Neural-Network-Based Distributed Generalized Nash Equilibrium Seeking for Uncertain Nonlinear Multiagent Systems

Wei Huo, Lingying Huang, Subhrakanti Dey, and Ling Shi, *Fellow, IEEE*

***Abstract*—This article investigates distributed variational generalized Nash equilibrium (v-GNE)-seeking problems in heterogeneous high-order nonlinear multiagent systems with partially unknown dynamics. To overcome the difficulties brought by high-order uncertain physical dynamics, we introduce a virtual decision with a primal–dual update for each player and design reference-tracking schemes to guide the players' outputs toward the v-GNE. Besides, we incorporate a shallow neural-network-based dynamic estimator to handle the unknown nonlinear dynamics. Through the Lyapunov analysis, we demonstrate that the players' behaviors converge to the v-GNE with an arbitrarily small error. Numerical simulations of connectivity control games and energy consumption games illustrate the effectiveness of our algorithm.**

***Index Terms*—Adaptive control, generalized Nash equilibrium (GNE), multiagent systems, neural network (NN).**

I. Introduction

NONCOOPERATIVE games have tremendous applications, including spectrum access in cognitive radio networks [1], energy consumption coordination in power grids [2], and path planning in unmanned vehicle systems [3]. A Nash equilibrium (NE) characterizes desirable and stable solutions where no player intends to unilaterally change its action. Generalized NEs (GNEs) additionally consider coupling constraints among players, such as limited shared resources [4] and safety distances [5]. Thus, GNE-seeking problems have become a focal point of research in game theory.

Manuscript received 7 June 2023; revised 1 August 2023; accepted 13 August 2023. Date of publication 29 November 2023; date of current version 19 September 2024. The work of Wei Huo and Lingying Shi was supported by the Hong Kong RGC General Research Fund under Grant 16211622. Recommended by Associate Editor Naira Hovakimyan. *(Corresponding author: Lingying Huang.)*

Wei Huo and Ling Shi are with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Kowloon, Hong Kong (e-mail: whuoaa@connect.ust.hk; eesling@ust.hk).

Lingying Huang is with the School of Electrical and Electronic Engineering, Nanyang Technology University, Singapore 639798 (e-mail: lingying.huang@ntu.edu.sg).

Subhrakanti Dey is with the Hamilton Institute, NUI Maynooth, Maynooth, Company, W23 F2H6 Maynooth, Ireland, and also with Uppsala University, 75105 Uppsala, Sweden (e-mail: subhra.dey@signal.uu.se).

Digital Object Identifier 10.1109/TCNS.2023.3337671

The rapid development of multiagent networks has increased interest in the distributed computation of GNEs. Unlike classical methods that rely on a central node to provide players with global information about all opponents' actions, participating players only access the decisions of their neighbors in distributed networks, i.e., a partial-decision setting. To search the GNE with coupled linear equation constraints in such settings, the authors in [6] and [7] proposed distributed algorithms. However, the convergence rate is slowed down by the diminishing step size. Later, the authors in [8] and [9] developed fully distributed variational GNE (v-GNE)-seeking algorithms with fixed step sizes in discontinuous-time and continuous-time systems, respectively.

The distributed GNE-seeking algorithms mentioned earlier are designed for static multiagent systems without inherent players' dynamics, which can be regarded as pure information-layer problems. However, in cyber-physical systems (CPSs), players typically have physical dynamics such as bicycle dynamics and Euler–Lagrange dynamics for mobile robot manipulators [10], [11]. Integrating physical dynamics into distributed GNE seeking significantly increases the difficulty since players' actions cannot be controlled freely. Ibrahim and Hayakawa [12] proposed a control law for linear time-invariant dynamic players by using passivation to asymptotically stabilize the NE of the game. In quadratic games, Guo and De Persis [13] allowed the players' dynamics to be general linear systems subjected to uncertainties and external disturbances by designing an internal model. However, these works assumed that the players have full knowledge of others' actions, thus becoming unsuitable for partial-information scenarios. Cai et al. [14] proposed distributed strategy-updating rules for aggregative games in multiagent systems with given general linear dynamics. Considering the coupling affine constraints, Deng et al. [15] developed a distributed algorithm for linear second-order players such that their outputs converge to the v-GNE. For more realistic nonlinear multiagent systems, Deng and Liang [16] designed a distributed NE-seeking algorithm for aggregative games of multiple Euler–Lagrange systems using feedback linearization and tracking control. Deng [17] further presented distributed NE-seeking controllers for aggregative games in which players have second-order nonlinear dynamics with uncertain parameters.

Despite the fruitful outcomes of distributed NE-seeking algorithms for players with dynamics, most literature assumes that the players have explicit knowledge of their dynamic functions.

However, the nonlinear dynamics in real-world complex engineering systems are often difficult to model explicitly, making it even harder for players to know them. Ye et al. [18] developed an extended state observer for first-order players but required the unknown dynamics to be sufficiently smooth with bounded first and second derivatives. Tang and Yi [19] only considered the cases where unknown dynamics could be perfectly linearly parameterized. With the rise of artificial intelligence, neural networks (NNs) are demonstrating their strength in system identification and control [20], [21]. Ye et al. [22] developed a robust NE-seeking strategy for mixed-order nonlinear multiagent systems using radial basis function (RBF) NNs. However, they assumed that each player knows the Lipschitz constant of nonlinear dynamics and updated NN weights using the projection algorithm with a positive constant to bound the NN weight matrix. In addition, most existing works for dynamic players [12], [13], [14], [16], [17], [19], [22], [23] have not considered coupling constraints, which are crucial for resource-limited and safety-aware applications. Thus, the main challenges in designing and analyzing distributed v-GNE-seeking algorithms for uncertain nonlinear multiagent systems can be summarized as follows.

1) *Convergence in CPSs*: In reality, players often have inherent physical-layer dynamics that significantly influence their cyber-layer outputs, making it difficult to ensure that outputs converge to the v-GNE. Existing controllers for distributed GNE seeking [6], [7], [8], [9] fail to guarantee convergence in CPSs since they do not account for players' physical dynamics.
2) *Requirements on unknown dynamics*: Most existing works on noncooperative multiagent systems with dynamics assume that players know the exact dynamics model [12], [14], [15], [16], [17]. Some literature on handling uncertainties has specific requirements for nonlinear models [18], [19] while others require players to have partial information about the uncertainties [22]. However, since this information is usually limited, there is a need to design a novel adaptive controller that requires the least about the dynamics and players' knowledge.
3) *Coupling constraints*: Due to inherent uncertainties, the coupling constraints are difficult to satisfy since players' output decisions cannot be directly determined by their control input. Existing algorithms for distributed games with unknown dynamics ignore the constraints [18], [19], [22], and applying them directly would result in constraint violation at equilibria.

Motivated by the previous observations, this article aims to study the problem of seeking the v-GNE in noncooperative multiagent systems with high-order uncertain dynamics and a nonlinear shared constraint. We propose a novel adaptive controller with reference tracking and NN-based nonlinearity compensation for players. Based on the designed loss function for NNs, the NN weights are updated through online gradient descent and error backpropagation. The main contributions of this work are summarized as follows.

1) *v-GNE seeking for uncertain nonlinear systems*: We propose a distributed algorithm to solve the v-GNE-seeking problem in high-order multiagent systems that have a coupling constraint and uncertain nonlinear dynamics. Notably, we do not have any requirement on the continuous nonlinear dynamics, and the proposed approach does not rely on any knowledge of the nonlinear model. To the best of our knowledge, we are the first to address this problem.
2) *Adaptive NN*: We propose a novel loss function for the RBF NN to learn the unknown dynamics and derive a closed-form weight update rule with gradient descent and static error backpropagation. In contrast to the existing literature that achieves the adaptive weight update rule based on Lyapunov analysis [24], [25], [26], we provide a more intuitive perspective for designing an adaptation law.
3) *Exponential convergence with arbitrary accuracy*: Our proof shows that the proposed controller can drive players' outputs with an exponential convergence rate to an arbitrarily small neighborhood of the v-GNE, which satisfies the shared constraint (see Theorem 1).

The rest of this article is organized as follows. The problem of distributed v-GNE seeking in high-order nonlinear multiagent systems is formulated in Section II. In Section III, a distributed controller with an NN-based dynamics estimator is proposed. Then, details on the convergence analysis are provided in Section IV, followed by numerical simulations in Section V. Finally, Section VI concludes this article.

Notations: $\mathbb{R}^p$ and $\mathbb{R}^p_+$ represent the sets of p-dimensional real numbers and nonnegative real numbers, respectively. For $\mathcal{N} = \{1, 2, \ldots, N\}$, $\text{col}(x_i)_{i\in\mathcal{N}}$ or $\text{col}(x_i)_{i=1}^N$ refers to the stacked vector obtained from vectors x_i, and $\text{diag}(A_i)_{i\in\mathcal{N}}$ denotes the block diagonal matrix with $A_1, A_2, \ldots, A_N$ the main diagonal. For a positive semidefinite matrix A, $s_m(A)$ and $s_M(A)$ denote its smallest nonzero and largest eigenvalues, respectively. $I_q \in \mathbb{R}^{q\times q}$ represents an identity matrix, $\mathbf{0}_{p\times q} \in \mathbb{R}^{p\times q}$ is a zero matrix, and $\mathbf{1}_q \in \mathbb{R}^q$ denotes a vector with all elements equal to one. The operator $\|\cdot\|$ is the induced-2 norm for matrices and the Euclidean norm for vectors while $\|\cdot\|_F$ means the Frobenius norm of a matrix. For any two matrices, $A \in \mathbb{R}^{n\times m}$, $B \in \mathbb{R}^{p\times q}$, $A \otimes B \in \mathbb{R}^{np\times mq}$ is the Kronecker product of A by B. For a matrix A and any integer $q > 0$, we denote $A_{(q)} = A \otimes I_q$. $[\cdot]^+$ means $\max\{\cdot, 0\}$.

Let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ describe the information exchange among a set of players indexed by $\mathcal{N} = \{1, 2, \ldots, N\}$, where $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ denotes the communication links. Player j can receive messages from player i if and only if $(i, j) \in \mathcal{E}$. The adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{N\times N}$ describes the structure of $\mathcal{G}$, where $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$, and $a_{ij} = 0$ otherwise. G is undirected if $a_{ij} = a_{ji}$, $\forall i, j \in \mathcal{N}$, and connected if there is a path between any pair of players. Let $D = \text{diag}(d_i)_{i\in\mathcal{N}}$, where $d_i = \sum_{j=1}^N a_{ij}$. Then, we can write the Laplacian matrix as $L = D - A$. If $\mathcal{G}$ is undirected and connected, L is a symmetric matrix with N real eigenvalues in ascending order, given by $0 < s_2(L) \le \cdots \le s_N(L) \le 2\max_i\{d_i\}$.

II. Problem Formulation

Consider a group of players $\mathcal{N} = \{1, 2, \ldots, N\}$ with the same order of physical dynamics. The output of player $i \in \mathcal{N}$ is

determined by the following dynamics:

$$\begin{aligned}\dot{x}_{i,j} &= x_{i,j+1}, \quad j = 1, \ldots, n-1\\ \dot{x}_{i,n} &= h_i(x_i) + u_i\\ y_i &= x_{i,1}\end{aligned} \tag{1}$$

where $n \geq 1$ represents the order of the player's physical system, $x_{i,j} \in \mathbb{R}^m$ and $x_i = \text{col}(x_{i,j})_{j=1}^n$ are states of player i, $u_i \in \mathbb{R}^m$ and $y_i \in \mathbb{R}^m$ are its control input and output, respectively, and $h_i(x_i) : \mathbb{R}^{mn} \to \mathbb{R}^m$ is a continuous function representing an unknown nonlinear dynamics.

Remark 1: The dynamic model in (1) is widely applicable to practical systems, including first-order systems ($n = 1$), such as velocity-actuated vehicles [27], and second-order systems ($n = 2$), such as Euler–Lagrange systems in mechanical engineering [10], [11], [16] and Van der Pol oscillators in power grids [28]. The unknown nonlinear dynamics make v-GNE seeking considerably more challenging in comparison to previous works that deal with either single or multiple integrators [9], [29].

Remark 2: In the literature on noncooperative multiagent systems with uncertain nonlinear dynamics, Ye [18] required the second derivative of the nonlinear dynamics to be bounded. However, we only assume $h_i(x_i)$ to be continuous. On the other hand, Tang and Yi [19] assumed that the unknown dynamics can be perfectly linearly parameterized by a known continuous and bounded function vector. Nonetheless, we do not need such a restriction. Furthermore, players in [22] were required to know the Lipschitz constant of the unmodeled dynamics. However, $h_i(x_i)$ in (1) can be non-Lipschitz, and players have no knowledge of it. Thus, our assumption regarding the nonlinear dynamics is weaker and more practical.

The output decision profile of all players is denoted as $\mathbf{y} = \text{col}(y_i)_{i\in\mathcal{N}} = (y_i, \mathbf{y}_{-i})$, where $\mathbf{y}_{-i} = \text{col}(y_j)_{j\in\mathcal{N}\setminus\{i\}}$ represents the output profile of all player i's opponents. Selfish player i aims to minimize its cost function, $f_i(y_i, \mathbf{y}_{-i}) : \mathbb{R}^{mN} \to \mathbb{R}$, over a feasible action set. The feasible decision set of all players is defined by a nonlinear coupling constraint as follows:

$$\Omega = \left\{ \mathbf{y} \in \mathbb{R}^{mN} \mid g(\mathbf{y}) = \sum_{i=1}^{N} g_i(y_i) \leq 0 \right\} \tag{2}$$

where the function $g_i(y_i) : \mathbb{R}^m \to \mathbb{R}$ is nonlinear for any $i \in \mathcal{N}$. For instance, in a power grid, each power generation unit incurs an operating cost. While units want to minimize their cost selfishly, they need to consider global constraints such as power balance and emission constraints, i.e., the total generated power should satisfy the total demand in the grid, and the total carbon emission cannot exceed the stipulated limit [30], [31]. These global constraints can be modeled as (2).

The game among N players is formulated as a coupled optimization problem

$$\begin{aligned}\min_{y_i\in\mathbb{R}^m} \quad & f_i(y_i, \mathbf{y}_{-i})\\ \text{s.t.} \quad & g(\mathbf{y}) = \sum_{i=1}^{N} g_i(y_i) \leq 0 \quad \forall i \in \mathcal{N}\end{aligned} \tag{3}$$

where f_i and g_i are private information of player i. It is assumed that players communicate with each other via an undirected and connected graph $\mathcal{G}$.

Definition 1: A GNE of game (3) is an action profile $\mathbf{y}^* = \text{col}(\text{y}_\text{i}^*)_{\text{i}\in\mathcal{N}}$ satisfying

$$f_i(y_i^*, \mathbf{y}_{-i}^*) \leq f_i(y_i, \mathbf{y}_{-i}^*) \quad \forall y_i : (y_i, \mathbf{y}_{-i}^*) \in \Omega, i \in \mathcal{N}.$$

Here are assumptions about players' cost functions and the shared constraint.

Assumption 1: $f_i(\mathbf{y})$ is convex in y_i, continuously differentiable, and $\frac{\partial f_i}{\partial y_i}(\mathbf{y})$ is globally l_i-Lipschitz for any $i \in \mathcal{N}$, i.e., $\|\frac{\partial f_i}{\partial y_i}(\mathbf{y}) - \frac{\partial f_i}{\partial \bar{y}_i}(\bar{\mathbf{y}})\| \leq l_i\|\mathbf{y} - \bar{\mathbf{y}}\|\ \forall \mathbf{y}, \bar{\mathbf{y}} \in \mathbb{R}^{mN}$.

Assumption 2: $g_i(y_i)$ is convex and κ_1-Lipschitz continuous $\forall i \in \mathcal{N}$. Besides, there exists $\mathbf{y} \in \mathbb{R}^{mN}$ such that $g(\mathbf{y}) < 0$.

To deal with the coupling constraint, we obtain the primal–dual characterization for each player i via a Lagrangian

$$\mathcal{L}_i(y_i, \mathbf{y}_{-i}, \lambda_i) = f_i(y_i, \mathbf{y}_{-i}) + \lambda_i g(\mathbf{y}) \tag{4}$$

where $\lambda_i \in \mathbb{R}_+$ is its Lagrangian multiplier. A variable $\mathbf{y}^* = (y_i^*, \mathbf{y}_{-i}^*)$ is a GNE of (3) if and only if there exists a dual variable $\lambda_i^* \in \mathbb{R}_+$ that satisfies the following Karush–Kuhn–Tucker (KKT) conditions [32]:

$$\nabla_i f_i(y_i^*, \mathbf{y}_{-i}^*) + \lambda_i^* \nabla_i g(\mathbf{y}^*) = 0 \tag{5a}$$

$$g(\mathbf{y}^*) \leq 0 \tag{5b}$$

$$\lambda_i^* g(\mathbf{y}^*) = 0 \tag{5c}$$

where $\nabla_i f_i(\mathbf{y}) = \frac{\partial f_i(\mathbf{y})}{\partial y_i}$, and $\nabla_i g(\mathbf{y}) = \frac{\partial g(\mathbf{y})}{\partial y_i} = \nabla g_i(y_i)$. If $\lambda_1^* = \lambda_2^* = \cdots = \lambda_N^* = \lambda^*$, the GNE with λ^* is called a v-GNE, which means that all players are equally penalized for the constraint violation [33]. A v-GNE of game (3), denoted as $\mathbf{y}^*$, solves the following variational inequality VI(F, Ω):

$$\langle F(\mathbf{y}^*), \mathbf{y} - \mathbf{y}^* \rangle \geq 0 \quad \forall \mathbf{y} \in \Omega$$

where F is the pseudogradient (game mapping) of the game defined as

$$F(\mathbf{y}) = \text{col}(\nabla_i f_i(y_i, \mathbf{y}_{-i}))_{i\in\mathcal{N}}. \tag{6}$$

Assumption 3: $F(\mathbf{y})$ of game (3) is μ-strongly monotone on Ω, i.e.,

$$(\mathbf{y} - \bar{\mathbf{y}})^T (F(\mathbf{y}) - F(\bar{\mathbf{y}})) \geq \mu \|\mathbf{y} - \bar{\mathbf{y}}\|^2 \quad \forall \mathbf{y}, \bar{\mathbf{y}} \in \Omega$$

where $m > 0$.

Moreover, based on Assumption 1, we can easily derive that $F(\mathbf{y})$ is $\sqrt{N}\bar{l}$-Lipschitz, i.e.,

$$\|F(\mathbf{y}) - F(\bar{\mathbf{y}})\| \leq \sqrt{N}\bar{l} \|\mathbf{y} - \bar{\mathbf{y}}\| \quad \forall \mathbf{y}, \bar{\mathbf{y}} \in \Omega \tag{7}$$

where $\bar{l} = \max_i\{l_i\}$.

Remark 3: Assumptions 1–3 are quite standard in the literature of distributed GNE-seeking problems [6], [7], [8], [9] and necessary to enable players to learn the unique v-GNE in a distributed fashion. Specifically, Assumptions 1 and 2 ensure that there exists a solution to VI(F, Ω) [34]. Additionally, Assumption 2 implies that Slater's condition holds, ensuring that the optimal duality gap is zero, and thus, KKT conditions (5)

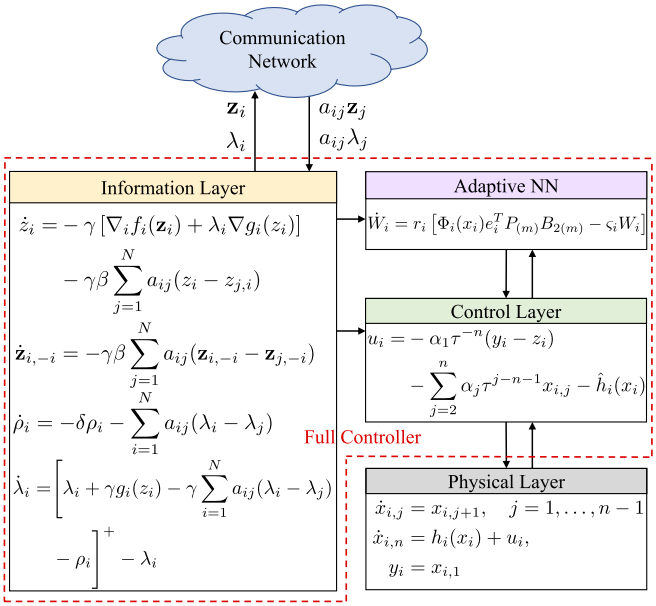

Fig. 1. Block diagram of player i with the proposed controller.

provide necessary and sufficient optimality conditions. Furthermore, Assumption 3 characterizes the uniqueness of the v-GNE in (3).

In this article, we will focus on the following problem.

Problem 1: For the high-order multiagent system (1) with unknown nonlinear dynamics, find a distributed controller u_i for player i such that given any $\varepsilon_{\mathbf{y}} > 0$, the outputs of players satisfy

$$\lim_{t\to\infty} \|\mathbf{y} - \mathbf{y}^*\|^2 \le \varepsilon_{\mathbf{y}}.$$

III. Algorithm Design

As the high-order players' decision outputs cannot be directly controlled due to inherent unknown nonlinear dynamics, we introduce an auxiliary variable $z_i \in \mathbb{R}^m$ for each player i as its virtual decision. We then solve an auxiliary noncooperative distributed game in the cyber layer for the players to drive z_i to y_i^*. Next, we propose distributed controllers for the players to track their virtual decisions. To compensate for the effect of nonlinear dynamics, the controller employs a shallow RBF NN to approximate the unknown nonlinear dynamics. The weight of the NN is adaptively updated based on a formulated online optimization problem.

Fig. 1 illustrates the proposed cyber-physical control system for player i. In the following, we will provide detailed explanations of the algorithms designed for the information layer, the control layer, and the adaptive NN.

A. Primal–Dual Update

We define $\mathbf{z} = \text{col}(z_i)_{i\in\mathcal{N}} = (z_i, \mathbf{z}_{-i})$ as the virtual decision profile of players, where $\mathbf{z}_{-i} = \text{col}(z_i)_{i\in\mathcal{N}\setminus\{i\}}$ is the virtual decision set of player i's opponents. The v-GNE of the virtual decisions satisfies the KKT conditions (5), making it a saddle point of the Lagrangian (4)

$$(\mathbf{z}^*, \lambda^*) = \max_{\lambda\in\mathbb{R}_+}\min_{z_i\in\mathbb{R}^m} \mathcal{L}_i(z_i, \mathbf{z}^*_{-i}, \lambda) \quad \forall i \in \mathcal{N}.$$

Therefore, the optimal primal variable $\mathbf{z}^*$ and the optimal dual variable λ^* can be obtained via iterative gradient descent and gradient ascent of the Lagrangian, respectively. In partial information settings, player i requires an estimation of all other players' virtual decisions to make gradient descent. Hence, we denote player i's estimate as $\mathbf{z}_i = \text{col}(z_{i,j})_{j\in\mathcal{N}} = (z_{i,i}, \mathbf{z}_{i,-i})$, where $\mathbf{z}_{i,-i} = \text{col}(z_{i,j})_{j\in\mathcal{N}\setminus\{i\}}$, $z_{i,j}$ is player i's estimate on player j's action, and $z_{i,i} = z_i$. Player i communicates $\mathbf{z}_i$ with its neighbors to compensate for the lack of the global information of the virtual action profile $\mathbf{z}$. Besides, player i keeps and communicates an estimate $\lambda_i \in \mathbb{R}_+$ of the global dual variable to ensure convergence to the v-GNE.

Inspired by the work in [8] and [9], we design the following distributed primal–dual algorithm for player i based on the KKT conditions in (5):

$$\dot{z}_i = -\gamma\left[\nabla_i f_i(\mathbf{z}_i) + \lambda_i\nabla g_i(z_i)\right] - \gamma\beta\sum_{j=1}^{N} a_{ij}(z_i - z_{j,i}) \tag{8a}$$

$$\dot{\mathbf{z}}_{i,-i} = -\gamma\beta\sum_{j=1}^{N} a_{ij}(\mathbf{z}_{i,-i} - \mathbf{z}_{j,-i}) \tag{8b}$$

$$\dot{\rho}_i = -\delta\rho_i - \sum_{i=1}^{N} a_{ij}(\lambda_i - \lambda_j) \tag{8c}$$

$$\dot{\lambda}_i = \left[\lambda_i + \gamma g_i(z_i) - \rho_i - \gamma\sum_{i=1}^{N} a_{ij}(\lambda_i - \lambda_j)\right]^+ - \lambda_i \tag{8d}$$

with $\boldsymbol{\rho}(0) = \text{col}(\rho_i(0))_{i\in\mathcal{N}} \in \text{Range}(L)$ and $\lambda_i(0) \in \mathbb{R}_+$, where $\beta, \gamma, \delta > 0$ are step sizes.

The update for z_i in algorithm (8) uses gradient descent of the Lagrangian function, with an additional term of consensus errors between its virtual primal variables and its neighbors' estimates. Similarly, the update for λ_i employs the projected gradient ascent of the local Lagrangian, with a term for consensus errors. Consensus communication schemes in the cyber layer facilitate players to reach an agreement on the estimates of global virtual decisions and dual variables. Additionally, an auxiliary variable ρ_i is employed to promote convergence of the dual estimates.

B. Distributed Controller Design

To regulate the output of high-order player i, we design the following controller:

$$u_i = -\alpha_1\tau^{-n}(y_i - z_i) - \sum_{j=2}^{n}\alpha_j\tau^{j-n-1}x_{i,j} - \hat{h}_i(x_i) \tag{9}$$

where $0 < \tau < 1$ is a control gain parameter, $\alpha_j > 0\ \forall j \in \{1, 2, \ldots, n\}$ are the step sizes for tracking the virtual action z_i, and $\hat{h}_i(x_i)$ is player i's approximations of $h_i(x_i)$, which will be introduced in Section III-C.

Since $y_i = x_{i,1}$, the tracking error of agent i is defined as $e_i = [(x_{i,1} - z_i)^T, \tau x_{i,2}^T, \tau^2 x_{i,3}^T, \ldots, \tau^{n-1} x_{i,n}^T]^T$. Based on (1) and the controller (9), the dynamics of the tracking error for player i can be written as

$$\dot{e}_i = \tau^{-1} A_{(m)} e_i - B_{1(m)} \dot{z}_i + \tau^{n-1} B_{2(m)} (h_i(x_i) - \hat{h}_i(x_i)) \tag{10}$$

where $A = \begin{bmatrix} \mathbf{0}_{(n-1)\times 1} & I_{n-1} \\ -\alpha_1 & [-\alpha_2, \ldots, -\alpha_n] \end{bmatrix}$, $B_1 = \begin{bmatrix} 1 \\ \mathbf{0}_{(n-1)\times 1} \end{bmatrix}$, and $B_2 = \begin{bmatrix} \mathbf{0}_{(n-1)\times 1} \\ 1 \end{bmatrix}$. We choose constants $\alpha_j, j \in \{1, 2, \ldots, n\}$ such that the matrix A is Hurwitz. For instance, when $m = 1$ and $n = 2$, we can choose $\alpha_1 = 1$ and $\alpha_2 = 2$ to make A Hurwitz. Therefore, there exists a positive-definite matrix $P \in \mathbb{R}^{n\times n}$ such that $A^T P + PA = -2I_n$.

Remark 4: The control gain τ is chosen to be a very small positive constant, and the precise relationship between τ and step sizes will be specified later. The designed controller (9) utilizes a high-gain feedback technique [35] and online nonlinearity compensation to effectively stabilize the high-order uncertain dynamics of the players and drive their outputs toward the trajectories of the desired virtual decisions. This results in the system being stabilized at the constrained optimal solutions.

C. NN-Based Dynamics Estimator

According to the universal approximation theorem [36], the continuous nonlinear function $h(v) : \mathbb{R}^{mn} \to \mathbb{R}^m$ can be approximated arbitrarily well on any compact set by an NN

$$h_{\text{NN}}(v) = W_{\text{NN}}^T \Phi(v) \tag{11}$$

where $v \in \mathbb{R}^{mn}$ is the input vector, $W_{\text{NN}} \in \mathbb{R}^{p\times m}$ is the weight vector, p is the number of neurons, $\Phi(v) = [\phi_1(v), \phi_2(v), \ldots, \phi_p(v)]^T \in \mathbb{R}^p$ is the basis function vector, and $\phi_j(v) : \mathbb{R}^{mn} \to \mathbb{R}$ is the RBF given by

$$\phi_j(v) = \exp\left[-\frac{(v - \mu_j)^T (v - \mu_j)}{2\sigma_j^2} \right]$$

in which $\mu_j \in \mathbb{R}^{mn}, \sigma_j > 0$ are the centers and width of the basis function, respectively.

Thus, on a compact set containing all possible trajectories of player i, for any given positive constant $\bar{\epsilon}_i$, there exists an ideal weight matrix W_i^* and the number of neurons p_i such that

$$h_i(x_i) = W_i^{*T} \Phi_i(x_i) + \epsilon_i \tag{12}$$

where $W_i^* \in \mathbb{R}^{p_i\times m}$, $\Phi_i \in \mathbb{R}^{p_i}$, and $\epsilon_i \in \mathbb{R}^m$ is the approximation error satisfying $\|\epsilon_i\| \le \bar{\epsilon}_i$.

Then, its nonlinear dynamics estimator is given by

$$\hat{h}_i(x_i) = W_i^T \Phi_i(x_i) \tag{13}$$

where $W_i \in \mathbb{R}^{p_i\times m}$ is the estimation of W_i^*.

We define a loss function for the player i's NN consisting of the tracking error and a regularization term as

$$J_i(W_i) = -\frac{1}{2\tau^n} e_i^T P A e_i + \frac{\varsigma_i}{2} \|W_i\|_F^2 \tag{14}$$

where $\varsigma_i > 0$ is the regularization parameter, usually avoiding overfitting in machine learning [37], and n is the order of player i's physical system mentioned in (1). We then formulate the following online optimization problem for player i to learn the optimal weights:

$$\min_{W_i} \quad J_i(W_i). \tag{15}$$

To solve the online optimization (15), we should obtain $\frac{\partial J_i}{\partial W_i}$. We get the partial differential of J_i

$$\partial J_i = -\frac{1}{2\tau^n} e_i^T (PA + AP^T) \partial e_i + \varsigma_i \text{Tr}\left[W_i^T \partial W_i \right].$$

Based on (10), it is derived that

$$\begin{aligned} \partial \dot{e}_i &= \tau^{-1} A_{(m)} \partial e_i - \tau^{n-1} B_{2(m)} \partial \hat{h}_i \\ &= \tau^{-1} A_{(m)} \partial e_i - \tau^{n-1} B_{2(m)} (\partial W_i)^T \Phi_i(x_i). \end{aligned}$$

In order to minimize $J_i(W_i)$, the following computationally and time-efficient static error backpropagation is presented [38], [39], [40]:

$$\partial \dot{e}_i = 0 \tag{16}$$

which results in $A_{(m)} \partial e_i = \tau^n B_{2(m)} (\partial W_i)^T \Phi_i(x_i)$. Therefore

$$\begin{aligned} \partial J_i &= -\frac{1}{2\tau^n} e_i^T PA \partial e_i - \frac{1}{2\tau^n} (\partial e_i)^T A^T P e_i + \varsigma_i \text{Tr}\left[W_i^T \partial W_i \right] \\ &= -\text{Tr}\left[B_{2(m)}^T P_{(m)}^T e_i \Phi_i(x_i)^T \partial W_i \right] + \varsigma_i \text{Tr}\left[W_i^T \partial W_i \right]. \end{aligned}$$

Based on the matrix derivatives [41], the gradient of $J_i(W_i)$ w.r.t. W_i is written as

$$\frac{\partial J_i(W_i)}{\partial W_i} = -\Phi_i(x_i) e_i^T P_{(m)} B_{2(m)} + \varsigma_i W_i.$$

Hence, we obtain the following gradient descent update rule for player i to learn the optimal weights:

$$\dot{W}_i = -r_i \frac{\partial J_i}{\partial W_i} = r_i \left[\Phi_i(x_i) e_i^T P_{(m)} B_{2(m)} - \varsigma_i W_i \right] \tag{17}$$

where $r_i > 0$ is the learning rate, also called adaptation gain.

Remark 5: Each player's NN adjusts online via the adaptive weight update rule defined in (17), without requiring any offline learning phase. Although RBF NNs are widely used in nonlinear multiagent systems for adaptive control, most existing weight update rules are designed using Lyapunov analysis [24], [25], [26], which lacks intuitive understanding. We present a novel closed-form loss function for the NN and formulate an online optimization problem for optimal weight learning. The weight update rule can be derived through direct gradient descent and backpropagation of tracking error, providing an intuitive perspective for adaptive control with RBF NNs.

Remark 6: Some studies use the projection algorithm to update the NN weights. However, this requires a positive constant to limit the weight matrix and ensure convergence of the error [22], [24], [42]. Unfortunately, we often lack knowledge of the weights or their upper bounds. To address this, we adopt a regularization technique commonly used in machine learning. This helps to reduce the potential for a high-gain control scheme,

particularly in cases where $W_i(t)$ might shift to extremely huge values.

The proposed controller for solving Problem 1 includes (8), (9), (13), and (17). Fig. 1 illustrates the entire closed-loop CPS for player i.

IV. Performance Analysis

This section analyzes the convergence performance of our proposed algorithm. First, we prove that the v-GNE of (3) is an equilibrium point of the auxiliary system (8). Then, we demonstrate that the entire closed-loop system, under the developed NN-based adaptive controller, can exponentially converge to the v-GNE with arbitrary accuracy.

To rewrite the closed-loop CPS in the proposed algorithms into a compact form, let us define

$$\mathcal{R}_i = \begin{bmatrix} \mathbf{0}_{m\times(i-1)m} & I_m & \mathbf{0}_{m\times(N-i)m} \end{bmatrix} \tag{18}$$

and $\mathcal{R} = \text{diag}(\mathcal{R}_i)_{i\in\mathcal{N}}$. Hence, $\mathcal{R}_i$ selects the ith m-dimensional component from an mN-dimensional vector, i.e., $\mathcal{R}_i\mathbf{z}_i = z_{i,i} = z_i$ and $\mathcal{R}\mathcal{Z} = \mathbf{z}$ where $\mathcal{Z} = \text{col}(\mathbf{z}_i)_{i\in\mathcal{N}}$.

Recalling that the dynamics of players are governed by (1), we can obtain the following compact form:

$$\begin{aligned}
\dot{\mathbf{x}} =& (I_N \otimes A_{(m)})\mathbf{x} - (I_N \otimes A_{(m)}B_{1(m)})\mathcal{R}\mathcal{Z} \\
&+ (I_N \otimes B_{2(m)})(H(\mathbf{x}) - \hat{H}(\mathbf{x})) && \text{(19a)}\\
\dot{\mathcal{Z}} =& -\gamma\mathcal{R}^T\left(\mathbf{F}(\mathcal{Z}) + G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda}\right) - \gamma\beta L_{(Nm)}\mathcal{Z} && \text{(19b)}\\
\dot{\boldsymbol{\rho}} =& -\delta\boldsymbol{\rho} + L\boldsymbol{\lambda} && \text{(19c)}\\
\dot{\boldsymbol{\lambda}} =& [\boldsymbol{\lambda} + \gamma\mathbf{g}(\mathcal{R}\mathcal{Z}) - \boldsymbol{\rho} - \gamma L\boldsymbol{\lambda}]^+ - \boldsymbol{\lambda} && \text{(19d)}
\end{aligned}$$

where $\boldsymbol{\rho} = \text{col}(\rho_i)_{i\in\mathcal{N}}$, $\boldsymbol{\lambda} = \text{col}(\lambda_i)_{i\in\mathcal{N}}$, $H(\mathbf{x}) = \text{col}(h_i(x_i))_{i\in\mathcal{N}}$, $\hat{H}(\mathbf{x}) = \text{col}(\hat{h}_i(x_i))_{i\in\mathcal{N}}$, $\mathbf{F}(\mathcal{Z}) = \text{col}(\nabla_i f_i(\mathbf{z}_i))_{i\in\mathcal{N}}$, $\mathbf{g}(\mathbf{z}) = \text{col}(g_i(z_i))_{i\in\mathcal{N}}$, and $G(\mathbf{z}) = \text{diag}(\nabla g_i(z_i))_{i\in\mathcal{N}}$. We call $\mathbf{F}(\mathcal{Z})$ defined on $\mathbb{R}^{N^2m}$ the extended pseudogradient since it is the extension of F in (6) to the augmented space of decisions and estimations. It should be noted that $\mathbf{F}(\mathbf{1}_N \otimes \mathbf{y}) = F(\mathbf{y})$, $\forall\mathbf{y} \in \mathbb{R}^{Nm}$. Moreover, the proposed CPS algorithm has an invariance property, namely that if $\boldsymbol{\rho}(0) \in \text{Range}(L)$, then $\boldsymbol{\rho} \in \text{Range}(L)$ along any trajectory of (19), by (19c).

The following lemma offers the relationship between a class of equilibria of system (8) and the v-GNE of the game (3).

Lemma 1: Under Assumptions 1–3, any equilibrium point $\text{col}(\bar{\mathcal{Z}}, \bar{\boldsymbol{\rho}}, \bar{\boldsymbol{\lambda}})$ of (8) with $\bar{\boldsymbol{\rho}} = \mathbf{0}$ satisfies $\bar{\mathcal{Z}} = \mathbf{1}_N \otimes \mathbf{y}^*$ and $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$, where the pair $(\mathbf{y}^*, \lambda^*)$ satisfies the KKT conditions in (5) with $\lambda_i^* = \lambda^*$. Therefore, $\mathbf{y}^*$ is the v-GNE of the game (3).

Proof: For any equilibrium $\text{col}(\bar{\mathcal{Z}}, \bar{\boldsymbol{\rho}}, \bar{\boldsymbol{\lambda}})$ of (8), it holds that

$$\begin{aligned}
\mathbf{0} =& \mathcal{R}^T\left(\mathbf{F}(\bar{\mathcal{Z}}) + G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}}\right) + \beta L_{(Nm)}\bar{\mathcal{Z}} && \text{(20a)}\\
\mathbf{0} =& -\delta\bar{\boldsymbol{\rho}} + L\bar{\boldsymbol{\lambda}} && \text{(20b)}\\
\mathbf{0} =& [\bar{\boldsymbol{\lambda}} + \gamma\mathbf{g}(\mathcal{R}\bar{\mathcal{Z}}) - \bar{\boldsymbol{\rho}} - \gamma L\bar{\boldsymbol{\lambda}}]^+ - \bar{\boldsymbol{\lambda}}. && \text{(20c)}
\end{aligned}$$

With $\bar{\boldsymbol{\rho}} = \mathbf{0}$, we have $L\bar{\boldsymbol{\lambda}} = 0$ by (20b), and therefore, there exists $\lambda^* \in \mathbb{R}_+$ such that $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$. Note that $(\mathbf{1}_N^T \otimes I_{Nm})\mathcal{R}^T = I_{Nm}$ and $(\mathbf{1}_N^T \otimes I_{Nm})L_{(Nm)} = \mathbf{0}$. Then, left multiplying both sides of (20a) with $(\mathbf{1}_N^T \otimes I_{Nm})$ yields

$$\mathbf{0} = \mathbf{F}(\bar{\mathcal{Z}}) + G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}}. \tag{21}$$

Substituting (21) into (20a) yields $L_{(Nm)}\bar{\mathcal{Z}} = 0$. Consequently, there is $\bar{\mathcal{Z}} = \mathbf{1}_N \otimes \mathbf{y}^*$ for some $\mathbf{y}^* \in \mathbb{R}^{Nm}$. Putting $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$ and $\bar{\mathcal{Z}} = \mathbf{1}_N \otimes \mathbf{y}^*$ into (21), we retrieve (5a) with $\lambda_i^* = \lambda^*$ for all $i \in \mathcal{N}$ since $\mathbf{F}(\bar{\mathcal{Z}}) = F(\mathbf{y}^*)$, $\mathcal{R}\bar{\mathcal{Z}} = \mathbf{y}^*$, and $G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}} = \nabla_{\mathbf{y}}g(\mathbf{y}^*)^T\lambda^*$. Then, using $\bar{\boldsymbol{\lambda}} = \mathbf{1}_N \otimes \lambda^*$ and $\bar{\mathcal{Z}} = \mathbf{1}_N \otimes \mathbf{y}^*$ into (20c), and premultiplying (20c) by $\mathbf{1}_N^T$, we retrieve (5b) and (5c) since $\mathbf{1}_N^T\mathbf{g}(\mathcal{R}\bar{\mathcal{Z}}) = g(\mathbf{y}^*)$, $\mathbf{1}_N^T L\bar{\boldsymbol{\lambda}} = 0$. ■

Remark 7: Due to the scenario of partial-decision information, the operator $\mathcal{R}^T\mathbf{F}(\mathcal{Z})$ is used instead of $F(\mathbf{z})$ in the noncooperative game. However, $\mathcal{R}^T\mathbf{F}$ is seldom monotone, causing difficulty in analyzing the convergence of distributed games. To address this issue, we use the restricted monotonicity property introduced in the following lemma, as inspired by the authors in [8] and [9]. This property is significant for our later convergence analysis.

Lemma 2: Consider Assumptions 1–3 hold and let

$$\Psi = \begin{bmatrix} \frac{\mu}{N} & -\frac{\bar{l}+\sqrt{N\bar{l}}}{2\sqrt{N}} \\ -\frac{\bar{l}+\sqrt{N\bar{l}}}{2\sqrt{N}} & \beta s_2(L) - \bar{l} \end{bmatrix}. \tag{22}$$

Then, for any $\beta > \frac{(\bar{l}+\sqrt{N\bar{l}})^2}{4\mu s_2(L)} + \frac{\bar{l}}{s_2(L)}$, $\Psi \succ 0$ and, for any $\mathbf{v} \in \mathbb{R}^{N^2m}$ and $\mathbf{v}' \in \text{Null}(\text{L}_{(\text{Nm})}) = \text{Range}(\mathbf{1}_\text{N} \otimes \text{I}_\text{Nm})$

$$\begin{aligned}
&(\mathbf{v}-\mathbf{v}')^T\left[\mathcal{R}^T\left(\mathbf{F}(\mathbf{v}) - \mathbf{F}(\mathbf{v}')\right) + \beta L_{Nn}(\mathbf{v}-\mathbf{v}')\right] \\
&\geq \bar{\mu}\left\|\mathbf{v}-\mathbf{v}'\right\|^2
\end{aligned} \tag{23}$$

where $\bar{\mu} = s_m(\Psi) > 0$.

Proof: The proof is similar to [8, Lemma 3]. ■

Lemma 3 (see [24], [43]): For a continuous function $V(t) \geq 0$ with bounded $V(0)\ \forall t \geq 0$, if it satisfies the following inequality:

$$\dot{V}(t) \leq -c_1V(0) + c_2$$

where c_1 and c_2 are positive constants, then there will be

$$V(t) \leq V(0)e^{-c_1t} + \frac{c_2}{c_1}\left(1 - e^{-c_1t}\right).$$

We are now ready to provide the convergence analysis for the CPS in (19).

Theorem 1: The multiagent system with the proposed controller exponentially converges to the v-GNE with an arbitrarily small error when the following conditions hold:

$$1 > \tau^n\kappa_5 + \tau\left(\frac{\gamma\omega^2}{2\bar{\mu}} + \frac{\gamma N\kappa_1^2\kappa_4^2}{2s_2(L)}\right) \tag{24}$$

$$\beta > \frac{(\bar{l}+\sqrt{N\bar{l}})^2}{4\mu s_2(L)} + \frac{\bar{l}}{s_2(L)} \tag{25}$$

where $\omega = \kappa_4(\bar{l} + \beta s_N(L) + \sqrt{N}\kappa_1\kappa_2)$, $\lambda^* \leq \kappa_2$, $\kappa_4 = \|PB_1\|, \kappa_5 = \|PB_2\|, \kappa_1$ and μ are mentioned in Assumptions 2 and 3, respectively, $\bar{\mu}$ is defined in Lemma 2, $\bar{l}$ is given in (7), N is the network size, and $s_2(L)$ is the second smallest eigenvalue of the Laplacian matrix L.

Proof: Let $\Theta = \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T + L^\dagger$, and $L^\dagger$ is the Moore–Penrose pseudoinverse of L. We construct a Lyapunov function

$$V = V_1 + V_2 + V_3 + V_4 + V_5 \tag{26}$$

where

$$V_1 = \frac{1}{2}\mathbf{e}^T\left(I_N \otimes P_{(m)}\right)\mathbf{e} \tag{27}$$

$$V_2 = \frac{1}{2}(\mathcal{Z}-\bar{\mathcal{Z}})^T(\mathcal{Z}-\bar{\mathcal{Z}}) \tag{28}$$

$$V_3 = \frac{1}{2}(\boldsymbol{\rho}-\bar{\boldsymbol{\rho}})^T\Theta(\boldsymbol{\rho}-\bar{\boldsymbol{\rho}}) \tag{29}$$

$$V_4 = \frac{1}{2}(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}) \tag{30}$$

$$V_5 = \frac{\tau^{n-1}}{2r_i}\sum_{i=1}^{N}\mathrm{Tr}\left[\tilde{W}_i^T\tilde{W}_i\right] \tag{31}$$

where $\tilde{W}_i = W_i - W_i^*$. Based on the properties of the pseudoinverse and $\mathrm{Null}(L) = \mathrm{Range}(\mathbf{1}_N)$, we can infer that $L^\dagger \succeq 0$ and $\mathrm{Null}(L^\dagger) = \mathrm{Null}(L)$. Since $\frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T \succeq 0$, we have $\Theta \succeq 0$. Therefore, $V \geq 0$.

For V_1, we have

$$\begin{aligned}\dot{V}_1 &= \mathbf{e}^T\left(I_N \otimes P_{(m)}\right)\dot{\mathbf{e}}_i \\ &= \tau^{-1}\mathbf{e}^T\left(I_N \otimes (PA)_{(m)}\right)\mathbf{e} - \mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\mathcal{R}\dot{\mathcal{Z}} \\ &\quad + \tau^{n-1}\mathbf{e}^T\left(I_N \otimes (PB_2)_{(m)}\right)\left(H(\mathbf{x}) - \hat{H}(\mathbf{x})\right).\end{aligned} \tag{32}$$

For the second term of (32), we have

$$\begin{aligned}&-\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\mathcal{R}\dot{\mathcal{Z}} \\ &= \gamma\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\left[F(\mathcal{Z}) + G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda} + \mathcal{R}\beta L_{Nn}\mathcal{Z}\right] \\ &= \gamma\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\left(F(\mathcal{Z}) - F(\bar{\mathcal{Z}})\right) \\ &\quad + \gamma\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda} - G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}}\right] \\ &\quad + \gamma\beta\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\mathcal{R}L_{Nmn}(\mathcal{Z}-\bar{\mathcal{Z}})\end{aligned} \tag{33}$$

where the first equality follows from $\mathcal{R}\mathcal{R}^T = I_{Nmn}$, and the second equality follows from Lemma 1. For the second term of (33)

$$\begin{aligned}&\gamma\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda} - G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}}\right] \\ &= \gamma\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda} - G(\mathcal{R}\mathcal{Z})^T\bar{\boldsymbol{\lambda}}\right. \\ &\quad \left. + G(\mathcal{R}\mathcal{Z})^T\bar{\boldsymbol{\lambda}} - G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}}\right] \\ &\leq \gamma\left\|G(\mathcal{R}\mathcal{Z})\right\|\|\mathbf{e}\|\|PB_1\|\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\| \\ &\quad + \gamma\left\|\bar{\boldsymbol{\lambda}}\right\|\|\mathbf{e}\|\|PB_1\|\left\|G(\mathcal{R}\mathcal{Z}) - G(\mathcal{R}\bar{\mathcal{Z}})\right\| \\ &\leq \gamma\sqrt{N}\kappa_1\kappa_4\|\mathbf{e}\|\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\| \\ &\quad + \gamma\sqrt{N}\kappa_1\kappa_2\kappa_4\|\mathbf{e}\|\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|.\end{aligned} \tag{34}$$

Therefore, we have the following for the second term of (32):

$$\begin{aligned}&\mathbf{e}^T\left(I_N \otimes (PB_1)_{(m)}\right)\left[F(\mathcal{Z}) + G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda} + \mathcal{R}\beta L_{Nn}\mathcal{Z}\right] \\ &\leq \gamma\kappa_4\left(\bar{l} + \beta s_N(L)\right)\|\mathbf{e}\|\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\| + \gamma\sqrt{N}\kappa_1\kappa_4\|\mathbf{e}\|\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\| \\ &\quad + \gamma\sqrt{N}\kappa_1\kappa_2\kappa_4\|\mathbf{e}\|\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|.\end{aligned} \tag{35}$$

For the third term of (32)

$$\begin{aligned}&\mathbf{e}^T\left(I_N \otimes (PB_2)_{(m)}\right)\left(H(\mathbf{x}) - \hat{H}(\mathbf{x})\right) \\ &= \sum_{i=1}^{N} e_i^T(PB_2)_{(m)}\left(h_i(\mathbf{x}_i) - \hat{h}_i(\mathbf{x}_i)\right) \\ &= \sum_{i=1}^{N} e_i^T(PB_2)_{(m)}\left(W_i^*\Phi_i(x_i) + \epsilon_i - W_i\Phi_i(x_i)\right) \\ &= \sum_{i=1}^{N} e_i^T(PB_2)_{(m)}\tilde{W}_i\Phi_i(x_i) + \mathbf{e}^T\left(I_N \otimes (PB_2)_{(m)}\right)\boldsymbol{\epsilon}\end{aligned}$$

where $\boldsymbol{\epsilon} = \mathrm{col}(\epsilon_i)_{i\in\mathcal{N}}$.

For V_2, we can derive

$$\begin{aligned}\dot{V}_2 &= (\mathcal{Z}-\bar{\mathcal{Z}})^T\dot{\mathcal{Z}} \\ &= \gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\left[-\mathcal{R}^T\left(\mathbf{F}(\mathcal{Z}) + G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda}\right) - \beta L_{(Nn)}\mathcal{Z}\right] \\ &= -\gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\mathcal{R}^T\left[\mathbf{F}(\mathcal{Z}) - \mathbf{F}(\bar{\mathcal{Z}})\right] \\ &\quad - \gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\mathcal{R}^T\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda} - G(\mathcal{R}\bar{\mathcal{Z}})^T\bar{\boldsymbol{\lambda}}\right] \\ &\quad - \gamma\beta(\mathcal{Z}-\bar{\mathcal{Z}})^T L_{(Nm)}(\mathcal{Z}-\bar{\mathcal{Z}})\end{aligned} \tag{36}$$

where the last equation follows from Lemma 1. For the first and the third terms of (36), Lemma 2 yields

$$\begin{aligned}&-\gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\mathcal{R}^T\left[F(\mathcal{Z}) - F(\bar{\mathcal{Z}})\right] \\ &\quad - \gamma\beta(\mathcal{Z}-\bar{\mathcal{Z}})^T L_{(Nm)}(\mathcal{Z}-\bar{\mathcal{Z}}) \\ &\leq -\gamma\bar{\mu}\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|^2\end{aligned} \tag{37}$$

if the condition (25) is satisfied.

For V_3, we have

$$\begin{aligned}\dot{V}_3 &= (\boldsymbol{\rho}-\bar{\boldsymbol{\rho}})^T\Theta\dot{\boldsymbol{\rho}} \\ &= (\boldsymbol{\rho}-\bar{\boldsymbol{\rho}})^T\Theta(-\delta\boldsymbol{\rho} + L\boldsymbol{\lambda}) \\ &= (\boldsymbol{\rho}-\bar{\boldsymbol{\rho}})^T\Theta\left[-\delta(\boldsymbol{\rho}-\bar{\boldsymbol{\rho}}) + L(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})\right] \\ &\leq -\delta s_m(\Theta)\|\boldsymbol{\rho}-\bar{\boldsymbol{\rho}}\|^2 + (\boldsymbol{\rho}-\bar{\boldsymbol{\rho}})^T(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})\end{aligned} \tag{38}$$

where the third equation holds from Lemma 1, and the inequality follows from $\boldsymbol{\rho} \in \mathrm{Range}(L)$ and $L^\dagger L = I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T$.

For V_4, we have

$$\begin{aligned}\dot{V}_4 &= (\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\dot{\boldsymbol{\lambda}} \\ &= (\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left\{[\boldsymbol{\lambda} + \gamma\mathbf{g}(\mathcal{R}\mathcal{Z}) - \boldsymbol{\rho} - \gamma L\boldsymbol{\lambda}]^+ - \boldsymbol{\lambda}\right\} \\ &\leq (\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left(\gamma\mathbf{g}(\mathcal{R}\mathcal{Z}) - \boldsymbol{\rho} - \gamma L\boldsymbol{\lambda}\right) \\ &\leq (\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left[\gamma\mathbf{g}(\mathcal{R}\mathcal{Z}) - \boldsymbol{\rho} - \gamma L\boldsymbol{\lambda} - \left(\gamma\mathbf{g}(\mathcal{R}\bar{\mathcal{Z}}) - \bar{\boldsymbol{\rho}} - \gamma L\bar{\boldsymbol{\lambda}}\right)\right] \\ &= \gamma(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left(\mathbf{g}(\mathcal{R}\mathcal{Z}) - \mathbf{g}(\mathcal{R}\bar{\mathcal{Z}})\right) - (\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T(\boldsymbol{\rho}-\bar{\boldsymbol{\rho}}) \\ &\quad - \gamma(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T L(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})\end{aligned} \tag{39}$$

where the two inequalities hold from [9, Lemma 1].

For V_5, we have

$$\begin{aligned}\dot{V}_5 &= \sum_{i=1}^{N}\frac{\tau^{n-1}}{r_i}\mathrm{Tr}\left[\tilde{W}_i^T\dot{\tilde{W}}_i\right]\\ &= \tau^{n-1}\sum_{i=1}^{N}\mathrm{Tr}\left\{\tilde{W}_i^T\left[\Phi_i(x_i)e_i^TP_{(m)}B_{2(m)}-\varsigma_iW_i\right]\right\}\\ &= \tau^{n-1}\sum_{i=1}^{N}\left\{e_i^T(PB_2)_{(m)}\tilde{W}_i^T\Phi_i(x_i)-\frac{\varsigma_i}{2}\mathrm{Tr}\left[\tilde{W}_i^T\tilde{W}_i\right]\right.\\ &\quad\left.+\frac{\varsigma_i}{2}\mathrm{Tr}\left[W_i^{*T}W_i^*-W_i^TW_i\right]\right\}\\ &\le \tau^{n-1}\sum_{i=1}^{N}e_i^TPB_2\tilde{W}_i^T\Phi_i(x_i)+\tau^{n-1}\sum_{i=1}^{N}\frac{\varsigma_i}{2}\mathrm{Tr}\left[W_i^{*T}W_i^*\right]\\ &\quad-r_m\underline{\varsigma}V_5 \end{aligned}\tag{40}$$

where $r_m=\min_i\{r_i\}$ and $\underline{\varsigma}=\min_i\{\varsigma_i\}$.

Therefore, the derivative of the constructed Lyapunov function has

$$\begin{aligned}\dot{V} &= \dot{V}_1+\dot{V}_2+\dot{V}_3+\dot{V}_4+\dot{V}_5\\ &\le \gamma\omega\|\mathbf{e}\|\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|+\tau^{n-1}\mathbf{e}^T(I_N\otimes(PB_2)_{(m)})\boldsymbol{\epsilon}\\ &\quad+\gamma\sqrt{N}\kappa_1\kappa_4\|\mathbf{e}\|\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\|-\tau^{-1}\|\mathbf{e}\|^2\\ &\quad-\gamma\bar{\mu}\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|^2-\delta s_m(\Theta)\|\boldsymbol{\rho}-\bar{\boldsymbol{\rho}}\|^2\\ &\quad-\gamma s_2(L)\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\|^2-r_m\underline{\varsigma}V_5+\Sigma_W\\ &\quad-\gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\mathcal{R}^T\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda}-G(\mathcal{R}\bar{\mathcal{Z}})\bar{\boldsymbol{\lambda}}\right]\\ &\quad+\gamma(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left(\mathbf{g}(\mathcal{R}\mathcal{Z})-\mathbf{g}(\mathcal{R}\bar{\mathcal{Z}})\right)\end{aligned}\tag{41}$$

where $\Sigma_W=\sum_{i=1}^{N}\frac{\varsigma_i}{2}\mathrm{Tr}[W_i^{*T}W_i^*]$. For the last two terms of (41), we have

$$\begin{aligned}&\gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\mathcal{R}^T\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda}-G(\mathcal{R}\bar{\mathcal{Z}})\bar{\boldsymbol{\lambda}}\right]\\ &\quad-\gamma(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left(\mathbf{g}(\mathcal{R}\mathcal{Z})-\mathbf{g}(\mathcal{R}\bar{\mathcal{Z}})\right)\\ &=\gamma(\mathbf{z}-\mathbf{x}^*)^T\left(\nabla\mathbf{g}(\mathbf{z})^T\boldsymbol{\lambda}-\nabla\mathbf{g}(\mathbf{x}^*)^T\bar{\boldsymbol{\lambda}}\right)\\ &\quad-\gamma(\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}})^T\left(\mathbf{g}(\mathbf{z})-\mathbf{g}(\mathbf{x}^*)\right)\\ &=\gamma\sum_{i=1}^{N}\left\{\lambda_i^T\left[\nabla g_i(z_i)(z_i-x_i^*)-(g_i(z_i)-g_i(x_i^*))\right]\right.\\ &\quad\left.-\lambda^{*T}\left[\nabla g_i(x_i^*)(z_i-x_i^*)-(g_i(z_i)-g_i(x_i^*))\right]\right\}.\end{aligned}$$

Based on the convexity of g_i, we have $\nabla g_i(z_i)(z_i-x_i^*)-(g_i(z_i)-g_i(x_i^*))\ge 0$, and $\nabla g_i(x_i^*)(z_i-x_i^*)-(g_i(z_i)-g_i(x_i^*))\le 0$. Since $\lambda_i(0)\in\mathbb{R}_+$, we have $\lambda_i\in\mathbb{R}_+$ for $t>0$. Moreover, according to $\lambda^*\in\mathbb{R}_+$, we obtain

$$\gamma(\mathcal{Z}-\bar{\mathcal{Z}})^T\mathcal{R}^T\left[G(\mathcal{R}\mathcal{Z})^T\boldsymbol{\lambda}-G(\mathcal{R}\bar{\mathcal{Z}})\bar{\boldsymbol{\lambda}}\right]\ge 0.\tag{42}$$

Furthermore, we have the following inequalities for the first three terms in (41) by applying Young's inequality:

$$\begin{aligned}&\gamma\omega\|\mathbf{e}\|\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|+\tau^{n-1}\mathbf{e}^T(I_N\otimes(PB_2)_{(m)})\epsilon\\ &+\gamma\sqrt{N}\kappa_1\kappa_4\|\mathbf{e}\|\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\|\\ &\le\left(\frac{\gamma\omega}{2\nu_1}+\frac{\gamma\sqrt{N}\kappa_1\kappa_4}{2\nu_2}+\frac{\tau^{n-1}\kappa_5}{2\nu_3}\right)\|\mathbf{e}\|^2+\frac{\gamma\nu_1\omega}{2}\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|^2\\ &+\frac{\gamma\nu_2\sqrt{N}\kappa_1\kappa_4}{2}\left\|\boldsymbol{\lambda}-\bar{\boldsymbol{\lambda}}\right\|^2+\frac{\nu_3\tau^{n-1}\kappa_5}{2}\bar{\epsilon}^2.\end{aligned}\tag{43}$$

Using (42) into (41), and letting $\nu_1=\frac{\bar{\mu}}{\omega}$, $\nu_2=\frac{s_2(L)}{\sqrt{N}\kappa_1\kappa_4}$, and $\nu_3=\frac{1}{2}$, we have

$$\begin{aligned}\dot{V} &\le -k_1V_1-\gamma\bar{\mu}V_2-\gamma s_2(L)V_3-\delta\frac{s_m(\Theta)}{s_M(\Theta)}V_4-r_m\underline{\varsigma}V_5\\ &\quad+\Sigma_W+\frac{\tau^{n-1}\kappa_5}{4}\|\bar{\epsilon}\|^2\\ &\le -k^*V+\Sigma_W+\frac{\tau^{n-1}\kappa_5}{4}\|\bar{\epsilon}\|^2\end{aligned}\tag{44}$$

where $k^*=\min\{k_1,\gamma\bar{\mu},\gamma s_2(L),\delta\frac{s_m(\Theta)}{s_M(\Theta)},r_m\underline{\varsigma}\}$, and $k_1=(\frac{1}{\tau}-\frac{\gamma\omega^2}{2\bar{\mu}}-\frac{\gamma N\kappa_1^2\kappa_4^2}{2s_2(L)}-\tau^{n-1}\kappa_5)/s_M(P)>0$ under the condition (24). According to Lemma 3, we conclude

$$\begin{aligned}V(t) &\le V(0)e^{-k^*t}+\frac{4\Sigma_W+\tau^{n-1}\kappa_5\|\bar{\epsilon}\|^2}{4\tilde{}k^*}\left(1-e^{-k^*t}\right)\\ &=\left(V(0)-\frac{4\Sigma_W+\bar{\epsilon}}{4\tilde{}k^*}\right)e^{-k^*t}+\frac{4\Sigma_W+\tau^{n-1}\kappa_5\|\bar{\epsilon}\|^2}{4\tilde{}k^*}.\end{aligned}\tag{45}$$

Since k_1, γ, δ, r_m can be arbitrarily large, k^* can be arbitrarily large so that $\frac{4\Sigma_W+\tau^{n-1}\kappa_5\|\bar{\epsilon}\|^2}{4\tilde{}k^*}$ is sufficiently small. ■

Remark 8: Due to the unknown nonlinear dynamics, controller design and convergence analysis become more complex. From (12), we can see that there is an approximation error even though the NN weight is optimal. Since the NN seldom learns the exact nonlinear dynamics, it is challenging for players' outputs to converge to the exact v-GNE, i.e., to rigorously prove $\dot{V}\le 0$. The convergence error leads to deviation from the optimality point and causes constraint violation at the equilibrium, which can be hazardous in safety-critical scenarios. To deal with the unavoidable error, we propose a controller and show that the error can be made arbitrarily small by choosing appropriate step sizes. Therefore, we cannot directly use the primal–dual update rule in [9] since it cannot achieve this goal. The equilibrium with $\bar{\boldsymbol{\rho}}=0$ and step sizes play key roles in our analysis.

It should be noted that the number of neurons in the one-layer RBF NN p does not affect the convergence proof or the step size conditions for *stabilization*. However, according to (45), the NNs approximation capacity, as indicated by the approximation error $\bar{\epsilon}$, affects the final *convergence accuracy*, given by the inequality

$$\begin{aligned}\|\mathbf{y}-\mathbf{y}^*\|^2 &\le \|\mathbf{e}\|^2+\left\|\mathcal{Z}-\bar{\mathcal{Z}}\right\|^2\\ &< \min\{2,1/s_m(P)\}V\\ &\le \min\{2,1/s_m(P)\}\frac{4\Sigma_W+\tau^{n-1}\kappa_5\|\bar{\epsilon}\|^2}{4\tilde{}k^*}\end{aligned}\tag{46}$$

where k^* depends on the step sizes. Therefore, when choosing the step sizes to achieve the desired convergence error $\varepsilon_{\mathbf{y}}$, the NNs approximation capacity must be taken into account. If the

TABLE I
COMPARISON OF SOME EXISTING WORKS WITH OURS

Algorithm	Players' dynamics	Coupling constraint
MIC [9]	×	✓
EOC [18]	First-order with uncertainties	×
LPC [19]	High-order with linearized uncertain parameters	×
Ours	High-order with uncertain nonlinearity	✓

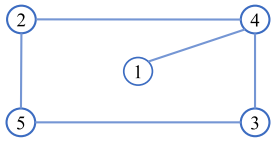

Fig. 2. Communication topology for vehicles.

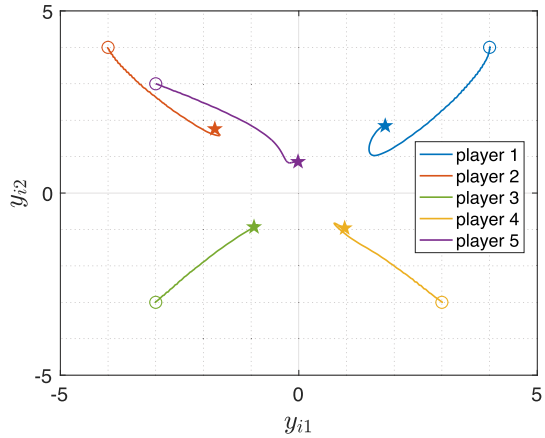

Fig. 3. Evolution of vehicles' positions in the 2-D plane. The hollow circles represent the initial positions of the vehicles while the filled stars denote the desired v-GNE $\mathbf{y}^*$.

approximation capacity is poor (i.e., $\bar{\epsilon}$ is large), we must choose small τ and large step sizes to make k^* sufficiently large. This ensures that (46) is smaller than the given $\varepsilon_{\mathbf{y}}$. Consequently, our algorithm ensures $(\mathbf{y}, \boldsymbol{\lambda})$ to be arbitrarily closed to the exact v-GNE $(\mathbf{y}^*, \boldsymbol{\lambda}^*)$. Even if the converged action profile $\bar{\mathbf{y}}$ violates the coupling constraint $g(\mathbf{y}) \leq 0$, we can drive it to a point that satisfies the coupling constraint, with a deviation arbitrarily small so that it remains within a tiny neighborhood of $\mathbf{y}^*$. In the future, we will study more advanced adaptive control and analysis methods to achieve exact convergence.

Remark 9: Incorporating the heavy-ball technique [44] or the second-order Nesterov method [45] may accelerate the primal–dual update in the information layer, which is an interesting future research direction. Moreover, heterogeneous step sizes may also aid convergence acceleration.

V. NUMERICAL SIMULATION

In this section, we evaluate the performance of our proposed controller through numerical simulations and compare it with some existing algorithms: the multi-integrator controller (MIC) for distributed v-GNE seeking developed in [9], the extended observer-based controller (EOC) designed in [18], and the linearly parameterized compensation (LPC) method proposed in [19]. A brief comparison between these works and ours is summarized in Table I.

A. Connectivity Control Game

Consider a connectivity control game with $N = 5$ velocity-actuated vehicles and a communication topology shown in Fig. 2. Each nonlinear vehicle moves in the 2-D plane, and x_i represents its position information, which is described as the following dynamics [24]:

$$\dot{x}_i = \begin{bmatrix} \dot{x}_{i1} \\ \dot{x}_{i2} \end{bmatrix} = u_i + \begin{bmatrix} x_{i2}\sin(\theta_{i1}x_{i1}) \\ x_{i1}\cos(\theta_{i12}x_{i2}^2) \end{bmatrix}$$
$$y_i = x_i.$$

Moreover, the cost function of vehicle i is

$$f_i(\mathbf{y}) = \iota_i(y_i) + c_i(\mathbf{y})$$

where $\iota_i(y_i)$ defines a local goal and $c_i(\mathbf{y})$ is an interconnection term corresponding to a collective goal [27]. The function $\iota_i(y_i)$ is defined as $\iota_i(y_i) = y_i^T w_{ii} y_i + y_i^T w_i$.

In our simulation, we set $\theta_{i1} = 10$, $\theta_{i2} = 11$ for $i \in \mathcal{N}$, $w_{11} = w_{55} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $w_{22} = w_{44} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$, $w_{33} = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$, $w_1 = [-6, -10]^T$, $w_2 = [14, -14]^T$, $w_3 = [12, 12]^T$, $w_4 = [-10, 10]^T$, and $w_5 = [2, -6]^T$. Moreover, $c_1(\mathbf{y}) = \|y_1 - y_4\|^2$, $c_2(\mathbf{y}) = \|y_2 - y_4\|^2$, $c_3(\mathbf{y}) = \|y_3 - y_1\|^2$, $c_4(\mathbf{y}) = \|y_4 - y_2\|^2$, and $c_5(\mathbf{y}) = \|y_5 - y_4\|^2$. Besides, $g_1(y_1) = \|y_1\|^2 - 7$, $g_2(y_2) = \|y_2\|^2 - 6$, $g_3(y_3) = \|y_3\|^2 - 1.44$, $g_4(y_4) = \|y_4\|^2 - 1.8$, and $g_5(y_5) = \|y_5\|^2 - 1$, which restricts a safe region for vehicles.

To implement the proposed algorithm, we can first set β according to (25) and γ to facilitate the convergence of virtual decisions $\mathbf{z}$. Subsequently, we can tune the control gain τ and choose large learning rates and regularization parameters to ensure that the players' outputs track the trajectories of $\mathbf{z}$. Finally, we can fine-tune the parameters in the NNs to reduce oscillations in the vicinity of the steady state. In this example, we choose $\alpha_1 = 12$, $\tau = 0.02$, $\gamma = 0.9$, $\beta = 3$, $\delta = 2$, $r_i = 3$, $\varsigma_i = 2$ for $i \in \mathcal{N}$, $p_i = 30$, μ_{ij} is uniformly distributed in $[-3, 3]$, and $\sigma_{ij} = 5\sqrt{2}$ for all $i \in \mathcal{N}, j \in \{1, 2, \ldots, p\}$.

Fig. 3 shows the trajectories of vehicles under the proposed algorithm, where the position decisions of vehicles approach an extremely small neighborhood of the v-GNE. Figs. 4 and 5 demonstrate the convergence performance and the evolutions of the coupling constraint under different algorithms, respectively.

The coupling constraint satisfaction only applies to the steady state and may be violated before the convergence due to the partial information setting. Players need to communicate with others and adjust their strategy to gradually get the desired outcome and satisfy the constraint. Therefore, existing literature

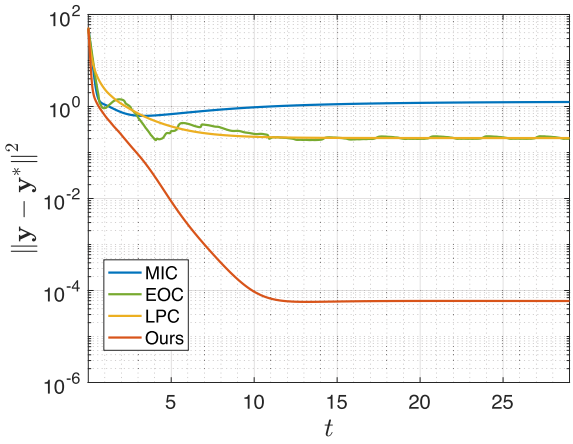

Fig. 4. Convergence performance of the vehicles' decisions under MIC, EOC, LPC, and our algorithms.

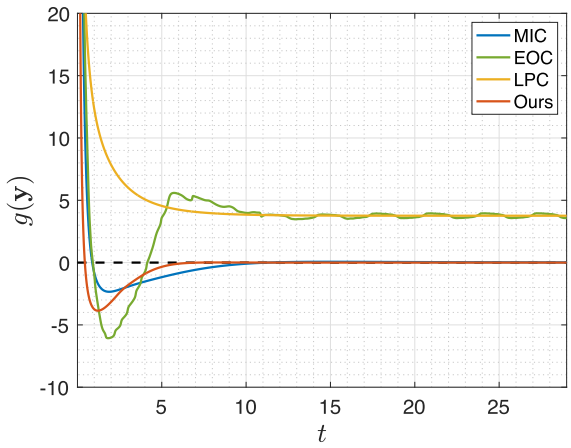

Fig. 5. Evolutions of the coupling constraint under MIC, EOC, LPC, and our algorithm. The coupling constraints convergent to or below the black dashed line indicate asymptotic satisfaction.

all focuses on pursuing asymptotic satisfaction of the coupling constraints [9], [46], [47], [48], [49], as we do in this work. From Figs. 4 and 5, we can see that our algorithm achieves exponential convergence to a small neighborhood of the v-GNE and asymptotic satisfaction of the coupling constraint. However, MIC fails to converge to the v-GNE due to ignoring vehicles' dynamics, and the outputs under EOC and LPC violate the constraint, and the errors are larger than those in our algorithm.

B. Energy Consumption Game

Consider an energy consumption game with $N = 6$ Van der Pol oscillators as the following second-order system [28]:

$$\begin{aligned} \dot{x}_{i,1} &= x_{i,2} \\ \dot{x}_{i,2} &= -b_i x_{i,1} + \bar{b}_i(1 - x_{i,1}^2)x_{i,2} + u_i \\ y_i &= x_{i,1} \end{aligned} \tag{47}$$

where $x_{i,1}$ and $x_{i,2}$ are oscillator i's electric current and voltage, respectively. The consumption function is defined as $f_i(y_i) = m_i(y_i - o_i) + (p_i \sum_{i=1}^{N} y_i + q_i)y_i$. These oscillators communicate based on the topology shown in Fig. 6. In our simulation, we set $b_i = 2$, $\bar{b}_i = 3$, $m_i = 2$, $p_i = 0.1$ $\forall i \in \mathcal{N}$, $o_1 = o_4 = o_5 = 8$, $o_2 = 10$, $o_3 = o_6 = 6$, $q_1 = 9.22$, $q_2 = 34.65$, $q_3 = 10.03$, $q_4 = 33$, $q_5 = 41.2$, $q_5 = 35.45$, and $g(\mathbf{y}) = \sum_{i=1}^{N} y_i - 33.5$ restricting a safe upper bound of the total current of oscillators. For the NN-based controller, we choose $\alpha_1 = \alpha_2 = 3$, $\tau = 0.1$, $\gamma = 5$, $\beta = 15$, $\delta = 5$, $r_i = 60$, $\varsigma_i = 5, p_i = 10$, μ_{ij} is uniformly distributed in $[-2, 2]$, and $\sigma_{ij} = 2.5$ $\forall i \in \mathcal{N}, j \in \{1, 2, \ldots, p\}$.

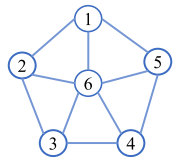

Fig. 6. Communication topology for oscillators.

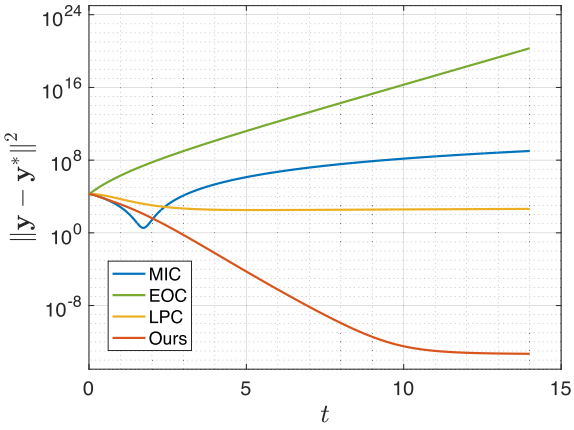

Fig. 7. Convergence performance of the oscillators' decisions under MIC, EOC, LPC, and our algorithm.

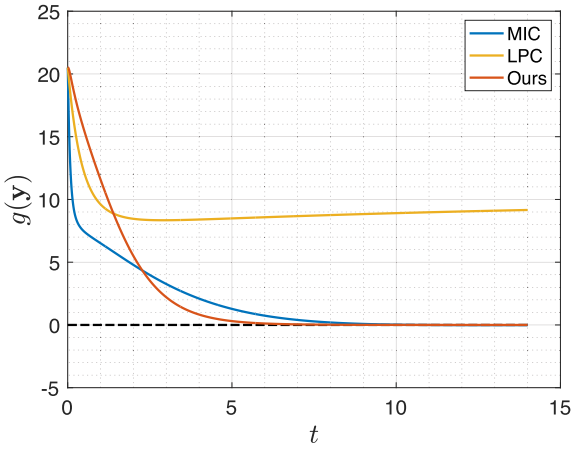

Fig. 8. Evolutions of the coupling constraint under MIC, LPC, and our algorithm. The coupling constraints convergent to or below the black dashed line indicate asymptotic satisfaction. We omit the evolution of the coupling constraint under EOC here since it diverges significantly.

Fig. 7 illustrates the convergence performance of various algorithms. It can be observed that the EOC proposed in [18] is unable to handle the high-order uncertain players. Moreover, the outputs of the oscillators diverge under MIC and converge to suboptimal solutions with LPC. In contrast, our algorithm successfully achieves convergence of players' decisions to the v-GNE with an extremely small error. Moreover, Fig. 8 reveals that the coupling constraint is violated when using LPC since it does not take this restriction into account while our algorithm gradually satisfies $g(\mathbf{y}) \leq 0$.

VI. Conclusion

This article develops a novel controller for distributed v-GNE-seeking problems where players possess inherent unknown nonlinear dynamics. The proposed methodology leverages NNs with Gaussian radial basis activation functions to learn the nonlinear dynamics in real time. It is guaranteed that the v-GNE can be achieved exponentially with an arbitrarily small error. Finally, simulation results demonstrate the effectiveness of this algorithm in learning the nonlinear dynamics to achieve the v-GNE.

Future directions may involve utilizing more advanced adaptive control to ensure convergence to the exact v-GNE, as well as accounting for time-varying external disturbances.

References

[1] D. Niyato and E. Hossain, "A game-theoretic approach to competitive spectrum sharing in cognitive radio networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2007, pp. 16–20.

[2] M. Ye and G. Hu, "Game design and analysis for price-based demand response: An aggregate game approach," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 720–730, Mar. 2017.

[3] U. Challita, W. Saad, and C. Bettstetter, "Deep reinforcement learning for interference-aware path planning of cellular-connected UAVs," in *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–7.

[4] C. W. Zaw, H. Nguyen, W. Tran, Z. S. Han, and C. S. Hong, "Generalized Nash equilibrium game for radio and computing resource allocation in co-located MEC," in *Proc. IEEE Int. Conf. Commun.*, 2020, pp. 1–6.

[5] L. Shi, J. Néstor, H. Marcano, and R. H. Jacobsen, "Multi-UAV path coordination through generalized potential games," in *Proc. Int. Conf. Automat. Robot. Appl.*, 2022, pp. 125–129.

[6] S. Liang, P. Yi, and Y. Hong, "Distributed Nash equilibrium seeking for aggregative games with coupled constraints," *Automatica*, vol. 85, pp. 179–185, 2017.

[7] K. Lu, G. Jing, and L. Wang, "Distributed algorithms for searching generalized Nash equilibrium of noncooperative games," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 2362–2371, Jun. 2019.

[8] L. Pavel, "Distributed GNE seeking under partial-decision information over networks via a doubly-augmented operator splitting approach," *IEEE Trans. Autom. Control*, vol. 65, no. 4, pp. 1584–1597, Apr. 2020.

[9] M. Bianchi and S. Grammatico, "Continuous-time fully distributed generalized Nash equilibrium seeking for multi-integrator agents," *Automatica*, vol. 129, 2021, Art. no. 109660.

[10] R. Ortega, J. A. L. Perez, P. J. Nicklasson, and H. J. Sira-Ramirez, *Passivity-Based Control of Euler–Lagrange Systems: Mechanical, Electrical and Electromechanical Applications*. Berlin, Germany: Springer, 2013.

[11] M. Lu and L. Liu, "Leader–following consensus of multiple uncertain Euler–Lagrange systems with unknown dynamic leader," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4167–4173, Oct. 2019.

[12] A. R. Ibrahim and T. Hayakawa, "Nash equilibrium seeking with linear time-invariant dynamic agents," in *Proc. Amer. Control Conf.*, 2019, pp. 1202–1207.

[13] M. Guo and C. De Persis, "Linear quadratic network games with dynamic players: Stabilization and output convergence to Nash equilibrium," *Automatica*, vol. 130, 2021, Art. no. 109711.

[14] X. Cai, F. Xiao, B. Wei, M. Yu, and F. Fang, "Nash equilibrium seeking for general linear systems with disturbance rejection," *IEEE Trans. Cybern.*, vol. 53, no. 8, pp. 5240–5249, Aug. 2023.

[15] Z. Deng, Y. Liu, and T. Chen, "Generalized Nash equilibrium seeking algorithm design for distributed constrained noncooperative games with second-order players," *Automatica*, vol. 141, 2022, Art. no. 110317.

[16] Z. Deng and S. Liang, "Distributed algorithms for aggregative games of multiple heterogeneous Euler–Lagrange systems," *Automatica*, vol. 99, pp. 246–252, 2019.

[17] Z. Deng, "Distributed Nash equilibrium seeking for aggregative games with second-order nonlinear players," *Automatica*, vol. 135, 2022, Art. no. 109980.

[18] M. Ye, "Distributed robust seeking of Nash equilibrium for networked games: An extended state observer-based approach," *IEEE Trans. Cybern.*, vol. 52, no. 3, pp. 1527–1538, Mar. 2022.

[19] Y. Tang and P. Yi, "Nash equilibrium seeking for high-order multiagent systems with unknown dynamics," *IEEE Trans. Control Netw. Syst.*, vol. 10, no. 1, pp. 321–332, Mar. 2023.

[20] M. M. Polycarpou and P. A. Ioannou, *Identification and Control of Nonlinear Systems Using Neural Network Models: Design and Stability Analysis*. Princeton, NJ, USA: Citeseer, 1991.

[21] A. J. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches*. Hoboken, NJ, USA: Wiley, 2006.

[22] M. Ye, L. Ding, and J. Yin, "Distributed robust-equilibrium seeking for mixed-order games by a neural-network-based approach," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 53, no. 8, pp. 4808–4819, 2023.

[23] D. Li, M. Ye, L. Ding, and S. Xu, "Distributed Nash equilibrium computation with uncertain dynamics and disturbances," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1376–1385, May/Jun. 2022.

[24] Z.-G. L. Hou, L. Cheng, and M. Tan, "Decentralized robust adaptive control for the multiagent system consensus problem using neural networks," *IEEE Trans. Syst., Man, Cybern. Part B*, vol. 39, no. 3, pp. 636–647, Jun. 2009.

[25] J. Qin, G. Zhang, W. X. Zheng, and Y. Kang, "Neural network-based adaptive consensus control for a class of nonaffine nonlinear multiagent systems with actuator faults," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3633–3644, Dec. 2019.

[26] X. Jin, X. Zhao, J. Yu, X. Wu, and J. Chi, "Adaptive fault-tolerant consensus for a class of leader-following systems using neural network learning strategy," *Neural Netw.*, vol. 121, pp. 474–483, 2020.

[27] M. S. Stankovic, K. H. Johansson, and D. M. Stipanovic, "Distributed seeking of Nash equilibria with applications to mobile sensor networks," *IEEE Trans. Autom. Control*, vol. 57, no. 4, pp. 904–919, Apr. 2012.

[28] M. Sinha, F. Dörfler, B. B. Johnson, and S. V. Dhople, "Uncovering droop control laws embedded within the nonlinear dynamics of Van der Pol oscillators," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 2, pp. 347–358, Jun. 2017.

[29] M. Ye, "Distributed Nash equilibrium seeking for games in systems with bounded control inputs," *IEEE Trans. Autom. Control*, vol. 66, no. 8, pp. 3833–3839, Aug. 2021.

[30] G. Binetti, A. Davoudi, F. L. Lewis, D. Naso, and B. Turchiano, "Distributed consensus-based economic dispatch with transmission losses," *IEEE Trans. Power Syst.*, vol. 29, no. 4, pp. 1711–1720, Jul. 2014.

[31] X. Yi, X. Li, T. Yang, L. Xie, T. Chai, and K. H. Johansson, "Distributed bandit online convex optimization with time-varying coupled inequality constraints," *IEEE Trans. Autom. Control*, vol. 66, no. 10, pp. 4620–4635, Oct. 2021.

[32] Y. Zou, B. Huang, Z. Meng, and W. Ren, "Continuous-time distributed Nash equilibrium seeking algorithms for non-cooperative constrained games," *Automatica*, vol. 127, 2021, Art. no. 109535.

[33] F. Facchinei and C. Kanzow, "Generalized Nash equilibrium problems," *Ann. Operations Res.*, vol. 175, no. 1, pp. 177–211, 2010.

[34] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Berlin, Germany: Springer, 2003.

[35] J. Liu et al., "Sliding mode control of grid-connected neutral-point-clamped converters via high-gain observer," *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 4010–4021, Apr. 2022.

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

[37] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, vol. 112. Berlin, Germany: Springer, 2013.

[38] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 1, no. 1, pp. 4–27, Mar. 1990.
[39] I. Choudhary, K. Assaleh, and M. AlHamaydeh, "Nonlinear autoregressive exogenous artificial neural networks for predicting buckling restrained braces force," in *Proc. Int. Symp. Mechatron. Appl.*, 2012, pp. 1–5.
[40] A. Sharifi, A. Sharafian, and Q. Ai, "Adaptive MLP neural network controller for consensus tracking of multi-agent systems with application to synchronous generators," *Expert Syst. Appl.*, vol. 184, 2021, Art. no. 115460.
[41] K. B. Petersen et al., "The Matrix Cookbook," *Tech. Univ. Denmark*, Kongens Lyngby, Denmark, 2008.
[42] D. Wang, H. Ma, and D. Liu, "Distributed control algorithm for bipartite consensus of the nonlinear time-delayed multi-agent systems with neural networks," *Neurocomputing*, vol. 174, pp. 928–936, 2016.
[43] S. S. Ge and C. Wang, "Adaptive neural control of uncertain MIMO nonlinear systems," *IEEE Trans. Neural Netw.*, vol. 15, no. 3, pp. 674–692, May 2004.
[44] R. Xin and U. A. Khan, "Distributed heavy-ball: A generalization and acceleration of first-order methods with gradient tracking," *IEEE Trans. Autom. Control*, vol. 65, no. 6, pp. 2627–2633, Jun. 2020.
[45] X. Wang, S. Yang, Z. Guo, and T. Huang, "A second-order projected primal-dual dynamical system for distributed optimization and learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 6568–6577, Sep. 2023.
[46] F. Liu, J. Yu, Y. Hua, X. Dong, Q. Li, and Z. Ren, "Dynamic generalized Nash equilibrium seeking for N-coalition noncooperative games," *Automatica*, vol. 147, 2023, Art. no. 110746.
[47] P. Yi and L. Pavel, "Distributed generalized Nash equilibria computation of monotone games via double-layer preconditioned proximal-point algorithms," *IEEE Trans. Control Netw. Syst.*, vol. 6, no. 1, pp. 299–311, Mar. 2019.
[48] Z. Deng, "Distributed generalized Nash equilibrium seeking algorithm for nonsmooth aggregative games," *Automatica*, vol. 132, 2021, Art. no. 109794.
[49] K. Lu, G. Li, and L. Wang, "Online distributed algorithms for seeking generalized Nash equilibria in dynamic environments," *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2289–2296, May 2021.

Wei Huo received the B.S. degree in electronic and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2020. She is currently working toward the Ph.D. degree in electrical and computer engineering with the Hong Kong University of Science and Technology, Hong Kong.

Her current research interests include multiagent systems, event-triggered mechanisms, distributed optimization, and privacy protection.

Lingying Huang received the B.S. degree in electrical engineering and automation from Southeast University, Jiangsu, China, in 2017, and the Ph.D. degree in electrical and computer engineering from the Hong Kong University of Science and Technology, Hong Kong, in 2021.

She is currently a Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. In 2015, she was in a summer program with Georgia Tech University, Atlanta, GA, USA. Her current research interests include intelligent vehicles, cyber-physical system security/privacy, networked state estimation, event-triggered mechanisms, and distributed optimization.

Subhrakanti Dey received the bachelor's and master's degrees in technology from the Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology, Kharagpur, Kharagpur, India, in 1991 and 1993, respectively, and the Ph.D. degree from the Department of Systems Engineering, Research School of Information Sciences and Engineering, Australian National University, Canberra, ACT, Australia, in 1996.

He is currently a Professor with Hamilton Institute, National University of Ireland, Maynooth, Ireland. Prior to this, he was a Professor with the Department of Engineering Sciences, Uppsala University, Uppsala, Sweden, from 2013 to 2017, a Professor with the Department of Electrical and Electronic Engineering, University of Melbourne, Parkville, Australia, from 2000 until early 2013, and a Professor of Telecommunications with the University of South Australia during 2017–2018. From 1995 to 1997, and 1998 to 2000, he was a Postdoctoral Research Fellow with the Department of Systems Engineering, Australian National University. From 1997 to 1998, he was a Postdoctoral Research Associate with the Institute for Systems Research, University of Maryland, College Park, MD, USA. His current research interests include wireless communications and networks, signal processing for sensor networks, networked control systems, and molecular communication systems.

Dr. Dey is currently a Senior Editor on the Editorial Board of the IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS, and an Associate Editor/Editor for *Automatica,* IEEE CONTROL SYSTEMS LETTERS, and IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He was also an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2007 to 2010 and 2014 to 2018, the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 2004 to 2007, and Elsevier *Systems and Control Letters* from 2003 to 2013.

Ling Shi (Fellow, IEEE) received the B.E. degree in electrical and electronic engineering from the Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2002, and the Ph.D. degree in control and dynamical systems from the California Institute of Technology, Pasadena, CA, USA, in 2008.

He is currently a Professor with the Department of Electronic and Computer Engineering, and the Associate Director of Robotics Institute, both with the Hong Kong University of Science and Technology. His research interests include cyber-physical systems security, networked control systems, sensor scheduling, event-based state estimation, and exoskeleton robots.

Dr. Shi was an editorial board member for the European Control Conference from 2013 to 2016. He was a Subject Editor for the *International Journal of Robust and Nonlinear Control* from 2015 to 2017, an Associate Editor for the IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS from 2016 to 2020, an Associate Editor for IEEE CONTROL SYSTEMS LETTERS from 2017 to 2020, and an Associate Editor for a special issue on Secure Control of Cyber-Physical Systems in the IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS from 2015 to 2017. He was also the General Chair of the 23rd International Symposium on Mathematical Theory of Networks and Systems (MTNS 2018). He is a member of the Young Scientists Class 2020 of the World Economic Forum.