A Distributed False Data Injection Cyber-Attack Detection in Discrete-Time Nonlinear Multi-Agent Systems Using Neural Networks

Amirreza Mousavi

A Thesis
in
the Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Master of Applied Science (Electrical and Computer Engineering) at
Concordia University
Montréal, Québec, Canada

December 2021

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: Seyed Amirreza Mousavi

Entitled: A Distributed False Data Injection Cyber-Attack Detection in Discrete-Time Nonlinear Multi-Agent Systems Using Neural Networks

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Electrical and Computer Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

______________________________ Chair
Dr. A.G. Aghdam

______________________________ Examiner
Dr. M. Ghafouri (CIISE)

______________________________ Examiner
Dr. A.G. Aghdam

______________________________ Thesis Supervisor(s)
Dr. R. Selmic

______________________________ Thesis Supervisor(s)

Approved by ______________________________
Dr. Yousef Shayan, Chair
Department of Electrical and Computer Engineering

____
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Date ______________________________

Abstract

A Distributed False Data Injection Cyber-Attack Detection in Discrete-Time Nonlinear Multi-Agent Systems Using Neural Networks

Amirreza Mousavi

In this thesis we study a detection method for the false data injection (FDI) attack class on discrete-time nonlinear multi-agent systems. The thesis considers and models three types of FDI attacks on multi-agent communication channels, including sensor channel, actuator channel, and neighbouring channel. To estimate the dynamics nonlinearity of each agent, we exploit a radial basis function neural network (RBFNN). We consider a leader-follower multi-agent system, where the communication between agents is modeled with an undirected graph. We proposed the weight tuning law for the RBFNN and introduced an NN-based distributed control law for each agent. The objective of each agent is to follow the leader and maintain the desired formation along the trajectory. We used the Lyapunov stability analysis to prove the uniform ultimate boundedness (UUB) of the formation error and neural network (NN) weights matrix and show that the multi-agent system reaches the desired w while following the leader.

Moreover, we proposed a distributed attack detection method to detect the FDI attack on each agent's sensor, actuator, and neighbouring communication channel. We designed an observer to estimate the state of each agent and used its estimation to form the residual signal for each agent. Using Lyapunov stability analysis, we show that when the system is reached its desired formation, the residual signal is UUB. We obtained abound for the residual signal and considered the bound as the attack detection threshold. We also provided the attack detectability condition for each agent.

The simulation results in MATLAB and Coppeliasim simulation environment are provided to demonstrate the performance of the detection methodology, proposed distributed control law, and neural network nonlinearity estimator, including three examples.

Acknowledgments

I would like to thank my supervisor Prof. Rastko R. Selmic, for his dedicated support and guidance. Prof. Selmic continuously provided encouragement and was always willing and enthusiastic to assist in any way he could throughout the research project. I cannot forget to thank my family and my wife Shima Savehshemshaki for all the unconditional support in these very intense academic years and send me endless messages of encouragement throughout the process. I would also like to thank my great friend Mr. Kiarash Aryankia for always being there for me and support me in any circumstances. Finally, many thanks to all participants that took part in the study and enabled this research to be possible.

Contents

List of Figures

List of Tables

List of Abbreviations

CPS = Cyber-Physical System

FDI = False Data Injection

NN = Neural Network

UUB = Uniformly Ultimate Boundedness

RBFNN = Radial Basis Function Neural Network

UGV = Unmanned Ground Vehicle

Chapter 1

Introduction

1.1 Literature Review

Cyber-Physical Systems (CPSs) are the integration of computation units and communication networks with physical processes [1]. In recent years, much attention has been devoted to studying CPSs due to their modern engineering applications such as traffic networks [2], [3], power systems [4], [5], Internet of things, and multi-agent systems [6]. Most of the aforementioned systems are connected to the Internet and wireless communication networks through communication channels that attackers can penetrate and change the transmitted data.

Various cyber-attack detection methods have been proposed in the literature of CPSs. In [7], a sensor coding mechanism is used to detect stealthy data injection attacks, which is designed by an intelligent attacker with a system model knowledge. In [8], a strategy is proposed to estimate and compensate attacks in the forward link of a nonlinear CPS. The proposed method is using nonlinear control theory with applied neural networks to develop cyber-attack observers for multi-agents. In [9], an adaptive framework is developed for the control design of cyber-physical systems in the presence of simultaneous adversarial sensor and actuator attacks. Authors in [10] investigated the attack detection problem of setpoint attacks on CPS and proposed a control architecture exploiting a command governor to detect the setpoint attack on networked control systems. Most of the works in cyber-attack detection are devoted to single-agent systems, while multi-agent systems can be a potential target for attackers because they possess too many communication channels and have extensive use in critical infrastructure.

With the development of computation and communication technologies, multi-agent systems have received significant attention from researchers due to their various applications in large-scale critical infrastructures such as smart grids systems, water distribution networks, telecommunication networks, and transportation systems [2–6]. Multi-agent systems are connected through communication channels that increase their vulnerability to external cyber-attacks. Security concerns related to these systems include physical security and cyber security, and combined cyber-physical threats. Various types of cyber-attacks have been reported in recent years [11–13], , which can deteriorate physical systems' performance and ultimately lead to failures or unsafe behavior. As a result, significant attention has been devoted to study the security of multi-agent systems. In multi-agent systems, we consider that each agent has three types of communication channels: (i) actuator channel, which transfers the control signal from the controller to the plant; (ii) sensor channel, which transfers the system output (the sensor measurements) from the agent plant to the controller; and (iii) neighboring communication channels through which each agent receives neighbors' data. These vulnerable communication channels are prone to cyber-physical attacks [14, 15].

Cyber-attack detection methods have been proposed in the literature for linear multi-agent systems [15–17]. In [15], the problem of cyber-attacks detection on the communication network of linear interconnected systems and multi-agent systems governed by a consensus-based control was investigated, and a distributed residual-based attack detection method was proposed for the attack detection on the neighboring communication channels. Authors in [16], proposed a data-driven switching controller to obtain the resilient control for the discrete-time linear multi-agent network under unconfined cyber-attacks. In [17], the attack detection problem for the interconnected stochastic systems was studied, where centralized and decentralized detection strategies for detecting attacks are developed, and the detection performances are characterized. In [18], authors presented a distributed False Data Injection (FDI) attack detection strategy for networked robots controlled by a distributed observer–controller scheme. A vector of residuals was introduced based on the designed observer, where fault detectability and isolability conditions were derived.

Results in [19] offer a distributed fault detection filtering approach for a class of continuous-time, nonlinear, multi-agent systems with uncertainties, measurement noise, and disturbances. In [20], a distributed event-triggered consensus problem for continuous-time, nonlinear, multi-agent systems with general directed communication topology under cyber-attacks has been addressed. Some distributed methods for attack detection in multi-agent systems have been recently proposed [21–26]. Most of the works consider linear or continuous models for multi-agent systems. Moreover, some methods assume that each agent is aware of the entire topology of the multi-agent system (centralized approach), [14, 18]. In this thesis, we consider a discrete-time nonlinear multi-agent system controlled by a displacement-based controller where each agent has access to its neighbors' information.

While some results in the literature assume that the neighboring communication channels are safe and investigate the cyber-attack detection problem on the actuator and sensor channels [21, 27, 28], other works consider that all the agents are safe, and the attacker can breach and compromise only the neighboring communication channels [15, 29, 30]. In this thesis, we assume that sensors, actuators, and neighboring communication channels can be simultaneously attacked, which is different from existing results in the literature.

1.2 Thesis Motivation and Contributions

In this thesis, we propose a cyber-attack detection method that relies on locally available information and communicated data by neighboring agents. We present a cyber-attack detection method in discrete-time, nonlinear multi-agent systems with an unknown nonlinearity in system dynamics using a neural network (NN)-based observer. Through Lyapunov stability analysis, a threshold is obtained for the proposed residual-based detector to detect agents' communication channels' attacks. We also developed a displacement-based method to maintain the desired formation. The Lyapunov stability theory is used to prove the uniform ultimate boundedness (UUB) of the formation error and to provide a bound for the formation error.

In multi-agent systems, consensus and formation control are two essential problems of interest. In consensus control, agents interact locally in order to reach a common value of a certain state [31]. The formation is defined as a configuration in a space where each agent is at the desired distance or angle from its neighbors [32]. In this thesis, the attack detection problem is addressed for the formation control of a class of discrete-time, nonlinear multi-agent systems. In the displacement-based formation control of multi-agent systems, it is necessary for each agent to communicate with its neighboring agents to achieve the formation objective [33]. However, communication channels are vulnerable to cyber-attacks. By changing the communication channels data, agents can receive the corrupted data, which can disturb the formation and cause collisions. Therefore, the security of the communication channels data is of paramount importance.

The attack model studied in this thesis is a false data injection attack on the actuator channel, sensor channel, and neighboring communication channels. Due to unknown nonlinearity in the system dynamics, the attacker is unable to know the agents' system dynamics. As a result, the covert attack studied in [23, 34] cannot be applied here.

Compared with the literature on cyber-attack detection in multi-agent systems, the main contributions of this thesis can be summarized as follows:

(i) An NN observer-based attack detection scheme for uncertain, discrete-time, nonlinear multi-agent systems is developed, and UUB of detection residual of each agent is proven.

(ii) The attack detectability condition is derived when three different types of communication channels are under FDI attacks. A distributed, NN-based observer and the controller for the formation control of discrete-time, nonlinear multi-agent systems are established.

(iii) The NN-based controller is used to compensate for unknown nonlinearities in the dynamics of discrete-time multi-agent systems, where UUB of formation error and NN weights estimation errors are rigorously proven.

(iv) It is demonstrated that the proposed system can detect FDI attacks on communication channels of multi-agent systems by implementing the proposed method in MATLAB and CoppeliaSim robot simulation environment.

1.3 Publication

- **A. Mousavi, K. Aryankia, and R. R. Selmic, "Cyber-attack detection in discrete-time nonlinear multi-agent systems using neural networks," In 2021 IEEE Conference on Control Technology and Applications (CCTA). IEEE, 2021.** [35]

  we presented an observer-based attack detection method to detect FDI attacks in neighboring communication channels of discrete-time, nonlinear multi-agent systems when the leader-follower graph topology is strongly connected.

- **R. R. Selmic, A. Mousavi, and K. Aryankia, "A Distributed FDI Cyber-Attack Detection in Discrete-Time Nonlinear Multi-Agent Systems Using Neural Networks" In European Journal of Control *(Under review).*** [36]

  In this paper, we study a more general attack detection problem for a class of discrete-time, nonlinear multi-agent systems when their sensor, actuator, and neighboring communication channels are simultaneously compromised by cyber-attacks. The leader-follower graph topology is assumed to contain a spanning tree with the leader as its root. We also provide a rigorous proof that the residual signal and formation error are UUB.

1.4 Thesis Overview

This thesis is organized as follows:

- In Chapter 2, the main concepts and definitions used along the thesis are defined and presented. Moreover, the basis of formation control and neural network are summarized and reviewed.
- In Chapter 3 an NN is developed to approximate the unknown nonlinearity dynamics of each agent, and a distributed displacement-based controller is proposed to maintain the desired formation. The Lyapunov stability analysis is used to prove the UUB of formation error.
- In Chapter 4, a distributed observer is proposed for the attack detection purpose, and the Lypunoc analysis is used to prove the UUB of attack detection residual. Moreover, the attack detectability condition is given.
- In Chapter 5, the performance of the proposed cyber-attack detection method is demonstrated through the simulation results in MATLAB and CoppeliaSim.
- In Chapter 6, the conclusion of this thesis, by summarizing the proposed results, is provided, and the future research directions are outlined.

Chapter 2

Preliminaries and Definitions

2.1 Graph Theory

In multi-agent systems, interaction among the agents is usually modeled by graphs. This thesis considers the weighted undirected graph to model interactions and communication among agents [37]. Let an undirected graph $\mathcal{G}$ be given by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the set of vertices is $\mathcal{V} = \{v_1, v_2, ..., v_N\}$, and the set of edges is $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. An edge in $\mathcal{G}$ is denoted by a pair (v_i, v_j). In the multi-agent system, the node v_i denotes the i-th agent, and an unordered pair of $(v_i, v_j) \in \mathcal{E}$, if the agent i and agent j can communicate with each other. Nodes i and j is are neighbors if $(v_j, v_i) \in \mathcal{E}$.

Definition 1. *The neighbor set of i-th agent, $\mathcal{N}_i = \{j | (v_j, v_i) \in \mathcal{E}\}$ is the set of all adjacent agents to the agent i.*

Non-negative symmetric matrix $\mathcal{A} = [a_{ij}]$ is adjacency matrix, where $a_{ij} > 0$ whenever there is an edge between and between node i and node j, otherwise $a_{ij} = 0$. Clearly, for undirected graphs $a_{ij} = a_{ji}$. Throughout this thesis, it is assumed that there are no self-loops, i.e., $a_{ii} = 0$, and the graph topology is fixed. The element a_{ij} is the weight between nodes υ_i and υ_j. The weighted degree of node υ_i, $d(\upsilon_i)$, is defined as follows:

$$d(\upsilon_i) = \sum_{j \in \mathcal{N}_i} a_{ij}. \tag{1}$$

The degree matrix $\Delta \in \mathbb{R}^{N \times N}$ is defined as $\Delta = diag(d(\upsilon_i))$, and the Laplacian is defined as $L = \Delta - \mathcal{A}$. A path from node i to node j is a sequence of successive edges in the form of $\{(\upsilon_i, \upsilon_m), (\upsilon_m, \upsilon_p), ..., (\upsilon_k, \upsilon_j)\}$. An undirected graph is said to have a spanning tree, if there exists a path from a node (called the root) to every other node in the graph [38]. We use υ_l to annotate the leader of the multi-agent system, which generates a reference signal for other nodes to follow. The leader only sends the information to other agents and cannot receive any information from other agents. We define the leader-followers graph as a graph that contains all followers and the leader. We consider that the leader-follower graph topology contains a spanning tree with the root node being the leader.

2.2 Notation

In this thesis, x^+ represents one-step ahead state, $tr(.)$ denotes the trace of matrix, and $\mathbf{1}_N \in \mathbb{R}^N$ is the vector of 1's. For an arbitrary matrix A, $\bar{\sigma}(A)$ denotes the maximum singular value of the matrix and $\underline{\sigma}(A)$ denotes the minimum singular value of the matrix A. For a vector x the notation $||x||$ denotes Euclidean norm and the Frobenius norm of a matrix A is $||A||_F = \sqrt{tr(A^T A)}$. With I_n, we denote an $n \times n$ identity matrix. With $\otimes$ we denote a Kronecker product.

2.3 Formation Control of Multi-Agent Systems

In multi-agent systems, formation control refers to the design of a control law to stabilize each agent's position with respect to its neighbours in order to maintain a predefined geometrical shape [39]. In the formation control of multi-agent systems, the interaction among agents determines agents' control variables and sensed variables [33]. A general categorization of formation control includes position-based control, displacement-based control, and distance-based control [40].

Position-based control: In the position-based formation control of multi-agent systems, each agent actively controls its position, and each agent reaches its desired positions without interacting with other agents. In the position-based control, each agent senses its positions with respect to a global coordinate system to control its positions to achieve the desired formation prescribed by the desired framework with respect to the global coordinate system. As a result, this type of formation control requires high sensing capability while the interaction among the agents is unnecessary. Interactions among the agents can be considered. In the position-based control, interactions among agents improves the control performance and maintains the formation along with the system trajectory [33].

Displacement-based control: In displacement-based formation control of the multi-agent system, each agent maintains the desired relative position (displacements) from its neighbors. It is assumed that all of the agents' local reference frames are aligned, i.e. agents' coordinate systems are orientation-aligned local coordinate systems [41]. In this approach, the control variable of each agent is its neighbors' displacement, and each agent controls its relative position with respect to its neighboring agents [42].

Distance-based control: In distance-based formation control of multi-agent systems, each agent controls the inter-agent distances of its neighboring agents. In distance-based formation control, agents need to interact to maintain their formation as a rigid body, and interaction topology among agents is usually described by graph rigidity or persistence [43]. Since agents control only their inter-agent distances to achieve the predefined desired distances, this type of formation control requires less sensing capability than other methods [33]. Agents can measure the relative positions of their adjacent agents within the local coordinate frames, and adjust the norms of the relative positions to control their formation [42]. On the other hand, this method requires many interactions among agents, which plays

	Position-based	Displacement-based	Distance-based
Control variables	Absolute position	Relative positions of neighbors	Norms of relative positions of neighbors (Interagent distances)
Coordinate systems	Global coordinate	Orientation-aligned local coordinate systems	Local coordinate systems

Table 1: Comparison of different types of formation control.

a crucial role in the control of multi-agent systems.

In Table 1, we compare the diffident types of formation control [33, 42]. Displacement-based control is moderate in terms of both sensing capability and interaction topology compared to the other approaches. In this thesis, we consider a leader-followers multi-agent system. We consider that all agents measure their states with respect to a global coordinate system. The control objective is to maintain the desired formation along their trajectory and follow the leader. In the leader-followers scenario, the leader knows the reference trajectory, and other agents should follow the leader while maintaining the desired formation. To achieve this goal, we use the displacement-based control approach. The consensus-based formation control can be categorized as a displacement-based control where the desired relative displacements of agents are set to zero.

2.4 Discrete-Time Nonlinear Multi-Agent Systems

In this thesis, we consider that the physical system is modeled as a nonlinear, discrete-time, multi-agent system that consists of N agents. The dynamics of each agent is given by

$$x_i^+ = f_i(x_i) + u_i + w_i, \qquad i \in \{1, 2, .., N\}, \tag{2}$$

where $x_i \in \mathbb{R}^n$ is the system state, $u_i \in \mathbb{R}^n$ is the control input, and $w \in \mathbb{R}^n$ is the disturbance. Nonlinear functions $f_i(.) \in \mathbb{R}^n \to \mathbb{R}^n$ is assumed to be locally Lipschitz. We also consider that the nonlinear dynamics f_i and disturbance w_i are unknown. The overall system dynamics can be written as

$$\mathbf{x}^+ = f(\mathbf{x}) + u + w, \tag{3}$$

where the stacked state vector is $\mathbf{x} = [x_1^T, ..., x_N^T]^T \in \mathbb{R}^{nN}$, $f(\mathbf{x}) = [f_1^T(x_1), ..., f_N^T(x_N)]^T \in \mathbb{R}^{nN} \to \mathbb{R}^{nN}$, stacked control input $u = [u_1^T, ..., u_N^T]^T \in \mathbb{R}^{nN}$, and $w = [w_1^T, ..., w_N^T]^T \in \mathbb{R}^{nN}$.

Assumption 1. *The unknown disturbance w is bounded by $||w|| \leq w_M$, with w_M a fixed bound.*

The leader dynamics is defined as follows:

$$x_l^+ = f_l(x_l), \tag{4}$$

where $f_l \in \mathbb{R}^n$, and satisfies the following assumption.

Assumption 2. *The unknown leader dynamics $f_l(x_l)$ is bounded by $||f_l(x_l)|| \leq F_M$, with a fixed bound F_M. The leader trajectory x_l is in a bounded region, i.e., $||x_l(t)|| < X_M, \forall t$, with X_M a constant bound.*

Assumption 3. *The leader trajectory x_l is in a bounded region, i.e., $||x_l(t)|| < X_M, \forall t$, with X_M a constant bound.*

Assumption 4. *The leader-follower graph topology is a weighted undirected connected graph that contains a spanning tree with the root node being the leader.*

Definition 2 ([44])**.** *Consider the following nonlinear system*

$$x^+ = F(x, t), \tag{5}$$

where x denotes the system state, and F is a nonlinear function. Let the initial time be t_0, and the initial condition be $x(t_0) = x_0$. The solution of (5) is said to be uniformly ultimately bounded (UUB), if for all initial states x_0, there exists a $b \in \mathbb{R}$ and a time $\mathcal{T}_f(c, x_0) \in \mathbb{Z}^+$ such that $||x|| \leq b$ for all $t \geq t_0 + \mathcal{T}_f$.

2.5 Cyber-Attack Classification

Attacks in multi-agent systems can be categorized as attacks on agents (nodes) and attacks on communication links (edges) [45]. Manipulating agents' sensors and actuators' data are considered as an attack on the node. Changing and jamming the neighboring data in a multi-agent system can be considered as an attack on edge. In the attack on nodes, the attacker injects disruptive signals into the sensor or actuator channels of the agents. In addition, sensor spoofing can be considered in this category. In an edge attack, the attacker launches a false signal into the neighboring communication channels of an agent to interfere with the original signal before it reaches the receiver or the attacker keeps the neighboring communication channel busy to avoid the transmission of the original signal.

In this thesis, we study the FDI attack on sensor channel and actuator channel of agents as an attack on nodes and study the FDI attack on neighboring communication channels of multi-agent systems as the attacks on edges.

2.5.1 Attack Model

This thesis focuses on the class of FDI attacks that alter the data transmitted through communication channels. Different FDI attacks can be performed according to the attacker's available resources. In this thesis, we focus on the detection of FDIs attacks on the sensor, actuator, and neighboring channels.

In this subsection, different types of FDI attack on distributed multi-agent systems are modeled [14]. We consider that the attacker has access to the agent communication channels, and it can change the actuator, sensor and neighboring communication channels data of each agents.

i) Attack on the actuator channel: Each agent uses the actuator channel to send its control input to the plant, and the attacker can perform the FDI attack on this channel and change the control input. The attack on the actuator channel can be modelled as

$$u_i^c = u_i + \kappa_i u_i^a, \tag{6}$$

where u_i^c is the corrupted control input, u_i^a is the attacker signal, and flag $\kappa_i = 1$ when there is an attack in the actuator channel ($\kappa_i = 0$ in the attack-free case),

$$\kappa_i = \begin{cases} 1, & \text{if agent } i \text{ is under actuator attack,} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

ii) Attack on the sensor channel: The system output is sent to the controller through the sensor channel, and the attacker can change the sensor data by injecting some false data into the sensor channel. We consider that the system's state is measurable by sensors [46]. As a result, the attack on the sensor channel can be modelled as follows:

$$x_i^c = x_i + \lambda_i x_i^a, \tag{8}$$

where x_i^c is the corrupted sensor data, x_i^a is the attacker signal, and flag $\lambda_i = 0$ when there is not an attack on sensor channel, otherwise $\lambda_i = 1$,

$$\lambda_i = \begin{cases} 1, & \text{if agent } i \text{ is under sensor attack,} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Since the controller uses the sensor data to generate control input, by corrupting the sensor channel data, the attacker can affect the control input.

iii) Attack on neighboring communication channels: We consider that each agent's control input is a function of the agent's sensor data and its neighboring data. Define ζ_i as aggregated outputs of i-th agent's neighbors. The agent i receives the data from its neighboring communication channels (see Fig. 1). The i-th agent's control input

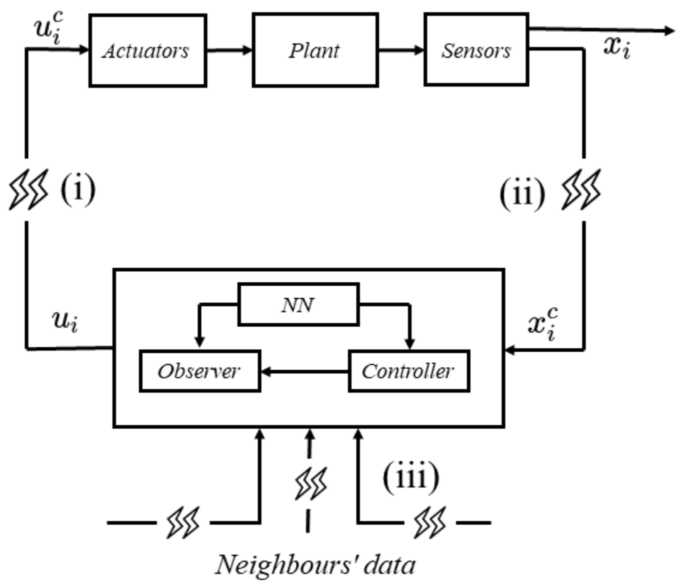


Figure 1: System architecture of the agent i.

can be expressed by a known function $\mathcal{F}_i$ as follows:

$$u_i = \mathcal{F}_i(x_i, \zeta_i). \tag{10}$$

Let $\bar{x}_j$ denotes the sensor measurement of the agent j that is sent to its neighbors, including agent $i \in \mathcal{N}_j$. Attack on neighboring communication channel of agent i occurs when the attacker changes the sensor data of agent $j \in \mathcal{N}_i$ by injecting $\bar{x}^a_{ji}$. Therefore, neighboring communication channel of agent i, in the presence of attacks, can be modelled as [15]:

$$\bar{x}^c_{ji} = \bar{x}_j + \Psi^i_j \bar{x}^a_{ji}, \tag{11}$$

where $\bar{x}^c_{ji}$ is the corrupted sensor data that agent i receives form agent j, and flag $\Psi^i_j = 1$ when there is an attack on neighboring communication channels and the attacker change the data that agent i receives form agent j, otherwise $\Psi^i_j = 0$,

$$\Psi^i_j = \begin{cases} 1, & \text{if the attacker change the data that agent } i \text{ receives form agent } j, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

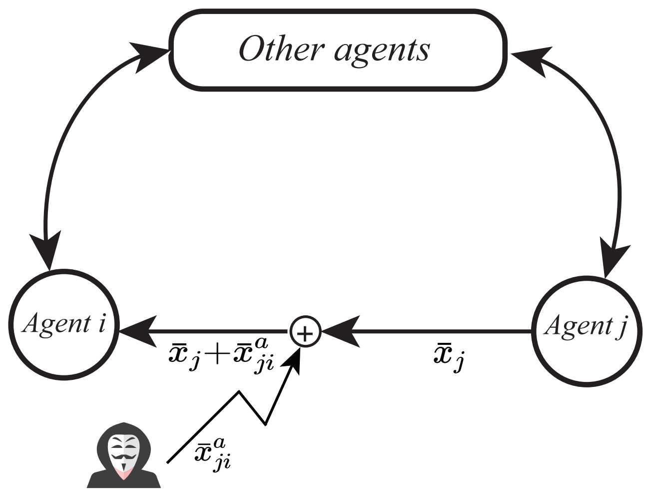


Figure 2: Multi-agent system under attack in neighboring communication channel.

When the attacker, by injecting $\bar{x}_{ji}^a$, changes $\bar{x}_j$ to $\bar{x}_{ji}^c$, the ζ_i is changed to ζ_i^c. Therefore, we have the following expression for the corrupted control input of agent i:

$$u_i' = \mathcal{F}_i(x_i, \zeta_i^c), \tag{13}$$

where u_i' is the i-th agent's control input, which has been affected by the attack on the agent's neighboring communication channels. Consequently, an attack on neighboring communication channels affects the agent's control input.

Remark 1. *The $\bar{x}_j$ is the sensor data that agent j sends to its neighbors and it is equal to x_j.*

All types of attacks can compromise the agent's control input, which can lead to degrading the control performance, formation and possibly causing collision between agents.

Remark 2. *Note that an attack on the actuator channel directly affects the agent plant, and an attack on the sensor and neighboring communication channels directly affects the controller and observer (see Fig. 1).*

We propose a distributed, residual-based attack detection method to detect the stated types of attacks. The attack detection method's objective is to enable each agent to detect attacks in its communication channels. By exploiting the proposed attack detection method, each agent can detect attacks on its sensor channel, actuator channel, and its neighboring communication channels.

2.6 Neural Networks

Neural networks are modelled based on the nervous system, with the neuron as the basic unit. In the context of the control system, neural networks have vast applications such as system identification, and nonlinearity estimation and adaptive control of nonlinear control systems. In this subsection, we provide a brief background on neural networks (NN) and model an NN to exploit it in closed-loop control systems. In the closed-loop control systems, NN can provide stabilizing controls for the system and maintain all its weights bounded.

2.6.1 Radial Basis Function Neural Networks

One of the most significant capabilities of neural networks is the function approximation property. NN is used to cope with the demand for controlling the uncertain and unknown nonlinear control systems using the capability of learning online through the stable closed-loop control process and exploiting the learned knowledge in the control tasks to improve the control performance and assure the stability of the closed-loop system.

The main feature of adaptive control is to approximate the unknown functions through online adjustment of controller parameters. This feature helps a designer to achieve the desired level for control performance. However, adaptive control's adjusting ability is limited since it needs to recalculate (or re-adapt) the controller parameters even for repeating the same control task.

Using the adaptive neural network enables us to approximate the unknown nonlinearity of system dynamics. Moreover, the adaptive tuning law establishes the boundedness of the neural network weights matrix in closed-loop systems. To derive the adaptive law, we

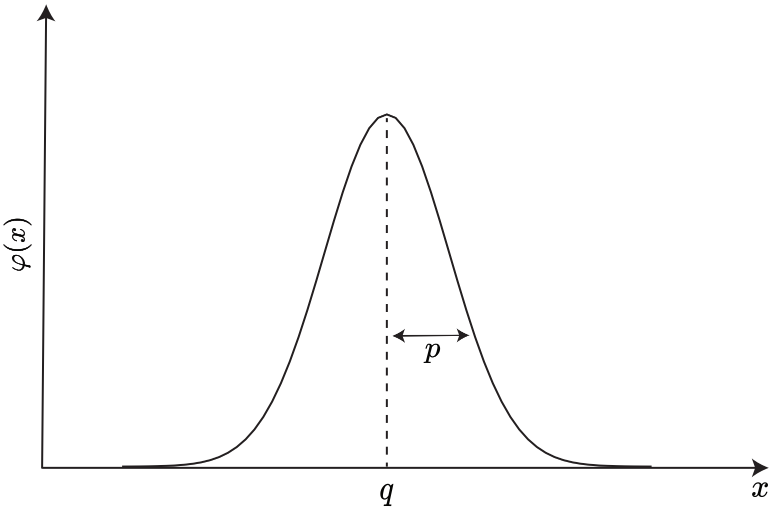


Figure 3: Gaussian activation function.

proposed to use the Lyapunov stability theorem.

In this thesis we used the RBF network for the function approximation, which is developed in [47]. The RBF networks belong to the class of linear in parameter neural networks. By considering the RBFNN input as $x = [x_1, ..., x_n]^T \in \mathbb{R}^n$, its output as $y = [y_1, ..., y_m]^T \in R^m$ and ϑ neurons in hidden layers over a compact set $\Omega_x \subset \mathbb{R}^n$, the RBFNN can be expressed as

$$y_i = \sum_{j=1}^{\vartheta} w_{ij}\varphi_j(x) = w_i^T \phi(x), \quad i = 1, ..., m, \tag{14}$$

where $w_i = [w_{i1}, w_{i2}, \ldots, w_{i\vartheta}]^T$ is the weight vector where w_{ij} is the weight between neuron j and output i. The activation function $\varphi_i(x)$ is a radial basis function with $q_i \in \mathbb{R}^n$ as the center of the activation function, and p_i as the width of the Gaussian functions. Moreover $\phi(x) = [\varphi_1(x), ..., \varphi_\vartheta(x)]^T$, and the Gaussian activation function is defined as

$$\varphi_i(x) = \exp\left[\frac{-(x-q_i)^T (x-q_i)}{p_i^2}\right], \quad i = 1, 2, \ldots, \vartheta. \tag{15}$$

By defining the weights matrix $W = [w_1, ..., w_m]$ the RBFNN can be re-written as

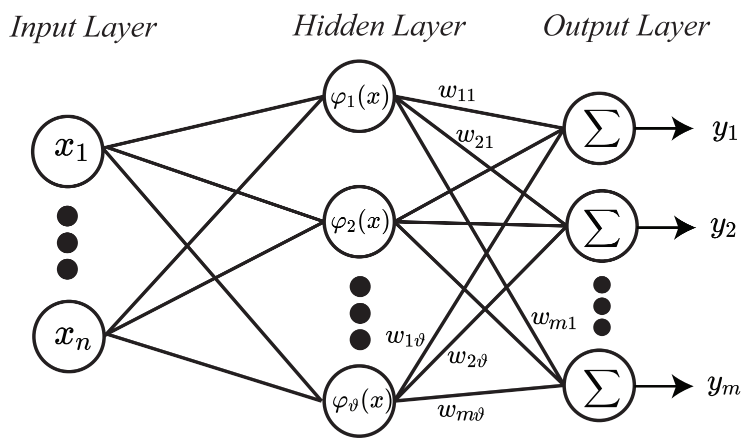


Figure 4: RBF neural network.

$$f(x) = W^T \phi(x). \tag{16}$$

In the next chapter, the described RBFNN is used for the approximation of the unknown system dynamics.

Chapter 3

A Formation Control Design for Discrete-Time Nonlinear Multi-Agent Systems

3.1 Problem Formulation

For the multi-agent described by (2), consisting of one leader and N followers, the objective of the system is to reach the desired formation shape and maintain the desired form along their trajectory. As a result, the objective of each agent can be defined as to reach a desired relative position with respect to its neighbouring agents while following the leader. We designed a distributed control law such that the relative position between neighbouring agents i and j converges to the bounded, desired, relative inter-agent displacement d_{ij}:

$$x_i - x_j \to d_{ij}, \quad i, j \in \{1, ..., N\}. \tag{17}$$

We define the desired, relative inter-agent displacement between agent i and the leader as d_i. As a result we can represent $d_{ij} = d_i - d_j$ as a desired relative inter-agent displacement between agent i and agent j. We define the tracking error between the agent i and the leader as

$$\delta_i = x_l - x_i - d_i. \tag{18}$$

The local formation error of i-th agent is given by

$$e_i = \sum_{j \in \mathcal{N}_i} a_{ij}(x_j - x_i - d_{ji}) + b_i(x_l - x_i - d_i), \tag{19}$$

where b_i is the direct gain from the leader to agent i and $b_i \geq 0$, with $b_i > 0$ for at least one agent. Consequently, there exists at least one agent (i-th) such that the leader sends its information to it, i.e. $b_i \neq 0$.

Remark 3. *The existence of a communication link between the leader and at least one of the agents is a standard consideration that can be seen in the formation control problem of multi-agent systems [48–51]. This consideration relaxes Assumption 5 in [52], where authors considered that the agents are aware of the leader's information.*

Let $e = [e_1^T, ..., e_N^T]^T \in \mathbb{R}^{nN}$, and matrix $B = diag(b_i)$. Similar to [53], the global form of formation error is given by

$$e = -[(L + B) \otimes I_n](\mathbf{x} - \mathbf{1}_N \otimes x_l - d), \tag{20}$$

where $d = [d_1^T, ..., d_N^T]^T \in \mathbb{R}^{nN}$. Since we considered $b_i \neq 0$ for at least one agent, matrix $L + B$ is full-rank and invertible [51]. The desired relative position between agents and the leader is bounded by $||d|| < d_M$, with d_M a fixed bound.

By defining the stacked tracking error as $\delta = [\delta_1^T, ..., \delta_N^T]^T \in \mathbb{R}^{nN}$ the global formation error can be rewritten as

$$e = -[(L + B) \otimes I_n]\delta. \tag{21}$$

The global formation error dynamics is given by

$$e^+ = -[(L + B) \otimes I_n](f(\mathbf{x}) + u + w - \mathbf{1}_N \otimes x_l^+ - d). \tag{22}$$

Lemma 1 ([51])**.** *Let the undirected graph contains a spanning tree, with the leader as a root and* $\exists b_i \neq 0$*. Then,* $||\delta|| \leq ||e||/\underline{\sigma}(L+B)$*.*

Lemma 2. *Let* $\bar{L} \in \mathbb{R}^{n\times n}$ *be a positive definite symmetric matrix. Select a positive diagonal matrix* $K \in \mathbb{R}^{n\times n}$ *such that* $\bar{\sigma}(K) < 1/\bar{\sigma}(\bar{L})$*, and define*

$$P = I - K\bar{L}. \tag{23}$$

Then, $||P|| < 1$*.*

Proof. Let $\bar{\sigma}(K) < 1/\bar{\sigma}(\bar{L})$, and $\lambda_i(K\bar{L})$ be i-th eigenvalue of $K\bar{L}$. Using the fact that for a positive definite symmetric matrix A, $\lambda_{max}(A) \leq \bar{\sigma}(A)$ [54], one has

$$\lambda_i(K\bar{L}) \leq \lambda_{max}(K)\lambda_{max}(\bar{L}) \leq \bar{\sigma}(K)\bar{\sigma}(\bar{L}) < 1. \tag{24}$$

As matrix $K\bar{L}$ is a positive definite matrix, it follows that $0 < \lambda_i(K\bar{L}) < 1$, for $i \in \{1,...,n\}$, which implies $0 < \lambda_i(P) < 1$. For a symmetric matrix P, one can get $||P|| = \sqrt{\lambda_{max}(P^TP)} = |\lambda_{max}(P)|$, and consequently $||P|| < 1$. □

Remark 4. *Note that if* $d_i = 0$ *for* $i \in \{1,...,N\}$ *in (18), the formation control setup becomes a consensus problem [33], where the result of this thesis can still be applied.*

3.2 Nonlinearity Dynamics Estimation Using Neural Network

To design a controller for the multi-agent system (2), it is required to approximate the nonlinearity $f_i(x_i)$. To approximate the unknown nonlinear dynamics, each agent has an RBFNN.

It has been proven in [55] that an RBF network (16), with sufficiently large number of nodes ϑ and appropriately placed node centers and variances, can approximate any continuous function $f_i(x) : \Omega_x \rightarrow R^m$ over a compact set $\Omega_x \subset R^n$ to an arbitrary accuracy as

$$f_i(x_i) = W_i^T \phi_i(x_i) + \epsilon_i \quad \forall x \in \Omega_x \tag{25}$$

where the $W_i \in \mathbb{R}^{\vartheta_i \times n}$ is the desired constant unknown weights matrix, $\phi_i(x_i) \in \mathbb{R}^{\vartheta_i \times n}$ is an NN activation function, and ϵ_i is the NN approximation error. The approximation of the overall dynamics nonlinearity $f(\mathbf{x}) = [f_1^T(x_1), ..., f_N^T(x_N)]^T$ can be written as

$$f(\mathbf{x}) = W^T \phi(\mathbf{x}) + \epsilon, \tag{26}$$

where the overall ideal NN weights matrix W is defined as $W = diag(W_1, ..., W_N)$, $\phi(\mathbf{x}) = [\phi_1^T(x_1), ..., \phi_N^T(x_N)]^T$, and $\epsilon = [\epsilon_1^T, ..., \epsilon_N^T]^T$.

The estimation of the nonlinearity in the dynamics of multi-agent systems using RBFNN is given by

$$\hat{f}_i(x_i) = \hat{W}_i^T \phi_i(x_i), \tag{27}$$

where $\hat{W}_i$ is the current estimated NN weights matrix. The overall nonlinearity $f(\mathbf{x})$ over a compact set Ω can be estimated as follows:

$$\hat{f}(\mathbf{x}) = \hat{W}^T \phi(\mathbf{x}), \tag{28}$$

where the estimation of the ideal weights matrix $\hat{W}$ is defined as $\hat{W} = diag(\hat{W}_1, ..., \hat{W}_N)$. The ideal weight vector W is an artificial quantity required for analytical purposes, and is defined as the value of $\hat{W}$ that minimizes $|\epsilon|$ for all $x \in \Omega_x \subset R^n$, i.e.

$$W \triangleq \arg \min_{W \in R^{\vartheta \times m}} \left\{ \sup_{x \in \Omega_x} \left| f(\mathbf{x}) - \hat{W}^T \phi(\mathbf{x}) \right| \right\} \tag{29}$$

From (15) one can write $||\phi(\mathbf{x})|| \leq \phi_M$, with a fixed bound ϕ_M. Considering sufficiently large number of neurons, the NN approximation error ϵ is bounded, i.e., $||\epsilon|| \leq \epsilon_M$ [47]. We use a standard assumption [48, 56–58] as follows:

Assumption 5. *Unknown ideal NN weights matrix W is bounded, i.e., $||W||_F \leq W_M$, with a fixed bound W_M.*

3.3 Control Design

Based on the defined control objective for the leader-followers multi-agent system (2) which is maintaining the desired formation along the trajectory, and by considering the estimation of unknown nonlinearity $\hat{f}_i(x_i)$ as (27), for each agent we propose the following distributed control law

$$u_i = -\hat{f}_i(x_i) + x_i + k_i e_i, \tag{30}$$

where the positive diagonal matrix $k_i \in \mathbb{R}^{n \times n}$ is the control gain. By using (27), the control law (30) can be rewritten as

$$u_i = -\hat{W}_i^T \phi_i(x_i) + x_i + k_i e_i, \tag{31}$$

or in the stacked form as

$$u = -\hat{W}^T \phi(\mathbf{x}) + \mathbf{x} + Ke, \tag{32}$$

where $K = diag(k_1, ..., k_N)$. By defining $\tilde{W}_i = W_i - \hat{W}_i$, the function estimation error is given by

$$\tilde{f}(\mathbf{x}) = f(\mathbf{x}) - \hat{f}(\mathbf{x}) = \tilde{W}^T \phi(\mathbf{x}) + \epsilon, \tag{33}$$

with $\tilde{W} = diag(\tilde{W}_i)$ being the NN weights matrix estimation error. Let the observer error $\tilde{x}_i = x_i - \hat{x}_i$ be the attack detection residual. Then, one can obtain the residual dynamics as

$$\tilde{x}_i^+ = G_i \tilde{x}_i + \tilde{W}_i^T \phi_i(x_i) + \epsilon_i + w_i + e_i. \tag{34}$$

Let $\tilde{\mathbf{x}} = [\tilde{x}_1^T, ..., \tilde{x}_N^T]^T \in \mathbb{R}^{nN}$. The stacked residual dynamics is given by

$$\tilde{\mathbf{x}}^+ = G\tilde{\mathbf{x}} + \tilde{W}^T\phi(\mathbf{x}) + w + \epsilon + e, \tag{35}$$

where $G = diag(G_1, ..., G_N) \in \mathbb{R}^{nN\times nN}$. Motivated by [59], let us consider the NN weights tuning law as follows

$$\hat{W}_i^+ = \hat{W}_i + \alpha\phi_i(x_i)\bar{h}_i^T - F_i\hat{W}_i, \tag{36}$$

where constant $\alpha > 0$, $F_i = \gamma I_{\vartheta_i}$, with $0 < \gamma < 1$, and

$$\bar{h}_i = \tilde{W}_i^T\phi_i(x_i) + \epsilon_i + w_i. \tag{37}$$

Theorem 1. *Consider a multi-agent system in the absence of attack that is modelled by a weighted undirected graph, with agents modelled by (2), under the Assumptions 1-5. Select the control law (31), and the NN weights matrix tuning law (36). If the following conditions hold*

$$\bar{\sigma}(K) < \frac{1}{\bar{\sigma}(\bar{L})}, \tag{38}$$

$$\alpha < \frac{1}{\phi_M^2}, \tag{39}$$

with $\bar{L} = (L + B) \otimes I_n$, and $P = I_{Nn} - K\bar{L}$, then the formation error and the NN weights matrix estimation error are UUB, with practical bounds given by (56) and (59) respectively.

Proof. Consider the following Lyapunov function candidate

$$V = \frac{1}{c}e^Te + \frac{1}{\alpha}tr(\tilde{W}^T\tilde{W}), \tag{40}$$

where $c = \frac{\bar{\sigma}^2(\bar{L})+\bar{\sigma}^2(\bar{L})\bar{\sigma}^2(\bar{P})}{1-\bar{\sigma}^2(P)}$. From Lemma 2, one can conclude that $c > 0$. Let us define

$$\mu = w + \epsilon. \tag{41}$$

From Assumption 1, μ is bounded by

$$||\mu|| \leq \mu_M, \tag{42}$$

where $\mu_M = \epsilon_m + w_M$. Moreover, let us define

$$\nu = \mathbf{1}_N \otimes x_l - \mathbf{1}_N \otimes f_l(x_l). \tag{43}$$

Then one has

$$||\nu|| \leq \nu_M, \tag{44}$$

where ν_M can be obtained using Assumptions 2 and 3. By defining

$$\theta = \tilde{W}^T \phi(\mathbf{x}), \tag{45}$$

from (22) the first difference of Lyapunov function candidate $V_1 = \frac{1}{c} e^T e$ is derived as follows:

$$\begin{aligned} \Delta V_1 = & -\frac{1}{c} e^T (I - P^T P) e - \frac{2}{c} \theta^T \bar{L}^T P e - \frac{2}{c} \mu^T \bar{L}^T P e - \frac{2}{c} \nu^T \bar{L}^T P e + \frac{1}{c} \theta^T \bar{L}^T \bar{L} \theta \\ & + \frac{2}{c} \theta^T \bar{L}^T \bar{L} \mu + \frac{2}{c} \theta^T \bar{L}^T \bar{L} \nu + \frac{1}{c} \mu^T \bar{L}^T \bar{L} \mu + \frac{1}{c} \nu^T \bar{L}^T \bar{L} \nu + \frac{1}{c} \mu^T \bar{L}^T \bar{L} \nu, \end{aligned} \tag{46}$$

and from (36), the first difference of Lyapunov function candidate $V_2 = \frac{1}{\alpha} tr(\tilde{W}^T \tilde{W})$ is derived as

$$\begin{aligned} \Delta V_2 = & \frac{1}{\alpha} tr[-2\alpha \tilde{W}^T \phi \phi^T \tilde{W} - 2\alpha \mu \phi^T \tilde{W} + 2\gamma \tilde{W}^T F \hat{W} + \alpha^2 \tilde{W}^T \phi \phi^T \phi \phi^T \tilde{W} \\ & + 2\alpha^2 \tilde{W} \phi \phi^T \phi \mu^T - 2\alpha \tilde{W} \phi \phi^T F \hat{W} + \alpha^2 \mu \phi^T \phi \mu^T - 2\alpha \mu \phi^T F \hat{W} + \hat{W}^T F^T F \hat{W}], \end{aligned} \tag{47}$$

From (46) and (47), the first difference of Lyapunov function candidate is obtained as

$$\begin{aligned}\Delta V = &-\frac{1}{c}e^T(I-P^TP)e-\frac{2}{c}\theta^T\bar{L}^TPe-\frac{2}{c}\mu^T\bar{L}^TPe-\frac{2}{c}\nu^T\bar{L}^TPe+\frac{1}{c}\theta^T\bar{L}^T\bar{L}\theta+\frac{2}{c}\theta^T\bar{L}^T\bar{L}\mu\\&+\frac{2}{c}\theta^T\bar{L}^T\bar{L}\nu+\frac{1}{c}\mu^T\bar{L}^T\bar{L}\mu+\frac{1}{c}\nu^T\bar{L}^T\bar{L}\nu+\frac{1}{c}\mu^T\bar{L}^T\bar{L}\nu+\frac{1}{\alpha}tr[-2\alpha\tilde{W}^T\phi\phi^T\tilde{W}\\&-2\alpha\mu\phi^T\tilde{W}+2\gamma\tilde{W}^TF\hat{W}+\alpha^2\tilde{W}^T\phi\phi^T\phi\phi^T\tilde{W}+2\alpha^2\tilde{W}\phi\phi^T\phi\mu^T-2\alpha\tilde{W}\phi\phi^TF\hat{W}\\&+\alpha^2\mu\phi^T\phi\mu^T-2\alpha\mu\phi^TF\hat{W}+\hat{W}^TF^TF\hat{W}].\end{aligned} \tag{48}$$

Reorganizing the terms in (48), yields

$$\begin{aligned}\Delta V \leq &-\frac{1}{c}e^T(I-P^TP)e-\frac{2}{c}\theta^T\bar{L}^TPe-\frac{2}{c}\mu^T\bar{L}^TPe-\frac{2}{c}\nu^T\bar{L}^TPe+\frac{1}{c}\theta^T\bar{L}^T\bar{L}\theta\\&+\frac{2}{c}\theta^T\bar{L}^T\bar{L}\mu+\frac{2}{c}\theta^T\bar{L}^T\bar{L}\nu+\frac{1}{c}\mu^T\bar{L}^T\bar{L}\mu+\frac{1}{c}\nu^T\bar{L}^T\bar{L}\nu+\alpha\phi_M^2\mu_M^2+\frac{2}{c}\mu^T\bar{L}^T\bar{L}\nu\\&-(2-\alpha\phi^T\phi)\theta^T\theta-2(1-\alpha\phi^T\phi)\theta^T\mu+\frac{1}{\alpha}tr[\gamma^2\hat{W}^T\hat{W}+2\gamma\tilde{W}^T(W-\tilde{W})\\&+2\alpha\gamma\mu\phi^T(\tilde{W}-W)].\end{aligned} \tag{49}$$

Consider any two vectors v_1 and $v_2 \in \mathbb{R}^{nN}$; applying Young's inequality and Cauchy inequality, one can write $v_1^Tv_2 \leq \frac{a_1||v_1||^2}{2a_2}+\frac{a_2||v_2||^2}{2a_1}$, where a_1 and a_2 are positive constants. From this one can write

$$-\frac{2}{c}\theta^T\bar{L}^TPe \leq \frac{2}{c}\bar{\sigma}(\bar{L})\bar{\sigma}(P)\big(\frac{a_1}{2a_2}||\theta||^2+\frac{a_2}{2a_1}||e||^2\big), \tag{50}$$

where $a_1 = 2\bar{\sigma}(P)\bar{\sigma}(\bar{L})$, and $a_2 = 1-\bar{\sigma}^2(P)$. Using the following equation

$$\frac{1}{c}\bar{\sigma}^2(\bar{L})+\frac{1}{c}\bar{\sigma}(\bar{L})\bar{\sigma}(P)\frac{a_1}{a_2} = 1, \tag{51}$$

and considering (50), (49) can be written as

$$
\begin{aligned}
\Delta V \leq & -\frac{1}{2c}[1-\bar{\sigma}^2(P)]e^T e - \frac{2}{c}\mu^T \bar{L}^T P e - \frac{2}{c}\nu^T \bar{L}^T P e + \frac{\bar{\sigma}^2(\bar{L})}{c}(\mu_M + \nu_M)^2 + 2\gamma\phi_M W_M \mu_M \\
& + \alpha\phi_M^2\mu_M^2 - (1-\alpha\phi^T\phi)\theta^T\theta - 2(1-\alpha\phi^T\phi)\theta^T\mu + 2\gamma\theta^T\mu + \frac{2}{c}\theta^T\bar{L}^T\bar{L}\mu + \frac{2}{c}\theta^T\bar{L}^T\bar{L}\nu \\
& + \frac{1}{\alpha}tr[\gamma^2\hat{W}^T\hat{W} + 2\gamma\tilde{W}^T(W-\tilde{W})].
\end{aligned}
\tag{52}
$$

By defining $B_M = 1-\alpha\phi_M^2$, $\Gamma_1 = 1+\gamma+\frac{\bar{\sigma}^2(\bar{L})}{c}$, and $\Gamma_2 = \frac{\bar{\sigma}^2(\bar{L})}{c}$ and completing the squares for θ, we have

$$
\begin{aligned}
\Delta V \leq & -\frac{1}{2c}[1-\bar{\sigma}^2(P)]e^T e - \frac{2}{c}\mu^T \bar{L}^T P e - \frac{2}{c}\nu^T \bar{L}^T P e - B_M||\theta - \frac{\Gamma_1}{B_M}\mu_M - \frac{\Gamma_2}{B_M}\nu_M||^2 \\
& + \frac{1}{B_M}(\Gamma_1\mu_M + \Gamma_2\nu_M)^2 - \frac{1}{\alpha}[\gamma(2-\gamma)||\tilde{W}||_F^2 - 2\gamma(1-\gamma)W_M||\tilde{W}||_F - \gamma^2 W_M^2] \\
& + \frac{\bar{\sigma}^2(\bar{L})}{c}(\mu_M+\nu_M)^2 + 2\gamma\phi_M W_M\mu_M + \alpha\phi_M^2\mu_M^2.
\end{aligned}
\tag{53}
$$

Now by completing the squares for $\tilde{W}$, one can have

$$
\begin{aligned}
\Delta V \leq & - B_M||\theta - \frac{\Gamma_1}{B_M}\mu_M - \frac{\Gamma_2}{B_M}\nu_M||^2 + -\frac{1}{\alpha}\gamma(2-\gamma)\left[||\tilde{W}||_F - \frac{1-\gamma}{2-\gamma}W_M\right]^2 \\
& - \frac{1}{c}\left[\frac{1}{2}(1-\bar{\sigma}^2(P))||e||^2 - 2\Lambda_1||e|| - \Lambda_2\right],
\end{aligned}
\tag{54}
$$

where $\Lambda_1 = \bar{\sigma}(\bar{L})\bar{\sigma}(P)(\mu_M+\nu_M)$, and Λ_2 is

$$
\Lambda_2 = \bar{\sigma}^2(\bar{L})(\mu_M+\nu_M)^2 + 2c\gamma\phi_M W_M\mu_M + \alpha c\phi_M^2\mu_M^2 + \frac{c}{B_M}(\Gamma_1\mu_M+\Gamma_2\nu_M)^2 + \frac{c}{\alpha}\frac{\gamma}{2-\gamma}W_M^2].
\tag{55}
$$

Then, $\Delta V < 0$ as long as (38), and (39) hold, and the quadratic term for e in (54) is positive which is guaranteed when

$$
||e|| > \frac{2\Lambda_1 + \sqrt{4\Lambda_1^2 + 2(1-\bar{\sigma}^2(P))\Lambda_2}}{1-\bar{\sigma}^2(P)} \triangleq e_M.
\tag{56}
$$

Similarly by reorganizing terms in (53) and completing square for e, one can obtain

$$\begin{aligned}\Delta V \leq &-\frac{1}{2c}(1-\bar{\sigma}^2(P))||e-\frac{2}{1-\bar{\sigma}^2(P)}\bar{\sigma}(\bar{L})\bar{\sigma}(P)(\mu_m+\nu_m)||^2-B_M||\theta-\frac{\Gamma_1}{B_M}\mu_M \\ &-\frac{\Gamma_2}{B_M}\nu_M||^2-\frac{1}{\alpha}[\gamma(2-\gamma)||\tilde{W}||_F^2-2\gamma(1-\gamma)W_M||\tilde{W}||_F-\xi],\end{aligned} \tag{57}$$

with

$$\xi = \gamma^2 W_M^2+\alpha(\mu_M+\nu_M)^2+\alpha^2\phi_M^2\mu_M^2+2\alpha\gamma\phi_M W_M\mu_M+\frac{\alpha}{B_M^2}(\Gamma_1\mu_M+\Gamma_2\nu_M)^2. \tag{58}$$

Then, $\Delta V < 0$ as long as (38), and (39) hold, and the quadratic term for $\tilde{W}$ in (54) is positive which is guaranteed when

$$||\tilde{W}||_F > \frac{\gamma(1-\gamma)W_M+\sqrt{\gamma^2(1-\gamma)^2W_M^2+\gamma(2-\gamma)\xi}}{\gamma(2-\gamma)} \triangleq \tilde{W}_M. \tag{59}$$

We conclude that ΔV is negative outside a compact set as long as (38) and (39) are satisfied and either (56) or (59) holds. According to a standard Lyapunov extension theorem [44], this demonstrates that the formation error and NN weights matrix estimates error are UUB. Lemma 1 shows that the tracking error vector $\delta(t)$ is bounded. That means all agents follow the leader while maintaining the desired formation. □

If either (56) or (59) holds, ΔV is negative and V decreases. Therefore, (56) and (59) provide practical bounds for the formation error and the NN weight estimation error respectively.

In this chapter, we defined the objective of a leader-followers multi-agent system as reaching a desired relative position with respect to its neighbouring agents while following the leader. We used an RBFNN to approximate the agent's unknown nonlinearity dynamics and designed a distributed control law to realize this objective. A Lyapunov stability analysis was used to prove the UUB of formation error and show that the stated objective is met.

Chapter 4

Attack Detection

In this chapter, we propose a detection method that enable each agent to detect FDI cyber-attacks in its sensor, actuator, and neighbouring communication channel. The designed attack detection method is capable of detecting these attacks separately and a combination of them. We propose a distributed observer for each agent, that is used to form a residual signal. We then design an attack detection threshold by applying the Lyapunov analysis.

4.1 Attack Detection Method

For a distributed cyber-attack detection, each agent has a dedicated NN-based observer that generates the residual signal. We propose the following observer to estimate the i-th agent's states

$$\begin{cases} \hat{x}_i^+ = \hat{W}_i^T \phi_i(x_i) + u_i - G_i(x_i - \hat{x}_i) - \sum\limits_{j \in \mathcal{N}_i} a_{ij}(x_j - x_i - d_{ji}) + b_i(x_l - x_i - d_i), \\ \hat{x}_i(0) = x_i(0), \end{cases} \tag{60}$$

where the diagonal matrix with nonnegative elements $G_i \in \mathbb{R}^{n \times n}$ is the observer gain. The observer dynamics has a neural network part, a standard observer part, and a part that depends on graph topology.

The system reaches its desired formation if no attack is injected into the system. We propose here a method to detect attacks. As a result, the following theorem is given.

Theorem 2. *Consider a multi-agent system that has reached the desired formation (i.e., $||e|| \leq e_M$) in the absence of an attack with the observer (60). If the following condition holds*

$$\bar{\sigma}(G) < \frac{1}{\sqrt{\eta}}, \tag{61}$$

with $\eta = 1 + (1 - \alpha\phi_M^2)^{-1}$, then, the residual $\tilde{\boldsymbol{x}}$ is UUB, with practical bounds given by (72).

Proof. Consider the desired formation is achieved if there is no injected attack to the multi-agent system, (i.e., $||e|| \leq e_M$). Let us define the following Lyapunov function candidate

$$V = V_1 + V_2, \tag{62}$$

where $V_1 = \tilde{\mathbf{x}}^T\tilde{\mathbf{x}}$, and $V_2 = \frac{1}{\alpha}tr(\tilde{W}^T\tilde{W})$. By considering (41), (43), and (45), the first difference of Lyapunov function candidate is given by

$$\Delta V = \Delta V_1 + \Delta V_2. \tag{63}$$

From (35) one can derive ΔV_1 as follows:

$$\begin{aligned}\Delta V_1 =& (\tilde{\mathbf{x}}^+)^T\tilde{\mathbf{x}}^+ - \tilde{\mathbf{x}}^T\tilde{\mathbf{x}} = -\tilde{\mathbf{x}}^T[I - G^TG]\tilde{\mathbf{x}} + 2\tilde{\mathbf{x}}^TG^T\theta + 2\theta^Te + 2\tilde{\mathbf{x}}^TG^T\mu + 2\tilde{\mathbf{x}}G^Te \\ &+ \theta^T\theta + 2\theta^T\mu + \mu^T\mu + e^Te + 2\mu^Te.\end{aligned} \tag{64}$$

By defining $F = diag(F_i)$, we use (36) to obtain ΔV_2

$$\begin{aligned}\Delta V_2 =& \frac{1}{\alpha}tr[(\tilde{W}^+)^T\tilde{W}^+ - \tilde{W}^T\tilde{W}] = \frac{1}{\alpha}tr[-2\alpha\tilde{W}^T\phi\phi^T\tilde{W} - 2\alpha\mu\phi^T\tilde{W} + 2\gamma\tilde{W}^TF\hat{W} \\ &+ \alpha^2\tilde{W}^T\phi\phi^T\phi\phi^T\tilde{W} + 2\alpha^2\tilde{W}\phi\phi^T\phi\mu^T - 2\alpha\tilde{W}\phi\phi^TF\hat{W} + \alpha^2\mu\phi^T\phi\mu^T \\ &- 2\alpha\mu\phi^TF\hat{W} + \hat{W}^TF^TF\hat{W}].\end{aligned} \tag{65}$$

Thus, from (64) and (65) one we can write ΔV as:

$$\begin{aligned}\Delta V =& 2\theta^T G\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T[I - G^T G]\tilde{\mathbf{x}} + 2\theta^T e + 2\tilde{\mathbf{x}}^T G^T e + 2\theta^T \mu \ + \theta^T\theta + 2\tilde{\mathbf{x}}^T G^T \mu + 2\mu^T e \\ &+ \mu^T\mu + e^T e + \frac{1}{\alpha} tr[-2\alpha \tilde{W}^T \phi\phi^T \tilde{W} - 2\alpha\mu\phi^T\tilde{W} + 2\gamma \tilde{W}^T F \hat{W} \\ &+ \alpha^2 \tilde{W}^T \phi\phi^T\phi\phi^T \tilde{W} + 2\alpha^2 \tilde{W}\phi\phi^T\phi\mu^T - 2\alpha\tilde{W}\phi\phi^T F\hat{W} + \alpha^2\mu\phi^T\phi\mu^T \\ &- 2\alpha\mu\phi^T F\hat{W} + \hat{W}^T F^T F \hat{W}].\end{aligned} \tag{66}$$

Reorganizing the terms in (66) yields

$$\begin{aligned}\Delta V \leq& -(1-\alpha\phi^T\phi)\theta^T\theta - 2(1-\alpha\phi^T\phi)\theta^T\mu + 2\theta^T e + 2\theta^T\mu + 2\theta^T G\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T[I - G^T G]\tilde{\mathbf{x}} \\ &+ 2\tilde{\mathbf{x}}^T G^T e + 2\tilde{\mathbf{x}}^T G^T\mu + \mu^T\mu + \alpha\phi^T\phi\mu^T\mu + e^T e + 2\mu^T e + \frac{1}{\alpha} tr[\gamma^2 \hat{W}^T \hat{W} \\ &+ 2\gamma\tilde{W}^T(W - \tilde{W}) + 2\alpha\gamma\mu\phi^T(\tilde{W} - W)].\end{aligned} \tag{67}$$

Completing the squares for θ, and considering Assumption 3, one can write

$$\begin{aligned}\Delta V \leq& -(1-\alpha\phi^T\phi)||\theta - \frac{1}{1-\alpha\phi^T\phi}G\tilde{\mathbf{x}} - \frac{\gamma + \alpha\phi^T\phi}{1-\alpha\phi^T\phi}\mu - \frac{1}{1-\alpha\phi^T\phi}e||^2 \\ &- \tilde{\mathbf{x}}^T\Big[I - (1 + \frac{1}{1-\alpha\phi^T\phi})G^T G\Big]\tilde{\mathbf{x}} + 2(\frac{\gamma+1}{1-\alpha\phi^T\phi})\tilde{\mathbf{x}}^T G^T\mu + 2\eta\tilde{\mathbf{x}}^T G^T e \\ &+ (-2\gamma + \frac{(1+\gamma)^2}{1-\alpha\phi^T\phi})\mu^T\mu + 2(\frac{\gamma+1}{1-\alpha\phi^T\phi})\mu^T e + \eta e^T e \\ &- \frac{1}{\alpha}[\gamma(2-\gamma)||\tilde{W}||_F^2 - 2\gamma(1-\gamma)W_M||\tilde{W}||_F - \gamma^2 W_M^2] + 2\gamma||\phi||W_M\mu_M.\end{aligned} \tag{68}$$

Completing the squares for $\tilde{W}$, considering (68), and using the fact that $||\phi(x)|| \leq \phi_M$, one has

$$\begin{aligned}\Delta V \leq& -(1-\alpha\phi^T\phi)||\theta - \frac{1}{1-\alpha\phi^T\phi}G\tilde{\mathbf{x}} - \frac{\gamma+\alpha\phi^T\phi}{1-\alpha\phi^T\phi}\mu - \frac{1}{1-\alpha\phi^T\phi}e||^2 \\ &- (1-\eta\bar{\sigma}^2(G))\left[||\tilde{\mathbf{x}}||^2 - \frac{1}{1-\eta\bar{\sigma}^2(G)}(2\rho_1||\tilde{\mathbf{x}}|| + \rho_2)\right] \\ &- \frac{1}{\alpha}\gamma(2-\gamma)\left[||\tilde{W}||_F - \frac{1-\gamma}{2-\gamma}W_M\right]^2,\end{aligned} \tag{69}$$

with

$$\rho_1 =(\frac{\gamma+1}{1-\alpha\phi_M^2})\bar{\sigma}(G)\mu_M + \eta\bar{\sigma}(G)e_M, \tag{70}$$

where e_M is derived using the result of Theorem 1 and is given by (56). Moreover,

$$\rho_2 =2\gamma\mu_M W_M + \frac{1}{\alpha}\frac{\gamma}{2-\gamma}W_M^2 + (-2\gamma + \frac{(1+\gamma)^2}{1-\alpha\phi_M^2})\mu_M^2 + 2\mu_M(\frac{\gamma+1}{1-\alpha\phi_M^2})e_M + \eta e_M^2. \tag{71}$$

Then, $\Delta V < 0$ as long as (61) holds, and the quadratic term for $\tilde{\mathbf{x}}$ in (69) is positive which is guaranteed when

$$||\tilde{\mathbf{x}}|| > \frac{\rho_1 + \sqrt{\rho_1^2 + (1-\eta\bar{\sigma}^2(G))\rho_2}}{1-\eta\bar{\sigma}^2(G)} \triangleq \pi. \tag{72}$$

In general ΔV is negative outside a compact set as long as (61) is satisfied and (72) holds. According to a standard Lyapunov extension theorem [44], this demonstrates that the residual is UUB. □

If (72) holds, ΔV is negative and V decreases. Therefore, these provide practical bound π for the residual signal.

Remark 5. *Theorem 2 shows that when the multi-agent system reaches the desired formation, in the absence of attacks, the residual is UUB, with the bound π ([44], Theorem 2.4.6). Thus, under an attack-free condition, one has $||\tilde{\mathbf{x}}|| \leq \pi$. Therefore, we consider π as the attack detection threshold.*

4.2 Attack Detectability Condition

The scalar π is the threshold for the stacked detection residual of a multi-agent system. For an arbitrary vector $\tilde{\mathbf{x}} = [\tilde{x}_1^T, ..., \tilde{x}_N^T]^T$, recall the norm inequality property $||\tilde{x}_i|| \leq ||\tilde{\mathbf{x}}||$. Therefore, the scalar π given by (72) is selected as each agent's residual's threshold. The detection threshold π guarantees that

$$||\tilde{x}_i|| \leq \pi, \tag{73}$$

when no attack affects the agents communications channels. Therefore, each agent local detection algorithm checks whether condition (73) is satisfied.

Let us define the variable s_i, as the overall effect of attacks on residual signal of the i-th agent, which is given by

$$s_i = \kappa_i u_i^a + \lambda_i G_i x_i^a + \hat{f}_i(x_i) - \hat{f}_i(x_i + \lambda_i x_i^a) + \sum_{j \in \mathcal{N}_i} a_{ij}(\Psi_j^i \bar{x}_{ji}^a - \lambda_i x_i^a) + b_i(\Psi_l^i \bar{x}_{li}^a - \lambda_i x_i^a). \tag{74}$$

The following theorem determines the attack detectability condition.

Theorem 3. *Consider a multi-agent system (2) that is modelled by an undirected graph. Select the control law (31), and the NN observer (60). An agent detects FDI attacks if the following inequality holds*

$$||\sum_{l=0}^{k-1} G_i^{k-l-1} s_i|| > \pi + ||\sum_{l=0}^{k-1} G_i^{k-l-1}(\tilde{W}_i^T \phi(x_i) + e_i + \epsilon_i + w_i)||. \tag{75}$$

Proof. Let us consider $\bar{u}_i$ as the control input of i-th agent when the agent is under sensor and neighboring channels' attacks. From (30), $\bar{u}_i$ can be written as

$$\begin{aligned} \bar{u}_i = &- \hat{f}_i(x_i + \lambda_i x_i^a) + x_i + \lambda_i x_i^a + k_i \sum_{j \in \mathcal{N}_i} a_{ij}(\bar{x}_j + \Psi_j^i \bar{x}_{ji}^a \\ &- x_i - \lambda_i x_i^a - d_{ji}) + b_i(\bar{x}_l + \Psi_l^i \bar{x}_{li}^a - x_i - \lambda_i x_i^a - d_i). \end{aligned} \tag{76}$$

From (2), (6), and (76) the system dynamics under sensor, actuator, and neighboring channels attacks can be written as

$$x_i^+ = f(x_i) + \bar{u}_i + \kappa_i u_i^a + w_i. \tag{77}$$

From (8), (11), and (60) the observer dynamics under attack can be written as

$$\begin{aligned}\hat{x}_i^+ =& f(x_i + \lambda_i x_i^a) + \bar{u}_i - G_i(x_i + \lambda_i x_i^a) - \sum_{j \in \mathcal{N}_i} a_{ij}(\bar{x}_j + \Psi_j^i \bar{x}_{ji}^a \\ & - x_i - \lambda_i x_i^a - d_{ji}) + b_i(\bar{x}_l + \Psi_l^i \bar{x}_{li}^a - x_i - \lambda_i x_i^a - d_i).\end{aligned} \tag{78}$$

Considering the detection residual, and from (77) and (78), the residual dynamics can be derived as

$$\tilde{x}_i^+ = G_i \tilde{x}_i + \tilde{W}_i^T \phi(x_i) + \epsilon_i + w_i + e_i + s_i. \tag{79}$$

The response of residual signal (34) under attack, when $x_i(0) = \hat{x}_i(0)$, can be written as

$$\tilde{x}_i = \sum_{l=0}^{k-1} G_i^{k-l-1}(\tilde{W}_i^T \phi(x_i) + e_i + \epsilon_i + w_i + s_i). \tag{80}$$

Using triangle inequality, one has

$$||\tilde{x}_i|| \geq || \sum_{l=0}^{k-1} G_i^{k-l-1} s_i|| - || \sum_{l=0}^{k-1} G_i^{k-l-1}(\tilde{W}_i^T \phi(x_i) + e_i + \epsilon_i + w_i)||. \tag{81}$$

If the following condition is satisfied

$$|| \sum_{l=0}^{k-1} G_i^{k-l-1} s_i|| - || \sum_{l=0}^{k-1} G_i^{k-l-1}(\tilde{W}_i^T \phi(x_i) + e_i + \epsilon_i + w_i)|| > \pi, \tag{82}$$

then, the residual exceeds the threshold, i.e., $||\tilde{x}_i|| > \pi$, which implies the following attack detectability condition

$$|| \sum_{l=0}^{k-1} G_i^{k-l-1} s_i|| > \pi + || \sum_{l=0}^{k-1} G_i^{k-l-1}(\tilde{W}_i^T \phi(x_i) + e_i + \epsilon_i + w_i)||. \tag{83}$$

This completes the proof. □

Note that in practice, it is sufficient that

$$||\tilde{x}_i|| > \pi, \tag{84}$$

for agent i to affirm that an attack has occurred in its communication channels.

Remark 6. *The overall attack effect s_i can be due to any type of attacks mentioned earlier including some combination of them.*

In this chapter, we developed a distributed attack detection method for detecting FDI attacks in sensor, actuator, and neighbouring communication channels of the discrete-time multi-agent systems with unknown nonlinearity dynamics. We used the Lyapunov stability analysis to prove the UUB of attack detection residual when the system reached the desired formation and provided a practical attack detection threshold. Moreover, an attack detectability condition is provided to determine the detection of attacks.

Chapter 5

Numerical Simulation and Implementation

In this section, we provide three examples of nonlinear multi-agent systems control and being under FDI attack. In the first example, we consider four different attack scenarios on the discrete-time nonlinear multi-agent system with unknown dynamics in the MATLAB environment. We evaluate our proposed control law in the attack-free scenario and attack detection method on under-attack scenarios. In the second example, we consider a multi-agent system of UGVs and implement two different attack scenarios on neighboring communication channels of the multi-agent system. We use the MATLAB environment to validate the UGVs detection systems and their behavior under attack conditions. In the third example, we used the CoppeliaSim simulation environment to emulate a real-life scenario. We implemented the first scenario of the second example in the CoppeliaSim and investigated the behavior of UGVs under attack in this environment.

5.1 Simulation Results

We conducted numerical simulations to evaluate the performance of the proposed attack detection method.

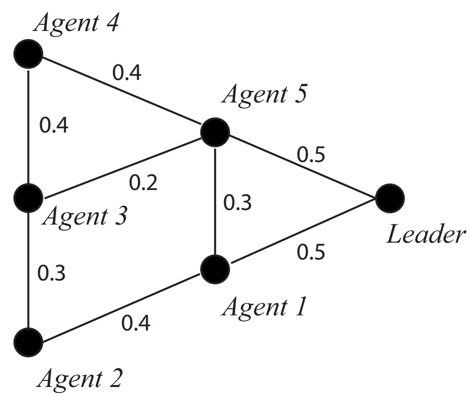


Figure 5: Communication topology and formation shape.

Example: Consider a multi-agent system with the leader and five agents as followers, modeled by the undirected graph as shown in Fig. 5. We consider the dynamics of each agent as [44]:

$$x_i^+ = \begin{bmatrix} \frac{\beta_i x_{i2}}{1+x_{i2}^2} \\ \frac{\beta_i x_{i1}}{1+x_{i2}^2} \end{bmatrix} + u_i + w_i, \qquad i \in \{1, 2, .., 5\}, \tag{85}$$

where $x_i = [x_{i1} \ x_{i2}]^T$. Variables x_{i1} and x_{i2} are the position in x and y coordinates, respectively. We set $\beta_1 = 0.3$, $\beta_2 = 0.5$, $\beta_3 = 0.6$, $\beta_4 = 0.9$, and $\beta_5 = 1$. In this simulation, we select the control parameters $k_i = 0.4I_2$, and the observer gain as $G_i = 0.1I_2$. The initial conditions for the follower agents are: $x_1(0) = [1, -6]^T$, $x_2(0) = [1, -7]^T$, and $x_3(0) = [1.5, -6]^T$, $x_4(0) = [0.5, -5]^T$, $x_5(0) = [1, -5]^T$. The disturbances in the dynamics of each agent are given by $w_1 = [0.03, 0.01]^T cos(3t), w_2 = [0.02, 0.02]^T sin(3t), w_3 = [0.01, 0.03]^T cos(2t), w_4 = [0.02, 0.03]^T cos(4t), w_5 = [0.02, 0.01]^T sin(3t)$.

RBFNN is selected with 9 neurons, centers q_j evenly spaced for each agent on $[-1, 2] \times [-7, -4]$ grid, and $F_i = 0.4I_9$, $\alpha = 0.01$.

The desired formation and communication topology is shown in Fig. 5, and the desired position of each agent with respect to the leader is given by

$$\begin{aligned} &d_1 = [-0.5, \ -0.5]^T, d_2 = [-1, \ -1]^T, d_3 = [-1, \ 0]^T, d_4 = [-1, \ 1]^T, \\ &d_5 = [-0.5, \ 0.5]^T. \end{aligned} \tag{86}$$

In the example, we first investigate the performance of the proposed neural network-based control law in an attack-free scenario for the multi-agent system (85). Then, we provide four attack scenarios for the multi-agent systems to demonstrate the effectiveness of the proposed attack detection method. We consider the leader trajectory as $x_{l1} = cos(t/15) + 2$, $x_{l2} = sin(t/15) - 6$.

In the attack-free scenario, we validate that the multi-agent system reaches and maintains the desired formation with the proposed control law (31). Fig. 6 shows the system trajectory along x and y coordinate in attack-free scenario, and Fig. 7 shows each agent formation error which shows that it is UUB. Fig. 8 shows the Frobenius norm of the neural network weights matrix.

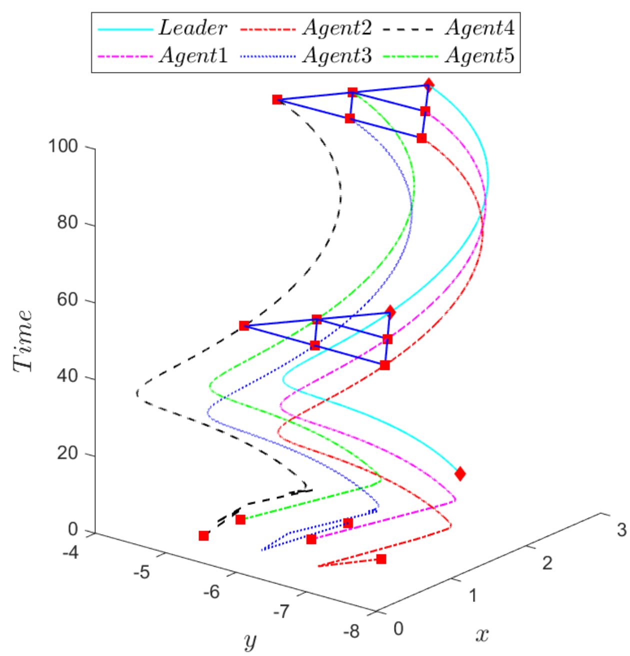


Figure 6: Multi-agent system formation in attack-free case.

Now, we consider four attack scenarios. For the first attack scenario, we consider the attacker injects false data on the agent 3 actuator channel

$$u_3^c = u_3 + \kappa_3 u_3^a, \tag{87}$$

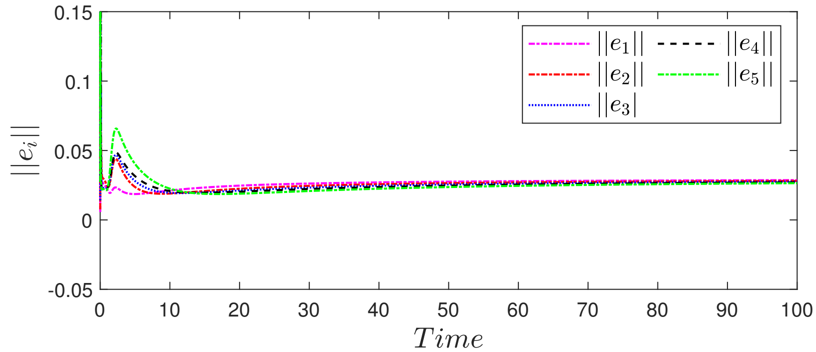


Figure 7: Formation error for all agents.

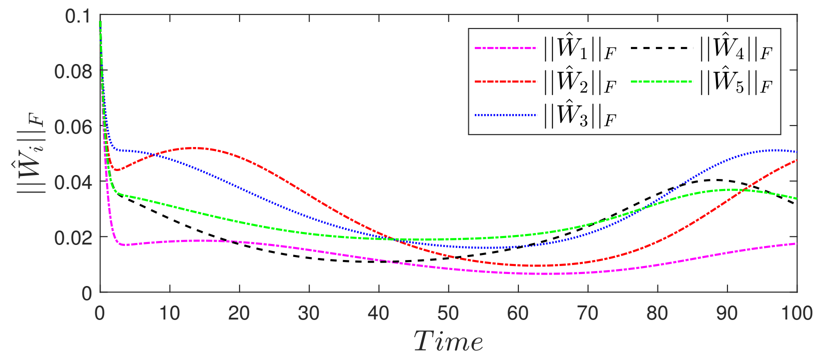


Figure 8: Neural network matrix weights Frobenius norm for all agents.

where $u_3^a = (1 - \exp(50 - t))[0.7, 0.7]^T$, and

$$\kappa_3 = \begin{cases} 1, & \text{for } 50 \leq t \leq 60 \\ 0, & \text{otherwise.} \end{cases} \tag{88}$$

In Fig. 9, the effect of attack can be seen on the system formation. The residual signal of agent 3, which is $||\tilde{x}_3||$ with the threshold π, is shown in Fig. 10. The attack increases the residual signal of agent 3 and it exceeds the threshold.

For the second scenario, we consider that the attacker injects false data on the sensor channel of agent 5

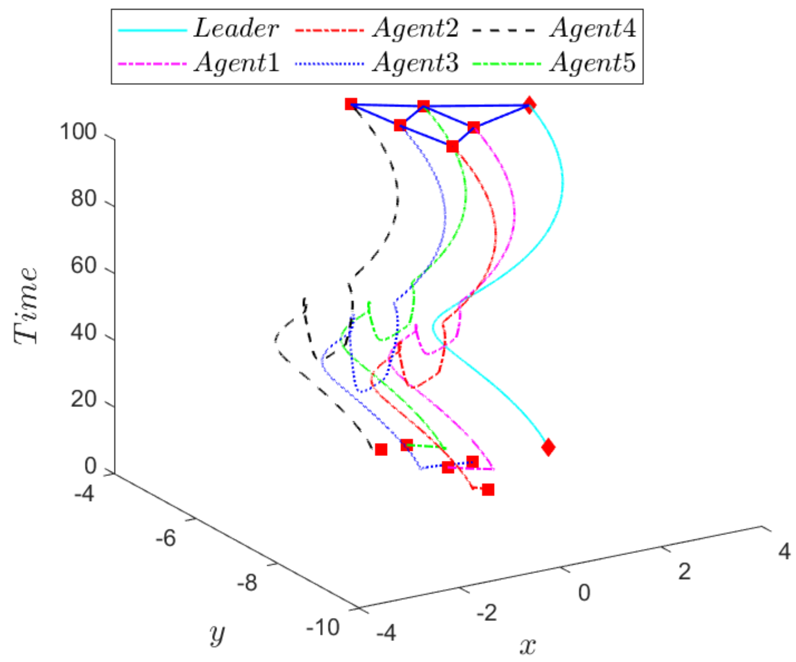


Figure 9: System trajectory under attack with attack on the actuator channel of agent 3.

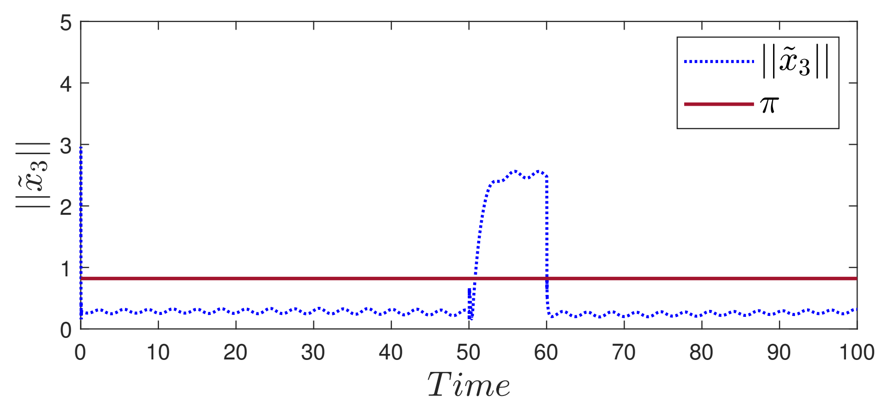


Figure 10: Residual signal of agent 3 in case 1: attack on the actuator channel of agent 3.

$$x_5^c = x_5 + \lambda_5 x_5^a, \tag{89}$$

where $x_5^a = [-2, -2.5]^T$, and

$$\lambda_5 = \begin{cases} 1, & \text{for } 50 \leq t \leq 60 \\ 0, & \text{otherwise.} \end{cases} \tag{90}$$

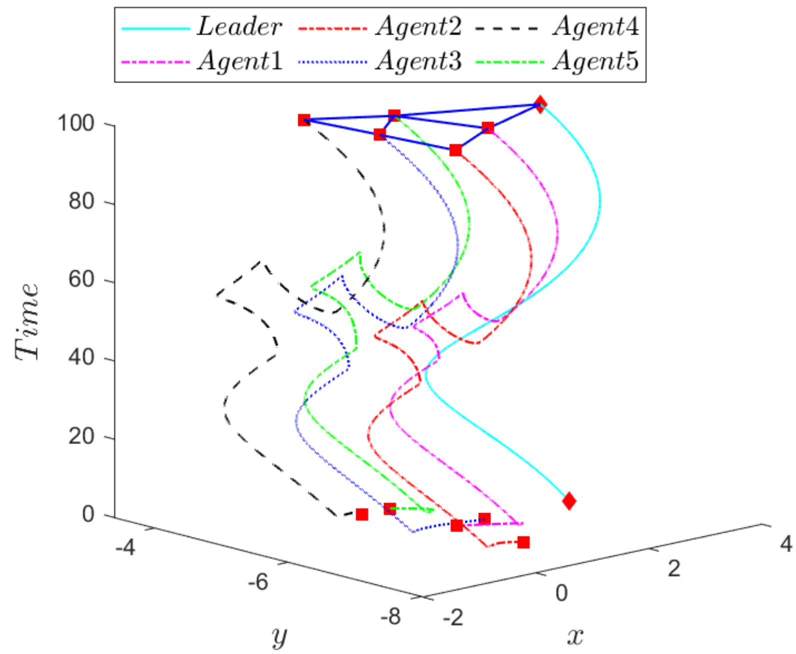


Figure 11: System trajectory under attack. Attack on the sensor channel of agent 5.

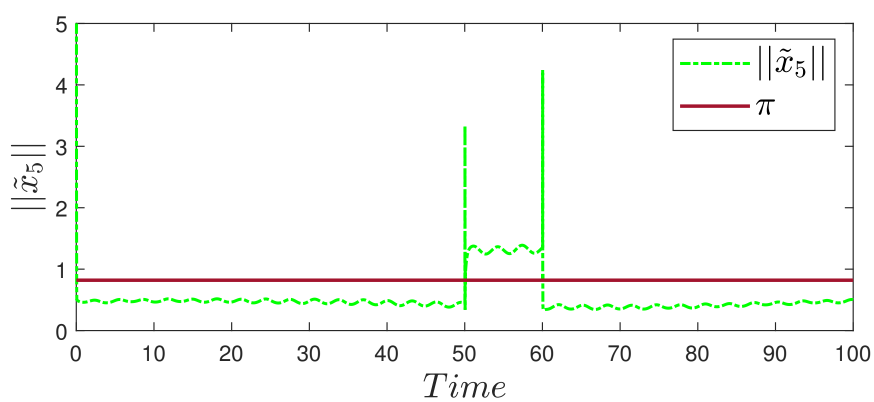


Figure 12: Residual signal of agent 5 in case 2: attack on sensor channel of agent 5.

In Fig. 11, the attack's effect on the agent 5 sensor channel can be seen. The residual signal of agent 5, $||\tilde{x}_5||$, increases during the attack and reveals the attack (Fig. 12).

For the third scenario, we consider the attacker performs the FDI attack on the neighboring communication channels of agent 4. We consider that the attacker injects the false data into the communication channel between agent 4 and agent 5, and the attacker changes the data that agent 4 receives from agent 5

$$\bar{x}^c_{54} = \bar{x}_5 + \Psi^4_5 \bar{x}^a_{54}, \tag{91}$$

where $\bar{x}^a_{54} = [-2, 2cos(2t)]^T$, and

$$\Psi^4_5 = \begin{cases} 1 & \text{for } 50 \leq t \leq 60 \\ 0 & \text{otherwise.} \end{cases} \tag{92}$$

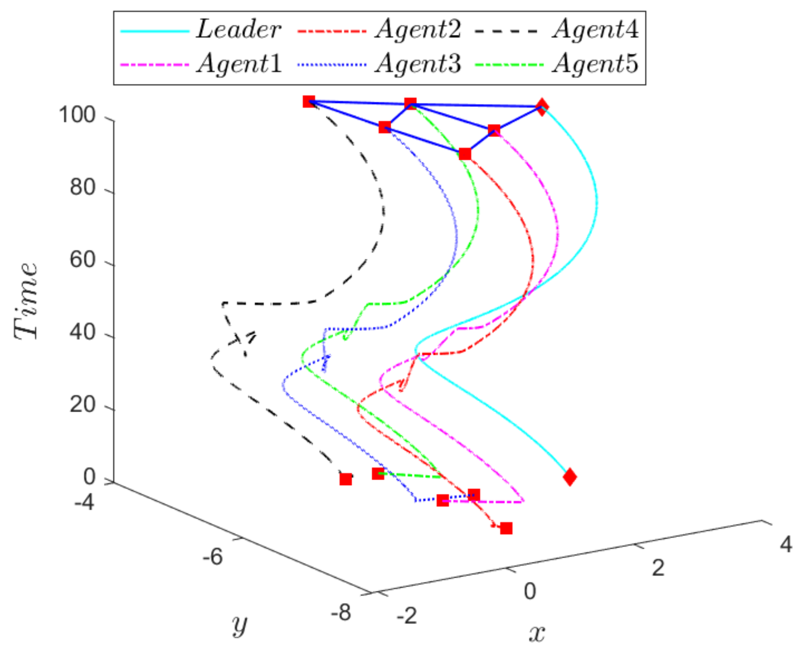


Figure 13: System trajectory under attack. Attack on the neighbouring communication channel between agent 4 and agent 5.

Fig. 13 shows the effect of the attack on the neighboring communication channel of agent 4, and the residual signal of agent 4 is shown in Fig. 14, which illustrates the detection of the attack.

For the last scenario, we consider that the attacker breaches all channels and distorts their data. The attacker injects the false data in the actuator channel of agent 3, in the sensor channel of agent 5, and changes the data that agent 4 receives from agent

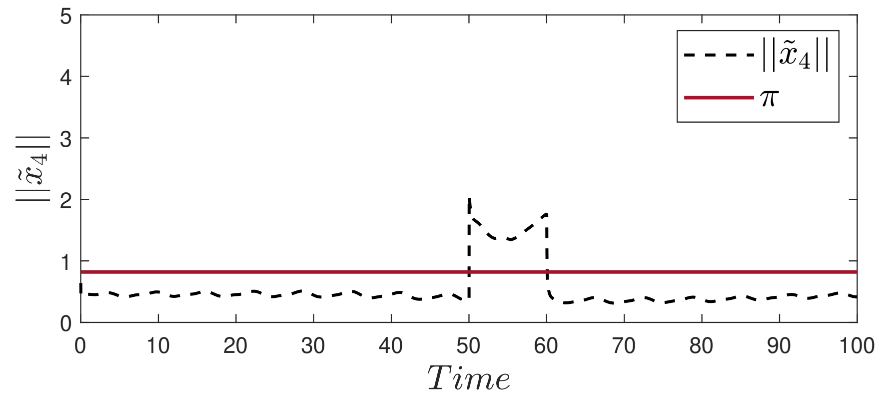


Figure 14: Residual signal of agent 4 in case 3: attack on the neighbouring communication channel of agent 4.

5. In fourth attack scenario, we simulated the multi-agent system (85) with previously mentioned attack signals. In Fig. 15, the effect of mentioned attacks on the performance of multi-agent system (85) is shown. Since agent 3, agent 4, and agent 5 are under attack, the residual signals of these agents are shown in Fig. 16, indicating attacks' detection. Consequently, all the attacked agents can detect the attack by the proposed detection method as all the attacked agents' residuals exceed the threshold during the attack, as shown in Fig. 16.

Example 2: In this example, we consider a multi-agent system of UGVs and implement two different attack scenarios. We consider four UGVs. One is considered the leader, and the others are the follower agents, modelled by single-integrator dynamics, similar to [23].

$$x_i^+ = x_i + sin(x_i) + u_i, \tag{93}$$

where $x_i = [x_{i1} \ x_{i2}]^T$. Variables x_{i1} and x_{i2} are the position in x and y coordinates, respectively and $u_i \in \mathbb{R}^2$ is the control input.

By considering $f_i(x_i) = x_i + sin(x_i)$ as the unknown nonlinearity, the UGVs dynamics can be expressed as follows:

$$x_i^+ = f_i(x_i) + u_i. \tag{94}$$

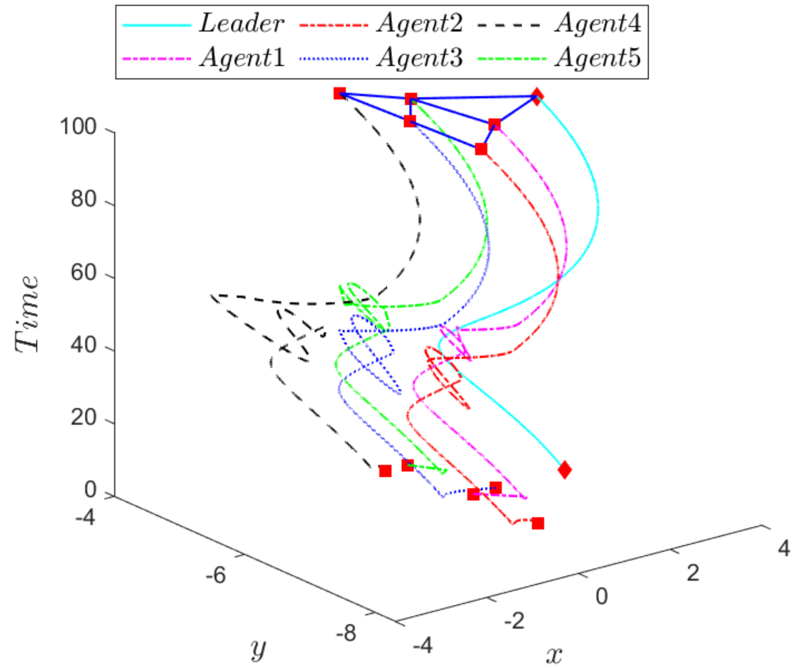


Figure 15: System trajectory when agent 3, 4, and 5 are under attack; combination of the three different types of attacks.

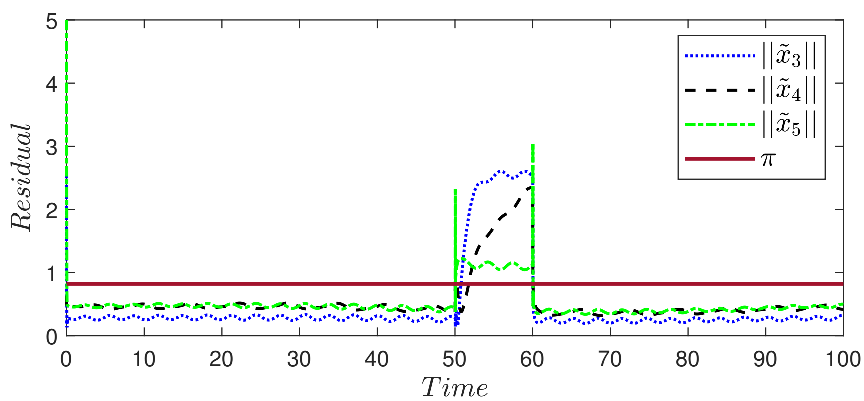


Figure 16: Residual signal of under attacked agents.

We used the controller (30), and observer (34) for each UGV and used the proposed attack detection method for detecting FDI attack in the communication channels of each agent.

In this example, we consider the control parameters $k_i = 0.5I_2$, and the observer gain as $G_i = 0.15I_2$. The initial conditions for the follower agents are: $x_1(0) = [1.5, -7]^T$,

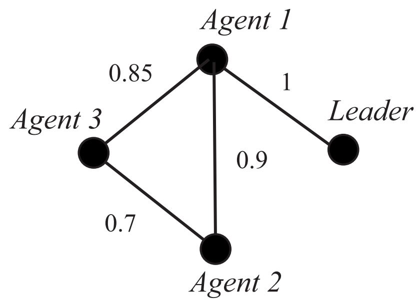


Figure 17: Communication topology and formation shape.

$x_2(0) = [1, -7.5]^T$, and $x_3(0) = [1.5, -7.5]^T$. RBFNN is selected with 9 neurons, centers q_j evenly spaced on $[1, 4] \times [-8, -2]$ for each agent, and $F_i = 0.35I_9$, $\alpha = 0.01$.

We consider the following leader trajectory $x_{l1} = 0.03t + 3$, and $x_{l2} = 0.03t - 3$. The desired formation and communication topology is shown in Fig. 17, and the desired position of each agent with respect to the leader is given by

$$d_1 = [-1,\ 1]^T, d_2 = [-2,\ 0]^T, d_3 = [-1,\ 1]^T. \tag{95}$$

Fig. 18a shows the system in the attack-free case, which illustrates reaching the desired formation. In the following, we consider two attack scenarios on neighboring communication channels of agents. In the first attack scenario, we consider that the leader signal is under attack, and the attack changes the leader signal by injecting false data on the communication channel between the leader and UGV 1. As a result, UGV 1 receives corrupted data from the leader.We consider the following leader trajectory $x_{l1} = 0.03t + 3$, and $x_{l2} = 0.03t - 3$. The desired formation and communication topology is shown in Fig. 17, and the desired position of each agent with respect to the leader is given by

$$\bar{x}^c_{l1} = \bar{x}_l + \Psi^1_l \bar{x}^a_{l1}, \tag{96}$$

where $\bar{x}^a_{l1} = [-1.4sin(t/6), cos(t/5)]^T$, and

$$\Psi^1_l = \begin{cases} 1 & \text{for } 40 \leq t \leq 50 \\ 0 & \text{otherwise.} \end{cases} \tag{97}$$

Fig. 18b shows the effect of this attack on the multi-agent systems. Since all agents receive the leader signal directly or indirectly, the attack affects all the agents and changes the desired formation. Since the leader is the neighbour of agent one, the neighbouring communication channel of UGV 1 is under attack. Fig. 19a shows the residual signal of UGV 1, which indicates the detection of the attack.

In the second scenario, we consider that the neighbouring channels of agents 2 and 3 are under FDI attacks, and the attacker changes the sensor data that UGV 2 and UGV 3 receive from UGV 1 as follows:

$$\begin{cases} \bar{x}^c_{12} = \bar{x}_1 + \Psi^2_1 \bar{x}^a_{12}, \\ \bar{x}^c_{13} = \bar{x}_1 + \Psi^3_1 \bar{x}^a_{13}, \end{cases} \tag{98}$$

where $\bar{x}^a_{12} = [-1.2sin(t), cos(t/15)]^T$, $\bar{x}^a_{13} = [-0.5, cos(t/3)]^T$, and

$$\Psi^3_1 = \Psi^2_1 = \begin{cases} 1, & \text{for } 40 \leq t \leq 50 \\ 0, & \text{otherwise.} \end{cases} \tag{99}$$

In this case, the neighboring communication channels of both UGVs 2 and 3 are under attack. The effect of this attack on the multi-agent system can be seen in Fig. 18c. The residual of UGV 2 and UGV 3 are displayed on Fig. 19b which are showing the detection of attack on both UGVs.

Example 3: In this example, we implement the first scenario of the *Example 2* in the Coppeliasim environment. The CoppeliaSim (formerly V-REP) is a virtual robot experimentation platform used to validate this thesis's theoretical results. The simulation

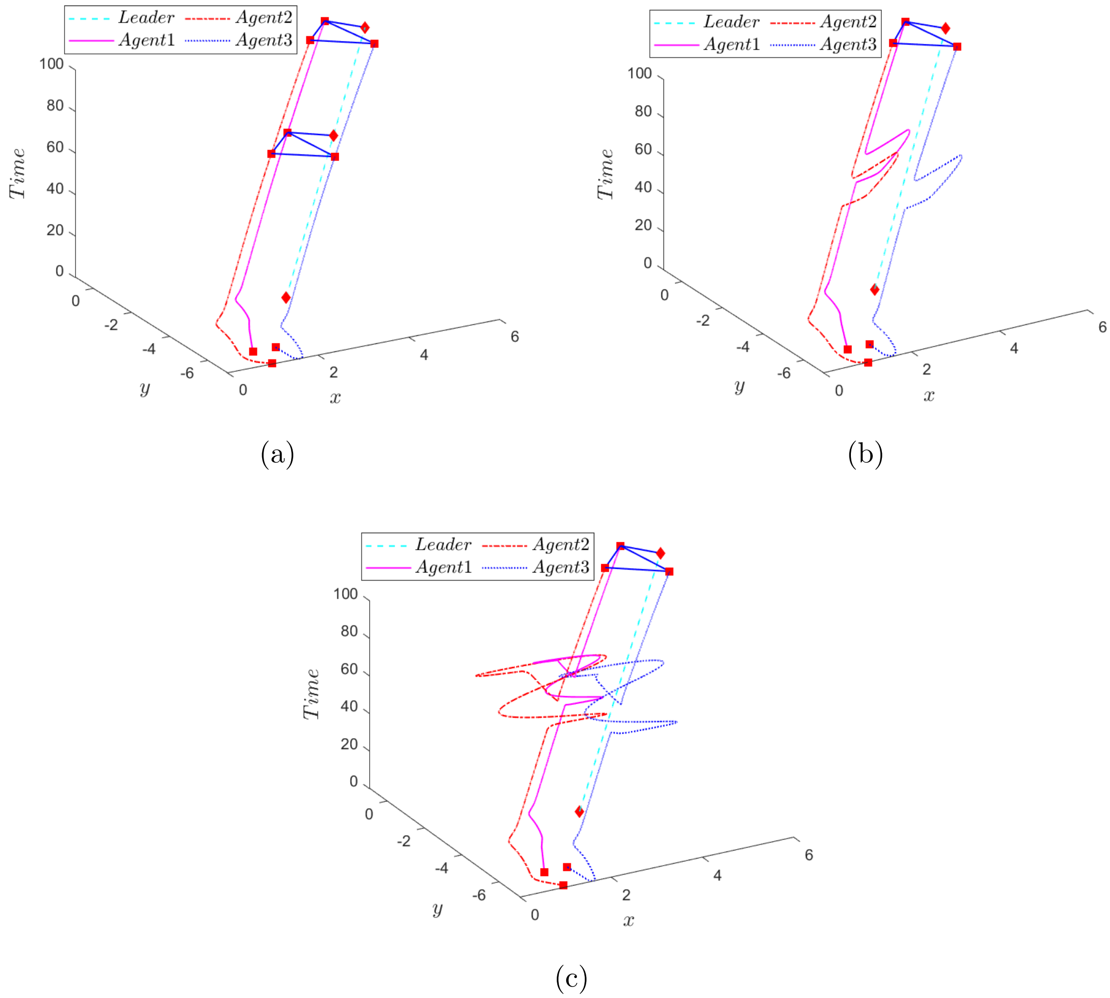


Figure 18: System trajectory. (a) system trajectory in attack-free case. (b) System trajectory under attack in the first scenario when agent 1 neighboring channel is under attack. (c) System trajectory under attack in the second scenario when agents 2 and 3 neighboring channels are under attack.

software CoppeliaSim enables us to have a real robot dynamics emulation that is close to the real robot platform. Four mobile robots (Pioneer P3-DX) are considered, one as the leader and the others are the follower robots (see Fig. 20). They are modeled by the dynamics (93). The control parameters are the same as that in *example 2*. In this attack scenario, the attacker injects the false data to the neighbouring communication between the leader and robot 1 to change the system formation to its desired formation (see Fig. 21).

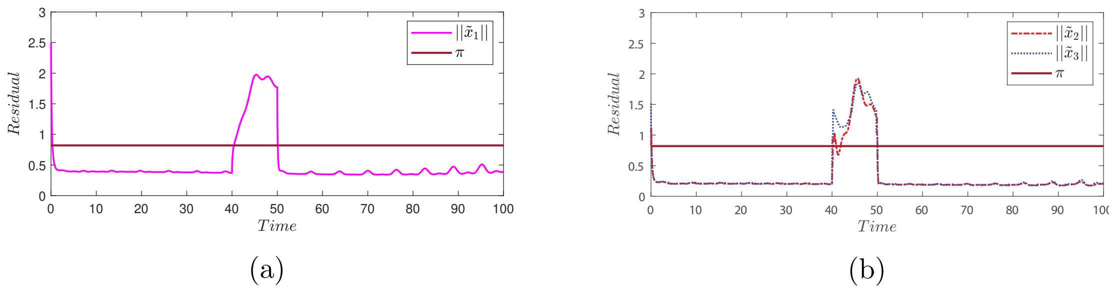


Figure 19: Residual signal. (a) Residual signal of agent 1 in the first scenario. (b) Residual signal of agents 2 and 3 in the second scenario.

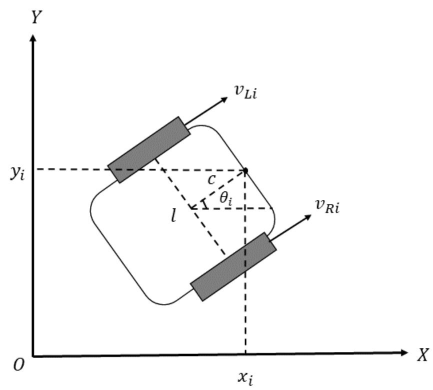


Figure 20: Schematic of Pioneer P3-DX robot.

The control input u_i in (93) is allocated to the left and right wheels using the following transfer matrix [60].

$$\begin{bmatrix} v_{Ri} \\ v_{Li} \end{bmatrix} = \begin{bmatrix} sin\theta_i + \frac{l}{2c}cos\theta_i & sin\theta_i - \frac{l}{2c}cos\theta_i \\ sin\theta_i - \frac{l}{2c}cos\theta_i & sin\theta_i + \frac{l}{2c}cos\theta_i \end{bmatrix} u_i, \tag{100}$$

where l is the distance between the two driving wheels, and we consider $c = l/2$. We consider the following leader trajectory $x_{l1} = 0.1t$, and $x_{l2} = -2$. The desired formation is the same as Fig. 21, and the desired position of each agent with respect to the leader is same as (95). Fig. 22 shows the initial positions of robots in the CoppeliaSim simulation environment. In the attack scenario, the neighbouring communication channel between leader and robot 1 is under FDI attack. In Fig. 23a robots are shown when they reach the

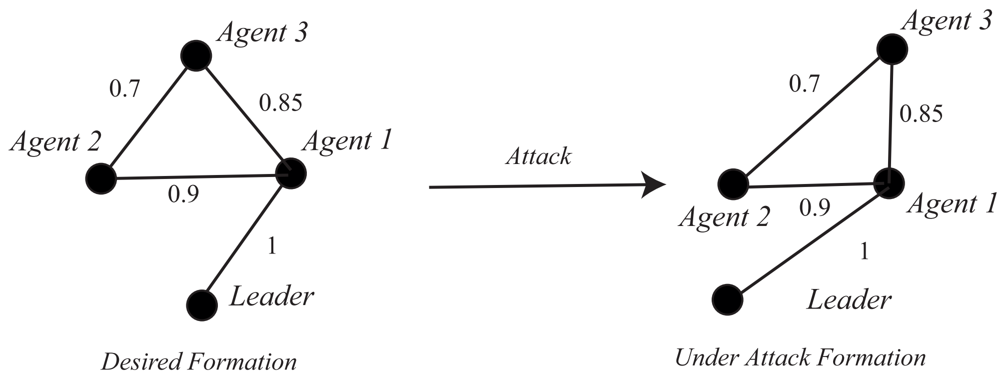


Figure 21: Example 3 desired formation shape and under attack formation.

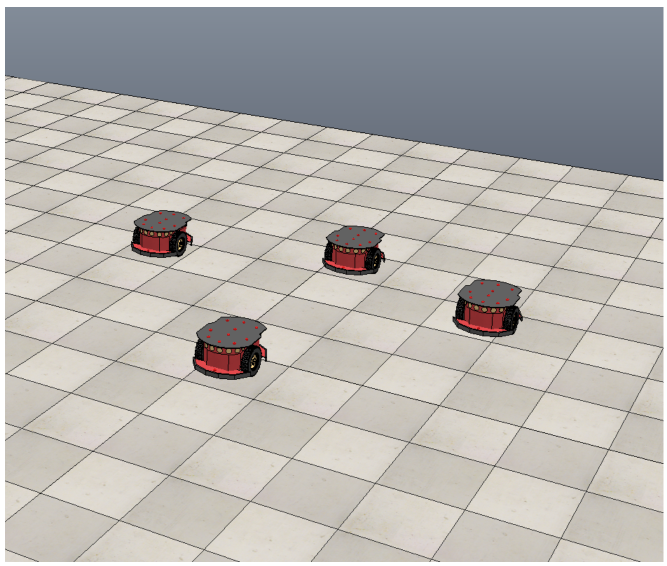
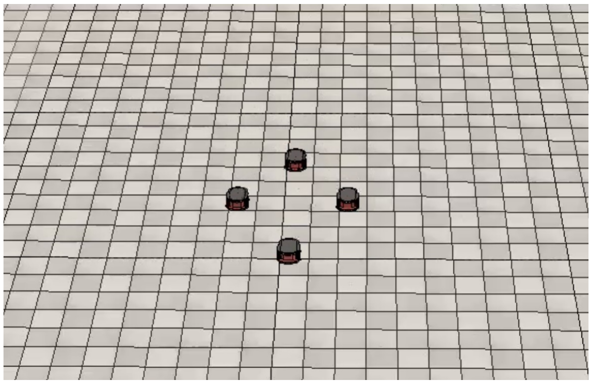


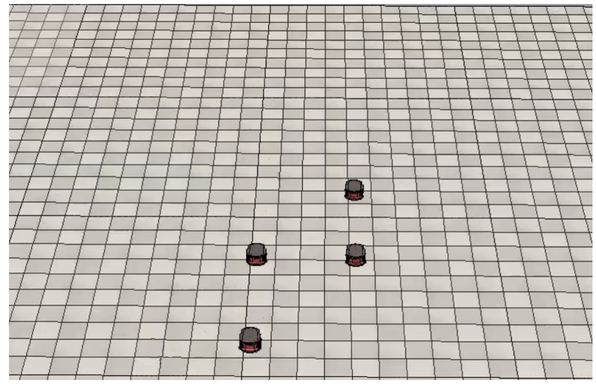
Figure 22: Four Pioneer P3-DX robots in the CoppeliaSim environment.

desired formation in the Coppeliasim environment, and Fig. 23b shows the multi-agent system under attack, indicating attacker changes the formation to its desired formation. Fig. 24 shows the generated residual signal of robots 1, 2 and 3 in the CoppeliaSim environment, which shows the detection of the attack. Since the formation is changed, all the agents detect the attack. The video of this simulation can be found here [1], which shows the disordering of the desired formation under attack and maintaining the formation before the attack and forming the formation again when the attack is over.

[1]https://www.dropbox.com/s/tgdcqt5nyalpe3d/thesis.mp4?dl=0

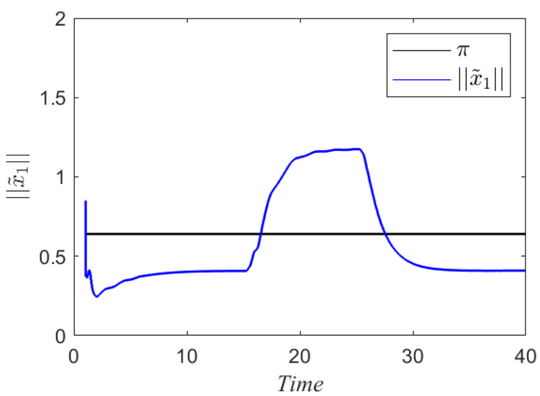


(a)

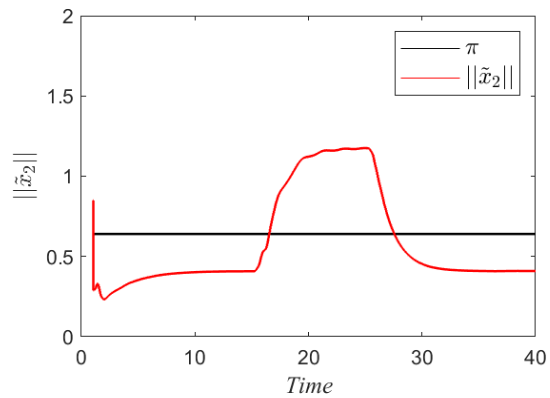


(b)

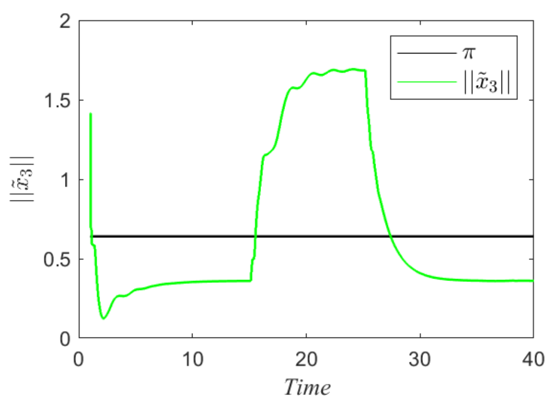
Figure 23: CoppeliaSim environment. (a) The multi-agent system reached to the desired formation. (b) The multi-agent system under attack.



(a)



(b)



(c)

Figure 24: Residual signals. (a) Residual signal of robot 1. (b) Residual signal of robot 2.(c) Residual signal of robot 3.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

We developed a distributed cyber-attack detection method for a class of discrete-time, nonlinear, heterogeneous, multi-agent systems in a formation control setting. We investigated the false data injection attack on agents' communication channels and proposed an NN-based observer to form a residual signal for each agent to detect attacks on its actuator, sensor, and neighboring communication channels. The Lyapunov stability analysis was used to prove the UUB of residual signal and obtain a practical bound as an attack detectability threshold. Moreover, we proposed a formation control for a class of discrete-time, nonlinear multi-agent systems and proved the UUB of the formation error. The simulation results were presented different types of attacks on the sensor, actuator, and neighboring communication channels of agents, as well as a combination of them. The proposed detection method cannot identify the type of attack. Further work is required to identify the type of attack after the attack has been detected.

6.2 Future Work

Some suggestions for future research in this area are outlined below:

- The proposed attack detection architecture lacks identification of the type of attacks. For future work, extending the attack detection method to classify the attack on the sensor, actuator, and neighboring communication channels of multi-agent systems can be considered.
- Design an attack mitigation action to decrease the effect of attacks on the system or develop a controller for attack compensation once the attack is detected by the proposed detection method.
- Develop the result of this research for the multi-agent systems modelled with the directed graph topology.
- The proposed attack detection threshold is conservative; designing a less conservative threshold can be another extension to this work.
- Develop a method to distinguish between the system fault and cyber-attack on communication channels.

References

[1] E. A. Lee, "Cyber physical systems: Design challenges," in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. IEEE, 2008, pp. 363–369.

[2] G. Xiong, F. Zhu, X. Liu, X. Dong, W. Huang, S. Chen, and K. Zhao, "Cyber-physical-social system in intelligent transportation," *IEEE/CAA Journal of Automatica Sinica*, vol. 2, no. 3, pp. 320–333, 2015.

[3] B. Chen, Z. Yang, S. Huang, X. Du, Z. Cui, J. Bhimani, X. Xie, and N. Mi, "Cyber-physical system enabled nearby traffic flow modelling for autonomous vehicles," in *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*. IEEE, 2017, pp. 1–6.

[4] C. A. Macana, N. Quijano, and E. Mojica-Nava, "A survey on cyber physical energy systems and their applications on smart grids," in *2011 IEEE PES conference on innovative smart grid technologies Latin America (ISGT LA)*. IEEE, 2011, pp. 1–7.

[5] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[6] R. Baheti and H. Gill, "Cyber-physical systems," *The impact of control technology*, vol. 12, no. 1, pp. 161–166, 2011.

[7] F. Miao, Q. Zhu, M. Pajic, and G. J. Pappas, "Coding schemes for securing cyber-physical systems against stealthy data injection attacks," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 106–117, 2016.

[8] F. Farivar, M. S. Haghighi, A. Jolfaei, and M. Alazab, "Artificial intelligence for detection, estimation, and compensation of malicious attacks in nonlinear cyber-physical systems and industrial iot," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2716–2725, 2019.

[9] X. Jin, W. M. Haddad, and T. Yucelen, "An adaptive control architecture for mitigating sensor and actuator attacks in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 6058–6064, 2017.

[10] W. Lucia, K. Gheitasi, and M. Ghaderi, "Setpoint attack detection in cyber-physical systems," *IEEE Transactions on Automatic Control*, 2020.

[11] S. Kriaa, M. Bouissou, and L. Piètre-Cambacédès, "Modeling the stuxnet attack with bdmp: Towards more formal risk assessments," in *2012 7th International Conference on Risks and Security of Internet and Systems (CRiSIS)*. IEEE, 2012, pp. 1–8.

[12] J. Weiss, "Aurora generator test," *Handbook of SCADA/Control Systems Security*, vol. 107, 2016.

[13] D. U. Case, "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, 2016.

[14] A. Teixeira, H. Sandberg, and K. H. Johansson, "Networked control systems under cyber attacks with applications to power networks," in *Proceedings of the 2010 American Control Conference*. IEEE, 2010, pp. 3690–3696.

[15] F. Boem, A. J. Gallo, G. Ferrari-Trecate, and T. Parisini, "A distributed attack detection method for multi-agent systems governed by consensus-based control," in

2017 IEEE 56th Annual Conference on Decision and Control (CDC). IEEE, 2017, pp. 5961–5966.

[16] W. Zhang, S. Mao, J. Huang, L. Kocarev, and Y. Tang, "Data-driven resilient control for linear discrete-time multi-agent networks under unconfined cyber-attacks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 2, pp. 776–785, 2021.

[17] R. Anguluri, V. Katewa, and F. Pasqualetti, "Attack detection in stochastic interconnected systems: Centralized vs decentralized detectors," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 4541–4546.

[18] F. Arrichiello, A. Marino, and F. Pierri, "Observer-based decentralized fault detection and isolation strategy for networked multirobot systems," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1465–1476, 2015.

[19] C. Keliris, M. M. Polycarpou, and T. Parisini, "A distributed fault detection filtering approach for a class of interconnected continuous-time nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 8, pp. 2032–2047, 2013.

[20] D. Ye and X. Yang, "Distributed event-triggered consensus for nonlinear multi-agent systems subject to cyber attacks," *Information Sciences*, vol. 473, pp. 178–189, 2019.

[21] A. Khazraei, H. Kebriaei, and F. R. Salmasi, "Replay attack detection in a multi agent system using stability analysis and loss effective watermarking," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 4778–4783.

[22] Z.-H. Pang, G.-P. Liu, D. Zhou, F. Hou, and D. Sun, "Two-channel false data injection attacks against output tracking control of networked systems," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 5, pp. 3242–3251, 2016.

[23] A. Barboni, H. Rezaee, F. Boem, and T. Parisini, "Detection of covert cyber-attacks in interconnected systems: A distributed model-based approach," *IEEE Transactions on Automatic Control*, 2020.

[24] X. Huang and J. Dong, "Reliable leader-to-follower formation control of multiagent systems under communication quantization and attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 89–99, 2019.

[25] Y. Wu and X. He, "Secure consensus control for multiagent systems with attacks and communication delays," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 1, pp. 136–142, 2017.

[26] Y. Wan, J. Cao, G. Chen, and W. Huang, "Distributed observer-based cyber-security control of complex dynamical networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 11, pp. 2966–2975, 2017.

[27] A. Barboni and T. Parisini, "Towards distributed accommodation of covert attacks in interconnected systems," in *2020 59th IEEE Conference on Decision and Control (CDC).* IEEE, 2020, pp. 5731–5736.

[28] G. D. L. Torre and T. Yucelen, "Adaptive architectures for resilient control of networked multiagent systems in the presence of misbehaving agents," *International Journal of Control*, vol. 91, no. 3, pp. 495–507, 2018.

[29] S. Sundaram and B. Gharesifard, "Consensus-based distributed optimization with malicious nodes," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton).* IEEE, 2015, pp. 244–249.

[30] M. Taheri, K. Khorasani, I. Shames, and N. Meskin, "Undetectable cyber attacks on communication links in multi-agent cyber-physical systems," in *2020 59th IEEE Conference on Decision and Control (CDC).* IEEE, 2020, pp. 3764–3771.

[31] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[32] F. Xiao, L. Wang, J. Chen, and Y. Gao, "Finite-time formation control for multi-agent systems," *Automatica*, vol. 45, no. 11, pp. 2605–2611, 2009.

[33] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[34] R. S. Smith, "Covert misappropriation of networked control systems: Presenting a feedback structure," *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 82–92, 2015.

[35] A. Mousavi, K. Aryankia, and R. R. Selmic, "Cyber-attack detection in discrete-time nonlinear multi-agent systems using neural networks," to appear in 2021 IEEE Conference on Control Technology and Applications (CCTA). IEEE, 2021.

[36] R. R. Selmic, A. Mousavi, and K. Aryankia, "A distributed FDI cyber-attack detection in discrete-time nonlinear multi-agent systems using neural networks," *submitted to European Journal of Control*, 2021.

[37] N. Monshizadeh, H. L. Trentelman, and M. K. Camlibel, "Projection-based model reduction of multi-agent systems using graph partitions," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 2, pp. 145–154, 2014.

[38] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[39] F. Mehdifar, C. P. Bechlioulis, F. Hashemzadeh, and M. Baradarannia, "Prescribed performance distance-based formation control of multi-agent systems," *Automatica*, vol. 119, p. 109086, 2020.

[40] K.-K. Oh and H.-S. Ahn, "Formation control and network localization via orientation alignment," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 540–545, 2013.

[41] X. Wang, B. Zerr, H. Thomas, B. Clement, and Z. Xie, "Pattern formation of multi-auv systems with the optical sensor based on displacement-based formation control," *International Journal of Systems Science*, vol. 51, no. 2, pp. 348–367, 2020.

[42] S.-M. Kang and H.-S. Ahn, "Design and realization of distributed adaptive formation control law for multi-agent systems with moving leader," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1268–1279, 2015.

[43] K. Aryankia and R. R. Selmic, "Neuro-adaptive formation control and target tracking for nonlinear multi-agent systems with time-delay," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 791–796, 2020.

[44] J. Sarangapani, *Neural Network Control of Nonlinear Discrete-Time Systems*. CRC press, 2018.

[45] H. Modares, R. Moghadam, F. L. Lewis, and A. Davoudi, "Static output-feedback synchronisation of multi-agent systems: a secure and unified approach," *IET Control Theory & Applications*, vol. 12, no. 8, pp. 1095–1106, 2018.

[46] H. Modares, B. Kiumarsi, F. L. Lewis, F. Ferrese, and A. Davoudi, "Resilient and robust synchronization of multiagent systems under attacks on sensors and actuators," *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 1240–1250, 2019.

[47] C. Wang and D. J. Hill, "Learning from neural control," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 130–146, 2006.

[48] A. Das and F. L. Lewis, "Cooperative adaptive control for synchronization of second-order systems with unknown nonlinearities," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 13, pp. 1509–1524, 2011.

[49] 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 Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1217–1226, 2014.

[50] Y. Yang, H. Xu, and D. Yue, "Observer-based distributed secure consensus control of a class of linear multi-agent systems subject to random attacks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 8, pp. 3089–3099, 2019.

[51] 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, 2012.

[52] K. Aryankia and R. R. Selmic, "Formation control and target tracking for a class of nonlinear multi-agent systems using neural networks," in *2020 European Control Conference (ECC)*. IEEE, 2020, pp. 160–165.

[53] A. Das and F. L. Lewis, "Distributed adaptive control for synchronization of unknown nonlinear networked systems," *Automatica*, vol. 46, no. 12, pp. 2014–2021, 2010.

[54] A. W. Marshall, I. Olkin, and B. C. Arnold, *Inequalities: Theory of Majorization and Its Applications*. Springer, 1979, vol. 143.

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

[56] K. Aryankia and R. R. Selmic, "Neural network-based formation control with target tracking for second-order nonlinear multi-agent systems," *IEEE Transactions on Aerospace and Electronic Systems*, 2021.

[57] F. Lewis, S. Jagannathan, and A. Yesildirak, *Neural Network Control of Robot Manipulators and Non-linear Systems*. CRC press, 2020.

[58] B. Cui, T. Ma, F. L. Lewis, C. Zhao, Y. Song, and C. Feng, "Distributed adaptive consensus control of heterogeneous multi-agent chaotic systems with unknown time delays," *IET Control Theory & Applications*, vol. 9, no. 16, pp. 2414–2422, 2015.

[59] S. Jagannathan and F. Lewis, "Discrete-time neural net controller for a class of nonlinear dynamical systems," *IEEE Transactions on Automatic Control*, vol. 41, no. 11, pp. 1693–1699, 1996.

[60] S. Li, R. Kong, and Y. Guo, "Cooperative distributed source seeking by multiple robots: Algorithms and experiments," *IEEE/ASME Transactions on mechatronics*, vol. 19, no. 6, pp. 1810–1820, 2014.