Scilight

Intelligence & Control
https://www.sciltp.com/journals/ic



Article

Check for updates

Aggregative Game-Driven Neurodynamic Approach for Hierarchical Stringent Multiple Manipulators Fuzzy Control

Yuqiao Chen [1,2], Jingxin Liu [1,3,*], Lijun Wang [1,3], Yuxiang Li [1,3] and Jingwen Tu [1,3]

[1] Chongqing Research Institute of Intelligent Mathematics and Autonomous Intelligence (CQRI-IM·AI*), Chongqing 401331, China
[2] School of Mechanical and Intelligent Manufacturing, Chongqing University of Science and Technology, Chongqing 401331, China
[3] School of Mathematical and Physical Sciences, Chongqing University of Science and Technology, Chongqing 401331, China
* Correspondence: liujx217@gmail.com

How To Cite: Chen, Y.; Liu, J.; Wang, L.; et al. Aggregative Game-Driven Neurodynamic Approach for Hierarchical Stringent Multiple Manipulators Fuzzy Control. *Intelligence & Control* **2026**, *2*(4), 6. https://doi.org/10.53941/ic.2026.100020

Received: 25 July 2026
Revised: 18 September 2026
Accepted: 12 October 2026
Published: 17 November 2026

Abstract: This paper investigates a class of multiple redundant manipulators collaborative operation with comprehensive constraint tasks from the perspective of aggregative game, and the aggregated information in the game model is not accessible to the participants (players). We incorporate each participant's estimation of the aggregated information to guide the creation of the algorithm. Since each manipulator only transmits the aggregated kinematic information, this aggregative game model reduces the communication cost of multiple manipulators network. The proposed neurodynamic algorithm is capable of handling tasks such as trajectory tracking, obstacle avoidance and posture maintenance, and can also optimize the rarely-discussed self-adjustment. In view of the unknowability of the kinematics of the manipulator, we introduce a neural operator based on fuzzy association rules to approximate the error. The convergence of the neurodynamic algorithm for generalized Nash equilibrium (GNE) is provable, and it can achieve distributed optimal control of the multiple manipulators.

Keywords: multiple redundant manipulators; aggregative game; neurodynamic algorithm; fuzzy association rules; generalized Nash equilibrium (GNE)

1. Introduction

In real applications, a single redundant manipulator may be unable to complete complex component or fragment assembly efficiently [1]. Multiple redundant-manipulator systems are therefore increasingly used in complex manufacturing, logistics sorting, space operations, and related fields [2]. Such systems must usually complete several control tasks simultaneously, including trajectory tracking [3], posture preservation [4], and obstacle avoidance [5]. These tasks interact and often have strict priority relationships. Constrained motion-planning studies in dynamic environments further show the importance of representing environmental interactions during autonomous operation [6]. Centralized control can coordinate the complete system, but it requires global information and is vulnerable to failure of the central processing unit. Distributed methods are consequently important for multiple-manipulator control [7]. Beyond manipulator-specific control, distributed algorithms are also employed to address time-varying general unbalanced network optimization [8] and multiblock separable optimization [9]. These studies provide algorithmic foundations for networked constrained control. Liu et al. developed a neural-network-based distributed method for fault-tolerant control of networked redundant manipulators [10], while Zhang et al. proposed a distributed scheme that improves communication efficiency in collaborative robot-arm systems [11]. However, these distributed methods do not jointly address task priorities and uncertain kinematics [12], and they do not resolve how aggregate-dependent decisions should be made when the aggregate is unavailable to individual manipulators.

In collaborative control of multiple redundant manipulators, the hierarchy among control tasks is essential for safe and reliable operation [13]. Obstacle avoidance has the highest priority because the distance between a

Copyright: © 2026 by the authors. This is an open access article under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).
Publisher's Note: Scilight stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

manipulator and nearby obstacles or other manipulators must remain safe [5]. Posture preservation then maintains the desired end-effector orientation and helps avoid singular configurations or joint limits [4]. Subject to these two tasks, trajectory tracking should use the remaining motion to follow the desired end-effector path accurately [14]. Null-space projection is a standard tool for implementing this hierarchy. Obstacle avoidance is adjusted through a potential-field construction, posture preservation is represented through a Jacobian-dependent transformation, and trajectory tracking is executed in the main task space [15,16]. This design prevents lower-priority commands from changing higher-priority motion. When tasks conflict, a lower-priority command can be relaxed or suspended while the safety constraint remains active [17]. After the first three tasks are satisfied, the remaining redundant degrees of freedom can redistribute selected joint velocities across the network. This local-joint self-adjustment is intended to make manipulator motion smoother and to moderate unnecessary joint motion or energy consumption [18].

The distributed and real-time characteristics of collaborative manipulator control make dynamic neural networks attractive because their differential equations can be evaluated in parallel and updated online [19]. Tan et al. combined a metaheuristic method with a recurrent neural network (RNN) to solve the manipulability-optimization problem of redundant manipulators [20]. Zhang et al. addressed perturbed manipulability optimization through a distributed RNN [21]. Kong et al. designed a zeroing neural network with fuzzy parameters for collaborative planning of multiple redundant manipulators [22]. Liu et al. used neural networks for fault-tolerant control of networked flexible manipulators [10]. Jin et al. exploited neural dynamics to study manipulator kinematics and distributed cooperative control with time delays [7], whereas Izadbakhsh et al. constructed differential-equation approximators for collaborative multi-manipulator control [23]. Together, these studies motivate a neurodynamic optimization algorithm that can learn uncertain kinematics while updating distributed control decisions.

Another challenge is to coordinate independent manipulators toward a collective objective while they share resources and environmental constraints. Individual malfunctions and mutually exclusive operations make game-theoretic coordination a natural formulation [24,25]. In a game-based controller, each manipulator acts as a decision maker and the coupled physical requirements define shared constraints. Existing game-based multi-manipulator methods commonly require each agent to retain or predict the complete action profile of the other manipulators [21,26,27]. The resulting local information dimension grows with the network size and can hinder scalability. In an aggregative game, by contrast, each player's cost depends on a low-dimensional statistic of all players' actions rather than on every individual action [28]. Distributed generalized Nash equilibrium (GNE) seeking under asynchrony and delays further supports the treatment of coupled constraints in game-theoretic decision making [29].

Despite these advances, existing game-based multi-manipulator control methods do not simultaneously resolve three issues. First, the aggregate action required by each local cost may be unavailable to an individual manipulator and must be estimated through the communication network. Second, obstacle avoidance, posture preservation, trajectory tracking, and network-level self-adjustment must be imposed in a strict physical hierarchy rather than as interchangeable constraints. Third, uncertain local kinematics must be learned online without exchanging private kinematic parameters. The unresolved problem is therefore to combine aggregate estimation, hierarchical physical constraints and online kinematic learning while separating local information requirements from protocol-dependent communication traffic. Because the manipulators operate cooperatively, their collaborative system can be represented as a coalition of players in an aggregative game, with each player using a local estimate of the aggregate action to construct its decision.

- For incomplete cooperation among multiple manipulators, we formulate an aggregative game in which every manipulator uses a local estimate of the aggregate action rather than retaining the complete action profile. This representation supplies the aggregate information required by the neurodynamic controller while reducing the local information dimension. The transmitted payload remains protocol dependent.
- The considered constraint tasks related to the multiple manipulators are comprehensive, including trajectory tracking, obstacle avoidance, posture preservation, and self-adjustment. The adjustment of local joints being discussed is an issue that is often overlooked. It aims to enhance the smoothness and energy efficiency of the manipulator's movement. The first three levels of tasks can be accomplished in the main space, while the self-adjustment can be realized in the remaining null space. The unknowability in kinematics is handled by neural operators with fuzzy association rules.
- A fuzzy-rule-based neural operator updates the unknown kinematic parameters within the projected dynamics for constrained equilibrium seeking. The same dynamics combine aggregate estimation, local feasibility projection, and the multipliers associated with the physical constraints.
- The designed neurodynamic algorithm has excellent approximation capabilities, which can minimize the tracking error to the greatest extent and enhance the distributed controllability of the manipulators. Meanwhile, it can search for the optimal control decision and GNE in the case of multiple levels of complex constraints.

The remaining parts of this article are organized as follows. Section 2 formulates the control problem. Section 3 presents the neurodynamic algorithm and its convergence analysis. Section 4 reports the simulations, and Section 5 gives the conclusions.

2. Problem Description

Notation 1. *Let $\theta_i = [\theta_{i\iota}]_{\iota=1}^{a} = [\theta_{i1}, \ldots, \theta_{ia}]^{\mathrm{T}} \in \mathbb{R}^a$ denote the joint-angle vector of manipulator i, where $a \geq 7$ is the number of degrees of freedom. Its end-effector position $p_i \in \mathbb{R}^3$ satisfies $p_i(\theta_i, \psi_i) = M_i(\theta_i)\psi_i$, where $M_i(\theta_i) \in \mathbb{R}^{3\times b}$ is the kinematic regression matrix. The vector $\psi_i \in \mathbb{R}^b$ contains the kinematic parameters formed from the link offsets and lengths. The end-effector velocity is $\dot{p}_i = \dot{M}_i(\theta_i, \dot{\theta}_i)\psi_i$, where $\dot{M}_i(\theta_i, \dot{\theta}_i) = \sum_{\iota=1}^{a} H_{i\iota}\dot{\theta}_{i\iota}$ and $H_{i\iota} = \partial M_i(\theta_i)/\partial\theta_{i\iota}$, $\dot{\theta}_i = [\dot{\theta}_{i\iota}]_{\iota=1}^{a}$. The communication graph is undirected and connected. Manipulator j is a neighbor of manipulator i when the two agents can exchange controller information.*

2.1. Learning-Based Trajectory Tracking

Because the kinematic parameters are unknown, manipulator i maintains the estimate $\bar{\psi}_i = [\bar{\psi}_{i\iota}]_{\iota=1}^{b}$ of the true vector ψ_i. The trajectory-tracking condition is

$$\bar{J}_i(\theta_i, \bar{\psi}_i)\dot{\theta}_i = \dot{p}_{\mathrm{d}i} - \rho(p_i - p_{\mathrm{d}i}), \tag{1}$$

where $p_{\mathrm{d}i}$ and $\dot{p}_{\mathrm{d}i}$ are the desired end-effector position and velocity, respectively. The scalar $\rho \in \mathbb{R}_+$ is the feedback coefficient, and $\bar{J}_i(\theta_i, \bar{\psi}_i) = \sum_{\iota=1}^{b} S_{i\iota}(\theta_i)\bar{\psi}_{i\iota} \in \mathbb{R}^{3\times a}$ is the estimate of the Jacobian matrix $J_i(\theta_i, \psi_i)$. Substitution of the parameter estimate gives

$$\sum\nolimits_{\iota=1}^{b} S_{i\iota}(\theta_i)\bar{\psi}_{i\iota}\dot{\theta}_i = \dot{p}_{\mathrm{d}i} - \rho(M_i(\theta_i)\psi_i - p_{\mathrm{d}i}), \tag{2}$$

where $S_{i\iota}(\theta_i) = \partial J_i(\theta_i, \psi_i)/\partial\psi_{i\iota}$. As $\bar{\psi}_i$ approaches ψ_i, Equation (2) approaches the nominal trajectory-tracking relation.

2.2. Posture Preservation

Posture preservation complements end-effector position control. For manipulator i, the rotational transformation between the end-effector and base frames is $v_{\mathrm{B}i} = R_i^{\mathrm{T}}(\theta_i)v_{\mathrm{E}i}$, where $v_{\mathrm{B}i}$ and $v_{\mathrm{E}i}$ express the same vector in the two frames. Setting $v_{\mathrm{E}i} = [0, 0, 1]^{\mathrm{T}}$ fixes the end-effector direction along its z-axis and gives $v_{\mathrm{B}i} = [\mathrm{r}_{31}, \mathrm{r}_{32}, \mathrm{r}_{33}]^{\mathrm{T}}$. We specify the end-effector orientation through the two-dimensional vector $v_i(\theta_i) = [\mathrm{r}_{31}, \mathrm{r}_{32}]^{\mathrm{T}}$. The equality constraint for maintaining the direction can be expressed as

$$J_{\mathrm{v}i}(\theta_i)\dot{\theta}_i = \dot{v}_{i0} - \epsilon(v_i - v_{i0}), \epsilon \in \mathbb{R}_{++}, \tag{3}$$

where $J_{\mathrm{v}i}(\theta_i) = \partial v_i(\theta_i)/\partial\theta_i \in \mathbb{R}^{2\times a}$ is the orientation Jacobian and ϵ is the corresponding feedback coefficient. The vectors v_{i0} and $\dot{v}_{i0}$ denote the orientation reference and its time derivative.

2.3. Obstacle Avoidance

Each manipulator must avoid obstacles in the shared workspace. Let $\mathrm{C}_i = [x_{\mathrm{c}i}, y_{\mathrm{c}i}, z_{\mathrm{c}i}]^{\mathrm{T}}$ be the point on manipulator i closest to the nearest obstacle $\mathrm{O}_i = [x_{\mathrm{o}i}, y_{\mathrm{o}i}, z_{\mathrm{o}i}]^{\mathrm{T}}$. The following inequality regulates motion toward that obstacle:

$$\mathrm{Vec}_{\mathrm{C}_i\mathrm{O}_i} J_{\mathrm{c}i}(\theta_i)\dot{\theta}_i \leq h_i \tag{4}$$

where $\mathrm{Vec}_{\mathrm{C}_i\mathrm{O}_i} = \mathrm{sgn}([x_{\mathrm{o}i} - x_{\mathrm{c}i}, y_{\mathrm{o}i} - y_{\mathrm{c}i}, z_{\mathrm{o}i} - z_{\mathrm{c}i}]^{\mathrm{T}})$ and $\mathrm{sgn}(\cdot)$ is the sign operator. $J_{\mathrm{c}i} : \mathbb{R}^a \mapsto \mathbb{R}^{3\times a}$ is the Jacobian matrix related to the critical location C_i. The deceleration term prevents an abrupt stop when an obstacle enters the safety region. It is parameterized by h_i, defined as $h_i = \mathcal{C}(d_i)\max\{\mathrm{Vec}_{\mathrm{C}_i\mathrm{O}_i} J_{\mathrm{c}i}(\theta_i)\dot{\theta}_i|_{d_i=\bar{d}}, 0\}$. Here, $\mathcal{C}(d_i)$ is the smoothing function

$$\mathcal{C}(d_i) = \begin{cases} 1, & d_i \geq \bar{d}, \\ \sin^2\left(\frac{\pi}{2} \cdot \frac{d_i - \underline{d}}{\bar{d} - \underline{d}}\right), & \underline{d} \leq d_i \leq \bar{d}, \\ 0, & d_i \leq \underline{d}, \end{cases} \tag{5}$$

which keeps the joint velocity continuous. The distances $\bar{d}$ and $\underline{d}$ are the outer activation threshold and hard safety threshold, respectively. As shown in Figure 1, obstacles beyond $\bar{d}$ do not activate the correction. As the distance approaches $\underline{d}$, motion toward the obstacle decreases to zero and then reverses.

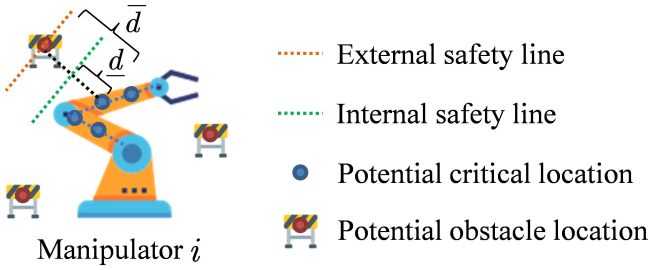


Figure 1. Diagram of safe distance between obstacle and critical locations.

2.4. Local Joint Velocity-Based Self-Adjustment

The controller first satisfies the higher-priority position, posture, and safety commands, then uses the remaining redundancy to redistribute selected joint velocities across the network. This redistribution can compensate when one arm approaches an undesirable joint motion or contributes less to the collective task. The coupling constraint specifies the desired network-level sum of selected joint velocities:

$$\sum_{i=1}^{N} A_i \dot{\theta}_i = \sum_{i=1}^{N} \dot{\mathscr{R}}_{\mathrm{d}i}, \dot{\mathscr{R}}_{\mathrm{d}i} \in \mathbb{R}^4, \tag{6}$$

where $\dot{\mathscr{R}}_{\mathrm{d}i}$ is the desired selected-joint velocity and $A_i \in \mathbb{R}^{4\times a}$ is a binary selection matrix. The matrices A_i identify which joints can absorb the residual motion. Selected joints on the other manipulators can compensate for a deviation from one manipulator, while the null-space projection preserves the higher-priority commands. Figure 2 illustrates this network-level balance. The constraint coordinates redundant motion but does not provide a complete actuator-fault recovery guarantee.

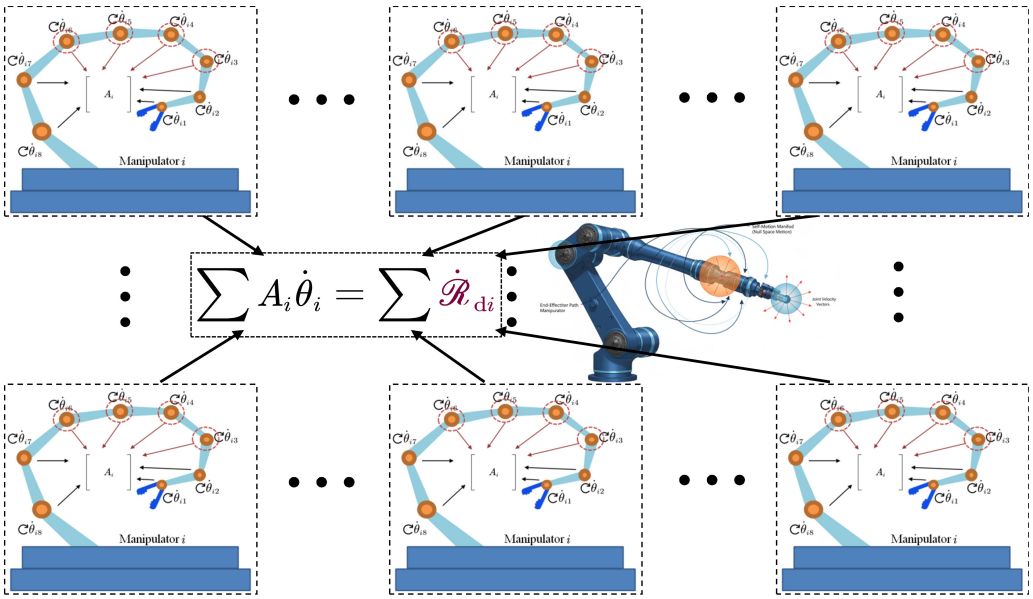


Figure 2. Schematic of self-regulation through local joint velocities in a system of redundant manipulators.

Incomplete cooperation can arise when a manipulator malfunctions or deviates from the collective objective. We therefore treat each manipulator as a player. Because each player's cost depends on a summary of the network action, the coordinated control problem takes the following aggregative-game form:

$$\begin{aligned} \min \ & \mathcal{J}_i(\dot{\theta}_i, \sigma(\dot{\theta})), \\ \text{s.t.} \ & J_i(\theta_i, \psi_i)\dot{\theta}_i = q_i, \\ & J_{\mathrm{v}i}(\theta_i)\dot{\theta}_i = q_{\mathrm{v}i}, \\ & \sum_{i=1}^{N} A_i \dot{\theta}_i = \sum_{i=1}^{N} \dot{\mathscr{R}}_{\mathrm{d}i}, \\ & \eta_i \dot{\theta}_i \le h_i, \ \dot{\theta}_i \in \Omega_i, \end{aligned} \tag{7}$$

where $\sigma(\dot{\theta})$ is the aggregate-action function, with $\dot{\theta} = [\dot{\theta}_i]_{i=1}^N \in \mathbb{R}^{Na}$, $\eta_i = \mathrm{Vec}_{\mathrm{C}_i\mathrm{O}_i} J_{\mathrm{c}i}(\theta_i) \in \mathbb{R}^{1\times a}$, $q_i = \dot{p}_{\mathrm{d}i} - \rho(p_i - p_{\mathrm{d}i})$, $q_{\mathrm{v}i} = \dot{v}_{i0} - \epsilon(v_i - v_{i0})$. For the joint decision $\dot{\theta}_{-i}$ of the other players, player i solves

$$\min \ \bar{\mathcal{J}}_i(\dot{\theta}_i, \dot{\theta}_{-i}), \tag{8}$$

where $\dot{\theta}_{-i} = [\dot{\theta}_j]_{j=1,j\neq i}^N$. The corresponding generalized Nash equilibrium (GNE) $\dot{\theta}^* = [\dot{\theta}_i^*]_{i=1}^N$ satisfies $\dot{\theta}_i^* \in \arg\min_{\dot{\theta}_i} \bar{\mathcal{J}}_i(\dot{\theta}_i, \dot{\theta}_{-i}^*)$, and can be characterized by the following variational inequality (VI) [30]:

$$\mathrm{VI}(\mathcal{F}, \Theta): \ (\dot{\theta} - \dot{\theta}^*)^{\mathrm{T}} \mathcal{F}(\dot{\theta}^*) \geq 0, \ \forall \dot{\theta} \in \Theta, \tag{9}$$

where $\mathcal{F}(\cdot)$ is the pseudo-gradient operator and $\mathcal{F}(\dot{\theta}^*) = [\nabla_{\dot{\theta}_i} \bar{\mathcal{J}}_i(\dot{\theta}_i^*, \dot{\theta}_{-i}^*)]_{i=1}^N = [\nabla_{\dot{\theta}_i} \mathcal{J}_i(\dot{\theta}_i^*, \sigma(\dot{\theta}^*))]_{i=1}^N$, Θ represents the nonempty constraint set of problem (7). For each manipulator i, the cost $\bar{\mathcal{J}}_i$ is continuously differentiable and convex in $\dot{\theta}_i$ for any fixed $\dot{\theta}_{-i}$, and the local constraint set $\Omega_i = \{\dot{\theta}_i \in \mathbb{R}^a : \underline{\dot{\theta}}_i \leq \dot{\theta}_i \leq \bar{\dot{\theta}}_i\}$ imposes componentwise lower and upper bounds on $\dot{\theta}_i$. The equality constraints $J_i(\theta_i, \psi_i)\dot{\theta}_i = q_i$, $J_{\mathrm{v}i}(\theta_i)\dot{\theta}_i = q_{\mathrm{v}i}$ and inequality constraint $\eta_i \dot{\theta}_i \leq h_i$ are handled in the main task space. The coupled constraint $\sum_{i=1}^N A_i \dot{\theta}_i = \sum_{i=1}^N \dot{\mathscr{R}}_{\mathrm{d}i}$ is enforced in the residual null space.

Remark 1. *The VI combines the individual game gradients with the shared feasible set. In Lemma 1, the multipliers impose the tracking, posture, safety, and self-adjustment constraints, while the projection enforces the joint-velocity bounds. A fixed point is therefore a feasible velocity profile under which no manipulator can reduce its cost through a unilateral feasible change.*

Lemma 1. *For a feasible action decision $\dot{\theta}_i^*$ of the manipulator i, if there exist multipliers $(\mu_i^*, \mu_{vi}^*, \nu_i^*, \lambda_i^*)$ such that*

$$\begin{aligned} \dot{\theta}_i^* =& \mathcal{P}_{\Omega_i}[\dot{\theta}_i^* - \nabla_{\dot{\theta}_i} \mathcal{J}_i(\dot{\theta}_i^*, \sigma(\dot{\theta}^*)) + J_i^{\mathrm{T}} \mu_i^* + J_{vi}^{\mathrm{T}} \mu_{vi}^* \\ & + A_i^{\mathrm{T}} \nu_i^* - \eta_i^{\mathrm{T}} \mathcal{P}_{\mathbb{R}_{\geq 0}}(\lambda_i^* + \eta_i \dot{\theta}_i^* - h_i)], \end{aligned} \tag{10a}$$

$$\lambda_i^* = \mathcal{P}_{\mathbb{R}_{\geq 0}}(\lambda_i^* + \eta_i \dot{\theta}_i^* - h_i), \tag{10b}$$

hold, then $\dot{\theta}_i^*$ is a GNE of the aggregative game (7).

Proof. Feasibility of $\dot{\theta}_i^*$ implies that $J_i \dot{\theta}_i^* = q_i$, $J_{\mathrm{v}i} \dot{\theta}_i^* = q_{\mathrm{v}i}$, and $\sum_{i=1}^N A_i \dot{\theta}_i^* = \sum_{i=1}^N \dot{\mathscr{R}}_{\mathrm{d}i}$ hold true. We use the sufficient direction of the variational-inequality KKT conditions (Equation (10.1.2) in [30]). The projection in (10b) gives $\lambda_i^* \geq 0$ and $\lambda_i^*(\eta_i \dot{\theta}_i^* - h_i) = 0$. Substituting (10b) into (10a) and using the projection property gives

$$-\nabla_{\dot{\theta}_i} \mathcal{J}_i(\dot{\theta}_i^*, \sigma(\dot{\theta}^*)) + J_i^{\mathrm{T}} \mu_i^* + J_{vi}^{\mathrm{T}} \mu_{vi}^* + A_i^{\mathrm{T}} \nu_i^* - \eta_i^{\mathrm{T}} \lambda_i^* \in \mathscr{N}_{\Omega_i}(\dot{\theta}_i^*).$$

For any feasible unilateral change z_i with the other decisions fixed at $\dot{\theta}_{-i}^*$, let $d_i = z_i - \dot{\theta}_i^*$. The equality constraints give $J_i d_i = J_{\mathrm{v}i} d_i = A_i d_i = 0$. Feasibility and complementarity give $\lambda_i^* \eta_i d_i \leq 0$. Taking the inner product of the normal-cone inclusion with d_i therefore gives

$$\left\langle \nabla_{\dot{\theta}_i} \mathcal{J}_i(\dot{\theta}_i^*, \sigma(\dot{\theta}^*)), d_i \right\rangle \geq -\lambda_i^* \eta_i d_i \geq 0.$$

By convexity in the unilateral decision, $\dot{\theta}_i^*$ minimizes player i's cost over its feasible unilateral decisions. This holds for every player, so $\dot{\theta}^*$ is a GNE of game (7). □

3. Main Results

To seek a GNE of problem (7) with unknown kinematic parameters, we first define the estimation-error objective

$$\min \ \mathrm{Err}(\bar{\psi}) = \sum_{i=1}^N \frac{1}{2} \| \dot{p}_i(\bar{\psi}_i) - \dot{p}_i + \varsigma(p_i(\bar{\psi}_i) - p_i) \|^2, \tag{11}$$

where $\varsigma \in \mathbb{R}_+$ is a weighting parameter. The objective vanishes when the estimated position and velocity match their measured values. We use the fuzzy-gain neural operator

$$\dot{\bar{\psi}}_i = -\Delta \gamma_i (\tilde{\dot{M}}_i + \varsigma M_i)^{\mathrm{T}} (\tilde{\dot{M}}_i \bar{\psi}_i - \dot{p}_i + \varsigma(M_i \bar{\psi}_i - p_i)), \tag{12}$$

to minimize (11), where $\Delta\gamma_i \in [0,1]$ is the adaptive learning gain. $\tilde{\dot{M}}_i = \dot{M}_i(\theta_i, \dot{\theta}_i + \varrho_i)$ is the perturbed kinematic-regression derivative, and $\varrho_i \in \mathbb{R}^a$ is an independent zero-mean perturbation. Table 1 defines $\Delta\gamma_i$ from the inputs $\upsilon_1 = \|M_i\bar{\psi}_i - p_i\| \in [0, 0.05]$ and $\upsilon_2 = \|\tilde{\dot{M}}_i\bar{\psi}_i - \dot{p}_i\| \in [0, 0.05]$. The input labels are small (S), medium (M), and large (L); the output labels additionally include very small (VS) and very large (VL). Centroid defuzzification converts the output label into a scalar gain [31].

Table 1. Fuzzy association rules.

$\boldsymbol{\Delta\gamma_i}$		$\boldsymbol{\upsilon_1}$		
		$\mathrm{S}_{\upsilon_1}:[0, 0.02)$	$\mathrm{M}_{\upsilon_1}:[0.02, 0.04)$	$\mathrm{L}_{\upsilon_1}:[0.04, 0.05]$
υ_2	$\mathrm{S}_{\upsilon_2}:[0, 0.02)$	VS: $[0, 0.01)$	S: $[0.01, 0.3)$	M: $[0.3, 0.7)$
	$\mathrm{M}_{\upsilon_2}:[0.02, 0.04)$	S: $[0.01, 0.3)$	M: $[0.3, 0.7)$	L: $[0.7, 0.9)$
	$\mathrm{L}_{\upsilon_2}:[0.04, 0.05]$	M: $[0.3, 0.7)$	L: $[0.7, 0.9)$	VL: $[0.9, 1]$

For the simulations, both inputs are clipped to $[0, 0.05]$ and evaluated with overlapping triangular memberships centred at 0, 0.025, and 0.05, each with half-width 0.025. Rule firing uses the minimum operator. The output centres for VS, S, M, L, and VL are 0.005, 0.20, 0.50, 0.80, and 0.95, respectively, and their firing-strength-weighted centroid gives $\Delta\gamma_i$.

Remark 2. *Larger residuals υ_1 and υ_2 produce a larger $\Delta\gamma_i$ and faster adaptation. Small residuals reduce the gain. When both residuals are sufficiently small, $\Delta\gamma_i$ may reach zero and stop the update, that is, $\dot{\bar{\psi}}_i = 0$. The learned parameters then remain available for position and Jacobian estimation [4].*

Define the parameter error $\varepsilon_i = \bar{\psi}_i - \psi_i$. For the neural solver (12), assume that the composite kinematic regressor remains sufficiently informative and satisfies the nonsingularity condition used in the proof, and that the fuzzy gain does not vanish before the composite residual reaches zero.

Theorem 1. *Under the neural solver (12), $\varepsilon_i \to \mathbf{0}$. Consequently, $M_i\bar{\psi}_i$ approximates p_i and $\sum_{\iota=1}^{b} S_{i\iota}(\theta_i)\bar{\psi}_{i\iota}$ approximates $J_i(\theta_i, \psi_i)$ as the parameter error tends to zero.*

Proof. Define the Lyapunov function $V_{\bar{\psi}_i} = \frac{1}{2}\|\bar{\psi}_i - \psi_i\|^2$. Along the neural solver (12), its derivative is

$$\begin{aligned}\dot{V}_{\bar{\psi}_i} &= (\bar{\psi}_i - \psi_i)^{\mathrm{T}}\dot{\bar{\psi}}_i \\ &= -\Delta\gamma_i(\bar{\psi}_i - \psi_i)^{\mathrm{T}}(\tilde{\dot{M}}_i + \varsigma M_i)^{\mathrm{T}}(\tilde{\dot{M}}_i + \varsigma M_i)(\bar{\psi}_i - \psi_i) \\ &= -\Delta\gamma_i\|(\tilde{\dot{M}}_i + \varsigma M_i)\varepsilon_i\|^2 \le 0.\end{aligned}$$

At steady state, either $\Delta\gamma_i = 0$ or $(\tilde{\dot{M}}_i + \varsigma M_i)\varepsilon_i = \mathbf{0}$. The stated gain condition excludes the first case while the composite residual is nonzero. Write $\tilde{\dot{M}}_i = \dot{M}_i + \dot{M}_{\varrho_i}$, where $\dot{M}_i = \sum_{\iota=1}^{a} H_{i\iota}\dot{\theta}_{i\iota}$ and $\dot{M}_{\varrho_i} = \sum_{\iota=1}^{a} H_{i\iota}\varrho_{i\iota}$, with $\varrho_{i\iota}$ the ι-th component of ϱ_i. Then $\mathbb{E}(\dot{M}_{\varrho_i}) = \sum_{\iota=1}^{a}\mathbb{E}(H_{i\iota})\mathbb{E}(\varrho_{i\iota}) = \mathbf{0}$, using the factorization of the expectation. The steady-state relation $(\tilde{\dot{M}}_i + \varsigma M_i)\varepsilon_i = \mathbf{0}$ gives $\mathbb{E}(\varepsilon_i^{\mathrm{T}}(\tilde{\dot{M}}_i + \varsigma M_i)^{\mathrm{T}}(\tilde{\dot{M}}_i + \varsigma M_i)\varepsilon_i) = \mathbb{E}(\varepsilon_i^{\mathrm{T}}\dot{M}_{\varrho_i}^{\mathrm{T}}\dot{M}_{\varrho_i}\varepsilon_i) + \mathbb{E}(2\varepsilon_i^{\mathrm{T}}\dot{M}_{\varrho_i}^{\mathrm{T}}(\dot{M}_i + \varsigma M_i)\varepsilon_i) + \mathbb{E}(\varepsilon_i^{\mathrm{T}}(\dot{M}_i + \varsigma M_i)^{\mathrm{T}}(\dot{M}_i + \varsigma M_i)\varepsilon_i) = 0$, where $\mathbb{E}(2\varepsilon_i^{\mathrm{T}}\dot{M}_{\varrho_i}^{\mathrm{T}}(\dot{M}_i + \varsigma M_i)\varepsilon_i) = 2\mathbb{E}(\varepsilon_i^{\mathrm{T}})\mathbb{E}(\dot{M}_{\varrho_i}^{\mathrm{T}})\mathbb{E}((\dot{M}_i + \varsigma M_i)\varepsilon_i) = 0$. Under the stated nonsingularity condition, $\mathrm{Var}(\dot{M}_{\varrho_i})$ is represented by a positive-definite matrix U. Hence, $\mathbb{E}(\varepsilon_i^{\mathrm{T}}U\varepsilon_i) = 0$ holds at the steady state $(\tilde{\dot{M}}_i + \varsigma M_i)\varepsilon_i = \mathbf{0}$. Because U is positive definite, this equality implies $\varepsilon_i \to \mathbf{0}$. Consequently, $M_i\bar{\psi}_i$ can approximate p_i, and $\bar{J}_i(\theta_i, \bar{\psi}_i)$ can approximate $J_i(\theta_i, \psi_i)$. □

Remark 3. *The fuzzy gain controls the adaptation rate, whereas the composite position–velocity regressor determines which parameter directions are observable. The Lyapunov argument makes the observable residual nonincreasing. Exact parameter recovery also requires sufficient kinematic excitation. Without that excitation, the update can reduce the measured residual without identifying every parameter component.*

3.1. Algorithm Design

Each manipulator lacks the other manipulators' local kinematic information and cannot evaluate the aggregate action directly. We therefore introduce a local aggregate estimate and solve game (7) using the following neurodynamic system:

$$\begin{aligned}\ddot{\theta}_i =& -\dot{\theta}_i+\mathcal{P}_{\Omega_i}[\dot{\theta}_i-\nabla_{\dot{\theta}_i}\mathcal{J}_i(\dot{\theta}_i,u_i)+J_i^{\mathrm{T}}\mu_i+J_{\mathrm{v}i}^{\mathrm{T}}\mu_{\mathrm{v}i}\\ &+A_i^{\mathrm{T}}\nu_i-\eta_i^{\mathrm{T}}\lambda_i+\sum\nolimits_{j\in\mathbb{N}_i}(u_i-u_j)],\end{aligned} \tag{13a}$$

$$\dot{u}_i =-\sum\nolimits_{j\in\mathbb{N}_i}(\dot{\theta}_i-\dot{\theta}_j+u_i-u_j)+\ddot{\theta}_i, \tag{13b}$$

$$\dot{\lambda}_i =-\lambda_i+\mathcal{P}_{\mathbb{R}_{\geq 0}}(\lambda_i+\eta_i\dot{\theta}_i-h_i) \tag{13c}$$

$$\dot{\mu}_i =-\sum\nolimits_{j\in\mathbb{N}_i}(\mu_i-\mu_j)-J_i\dot{\theta}_i+q_i, \tag{13d}$$

$$\dot{\mu}_{\mathrm{v}i} =-\sum\nolimits_{j\in\mathbb{N}_i}(\mu_{\mathrm{v}i}-\mu_{\mathrm{v}j})-J_{\mathrm{v}i}\dot{\theta}_i+q_{\mathrm{v}i}, \tag{13e}$$

$$\dot{\nu}_i =-\sum\nolimits_{j\in\mathbb{N}_i}(\nu_i-\nu_j)-\sum\nolimits_{k=1}^{N}A_k\dot{\theta}_k+\sum\nolimits_{k=1}^{N}\dot{\mathscr{R}}_{\mathrm{d}k}, \tag{13f}$$

where $\mathcal{P}_{\Omega_i}[\cdot]$ is the projection operator, $\mathbb{N}_i$ is the neighbor set of manipulator i, and $u_i\in\mathbb{R}^a$ is its local aggregate estimate. The initial states satisfy $\dot{\theta}_{i0}\in\Omega_i$, $\lambda_{i0}\in\mathbb{R}_+$, $\mu_{i0}\in\mathbb{R}^3$, $\mu_{\mathrm{v}i0}\in\mathbb{R}^2$, $\nu_{i0}\in\mathbb{R}^4$, $u_{i0}=\dot{\theta}_{i0}$. Manipulator i evaluates the local gradient as $\nabla_{\dot{\theta}_i}\mathcal{J}_i(\dot{\theta}_i,u_i)=\nabla_{\dot{\theta}_i}\mathcal{J}_i(\dot{\theta}_i,x)\mid_{x=u_i}+\frac{1}{N}\nabla_x\mathcal{J}_i(\dot{\theta}_i,x)\mid_{x=u_i}$ using its estimate u_i.

Remark 4. *The decision variable is the joint velocity $\dot{\theta}_i$, whose dynamics are given by (13a). The projection $\mathcal{P}_{\Omega_i}$ enforces the local velocity bounds. The multipliers μ_i and $\mu_{\mathrm{v}i}$ impose the tracking and posture equalities through (13d) and (13e). Equation (13f) updates ν_i for the network-level self-adjustment equality, while (13c) updates λ_i for the obstacle-avoidance inequality. Because agents do not exchange private kinematic parameters, Equation (13b) supplies each player with the local aggregate estimate used in the GNE search.*

Let $\mathcal{F}_u(\dot{\theta})$ denote the extended pseudo-gradient $[\nabla_{\dot{\theta}_i}\mathcal{J}_i(\dot{\theta}_i,u_i)]_{i=1}^N$, and stack the variables as $\ddot{\theta}=[\ddot{\theta}_i]_{i=1}^N$, $u=[u_i]_{i=1}^N$, $\lambda=[\lambda_i]_{i=1}^N$, $\mu=[\mu_i]_{i=1}^N$, $\mu_{\mathrm{v}}=[\mu_{\mathrm{v}i}]_{i=1}^N$, $\nu=[\nu_i]_{i=1}^N$. Set $\Omega=\prod_i\Omega_i$, $\eta=\mathrm{diag}_i(\eta_i)$, and $(h,q,q_{\mathrm{v}})=([h_i],[q_i],[q_{\mathrm{v}i}])_{i=1}^N$. The repeated coupling block is $A=\mathbf{1}_N\otimes[A_1,\ldots,A_N]\in\mathbb{R}^{4N\times Na}$.

$$\begin{bmatrix}\ddot{\theta}\\ \dot{u}\\ \dot{\lambda}\\ \dot{\mu}\\ \dot{\mu}_{\mathrm{v}}\\ \dot{\nu}\end{bmatrix}=\begin{bmatrix}-\dot{\theta}+\mathcal{P}_{\Omega}[\dot{\theta}-\xi]\\ -(L\otimes I_a)(\dot{\theta}+u)+\ddot{\theta}\\ -\lambda+\mathcal{P}_{\mathbb{R}_+^N}[\lambda+\eta\dot{\theta}-h]\\ -(L\otimes I_3)\mu-J\dot{\theta}+q\\ -(L\otimes I_2)\mu_{\mathrm{v}}-J_{\mathrm{v}}\dot{\theta}+q_{\mathrm{v}}\\ -(L\otimes I_4)\nu-A\dot{\theta}+\dot{\mathscr{R}}_{\mathrm{d}}\end{bmatrix}, \tag{14}$$

where $\xi=\mathcal{F}_u(\dot{\theta})-J^{\mathrm{T}}\mu-J_{\mathrm{v}}^{\mathrm{T}}\mu_{\mathrm{v}}-A^{\mathrm{T}}\nu+\eta^{\mathrm{T}}\lambda-(L\otimes I_a)u$, $J=\mathrm{diag}[J_i]_{i=1}^N$, $J_{\mathrm{v}}=\mathrm{diag}[J_{\mathrm{v}i}]_{i=1}^N$, $\dot{\mathscr{R}}_{\mathrm{d}}=[\sum_{i=1}^N\dot{\mathscr{R}}_{\mathrm{d}i}^{\mathrm{T}},\ldots,\sum_{i=1}^N\dot{\mathscr{R}}_{\mathrm{d}i}^{\mathrm{T}}]^{\mathrm{T}}\in\mathbb{R}^{4N}$. Because $\bar{\mathcal{J}}_i$ is continuously differentiable and convex in $\dot{\theta}_i$, its gradient is locally Lipschitz under the stated regularity assumptions. Algorithm (14) couples each action update to a local estimate of the aggregate action.

3.2. Convergence Analysis

To analyze convergence of algorithm (13) to a GNE of game (7), define the Lyapunov function

$$V_y=-\langle\Gamma(y),\Lambda(y)\rangle-\frac{1}{2}\|\Lambda(y)\|^2+\frac{1}{2}\|y-y^*\|^2, \tag{15}$$

where $y=(\dot{\theta},u,\lambda,\mu,\mu_{\mathrm{v}},\nu)$ and $y^*=(\dot{\theta}^*,u^*,\lambda^*,\mu^*,\mu_{\mathrm{v}}^*,\nu^*)$. The mappings Λ and Γ are

$$\Lambda: y\mapsto(14), \tag{16}$$

$$\Gamma : y \mapsto \begin{bmatrix} \mathcal{F}_u(\dot{\theta}) - J^{\mathrm{T}}\mu - J_{\mathrm{v}}^{\mathrm{T}}\mu_{\mathrm{v}} - A^{\mathrm{T}}\nu + \eta^{\mathrm{T}}\lambda - (L \otimes I_a)u \\ (L \otimes I_a)\dot{\theta} + (L \otimes I_a)u - \ddot{\theta} \\ -\eta\dot{\theta} + h \\ (L \otimes I_3)\mu + J\dot{\theta} - q \\ (L \otimes I_2)\mu_{\mathrm{v}} + J_{\mathrm{v}}\dot{\theta} - q_{\mathrm{v}} \\ (L \otimes I_4)\nu + A\dot{\theta} - \dot{\mathscr{R}}_{\mathrm{d}} \end{bmatrix}, \tag{17}$$

and $\dot{\theta}^*$, u^*, λ^*, μ^*, μ_{v}^*, ν^* are taken from Lemma 1. Assume that the feasible set is nonempty, the communication graph is connected, and the extended pseudo-gradient is monotone on the feasible set.

Theorem 2. *For any admissible initial state y_0, the solution $(\dot{\theta}, u, \lambda, \mu, \mu_{\mathrm{v}}, \nu)$ of (14) converges to a GNE of game (7).*

Proof. Define $Y = \Omega \times \mathbb{R}^{aN} \times \mathbb{R}_+^N \times \mathbb{R}^{3N} \times \mathbb{R}^{2N} \times \mathbb{R}^{4N}$. The projection property gives $\langle y - \Gamma(y) - \mathcal{P}_Y[y - \Gamma(y)], y - \mathcal{P}_Y[y - \Gamma(y)]\rangle \le 0$. Using $\Lambda(y) = -y + \mathcal{P}_Y(y - \Gamma(y))$ gives $-\langle \Gamma(y), \Lambda(y)\rangle \ge \|\Lambda(y)\|^2$. Together with (15), this inequality yields $V_y \ge \frac{1}{2}\|\Lambda(y)\|^2 + \frac{1}{2}\|y - y^*\|^2 \ge 0$. Consequently, V_y is radially unbounded with respect to $\dot{\theta}$, u, λ, μ, μ_{v}, and ν. The projection property also gives $\langle y - \Gamma(y) - \mathcal{P}_Y[y - \Gamma(y)], y^* - \mathcal{P}_Y[y - \Gamma(y)]\rangle \le 0$. Substitute $\mathcal{P}_Y[y - \Gamma(y)] = y + \Lambda(y)$ into this inequality and expand the inner product to obtain

$$\langle \Gamma(y) + y - y^*, \Lambda(y)\rangle \le -\langle \Gamma(y), y - y^*\rangle - \|\Lambda(y)\|^2. \tag{18}$$

Differentiating V_y along the solution of (14) gives

$$\begin{aligned} \dot{V}_y =& \langle \Gamma(y) - \partial\Gamma(y)\Lambda(y) + \Lambda(y) + y - y^*, \dot{y}\rangle \\ =& \langle \Gamma(y) + y - y^*, \Lambda(y)\rangle - (\Lambda(y))^{\mathrm{T}}\partial\Gamma(y)\Lambda(y) + \|\Lambda(y)\|^2, \end{aligned} \tag{19}$$

which, together with (18), gives

$$\dot{V}_y \le -\langle \Gamma(y), y - y^*\rangle - (\Lambda(y))^{\mathrm{T}}\partial\Gamma(y)\Lambda(y), \tag{20}$$

where $\partial\Gamma(y) = \begin{bmatrix} f_{\dot{\theta}} & -(L \otimes I_a) & \eta^{\mathrm{T}} & -J^{\mathrm{T}} & -J_{\mathrm{v}}^{\mathrm{T}} & -A^{\mathrm{T}} \\ L \otimes I_a & L \otimes I_a & 0 & 0 & 0 & 0 \\ -\eta & 0 & 0 & 0 & 0 & 0 \\ J & 0 & 0 & L \otimes I_3 & 0 & 0 \\ J_{\mathrm{v}} & 0 & 0 & 0 & L \otimes I_2 & 0 \\ A & 0 & 0 & 0 & 0 & L \otimes I_4 \end{bmatrix}$, with $f_{\dot{\theta}} = \partial\mathcal{F}_u(\dot{\theta})/\partial\dot{\theta}$. The monotonicity of the extended pseudo-gradient makes the symmetric part of $f_{\dot{\theta}}$ positive semidefinite. The undirected communication graph also makes L symmetric positive semidefinite. Hence, $y^{\mathrm{T}}\partial\Gamma(y)y = \dot{\theta}^{\mathrm{T}} f_{\dot{\theta}}\dot{\theta} + u^{\mathrm{T}}(L \otimes I_a)u + \mu^{\mathrm{T}}(L \otimes I_3)\mu + \mu_{\mathrm{v}}^{\mathrm{T}}(L \otimes I_2)\mu_{\mathrm{v}} + \nu^{\mathrm{T}}(L \otimes I_4)\nu \ge 0$. Thus, the symmetric part of $\partial\Gamma(y)$ is positive semidefinite and $(\Lambda(y))^{\mathrm{T}}\partial\Gamma(y)\Lambda(y) \ge 0$. Equation (20) therefore gives $\dot{V}_y \le -\langle \Gamma(y), y - y^*\rangle$. Since y^* satisfies the VI equilibrium condition, $\langle \Gamma(y^*), y - y^*\rangle \ge 0$. Subtracting this term and using monotonicity gives

$$\begin{aligned} \dot{V}_y \le& -\langle \Gamma(y) - \Gamma(y^*), y - y^*\rangle \\ =& -\langle \mathcal{F}_u(\dot{\theta}) - \mathcal{F}_u(\dot{\theta}^*), \dot{\theta} - \dot{\theta}^*\rangle - (u - u^*)^{\mathrm{T}}(L \otimes I_a)(u - u^*) \\ & -(\mu - \mu^*)^{\mathrm{T}}(L \otimes I_3)(\mu - \mu^*) - (\mu_{\mathrm{v}} - \mu_{\mathrm{v}}^*)^{\mathrm{T}}(L \otimes I_2)(\mu_{\mathrm{v}} - \mu_{\mathrm{v}}^*) \\ & -(\nu - \nu^*)^{\mathrm{T}}(L \otimes I_4)(\nu - \nu^*) \le 0. \end{aligned} \tag{21}$$

The lower bound on V_y and the derivative estimate (21) establish Lyapunov stability of the neurodynamic aggregation algorithm. Define $\mathcal{W}_1 = \{(\dot{\theta}, u, \lambda, \mu, \mu_{\mathrm{v}}, \nu) : \dot{V}_y = 0\}$ and let $\mathcal{W}_2$ be its largest weakly invariant subset. LaSalle's invariance principle implies that $(\dot{\theta}, u, \lambda, \mu, \mu_{\mathrm{v}}, \nu)$ converges to $\mathcal{W}_2$. On this invariant set, $\Lambda(y) = 0$; the consensus residuals vanish and the projected equilibrium conditions reduce to those in Lemma 1. Hence, $\dot{\theta}$ converges to a GNE of game (7), which yields the distributed equilibrium control of the multi-manipulator system. □

Remark 5. *The projection keeps the joint velocities feasible, the consensus dynamics align the local aggregate estimates, and the multipliers impose the physical constraints. Under monotonicity, these components make the Lyapunov function nonincreasing. LaSalle's invariance principle then confines the limiting motion to the feasible stationary points characterized by the GNE conditions.*

4. Simulation Experiments

We evaluated the controller under simultaneous trajectory-tracking, posture-preservation, obstacle-avoidance, and self-adjustment tasks. The system comprised $N = 4$ seven-degree-of-freedom manipulators with bases at $(0, 0, 0)$, $(1.5, 0, 0)$, $(0, 1.5, 0)$, and $(1.5, 1.5, 0)$. The communication graph was undirected and complete. We placed three static obstacles in the workspace and set the outer and hard distance thresholds to $\bar{d} = 0.5$ m and $\underline{d} = 0.15$ m. We bounded each joint velocity within $[-2.5, 2.5]$ rad/s. Joints $[1, 3, 5, 7]$ participated in network-level self-adjustment, with a desired summed velocity of zero.

Each simulation lasted 8 s with an integration step of 0.005 s. We set the feedback gains to $\rho = 5$ and $\epsilon = 0.15$. The controller Jacobian used the current estimates of seven kinematic parameters. We did not use exponential moving averages, home-posture regularization, velocity damping, or display-only obstacle slowdown. We varied the initial joint states and parameter estimates across five paired random seeds $\{11, 23, 37, 53, 71\}$.

We prescribed periodic three-dimensional end-effector motion with manipulator-dependent phase shifts and a time-varying two-dimensional orientation reference:

$$p_{\mathrm{d}i}(t) = b_i + [0.25, 0.25, 0.85]^{\mathrm{T}} + [0.08\cos\xi_i,\ 0.08\sin\xi_i,\ 0.03\sin(2\xi_i)]^{\mathrm{T}},$$
$$v_i^{\mathrm{ref}}(t) = [0.1\sin(0.5t + \phi_i),\ 0.1\cos(0.5t + \phi_i)]^{\mathrm{T}},$$

where b_i is the base position, $\xi_i = 0.8\pi t + \phi_i$, and $\phi_i = (i - 1)\pi/2$.

The convex aggregative stage cost was

$$\mathcal{J}_i(\dot{\theta}_i, \sigma(\dot{\theta})) = \frac{1}{2}\dot{\theta}_i^{\mathrm{T}} P_i \dot{\theta}_i + \frac{\beta}{2}\|\dot{\theta}_i - \sigma(\dot{\theta})\|^2 + \omega_i^{\mathrm{T}}\sigma(\dot{\theta}), \tag{22}$$

where $\beta = 1$ and $P_i = (3 + 0.5i)I$ for $i = 1, \ldots, 4$. We sampled each component of the fixed nonnegative vector ω_i uniformly from $[0.01, 0.08]$ using the paired seed. The coupling term introduces the local aggregate estimate into the control decision and penalizes deviation from the collective action.

We compared five matched conditions. The proposed method used distributed aggregate estimation and fuzzy kinematic learning. For the baseline, we adapted the off-diagonal partial-decision profile-consensus update from Equation (8b) of Huo et al. [24]. We used each player's current local decision as its diagonal block and kept the reported gains $\gamma = 0.9$ and $\beta = 3$. Each player stored an estimate of the complete action profile. To isolate the effect of this change to the aggregate-information layer, both methods used the same kinematic learner, task projections, constraints, game correction, and plant. The full-information reference replaced the local estimate with the true aggregate. The fixed-gain control retained distributed estimation and set $\Delta\gamma_i = 0.30$. The local-only control substituted each manipulator's action for the aggregate estimate.

Steady tracking RMSE is evaluated over the final 20% of each run and then averaged over manipulators. The aggregate-estimation RMSE averages the norm of the local-estimate error over time and agents. The projected-gradient diagnostic is the RMSE of the game gradient after projection onto the tracking/orientation null spaces and the tangent space of the network self-adjustment equality. It quantifies stationarity within these projected subspaces; active velocity-bound and obstacle-complementarity residuals are outside this diagnostic. Runtime is the observed wall-clock time per integration step. Table 2 reports the mean and standard deviation across the five paired seeds.

Table 2. Comparison over five paired random seeds. Values are mean ± standard deviation. Lower values are better.

Method	Track RMSE (10^{-3} m)	Proj. Gradient	Aggregate RMSE	Cost	Runtime (ms/Step)
Proposed	3.789 ± 0.788	1.816 ± 0.429	0.731 ± 0.041	9.022 ± 1.342	4.974 ± 0.026
Huo et al. [24]	3.787 ± 0.790	1.801 ± 0.399	0.632 ± 0.033	8.912 ± 1.178	5.034 ± 0.012
Full information	3.822 ± 0.788	1.824 ± 0.452	0	8.942 ± 1.204	4.942 ± 0.019
Fixed gain	3.841 ± 0.872	1.844 ± 0.431	0.737 ± 0.047	9.109 ± 1.362	4.880 ± 0.022
Local only	3.798 ± 0.740	1.860 ± 0.452	1.600 ± 0.143	8.916 ± 1.142	4.923 ± 0.020

Figure 3 and Table 2 summarize the matched comparison. All conditions attained mean steady-state tracking root-mean-square errors (RMSEs) below 4×10^{-3} m. Relative to the Huo et al. baseline, the proposed condition retained effectively identical tracking accuracy (3.789 ± 0.788 versus 3.787 ± 0.790 in units of 10^{-3} m) and reduced mean runtime per step by 1.20%. The proposed condition's projected-gradient RMSE, aggregate-estimation RMSE, and cost were 0.80%, 15.75%, and 1.225% higher than the Huo et al. baseline values, respectively, defining a

resource–accuracy trade-off rather than a single-metric comparison. Relative to the internal controls, the proposed condition yielded a projected-gradient RMSE of 1.816, 1.51% below fixed gain and 2.36% below local only.

The proposed method achieved similar tracking accuracy with one-quarter of the local information dimension and one-quarter of the exchanged scalars of the Huo et al. baseline.

Under the complete graph and one-vector-per-directed-edge accounting, the proposed condition transmitted 84 scalars per step, compared with 336 for the profile-consensus baseline. Its local information dimension was likewise 7 rather than 28. Both reductions arise because each agent exchanges and stores one aggregate estimate instead of a complete four-manipulator action profile. These counts quantify the implemented direct-exchange protocols; alternative protocols may yield different exchange patterns.

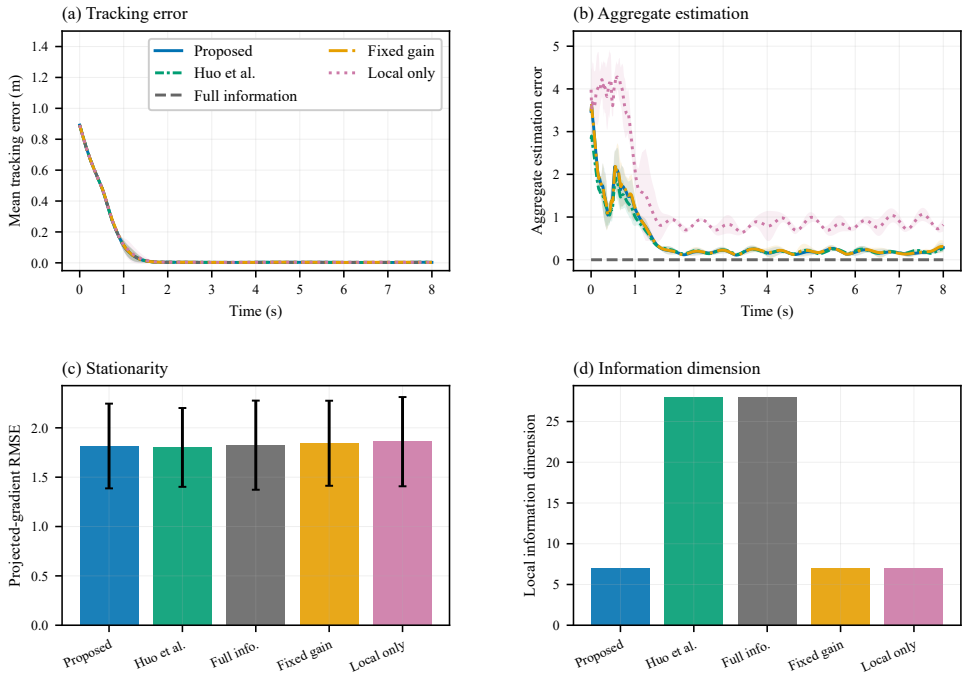


Figure 3. Matched comparison over five paired seeds. (**a**) Mean end-effector tracking error. (**b**) Aggregate-estimation error. (**c**) Projected-gradient RMSE. (**d**) Local information dimension.

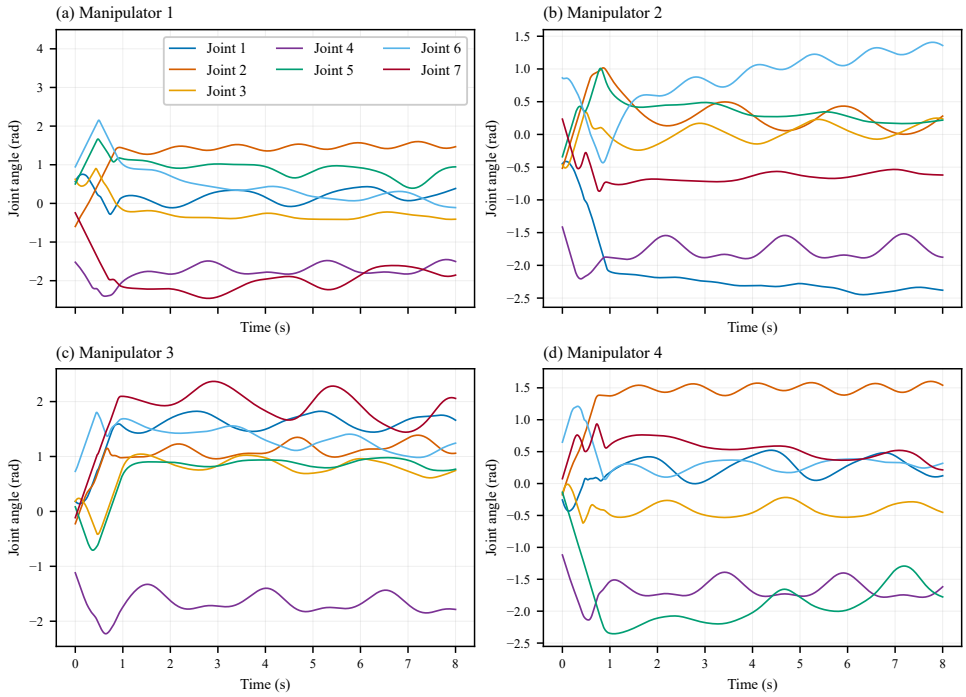


Figure 4. Joint-angle trajectories of the proposed method for Manipulators 1–4 in (**a**–**d**), respectively.

Figure 4 shows joint trajectories from the proposed condition with seed 11. Panels a–d correspond to Manipulators 1–4, and the colours identify individual joints. All trajectories remained bounded over the simulated horizon. Across the five proposed-condition runs, none of the 112,140 recorded joint-velocity components exceeded the prescribed interval; the largest absolute sampled velocity was 2.5 rad/s.

Figure 5 reports the kinematic-learning results. After the transient, the adaptive gains decreased and remained below the fixed gain for most of the steady regime. Final parameter RMSE was 0.0683 ± 0.0079 for proposed and 0.0671 ± 0.0067 for fixed gain. The fuzzy rule base therefore achieved comparable parameter estimation while operating with a lower late-stage gain. Figure 6 reports the self-adjustment residual, obstacle distance, and aggregative cost. The self-adjustment residual decayed to a small neighborhood after the initial transient. Every sampled trajectory remained above the hard threshold $\underline{d} = 0.15$ m. The mean minimum distance was 0.398 ± 0.010 m, and no sampled safety violations occurred. Entry into the outer threshold during the transient activated the obstacle correction without an additional display-only slowdown.

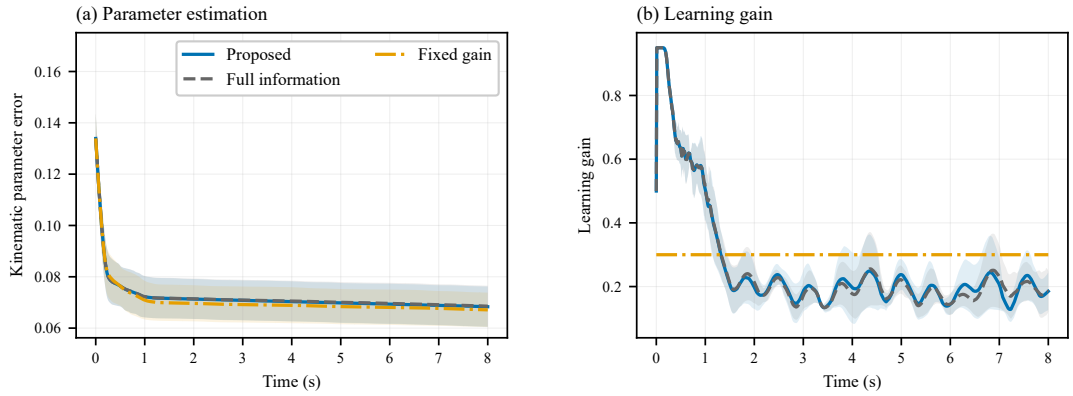


Figure 5. Kinematic learning over five paired seeds. (**a**) Parameter-estimation error. (**b**) Learning gain with the fixed-gain reference. Curves with shading: mean ± one standard deviation.

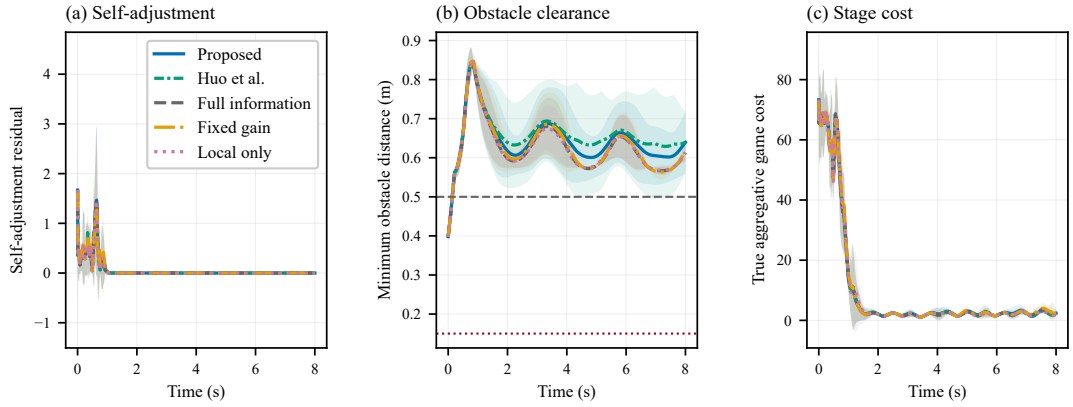


Figure 6. Constraints and cost over five paired seeds. (**a**) Self-adjustment residual. (**b**) Minimum obstacle distance with hard (dotted) and outer (dashed) thresholds. (**c**) True aggregative stage cost. Curves with shading: mean ± one standard deviation.

5. Conclusions

Considering the complex and comprehensive task constraints and the potential mutual exclusivity among multiple manipulators, this paper has modeled a class of multiple manipulators cooperative control problems with integrated tasks as an aggregative game. Each manipulator is regarded as a participant in the aggregative game, where the aggregated information is not known to the participants. The corresponding tasks include trajectory tracking, obstacle avoidance, posture preservation, and rarely involved self-adjustment. We have assigned a variable to each participant to estimate the aggregated information of kinematics, thereby guiding the generation of the neurodynamic algorithm. During the execution of the algorithm, each manipulator only transmits the aggregated kinematic information, thus reducing the communication cost of the multiple manipulators network. The proposed neurodynamic algorithm can track the motion trajectories of the manipulators, safely avoid obstacles, and maintain

the posture, while optimizing self-adjustment. We have considered using neural operators with fuzzy association rules to address the issue of the manipulators' kinematics that is not well-known. We have also proved the convergence of the neurodynamic algorithm to the GNE in the aggregative game and the effectiveness of the distributed optimal control of multiple manipulators. Future work will focus on modeling uncertain aggregative games [32] to explore multiple manipulators control or predictive control [33].

Author Contributions

Y.C.: conceptualization, methodology, software; J.L.: data curation, writing—original draft preparation; L.W.: visualization, investigation; Y.L.: supervision; J.T.: software, validation. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported in part by the National Natural Science Foundation of China (No. 62606058); in part by the Natural Science Foundation of Chongqing (Nos. CSTB2024NSCQ-MSX1087, CSTB2024NSCQMSX0077); in part by the Chongqing Social Science Planning Doctoral Project (No. 2022BS064); in part by the Science and Technology Research Program of Chongqing Municipal Education Commission (Nos. KJQN202301541, KJQN202501553).

Data Availability Statement

Not applicable.

Conflicts of Interest

The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies

No AI tools were utilized for this paper.

References

1. Bai, Q.; Li, P.; Tian, W.; et al. Coordinated Motion Planning of the Mobile Redundant Manipulator for Processing Large Complex Components. *Int. J. Adv. Manuf. Technol.* **2022**, *121*, 6703–6721. https://doi.org/10.1007/s00170-022-09785-x.
2. Ma, Y.; Zhu, X.; Fei, H.; et al. Teleoperated Cyber-Physical System for Cable-Driven Hyper-Redundant Flexible Manipulation. *IEEE Trans. Ind. Cyber-Phys. Syst.* **2025**, *3*, 588–596. https://doi.org/10.1109/ticps.2025.3614603.
3. Yao, K.; Billard, A. Exploiting Kinematic Redundancy for Robotic Grasping of Multiple Objects. *IEEE Trans. Robot.* **2023**, *39*, 1982–2002. https://doi.org/10.1109/tro.2023.3253249.
4. Xie, Z.; Jin, L. A Fuzzy Neural Controller for Model-Free Control of Redundant Manipulators with Unknown Kinematic Parameters. *IEEE Trans. Fuzzy Syst.* **2024**, *32*, 1589–1601. https://doi.org/10.1109/tfuzz.2023.3328545.
5. Zhang, M.; Guo, J.; Zhang, Z. A Distributed Slack Barrier Recurrent Neural Network for Multiple Redundant Manipulators Collaborative System in Obstacles Environment. *IEEE Trans. Syst. Man Cybern. Syst.* **2025**, *55*, 6299–6311. https://doi.org/10.1109/tsmc.2025.3578566.
6. Huang, T.; Pan, H.; Sun, W.; et al. Sine Resistance Network-Based Motion Planning Approach for Autonomous Electric Vehicles in Dynamic Environments. *IEEE Trans. Transp. Electrif.* **2022**, *8*, 2862–2873. https://doi.org/10.1109/tte.2022.3151852.
7. Jin, L.; Zheng, X.; Luo, X. Neural Dynamics for Distributed Collaborative Control of Manipulators with Time Delays. *IEEE/CAA J. Autom. Sin.* **2022**, *9*, 854–863. https://doi.org/10.1109/jas.2022.105446.
8. Li, H.; Lu, Q.; Huang, T. Distributed Projection Subgradient Algorithm over Time-Varying General Unbalanced Directed Graphs. *IEEE Trans. Autom. Control.* **2019**, *64*, 1309–1316. https://doi.org/10.1109/tac.2018.2849616.
9. Li, H.; Wu, X.; Wang, Z.; et al. Distributed Primal–Dual Splitting Algorithm for Multiblock Separable Optimization Problems. *IEEE Trans. Autom. Control.* **2022**, *67*, 4264–4271. https://doi.org/10.1109/tac.2021.3116116.
10. Liu, Y.; Yao, X.; Zhao, W. Distributed Neural-Based Fault-Tolerant Control of Multiple Flexible Manipulators with Input Saturations. *Automatica* **2023**, *156*, 111202. https://doi.org/10.1016/j.automatica.2023.111202.
11. Zhang, J.; Jin, L.; Yang, C. Distributed Cooperative Kinematic Control of Multiple Robotic Manipulators with an Improved Communication Efficiency. *IEEE ASME Trans.Mechatron.* **2022**, *27*, 149–158. https://doi.org/10.1109/tmech.2021.3059441.
12. Xie, Z.; Jin, L.; Luo, X. Kinematics-Based Motion-Force Control for Redundant Manipulators with Quaternion Control. *IEEE Trans. Autom. Sci. Eng.* **2023**, *20*, 1815–1828. https://doi.org/10.1109/tase.2022.3186668.
13. Ginnante, A.; Simetti, E.; Caro, S.; et al. Task Priority Based Design Optimization of a Kinematic Redundant Robot. *Mech. Mach. Theory* **2023**, *187*, 105374. https://doi.org/10.1016/j.mechmachtheory.2023.105374.

14. Sun, W.; Yuan, Y. Passivity Based Hierarchical Multi-Task Tracking Control for Redundant Manipulators with Uncertainties. *Automatica* **2023**, *155*, 111159. https://doi.org/10.1016/j.automatica.2023.111159.
15. Xie, Z.; Jin, L.; Lv, X. A Hierarchical Control and Learning Network for Redundant Manipulators with Unknown Physical Parameters. *IEEE Trans. Ind. Electron.* **2025**, *72*, 7106–7116. https://doi.org/10.1109/tie.2024.3497304.
16. Karami, A.; Sadeghian, H.; Keshmiri, M.; et al. Hierarchical Tracking Task Control in Redundant Manipulators with Compliance Control in the Null-Space. *Mechatronics* **2018**, *55*, 171–179. https://doi.org/10.1016/j.mechatronics.2018.09.005.
17. Zhou, H.; Lin, X. Intelligent Redundant Manipulation for Long-Horizon Operations with Multiple Goal-Conditioned Hierarchical Learning. *Adv. Robot.* **2025**, *39*, 291–304. https://doi.org/10.1080/01691864.2025.2475295.
18. Sewlia, M.; Verginis, C.K.; Dimarogonas, D.V. Trajectory Tracking for Multi-Manipulator Systems in Constrained Environments. *arXiv* **2025**, arXiv:2512.14206.
19. Yang, X.; Zhao, Z.; Ma, B.; et al. Kinematic and Dynamic Manipulability Optimizations of Redundant Manipulators Based on RNN Model. *IEEE Trans. Ind. Inform.* **2024**, *20*, 5763–5773. https://doi.org/10.1109/TII.2023.3334305.
20. Tan, J.; Shang, M.; Jin, L. Metaheuristic-Based RNN for Manipulability Optimization of Redundant Manipulators. *IEEE Trans. Ind. Informat.* **2024**, *20*, 6489–6498. https://doi.org/10.1109/tii.2023.3348830.
21. Zhang, J.; Jin, L.; Cheng, L. RNN for Perturbed Manipulability Optimization of Manipulators Based on a Distributed Scheme: A Game-Theoretic Perspective. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 5116–5126. https://doi.org/10.1109/tnnls.2020.2963998.
22. Kong, Y.; Chen, S.; Jiang, Y.; et al. Zeroing Neural Network with Fuzzy Parameter for Cooperative Manner of Multiple Redundant Manipulators. *Expert Syst. Appl.* **2023**, *212*, 118735. https://doi.org/10.1016/j.eswa.2022.118735.
23. Izadbakhsh, A.; Nikdel, N.; Deylami, A. Cooperative and Robust Object Handling by Multiple Manipulators Based on the Differential Equation Approximator. *ISA Trans.* **2022**, *128*, 68–80. https://doi.org/10.1016/j.isatra.2021.11.010.
24. Huo, W.; Huang, L.; Dey, S.; et al. Neural-Network-Based Distributed Generalized Nash Equilibrium Seeking for Uncertain Nonlinear Multiagent Systems. *IEEE Trans. Control. Netw. Syst.* **2024**, *11*, 1323–1334. https://doi.org/10.1109/tcns.2023.3337671.
25. Liu, M.; Mu, W.; Lv, X.; et al. Neural Network for Distributed Collaboration of Multiple Manipulators with Switching Topologies: A Game-Theoretic Perspective. *IEEE Trans. Syst. Man Cybern. Syst.* **2025**, *55*, 3213–3221. https://doi.org/10.1109/tsmc.2025.3540403.
26. Zhang, J.; Jin, L.; Wang, Y.; et al. A Collaboration Scheme for Controlling Multimanipulator System: A Game-Theoretic Perspective. *IEEE ASME Trans. Mechatron.* **2023**, *28*, 128–139. https://doi.org/10.1109/tmech.2022.3193136.
27. Tong, Y.; Liu, H.; Yang, T.; et al. A Game Theory Perspective for Multimanipulators with Unknown Kinematics. *IEEE Trans. Cogn. Dev. Syst.* **2026**, *18*, 167–185. https://doi.org/10.1109/tcds.2025.3575530.
28. Li, H.; Li, J.; Ran, L.; et al. A Survey of Distributed Algorithms for Aggregative Games. *IEEE/CAA J. Autom. Sin.* **2025**, *12*, 859–871. https://doi.org/10.1109/jas.2024.124998.
29. Li, H.; Ran, L.; Zheng, L.; et al. Convergence Analysis of Distributed Generalized Nash Equilibria Seeking Algorithm with Asynchrony and Delays. *IEEE Trans. Autom. Control.* **2025**, *70*, 642–648. https://doi.org/10.1109/tac.2024.3439652.
30. Facchinei, F.; Pang, J.-S. *Finite-Dimensional Variational Inequalities and Complementarity Problems*; Springer New York: New York, NY, USA, 2003. https://doi.org/10.1007/b97544.
31. Wu, C.H.; Ouyang, C.S.; Chen, L.W.; et al. A New Fuzzy Clustering Validity Index with a Median Factor for Centroid-Based Clustering. *IEEE Trans. Fuzzy Syst.* **2015**, *23*, 701–718. https://doi.org/10.1109/tfuzz.2014.2322495.
32. Liu, J.; Liao, X.; Dong, J.S.; et al. Continuous-Time Distributed Generalized Nash Equilibrium Seeking in Nonsmooth Fuzzy Aggregative Games. *IEEE Trans. Control. Netw. Syst.* **2024**, *11*, 1262–1274. https://doi.org/10.1109/tcns.2023.3336829.
33. Yan, J.; Jin, L.; Hu, B. Data-Driven Model Predictive Control for Redundant Manipulators with Unknown Model. *IEEE Trans. Cybern.* **2024**, *54*, 5901–5911. https://doi.org/10.1109/tcyb.2024.3408254.