Distributed Fault Detection in Formation of Multi-Agent Systems with Attack Impact Analysis

Arefeh Amrollahi Biyooki

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 at

Concordia University

Montréal, Québec, Canada

May 2019

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Arefeh Amrollahi Biyooki

Entitled: Distributed Fault Detection in Formation of Multi-Agent Systems with Attack Impact Analysis

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

Master of Applied Science

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

Signed by the final examining committee:

_____ Dr. M. Kahrizi, Chair

_____ Dr. F. Haghighat, External Examiner

_____ Dr. M. Kahrizi, Examiner

_____ Dr. K. Khorasani, Supervisor

Approved by: _____

Dr. W. E. Lynch, Chair

Department of Electrical and Computer Engineering

_____ 20 _____

Dr. A. Asif

Dean, Faculty of Engineering and Computer Science

ABSTRACT

Distributed Fault Detection in Formation of Multi-Agent Systems with Attack Impact Analysis

Arefeh Amrollahi Biyooki

Autonomous Underwater Vehicles (AUVs) are capable of performing a variety of deep-water marine applications as in multiple mobile robots and cooperative robot reconnaissance. Due to the environment that AUVs operate in, fault detection and isolation as well as the formation control of AUVs are more challenging than other Multi-Agent Systems (MASs). In this thesis, two main challenges are tackled.

We first investigate the formation control and fault accommodation algorithms for AUVs in presence of abnormal events such as faults and communication attacks in any of the team members. These undesirable events can prevent the entire team to achieve a safe, reliable, and efficient performance while executing underwater mission tasks. For instance, AUVs may face unexpected actuator/sensor faults and the communication between AUVs can be compromised, and consequently make the entire multi-agent system vulnerable to cyber-attacks. Moreover, a possible deception attack on network system may have a negative impact on the environment and more importantly the national security. Furthermore, there are certain requirements for speed, position or depth of the AUV team. For this reason, we propose a distributed fault detection scheme that is able to detect and isolate faults in AUVs while maintaining their formation under security constraints. The effects of faults and communication attacks with a control theoretical perspective will be studied.

Another contribution of this thesis is to study a state estimation problem for a linear

dynamical system in presence of a Bias Injection Attack (BIA). For this purpose, a Kalman Filter (KF) is used, where we show that the impact of an attack can be analyzed as the solution of a quadratically constrained problem for which the exact solution can be found efficiently. We also introduce a lower bound for the attack impact in terms of the number of compromised actuators and combination of sensors and actuators. The theoretical findings are accompanied by simulation results and numerical can study examples.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor Prof. K. Khorasani. I am thankful to him for giving me the opportunity to join his research group at Concordia University. Completion of my masters would have not been possible without his kind support, motivation, and invaluable guidance.

Second, I would like to thank my committee members, Prof. M. Kahrizi, Prof. F. Haghighat for serving as my committee members and for their comments and suggestions.

Moreover, I would also like to thank members of the control lab team at Concordia University, especially: Esmaeil Alizadeh, Dr. Amir Baniamerian, Dr. Bahar Pourbabaee, Mehdi Taheri, Reza Bahrevar, Mahsa Khoshab, Maryam Abdollahi, Farinaz Kouhrangiha, Yanyan Shen, Rezvan Nozari for all the stimulating discussions and exchanges, as well as for all the fun and nice time we have had throughout these years.

Last, but by no means least, my special thanks go to my husband Ahmad, for his love and support over the years. He has always been a pillar of strength, inspiration and support and I also would like to thank to my daughter Samin. I especially thank my brother MohamadAli. I feel that my father who passed away recently is looking at this work from the Heaven. Additionally, I would like to thank my mother for her role in encouraging me to finish this piece of work.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AD Attack Detection

ADT Attack Detection Time

AUV Autonomous Underwater Vehicle

BIA Bias Injection Attack

CPS Cyber-Physical System

CST Compound Scalar Testing

CUMSUM Cumulative Sum

DOS Denial of Service Attack

DR Detection Rate

EKF Extended Kalman Filter

FD Fault Detection

FDI Fault Detection and Isolation

FDIR Fault Detection, Isolation and Recovery

FE Fault Estimation

FN False Negative

FP False Positive

FTC Fault Tolerant Control

GLR Generalized Likelihood Ratio

KF Kalman Filter

LKF Linear Kalman Filter

LOE Loss of Effectiveness

LTI Linear-Time Invariant

MAS Multi-Agent System

MMR Multiple Mobile Robot

PDF Probability Density Function

SA Secured Actuators

SPRT Sequential Probability Ratio Test

SS Secured Sensors

TN True Negative

TP True Positive

UAV Unmanned Aerial Vehicle

UKF Unscented Kalman Filter

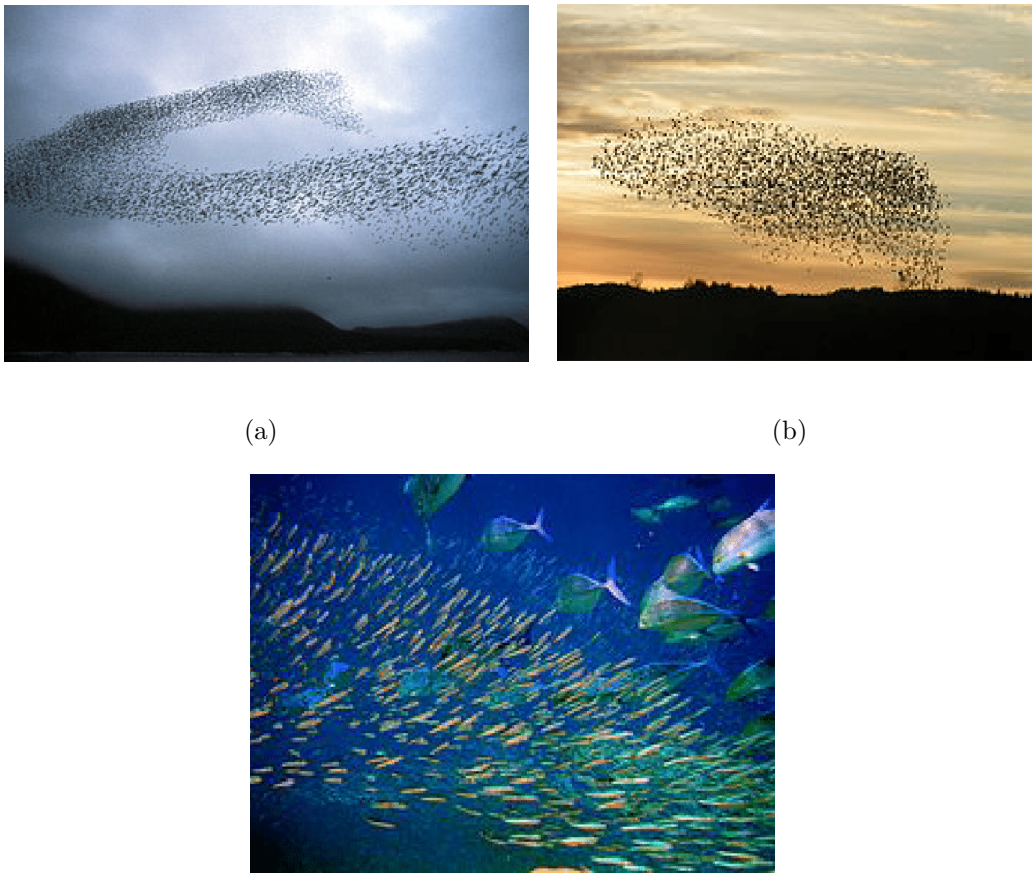
Chapter 1

Introduction

1.1 Motivation

Recently autonomous Multi-Agent Systems (MASs) and their coordinated movements have been the subject of many studies. These systems are controlled by cooperative control laws and are designed to act in a way that serve the group common purpose. Moreover, MASs can be found in biology, physics, applied mathematics, mechanics and control theory. The applications of MASs can be seen from the motion of a flock of birds, a herd of land animals, a school of fish and a swarming in natural systems as illustrated in Figure 1.1. Cooperative control of MASs can be applied to various applications such as multiple Unmanned Aerial Vehicles (UAVs) [1, 2], multiple Autonomous Underwater Vehicles (AUVs) [3, 4], multiple mobile robots, [5, 6], and multiple satellite clusters [7]. MASs are capable of executing more complex tasks due to their great advantages such as improving system efficiency, flexibility and reliability, reducing cost, and providing new capabilities that cannot be incorporated

within a single agent system.



(a) (b)

(c)

Figure 1.1: Examples of biological systems exhibiting formation behaviors in nature: (a), (b) flocks of birds, (c) schools of fishes [8].

Generally, the goal in MAS is to maintain a desired group shape formation. For instance, a group of satellites may be required to set themselves into a specific formation in order to efficiently provide and communicate coverage over a particular trajectory around the globe. Also, a group of UAVs may be required to fly in a particular formation to provide surveillance of a region or to decrease localization errors when positioning a target of interest. Formation control is the problem of controlling the relative position and orientation of robots

within a group in order to reach a desirable team formation. As the complexities of the missions increase UAVs and AUVs could not accomplish the mission tasks individually, and hence, application of MAS has provided lots of opportunities for various researches. The basic idea is to use relatively inexpensive, simple and small AUVs instead of expensive specialized ones to cooperatively fulfill the challenging AUV missions. The proposed approach can increase the overall reliability of the system or fulfill the missions that cannot be executed by a single AUV while decreasing the mission complexity. The degree of autonomy, capability of Fault Detection and Isolation (FDI) and identification are the crucial factors for AUV systems to successfully accomplish their mission tasks [9].

1.2 Literature Review

1.2.1 Multi-Agent Formation Control

There are many benefits in using a group of homogeneous or heterogeneous agents, instead of a single agent to accomplish a task. When comparing the mission outcome of a group of MASs controlling in the same environment to that of a single vehicle, the overall performance is easy to evaluate as MAS group can enhance 1) task allocation, 2) performance, 3) the time duration required, 4) the system effectiveness and 5) the safety to accomplish the desired outcome. Besides, the technical advances in wireless communication, sensors, and embedded systems have all play a part in the development of new generation of coordination algorithms for unmanned aerial/ground/underwater/space vehicles [3].

Generally, physical faults occur in an individual member of the team or communication among the team members. Moreover, we should consider different faults in various applications that are applicable for multiple mobile robots, multiple AUVs [10] and a fleet of marines, UAVs [11] and cooperative robot reconnaissance [12]. Control of MAS behavior is categorized into centralized and decentralized systems.

1.2.1.1 Centralized Control

In centralized systems, a powerful core central unit makes decisions and communicates them within the vehicles in the team. This core unit can optimize vehicle coordination, accommodate individual vehicle faults and monitor the accomplishment of the mission. However, it is possible that any fault in the central unit leads to a failure of the entire system. Most of the available literature on FDI of MASs are based on centralized fault diagnosis architectures [13]. In practice, it is quite challenging to address the problem of FDI in a network of MASs in the context of a centralized architecture due to the stringent computational resources and communication bandwidth limitations. The Figure 1.2 shows a centralized control architecture, consider the dynamics of each agents $i = \{1, \ldots, N_v\}$ and x_i, u_i are the state and control input.

Few drawbacks of a centralized system are as follows: a more complex controller design in case of having different agents in the network, task allocation, and more difficult task allocation and measurement. Moreover, the agents will not operate if the centralized control fails and the entire network breaks down.

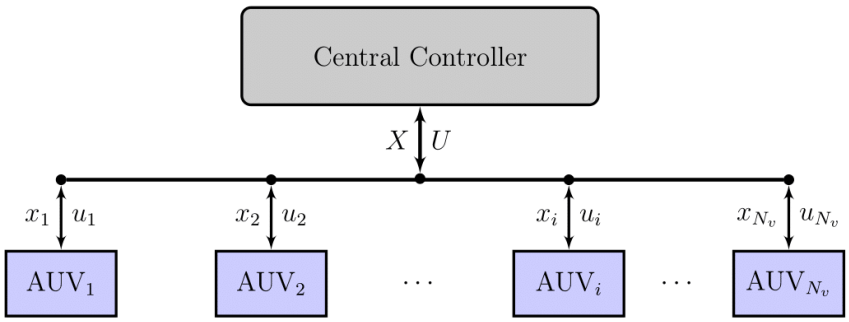


Figure 1.2: Centralized control of n MASs [9].

1.2.1.2 Decentralized Control

In the past decade, MASs have been the subject of considerable research due to their extensive applications in a wide variety of areas involving chemical manufacturing industries, geographical exploration, building automation and military industries [14–17]. It should be emphasized that the consensus problem for MASs have attracted special interest because of its clear practical insights. So far, various research as well as design methodologies have been exploited to deal with the consensus problems for different types of MAS [18–20].

In networks, individual vehicles can communicate and share information, but each vehicle receives particular task allocation to achieve part of the global mission. Some advantages of using decentralized networks are robustness to single agent failure, better system stability, better time constraints on applications, less communication load, and less computational power required from the agents. Since FDI modules in networks usually receive only local information, they will inevitably be unable to take into account the information corresponding to the neighbouring agents in their solution [21]. Consequently, the only viable and feasible

architecture is a distributed one that is more suitable than a centralized architecture due to its lower complexity and use of fewer network resources. Decentralized control architecture is more effective than a decentralized control due to information exchanges among the neighbouring agents. Figure 1.3 shows a decentralized control strategy of MASs.

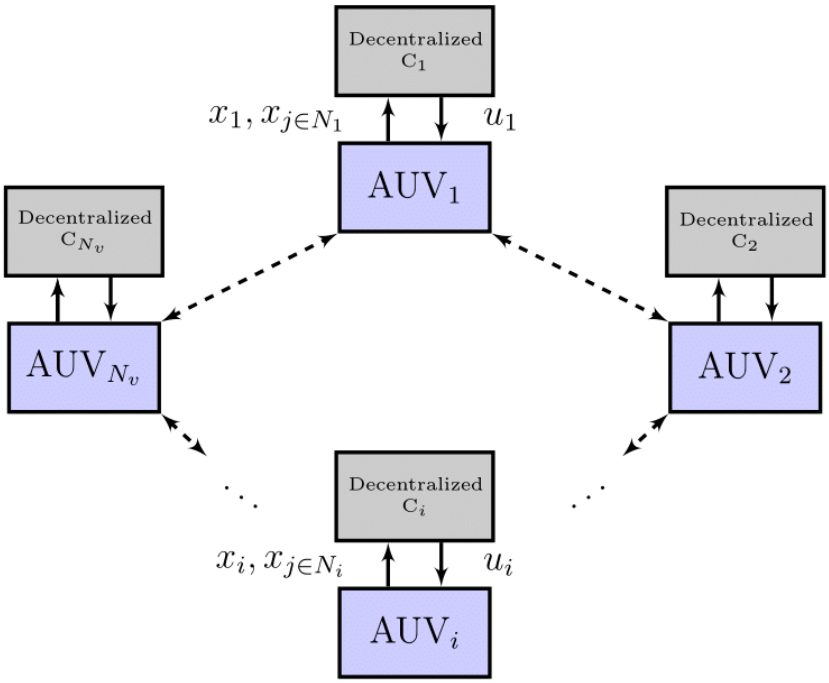


Figure 1.3: Decentralized control strategy of n MASs [22].

Some advantages of decentralized control strategy are the increased flexibility, individual control of each agent, more robustness of the system as compared to the other architectures, and a better communication among the direct neighbouring agents.

Different coordinate architectures and strategies have been developed for both centralized and decentralized methods. For instance, coordination of Multiple Mobile Robot (MMR) group includes behavior-based [23], virtual structure [24], leader-follower [25, 26], graph-based [27] and potential field approaches [28]. Figure 1.4 illustrates the block diagram

of different coordination strategies within MASs.

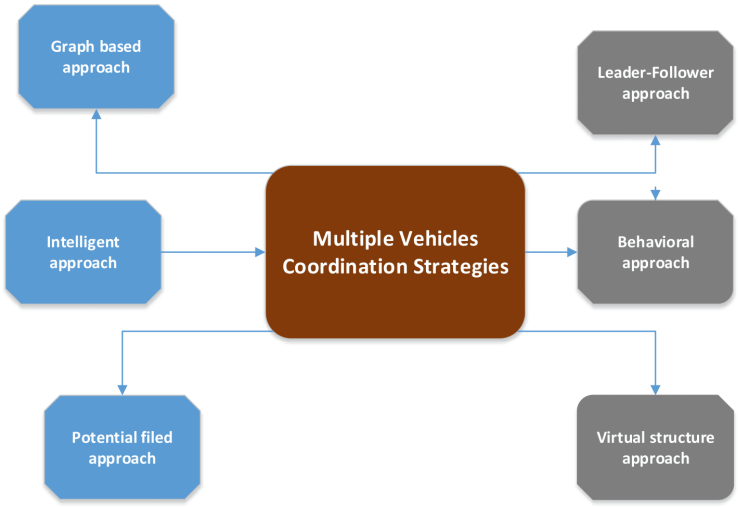


Figure 1.4: Coordinate strategies of MASs [9].

1.2.2 Fault Tolerant Control in Formation of Multi-Agent Systems

The FDI of single agent and MASs have been thoroughly researched in the literature [29–33]. The physical limitations and constraints of practical systems and saturation faults are challenging phenomena, such as deflection of control surfaces of UAVs, voltage limits on electrical motors, and flow rates of hydraulic actuator faults [34].

The problem of Fault Tolerant Control (FTC) in Markovian jump systems is investigated in the literature [35]. An adaptive observer-based approach is developed to detect the occurrence and to estimate the severity of actuator faults in the systems. A statistical local approach for the faults with very small amplitudes is considered in [36]. The FDI techniques based on sliding-mode observers for the robust decentralized system is studied in [37]. In [38], FD and Fault Estimation (FE) of network of sensing systems with incomplete

measurements is presented. A consensus based overlapping decentralized FDI approach are studied in [31]. In [39], the improvements in design and analysis of actuator FDI for a team of MAS are presented. A cooperative hierarchical actuator fault accommodation for formation of flying vehicles with absolute measurements and with various relative measurements are investigated in [40, 41]. The agents are modeled as Linear-Time Invariants (LTIs) and local fault recovery module can detect the LOE actuator faults and partially recover the faulty agents. A decentralized formation-level fault recovery module is also designed to enhance the overall performance of the team in [42].

The consensus problem for a second-order MAS is presented in [43] and a cooperative algorithm is proposed which leads the team to achieve consensus in presence of random directional communication link failures.

1.2.3 A Review of Attack and Security Approaches

Cyber-Physical System (CPS) consist of both logical elements such as embedded computers and physical elements connected by communication channels such as Internet [44]. Cyber attacks on control systems compromising measurement and actuator data integrity and availability have been considered in [45], where the authors have modeled attack effects on the physical dynamics. In [46], several attack scenarios have been simulated and analyzed on the Tennessee-Eastman process control system to study the attack impacts and its detectability.

Cyber security is one of the major concerns in Cyber-Physical Systems (CPSs). The cyber security aspect of CPS can be classified into two classes namely: 1) the information

security which is considered by encryption and data security, and 2) secure control theory which studies how cyber attacks affects the control system of physical dynamics. The safety tools using only information security are not sufficient for secure control of CPS because they cannot describe the systems macro-behavior, hence they should be complemented with a secure control theory.

Secure control theory focuses on an attack model that explains the uncertain and erratic nature of cyber attacks [47] such as the Denial of Service Attack (DOS). DOS refers to obstructing communication among networked agents and attacking the routing protocols. As a countermeasure, in the literature [48] provides a secure control scheme in presence of DOS attacks. Another game theoretic approach achieving a robust and resilient control against DOS attacks is studied in [49,50]. Another type of attack model is the deception attack that basically occurs by injecting incorrect information from sensors or to controllers [51]. For the deception attack model, the attacker can receive the secret key or compromise certain cyber elements in order to falsify the data. This type of attack has been researched in the deception attack of electric power distribution systems in the area of secure control theory [52]. The monitoring system could not detect all feasible deception attacks since the attacker with a good knowledge of the system can design a sophisticated deception attack that is approximately or absolutely undetectable.

The CPS is modeled as a stochastic linear system with a Gaussian white noise. Malicious stealthy deception attacks attempt to drive the state estimate of the Kalman Filter (KF) away from the actual state without being detected. Investigating all possible attacks

in MAS is an important problem in many areas. Attack or vulnerability taxonomies are designed for various methods [53] as provided below

- to develop automated tools for performing security assessment,
- to provide a way to explore unknown attacks,
- to understand the attacks implications and the defense mechanism against them.

For consecutive attacks, the maximum number of attacks is evaluated by recursive Riccati equations. For randomly launched deception attacks, the desired attack probability is obtained by solving a Riccati equation in [54]. Deception attacks compromising integrity have received much attention. For instance, replay attacks on the sensor measurements, which is an appropriate kind of deception attack have been studied. The authors in [55] consider that all existing sensors are attacked and suitable counter-measures to detect the attacks are considered. For this attack, adversaries do not have any model knowledge but they are able to receive and corrupt the sensor data. Recently, another form of deception attacks has been investigated and that is false-data injection attacks. For instance, in power networks, an adversary with perfect model knowledge has been investigated in [56]. In particular, in [57] the authors analyzed attack policies with limited model knowledge and performed experiments on a power system control software showing that such attacks are stealthy and can induce the erroneous belief that the system is at an unsafe state.

Data injection attacks on dynamic control systems have been studied. In [58], authors consider the set of attack policies for covert (undetectable) false-data injection attacks with detailed model knowledge and a full access to all sensor and actuator channels that describe

the set of undetectable false-data injection attacks for infinite adversaries with full-state information, but possibly compromising only a subset of the existing sensors and actuators. In the context of MAS, optimal adversary policies for data injection using full model knowledge and state information are proposed in [59]. The work in [60] considers the detection and reduction of false-data injection attacks on linear information dissemination algorithms over communication networks.

The deception attack is also known as false data-injection attacks or malicious attacks. The work in [61] illustrates that stealth attacks exist if the number of compromised measurements reach a certain value. Moreover, two models on sparse stealth attacks are constructed for two typical scenarios namely: 1) a model where random attacks in arbitrary measurements can be compromised, and 2) a model where targeted attacks in specified states are modified.

The deception attack occurs when an adversary replaces the true data exchanged between the agents by false signals. The deception or acceptance of false data means that there is an unauthorized access and interface with an asset. The attacker changes a message that contains data determining an action to be done, and the recipient thinks that this message is true and acts accordingly. Another form of this attack occurs when some information is modified and an authorized user accepts and approves the change while it is wrong. A famous example of this type of threat is a man in the middle. In such attacks, malicious users intercept and modify the data transmitted between users through a network connection, without the recipient and sender realizing the presence of the intermediary. In

deception attacks, false information is injected into sensors or controllers. The attacker obtains the secret key or compromises some cyber elements in order to falsify the data rather than obstruct it. These types of attacks have been studied in different applications such as power grid [62] and sensor networks [63].

Different versions of the deception attack are defined as follows:

a. **Fake messages:** Having a fake identity makes it easy for an attacker to send deceptive information to others.

b. **Fake agents:** An attacker plays many roles in a genuine interaction. Many fake bidder agents are created by an attacker to demolish the auction.

c. **Fake services:** An attacker creates an entirely fake interaction with fake agents for deception or disinformation such as a completely fake auction interaction with pseudo bidders to attract real agents and betray them.

d. **Reputation attack:** A group of attackers work together to deceive the reputation mechanism such that many malicious agents may collude to increase their own rank and thus defraud a trust service into believing that they are trusted agents [64].

The interaction between physical and cyber system make control systems different from conventional IT systems. Malicious actions can enter at network gateways in the close loop and cause vulnerability effects.

Deception attack models for dynamical networked control system are as follows:

- **Zero dynamics:** Only sensor or actuator along with the plant model are considered.

- **Covert:** Sensor and actuator as well as the plant model are considered.
- **Replay:** Only sensor, but no plant model is considered.
- **Bias:** Sensor and actuator as well as steady-state model are considered.

The FDI approaches can be applied for detecting the negative impact of cyber attacks on networked control systems. However, these tools might be exploited more successfully if similarities and differences between faults and attacks are well investigated. Both faults and attacks can be developed at an unknown time instant and may lead to an unreliable variation in the behavior of physical systems. The faults and attacks can be modeled by incorporating additive signals in the state space model of the system. Faults and attacks have inherently distinct features, making it difficult for traditional FDI techniques to implement in case of cyber-physical attacks.

The most important difference between a fault and an attack lies in that the fault occurs without deliberate human intervention that occur in the system components such as actuators, sensors, or communication channels, while the attack is an intentional action performed by malicious adversaries. In addition, simultaneous faults are suggested to be non-colluding while cyber attacks could be performed in a coordinated way. For these reasons, cyber-physical attacks may cause more destructive damage to the system than faults.

Another point is that cyber-physical attacks are much more difficult to detect and isolate, particularly when they are implemented in a coordinated way. The attackers can be controlled for partially or completely bypassing traditional irregularity detectors [65, 66], false data injection attacks [67], zero-dynamics attacks or covert attacks [68]. Also, it is

necessary to apply some priority methods for analyzing that the attacks are detectable and identifiable before performing detection and isolation techniques.

Faults often occur for a long time until they are detected, isolated and repaired but malicious attacks may be performed within a short time period due to the limited resources and covert intention of the adversaries [69, 70]. Moreover, for safety-critical applications, it is required to detect the attacks with a detection delay that is upper bounded by a certain prescribed value [71–73]. For these reasons, the detection and identification of attacks should be formulated as a sequential detection and isolation of transient changes in stochastic dynamical systems. Norm-based state estimator is applied in [74] to guarantee the boundedness of estimation errors caused by noise, modeling errors and cyber-attacks. Some work on state estimation on stochastic time-varying nonlinear systems subject to randomly occurring deception attacks are considered in [75] through framework of KF.

1.3 Thesis Contributions

- In Chapter 3, a performance analysis of a model-based distributed fault detection scheme for Multi-Agent Systems (MASs) is presented. Next, potential attack scenarios in MAS as well as the implementation and analysis of Fault Detection (FD) for MAS under deception attacks and fault scenarios are presented. The major contribution of this chapter is to develop a methodology and framework to distinguish faults and attacks.

- In Chapter 4, the Bias Injection Attack (BIA) strategy to the Kalman filtering problem

is extended, and the impact of the worst-case BIA in a stochastic setting is analyzed by a quadratically-constrained optimization program. The main contribution of this chapter is to use a combination of sensors and actuators in order to have minimum errors in the output and to also maintain estimation errors lower than an upper bound. Finally, we need to determine and choose which sensors and actuators should be secured.

1.4 Thesis Layout

- **Chapter 1** provides a literature review on various aspects of multi-agent formation control, Fault Tolerant Control (FTC) for formation of Multi-Agent Systems (MASs), and attack and security approaches.

- **Chapter 2** provides a background information on the topics that are used in this thesis. This chapter talks about the consensus problem in Multi-Agent Systems (MASs), formation graph modeling, modeling of Autonomous Underwater Vehicles (AUVs), fault diagnosis, and finally an introduction to Linear Kalman Filters (LKFs).

- **Chapter 3** introduces the proposed distributed Fault Detection (FD) in the formation of AUVs and the communication network among the AUVs in the presence of a deception attack. Moreover, each agent has a senor which measures the relative position of the neighbouring agents, the relative position of the sensor and the difference between the estimated agent position and the estimated neighbouring agent position are proposed. All the simulation results are provided in Section 3.4.

- **Chapter 4** introduces the problem of estimating the attack impact in monitoring systems. To tackle this issue, a Kalman filter equipped with a chi-squared detector under Bias Injection Attack (BIA) is presented. Based on the worst-case attack impact, a lower bound on the attack impact, the main goal is to pick which sensors or actuators and the combination of sensors and actuators should be secured. The simulation results are provided in Section 4.6.

- **Chapter 5** presents the conclusion as well as potential future work.

Chapter 2

Background Information

In this chapter, a review on background information needed for this work is presented. First, the consensus problem in MAS is given in Section 2.2. A formation graph modeling is presented in Section 2.1. Next, the dynamic model of Autonomous Underwater Vehicle (AUV) is explained in Section 2.3. Fault Detection and Isolation (FDI) is introduced in Section 2.4. Finally, in Sections 2.5 and 2.7, different types of actuator faults and the Kalman Filter (KF) method are presented.

2.1 Formation Graph Modeling

Assume that the $\mathcal{N}$ agents evolving in a d-dimensional state space where $x_i \in \mathbb{R}^d$ and $i = 1, \ldots, \mathcal{N}$. Let $\mathcal{V} = \{1, \ldots, \mathcal{N}\}$ be a set of vertices in graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is in the set of unordered pairs of vertices. An edge (v_i, v_j) is in $\mathcal{E}$ if agents i and j can interact with each other and the degree matrix is diagonal matrix $\mathcal{D} =$

$diag(\deg(v_1), \ldots, \deg(v_N))$ where the number of vertices adjacent equal v_i. The neighbour of the node i is denoted by $\mathcal{N} = \{j \mid (i,j) \in \mathcal{E}\}$. The weighted adjacency matrix $\mathcal{A} = [a_{ij}] \in \{0,1\}^{N \times N}$, where a_{ij} is defined as follows:

$$a_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in \mathcal{E}, \\ 0 & \text{if otherwise.} \end{cases}$$

Another matrix that we may assign to a weighted graph is Laplacian matrix. It is defined by $\mathcal{L} = \mathcal{D} - \mathcal{A}$, where $\mathcal{D}$ denotes diagonal weighted in-degree matrix. The adjacency and Laplacian matrices associated with our example graph shown in Figure 3.1 are given as follows:

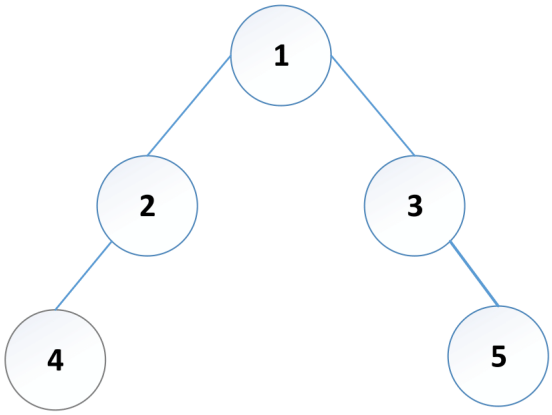


Figure 2.1: AUV formation with N=5.

$$\mathcal{L} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}, \quad \mathcal{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{2.1}$$

The graph Laplacian is defined by $\mathcal{L}$ is a positive semi-definitive matrix and hence $\mathcal{L} = \mathcal{D} - \mathcal{A}$ $\mathcal{L} = \mathcal{L}^{\mathcal{T}} \succeq 0$ [76].

In order to use the Laplacian matrix in the cooperative control design procedure, we first transform Laplacian matrix into its normal Jordan form $\mathcal{L} = \mathcal{M}\mathcal{J}\mathcal{M}^1$, where $\mathcal{M}$ denotes the transformation matrix. The main diagonal elements of matrix $\mathcal{J}$ are eigenvalues of Laplacian matrix and columns of the transformation matrix are their associated right eigenvectors [77].

Defitinion 1: Consider the MAS with the communication graph $\mathcal{G}$ and the proximity graph A is a simple graph in which two vertices are connected by an edge if and only if the vertices satisfy particular geometric requirements. The induced $\delta - disk$ proximity graph is specified by $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ and $(v_i, v_j) \in \mathcal{E}(t) \Leftrightarrow ||x_i(t) - x_j(t)|| \leq \delta$. Figure 2.2 illustrates the notation of the proximity graph.

Lemma 2.1.1 *Let $\mathcal{L}$ be the Laplacian associated with an undirected graph $\mathcal{G}$. Then, $\mathcal{L}$ has at least one zero eigenvalue and all of nonzero eigenvalues are positive. Furthermore, the matrix $\mathcal{L}$ has exactly one zero eigenvalue if and only if the undirected graph $\mathcal{G}$ is connected,*

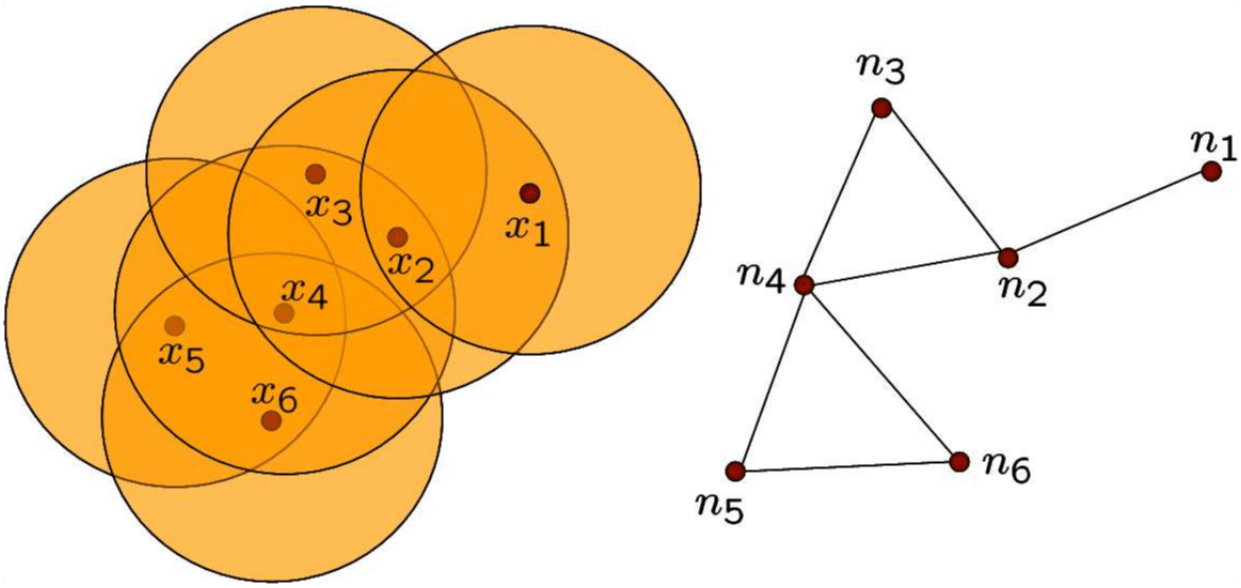


Figure 2.2: δ-disk proximity graph [76].

and the eigenvector associated with zero is $\mathbf{1}$ *[78].*

2.2 Consensus Problem in Multi-Agent Systems

The consensus problem is a particular example of a cooperative distributed algorithm that has been extensively studied by control and computer scientists in the field of distributed computing [79–81]. It is based on reaching an agreement on a certain quantity among all the nodes in a network that is a function of the initial state of each node. This problem has gained an increasing amount of attention from the control community due to its application in the distributed control of networked dynamic systems, such as in sensor fusion [82], decentralized estimation [83], and control of MAS like flocking, rendezvous, deployment, containment and formation [84]. Several studies have been connected in order to understand how topological properties of the network affect the performance of a group of dynamic agents using the

consensus algorithm [85]. These studies also cover a distributed control law, analyzing convergence analysis, an stability [78,86], controllability [87] and observability [88,89] properties among others.

The consensus problem in MAS is to determine a distributed control strategy that ensures all agents agree on a common value for a variable of interest. This value is generally called the consensus value and can represent a state or an output of the agent. Consensus algorithms are one of the tools that are applied for analysis of distributed systems, even when the network information structure has a vital effect on the control design where only part of the information is available to each agent. A consensus control means to design a networked interaction protocol such that all agents reach an agreement on their certain variables of common interest asymptotically or in a finite time. Each agent moves toward the centroid of its neighbouring set as given below for the single-integrator dynamics

$$\dot{x}_i(t) = u, \tag{2.2}$$

where

$$u = \sum_{j \in N} (x_i - x_j). \tag{2.3}$$

and equations (2.2) and (2.3) can be rewritten as follows

$$\dot{x}_i = -\deg(v_i)x_i + \sum_{i=1}^{\mathcal{N}} a_{ij}x_j. \tag{2.4}$$

where

$$a_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in \mathcal{E}, \\ 0 & \text{if otherwise.} \end{cases}$$

Hence, consensus or agreement dynamics is given as follows:

$$u = -\mathcal{L}x. \tag{2.5}$$

where $x = [x_1, \ldots, x_n]^T$.

2.3 Modeling of Autonomous Underwater Vehicles

In this section, important coordinate frames that are necessary for developing the AUV model are explained. Then, the dynamics of an AUV is presented. Finally, nonlinear equations of AUV motion in the horizontal plane are described.

2.3.1 Coordinate Frames

The Autonomous Underwater Vehicle (AUV) in six degrees of freedom depends on the choice of coordinates. The equations of motion for the AUV can be defined using two coordinate frames as:

- World-fixed reference frame $\{W\}$ is defined by axes (O_W-X_W, Y_W, Z_W) where O_W is the origin of the W frame, X_W axis points to the north, Y_W axis points to the east, and the Z_W axis points to the center of the earth.

- Body-fixed reference frame $\{B\}$ is defined by axes (O_B -X_B, Y_B, Z_B) where O_B is the origin of the B frame. The body-fixed frame with orthogonal axes X_B, Y_B, Z_B is coupled to the vehicle, the X-axis points to the forward direction, Y-axis points to the right of the vehicle and the Z-axis points vertically down. The origin O_B is selected to coincide with the center of gravity of the vehicle. The directions of coordinate frames are displayed in Figure 2.3 where degrees of freedom of the frame $\{B\}$ are: *surge, sway, heave, roll, pitch* and *yaw* [9]. Moreover, we have

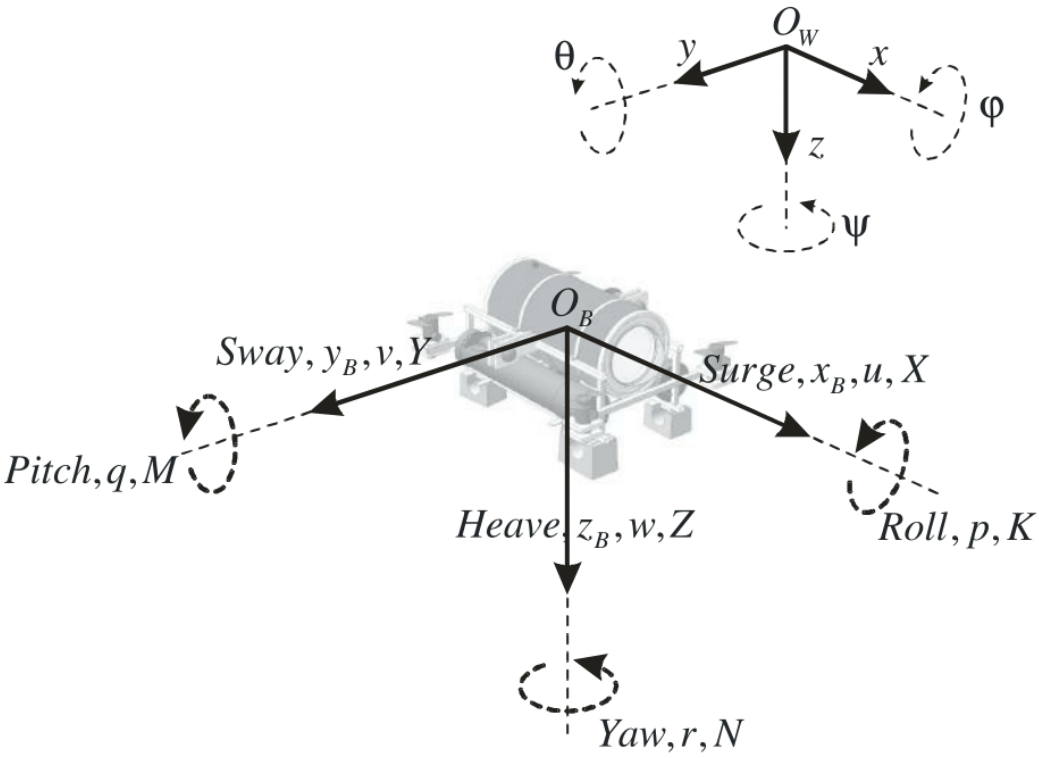


Figure 2.3: Definition of the reference frames $\{W\}$ and $\{B\}$ [90].

- $\eta = [\eta_1^T, \eta_2^T]^T$ where $\eta_1 = [p_x, p_y, p_z]^T$ and $\eta_2 = [p_\varphi, p_\theta, p_\psi]^T$ denote the position and orientation of $\{B\}$ expressed in $\{W\}$, respectively.

- $\nu = [\nu_1^T, \nu_2^T]^T$ where $\nu_1 = [v_u, v_v, v_w]^T$ and $\nu_2 = [v_p, v_q, v_r]^T$ denote the linear velocities and angular velocities of the vehicle expressed in $\{B\}$, respectively.

- $\tau = [\tau_1^T, \tau_2^T]^T$ where $\tau_1 = [X, Y, Z]^T$ and $\tau_2 = [M, N, K]^T$ denote the total forces and moments acting on the vehicle expressed in $\{B\}$, respectively.

Euler Angles

Euler angles relate two coordinate systems in terms of their orientation, as the orientation of the $\{B\}$-frame with respect to the $\{W\}$-frame. To orientate one coordinate system with respect to another, it will be subjected to a sequence of three rotations. The Euler convention used to describe the orientation from a body to world is the z-y-x convention. The B-frame is first rotated around the z-axis, then around the y-axis, and finally around the x-axis. This sequence of rotations corresponds to the rotation angles of yaw (p_ψ), $p$itch (p_θ) and $r$oll (p_φ), respectively. The rotation matrix used to describe the orientation of the B-frame with respect to the W-frame is illustrated as follows:

$$\begin{aligned} R_B^W &= (p_\varphi, p_\theta, p_\psi) = R_z(p_\psi) R_y(p_\theta) R_x(p_\varphi), \\ R^{30}(p_\varphi, p_\theta, p_\psi) &= R^{32}(p_\psi) R^{21}(p_\theta) R^{10}(p_\varphi), \end{aligned} \tag{2.6a}$$

$$R_B^W(\eta_2)^{-1} = R_W^B(\eta_2) \tag{2.6b}$$

$$R_z(p_\psi) = \begin{bmatrix} \cos(p_\psi) & -\sin(p_\psi) & 0 \\ \sin(p_\psi) & \cos(p_\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.7a}$$

$$R_y(p_\theta) = \begin{bmatrix} \cos(p_\theta) & 0 & \sin(p_\theta) \\ 0 & 1 & 0 \\ -\sin(p_\theta) & 0 & \cos(p_\theta) \end{bmatrix} \tag{2.7b}$$

$$R_x(p_\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(p_\varphi) & -\sin(p_\varphi) \\ 0 & \sin(p_\varphi) & \cos(p_\varphi) \end{bmatrix} \tag{2.7c}$$

The transformation matrix can be written as

$$J_B^W(\eta_2) = \begin{bmatrix} 1 & \sin(p_\varphi)\tan(p_\theta) & \cos(p_\varphi)\tan(p_\theta) \\ 0 & \cos(p_\varphi) & -\sin(p_\varphi) \\ 0 & \sin(p_\varphi)/\cos(p_\theta) & \cos(p_\varphi)/\cos(p_\theta) \end{bmatrix}, \tag{2.8a}$$

$$J_B^W(\eta_2)^{-1} = J_W^B(\eta_2) \tag{2.8b}$$

The consequence of using the Euler angles is to describe the vehicle motion by $J_B^W(\eta_2)$ can not be displayed for a pitch angle of $p_\theta = \pm\pi/2$. However, the AUV will always operate far from this singular point. The kinematic AUV equations relative to the earth-fixed frame can

be converted into the following compact 6-dimensional matrix form,

$$\dot{\eta} = J(\eta)\nu \Longleftrightarrow \begin{bmatrix} \dot{\eta}_1 \\ \\ \dot{\eta}_2 \end{bmatrix} = \begin{bmatrix} R_B^W(\eta_2) & 0 \\ \\ 0 & J_B^W(\eta_2) \end{bmatrix} \begin{bmatrix} \nu_1 \\ \\ \nu_2 \end{bmatrix}. \tag{2.9}$$

2.3.2 AUV Nonlinear Equation of Motion

The general description of AUV equations of motion is obtained by applying the Newton-Euler formulation to a rigid body in the fluid. The dynamic equations of a six degrees of freedom AUV are in the form of [90]

$$M\dot{v} + C(v)v + D(v)v + g(\eta) = \tau, \tag{2.10a}$$

$$\dot{\eta} = J(\eta)\nu. \tag{2.10b}$$

- **Mass and Inertia Matrix**

The mass and inertia consist of a rigid body mass (M_{RB}) and added mass M_A as illustrated below:

$$M = M_{RB} + M_A. \tag{2.11}$$

- **Coriolis and Centripetal Matrix**:

The coriolis and centripetal matrix consists of a rigid body and an added mass term that include:

$$C(v) = C_{RB}(v) + C_A(v). \tag{2.12}$$

- **Hydrodynamic Damping Matrix**:

The hydrodynamic damping of underwater vehicles usually contains drag and lift forces. However, the AUV operates only at a low speed which makes lift forces negligible as compared to drag forces. The drag forces can be separated into a linear and a quadratic term as follows

$$D(v) = D_q(v) + D_l(v). \tag{2.13}$$

where $D_q(v)$ and $D_l(v)$ are the quadratic and linear drag terms, respectively.

2.3.3 AUV Nonlinear Equations of Motion in Horizontal Plane

In this thesis, we assume that the AUVs move in a horizontal plane. Moreover, the AUV is symmetric in all planes and the origin of the body-fixed frame O_B, the center of gravity C_G, and the center of buoyancy C_B of the AUV coincide with each other, *i.e.* $O_B = C_G = C_B = [0 \quad 0 \quad 0]$. Kinematic and dynamic equations of motion in the *surge, sway*, and *yaw* are provided below

$$\dot{p}_x = v_u \cos(p_\psi) - v_v \sin(p_\psi), \tag{2.14a}$$

$$\dot{p}_y = v_u \sin p_\psi + v_v \cos(p_\psi), \tag{2.14b}$$

$$\dot{p}_\psi = v_r, \tag{2.14c}$$

$$m_u \dot{v}_u - m_v v_v v_r + d_v v_v = \tau_u, \tag{2.14d}$$

$$m_u \dot{v}_u - m_v v_v v_r + d_v v_v = \tau_v, \tag{2.14e}$$

$$m_r \dot{v}_r + (m_v - m_u) v_u v_v + d_r v_r = \tau_r. \tag{2.14f}$$

Equation (2.14) can be combined in a matrix form as follows

$$M\dot{v} + C(v)v + Dv = \tau, \tag{2.15a}$$

$$\dot{\eta} = J(p_\psi)\nu, \tag{2.15b}$$

where $v = [v_u \quad v_v \quad v_r]^T$ and $\eta = [p_x \quad p_\psi \quad p_z]^T$ are the vectors of linear and angular velocities absolute positions and orientation in the world-fixed frame. $J(p_\psi)$ in equation (2.15b) is obtained as follows

$$J(p_\psi) = \begin{bmatrix} \cos(p_\psi) & -\sin(p_\psi) & 0 \\ \sin(p_\psi) & \cos_{p_\psi} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.16}$$

The total mass and inertia constant parameters along X, Y and about Z axes are given by

$$M = \begin{bmatrix} m_u & 0 & 0 \\ 0 & m_v & 0 \\ 0 & 0 & m_r \end{bmatrix}, \tag{2.17}$$

The drag matrix D defined in equation (2.15a) consists of parameters d_u, d_v, d_r along X, Y and Z axes, respectively, as

$$D = \begin{bmatrix} d_u & 0 & 0 \\ 0 & d_v & 0 \\ 0 & 0 & d_r \end{bmatrix}. \tag{2.18}$$

The Coriolis and centripetal matrix $C(v)$ is expressed as follows:

$$C(v) = \begin{bmatrix} 0 & 0 & -m_v v_v \\ 0 & 0 & m_u v_u \\ m_v v_v & -m_u v_u & 0 \end{bmatrix}, \tag{2.19}$$

Finally, the vector τ in equation (2.15a) given below

$$\tau = \begin{bmatrix} \tau_u \\ \tau_v \\ \tau_r \end{bmatrix}. \tag{2.20}$$

is the control input with τ_u τ_v and τ_r as the total forces and torques produced by the actuators in *surge, sway* and *yaw* directions, respectively [9].

2.4 Fault Diagnosis

Fault Detection and Isolation (FDI) methods are developed based on the concept of redundancy, which can be either hardware redundancy or analytical redundancy as illustrated in Figure 2.4 [91]. In order to improve the reliability of a system, fault diagnosis is usually used to monitor, locate, and detect a fault by applying the concept of redundancy, either hardware or analytical redundancy [92].

Hardware Redundancy

The essential concept of the hardware redundancy is to have identical components with the same input signals so that the duplicated output signals can be compared, leading to diagnosis decision by a variety of methods such as limit checking and majority voting. Hardware redundancy is reliable, but expensive and increases the weight as well as occupying more space [92]. The common technique used in the hardware redundancy approaches are the cross channel monitoring method, residual generation using parity generation, and signal processing methods such as wavelet transformation [92].

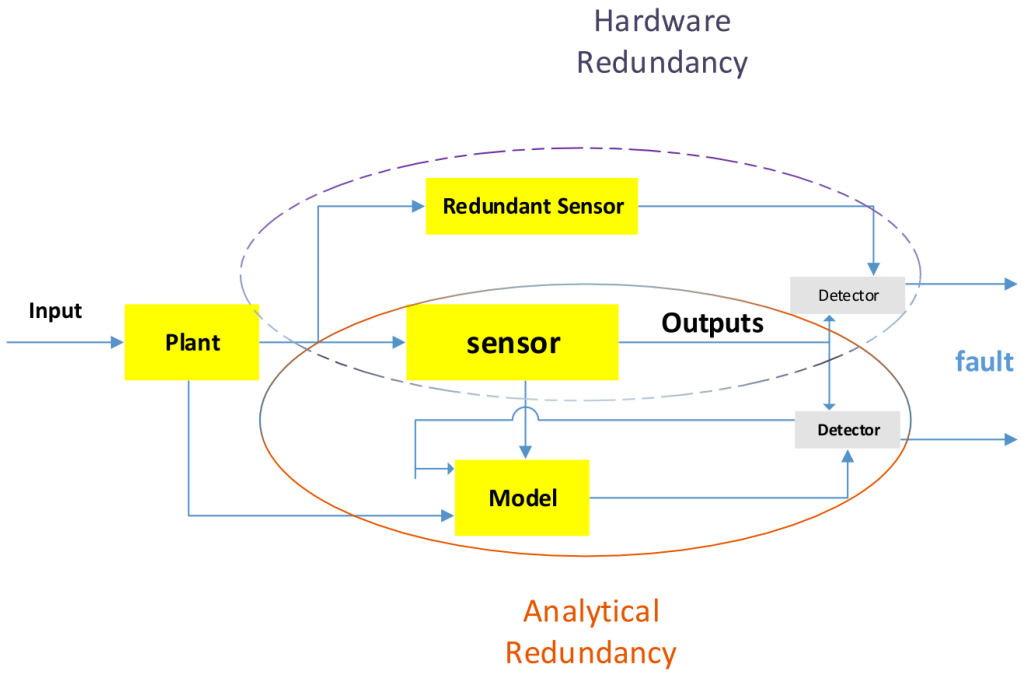


Figure 2.4: Illustration of hardware and analytical redundancy for FDI purposes [91].

Analytical Redundancy

A general structure of an analytical redundancy-based (or model-based) Fault Detection, Isolation and Recovery (FDIR) of the system is illustrated in Figure 2.4. Usually, analytical redundancy approaches are classified into quantitative model-based, model-based methods, and qualitative model-based methods. This classification of the diagnostic system is shown in Figure 2.5. The analytical redundancy approaches (also termed functional, inherent or

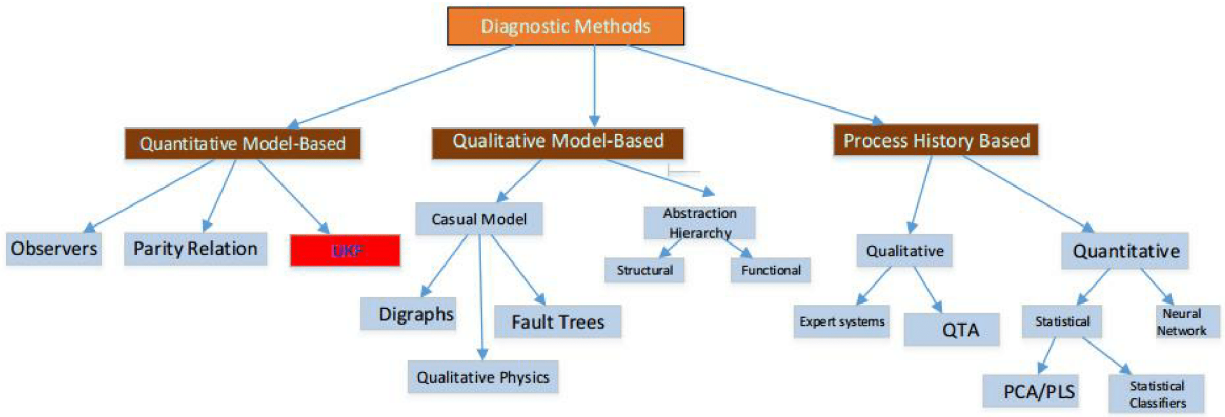


Figure 2.5: Classification of fault diagnostic algorithms [93]

.

artificial redundancy) work on the basis of the functional dependence among the states, input and the output of the system. The analytical redundancy can be categorized into two classes as follows [94]:

- Direct redundancy: This method is achieved using a relationship between different sensor measurements and differences among various sensor measurements that are useful in obtaining other sensors.

- Temporal redundancy: This method depends on differential or various relationships

among various sensors output and actuator inputs. With process input and output data, temporal redundancy is useful for sensor and actuator fault detection [94].

The fault diagnosis problem can be divided into three steps. First, a set of variables known as residuals can be generated by using one or more residual generation filters. These residuals should ideally be zero (or have zero means) under no-fault conditions. The second step is to make decisions on whether a fault has occurred (fault detection) determine the type of faults that have occurred based on the residuals (fault isolation). Finally, the controller is reconfigured in an online manner in case a fault is detected. Furthermore, a practical system is usually subjected to noise and unknown disturbances. This results in residuals that are not zero, even under no-fault conditions. [94].

- Residual Generation: Common approaches to generate residuals for FDI purposes are observer-based and Kalman Filter (KF) methods [95], parity relation methods [96], and parameter estimation methods.

- Robust Residual Evaluation: It is an essential strategy to develop robust hypothesis testing algorithms to evaluate the residual. The strategy concentrates on a robust approach of detecting a change in signal or system parameters that corresponds to a fault. The simplest decision rule is to declare that a fault occurs when the instantaneous value of a residual exceeds a threshold. More sophisticated residual evaluation methods are based on statistical decision theories such as Generalized Likelihood Ratio (GLR) test or a sequential probability ratio test [97, 98].

Fault Detection, Isolation and Recovery (FDIR) techniques are an important problem in many disciplines. The there steps in FDIR are defined as follows: Diagnosis methods are classified into three categories as follows:

- **Fault detection**: to make a binary decision either that something has gone wrong or that everything is fine (determine the time of a fault occurrence).
- **Fault isolation:** to determine the location of the fault, e.g, which sensor or actuator has become faulty.
- **Fault identification:** to estimate the size and type or nature of the fault.

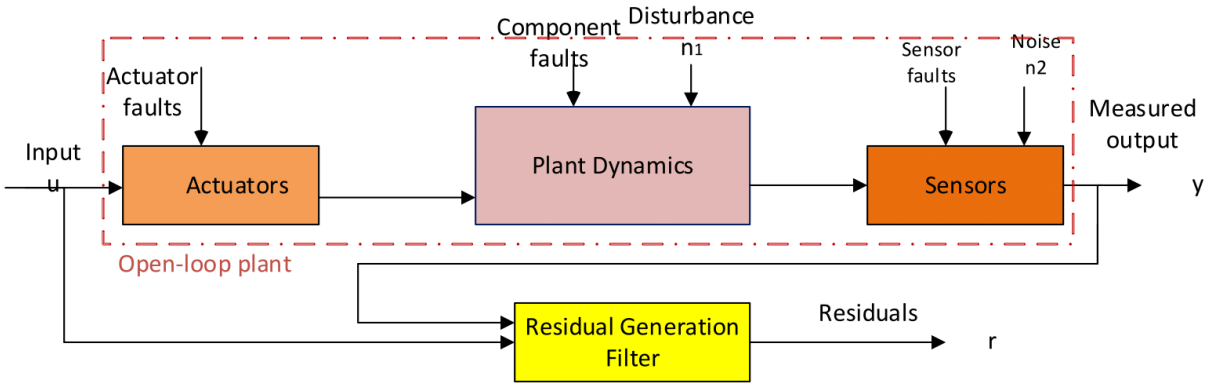


Figure 2.6: Fault representation model used for designing the residual generation filter [91].

2.4.1 Residual Generation

As illustrated in Figure 2.6 the residual signal is generated based on the input vector $u(t)$ and the output vector $y(t)$, and hence [91]

$$r(t) = g(u(t), y(t)). \tag{2.21}$$

The residual signal $r(t)$ can be generated as a difference between the measured output $y(t)$ and an estimated output $\hat{y}(t)$ as illustrated in equation (2.22),

$$r(t) = y(t) - \hat{y}(t). \tag{2.22}$$

Decision Making

Most decision logics use the history and trend of the residuals and utilize powerful or optimal statistical test techniques. Few well-known examples of these statistical tests are as follows [91]:

1. Sequential Probability Ratio Test (SPRT),
2. Cumulative Sum (CUMSUM),
3. Generalized Likelihood Ratio (GLR),
4. Local approach.

Reconfiguration

The reconfiguration step involves modifying the controller in response to faults that are detected and isolated in order to ensure safe or satisfactory system operation system. There are different reconfiguration control methods based on online learning or system identification. Figure 2.7 illustrates a summary of various FDIR techniques.

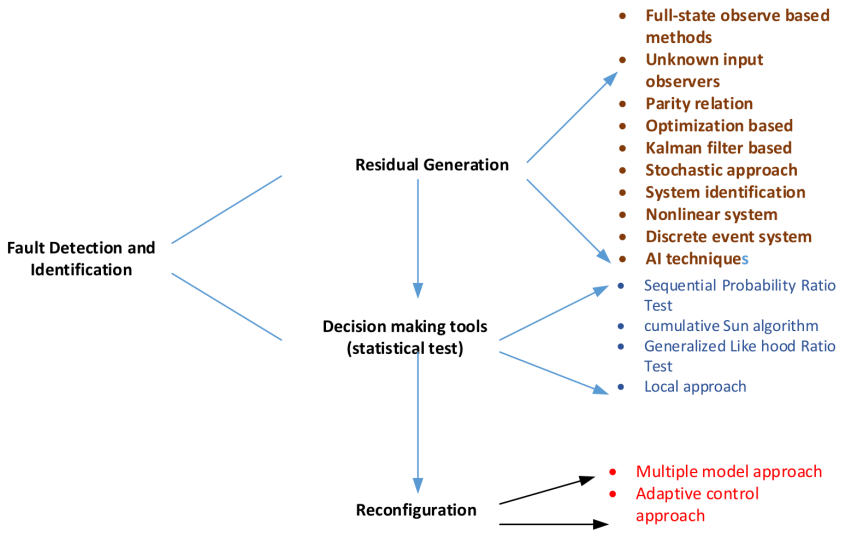


Figure 2.7: Classification of FDIR techniques [91].

In the design of a fault diagnosis system, the following tasks and questions should be considered [99]:

- How to handle the noise in the system?
- How to handle multiple faults?
- How to handle disturbances (additive uncertainty)?
- How to handle modeling errors (multiplicative uncertainty)?
- How to handle nonlinearites?
- How to cope with detection delays?
- How to overcome the complexity in a FDI algorithm design?

- How to minimize the complexity in a FDI algorithm implementation (or execution)?
- What are the requirements for a prior modeling information?
- How good self learning and adaptive capabilities are?

2.5 Actuator Fault

A fault can occur in actuator, sensor or the plant. In terms of induced effects of faults on the system performance, faults can be either additive or multiplicative. Actuator faults have an important effect on the AUV system performance and may lead to the system malfunction or failure. Actuator faults are commonly defined sudden change of the control input u to u_{ci} (the actual input generated by the faulty actuator). The effectiveness coefficient matrix of the actuator control parameters is expressed as follows:

1. Loss of Effectiveness (LOE): A decrease in the actuator gain that results in a deflection that has a smaller gain with respect to the nominal value. Hence, the dynamics of the i-th faulty agent can be modeled after the injection of fault at time t_f as

$$u_i(t) = \bar{k_i} u_{ci}(t), \qquad 0 < \bar{k_i} < 1 \quad \forall t \geq t_f \tag{2.23}$$

2. *Lock in place*: The actuator is locked to a certain position at an unknown time t_f and does not respond to subsequent commands as can be noted from equation (2.24)

$$u_i(t) = u_{ci}(t_f) = k \qquad \forall t \geq t_f \tag{2.24}$$

where k is constant number.

3. *Outage*: In this scenario, the actuator produces zero force and moment hence it becomes ineffective, hence,

$$u_i(t) = 0, \qquad \forall t \geq t_f \tag{2.25}$$

where the actuator input and output of the i^{th} actuator are represented as $u_{ci}(t_f)$ the time that fault is injected to the i^{th} actuator is denoted by t_f, the actuator effectiveness coefficient of the i^{th} actuator is defined as $k_i(t) \in [\epsilon_i, 1]$ where the minimum effectiveness. most of the works have considered the actuator redundancy of underwater vehicles. On the other hand, like most of the mechanical systems, Loss of Effectiveness (LOE) in actuators of underwater vehicles is highly probable. Therefore, in order to overcome the lack of this topic in literature, in this thesis we focused on LOE faults in actuators of underwater vehicles.

2.6 Sensor Fault

The AUVs are equipped with different type of sensors. Kalman Filter (KF) is one of the most common approaches in sensor fusion that provides the required signals for the controller. Common sensors used in AUVs are:

- IMU (Inertial Measurement Unit): It provides information about the vehicles linear acceleration and angular velocity through a combination of accelerators, magnetometers and gyroscopes.

- Depth sensor: It measures the depth of AUV by measuring the water pressure.

- Altitude and frontal sonars: They indicate the existence of obstacles and the distance from the sea bottom.
- Ground speed sonar: This sensor provides the linear velocity of the vehicle with reference to the ground.
- Current meter: It determines the relative measurements between the vehicle's velocity and the water.
- GNSS (Global Navigation Satellite System): It is utilized to reset the drift error of the IMU and localize exactly the vehicle. It only works only at the surface.
- Compass: It measures the vehicle's yaw.
- Baseline acoustic: It determines the exact localization of the vehicle in a specific range of underwater environment by using one or more transmitters.

The various sensors that are used in an AUV depends on its application. Each sensor can become faulty. The output zeroing or external disturbances can be considered as a failure in AUV sensors [100].

2.7 Kalman Filter (KF)

Linear Kalman Filters (LKFs) are widely used to estimate system states and parameters. Linear Kalman Filters (LKFs), or just KF is a linear quadratic estimation problem that has two steps, namely the prediction and the update (correction). The state variables are

estimated in the prediction step and are corrected upon receiving the next measurement. The LKF can be described for the linear system as given by:

$$\begin{aligned} x(k+1) &= A(k)x(k) + B(k)u(k) + \omega(k), \qquad x(0) = x_0 \\ z(k) &= H(k)x(k) + \upsilon(k), \end{aligned} \tag{2.26}$$

where $x(k), z(k), \omega(k), \upsilon(k)$ and x_0 are the state variables, outputs, process and measurement noise and the initial values of the states, respectively, $A(k)$, $B(k)$, and $H(k)$ are the time-varing state-space matrices. The $w(k)$ and $v(k)$ are the zero mean Gaussian noise with zero mean and the covariance matrices $W(k)$ and $V(k)$, respectively. The covariance of the two noise models are given by

$$Q = \mathbb{E}[w(k)w^T(k)], \tag{2.27a}$$

$$R = \mathbb{E}[v(k)v^T(k)]. \tag{2.27b}$$

The mean squared error is given by equation (2.29) and this is equivalent to

$$\mathbb{E}[e(k)e^T(k)] = P(k) \tag{2.28}$$

where $P(k)$ is the error covariance matrix at time k and is calculated as follows:

$$P(k) = \mathbb{E}[e(k)e^T(k)] = \mathbb{E}[(x(k) - \hat{x}(k))(x(k) - \hat{x}(k))^T] \tag{2.29}$$

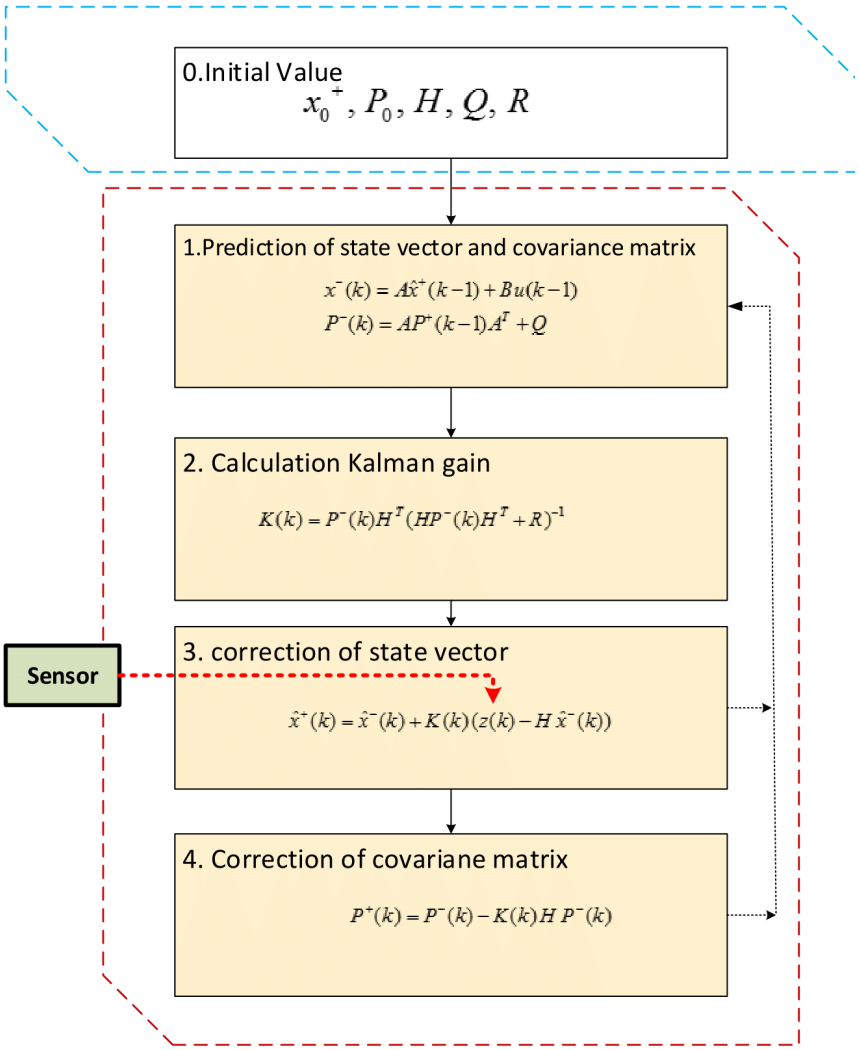


Figure 2.8: Flowchart for the KF [101].

Moreover, the KF estimates the process state at any given time. In the next step, the feedback can be obtained in the form of noisy measurements. Two main groups, time update equations and measurement equations are calculated in KF. The time update equations obtain the current state and error covariance estimates to receive the a priori estimates for the next step. The measurement update equations obtain a new measurement into the a

priori estimate to receive an enhanced a posteriori estimate. These details are shown in Figure 2.8.

2.8 Conclusion

In this chapter, background information that is necessary for model of multi Autonomous Underwater Vehicles (AUVs) were presented. The consensus problem in MAS was explained and the fundamental concepts of formation graph modeling were illustrated. In the next section, the dynamic model of AUV was introduced. The essential topics of Fault Detection and Isolation (FDI) were discussed. Finally, different types of actuator faults and a brief review of Linear Kalman Filter (LKF) were presented. The objectives of this thesis are: 1) analyzing the performance of a model-based distributed fault detection scheme for MAS, 2) considering potential attack scenarios in MAS as well as their implementation, 3) the analysis of FD for MAS under BIA, 4) considering various fault scenarios and developing a framework to distinguish faults and attacks. Moreover, the following two problems are addressed in Chapter 4. First, an estimation problem in case of attack impact in the presence of cyber-attacks is studied. Then, we analyze the problem of finding the worst-case BIAs.

Chapter 3

Distributed Fault and Bias Injection Attack Detection in the Formation of AUVs

3.1 Introduction

In this chapter, the fault detection and Bias Injection Attack (BIA) in a network of AUVs formation using Unscented Kalman Filters (UKFs) are presented. Both actuator and sensor faults of AUVs are investigated. Moreover, the problem of cyber attack detection and isolation is studied among communication of AUVs in the formation control. The communication network among the AUVs ensures that the effects to a BIA from the attacker and AUV formation control using a consensus algorithm to reach a pre-specified formation is

achieved.

3.2 AUV System under Study

The underlying system is nonlinear and the actuator fault and sensor fault are incorporated as illustrated in equation (3.1). We assume to have five AUVs that have nonlinear dynamics and each node i has two states as governed by

$$\begin{aligned} \dot{X}_i &= f(X_i, u_a.F_{a,i}) + w_i, \quad i = 1, \ldots, N \\ y_i &= h(X_i) + F_{s,i} + v_i, \end{aligned} \tag{3.1}$$

where $X_i = [x_i, v_i]^T \in \mathbb{R}^2$ illustrates the state variable, u_i represents the control input, w_i and v_i represent the external disturbances and $F_{a,i}$, $F_{s,i}$ are the actuator (LOE) and bias sensor fault of the i^{th} AUVs, respectively.

3.2.1 Formation Control of AUVs

This chapter considers a triangular formation consisting of the five agents. This simple setup has been considered in the formation control in [102–105] and we refer the reader to these papers for additional background and references on controlling triangular formations. A triangular formation of five AUVs is shown in Figure 3.1.

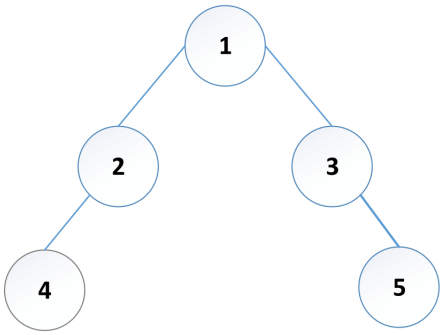


Figure 3.1: AUV formation with N=5.

The control input u_i should be a function of the relative state of the vehicles, that is where the control law is given as follows [13]:

$$u_i(t) = -k_i v_i(t) + \sum_{j \in N_i} [(x_j(t) - x_i(t)) + \gamma(v_j(t) - v_i(t))]. \tag{3.2}$$

where k and γ are the feedback gains for the velocity consensus. Applying a feedback control for the position and velocity consensus, the augmented control input $U = [u_1, \ldots, u_N]^T$ can be written as:

$$U = (K \otimes I_m)(X - X_d) + (G \otimes I_m)V, \tag{3.3}$$

where $\otimes$ denotes the Kronecker product, X_d is the desired state vector and V is the external control input (reference input) and X, X_d are given by

$$X = \begin{bmatrix} x_1^T & x_2^T & \ldots & x_N^T, v_1^T, v_2^T, \ldots & v_N^T \end{bmatrix}^T, \tag{3.4}$$

$$X_d = \begin{bmatrix} x_{d,1}^T & x_{d,2}^T & \ldots & x_{d,N}^T, v_{d,1}^T, v_{d,2}^T, \ldots & v_{d,N}^T \end{bmatrix}.^T \tag{3.5}$$

In equation (3.3), K and G are the control gain matrices that are given in equations (3.6a) and (3.6b) respectively, that is

$$K = \begin{bmatrix} 0_N & 0_N \\ -\mathcal{L} & -\gamma\mathcal{L} - kI_N \end{bmatrix}, \tag{3.6a}$$

$$G = \begin{bmatrix} 0_N \\ -I_N \end{bmatrix}, \tag{3.6b}$$

where $\mathcal{L}$ is the Laplacian matrix and $k = diag(k_1, \ldots, k_N)$.

The formation performance with fault and without fault on the actuator and sensor of one of the nodes of the AUVs will be studied in Section 3.4. Moreover, the FDI scheme for the formation control is illustrated in Figure 3.2.

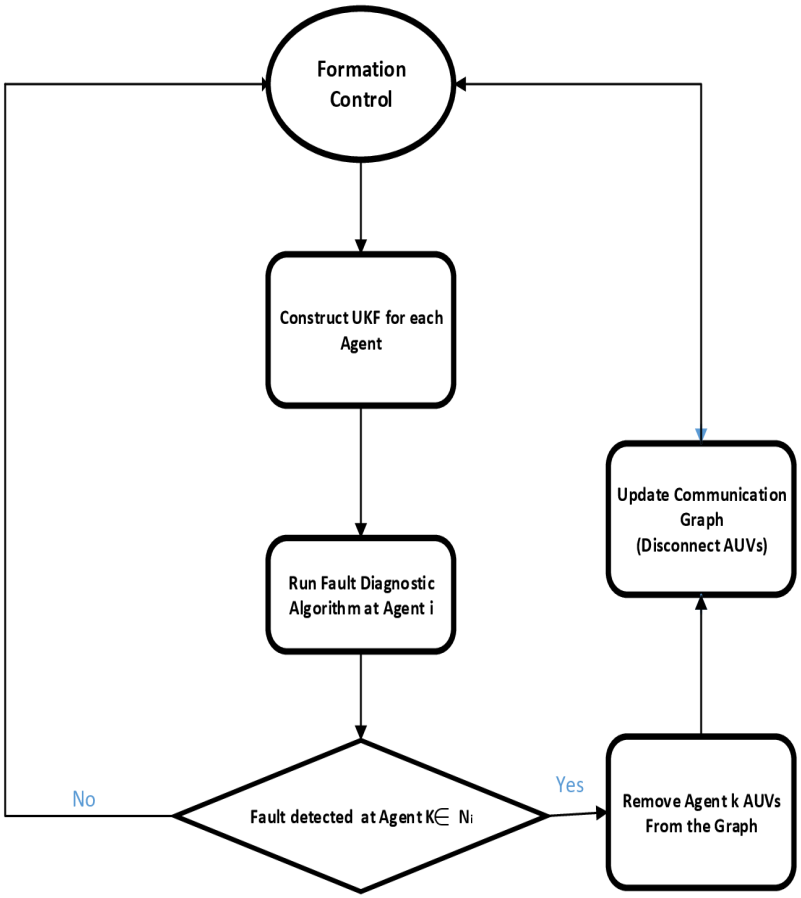


Figure 3.2: The FDI strategy for the formation control of a team of AUVs [13].

The results in [13] provide methods to remove the agent under fault or BIA from the network [13]. 1) The agent removed may correspond to either unexpected changes in the system, or 2) the removal of faulty agent or 3) agent under d. In three scenarios, it is desirable to maintain the detection capabilities of the distributed FD scheme despite the model changes. A distributed FD scheme does not require the full knowledge of the network [106].

3.2.2 Unscented Kalman Filters (UKF)

The Unscented Kalman Filter (UKF) are a variant of KF that is applied for nonlinear systems. The UKF uses the Unscented Transformation to approximate the probability density by deterministic sampling of points (called sigma points) representing the underlying distribution as a Gaussian process. The nonlinear transformation of these points are expected to represent the posterior distribution. The UKF tends to be more accurate and more robust as compared to the Extended Kalman Filter (EKF), particularly for highly nonlinear systems. An illustration of sampling and the calculation of mean and covariances for the Extended Kalman Filter (EKF) and UKF are shown in Figure 3.3. During the prediction step, the estimated state and covariances are augmented with the mean and covariances of the process noise. The complete procedure of UKF implementation is provided below [107].

The method below is calculating the nonlinear transformation. Consider propagating a random variable x (dimension L) through a nonlinear function $y = g(x)$. Assume x has mean $\bar{x}$ and covariance P_x. To calculate the statistics of y, a matrix $\mathcal{X}$ of $2L + 1$ sigma vectors $\mathcal{X}_i$ is given as follows:

$$\begin{aligned} \mathcal{X}_o &= \bar{x} \\ \mathcal{X}_i &= \bar{x} + (\sqrt{(L+\lambda)P_x})_i \quad i = 1, \ldots, L \\ \mathcal{X}_i &= \bar{x} + (\sqrt{(L+\lambda)P_x})_{i-L} \quad i = L+1, \ldots, 2L \end{aligned} \tag{3.7}$$

where $\lambda = \alpha^2(L + k) - L$ is a scaling parameter. The constant α determines the spread of the sigma points around $\bar{x}$. α is a positive small values, $(1 \leq \alpha \leq 1e-4)$, $\beta = 2$ and k is set

to 0 or $3 - L$. These sigma vectors are propagated through the nonlinear function,

$$\mathcal{Y}_i = g(\mathcal{X}_i) \quad i = 0, \ldots, 2L \tag{3.8}$$

and the mean and covariance for y are equal a weighted sample mean and covariance of the posterior sigma points,

$$\bar{y} \approx \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_i \tag{3.9}$$

$$P_y \approx \sum_{i=0}^{2L} W_i^{(c)} \{\mathcal{Y}_i - \bar{y}\}\{\mathcal{Y}_i - \bar{y}\}^T \tag{3.10}$$

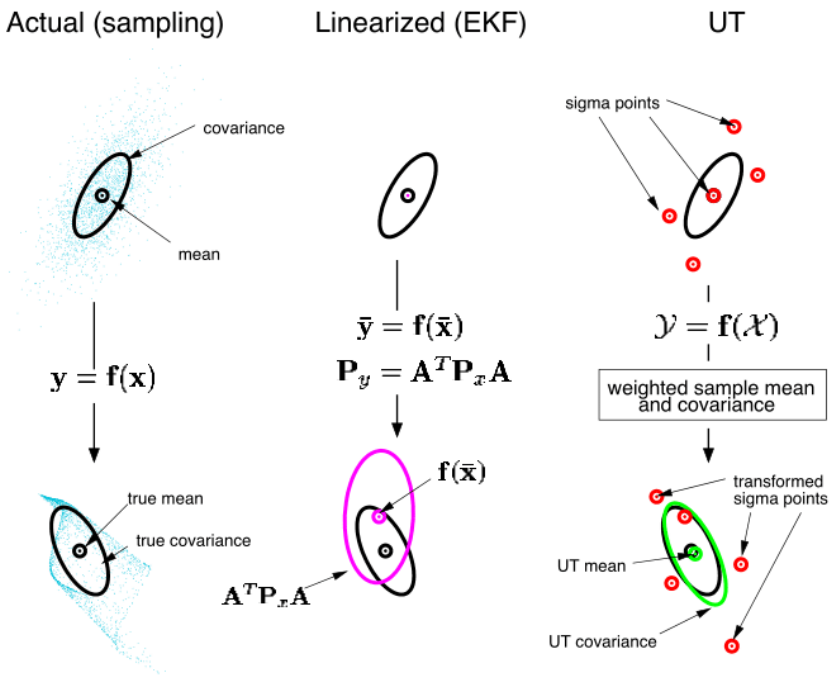


Figure 3.3: The computational procedure of UKF [107] .

Initialization:

$X_0 = [0]_{2L\times 1}$

$$\hat{X}_0 = \mathbb{E}[X_0] \tag{3.11}$$

$$P_0 = \mathbb{E}[(X_0 - \hat{X}_0)(X_0 - \hat{X}_0)^T] \tag{3.12}$$

$$\hat{X}_0^a = \mathbb{E}[X^a] = [\hat{X}_0^T \quad 0 \quad 0]^T \tag{3.13}$$

For $k \in \{1, \ldots, \infty\}$ time update in equation (3.1)

- Calculate sigma points:

$$\mathcal{X}_{k-1}^a = [\hat{X}_{k-1}^a \quad \hat{X}_{k-1}^a \pm \sqrt{(L+\lambda)P_{k-1}^a}] \tag{3.14}$$

Time Update:

- Evaluate the sigma points using the dynamical model in equation (3.1)

$$\mathcal{X}_{k|k-1}^x = f[\hat{X}_{k-1}^x, \mathcal{X}_{k-1}^v] \tag{3.15}$$

- Estimate the prediction state:

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{X}_{i,k|k-1}^x \tag{3.16}$$

- Estimate the error covariance:

$$P^{-}_{k|k-1} = \sum_{i=0}^{2L} W_i^c (\mathcal{X}^x_{i,k|k-1} - \hat{x}_{k|k-1})(\mathcal{X}_{i,k|k-1} - \hat{x}_{k|k-1})^T + R^v \tag{3.17}$$

Measurement Update:

- Evaluate the sigma points from the noisy measurement:

$$\mathcal{Y}_{i,k|k-1} = h(\mathcal{X}_{i,k|k-1}) \tag{3.18}$$

- Estimate the predicted measurement:

$$\hat{y}_{k|k-1} = \sum_{L=1}^{2n} W_i^m \mathcal{Y}_{i,k|k-1} \tag{3.19}$$

- Estimate the innovation covariance matrix:

$$P_{k|k-1} = \sum_{i=1}^{2n} W_i^c (\mathcal{Y}_{i,k|k-1} - \hat{y}_{k|k-1})(\mathcal{Y}_{i,k|k-1} - \hat{y}_{k|k-1})^T + R^W \tag{3.20}$$

- Estimate the cross-covariance matrix:

$$P_{xy,k|k-1} = \sum_{i=1}^{2n} W_i^c (\mathcal{X}_{i,k|k-1} - \hat{x}_{k|k-1})(\mathcal{Y}_{i,k|k-1} - \hat{y}_{k|k-1})^T \tag{3.21}$$

- Calculate the Kalman gain:

$$K_k = P_{xy,k|k-1} P^{-1}_{k|k-1} \tag{3.22}$$

- Estimate the update state:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_{k|k-1}) \tag{3.23}$$

- Estimate the error covariance:

$$P_{k|k} = P^{-}_{k|k-1} - K_k P_{k|k-1} K_k^T \tag{3.24}$$

where R^v= Process noise covariance, R^w=measurement noise covariance, $X^a = [X^T \quad v^T \quad w^T]^T$, $\mathcal{X}^a = [(\mathcal{X}^x)^T \quad (\mathcal{X}^v)^T \quad (\mathcal{X}^w)^T]^T$

3.2.3 Fault Detection Strategy

Residual generation is an integral part of the fault detection strategy. The residual signal $r_{i,k}$ is defined by equation (2.22) as

$$r_{i,k} = y_{i|k} - \hat{y}_{i|k}, \tag{3.25}$$

where $r_{i,k}$ represents the residual signal of agent i at time k. In case of no fault, the residual signal has a zero-mean Gaussian distribution with a constant covariance matrix $\mathcal{P}_{r_i} = H_i P_{i,k|k-1} H_i^T + R^T_{i,k}$, where $H_i = \partial h(x_{i,k|k-1})/\partial x_{i,k|k-1}$ is the linearized measurement matrix around the operation point $\hat{x}_{i,k|k-1}$. Hence, faults can be diagnosed by testing the

following hypothesis:

$$\begin{cases} \mathcal{H}_0 : r_{i,k} \sim \mathcal{N}(0, \mathcal{P}_{r_i}), \\ \mathcal{H}_1 : r_{i,k} \nsim \mathcal{N}(0, \mathcal{P}_{r_i}), \end{cases} \tag{3.26}$$

where $\mathcal{N}(0, \mathcal{P}_{r_i})$ is the Probability Density Function (PDF) of a Gaussian random variable with a mean of zero and a covariance of $\mathcal{P}_{r_i}$. In this thesis, we consider the above hypothesis test by checking the 'power' of the residuals $r_{i,k}^T \mathcal{P}_{r_i}^{-1} r_{i,k}$.

Compound Scalar Testing (CST)Algorithm

The CST is defined as follows:

$$\begin{cases} \text{Accept} \quad \mathcal{H}_0 \quad \text{if} \quad r_{i,k}^T \mathcal{P}_{r_i}^{-1} r_{i,k} \leq h, \\ \text{Accept} \quad \mathcal{H}_1 \quad \text{if} \quad r_{i,k}^T \mathcal{P}_{r_i}^{-1} r_{i,k} > h, \end{cases} \tag{3.27}$$

where h is a threshold value for the fault diagnosis and is greater than $\mathbb{E}[r_{i,k}^T \mathcal{P}_r^{-1} r_{i,k}]$ to reduce the false alarm rate. If the null hypothesis $\mathcal{H}_0$ is used accepted the fault diagnosis algorithm declares there is no fault in an agent i. On the other hand if $\mathcal{H}_1$ is accepted then there is a fault in the agent i.

Generalized Likelihood Ratio (GLR)Algorithm

Another important aspect of a general control system is to integrate the FDI scheme with a fault identification or estimation module in order to estimate the severity of a fault that has occurred in different components of the system such as sensors and actuators. A modified version of the GLR scheme is developed to estimate the severity of a sensor fault. The GLR

was initially proposed and subsequently modified in [108–111]. The GLR can be used instead of a maximum likelihood test.

Let us define $S_j^k(\theta_1)$ as in equation (3.28) according to

$$S_j^k(\theta_1) = \sum_{i=j}^{k} \ln \frac{p_{\theta_1}(y_i)}{p_{\theta_0}(y_i)}, \tag{3.28}$$

where θ_0 and θ_1 are the sets that characterize the distributions of observations, p_{θ_0} and p_{θ_1}, before and after the change. We now define g_k in the GLR algorithm that can be calculated by using the minimum magnitude of change as follows:

$$g_k = \max_{1 \le j \le k} \sup_{\theta_1} S_j^k(\theta_1). \tag{3.29}$$

The detection time t_a is the minimum value of k at which $g_k > h$, where h is the detector threshold and the magnitude jump $(\theta_1 - \theta_0)$ and t_a is calculated. The conditional maximum likelihood estimate for the change time is the value of j at which the maximum value of g_k is reached. Therefore, the conditional maximum likelihood change magnitude and time are given by

$$(\tilde{J}, \theta_1) = \arg \max_{1 \le j \le t_a} \sup_{\theta_1} S_j^k(\theta_1) = \sum_{i=j}^{t_a} \ln \frac{p_{\theta_1}(y_i)}{p_{\theta_0}(y_i)}, \tag{3.30}$$

and $\hat{t}_a = \tilde{J}$ [98]. The key point is that the GLR test is due to from the fact that only the mean value of these residuals may change when the analyzed system is affected by additive

failure and the test statistic is given by

$$g_k = \frac{1}{2\sigma_n^2} \max_{1\leq j\leq k} \frac{1}{k-j+1} \Big[\sum_{k}^{i=j} y_i \Big]^2 \underset{H_0}{\overset{H_1}{\gtrless}} \gamma \tag{3.31}$$

where σ_n^2 is the variance, while p_{θ_1} is assumed unknown, γ is the threshold, $\mathcal{H}_1$ and $\mathcal{H}_0$ are two hypothesis test from equation (3.27).

3.3 Difference between Faults and Bias Injection Signals

The FDI approaches can be applied for detecting negative impacts of cyber attacks on networked control systems. However, these tools might be exploited more successfully if we could figure out the similarities and differences between faults and attacks. Both faults and attacks can be developed at an unknown time instant and they cause unreliable variation in the behavior of physical systems. The faults and attacks can be modeled by additive signals in the discrete-time state space model. Faults and attacks possess inherently distinct features, making it difficult for traditional FDI techniques to be implemented. Deception attack signal is an intelligent signal in which the attacker has the knowledge of the system. Firstly, the most significant difference between a fault and an attack lies in that fault is shown as a phenomenon that occurs randomly in each component such as actuators, sensors, or communication channels of a system. Moreover, simultaneous faults are suggested to be non-colluding while cyber attacks could be performed in a coordinated way. For these reasons,

cyber-physical attacks may cause more destructive damage to the system than faults.

Another point is that cyber-physical attacks are much more difficult to detect and isolate since they can be implemented in a coordinated way. The attack vectors can be controlled for partially or completely bypassing traditional irregularity detectors [65, 66], false data injection attacks [67], zero-dynamics attacks or covert attacks [68].

Faults often occur for a long time until they are detected, isolated and repaired, but malicious attacks may be performed within a short time period due to the limited resources of the adversaries [69, 70]. Moreover, for safety-critical applications, it is required to detect the attacks with the detection delay upper bounded by a certain prescribed value [71–73]. For these reasons, the detection and identification of attacks should be formulated as the sequential detection and isolation of transient changes in stochastic dynamical systems. Norm-based state estimator is applied in [74] to guarantee the boundedness of estimation errors caused by noise, modeling errors as well as cyber-attacks. Moreover, the deployment of bad data detectors results in a bound constraint of attacks. Certain issues about state estimation on stochastic time-varying nonlinear systems subject to randomly occurring BIA have been investigated in the framework of Kalman filtering in [75].

3.3.1 Bias Injection Attack Model

The BIA can be modeled as follows:

$$u_a(k) = u_i(k) + B_a a_u(k), \tag{3.32}$$

where the control input u_i, $a_u(k) \in \mathbb{R}$ represents the signal that the attacker injects. The vector B_a denotes the attack signature and is therefore directly dependent on the attacker. In the proposed deception each agent is equipped with a UKF that estimates its states, and then produces a residual for each output measurement. These residuals are normally enough to detect and isolate sensor and actuator faults of agents. Here, it is assumed that the attacks are acting on the communication models of agents as shown in Figure 3.1. The set $\mathcal{N}_i$ is the defined UKF that estimate states of the neighbouring agents. The innovation $r_{i,k}$ is defined as follows:

$$r_{i,k} = y_{i|k} - \hat{y}_{i|k}, \tag{3.33}$$

Remark 3.3.1 *By sending the innovation signal $r_{i,k}$ rather than the measurement $y_{i|k}$ or the local estimate $\hat{y}_{i|k}$ the innovation $r_{i,k}$ will approach a steady state distribution that can be easily checked by a false data detector [112].*

It is also assumed that each agent measures the relative measurement of the neighbouring agents, the relative measurement of the sensor, and also the difference between the actual neighbouring agent's measurement and the estimated neighbouring agent's measurement residual signals denoted by $r_{i,j}$ is given by

$$r_{i,j} = \mathsf{y}_{i|k} - \hat{\mathsf{y}}_{i|k}. \tag{3.34}$$

where $\mathsf{y}_{i|k}$ (actual relative measurement) and $\hat{\mathsf{y}}_{i|k}$ (estimated relative measurement) are presented in Figure 3.4. The idea is that each agent detects and isolates the fault and attack by looking at $r_{i,k}$ and $r_{i,j}$ signals. Whenever there is a fault in any agent i, both $r_{i,k}$, $r_{i,j}$ and

$r_{j,i}$ ($j \in \mathcal{N}_i$) are affected but when BIA happens it only affects the $r_{i,j}$ but not $r_{i,k}$. Therefore, with a proper inference system, faults and BIA would be distinguished. However, this problem gets more complicated with different assumptions on the attacker. Depending on the amount of information the attacker has from the agent (both model and current state), different scenarios are possible. In the most extreme case the attacker can produce $(u_i, r_{i,k})$ pair that appears normal and in line with the sensor when the agent i is faulty. This BIA can only happen when the attacker has prefect information of the agent i.

Figures 3.4 and 3.5 illustrate the overall framework for five agents, where important consideration are as follows:

- **Privacy**: Only communicating the innovation ($r_{i,k}$) rather than the state estimate($\hat{x}$),
- **Analytical redundancy**: the model of the agent is used.

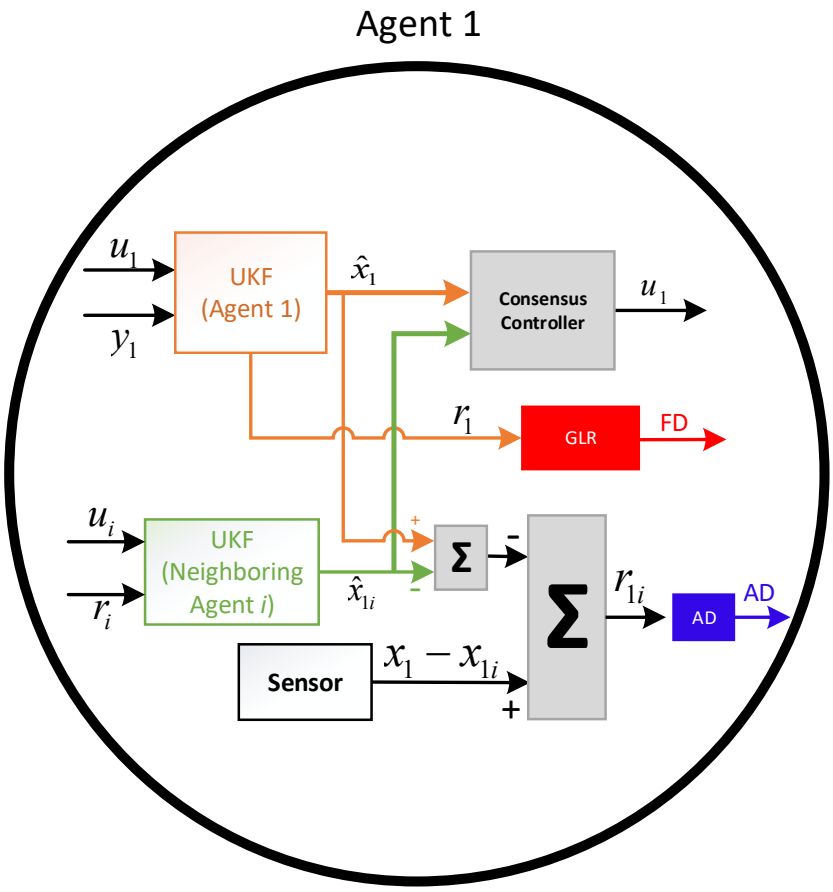


Figure 3.4: Overall framework of fault and attack diagnosis for Agent 1 where FD denotes Fault Detection and AD denotes Attack Detection.

As can be seen in Figure 3.4, a UKF is designed to detect the fault for an individual agent. Multiple UKFs are deigned for neighbouring agents to detect a BIA. One sensor is considered to receive actual relative information. We receive the relative residual signal $r_{i,j}$ for detecting a BIA among the commutations of every agent.

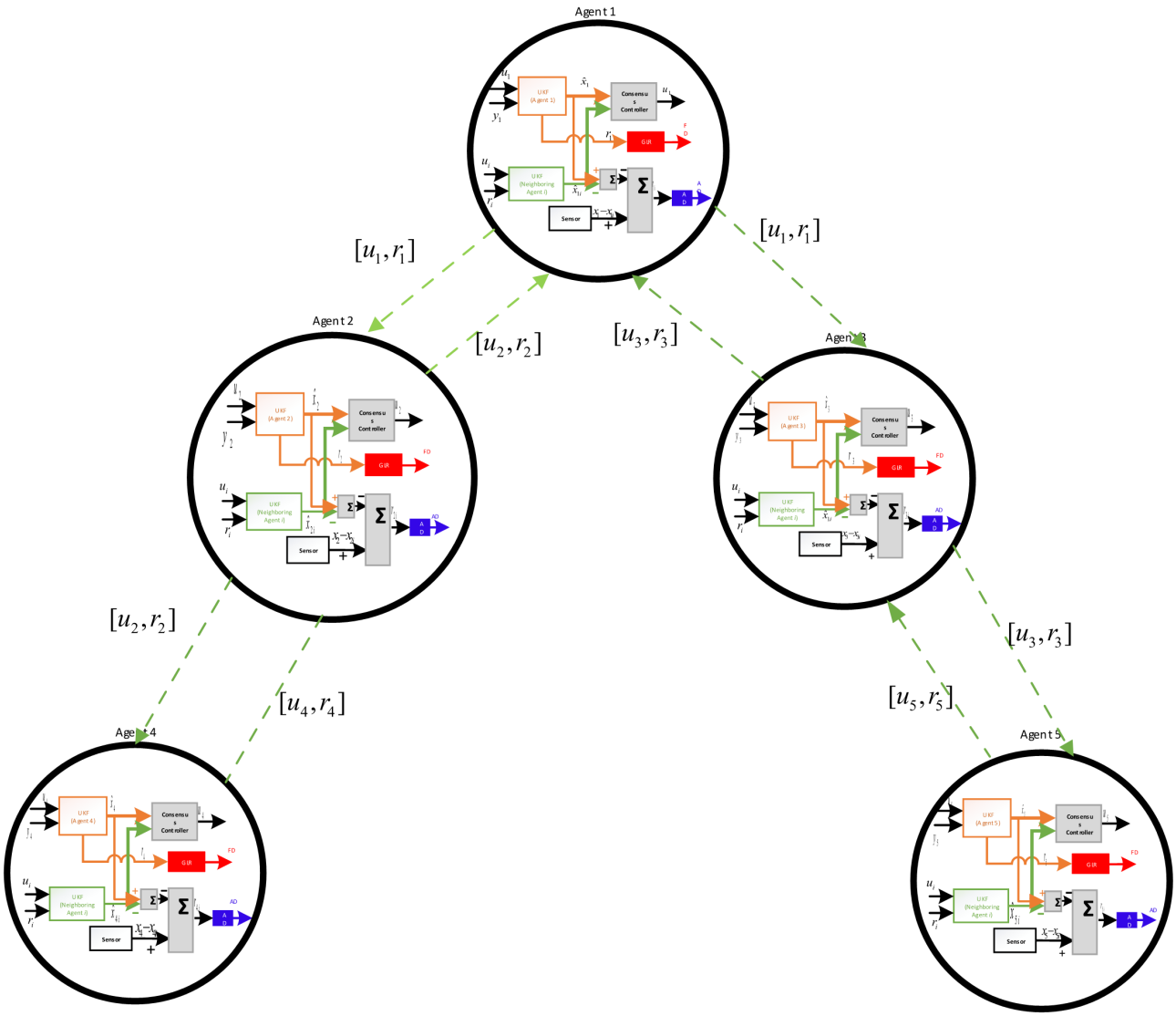


Figure 3.5: Overall framework of faults and attacks diagnosis for five agents.

3.4 Simulation Result

In this section, simulation results and performance evaluation for the formation of nonlinear AUVs given Section in 2.3.3 and FD scheme for different scenarios are presented. The system with no fault is presented in Section 3.4.1, while in Section 3.4.2 the system with faults and

the applied FD method are given. The AUV system with BIA and also the incorporated fault are provided in Sections 3.4.4 and 3.4.5. The formation of nonlinear AUVs is in a regular triangle that is given in Figure 3.5. Without loss of generality details of the dynamical model parameters of each AUV is indicated in Table 3.1 where the sets of model parameters are taken from [9].

Table 3.1: AUV Parameters [9].

Total mass and interim matrix (kg)	$M = diag[200\ 250\ 80]$
Linear drag matrix (Ns/m)	$D = diag[170\ 100\ 50]$
Sampling time (second)	0.2

$$C(v) = \begin{bmatrix} 0 & 0 & -250v_v \\ 0 & 0 & 200v_u \\ 250v_v & -200v_u & 0 \end{bmatrix}, \tag{3.35a}$$

$$\mathcal{L} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix} \tag{3.35b}$$

Moreover, Table 3.2 summarizes the initial values for each agent and equations (3.35a) and (3.35b) given $C(v)$ and the Laplacian matrix $\mathcal{L}$, respectively.

In addition, the UKF parameters are set as $\alpha = 1e-2$, $\beta = 2$, $\kappa = 3-n$ and $n = 6$,

Table 3.2: Initial Conditions.

	$[\rho_x(0)\ \rho_y(0)\ \rho_\psi(0)\ v_u(0)\ v_v(0)\ v_r(0)]$
AUV_1	$[0\ 0\ \frac{\pi}{4}\ 0\ 0\ 0]$
AUV_2	$[-1\ -1\ \frac{\pi}{4}\ 0\ 0\ 0]$
AUV_3	$[-1\ 1\ \frac{\pi}{4}\ 0\ 0\ 0]$
AUV_4	$[-2\ -2\ \frac{\pi}{4}\ 0\ 0\ 0]$
AUV_5	$[-2\ 2\ \frac{\pi}{4}\ 0\ 0\ 0]$

and we assume that the mission starts at $t = 0$ and will be terminated at time $t = 2000$ sec.

3.4.1 No Fault Scenario

The simulation results of the formation trajectories of all the five AUVs in a normal operation (with no fault) is given in Figure 3.6. The goal of formation is that a group of AUV maintain a desired formation (triangle).

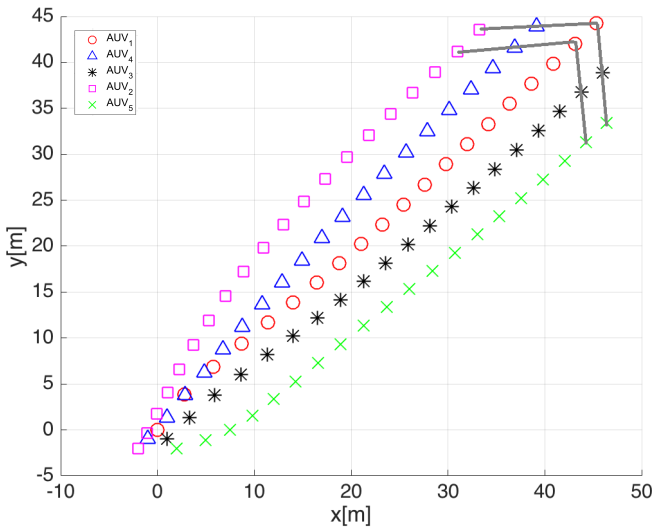


Figure 3.6: Formation trajectories for healthy team.

Figures 3.7 illustrate the estimated orientation of each agent with no fault, respectively.

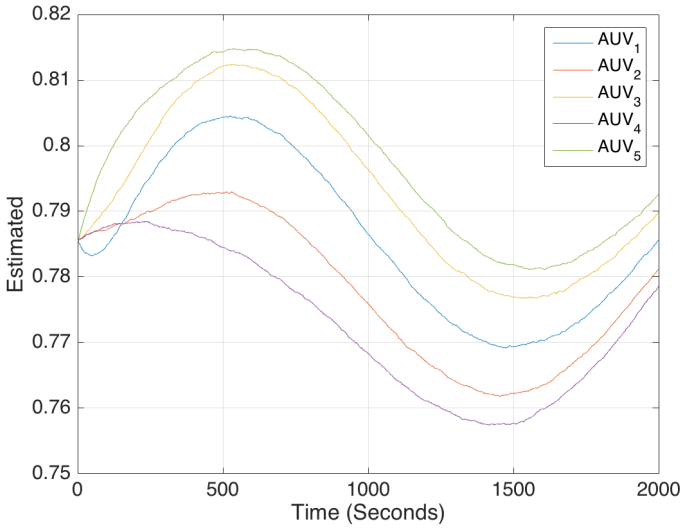


Figure 3.7: The estimated trajectories corresponding to agents 1 to 5 in case of no fault.

Figure 3.8 shows the control input for all the five agents in case of healthy system.

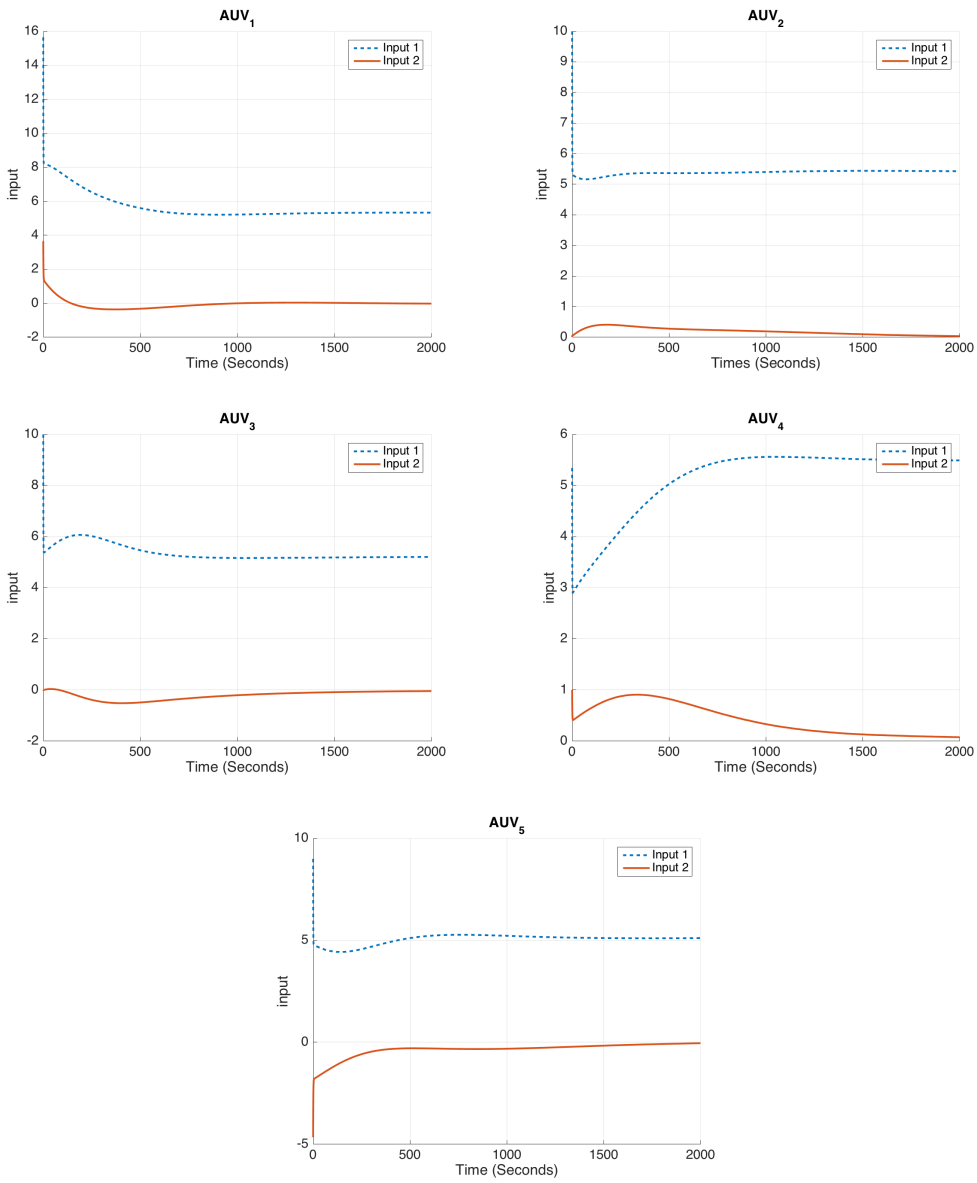


Figure 3.8: Control input signals corresponding to all five agents in case of no fault.

Figure 3.9 shows residual signals $r_{i,k}$ in equation (3.25) generated for all five agents. As can be seen, the residual signal is almost zero (with some random noise) for the healthy team.

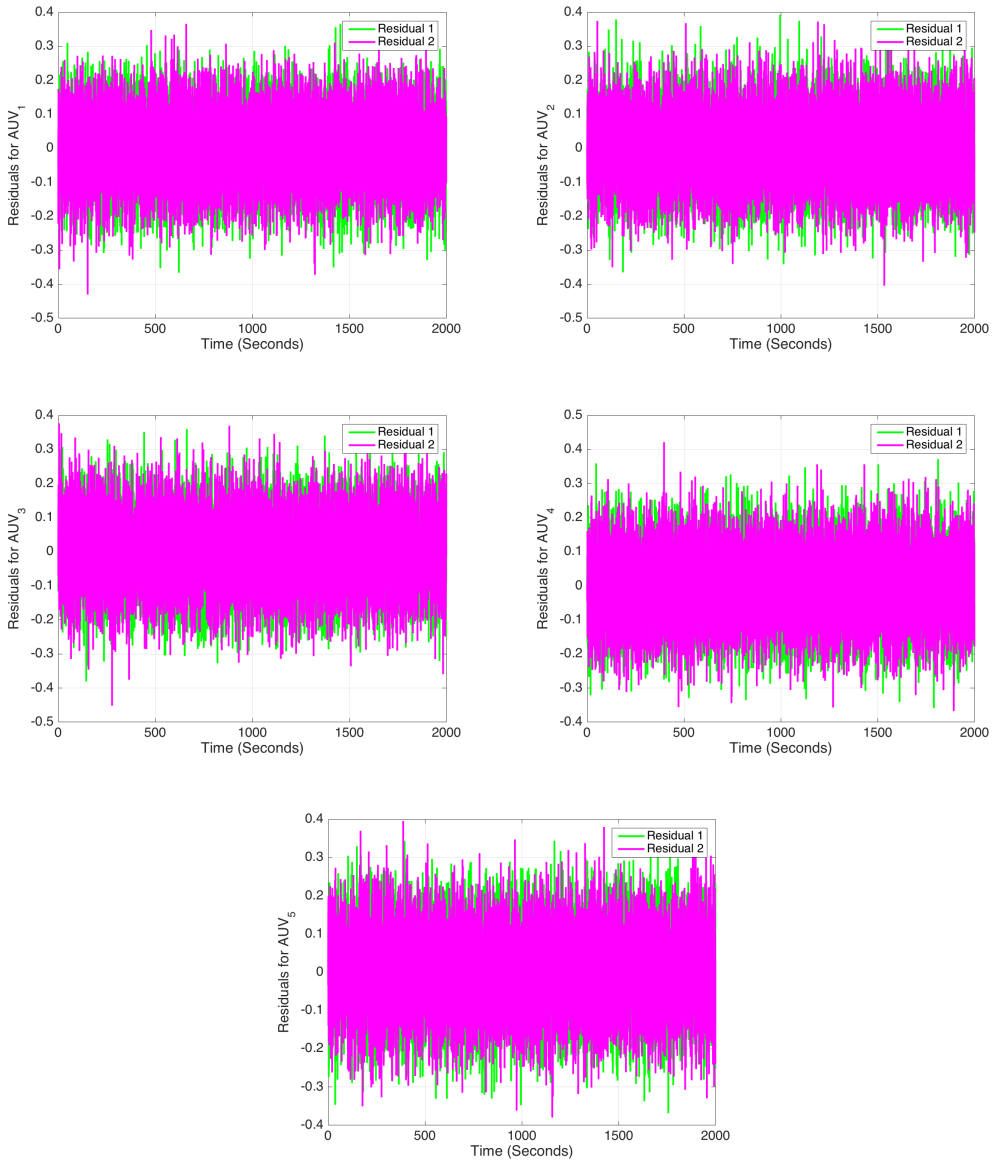


Figure 3.9: Residual signals ($r_{i,k}$) corresponding to all the five agents in case of no fault.

Figure 3.10 shows the g_{max} value as defined in equation (3.29). As can be seen, g_{max} values are noisy for all the agents.

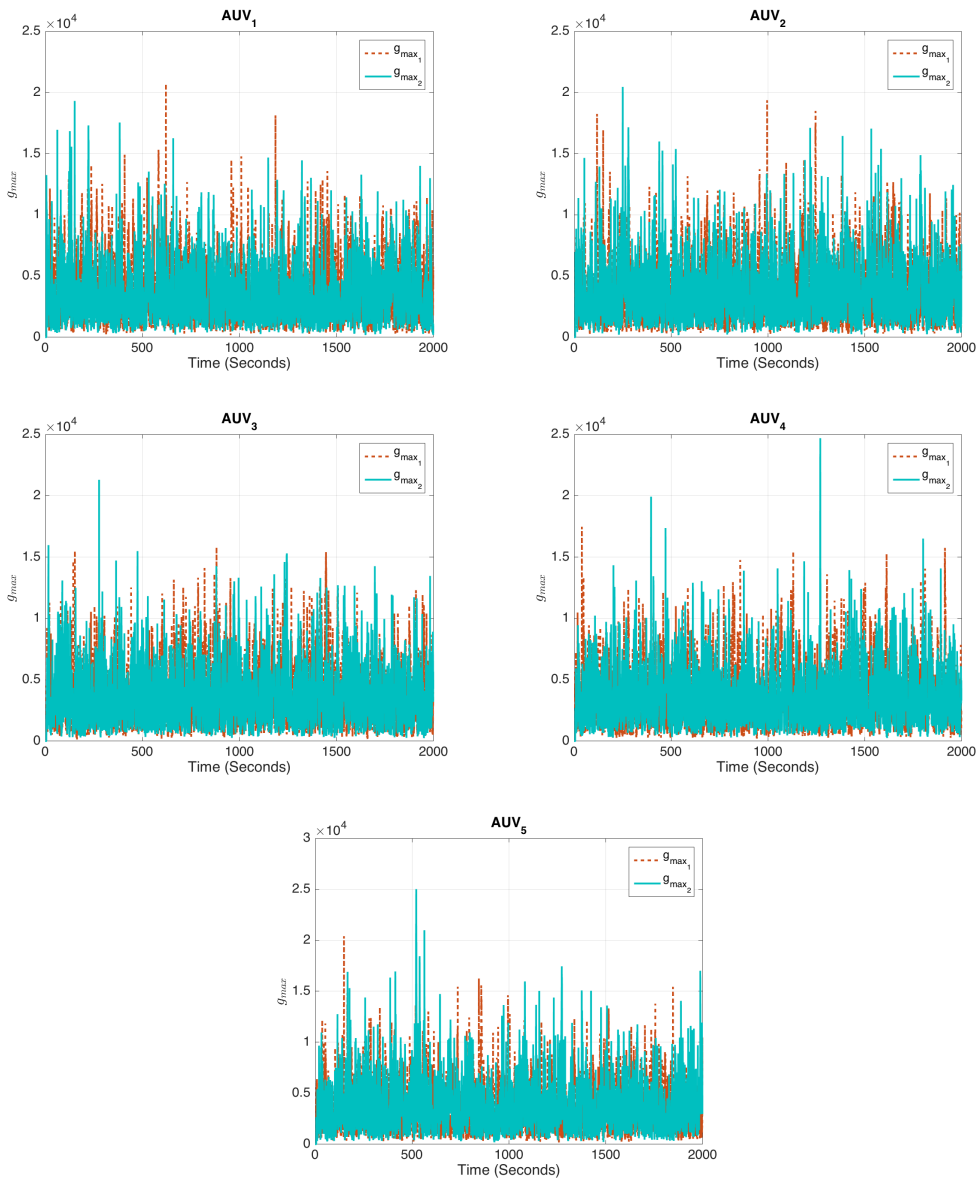


Figure 3.10: The result of GLR algorithm corresponding to all five agents in case of no fault.

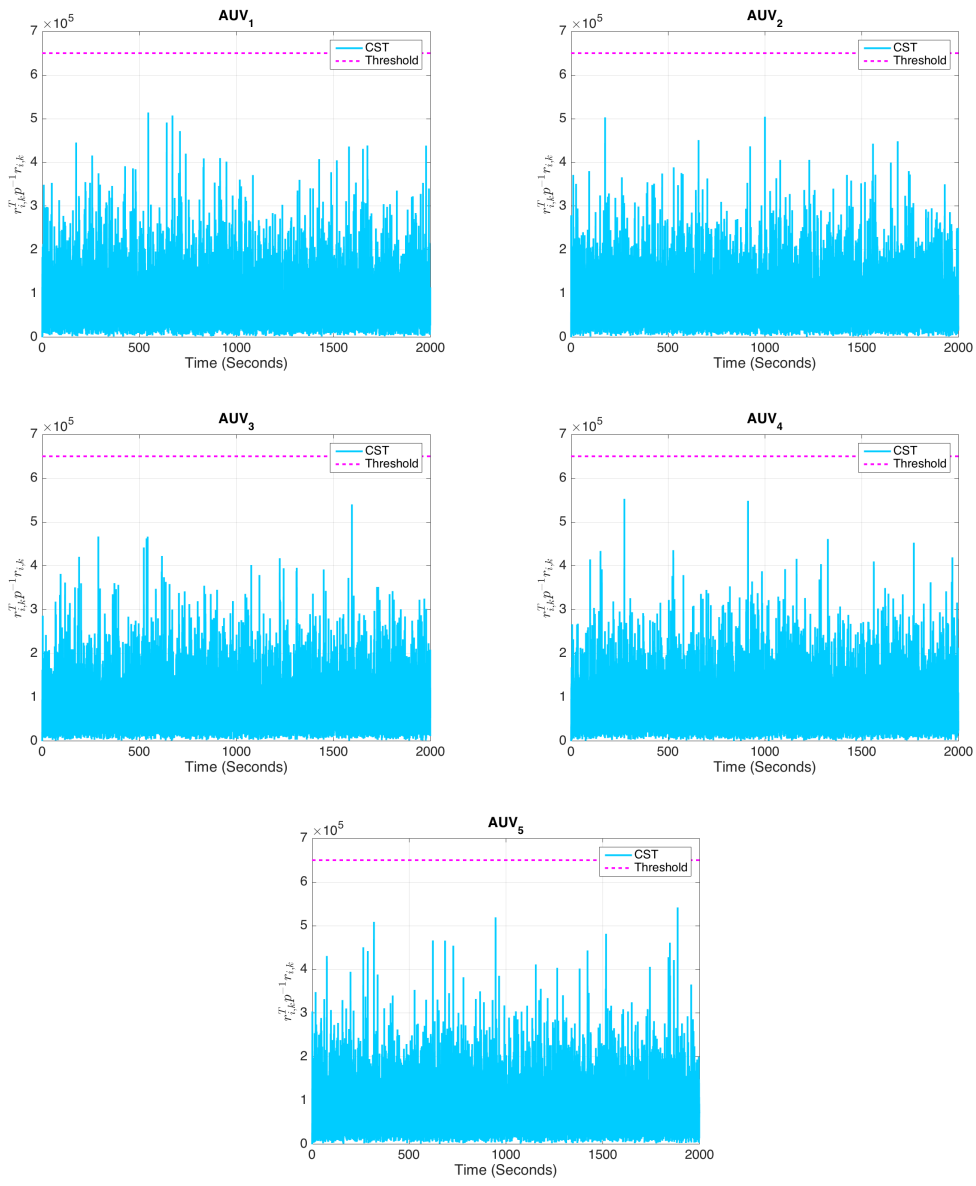


Figure 3.11: CST for all five agents in case of no fault.

Figure 3.11 shows the CST in equation (3.27), when the system is healthy as we can see that the CST is under threshold h. The simulation results of fault detection in equation

(3.36) equals zero for a team of healthy agents are presented in Figure 3.12.

$$FD = \begin{cases} 1 & \text{faulty} \\ \\ 0 & \text{no fault} \end{cases} \tag{3.36}$$

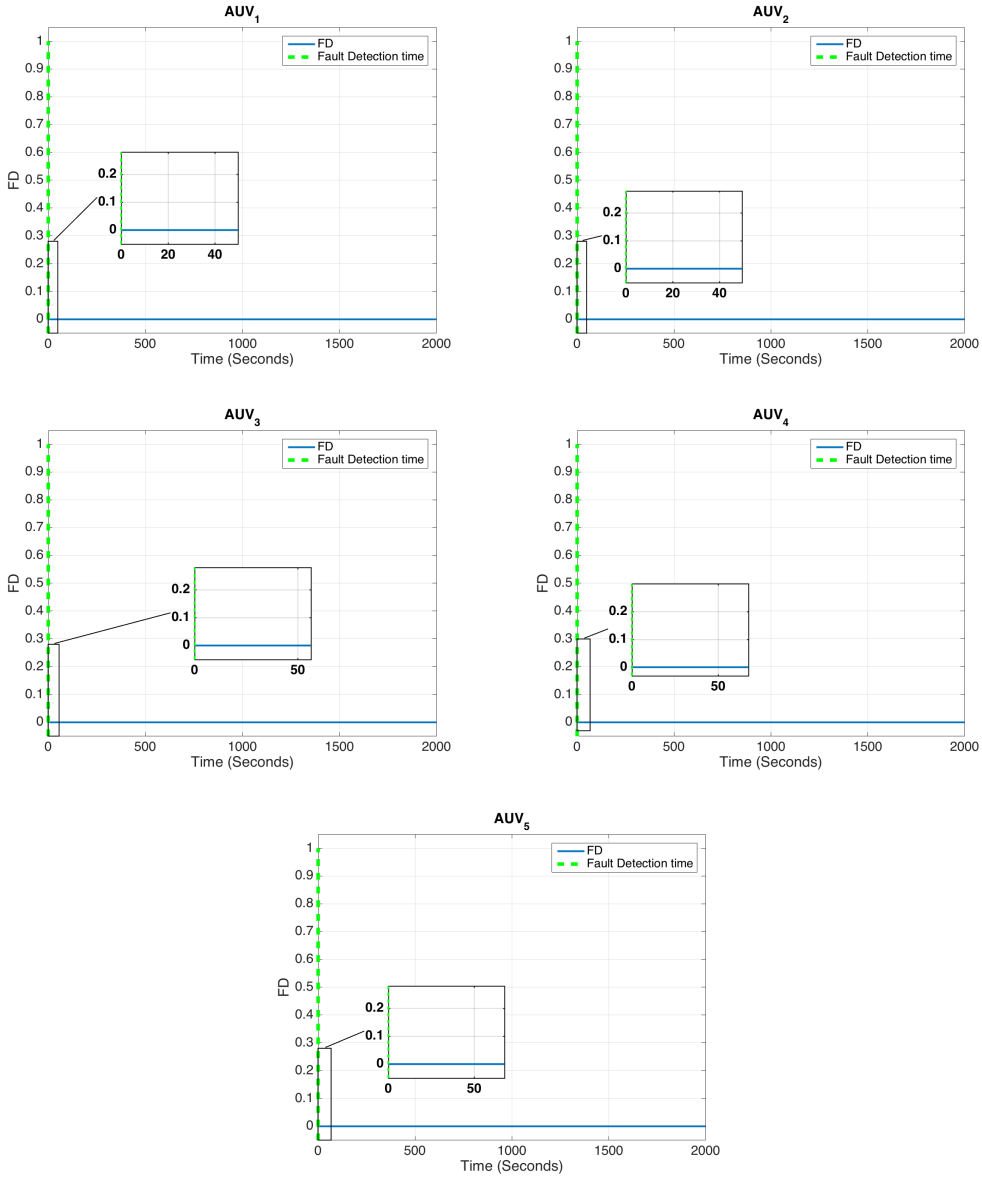


Figure 3.12: The fault detection simulation results for all five agents in case of no fault.

3.4.2 Actuator (LOE) and Sensor Fault

Figure 3.13 shows that both Agents 1 and 4 are under 90% LOE fault for the first actuator. Moreover, Agent 2 experiences 45 degree bias sensor fault. In addition, the sensor fault is injected to Agent 5 with an offset of $10m$.

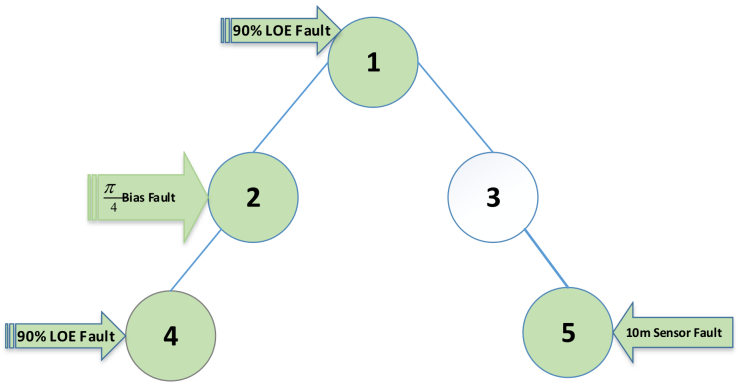


Figure 3.13: All AUVs under various faults.

As can be seen in Figures 3.14, the formation trajectories for four of AUVs, namely Agents 1, 2, 4, and 5 do not follow triangular formation due to the presence of faults.

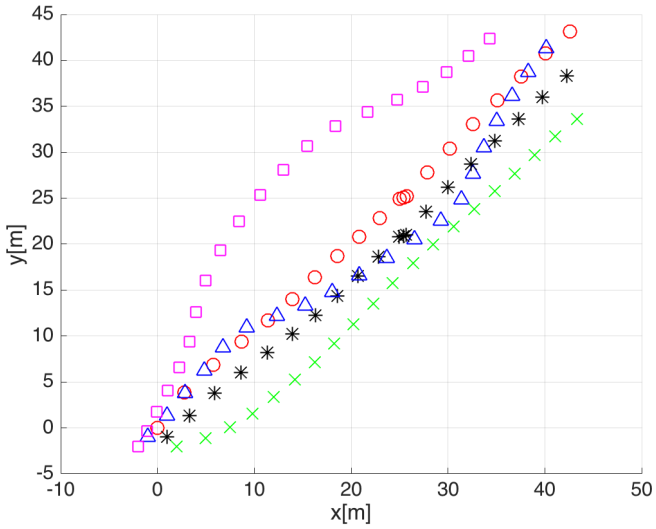


Figure 3.14: Formation trajectory in the presence of fault.

Figure 3.15 illustrates the estimated orientation of each of agents in presence of faults.

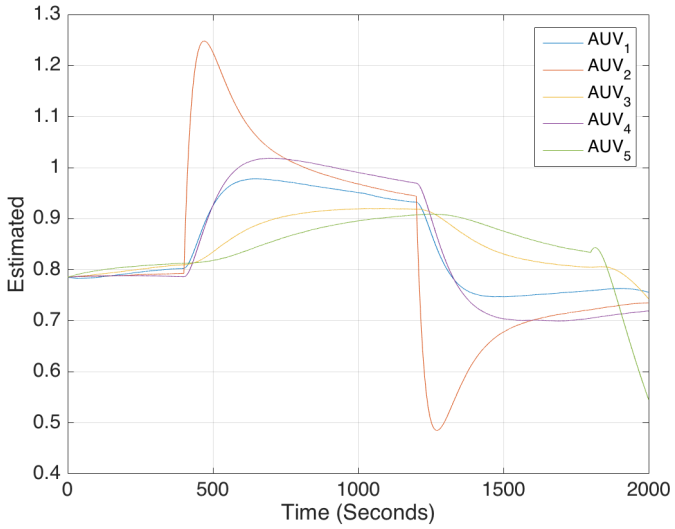


Figure 3.15: The estimated trajectories ($\hat{x}$) corresponding to all five agents in presence of faults.

Figure 3.16 shows the measured trajectories. As can be seen, Agent 2 has a jump around $t = 400\,\text{sec}$ due to a 45 degrees sensor bias fault that lasts until $t = 1200\,\text{sec}$. Note that horizontal AUVs are considered here.

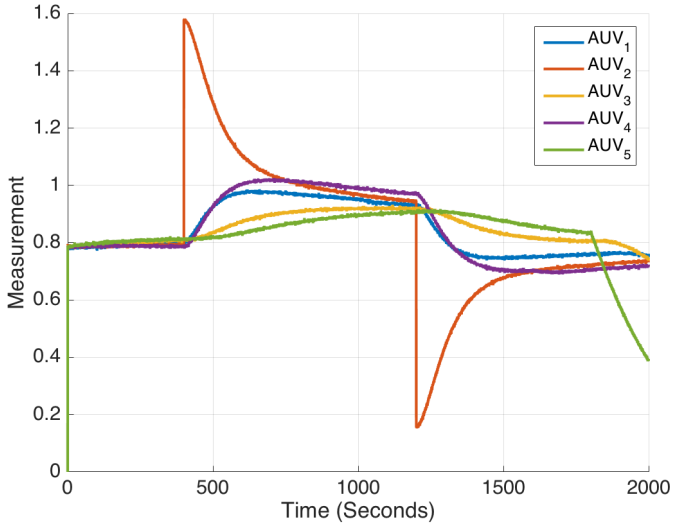


Figure 3.16: The measured trajectories corresponding to all five agents in presence of faults.

Figure 3.17 shows the control inputs for all the five faulty agents. As noted, controller has more oscillation when compared to the healthy system in Figure 3.8.

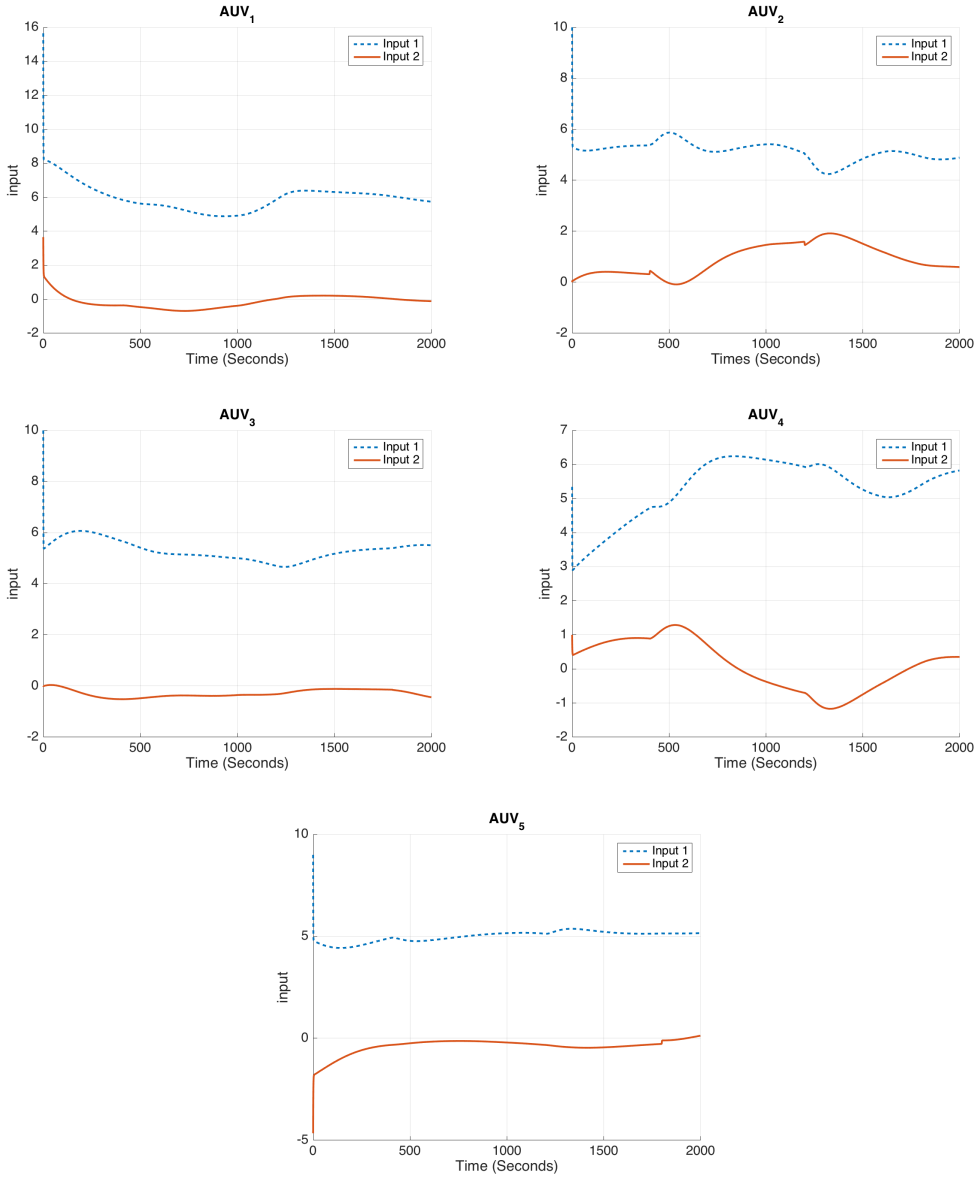
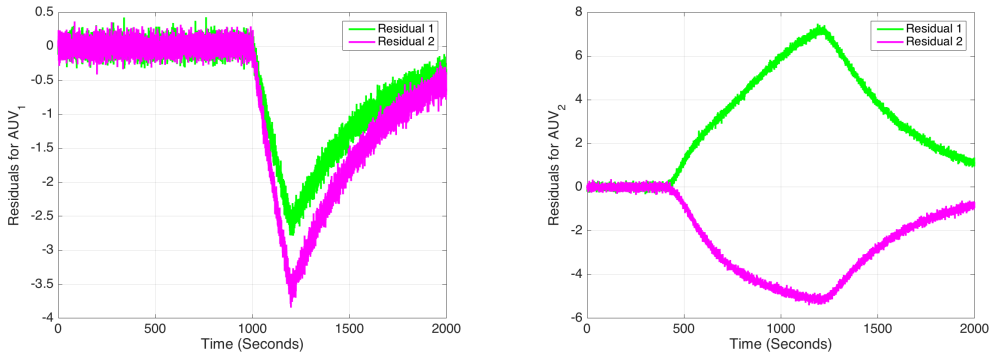
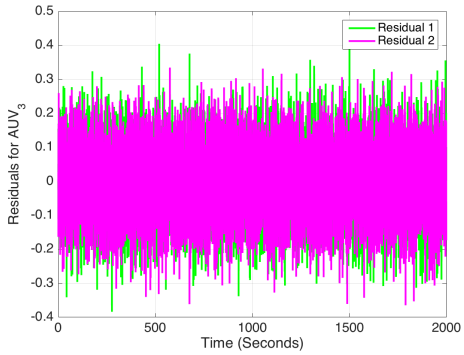


Figure 3.17: Control input for each agent in presence of faults.

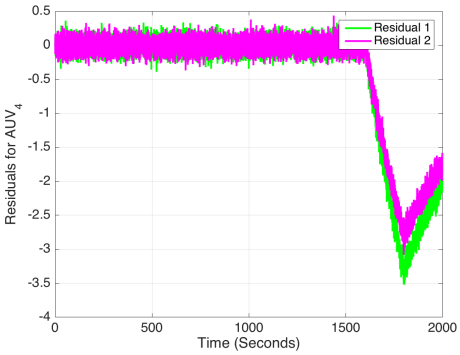


(a)

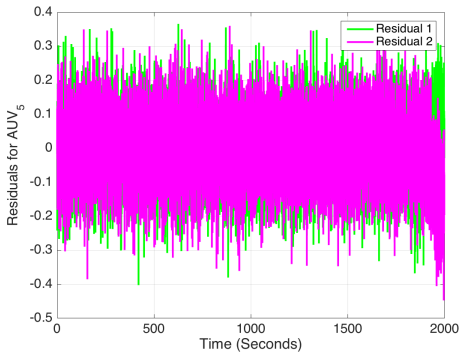
(b)



(c)



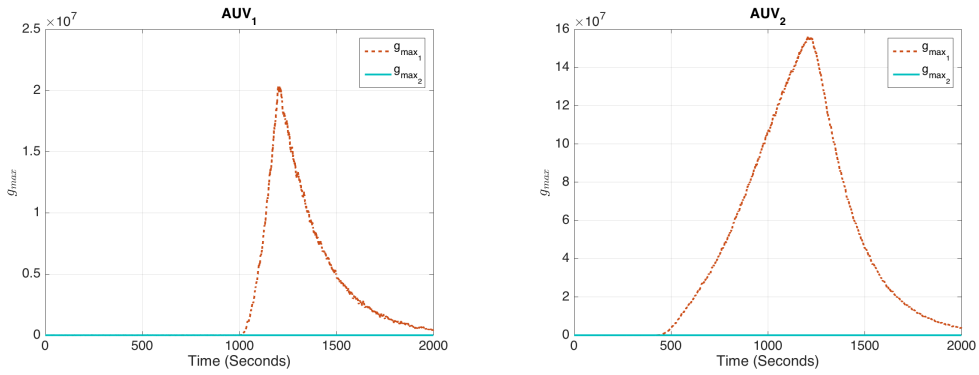
(d)



(e)

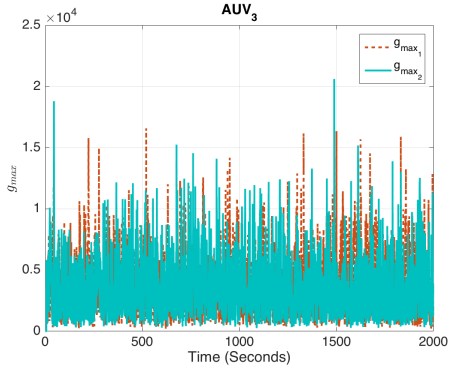
Figure 3.18: (a) Residual signal $r_{i,k}$ for Agent 1 (first actuator 90% LOE), (b) $r_{i,k}$ for Agent 2 (45 degree bias sensor fault), (c) no fault, (d) Residual signal $r_{i,k}$ for Agent 4 (first actuator 90% LOE), (e) $10m$ sensor fault

The residual signals $r_{i,k}$ that are generated for all agents are illustrated in Figure 3.18 and residual signals corresponding to all the five agents due to 90% LOE fault for the first actuator Agent 1 and Agent 3, Agent 2 experiencing a 45 degree bias sensor fault, and Agent 5 experiencing a sensor fault of $10m$ offset.

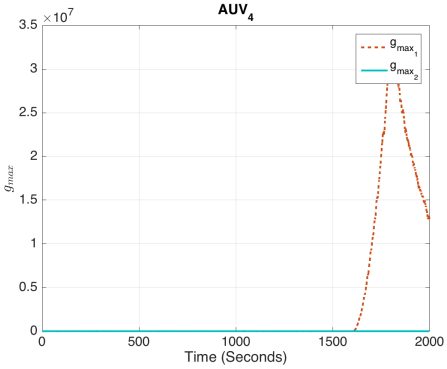


(a)

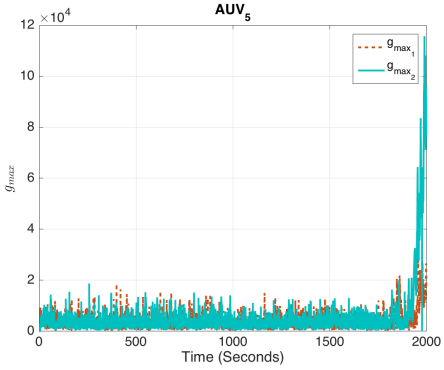
(b)



(c)



(d)



(e)

Figure 3.19: GLR algorithm corresponding to Agents 1 to 5 with the presences of faults.

Moreover, g_k value (GLRs algorithm) for all the agents are provided in Figure 3.19. As

can be seen in Figure 3.19 the g_k are not the same. Moreover, CSTs are above false alarm rate for the faulty Agents 1, 2, 3, 5 as shown in Figure 3.30.

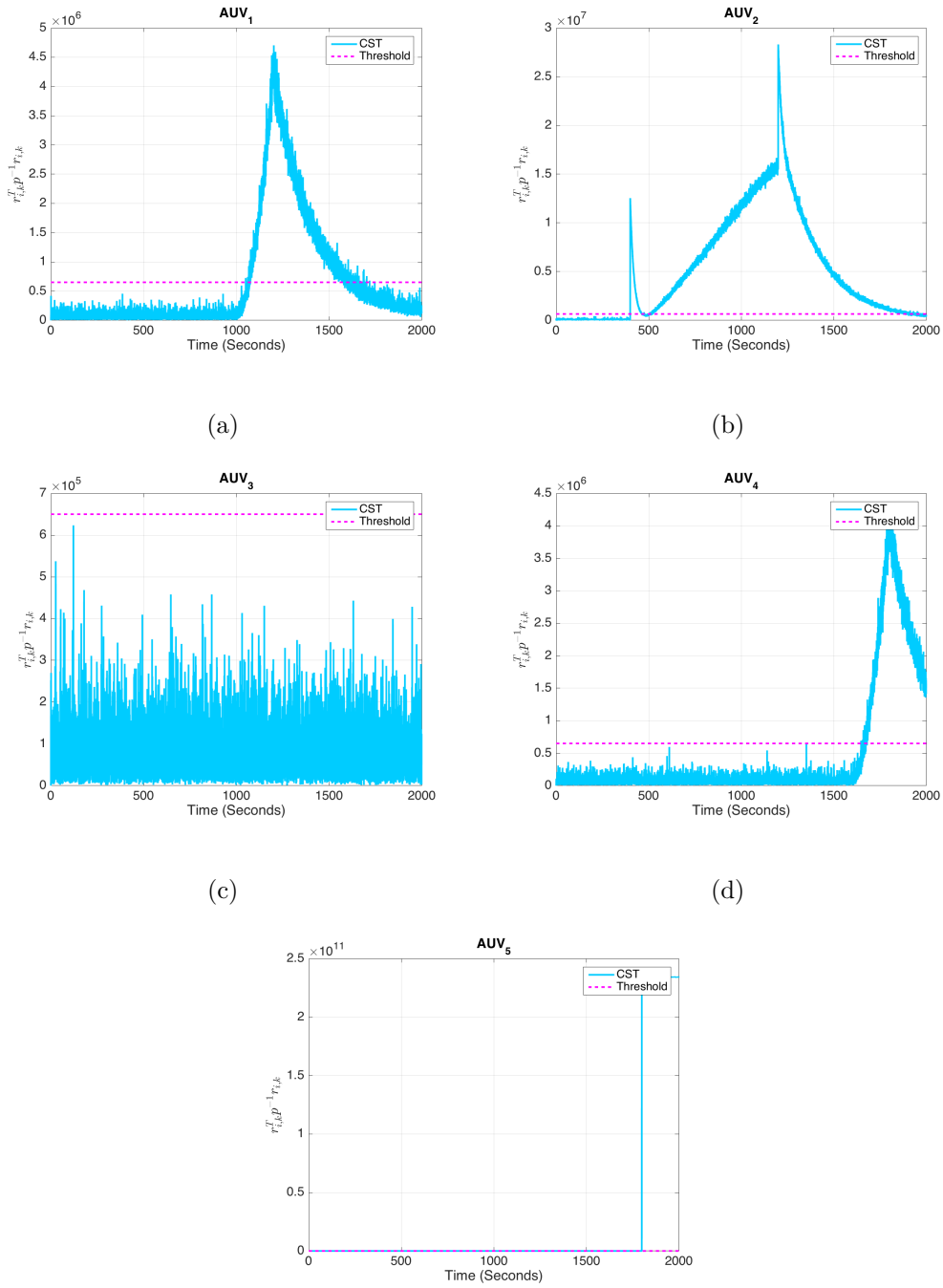
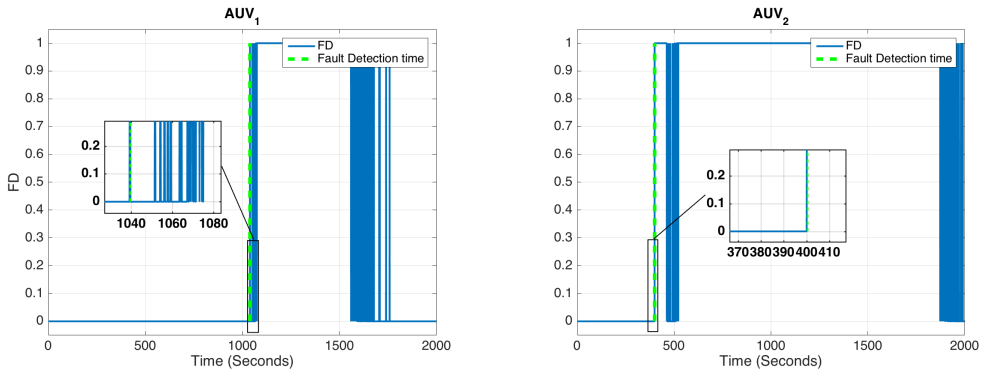


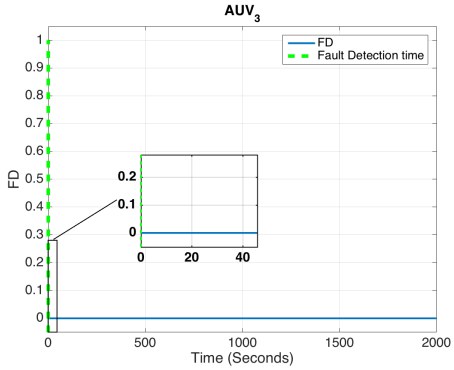
Figure 3.20: The CST for all the agents when the system is under fault.

Figure 3.20 illustrates the result of fault detection for all agents.

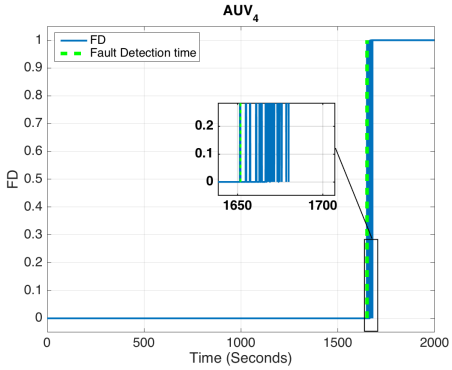


(a)

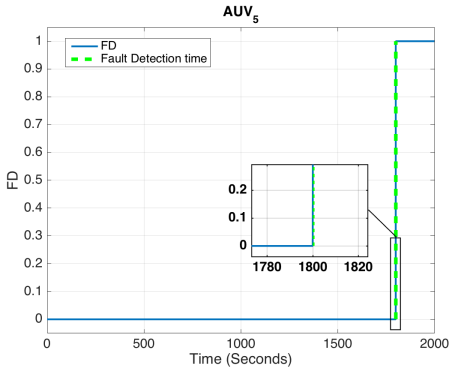
(b)



(c)



(d)



(e)

Figure 3.21: Fault detection (FD) for all agents.

In case of Agent 1 experiences 90% LOE fault, the fault detection time equals

t_d (fault detection time) = 39.2 sec (196 samples × 0.2 Step Time). In case of Agent 2 experiencing 45 degree bias sensor fault, the fault detection time equals $t_d = 0.4$ sec. Moreover, when Agent 4 experiences a 90% LOE for the first actuator, $t_d = 44$ sec. The fault detection time $t_d = 0.4$ sec when Agent 5 experiences a $10m$ offset sensor fault is given in Table 3.3.

Table 3.3: Fault detection time when the system is under fault where the sampling period is 0.2 sec.

Number of Agent	Type of Abnormal Event	When a fault is started	When a fault is terminated	When a fault is detected	Fault Detection Time [Samples]
Agent 1	First Actuator 90% LOE Fault	5000	6000	5196	196
Agent 2	45 degree Bias Sensor Fault	2000	6000	2002	2
Agent 4	First Actuator 90% LOE Fault	8000	9000	8220	220
Agent 5	Bias Sensor Fault $10m$	9000	10000	9002	2

3.4.3 No Fault and No Attack

The relative residual signals measured for Agent 2 from Agent 1 and measured Agent 2 from Agent 4 are close to zero. Figures 3.22 and 3.23 illustrate the results where there is no fault and no attack.

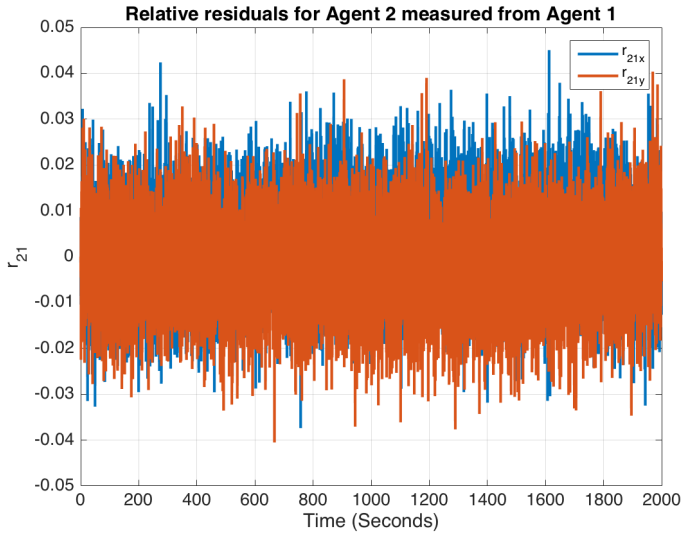


Figure 3.22: Relative residual signal for Agent 2 measured from Agent 1 for the healthy system.

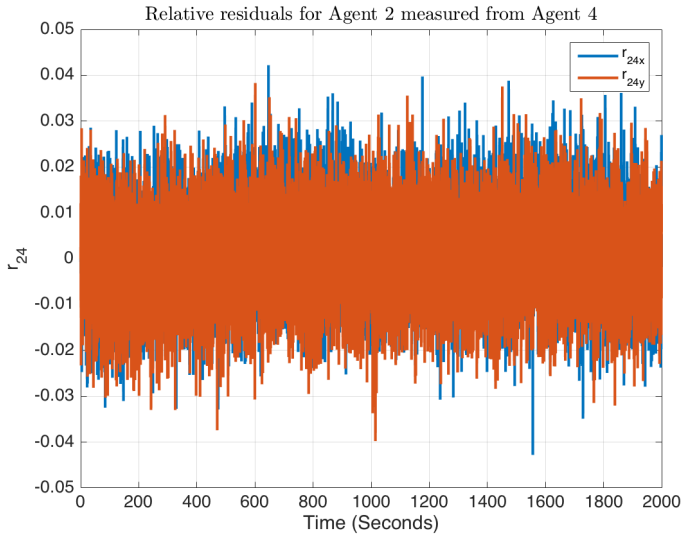


Figure 3.23: Relative residual signal for Agent 2 measured from Agent 4 for the healthy system.

We can conclude that the residual signal $r_{i,j}$ in equation (3.34) is equal to zero for the healthy system. The residual signals r_{21} and r_{24} show Agent 2 is under attack and Agents 1 and 4 are neighbours (r_{21} and $r_{24} = 0$ for healthy system). The residual signals r_{21} and r_{24} that are generated from the developed detection algorithm are shown in Figure 3.22. In this case, the position (x, y) measured from Agent 2 for the sensor and the relative states are equivalent as shown in Figure 3.24.

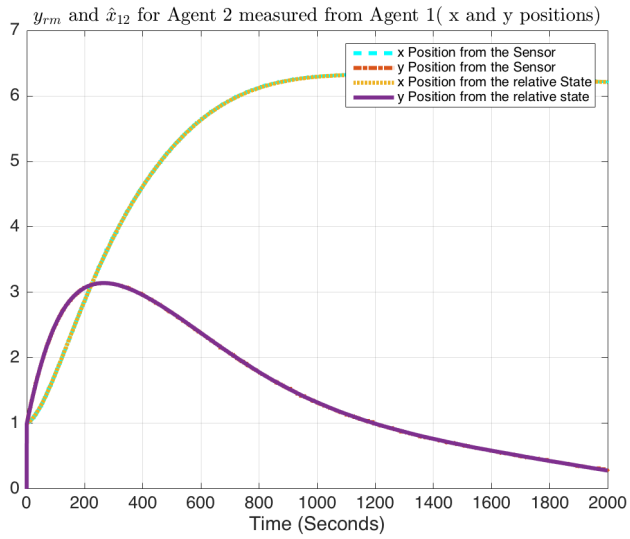


Figure 3.24: The position (x, y) from the sensor and relative state measured from Agent 2 for the healthy system.

3.4.4 Injection of the Bias Injection Attack Case

In Figure 3.27 the x and y position signals are getting away from each other and both are diverging after the attack signal was injected at $t_d = 400\,\text{sec}$. Consequently, r_{21_x} and r_{21_y} in Figure 3.28 are increasing with r_{21_y} diverging at a faster pace. This illustrates the case where an agent in the team of AUVs suffers from a BIA (modeled based on equation (3.32)). By

observing both Figures 3.28 and 3.29, when Agent 2 is under an attack, the relative residuals for Agent 2 measured from Agents 1 and 4 are NOT close to zero.

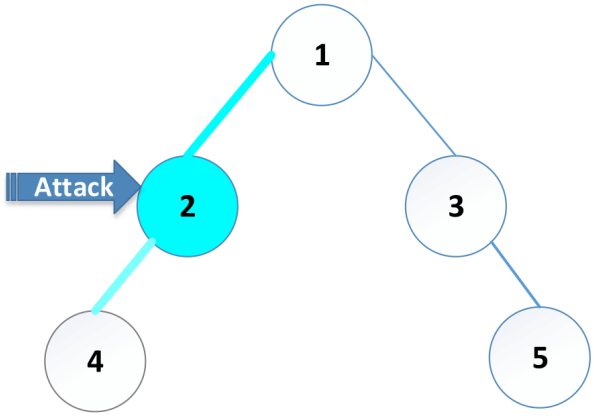


Figure 3.25: All AUVs are in formation when Agent 2 is under a BIA.

In this case, we can see that the Agent 2 does not follow the formation trajectories as shown in Figure 3.26.

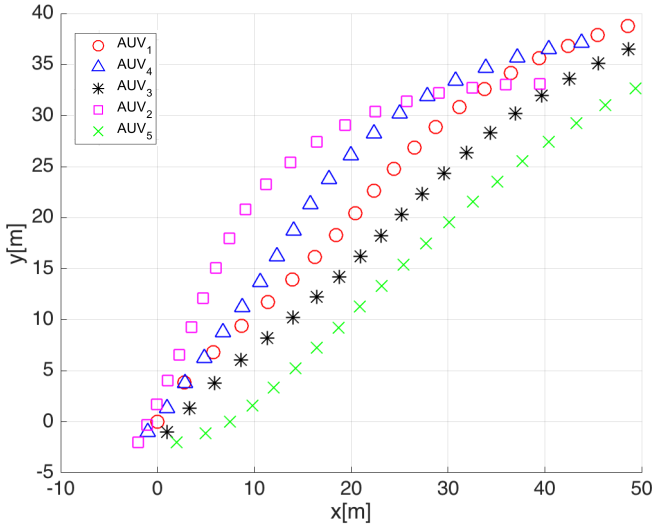


Figure 3.26: Formation trajectories of the five AUVs in presence of a BIA in Agent 2.

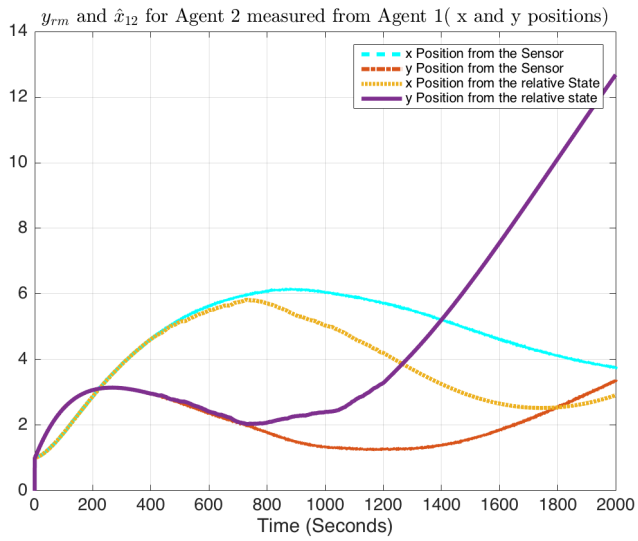


Figure 3.27: The position (x, y) from the sensor and relative state for Agent 2 when Agent 2 is under an attack.

Figure 3.27 shows that the position (x, y) from the sensor and relative state for Agent 2 are not equivalent since the system is under a BIA. The residual signals r_{21} and r_{24} that are generated from the developed detection algorithm are shown in Figures 3.28 and 3.29 which shows that Agent 2 is under a BIA.

Figure 3.28: Relative residuals for Agent 2 measured from Agent 1 when Agent 2 is under an attack.

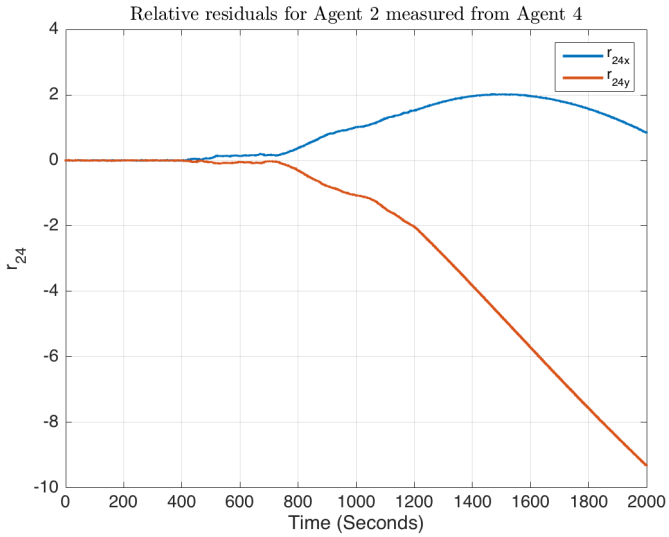
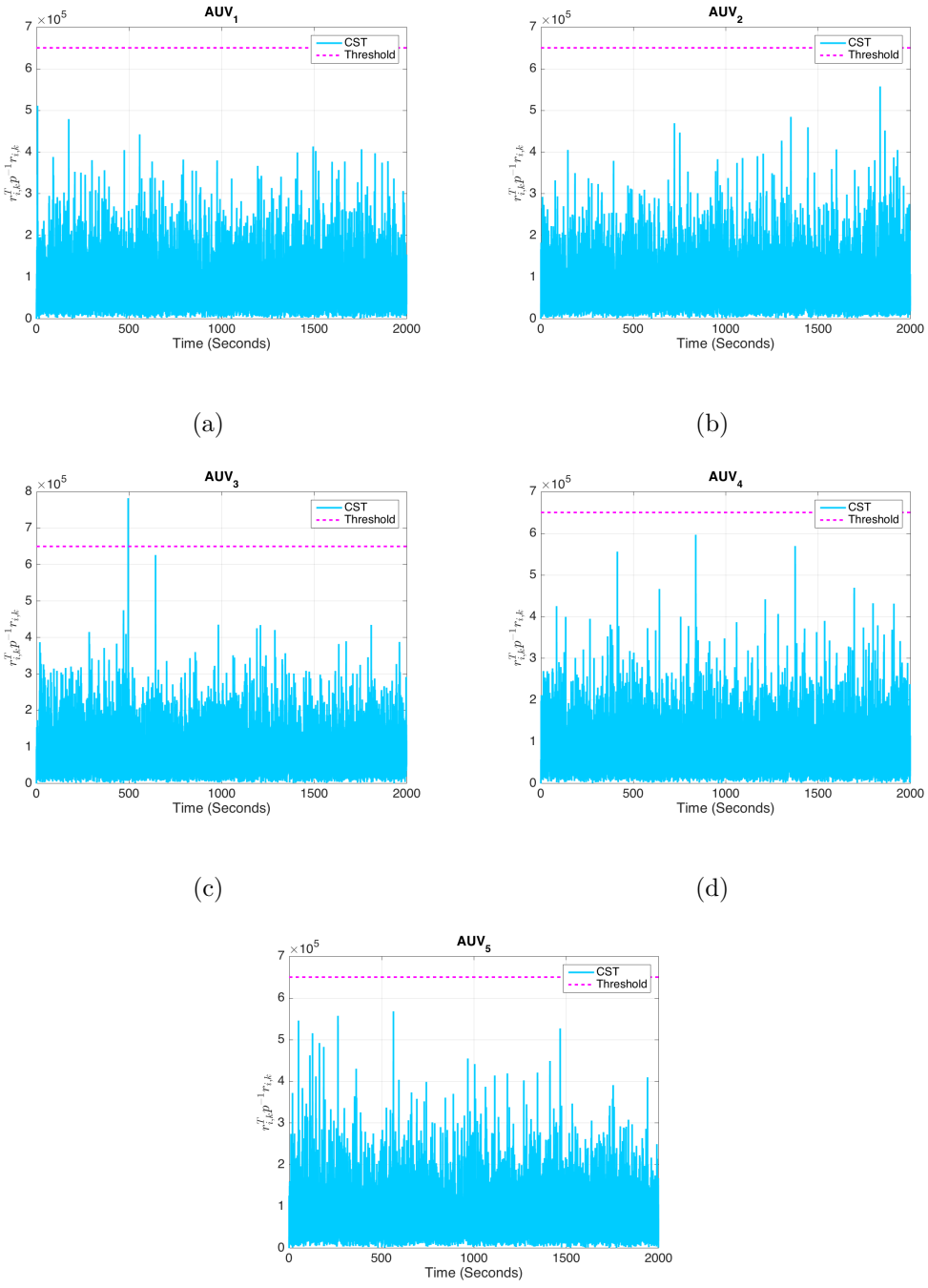


Figure 3.29: Relative residuals for Agent 2 measured from Agent 4 when Agent 2 is under an attack.

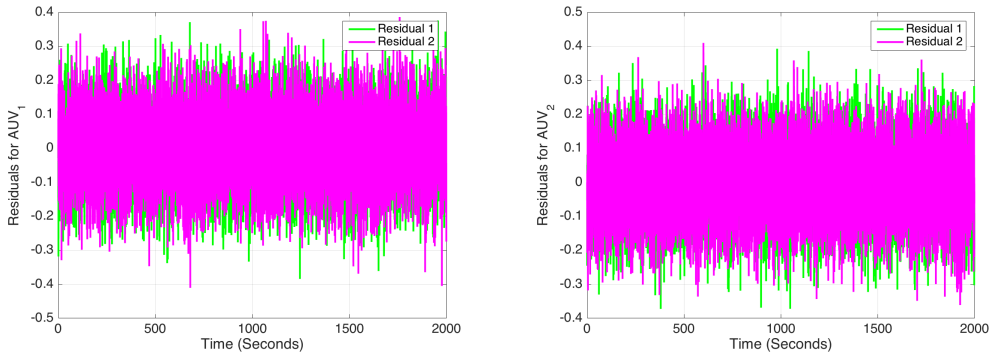
Moreover, Figure 3.31 illustrates residual signals ($r_{i,k}$) for all agents when Agent 2 is under a BIA but with no fault. As can be seen, residual signals are close to zero for all the six states of AUVs, so we conclude that the system is not under fault. Figure 3.30 shows the CST for all agents that are under the threshold when the system is under BIA.Note the

BIA is an intelligent signal that is difficult to be detected.



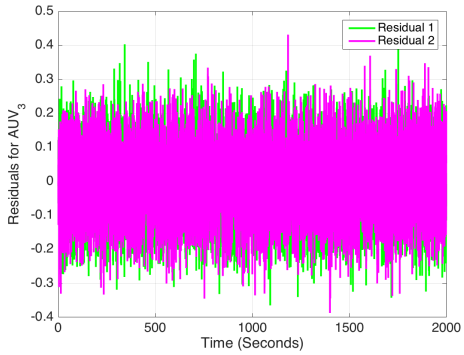
(a) (b) (c) (d) (e)

Figure 3.30: The CST for all agents with Agent 2 experiencing a BIA.

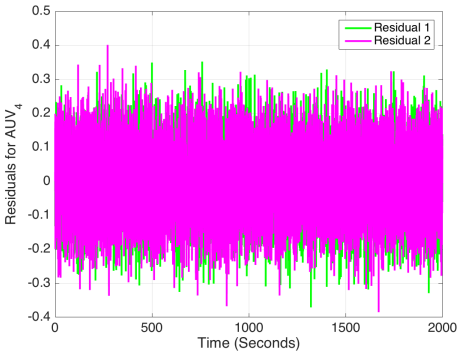


(a)

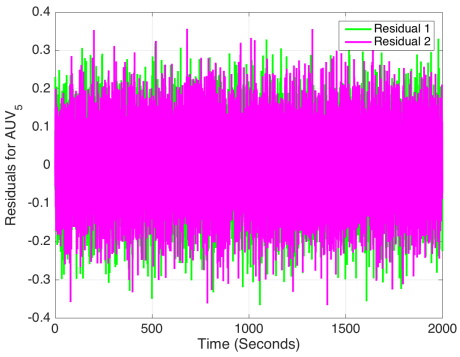
(b)



(c)



(d)



(e)

Figure 3.31: Residual signals ($r_{i,k}$) for all agents with Agent 2 experiencing a BIA

We can see in Figure 3.31 the residual signal $r_{i,k}$ equals to zero but the system is under

attack since the BIA is an intelligent signal and undetectable in the system but with the developed frame work in Figure 3.1, we can detect the BIA with residual $r_{i,j}$ as shown in Figures 3.28 and 3.29.

3.4.5 Injection of both Bias Injection Attack and Fault

In this case, Agent 2 experiences a 45 degree bias sensor fault as well as BIA among the communication channels. Agent 4 is under 80% LOE fault for the second actuator and Agent 1 is under 90% LOE fault for the first actuator as demonstrated in Figure 3.32.

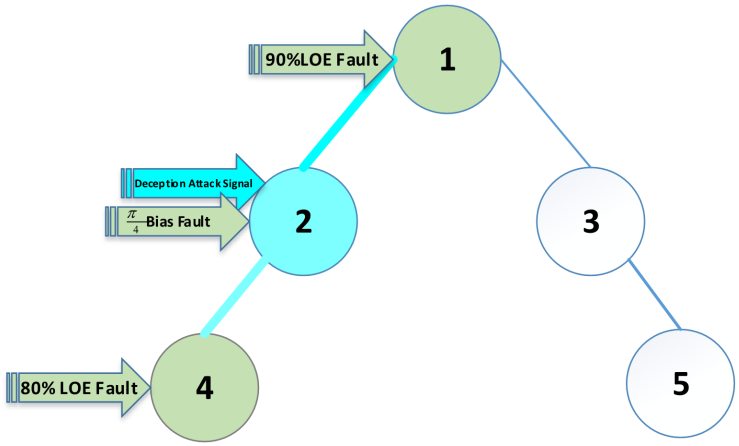


Figure 3.32: All AUVs in formation under attacks as well as faults.

It can be easily seen that all agents except Agents 3 and 5 do not have the same trajectories. Figure 3.33 illustrates that Agents 1, 2 and 4 do not follow the formation.

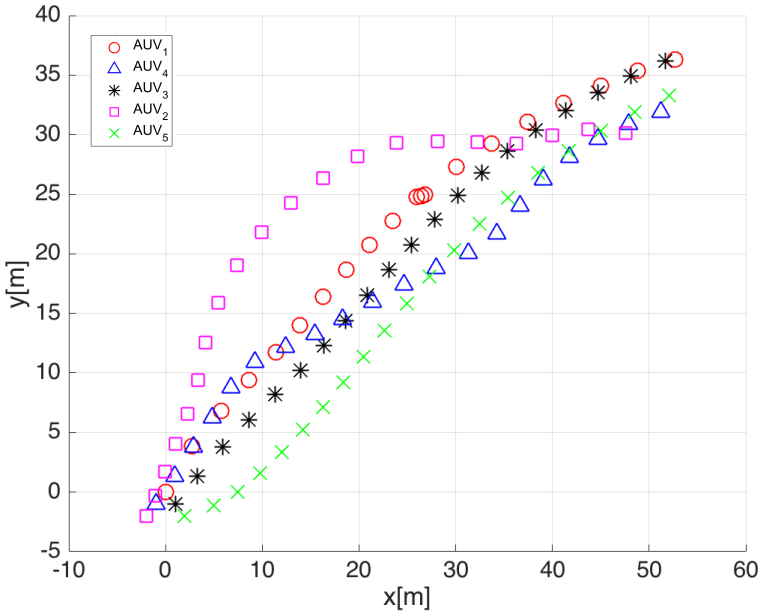


Figure 3.33: Formation trajectories in the presence of both attacks and faults.

Figure 3.34 illustrates that the position (x, y) from the sensor and relative states for Agent 2 are not equivalent because the system is under a communication attack or fault. Hence, when both fault and attack are injected to the system, position signals (x, y) and the estimated position signals $(\hat{x}, \hat{y})$ have more oscillations as compared to the system that is under attack as illustrated in Figure 3.25. Moreover, with the FD algorithm, we can detect whether agents are under BIA or fault.

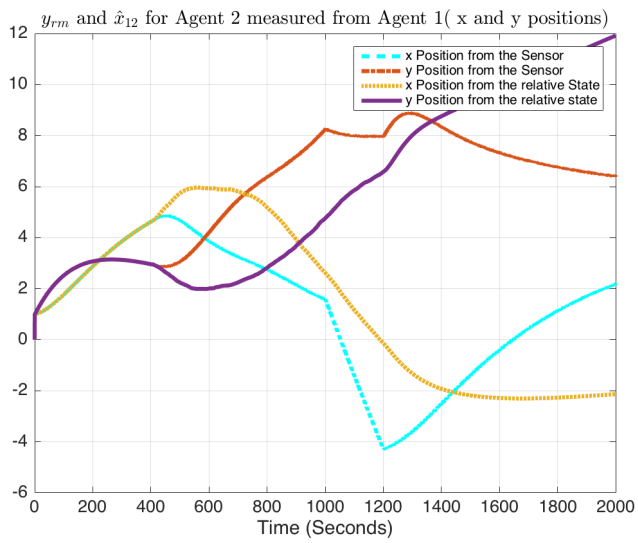


Figure 3.34: The position (x, y) from the sensor and relative state for Agent 2 when Agent 2 is under communication attack and fault.

Figure 3.35 illustrates the relative residual signals for Agent 2 measured from Agent 1 when Agent 2 is under attack.

Figure 3.35: Relative residuals for Agent 2 measured from Agent 1 when the team under attack and fault.

Figure 3.36 illustrates when Agent 2 is under attack and the relative residual signals

for Agent 2 measured from Agent 4 when Agent 2 is under attack and residual signals r_{21} and r_{24} deviate from zero starting at $t = 400\,\text{sec}$.

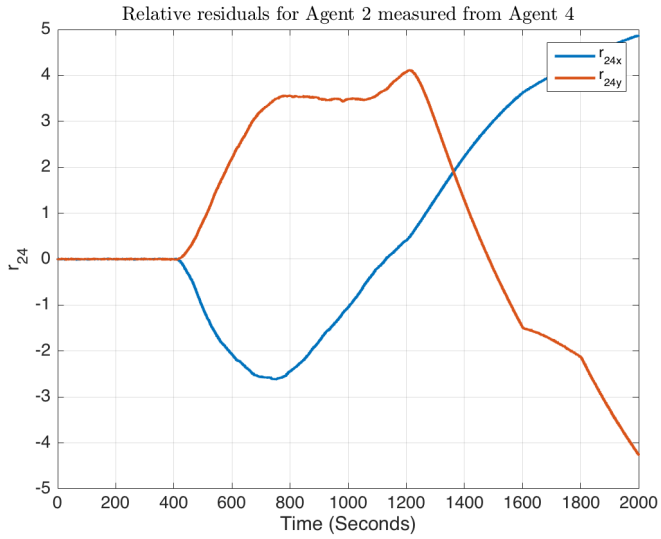


Figure 3.36: Relative residual signals for Agent 2 measured from Agent 4 when the team is under attacks and faults.

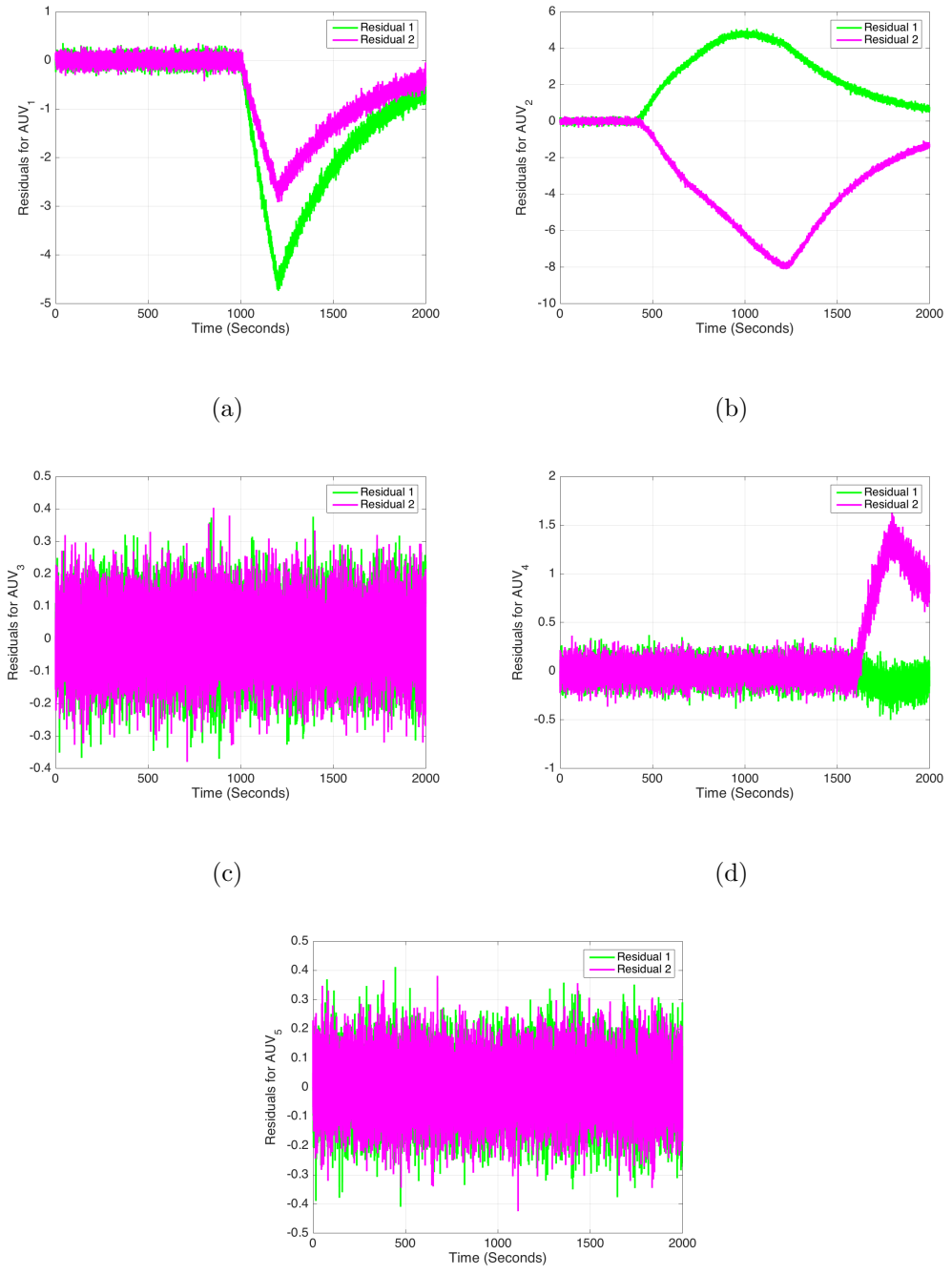
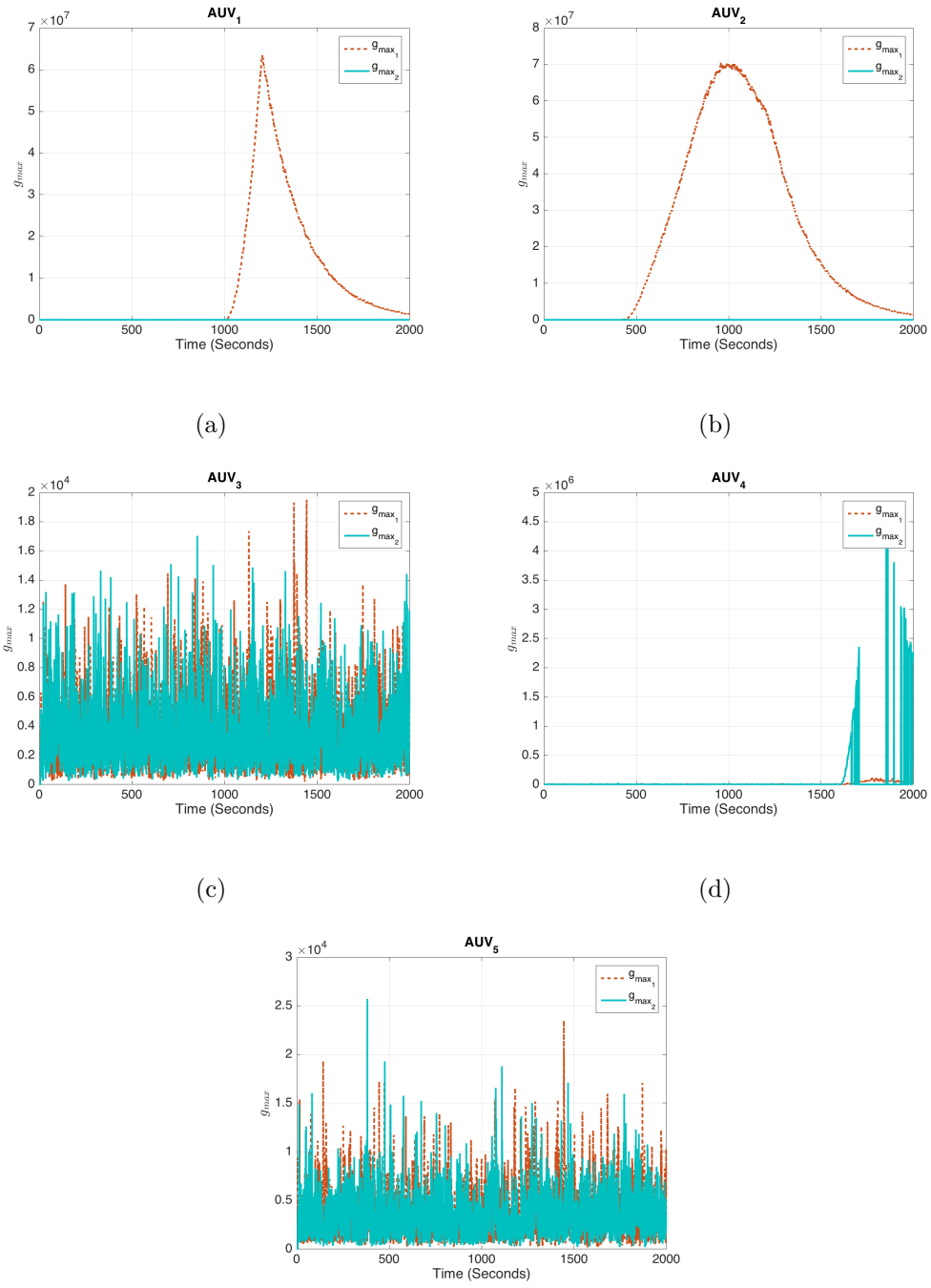


Figure 3.37: (a) Residual signal $r_{i,k}$ for Agent 1 (first actuator 90% LOE), (b) Residual signal $r_{i,k}$ for Agent 2 (45 degree bias sensor fault), (c) no fault, (d) Residual signal $r_{i,k}$ for Agent 4 (second actuator 80% LOE), (e) no fault.

Figure 3.37 depicts residuals signals $r_{i,k}$ in equation (3.25) corresponding to all agents in case of Agent 2 experiencing 45 degree bias sensor fault, a BIA among communication for Agent 2, Agent 4 experiencing 80% LOE fault in the second actuator, and Agent 1 experiencing 90% LOE fault in its first actuator and when Agents 1, 2, 4 are under fault.



(a) (b) (c) (d) (e)

Figure 3.38: The GLR algorithm for all agents under the BIA and faults.

Figure 3.38 depicts the g_k value equation (3.29) that is not equivalent when the system

is under BIA, sensor fault and actuator fault. The CSTs are above threshold for the faulty Agent 1,2 and 4 as shown in Figure 3.39.

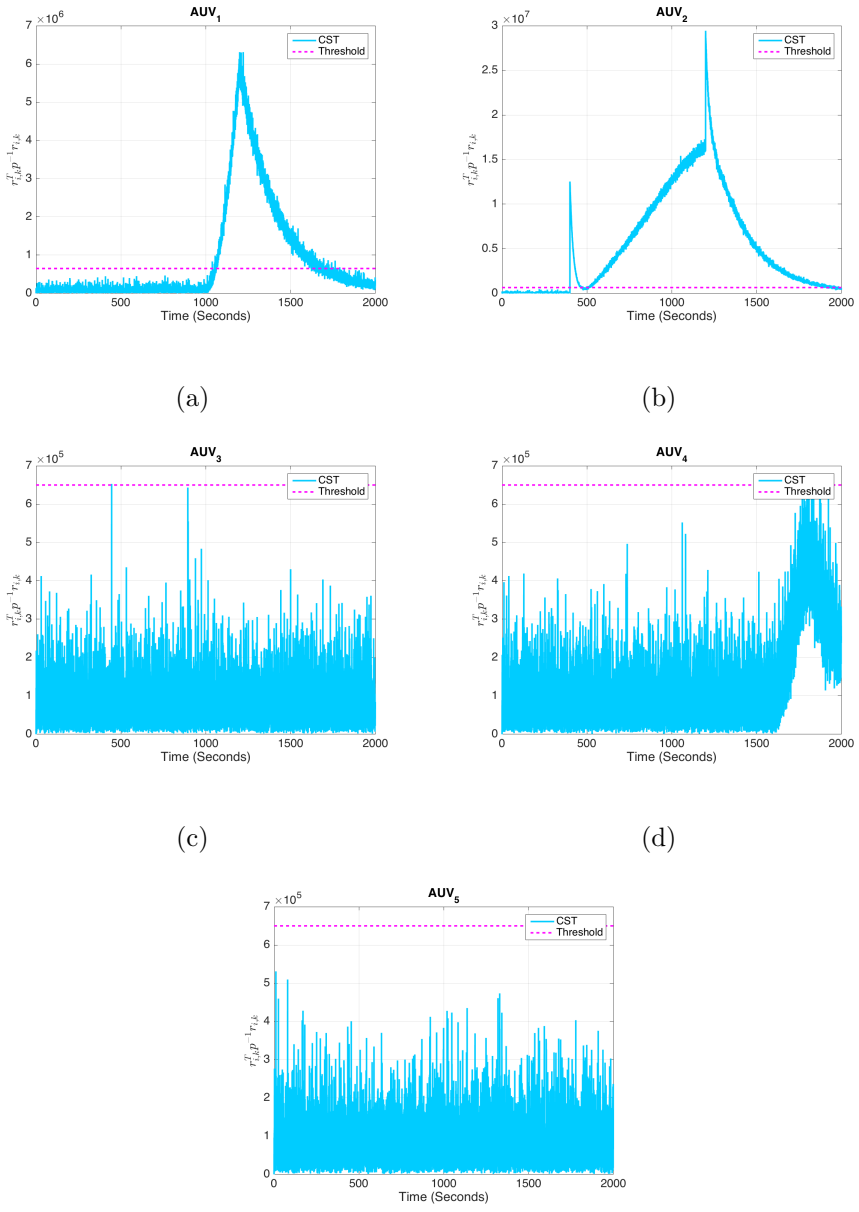
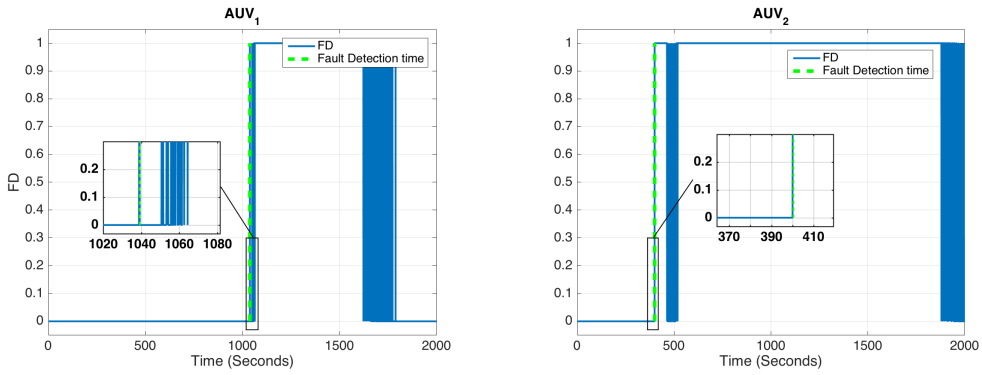
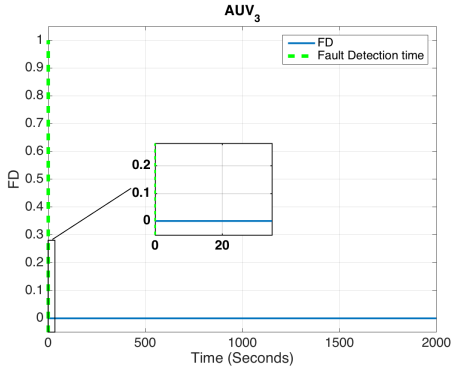


Figure 3.39: The CST for all agents under a BIA and fault.

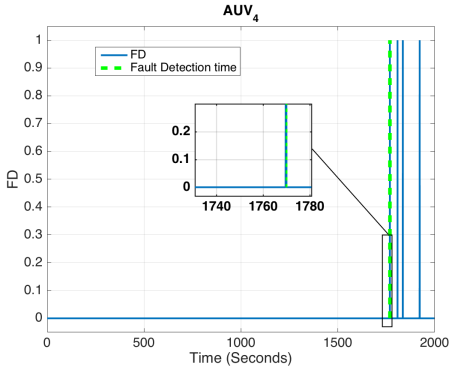


(a)

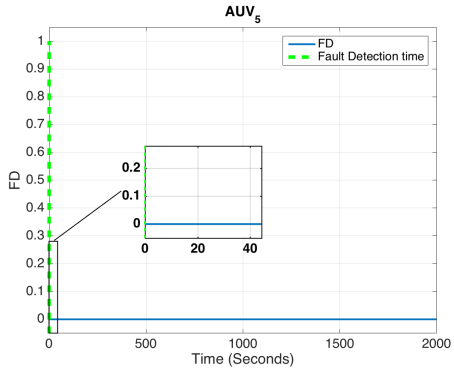
(b)



(c)



(d)



(e)

Figure 3.40: FD signals for all agents under a BIA and fault.

We define a binary AD signal for the residual signal $r_{i,j}$ as follows:

$$AD = \begin{cases} 1 & r_{i,j} > h_a \\ \\ 0 & r_{i,j} < h_a \end{cases} \tag{3.37}$$

When Agent 2 is under attack, the UKF that is developed for the neighbouring agent will yield the residual $r_{i,j}$ equation (3.34) where thereshold $(h_a) = 0.05$, $AD = 1$ when there is an attack and $AD = 0$, otherwise. As noted in equation (3.37), AD is obtained by applying a threshold mechanism. The threshold is set to 0.05 as illustrated in Figures 3.22 and 3.23.

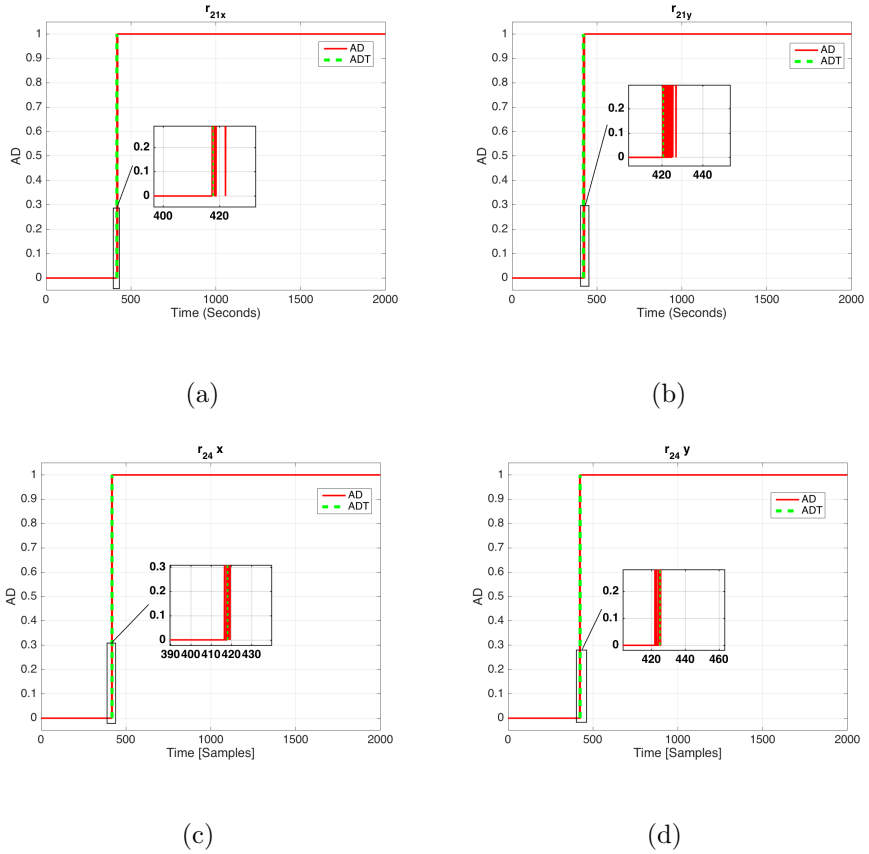


Figure 3.41: The AD when the Agent 2 is under attack.

Table 3.4: Fault detection time when the system is under BIA and fault.

Number of Agent	Type of Abnormal Event	When a fault is started.	When a fault is terminated	When a fault is detected	Detection Time [Samples]
Agent 1	First Actuator 90% LOE Fault	5000	6000	5001	1
Agent 2	45 degree Bias Sensor Fault	2000	6000	2002	2
Agent 4	Second Actuator 80% LOE Fault	8000	9000	8376	376

Figure 3.38 depicts the FD value equation (3.36) that is binary signal when the system is under BIA, sensor fault and actuator fault. The FDI times for Agent 1, Agent 2 and Agent 4 are t_d (fault detection time) = 140 (samples) $\times 0.2$ (step time) = 28 sec, $t_d = 0.4$ sec and $t_d = 64$ sec, respectively.

3.5 Performance Analysis

The performance of the developed FD and AD scheme can be analyzed by a confusion matrix (also known as the binary contingency table) approach as applied as following:

Given a classifier and a data instance, there are four possible outcomes that are:

- True Positive (TP): if the instance has been correctly identified,
- True Negative (TN): if the instance has been correctly rejected,
- False Positive (FP): if the instance has been incorrectly identified,
- False Negative (FN): if the instance has been incorrectly rejected,

In order to measure the performance measures different results are used and the Detection Rate (DR) is computed. This metric is given by:

$$\text{Detection Rate (DR)} = \frac{TP}{TP + FN} \tag{3.38}$$

Fault	TP	FN	FP	TN	DR
90% LOE Fault (Agent 1)	671	281	2210	6791	77%
45 degree Bias Sensor Fault (Agent 2)	3780	219	3803	2198	95%
80% LOE Fault (Agent 4)	543	456	1001	8000	56%

Table 3.5: The confusion matrix when the system is under both fault and BIA.

The parameters for confusion matrix when the system is under fault is illustrated in Tables 3.5, 3.6 and 3.7 which show parameters for the confusion matrix when only Agent 2 is under BIA.

BIA	ADT	TP	TN	FP	FN	DR
Agent 2(r_{21_x})	2076	3913	2000	4001	86	98%
Agent 2(r_{21_y})	2116	3877	2000	4001	122	97%

Table 3.6: The confusion matrix for relative residuals r_{21} for both x and y positions.

BIA	ADT	TP	TN	FP	FN	DR
Agent 2(r_{24_x})	2169	3908	2000	4001	91	98%
Agent 2(r_{24_y})	2224	3877	2000	4001	122	97%

Table 3.7: The confusion matrix for relative residuals r_{24} for both x and y positions.

The BIA signal is injected at $t = 400\,\text{sec}$ (2000 samples$\times$ 0.2 step time= 400 sec). The BIA is between the communication of the following agents: (Agent 2, 1) and (Agent 2, 4) as illustrated in the Figures 3.35 and 3.36. Two residual signals r_{21} and r_{24} in both x and y positions as illustrated in Tables 3.6 and 3.7, where the Attack Detection Time (ADT) for r_{21_x} and r_{21_y} are $t = 415\,\text{sec}$ (2076 samples) and $t = 433\,\text{sec}$ (2169 samples) and $t = 444\,\text{sec}$ (2224 samples), respectively.

3.6 Conclusion

In this chapter, a fault detection as well as BIA in AUVs while maintaining their formation is investigated. A distributed fault detection scheme is proposed that is based on the Generalized Likelihood Ratio (GLR) and Unscented Kalman Filter (UKF) methods. Moreover, each agent has a sensor that measures the actual relative position of neighbouring agents. Every agent has multiple UKF that measures estimated relative positions. The main observation is that each agent can detect faults as well as BIAs. All of the simulation results were provided in Section 3.4. The following suggestion can be considered for future work. 1) considering various stochastic switching network typologies, 2) considering disturbance and uncertainties for homogeneous agents, 3) studying various types of faults such as lock-in-place, float, hard over, and 4) extending the work to a team of nonlinear agents.

Chapter 4

Attack Impacts on Linear-Time Invariant Agents

4.1 Introduction

Cyber-Physical Systems (CPSs) have been extensively used in various fields, such as healthcare, environmental monitoring, intelligent transportation and aerospace information security [113]. An important type of cyber-attack is studied in the control community that is known as the false data injection. In such attacks, the attacker interrupts and changes some of the measurements and/or control signals in a coordinated way. These attacks can have a malicious goal such as destabilizing the system or considerably increasing the estimation error while staying undetected by detection algorithms. Different approaches for detecting false data injection attacks and mitigating their impacts have been studied in [114].

The problem of security allocation in stochastic linear dynamical systems have been

studied [52]. A state estimation problem where a KF is used as an estimator, and a chi-squared test is used to detect anomalies have been considered [115]. Furthermore, analyzing on protection against Bias Injection Attack (BIA) has been investigated in [116]. In BIAs, the attacker goal is to increase the mean square estimation error by adding a constant bias to some of the sensor measurements while remaining undetected. The finding of a criterion for selecting sensors for security is the main goal in order to mitigate the attack impacts [117]. In this chapter, an estimation problem in the presence of bias injection attack is considered. The Kalman Filter (KF) is used to estimate the system state, and a chi-squared test is used for detecting anomalies such as BIA and our contribution is to provide the attack impact and derive a lower bound of the impact attack in terms of the combination number of sensors and actuators.

4.2 The System Under Study

The plant is considered as a linear time-invariant system given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + B_a a_u(k) + w(k), \\ y_a(k) &= Cx(k) + Da_y(k) + v(k), \end{aligned} \tag{4.1}$$

where $A \in \mathbb{R}^{n\times n}$, $x(k) \in \mathbb{R}^n$ is the system state vector, $B \in \mathbb{R}^{n\times m}$ and $u(k) \in \mathbb{R}^m$ is the control matrix and input, $y_a(k) \in \mathbb{R}^p$ is the measurement output under only attack, and $w(k) \in \mathbb{R}^n$ and $v(k) \in \mathbb{R}^p$ are the process and measurement noises, respectively. The vector $a_y(k) \in \mathbb{R}^{n_a}$ indicates the signal that the attacker injects in the output. The matrix

$D \in \mathbb{R}^{p \times n_a}$ is the matrix that maps $a_y(k)$ to the measurement output $y_a(k)$. The elements $(i_1, 1), (i_2, 2), \ldots, (i_{n_a}, n_a)$ of the matrix D are ones, and the rest is zero. The indices $i_1 < i_2 < \cdots < i_{n_a}$ denote the measurements that are under the attacker's control.

The secured sensors will not be affected by the attacker. The secured sensors as explained secure some of the existing or placing additional sensors, such as to make undetectable attacks introduced harder to achieve [52]. Assume that the attacker controls the specific number of sensors. The secured sensors have rows of zeros in D. The vector $a_u(k) \in \mathbb{R}^{m_a}$ denotes the signal that the attacker injects in the impact, and m_a is the number of the attacker inputs. The matrix $B_a \in \mathbb{R}^{n \times m_a}$ denotes the attack matrix. The elements $(j_1, 1), (j_2, 2), \ldots, (j_{m_a}, m_a)$ of the matrix B_a are one, and the rest are zero. The indices $j_1 < j_2 < \cdots < j_{m_a}$ in B_a denote the measurement that are under the attackers control, in the vector $u(k)$. The secured actuators cannot be affected by the attacker. The rows of B_a matrix are equal to zero for the secured actuators. The added random Gaussian noise signals are independent and identically distributed vectors with $(w(k) \sim \mathcal{N}(0, \Sigma_w), v(k) \sim \mathcal{N}(0, \Sigma_v))$. The state-space realization (A, B, C) has no invariant zeros. In particular, this assumption implies that the pair (A, C) is observable and the pair $(\Sigma_r^{-\frac{1}{2}}, A)$ is stabilizable. The attack matrices B_a and D are dependent on the physical structure of actuators and sensors. Moreover, the attack matrices should satisfy the following criteria:

$$\text{span } D \subset \text{span } C \quad \textit{and} \quad \text{span } B_a \subset \text{span } B.$$

4.2.1 The Monitoring System

The state vector $x(k)$ is estimated by using the KF to reach a steady state. Let $\hat{x}(k)$ be the estimate of the system state vector under attack. Then, the estimator dynamics can be represented as follows:

$$\hat{x}(k+1 \mid k) = (A - KC)\hat{x}(k \mid k-1) + Ky_a(k) + Bu(k), \tag{4.2}$$

where the Kalman gain K and error covariance matrix Σ_e are provided below:

$$\Sigma_e = \mathbb{E}((\hat{x}(k+1) - x(k+1))(\hat{x}(k+1) - x(k+1)))^T, \tag{4.3}$$

$$K = A\Sigma_e C^T (C\Sigma_e C^T + \Sigma_w)^{-1},$$

and the steady state Riccati equation of the error covariance matrix Σ_e is given by

$$\Sigma_e = A\Sigma_e A^T - A\Sigma_e C^T (C\Sigma_e C^T + \Sigma_w)^{-1} C\Sigma_e A^T + \Sigma_w,$$

since (C, A) is observable. Note that the matrix $A-KC$ is asymptotically stable [118]. While a CPS is under attack, the monitoring system can compare a sequence of the compromised data to the expected output of the healthy system in order to detect any cyber attacks. Since a cyber attack can be regarded as a new attack capabilities in the CPS, existing fault diagnosis algorithms can be used to detect such attacks.

4.2.2 Detection Algorithm

In this chapter, we consider a popular attack detection algorithm that constantly checks the statistical properties of the residual signal generated by the steady-state KF. Such a setup has wide engineering applications and can be extended to nonlinear and non-Gaussian processes [119]. The residual signal that is considered in the detection algorithm is defined as follows:

$$r(k) = y_a(k) - C\hat{x}(k \mid k-1), \tag{4.4}$$

where $C\hat{x}(k \mid k-1)$ denotes an estimate of $Cx(k)$, and consequently $r(k)$ shows the difference between $y_a(k)$ and its modeled behavior. In case of no anomaly, $r(k)$ is a white Gaussian process with a zero mean and a covariance matrix given in equation (4.5) [117].

$$\Sigma_r = C\Sigma_e C^T + \Sigma_w. \tag{4.5}$$

Therefore, cyber attacks can be diagnosed by testing the hypotheses in equation (4.6). In this chapter, we only consider BIA

$$\begin{cases} \mathcal{H}_0 : & r(k) \sim \mathcal{N}(0, \Sigma_r), \\ \mathcal{H}_1 : & r(k) \nsim \mathcal{N}(0, \Sigma_r), \end{cases} \tag{4.6}$$

where $\mathcal{N}(0, \Sigma)$ illustrates the Gaussian distribution with zero mean and the covariance Σ. There are various statistical hypothesis testing algorithms such as the Sequential Probability Ratio Test (SPRT) [120], the Cumulative Sum (CUMSUM) [121], Generalized Likelihood Ratio (GLR) test [122]. An easy way to realize the residual signal is through Compound

Scalar Testing (CST) that checks $r(k)^T\Sigma_r^{-1}r(k)$ [92], and the corresponding hypothesis test is as follows:

$$\begin{cases} \text{Accept} \quad \mathcal{H}_0 \quad \text{if} \quad r(k)^T\Sigma_r^{-1}r(k) \leq \tau, \\ \text{Accept} \quad \mathcal{H}_1 \quad \text{if} \quad r(k)^T(k)\Sigma_r^{-1}r(k) > \tau, \end{cases} \tag{4.7}$$

where $y_a(k) \in \mathbb{R}^p$ is the measurement output under only attack, $\tau > p^1$ is a threshold value. If there are attacks, $r^T(k)\Sigma_r^{-1}r(k) > \tau$, and hence follows the χ_k^2 distribution with p degrees of freedom. Then, $\mathbb{E}[r^T(k)\Sigma_r^{-1}r(k)] = I_p$ and it is highly likely that the null hypothesis $\mathcal{H}_0$ will be accepted due to the statistical characteristics of the χ_k^2 distribution and CST declares $r(k)^T\Sigma_r^{-1}r(k) > \tau$, implying that there is an anomaly in the system that may be induced by cyber-attacks. Since the entire testing process is stochastic, there is a chance of getting false alarms according to the threshold value τ as illustrated in Figure 4.1.

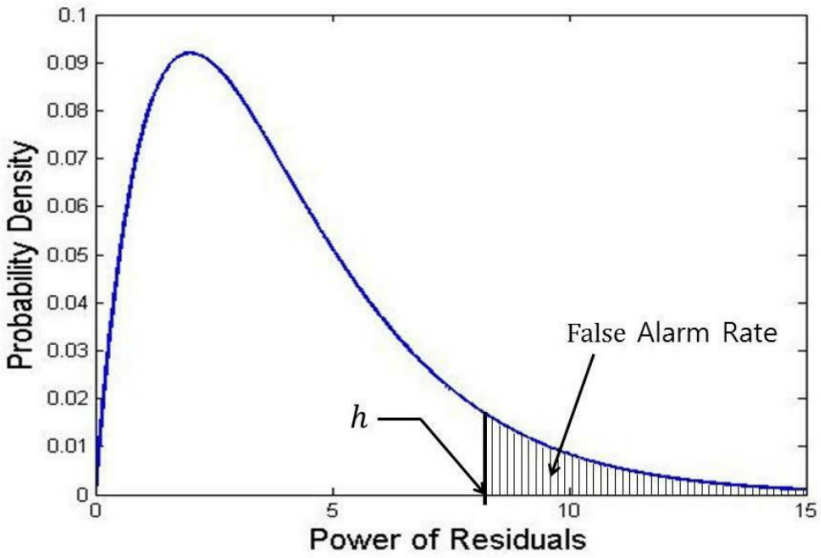


Figure 4.1: Probability distribution of residual power [6].

The presence of anomalies change the distribution of $r(k)$, and a simple method is used to test whether the Euclidean distance is more than a large threshold ($\tau > 0$). The random

variable χ_k^2 is distributed according to a chi-squared test as follows:

$$\chi_k^2 = r^T(k)\Sigma_r^{-1}r(k) = \left\|\Sigma_r^{-\frac{1}{2}}r(k)\right\|_2^2. \tag{4.8}$$

In absence of anomalies, the random variable χ_k^2 in equation (4.8) is usually relatively small. When χ_k^2 exceeds the threshold $\chi_k^2 > \tau$, it might be an indication that an anomaly has occurred. However, it is important to realize that any threshold τ in equation (4.9) may be crossed even when anomalies are not present and hence generating false alarms. These false alarms may occur due to random noise with the probability distribution function as follows:

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}r(k)\right\|_2^2 > \tau) := \alpha. \tag{4.9}$$

Large values of τ would decrease the false alarm probability α along with the sensitivity to detection of anomalies.

4.3 Non-Central Chi-Squared Distribution

Let X be an p-dimensional Gaussian random vector with a mean value μ_X and the covariance matrix of $\Sigma_X = I_p$. This distribution is illustrated by two parameters, where the first parameter is the number of degrees of freedom (n), that is the dimension of the random vector X. The other parameter is the non-centrality parameters that is explained by λ. This parameter introduces the squared Euclidean norm of the mean value of X that is $\lambda = \|\mu_x\|_2^2$.

Complementary cumulative distribution function of the random variable χ^2_μ is given below

$$\mathbb{P}(\chi^2_\mu > \tau) = Q_{\frac{n}{2}}(\sqrt{\lambda}, \sqrt{\tau}), \tag{4.10}$$

where $\tau > 0$ and $Q_{\frac{n}{2}}(\sqrt{\lambda}, \sqrt{\tau})$ denotes the generalized Marcum Q-function, which is defined by $Q_v(a, b)$ and is continuous in a and b , $v > 0$ [123].

Lemma 4.3.1 *The generalized Marcum Q-function $Q_v(a, b)$ is strictly increasing in a for all $a \geq$ and $b, v > 0$.*

Proof. *We refer the interested reader to [124].* □

From Lemma 4.3.1, it follows directly that for fixed n, and τ, $Q_{\frac{n}{2}}(\sqrt{\lambda}, \sqrt{\tau})$ is strictly increasing in λ.

4.4 Generalized Eigenvalues and Eigenvectors

Definition 4.4.1 *[125] Let M, N be matrices $\mathbb{C}^{n\times n}$. The set of generalized eigenvalues of the matrix pencil (pair) (M, N) is given by $\lambda(M, N) = \{\lambda \in \mathbb{C} : det(M - \lambda N) = 0\}$.*

The *generalized eigenvector* x of (M, N) is a nontrivial solution to the equation $Mx = \lambda Nx$ $N \succ 0$ and $M \succeq 0$. The pencil (M, N) has exactly n real non-negative generalized eigenvalues.

Lemma 4.4.1 *Let $M \succeq 0$ and $N \succ 0$ with $M, N \in \mathbb{R}^{N\times N}$, and let $0 \leq \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the generalized eigenvalues of the pencil (M, N). Then, for any $\{j \in 1, \ldots, n\}$ we have*

$$\lambda_j = \min_{\substack{U \subseteq \mathbb{R}^n \\ \dim(U)=j}} \max_{\substack{x \in U \\ x \neq 0}} \frac{x^T M x}{x^T N x}, \tag{4.11}$$

where U denotes a subspace of the vector space $\mathbb{R}^n$.

Proof. We refer the interested reader to [126]. □

4.5 Bias Injection Attacks (BIAs)

In the BIA, a false data injection is considered where the adversary's goal is to inject a constant bias in the system without being detected. Furthermore, the bias is calculated so that the impact at the steady-state is maximized and the attack a_{k+1} is described by [117]

$$a_{k+1} = \beta a_k + (1 - \beta) a^*_\infty, \tag{4.12}$$

where $a_0 = 0$, $0 < \beta < 1$ and a^*_∞ will be provided in Theorem 4.5.5.

The BIAs are classified into the following three cases:

- **Case 1:** The attacker is able to dispute the data from sensors [117].
- **Case 2:** The attacker is able to dispute the data from actuators.
- **Case 3:** The attacker is able to dispute the data from sensors and controllers.

The contribution of this chapter is to extend the analysis and mitigation results of the of bias injection attack for actuators (Case 2) in Section 4.5.2 and also in the simultaneous attack on both sensors and actuators (Case 3) in Section 4.5.3. The end result would be a

condition on the minimum number of sensors/actuators that should be secured in a system to maintain the estimation error less than a pre-specified upper bound.

The contributions are two fold for Case 2 in Section 4.5.2 and Case 3 in Section 4.5.3. First, the estimation problem in presence of cyber-attacks will be considered. Second, we formulate the problem of finding the worst case bias injection attacks. Finally, we provide an analysis of the attack impact and derive a lower bound of the attack impact in terms of the number of actuators for Case 2 and combination of sensors/actuators for Case 3. A numerical example is presented for Case 2 in Section 4.6.2. Numerical examples for the combination of sensors and actuators (Case 3) are presented in Section 4.6.3.

4.5.1 Case 1: Secured Sensors

In this scenario, the attacker designs a BIA to corrupt only sensors of the system. A linear time-invariant system is given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w(k), \\ y_a(k) &= Cx(k) + Da_y(k) + v(k). \end{aligned} \tag{4.13}$$

The attack measurement $y_a(k)$ is used to obtain the state estimates. The KF is used to estimate $\hat{x}_{a_y}(k \mid k-1)$ in equation (4.14). It is the sum of the response to $y(k)$ and $a(k)$ which is given by

$$\hat{x}_{a_y}(k \mid k-1) = \hat{x}(k \mid k-1) + \Delta\hat{x}_y(k \mid k-1), \tag{4.14}$$

where $\Delta\hat{x}_y(k \mid k-1)$ is only dependent on the attack signal that is propagated by

$$\hat{x}_{a_y}(k+1 \mid k) = (A - KC)\hat{x}_{a_y}(k \mid k-1) + Ky_a(k) + Bu(k). \tag{4.15}$$

Moreover, in the event of a BIA, the attack signal $a_y(k)$ converges slowly to some constant vector [116]. We formulate the problem of finding the vector a_y that maximizes the mean square estimation error in the steady state and does not increase the alarm probability more than a certain threshold by substituting the equation (4.14) into equation (4.15) as

$$\hat{x}(k+1 \mid k)+\Delta\hat{x}_y(k+1 \mid k) = (A-KC)(\hat{x}(k \mid k-1)+\Delta\hat{x}_y(k \mid k-1))+Ky(k)+KDa_y(k)+Bu(k),$$

$$\Delta\hat{x}_y(k+1 \mid k) = (A - KC)\Delta\hat{x}_y(k \mid k-1) + KDa_y(k). \tag{4.16}$$

The matrix $A - KC$ of the Kalman filter is asymptotically stable, hence, both $\Delta\hat{x}_y(k)$ and $\Delta r(k)$ reach steady states. The steady state equation for $\Delta\hat{x}_y(k)$ is given as follows:

$$\Delta\hat{x}_y = (A - KC)\Delta\hat{x}_y + KDa_y, \tag{4.17}$$

The steady state equation for $\Delta r(k)$ is given by

$$\Delta r = Da_y - C\Delta\hat{x}_y. \tag{4.18}$$

The solution to equation (4.16) is calculated by

$$\Delta\hat{x}_y = (I_n - A + KC)^{-1}KDa_y := G_{\hat{x}}KDa_y, \tag{4.19}$$

where the matrix $I_n - A + KC$ is invertible and $G_{\hat{x}} = (I_n - A + KC)^{-1}$. Substituting equation (4.19) into equation (4.18) yields

$$\Delta r = Da_y - CG_{\hat{x}}Da_y = (I_p - CG_{\hat{x}})Da_y := G_r Da_y, \tag{4.20}$$

where $G_r = (I_p - CG_{\hat{x}})$.

Remark 4.5.1 *Note that the bias injection attack is not detected during the transient phase. Moreover, the attacker can smoother the transient behavior by adding the attack slowly [116].*

There is an error between the state vector $x(k)$ and the corrupted estimate $\hat{x}_{a_y}(k)$. We specify the objective of the attacker as to maximize the expected state estimation error as shown below,

$$x(k) - \hat{x}_{a_y}(k) = e(k) - \Delta\hat{x}_y(k),$$

$$\max_{a_y \in \mathbb{R}^{n_a}} \mathbb{E}\|e(k) - \Delta\hat{x}_y\|_2^2. \tag{4.21}$$

The objective is that $\mathbb{E}\,\|e(k) - \Delta\hat{x}_y\|_2^2 = \mathbb{E}\{\|e(k)\|_2^2 - 2\Delta\hat{x}_y^T e(k) + \|\Delta\hat{x}_y\|_2^2\}$ and the term $\mathbb{E}\{2\Delta\hat{x}_y^T e(k)\}$ is equal to zero because $\Delta\hat{x}_y$ is constant that is defined in equation (4.21), and $e(k)$ has a zero mean. The term $\mathbb{E}\,\|e(k)\|_2^2 = Tr(\Sigma_e)$ is constant, and is part of the error that comes from noise. In order to solve equation (4.21), the attacker should find the attack signal such that the following expected value is maximized,

$$\mathbb{E}\|\Delta\hat{x}_y\|_2^2 = \|G_{\hat{x}}KDa_y\|_2^2.$$

The BIAs preserve the nature of the distribution of the residual signal since it remains

Gaussian. However, the attack changes the mean value of the distribution and the probability of alarms can increase. As a result, the constraint for the attacker is to not considerably increase the alarm probability. This constraint can be modeled as follows:

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2 > \tau) \leq \alpha + \Delta\alpha \leq 1. \tag{4.22}$$

In order to investigate the worst cyber attack the following optimization problem is formalized that represents the intent of the cyber attacks.

Problem 1:

$$\begin{aligned} &\max_{a\in\mathbb{R}^{n_a}} \|G_{\hat{x}} K D a_y\|_2^2, \\ s.t. \quad &\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2 > \tau) \leq \alpha + \Delta\alpha. \end{aligned} \tag{4.23}$$

where $\Delta\alpha > 0$, $\alpha + \Delta\alpha \leq 1$ is a threshold and the alarm probability in the presence of attacks raises to $\alpha + \Delta\alpha$ and the attack will remain undetected.. Few assumptions will be considered for the worst case scenario for the attacker.

Assumption 4.5.1 *The attacker is assumed to have the full knowledge of the CPS and the detection algorithm. Moreover, the attacker is able to control all unsecured sensors.*

Under the above assumption, the attacker can use the full system model to design the attack strategy. The worst attackers with limited information about the system could only cause less trouble to the system than those with perfect knowledge. For a particular value of δ in equation (4.24), Problem 1 can be modeled as Problem 2. It has an analytical solution and we will prove that for the particular choice of δ Problem 1 is equivalent to Problem 2.

Problem 2:

$$\begin{aligned} &\max_{a\in\mathbb{R}^{n_a}} \|G_{\hat{x}} K D a_y\|_2^2, \\ s.t. \quad &\left\|\Sigma_r^{-\frac{1}{2}}(G_r D a_y)\right\|_2^2 \leq \delta^2. \end{aligned} \tag{4.24}$$

The random variable $\Sigma_r^{-\frac{1}{2}} r(k)$ in equation (4.24) has a Gaussian distribution with a unit covariance and a zero mean, while the goal of the bias injection attack is only to change the mean value, the attacker can change the distribution form zero to $\Sigma_r^{-\frac{1}{2}}\Delta r$. This is due to the fact that the spherical symmetry of the distribution, the spherical symmetry of the integrated area, and the direction of change are not effected. It should be noted that $\left\|\Sigma_r^{-\frac{1}{2}}\Delta r\right\|_2^2$ has an impact on the alarm rate.

Lemma 4.5.1 *The following alarm probability*

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2 > \tau)$$

is approximately increasing in $\left\|\Sigma_r^{-\frac{1}{2}}(\Delta r)\right\|_2^2 = \left\|\Sigma_r^{-\frac{1}{2}}(G_r D a_y)\right\|_2^2$ *[117].*

Theorem 4.5.1 *If Assumption 4.5.1 holds, then there exists* $\delta \in \mathbb{R}$ *such that Problem 1 and Problem 2 are equivalent [117].*

Let $\bar{r} \in \mathbb{R}^P$ be a unit vector and δ satisfies the following criterion

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\delta r)\right\|_2^2 > \tau) \leq \alpha + \Delta\alpha. \tag{4.25}$$

The generalized Marcum Q-function increases in the non-centrality parameter for $p > 0$. Therefore, the non-centrality parameter is equal to $\left\|\Sigma_r^{-\frac{1}{2}}(G_r D a_y)\right\|_2^2$ and such δ exists, since

the alarm probability is the Marcum Q-function, and this function is continuous and strictly increasing in δ^2 for p, $\tau > 0$.

4.5.1.1 Mitigating the Impact of the Sensor Attacks

It is always of interest to first check if the attacker is able to increase the error arbitrarily and large damage that is undetectable at the same time. A sufficient condition is the existence of $Da_y \neq 0$ such that $\Sigma_r^{-\frac{1}{2}} G_r Da_y = \Sigma_r^{-\frac{1}{2}} \Delta r = 0$. From equation (4.18), $Da_y = C\Delta\hat{x}_y$ and the solution for Problem 2 is analyzed as follows [117]:

$$\begin{aligned} \Delta\hat{x}_y &= (A - KC)\Delta\hat{x}_y + KDa_y, \\ &= A\Delta\hat{x}_y + K(Da_y - C\Delta\hat{x}_y) = A\Delta\hat{x}_y. \end{aligned} \tag{4.26}$$

Corollary 4.5.1 *[117] If the matrix A does not have an eigenvalue equal to 1, it follows that null$\{G_r\} = \{\varnothing\}$.*

The corollary (4.5.1) has nontrivial solution, only when the matrix A has $\lambda = 1$.

Assumption 4.5.2 *The matrix A does not have an eigenvalue of 1.*

Under Assumption 4.5.2, the solution of Problem 2 can be considered as follows:

Theorem 4.5.2 *Suppose that Assumption 4.5.2 holds. Then, the solution of Problem 2 is obtained by [116]*

$$a_y^* = \pm \frac{\delta}{\left\| \Sigma_r^{-\frac{1}{2}} (G_r D\nu^*) \right\|_2} \nu^*,$$

where ν^* is the unit generalized eigenvector that corresponds to the maximal generalized eigenvalue λ^* of the pencil

$$(D^T G_{\hat{x}}^T G_{\hat{x}} D, D^T G_r^T \Sigma_r^{-1} G_r D), \tag{4.27}$$

and the maximal increase of mean squared error is defined as

$$\left\| G_{\hat{x}} D a_y^* \right\|_2^2 = \lambda^* \delta^2. \tag{4.28}$$

Therefore, we can conclude that attack has an impact on two parameters λ^* and δ^2. The parameter δ^2 and λ^* depend on the properties of equation (4.27). The corresponding simulation results are provided in Section 4.6.1.

4.5.2 Case 2: Secured Actuators

Our contribution is to provide the analysis of attack impact and derive a lower bound of attack impact in terms of the number of actuators. If the attacker can attack only the actuators of the linear system, then the linear system model will be as follows:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + B_a a_u(k) + w(k), \\ y_a(k) &= Cx(k) + v(k). \end{aligned} \tag{4.29}$$

where the attacker changes the value of $a_u(k)$. Note that the secured actuator cannot be affected by the attacker. When the rows of B_a matrix for the corresponding actuators are equal to zero, the extract attack impact is used to construct the state estimate, as follows

$$\hat{x}_{a_u}(k \mid k-1) = \hat{x}(k \mid k-1) + \Delta\hat{x}_u(k \mid k-1),$$

$$\hat{x}_{a_u}(k+1 \mid k) = (A - KC)\hat{x}_{a_u}(k \mid k-1) + Ky(k) + Bu(k) + B_a a_u(k), \tag{4.30}$$

$$\Delta\hat{x}_u(k+1 \mid k) = (A - KC)\Delta\hat{x}_u(k \mid k-1) + B_a a_u(k). \tag{4.31}$$

Consider the attack signal and the residual signal is given by

$$y - C\hat{x}_{a_u}(k) = r(k) - C\Delta\hat{x}_u(k) = r(k) + \Delta r(k), \tag{4.32}$$

and the steady state equations for $\Delta\hat{x}_u$ and Δr are as follows:

$$\Delta\hat{x}_u = (A - KC)\Delta\hat{x}_u + B_a a_u, \tag{4.33}$$

$$\Delta r = B_a a_u - C\Delta\hat{x}_u, \tag{4.34}$$

where the matrix $A - KC$ is stable and the matrix $I_n - A + KC$ is invertible. Then, the solution to equation (4.31) is obtained by

$$\Delta\hat{x}_u = (I_n - A + KC)^{-1} B_a a_u := G_{\hat{x}} B_a a_u, \tag{4.35}$$

where $G_{\hat{x}} = (I_n - A + KC)^{-1}$. Substituting equation (4.35) into equation (4.36) yields

$$\Delta r = B_a a_u - CG_{\hat{x}} B_a a_u = G_r B_a a_u, \tag{4.36}$$

where $G_r = (I_p - CG_{\hat{x}})$. The error between $x(k)$ and the corrupted estimate $\hat{x}_{a_u}(k)$ is given by

$$x(k) - \hat{x}_{a_u}(k) = e(k) - \Delta\hat{x}_u(k). \tag{4.37}$$

The goal of the attacker is to increase the mean square error in equation (4.38) once $\Delta\hat{x}_u(k)$ reaches steady state $\Delta\hat{x}_u$, that is

$$\max_{a\in\mathbb{R}^{m_a}} \mathbb{E}\|e(k) - \Delta\hat{x}_u\|_2^2. \tag{4.38}$$

The attacker's desire is to find a_u that maximizes

$$\mathbb{E}\|\Delta\hat{x}_u\|_2^2 = \|G_{\hat{x}}B_a a_u\|_2^2 .$$

The attacker problem for Case 2 is now given below

Problem 3:

$$\begin{aligned} &\max_{a\in\mathbb{R}^{m_a}} \|G_{\hat{x}}B_a a_u\|_2^2 , \\ s.t. \qquad &\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k) + \Delta r)\right\|_2^2 > \tau) \leq \alpha + \Delta\alpha. \end{aligned} \tag{4.39}$$

where $\Delta\alpha > 0$, $\alpha + \Delta\alpha \leq 1$ is a threshold and the alarm probability in the presence of attacks raises to $\alpha + \Delta\alpha$ and the attack will remain undetected.. Few assumptions will be considered for the worst case scenario for the attacker.

Assumption 4.5.3 *The attacker is assumed to have full knowledge of the CPS and the detection algorithm is used in Problem 3. Moreover, the attacker is able to control all the unsecured actuators.*

Under this assumption, the attacker can use the full system model to design the attack strategy. That is, the worst attackers with limited information about the system could only cause less trouble to the system than those with perfect knowledge. Then, for a particular value of δ in equation (4.40), Problem 3 can be modeled as a quadratic problem with a constraint where Problem 4 has an analytical solution, and the problem statement as the worst case impact of BIA is considered as given below,

Problem 4:

$$\begin{aligned} &\max_{a\in\mathbb{R}^{m_a}} \|G_{\hat{x}} B_a a_u\|_2^2 \\ s.t. \quad &\left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2 \leq \delta^2. \end{aligned} \tag{4.40}$$

The random variable $\Sigma_r^{-\frac{1}{2}} r(k)$ in equation (4.40) has a Gaussian distribution with a unit covariance and zero mean. The bias injection attack is only to change the mean value, the attacker can change the distribution from zero to $\Sigma_r^{-\frac{1}{2}}\Delta r$. This is due to the fact that given the spherical symmetry of the distribution and the spherical symmetry of the integrated area the direction of change is not effected. The magnitude $\left\|\Sigma_r^{-\frac{1}{2}}\Delta r\right\|_2^2$ has an effect on the alarm rate.

Lemma 4.5.2 *The following alarm probability* (τ)

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2 > \tau),$$

is approximately increasing in $\left\|\Sigma_r^{-\frac{1}{2}}(\Delta r)\right\|_2^2 = \left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2$.

Proof. The random variable $\Sigma_r^{-\frac{1}{2}}(r+\Delta r)$ is Gaussian with the mean value $\Sigma_r^{-\frac{1}{2}}(\Delta r) =$

$\Sigma_r^{-\frac{1}{2}} G_r B_a a_u$ and the covariance matrix $\mathbb{E}\{\Sigma_r^{-\frac{1}{2}} r(k) r^T(k) (\Sigma_r^{-\frac{1}{2}})^T\} = I_p$. Hence, $\left\|\Sigma_r^{-\frac{1}{2}}(r(k) + \Delta r)\right\|_2^2$ follows the non-central chi-squared distribution as explained by two parameters that are the degrees of freedom n (the dimension of the random variable $r(k) \in \mathbb{R}^p$), and the non-centrality parameter (the magnitude of the mean value $\left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2$). The distribution function of the non-central chi-squared random variable is represented by the alarm probability. This function is equal to Marcum Q-function that is defined in [124]. The generalized Marcum Q-function increases in the non-centrality parameter for $p > 0$. Therefore, the non-centrality parameter is equal to $\left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2$. □

After finding the solution for the optimization Problem 3, the next step is to prove that Problem 3 can be transformed into equation (4.40)

Theorem 4.5.3 *If Assumption 4.5.3 holds, then there exists $\delta \in \mathbb{R}$ such that Problem 3 and Problem 4 are equivalent.*

Let $\bar{r} \in \mathbb{R}^P$ be a unit vector and δ satisfies the following criterion

$$\mathbb{P}\left(\left\|\Sigma_r^{-\frac{1}{2}}(r(k) + \delta r)\right\|_2^2 > \tau\right) \leq \alpha + \Delta\alpha. \tag{4.41}$$

The alarm probability equals to the Marcum Q-function, and this function is continuous and increasing in δ^2 for $p, \tau > 0$ [124]. Assume that there exists a_u that satisfies the constraint in equation (4.22)

$$\mathbb{P}\left(\left\|\Sigma_r^{-\frac{1}{2}}(r(k) + G_r B_a a_u)\right\|_2^2 > \tau\right) \leq \alpha + \Delta\alpha. \tag{4.42}$$

Consider the constraint was defined in equation (4.40),

$$\left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2 > \delta^2.$$

Also, we have

$$\left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2 > \delta^2 = \|\delta\bar{r}\|_2^2 .$$

From equation (4.41) and equation (4.42), we have the following

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k) + G_r B_a a_u)\right\|_2^2 > \tau) \leq \mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k) + \delta r)\right\|_2^2 > \tau). \tag{4.43}$$

This is in contradiction with Lemma (4.5.2) in which the alarm rate is increasing in $\left\|\Sigma_r^{-\frac{1}{2}}(G_r B_a a_u)\right\|_2^2$.

4.5.2.1 Mitigating the Impact of the Actuators Attack

Assume that the attacker is able to increase the error arbitrarily. A sufficient condition for the existence of $B_a a_u \neq 0$ such that $\Sigma_r^{-\frac{1}{2}} G_r B_a a_u = \Sigma_r^{-\frac{1}{2}} \Delta r = 0$. From equation (4.34), $B_a a_u = C\Delta\hat{x}_u$ and the solution for Problem 4 is analyzed as follows [117]:

$$\begin{aligned} \Delta\hat{x}_u &= (A - KC)\Delta\hat{x}_u + B_a a_u, \\ &= A\Delta\hat{x}_u + (B_a a_u - KC\Delta\hat{x}_u) = A\Delta\hat{x}_u, \end{aligned} \tag{4.44}$$

Note that the attack mitigation is borrowed from equation (4.33)

Corollary 4.5.2 *The matrix $G_{\hat{x}}$ is positive definite. Hence, null$\{G_{\hat{x}}\} = 0$.*

Hence, Problem 4 has analytical solution (provided below) that is bounded.

$$a_u^* = \pm \frac{\delta}{\left\| \Sigma_r^{-\frac{1}{2}} (G_r B_a \nu^*) \right\|_2} \nu^*,$$

where ν^* is the unit length generalized eigenvector that is equivalent to the maximal generalized eigenvalue λ^* of the pencil matrix is obtained by

$$(B_a^T G_{\hat{x}}^T G_{\hat{x}} B_a, B_a^T G_r^T \Sigma_r^{-1} G_r B_a). \tag{4.45}$$

Also, the maximal increase of mean squared error is defined as

$$\|G_{\hat{x}} B_a a_u^*\|_2^2 = \lambda^* \delta^2. \tag{4.46}$$

Since in general we do not know how much is the attacker willing to risk, λ^* can be used as an estimate of the attack impact in equation (4.46). The corresponding simulation results are provided in Section 4.6.2.

4.5.3 Case 3: Secured Sensors and Actuators

In this scenario, we consider BIAs that are more complex than previous two cases by considering both the actuator and sensor attack. Our contribution is to provide the attack impact and derive a lower bound of the impact attack in term of the combination number of sensors and actuators. The attacker can inject attack sequences to both sensors and actuators of

the system as follows:

$$\begin{aligned}
\Delta\hat{x}(k+1) &= \Delta\hat{x}_y(k+1) + \Delta\hat{x}_u(k+1),\\
\Delta\hat{x}(k+1) &= (A-KC)\Delta\hat{x}(k) + KDa_y(k) + B_a a_u(k),\\
\Delta r(k) &= Da_y(k) - C\Delta\hat{x}(k).
\end{aligned} \tag{4.47}$$

$$\Delta\hat{x} = (A-KC)\Delta\hat{x} + KDa_y + B_a a_u, \tag{4.48}$$

Since the matrix $A-KC$ is stable and the matrix $I_n - A + KC$ is invertible, hence

$$\Delta\hat{x} = (I_n - A + KC)^{-1}(KDa_y + B_a a_u) = G_{\hat{x}}KDa_y + G_{\hat{x}}B_a a_u, \tag{4.49}$$

$$\Delta r = Da_y - C(G_{\hat{x}}KDa_y + G_{\hat{x}}B_a a_u) = G_r Da_y - CG_{\hat{x}}B_a a_u, \tag{4.50}$$

$$G_r = (I_p - CG_{\hat{x}}K),$$

$$G_{\hat{x}} = (I_n - A + KC)^{-1}.$$

The attacker problem in Case 3 is now given below:

Problem 5:

$$\begin{aligned}
&\max_{a\in\mathbb{R}^{m_a\times n_a}} \left\|G_{\hat{x}}KDa_y + G_{\hat{x}}B_a a_u\right\|_2^2,\\
s.t. \qquad &\mathbb{P}\left(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2 > \tau\right) \le \alpha + \Delta\alpha.
\end{aligned} \tag{4.51}$$

where $\Delta\alpha > 0$, $\alpha + \Delta\alpha \le 1$ is a threshold and the alarm probability in the presence of attacks raises to $\alpha + \Delta\alpha$ and the attack will remain undetected. Few assumptions will be considered for the worst case scenario for the attacker.

Assumption 4.5.4 *The attacker knows the structure of Problem 5 and the attacker is able to control all the sensors and actuators that are not secured.*

Problem 6:

$$\begin{aligned} &\max_{a\in\mathbb{R}^{m_a\times n_a}} \left\|G_{\hat{x}}KDa_y + G_{\hat{x}}B_a a_u\right\|_2^2, \\ s.t. \qquad &\left\|\Sigma_r^{-\frac{1}{2}}(G_r Da_y - CG_{\hat{x}}B_a a_u)\right\|_2^2 \leq \delta^2. \end{aligned} \tag{4.52}$$

The attack can induce unbounded estimation error by the measuring a_u and a_y. We have to look at the space of a_u and a_y. Problem 6 for Case 3 can be defined as follows:

$$\begin{aligned} &\max_{a\in\mathbb{R}^{m_a\times n_a}} \left\|G_{x_a}a\right\|_2^2, \\ s.t. \qquad &\left\|\Sigma_r^{-\frac{1}{2}}(G_{r_a}a)\right\|_2^2 \leq \delta^2, \end{aligned} \tag{4.53}$$

where matrices G_{x_a} and G_{r_a} are given below

$$G_{x_a} = \begin{bmatrix} G_{\hat{x}}KD & 0 \\ 0 & G_{\hat{x}}B_a \end{bmatrix},$$

$$G_{r_a} = \begin{bmatrix} \Sigma_r^{-\frac{1}{2}}G_r D & 0 \\ 0 & -\Sigma_r^{-\frac{1}{2}}CG_{\hat{x}}B_a \end{bmatrix},$$

Lemma 4.5.3 *The optimization problem (4.53) is bounded if and only if* $\ker\{G_{r_a}\} \subset \ker\{G_{x_a}\}$

where a is defined as

$$a = \begin{bmatrix} a_y \\ a_u \end{bmatrix}.$$

Lemma 4.5.4 *The following alarm probability*

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2 > \tau),$$

is approximately increasing in $\left\|\Sigma_r^{-\frac{1}{2}}(\Delta r)\right\|_2^2 = \left\|\Sigma_r^{-\frac{1}{2}}(G_{r_a}a)\right\|_2^2$.

Proof. The random variable $\Sigma_r^{-\frac{1}{2}}(r+\Delta r)$ is Gaussian with the mean value $\Sigma_r^{-\frac{1}{2}}(\Delta r) = \Sigma_r^{-\frac{1}{2}}G_{r_a}a$ and the covariance matrix $\mathbb{E}\{\Sigma_r^{-\frac{1}{2}}r(k)r^T(k)(\Sigma_r^{-\frac{1}{2}})^T\} = I_p$. Hence, $\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\Delta r)\right\|_2^2$ follows the non-central chi-squared distribution. The non-centrality parameter is the magnitude of the mean value $\left\|\Sigma_r^{-\frac{1}{2}}(G_{r_a}a)\right\|_2^2$. □

Theorem 4.5.4 *If Assumption 4.5.4 holds, then there exists* $\delta \in \mathbb{R}$ *such that Problem 5 and Problem 6 are equivalent.*

The proof is similar to Theorem 4.3.3.

Let $\bar{r} \in \mathbb{R}^P$ be a unit vector and δ satisfies the following criterion

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\delta r)\right\|_2^2 > \tau) \leq \alpha + \Delta\alpha. \tag{4.54}$$

Assume that there exists a that satisfies the constraint in equation (4.25)

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+G_{r_a}a)\right\|_2^2 > \tau) \leq \alpha + \Delta\alpha. \tag{4.55}$$

Consider the constraint that was defined in equation (4.52),

$$\left\|\Sigma_r^{-\frac{1}{2}}(G_{r_a}a)\right\|_2^2 > \delta^2.$$

Also, we have

$$\left\|\Sigma_r^{-\frac{1}{2}}(G_{r_a}a)\right\|_2^2 > \delta^2 = \|\delta\bar{r}\|_2^2,$$

From equation (4.54) and equation (4.55), we have the following

$$\mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+G_{r_a}a)\right\|_2^2 > \tau) \leq \mathbb{P}(\left\|\Sigma_r^{-\frac{1}{2}}(r(k)+\delta r)\right\|_2^2 > \tau). \tag{4.56}$$

This is in contradiction with Lemma (4.5.4) in which the alarm rate is increasing in $\left\|\Sigma_r^{-\frac{1}{2}}(G_{r_a}a)\right\|_2^2$.

4.5.3.1 Mitigating the Impact of Sensors and Actuators Attacks

Here, the optimal combination of secured sensors and actuators is proposed. Moreover, a condition on the necessary number of the sensors and actuators to be secured is considered, so that the impact of the BIA is less than the desired threshold. Assume that the attacker is able to increase the error arbitrarily.

$$\begin{aligned}\Delta\hat{x} &= (A-KC)\Delta\hat{x} + KDa_y + B_a a_u,\\ &= A\Delta\hat{x} + K(Da_y - C\Delta\hat{x}) + B_a a_u.\end{aligned} \tag{4.57}$$

Assumption 4.5.5 *We assume A does not have an eigenvalue of 1.*

Corollary 4.5.3 *We know from corollaries 4.5.1 and 4.5.2 that null$\{G_x\} = \{\varnothing\}$ and if A*

does not have an eigenvalue of 1, the we have null$\{G_r\} = \{\varnothing\}$. *Moreover, according to Lemma 4.5.3, for boundedness of optimization in equation (4.5.3), we have to prove that* $\ker\{G_{r_a}\} \subset \ker\{G_{x_a}\}$, *the matrix* G_{r_a} *is a Jordan-block matrix and its eigenvalues are the union of eigenvalues of matrices* $-\Sigma_r^{-\frac{1}{2}} C G_{\hat{x}} B_a$, $\Sigma_r^{-\frac{1}{2}} G_r D$ *and* G_{x_a} *is Jordan-block too. Now, assuming A does not have eigenvalue equal to one, it follows that* $\ker\{G_{r_a}\} \subset \ker\{G_{x_a}\}$.

The solution of Problem 6 under Assumption 4.5.5 is given in [116].

Theorem 4.5.5 *Suppose Assumption 4.5.5 holds. Then, the solution of Problem 6 is given by*

$$a^* = \pm \frac{\delta}{\left\| \Sigma_r^{-\frac{1}{2}} (G_{r_a} \nu^*) \right\|_2} \nu^*,$$

where ν^* is the unit generalized eigenvector that corresponds to the maximal generalized λ^* of the pencil in Theorem 4.4.1.

$$(G_{x_a}^T G_{x_a}, G_{r_a}^T \Sigma_r^{-1} G_{r_a}), \tag{4.58}$$

and the maximal increase of mean squared error is

$$\left\| \Sigma_r^{-\frac{1}{2}} (G_{x_a} a^*) \right\|_2^2 = \lambda^* \delta^2. \tag{4.59}$$

The results for the different combination of the secured sensors and actuators are provided in Section 4.6.3.

4.6 Simulation Result

To demonstrate the effectiveness of the proposed results a linearized model of the quadruple tank process is used. The parameters of the linear time-invariant system that is defined in equation (4.1) are chosen as follows [117]:

$$A = \begin{bmatrix} 0.9 & 0 & 0.3 & 0 \\ 0 & 0.9 & 0 & 0.3 \\ 0 & 0 & 0.9 & 0 \\ 0 & 0 & 0 & 0.9 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\Sigma_\nu = 10^{-2} \begin{bmatrix} 0.8 & 0.2 & 2.7 & 0.7 \\ 0.2 & 0.8 & 0.7 & 2.7 \\ 2.7 & 0.7 & 9.0 & 2.3 \\ 0.7 & 2.7 & 2.3 & 9.0 \end{bmatrix}, \quad \Sigma_w = \begin{bmatrix} 2.5 & 0 & 0 & 0 \\ 0 & 2.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix},$$

and $\delta^2 = 1$ when the attack impact in equation (4.59) is equal to the largest generalized eigenvalue of A.

4.6.1 Case 1 (Secured Sensors)

In this case, sensor attacks are considered. When none of the sensors is secured, the matrix $D = I_p$ will be an identity matrix, and the pencil matrix is equal to $(G_{\hat{x}}^T G_{\hat{x}}, G_{\hat{r}}^T \Sigma_r^{-1} G_{\hat{r}}^T)$. The generalized eigenvalues is defined as $\lambda = \{0.0156, 0.021, 65.45, 116.32\}$.

We consider for simplicity $\delta^2 = 1$. The maximal generalized eigenvalue for different

combination of secured sensors are provided in Table 4.1. Since, the second largest generalized eigenvalue is $\lambda_3 = 65.45$ and the third largest eigenvalue equals 0.02, so, the necessary number of sensors should be secured that will be analyzed. When Sensors $\{1, 3\}$ are secured, reducing maximal impact $116.32/1.43 \approx 81$ times in Table 4.1. From Table 4.1, we can see that Sensor 3 is more important to be secured than Sensor 2. We can see considerable estimation errors in scenarios where Sensors $\{1, 2\}$ ($\lambda^* = 2.68$) and Sensors $\{1, 3\}$ ($\lambda^* = 84.49$) are secured in Table 4.1. As a result, when the sensor has more noise, it is important to be secured under an attack.

Table 4.1: Maximal generalized eigenvalues for different combination of secured sensors.

	Secured Sensors	λ^*		Secured Sensors	λ^*
1	$\{1, 2, 3, 4\}$	0	9	{1,4}	2.15
2	$\varnothing$	116.32	10	{2,3}	2.15
3	{1}	84.57	11	{2,4}	84.49
4	{2}	84.73	12	{3,4}	1.43
5	{3}	84.73	13	{1,2,3}	2.14
6	{4}	84.73	14	{1,2,4}	2.14
7	{1,2}	2.68	15	{1,3,4}	1.42
8	{1,3}	84.49	16	{2,3,4}	1.42

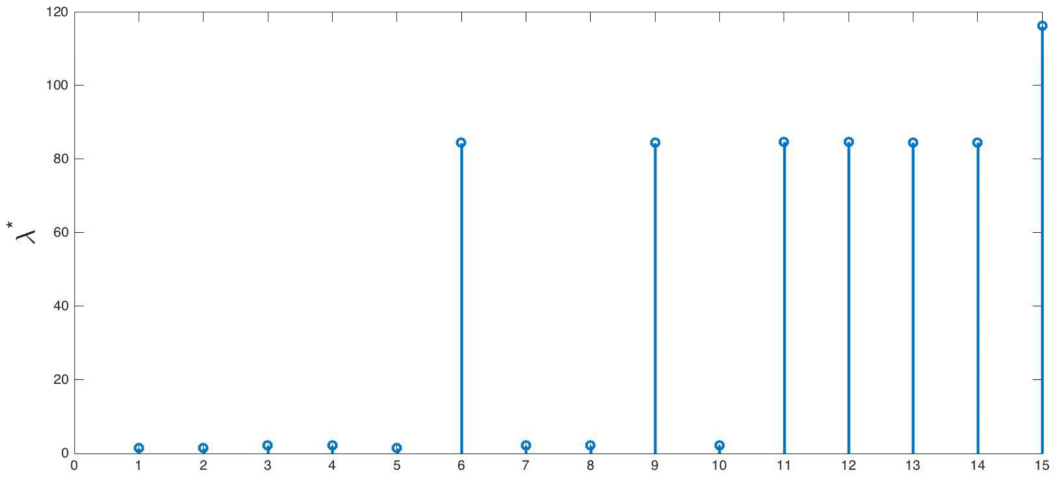


Figure 4.2: Maximal generalized eigenvalues for different combination of secured sensors.

4.6.2 Case 2 (Secured Actuators)

In this case, actuator attacks are considered. If none of actuators are secured, then $B_a = I_n$ and the pencil matrix in Theorem 4.4.1 is equal to

$$(B_a^T G_{\hat{x}}^T G_{\hat{x}} B_a, B_a^T G_r^T \Sigma_r^{-1} G_r B_a).$$

and the worst case eigenvalue is equal to 2.7 as shown in Table 4.2 as $\lambda = \{0.65, 0.72, 2.7, 2.72\}$.

The maximal generalized eigenvalue for different combination of actuators are shown in Table 4.2.

Table 4.2: Maximal generalized eigenvalues for different combination of the secured actuators.

	Secured Actuators	λ^*		Secured Actuators	λ^*
1	$\{1, 2, 3, 4\}$	0	9	{1,4}	2.52
2	$\varnothing$	2.72	10	{2,3}	2.52
3	{1}	2.71	11	{2,4}	2.70
4	{2}	2.71	12	{3,4}	2.52
5	{3}	2.71	13	{1,2,3}	0.77
6	{1,2}	2.7	14	{1,2,4}	0.77
7	{1,3}	2.70	15	$\{1, 3, 4\}$	2.51
8	{1,4}	2.52	16	{2,3,4}	2.51

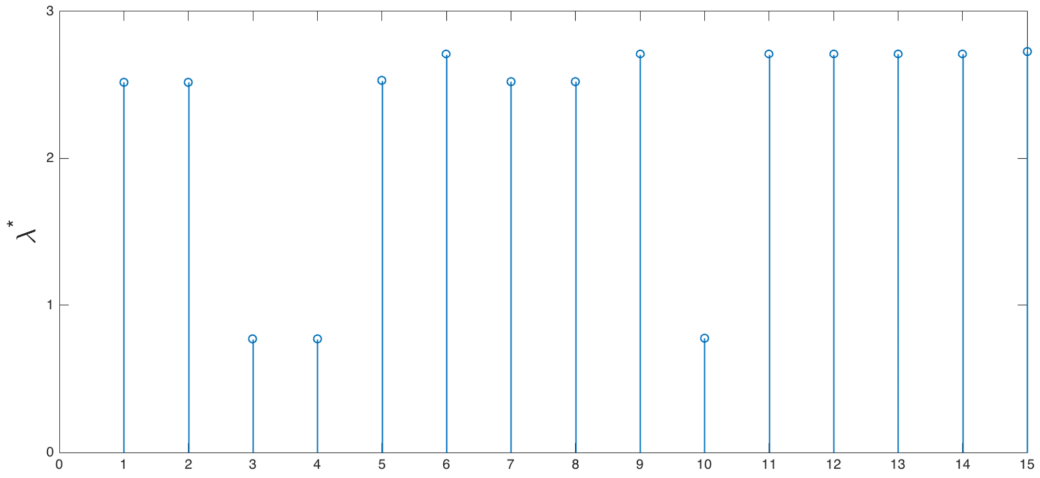


Figure 4.3: Maximal generalized eigenvalues for different combination of secured actuators.

4.6.3 Case 3 (Secured Sensors and Actuators)

In this case, bias injection attacks on both sensors and actuators are considered that is more complex than the previous two cases. None of the sensors and actuators are secured, hence $D = I_p$ and $B_a = I_n$. Moreover, the pencil matrix is equal to

$$(G_{x_a}^T G_{x_a}, G_{r_a}^T \Sigma_r^{-1} G_{r_a}).$$

The generalized eigenvalues are $\lambda_1 = 0.015$, $\lambda_2 = 0.021$, $\lambda_3 = 0.65$, $\lambda_4 = 0.73$, $\lambda_5 = 2.69$, $\lambda_6 = 2.72$, $\lambda_7 = 65.45$ and $\lambda_8 = 116.32$. The maximum generalized eigenvalues are calculated for different combination of secured sensors and actuators and they are provided in Tables 4.3, 4.4, and 4.5. As a result, if the number of secured actuators and sensors are increased, we do not have less estimation error. The minimum estimation error depends on which sensors and actuators are secured. The main observation is in Table (4.5) when the Sensor $\{1, 2\}$ are secured in which the estimation error has the least value.

Table 4.3: Maximal generalized eigenvalues for different combination of the secured sensors and actuators.

	Secured Sensors (SS)	Secured Actuators (SA)	λ^*		SS	SA	λ^*
91	{1,2,3,4}	{2}	2.71	125	{2,3,4}	{3}	2.71
92	{1,2,3,4}	{1}	2.71	126	{2,3,4}	{3}	2.71
93		{1,2,3,4}	116.32	127	{2,3,4}	{1}	2.71
94	{4}	{2,3,4}	84.73	128	{1}	{2,3,4}	84.57
95	{4}	{1,3,4}	84.73	129	{1}	{1,3,4}	84.57
96	{4}	{1,2,4}	84.73	130	{1}	{1,2,4}	84.57
97	{4}	{1,2,3}	84.73	131	{1}	{1,2,4}	84.57
98	{3}	{2,3,4}	84.73	132	{1,4}	{3,4}	2.52
99	{3}	{1,3,4}	84.73	133	{1,4}	{2,4}	2.70
100	{3}	{1,2,4}	84.73	134	{1,4}	{2,3}	2.52
101	{3}	{!,2,3}	84.73	135	{1,4}	{1,4}	2.52
102	{3,4}	{3,4}	3.52	136	{1,4}	{1,3}	2.70
103	{3,4}	{2,4}	2.70	137	{1,4}	{1,2}	2.15
104	{3,4}	{2,3}	2.52	138	{1,3}	{3,4}	84.49
105	{3,4}	{1,4}	2.52	139	{1,3}	{2,4}	84.49
140	{1,3}	{2,3}	84.49	145	{1,3,4}	{3}	2.71
141	{1,3}	{1,2}	84.49	146	{1,3,4}	{2}	2.71
142	{1,3}	{1,3}	84.49	147	{1,3,4}	{1}	2.71

Table 4.4: Maximal generalized eigenvalues for different combination of secured sensors and actuators.

	SS	SA	λ^*		SS	SA	λ^*
143	{1,3}	{1,2}	84.49	148	{1,2}	{3,4}	2.52
144	{1,3,4}	{4}	2.71	149	{1,2}	{2,4}	2.52
151	{1,2}	{3,4}	2.52	186	{3,4}	{1,4}	84.57
152	{1,2}	{1,3}	2.72	187	{3,4}	{1,3}	84.57
153	{1,2}	{1,2}	2.26	188	{2}	{1,2}	84.57
154	{1,2,4}	{4}	2.71	189	{2,4}	{4}	84.57
155	{1,2,4}	{3}	2.71	190	{2,4}	{3}	84.57
156	{1,2,4}	{2}	2.71	191	{2,4}	{2}	84.57
157	{1,2,4}	{1}	2.71	192	{2,4}	{1}	84.57
158	{1,2,3}	{3}	2.71	193	{2,3}	{4}	2.71
159	{4}	{3}	2.71	194	{2,3}	{3}	2.71
160	{4}	{2}	2.71	195	{2,3}	{2}	2.71
161	{4}	{2}	2.71	196	{2,3}	{1}	2.71
162	{1,2,3,4}		2.71	197	{1,2,3}	{1}	2.71
163		{2,3,4}	116.3	198	{2,3,4}	{1}	2.71
164		{1,3,4}	116.3	199	{1,4}	{2,4}	84.57
165		{1,2,4}	116.3	200	{1}	{2,3}	84.57
166		{1,2,3}	116.3	201	{1}	{1,4}	84.57

Table 4.5: Maximal generalized eigenvalue for different combination of secured sensors and actuators.

	SS	SA	λ^*		SS	SA	λ^*
221		{2,3}	116.32	239	{2,3}		2.72
222		{1,4}	116.32	240	{1}	{4}	84.57
223		{1,3}	116.32	241	{1}	{3}	84.57
224		{1,2}	84.73	242	{1}	{2}	84.57
225	{4}	{4}	84.73	243	{1}	{1}	84.57
226	{4}	{3}	84.73	244	{1,4}		2.72
227	{4}	{2}	84.73	245	{1,3}		84.49
228	{4}	{1}	84.73	246	{1,2}		2.72
229	{3}	{4}	84.73	247		{4}	116.32
230	{3}	{3}	84.73	248		{3}	116.32
231	{3}	{2}	84.73	249		{2}	116.32
232	{3}	{1}	84.73	250		{1}	116.32
233	{3,4}		2.71	251	{4}		84.73
234	{2}	{4}	84.57	252	{3}		84.73
235	{2}	{3}	84.57	253	{2}		84.73
236	{2}	{2}	84.57	254	{1}		84.73
237	{2}	{1}	84.57	255	$\varnothing$	$\varnothing$	116.32
238	{2,4}		84.57	256	{1,2,3,4}	{1,2,3,4}	0

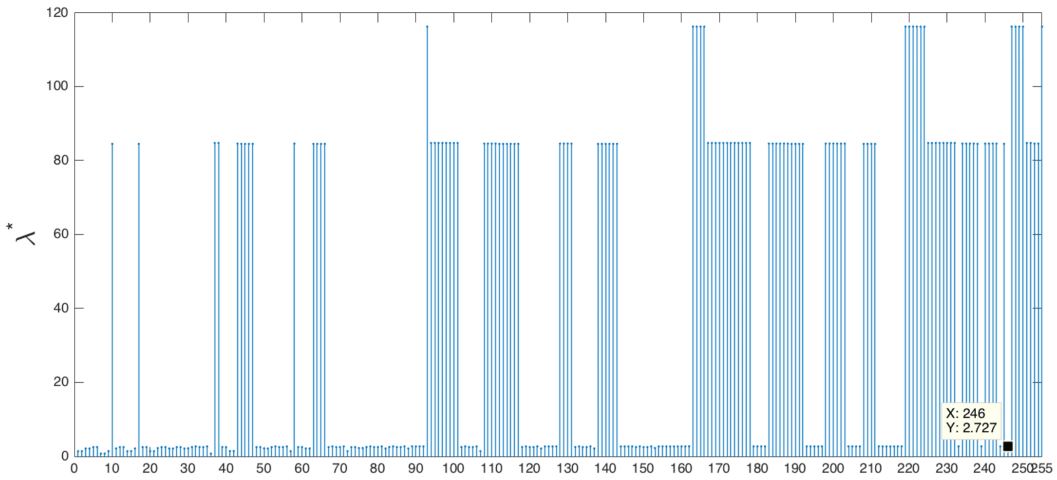


Figure 4.4: Maximal generalized eigenvalue for different combination of the secured sensors and actuators.

4.7 Conclusion

In this chapter, the main underling problem considered is that it is expensive to secure all sensors and actuators. Hence, we have to choose which sensors and actuators should be secured. The final goal is to find the optimal combination of sensors and actuators that has the minimum error in the output or to maintain an estimation error lower than an upper bound. Since, the worst case scenario is considered, it is assumed that attackers have the knowledge of the system. Therefore, an optimization problem is considered from attacker's point of view in which the attacker tries to the maximize the estimation error in the induced plant while remaining undetected. The problem of estimating the impact of BIA was also presented. The system consisted of a noisy plant, a Kalman filter, and an χ^2-squared anomaly detector. Moreover, the lower bound of the attack impact is also investigated. The

corresponding simulation results were provided in Sections 4.6.1, 4.6.2 and 4.6.3. The main observation from Table 4.5 is that Sensors $\{1, 2\}$ are secured with the minimum estimation error. For the combination of sensors and actuators, Scenario 194 with an estimation error of 2.71 can be considered in Table 4.4. Other scenarios with different estimation errors can also be analyzed. For future work, a more general system model such as linear time-varying or nonlinear and hybrid can be studied, and consequently, the complexity of network topology and protocols can be extended.

Chapter 5

Conclusions and Future Work

In this chapter, we summarize the results of this thesis and present potential directions for future work.

5.1 Conclusion

- **In Chapter 3**, possible detection and isolation fault strategy and deception attack techniques on a formation control of Autonomous Underwater Vehicles (AUVs) were presented. The distributed fault detection scheme based on the Generalized Likelihood Ratio (GLR) and the Unscented Kalman Filter (UKF) were used. The developed Fault Detection (FD) methodology is able to detect a faulty AUV in the formation was presented.

- **In Chapter 3**, the system was under both actuator (LOE), bias sensor fault as well as a deception attack. The consensus problem of multi AUVs was also studied. A method

to detect and isolate faulty agents under a deception attack was proposed in Figure 3.5. The performance of the developed FD and AD schemes are analyzed and the confusion matrix was provided in Tables 3.5, 3.6 and 3.7. We have simultaneously detected deception attacks and faults for each agent. All simulation results are presented in Section 3.4.

- **In Chapter 4**, the problem of estimating the impact of Bias Injection Attack (BIA) is considered. A Kalman Filter (KF) is used to estimate the system states and a χ^2-squared anomaly detector is utilized to select the secured sensors/actuators. We investigated the problem of finding the worst case Bias Injection Attack (BIA) that is reduced to a quadratic program for which the optimal value can be found using well-known algorithms.

- **In Chapter 4**, our contribution is to provide the analysis of attack impact and derive a lower bound of attack impact in terms of the number of actuators and the combination of sensors and actuators. A lower bound of the attack impact is also derived and the corresponding simulation results are illustrated.

5.2 Suggestions for Future Work

The potential future work for Chapter 3 are:

- To extend the proposed framework for various stochastic switching network typologies where the network architecture of agents are changing over time.

- To extend the proposed methodology for homogeneous agents with the disturbance and uncertainties in FDI estimations.
- To extend the proposed framework for outage faults in a team of nonlinear agents.
- To use 6-DOF AUV model resulting in a wider application in underwater missions.
- To extend the proposed FTC to address various types of faults in actuators such as lock-in-place, float, and hard over.
- To extend the proposed framework for different types of attack such as covert, replay and zero dynamic attacks.
- To extend the design of the proposed fault-tolerant control schemes for scenarios with fewer redundancy.
- To improve the performance of the proposed methodologies.

Potential future directions for Chapter 4 are:

- To work on a more general system model such as linear time-varying or nonlinear and hybrid.
- To conduct a detailed analysis of cyber attacks within the Cyber-Physical System (CPS) network. Compared to the single agent, analyzing the impact of cyber attacks on CPS is more challenging due to the complexity of network topology and protocols.

Bibliography

[1] O. Saif, I. Fantoni, and A. Zavala-Rio, "Flocking of multiple unmanned aerial vehicles by lqr control," in *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on*. IEEE, 2014, pp. 222–228.

[2] Y. F. Chen, N. K. Ure, G. Chowdhary, J. P. How, and J. Vian, "Planning for large-scale multiagent problems via hierarchical decomposition with applications to uav health management," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 1279–1285.

[3] Y. Zhang and H. Mehrjerdi, "A survey on multiple unmanned vehicles formation control and coordination: normal and fault situations," in *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*. IEEE, 2013, pp. 1087–1096.

[4] M. Davoodi, N. Meskin, and K. Khorasani, "A single dynamic observer-based module for design of simultaneous fault detection, isolation and tracking control scheme," *International Journal of Control*, pp. 1–16, 2017.

[5] A. Souliman, A. Joukhadar, H. Alturbeh, and J. F. Whidborne, "Intelligent collision avoidance for multi agent mobile robots," in *Intelligent Systems for Science and Information*. Springer, 2014, pp. 297–315.

[6] C. Kwon, "High assurance control of cyber-physical systems with application to unmanned aircraft systems," Ph.D. dissertation, Purdue University, 2017.

[7] P. Salvatori, A. Neri, C. Stallo, and F. Rispoli, "A multiple satellite fault detection approach for railway environment," in *ITS Telecommunications (ITST), 2017 15th International Conference on*. IEEE, 2017, pp. 1–5.

[8] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative control of multi-agent systems: optimal and adaptive design approaches*. Springer Science & Business Media, 2013.

[9] S. Sedaghati, "Formation control and fault accommodation for a team of autonomous underwater vehicles," 2015.

[10] D. J. Stilwell and B. E. Bishop, "Platoons of underwater vehicles," *IEEE control systems*, vol. 20, no. 6, pp. 45–52, 2000.

[11] A. Richards and J. How, "Decentralized model predictive control of cooperating uavs," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 4. IEEE, 2004, pp. 4286–4291.

[12] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE transactions on robotics and automation*, vol. 14, no. 6, pp. 926–939, 1998.

[13] I. Shames, A. M. Teixeira, H. Sandberg, and K. H. Johansson, "Distributed fault detection for interconnected second-order systems," *Automatica*, vol. 47, no. 12, pp. 2757–2764, 2011.

[14] Y. Azuma and T. Ohtsuka, "Receding horizon nash game approach for distributed nonlinear control," in *SICE Annual Conference (SICE), 2011 Proceedings of*. IEEE, 2011, pp. 380–384.

[15] W. B. Dunbar, "Distributed receding horizon control of dynamically coupled nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1249–1263, 2007.

[16] E. Franco, L. Magni, T. Parisini, M. M. Polycarpou, and D. M. Raimondo, "Cooperative constrained control of distributed agents with nonlinear dynamics and delayed information exchange: A stabilizing receding-horizon approach," *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 324–338, 2008.

[17] C. Langbort, R. S. Chandra, and R. D'Andrea, "Distributed control design for systems interconnected over an arbitrary graph," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1502–1519, 2004.

[18] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE transactions on automatic control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[19] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on automatic control*, vol. 51, no. 3, pp. 401–420, 2006.

[20] V. Ugrinovskii, "Distributed robust estimation over randomly switching networks using h consensus," *Automatica*, vol. 49, no. 1, pp. 160–168, 2013.

[21] G. Ferrari-Trecate, L. Galbusera, M. P. E. Marciandi, and R. Scattolini, "Model predictive control schemes for consensus in multi-agent systems with single-and double-integrator dynamics," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2560–2572, 2009.

[22] P. Banala, *Output feedback H-infinity control design for multi-agent systems*. Southern Illinois University at Carbondale, 2011.

[23] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 933–941, 2003.

[24] R. W. Beard, J. Lawton, F. Y. Hadaegh *et al.*, "A coordination architecture for spacecraft formation control," *IEEE Transactions on control systems technology*, vol. 9, no. 6, pp. 777–790, 2001.

[25] J. P. Desai, J. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 4. IEEE, 1998, pp. 2864–2869.

[26] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *IEEE transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908, 2001.

[27] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 3. IEEE, 2001, pp. 2968–2973.

[28] P. Ogren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic control*, vol. 49, no. 8, pp. 1292–1302, 2004.

[29] H. Wang and S. Daley, "Actuator fault diagnosis: an adaptive observer-based technique," *IEEE transactions on Automatic Control*, vol. 41, no. 7, pp. 1073–1078, 1996.

[30] Y. Shang, "Consensus recovery from intentional attacks in directed nonlinear multiagent systems," *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 14, no. 6, pp. 355–361, 2013.

[31] S. Stanković, N. Ilić, Ž. Djurović, M. Stanković, and K. H. Johansson, "Consensus based overlapping decentralized fault detection and isolation," in *Control and Fault-Tolerant Systems (SysTol), 2010 Conference on*. IEEE, 2010, pp. 570–575.

[32] N. Meskin and K. Khorasani, "Actuator fault detection and isolation for a network of unmanned vehicles," *IEEE Transactions on Automatic Control*, vol. 54, no. 4, pp. 835–840, 2009.

[33] M. Davoodi, K. Khorasani, H. Talebi, and H. Momeni, "A robust semi-decentralized fault detection strategy for multi-agent systems: an application to a network of micro-air vehicles," *International Journal of Intelligent Unmanned Systems*, vol. 1, no. 1, pp. 21–35, 2013.

[34] S. Bezzaoucha, B. Marx, D. Maquin, and J. Ragot, "Linear feedback control input under actuator saturation: A takagi-sugeno approach," in *2nd International Conference on Systems and Control, ICSC 2012*, 2012, p. CDROM.

[35] H. Dong, Z. Wang, S. X. Ding, and H. Gao, "Finite-horizon estimation of randomly occurring faults for a class of nonlinear time-varying systems," *Automatica*, vol. 50, no. 12, pp. 3182–3189, 2014.

[36] Q. Zhang, M. Basseville, and A. Benveniste, "Fault detection and isolation in nonlinear dynamic systems: A combined input–output and local approach," *Automatica*, vol. 34, no. 11, pp. 1359–1373, 1998.

[37] X.-G. Yan and C. Edwards, "Nonlinear robust fault reconstruction and estimation using a sliding mode observer," *Automatica*, vol. 43, no. 9, pp. 1605–1614, 2007.

[38] X. He, Z. Wang, Y. Liu, and D. Zhou, "Least-squares fault detection and diagnosis for networked sensing systems using a direct state estimation approach," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1670–1679, 2013.

[39] K. C. Daly, E. Gai, and J. V. Harrison, "Generalized likelihood test for fdi in redundant sensor configurations," *Journal of Guidance, Control, and Dynamics*, 2012.

[40] S. Azizi and K. Khorasani, "Cooperative actuator fault accommodation in formation flight of unmanned vehicles using relative measurements," *International Journal of Control*, vol. 84, no. 5, pp. 876–894, 2011.

[41] ——, "A hierarchical architecture for cooperative actuator fault estimation and accommodation of formation flying satellites in deep space," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1428–1450, 2012.

[42] H. Li, X. Liao, T. Huang, W. Zhu, and Y. Liu, "Second-order global consensus in multiagent networks with random directional link failure," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 3, pp. 565–575, 2015.

[43] W. Yu, G. Chen, and M. Cao, "Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems," *Automatica*, vol. 46, no. 6, pp. 1089–1095, 2010.

[44] A. A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems." in *HotSec*, 2008.

[45] S. Gorman, "Electricity grid in us penetrated by spies," *The Wall Street Journal*, vol. 8, 2009.

[46] A. Giani, S. Sastry, K. H. Johansson, and H. Sandberg, "The viking project: an initiative on resilient control of power networks," in *Resilient Control Systems, 2009. ISRCS'09. 2nd International Symposium on.* IEEE, 2009, pp. 31–35.

[47] V. D. Gligor, "A note on denial-of-service in operating systems," *IEEE Transactions on Software Engineering*, no. 3, pp. 320–324, 1984.

[48] S. Amin, A. A. Cárdenas, and S. Sastry, "Safe and secure networked control systems under denial-of-service attacks." in *HSCC*, vol. 5469. Springer, 2009, pp. 31–45.

[49] Q. Zhu and T. Başar, "Robust and resilient control design for cyber-physical systems with an application to power systems," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on.* IEEE, 2011, pp. 4066–4071.

[50] A. Gupta, C. Langbort, and T. Başar, "Optimal control in the presence of an intelligent jammer with limited actions," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 1096–1101.

[51] Y. Mo, E. Garone, A. Casavola, and B. Sinopoli, "False data injection attacks against state estimation in wireless sensor networks," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 5967–5972.

[52] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, p. 13, 2011.

[53] V. M. Igure and R. D. Williams, "Taxonomies of attacks and vulnerabilities in computer systems," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 1, 2008.

[54] D. Ding, G. Wei, S. Zhang, Y. Liu, and F. E. Alsaadi, "On scheduling of deception attacks for discrete-time networked systems equipped with attack detectors," *Neurocomputing*, vol. 219, pp. 99–106, 2017.

[55] K. Xing, S. S. R. Srinivasan, M. Jose, J. Li, X. Cheng *et al.*, "Attacks and countermeasures in sensor networks: a survey," in *Network security*. Springer, 2010, pp. 251–272.

[56] J. Hu, S. Liu, D. Ji, and S. Li, "On co-design of filter and fault estimator against randomly occurring nonlinearities and randomly occurring deception attacks," *International Journal of General Systems*, vol. 45, no. 5, pp. 619–632, 2016.

[57] Z.-H. Pang and G.-P. Liu, "Design and implementation of secure networked predictive control systems under deception attacks," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 5, pp. 1334–1342, 2012.

[58] A. Teixeira, G. Dán, H. Sandberg, and K. H. Johansson, "A cyber security study of a scada energy management system: Stealthy deception attacks on the state estimator," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 11 271–11 277, 2011.

[59] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Revealing stealthy attacks in control systems," in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*. IEEE, 2012, pp. 1806–1813.

[60] A. Teixeira, H. Sandberg, G. Dán, and K. H. Johansson, "Optimal power flow: Closing the loop over corrupted data," in *American Control Conference (ACC), 2012*. IEEE, 2012, pp. 3534–3540.

[61] J. Hao, R. J. Piechocki, D. Kaleshi, W. H. Chin, and Z. Fan, "Sparse malicious false data injection attacks and defense mechanisms in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1–12, 2015.

[62] Y. Liu and Y. Jia, "H consensus control of multi-agent systems with switching topology: a dynamic output feedback protocol," *International Journal of Control*, vol. 83, no. 3, pp. 527–537, 2010.

[63] A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. S. Sastry, "Cyber security analysis of state estimators in electric power systems," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 5991–5998.

[64] S. Bijani and D. Robertson, "A review of attacks and security approaches in open multi-agent systems," *Artificial Intelligence Review*, pp. 1–30, 2014.

[65] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*. IEEE, 2009, pp. 911–918.

[66] Y. Mo, R. Chabukswar, and B. Sinopoli, "Detecting integrity attacks on scada systems," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1396–1407, 2014.

[67] L. Ni, "Fault-tolerant control of unmanned underwater vehicles," 2001.

[68] R. S. Smith, "A decoupled feedback structure for covertly appropriating networked control systems," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 90–95, 2011.

[69] S. Amin, X. Litrico, S. Sastry, and A. M. Bayen, "Cyber security of water scada systemspart i: Analysis and experimentation of stealthy deception attacks," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1963–1970, 2013.

[70] R. C. Cavalcante, I. I. Bittencourt, A. P. da Silva, M. Silva, E. Costa, and R. Santos, "A survey of security in multi-agent systems," *Expert Systems with Applications*, vol. 39, no. 5, pp. 4835–4846, 2012.

[71] B. Bakhache and I. Nikiforov, "Reliable detection of faults in measurement systems," *International Journal of adaptive control and signal processing*, vol. 14, no. 7, pp. 683–700, 2000.

[72] B. K. Guepie, L. Fillatre, and I. Nikiforov, "Detecting an abrupt change of finite duration," in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*. IEEE, 2012, pp. 1930–1934.

[73] ——, "Sequential monitoring of water distribution network," *IFAC Proceedings Volumes*, vol. 45, no. 16, pp. 392–397, 2012.

[74] M. Pajic, J. Weimer, N. Bezzo, O. Sokolsky, G. J. Pappas, and I. Lee, "Design and implementation of attack-resilient cyberphysical systems: With a focus on attack-resilient state estimators," *IEEE Control Systems*, vol. 37, no. 2, pp. 66–81, 2017.

[75] D. Ding, Y. Shen, Y. Song, and Y. Wang, "Recursive state estimation for discrete time-varying stochastic nonlinear systems with randomly occurring deception attacks," *International Journal of General Systems*, vol. 45, no. 5, pp. 548–560, 2016.

[76] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[77] I. Saboori, "Cooperative and consensus-based control for a team of multi-agent systems," Ph.D. dissertation, Concordia University, 2016.

[78] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on automatic control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[79] N. A. Lynch, *Distributed algorithms*. Morgan Kaufmann, 1996.

[80] G. Cybenko, "Dynamic load balancing for distributed memory multiprocessors," *Journal of parallel and distributed computing*, vol. 7, no. 2, pp. 279–301, 1989.

[81] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.

[82] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*. IEEE, 2005, pp. 6698–6703.

[83] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*. IEEE, 2005, pp. 8179–8184.

[84] F. Bullo, J. Cortes, and S. Martinez, *Distributed control of robotic networks: a mathematical approach to motion coordination algorithms*. Princeton University Press, 2009.

[85] 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.

[86] R. O. Saber and R. M. Murray, "Consensus protocols for networks of dynamic agents," 2003.

[87] M. Ji and M. Egerstedt, "A graph-theoretic characterization of controllability for multi-agent systems," in *American Control Conference, 2007. ACC'07*. IEEE, 2007, pp. 4588–4593.

[88] ——, "Observability and estimation in distributed sensor networks," in *Decision and Control, 2007 46th IEEE Conference on*. IEEE, 2007, pp. 4221–4226.

[89] J. Sandhu, M. Mesbahi, and T. Tsukamaki, "Relative sensing networks: observability, estimation, and the control structure," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*. IEEE, 2005, pp. 6400–6405.

[90] J. Vervoort, "Modeling and control of an unmanned underwater vehicle," 2009.

[91] I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 3, pp. 636–653, 2010.

[92] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniquespart i: Fault diagnosis with model-based and signal-based approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3757–3767, 2015.

[93] M. Ruiz, J. Colomer, and J. Meléndez, "Combination of statistical process control (spc) methods and classification strategies for situation assessment of batch process," *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, vol. 10, no. 29, 2006.

[94] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis: Part i: Quantitative model-based methods," *Computers & chemical engineering*, vol. 27, no. 3, pp. 293–311, 2003.

[95] R. N. Clark, D. C. Fosth, and V. M. Walton, "Detecting instrument malfunctions in control systems," *IEEE Transactions on Aerospace and Electronic Systems*, no. 4, pp. 465–473, 1975.

[96] J. Gertler, "Fault detection and isolation using parity relations," *Control engineering practice*, vol. 5, no. 5, pp. 653–661, 1997.

[97] R. K. Mehra and J. Peschon, "An innovations approach to fault detection and diagnosis in dynamic systems," *Automatica*, vol. 7, no. 5, pp. 637–640, 1971.

[98] M. Basseville, I. V. Nikiforov *et al.*, *Detection of abrupt changes: theory and application*. Prentice Hall Englewood Cliffs, 1993, vol. 104.

[99] J. Chen and R. J. Patton, *Robust model-based fault diagnosis for dynamic systems*. Springer Science & Business Media, 2012, vol. 3.

[100] S. Shahrokhi Tehrani, "Fault detection, isolation and identification of autonomous underwater vehicles using dynamic neural networks and genetic algorithms," Ph.D. dissertation, Concordia University, 2015.

[101] B. Amer, M. Atia, M. Hefhawi, and A. Noureldin, "An adaptive positioning system for smartphones in zigbee networks using channel decomposition and particle swarm optimization," 2015.

[102] M. Cao, A. S. Morse, C. Yu, B. D. Anderson, and S. Dasguvta, "Controlling a triangular formation of mobile autonomous agents," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 3603–3608.

[103] B. D. Anderson, C. Yu, S. Dasgupta, and A. S. Morse, "Control of a three-coleader formation in the plane," *Systems & Control Letters*, vol. 56, no. 9-10, pp. 573–578, 2007.

[104] M. Cao, C. Yu, A. S. Morse, B. Anderson, and S. Dasgupta, "Generalized controller for directed triangle formations," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 6590–6595, 2008.

[105] S. Li and X. Wang, "Quickest attack detection in multi-agent reputation systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 653–666, 2014.

[106] A. Teixeira, "Toward cyber-secure and resilient networked control systems," Ph.D. dissertation, KTH Royal Institute of Technology, 2014.

[107] A. UmaMageswari, J. J. Ignatious, and R. Vinodha, "A comparitive study of kalman filter, extended kalman filter and unscented kalman filter for harmonic analysis of the non-stationary signals," *International Journal of Scientific & Engineering Research*, vol. 3, no. 7, pp. 1–9, 2012.

[108] J. Prakash, S. C. Patwardhan, and S. Narasimhan, "A supervisory approach to fault-tolerant control of linear multivariable systems," *Industrial & Engineering Chemistry Research*, vol. 41, no. 9, pp. 2270–2281, 2002.

[109] S. C. Patwardhan, S. Manuja, S. Narasimhan, and S. L. Shah, "From data to diagnosis and control using generalized orthonormal basis filters. part ii: Model predictive and fault tolerant control," *Journal of Process Control*, vol. 16, no. 2, pp. 157–175, 2006.

[110] A. P. Deshpande and S. C. Patwardhan, "Online fault diagnosis in nonlinear systems using the multiple operating regime approach," *Industrial & Engineering Chemistry Research*, vol. 47, no. 17, pp. 6711–6726, 2008.

[111] A. Willsky and H. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Transactions on Automatic control*, vol. 21, no. 1, pp. 108–112, 1976.

[112] Z. Guo, D. Shi, K. H. Johansson, and L. Shi, "Optimal linear cyber-attack on remote state estimation," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 4–13, 2017.

[113] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: risk assessment, detection, and response," in *Proceedings of the 6th ACM symposium on information, computer and communications security*. ACM, 2011, pp. 355–366.

[114] O. Vukovic, K. C. Sou, G. Dan, and H. Sandberg, "Network-aware mitigation of data integrity attacks on power system state estimation," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 6, pp. 1108–1118, 2012.

[115] D. Ding, Z. Wang, Q.-L. Han, and G. Wei, "Security control for discrete-time stochastic nonlinear systems subject to deception attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2016.

[116] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," *Automatica*, vol. 51, pp. 135–148, 2015.

[117] J. Miloševič, T. Tanaka, H. Sandberg, and K. H. Johansson, "Analysis and mitigation of bias injection attacks against a kalman filter," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8393–8398, 2017.

[118] B. Anderson and J. Moore, "Optimal filtering. 2005."

[119] W. Xue, Y.-Q. Guo, and X.-D. Zhang, "Application of a bank of kalman filters and a robust kalman filter for aircraft engine sensor/actuator fault diagnosis," *International Journal of Innovative Computing, Information and Control*, vol. 4, no. 12, pp. 3161–3168, 2008.

[120] D. P. Malladi and J. L. Speyer, "A generalized shiryayev sequential probability ratio test for change detection and isolation," *IEEE Transactions on Automatic Control*, vol. 44, no. 8, pp. 1522–1534, 1999.

[121] I. V. Nikiforov, "A generalized change detection problem," *IEEE Transactions on Information theory*, vol. 41, no. 1, pp. 171–187, 1995.

[122] D. Dionne, H. Michalska, Y. Oshman, and J. Shinar, "Novel adaptive generalized likelihood ratio detector with application to maneuvering target tracking," *Journal of guidance, control, and dynamics*, vol. 29, no. 2, pp. 465–474, 2006.

[123] Y. Sun and Á. Baricz, "Inequalities for the generalized marcum q-function," *Applied Mathematics and Computation*, vol. 203, no. 1, pp. 134–141, 2008.

[124] Y. Sun, Á. Baricz, and S. Zhou, "On the monotonicity, log-concavity, and tight bounds of the generalized marcum and nuttall q-functions," *IEEE Transactions on Information Theory*, vol. 56, no. 3, pp. 1166–1186, 2010.

[125] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.

[126] H. Avron, E. Ng, and S. Toledo, "Using perturbed qr factorizations to solve linear least-squares problems," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 2, pp. 674–693, 2009.