \geqslant \eta I_N$ $\forall t \geqslant t_0$, from the definition of PE [19]. Thus, the condition of (36) is satisfied since $\beta > 0$ is a design parameter and $\mathcal{L}$ has all the nonzero eigenvalues with positive real parts [25, 26]. This means that the estimation error of the NN weight $\tilde{W}_{il}$ converges to a neighborhood close to zero. The definition of the weight estimation error, i.e., $\tilde{W}_{il} = \hat{W}_{il} - W_l$, implies that all the agents converge to a neighborhood close to the common optimal weight W_l and a consensus between all the agents is achieved. The convergence of $\hat{W}_l \to W_l$ implies that, along the periodic trajectory $\phi_l(\chi_i(t))\,|_{t \geqslant T_i}$, we have

$$H(\chi_i) = W_l^{\mathrm{T}} S_l(\chi_i) + \epsilon_l = \hat{W}_l^{\mathrm{T}} S_l(\chi_i) - \tilde{W}_l^{\mathrm{T}} S_l(\chi_i) + \epsilon_l$$

$$= \hat{W}_l^{\mathrm{T}} S_l(\chi_i) + \epsilon_{l,1} = \bar{W}_l^{\mathrm{T}} S_l(\chi_i) + \epsilon_{l,2}, \quad \forall i \in \mathbf{I}\,[1, N]\,, \tag{37}$$

where $\epsilon_{l,1} = \epsilon_l - \tilde{W}_l^{\mathrm{T}} S_l(\chi_i) = \mathcal{O}(\|\epsilon_l\|)$ because of the convergence of $\tilde{W}_l^{\mathrm{T}} \to 0$. The last equality is obtained according to (20), where $\bar{W}_l$ is the corresponding sub-vector of $\bar{W}$ along the periodic trajectory $\phi_l(\chi_i(t))\,|_{t\geqslant T_i}$ and $\epsilon_{l,2}$ is an approximation error using $\bar{W}_l^{\mathrm{T}} S_l(\chi_i)$. This apparently leads to $\epsilon_{l,2} = \mathcal{O}(\epsilon_{l,1})$ after a transient time.

However, from the definition of the localization of the Gaussian RBF NNs, after time T_i along the tracking orbit $\phi_{il}(\chi_i)|_{t\geqslant T_i}$, we have

$$\hat{W}^{\mathrm{T}} S(\chi_i) = \hat{W}_l^{\mathrm{T}} S_l(\chi_i) + \hat{W}_{\bar{l}}^{\mathrm{T}} S_{\bar{l}}(\chi_i), \quad \forall i \in \mathbf{I}\,[1, N]\,, \tag{38}$$

for the remaining neurons with centers far away from the trajectory $\phi_l(\chi_i(t))\,|_{t\geqslant T_i}$; $\|S_{\bar{l}}(\chi_i)\|$ becomes very small because of the localization property of the Gaussian RBF NNs. From the adaptation law in (18) with $\hat{W}(0) = 0$, it can be seen that the small values of $S_{\bar{l}}(\chi_i)$ activate the adaptation of the associated neural weights $\hat{W}_{\bar{l}}^{\mathrm{T}}$ only slightly. Thus, both $\hat{W}_{\bar{l}}^{\mathrm{T}}$ and $\hat{W}_{\bar{l}}^{\mathrm{T}} S_{\bar{l}}(\chi_i)$, as well as $\bar{W}_{\bar{l}}^{\mathrm{T}}$ and $\bar{W}_{\bar{l}}^{\mathrm{T}} S_{\bar{l}}(\chi_i)$, remain very small along the periodic trajectory $\phi_l(\chi_i(t))\,|_{t\geqslant T_i}$. This means that the entire RBF NN $\hat{W}^{\mathrm{T}} S(\chi_i)$ and $\bar{W}^{\mathrm{T}} S(\chi_i)$ can be used to cooperatively approximate the unknown function $H(\chi_i)$ accurately along the periodic trajectory $\phi_l(\chi_i(t))\,|_{t\geqslant T_i}$; i.e.,

$$\begin{aligned} H(\chi_i) &= \hat{W}_l^{\mathrm{T}} S_l(\chi_i) + \epsilon_{l,1} = \hat{W}^{\mathrm{T}} S(\chi_i) + \epsilon_1 \\ &= \bar{W}_l^{\mathrm{T}} S_l(\chi_i) + \epsilon_{l,2} = \bar{W}^{\mathrm{T}} S(\chi_i) + \epsilon_2, \quad \forall i \in \mathbf{I}\,[1, N]\,, \end{aligned}$$

with the approximation accuracy level of $\epsilon_1 = \epsilon_{l,1} - \hat{W}_{\bar{l}}^{\mathrm{T}} S_{\bar{l}}(\chi_i) = \mathcal{O}(\epsilon_{l,1}) = \mathcal{O}(\epsilon)$ and $\epsilon_2 = \epsilon_{l,2} - \bar{W}_{\bar{l}}^{\mathrm{T}} S_{\bar{l}}(\chi_i) = \mathcal{O}(\epsilon_{l,2}) = \mathcal{O}(\epsilon),\ \ldots, \epsilon_n = \epsilon_{l,n} - \bar{W}_{\bar{l}}^{\mathrm{T}} S_{\bar{l}}(\chi_i) = \mathcal{O}(\epsilon_{l,n}) = \mathcal{O}(\epsilon)$. This ends the proof.

Remark 2. Eq. (37) constitutes a key equation in our proof. The means of obtaining this equation are clarified in the sentences below. Similar results have been frequently obtained in many existing deterministic learning studies [1, 8, 29]. For a more quantitative analysis of the error terms, such as $\epsilon_l, \epsilon_{l,1}$ and $\epsilon_{l,2}$, readers are referred to [19, 20].

Remark 3. In the case of a not-connected graph, where some nodes are separated from the others, the information cannot be sufficiently exchanged, because these nodes cannot receive any information sent from the others. Thus, their learning process is independent of the others and their NN weights converge only to their local optimal values in the region of their neighborhood instead over a domain consisting of the union of all state orbits. This means that the generalization ability of the NN cannot be improved.

Remark 4. The first part of the proof of Theorem 1 shows the boundedness of the closed-loop system signals, including the system states and the control torque. The filter-based control we use is unlike other techniques (e.g., backstepping) that may require dynamic surface control to overcome the explosion of terms [35, 36]. Additionally, the second part of the proof shows that a careful choice of the design parameters guarantees the system stability, as well as the convergence of the error dynamics to a small neighborhood close to zero. This achieves our objective in designing a stable tracking control and accurate learning system without using dynamic surface control.

Remark 5. The proof of Theorem 1 shows that, by exchanging weight information among the robot agents, a consensus is reached in a neighborhood close to the optimum weight. This can be achieved provided that the reference trajectories are recurrent. Thus, a common optimum estimation about the robots' unknown function can be obtained. The common estimation results lead to a beneficial capability, that is, to the use of the common optimized weights as a learned experience. This learned experience can be used in different tasks with different reference trajectories without re-performing the NN learning process. This property is explained in the following section.

4 Learning control using experienced neural networks

In this section, we further address the second objective of Problem 1, that is, to achieve an accurate control performance without re-adapting the NNs to the nonlinear uncertain dynamics. To this end, consider the multiple robot manipulator system (10) and the reference models dynamics (11) with recurrent orbits $\phi_d(\chi_d)$. Now, we design an NN learning control scheme using the learned knowledge result of Section 3 such that all the signals in the closed-loop system remain bounded and the tracking error converges exponentially close to zero using the control law (17) after replacing the dynamic NN term by $\bar{W}^{\mathrm{T}}S(\chi_i)$, i.e.,

$$\tau_i = \bar{W}^{\mathrm{T}}S(\chi_i) + M(x_{i1})(\ddot{x}_{di,1} - \lambda_i\dot{e}_i) - K_i r_i, \quad \forall i \in \mathbf{I}\,[1, N]\,, \tag{39}$$

where $\bar{W}^{\mathrm{T}}S(\chi_i) = [\,\bar{W}_1^{\mathrm{T}}S_1(\chi_i)\ \bar{W}_2^{\mathrm{T}}S_2(\chi_i)\ \cdots\ \bar{W}_n^{\mathrm{T}}S_n(\chi_i)\,]^{\mathrm{T}}$ is the accurate RBF NN approximation of the nonlinear uncertain function $H(\chi_i)$ along the recurrent trajectory $\phi_l(\chi_i(t))\,|_{t\geqslant T_i}$. On this basis, we have the following theorem on learning control using experiences.

Theorem 2. Given the multi-robot manipulators system consisting of the plants (10) and the reference models dynamics (11) with the network communication topology $\mathcal{G}$ under Assumptions 1–4, the tracking control performance (i.e., the trajectory tracking error converges exponentially close to zero) can be achieved using the constant RBF NN control law (39) with the constant weights obtained from (20).

Proof. The closed-loop system of each robot can be formed by involving the local controller in (39), the robotic system dynamics in (10), and the results shown in Theorem 1, in particular the result showing that weights converge to a small vicinity of the optimum values W, (i.e., $\bar{W}_i$ is approximately equal to $\hat{W}_i$, which pushes $\tilde{W}_i$ to a negligible value). From (19), using the control law in (39), we have the closed-loop system as

$$\dot{r}_i = M_i^{-1}(x_{i,1})\,(-K_i r_i - \epsilon_i)\,, \quad \forall i \in \mathbf{I}\,[1, N]\,. \tag{40}$$

Considering the Lyapunov function candidate $V_r = \frac{M(x_{i,1})}{2}\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i$, the derivative $\dot{V}_r$ is

$$\dot{V}_r = \frac{\dot{M}(x_{i,1})}{2}\sum_{i=1}^{N} r_i^{\mathrm{T}} r_i + M(x_{i,1})\sum_{i=1}^{N} r_i^{\mathrm{T}} \dot{r}_i. \tag{41}$$

Following an argument similar to that used to prove Theorem 1 and inequality (29), we select the constant $K_i = K_{i1} + K_{i2} + Y$ such that K_{i1} and K_{i2} are positive values,

$$\dot{V}_r \leqslant -\sum_{i=1}^{N}\left(r_i\epsilon_i + r_i^{\mathrm{T}}K_{i1}r_i + r_i^{\mathrm{T}}K_{i2}r_i\right). \tag{42}$$

Following the same procedures in Theorem 1, part (i), we can show that

$$-\frac{r_i^{\mathrm{T}}K_{i2}r_i}{2} - r_i\epsilon_i \leqslant \frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})}. \tag{43}$$

Substituting (43) into (42) yields

$$\begin{aligned}
\dot{V} &\leqslant -\sum_{i=1}^{N}\frac{r_i^{\mathrm{T}}K_{i2}r_i}{2} + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})} - \sum_{i=1}^{N} r_i^{\mathrm{T}}K_{i1}r_i \\
&\leqslant -\frac{1}{2}\sum_{i=1}^{N} r_i^{\mathrm{T}}\left(2K_{i1} + K_{i2}\right) r_i + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})} \\
&\leqslant -\frac{1}{2}\rho\sum_{i=1}^{N}\left(r_i^{\mathrm{T}}r_i\right) + \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})} \leqslant -\rho V_r + \delta,
\end{aligned} \tag{44}$$

where $\rho = \min\{2K_{i1}, K_{i2}\}$ and $\delta = \sum_{i=1}^{N}\frac{\|\epsilon^*\|^2}{2\lambda_{\max}(K_{i2})}$. Then, V_r satisfies

$$0 \leqslant V_r(t) \leqslant \frac{\delta}{\rho} + V_r(0)\mathrm{e}^{-\rho t}. \tag{45}$$

By following an argument similar to that used to prove part (i) of Theorem 1, we can conclude that all the signals in the closed-loop system remain bounded and the position error of each agent $e_i = x_{i,1} - x_{di,1}$

Table 1 Parameters of the robot

Parameter	Value
m_1 (kg)	0.8
m_2 (kg)	2.3
l_1 (m)	1
l_2 (m)	1
$I_1 \times 10^{-3}$ $(\mathrm{kg}\cdot\mathrm{m}^2)$	61.25
$I_2 \times 10^{-3}$ $(\mathrm{kg}\cdot\mathrm{m}^2)$	20.42

($\forall i \in \mathbf{I}[1, N]$) converges to a small neighborhood close to zero in a finite time, where the size of this neighborhood can be determined by appropriately choosing $\lambda_{\max}(K_{i1})$ and $\lambda_{\max}(K_{i2})$ for all $i \in \mathbf{I}[1, N]$. This ends the proof and fulfills the control objective of Problem 1.

Remark 6. As compared to the results shown in Section 3 using (10), (11), (17), and (33), the results in this section do not require any online RBF NN adaptation for the robots. This notably reduces the computational expense and hence facilitates the implementation of the controller. The simulations in the following section provide further details.

5 Simulation studies

In this section, we demonstrate the effectiveness of the proposed approach by considering multiple homogeneous 2-DOF robot manipulator systems in the form of (10) with the associated parameters given by

$$M(q_i) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \quad C(q_i, \dot{q}_i) = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}, \quad F(\dot{q}_i) = \begin{bmatrix} F_{11} \\ F_{21} \end{bmatrix}, \quad G(q_i) = \begin{bmatrix} G_{11} \\ G_{21} \end{bmatrix},$$

with $M_{11} = m_1 l_{c1}^2 + m_2 \left(l_1^2 + l_{c2}^2 + 2 l_1 l_{c2} \cos(q_{i2})\right) + I_1 + I_2$, $M_{12} = m_2 \left(l_{c2}^2 + l_1 l_{c2} \cos(q_{i2})\right) + I_2$, $M_{21} = m_2 \left(l_{c2}^2 + l_1 l_{c2} \cos(q_{i2})\right) + I_2$, $M_{22} = m_2 l_{c2}^2 + I_2$, $C_{11} = -m_2 l_1 l_{c2} \dot{q}_{i2} \sin(q_{i2})$, $C_{12} = -m_2 l_1 l_{c2} (\dot{q}_{i1} + \dot{q}_{i2}) \sin(q_{i2})$, $C_{21} = m_2 l_1 l_{c2} \dot{q}_{i1} \sin(q_{i2})$, $C_{22} = 0$, $G_{11} = (m_1 l_{c2} + m_2 l_1)\, g \cos(q_{i1}) + m_2 l_{c2} g \cos(q_{i1} + q_{i2})$, $G_{22} = m_2 l_{c2} g \cos(q_{i1} + q_{i2})$, where l_1, l_2, m_1, and m_2 are the lengths and masses of the first and second links for the agents, respectively. l_{c1} and l_{c2} are the halves of these lengths, F_{11}, F_{21} are constants, and I_1 and I_2 are the inertia of the first and second links, respectively; their values are shown in Table 1. We used $N = 10$ manipulators exchanging their estimated weight information to obtain common accurate nonlinear uncertain function approximation. The following signals were constructed to be periodic reference trajectories for each individual robot to follow, $x_{d1,1} = [0.8\sin(t), 0.8\cos(t)]^{\mathrm{T}}$, $x_{d2,1} = [\cos(2t), \sin(2t)]^{\mathrm{T}}$, $x_{d3,1} = [\sin(0.5t), \cos(0.5t)]^{\mathrm{T}}$, $x_{d4,1} = [\sin^2(t), \cos^2(t)]^{\mathrm{T}}$, $x_{d5,1} = [0.5\cos^2(t), 0.5\sin^2(t)]^{\mathrm{T}}$, $x_{d6,1} = [0.5\sin^2(t), \cos(t)]^{\mathrm{T}}$, $x_{d7,1} = [\sin(t), 0.5\cos^2(t)]^{\mathrm{T}}$, $x_{d8,1} = [0.5\sin(2t), 0.5\cos^2(t)]^{\mathrm{T}}$, $x_{d9,1} = [0.5\sin(2t) + \cos(t), 0.5\cos(2t) + \sin(t)]^{\mathrm{T}}$, $x_{d10,1} = [\cos(t), \sin(t)]^{\mathrm{T}}$, where $x_{di,1} \in \mathbb{R}^2$, $i \in \mathbf{I}[1, 10]$ is the position of the desired trajectory. From these reference signals, it can be shown that Assumption 3 is satisfied. A connected undirected network topology $\mathcal{G}$ was considered, as shown in Figure 1 to satisfy Assumption 4.

5.1 Simulation for learning control

We first examined the learning control performance CDL based on the above system setup, using the control law (17) and the RBF-NN weight update law (18). For each $i \in \mathbf{I}[1, 10]$, we constructed the Gauss RBF NN $\hat{W}_i S_i(\chi_i)$ using $21 \times 21 = 441$ neuron nodes with the centers evenly placed over a state space of $[-1.2, 1.2] \times [-1.2, 1.2]$ that is determined to cover the state space of the robot manipulator system. The widths ς_i were chosen as 0.6 to guarantee even distribution of the neurons. The controller and update law parameters were selected as in other studies in the literature, e.g., [1, 20], such that $\Gamma_i = 10$, $\beta = 5$. K_{i1} and K_{i2} were selected to be sufficiently large to obtain accurate tracking as proved in Theorem 1, that is, 40 and 100, respectively. $\lambda = 20$ and $\sigma_i = 0.00001$ were chosen as in most studies in the literature, e.g., [1, 14]. The position initial conditions were $x_{1,1}(0) = [\,0.3\ 0.6\,]^{\mathrm{T}}$,

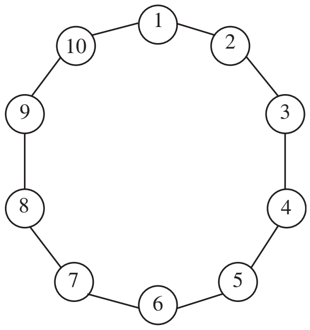

Figure 1 Network topology $\mathcal{G}$.

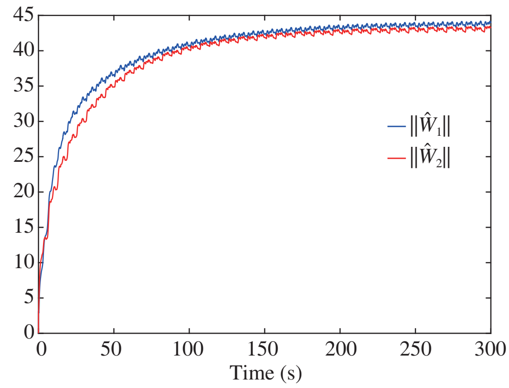

Figure 2 (Color online) $\mathcal{L}_2$ norm of partial neural network weights for Agent 1.

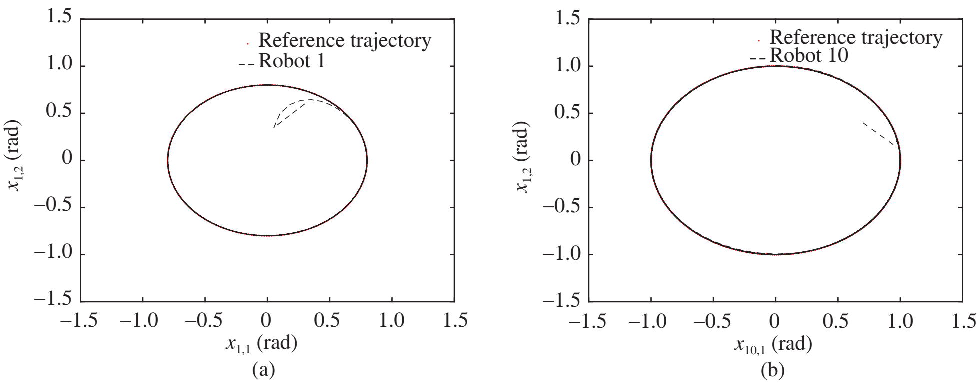

Figure 3 (Color online) Phase plane trajectories of agents. (a) Robot 1; (b) Robot 10.

$x_{2,1}(0) = [\,0.1\ 0.7\,]^{\rm T}$, $x_{3,1}(0) = [\,0.5\ 0.2\,]^{\rm T}$, $x_{4,1}(0) = [\,0.8\ 0.4\,]^{\rm T}$, $x_{5,1}(0) = [\,0.7\ 0.4\,]^{\rm T}$, $x_{6,1}(0) = [\,0.05\ 0.05\,]^{\rm T}$, $x_{7,1}(0) = [\,0.2\ 0.6\,]^{\rm T}$, $x_{8,1}(0) = [\,0.5\ 0.5\,]^{\rm T}$, $x_{9,1}(0) = [\,0.9\ 0.3\,]^{\rm T}$ and $x_{10,1}(0) = [\,0.9\ 0.9\,]^{\rm T}$ and the initial conditions for the estimated weights $\hat{W}_i(0) = [\,\hat{W}_{i,1}(0)\ \hat{W}_{i,2}(0)\,]^{\rm T}, \hat{W}_{i,1}$ and $\hat{W}_{i,2} \in \mathbb{R}^{441}$, $\forall i \in \mathbf{I}[1, 10]$ were $\hat{W}_{1,1}(0) = \hat{W}_{1,2}(0) = [\,0\ 0\ \cdots\ 0\,]^{\rm T}$, $\hat{W}_{2,1}(0) = \hat{W}_{2,2}(0) = [\,0.5\ 0.5\ \cdots\ 0.5\,]^{\rm T}$, $\hat{W}_{3,1}(0) = \hat{W}_{3,2}(0) = [\,0.1\ 0.1\ \cdots\ 0.1\,]^{\rm T}$, $\hat{W}_{4,1}(0) = \hat{W}_{4,2}(0) = [\,0.2\ 0.2\ \cdots\ 0.2\,]^{\rm T}$, $\hat{W}_{5,1}(0) = \hat{W}_{5,2}(0) = [\,0.3\ 0.3\ \cdots\ 0.3\,]^{\rm T}$, $\hat{W}_{6,1}(0) = \hat{W}_{6,2}(0) = [\,0.7\ 0.7\ \cdots\ 0.7\,]^{\rm T}$, $\hat{W}_{7,1}(0) = \hat{W}_{7,2}(0) = [\,-0.5\ -0.5\ \cdots\ -0.5\,]^{\rm T}$, $\hat{W}_{8,1}(0) = \hat{W}_{8,2}(0) = [\,-0.7\ -0.7\ \cdots\ -0.7\,]^{\rm T}$, $\hat{W}_{9,1}(0) = \hat{W}_{9,2}(0) = [\,-0.2\ -0.2\ \cdots\ -0.2\,]^{\rm T}$, $\hat{W}_{10,1}(0) = \hat{W}_{10,2}(0) = [\,1\ 1\ \cdots\ 1\,]^{\rm T}$. The simulation was run for 300 s. the results are plotted in Figure 2 through 6. The performance of the scheme in terms of estimating the optimal weight vectors W is shown in Figure 2, which indicates perfect convergence of the estimated weights, $\hat{W}_{1,1}$ and $\hat{W}_{1,2} \in \mathbb{R}^{441}$, for Robot 1 as an example, to the common optimum weight $W \in \mathbb{R}^{441}$. For simplicity of presentation, we decided to demonstrate the results of two sample robots, Agents 1 and 10. In Figure 3, we show the reference orbits and the actual trajectories of these robots. We show in Figure 4 the robot position tracking control responses and the tracking errors for the two robots. It can be observed that the tracking performance of the robots is satisfactory, despite the nonlinear uncertainties in the system. We also show the NN approximation results of the unknown system dynamics $H(\chi_1)$ and $H(\chi_{10})$ plotted in Figure 5 using RBF NNs $\hat{W}_1 S_1(\chi_1)$ and $\hat{W}_{10} S_{10}(\chi_{10})$, respectively. It is obvious that the cooperative learning succeeded in achieving accurate approximation of the unknown uncertain nonlinearities and the learned knowledge can be stored using constant NNs, as shown in Figure 6.

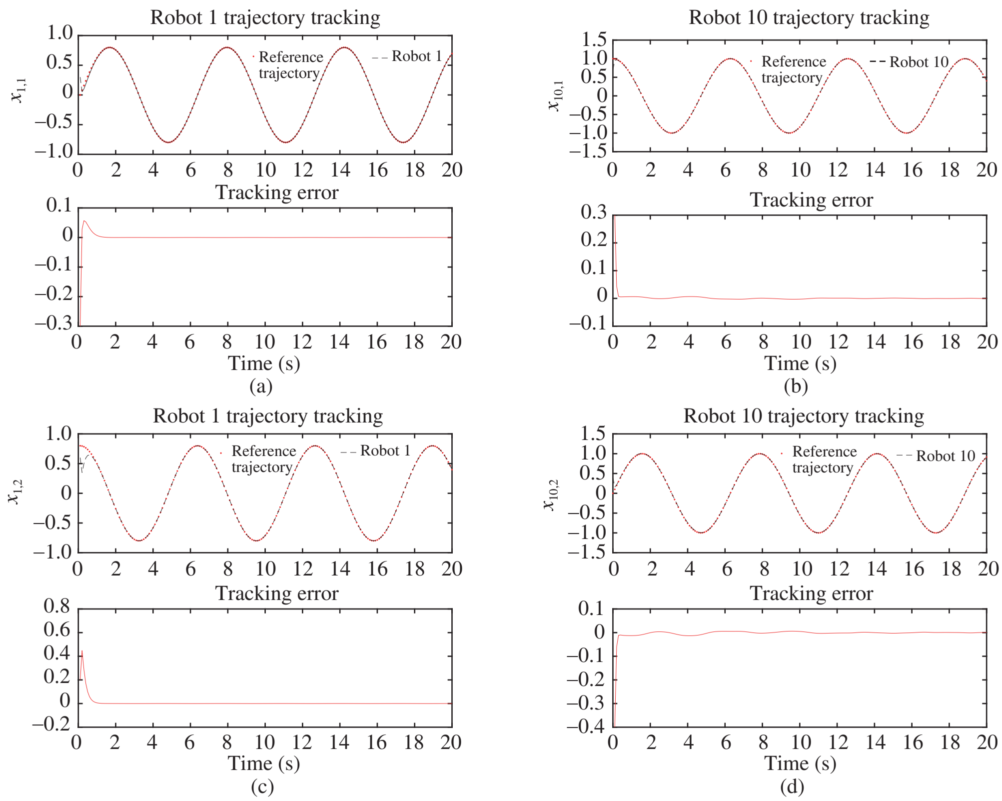

Figure 4 (Color online) Position tracking control. (a) $x_{1,1} \to x_{d1,1}$; (b) $x_{3,1} \to x_{d10,1}$; (c) $x_{1,2} \to x_{d1,2}$; (d) $x_{3,2} \to x_{d10,2}$.

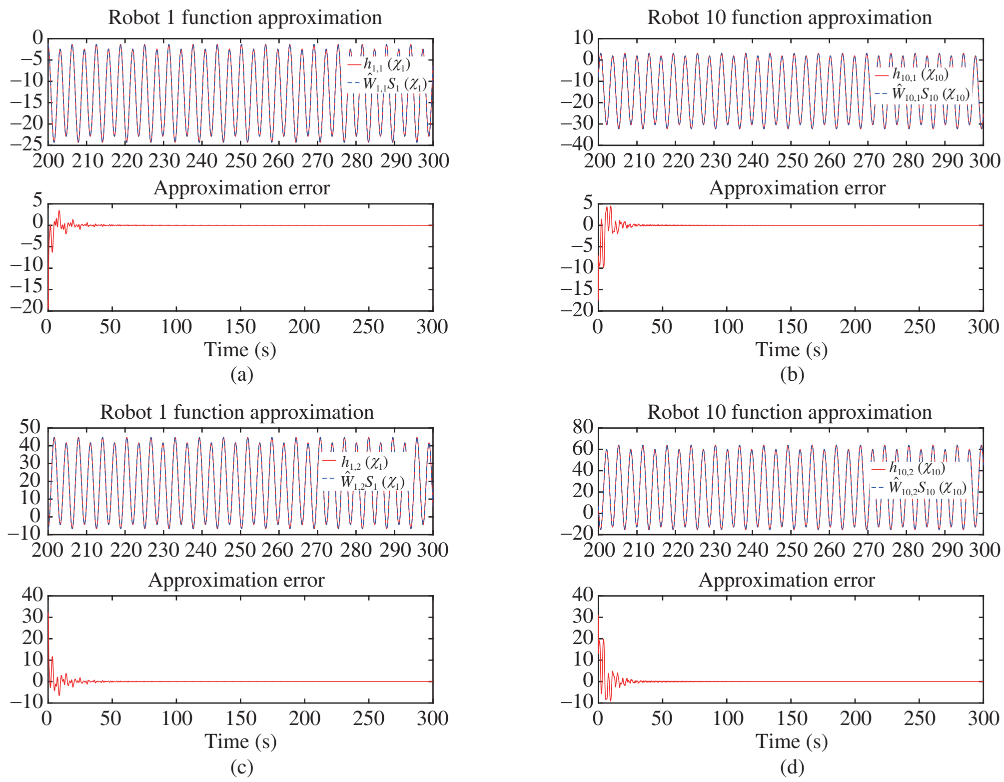

Figure 5 (Color online) Function approximation. (a) $\hat{W}_{1,1}S_1(\chi_1) \to h_1(\chi_1)$; (b) $\hat{W}_{10,1}S_{10}(\chi_{10}) \to h_1(\chi_{10})$; (c) $\hat{W}_{1,2}S_1(\chi_1) \to h_2(\chi_1)$; (d) $\hat{W}_{10,2}S_{10}(\chi_{10}) \to h_2(\chi_{10})$.

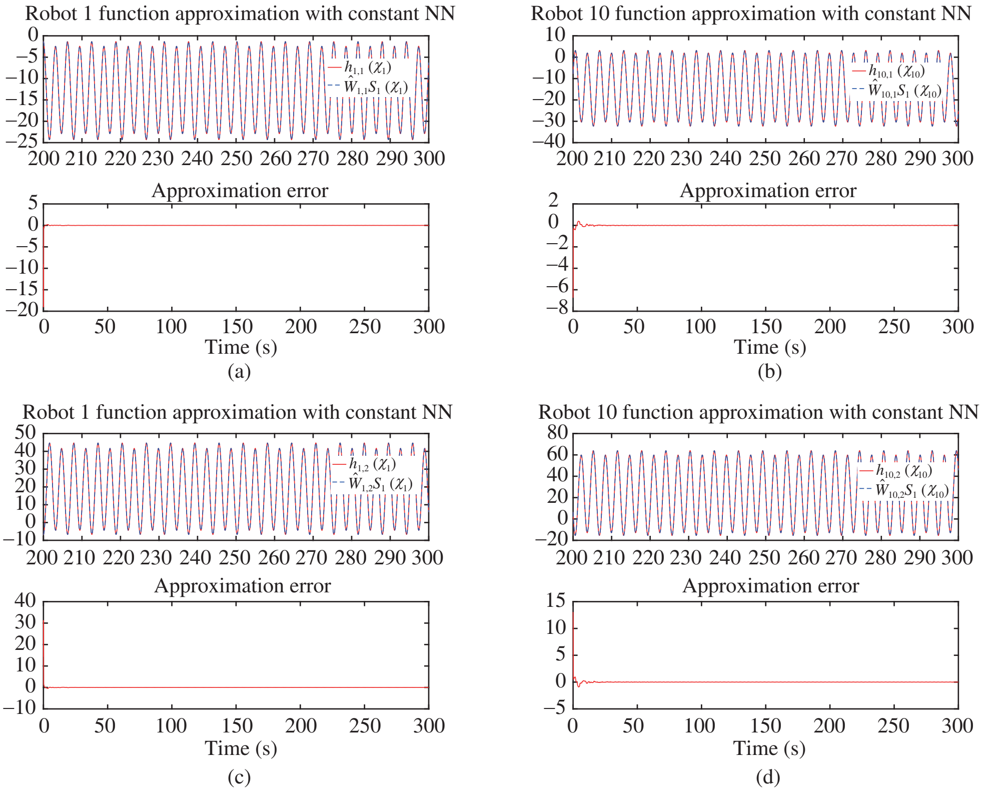

Figure 6 (Color online) Function approximation. (a) $\bar{W}_{1,1}S_1(\chi_1) \rightarrow h_1(\chi_1)$; (b) $\bar{W}_{10,1}S_{10}(\chi_{10}) \rightarrow h_1(\chi_{10})$; (c) $\bar{W}_{1,2}S_1(\chi_1) \rightarrow h_2(\chi_1)$; (d) $\bar{W}_{10,2}S_{10}(\chi_{10}) \rightarrow h_2(\chi_{10})$.

5.2 Simulation for learning control using experience

We further examined the control performance of the multi-robot manipulator system using the experience obtained from the cooperative learning results of the CDL control by employing the control law (39) with no weight update law. To this end, we considered the same system dynamics (10), reference trajectories (11), initial conditions, and control gains for fairness of comparison. The simulation results for the same robots (i.e., Agents 1 and 10) are plotted in Figures 7 and 8. It can be seen in Figure 7 as compared to Figure 4 that a much better performance is obtained despite the subtle difference between the two control laws (17) and (39). Recall that these satisfactory results are obtained without recalculation/re-adaptation of the NN weights, which is beneficial in that it reduces the computational complexity and saves the system resources/energy, especially if a large number of neurons is involved in the control process. Control input responses using the two different control laws (17) and (39) can be observed in Figure 8.

6 Conclusion

This paper addressed the problem of trajectory tracking control of multi-robot manipulators in the presence of homogeneous nonlinear uncertainties. The proposed approach is divided into two parts. The first part comprises cooperative deterministic learning using RBF NNs. It estimates the NN weight information via inter-agent communication to reach a common approximation of the uncertainties among agents. The second part is aimed to re-utilize the learned knowledge in any given reference trajectory and then regulate the robot's position and velocity accordingly. In this part, no information exchange or weight update occurs, and hence, the computational burden is reduced. This results in a reduction in the required system resources/energy, especially if a large number of neurons is involved in the control process. The CDL control law employs the cooperative learning concept to overcome the nonlinearity

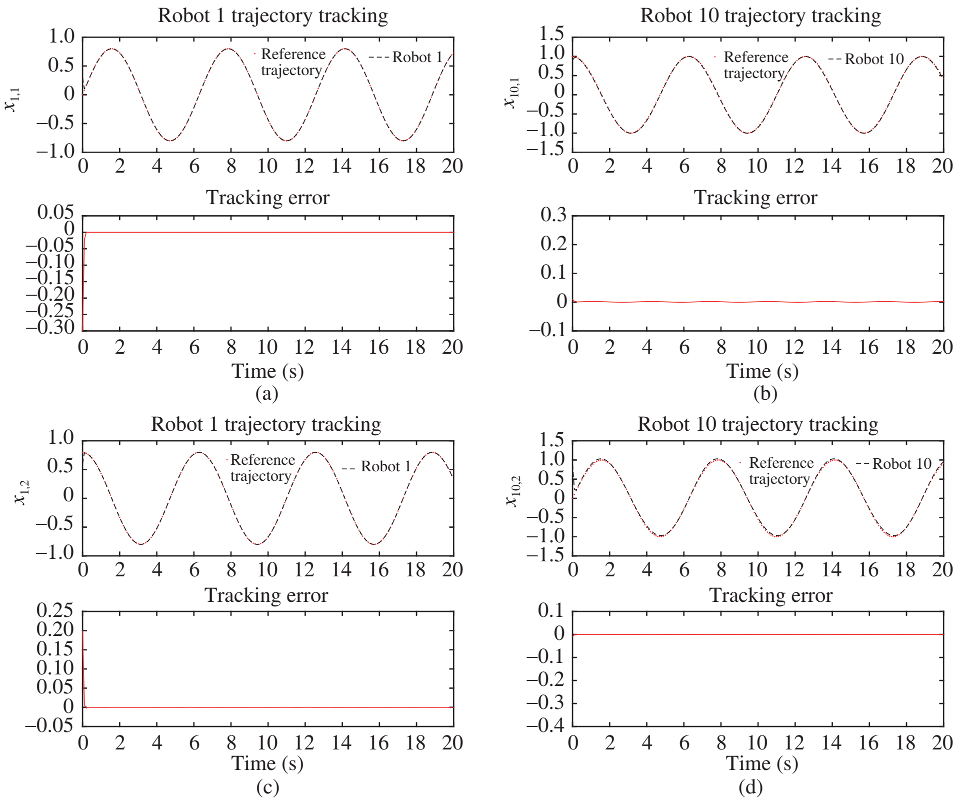

Figure 7 (Color online) Position tracking control using experience. (a) $x_{1,1} \to x_{d1,1}$; (b) $x_{10,1} \to x_{d10,1}$; (c) $x_{1,2} \to x_{d1,2}$; (d) $x_{10,2} \to x_{d10,2}$.

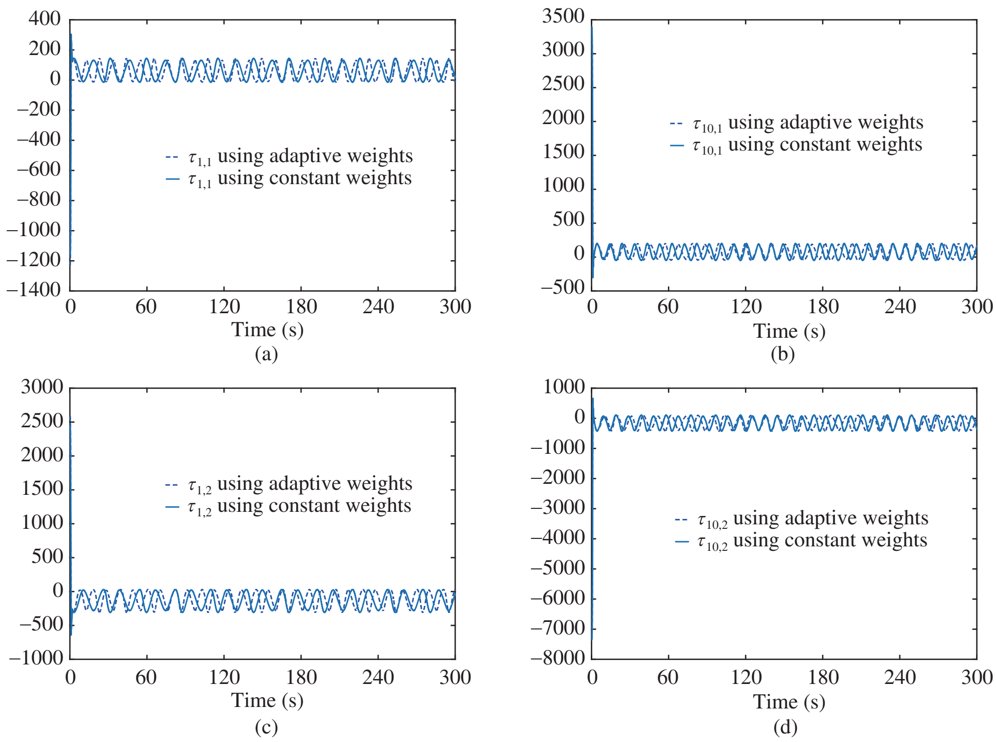

Figure 8 (Color online) Control torques (N·m). (a) Robot 1 control torque $\tau_{1,1}$; (b) Robot 10 control torque $\tau_{10,1}$; (c) Robot 1 control torque $\tau_{1,2}$; (d) Robot 10 control torque $\tau_{10,2}$.

and uncertainties in the robots model. Extensive numerical simulations for a team of 2-DOF robot manipulators demonstrated the distinctive capabilities of the technique.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61773194) and Science and Technology Project of Longyan City (Grant No. 2017LY69).

References

1 Yuan C, Licht S, He H. Formation learning control of multiple autonomous underwater vehicles with heterogeneous nonlinear uncertain dynamics. IEEE Trans Cybern, 2017, doi: 10.1109/TCYB.2017.2752458
2 Yang C, Ganesh G, Haddadin S, et al. Human-like adaptation of force and impedance in stable and unstable interactions. IEEE Trans Robot, 2011, 27: 918–930
3 Chung S J, Slotine J J E. Cooperative robot control and concurrent synchronization of lagrangian systems. IEEE Trans Robot, 2009, 25: 686–700
4 Li Z J, Tao P Y, Ge S S, et al. Robust adaptive control of cooperating mobile manipulators with relative motion. IEEE Trans Syst Man Cybern B, 2009, 39: 103–116
5 Huang J, Wen C, Wang W, et al. Adaptive finite-time consensus control of a group of uncertain nonlinear mechanical systems. Automatica, 2015, 51: 292–301
6 Liu Y J, Lu S, Li D, et al. Adaptive controller design-based ABLF for a class of nonlinear time-varying state constraint systems. IEEE Trans Syst Man Cybern, 2017, 47: 1546–1553
7 Liu Y J, Tong S. Barrier Lyapunov functions for Nussbaum gain adaptive control of full state constrained nonlinear systems. Automatica, 2017, 76: 143–152
8 Li D P, Li D J, Liu Y J, et al. Approximation-based adaptive neural tracking control of nonlinear MIMO unknown time-varying delay systems with full state constraints. IEEE Trans Cybern, 2017, 47: 3100–3109
9 Liu L, Wang Z, Zhang H. Adaptive fault-tolerant tracking control for MIMO discrete-time systems via reinforcement learning algorithm with less learning parameters. IEEE Trans Automat Sci Eng, 2017, 14: 299–313
10 Zeinali M, Notash L. Adaptive sliding mode control with uncertainty estimator for robot manipulators. Mech Mach Theor, 2010, 45: 80–90
11 He W, Chen Y, Zhao Y. Adaptive neural network control of an uncertain robot with full-state constraints. IEEE Trans Cybern, 2015, 46: 620–629
12 Li Z, Deng J, Lu R, et al. Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach. IEEE Trans Syst Man Cybern Syst, 2016, 46: 740–749
13 Liang X, Wang H, Liu Y H, et al. Adaptive task-space cooperative tracking control of networked robotic manipulators without task-space velocity measurements. IEEE Trans Cybern, 2016, 46: 2386–2398
14 Yuan C, Abdelatti M, Dong X, et al. Cooperative deterministic learning control of multi-robot manipulators. In: Proceedings of Chinese Control Conference, Wuhan, 2018. accepted
15 Xiao H, Li Z, Chen P C L. Formation control of leader-follower mobile robots' systems using model predictive control based on neural-dynamic optimization. IEEE Trans Ind Electron, 2016, 63: 5752–5762
16 Li Z, Xia Y, Wang D, et al. Neural network-based control of networked trilateral teleoperation with geometrically unknown constraints. IEEE Trans Cybern, 2016, 46: 1051–1064
17 Li Z, Yang C, Su C Y, et al. Decentralized fuzzy control of multiple cooperating robotic manipulators with impedance interaction. IEEE Trans Fuzzy Syst, 2015, 23: 1044–1056
18 Wang C, Hill D J. Deterministic Learning Theory for Identification, Recognition and Control. Boca Raton: CRC Press, 2009
19 Yuan C, Wang C. Persistency of excitation and performance of deterministic learning. Syst Control Lett, 2011, 60: 952–959
20 Yuan C, Wang C. Performance of deterministic learning in noisy environments. Neurocomputing, 2012, 78: 72–82
21 Yuan C Z, Wang C. Design and performance analysis of deterministic learning of sampled-data nonlinear systems. Sci China Inf Sci, 2014, 57: 032201
22 Xu B, Yang C G, Shi Z K. Reinforcement learning output feedback NN control using deterministic learning technique. IEEE Trans Neural Netw Learning Syst, 2014, 25: 635–641
23 Wang C, Hill D J. Deterministic learning and rapid dynamical pattern recognition. IEEE Trans Neural Netw, 2007, 18: 617–630
24 Chen W, Hua S, Zhang H. Consensus-based distributed cooperative learning from closed-loop neural control systems. IEEE Trans Neural Netw Learn Syst, 2015, 2: 331–345
25 Ren W, Beard R W, Atkins E M. Information consensus in multivehicle cooperative control. IEEE Control Syst Mag, 2007, 27: 71–82
26 Ni W, Cheng D. Leader-following consensus of multi-agent systems under fixed and switching topologies. Syst Control Lett, 2010, 59: 209–217
27 Wang C, Hill D J. Learning from neural control. IEEE Trans Neural Netw, 2006, 17: 130–146
28 Shilnikov L P, Shilnikov A L, Turaev D V, et al. Methods of Qualitative Theory in Nonlinear Dynamics. Singapore: World Scientific, 2001
29 Olfati-Saber R, Fax J A, Murray R M. Consensus and cooperation in networked multi-agent systems. Proc IEEE,

2007, 95: 215–233
30 Dai S L, Wang C, Luo F. Identification and learning control of ocean surface ship using neural networks. IEEE Trans Ind Inf, 2012, 8: 801–810
31 Yuan C, He H, Wang C. Cooperative deterministic learning-based formation control for a group of nonlinear uncertain mechanical systems. IEEE Trans Ind Inf, 2018, doi: 10.1109/TII.2018.2792455
32 Jadbabaie A, Jie L A, Morse A S. Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Trans Automat Contr, 2003, 48: 988–1001
33 Vicsek T, Czirók A, Ben-Jacob E, et al. Novel type of phase transition in a system of self-driven particles. Phys Rev Lett, 1995, 75: 1226–1229
34 Khalil H. Nonlinear Systems. Upper Saddle River: Prentice-Hall, 2002
35 Swaroop D, Hedrick J K, Yip P P, et al. Dynamic surface control for a class of nonlinear systems. IEEE Trans Automat Contr, 2000, 45: 1893–1899
36 Xu B, Shi Z, Yang C, et al. Composite neural dynamic surface control of a class of uncertain nonlinear systems in strict-feedback form. IEEE Trans Cybern, 2014, 44: 2626–2634