Robust Adaptive Leader-Following Formation Control of Nonlinear Multiagents Using Three-Layer Neural Networks

Kiarash Aryankia and Rastko Selmic, Senior Member, IEEE

Abstract—This article studies a formation control problem for a group of heterogeneous, nonlinear, uncertain, input-affine, second-order agents modeled by a directed graph. A tunable neural network (NN) is presented, with three layers (input, two hidden, and output) that can approximate an unknown nonlinearity. Unlike one- or two-layer NNs, this design has the advantage of being able to set the number of neurons in each layer ahead of time rather than relying on trial and error. The NN weights tuning law is rigorously derived using the Lyapunov theory. The formation control problem is tackled using a robust integral of the sign of the error feedback and NNs-based control. The robust integral of the sign of the error feedback compensates for the unknown dynamics of the leader and disturbances in the agent errors, while the NN-based controller accounts for the unknown nonlinearity in the multiagent system. The stability and semi-global asymptotic tracking of the results are proven using the Lyapunov stability theory. The study compares its results with two others to assess the effectiveness and efficiency of the proposed method.

Index Terms—Formation control, multilayer neural networks (NNs), nonlinear multiagent system, second-order system.

I. INTRODUCTION

Multiagent systems have received significant attention from researchers due to their various applications. Multiagent systems have a significant application in formation control, which has been inspired by nature. In [1], various categories of formation control problems were reviewed. The displacement control, one of the main categories of formation control problems, includes leader-following consensus problem and distributed cooperative tracking control problem [2], [3], [4], [5]. In multiagent system formation control, single and double integrator dynamics are frequently used to model the system [4], [6], [7], as well as nonlinear dynamics [2], [3], [8], [9], [10]. Some literature that considered unknown nonlinear dynamics utilized neural networks (NN) or fuzzy logic controllers to approximate the uncertain nonlinearity in the system dynamics. There are two main reasons for this: first, the multilayer NNs and fuzzy logic controllers are universal approximators that can approximate any continuous and smooth function over a compact set with desired accuracy [3], [11] and second, a precise model for the dynamics of a multiagent system is not required, which eliminates an extra effort to acquire accurate system model [12].

In formation control problems, intelligent control solutions mainly used one-layer NNs, for example, radial basis function NNs (RBFNNs) [2], [10], [13], [14] or two-layer NNs [8], to compensate for the unknown nonlinearities in the system dynamics. These studies used the universal approximation property, where the RBFNNs and the Gaussian activation functions were utilized. In RBFNN, hidden layer neurons are usually uniformly distributed on a regular lattice, or they are randomly initialized. Wang and Hill [15] constructed an RBFNN, where the distance of any two adjacent centers of the bell-shaped activation functions was the same as the width of the activation function.

An RBFNN can approximate a nonlinearity if the input variables of the activation function are located in certain neighborhoods of the RBF centers [15]. Thus, one can select a large number of neurons (not a priori known) and distribute activation functions over a regular lattice uniformly to cover the compact set. A one-layer NN with a tunable, hidden layer in the form of linear-in-the-parameters can approximate a nonlinearity in the dynamics of a nonlinear system at the desired accuracy. The exact number of neurons for control systems design is not clearly established. Although the self-organizing receptive field method was introduced in [16] to optimize the locations in RBFNN, these results have not been used in the formation control problem to approximate the unknown nonlinearity.

These limitations and restrictions motivated us to propose a three-layer NN with two hidden layers, where the numbers of neurons in hidden layers are set based on the order of the agents’ model (see Remark 9). The main motivation for employing a three-layer NN in this study is to overcome some limitations of simpler NN-based control methods that rely on trial-and-error hyperparameter tuning. Increasing the number of layers and neurons inevitably increases computational complexity. However, setting the number of neurons a priori reduces the controller design time.
To address the issue of the number of neurons in an NN, a self-structuring NN was proposed in [12], where a neuron is divided into two by satisfying a specific condition, but the output of these two neurons remains identical to the output of a neuron without division. This work was extended to a nonlinear multiagent system [8], where authors designed an observer to estimate the states of the leader for each agent and addressed the cooperative tracking control problem of a leader. Compared with existing results, we used two hidden layers instead of one [8], [12], [17], [18], [19], and we developed tuning laws for all three layers. The other novelty is that the number of neurons is fixed.

Xian et al. [20] used robust integral of the sign of the error (RISE) feedback to eliminate the chattering effect of the sliding mode controller and improve the system performance to semi-global asymptotic stability instead of being uniformly ultimate bounded (UBU). These findings were further extended in [17], [18], [21], [22], and [23]. We also employed this technique to enhance the performance and stability of the closed-loop system, even in the presence of disturbances and NN approximation errors. Although a RISE feedback control can potentially exceed the actuator’s limit due to the large integral gain, it has been shown that it can be implemented in real-time applications, for example, [23]. Xian et al. [20] used the RISE feedback to compensate for nonlinearity in a class of higher order, multi-input multioutput systems. These results were extended in [17], where a tracking problem of an uncertain system was addressed by an NN-based controller. Patre et al. [17] used the discontinuous projection algorithm, which requires an a priori knowledge of the convex set and maximum and minimum of NN’s weights [24]. In a similar study, Dierks and Jagannathan [21] used two-layer NN (input, hidden, and output layers), where only the weights between two outer layers can be tuned. The tuning laws in [21] were derived using the Lyapunov stability theory. These results were further extended in [18] and [23], where a discontinuous projection algorithm has tuned the NN weights matrices between any two consecutive layers for a two-layer NN.

Compared to [25], here, we consider the general nonlinear uncertain, input affine, multiagents modeled by a directed graph. Furthermore, we have included rigorous mathematical analysis and presented extensive simulation results to validate and verify the efficiency of the proposed method. We have also introduced a performance index to demonstrate the advantages of the proposed method in comparison with the two previously established results.

The novelty of our work in comparison with [10], [17], [18], [20], [21], [23], and [24] is that we employ a three-layer NN comprised of an input layer, two hidden layers, and an output layer. Moreover, all NN weights matrices are tunable, and the tuning laws are derived through the Lyapunov theory.

The NN-based formation control results addressing the displacement based, leader-following problem have demonstrated only uniform ultimate bounded stability due to NN approximation error and lack of communication between the leader and all followers [3], [5], [8], [26], [27], [28]. In this article, we propose a RISE feedback controller with an NN with two hidden layers to achieve semi-global, asymptotic stability for the leader-following formation control for heterogeneous, second-order, uncertain, input-affine, nonlinear multiagent systems.

The main contributions of this article are listed as follows. 1) We consider that the leader is connected to at least one of the other agents. Unlike existing literature on NN-based control, such as [17], [18], [19], [21], and [23], where the desired trajectory signals (i.e., information from the leader) are typically fed into the first layer of the NN, our approach differs by designating the NN without utilizing these signals. Instead, we achieve leader following through a RISE feedback controller that incorporates a robustifying term to account for the effects of the leader dynamics and unknown disturbances in the tracking error of each agent.

2) We determined the number of neurons in each layer and derived the corresponding tuning laws for the NN weights matrices. While the majority of the existing literature on NN-based control has focused on one- or two-layer NNs [2], [5], [18], [19], [23], [28], [29], we have developed our approach using a three-layer NN architecture where the number of neurons in each NN layer is set based on a priori knowledge of the system, such as the number of system states.

3) The semi-global, asymptotic, leader-following performance is rigorously proven, which distinguishes our work from [2], [3], [5], and [27].

Following is an overview of the remainder of this article. In Section II, preliminaries and problem formulation are presented. Section III provides the control design of RISE feedback and adaptive NN weights matrices tuning laws. The proposed method’s stability is rigorously proven using the Lyapunov stability theory. Its effectiveness is demonstrated through numerical simulations and comparisons with two other methods in Section IV. The conclusion is summarized in Section V.

II. PRELIMINARIES AND PROBLEM FORMULATION

A. Notation

In this article, \( \mathbb{R}^n \) represents \( n \)-dimensional Euclidean space, and \( \mathbb{R}^{n \times m} \) refers to the set of \( n \times m \) real matrices. Furthermore, \( I_n \) denotes an \( n \times n \) identity matrix, and \( 1_N \) is an all-one vector of size \( N \). The notation \( \text{diag}(a_1, a_2, \ldots) \) refers to a block diagonal matrix with diagonal elements of \( a_i \). The operation \( \text{vec}(\cdot) \) creates a stacked vector from a matrix in a column-wise fashion. The sum of a square matrix’s diagonal elements is \( \text{tr}(A) \). The Kronecker product is denoted by \( \otimes \), and the transpose of a matrix \( A \) is \( A^T \). The minimum and maximum singular values of a matrix \( A \) are denoted by \( \sigma(A) \) and \( \sigma(A) \), respectively. The 2-norm of a vector or matrix is represented as \( \| \cdot \| \). The Frobenius norm of a matrix is represented as \( \| \cdot \|_F \). The Manhattan norm of a vector \( x \in \mathbb{R}^d \) is \( \| x \|_1 = \sum_{i=1}^{n} |x_i| \). The partial derivative of a function with respect to a variable is represented by a superscript \( \dot{\cdot} \), and the signum function is denoted by \( \text{sgn}(\cdot) \). The notation \( \bigcap_{\mu(H)=0} \) refers to the intersection over all sets \( H \) of the Lebesgue measure zero, and \( \bar{\omega} \) refers to the convex closure.
B. Graph Theory

We model interaction among agents using a directed graph $G = (V, E)$. The graph comprises $N$ vertices, represented by a nonempty set $V = \{v_1, \ldots, v_N\}$, and an edge set $E \subseteq V \times V$ representing connections. The graph is assumed to be simple, with no repeated edges or self-loops. The adjacency or connectivity matrix is $A = [a_{ij}]$, where $a_{ij} > 0$ if $(v_j, v_i) \in E$, and $a_{ij} = 0$ otherwise, with $a_{ii} = 0$.

The definition of a neighboring set $N_i$, and a path can be found in [30]. The in-degree matrix $D = \text{diag}(\text{deg}_i)$ is a diagonal matrix whose elements are the sum of incoming edge weights to vertex $i$, that is, $\text{deg}_i = \sum_{j \in N_i} a_{ij}$. The graph Laplacian $L$ is calculated as $L = D - A$.

The leader adjacency matrix $B = \text{diag}(b_1, \ldots, b_N)$ is a diagonal matrix where $b_i > 0$ if the leader sends information to agent $i$ and $b_i = 0$ otherwise. The directed graph is assumed to have a spanning tree with the leader as the root (node 0), implying that there is a directed path from the leader to every other node.

The definition of an M-matrix can be found in [3] and [31]. We present a lemma similar to [5, Lemma 2] and is utilized later in the proof.

Lemma 1: Let the directed graph $G = (V, E)$ be strongly connected. If the leader is connected to at least one of the agents, then $L + B$ is a nonsingular M-matrix, and the following inequalities hold:

$$
\Pi_1 \triangleq P(L + B) + (L + B)^T P > 0
$$

$$
\Pi_2 \triangleq (L + B)P + P(L + B)^T > 0
$$

where $P = \text{diag}(p_i)$ is a positive-definite matrix and $p = [p_1, \ldots, p_N]^T \triangleq (L + B)^{-1} 1_N$.

Proof: We consider that the graph $G$ is strongly connected. As a result, the matrix $L + B$ is an irreducible, diagonally dominant M-matrix [5, Remark 3], [31, Lemma 4.32], consequently nonsingular. Similarly to the approach that is used in the proof of [31, Th. 4.25], define the vector $p$ as $(L + B)p = 1_N > 0$. Note that $p > 0$ can be expressed as $p = (L + B)^{-1} 1_N$. Define the matrix $P$ as $P = \text{diag}(p_1, \ldots, p_n)$, where $p_i$ are the elements of the vector $p$.

In order to prove that $\Pi_1 > 0$, let us show that $P(L + B)$ is a diagonally dominant matrix. One can define the vector $n = P(L + B)1_N$ and write its elements as $n_i = p_i l_i$, with $l_i = d_i + b_i - \sum_{j=1, j \neq i}^N a_{ij}$. Note that $n_i$ is the scaled element of $i$th row in the matrix $L + B$. Therefore, from Gershgorin’s Theorem, and using the fact that any $i$th row is scaled by a positive value, it is clear that $P(L + B)$ is a diagonally dominant M-matrix and nonsingular. Consequently, one can conclude that its transpose, $(L + B)^T P$, is a positive-definite matrix as well as $\Pi_1$.

To prove that $\Pi_2 > 0$, from the definition of $p$, one can write $(L + B)P1_N = 1_N$. By virtue of the Laplacian matrix $L$, let us define $q = (L + B)P1_N = (D - A + B)P1_N$, then one can write $q_i = (d_i + b_i)p_i - \sum_{k=1, k \neq i}^N a_{ik} p_k$. From $(L + B)p = 1_N$, one has $(d_i + b_i)p_i - \sum_{k=1, k \neq i}^N a_{ik} p_k = 1$. Then, one can show that $(L + B)P$ is a (row) strictly diagonally dominant matrix, that is, for any row, $(d_i + b_i)p_i > \sum_{k=1, k \neq i}^N a_{ik} p_k$. From Gershgorin’s Theorem, it follows that $(L + B)P$ is a positive-definite matrix as well as $P(L + B)^T$. Therefore, one can express $\Pi_2 = (L + B)P + P(L + B)^T > 0$. This completes the proof.

C. Problem Formulation

A multiagent system with $N$ agents is considered, with the dynamics of each agent given by

$$
\dot{p}_i = v_i
$$

$$
\dot{v}_i = f_i(p_i, v_i) + g_i(p_i, v_i)u_i(t) + w_i(t), \quad i = 1, \ldots, N
$$

where the variables $p_i \in \mathbb{R}^n$ and $v_i \in \mathbb{R}^n$ denote the position and velocity states of each agent, respectively. The functions $f_i(p_i, v_i) \in \mathbb{R}^n$ and $g_i(p_i, v_i) \in \mathbb{R}^{n \times n}$ are unknown nonlinear $C^2$ functions. The control input is represented by $u_i \in \mathbb{R}^n$, while a disturbance affecting the agent is denoted by $w_i \in \mathbb{R}^n$.

The dynamics of the multiagent system can be written in a compact form

$$
\dot{x}_1 = x_2
$$

$$
\dot{x}_2 = f(x_1, x_2) + g(x_1, x_2)u + w
$$

where $x_1$ and $x_2$ are stacked vectors of positions and velocities of all agents, respectively. The stacked vector $x_1 = [p_1^T, p_2^T, \ldots, p_N^T]^T \in \mathbb{R}^{nN}$ and $x_2 = [v_1^T, v_2^T, \ldots, v_N^T]^T \in \mathbb{R}^{nN}$. The overall states of the multiagent system are represented by $x = [x_1^T, x_2^T]^T$. The function $f(x) = [f_1^T(p_1, v_1), f_2^T(p_2, v_2), \ldots, f_N^T(p_N, v_N)]^T \in \mathbb{R}^{nN}$ is the stacked nonlinearity of all agents. The control input gain matrix is $g(x) = \text{diag}(g_1(p_1, v_1), g_2(p_2, v_2), \ldots, g_N(p_N, v_N)) \in \mathbb{R}^{nN \times nN}$, the stacked control input vector is $u = [u_1^T, u_2^T, \ldots, u_N^T]^T \in \mathbb{R}^{nN}$, and the stacked disturbance vector is $w = [w_1^T, w_2^T, \ldots, w_N^T]^T \in \mathbb{R}^{nN}$.

The leader’s dynamics are given by

$$
\dot{p}_l = v_l
$$

$$
\dot{v}_l = f_l(p_l, v_l)
$$

where $p_l, v_l \in \mathbb{R}^n$ denote the position and velocity states of the leader, respectively. Here, we use some standard assumptions [17], [18], [22], [28], [29], [32], [33].

Assumption 1: The trajectory of the leader and its first three-time derivatives are bounded, that is, $p_l, v_l, \dot{v}_l, \ddot{v}_l \in L_\infty$. The bounds of $p_l, v_l, \dot{v}_l$, and $\ddot{v}_l$ are considered to be unknown.

Assumption 2: The gain matrix $g_l(p_l, v_l)$ is a symmetric, positive-definite matrix, and its inverse satisfies the following inequality:

$$
\bar{g}||x||^2 \leq x^T \bar{g}^{-1} x \leq \bar{g}||x||^2
$$

where $x \in \mathbb{R}^n$, and unknown constants $\bar{g}$ and $\bar{g}$ are positive.

Remark 1: Assumption 2 states the condition for the controllability of the system (2). These bounds are considered for analytical purposes, and their exact values are considered unknown for control design. We removed an a priori knowledge about $g$ in our control design. Its exact value is not required, and this distinguishes our work from [2], [28], [29], and [34]. Furthermore, it is worth noting that many nonlinear systems satisfying this assumption and represented
by (2) can be found in practical scenarios, including systems described by Euler–Lagrange formulations (see, for example, [17], [18], and [35]).

Assumption 3: The disturbance and the first two time derivatives \((\dot{w}_i, \ddot{w}_i, \dddot{w}_i)\) are considered to be bounded with unknown bounds.

Assumption 1 is necessary as uncertainties or variations in the leader’s trajectory are common in real-world scenarios. However, boundedness ensures that the leader’s trajectory will not diverge uncontrollably. Disturbances are often present in real-world systems and can lead to instability and poor performance. By assuming that the disturbance is bounded in Assumption 3, the controller can be designed to mitigate the effects of the disturbance. The bounds in Assumptions 1 and 3 are unknown and defined only for analytical purposes.

Let us define the error vector for \(i\)th agent as follows [9]:

\[
e_i = \sum_{j \in N_i} a_{ij} (p_j - p_i - d_i + d_j) + b_i (p_j - p_i - d_i)
\]

where the constant vectors \(d_i \in \mathbb{R}^n, i \in \{1, \ldots, N\}\) represent the desired relative position between agent \(i\) and the leader. Note that we define the desired relative position between agents \(i\) and \(j\) as \(d_{ij} = d_i - d_j\). For the agent \(i\), define \(\delta_i\) as

\[
\delta_i = \sum_{j \in N_i} a_{ij} (v_j - v_i) + b_i (v_j - v_i).
\]

The control objective is to design a robust, distributed controller for each agent using local information that ensures all agents achieve the desired formation and maintain the leader’s velocity

\[
\lim_{t \to \infty} ||p_i - p_l - d_i|| = 0
\]

\[
\lim_{t \to \infty} ||v_i - v_l|| = 0.
\]

D. Neural Networks

Based on the universal approximation theorem, for a smooth nonlinear function of \(y(x)\), over a compact set of \(\Omega_x\), there exists a three-layer NN [36], such that

\[
y(x) = W^T \sigma_1 (Z^T \sigma_2 (V^T X)) + \epsilon
\]

where \(V, Z, \text{ and } W\) are ideal NN weights matrices, \(\sigma_1\) and \(\sigma_2\) are the activation functions, and \(\epsilon\) is the NN approximation error.

Assumption 4: The unknown ideal NN weights matrices \(V, Z, \text{ and } W\) are bounded with fixed bounds such that

\[
||V||_F \leq V_m, \quad ||Z||_F \leq Z_m, \quad ||W||_F \leq W_m.
\]

Assumption 5: For the vector \(x \in \mathbb{R}_0^\infty\), the approximation error \(\epsilon\), and its first and second time derivatives are bounded with a fixed bound.

Assumption 4 is essential because it limits the unknown ideal NN weights matrices, \(V, Z, \text{ and } W\), to fixed bounds, \(V_m, Z_m, \text{ and } W_m\), respectively. By limiting the weights matrices, the NN-based controller can be designed with stable performance. Assumption 5 also deals with the approximation error of the NN-based controller. The approximation error can lead to instability and poor performance if it is not bounded. The fixed bounds in Assumption 5 allow for designing a stable and robust controller. These considerations are common and standard assumptions found in NN-based control methods [3], [18], [35]. The bounds defined in Assumptions 4 and 5 are considered unknown and are not used in the controller design; they are only defined for analytical purposes.

In this article, we use a three-layer NN to approximate an unknown nonlinear function over a compact set \(\Omega_x\). The first hidden layer has \(m_1\) neurons, and the second hidden layer has \(m_2\) neurons. Considering \(\hat{y} = [\hat{y}_1, \ldots, \hat{y}_{n_2}]^T \in \mathbb{R}^{n_2}\), the output of NN can be written as follows:

\[
\hat{y} = \hat{W}^T \sigma_1 (\hat{Z}^T \sigma_2 (\hat{V}^T X))
\]

where \(X = [1, X^T]^T \in \mathbb{R}^{n+1}\) with \(X = [x_1, \ldots, x_n]^T \in \mathbb{R}^n\), \(Y\) is the NN weights matrices \(\hat{V} \in \mathbb{R}^{(n_1+1) \times m_1}\), \(\hat{Z} \in \mathbb{R}^{(m_1 \times m_2) \times n_2}\), and \(\hat{W} \in \mathbb{R}^{(m_2 \times n_2)}\). We use \(\sigma_1(\cdot) \in \mathbb{R}^{m_1+1}\) and \(\sigma_2(\cdot) \in \mathbb{R}^{m_2+1}\) to denote the activation functions of hidden layers. We use the sigmoid function for both hidden layers, that is, for a scalar \(x_i\), one has \(\sigma_k(s_i) = 1/(1 + e^{-x_i})\) for \(k \in \{1, 2\}\). The activation function of the output layer is linear. The ideal NN weights matrices \(V, Z, \text{ and } W\) are defined as follows:

\[
V, Z, W = \arg \min_{\hat{V}, \hat{Z}, \hat{W}} \left\{ \sup_{x \in \Omega_x} ||y(x) - \hat{y}(x)|| \right\}.
\]

Let us define \(\hat{y} = y - \hat{y}, \hat{\sigma}_k = \sigma_k - \hat{\sigma}_k, \hat{V} = V - \hat{V}, \hat{Z} = Z - \hat{Z}, \text{ and } \hat{W} = W - \hat{W}\). We define \(\sigma_1(\cdot) = \sigma_1(Z^T \sigma_2), \text{ with } \sigma_2 \neq \sigma_2(V^T X), \text{ and } \sigma_1 \neq \sigma_1(\hat{Z}^T \sigma_2)\). Considering \(\hat{\sigma}_1 = \hat{\sigma}_1(Z^T \sigma_2), \text{ and } \hat{\sigma}_2 = \hat{\sigma}_2(V^T X).\) Moreover, we express \(\hat{\sigma}_1(\cdot) = [d\sigma_1(s_i)/(d\sigma_2)]_{s=\hat{\sigma}_2} \text{ and } \hat{\sigma}_2 = [d\sigma_2(s_i)/(d\sigma_2)]_{s=\hat{\sigma}_2}\). The estimation error satisfies the following:

\[
y - \hat{y} = \hat{W}^T \hat{\sigma}_1 (\hat{Z}^T \hat{\sigma}_2 (\hat{V}^T X)) + \hat{W}^T \hat{\sigma}_1 \hat{Z}^T (\hat{\sigma}_2 - \hat{\sigma}_2(V^T X)) + \hat{W}^T \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 (\hat{V}^T X) + \epsilon.
\]

with

\[
\epsilon = \hat{W}^T \hat{\sigma}_1 \hat{Z}^T O_1(\cdot) + \hat{W}^T \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 (V^T X) + \hat{W}^T \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 \hat{V}^T X + W^T \sigma_1(\hat{Z}^T \sigma_2) + W^T \sigma_1(Z^T \sigma_2) + W^T O_1(\cdot) + \epsilon.
\]

where \(O_1(\cdot) \text{ and } O_2(\cdot)\) are higher-order terms of the Taylor series expansion of \(\sigma_1(Z^T \sigma_2(V^T X))\) at \(Z^T \sigma_2(V^T X)\) and \(\sigma_2(V^T X)\) at \(\hat{Z}^T \sigma_2(V^T X)\), respectively.

Proof: Following the procedure in [12] and [35], from (9) and (11), one can write:

\[
y - \hat{y} = W^T \sigma_1 (Z^T \sigma_2 (V^T X)) - \hat{W}^T \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 (\hat{V}^T X) + \epsilon.
\]

By adding and subtracting \(W^T \sigma_1 + \hat{W}^T \hat{\sigma}_1\), and rearranging (15), one has

\[
y - \hat{y} = \hat{W}^T \hat{\sigma}_1 + W^T \sigma_1 + \hat{W}^T \hat{\sigma}_1 + \epsilon.
\]

Taylor series expansions of \(\sigma_1(Z^T \sigma_2(V^T X))\) at \(Z^T \sigma_2(V^T X)\) and \(\sigma_2(V^T X)\) at \(V^T X\) is

\[
\sigma_1(Z^T \sigma_2(V^T X)) = \sigma_1(\hat{Z}^T \sigma_2(V^T X)) + \hat{\sigma}_1(Z^T \sigma_2(V^T X) - \hat{Z}^T \sigma_2(V^T X)) + O(\cdot)
\]

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
\[
\begin{align*}
\sigma_2(V^T X) &= \hat{\sigma}_2 + \hat{\sigma}_2 V^T X + O_2(\cdot), \\
\end{align*}
\]
\[ (18) \]

Note that \(O_1(\cdot)\) and \(O_2(\cdot)\) are shortened notations for \(O_1(Z^T \sigma_2(V^T X) - \hat{Z}^T \sigma_2(V^T X))\) and \(O_2(V^T X)\), respectively.

From (17) and (18), it is evident that
\[
\hat{\sigma}_1 = \hat{\sigma}_1[Z^T \sigma_2 - \hat{Z}^T \sigma_2] + O_1(\cdot)
\]
\[
\hat{\sigma}_2 = \hat{\sigma}_2 V^T X + O_2(\cdot).
\]
\[ (19) \]

Substituting (19) in (16), one has
\[
y - \hat{y} = \hat{W}^T \left( \hat{\sigma}_1 + \hat{\sigma}_1[Z^T \sigma_2 - \hat{Z}^T \sigma_2] + O_1(\cdot) \right) + \hat{W}^T \hat{\sigma}_1 + \epsilon.
\]
\[ (20) \]

Rearranging (20), yields
\[
y - \hat{y} = \hat{W}^T \left( \hat{\sigma}_1 - \hat{\sigma}_1[Z^T \sigma_2 - \hat{Z}^T \sigma_2] + \hat{\sigma}_1 Z^T \hat{\sigma}_2 + \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 + \hat{\sigma}_1 O_1(\cdot) \right) + \epsilon.
\]
\[ (21) \]

From (19), one has
\[
y - \hat{y} = \hat{W}^T \left( \hat{\sigma}_1 - \hat{\sigma}_1[Z^T \sigma_2 - \hat{Z}^T \sigma_2] + \hat{\sigma}_1 Z^T \hat{\sigma}_2 + \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 + \hat{\sigma}_1 O_1(\cdot) \right) + \epsilon.
\]
\[ (22) \]

Following the same procedure for \(\hat{W}^T \hat{\sigma}_1\), from (19), and by substituting (19), one can write:
\[
y - \hat{y} = \hat{W}^T \left( \hat{\sigma}_1 - \hat{\sigma}_1[Z^T \sigma_2 - \hat{Z}^T \sigma_2] + \hat{\sigma}_1 Z^T \hat{\sigma}_2 + \hat{\sigma}_1 \hat{Z}^T \hat{\sigma}_2 + \hat{\sigma}_1 O_1(\cdot) \right) + \epsilon.
\]
\[ (23) \]

From (18) and (23), one can derive \(\bar{\epsilon}\) as (14).

The following lemma provides a bound for \(\bar{\epsilon}\).

**Lemma 3:** The term \(\bar{\epsilon}\) (14) satisfies the following inequality:
\[
||\bar{\epsilon}|| \leq \Gamma \mu
\]
\[ (24) \]

where \(\Gamma\) is an unknown positive constant, and \(\mu\) is given by
\[
\begin{align*}
\mu &= \|\hat{W}\|_F + \|\hat{W}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{Z}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{Z}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{Z}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{Z}\|_F + 1. \\
\end{align*}
\]
\[ (25) \]

**Proof:** Following the procedure in [35, Lemma 4.3.1], [12], as well as using the fact that \(\sigma_1(\cdot)\) and its derivative are bounded, and from (19), one can show that:
\[
\begin{align*}
||O_2(\cdot)|| &\leq c_0 + c_1 ||\hat{X}|| + c_2 \|\hat{\sigma}_1\|_F \|\hat{X}\|_F \\
||O_1(\cdot)|| &\leq c_3 + c_4 \|\hat{Z}\|_F \\
\end{align*}
\]
\[ (26) \]

where \(c_i, \ i \in \{0, ..., 4\}\) are positive constants. Using Assumption 4, and by substituting (26) in (14), one has
\[
\begin{align*}
\bar{\epsilon} &\leq \|\hat{W}\|_F \|\hat{W}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{Z}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{Z}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{Z}\|_F + 1. \\
\end{align*}
\]
\[ (25) \]

From (28), one has
\[
\begin{align*}
||\bar{\epsilon}|| &\leq \Gamma \left( \|\hat{W}\|_F + \|\hat{W}\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{W}\|_F \|\hat{Z}\|_F \|\hat{\sigma}_1\|_F \|\hat{X}\|_F + \|\hat{Z}\|_F + 1 \right) \\
\end{align*}
\]
\[ (29) \]

with
\[
\Gamma \triangleq \max\{d_0 c_0 + d_3, c_1 + d_1, c_2, d_2, d_4, d_5, d_6 c_3 + d_7 + c_4\}
\]
\[ (30) \]

From (30), it is straightforward to verify that selecting \(\mu\) as (25) implies (24).

**Remark 2:** It is common to consider that the leader’s information (4) is known and use it in the NN input layer [17], [18], [19], [21], [23]. To relax this requirement, we consider that the leader dynamics are unknown, and the NN is not designed using the leader’s information, thus offering more flexibility in the controller design.

**Remark 3:** Lemma 3 provides an upper bound for \(\bar{\epsilon}\) using a three-layer NN. Note that this error is bounded by \(||\bar{\epsilon}|| \leq \Gamma \mu\), where \(\Gamma\) is considered to be an unknown constant. Therefore, one can write it as \(||\bar{\epsilon}|| \leq \bar{\epsilon}_M\), where \(\bar{\epsilon}_M\) is considered unknown. Note that we consider this bound for analytical purposes, and its exact value is not necessary for the controller design.

**Remark 4:** For a sufficiently smooth function over a compact set, the Stone–Weierstrass Theorem states that the function can be approximated using enough number of neurons, and the universal approximation property of NN guarantees the boundedness of the approximation error [3], [17], [36], [37].
III. Controller Design

This section details the controller design of RISE feedback and adaptive NN weights matrices tuning laws.

Let us define 
$$\delta = [\delta_1^T, \ldots, \delta_N^T]^T \in \mathbb{R}^{N \times 1},$$
and 
$$e = [e_1^T, \ldots, e_N^T]^T \in \mathbb{R}^{N \times 1}.$$ Then, one can write (6) and (7) as follows:

$$e = [(L + B) \otimes I_n](x_k - p_t - d)$$
$$\delta = [(L + B) \otimes I_n](x_k - v_t) \quad (31)$$

where 
$$p_t = 1_N \otimes p_t,$$ 
$$d = [d_1^T, \ldots, d_N^T]^T$$ 
and 
$$v_t = 1_N \otimes v_t.$$ Let us define an auxiliary variable of 
$$\zeta = \delta + k_1e$$ as in [3], where 
$$k_1$$ is a positive constant gain. Then, the stacked vector 
$$\xi = [\xi_1^T, \ldots, \xi_N^T]^T \in \mathbb{R}^{N \times 1}$$ is defined as follows:

$$\xi = \delta + k_1e \quad (32)$$

Let us define 
$$f_i = 1_N \otimes f_i,$$ then one can find the time derivative of (32) as follows:

$$\dot{\zeta} = [(L + B) \otimes I_n](f(x) + g(x)u + w - f_i) + k_1 \delta. \quad (33)$$

A. Robust Integral of the Sign of the Error Feedback

Let us define the filtered error as follows:

$$r = \zeta + k_2 \zeta \quad (34)$$

where 
$$k_2$$ is a positive constant. The time derivative of (34) is given by

$$\dot{r} = \zeta + k_2 \dot{\zeta}. \quad (35)$$

From Assumption 2, Remark 1, Lemma 1, and (35), one can express the following equation similar to [19] as:

$$\dot{G}(x)Qr = \dot{G}(x)Q(\dot{\zeta} + k_2 \dot{\zeta}) \quad (36)$$

where 
$$\dot{G}(x) = G^{-1}(x)$$ and 
$$\dot{Q} = [(L + B) \otimes I_n]^{-1}.$$ By adding and subtracting 
$$1/2\dot{G}(x)Qr + k_2 \zeta,$$ and from (31)–(33), (36) can be rewritten as follows:

$$\dot{G}(x)Qr = \dot{G}(x)(\dot{k}_2 + k_1 f_i)(x) + (k_2 + k_1)g(x)u + k_1 k_2(x_2 - v_t)$$
$$+ f_i(x) + g(x)u + g(x)u + w - f_i$$
$$+ \frac{1}{2}\dot{G}(x)Qr + k_2 \zeta - \frac{1}{2}\dot{G}(x)Qr - k_2 \zeta$$
$$= -\frac{1}{2}\dot{G}(x)Qr - k_2 \zeta + \dot{u} + N_1 + N_2 \quad (37)$$

where

$$\dot{G}(x) = \frac{\partial G}{\partial x_2}x_2 + \frac{\partial G}{\partial x_2}(f(x) + g(x)u + w)$$
$$+ \frac{\partial f}{\partial x_1}x_2 + \frac{\partial f}{\partial x_2}(f(x) + g(x)u + w)$$
$$+ f_i = 1_N \otimes \left(\frac{\partial f_i}{\partial v_t}v_t + \frac{\partial f_i}{\partial v_i}f_i\right) \quad (38)$$

Using the fact that 
$$G(x)g(x) = I_N,$$ and from Assumption 2, 
$$N_1$$ and 
$$N_2$$ are given by

$$N_1 = (k_1 + k_2)(G(x)f(x) + u) + G(x)\frac{\partial f}{\partial x_1}x_2$$
$$+ \frac{\partial f}{\partial x_2}(f(x) + g(x)u) + k_2 x_2 + k_2 k_1(x_1 - d)$$
$$+ \frac{1}{2}\left(\frac{\partial G}{\partial x_1}x_2 + \frac{\partial G}{\partial x_2}(f(x) + g(x)u)\right)\left(f(x) - g(x)u\right)$$
$$+ (k_1 + k_2)x_2 + k_1 k_2(x_1 - d) + k_1 k_2 G(x)x_2 \quad (39)$$

Note that functions 
$$N_1 = [N_1^T, \ldots, N_1^N]^T \in \mathbb{R}^{N \times 1},$$
and 
$$N_2 = [N_2^T, \ldots, N_2^N]^T \in \mathbb{R}^{N \times 1},$$ consist of unknown terms. The difference between 
$$\dot{N}_1,$$ and 
$$\dot{N}_2$$ is that, for each agent, we use NN to approximate 
$$\dot{N}_1,$$ and a time-varying robustifying term to compensate for 
$$\dot{N}_2.$$ The term 
$$\dot{N}_1$$ is a function of available states and the agent’s control input. Note that feeding the agent’s control input into the input layer of the NN causes the circular design problem. To avoid this problem, similar to [12] and [18], we created the NN input from the NN weights matrices, states of the system dynamics, the robustifying term, and 
$$\zeta.$$ This distinguishes our results from [19].

Considering (37), the NN-based control law is derived as follows:

$$\dot{u} = -(k_3 + k_4)\zeta - k_2(k_3 + k_4)\zeta - \dot{N}_1 - \kappa(t) \text{sgn}(\zeta) \quad (40)$$

where 
$$k_3$$ and 
$$k_4$$ are positive constants, 
$$\dot{N}_1$$ is the output of NN, and 
$$\kappa(t) = [\text{diag}(\xi_1, \ldots, \xi_N)] \otimes I_n \in \mathbb{R}^{N \times N \times N}$$ is the time-varying gain. The last term in (41) is responsible for compensating the unknown term of 
$$\dot{N}_2$$ in (37) and the NN approximation error. For the agent 
$$i,$$ let us define 
$$N_2^i \triangleq \kappa(t) \text{sgn}(\zeta).$$

The block diagram of the proposed control law is illustrated in Fig. 1.

The discontinuous nature of the sign function allows the controller to switch quickly between control modes, which can help to compensate for sudden changes in the system or disturbances. It also should be noted that the discontinuous nature of the sign function can lead to a chattering effect. We use RISE feedback, coupled with a tunable NN-based controller, to eliminate the chattering effect and improve the controller’s performance.

We choose the following NN input vector for each agent:

$$\tilde{x}_i = \left[p_t^T, v_t^T, \xi_i, \dot{\xi}_i, \dot{V}_i^T, \dot{\tilde{z}}_i^T, \dot{W}_i^T, \kappa(t) \right]^T \in \mathbb{R}^{3n+4} \quad (42)$$

and 
$$X_i = [1, \tilde{X}_i^T]^T \in \mathbb{R}^{3n+5}.$$ Let 
$$N_1 \triangleq y(X_i),$$ then the NN output for agent 
$$i$$ is described as (11), that is, 
$$\tilde{Y}_i \triangleq \tilde{y}(X_i).$$ Let 
$$V = \text{diag}(V_1) \in \mathbb{R}^{(3n+5)N \times 1},$$ 
$$\tilde{Z} = \text{diag}(Z_1) \in \mathbb{R}^{(m+1)N \times 1},$$ 
$$\tilde{W} = \text{diag}(W_1) \in \mathbb{R}^{(m+1)N \times 1},$$ 
$$\dot{\tilde{V}} = \text{diag}(\dot{\tilde{V}}),$$ 
$$\dot{Z} = \text{diag}(\dot{Z}_1),$$ 
$$\dot{W} = \text{diag}(\dot{W}_1),$$ 
$$\tilde{\sigma}_i = [\tilde{\sigma}_i^T, \ldots, \tilde{\sigma}_i^N]^T,$$ 
and 
$$\tilde{\sigma}_i = [\tilde{\sigma}_i^T, \ldots, \tilde{\sigma}_i^N]^T, s \in [1, 2].$$ Let us define 
$$\tilde{X} = [\tilde{X}_1^T, \ldots, \tilde{X}_N^T] \in \mathbb{R}^{3n+5}.$$ Define 
$$\tilde{N}_1 = \tilde{W}^T \sigma_1(\tilde{Z}_1^T \sigma_2(\tilde{V}_1^T \tilde{X}_1))$$ and 
$$\hat{N}_1' = [N_1^T, \ldots, N_1^N]^T,$$ with 
$$\tilde{N}_1 = \hat{N}_1' - \tilde{N}_1,$$ as

$$\tilde{N}_1 = \tilde{W}_i^T \sigma_1(\tilde{\sigma}_i \tilde{\sigma}_i^T \sigma_2(\tilde{V}_i^T \tilde{X}_i))$$
$$+ \tilde{W}_i^T \sigma_1(\tilde{\sigma}_i \tilde{\sigma}_i^T \sigma_2(\tilde{V}_i^T \tilde{X}_i)) \quad (43)$$

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Lemma 2, and Remark 3, one has
\[
\frac{\Omega_1}{\kappa}
\]
Assumptions 1, 3, (33), and (34). Then it follows that
\[
5642 \text{ IEEE TRANSACTIONS ON CYBERNETICS, VOL. 54, NO. 10, OCTOBER 2024}
\]
where
\[
\dot{u}
\]
Remark 5: Let
\[
\hat{\xi}(\tau) = \frac{1}{2} \left( \dot{\xi}(\tau) \right)_{||}^2
\]
Assumption 1), signals \( u \) is bounded. As the
\[
\hat{\xi}(\tau) = \frac{1}{2} \left( \dot{\xi}(\tau) \right)_{||}^2
\]
Remark 6: The unknown term \( \mathcal{N}_2 \) includes disturbance, its
time derivative, the leader's velocity, and its time derivative.
Following the similar procedure as in [19] and [20], consider-
the smooth functions for \( f \) and \( g \) in system dynamics as well as
Assumptions 1 and 3, one has \( ||\mathcal{N}_2|| \leq d_M \) and \( ||\mathcal{N}_2|| \leq d_M \), where \( d_M \) and \( d_M \) are considered to be
unknown.
Take the time derivative of (14), recall Remark 5, and consider
Assumption 5, the boundedness of the activation
function and its derivative are bounded. Using Assumption 4,
for an arbitrary set of $\Omega_4$, the inputs of the NN ($\chi_i(t)$) are bounded and their derivatives ($\dot{\chi}_i$) are bounded. Therefore, we can conclude that there exist upper bounds for $\dot{N}_i$ and $\ddot{N}_i$, namely, $\|\dot{N}_i\| \leq N_M$ and $\|\ddot{N}_i\| \leq N_M$, which are considered to be unknown. This fact also can be found in [17], [19], [20], and [38].

Remark 7: As continuous differentiability is a sufficient condition for being locally Lipschitz [39, Corollary 4.1.1], [40, Th. 8.4] can be written as [20, Lemma 2].

B. Stability Analysis

We prove in this section that the proposed control law (41) ensures stability in the leader-following formation control for a group of heterogeneous, second-order, uncertain, nonlinear multiagents (2). Before presenting the main result of this article, the following lemma is needed.

Lemma 4: Define an auxiliary function $L_i(t)$ as follows:

$$
L_i(t) \triangleq r_i^T(\dot{N}_i + \dot{N}_i(t)) + \dot{\chi}_i^T N_i + W_i^2 + V_i^2 + Z_i^2 \sgn(\chi_i).
$$

Set $\kappa_i = d_i + \bar{e}_i + (1/k_2) \max\{d_i + \bar{e}_i + N_M, \ldots, N_M, \ldots, N_M\}$. Then, the following inequality holds:

$$
M_i \triangleq \kappa_i \|\dot{\chi}_i(t)\|_{1} - \kappa_i \|\dot{\chi}_i(t)\|_{1} (\dot{N}_i + \dot{N}_i(0)) - \int_{0}^{t} L_i(s) ds \geq 0.
$$

Proof: For a discontinuous function $L(t)$, using differential inclusion [41], [42], there exists an absolutely continuous Filippov’s solution with an upper semi-continuous, compact, and convex set-valued map $K[\mathcal{C}]$. From (34) and by integrating (48) [38], one has

$$
\int_{0}^{t} L_i(s) ds \leq \int_{0}^{t} \zeta_i^T(s) (\dot{N}_i(s) + \dot{N}_i(s)) ds - \kappa_i \int_{0}^{t} \zeta_i^T(s) \sgn(\chi_i(s)) ds
$$

Using the Cauchy–Schwarz inequality, and the definition of $\kappa_i$, one has

$$
\begin{align*}
\int_{0}^{t} L_i(s) ds & \leq \kappa_i \|\dot{\chi}_i(t)\|_{1} (\dot{N}_i + \dot{N}_i(0)) - \int_{0}^{t} L_i(s) ds \\
& = \kappa_i \|\dot{\chi}_i(t)\|_{1} (\dot{N}_i + \dot{N}_i(0)) - \kappa_i \|\dot{\chi}_i(t)\|_{1} (\dot{N}_i + \dot{N}_i(0)) - \int_{0}^{t} L_i(s) ds \\
& \leq \kappa_i \|\dot{\chi}_i(t)\|_{1} (\dot{N}_i + \dot{N}_i(0)) - \kappa_i \|\dot{\chi}_i(t)\|_{1} (\dot{N}_i + \dot{N}_i(0)) - \int_{0}^{t} L_i(s) ds
\end{align*}
$$

Therefore, (49) is valid and this completes the proof.

Remark 8: Although a high number of neurons and three layers implies a higher computational burden, we should note that the proposed method reduces the time allotted to tune one-layer NN, for example, RBFNN centroids, the width of activation functions, and the number of neurons. In the NN control literature, it is commonly assumed the existence of a large enough positive integer for ideal neuron numbers for a one-layer NN [5], [26]. There is a tradeoff between the setting of neuron and layer numbers and the computation complexity. Here, we can streamline the controller design process by determining the number of neurons in each layer beforehand. The NN approximation error is then compensated with the robustifying term.

Remark 9: To determine the number of neurons in each layer, we utilize an a priori information based on the number of position/velocity states, denoted by $n$. The numbers of neurons for the first and second hidden layers are set to $m_1 = 3n + 4$ and $m_2 = 2(3n + 4) + 2$, for each agent. From [44, Th. 2.4], one can see that a three-layer NN with $3n + 4$ inputs, $3n + 4$ neurons in the first hidden layer, $2(3n + 4) + 2$ neurons in the second hidden layer, and the sigmoid activation function, can approximate function $\dot{N}_i$ with arbitrary accuracy.

Theorem 1: Let the multiagent system (2) be modeled by a strongly connected directed graph. For each agent, select the number of the hidden layer neurons in (11) as $m_1 = 3n + 4$, $m_2 = 2(3n + 4) + 2$, and the NN input as (42). Under Assumptions 1–5, choose $k_1 > (1/2)$ and

$$
k_2 > \frac{1 + \sigma(P(L + B))}{\sigma(P(L + B))}.
$$

Set the control input as follows:

$$
u_i(t) = -(k_3 + k_4)\chi_i(t) + (k_3 + k_4)\chi_i(0)
$$

where $k_3$ and $k_4$ satisfy the following conditions:

$$
k_3 > \frac{k_2}{2\sigma(P(L + B))}, \; k_4 > \frac{\sigma(P(L + B))}{2}.
$$

Set the time-varying gain $\kappa_i(t)$ as (47), and the NNs weights matrices tuning laws as follows:

$$
\dot{\chi}_i = \gamma_i(k_2 \chi_i^T \dot{\chi}_i + \dot{\chi}_i^T \chi_i - \|\dot{\chi}_i\|_1 \dot{\chi}_i)
$$

where we used the notation $\sgn(\chi_i) = [\sgn(\chi_{i1}), \ldots, \sgn(\chi_{in})]^T$. The definition of the set-valued map $\sgn(\chi_i)$, for $x_i \in R$ is given in [42] and [43].
Then, all the closed-loop signals are bounded, $\zeta \to 0$ as $t \to \infty$, and agents achieve the desired formation and maintain the leader's velocity.

**Proof:** Let us define the Lyapunov function candidate

$$V = \frac{1}{2}e^T e + \frac{1}{4}\xi^T \Pi_1 \xi + \frac{1}{2}r^T G Q r + M + \frac{1}{2}k^T k + \tilde{y}$$

with the semi-positive term $M \triangleq \sum_{i=1}^N M_i$. The variable $M_i$ is defined using Lemma 4, and $\tilde{y}$ is given by

$$\tilde{y} \triangleq \frac{1}{2} \text{Tr} \left( \dot{\tilde{w}}^T \alpha - \tilde{W} \right) + \frac{1}{2} \text{Tr} \left( \tilde{Z} \beta - \tilde{Z} \right) + \frac{1}{2} \text{Tr} \left( \tilde{V} \gamma - \tilde{V} \right)$$

Let us define $k = [k_1, \ldots, k_N]^T \in \mathbb{R}^N$, $\kappa = [k_{d1}, \ldots, k_{dN}]^T \in \mathbb{R}^N$, and $\hat{k} = [\hat{k}_1, \ldots, \hat{k}_N]^T \in \mathbb{R}^N$, where $\hat{k}_i \triangleq k_i - k_{di}$. Moreover, let $\chi = [e^T, \xi^T, \bar{Y}]^T$, $\sqrt{M}$, $\text{vec}(\alpha^{-1} \tilde{W})^T$, $\text{vec}(\beta^{-1} \tilde{Z})^T$, $\text{vec}(\gamma^{-1} \tilde{V})^T$, $\hat{k}^T$. It can be shown that the following inequalities are valid for (56):

$$L_b(\chi) \leq V \leq U_b(\chi)$$

with

$$L_b(\chi) = \eta ||\chi||^2, \quad U_b(\chi) = \tilde{\eta} ||\chi||^2$$

where $\eta \triangleq \min\{1/(1/2)g(Q), (1/2), (1/4)g(\Pi_1)\}$ and $\tilde{\eta} \triangleq \max\{1/(1/2)g(Q), 1, (1/4)g(\Pi_1)\}$. To use [20, Lemma 2], we should study the existence of Filippov's solution for $\dot{\chi} = f(\chi, t)$, as $\dot{M}$ and $\hat{k}$ are differential equations with discontinuous right-hand side [45]. Using differential inclusion [41], [42], an absolutely continuous Filippov's solution exists for $\dot{\chi} \in K[\mathcal{F}](\chi, t)$ where an upper semi-continuous, compact and convex set $K[\mathcal{F}](\chi, t)$ is defined as follows:

$$K[\mathcal{F}](\chi, t) \triangleq \bigcap_{R=0}^\infty \bigcap_{\mu=0} \gamma(\mathcal{F}(B(\chi, R) \setminus H, t))$$

where $B(\chi, R)$ is the closed ball with center $\chi$ and the radius $R$. Using (60) and from [46, Th. 2.2], the time derivative of the Lyapunov function candidate exists almost everywhere

$$\dot{V}(\chi) \in \nabla \mathcal{F}[\mathcal{F}](\chi, t)$$

As the Lyapunov function candidate (56) is continuously smooth, (61) can be written as follows:

$$\dot{V}(\chi) \in \nabla \mathcal{F}[\mathcal{F}](\chi, t) \subset \sum_{\Lambda \in \partial \mathcal{F}(\chi)}$$

Using (57) and (62), the time derivative of (56) is given by

$$\dot{V} = e^T \dot{e} + \frac{1}{2} \xi^T \Pi_1 \dot{\xi} + \frac{1}{2} r^T G Q r + r^T G Q r + M + k^T \dot{k} + \text{Tr} \left( \dot{\tilde{W}}^T \alpha^{-1} \tilde{W} \right) + \text{Tr} \left( \dot{\tilde{Z}}^T \beta^{-1} \tilde{Z} \right) + \text{Tr} \left( \dot{\tilde{V}}^T \gamma^{-1} \tilde{V} \right)$$.  

Substituting (32), (34), (44), and (49) in (63) yields

$$\dot{V} = e^T (\zeta - k_1) + \frac{1}{2} \xi^T \Pi_1 (r - k_2 \zeta) + r^T (-k_2 \zeta - (k_3 + k_4) r - \kappa(t) \text{sgn}(\zeta) + \dot{\tilde{W}}^T (\delta_1 - \delta_1^T \bar{Z} \delta_2 - \delta_1^T \bar{Z} \delta_1 ^T \bar{V} \lambda) + \dot{\tilde{W}}^T \delta_1 (\delta_2 - \delta_1^T \bar{V} \lambda) + \dot{\tilde{W}}^T \delta_1^T \bar{Z} \delta_2^T \bar{V} \lambda) + N_2 + \delta^T + k^T \dot{k} - \sum_{i=1}^N (r_i (N_i - k_{di} \text{sgn}(\zeta)) + \zeta_i \delta_1 N_i + (W_m^2 + V_m^2 + Z_m^2) \zeta_i \text{sgn}(\zeta) + \text{Tr} (\tilde{W} \alpha^{-1} \tilde{W}) + \text{Tr} (\tilde{Z} \beta^{-1} \tilde{Z}) + \text{Tr} (\tilde{V} \gamma^{-1} \tilde{V})$$.

Reorganizing (64), from (34) and (43), one has

$$\dot{V} \leq e^T \zeta - k_1 e + \frac{1}{2} \xi^T \Pi_1 (r - k_2 \zeta) + r^T (-k_2 \zeta - (k_3 + k_4) r + \text{Tr} (\tilde{W} \alpha^{-1} \tilde{W}) + \text{Tr} (\tilde{Z} \beta^{-1} \tilde{Z}) + \text{Tr} (\tilde{V} \gamma^{-1} \tilde{V})$$.

By substituting (46) and (55) in (65), one can get

$$\dot{V} \leq e^T \zeta - k_1 e + \frac{1}{2} \xi^T \Pi_1 (r - k_2 \zeta) + r^T (-k_2 \zeta - (k_3 + k_4) r + \text{Tr} (\tilde{W} \alpha^{-1} \tilde{W}) + \text{Tr} (\tilde{Z} \beta^{-1} \tilde{Z}) + \text{Tr} (\tilde{V} \gamma^{-1} \tilde{V})$$.

Applying Young's inequality for $\text{Tr}(\tilde{W} \alpha^{-1} \tilde{W})$, one has $\text{Tr}(\tilde{W} \alpha^{-1} \tilde{W}) \leq (W_m^2/2) - (||\tilde{W}||^2)/2$. Following the same procedure for $\text{Tr}(\tilde{Z} \beta^{-1} \tilde{Z})$ and $\text{Tr}(\tilde{V} \gamma^{-1} \tilde{V})$, one can write

$$\dot{V} \leq e^T (\zeta - k_1 e) + \frac{1}{2} \xi^T \Pi_1 (r - k_2 \zeta) + r^T (-k_2 \zeta - (k_3 + k_4) r + \text{Tr} (\tilde{W} \alpha^{-1} \tilde{W}) + \text{Tr} (\tilde{Z} \beta^{-1} \tilde{Z}) + \text{Tr} (\tilde{V} \gamma^{-1} \tilde{V})$$.

From (1) and applying the Cauchy–Schwarz inequality and Young’s inequality, the following can be obtained:

$$-k_2 r^T \zeta \leq k_2 \frac{2}{\sigma(P+L+B)||\zeta||^2} + k_2 \frac{||r||^2}{2\sigma(P+L+B)}$$.

Substituting (68) in (67) and from (1) yields

$$\dot{V} \leq - \left( k_1 - \frac{1}{2} \right) ||e||^2 - \left( k_2 \frac{2}{\sigma(P+L+B)} - 1 + \frac{1}{\sigma(P+L+B)} \right)||\zeta||^2$$.
\[-\left( k_3 + k_4 - \frac{\sigma(P(L + B))}{2} - \frac{k_2}{2\sigma(P(L + B))}\right)||r||^2 \right).
\]

From the given conditions for gains $k_1, k_2, k_3,$ and $k_4$ in Theorem 1, one has
\[
\dot{V} \leq -\lambda_1||e||^2 - \lambda_2||\xi||^2 - \lambda_3||r||^2
\]
where $\lambda_1, \lambda_2,$ and $\lambda_3$ are positive constants. Consequently, (70) can be rewritten as follows:
\[
\dot{V} \leq -U(\chi)
\]
where $U(\chi) \triangleq -\lambda||e^T, \xi^T, r^T||^2,$ with $\lambda \triangleq \min(\lambda_1, \lambda_2, \lambda_3).$ Note that in (71), $U(\chi)$ is a positive semi-definite function over $\Omega_\chi.$ From (56) and (71), one has $V \in L_\infty$ over $\Omega_\chi,$ hence, $e, \xi, r, \tilde{W}, \tilde{Z}, \tilde{V},$ and $\bar{k}$ are bounded on set $\Omega_\chi.$ Consequently, from (32) $\delta \in L_\infty.$ From Assumption 1 and (31), one can conclude that $x \in L_\infty.$ From (34), as $r \in L_\infty$ and $\xi \in L_\infty,$ one can obtain $\xi \in L_\infty.$ Using Lemma 1, Assumptions 1–3, and (33), it can be deduced that $u \in L_\infty.$ Based on Assumptions 4, (41), and (44) are also bounded. From the boundedness of $e, \delta, \xi,$ and $r$ and the closed-loop terms, $U(\chi)$ is uniformly continuous, where [20, Lemma 2] can be applied. Let $\Omega_\chi = \{\chi \mid ||\chi|| < B(0, R)\},$ and the initial conditions belong to the compact set $S \subset \Omega_\chi$ as follows:
\[
S \triangleq \left\{ \chi \in \Omega_\chi \mid U_b(\chi) \leq \eta R^2 \right\}
\]
where the origin is within the set of $S.$ Then, the lemma [20, Lemma 2] is used to conclude $||e||^2 \to 0, ||\xi||^2 \to 0, ||r||^2 \to 0,$ as $t \to \infty$ $\forall \chi(0) \in S.$ From (31) and (32), one can conclude that as $||e|| \to 0,$ and $||\delta|| \to 0,$ (8) holds. Consequently, by increasing $R,$ the semi-global asymptotic stability is obtained [20]. This completes the proof.

Remark 10: Note that, we have considered the underlying graph $G(V, E)$ to be a strongly connected directed graph and that the leader is connected to at least one agent. These considerations imply the existence of a spanning tree with the leader as its root (see [26, Remark 1]). Therefore, without loss of generality, one can prove that Theorem 1 is valid when there exists a spanning tree with the leader as its root.

Remark 11: The results of Theorem 1 can be used to determine the hyperparameters for the NN. Specifically, one can set the number of hidden-layer neurons in (11) as $m_1 = 3n + 4$ and $m_2 = 2(3n + 4) + 2,$ where $n$ is the number of position states. The NN input can be defined using (42), and the constants $\alpha_i, \beta_i,$ and $\gamma_i$ can be set to sufficiently small values. To select the control parameters, one can choose $k_1 > (1/2)$ and $k_2$ as defined in (52). Additionally, $k_3$ and $k_4$ can be set using (54). Note that the conditions outlined in (52) and (54) present the primary parameter choices. By opting for higher values of $k_1, k_2,$ and $k_3,$ one can achieve not only an elevated level of system responsiveness but also a quicker settling time. Nevertheless, adjusting these tuning parameters will directly impact energy consumption.

IV. SIMULATION RESULTS

To illustrate the performance of our proposed method, we studied a multiagent system that consists of five two-link robot arms. The directed graph that models the system is shown in Fig. 2. The dynamics of each agent are given in [2] and [34].

Let $p_1 = [p_{1i1}, p_{1i2}]^T \in \mathbb{R}^2$ and $p_2 = [p_{2i1}, p_{2i2}]^T \in \mathbb{R}^2$ be the joint position and velocity states of $i$th robot arm, respectively. Consider that the control input is $\tau_i \triangleq u_i.$ Each robot arm has the following parameters: $g = 9.8 \text{ m/s}^2, r_1 = 1 \text{ m}, r_2 = 1 \text{ m}, m_{i1} = 0.8 \text{ kg},$ and $m_{i2} = 1.7 \text{ kg},$ $w_i = [-0.12\cos(t), 0.1\sin(t)]^T, i \in [1, \ldots, 5].$ The leader dynamics are given by
\[
\dot{p}_l = v_l
\]
\[
\dot{v}_l = \begin{bmatrix}
-p_{l1} + 0.2(1 - p_{l1}^2)v_l \\
-p_{l2} + 0.3(1 - p_{l2}^2)v_l
\end{bmatrix}
\]
where $p_1 = [p_{1i1}, p_{1i2}]^T \in \mathbb{R}^2$ and $v_l = [v_{l1}, v_{l2}]^T \in \mathbb{R}^2$ are the position and velocity states of the leader. The initial conditions for the five robots are: 1) $p_{1i1}(0) = [2.1, 0]^T; 2) p_{2i1}(0) = [2.5, 0]^T; 3) p_{3i1}(0) = [-1.1, 2]^T; 4) p_{4i1}(0) = [-1.8, 0.7]^T; and 5) p_{5i1}(0) = [-1.7, 0]^T,$ and zero initial velocities for all agents. The desired displacement with respect to the leader for each agent is given as $d_1 = [0, 0]^T, d_2 = [-\sin(2\pi/5), d\cos(2\pi/5)]^T, d_3 = [-\sin(\pi/5), -d\cos(\pi/5)]^T, d_4 = [\sin(\pi/5), -d\cos(\pi/5)]^T,$ and $d_5 = [\sin(2\pi/5), d\cos(2\pi/5)]^T$ with $d = 1.$ The leader initial states are $p_l(0) = [1, -1]^T$ and $v_l(0) = [0, 0]^T,$ respectively. All weights of edges in Fig. 2 are equal to one as well as the leader’s edge weight ($b_1 = 1$).

Following Remark 9 and considering the system dimension $n = 2,$ we adopted the NN architecture for each agent consisting of 10 input neurons, 10 neurons in the first hidden layer, 22 neurons in the second hidden layer, and 2 output neurons. The activation function for these layers is a sigmoid function. The activation function for the output layer is linear. The constants $\alpha_i, \beta_i, \gamma_i$ are chosen as $\alpha_i = (1/20), \beta_i = (1/20), \gamma_i = (1/20).$ The gains were selected as $k_1 = 4, k_2 = 37.5, k_3 = 370,$ and $k_4 = 12.$

Figs. 3–7 present the simulation results for the multiagent system. Fig. 3 shows the trajectories of both the agents and the leader, while their velocities are depicted in Fig. 4, where the agents’ velocities converge to the leader’s velocity. The Frobenius norm of the NN weights matrices, $\tilde{V}_i, \tilde{Z}_i,$ and $\tilde{W}_i,$ are illustrated in Fig. 5. The time-varying gains $\kappa_i$ are shown in Fig. 6, where a dead-zone of size $b = 0.005$ has been implemented to prevent $\kappa_i(t)$ from reaching high values, as done in [19] and [38]. The error signals $e_i(t),$ their time
Fig. 3. Position state of the leader (solid line) and the agents. The configuration achieved the desired position and followed the leader.

Fig. 4. Velocity state of the leader (solid line) and agents. The agents maintained the leader’s velocity.

Fig. 5. Frobenius norm of NN weights matrices for agents.

derivatives $\delta_i(t)$, and the filtered errors $\zeta_i(t)$ are depicted in Fig. 7. These results verify the performance of the proposed method based on RISE feedback and the NN-based controller.

We conducted a comparative analysis with two previously established results on leader-following consensus control of the nonlinear system with unknown nonlinearities using NNs, that is, RBFNN [2] and another one-layer NN method [3]. We determined the stability of the proposed method and compared it with UUB stability results of the other two methods. The results of our analysis demonstrated the effectiveness and superiority of the proposed method in terms of stability, efficiency, and performance. We included the desired displacements and applied these methods similar to [30, p. 127]. We define the average of the control inputs cost function (ACI), $v(t)$ and the average of the formation errors cost function (AFE), $\vartheta(t)$, as follows:

$$v(t) = \frac{1}{2N} \sum_{i=1}^{N} u_i^T(t)u_i(t)$$

$$\vartheta(t) = \frac{1}{2N} \sum_{i=1}^{N} ||e_i(t)||_1. \tag{74}$$

These two functions are used as performance indices to evaluate the effectiveness of the proposed methods in comparison with [2] and [3]. Fig. 8(a) shows the semi-logarithmic graph of the ACI function. Considering energy consumption, the proposed method (solid line) is similar to the other two methods. As shown in Fig. 8(b), our proposed method achieves faster settling and converges to zero, unlike the other two methods where the error remained bounded.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
V. CONCLUSION

This article developed the leader-following formation control of the heterogeneous, second-order, uncertain, input-affine, nonlinear multiagent systems modeled by a directed graph. The unknown nonlinearity in the dynamics of a multiagent system was approximated by a tunable, three-layer NN consisting of an input layer, two hidden layers, and an output layer. The proposed method can a priori set the number of neurons in each layer of the NN. The NN weights tuning laws were derived using the Lyapunov theory. A robust integral of the sign of the error feedback control with an NN was developed that guarantees semi-global asymptotic leader tracking. The boundedness of the closed-loop signals and asymptotic leader tracking formation were proven using the Lyapunov stability theory. The article results were compared with two previous results, which showed the effectiveness of the proposed method.

REFERENCES

[1] K.-K. Oh, M.-C. Park, and H.-S. Ahn, “A survey of multiagent formation control,” Automatica, vol. 53, pp. 424–440, Mar. 2015.
[2] G.-X. Wen, C. P. Chen, Y.-J. Liu, and Z. Liu, “Neural-network-based adaptive leader-following consensus control for second-order non-linear multi-agent systems,” IET Control Theory Appl., vol. 9, no. 13, pp. 1927–1934, 2015.
[3] A. Das and F. L. Lewis, “Cooperative adaptive control for synchronization of second-order systems with unknown nonlinearities,” Int. J. Robust Nonlin. Control, vol. 21, no. 13, pp. 1509–1524, 2011.
[4] R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” IEEE Trans. Autom. Control, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
[5] A. Das and F. L. Lewis, “Distributed adaptive control for synchronization of unknown nonlinear networked systems,” Automatica, vol. 46, no. 12, pp. 2014–2021, Dec. 2010.
[6] H. Zhang, D. Yue, X. Yin, S. Hu, and C. Xia Dou, “Finite-time distributed event-triggered consensus control for multi-agent systems,” Inf. Sci., vol. 339, pp. 132–142, Apr. 2016.
[7] W. Ren, “Multi-vehicle consensus with a time-varying reference state,” Syst. Control Lett., vol. 56, nos. 7–8, pp. 474–483, 2007.
[8] G. Chen and Y.-D. Song, “Cooperative tracking control of nonlinear multiagent systems using self-structuring neural networks,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 8, pp. 1496–1507, Aug. 2014.
[9] Q. Shi, T. Li, J. Li, C. P. Chen, Y. Xiao, and Q. Shan, “Adaptive leader-following formation control with collision avoidance for a class of second-order nonlinear multi-agent systems,” Neurocomputing, vol. 350, pp. 282–290, Jul. 2019.
[10] Y.-Y. Chen, R. Huang, Y. Ge, and Y. Zhang, “Spherical formation tracking control of nonlinear second-order agents with adaptive neural flow estimate,” IEEE Trans. Neural Netw. Learn. Syst., vol. 33, no. 10, pp. 5716–5727, Oct. 2022.
[11] J. L. Castro, “Fuzzy logic controllers are universal approximators,” IEEE Trans. Syst., Man, Cybern., vol. 25, no. 4, pp. 629–635, Apr. 1995.
[12] J.-H. Park, S.-H. Huh, S.-I. Kim, S.-I. Soo, and G.-T. Park, “Direct adaptive controller for nonaffine nonlinear systems using self-structuring neural networks,” IEEE Trans. Neural Netw., vol. 16, no. 2, pp. 414–422, Mar. 2005.
[13] X. Wang, H. Wang, T. Huang, and J. Kurths, “Neural-network-based adaptive tracking control for nonlinear multiagent systems: The observer case,” IEEE Trans. Cybern., vol. 53, no. 1, pp. 158–160, Jan. 2023.
[14] K. Lu, Z. Liu, G. Lai, C. L. P. Chen, and Y. Zhang, “Adaptive consensus tracking control of uncertain nonlinear multiagent systems with predefined accuracy,” IEEE Trans. Cybern., vol. 51, no. 1, pp. 405–415, Jan. 2021.
[15] C. Wang and D. J. Hill, “Learning from neural control,” IEEE Trans. Neural Netw., vol. 17, no. 1, pp. 130–146, Jan. 2006.
[16] J. Moody and C. Darken, “Learning with localized receptive fields,” Dept. Comput. Sci., New Haven, CT, USA, Rep. YALEU/DCS/RR-649, 1988.
[17] P. M. Patre, W. MacKunis, K. Kaiser, and W. E. Dixon, “Asymptotic tracking for uncertain dynamic systems via a multilayer neural network feedforward and RISE feedback control structure,” IEEE Trans. Autom. Control, vol. 53, no. 9, pp. 2180–2185, Oct. 2008.
[18] J. Shin, S. Kim, and A. Tsourdos, “Neural-networks-based adaptive control for an uncertain nonlinear system with asymptotic stability,” Int. J. Control, Autom. Syst., vol. 16, no. 4, pp. 1989–2001, Apr. 2018.
[19] Q. Yang, S. Jagannathan, and Y. Sun, “Robust integral of neural network and error sign control of MIMO nonlinear systems,” IEEE Trans. Neural Netw. Learn. Syst., vol. 26, no. 12, pp. 3278–3286, Dec. 2015.
[20] B. Xian, D. M. Dawson, M. S. De Queiroz, and J. Chen, “A continuous asymptotic tracking controller for uncertain nonlinear systems,” IEEE Trans. Autom. Control, vol. 49, no. 7, pp. 1200–1211, Jul. 2004.
[21] T. Dierks and S. Jagannathan, “Neural network control of mobile robot formations using RISE feedback,” IEEE Trans. Syst., Man, Cybern., Part B, Cybern., vol. 39, no. 2, pp. 332–347, Apr. 2009.
[22] N. Fischer, Z. Kan, R. Kamalapurkar, and W. E. Dixon, “Saturated RISE feedback control for a class of second-order nonlinear systems,” IEEE Trans. Autom. Control, vol. 59, no. 4, pp. 1094–1099, Apr. 2014.
[23] Z. Yao, J. Yao, and W. Sun, “Adaptive RISE control of hydraulic systems with multilayer neural-networks,” IEEE Trans. Ind. Electron., vol. 66, no. 11, pp. 8638–8647, Nov. 2019.
[24] B. Yao and M. Tomizuka, “Adaptive robust control of SISO nonlinear systems in a semi-strict feedback form,” Automatica, vol. 33, no. 5, pp. 893–900, May 1997.
[25] K. Aryanian and R. R. Selmic, “Formation control for a class of nonlinear multi-agent systems using three-layer neural networks,” in Proc. Amer. Control Conf. (ACC), 2023, pp. 1667–1672.
[26] H. Zhang and F. L. Lewis, “Adaptive cooperative tracking control of higher-order nonlinear systems with unknown dynamics,” Automatica, vol. 48, no. 7, pp. 1432–1439, Jul. 2012.
[27] C. P. Chen, G.-X. Wen, Y.-J. Liu, and F.-Y. Wang, “Adaptive consensus control for a class of nonlinear multiagent time-delay systems using neural networks,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 6, pp. 1217–1226, Jun. 2014.
[28] G. Wen, C. P. Chen, Y.-J. Liu, and Z. Liu, “Neural network-based adaptive leader-following consensus control for a class of nonlinear multi-agent state-delay systems,” IEEE Trans. Cybern., vol. 47, no. 8, pp. 2151–2160, Aug. 2017.
[29] S. Ge, F. Hong, and T. H. Lee, “Adaptive neural control of nonlinear time-delay systems with unknown virtual control coefficients,” IEEE Trans. Syst., Man, Cybern., Part B, Cybern., vol. 34, no. 1, pp. 499–516, Feb. 2004.
[30] M. Mesbahi and M. Egerstedt, Graph Theoretic Methods in Multiagent Systems, vol. 33. Princeton, NJ, USA: Princeton Univ. Press, 2010.
[31] Z. Qu, Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles. London, U.K.: Springer, 2009.
[32] S. S. Ge, F. Hong, and T. H. Lee, “Adaptive neural network control of nonlinear systems with unknown time delays,” IEEE Trans. Autom. Control, vol. 48, no. 11, pp. 2004–2010, Nov. 2003.
[33] M. Wang, B. Chen, and P. Shi, “Adaptive neural control for a class of perturbed strict-feedback nonlinear time-delay systems,” IEEE Trans. Syst., Man, Cybern., Part B, Cybern., vol. 38, no. 3, pp. 721–730, Jun. 2008.
[34] K. Aryankia and R. R. Selmic, “Neural network-based formation control with target tracking for second-order nonlinear multiagent systems,” IEEE Trans. Aerosp. Electron. Syst., vol. 58, no. 1, pp. 328–341, Feb. 2022.
[35] F. Lewis, S. Jagannathan, and A. Yesildirak, Neural Network Control of Robot Manipulators and Non-linear Systems. Boca Raton, FL, USA: CRC Press, 1998.
[36] A. F. Filippov, Differential Equations With Discontinuous Righthand Sides: Control Systems, vol. 18. Dordrecht, The Netherlands: Springer, 2013.
[37] M. H. Stone, “The generalized weierstrass approximation theorem,” Math. Mag., vol. 21, no. 5, pp. 237–254, 1948.
[38] B. Fan, Q. Yang, S. Jagannathan, and Y. Sun, “Asymptotic tracking controller design for nonlinear systems with guaranteed performance,” IEEE Trans. Cybern., vol. 48, no. 7, pp. 2001–2011, Jul. 2018.
[39] J. Cortes, “Discontinuous dynamical systems,” IEEE Control Syst. Mag., vol. 39, no. 9, pp. 1910–1914, Sep. 1994.

Kiarash Aryankia received the Ph.D. degree in electrical and computer engineering from Concordia University, Montreal, QC, Canada, in 2023.

His research interests include the formation control of multi-agent systems, neural network control, and nonlinear control.

Rastko Selmic (Senior Member, IEEE) received the B.S. degree in electrical engineering from the University of Belgrade, Belgrade, Serbia, in 1994, and the M.S. and Ph.D. degrees in electrical engineering from the University of Texas at Arlington, Arlington, TX, USA, in 1997 and 2000, respectively.

He is a Professor with the Electrical and Computer Engineering Department, Concordia University in Montreal, QC, Canada, where he served as an Associate Chair of Graduate Studies and a Graduate Program Director. Until 2017, he was an AT&T Professor of Electrical Engineering with Louisiana Tech University, Ruston, LA, USA. He was a Research Fellow with the U.S. Air Force Research Laboratory. From 1997 to 2002, he was a Lead DSP Systems Engineer with Signalogic Inc., Dallas, TX, USA. He has authored/coauthored a U.S. patent with four additional reports of invention, 140 journal, and conference papers, four book chapters, and books Wireless Sensor Networks: Security, Coverage, and Localization (Springer, 2016), and Neuro-Fuzzy Control of Industrial Systems with Actuator Nonlinearities (SIAM Press, Philadelphia, PA, USA, 2002). His current research interests include the formation control of UAVs, smart sensors and actuators, cooperative sensing and control, and gesture-based control.

Dr. Selmic received the 2023 Teaching Excellence Award from Gina Cody School of Engineering and Computer Science, Concordia University, the 2009 IFM Award for Outstanding Publication, the ARRI Invention Award in 2000, the first prize at the IEEE Fort Worth Section Graduate Paper Contest in 1999, was a finalist for the Best Paper Award at IEEE International Conference on Control Applications in 1998, and the ARRI Best Paper Award in 1997. He served as an Associate Editor for IEEE TRANSACTIONS ON NEURAL NETWORKS and currently serves as an Associate Editor for IEEE TRANSACTIONS ON CYBERNETICS and Frontiers in Control Engineering. He is a Professional Engineer of Ontario.