Composite Distributed Learning and Synchronization of Nonlinear Multi-Agent Systems with Complete Uncertain Dynamics

Emadodin Jandaghi, Dalton L. Stein, Adam Hoburg, Paolo Stegagno, Mingxi Zhou, Chengzhi Yuan

Abstract— **This paper addresses the problem of composite synchronization and learning control in a network of multi-agent robotic manipulator systems with heterogeneous nonlinear uncertainties under a leader-follower framework. A novel two-layer distributed adaptive learning control strategy is introduced, comprising a first-layer distributed cooperative estimator and a second-layer decentralized deterministic learning controller. The first layer is to facilitate each robotic agent's estimation of the leader's information. The second layer is responsible for both controlling individual robot agents to track desired reference trajectories and accurately identifying/learning their nonlinear uncertain dynamics. The proposed distributed learning control scheme represents an advancement in the existing literature due to its ability to manage robotic agents with completely uncertain dynamics including uncertain mass matrices. This allows the robotic control to be environment-independent which can be used in various settings, from underwater to space where identifying system dynamics parameters is challenging. The stability and parameter convergence of the closed-loop system are rigorously analyzed using the Lyapunov method. Numerical simulations validate the effectiveness of the proposed scheme.**

I. INTRODUCTION

Robotics has many applications from manufacturing to surgical procedures [1], [2], [3], [4]. However, controlling robots in space and underwater is challenging due to unpredictable robot dynamics in such environments. Also, the increasing demand for high precision and operational complexity has shifted the focus towards cooperatively utilizing multiple standard robots. This is beneficial with improved efficiency, cost reduction, and redundancy [5], [6]. To this end, numerous studies have been done to formulate diverse decentralized control strategies for coordinating multiple robotic arms [7].

Despite extensive research, one critical outstanding challenge is the management of model uncertainties which can impair the performance of distributed control systems [8]. Some studies have examined the role of uncertainties in robotic arm control [9], however, these did not extend their findings to the synchronization of multiple robots. [10] explored a virtual leader-follower strategy for robot manipulators under uncertainties and disturbances. While a high-gain observer was used for velocity estimation, this approach risks exciting unmodeled high-frequency dynamics and magnifying measurement noise. Such issues make controller implementation challenging and necessitate meticulous parameter tuning. Another drawback in the existing literature is the assumption of homogeneous system dynamics across all robots, which is often not the case in real-world applications. Meanwhile, [11] ignored system uncertainties and assumed full model knowledge. Furthermore, while some research has considered non-identical robotic systems [10], others [12] mostly focused solely on achieving adaptive tracking control, without consideration for the convergence of controller parameters to their optimal states. Previous literature has not fully tackled the learning of system uncertainties without certain assumptions. These include having a known Mass or Inertia matrix [13], and using large gains to suppress errors caused by the Coriolis and Centripetal force matrix [9]. In contrast, our work uniquely identifies the nonlinear uncertain dynamics of each robot without making any assumptions on certainty or structure of each system.

In this study, we make a significant step forward compared to previous research. This framework is completely independent of any multi-agent system attributes, rendering this method universally applicable to any nonlinear Euler-Lagrange (EL) system dynamics. We tackle the challenges of achieving synchronization control and integrating learning capabilities into multi-robot systems with heterogeneous nonlinear uncertain dynamics without any knowledge of each robotic system's nonlinearities. Our control architecture employs a fixed directed graph communication with a virtual leader with linear dynamics. The control strategy is dual-layered: The first layer focuses on cooperative estimation, allowing agents to have inter-agent communication and share leader-estimated states and system matrices. The second layer involves a decentralized adaptive learning controller to regulate each robot's state and pinpoint its unique dynamics using first-layer estimates. The second layer operates without any data sharing, allowing each local robot to implement its adaptive learning controller in a completely decentralized way. Our adaptive learning control law uses precise function approximation with Radial Basis Function (RBF) Neural Networks (NN) for the identification of systems uncertain dynamics. This enhances control over robotics in space and underwater environments, where the system dynamics are often unpredictable. We confirm the efficacy of the approach through mathematically rigorous analysis and comprehensive simulation studies.

This work is supported in part by the National Science Foundation under Grant CMMI-1952862 and CMMI-2154901.

E. Jandaghi, D. L. Stein and C. Yuan are with the Department of Mechanical, Industrial and Systems Engineering, Kingston, RI 02881, USA, emadjandaghi@uri.edu; daltonstein98@uri.edu; cyuan@uri.edu. A. Huborg and P. Stegagno are with the Department of Electrical, Computer and Biomedical Engineering, Kingston, RI 02881, USA, ajhoburg@uri.edu; pstegagno@uri.edu. M. Zhou is with the Graduate School of Oceanography, , Kingston, RI 02881, USA, mzhou@uri.edu

II. PRELIMINARIES AND PROBLEM STATEMENT

A. Notation and Graph Theory

We denote the set of real numbers as $\mathbb{R}$. $\mathbb{R}^{m\times n}$ is the set of real $m\times n$ matrices, and $\mathbb{R}^n$ is the set of real $n\times 1$ vectors. $A\otimes B$ signifies the Kronecker product of matrices A and B. Given two integers k_1 and k_2 with $k_1<k_2$, $\mathbf{I}[k_1,k_2]=\{k_1,k_1+1,\ldots,k_2\}$. For a vector $x\in\mathbb{R}^n$, its norm is defined as $|x|:=(x^Tx)^{1/2}$. For a square matrix A, $\lambda_i(A)$ denotes its i-th eigenvalue, while $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ represent its maximum and minimum eigenvalues, respectively. A directed graph $G=(V,E)$ comprises nodes in the set $V=\{1,2,\ldots,N\}$ and edges in $E\subseteq V\times V$. An edge from node i to node j is represented as (i,j), with i as the parent node and j as the child node. Node i is also termed a neighbor of node j. N_i is considered as the subset of V consisting of the neighbors of node i. A sequence of edges in G, $(i_1,i_2),(i_2,i_3),\ldots,(i_k,i_{k+1})$, is called a path from node i_1 to node i_{k+1}. Node i_{k+1} is reachable from node i_1. A directed tree is a graph where each node, except for a root node, has exactly one parent. The root node is reachable from all other nodes. A directed graph G contains a directed spanning tree if at least one node can reach all other nodes. The weighted adjacency matrix of G is a non-negative matrix $A=[a_{ij}]\in\mathbb{R}^{N\times N}$, where $a_{ii}=0$ and $a_{ij}>0 \implies (j,i)\in E$. The Laplacian of G is denoted as $L=[l_{ij}]\in\mathbb{R}^{N\times N}$, where $l_{ii}=\sum_{j=1}^{N}a_{ij}$ and $l_{ij}=-a_{ij}$ if $i\neq j$. From [14], L has one zero eigenvalue and remaining eigenvalues with positive real parts if and only if G has a directed spanning tree.

B. Radial Basis Function NNs

The RBF Neural Networks (NN) can be described as $f_{nn}(Z)=\sum_{i=1}^{N}w_is_i(Z)=W^TS(Z)$, where $Z\in\Omega_Z\subseteq\mathbb{R}^q$ and $W=w_1,...,w_N^T\in\mathbb{R}^N$ as input and weight vectors respectively. N indicates the number of NN nodes, $S(Z)=[s_1(||Z-\mu_i||),...,s_N(||Z-\mu_i||)]^T$ with $s_i(\cdot)$ is a RBF, and $\mu_i(i=1,...,N)$ is distinct points in the state space. The Gaussian function $s_i(||Z-\mu_i||)=\exp\left[-\frac{(Z-\mu_i)^T(Z-\mu_i)}{\eta_i^2}\right]$ is generally used for RBF, where $\mu_i=[\mu_{i1},\mu_{i2},...,\mu_{iN}]^T$ is the center and η_i is the width of the receptive field. The Gaussian function categorized by localized radial basis function s in the sense that $s_i(||Z-\mu_i||)\to 0$ as $||Z||\to\infty$. It has been shown in [15], for any continuous function $f(Z):\Omega_Z\to R$ where $\Omega_Z\subset R^p$ is a compact set, there exists an ideal constant weight vector W^*, such that for each $\epsilon^*>0$, $f(Z)=W^{*T}S(Z)+\epsilon(Z)$, $\forall Z\in\Omega_Z$, where $\epsilon(Z)$ is the approximation error which can be made arbitrarily small given a sufficiently large node number N. Moreover, according to [16], given any continuous recurrent trajectory $Z(t):[0,\infty)\to R^q$, $Z(t)$ remains in a bounded compact set $\Omega_Z\subset\mathbb{R}^q$, for RBF NN $W^TS(Z)$ with centers placed on a regular lattice (large enough to cover compact set Ω_Z), the regressor subvector $S_\zeta(Z)$ consisting of RBFs with centers located in a small neighborhood of $Z(t)$ is persistently exciting (PE).

C. Problem Statement

We consider a multi-robot manipulator system of N robots with heterogeneous uncertain nonlinear dynamics:

$$M_i(\theta_i)\ddot{\theta}_i+C_i(\theta_i,\dot{\theta}_i)\dot{\theta}_i+g_i(\theta_i)=\tau_i,\quad i\in\mathbf{I}[1,N],\quad (1)$$

where $\tau_i\in\mathbb{R}^n$ is the vector of input signals. The subscript i denotes the i^{th} robotic agent. For each $i\in\mathbf{I}[1,N]$, $\theta_i=[\theta_{i1},\theta_{i2},...,\theta_{in}]^T\in\mathbb{R}^n$ are the joint positions, $\dot{\theta}_i,\ddot{\theta}_i$ are the joint velocities and accelerations, respectively. $M\in\mathbb{S}_+^n$ is a positive definite mass matrix, $C\in\mathbb{R}^{n\times n}$ is the matrix containing Coriolis and Centripetal forces, $g\in\mathbb{R}^n$ is the gravity term. It is crucial to note that as opposed to our previous work [13], we make no assumptions about the certainty of any system matrices (M,C, and g) for control design. We rewrite the system dynamics as:

$$\begin{aligned}\dot{x}_{i1}&=x_{i2}\\ \dot{x}_{i2}&=M_i^{-1}(x_{i1})[\tau_i-C_i(x_i)x_{i2}-g_i(x_{i1})]\end{aligned}\quad (2)$$

where $x_{i1}=\theta_i$, $x_{i2}=\dot{\theta}_i$, and let x_i be a column vector of x_{i1} and x_{i2}. The virtual leader's dynamics is:

$$\dot{\chi}_0=A_0\chi_0 \quad (3)$$

Here, "0" marks the leader node. The leader state χ_0 is a column vector of x_{01} and x_{02}. A_0 is a constant matrix. We make two assumptions, which do not sacrifice generality:

Assumption 1. *All eigenvalues of A_0 are imaginary.*

Assumption 2. *The digraph G has a directed spanning tree rooted at node 0.*

Assumption 1 ensures all the states of the leader dynamics remain periodic and uniformly bounded. The Laplacian L of the graph can be divided as follows using Assumption 2: $L=\begin{bmatrix}\sum_{j=1}^{N}a_{0j} & -[a_{01},\ldots,a_{0N}]\\ -\Phi_{1N} & H\end{bmatrix}$. Here, Φ is a diagonal matrix and H has only positive real parts for its non-zero eigenvalues. Given the robotic systems (2) and a leader (3), with Assumptions 1-2, we aim to design such a controller that: (1) *Cooperative Synchronization:* All robots should synchronize to the leader, i.e., $\lim_{t\to\infty}(x_{i1}(t)-x_{01}(t))=0 \quad \forall i\in\mathbf{I}[1,N]$. (2) *Decentralized Learning:* Each robot should learn its own dynamics via its local adaptive controller using RBF NNs.

To this end, a two-layer framework is proposed: The first layer uses a distributed observer to estimate the leader's state and system information; The second layer uses a decentralized controller for synchronization control and dynamics learning. Only the first layer requires data sharing between nearby robots and the second layer just works on local data.

III. DISTRIBUTED ADAPTIVE CONTROL

A. First Layer: Distributed Cooperative Estimator

In a distributed control setting, the leader's information including χ_0 and A_0, may not be available to all agents. This constraint leads us to create a distributed cooperative estimator, enabling each agent to estimate the leader's state and dynamics through inter-agent collaboration:

$$\dot{\hat{\chi}}_i(t) = A_i(t)\hat{\chi}_i(t) + \beta_{i1}\sum_{j=0}^{N} a_{ij}(\hat{\chi}_j(t) - \hat{\chi}_i(t)),\ \forall i \in I[1, N] \tag{4}$$

The observer states for each robot $\hat{\chi}_i = [\hat{x}_{i1}, \hat{x}_{i2}]^T$ are used to estimate the leader's state $\chi_0 = [x_{01}, x_{02}]^T$. β_{i1} are positive design constant numbers. The time-varying system parameters $\hat{A}_i(t)$ are updated by the following for all i:

$$\dot{\hat{A}}_i(t) = \beta_{i2}\sum_{j=0}^{N} a_{ij}(\hat{A}_j(t) - \hat{A}_i(t)), \quad \forall i \in I[1, N] \tag{5}$$

$\hat{A}_i$ are used to estimate the leader's system matrix A_0 and have dimensions $n \times n$. The constants β_{i2} are all positive numbers. By defining $\tilde{\chi}_i = \hat{\chi}_i - \chi_0$ and $\tilde{A}_i = \hat{A}_i - A_0$, we obtain the error dynamics for agent i:

$$\begin{aligned}\dot{\tilde{\chi}}_i(t) &= A_0\tilde{\chi}_i(t) + \tilde{A}_i(t)\tilde{\chi}_i(t) + \tilde{A}_i(t)\chi_0(t)\\ &\quad + \beta_{i1}\sum_{j=0}^{N} a_{ij}(\tilde{\chi}_j(t) - \tilde{\chi}_i(t)), \quad \forall i \in I[1, N]\\ \dot{\tilde{A}}_i(t) &= \beta_{i2}\sum_{j=0}^{N} a_{ij} - (\tilde{A}_j(t) - \tilde{A}_i(t)), \quad \forall i \in I[1, N].\end{aligned} \tag{6}$$

Define $\tilde{\chi} = \text{col}\{\tilde{\chi}_1, \dots, \tilde{\chi}_N\}$, $\tilde{A} = \text{col}\{\tilde{A}_1, \dots, \tilde{A}_N\}$, $\tilde{A}_b = \text{diag}\{\tilde{A}_1, \dots, \tilde{A}_N\}$, $B_{\beta_1} = \text{diag}\{\beta_{11}, \dots, \beta_{N1}\}$, and $B_{\beta_2} = \text{diag}\{\beta_{12}, \dots, \beta_{N2}\}$, we can further obtain the error dynamics for the entire network:

$$\begin{aligned}\dot{\tilde{\chi}}(t) &= ((I_N \otimes A_0) - B_{\beta_1}(H \otimes I_{2n}))\tilde{\chi}(t)\\ &\quad + \tilde{A}_b(t) \otimes \tilde{\chi}(t) + \tilde{A}_b(t)(1_N \otimes \chi_0(t)),\\ \dot{\tilde{A}}(t) &= -B_{\beta_2}(H \otimes I_n)\tilde{A}(t).\end{aligned} \tag{7}$$

Theorem 1. *Given the error system* (7) *and under Assumptions 1 and 2, for all* $i \in I[1, N]$ *and any initial conditions* $\chi_0(0), \hat{\chi}_i(0), \hat{A}_i(0)$, *we have* $\lim_{t\to\infty} \tilde{A}_i(t) = 0$ *and* $\lim_{t\to\infty} \tilde{\chi}_i(t) = 0$ *exponentially.*

The proof of Theorem 1 is similar to that of [13], which will be omitted here to save some space.

B. Second Layer: Decentralized Learning Control

Given that the leader's state information χ_0 is unavailable to all robotic agents, $\hat{\chi}_i$ will serve as the tracking reference signal for control design of each local agent. The learning control objective is met when each agent's system states x_i converge to $\hat{\chi}_i$ in a decentralized manner. For stability analysis, we first state the following property of the robotic system: the matrix $\dot{M}_i(\theta_i) - 2C_i(\theta_i, \dot{\theta}_i)$ is skew-symmetric which means $\frac{1}{2}\dot{M}_i(\theta_i) = 2C_i(\theta_i, \dot{\theta}_i)$. To this end, consider the i^{th} robot with a reference signal in its controller as $\dot{x}_{ri}$ and $\ddot{x}_{ri}$, while let a filtered output signal r_i be:

$$r_i = \dot{x}_{i1} - \dot{x}_{ri} = \dot{e}_i + \lambda_i e_i \quad \forall i \in I[1, N] \tag{8}$$

where $\lambda_i > 0$ and $e_i \in \mathbb{R}^n$ is the position tracking error: $e_i = x_{i1} - \hat{x}_{i1}$. $\dot{x}_{ri2}$ is the second derivative of reference signal with:

$$M_i(x_{i1})\ddot{x}_{ri} + C_i(x_i, \dot{x}_i)\dot{x}_{ri} + g_i(x_i) = H_i(\chi_i) \tag{9}$$

where $\chi_i = col\{x, \dot{x}, \dot{x}_r, \ddot{x}_r\}$. The unknown nonlinear function $H_i(\chi_i)$ is aimed to be approximated using the RBF NN. Specifically, according to Section II-B, there exist RBF NNs $W_i^{*T}S_i(\chi_i)$ such that $H_i(\chi_i) = W_i^{*T}S_i(\chi_i) + \epsilon_i(\chi_i),\ \forall i \in I[1, N]$, with W_i^* as the ideal constant weights, and $|\epsilon_i(\chi_i)| \le \epsilon_i^*$ is the ideal approximation errors which can be made arbitrarily small given sufficiently large number of neurons. Assuming $\hat{W}_i$ as the estimate of W_i^*, we construct the decentralized learning control law as:

$$\tau_i = \hat{W}_i^T S_i(\chi_i) - K_i r_i, \quad \forall i \in I[1, N]. \tag{10}$$

where $\hat{W}_i^T S_i(\chi_i) = [\hat{W}_{i1}^T S_{i1}(\chi_i), \dots, \hat{W}_{in}^T S_{in}(\chi_i)]^T$ and $K_i \in \mathbb{R}^{n\times n}$. A robust self-adaptation law for online updating $\hat{W}_i$ is constructed as:

$$\dot{\hat{W}}_{ij} = -\Gamma_i[S_{ij}(\chi_i)r_{ij} + \sigma_i\hat{W}_{ij}],\ \forall i \in I[1, N],\ \forall j \in I[1, n]. \tag{11}$$

where Γ_i and σ_i are positive constant numbers with σ_i being very small, r_{ij} represents the j^{th} element of r_i. Substituting (9) and (10) into (8) for all $i \in I[1, N]$ yields:

$$\dot{r}_i = M_i^{-1}(x_i)(\tilde{W}_i^T S_i(\chi_i) - \epsilon_i(\chi) - K_i r_i - C_i(x_i, \dot{x}_i)r_i) \tag{12}$$

where $\tilde{W}_i := \hat{W}_i - W_i^*$.

Theorem 2. *Given systems* (11) *and* (12)*. If there exists a sufficiently large compact set* Ω_{χ_i} *such that* $\chi_i \in \Omega_{\chi_i}\ \forall i \in \mathbf{I}[1, N]$*, then for any bounded initial conditions with* $\hat{W}_i(0) = 0 (\forall i \in \mathbf{I}[1, N])$ *we have: (i) all the signals in the system remain uniformly bounded; (ii) the position tracking error* $x_{i1} - \hat{x}_{i1}$ *converges exponentially to a small neighborhood around the origin, by choosing the design parameters with* $K_i \in \mathbb{S}_+^n\ \forall i \in \mathbf{I}[1, N]$*. (iii) along the system trajectory denoted by* $\varphi(\chi_i(t))|_{t\ge T_i}$ *starting from* T_i *which represents the settling time of tracking control, the local estimated neural weights* $\hat{W}_{i\psi}$ *converge to small neighborhoods close to the corresponding ideal values* $W_{i\phi}^*$*, and locally-accurate identification of nonlinear uncertain dynamics defined in* (9) *can be obtained by* $\hat{W}_i^T S_i(\chi_i)$ *as well as* $\bar{W}_i^T S_i(\chi_i)$ *along the system trajectory* $\psi(\chi_i(t))|_{t\ge T_i}$*, where*

$$\bar{W}_i^T = mean_{t\in[t_{ia}, t_{ib}]}\hat{W}_i(t), \quad \forall i \in \mathbf{I}[1, N], \tag{13}$$

with $[t_{ia}, t_{ib}](t_{ib} > t_{ia} > T_i)$ *being a time segment after the transient period of tracking control.*

Proof. (i) Consider the following Lyaponuv function candidate for (11) and (12):

$$V_i = \frac{1}{2}r_i^T M_i(x_i)r_i + \frac{1}{2}\sum_{j=1}^{n}\tilde{W}_{ij}^T\Gamma^{-1}\tilde{W}_{ij}. \tag{14}$$

Temporal differentiation of (14) yields,

$$\dot{V}_i = r_i^T M_i(x)\dot{r}_i + \frac{1}{2}r_i^T\dot{M}_i(x)r_i + \sum_{j=1}^{n}\dot{\tilde{W}}_{ij}^T\Gamma_i^{-1}\tilde{W}_{ij}. \tag{15}$$

Exploiting (12) yields

$$\dot{V}_i = -r_i^T K_i r_i - r_i^T\epsilon(\chi_i) + r_i^T\tilde{W}_i^T S_i(\chi) + \sum_{j=1}^{n}\dot{\tilde{W}}_{ij}^T\Gamma_i^{-1}\tilde{W}_{ij}. \tag{16}$$

Utilizing $\dot{\hat{W}} = \dot{\tilde{W}}$ in (11), and substituting (11) into (16),

$$\begin{aligned}\dot{V}_i &= -r_i^T K_i r_i - r_i^T \epsilon(\chi_i) + r_i^T \tilde{W}_i^T S_i(\chi_i) \\ &\quad + \sum_{j=1}^{n} \left(-\Gamma_i [S_{ij}(\chi_i) r_{ij} + \sigma_i \hat{W}_{ij}]\right)^T \Gamma_i^{-1} \tilde{W}_{ij} \\ &= -r_i^T K_i r_i - r_i^T \epsilon(\chi_i) - \sum_{j=1}^{n} \sigma_i \hat{W}_{ij}^T \tilde{W}_{ij}\end{aligned} \tag{17}$$

Select $K_i = K_{i1} + K_{i2}$ with K_{i1} and K_{i2} being positive definite to yield:

$$\dot{V}_i = -r_i^T K_{i1} r_i - r_i^T K_{i2} r_i - r_i^T \epsilon(\chi_i) - \sum_{j=1}^{n} \sigma_i \hat{W}_i^T j \tilde{W}_{ij}. \tag{18}$$

As a result, we have for all $i \in \mathbf{I}[1, N]$ and $j \in \mathbf{I}[1, n]$:

$$\begin{aligned}-\sigma_i \hat{W}_i^T j \tilde{W}_{ij} &= -\sigma_i (\tilde{W}_{ij} + W_{ij}^*)^T \tilde{W}_{ij} \leq -\sigma_i \tilde{W}_{ij}^2 - \sigma_i \tilde{W}_{ij} W_{ij}^* \\ &\leq -\frac{\sigma_i}{2} \|\tilde{W}_{ij}\|^2 + \frac{\sigma_i}{2} \|W_{ij}^*\|^2.\end{aligned} \tag{19}$$

Using the same approach we have:

$$-r_i^T K_{i2} r_i - r_i^T \epsilon_i \leq \frac{\epsilon_i^T \epsilon_i}{4\lambda_{\min}(K_{i2})} \leq \frac{||\epsilon_i^*||^2}{4\lambda_{\min}(K_{i2})}. \tag{20}$$

Based on (19) and (20), we conclude:

$$\begin{aligned}\dot{V}_i \leq & - r_i^T K_{i1} r_i + \frac{||\epsilon_i^*||^2}{4\lambda_{\min}(K_{i2})} \\ & - \frac{1}{2} \sum_{j=1}^{n} \sigma_i \|\tilde{W}_{ij}\|^2 + \frac{1}{2} \sum_{j=1}^{n} \sigma_i \|W_{ij}^*\|^2. \quad \forall i \in I[1, N]\end{aligned} \tag{21}$$

Given that K_{i1} is positive definite, we can conclude that $\dot{V}_i$ is negative definite under the condition that:

$$\|r_i\| > \frac{\|\epsilon_i^*\|}{\sqrt{2\lambda_{\min}(K_{i1})\lambda_{\min}(K_{i2})}} + \sqrt{\frac{2\sigma_i}{\lambda_{\min}(K_{i1})}} \sum_{j=1}^{n} \|W_{ij}^*\| \tag{22}$$

$$\text{or} \quad \sum_{j=1}^{n} \|\tilde{W}_{ij}\| > \frac{\|\epsilon_i\|}{\sqrt{2\sigma_i \lambda_{\min}(K_{i2})}} + \sum_{j=1}^{n} \|W_{ij}^*\|. \quad \forall i \in I[1, N]$$

The signals r_i and $\tilde{W}_{ij}$ are bounded, leading to boundedness of $\tilde{W}_{ij}$. As a result, $\hat{W}_{ij}$ is also bounded. Given that the regressor vector $S_{ij}(\chi_i)$ is bounded according to [17], the feedback control law τ_i from (10) is bounded as well.

(ii) For the second part of the proof, we examine the Lyapunov function for the dynamics of r_i given by (12) as,

$$V_{ri} = \frac{1}{2} r_i^T M_i(x_{i1}) r_i. \tag{23}$$

Temporal differentiation of (23) and substituting in (12) for $M_i(\theta)\dot{r}_i$ yields $\dot{V}_{ri} = -r_i^T K_i r_i + r_i^T \tilde{W}_i^T S_i(\chi_i) - \epsilon_i^T r_i$. Using an approach similar to the one used for inequality (20), and let $K_i = K_{i1} + 2K_{i2}$, we can demonstrate:

$$-r_i^T K_{i2} r_i - r_i^T \epsilon_i \leq \frac{\epsilon_i^2}{4\lambda_{\min}(K_{i2})} \leq \frac{\epsilon_i^{*2}}{4\lambda_{\min}(K_{i2})}, \quad \forall i \in I[1, N] \tag{24}$$

$$-r_i^T K_{i2} r_i + r_i \tilde{W}_i^T S_i(\chi_i) \leq \frac{\tilde{W}_i^{*2} s_i^{*2}}{4\lambda_{\min}(K_{i2})}, \quad \forall i \in I[1, N]. \tag{25}$$

where $||S_i(\chi_i)|| \leq s_i^*$ for all $i \in I[1, N]$. The existence of such a s_i^* is confirmed by [17]. Therefore, we arrive at:

$$\dot{V}_{ri} \leq -r_i^T \lambda_{\min}(K_{i1}) r_i + \delta_i \leq -\rho_i V_{ri} + \delta_i, \ \forall i \in I[1, N] \tag{26}$$

where $\rho_i = \min\{2\lambda_{\min}(K_{i1}), \frac{2\lambda_{\min}(K_{i1})}{\lambda_{\max}(M_i)}\}$, $\delta_i = \frac{\delta_i^{*2}}{4\lambda_{\min}(K_{i2})} + \frac{\tilde{W}_i^{*2} s_i^{*2}}{4\lambda_{\min}(K_{i2})}$. Solving (26) gives us:

$$0 \leq V_{ri}(t) \leq V_{ri}(0) \exp(-\rho_i t) + \frac{\delta_i}{\rho_i}, \ \forall t \geq 0, \forall i \in I[1, N] \tag{27}$$

This implies that there exists a finite time $T_i > 0$ such that r_i will exponentially converge to a small area around zero. This further confirms that the tracking errors e_i will also reach a small zone around zero. The size of this zone can be minimized by carefully choosing K_{i1} with $\lambda_{\min}(K_{i1}) > 0$.

(iii) From the above proof, since $e_i = x_{i1} - \hat{x}_{i1}$, $\hat{x}_{i1}$ will eventually converge to x_{01}, and x_{01} is a periodic signal under Assumption 1, x_{i1} will also become a periodic signal after a finite time T_i. Furthermore, while e_i converges to zero, $\dot{e}_i$ converges to zero, leading x_{i2} converges to x_{02}. Since the leader dynamics is a smooth continuous LTI system, periodicity of x_{01} implies that $\dot{x}_{01}$ is also periodic and thus x_{i2} is periodic after a finite time T_i. Consequently, the inputs of RBF NNs (χ_i) are made as periodic signals for all $t \geq T_i$. According to Section II-B, the partial PE condition of the localized RBF NN regression subvector $S_{i\phi}(\chi_i)$ along the system trajectory $\phi_i(\chi_i(t)|_{t \geq T_i}$ is guaranteed, where $\hat{W}_i^T S_i(\chi_i) = \hat{W}_{i\phi}^T S_{i\phi}(\chi_i) + \hat{W}_{i\bar{\phi}}^T S_{i\bar{\phi}}(\chi_i)$. Accordingly, system dynamics of (19) can be expressed as: $\dot{r}_i = M_i^{-1}(x_{i1})(\widetilde{W}_{i\phi}^T S_{i\phi}(\chi_i) - K_i r_i - C_i(\chi_i) r_i - \epsilon'_{i\phi})$, where $\epsilon'_{i\phi} = \epsilon_i - \hat{W}_{i\bar{\phi}}^T S_{i\bar{\phi}}(\chi_i)$ is the localized ideal NN approximation error along the tracking trajectory. Thus, the overall closed-loop adaptive learning system can be described by:

$$\begin{pmatrix} \dot{r}_i \\ \dot{\widetilde{W}}_{i\phi,1} \\ \dot{\widetilde{W}}_{i\phi,2} \\ \vdots \\ \dot{\widetilde{W}}_{i\phi,n} \end{pmatrix} = \begin{pmatrix} -M_i^{-1}(x_{i1})K_i & M_i^{-1}(x_{i1}) \begin{bmatrix} S_{i\phi,1}^T(\chi_i) & 0 & 0 & 0 \\ 0 & S_{i\phi,2}^T(\chi_i) & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & S_{i\phi,n}^T(\chi_i) \end{bmatrix} \\ \begin{matrix} -\Gamma_i S_{i\phi,1}(\chi_i) \\ -\Gamma_i S_{i\phi,2}(\chi_i) \\ \vdots \\ -\Gamma_i S_{i\phi,n}(\chi_i) \end{matrix} & 0 \end{pmatrix}$$

$$\times \begin{pmatrix} r_i \\ \widetilde{W}_{i\phi,1} \\ \widetilde{W}_{i\phi,2} \\ \vdots \\ \widetilde{W}_{i\phi,n} \end{pmatrix} + \begin{pmatrix} -M_i^{-1}(x_{i1})\epsilon_{i\phi} \\ -\Gamma_i \sigma_i \widehat{W}_{i\phi,1} \\ -\Gamma_i \sigma_i \widehat{W}_{i\phi,2} \\ \vdots \\ -\Gamma_i \sigma_i \widehat{W}_{i\phi,n} \end{pmatrix}, \forall i \in I[1, N] \tag{28}$$

$$\begin{pmatrix} \dot{\widetilde{W}}_{i\bar{\phi},1} \\ \dot{\widetilde{W}}_{i\bar{\phi},2} \\ \vdots \\ \dot{\widetilde{W}}_{i\bar{\phi},n} \end{pmatrix} = \begin{pmatrix} -\Gamma_i (S_{i\bar{\phi},1}(\chi_i) r_i + \sigma_i \hat{W}_{i\bar{\phi},1}) \\ -\Gamma_i (S_{i\bar{\phi},2}(\chi_i) r_i + \sigma_i \hat{W}_{i\bar{\phi},2}) \\ \vdots \\ -\Gamma_i (S_{i\bar{\phi},n}(\chi_i) r_i + \sigma_i \hat{W}_{i\bar{\phi},n}) \end{pmatrix}, \forall i \in I[1, N] \tag{29}$$

Based on [17], the local approximation error $\epsilon'_{i\phi}$ and $\hat{W}_{i\bar{\phi}}^T S_{i\bar{\phi}}(\chi_i)$ are both small. Moreover, $\epsilon'_{i\phi}$ is proportional to ϵ_i. Studies [17] [18] have thoroughly examined the stability and convergence of the above closed-loop system. Specifically, it's established that the PE condition of $S_{i\phi}(\chi_i)$ leads to exponential convergence of $(r_i, \tilde{W}_{i\phi})$ to zero.

Further, since $\epsilon_{i\phi}$ is proportional to ϵ_i, and σ_i can be made as small as desired, $\tilde{W}_{i\phi}$ will also converge to an arbitrarily small vicinity of zero. As such, we have $H_i(\chi_i) = \hat{W}_{i\phi}^T S_{i\phi}(\chi_i) + \epsilon_{i\phi,1} = \bar{W}_{i\phi}^T S_{i\phi}(\chi_i) + \epsilon_{i\phi,2}$, $\forall i \in I[1, N]$, where $\epsilon_{i\phi,1}$ and $\epsilon_{i\phi,2}$ are approximation errors. They are proportional to $\epsilon_{i\phi}$ due to the proven convergence of $\tilde{W}_{i\phi}$ to

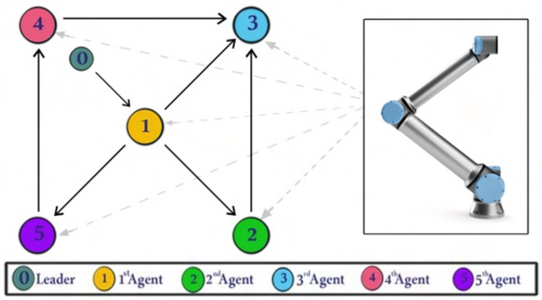

Fig. 1: Network Topology with Agent 0 as the Virtual Leader

zero. As for neurons far from the trajectory, $S_{i\bar{\phi}}(\chi_i)$ is minimal, impacting the neural weight adaptation only slightly. Therefore, the full RBF NN can still accurately approximate the unknown function $H_i(\chi_i)$ along the trajectory for $t \geq T_i$. I.e., $Hi(\chi_i) = \hat{W}_i^T S_i(\chi_i) + \epsilon_{i1} = \bar{W}_i^T S_i(\chi_i) + \epsilon_{i2}$. The approximation errors ϵ_{i1} and ϵ_{i2} are proportional to $\epsilon_{i\phi,1}$ and $\epsilon_{i\phi,2}$ respectively. This concludes the proof.

IV. SIMULATION

The multiple 2-DOF robot manipulator system, as described in (2), is considered with parameters:

$$M_i(q_i) = \begin{bmatrix} M_{i11} & M_{i12} \\ M_{i21} & M_{i22} \end{bmatrix}, C_i(q_i, \dot{q}_i) = \begin{bmatrix} C_{i11} & C_{i12} \\ C_{i21} & C_{i22} \end{bmatrix},$$

$$F_i(\dot{q}_i) = \begin{bmatrix} F_{i11} \\ F_{i21} \end{bmatrix}, g_i(q_i) = \begin{bmatrix} g_{i11} \\ g_{i21} \end{bmatrix}.$$

$$M_{i11} = m_{i1}l_{ic1}^2 + m_{i2}(l_{i1}^2 + l_{ic2}^2 + 2l_{i1}l_{ic2}\cos(qi2)) + I_{i1} + I_{i2},$$
$$M_{i12} = m_{i2}(l_{ic2}^2 + l_{i1}l_{ic2}\cos(q_{i2})) + I_{i2},$$
$$M_{i21} = m_{i2}(l_{ic2} + l_{i1}l_{ic2}\cos(qi2)) + I_{i2},\ M_{i22} = m_{i2}l_{ic2}^2 + I_{i2},$$
$$C_{i11} = -m_{i2}l_{i1}l_{ic2}\dot{q}_{i2}\sin(q_{i2}),\ G_{i22} = m_{i2}l_{ic2}g\cos(q_{i1} + q_{i2})$$
$$C_{i12} = -m_{i2}l_{i1}l_{ic2}(\dot{q}_{i1} + \dot{q}_{i2})\sin(q_{i2}),$$
$$C_{i21} = m_{i2}l_{i1}l_{ic2}\dot{q}_{i1}\sin(q_{i2}),\ C_{i22} = 0,$$
$$g_{i11} = (m_{i1}l_{ic2} + m_{i2}l_{i1})g\cos(q_{i1}) + m_{i2}l_{ic2}g\cos(q_{i1} + q_{i2}).$$

where l_{ic1}, l_{ic2} represent half of link lengths l_{i1}, l_{i2}, respectively, and F_{i11}, F_{i21} are defined as constants. The inertia of the links is given by I_{i1} and I_{i2}, and the detailed values are provided in Table I. By employing $N = 5$ manipulators, the robotic agents are set to follow the reference trajectory provided by the virtual leader. A leader dynamics is formulated to generate a periodic signal for the synchronization control:

$$\begin{bmatrix} \dot{x}_{01} \\ \dot{x}_{02} \end{bmatrix}, = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{01} \\ x_{02} \end{bmatrix}, \begin{bmatrix} x_{01}(0) \\ x_{02}(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0.8 \\ 0.8 \\ 0 \end{bmatrix}$$

The network topology G amongst the five robots is depicted in Fig 1. The system states encompass both measured signals and reference signals, totaling eight signals. If we use a neural network with 4 nodes for each signal, we end up with a massive number of nodes $4^8 = 65,536$ which is computationally expensive to train. However, the reference signals are expressed in terms of the measured signals, as shown in (8), which accentuates the significance of the measured signals. Consequently, we construct the RBF NN using the dominant four dimensions of the system state. This strategy reduces the size of the NN to $4^4 = 256$ which not only saves computational resources but also allows us

TABLE I: Parameters of the robot.

Parameter	Robot number				
	1	2	3	4	5
$m1$ (kg)	2	2.2	2.3	1.9	2.4
$m2$ (kg)	0.85	0.9	1	0.9	1.5
$l1$ (m)	0.35	0.5	0.6	0.52	0.57
$l2$ (m)	0.31	0.4	0.5	0.48	0.53
$I1 \times 10^{-3}$ (kg m^2)	61.25	70	72.14	67.21	73.42
$I2 \times 10^{-3}$ (kg m^2)	20.42	25.21	27.1	25.4	22.63

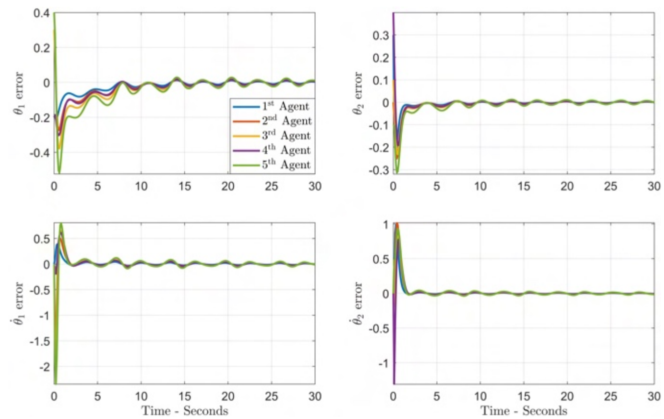

Fig. 2: Tracking error for each agent.

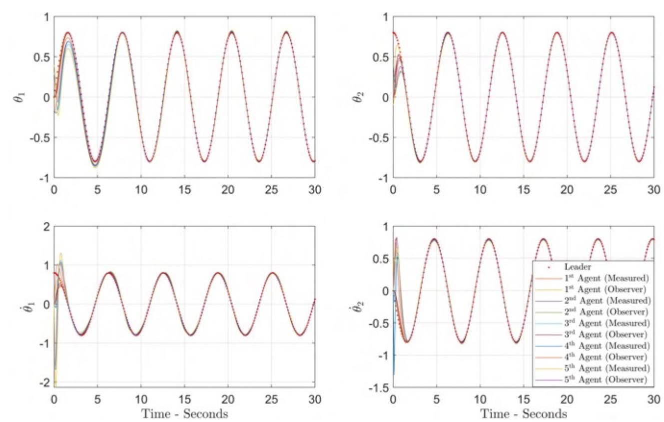

Fig. 3: Joint angle tracking response for each agent

to build a more precise model by dedicating more nodes to each of these four key dimensions. The range of each dimension lies within $[-1.2, 1.2]$, with a width parameter $\gamma_i = 0.8$. The observer and controller parameters are set to $\beta_1 = \beta_2 = 1$, $\gamma_i = 10$, $K = 10$, and $\sigma_i = 0.001$ for all $i \in [1, 5]$. The initial conditions are defined as $x_{11}(0) = [0.2\ 0.1]^T$, $x_{21}(0) = [0.3\ 0.5]^T$, $x_{31}(0) = [0.4\ 0.1]^T$, $x_{41}(0) = [0.2\ 0.5]^T$, $x_{51}(0) = [0.4\ 0.1]^T$, and $x_{i2} = [0\ 0]^T$, $\forall i \in [1, 5]$. Initial conditions for all distributed observer states $\hat{\chi}_i$ and NN weights $\hat{W}_i$ are uniformly initialized to zero. We conducted an assessment of two key components: the distributed cooperative estimator (4) and (5), and the decentralized learning control law (10) and (11). Despite varying complexities and nonlinear uncertainties among multiple robotic agents, the results indicate satisfactory tracking performance, as confirmed by Fig. 2, which shows a rapid convergence of tracking errors to zero. The tracking response of joint angles from the second layer in comparison

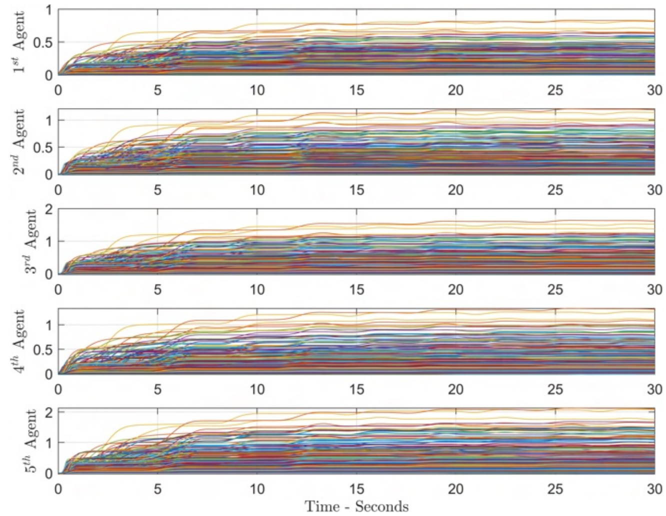

Fig. 4: NN weights Convergence for all agents.

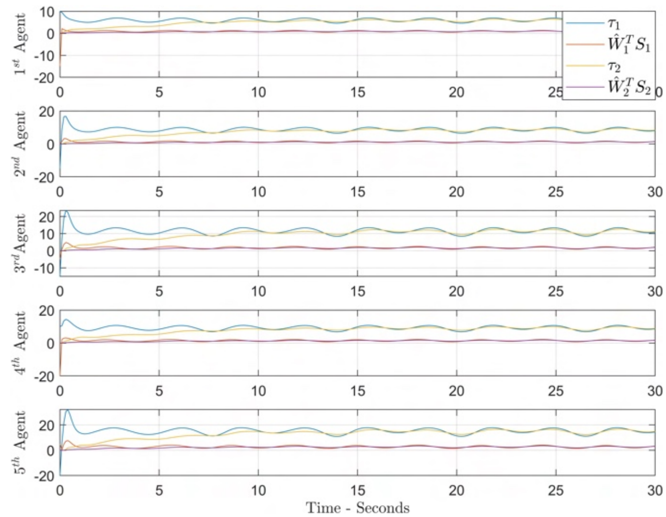

Fig. 5: NN approximation

to the desired signals originating from the first layer and the leader's signal is represented in Fig. 3. It is observed that all signals rapidly converge to align with the leader's signal within the initial seconds. Fig. 4 shows how quickly the Neural Network weights settle for all robotic agents. Moreover, we graphed the NN approximation associated with the unknown dynamic variables $H_i(\chi_i)$ for each agent in Fig. 5. In our controller, all system parameters are treated as having unstructured uncertainty, indicating that the control algorithm is environment-independent. Whether the system operates underwater, where buoyancy forces vary with the depth of a robot's arm, or in space, where the inertia matrix is unknown, the algorithm adapts accordingly. The identified system nonlinear dynamics can be saved and reused in the controller, even after the system restarts.

V. CONCLUSION

In this study, The challenge of implementing composite synchronization and adaptive learning control for a network of multi-robot manipulators, with complete nonlinear uncertain dynamics was successfully met. The novel approach contains a distributed cooperative estimator in the first layer for estimating the virtual leader's states, and a decentralized adaptive learning controller in the second layer to track the leader's states while identifying each robot's distinct nonlinear uncertain dynamics. The key strengths of This method include simultaneously achieving synchronization control and learning robots' complete nonlinear uncertain dynamics in a decentralized manner. This improves robotics control in space and underwater settings, where system dynamics are typically uncertain. The method's effectiveness is verified through detailed mathematical validation and simulations.

REFERENCES

[1] R. Cui and W. Yan, "Mutual synchronization of multiple robot manipulators with unknown dynamics," *Journal of Intelligent & Robotic Systems*, vol. 68, pp. 105–119, 2012.

[2] N. Seenu, K. C. RM, M. Ramya, and M. N. Janardhanan, "Review on state-of-the-art dynamic task allocation strategies for multiple-robot systems," *Industrial Robot: the international journal of robotics research and application*, vol. 47, no. 6, pp. 929–942, 2020.

[3] E. Jandaghi, X. Chen, and C. Yuan, "Motion dynamics modeling and fault detection of a soft trunk robot," in *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2023, pp. 1324–1329.

[4] S. Ghafoori, A. Rabiee, M. Jouaneh, and R. Abiri, "A novel seamless magnetic-based actuating mechanism for end-effector-based robotic rehabilitation platforms," 2024.

[5] C. Yuan, S. Licht, and H. He, "Formation learning control of multiple autonomous underwater vehicles with heterogeneous nonlinear uncertain dynamics," *IEEE transactions on cybernetics*, vol. 48, no. 10, pp. 2920–2934, 2017.

[6] N. Hazon and G. A. Kaminka, "On redundancy, efficiency, and robustness in coverage for multiple robots," *Robotics and Autonomous Systems*, vol. 56, no. 12, pp. 1102–1114, 2008.

[7] J. Huang, C. Wen, W. Wang, and Y.-D. Song, "Adaptive finite-time consensus control of a group of uncertain nonlinear mechanical systems," *Automatica*, vol. 51, pp. 292–301, 2015.

[8] M. Abdelatti, C. Yuan, W. Zeng, and C. Wang, "Cooperative deterministic learning control for a group of homogeneous nonlinear uncertain robot manipulators," *Science China Information Sciences*, vol. 61, pp. 1–19, 2018.

[9] Q. Liu, D. Li, S. S. Ge, R. Ji, Z. Ouyang, and K. P. Tee, "Adaptive bias rbf neural network control for a robotic manipulator," *Neurocomputing*, vol. 447, pp. 213–223, 2021.

[10] A. Rodriguez-Angeles and H. Nijmeijer, "Mutual synchronization of robots via estimated state feedback: a cooperative approach," *IEEE Transactions on control systems technology*, vol. 12, no. 4, pp. 542–554, 2004.

[11] S.-J. Chung and J.-J. E. Slotine, "Cooperative robot control and concurrent synchronization of lagrangian systems," *IEEE transactions on Robotics*, vol. 25, no. 3, pp. 686–700, 2009.

[12] H. Wang, "Flocking of networked uncertain euler–lagrange systems on directed graphs," *Automatica*, vol. 49, no. 9, pp. 2774–2779, 2013.

[13] X. Dong, C. Yuan, P. Stegagno, W. Zeng, and C. Wang, "Composite cooperative synchronization and decentralized learning of multi-robot manipulators with heterogeneous nonlinear uncertain dynamics," *Journal of the Franklin Institute*, vol. 356, no. 10, pp. 5049–5072, 2019.

[14] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on automatic control*, vol. 50, no. 5, pp. 655–661, 2005.

[15] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural computation*, vol. 3, no. 2, pp. 246–257, 1991.

[16] C. Wang and D. J. Hill, *Deterministic learning theory for identification, recognition, and control*. CRC Press, 2018.

[17] ——, *Deterministic learning theory for identification, recognition, and control*. CRC Press, 2009.

[18] C. Yuan and C. Wang, "Persistency of excitation and performance of deterministic learning," *Systems & control letters*, vol. 60, no. 12, pp. 952–959, 2011.