UNIVERSITY OF PANNONIA

DOCTORAL THESIS

Model based fault diagnosis in networked linear time invariant systems

DOI:10.18136/PE.2024.884

Author:
Wijaya Kurniawan

Supervisors:
Prof. Katalin M. Hangos
Prof. Lorinc Marton

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

UNIVERSITY OF PANNONIA
DOCTORAL SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY

2024

MODEL BASED FAULT DIAGNOSIS IN NETWORKED LINEAR TIME INVARIANT SYSTEMS

Thesis for obtaining a PhD degree in the Doctoral School of Information Science and Technology of the University of Pannonia

in the research field of Computer Sciences

Written by WIJAYA KURNIAWAN

Supervisor(s): KATALIN M. HANGOS, LORINC MARTON

propose acceptance (yes / no) …….

(supervisor/s)

As reviewer, I propose acceptance of the thesis:

Name of Reviewer: …..... …..... yes / no

…….

(reviewer)

Name of Reviewer: …..... …..... yes / no

…….

(reviewer)

The PhD-candidate has achieved ….....% at the public discussion.

Veszprém, …….

(Chairman of the Committee)

The grade of the PhD Diploma …..... (…….. %)
Veszprém,

…….

(Chairman of UDHC)

Declaration of Authorship

I, Wijaya Kurniawan, declare that this thesis titled, "Model based fault diagnosis in networked linear time invariant systems" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Acknowledgements

First of all, I would like to express my gratitude to my supervisors Prof. Katalin M. Hangos and Prof. Lorinc Marton for their constant support and tireless help in guiding me. Their advice and patience were invaluable while I was studying.

I would also like to thank all staff members of the Doctoral School of Information Science and Technology, and the International Office of the University of Pannonia, Hungary who have provided my needs as a student here.

Next, I am also grateful to my family who have been my motivation to study. Also, I give a special thanks to my wife who was willing to let me go far away for years to carry out and complete my studies.

Last, but not least, I would like to thank the Hungarian government for allowing me to study in this beautiful country by awarding me the Stipendium Hungaricum scholarship. Additionally, my research was financially supported by the Hungarian National Research, Development, and Innovation Fund, financed under the funding scheme K_19, project no. 131501.

Abstract

As network technologies such as process networks or transportation networks become more important to today's society, the demand for their availability and reliability is increasing. A fault in one or more subsystems can be propagated through the network and will degrade the performance of the overall networked systems. Thus, Fault Detection and Isolation (FDI) must be considered there. The two main classes of networked systems studied here are robot platoons and heat exchange networks.

Since the reliability of data driven fault diagnosis methods depends on the availability and quality of the data, the model based approaches are still relevant, especially for safety critical systems. This dissertation aims to enrich the existing FDI literature for networked linear time invariant systems.

First, the model based sensor and actuator fault diagnosis methods are studied for robot platoons that move in a leader-follower scheme. For sensor fault diagnosis, a model for the robot platoons that captures the effect of fault propagation is constructed. From that model, a bank of unknown input observers is designed as residual generators to detect and isolate the sensor faults. A threshold computation approach to prevent false alarms when there are no faulty sensors is also given. Meanwhile, for actuator fault diagnosis, a filtering technique is proposed that eliminates the need for network communication in isolating the actuator fault. A Proportional Integral (PI) observer is then designed to estimate the actuator fault's magnitude.

Secondly, model based methods for parameter fault detection, isolation, and estimation in Heat Exchange Networks (HENs) are introduced. The considered fault is a significant change in the heat transfer coefficient that determines the effect of environmental temperature on the temperature inside the network. The network topology contains splitting and joining connections. The sensor placement problem for each type of connection is addressed by using the signed directed graph analysis. However, if the fault occurs only at a certain location along the length of a tube in the HEN, it is found that fault localisation can hardly be performed. A bank of linear observers for fault isolation and a bank of nonlinear observers based on parameter adaptation techniques for fault estimation are proposed.

Thirdly, a model based method for input faults in HENs that contain loops in addition to splitting and joining connections is also presented. By using steady state value analysis, the effect of fault propagation in the network loops is analysed. Based on the analysis results, a fault isolation algorithm requiring only two sensors is constructed. To estimate the fault's magnitude, a PI observer is then proposed. The proposed method shows good robustness against the heat transfer coefficient parameter uncertainty. Moreover, it can also isolate and estimate the slowly developing incipient faults.

Kivonat

Ahogy a hálózati technológiák, például a folyamathálózatok vagy a szállítási hálózatok egyre fontosabbá válnak a mai társadalom számára, egyre növekszik az igény elérhetőségük és megbízhatóságuk iránt. Egy vagy több alrendszer hibája továbbterjedhet a hálózaton, és rontja a teljes hálózati rendszer teljesítményét. Így itt használni kell a hibafelismerés és -leválasztás (FDI). A dolgozatban vizsgált hálózati rendszerek két fő osztálya a robotszakasz és a hőcserélő hálózat.

Mivel az adatvezérelt hibadiagnosztikai módszerek megbízhatósága az adatok elérhetőségétől és minőségétől függ, a modellalapú megközelítések nagy fontosságúak, különösen a biztonság szempontjából kritikus rendszerek esetében. Jelen disszertáció célja, hogy gazdagítsa a hálózatos lineáris időinvariáns rendszerek meglévő FDI-irodalmát.

Először a modell alapú szenzor- és aktuátorhiba-diagnosztikai módszereket tárgyalom olyan robotszakaszoknál, amelyek vezető-követő sémában mozognak. Az érzékelők hibáinak diagnosztizálásához a robotcsoportokhoz egy olyan modellt készítettem, amely rögzíti a hiba terjedésének hatását. Ebből a modellből egy ismeretlen bemeneti megfigyelőkből álló csoport maradék hibákat képez az érzékelő hibáinak észlelésére és elkülönítésére. Küszöbérték számítással lehetséges a téves riasztások megelőzése, ha nincsenek hibás érzékelők. Az aktuátorhiba-diagnosztikához olyan szűrési technikát javasoltam, amely kiküszöböli a hálózati kommunikáció szükségességét az aktuátor hibájának elkülönítése során. Ezután egy arányos integrál (PI) megfigyelőt terveztem, hogy megbecsülje az aktuátor hibájának nagyságát.

Modell alapú módszereket vezettem be a hőcserélő hálózatokban (HEN) paraméterhiba észlelésére, elkülönítésére és becslésére. A figyelembe vett hiba a hőátbocsátási tényező jelentős változása, amely meghatározza a környezeti hőmérséklet hatását a hálózaton belüli hőmérsékletre. A hálózati topológia elágazásokat és összefolyásokat, mint kapcsolatokat tartalmaz. Az érzékelő elhelyezési problémáját minden egyes kapcsolattípus esetében előjeles irányított gráfelemzés segítségével oldottam meg. Ha azonban a hiba a HEN-ben egy cső hosszában csak egy bizonyos helyen jelentkezik, akkor azt mutattam meg, hogy a hiba lokalizálása nem végezhető el. Javasoltam egy lineáris megfigyelőkből álló bankot a hibaérzékeléshez és egy nemlineáris megfigyelőkből álló bankot, amely paraméteradaptációs technikákon alapul a hibabecsléshez.

Bemutattam egy modellalapú módszert a HEN-ek bemeneti hibáira, amelyek a elágazási és összefolyási mellett hurkokat is tartalmaznak. Állandósult állapotú értékelemzés segítségével elemeztem a hibaterjedés hatását a hálózati hurokban. Az elemzési eredmények alapján egy csak két érzékelőt igénylő hibaleválasztó algoritmust készítettem. A hiba nagyságának becsléséhez egy PI megfigyelőt javasoltam. A javasolt módszer jó robusztusságot mutat a hőátadási tényező paraméterének bizonytalanságával szemben. Ezenkívül képes elkülöníteni és megbecsülni a lassan kialakuló kezdődő hibákat.

Zusammenfassung

Da Netzwerktechnologien wie Prozess-oder Transportnetzwerke in unserer heutigen Gesellschaft immer wichtiger werden, steigt auch der Bedarf an deren Verfügbarkeit und Zuverlässigkeit. Fehler in einem oder mehreren Subsystemen kann sich über das Netzwerk ausbreiten und die Leistung der gesamten vernetzten Systeme beeinträchtigen. Entsprechend muss im diesen Systemen eine adäquate Fehlererkennung und -isolierung (FDI) umgesetzt werden. Die beiden Hauptklassen vernetzter Systeme, die in dieser Arbeit untersucht werden, sind Roboterzüge und Wärmeaustauschnetzwerke.

Da die Zuverlässigkeit datengesteuerter Fehlerdiagnosemethoden von der Verfügbarkeit und Qualität der Daten abhängt, sind modellbasierte Ansätze insbesondere für sicherheitskritische Systeme von besonderer Relevanz. Das Ziel dieser Dissertation ist es, die bestehende FDI-Methoden für vernetzte lineare zeitinvariante Systeme zu erweitern.

Zunächst werdendazu die modellbasierten Fehlerdiagnosemethoden für Sensoren und Aktoren für Roboterzüge untersucht, die sich in einem Leader-Follower-Schema bewegen. Für die Fehlerdiagnose an den Sensoren wird ein mathematisches Modell der Roboterzüge erstellt, welches die Auswirkungen der Fehlerausbreitung erfasst. Basierend auf diesem Modell wird ein Satz an Beobachtern konzipiert, um die Sensorfehler zu erkennen und zu isolieren. Außerdem wird ein Ansatz zur Bestimmung der Schwellwerte zur Vermeidung von Fehlalarmen präsentiert. Für die Fehlerfdiagnose der Aktuatorik wird eine Filtertechnik vorgeschlagen, welche die Notwendigkeit einer Netzwerkkommunikation zur Isolierung der Fehler überflüssig macht. Anschließend wird ein Proportional-Integral (PI) Beobachter entwickelt, um die Größe des Aktuatorfehlers abzuschätzen.

Im weiteren Verlauf der Arbeit werden modellbasierte Methoden zur Erkennung, -isolierung und -schätzung von parametrischen Fehlern in Wärmeaustauschnetzwerkenok (HEN) vorgestellt. Der betrachtete Fehler bewirkt eine signifikante Änderung des Wärmeübergangskoeffizienten, der den Einfluss der Umgebungstemperatur auf die Temperatur im Netzwerk bestimmt. Die Netzwerktopologie an sich enthält sich teilende als auch sich vereinende Verbindungen. Das Problem der Sensorplatzierung für beide Verbindungstypen wird mithilfe der vorzeichenbehafteten gerichteten Diagrammanalyse gelöst. Wenn der Fehler jedoch nur an einer bestimmten Stelle entlang der Länge eines Rohrs im HEN auftritt, zeigt sich, dass eine Fehlerlokalisierung kaum möglich ist. Daher werden ein Satz an linear Beobachterzur Fehlerisolierung sowie ein nichtlinearer Beobachter basierend auf Parameteridentifikationstechniken vorgeschlagen.

Zum Abschluss wird eine modellbasierte Methode für Eingangfehler in HENs vorgestellt, die neben der sich teilenden und vereinenden Verbindungen auch uckführkreise betrachtet. Mithilfe der stationären Wertanalyse wird die Auswirkung der Fehlerausbreitung in den Netzwerkkreisen analysiert. Basierend auf den Analyseergebnissen wird ein Fehlerisolationsalgorithmus konstruiert, der nur zwei Sensoren erfordert. Um das Ausmaß des Fehlers abzuschätzen, wird dann ein PI-Beobachter vorgeschlagen. Die vorgeschlagene Methode zeigt eine erhöhte Robustheit gegenüber Parameterveränderungn des Wärmeübergangskoeffizienten. Darüber hinaus ist es zudem möglich, sich langsam entwickelnde Fehler zu isolieren und abzuschätzen.

Contents

List of Figures

List of Tables

List of Abbreviations

ACC	**A**daptive **C**ruise **C**ontrol
AI	**A**rtificial **I**ntelligence
DOF	**D**egree **O**f **F**reedom
FDI	**F**ault **D**etection and **I**solation
FDIA	**F**alse **D**ata **I**njection **A**ttack
FSF	**F**ull **S**tate **F**eedback
FVT	**F**inal **V**alue **T**heorem
GPS	**G**lobal **P**ositioning **S**ystem
HEN	**H**eat **E**xchanger **N**etwork
LPF	**L**ow **P**ass **F**ilter
LPS	**L**umped **P**arameter **S**ystem
LQE	**L**inear **Q**uadratic **E**stimator
LQR	**L**inear **Q**uadratic **R**egulator
LTI	**L**inear **T**ime **I**nvariant
MIMO	**M**ulti **I**nput **M**ulti **O**utput
ML	**M**achine **L**earning
ODE	**O**rdinary **D**ifferential **E**quation
PCA	**P**rincipal **C**omponent **A**nalysis
PI	**P**roportional **I**ntegral
PRBS	**P**seudo **R**andom **B**inary **S**ignal
SDG	**S**igned **D**irected **G**raph
UAV	**U**nmanned **A**erial **V**ehicle
UIO	**U**nknown **I**nput **O**bserver
WSN	**W**ireless **S**ensor **N**etwork

List of Symbols

$\mathbb{N}$	the set of all natural numbers
$\mathbb{N}^*$	the set of non-zero natural numbers, $\mathbb{N}^* = \mathbb{N} \setminus \{0\}$
$\mathbb{R}$	the set of real numbers
$\mathbb{R}^*$	the set of non-zero real numbers, $\mathbb{R}^* = \mathbb{R} \setminus \{0\}$
$\mathbb{R}^n$	the set of column vectors with $n \in \mathbb{N}^*$ real elements
$\mathbb{R}^{n \times m}$	the set of $n \times m$ matrices with real elements, where $m, n \in \mathbb{N}^*$
$\lvert x \rvert$	absolute value of a scalar x
$x \in \mathbb{R}^n$	a column vector in $\mathbb{R}^n$ with elements, $x = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix}^T$
$\lVert x \rVert_\infty$	infinity norm of $x \in \mathbb{R}^n$ defined as $\lVert x \rVert_\infty = \max_{1 \leq i \leq n} \lvert x_i \rvert$
$A \in \mathbb{R}^{m \times n}$	an $m \times n$ matrix with real elements
$A^T \in \mathbb{R}^{n \times m}$	transpose of A
A^{-1}	inverse of $A \in \mathbb{R}^{n \times n}$
$\operatorname{rank}(A)$	the rank of A
I	identity matrix
0	zero matrix
x	state of a given system
y	output of a given system
u	input of a given system
$\dot{x} = \frac{\mathrm{d}x}{\mathrm{d}t}$	continuous time derivative
$x^{(j)}$	state of the jth subsystem
$y^{(j)}$	output of the jth subsystem
$u^{(j)}$	input of the jth subsystem
x_i	the ith state of a given system
y_i	the ith output of a given system
u_i	the ith input of a given system

Chapter 1

Introduction

1.1 Background and motivations

The complexity and the automation degree of technological processes are continuously growing in this modern era. Today, their standards pay close attention to issues regarding high availability, reliability, safe operation, and cost efficiency. Thus, every effort to enhance the reliability and robustness of individual components such as sensors, actuators, and controllers (or computers) is critical. However, a fault free situation can hardly be guaranteed. Hence, Fault Detection and Isolation (FDI) techniques are critical to ensure a reliable operation. Although Linear Time Invariant (LTI) systems form the known well investigated class of dynamic systems, their FDI methods still need improvements and refinements mostly in complex networked cases. Such advancements in the FDI methods may utilize the specialties of the application area of primary importance.

By using the minimum number of sensors placed cleverly, this present study aims to enrich the existing FDI literature specifically for model based fault diagnosis methods on networked LTI systems. Despite being limited to the LTI systems, this special class provides a mathematical framework to analyse and understand a wide range of phenomena in nature. Moreover, many nonlinearities can be approximated locally with linear models by defining a suitable domain or equilibrium point [1, 2, 3].

Two main classes of networked LTI systems are addressed in this study: robot platoons as an important part of modern transportation technologies, and Heat Exchange Networks (HENs) as one of the important processes of industries. The selection of robot platoons and HEN is based on their popularity in representing networked LTI systems over time in many literatures either in system or control theory. Some small examples where the robot or vehicle platoons are modelled by LTI systems for the sake of control purposes can be seen in the works of [4, 5, 6, 7]. In the case of HEN, the works by [8, 9, 10, 11] demonstrate the modelling of HEN as LTI system.

1.1.1 Model based FDI in networked LTI systems

Model based and data driven methods in FDI FDI mainly deals with monitoring processes to ensure the operation safety of control systems. There exist many different fault diagnosis techniques [12]. Yet, the important main categories comprise model based [13] and data driven methods [14, 15]. Data driven methods rely on data from system operations that are used to train machine learning (ML) or artificial intelligence (AI) data structures to detect and identify faults. Nowadays, these

methods are gaining popularity because they are easier to implement due to the current development of digital computing technology. However, their reliability strictly depends on the availability and quality of the data [16].

Meanwhile, even though model based methods of FDI emerged in the early 70s, their performance has been proven in many successful implementations in the industrial processes [17]. There are many different techniques and each of them is usually developed for different purposes. Mostly, these methods are based on dynamic system models. Model based methods often use models derived from first principles that rely heavily on the knowledge of basic physical and/or chemical principles. Model based fault diagnosis methods are also intimate with the modern control theory [18].

The importance of fault diagnosis has given rise to a lot of research that has been carried in the last decades. A book by [19] discusses fault detection and isolation from a computational perspective. There is also a book by [20] explaining how to do fault diagnosis based on system identification techniques. The current emerging trend in Artificial Intelligence (AI) has also been exploited in [21, 22, 23] which represents data driven based FDI. In the case of a model based approach, a simple linear model is proposed to detect and isolate faults in electrical grids by [24]. In a recent article, [25] has also made a summary of many methods that have been researched to solve fault detection and identification problems.

The importance and significance of FDI in networked systems On the other hand, nowadays, the modern society relies on networks, e.g., computer networks, transportation networks, social networks, electrical networks, etc. Although they could comprise simple elements, their large number and interconnections make them an important subclass of complex systems. As a dynamic system, a network presents many theoretical challenges during control or diagnosis method design. There have been many excellent surveys about this in systems and control literature [26, 27, 28, 29, 30, 31, 32].

Even though there is already abundant literature for fault diagnosis for networked systems, the number of works related to FDI in networked systems is advancing slowly compared to the rapid growth of the literature related to networked systems' complexity and communication techniques (protocols) development [33]. There are still many various challenges because some approaches are effective only in handling some specific applications or faults. Nevertheless, the importance of network technologies in large scale control systems, as a powerful tool for data transfer, enforces the continuity of research in studying and developing fault diagnosis techniques for networked systems to ensure its reliability, availability, and security [34].

Generally, each subsystem in a networked system is designed with a specific function to accomplish a common goal in the network. A fault in a subsystem can propagate to others and will destroy or degrade the performance of the overall system [35]. Thus, FDI is highly important to consider in the networked systems [36].

The importance of FDI in networked systems is proven by many previous research studies. [37] introduced the method to detect and isolate sensor faults in a Wireless Sensor Network (WSN) installed on the wind turbine. In terms of smart grid for power systems, [38] proposed a model based sensor fault detection and isolation method. Meanwhile, still in the same smart grid, some studies to detect cyber physical attacks are done by [39, 40, 41]. [42, 43] did some research to handle faults occurring in the formation control of multi agent systems. Based on the H_∞ method, a filter is proposed to detect stochastic additive faults by [44]. [45] investigated how to estimate additive faults as unknown input by using a group of distributed filters.

[46] proposed the detection and isolation for cyber attack targeting sensors by using sliding mode observer. A distributed soft fault detection for WSN by using the approach of Fuzzy systems is done by [47]. An observer based fault diagnosis for networks has also been done by [48, 49, 50, 51, 52].

The sensor placement problem for FDI in networked systems It is preferred to install as few sensors as possible in the networked systems for either physical or financial reasons [53, 54, 55]. On the other hand, there could be complex topologies in realistic networks (e.g. branching connections and loops) [56, 57]. Thus, by considering the network topology, the sensor placement problem needs to be examined so that the existence of faults in the subsystems inside the network can be detected and isolated [58].

1.1.2 FDI in robot platoons

The importance of FDI in either robot or vehicle platoons can be seen from the large amount of existing literatures. [59] used parity equations to detect faulty sensors and actuators. [60] designed a fault detection technique and a sliding mode controller to compensate for the detected faults. A fault classification based on Machine Learning (ML) is also proposed by [61]. After devising an algorithm to detect and isolate the faults, [62] suggested using the model based approaches for vehicle platoons.

Most of the literature can be grouped into 2 categories. The first category is those who prioritize control objectives over fault identification (i.e., fault mitigation or the fault is treated as disturbances) such as in [63, 64, 60, 65]. Meanwhile, the second category is those who concentrate on fault isolation and estimation. This thesis tries to contribute to the second category in a networked environment. In this case, compared to the work of [62] which uses many sensors, and [59] which uses parity equations and nonlinear observers, model based fault diagnosis approaches that use conventional sensors and Unknown Input Observers (UIOs) are explored. UIO belongs to the group of linear observers. [66] and [67] have investigated the use of UIO to detect a faulty agent in the multi agent systems but have not gone into detail about the sources of the fault. [68] designed a centralized UIO for fault estimation purposes in a leader-follower linear multi agent systems. On the other hand, related to those previous works, this thesis explores how to design a bank of local UIOs in every subsystem to detect and isolate faulty sensors. Thus, it is more scalable compared to the global UIO. By considering simultaneous actuator fault and communication fault in the platoon's subsystems, actuator fault diagnosis is investigated that can be decoupled from communication faults.

1.1.3 FDI in heat exchanger network

Heat exchanger networks are a simple but important class of process systems. For FDI in process systems, much literature can be found. A number of possible neural network architectures for FDI in process systems are studied by [69]. Meanwhile, [70] proposed a hybrid technique by using the Principal Component Analysis (PCA) and a Bayesian network. A review about data driven methods for fault diagnosis in process systems has also been done by [71]. In addition, [72] did a literature review about data driven techniques that have been developed for chemical process systems. Recently, the interconnections among fault diagnosis, risk assessment, and abnormal situation management are analysed by [73].

As important as it is, many have researched parameter fault diagnosis in a heat exchange process, such as [74, 75, 76]. Moreover, the currently popular data driven methods have been exploited in much literature in handling parameter fault diagnosis in HEN [77, 78, 79]. However, in terms of network topology, splitting and joining connections are common to be found in the HEN. Combined with the sparsity of sensors and considering that it is not realistic to install sensors at every point or preferred place, it is important to investigate the sensor placement problem for fault diagnosis purposes in HEN. Yet, much literature neither includes the existence of these branching connections nor the sensor placement problem. Thus, this thesis tries to contribute by dealing first with investigating the sensor placement problem in correlation with branching connections' existence in the HEN. Then, by taking into account the obtained results, the parameter fault diagnosis using model based approach is explored.

Aside from joining and splitting connections, the existence of loops in networked LTI systems is also common as part of the network topology. In the case of HEN, the loop exists for energy or material efficiency and is a vital component in the control system. In a loop, a fault in a subsystem can propagate back to the subsystem where the fault first occurred. Here, fault estimation becomes more challenging as the fault's amplitude may become smaller [80]. Much of the literature discusses such problems using data driven approaches such as [81] and [82]. Therefore, this thesis tries to contribute to the same problem by proposing a model based approach as an alternative. Moreover, the approach considers that the proposed method uses the number of sensors as minimum as possible.

1.2 The aims of the research

The aim of this research is to develop model based fault diagnosis methods for networked LTI systems, specifically robot platoons and heat exchange networks. The following goals are set:

1. To develop networked LTI system models that can meet the objectives of fault detection, isolation, and estimation purposes.

   As robot platoons and heat exchange networks are two important classes of networked systems, many previous works discuss their dynamical modelling. However, based on them, this study aims to construct suitable models that can be used for fault diagnosis. To do this, after the fault has been modelled (i.e. sensor fault, actuator fault, or parameter fault), a dynamical system model of the networked LTI systems will be constructed that includes the fault's model. This problem is challenging because, in addition to the model of the fault being included, the network's model should also be able to capture the fault's effect propagation from one subsystem to the others.

2. To solve the sensor placement problem that is necessary for fault diagnosis in the networked LTI systems.

   The sensor placement problem is the problem of selecting measurement variables (i.e. where to install sensors) to maximize some objectives while satisfying various design constraints. Here, the objective is that every possible fault should be able to be detected and isolated. Meanwhile, the constraints are related to physical and financial reasons, i.e., the number of sensors used is as minimum as possible and they are installed at realistic places, for example at the end of connections. This problem is challenging because, as the fault can

propagate to the other subsystems, the source of the fault should still be able to be localized. Furthermore, as parameter fault is considered, suitable approaches that are independent of parameters should be explored. It should be noted that, by allowing sensors to be put only at some specific points, unmeasurable states will emerge.

3. To investigate the effect of network topology on fault diagnosis in networked LTI systems using the model based approach.

   This study also investigates the relationship between network topology and fault diagnosis by using the model based approaches. To make it similar to the real world environment, the features that are considered in the network topology are splitting connections, joining connections, and loops. This problem is challenging because, combined with the previously mentioned sensor placement problem, the presence of unmeasurable states could make the fault effect propagation difficult to track.

1.3 The structure of this work

The basic notions that are used in these studies are briefly reviewed in Chapter 2. Then, a model based actuator and sensor fault diagnosis for robot platoons are described in Chapter 3. After that, a model based approach for parameter fault diagnosis in heat exchange networks that contains branching connections is discussed in Chapter 4. Chapter 5 presents a model based approach for additive fault diagnosis in the heat exchange networks that contain loops. Finally, the theses points and future works are summarized in Chapter 6.

Meanwhile, the appendices contain some basic knowledge and background information that are useful to provide understanding in this study. The basic theory about the LTI system is written in Appendix A. On the other hand, Appendix B gives the necessary background information about the networked system used in this thesis. After that, some basic knowledge about observers can be found in Appendix C. Appendix D presents a summary of a special observer for a class of nonlinear systems. The last Appendix E contains a brief discussion about the future works of this study.

Chapter 2

Basic notions

2.1 Fault modelling in LTI systems

A fault (f) is a phenomenon that changes the behaviour of the technological processes such that it no longer satisfies its original purpose. Generally, it induces considerable deviation from the normal behaviour of a system that is caused by some unexpected events [12] and it can be categorized into [18]:

1. Actuator faults (f_a): these faults cause changes in the actuator's behaviour.
2. Sensor faults (f_s): these faults directly affect the system's measurement.

   In the case of LTI systems (see the details of LTI systems in Appendix A), the actuator and sensor faults can be modelled by incorporating them into the general model in Eq (A.3) as:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B(\boldsymbol{u} + \boldsymbol{f_a}) \\ \boldsymbol{y} &= C\boldsymbol{x} + D\boldsymbol{u} + \boldsymbol{f_s} \end{aligned} \tag{2.1}$$

3. Parameter/process faults: these faults directly affect the system's dynamic, for example a drastic change in system's parameter (parameter faults).

   The parameter faults are modelled by changing the related matrix in the LTI model because they change the system's dynamic:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A_f\boldsymbol{x} + B_f\boldsymbol{u} \\ \boldsymbol{y} &= C\boldsymbol{x} + D\boldsymbol{u} \end{aligned} \tag{2.2}$$

   where A_f and B_f are matrices containing the changed parameters because of the faults.

Throughout this research, the fault is assumed to be an unknown vector that has zero value in fault free condition and it is a deterministic time function.

2.2 Fault detection, isolation, and estimation

Here, some definitions and models related to the fault diagnosis process are presented [12, 18, 19]:

1. Fault detection: a binary decision about either there are some presences of any fault or the absence of all faults.
2. Fault isolation: the process to determine the occurrence location of the faults.

3. Weak fault isolation: fault isolation process where the faults are assumed to happen one at a time (no simultaneous faults).

4. Fault estimation: the process of determining the magnitude of the faults and, if it is possible, the type of the fault based on the available measurements.

2.2.1 Fault detection

Consider a residual generator for system in Eq (2.1) that has the following model:

$$\begin{aligned} \dot{\boldsymbol{z}} &= A_z \boldsymbol{z} + B_{zu} \boldsymbol{u} + B_{zy} \boldsymbol{y} \ , \ \boldsymbol{z}(\boldsymbol{t_0}) = \boldsymbol{z_0} \\ \boldsymbol{r} &= C_z \boldsymbol{z} + D_{ru} \boldsymbol{u} + D_{ry} \boldsymbol{y} \end{aligned} \tag{2.3}$$

where $\boldsymbol{z}$ is the residual generator's states and $\boldsymbol{r}$ is the residual signal (or residues). A_z, B_{zu}, B_{zy}, C_z, D_{ru}, and D_{ry} are matrices with appropriate dimensions.

By taking the previous fault diagnosis definitions, the fault detection problem can be redefined as follows [18, 19]: given a system as in Eq (2.1), design a residual generator as in Eq (2.3) such that:

1. In the absence of fault (both $\boldsymbol{f_a} = \boldsymbol{0}$ and $\boldsymbol{f_s} = \boldsymbol{0}$), the residues should be asymptotically decays to zero ($\lim_{t\to\infty} \boldsymbol{r} = \boldsymbol{0}$).

2. The residues $\boldsymbol{r}$ should be affected by the fault (i.e. if $\boldsymbol{f_a} \neq \boldsymbol{0} \vee \boldsymbol{f_s} \neq \boldsymbol{0}$, then $\lim_{t\to\infty} \boldsymbol{r} \neq \boldsymbol{0}$).

2.2.2 Fault isolation

In case of fault isolation, a bank of residual generators is designed in such a way that each residues is only affected by a specific fault [12, 18, 19]. Thus, by observing the produced residues, the fault can be isolated.

For example, consider that a system may be subjected to three different faults f_1, f_2, and f_3. Then, design a bank of three residual generators that produces three residues r_1, r_2, and r_3 so that the effects of fault on the residues satisfies:

TABLE 2.1: Effects of fault on residues: example 1

signals	f_1	f_2	f_3
r_1	X	0	0
r_2	0	X	0
r_3	0	0	X

where X indicates a non zero value.

Table 2.1 shows that the residual generators solve the fault isolation problem.

However, in real world applications, only weak fault isolation is commonly realized because strong fault isolation (for simultaneous faults) usually demands a large number of measurements. In addition, some sufficient conditions may be not fulfilled to obtain a diagonal structure as in Table 2.1, or the system structure does now allow it (i.e., due to the lack of observability). In this situation, some faults should be grouped so that some residues are affected by a specific group of faults.

TABLE 2.2: Effects of fault on residues: example 2

signals	f_1	f_2	f_3
r_1	X	0	X
r_2	0	X	0

Table 2.2 shows an example where both the effect of faults f_1 and f_3 are grouped to be detected by the residual signal r_1 and residual signal r_2 is only affected by fault f_2.

This combination of fault detection and isolation is considered as the basis of every fault diagnosis process.

2.2.3 Fault estimation

Fault estimation represents the next enhancement and is mostly done after the fault has been isolated. It is a challenging problem and some methods usually can only be applied to some specific type of faults [12, 18, 19]. Generally, the steady state value of the fault signal can be estimated and the estimation problem in case of a constant fault is formulated as follows: obtain the fault estimate $\hat{\boldsymbol{f}}$ such that $\lim_{t\to\infty} \|\boldsymbol{f} - \hat{\boldsymbol{f}}\| = \boldsymbol{0}$.

By assuming that the fault can be modelled as a constant signal, a common fault estimation method is to consider the fault as an additional state so that an observer can be designed to estimate it. The system model that is augmented by the fault is called an extended state space model and it can be written in the following general form:

$$\begin{aligned} \begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{f}} \end{bmatrix} &= \begin{bmatrix} A & F_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{f} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \boldsymbol{u} \\ \boldsymbol{y} &= \begin{bmatrix} C & F_y \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{f} \end{bmatrix} \end{aligned} \tag{2.4}$$

where $\boldsymbol{f}$ is the considered fault, 0 is the zero matrix, F_x is the fault distribution matrix with respect to system's states, and F_y is the fault distribution matrix with respect to system's measurements.

Then, some observer design techniques in Appendix C can be applied to Eq (2.4) for fault estimation purposes.

Chapter 3

Actuator and sensor fault diagnosis in robot platoons

The current technological advancement in communication and networks has pushed the popularity of technologies such as networked control systems, the internet of everything, and distributed control. Combined with the common cooperation phenomenon in nature, such as flocking birds, swarming insects, and human collaboration, they inspired the emergence of cooperative robotics theory [83, 84, 85, 86]. Much research focuses on such topics because they can solve many tasks that can not be done by only one robot in many wide and diverse areas such as space exploration, domestic help, healthcare, military operations, and Wireless Sensor Networks (WSNs) [87, 88]. As there are multiple robots that cooperate to accomplish predefined tasks, many critical issues need to be handled in such systems in the area of communication and control [89, 90]. One attractive example in cooperative robotics that becomes a concern for many researchers is the robot or vehicle platoons where some robots or vehicle subsystems move together in a leader-follower scheme [91, 92].

To be able to achieve a common goal for the platoons, each subsystem should have information about the states of its neighbouring subsystem in addition to its own states [93]. For that reason, it is common to have network communication between them either using a vehicle to vehicle or vehicle to cloud communication [94]. However, this makes the platoons more prone to network disturbances that can be generally categorized into communication faults and False Data Injection Attacks (FDIA) [59, 95]. As the attacker is assumed to have sufficient knowledge about the system's behaviour, FDIA is harder to detect and diagnose compared to communication faults [96, 97, 98, 99]. Thus, among those two, FDIA is a currently rising topic with fewer research results.

Apart from the network disturbances, the system's performance is also affected by component faults such as actuator faults and sensor faults [17, 100]. To handle them, most control strategies can be categorized into passive and active approaches [101, 102, 103, 104]. In the passive approach, the faults information is not required as it prioritizes the control objectives (e.g. the faults are treated like some unknown disturbances). Meanwhile, in the active approach, the faults information is obtained through fault detection and isolation processes in addition to the control strategies. Thus, in many cases where the achievement of the goals is more important than the fault diagnosis, such as a consensus based problem, the passive approach is preferred [105]. However, if the faults need to be compensated more accurately, an active approach should be considered [106, 107].

Actuator faults and network disturbances usually enter into the same input channels in the subsystems [108, 109, 110]. Some research shows that the effect of network disturbances, either communication faults or FDIA, can be ignored on the actuator

fault diagnosis processes by proper fault oriented modelling, filtering, and estimation techniques [64, 111]. Meanwhile, as sensor faults on a subsystem can propagate through the network, its isolation problems should also be acknowledged so that they will not affect other subsystems and disturb the coordination.

On the other hand, Unknown Input Observer (UIO) is a popular approach in a model based fault diagnosis because it can decouple the influence of unknown inputs on state estimation. In [17], it was shown that weak faults isolation can be achieved using this UIO. Residual generators are built from these UIOs for fault diagnosis purposes which are sensitive to some groups of faults but insensitive to others. Chakrabarty et al have further explored the implementation of this UIO to accommodate bounded exogenous inputs and delayed measurements [112, 113]. Xu et al combined UIO with set-theoretic methods for robust fault detection purposes [114]. Sensor faults detection on a Unmanned Aerial Vehicle (UAV) using UIO has also been researched by Zuo et al [115].

As the network contributes to many unpredicted unknown inputs such as network disturbances and faults propagation, this unknown inputs decoupling feature becomes more important in networked control systems. Thus, much research to implement UIO as states observer in this networked environment has been done. Taha et al investigated state estimation conducted by UIO which is connected to an observed plant via network [116]. UIO for interconnected second-order system investigation has also been done by Shames et al [117]. Chen et al and Chakrabarty et al explored the design of UIO for a class of interconnected nonlinear systems [118, 119].

In terms of fault diagnosis in a networked environment, Liu et al and Shames et al explored the utilization of UIO to detect a faulty agent in a multi-agent system [66, 67]. Zhang et al investigated how to do either actuator or sensor faults identification for each agent using a global UIO in networked control systems [68].

Based on those backgrounds and motivations, this research explores the actuator and sensor fault diagnosis problems in the robot platoons that move in a leader-follower scheme. With the hope that the results can be used to further design some fault tolerant controls for the said system, 2 different scenarios are investigated: 1. detection, isolation, and estimation purposes for actuator faults, and 2. detection and isolation purposes for sensor faults.

As mentioned earlier, for actuator fault diagnosis, the problem is that actuator faults and network disturbances enter the subsystem from the same input channel. Thus, a fault diagnosis method is studied that uses relative output information to eliminate the need for information interchange among the neighbouring subsystems so that the actuator faults isolation process is decoupled from network disturbances. To do this, the subsystems are modelled based on which the non-measurable states of the neighbouring subsystems can be determined by proper filtering techniques. Then, the model is used as the basis for designing a Proportional Integral (PI) observer for actuator fault estimation. The pole placement design method and Linear Quadratic Estimator (LQE) are applied to determine the observer's gain. The author's related publication for actuator fault diagnosis is [A1].

Meanwhile, for sensor faults, the design of a bank of local UIOs as residual generators is studied to detect and isolate faulty sensors in each subsystem of the robot platoons. It is done by extending the UIO based sensor faults isolation problem into networked control systems. This local fault isolation scheme has advantages in terms of scalability features compared to a global UIO. The author's related publication for sensor fault diagnosis is [A2].

The research questions that we are going to tackle are as follows:

1. How to develop the model of robot platoons that captures the actuator and sensor fault propagations from one subsystem to the others?
2. How to isolate a specific sensor fault in a subsystem using a bank of local observers?
3. How to decouple the actuator fault and the communication fault that come into the same input channel in a subsystem?
4. How to estimate the previously mentioned actuator fault?

3.1 The modelling of the controlled robot platoons

This section discusses the modelling of the controlled robot platoons that is used as the same basis for both actuator fault diagnosis and sensor fault diagnosis scenarios. It is said to be controlled because the model is derived from the Adaptive Cruise Control (ACC) systems which is used to determine the appropriate control input for the vehicle following system [91].

3.1.1 Fault free model of robot platoons

Consider that several robots move along an axis and each of them has a double integrator dynamic as follows:

$$m^{(j)}\ddot{x}^{(j)} = u^{(j)} \tag{3.1}$$

where $j = 2, 3, 4, ..., N$ represents the jth follower robot with N being the number of the robots. $j = 1$ is allocated to represent the leader robot. In addition, it holds that $\ddot{x}^{(j)} = \dot{v}^{(j)} = a^{(j)}$ where $x^{(j)}$ is the robot's position, $v^{(j)}$ is the robot's velocity, $a^{(j)}$ is the robot's acceleration, $m^{(j)}$ is the robot's mass, and $u^{(j)}$ is the control input of the jth robot.

According to [92], two conditions that must be achieved in this leader-following scheme are individual stability and string stability:

1. *Individual stability* means that the spacing error of the concerned robot should converge to zero when the preceding robot has a constant speed. The spacing error describing the distance with the preceding robot can be expressed as:

$$\begin{aligned} \epsilon^{(j)} &= x^{(j)} - x^{(j-1)} + l^{(j-1)} \\ e^{(j)} &= x^{(j)} - x^{(j-1)} + L^{(j)} \end{aligned} \tag{3.2}$$

where $\epsilon^{(j)}$ is the measured inter-robot spacing, $e^{(j)}$ is the spacing error of the jth robot, and $L^{(j)}$ is some desired value of inter-robot spacing that includes the length of the preceding robot $l^{(j-1)}$. All of these are shown in Fig 3.1.

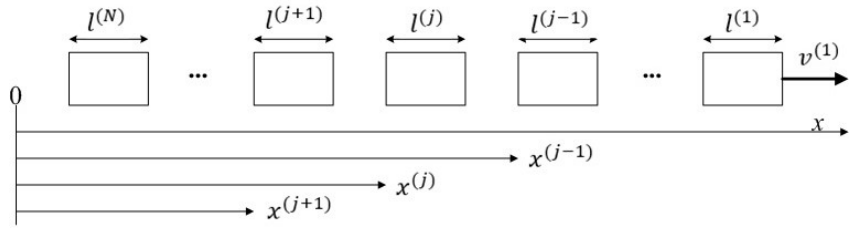

FIGURE 3.1: Robot platoon block diagram

2. *String stability* means that the spacing errors on each robot should not amplify towards the end of the string ($\|e^{(j)}\|_\infty \leq \|e^{(j-1)}\|_\infty$). To satisfy both individual stability and string stability, a constant time-gap policy is used to formulate the control input for each robot. In this policy, the desired inter-robot spacing $L^{(j)}$ is not made constant. Instead, it varies in proportional with velocity as well as the spacing error $e^{(j)}$ as follows:

$$\begin{aligned} L^{(j)} &= l^{(j-1)} + h\dot{x}^{(j)} \\ e^{(j)} &= \epsilon^{(j)} + h\dot{x}^{(j)} \end{aligned} \tag{3.3}$$

where h is a prescribed positive constant parameter called time-gap and $\epsilon^{(j)}$ is as given in Eq (3.2).

A previous research [92] found that a suitable control input $u_c^{(j)}$ for this constant time-gap policy is in the form of:

$$\begin{aligned} u_c^{(j)} = \ddot{x}^{(j)} &= -\frac{1}{h}(\dot{\epsilon}^{(j)} + \lambda e^{(j)}) \\ &= -\frac{1}{h}(\dot{x}^{(j)} - \dot{x}^{(j-1)} + \dot{l}^{(j-1)} + \lambda x^{(j)} - \lambda x^{(j-1)} + \lambda l^{(j-1)} + \lambda h\dot{x}^{(j)}) \\ &= -\frac{\lambda}{h}x^{(j)} + (-\frac{1}{h} - \lambda)\dot{x}^{(j)} + \frac{\lambda}{h}x^{(j-1)} + \frac{1}{h}\dot{x}^{(j-1)} + (-\frac{\lambda}{h})l^{(j-1)} \end{aligned} \tag{3.4}$$

where λ is a constant chosen such that $\lambda > 0$.

For the sake of convenience, Eq (3.4) is rewritten as:

$$u_c^{(j)} = k_1^{(j)}x^{(j)} + k_2^{(j)}v^{(j)} + k_3^{(j)}x^{(j-1)} + k_4^{(j)}v^{(j-1)} + k_5^{(j)}l^{(j-1)} \tag{3.5}$$

where $k_1^{(j)} = -\frac{\lambda}{h}$, $k_2^{(j)} = -\frac{1}{h} - \lambda$, $k_3^{(j)} = \frac{\lambda}{h}$, $k_4^{(j)} = \frac{1}{h}$, and $k_5^{(j)} = -\frac{\lambda}{h}$.

3.1.2 Measurements in robot platoons

It is assumed that each robot is equipped with conventional sensors which are a Global Positioning System (GPS)-based sensor, a velocity sensor, and a radar sensor. Thus, the considered measurement outputs are:

1. $y_1^{(j)}$ is a position measurement by using a GPS-based sensor (S1).
2. $y_2^{(j)}$ is a velocity measurement obtained from the same GPS-based sensor (S1). For GPS-based velocity measurement, please refer to [120].
3. $y_3^{(j)}$ is a velocity measurement by using the wheel-mounted velocity sensor (S2) on the robot.
4. $y_4^{(j)}$ is an inter-vehicle distance measurement by using a radar-based sensor (S3).

In addition, it is also assumed that each robot receives the position information from the preceding robot ($x^{(j-1)}$) through wireless inter-vehicle network communication. The velocity from the preceding robot ($v^{(j-1)}$) can also be obtained by differentiating this information ($\dot{x}^{(j-1)}$).

Note that the common use of local sensors and inter-vehicle communication introduces measurement redundancy. Velocity can either be obtained from the GPS-based sensor or measured by the wheel-mounted velocity sensor. In the same manner, the inter-vehicle distance can also either be directly measured or be obtained by

the difference between the GPS-based measurement and the position of the preceding vehicle which is received through the wireless communication network.

By taking account of those measurements along with Eqs (3.1) and (3.5), the state space model for the jth follower robot ($j = 2, 3, 4, ..., N$) can be written as follows:

$$\begin{aligned}
\begin{bmatrix} \dot{x}_1^{(j)} \\ \dot{x}_2^{(j)} \\ \dot{x}_3^{(j)} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \frac{k_5^{(j)}}{m^{(j)}} & \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}}{m^{(j)}} \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} l^{(j-1)} \\ x^{(j-1)} \\ v^{(j-1)} \end{bmatrix} \\
\begin{bmatrix} y_1^{(j)} \\ y_2^{(j)} \\ y_3^{(j)} \\ y_4^{(j)} \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix}
\end{aligned} \tag{3.6}$$

where $x_1^{(j)} = x^{(j)}$, $x_2^{(j)} = v^{(j)} = \dot{x}^{(j)}$, and $x_3^{(j)} = x^{(j)} - x^{(j-1)}$.

Special case: the leader robot

It is considered that the leader robot ($j = 1$) moves with a prescribed controlled speed and it does not receive information from the other robots. The needed control input is calculated only from the sensor measurements without being affected by the states of the other robots. A Proportional Integral (PI) velocity controller is assumed for this leader as follows:

$$\begin{aligned}
u_c^{(1)} &= k_I \int_0^t (v_{ref} - v^{(1)})\, d\tau + k_P (v_{ref} - v^{(1)}) \\
u_c^{(1)} &= -k_I x^{(1)} - k_P v^{(1)} + u_{ff} \\
u_{ff} &= k_I \int_0^t v_{ref}\, d\tau + k_P v_{ref}
\end{aligned} \tag{3.7}$$

where k_I is the integral control gain, k_P is the proportional control gain, u_{ff} is the feed-forward control term, and v_{ref} is a predefined speed setpoint for the leader robot.

The assumed measurement outputs for the leader robot are:

1. $y_1^{(1)}$ is a GPS-based position measurement.
2. $y_2^{(1)}$ is a velocity measurement based on the same GPS sensor.
3. $y_3^{(1)}$ is a velocity measurement by using the wheel-mounted velocity sensor on the robot.

Thus, the state space model for the leader robot is:

$$\begin{aligned}
\begin{bmatrix} \dot{x}_1^{(1)} \\ \dot{x}_2^{(1)} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{k_I}{m^{(1)}} & -\frac{k_P}{m^{(1)}} \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m^{(1)}} \end{bmatrix} u_{ff} \\
\begin{bmatrix} y_1^{(1)} \\ y_2^{(1)} \\ y_3^{(1)} \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix}
\end{aligned} \tag{3.8}$$

where $x_1^{(1)} = x^{(1)}$ and $x_2^{(1)} = v^{(1)} = \dot{x}^{(1)}$.

3.2 Sensor fault diagnosis in robot platoons

This section is provided to discuss the sensor fault diagnosis scenario in the robot platoons.

3.2.1 Model of robot platoons with sensor fault

Sensors can be affected by additive faults and measurement noises. The measurement noises are considered to be high frequency and low amplitude signals. Meanwhile, the faults are assumed to be low frequency and high magnitude signals (e.g. sensor biases). Furthermore, they have a considerable influence on the dynamic behaviour of the robot platoons. Hence, the output measurements for the follower robot ($j = 2, 3, 4, ..., N$) in Eq (3.6) can now be modelled as:

$$\begin{aligned} \boldsymbol{y}^{(j)} &= \tilde{C}^{(j)}\boldsymbol{x}^{(j)} + \boldsymbol{f_s}^{(j)} + \boldsymbol{w}^{(j)} \\ \tilde{C}^{(j)} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{3.9}$$

where $\boldsymbol{y}^{(j)} = [y_1^{(j)}\ y_2^{(j)}\ y_3^{(j)}\ y_4^{(j)}]^T$ is the measurement, $\boldsymbol{x}^{(j)} = [x_1^{(j)}\ x_2^{(j)}\ x_3^{(j)}]^T$ is the subsystem's states, $\boldsymbol{f_s}^{(j)} = [f_{s1}^{(j)}\ \dot{f}_{s1}^{(j)}\ f_{s2}^{(j)}\ f_{s3}^{(j)}]^T$ is the sensor fault which contains position sensor fault ($f_{s1}^{(j)}$), derivative of the position sensor fault ($\dot{f}_{s1}^{(j)}$), velocity sensor fault ($f_{s2}^{(j)}$), distance sensor fault ($f_{s3}^{(j)}$), and $\boldsymbol{w}^{(j)} = [w_1^{(j)}\ \dot{w}_1^{(j)}\ w_2^{(j)}\ w_3^{(j)}]^T$ is the measurement noises.

The received information from the network is also considered to be affected by network disturbances as follows:

$$i^{(j-1)} = x^{(j-1)} + \delta i^{(j-1)} \tag{3.10}$$

where $i^{(j-1)}$ is the received position information from the preceding robot, $x^{(j-1)}$ is the position information from the preceding robot, and $\delta i^{(j-1)}$ represents the disturbance on the signal received through the network due to the possible position sensor fault of the preceding robot.

The control input for each robot in the robot platoons can be calculated from the enumerated sensor measurements which are affected both by sensor faults and network disturbance. As there are redundancies in the measurements, there are also several ways to compute the control signal $u_c^{(j)}$. Based on Eqs (3.5), (3.9), and (3.10), one possible way to compute it is:

$$\begin{aligned} u_c^{(j)} &= k_1^{(j)}y_1^{(j)} + k_2^{(j)}y_3^{(j)} + k_3^{(j)}i^{(j-1)} + k_4^{(j)}\dot{i}^{(j-1)} + k_5^{(j)}l^{(j-1)} \\ u_c^{(j)} &= k_1^{(j)}(x_1^{(j)} + f_{s1}^{(j)} + w_1^{(j)}) + k_2^{(j)}(x_2^{(j)} + f_{s2}^{(j)} + w_2^{(j)}) + k_3^{(j)}(x^{(j-1)} + \delta i^{(j-1)}) \\ &\quad + k_4^{(j)}(\dot{x}^{(j-1)} + \dot{\delta} i^{(j-1)}) + k_5^{(j)}l^{(j-1)} \\ u_c^{(j)} &= k_1^{(j)}x^{(j)} + k_2^{(j)}v^{(j)} + k_3^{(j)}x^{(j-1)} + k_4^{(j)}v^{(j-1)} + k_5^{(j)}l^{(j-1)} + \tilde{D}^{(j)} \end{aligned} \tag{3.11}$$

where $\tilde{D}^{(j)}$ represents an unknown disturbance term which sums up the additive faults and the noises terms that arrive through the networks or from the local faulty

sensors as follows:

$$\tilde{D}^{(j)} = k_1^{(j)} f_{s1}^{(j)} + k_2^{(j)} w_1^{(j)} + k_2^{(j)} f_{s2}^{(j)} + k_2^{(j)} w_2^{(j)} + k_3^{(j)} \delta i^{(j-1)} + k_4^{(j)} \dot{\delta} i^{(j-1)} \tag{3.12}$$

If we implement the control input using measurements from other possibilities, then $u_c^{(j)}$ will have the same form as in Eq (3.11) and only $\tilde{D}^{(j)}$ will change.

Thus, the state space equation with unknown disturbance term for the jth robot in the robot platoons is:

$$\begin{aligned}
\begin{bmatrix} \dot{x}_1^{(j)} \\ \dot{x}_2^{(j)} \\ \dot{x}_3^{(j)} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \frac{k_5^{(j)}}{m^{(j)}} & \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}}{m^{(j)}} \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} l^{(j-1)} \\ x^{(j-1)} \\ v^{(j-1)} \end{bmatrix} \\
&\quad + \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}}{m^{(j)}} & \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta i^{(j-1)} \\ \dot{\delta} i^{(j-1)} \\ f_{s1}^{(j)} + w_1^{(j)} \\ f_{s2}^{(j)} + w_2^{(j)} \end{bmatrix} \\
\begin{bmatrix} y_1^{(j)} \\ y_2^{(j)} \\ y_3^{(j)} \\ y_4^{(j)} \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix} + \begin{bmatrix} f_{s1}^{(j)} \\ \dot{f}_{s1}^{(j)} \\ f_{s2}^{(j)} \\ f_{s3}^{(j)} \end{bmatrix} + \begin{bmatrix} w_1^{(j)} \\ \dot{w}_1^{(j)} \\ w_2^{(j)} \\ w_3^{(j)} \end{bmatrix}
\end{aligned} \tag{3.13}$$

Furthermore, Eq (3.13) can be written as in Eq (C.9) in the following form:

$$\begin{aligned}
\dot{\boldsymbol{x}}^{(j)} &= \tilde{A}^{(j)} \boldsymbol{x}^{(j)} + \tilde{B}^{(j)} \boldsymbol{u}^{(j)} + \tilde{E}^{(j)} \boldsymbol{d}^{(j)} \\
\boldsymbol{y}^{(j)} &= \tilde{C}^{(j)} \boldsymbol{x}^{(j)} + \boldsymbol{f_s}^{(j)} + \boldsymbol{w}^{(j)}
\end{aligned}$$

where

$$\begin{aligned}
\tilde{A}^{(j)} &= \begin{bmatrix} 0 & 1 & 0 \\ \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \tilde{B}^{(j)} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{k_5^{(j)}}{m^{(j)}} & \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}}{m^{(j)}} \\ 0 & 0 & -1 \end{bmatrix}, \\
\tilde{E}^{(j)} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}}{m^{(j)}} & \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \tilde{C}^{(j)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\
\boldsymbol{x}^{(j)} &= \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix}, \quad \boldsymbol{u}^{(j)} = \begin{bmatrix} l^{(j-1)} \\ x^{(j-1)} \\ v^{(j-1)} \end{bmatrix}, \quad \boldsymbol{d}^{(j)} = \begin{bmatrix} \delta i^{(j-1)} \\ \dot{\delta} i^{(j-1)} \\ f_{s1}^{(j)} + w_1^{(j)} \\ f_{s2}^{(j)} + w_2^{(j)} \end{bmatrix}, \\
\boldsymbol{f_s}^{(j)} &= \begin{bmatrix} f_{s1}^{(j)} \\ \dot{f}_{s1}^{(j)} \\ f_{s2}^{(j)} \\ f_{s3}^{(j)} \end{bmatrix}, \quad \boldsymbol{w}^{(j)} = \begin{bmatrix} w_1^{(j)} \\ \dot{w}_1^{(j)} \\ w_2^{(j)} \\ w_3^{(j)} \end{bmatrix}
\end{aligned} \tag{3.14}$$

Special case: the leader robot

The control input of the leader robot ($j = 1$) only depends on it's own sensor measurements. The unknown inputs, which act as disturbance for the sensor faulty model of the leader robot, come from this leader's sensor fault. With this additive sensor fault, the leader's measurements become:

$$\begin{aligned} \boldsymbol{y^{(1)}} &= \tilde{C}^{(1)}\boldsymbol{x^{(1)}} + \boldsymbol{f_s^{(1)}} + \boldsymbol{w^{(1)}} \\ \tilde{C}^{(1)} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{3.15}$$

where $\boldsymbol{y^{(1)}} = [y_1^{(1)}\ y_2^{(1)}\ y_3^{(1)}]^T$ is the leader's measurement, $\boldsymbol{x^{(1)}} = [x_1^{(1)}\ x_2^{(1)}]^T$ is the leader's states, $\boldsymbol{f_s^{(1)}} = [f_{s1}^{(1)}\ \dot{f}_{s1}^{(1)}\ f_{s2}^{(1)}]^T$ is the leader's sensor fault which contains position sensor fault ($f_{s1}^{(1)}$), derivative of the position sensor fault ($\dot{f}_{s1}^{(1)}$), velocity sensor fault ($f_{s2}^{(1)}$), and $\boldsymbol{w^{(1)}} = [w_1^{(1)}\ \dot{w}_1^{(1)}\ w_2^{(1)}]^T$ is the leader's measurement noises.

Then, based on Eqs (3.7) and (3.15), one possible way to compute the control input of the leader $u_c^{(1)}$ is:

$$\begin{aligned} u_c^{(1)} &= -k_I y_1^{(1)} - k_P y_3^{(1)} + u_{ff} \\ u_c^{(1)} &= -k_I(x_1^{(1)} + f_{s1}^{(1)} + w_1^{(1)}) - k_P(x_2^{(1)} + f_{s2}^{(1)} + w_2^{(1)}) + u_{ff} \\ u_c^{(1)} &= -k_I x_1^{(1)} - k_P x_2^{(1)} + u_{ff} + \tilde{D}^{(1)} \end{aligned} \tag{3.16}$$

where $\tilde{D}^{(1)}$ contains the additive fault-induced disturbance and noise terms as follows:

$$\tilde{D}^{(1)} = -k_I f_{s1}^{(1)} - k_I w_1^{(1)} - k_P f_{s2}^{(1)} - k_P w_2^{(1)} \tag{3.17}$$

Thus, the state space equation with sensor faults for the leader robot in the robot platoons is:

$$\begin{aligned} \begin{bmatrix} \dot{x}_1^{(1)} \\ \dot{x}_2^{(1)} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{k_I}{m^{(1)}} & -\frac{k_P}{m^{(1)}} \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m^{(1)}} \end{bmatrix} u_{ff} + \begin{bmatrix} 0 & 0 \\ -\frac{k_I}{m^{(1)}} & -\frac{k_P}{m^{(1)}} \end{bmatrix} \begin{bmatrix} f_{s1}^{(1)} + w_1^{(1)} \\ f_{s2}^{(1)} + w_2^{(1)} \end{bmatrix} \\ \begin{bmatrix} y_1^{(1)} \\ y_2^{(1)} \\ y_3^{(1)} \end{bmatrix} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix} + \begin{bmatrix} f_{s1}^{(1)} \\ \dot{f}_{s1}^{(1)} \\ f_{s2}^{(1)} \end{bmatrix} + \begin{bmatrix} w_1^{(1)} \\ \dot{w}_1^{(1)} \\ w_2^{(1)} \end{bmatrix} \end{aligned} \tag{3.18}$$

Furthermore, Eq (3.18) can be written as in Eq (C.9) in the following form:

$$\begin{aligned}
\dot{\boldsymbol{x}}^{(\mathbf{1})} &= \tilde{A}^{(1)}\boldsymbol{x}^{(\mathbf{1})} + \tilde{B}^{(1)}\boldsymbol{u}^{(\mathbf{1})} + \tilde{E}^{(1)}\boldsymbol{d}^{(\mathbf{1})} \\
\boldsymbol{y}^{(\mathbf{1})} &= \tilde{C}^{(1)}\boldsymbol{x}^{(\mathbf{1})} + \boldsymbol{f}_{\boldsymbol{s}}^{(\mathbf{1})} + \boldsymbol{w}^{(\mathbf{1})}
\end{aligned}$$

where

$$\begin{aligned}
\tilde{A}^{(1)} &= \begin{bmatrix} 0 & 1 \\ -\frac{k_I^{(1)}}{m^{(1)}} & -\frac{k_p^{(1)}}{m^{(1)}} \end{bmatrix}, \quad \tilde{B}^{(1)} = \begin{bmatrix} 0 \\ \frac{1}{m^{(1)}} \end{bmatrix}, \\
\tilde{E}^{(1)} &= \begin{bmatrix} 0 & 0 \\ -\frac{k_I^{(1)}}{m^{(1)}} & -\frac{k_p^{(1)}}{m^{(1)}} \end{bmatrix}, \quad \tilde{C}^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}, \\
\boldsymbol{x}^{(\mathbf{1})} &= \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix}, \quad \boldsymbol{u}^{(\mathbf{1})} = u_{ff}, \quad \boldsymbol{d}^{(\mathbf{1})} = \begin{bmatrix} f_{s1}^{(1)} + w_1^{(1)} \\ f_{s2}^{(1)} + w_2^{(1)} \end{bmatrix}, \\
\boldsymbol{f}_{\boldsymbol{s}}^{(\mathbf{1})} &= \begin{bmatrix} f_{s1}^{(1)} \\ \dot{f}_{s1}^{(1)} \\ f_{s2}^{(1)} \end{bmatrix}, \quad \boldsymbol{w}^{(\mathbf{1})} = \begin{bmatrix} w_1^{(1)} \\ \dot{w}_1^{(1)} \\ w_2^{(1)} \end{bmatrix}
\end{aligned} \tag{3.19}$$

3.2.2 UIO-based sensor fault isolation in robot platoons

Sensor fault isolation refers to a process for determining in which sensor the fault has occurred. One way to accomplish this is by using structured residual signals. These structured residual signals are made in such a way that each of them is sensitive (or insensitive) to a certain group of faults while insensitive (or sensitive) to the others. This property becomes the basis to solve the isolation problem. As mentioned earlier, because UIO can decouple some unknown input signals so that it is insensitive to them, it has the potential to produce the needed structured residual signals for sensor faults isolation. One remark is that this scheme can only isolate a single or a certain group of faults one at a time (weak fault isolation).

Consider an LTI system with sensor faults $\boldsymbol{f_s}$ and unknown disturbance $\boldsymbol{d}$ as follows:

$$\begin{aligned}
\dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} + E\boldsymbol{d} \\
\boldsymbol{y} &= C\boldsymbol{x} + \boldsymbol{f_s}
\end{aligned} \tag{3.20}$$

where $C \in \mathbb{R}^{n\times m}$, $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{y} \in \mathbb{R}^m$, and $\boldsymbol{f_s} \in \mathbb{R}^m$.

To apply the UIO for weak sensor fault isolation, we delete one by one of the rows in the C matrix in Eq (3.20) related with each sensor fault:

$$\boldsymbol{y_i} = C_i\boldsymbol{x} + \boldsymbol{f_{si}} \tag{3.21}$$

where $C_i \in \mathbb{R}^{n\times(m-1)}$ is the C matrix without the ith row, $\boldsymbol{y_i} \in \mathbb{R}^{m-1}$ is the $\boldsymbol{y}$ vector without the ith entry, and $\boldsymbol{f_{si}} \in \mathbb{R}^{m-1}$ is the $\boldsymbol{f_s}$ vector without the ith entry.

Then, a bank of m UIOs for the system can be derived with the following dynamics:

$$\begin{aligned}
\dot{\boldsymbol{z}}_{\boldsymbol{i}} &= F_i\boldsymbol{z_i} + T_iB\boldsymbol{u} + K_i\boldsymbol{y_i} \\
\boldsymbol{r_i} &= (I - C_iH_i)\boldsymbol{y_i} - C_i\boldsymbol{z_i}
\end{aligned} \tag{3.22}$$

with $i = 1, 2, 3, ..., m$, $\boldsymbol{z_i} \in \mathbb{R}^n$ is the ith sensor fault UIO's state vector, $\boldsymbol{r_i} \in \mathbb{R}^{m-1}$ is the ith sensor fault UIO's residual signal, and the F_i, T_i, K_i, and H_i matrices are

chosen in such a way to satisfy the following conditions:

$$\begin{aligned} H_i C_i E &= E \\ T_i &= I - H_i C_i \\ F_i &= T_i A - K_{1i} C_i \\ K_{2i} &= F_i H_i \\ K_i &= K_{1i} + K_{2i} \end{aligned} \tag{3.23}$$

where K_{1i} is chosen to stabilize F_i.

Eq (3.22) shows that $\boldsymbol{z_i}$ is affected by all inputs except the ith entry related to a specific sensor fault ($\boldsymbol{y_i}$). If a fault occurs, all the residual signals will be triggered except for the ith one (i.e. $\boldsymbol{r_i}$ is *insensitive* to the ith sensor fault). Hence, this bank of UIOs can be used as residual generators for weak sensor fault isolation.

Case 1: the follower robot

In the robot platoons, the network disturbance comes from the position sensor fault measurement of the preceding robot which is transmitted to the robot behind it. For the jth robot, Eq (3.14) shows that we can treat both the network disturbance from the preceding $(j-1)$th robot and the effect of its own sensor faults on control as the unknown input part $\boldsymbol{d^{(j)}}$.

The output matrix $\tilde{C}^{(j)}$ is given by Eq (3.9). The $\tilde{C}_1^{(j)}$, $\tilde{C}_2^{(j)}$, and $\tilde{C}_3^{(j)}$ matrices, which correspond to the design of UIO for weak sensor fault isolation, are obtained in the same manner as in Eq (3.21) so that we get:

$$\begin{aligned} \tilde{C}_1^{(j)} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \tilde{C}_2^{(j)} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \tilde{C}_3^{(j)} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \end{aligned} \tag{3.24}$$

One remark for UIO which is designed to be insensitive to a fault in S1 is that both the first and second row of the $\tilde{C}^{(j)}$ matrix is deleted to form $\tilde{C}_1^{(j)}$ because $y_2^{(j)}$ represents velocity which is obtained from the same sensor to obtain $y_1^{(j)}$. Thus, GPS-based velocity measurement is not treated as an extra sensor that needs faults isolation.

By using those new matrices in Eq (3.24), three UIOs for each sensor faults isolation can be designed for each robot according to Eqs (3.22) and (3.23) to produce residual signals $r_1^{(j)}$, $r_2^{(j)}$, and $r_3^{(j}$ which are insensitive to fault in S1, S2, and S3, respectively. These residual signals produced by each of those UIOs can be used to perform weak sensor fault isolation as shown in Fig 3.2.

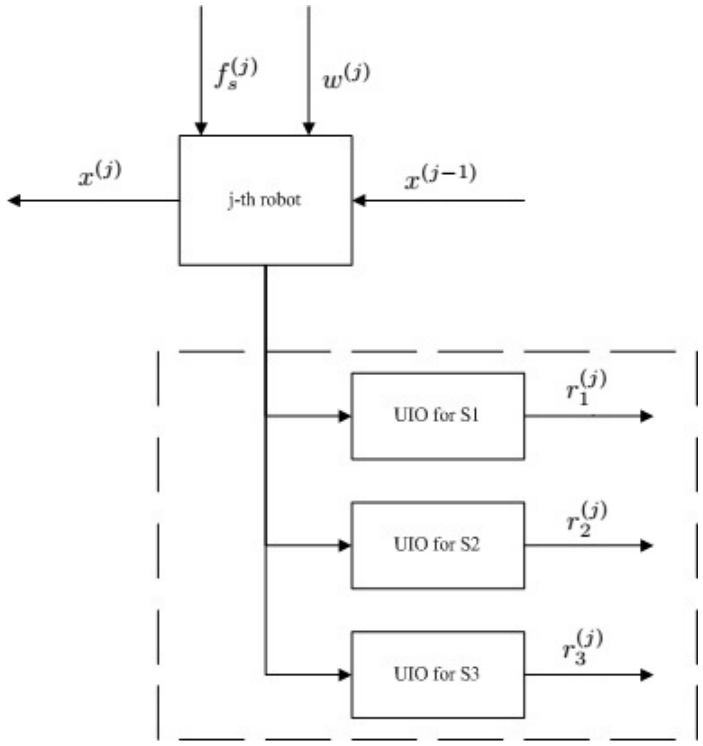

FIGURE 3.2: A bank of local UIOs for sensor faults isolation on each robot

Case 2: The leader robot

Sensor faults isolation for the leader robot has a rather different scheme because its states are not affected by the states of the other robots. One more thing to be noted is that it only has redundancy in velocity measurement which comes from both the GPS-based sensor measurement (S1) and the wheel-mounted velocity sensor measurement (S2).

As the leader robot has redundancy in the velocity measurement, we can use the same method as in the case of the follower robot to design the UIO which is insensitive to a fault in S2. Thus, based on Eq (3.19), we use the following matrices to construct the UIO that produces $r_2^{(1)}$:

$$\begin{aligned}
\tilde{A}_2^{(1)} &= \begin{bmatrix} 0 & 1 \\ -\frac{k_I}{m^{(1)}} & -\frac{k_P}{m^{(1)}} \end{bmatrix} \\
\tilde{B}_2^{(1)} &= \begin{bmatrix} 0 \\ \frac{1}{m^{(1)}} \end{bmatrix} \\
\tilde{C}_2^{(1)} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
\tilde{E}_2^{(1)} &= \begin{bmatrix} 0 & 0 \\ -\frac{k_I}{m^{(1)}} & -\frac{k_P}{m^{(1)}} \end{bmatrix}
\end{aligned} \tag{3.25}$$

As the position measurement of the leader does not have any redundancy, we refer to Eq (3.16) to design a residual generator for fault isolation in S1 by neglecting the disturbance parts $E^{(1)}$ and $D^{(1)}$ so that the produced residual signal will respond

to any sensor bias fault in S2. We can do that because we perform weak sensor fault isolation that assumes no simultaneous fault. Thus, we get:

$$\dot{x}_2^{(1)} = -\frac{k_P}{m^{(1)}} x_2^{(1)} - \frac{k_I}{m^{(1)}} x_1^{(1)} + \frac{1}{m^{(1)}} u_{ff} \tag{3.26}$$

After that, based on Eq (3.26), the matrices to construct the UIO that produces $r_1^{(1)}$ are:

$$\begin{aligned} \tilde{A}_1^{(1)} &= -\frac{k_P}{m^{(1)}} \\ \tilde{B}_1^{(1)} &= \begin{bmatrix} -\frac{k_I}{m^{(1)}} & \frac{1}{m^{(1)}} \end{bmatrix} \\ \tilde{C}_1^{(1)} &= 1 \\ \tilde{E}_1^{(1)} &= 0 \end{aligned} \tag{3.27}$$

3.2.3 Threshold computation for sensor fault isolation in robot platoons

As previously explained, to perform weak sensor fault isolation, the UIOs are treated as residual generators in which each residual is insensitive to a specific fault while sensitive to the others. Consequently, a suitable threshold need to be specified because we determine the faulty sensor by observing noisy measurement signals. By substituting Eq (3.21) augmented with the measurement noise vector $\boldsymbol{w} \in \mathbb{R}^m$ into Eq (3.22), we get:

$$\begin{aligned} \dot{\boldsymbol{z}}_{\boldsymbol{i}} &= F_i \boldsymbol{z}_{\boldsymbol{i}} + T_i B \boldsymbol{u} + K_i (C_i \boldsymbol{x} + \boldsymbol{f}_{\boldsymbol{si}} + \boldsymbol{w}_{\boldsymbol{i}}) \\ \boldsymbol{r}_{\boldsymbol{i}} &= (I - C_i H_i)(C_i \boldsymbol{x} + \boldsymbol{f}_{\boldsymbol{si}} + \boldsymbol{w}_{\boldsymbol{i}}) - C_i \boldsymbol{z}_{\boldsymbol{i}} \end{aligned} \tag{3.28}$$

where $\boldsymbol{w}_{\boldsymbol{i}} \in \mathbb{R}^{m-1}$ is the $\boldsymbol{w}$ vector without the ith entry.

Thus, the sole effect of the measurement noises on the residuals ($\boldsymbol{r}_{\boldsymbol{i(noise)}}$) is defined by the following equation:

$$\begin{aligned} \dot{\boldsymbol{z}}_{\boldsymbol{i}} &= F_i \boldsymbol{z}_{\boldsymbol{i}} + K_i \boldsymbol{w}_{\boldsymbol{i}} \\ \boldsymbol{r}_{\boldsymbol{i(noise)}} &= (I - C_i H_i) \boldsymbol{w}_{\boldsymbol{i}} - C_i \boldsymbol{z}_{\boldsymbol{i}} \end{aligned} \tag{3.29}$$

Now, let the corresponding transfer function matrix of the state space model in Eq (3.29) be G_{rwi}. By assuming finite energy input signal, it is known that $\|r_i\| \leq \|G_{rwi}\|_\infty \|w_i\|$, where $\|\cdot\|$ here denotes the L_2 norm, and the infinity norm of the linear system is defined as $\|G_{rwi}\|_\infty = \sup_\omega \|G(j\omega)\| = \sup_\omega \overline{\sigma}(G(j\omega))$. Here, $\overline{\sigma}$ denotes the maximum singular value.

Then, the threshold values (th_i) to minimize the occurrence of false alarms can be chosen as:

$$th_i = \|G_{rwi}\|_\infty \max \|w_i\| \tag{3.30}$$

where $\max \|w_i\|$ is an estimation of maximum value of $\|wi\|$ based on a priori measurements.

Hence, the weak sensor fault isolation is achieved by formulating the decision signal σ_i based on the following relation to indicate that the ith sensor is faulty:

$$\sigma_i = \begin{cases} 1 & \text{if } \|r_i\| \leq th_i \text{ and } \|r_{\hat{i}}\| > th_{\hat{i}},\ \forall \hat{i} \neq i \\ 0 & \text{otherwise.} \end{cases} \tag{3.31}$$

3.3 Sensor fault isolation using UIO in networked LTI control systems

This section aims to generalize the sensor fault isolation using UIO from the case of robot platoon into the networked LTI control systems.

3.3.1 Sensor fault in networked LTI control systems

In networked control systems, besides the local control inputs, the states of each subsystem in that network are also affected by the outputs of its neighbouring subsystems. This is shown in Eq (3.32) as follows:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(j)} &= A^{(j)}\boldsymbol{x}^{(j)} + B^{(j)}\boldsymbol{u}^{(j)} + I^{(j)}\boldsymbol{i}^{(j)} \\ \boldsymbol{y}^{(j)} &= C^{(j)}\boldsymbol{x}^{(j)} \end{aligned} \tag{3.32}$$

where subscript $j = 1, 2, 3, ..., N$ represents the jth subsystem, $\boldsymbol{x}^{(j)} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{y}^{(j)} \in \mathbb{R}^m$ is the measured output vector, $\boldsymbol{u}^{(j)} \in \mathbb{R}^r$ is the control input vector, and $\boldsymbol{i}^{(j)} \in \mathbb{R}^p$ is the interconnection input vector which represents the outputs from the neighbouring subsystems. $I^{(j)}$ is an interconnection input matrix with an appropriate dimension.

By assuming static linear interconnections, the interconnection inputs $\boldsymbol{i}^{(j)}$ can be generally written as:

$$\begin{aligned} \boldsymbol{i}^{(j)} &= L\boldsymbol{y}^{(k)} \\ \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ . \\ . \\ . \\ i_N \end{bmatrix} &= \begin{bmatrix} 0 & L_{12} & L_{13} & . & . & . & L_{1N} \\ L_{21} & 0 & L_{23} & . & . & . & L_{2N} \\ L_{31} & L_{32} & 0 & . & . & . & L_{3N} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ L_{N1} & L_{N2} & L_{N3} & . & . & . & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ . \\ . \\ . \\ y_N \end{bmatrix} \end{aligned} \tag{3.33}$$

where L is called the interconnection matrix. This L matrix contains the matrices L_{jk} which describes the relations between output measurements from the kth subsystem $\boldsymbol{y}^{(k)}$ being connected as interconnection inputs to the jth subsystem $\boldsymbol{i}^{(j)}$.

For N connected LTI subsystems subjected to sensor faults and measurement noises, each of them has dynamics described by state space equations as follows:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(j)} &= A^{(j)}\boldsymbol{x}^{(j)} + B^{(j)}\boldsymbol{u}^{(j)} + I^{(j)}(\boldsymbol{i}^{(j)} + \boldsymbol{\delta i}^{(j)}) \\ \boldsymbol{y}^{(j)} &= C^{(j)}\boldsymbol{x}^{(j)} + \boldsymbol{f}_s^{(j)} + \boldsymbol{w}^{(j)} \end{aligned} \tag{3.34}$$

where $\boldsymbol{f}_s^{(j)} \in \mathbb{R}^m$ is the sensor fault vector, $\boldsymbol{w}^{(j)} \in \mathbb{R}^m$ is the measurement noises vector, and $\boldsymbol{\delta i}^{(j)} \in \mathbb{R}^p$ is the disturbance term that describes the sensor faults propagation through network such that:

$$\boldsymbol{\delta i}^{(j)} = \sum_{k=1}^{N} L_{jk}(\boldsymbol{f}_s^{(k)} + \boldsymbol{w}^{(k)}) \tag{3.35}$$

Now, consider a general linear control algorithm where the control input $\boldsymbol{u}^{(j)}$ depends on the output measurements $\boldsymbol{y}^{(j)}$ so that:

$$\boldsymbol{u}^{(j)} = K^{(j)}\boldsymbol{y}^{(j)} + \boldsymbol{u}_c^{(j)} \tag{3.36}$$

where $K^{(j)}$ is the controller gain matrix and $\boldsymbol{u}_c^{(j)}$ is a feed forward term.

Hence, by rearranging Eq (3.36), it can be seen that the sensor faults are now affecting the dynamics of the subsystem as follows:

$$\begin{aligned}\dot{\boldsymbol{x}}^{(j)} &= \left[A^{(j)} + B^{(j)}K^{(j)}C^{(j)}\right]\boldsymbol{x}^{(j)} + \begin{bmatrix}B^{(j)} & I^{(j)}\end{bmatrix}\begin{bmatrix}\boldsymbol{u}_c^{(j)} \\ \boldsymbol{i}^{(j)}\end{bmatrix} \\ &\quad + \left[I^{(j)}\boldsymbol{\delta i}^{(j)} + B^{(j)}K^{(j)}\boldsymbol{f}_s^{(j)} + B^{(j)}K^{(j)}\boldsymbol{w}^{(j)}\right] \\ \boldsymbol{y}^{(j)} &= C^{(j)}\boldsymbol{x}^{(j)} + \boldsymbol{f}_s^{(j)} + \boldsymbol{w}^{(j)}\end{aligned} \tag{3.37}$$

3.3.2 UIO for sensor fault isolation in networked LTI control systems

In this section, we show that the UIO presented in Appendix C.5 can be applied to develop proper weak sensor fault isolation despite fault propagation through the network. The resulting observer can be implemented in a distributed way which means that it uses local subsystem's measurements and the measurements of the neighbouring subsystems. Thus, it is scalable.

The model in Eq (3.37) can be written in a more compact way as follows:

$$\begin{aligned}\dot{\boldsymbol{x}}^{(j)} &= \hat{A}^{(j)}\boldsymbol{x}^{(j)} + \hat{B}^{(j)}\hat{\boldsymbol{u}}^{(j)} + \hat{E}^{(j)}\hat{\boldsymbol{d}}^{(j)} \\ \boldsymbol{y}^{(j)} &= C^{(j)}\boldsymbol{x}^{(j)} + \boldsymbol{f}_s^{(j)} + \boldsymbol{w}^{(j)}\end{aligned} \tag{3.38}$$

where

$$\begin{aligned}\hat{A}^{(j)} &= A^{(j)} + B^{(j)}K^{(j)}C^{(j)} \\ \hat{B}^{(j)} &= \begin{bmatrix}B^{(j)} & I^{(j)}\end{bmatrix} \\ \hat{E}^{(j)} &= \begin{bmatrix}I^{(j)} & B^{(j)}K^{(j)}\end{bmatrix} \\ \hat{\boldsymbol{d}}^{(j)} &= \begin{bmatrix}\boldsymbol{\delta i}^{(j)} \\ \boldsymbol{f}_s^{(j)} + \boldsymbol{w}^{(j)}\end{bmatrix} \\ \hat{\boldsymbol{u}}^{(j)} &= \begin{bmatrix}\boldsymbol{u}_c^{(j)} \\ \boldsymbol{i}^{(j)}\end{bmatrix}\end{aligned} \tag{3.39}$$

Remark that Eq (3.38) has the same form as Eq (C.9) such that the UIO design as explained in Appendix C.5 can be applied for weak sensor fault isolation purpose. In addition, it is also seen that the input $\hat{\boldsymbol{u}}^{(j)}$ depends only on the control inputs of the jth subsystem and the interconnection inputs from the measured outputs of the neighbouring subsystems.

However, instead of making the UIO a residual generator that is only sensitive to certain faults, the UIO is designed to be insensitive. This is because the ideal situation in terms of UIO rank requirements is difficult to fulfil in real world implementations. So, to exploit maximum design freedom, each UIO is designed to be insensitive to one or more specific faults [17].

3.4 Simulation results for sensor fault diagnosis

In these simulations, the robot platoons consists of five robots where one robot acts as the leader while the remaining four robots act as the followers. Each of them transmits its GPS-based position measurement to the robot behind it.

In this simulation, the parameters for the controlled leader robot in Eq (3.19) are: $k_I = 0.05$, $k_P = 2$, and $m^{(1)} = 1$ where the values of k_I and k_P are the PI controller gain values for leader's velocity which is determined to get fast settling time with small overshoot.

Meanwhile, the parameters for the remaining four of the follower robot ($j = 2, 3, 4, 5$) in Eq (3.14) are obtained from Eq (3.5) in which $m^{(j)} = 1$, $l^{(j-1)} = 3$, $h = 2$, and $\lambda = 20$ refers to the adaptive cruise control examples in [92]. From those values, it yields that: $k_1^{(j)} = -10$, $k_2^{(j)} = -20.5$, $k_3^{(j)} = 10$, $k_4^{(j)} = 0.5$, and $k_5^{(j)} = -10$.

By using the new $\tilde{C}_i^{(j)}$ matrices as in Eqs (3.24), (3.25), and (3.27), the rank and detectability conditions necessary to UIO existence in Eq (C.18) are checked and it turns out that they are fulfilled.

Subsequently, the UIO's matrices H_i, T_i, F_i, and K_i are then computed to design the residual generators for each robot. The observer gains were determined using pole placement to satisfy all the conditions in Eq (3.23) with poles: at -15 for the UIO related to the leader's sensor S1, at -15 and -10 for the UIO related to the leader's sensor S2, at -15 and -10 for both the UIO related to the follower's sensor S1 and sensor S3, and at -15, -10, and -5 for the UIO related to the follower's sensor S2.

Note that for the leader robot, it does not have both S3 and the related UIO for that sensor. Consequently, it does not have distance measurement as one of its states since it is the leader.

After that, Eq (3.29) and Eq (3.30) are used to calculate the residuals threshold by considering measurement noises where $\max \|w_i\| = 0.35$ in the simulation. Also, it is obtained that $\|G_{rwi}\|_\infty$ are 1 for the followers and 2.6 for the leader.

Lastly, weak sensor fault isolation is done by observing the produced residual signals from the UIOs and then comparing them with the relationship shown in Eq (3.31). In this simulation, the leader's desired speed is set to 10 $\frac{m}{s}$.

There are three different scenarios that are simulated. In these scenarios, all of the sensor faults are simulated as sensor bias errors using a step function where the magnitude is increased from 0 to 10 at the 10th second.

In the first scenario, the fault is happening at sensor S1 in the first robot (leader) by which the system's response and UIO's residual signals are shown in Figs 3.3, 3.4, 3.5, and 3.6 respectively.

In the second scenario, the fault is happening at sensor S2 in the third robot by which the system's response and UIO's residual signals are shown in Figs 3.7, 3.8, 3.9, and 3.10 respectively.

Finally, in the third scenario, the fault is happening at sensor S3 in the 5th robot by which the system's response and UIO's residual signals are shown in Figs 3.11, 3.12, 3.13, and 3.14 respectively.

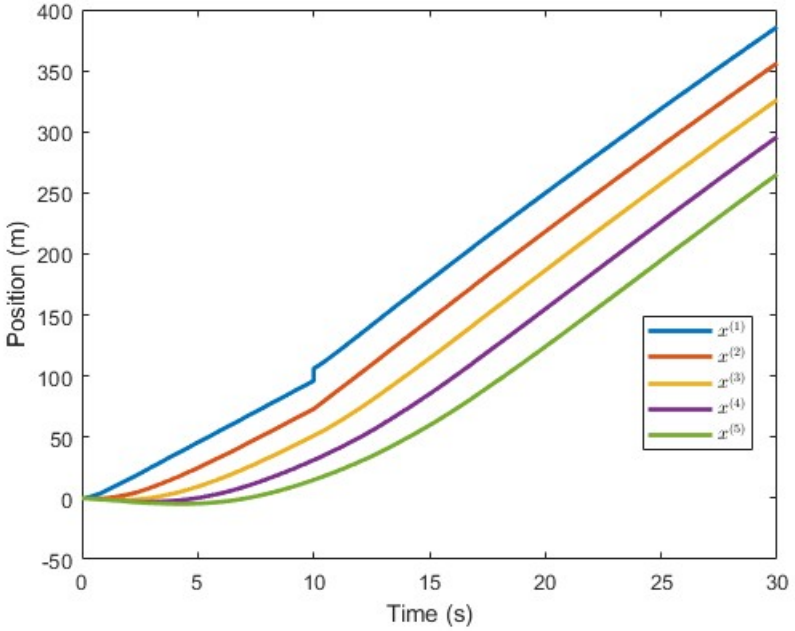

FIGURE 3.3: Robot's positions with faulty sensor S1 in the 1st robot

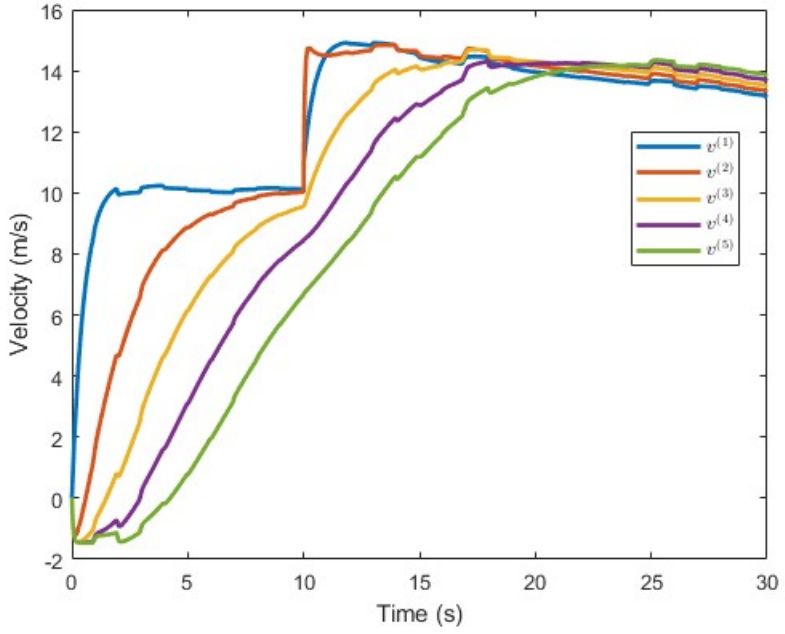

FIGURE 3.4: Robot's velocities with faulty sensor S1 in the 1st robot

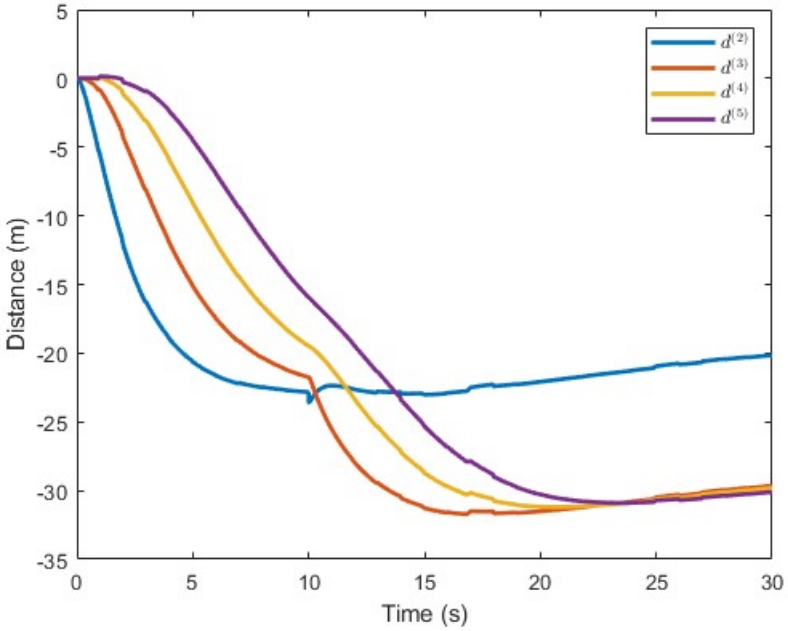

FIGURE 3.5: Robot's distances with faulty sensor S1 in the 1st robot

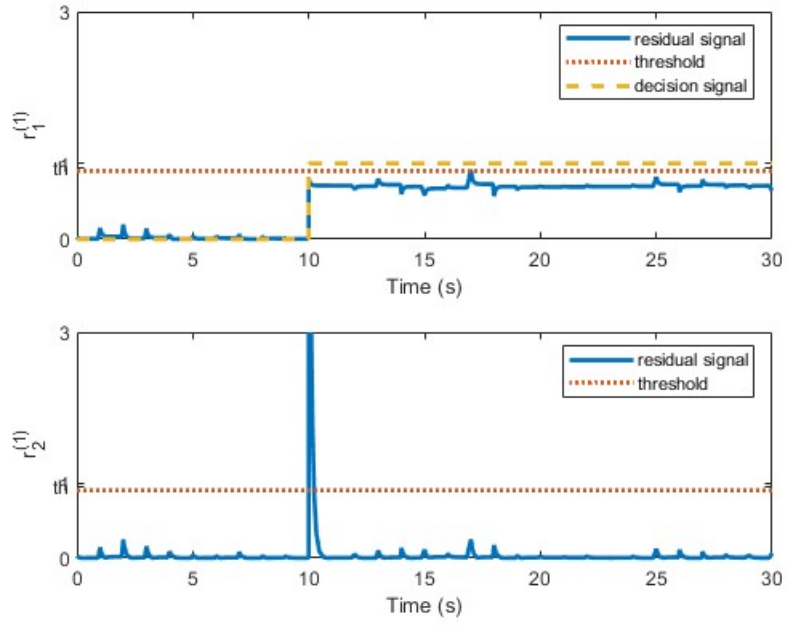

FIGURE 3.6: Residual signals with faulty sensor S1 in the 1st robot

For the first scenario, Figs 3.3, 3.4, and 3.5 shows that fault in S1 in the first robot is affecting all the states of the 2nd, 3rd, 4th, and 5th robots. This is happening because, as mentioned earlier, the transmitted information in the platoons' network comes from the GPS-based position measurement (S1). Besides that, based on the previous relation shown in the Eq (3.31), Fig 3.6 shows a significant change of amplitude in the residual signals produced by the UIO for sensor S2 indicating that a fault occurred in sensor S1 at the 10th second, which is correct.

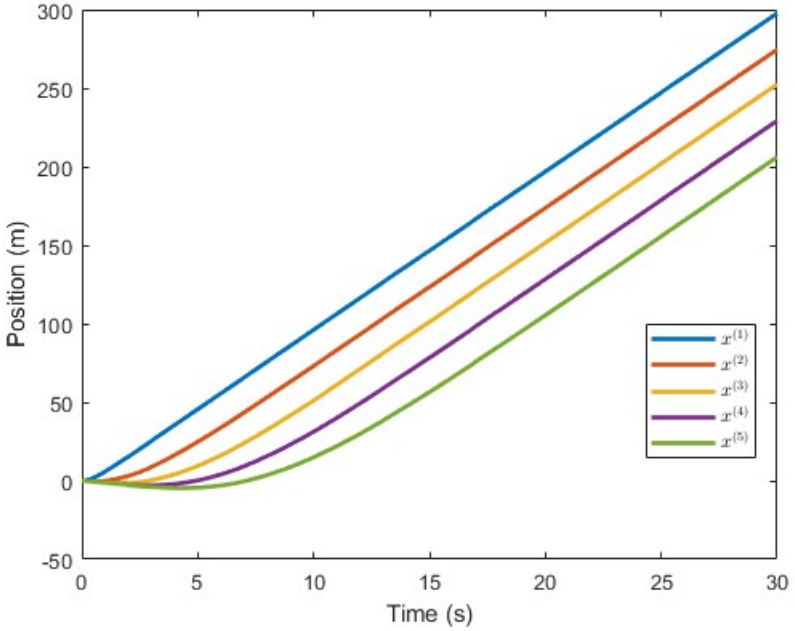

FIGURE 3.7: Robot's positions with faulty sensor S2 in the 3rd robot

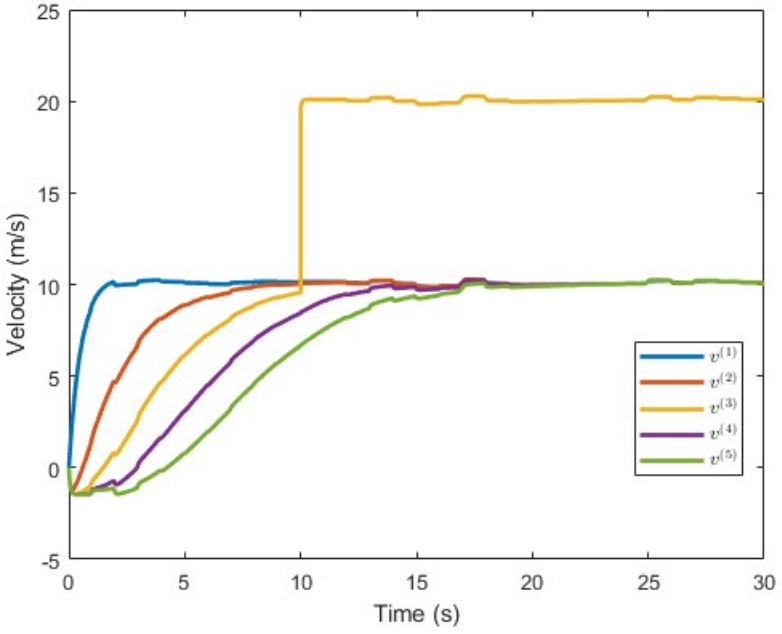

FIGURE 3.8: Robot's velocities with faulty sensor S2 in the 3rd robot

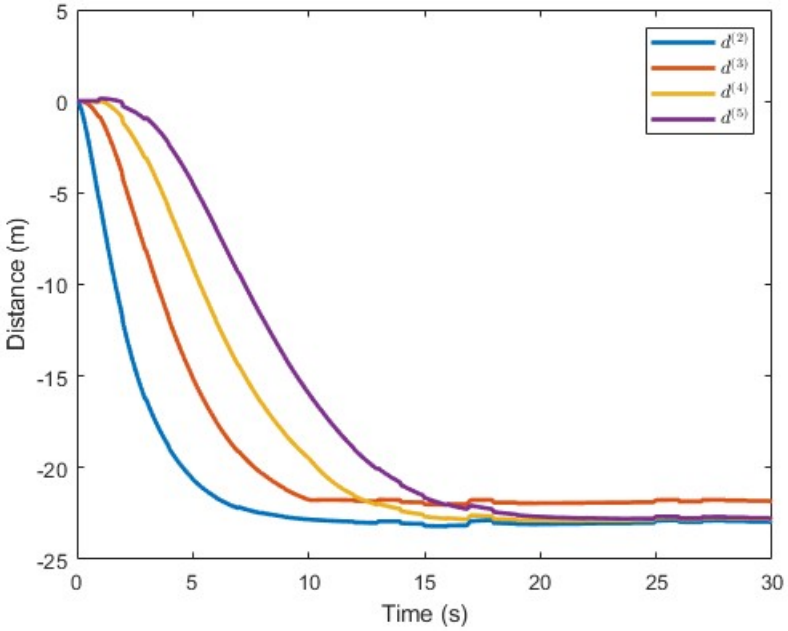

FIGURE 3.9: Robot's distances with faulty sensor S2 in the 3rd robot

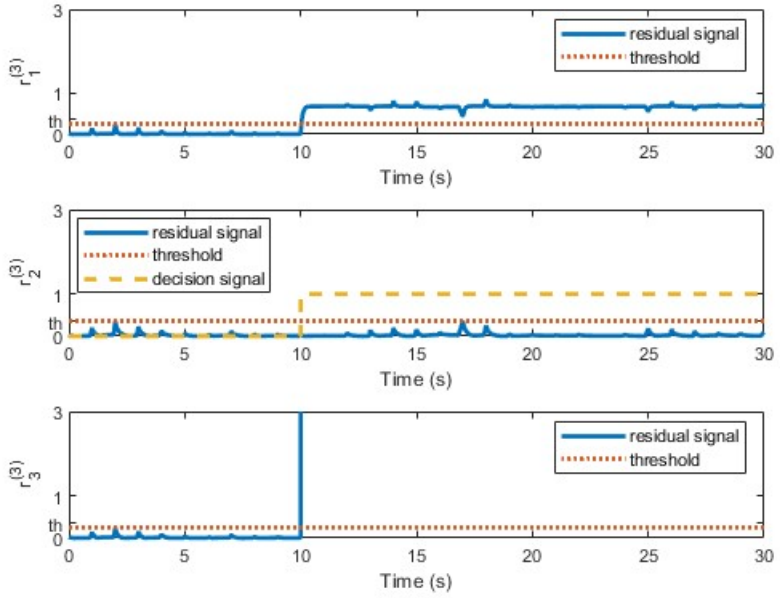

FIGURE 3.10: Residual signals with faulty sensor S2 in the 3rd robot

For the second scenario, Figs 3.7, 3.8, and 3.9 shows that fault in S2 in the third robot does not directly affect the states of the 4th and 5th robots because the measurement of sensor S2 is not the information that is transmitted to the robot behind it. In addition, Fig 3.10 shows a change of amplitude exceeding the threshold in both the residual signals from this robot's UIO for sensor S1 and sensor S3 indicating that sensor S2 is faulty at the 10th second, which is correct.

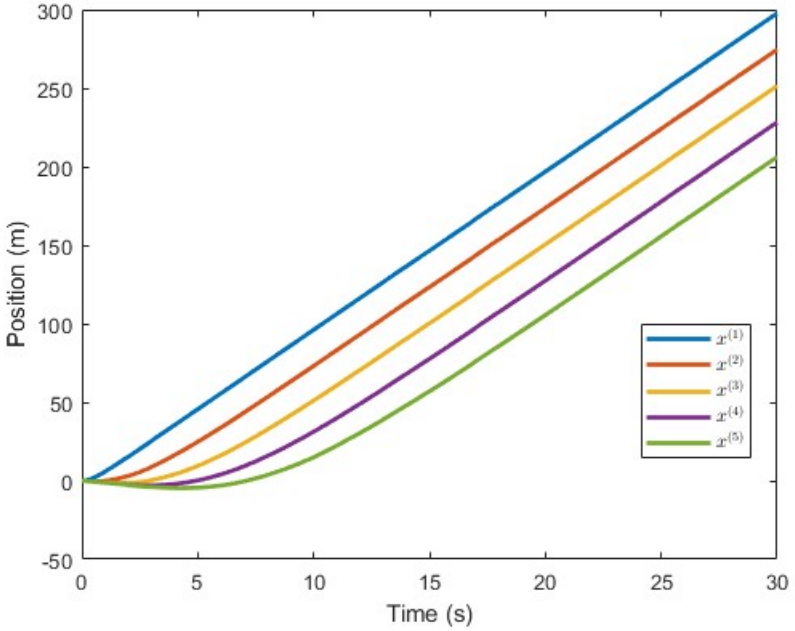

FIGURE 3.11: Robot's positions with faulty sensor S3 in the 5th robot

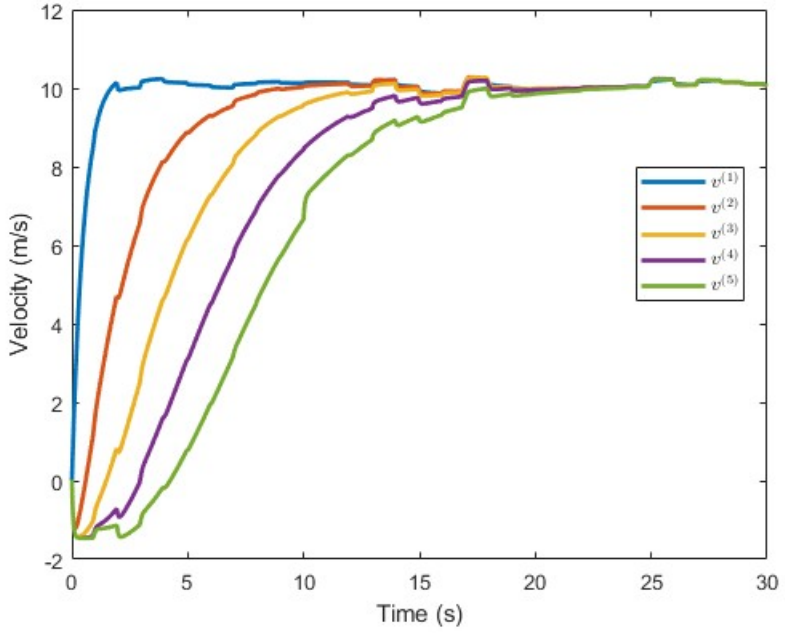

FIGURE 3.12: Robot's velocities with faulty sensor S3 in the 5th robot

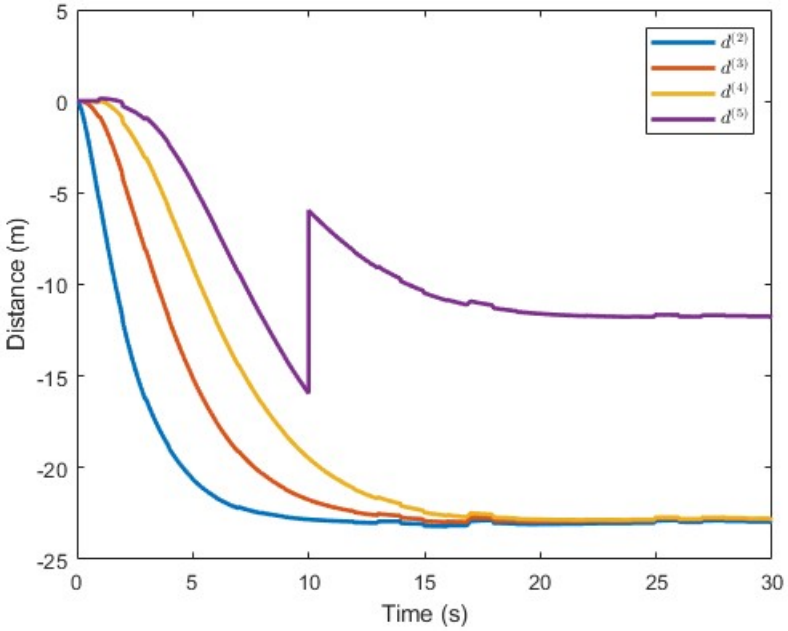

FIGURE 3.13: Robot's distances with faulty sensor S3 in the 5th robot

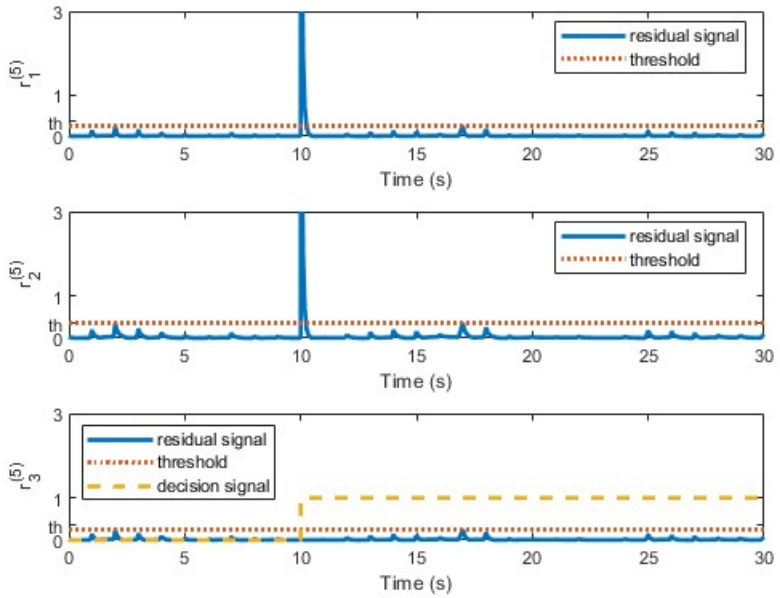

FIGURE 3.14: Residual signals with faulty sensor S3 in the 5th robot

Lastly, for the third scenario, Figs 3.11, 3.12, and 3.13 shows that a fault in S3 in the fifth robot does not directly affect any other robots at all because this 5th robot is the last in the platoons. Furthermore, Fig 3.14 shows a significant change of amplitude in this last robot's UIO residual signal for both sensor S1 and sensor S2 indicating that a fault happened at the 10th second in sensor S3, which is correct.

3.5 Actuator fault diagnosis in robot platoons

This section is provided to discuss the actuator fault diagnosis scenario in the robot platoons. Moreover, since the state of the leader robot is not affected by the state of neighbouring subsystems, the discussion only concerns the follower robots ($j = 2, 3, 4, ..., N$) because the proposed method is designed with the main aim of decoupling between actuator fault and network communication fault. As both of them enter the subsystem from the same input channel, such decoupling is necessary so that actuator fault diagnosis can be done.

3.5.1 Model of robot platoons with actuator fault

The actuator fault is modelled as an input signal that additively modifies the control input so that Eq (3.5) becomes:

$$u^{(j)} = u_c^{(j)} + f_a^{(j)} \tag{3.40}$$

where $u^{(j)}$ is the modified control input augmented with actuator fault $f_a^{(j)}$.

Then, based on Eqs (3.6) and (3.40), the related state space model for the jth robot with actuator fault can be written as follows:

$$\begin{aligned}
\begin{bmatrix} \dot{x}_1^{(j)} \\ \dot{x}_2^{(j)} \\ \dot{x}_3^{(j)} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \frac{k_5^{(j)}}{m^{(j)}} & \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}}{m^{(j)}} \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} l^{(j-1)} \\ x^{(j-1)} \\ v^{(j-1)} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} f_a^{(j)} \\
\begin{bmatrix} y_1^{(j)} \\ y_2^{(j)} \\ y_3^{(j)} \\ y_4^{(j)} \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^{(j)} \\ x_2^{(j)} \\ x_3^{(j)} \end{bmatrix}
\end{aligned} \tag{3.41}$$

3.5.2 Filter design for state estimation

Eq (3.6) shows that the states from the preceding $(j-1)$th robot acts as the input for the jth robot. This input can be affected by network disturbances if it is obtained from the information transmission through the network communication.

However, in addition to network communication, it is seen that $x^{(j-1)}$ can be obtained from the relative output information in the jth subsystem as follows:

$$x_m^{(j-1)} = y_1^{(j)} - y_4^{(j)} + w_m^{(j)} \tag{3.42}$$

where $x_m^{(j-1)}$ is the measured position of the $(j-1)$th subsystem based on the relative output information from the jth subsystem and $w_m^{(j)}$ is the measurement noise. This indicates redundancy in the position measurement and will be explored during the actuator fault diagnosis process.

On the other hand, $v^{(j-1)}$ is not directly available from the measurements. Yet, it can be estimated from $x_m^{(j-1)}$ by differentiation as follows:

$$v_m^{(j-1)} = \frac{dx_m^{(j-1)}}{dt} = \frac{d(y_1^{(j)} - y_4^{(j)})}{dt} + \frac{dw_m^{(j)}}{dt} \tag{3.43}$$

where $v_m^{(j-1)}$ is the computed velocity of the $(j-1)$th subsystem based on the relative output information from the jth subsystem.

As $\dot{w}_m^{(j)}$ is a high frequency signal, a low pass filter (LPF) is needed to get rid of its effect. The transfer function of the LPF is considered in the following form:

$$H(s) = \frac{1}{(T_{f1}s+1)(T_{f2}s+1)} \tag{3.44}$$

where $T_{f1}^{(j)}, T_{f2}^{(j)} > 0$ are the filter parameters.

Then, by assuming zero initial values and by utilizing the Laplace transformation property in function differentiation, we can obtain a velocity estimate $v_e^{(j-1)}$ as:

$$v_e^{(j-1)}(s) = \frac{s}{(T_{f1}^{(j)}s+1)(T_{f2}^{(j)}s+1)} x_m^{(j-1)}(s) \tag{3.45}$$

where $v_e^{(j-1)}(s)$ is in Laplace domain.

In the time domain, Eq (3.45) can be written in a state space form as follows:

$$\begin{aligned}
\begin{bmatrix} \dot{x}_{f1}^{(j)} \\ \dot{x}_{f2}^{(j)} \end{bmatrix} &= \begin{bmatrix} a_{f1}^{(j)} & a_{f2}^{(j)} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_{f1}^{(j)} \\ x_{f2}^{(j)} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} x_m^{(j-1)} \\
v_e^{(j-1)} &= \begin{bmatrix} c_f^{(j)} & 0 \end{bmatrix} \begin{bmatrix} x_{f1}^{(j)} \\ x_{f2}^{(j)} \end{bmatrix}
\end{aligned} \tag{3.46}$$

where

$$a_{f1}^{(j)} = \frac{-(T_{f1}^{(j)} + T_{f2}^{(j)})}{T_{f1}^{(j)} T_{f2}^{(j)}};\ a_{f2}^{(j)} = \frac{-1}{T_{f1}^{(j)} T_{f2}^{(j)}};\ c_f^{(j)} = \frac{1}{T_{f1}^{(j)} T_{f2}^{(j)}}.$$

Thus, by using relative output information and LPF, the necessity of network communication to get the position and velocity of the $(j-1)$th robot can be eliminated.

3.5.3 Modified subsystem model for actuator fault diagnosis

By combining Eqs (3.41) and (3.46), the robot model with actuator faults reads as follows:

$$\begin{aligned}
\begin{bmatrix} \dot{x}^{(j)} \\ \dot{v}^{(j)} \\ \dot{x}_{f1}^{(j)} \\ \dot{x}_{f2}^{(j)} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} & \frac{k_4^{(j)} c_f^{(j)}}{m^{(j)}} & 0 \\ 0 & 0 & a_{f1}^{(j)} & a_{f2}^{(j)} \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x^{(j)} \\ v^{(j)} \\ x_{f1}^{(j)} \\ x_{f2}^{(j)} \end{bmatrix} \\
&\quad + \begin{bmatrix} 0 & 0 \\ \frac{k_3^{(j)}}{m^{(j)}} & \frac{k_5^{(j)}}{m^{(j)}} \\ 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_m^{(j-1)} \\ l^{(j-1)} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} f_a^{(j)} \\
\begin{bmatrix} y_1^{(j)} \\ y_2^{(j)} \\ y_3^{(j)} \\ y_4^{(j)} \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x^{(j)} \\ v^{(j)} \\ x_{f1}^{(j)} \\ x_{f2}^{(j)} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} x_m^{(j-1)} \\ l^{(j-1)} \end{bmatrix}
\end{aligned} \tag{3.47}$$

Furthermore, Eq (3.47) can be written as in Eq (B.1) extended by an actuator fault term in the following form:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(j)} &= \bar{A}^{(j)}\boldsymbol{x}^{(j)} + \bar{B}^{(j)}\boldsymbol{u}^{(j)} + \bar{L}^{(j)}\boldsymbol{x}^{(\boldsymbol{j-1})} + \bar{F}^{(j)}\boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})} \\ \boldsymbol{y}^{(j)} &= \bar{C}^{(j)}\boldsymbol{x}^{(j)} + \bar{D}^{(j)}\boldsymbol{u}^{(j)} + \bar{E}^{(j)}\boldsymbol{x}^{(\boldsymbol{j-1})} \end{aligned} \tag{3.48}$$

where

$$\bar{A}^{(j)} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{k_1^{(j)}}{m^{(j)}} & \frac{k_2^{(j)}}{m^{(j)}} & \frac{k_4^{(j)}c_f^{(j)}}{m^{(j)}} & 0 \\ 0 & 0 & a_{f1}^{(j)} & a_{f2}^{(j)} \\ 0 & 0 & 1 & 0 \end{bmatrix}; \bar{B}^{(j)} = \begin{bmatrix} 0 \\ \frac{k_5^{(j)}}{m^{(j)}} \\ 0 \\ 0 \end{bmatrix};$$

$$\bar{L}^{(j)} = \begin{bmatrix} 0 \\ \frac{k_3^{(j)}}{m^{(j)}} \\ 1 \\ 0 \end{bmatrix}; \bar{F}^{(j)} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}; \bar{C}^{(j)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix};$$

$$\bar{D}^{(j)} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}; \bar{E}^{(j)} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \end{bmatrix}; \boldsymbol{x}^{(j)} = \begin{bmatrix} x^{(j)} & v^{(j)} & x_{f1}^{(j)} & x_{f2}^{(j)} \end{bmatrix}^T;$$

$$\boldsymbol{u}^{(j)} = I^{(j-1)}; \boldsymbol{x}^{(\boldsymbol{j-1})} = x_m^{(j-1)}; \boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})} = f_a^{(j)}$$

3.5.4 Actuator fault estimator design in robot platoons

By assuming that $\boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})}$ is constant, a PI observer can be used to estimate its value [121]. The basic knowledge as in Appendix C.4 is used here by replacing disturbance $\boldsymbol{d}$ in Eq (C.9) with actuator fault $\boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})}$ and other further necessary modifications.

As a start, the actuator fault $\boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})}$ is firstly included in the state vector $\boldsymbol{x}^{(j)}$ to form a new extended state $\boldsymbol{z}^{(j)} = [[\boldsymbol{x}^{(j)}]^T \ \boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})}]^T$ so that we can write an extended state space version of Eq (3.48) as follows:

$$\begin{aligned} \dot{\boldsymbol{z}}^{(j)} &= \begin{bmatrix} \bar{A}^{(j)} & \bar{F}^{(j)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(j)} \\ \boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})} \end{bmatrix} + \begin{bmatrix} \bar{B}^{(j)} & \bar{L}^{(j)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{u}^{(j)} \\ \boldsymbol{x}^{(\boldsymbol{j-1})} \end{bmatrix} \\ &= \bar{A_z}^{(j)}\boldsymbol{z}^{(j)} + \bar{B_z}^{(j)}\boldsymbol{u}_{\boldsymbol{z}}^{(\boldsymbol{j})} \\ \boldsymbol{y}^{(j)} &= \begin{bmatrix} \bar{C}^{(j)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^{(j)} \\ \boldsymbol{f}_{\boldsymbol{a}}^{(\boldsymbol{j})} \end{bmatrix} + \begin{bmatrix} \bar{D}^{(j)} & \bar{E}^{(j)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{u}^{(j)} \\ \boldsymbol{x}^{(\boldsymbol{j-1})} \end{bmatrix} \\ &= \bar{C_z}^{(j)}\boldsymbol{z}^{(j)} + \bar{D_z}^{(j)}\boldsymbol{u}_{\boldsymbol{z}}^{(\boldsymbol{j})} \end{aligned} \tag{3.49}$$

If $(\bar{A_z}^{(j)}, \bar{C_z}^{(j)})$ is an observable pair, then a PI observer can be designed for the extended state space model as in Eq (3.49) which estimates not only the states but

also the actuator fault. The state space model of the PI observer is:

$$\begin{aligned}
\dot{\hat{\boldsymbol{z}}}^{(j)} &= \left(\begin{bmatrix} \bar{A}^{(j)} & \bar{F}^{(j)} \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} \bar{L_P}^{(j)} \\ \bar{L_I}^{(j)} \end{bmatrix} \begin{bmatrix} \bar{C}^{(j)} & 0 \end{bmatrix} \right) \begin{bmatrix} \hat{\boldsymbol{x}}^{(j)} \\ \hat{\boldsymbol{f}}_{\boldsymbol{a}}^{(j)} \end{bmatrix} \\
&+ \begin{bmatrix} \bar{L_P}^{(j)} \\ \bar{L_I}^{(j)} \end{bmatrix} y^{(j)} + \begin{bmatrix} \bar{B}^{(j)} & \bar{L}^{(j)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{u}^{(j)} \\ \boldsymbol{x}^{(j-1)} \end{bmatrix} \\
&= (\bar{A_z}^{(j)} - \bar{L_z}^{(j)} \bar{C_z}^{(j)}) \hat{\boldsymbol{z}}^{(j)} + \bar{L_z}^{(j)} y^{(j)} + \bar{B_z}^{(j)} \boldsymbol{u}_{\boldsymbol{z}}^{(j)} \\
&= \bar{A_z}^{(j)} \hat{\boldsymbol{z}}^{(j)} + \bar{L_z}^{(j)} (y^{(j)} - \hat{y}^{(j)}) + \bar{B_z}^{(j)} \boldsymbol{u}_{\boldsymbol{z}}^{(j)} \\
\hat{y}^{(j)} &= \begin{bmatrix} \bar{C}^{(j)} & 0 \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}^{(j)} \\ \hat{\boldsymbol{f}}_{\boldsymbol{a}}^{(j)} \end{bmatrix} + \begin{bmatrix} \bar{D}^{(j)} & \bar{E}^{(j)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{u}^{(j)} \\ \boldsymbol{x}^{(j-1)} \end{bmatrix} \\
&= \bar{C_z}^{(j)} \hat{\boldsymbol{z}}^{(j)} + \bar{D_z}^{(j)} \boldsymbol{u}_{\boldsymbol{z}}^{(j)}
\end{aligned} \tag{3.50}$$

If $\bar{L_z}^{(j)} = [\bar{L_P}^{(j)} \ \bar{L_I}^{(j)}]^T$ is chosen such that $(\bar{A_z}^{(j)} - \bar{L_z}^{(j)} \bar{C_z}^{(j)})$ is Hurwitz, then $\lim_{t \to \infty} (\boldsymbol{z}^{(k)} - \hat{\boldsymbol{z}}^{(k)}) = 0$, i.e. the actuator fault will be correctly estimated.

To choose the appropriate observer's gain $\bar{L_z}^{(j)}$, one can use the easiest way such as the pole placement method. This method is popular because the obtained gain can manipulate the estimator's dynamic according to the designer's wish by placing the poles in the desirable locations [122]. Meanwhile, if measurement noises are considered, the LQE methods can be applied [123].

Remark: It is already checked that the pair (A, C) in system (3.47) (or (3.48)) is observable. However, if we delete both the second and the third row of the C matrix in the related system's output equation (i.e. $y_2^{(j)}$ and $y_3^{(j)}$ do not exist), it is found that the resulting system is still observable. In other words, the proposed actuator fault diagnosis method will still work even though no velocity measurement is available.

3.6 Simulation results for actuator fault diagnosis

In this simulation, a leader-follower scheme in the robot platoons consisting of 3 robots is built based on Eq (3.48) except for the leader robot. The first robot ($j = 1$), which acts as the leader, moves at a pre-determined controlled speed. The leader is followed by the second robot ($j = 2$) which is then followed by the third robot ($j = 3$) consecutively. Furthermore, an LPF based on Eq (3.46) is designed for each of the second and third robots as the followers by which its parameters are chosen so that the filter's dynamics are faster than the robot's dynamics. The used parameters for the follower robot's model and filter are shown in Table 3.1 where each robot is assumed to have a mass of 1 ($m^{(j)} = 1$) and a length of 3 ($l^{(j)} = 3$).

TABLE 3.1: Robot model and filter parameters

Parameters	$k_1^{(j)}$	$k_2^{(j)}$	$k_3^{(j)}$	$k_4^{(j)}$	$k_5^{(j)}$	$T_{f1}^{(j)}$	$T_{f2}^{(j)}$
For $j = 2, 3$	-10	-20.5	10	0.5	-10	0.05	0.1

To estimate the actuator fault, an observer based on Eq (3.50) is designed for every follower. As their sensor measurements are subjected to band limited white noise with 10^{-4} noise power and 0.1 sample time in the simulation, two methods are used in determining the needed observer's gain to check the performance: the common pole placement design method and LQE. For pole placement, the command *place* in MATLAB is used to get the necessary gains in placing the poles at -5, -5.1,

-5.2, -5.3, and -5.4. Meanwhile, for LQE, the command *lqe* in MATLAB is used to get the necessary gains with the same $Q = 1$ but with different R which are:

$$R_1 = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}, R_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, R_3 = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

where Q is the process noise variance and R is the measurement noise variance.

The position ($x^{(j)}$) for each robot when there are no faults is shown in Fig. 3.15. It is seen that each follower keeps a predefined distance from its neighbour.

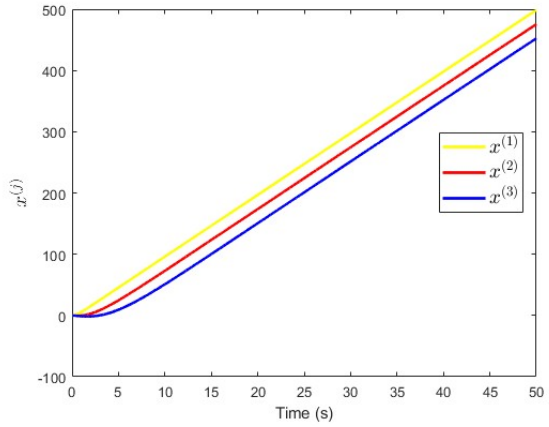

FIGURE 3.15: Robots' positions when there are no actuator faults

Fig 3.16 shows the output difference between the system that uses network communication as in Eq (3.41) and the system without network communication as in Eq (3.47). It is seen that the model without network communication is good enough to replace the one that uses network communication.

For the leader, there is no difference because it does not have network communication (as a leader, it does not have a preceding subsystem to be followed).

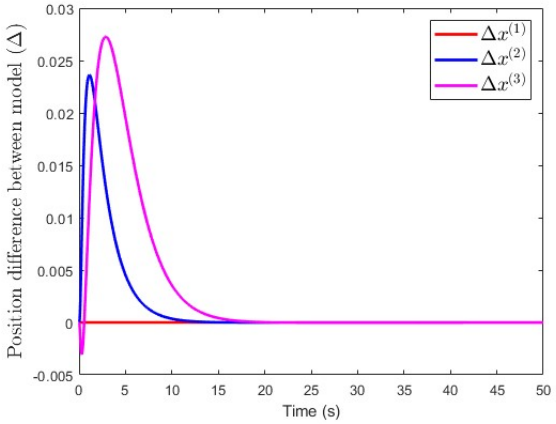

FIGURE 3.16: Difference in position responses of the models with and without network communication

In the next scenario, an actuator fault with an amplitude of 150 is injected into the second robot ($f_a^{(2)} = 150$) at the 25th second. The position of each robot is shown

in Fig 3.17. Meanwhile, the observer's fault estimation using pole placement and LQE are shown in Figs 3.18 and 3.19 respectively.

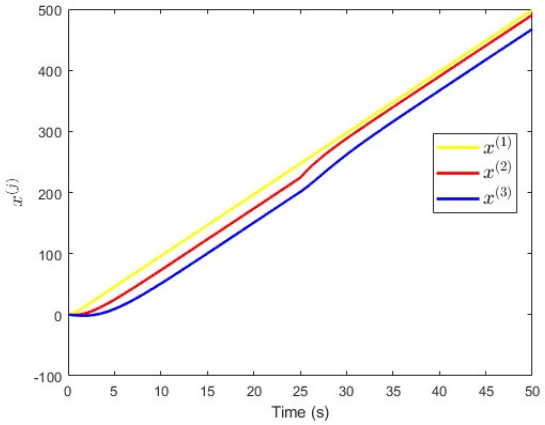

FIGURE 3.17: Robots' positions when there is an actuator fault in the 2nd robot

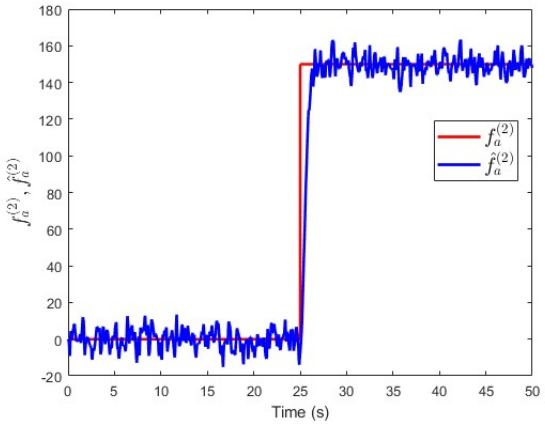

FIGURE 3.18: Actuator fault signal and its estimate in the 2nd robot using pole placement

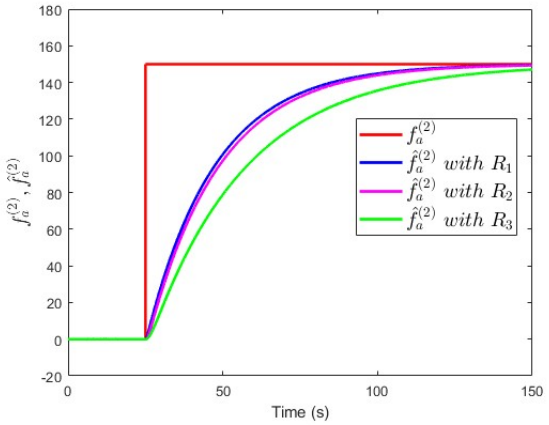

FIGURE 3.19: Actuator fault signal and its estimate in the 2nd robot using LQE

It is seen that the actuator fault causes a deviation in the motion of the second and third robots consecutively. After the fault occurred, the position between the second and the third robots got farther away because the platoons implemented a constant time gap policy. In this policy, the desired inter spacing between subsystems varies proportionally to the velocity as seen in Eq (3.3). Thus, when their velocity increases because of the fault, the gap between them is widening. On the other hand, by using pole placement, the observer successfully estimates the fault in an instant even though it is affected by the measurement noise. By using LQE, the noise's effect is eliminated. However, for each value of R_1, R_2, and R_3, they take a much longer time to estimate the fault which is around 100 seconds at best.

In the last scenario, an actuator fault with an amplitude of 100 is injected into the third robot ($f_a^{(3)} = 100$) at the 25th second. The position of each robot is shown in Fig 3.20. Meanwhile, the observer's fault estimation using pole placement and LQE are shown in Figs 3.21 and 3.22 respectively.

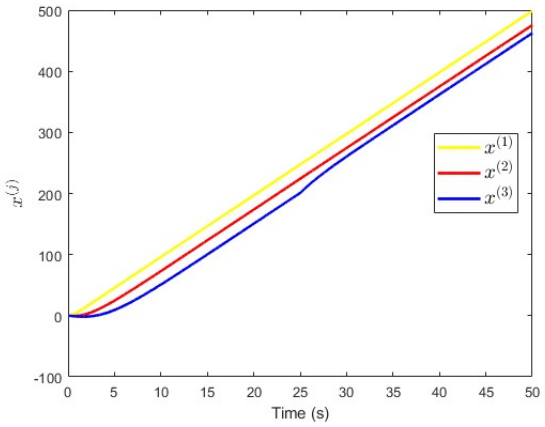

FIGURE 3.20: Robots' positions when there is an actuator fault in the 3rd robot

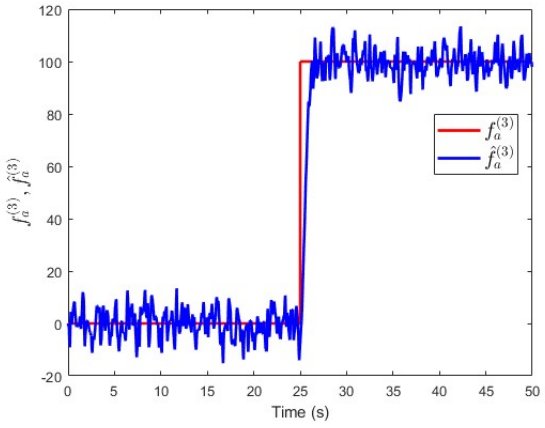

FIGURE 3.21: Actuator fault signal and its estimate in the 3rd robot using pole placement

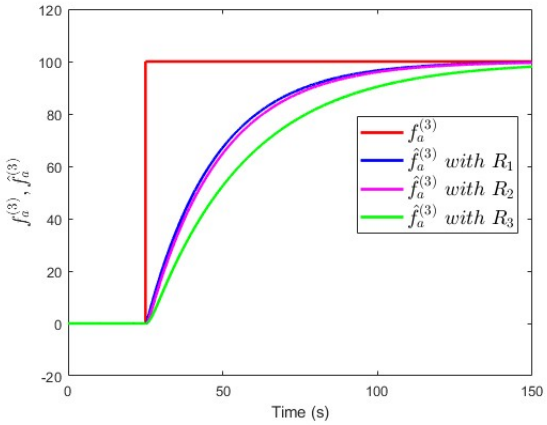

FIGURE 3.22: Actuator fault signal and its estimate in the 3rd robot using LQE

It is seen that the actuator fault causes a deviation only in the motion of the third robot. On the other hand, by using pole placement, the observer also successfully estimates the fault in an instant even though it is still affected by the measurement noise. By using LQE, the noise's effect is eliminated. However, for each value of R_1, R_2, and R_3, the results are the same as before where they take a much longer time to estimate the fault which is around 100 seconds at best.

3.7 Summary and discussion

In this chapter, the actuator and sensor fault diagnosis in the robot platoons that move in a leader-follower scheme is considered. For actuator fault diagnosis, no sensor fault is assumed. Likewise, for sensor fault diagnosis, no actuator fault is assumed. Each fault diagnosis problem is studied separately. In the case of actuator fault diagnosis, the detection, isolation, and estimation of the fault are the goals. Meanwhile, in the case of sensor fault diagnosis, the goals are to detect and isolate the faulty sensors in each robot of the platoons.

As a networked system, the states of each robot in the platoons are affected by the states of the preceding robot, except the leader. Network communication between subsystems is considered for the sake of state information transmission. It is assumed that the control input for each robot is computed from the time-gap policy used in the Adaptive Cruise Control systems. In addition, some conventional sensors are considered in each robot which are: a GPS-based sensor, a wheel-mounted velocity sensor, and a radar sensor for inter-vehicle distance measurement.

As actuator fault and network communication fault enter the subsystem through the same input channel, their decoupling needs to be considered during the actuator fault diagnosis. To do this, by exploiting measurement redundancy from the sensors, a filtering technique is proposed to get rid of the necessity of network communication so that the communication fault can be neglected. Thus, a new model that does not include network communication is constructed. Then, by using the new model as the basis, a PI observer is designed for the actuator fault estimation purpose. If the observer's gain is determined from the pole placement design method, it is found that the fault is estimated rapidly but the measurement noise was not suppressed. Meanwhile, by using the LQE design to determine the observer's gain, the measurement noise is eliminated in exchange for a much longer settling time.

Unlike in the actuator fault diagnosis, network communication is considered during the sensor fault diagnosis because the sensor fault from a specific subsystem can propagate through the network so that it disturbs the whole networked system. Thus, a networked system model that can describe the expected behaviour of the robot platoons with sensors fault and their propagation is constructed. By treating the fault propagation as the unknown input, a bank of distributed Unknown Input Observers for each robot is designed to detect and isolate the faulty sensor. As sensor measurement noise is considered, a threshold computation method is also introduced to prevent a false alarm when there are no faulty sensors. It is found that the weak sensor fault isolation in the robot platoons can be achieved by using this local UIOs approach in each subsystem which is more scalable than a single global UIO for the whole networked system.

The proposed sensor and actuator fault diagnosis methods here are developed on the robot platoons which move along one axis. Thus, for another system's implementation, preliminary steps must be taken. For example, in the case of Unmanned Aerial Vehicles (UAV)s, some decouplings need to be done as their movement is in more than one axis (i.e. pitch, roll, yaw). Moreover, as the proposed method targets weak fault isolation, some faults should be grouped (fault grouping).

Compared to other works in robot and vehicle platoons such as [63, 64, 60, 65] that prioritizes the control objectives over fault identification, this thesis concentrates on fault detection, isolation, and estimation so that the fault can be better compensated. The proposed model based methods use only 3 conventional sensors compared to [62], and a bank of local UIOs which belongs to the group of linear observers compared to [59] that use nonlinear observers. In comparison with [66] and [67] where the UIO is used only to detect a faulty agent, the UIO in this thesis is designed to detect a specific faulty sensor. Furthermore, a bank of local UIOs means that they are more scalable compared to the global UIO introduced by [68].

Chapter 4

Parameter fault diagnosis in heat exchange networks

Heat exchange units form an important class of operating elements in process systems [124]. They are commonly used for heating/cooling systems of a large building or a district of living houses. Here, one can find a heating/cooling source that serves a network of connected consumers where the heating/cooling medium (e.g. water) circulates in closed and well isolated tubes. In many cases, because of financial reasons, there are only a few sensors in such systems that are placed at the source and consumers.

The dynamic models of heat exchange units can be derived from first engineering principles, in particular from energy balances, and they are in the simplest lumped parameter case linear or bilinear state space models depending on the chosen or available input variables [125]. Thus, the system's states (e.g. temperatures) are spatially distributed along the tube length.

In most cases, heat exchange units are used in multiple instances forming a Heat Exchange Network (HEN), which is a networked dynamic system. A recent book by [126] describes network based methods for analysing structural properties of dynamic systems in general, and heat exchanger networks in particular.

Generally, the fault diagnosis design in networks of dynamic systems represents a hard problem due to the fault effect propagation through the network connections. In some cases, the specifics of the interconnections and the subsystems have to be explored to design fault diagnosis methods with guaranteed performances [127, 128]. Some also tried to use heuristic approaches such as fuzzy neural network [129] or machine learning [130] to handle it. A fuzzy observer based fouling detection method for heat exchangers was proposed in [131]. A nonlinear high gain observer based fault diagnosis method was introduced in [132] for such heat exchangers in which exothermic chemical reactions take place.

From the perceptive of fault types, a huge literature also exists. In the study of [133], several types of faults in heating networks were identified: leakage, pressure losses, and heat losses. A neural network based leakage detection method was proposed in [134] for heating networks with small and constant supply flows. The deterioration detection of the heat transfer surface by ageing was investigated in [135]. A recent study by [136] has also proposed a dissipation based distributed fault diagnosis that can detect and isolate actuator faults in HENs as the case study. A review of fault diagnosis strategies for district heating and cooling systems is presented in [15].

The general aim of this research is to develop simple yet powerful diagnosis methods for networked heat exchange units. In this present study, the particularities of the addressed system class will be explored.

In this section, the considered fault in HENs is the change in the heat transfer coefficient caused by the deterioration of the heat transfer surface because of ageing or degradation of the isolation of the heat transferring tubes. The fault diagnosis method is built upon a simple specific model of HENs. Thus, a model based approach for parameter fault diagnosis is proposed.

Two different fault scenarios are investigated in this study by which the goals are to detect, isolate, and estimate the faults. In the first scenario, the parameter fault is considered to happen along the length of the tube in an element in the HENs which contains splitting and joining connections. The author's related publication for this first scenario is [A3]. Meanwhile, in the second scenario, the parameter fault is considered to happen only in a certain position along the length of the tube connecting the elements in the HENs. In this second scenario, to differentiate it from the first scenario, we change the fault isolation term into fault localization, i.e. finding the isolation breakdown position along the tube. For this second scenario, the author's related publication is [A4]. The two scenarios are described using different system models (see Section 4.2 for details).

The research questions that we are going to tackle are as follows:

1. In the heat exchange networks that contain splitting and joining connections, how to identify a parameter fault that is happening either along the length of a heat exchange element or in the pipes connecting the subsystems?

2. Which sensor measurements are necessary to isolate the previously mentioned parameter faults?

3. By using the previously mentioned sensor measurements, how to estimate the magnitude of the fault?

4.1 Fault free model of heat exchange networks

Generally, the dynamic model of elements in HENs can be built from the energy conservation equations in their lumped model form (see Appendix A.2). Furthermore, even the distributed delay interconnections between elements can be represented in the same way (see Appendix B.1). Thus, the model of every element (heat exchange units) in HENs and their interconnections can be obtained by applying Eq (A.7). However, to make it more general, the environmental temperature will be considered by adding the heat transfer term $k_E(T_{EXT} - x_i)$ where $k_E > 0$ is the heat transfer coefficient and T_{EXT} represents the temperature of the environment. Meanwhile, the input u in Eq (A.7) is considered as the first input u_1 of the system. For connecting elements, this heat transfer term is also assumed to exist in the heat isolation wall. In the case of heating devices, T_{EXT} represents the temperature of an external heat source in which heat is transferred to the devices. Thus, by applying this addition,

the system's model for HENs can be written as follows:

$$
\begin{aligned}
CL_j &:= \begin{cases} \dot{\boldsymbol{x}}^{(j)} = A^{(j)}\boldsymbol{x}^{(j)} + B^{(j)}\boldsymbol{u}^{(j)} \\ y^{(j)} = C\boldsymbol{x}^{(j)} \end{cases} \\
A^{(j)} &= \begin{bmatrix} -v^{(j)} - k_E^{(j)} & 0 & ... & 0 \\ v^{(j)} & -v^{(j)} - k_E^{(j)} & ... & 0 \\ 0 & v^{(j)} & ... & 0 \\ 0 & ... & ... & 0 \\ 0 & ... & ... & -v^{(j)} - k_E^{(j)} \end{bmatrix} \\
B^{(j)} &= \begin{bmatrix} v^{(j)} & 0 & \dots & 0 \\ k_E^{(j)} & k_E^{(j)} & \dots & k_E^{(j)} \end{bmatrix}^T, \quad C = \begin{bmatrix} 0 & 0 & 0 & ... & 1 \end{bmatrix}
\end{aligned}
\tag{4.1}
$$

where CL is used to refer the model of a heat exchange unit or connection and $j = 1, 2...N$ represents the jth CL unit. The input is $\boldsymbol{u}^{(j)} = [u_1^{(j)}\ T_{EXT}]^T$ where u_1 is the inlet temperature and T_{EXT} is the external temperature from the environment. The state vector $\boldsymbol{x}^{(j)} \in \mathbb{R}^n$ contains the temperatures x_i along the tube of the jth subsystem where $i = 1, 2...n$ represents the position of a flow element in the tube. The measured output of the system is the last state variable $x_n^{(j)}$, i.e. $y^{(j)} = x_n^{(j)}$.

Note that the input $u_1^{(j)}$ comes from the output of another CL unit which may happen to pass through a joining connection. Also, the output $y^{(j)}$ becomes the input of another CL unit which may happen to pass through a splitting connection. For those splitting and joining connections, Kirchoff law as in Eqs (B.4) and (B.6) is applied (see Appendix B.2).

Furthermore, the model will always be stable as long as k_E is positive because the eigenvalues are already stable shown by the negative sign of the diagonal entries of the A matrix (see Gershgorin Circle Theorem for details [137]).

4.2 Parameter fault modelling in heat exchange networks

The heaters and consumers of HENs could be connected by such transmitting elements (pipes) that can hardly be monitored e.g. due to the unfavourable spatial placements or excessive lengths. In these cases, it is important to estimate the internal states and operating conditions of these elements based only on measurements performed at their terminals.

A critical parameter in the elements of a heat exchange network is represented by the heat transfer coefficient $k_E^{(j)}$. Its decrease leads to the performance degradation of the HENs. The slackening of the heat transfer coefficient is treated as a fault event in the network. The fault is modelled as a multiplicative parameter uncertainty in the system:

$$
k_{Ef}^{(j)} = (1 - f_j)k_E^{(j)} \tag{4.2}
$$

where $k_{Ef}^{(j)}$ is the modified heat transfer parameter because of the fault f_j and $f_j \in [0, 1)$ is a piece-wise continuous fault signal in the jth subsystem with sparse changes.

As mentioned earlier, the fault diagnosis problem in this section has two different scenarios with two different parameter fault modellings. In the first scenario, the parameter fault happens along the tube in an element in the HENs (no simultaneous faults in many elements at the same time). Meanwhile, in the second scenario, the

parameter fault happens in a certain position along the length of the tube in a predetermined element in the HENs (no simultaneous faults in many positions at the same time). For the sake of convenience, the first scenario is named *scenario A*. On the other hand, the second scenario is named *scenario B*. Moreover, to differentiate it between different scenarios, the fault isolation term in *scenario B* is changed into fault localization.

Scenario A

The fault signal in Eq (4.2) can be incorporated into the state space model in Eq (4.1) as follows:

$$\dot{\boldsymbol{x}}^{(j)} = A^{(j)}\boldsymbol{x}^{(j)} + B^{(j)}\boldsymbol{u}^{(j)} + f_j\boldsymbol{h}^{(j)}(\boldsymbol{x}^{(j)}) \ , \ y^{(j)} = C\boldsymbol{x}^{(j)} \tag{4.3}$$

where the ith entry of $\boldsymbol{h} \in \mathbb{R}^n$ is $h_i^{(j)} = k_E^{(j)}(x_i^{(j)} - T_{EXT})$.

Special case: faulty model with splitting connection. Consider two interconnected CL type connections where the first (j) is before the split while the second (k) is one of the split elements after the split. The state vector of this system is $\boldsymbol{x}^{(j,k)} = [\boldsymbol{x}^{(j)T}\ \boldsymbol{x}^{(k)T}]^T \in \mathbb{R}^{2n}$. Assume that the fault can only happen either before or after the split so that it will not affect all of the states.

The fault diagnosis oriented model in Eq (4.3) in this case can be extended by adding a fault distribution matrix $F^{(j,k)}$ to the last term of the model:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(j,k)} &= A^{(j,k)}\boldsymbol{x}^{(j,k)} + B^{(j,k)}\boldsymbol{u}^{(j,k)} + F^{(j,k)}\boldsymbol{h}^{(j,k)}(\boldsymbol{x}^{(j,k)}) \\ y^{(j,k)} &= C^{(j,k)}\boldsymbol{x}^{(j,k)} \end{aligned} \tag{4.4}$$

where

$$\begin{aligned} F^{(j,k)} &= \begin{bmatrix} f_jI & 0 \\ 0 & 0 \end{bmatrix} \text{ if the fault happens before the split} \\ F^{(j,k)} &= \begin{bmatrix} 0 & 0 \\ 0 & f_kI \end{bmatrix} \text{ if the fault happens after the split} \end{aligned} \tag{4.5}$$

Here, $I \in \mathbb{R}^{n\times n}$ is the identity matrix and $0 \in \mathbb{R}^{n\times n}$ is the zero matrix.

Scenario B

By assuming that there are no simultaneous faults, parameter fault in Eq (4.2) can be incorporated into the state space model in Eq (4.1) only on the ith state as follows:

$$\dot{\boldsymbol{x}} = A_{f,j}\boldsymbol{x} + B_{f,j}\boldsymbol{u} \ , \ \boldsymbol{y} = C\boldsymbol{x}$$

$$A_{f,j} = \begin{bmatrix} -(v+k_E) & 0 & ... & ... & ... & 0 \\ v & -(v+k_E) & ... & ... & ... & 0 \\ 0 & v & ... & ... & ... & 0 \\ 0 & ... & ... & ... & ... & 0 \\ 0 & ... & v & -(v+k_{Ef}) & ... & 0 \\ 0 & ... & ... & ... & ... & 0 \\ 0 & ... & ... & ... & -(v+k_E) & 0 \\ 0 & ... & ... & ... & v & -(v+k_E) \end{bmatrix} \tag{4.6}$$

$$B_{f,j} = \begin{bmatrix} v & 0 & ... & 0 & ... & 0 \\ k_E & k_E & ... & k_{Ef} & ... & k_E \end{bmatrix}^T \ , \ C = \begin{bmatrix} 0 & 0 & 0 & ... & 1 \end{bmatrix}$$

The superscript $...^{(j)}$ in this *scenario B* is not written because it is considered to be predetermined or already specified.

This can be seen in Fig 4.1.

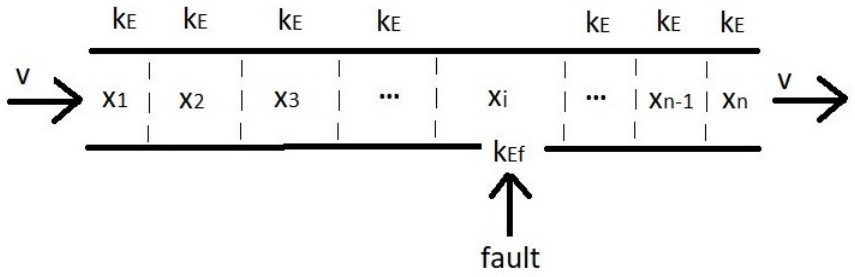

FIGURE 4.1: k_{Ef} fault position in the ith section of a tube

By substituting Eq (4.2) into Eq (4.6) and derive it more, we get:

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u} + F_j \ , \ y = C\boldsymbol{x}$$

$$A = \begin{bmatrix} -(v+k_E) & 0 & 0 & ... & ... & ... & 0 \\ v & -(v+k_E) & 0 & ... & ... & ... & 0 \\ 0 & v & -(v+k_E) & ... & ... & ... & 0 \\ 0 & ... & ... & ... & v & -(v+k_E) & 0 \\ 0 & ... & ... & ... & ... & v & -(v+k_E) \end{bmatrix} \tag{4.7}$$

$$B = \begin{bmatrix} v & 0 & ... & 0 \\ k_E & k_E & ... & k_E \end{bmatrix}^T \ , \ F_j = \begin{bmatrix} 0 & ... & f_j h(x_i) & ... & 0 \end{bmatrix}^T$$

where $h(x_i) = k_E(x_i - T_{EXT})$ and only the ith entry of $F_j \in \mathbb{R}^{n\times 1}$ has a non-zero value. A and B are the state and input matrices in the fault free case as in Eq (4.1).

4.3 Structural observability analysis for sensor placement in heat exchange networks

In a networked system, it is rare to install many sensors along the interconnections and at every subsystem's outputs because of either physical or financial reasons. Thus, sensor placement is another problem that needs to be considered for fault diagnosis purposes. This problem is related to the system's observability.

However, despite the observability defined by Kalman, the possible complexity that may arise in the network topology make a graph based approach intuitively preferable. To do this, a Signed Directed Graph (SDG) is drawn from the structural state space model of the related system [138, 139, 140]. In SDG, the vertices represent the inputs, outputs, and states, while the edges represent the relations between them. Therefore, an edge (p_i, p_j) from p_i vertex to p_j vertex exists if and only if there is a direct relation between them. Then, the observability analysis is done by checking the fulfilment of the following two conditions (details are in Theorem 14.4 in [138]):

1. There is at least one path from every state vertices to at least one of the outputs vertices.

2. There is at least one cycle family which touches every state vertices.

A *cycle family* means a set of vertices with disjoint cycles. The first condition is also called output reachability. It implies that each change in the states can be

detected at least in one of the output vertices. Meanwhile, the second condition is related to the structural rank (s-rank) of the observable pair of (A, C) in Eq (A.3) to ensure that it has a full rank. If both of those conditions are satisfied, then the system is called *structurally observable* or *s-observable*.

As SDG is a structural graph based approach, it is a parameter independent tool. Thus, in addition to being more intuitively preferable in terms of dealing with the network topology, it can also be applied to nonlinear systems.

Scenario A

In the CL type connection, Eq (4.1) shows that each state affects its successive state. The last state is measured by a sensor so that it is directly connected to the output vertex. The fault representing the change in $k_E^{(j)}$ parameter is influencing all of the states. Thus, we can make a *condensed graph* where all of the states are represented by just one vertex except the fault because we want to investigate whether that fault is observable or not concerning the sensor placement. Moreover, it implies that the first condition is already satisfied so we only need to check the fulfilment of the second condition.

As an example, consider three CL type connections CL_1, CL_2, and CL_3 which are joined into one CL type connection CL and then it is split into three CL type connections CL_a, CL_b, and CL_c as shown in Fig 4.2. The y vertices are representing the possibility of sensor placement in the joining and splitting connections.

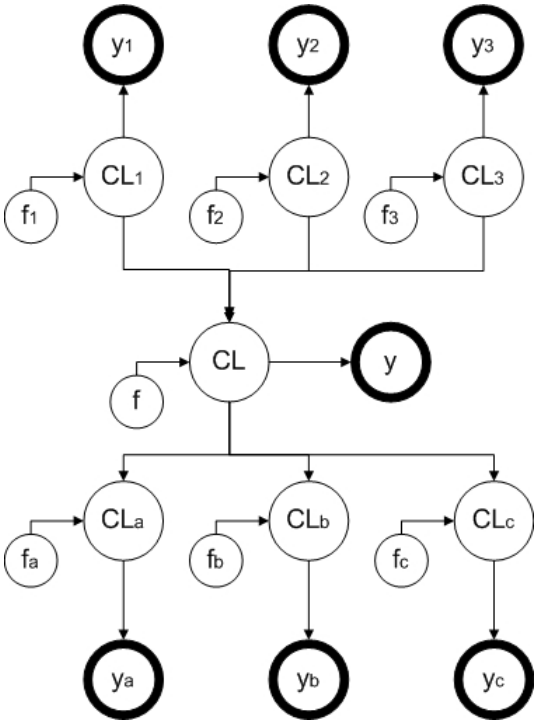

FIGURE 4.2: SDG of joining and splitting CL type connections

For *joining connection*, it is seen that (f_1, CL_1, CL, y) is not in a disjoint cycle family with (f_2, CL_2, CL, y) because they have two common vertices CL and y. This is also valid for the other cycles (f_3, CL_3, CL, y). Thus, with only one sensor put at the end of the CL type connection after the join, a single observer to isolate all the faults can not be constructed because the second observability condition is not fulfilled.

However, a *bank of observers* can be built by which each observer is constructed specifically just to detect a single specific fault. For each observer, we treat the output of the remaining CL type connections as an additional disturbance. The drawback of this approach is that a disturbance decoupling must be designed to compensate for the additional disturbances from the remaining CL type connections other than the one that we build a specific observer to estimate the fault happening there.

One other way to make the *joining connection* be s-observable for the whole fault vertices is *by adding a sensor at each CL type connection output in addition to the one at the joining connection output*. With this treatment, it will have exactly one cycle family for each fault vertex in the interconnected subsystems which are (f, CL, y), (f_1, CL_1, y_1), (f_2, CL_2, y_2), and (f_3, CL_3, y_3).

For *splitting connection*, it can be seen that each output vertex has two cycle families. One cycle touches the fault and state vertices of the CL type connection before it is split, while the other one touches the fault and states vertices of its related CL type connection after the split. Thus, *with just one sensor put at the output of a specific CL type connection after the split, an observer can always be constructed to estimate a fault happening at the CL type connection before the split*. Moreover, based on that same sensor, another observer to estimate a specific fault that occurs in the related CL type connection after a split can also be constructed. It should also be remarked that, for this splitting connection, only some of the states are affected by the fault depending on whether it happened before or after the splits as seen in Eqs (4.4) and (4.5).

Scenario B

From Eq (4.6), it is seen that each state affects its consecutive state. To make it simpler, but without loss of generalization, assume that the system consists of 5 states ($n = 5$) and the third state is affected by a parameter fault. Hence, we can draw the SDG as shown in Fig 4.3. The output vertices y_1, y_2, y_3, y_4, and y_5 represent the possibility of sensor placement in the related system. Meanwhile, the f vertex represents the parameter fault.

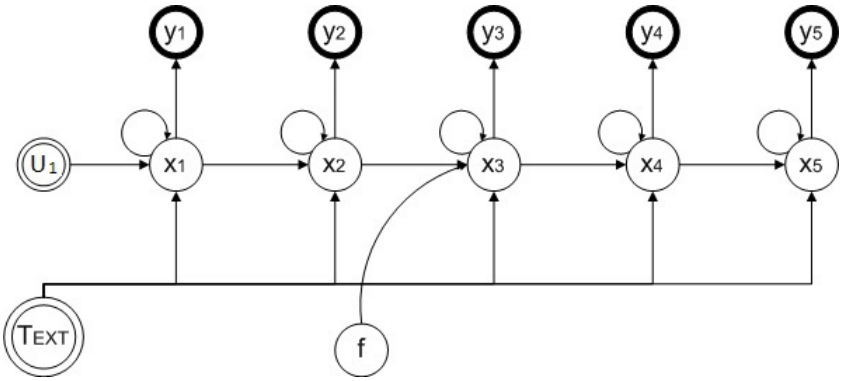

FIGURE 4.3: SDG for parameter fault in a certain location along the length of a tube

If minimizing the number of sensors is preferred, it is seen that y_5 is the best position to place the sensor. From that position, every state vertices have a path to the output vertex and there is one cycle family which touches all of the state vertices (y_5, x_1, x_2, x_3, x_4, x_5). Thus, both of the s-observability conditions are fulfilled. This shows that using just one sensor at the end of the connection to measure the last state is enough to estimate all of the system's states. Moreover, it is also seen that

there exists a path from f to y_5 which implies that we can use that same sensor to estimate the parameter fault in the considered system. This will always hold even if the fault affects other states.

4.4 Adaptive observer design for parameter fault estimation in heat exchange networks

A critical issue in the formulated fault diagnosis problem is that there are unmeasurable states in the system. [141] has developed a fault observer based method where a nonlinear fault distribution function depends not only on the inputs and outputs but also on estimated states (for the related article's main results, see Appendix D). By taking the basic idea from this previous research, we modified it to suit our case.

Scenario A

The dynamics of the CL type connections with fault is as shown in the Eq (4.3). Define the estimation errors as:

$$\begin{aligned} \boldsymbol{e}_x^{(j)} &= \boldsymbol{x}^{(j)} - \hat{\boldsymbol{x}}^{(j)} \ , \ e_y^{(j)} = y^{(j)} - \hat{y}^{(j)} = C\boldsymbol{e}_x^{(j)} \\ e_f^{(j)} &= f_j - \hat{f}_j \end{aligned} \tag{4.8}$$

where $\hat{f}_j$ is the estimated fault.

Consider an observer for faulty case as follows:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}}^{(j)} &= A^{(j)}\hat{\boldsymbol{x}}^{(j)} + B^{(j)}\boldsymbol{u}^{(j)} + \hat{f}_j\boldsymbol{h}^{(j)}(\hat{\boldsymbol{x}}^{(j)}) + K_x^{(j)}e_y^{(j)} \\ \hat{\boldsymbol{y}}^{(j)} &= C\hat{\boldsymbol{x}}^{(j)} \end{aligned} \tag{4.9}$$

along with the following parameter adaptation equation:

$$\dot{\hat{f}}_j := \begin{cases} K_f^{(j)}\boldsymbol{h}^{(j)}(\hat{\boldsymbol{x}}^{(j)})e_y^{(j)}, \text{ if } \|\hat{f}_j\| \leq \sigma \\ 0, \text{ otherwise} \end{cases} \tag{4.10}$$

Here, $K_x^{(j)}, K_f^{(j)T} \in \mathbb{R}^n$ are the observer and adaptation gain vectors. σ is a design constant to guarantee the stability and robustness of the adaptation law so that e_f is always bounded [142].

By assuming piece-wise constant fault signal, we can further derive the state and fault estimation error to get:

$$\begin{aligned} &\begin{bmatrix} \dot{\boldsymbol{e}}_x^{(j)} \\ \dot{e}_f^{(j)} \end{bmatrix} = A_e \begin{bmatrix} \boldsymbol{e}_x^{(j)} \\ e_f^{(j)} \end{bmatrix} \\ &\text{where} \\ &A_e = \begin{bmatrix} (A^{(j)} - K_x^{(j)}C + f_j k_E^{(j)} I & \boldsymbol{h}^{(j)}(\hat{\boldsymbol{x}}^{(j)}) \\ -K_f^{(j)}\boldsymbol{h}^{(j)}(\hat{\boldsymbol{x}}^{(j)})C & 0 \end{bmatrix} \end{aligned} \tag{4.11}$$

According to Gershgorin Circle Theorem, if $A^{(j)} - K_x^{(j)}C$ is stable, then $A^{(j)} - K_x^{(j)}C + f_j k_E^{(j)} I$ is also stable as long as $k_E^{(j)} f_j \geq 0$ has small norm. This is most likely fulfilled because $f_j \in [0, 1)$ (see Eq (4.2)). Furthermore, the estimation error $\boldsymbol{e}_x^{(j)}$ and

$e_f^{(j)}$ will converge to zero if A_e is stable for all $\hat{\boldsymbol{x}}^{(j)}$ and f_j. This stability can be checked online during the adaptation process. Thus, an observer described by Eq (4.9) along with its adaptation as in Eq (4.10) can be used to estimate the fault and it is called a *fault estimator*.

Special case, fault detector and fault estimator for splitting connection. As shown earlier, for splitting connection, we can put the sensors only at the end of each element after the split (see subsection 4.3). With this configuration, there will be one specific sensor that is used as an input to two observers for fault estimation either before or after the split. However, by *assuming no simultaneous faults*, the difference in fault distribution matrix shown in Eq (4.5) causes the need of a *fault detector* to determine where the fault is happened before fault estimation can be carried out.

Consider a bank of linear observers:

$$FD_k := \begin{cases} \dot{\hat{\boldsymbol{x}}}^{(j,k)} & = A^{(j,k)}\hat{\boldsymbol{x}}^{(j,k)} + B^{(j,k)}\boldsymbol{u}^{(j,k)} + K_x e_y^{(j,k)} \\ \hat{y}^{(j,k)} & = C^{(j,k)}\hat{\boldsymbol{x}}^{(j,k)} \end{cases} \tag{4.12}$$

where $e_y^{(j,k)} = y^{(j,k)} - \hat{y}^{(j,k)}$ and FD_k is a fault detector based on the measurement at the end of the kth split elements (elements after the split).

If K_x is chosen such that $A^{(j,k)} - K_x C^{(j,k)}$ is stable, then $\lim_{t\to\infty} \boldsymbol{e_x^{(j,k)}} = 0$ and $\lim_{t\to\infty} e_y^{(j,k)} = 0$ for non-faulty condition. It is also easily inferred that when a fault occurs before the split, it will propagate to all of the split elements. Thus, the following algorithm can be used to determine where the fault is happening in the splitting connection (fault isolation logic):

$$e_y^{(j,k)} \begin{cases} = 0 \;\forall\; FD_k, & \text{no fault} \\ \neq 0 \;\forall\; FD_k, & \text{a fault has occured before the split} \\ \neq 0 \;\exists\; FD_k, & \text{a fault has occured at the } k\text{th split element} \end{cases} \tag{4.13}$$

Then, after a fault has been detected, a specific nonlinear observer is activated to estimate the related fault. For this purpose, a bank of nonlinear observers is constructed as follows:

$$\begin{aligned} FE_j := \begin{cases} \dot{\hat{\boldsymbol{x}}}^{(j,k)} & = A^{(j,k)}\hat{\boldsymbol{x}}^{(j,k)} + B^{(j,k)}\boldsymbol{u}^{(j,k)} \\ & \quad + \hat{F}_j^{(j,k)}\boldsymbol{h}^{(j,k)}(\hat{\boldsymbol{x}}^{(j,k)}) + K_x e_y^{(j,k)} \\ y^{(j,k)} & = C^{(j,k)}\hat{\boldsymbol{x}}^{(j,k)} \\ \hat{F}_j^{(j,k)} & = \begin{bmatrix} \hat{f}_j I & O \\ O & O \end{bmatrix} \end{cases} \\ FE_k := \begin{cases} \dot{\hat{\boldsymbol{x}}}^{(j,k)} & = A^{(j,k)}\hat{\boldsymbol{x}}^{(j,k)} + B^{(j,k)}\boldsymbol{u}^{(j,k)} \\ & \quad + \hat{F}_k^{(j,k)}\boldsymbol{h}^{(j,k)}(\hat{\boldsymbol{x}}^{(j,k)}) + K_x e_y^{(j,k)} \\ y^{(j,k)} & = C^{(j,k)}\hat{\boldsymbol{x}}^{(j,k)} \\ \hat{F}_k^{(j,k)} & = \begin{bmatrix} O & O \\ O & \hat{f}_k I \end{bmatrix} \end{cases} \end{aligned} \tag{4.14}$$

with the following adaptation equations:

$$\dot{\hat{f}}_j = \dot{\hat{f}}_k := \begin{cases} K_f^{(j,k)}\boldsymbol{h}^{(j,k)}(\hat{\boldsymbol{x}}^{(j,k)})e_y^{(j,k)}, \text{ if } (\|\hat{f}_j\| \leq \sigma) \vee (\|\hat{f}_k\| \leq \sigma) \\ 0, \text{ otherwise} \end{cases} \tag{4.15}$$

Here, FE_j is a fault estimator based on the measurement at the end of any of the split elements to estimate the fault that happened before the split, and FE_k is a fault estimator based on the measurement at the end of the kth split elements to estimate the fault happened there.

In the same manner as before in Eq (4.11), with $\|F^{(j,k)}\| \in [0, 1)$, the same conclusion can be drawn for either FE_j or FE_k as long as the fault and k_E have small norm. Thus, a bank of linear observers described by Eq (4.12) along with the algorithm described by Eq (4.13) can be used as fault detectors. Meanwhile, a bank of nonlinear observers described by Eq (4.14) along with its adaptation described by Eq (4.15) can be used as fault estimators.

Scenario B

By using the same parameter adaptation equation as in Eq (4.10), a bank of nonlinear observers is constructed by which each of them is specifically designed to detect a fault in a section (a flow element position) of the tube. For example, observer 1 is designed to estimate a fault that happened in the 1st section of the tube, observer 2 is designed to estimate a fault that happened in the 2nd section of the tube, and so on.

To make $\lim_{t\to\infty} e_f = 0$ for the ith section ($i = 1 \dots n$), the following fault estimator observer can be constructed:

$$FE_i := \begin{cases} \dot{\hat{x}} & = A\hat{x} + B\boldsymbol{u} + \hat{F}_j + K_x e_y \\ y & = C\hat{x} \\ \hat{F}_j & = \begin{bmatrix} 0 & \dots & \hat{f}_j h(\hat{x}_i) & \dots & 0 \end{bmatrix}^T \end{cases} \tag{4.16}$$

Here, FE_i is an observer specifically designed to estimate a fault that happened in the ith section in the tube.

The diagram of this bank of observers is shown in Fig 4.4.

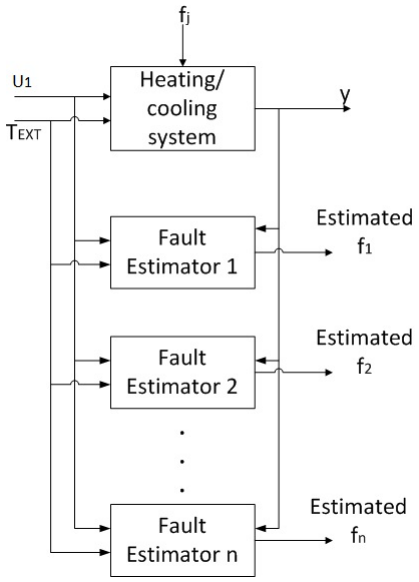

FIGURE 4.4: A bank of observers for fault estimation in a tube

4.5 Sensitivity analysis for parameter fault localization in heat exchange networks

This subsection is dedicated to *scenario B*.

Using a bank of observers described by Eq (4.16), it is expected that the fault can also be localized in addition to be estimated. To investigate it, by assuming zero initial conditions, we took the Laplace transform of Eq (4.6) so that we get:

$$\begin{aligned} Y(s) = X_n(s) = & \frac{v^n}{[s+(v+k_E)]^{n-1}[s+(v+k_{Ef})]} U_1(s) \\ & + \sum_{k=1}^{n-i} \frac{v^{k-1}}{[s+(v+k_E)]^k} k_E T_{EXT}(s) \\ & + \frac{v^{n-i}}{[s+(v+k_E)]^{n-i}[s+(v+k_{Ef})]} k_{Ef} T_{EXT}(s) \\ & + \sum_{k=1}^{i-1} \frac{v^{n-i+k}}{[s+(v+k_E)]^{n-i+k}[s+(v+k_{Ef})]} k_E T_{EXT}(s) \end{aligned} \tag{4.17}$$

where $Y(s)$, $X_n(s)$, $U_1(s)$, and $T_{EXT}(s)$ are the Laplace transform of $y(t)$, $x_n(t)$, $u_1(t)$, and $T_{EXT}(t)$, respectively.

From Eq (4.17), it is seen that the transfer function from $U_1(s)$ to $Y(s)$ does not contain the i variable which represents the ith section in the tube where the fault occurred. Thus, even if we manipulate or tweak the input u_1 with a signal which contains a wide spectrum of frequency (e.g. inserting a white noise signal, a saw-tooth signal, or a Pseudo Random Binary Signal (PRBS)), the information about the fault position can not be obtained.

On the other hand, the transfer function from $T_{EXT}(s)$ to $Y(s)$ contains the i variable that looks promising to use so that the fault position can be obtained. However, it is impractical to manipulate or tweak the input T_{EXT} because it represents the external temperature which comes from the outside environment.

Besides the transfer function as in Eq (4.17), we can also use the steady state value of the plant's output with respect to the position variation of the fault.

In Laplace transform, there is a property about Final Value Theorem (FVT) as follows:

$$\lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s) \tag{4.18}$$

where $F(s)$ is the Laplace transform of $f(t)$.

By assuming that $u_1(t)$ and $T_{EXT}(t)$ are step function from 0 to U_1 and from 0 to T_{EXT} respectively so that $U_1(s) = \frac{U_1}{s}$ and $T_{EXT}(s) = \frac{T_{EXT}}{s}$, we can use FVT to get a steady state value of $y(t) = x_n(t)$ in Eq (4.17) as follows:

$$\begin{aligned} y_{ss} = & \frac{v^n}{(v+k_E)^{n-1}(v+k_{Ef})} U_1 + \sum_{k=1}^{n-i} \frac{v^{k-1}}{(v+k_E)^k} k_E T_{EXT} \\ & + \frac{v^{n-i}}{(v+k_E)^{n-i}(v+k_{Ef})} k_{Ef} T_{EXT} \\ & + \sum_{k=1}^{i-1} \frac{v^{n-i+k}}{(v+k_E)^{n-i+k}(v+k_{Ef})} k_E T_{EXT} \end{aligned} \tag{4.19}$$

where y_{ss} is the steady state value of $y(t)$ given that the input $u_1(t)$ and $T_{EXT}(t)$ are step function.

Now, consider two faults occurred in two different positions i_1 and i_2 where $i_2 > i_1$. Let the steady state values in the presence of these faults are y_{ss1} and y_{ss2} respectively. If we compute the difference between these two steady state values, we get:

$$\begin{aligned} y_{ss2} - y_{ss1} = & - \sum_{k=n-i_2+1}^{n-i_1} \frac{v^{k-1}}{(v+k_E)^k} k_E T_{EXT} \\ & + T_{EXT} \left[\left(\frac{(v+k_E)^{i_2}}{v^{i_2}} - \frac{(v+k_E)^{i_1}}{v^{i_1}} \right) \frac{v^n}{(v+k_E)^n} \right] \\ & \left(\frac{k_{Ef}}{(v+k_{Ef})} + \frac{k_E}{(v+k_{Ef})} \sum_{k=i_1}^{i_2-1} \frac{v^k}{(v+k_E)^k} \right) \end{aligned} \tag{4.20}$$

Note that $\frac{v}{v+k_E} \in (0,1)$. Eq (4.20) shows that if $k_E > 0$ is sufficiently large compared to v and n is also large, then the effect of the fault can hardly be distinguished.

4.6 Simulation results

To verify and validate the proposed fault diagnosis method, some simulations is done using MATLAB/Simulink.

Scenario A

In this scenario, a simulation model of a HEN connecting a heater to two consumers is used. The energy output of a heater is distributed to the consumers via a hot pipe (H) which is then split into hot pipe 1 (H1) and hot pipe 2 (H2). The fluid from the consumers is fed back to the heater via cold pipe 1 (C1) and cold pipe 2 (C2) which are then joined into a cold pipe (C). It is assumed that the dynamics of each pipe can be represented as CL type connection with $n = 5$, $v = v_H = v_C = 2$, and $k_E = k_H = k_{H1} = k_{H2} = k_C = k_{C1} = k_{C2} = 1$. The splitting connection from H to H1 and H2 splits the mass flow equally (0.5 : 0.5). The diagram of this case study is shown in Fig 4.5. In this figure, the fault diagnostic blocks contain either a fault estimator for joining connection, or both a fault detector and a fault estimator for splitting connection.

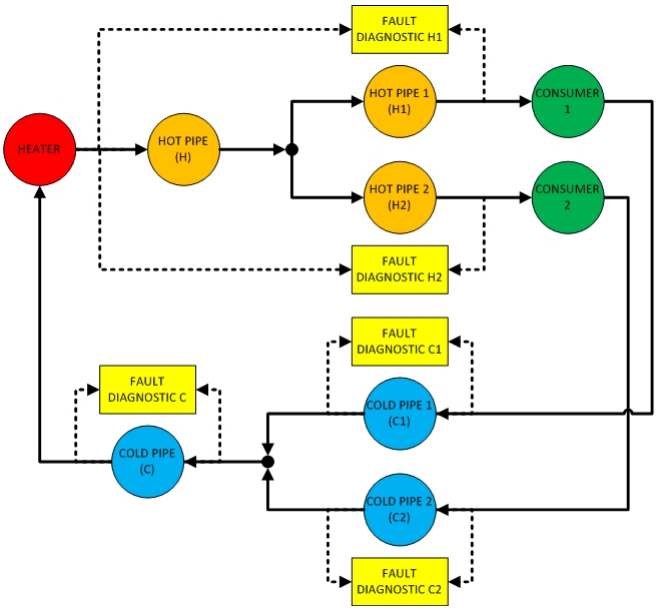

FIGURE 4.5: Heat exchange networks case study

To be able to detect and estimate the fault in each pipe, the sensor placement is done based on the previous structural observability analysis. For splitting connection in the hot pipe, it is enough to put the sensors at H for input measurement and at the end of H1 and H2 for output measurements. Meanwhile, for joining connection in the cold pipe, we use the multiple sensors approach. Thus, the sensors are put at the beginning of C1, C2, and C for input measurements and the end of C1, C2, and C for output measurements. The fault estimator for each pipe is constructed using those measurements. For hot pipes, fault detectors are also constructed. The fault estimator for hot pipe H is a special case because it can be constructed based on either the sensor measurement at the end of H1 or H2.

Eq (4.12) is used to construct the fault detectors for the splitting connection. As fault isolation logic, Eq (4.13) is applied with $k = 1, 2$. Meanwhile, as we used multiple sensors approach for the joining connection, only fault estimators are constructed to estimate specific faults for each cold pipe. To construct the fault estimators, we use Eq (4.14) for splitting connection and Eq (4.9) for joining connection.

After the fault detectors and fault estimators are constructed, the related measurements are fed into each of those observers. Then, when a fault is happening in the splitting connection, the fault isolation logic will activate a specific fault estimator based on the error signals from the fault detectors. It should be remarked that this configuration works on the assumption that *no simultaneous faults happen in the splitting connection*. To be able to detect and estimate simultaneous faults in the splitting connection, we must use the same multiple sensors approach as in the joining connection.

In this simulation, the fault detector gain K_x is chosen using the pole placement method while the fault estimator gain K_f is chosen so that the settling time is small enough without oscillation. Meanwhile, the chosen design constant is $\sigma = 1.5$ to allow some small overshoot in the observer's estimation process when the fault is near its maximum value.

First, a fault at the 15th second with an amplitude of 0.5 is introduced into the system. Fig 4.6 shows the error signals from the fault detector H1 and H2 when this fault is happening in the hot pipe H before the split. It can be seen that both the error signals e_{H1} and e_{H2} have the same non-zero value indicating that a fault has

occurred at the hot pipe H. The estimated fault from the fault estimator for this hot pipe H is shown in Fig 4.7 which displays that the fault is successfully estimated. Furthermore, this fault estimator can also estimate the unmeasurable states which are shown in Fig 4.8.

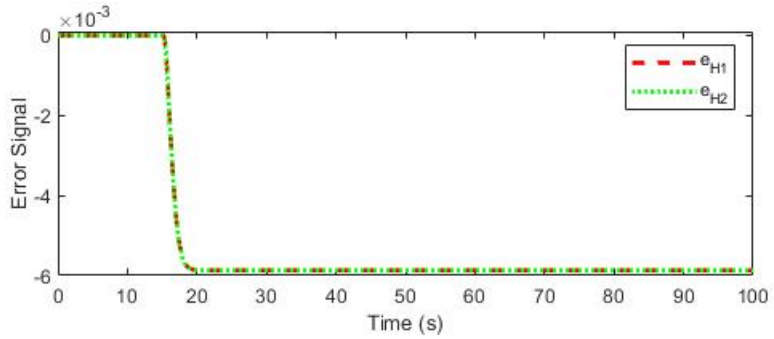

FIGURE 4.6: Error signals of fault detectors, fault at H

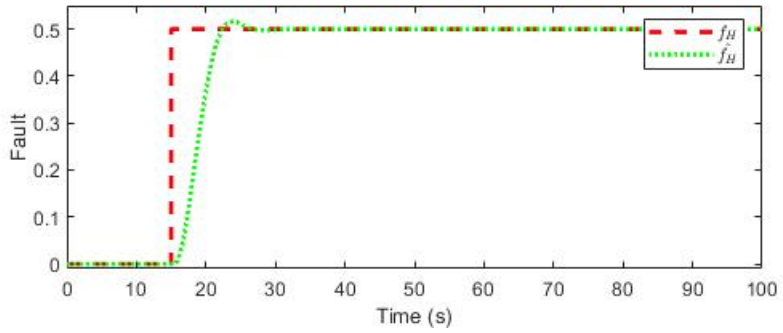

FIGURE 4.7: Fault estimation, fault at H

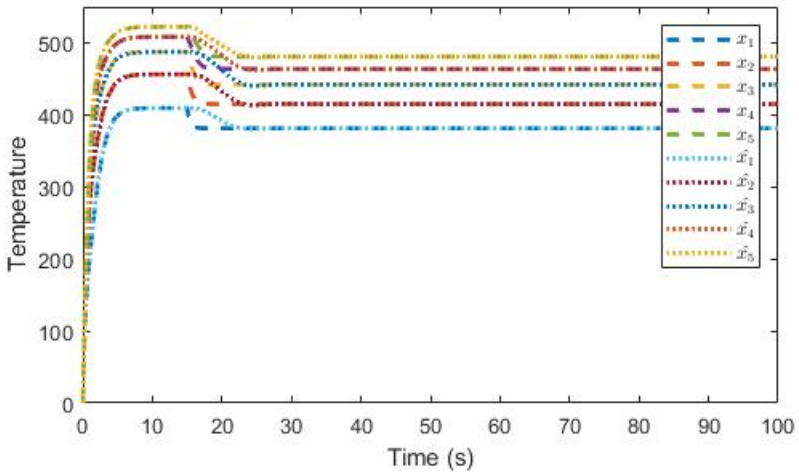

FIGURE 4.8: State estimation of fault estimator, fault at H

In the second simulation, the fault is simulated to be happening at the hot pipe H2 after the split. Fig 4.9 shows the error signals from the fault detector H1 and H2. It is seen that only the error signal e_{H2} has a non-zero value indicating that a fault is happening in the hot pipe H2. The estimated fault from the fault estimator for

hot pipe H2 is shown in Fig 4.10 which reveals that it also successfully estimated the fault. Meanwhile, the state estimation from this fault estimator is shown in Fig 4.11.

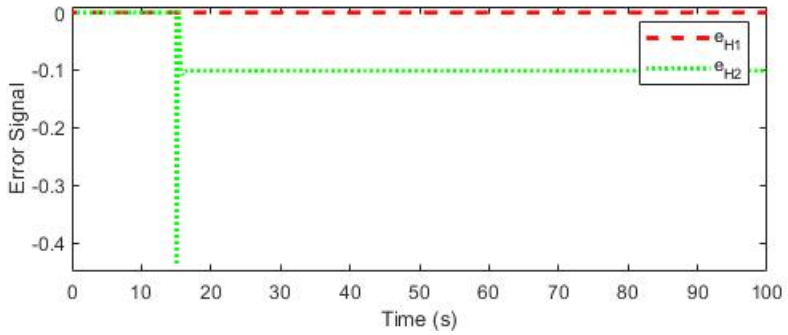

FIGURE 4.9: Error signals of fault detectors, fault at H2

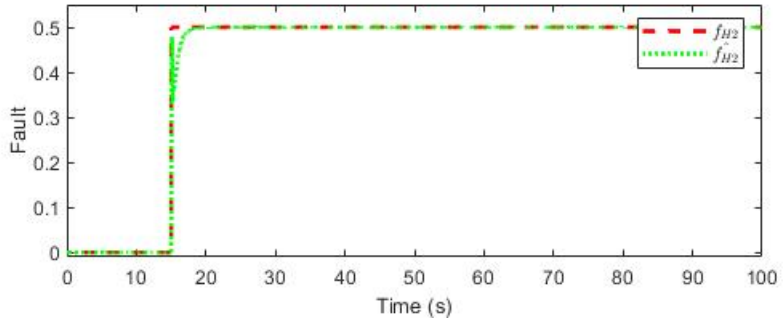

FIGURE 4.10: Fault estimation, fault at H2

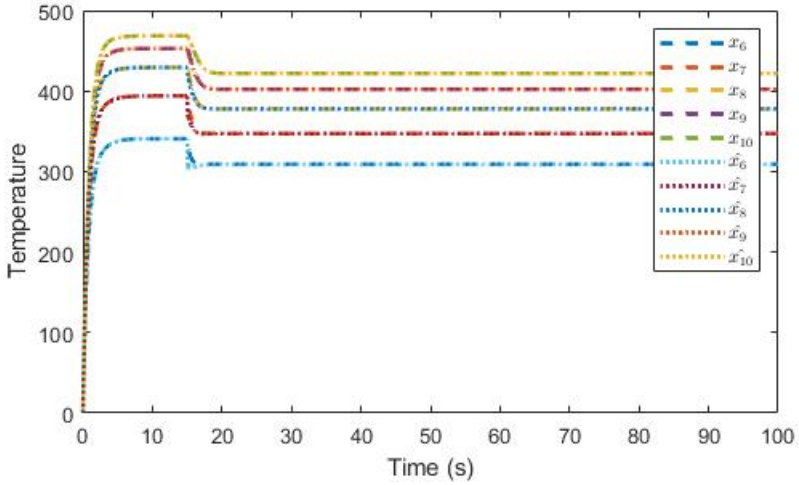

FIGURE 4.11: State estimation of fault estimator, fault at H2

Scenario B

In this simulation, it is assumed that the dynamics can efficiently be represented using 5 states/sections ($n = 5$). The other parameters are $v = 3$ and $k_E = 2$. The inputs are $u_1 = 600K$ and $T_{EXT} = 500K$. All of those numbers are chosen for the

sake of convenience. In the real world, they have many various values based on the units and equipment/medium type.

For the observers, the fault detector gains K_x is chosen using the pole placement method while the fault estimator gain K_f is chosen to achieve a fast but sufficiently damped observer response. Meanwhile, the chosen design constant is $\sigma = 1.5$ to allow some small overshoot in the observer's estimation process when the fault is near its maximum value.

There are 3 scenarios of fault occurrence in different sections that are simulated. For every scenario, a fault is introduced into the system at the 20th second. Fig 4.12 shows the results when a fault with an amplitude of 0.4 ($f = 0.4$) happened in the 1st section of the tube ($i = 1$). Fig 4.13 shows the results when a fault with an amplitude of 0.2 ($f = 0.2$) happened in the 3rd section of the tube ($i = 3$). Lastly, Fig 4.14 shows the results when a fault with an amplitude of 0.6 ($f = 0.6$) happened in the 5th section of the tube ($i = 5$). In all of those figures, the real fault is plotted using dotted lines while the estimated fault from each observer is plotted using dashed lines.

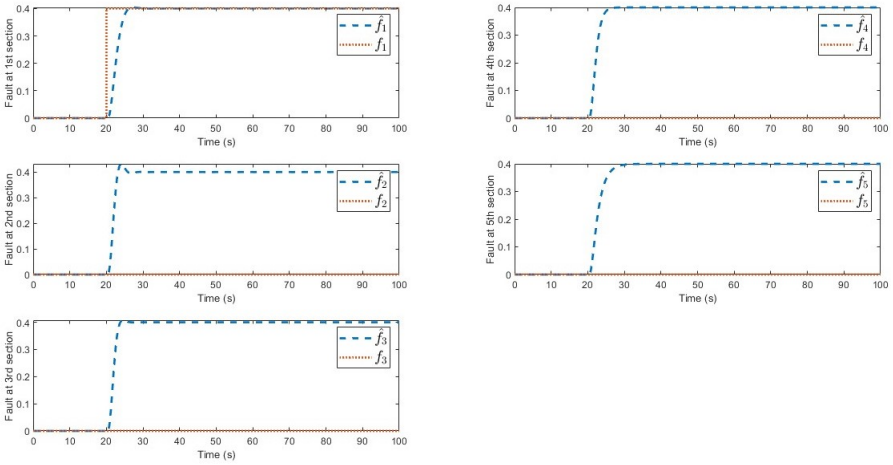

FIGURE 4.12: f and $\hat{f}$ for $f = 0.4$ in the 1st section of the connection

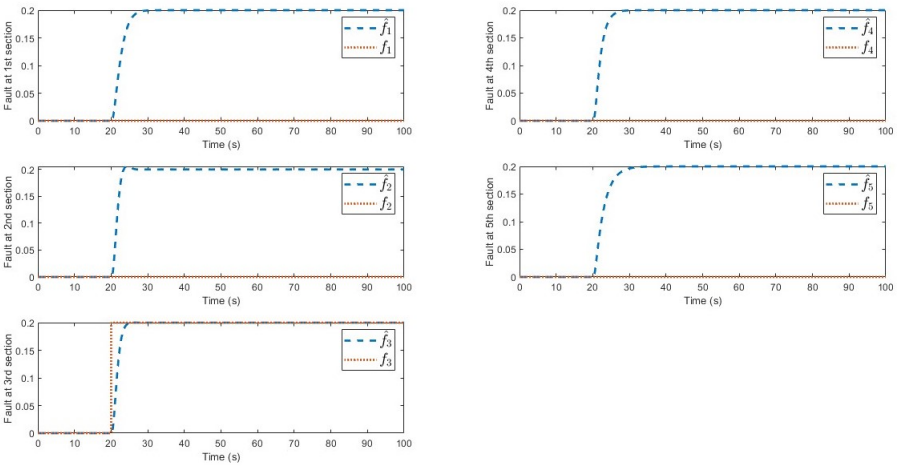

FIGURE 4.13: f and $\hat{f}$ for $f = 0.2$ in the 3rd section of the connection

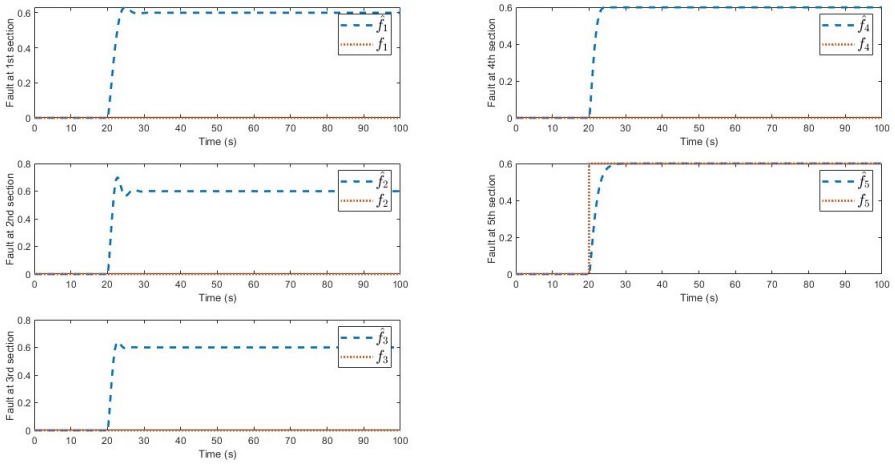

FIGURE 4.14: f and $\hat{f}$ for $f = 0.6$ in the 5th section of the connection

In the figures, it can be seen that all the faults in every scenario are correctly estimated by the observers. However, it is also shown that every observer in each scenario produced the same fault estimation results. Thus, even though the fault estimation is successful, it can not detect in which section the fault has occurred in the system.

Fault localization results. In the previous section (see subsection 4.5), we have shown that the fault position can only be extracted by tweaking the input T_{EXT} which is impractical to do. Moreover, here, we will also show that it is still difficult to get this information even if we can tweak it.

Using Eq (4.19), the steady state value y_{ss} of a system with 5 states ($n = 5$) is calculated for each value of $i = 1 \dots 5$ representing the position of the fault for a fixed input variable of $u_1 = 600$ and $T_{EXT} = 500$ with several different parameters values of v, k_E, and k_{Ef}. The results are shown in Table 4.1. It is seen that the steady state value of the plant's output is always the same wherever the fault is happening.

TABLE 4.1: Steady state value y_{ss} for each fault's position i with several different parameter values of v, k_E, and k_{Ef}

parameters values	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
$v = 2, k_E = 1, k_{Ef} = 0.5$	515.8	515.8	515.8	515.8	515.8
$v = 3, k_E = 2, k_{Ef} = 1.5$	508.6	508.6	508.6	508.6	508.6
$v = 8, k_E = 3, k_{Ef} = 0.75$	525.6	525.6	525.6	525.6	525.6
$v = 10, k_E = 4, k_{Ef} = 2$	521.7	521.7	521.7	521.7	521.7
$v = 15, k_E = 5, k_{Ef} = 3.75$	525.3	525.3	525.3	525.3	525.3

Fig 4.15 shows the Bode plot from input 1 (u_1) and input 2 (T_{EXT}) to output y for $i = 1, 2, 3, 4, 5$ (f_1, f_2, f_3, f_4, f_5) which looks almost identical to each other for each value of i. This reinforces the previous results that the position of the fault can hardly be isolated using only a measurement at the end of the connection (last state variable x_n) even if we excite the system in the entire reasonable frequency range.

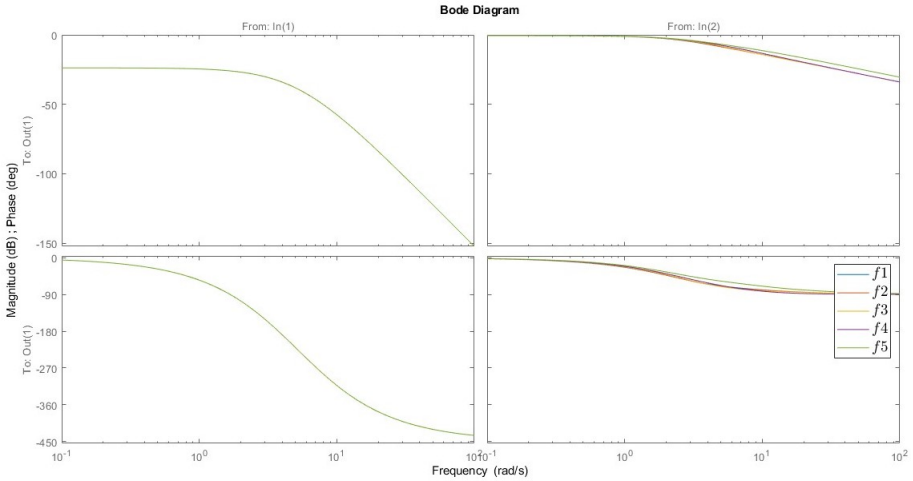

FIGURE 4.15: Bode diagram for fault in each section of the connection

4.7 Summary and discussion

In this chapter, model based methods are proposed to handle parameter fault diagnosis problems in HENs by considering two different scenarios. In the first scenario, the considered fault is the change in the heat transfer coefficient k_E along the length of the tube in an element in the HENs. Thus, the goal is to detect and isolate which tubes experience the parameter fault in the HENs and also to estimate the fault. Meanwhile, in the second scenario, the goal is to detect, isolate, and estimate the parameter fault that happens only at a certain location along the length of a specific tube.

As the network contains realistic connections in the form of joining and splitting elements, the investigation of sensor placement problems to identify every possible fault around the branching elements is needed. For that, an analysis of observability conditions using SDG is done. It is found that, in the case of splitting connections, it is enough to put the sensors only at the end of the elements after the split to detect and estimate the fault that occurred either before or after the split. However, in the case of joining connections, a sensor at the end of the joining elements before the connection is necessary. Moreover, in the case of a specific tube, the SDG based analysis also shows that using only one sensor at the end of that tube is enough to estimate the states when there is no fault and the fault when there is a single parameter fault at a certain location along the tube's length.

With the change of heat transfer coefficient being the model parameter, the fault's presence in HENs yields bilinear fault-input terms into the model. Thus, a linear observer is not suitable to solve the parameter fault estimation problem. However, a bank of linear observers can still be used and is especially necessary to detect and isolate the parameter fault in splitting connections because, there, the same sensor is used when the fault happens either before or after the split. To estimate the parameter fault, a bank of nonlinear observers is proposed that is based on the parameter adaptation equation.

To generalize the implementation of the proposed method, some properties must be fulfilled in the networked LTI systems which are: 1. The systems are positive. This positivity in LTI systems will always result in the same structure of the A matrix in the system's state space model $\dot{\boldsymbol{x}} = A\boldsymbol{x}$ where the off-diagonal elements are non-negative, i.e., a Metzler matrix [143]. Moreover, as the diagonal elements are

negative, this implies that the systems are also stable [144, 145, 146] (see Appendix A.2), 2. The network elements can be modelled as a 2ISO state space model as shown in Eq (4.1), 3. The fault can be modelled as a multiplicative parameter fault as shown in Eq (4.2), 4. The mass and/or energy conservation applies in the network's splitting and joining connections (see Appendix B.2), and 4. The faulty parameters affect all of the system states and the system's second input as shown in Eq (4.3).

In the case of a single parameter fault in a certain location in a specific tube, even though the proposed nonlinear observer can estimate it, fault isolation can not be done. It is because, by using sensitivity analysis on this specific case, it is found that the fault localization depends on the values of the environmental temperature which is not a practically changeable input variable. Moreover, even if we can tweak this impractical input variable, a steady state value calculation and frequency response analysis using the Bode plot show that the effect of different fault positions is hard to distinguish.

Here, the proposed method can be generalized to networked LTI systems where the first four properties are the same as in the previous case. The different property is only on the fifth by which the faulty parameters now affect only one of the system states and the system's second input as shown in Eq (4.7).

By proposing a model based method, this thesis provides an alternative to the currently popular data driven approaches in handling parameter fault diagnosis in HEN that have been carried out by many people before [77, 78, 79]. Furthermore, compared to much other works such as [74, 75, 76], the proposed method takes into account the sensor placement problem in correlation with the presence of joining and splitting connections. As a result, the method ensures the use of the minimum number of sensors possible. However, this leads to the existence of unmeasurable states which, combined with the parameter fault, induces a nonlinear fault distribution in the system. Thus, the nonlinear observer is applied for fault estimation purposes that can handle unmeasurable states.

Chapter 5

Input fault diagnosis in networks of heat exchangers

The FDI problem in industrial networks, such as heat exchange networks, presents special challenges. For the sake of energy and/or material efficiency, such networks usually contain loops. Therefore, fault diagnosis is harder to perform as the effect of the fault is propagated back to the subsystem where it first occurred [147, 148, 149, 104, 150]. Recent research has approached the fault isolation problem in such a loop using an improved deep neural network [82].

Another difficulty in the complex networked industrial systems related to FDI is the sparsity of sensors. Because of financial reasons, it is rare to install a multitude of sensors along the connections of the interconnected subsystems in the network. Commonly, they are placed at the end of a connection or at some of the subsystem's outputs. This motivates the development of an FDI method that groups network elements for fault identification [151, 152].

Furthermore, the interconnections that define the network topology make a networked dynamic system complex even if one considers simple elements (subsystems) in the network. Therefore, a practically important subclass of heat exchange systems is considered in this work where the elements are linear heat exchange systems connected by linear (possibly dynamic) connections.

Despite being a special case, it is a very important subclass because it can describe the dynamic behaviour of processes that serve our daily basic needs, e.g., domestic heating/cooling systems. Even some cellular processes belong to a similar linear subclass [124]. The fault that occurred in one of the subsystems is considered to be generated by an external source that can be treated as an extra input in a subsystem. Physically, this kind of fault represents some leakage phenomenon that is commonly found inside the networked linear heat exchange systems [153].

A model-based approach is proposed in this research to handle the fault diagnosis problem in a network of linear heat exchange systems, which may contain cycles/loops. The proposed approach does not need high computational costs. The burden of high computational power is common for fault diagnosis in a network [117].

The research questions that we are going to tackle are as follows:

1. In a heat exchange network with loops, how can a fault be identified in a subsystem or group of subsystems regardless of the fault effect propagation through the network loops and branches?

2. Which sensor measurements are necessary to isolate a fault in a subsystem of a heat exchange network (or group of subsystems), and how can measurement noise be handled during fault diagnosis?

3. If fault isolation can be conducted, how can the magnitude of the fault be estimated?

The author's related publication for this study is [A5].
The following modelling assumptions are used in this chapter:

A1 Only the transport of a single conserved extensive quantity (in this study, it is energy) is considered in the process systems. Thus, we have either energy-transport or mass-transport systems. Heat exchanger networks and domestic heating/cooling systems belong to the linear energy-transport class.

A2 Only linear convection and transfer is considered without any linear source.

A3 Constant overall mass and constant physicochemical parameters (such as density, specific heat, heat transfer coefficient, and convective flow rate) are assumed.

A4 One inlet and one outlet flow are considered where the inputs of the systems are the intensive variable (temperature or concentration) at the inlet and that of the balance volume with which transfer is assumed. Meanwhile, the output is the intensive variable (temperature or concentration) at the outlet.

Remark: In this chapter, the used subsystem's model is the same as in chapter 4, specifically Eq (4.1) in Section 4.1. However, for the sake of convenience to differentiate the discussion here from the one in Chapter 4, the input in the subsystem's model in Eq (4.1) is changed into $\boldsymbol{u}_X^{(j)} = [u^{(j)}\ u_E^{(j)}]^T$ which consists of $u^{(j)}$ as the intensive variable of the jth subsystem at the inlet and $u_E^{(j)}$ as the intensive variable of the external balance volume (environment).

5.1 Additive fault modelling in networks of linear heat exchange systems

In this chapter, the considered fault is a constant input signal which additively modifies the external intensive variable input signal $u_E^{(j)}$. The faulty external variable input signal $u_{Ef}^{(j)}$ has the form:

$$u_{Ef}^{(j)} = u_E^{(j)} + f^{(j)} \tag{5.1}$$

where $f^{(j)}$ is the fault signal in the jth subsystem.

Such fault modelling can describe several fault events: the unforeseen appearance of an unknown external source, or change in the heat transfer coefficient $k_E^{(j)}$.

If there is a change $f_k^{(j)}$ in the heat transfer coefficient, i.e., $k_{Ef}^{(j)} = k_E^{(j)} + f_k^{(j)}$, then the second input of the model in Eq (4.1) can be rephrased as:

$$\begin{aligned} (k_E^{(j)} + f_k^{(j)})u_E^{(j)} &= k_E^{(j)}(u_E^{(j)} + f^{(j)}), \\ \text{where } f^{(j)} &= \frac{f_k^{(j)} u_E^{(j)}}{k_E^{(j)}}. \end{aligned} \tag{5.2}$$

Note that Eq (5.2) shows the relationship between additive and multiplicative fault modelling. In other words, although the proposed method is built on the additive fault model, it can also be used for the multiplicative fault model.

During the fault diagnosis algorithm design, we assumed that:

A5 The probability of multiple fault events happening at the same time in the network is negligible (weak fault isolation), i.e., there are no simultaneous faults during the fault isolation and estimation processes.

5.2 Fault effect propagation in the network loops

To investigate the fault's effect on the subsystems of the network, the state space model in Eq (4.1) is converted into input-output realization as follows:

$$\begin{aligned} y^{(j)}(s) &= S_1^{(j)}(s)u^{(j)}(s) + S_2^{(j)}(s)u_E^{(j)}(s) \\ S_1^{(j)}(s) &= \left(\frac{v^{(j)}}{s + v^{(j)} + k_E^{(j)}}\right)^n \\ S_2^{(j)}(s) &= \sum_{h=1}^{n}\left[\frac{(v^{(j)})^{(h-1)}}{(s + v^{(j)} + k_E^{(j)})^h}\right] k_E^{(j)} \end{aligned} \tag{5.3}$$

where $S_1^{(j)}(s)$ is the transfer function in Laplace domain from $u^{(j)}(s)$ to $y^{(j)}(s)$ and $S_2^{(j)}(s)$ is the transfer function in Laplace domain from $u_E^{(j)}(s)$ to $y^{(j)}(s)$. Zero initial states are assumed.

Fig 5.1 shows the proposed realization for the fault effect analysis. For the sake of convenience, the notation $S_i^{(j)}$ will be used instead of $S_i^{(j)}(s)$ from here on.

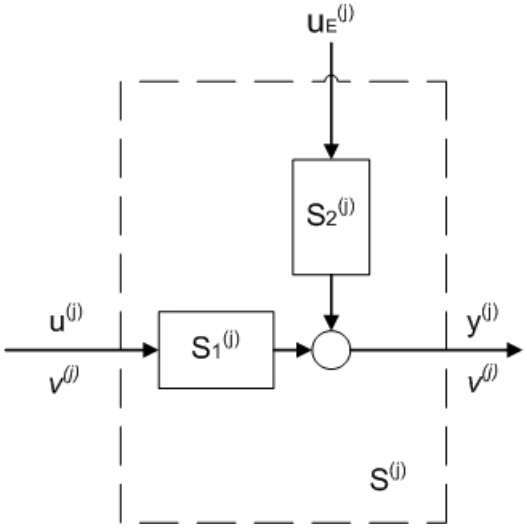

FIGURE 5.1: Input output representation of a subsystem.

Now, consider a loop in the process network as shown in Fig 5.2. There, $S^{(j)}$ represents the block diagram of the jth subsystem which contains $S_1^{(j)}$ and $S_2^{(j)}$ as shown in Fig 5.1. Meanwhile, $i^{(j)}$, where $j = 1 \dots m$, represents the inflows of the subsystems that are not part of the loop (possible joining connections), and $o^{(j)}$, where $j = 1 \dots m$, represents the outflows from the loop in splitting connections.

It is considered that the sensors are placed at the outputs $y^{(l)}$ and $y^{(m)}$ where $m \geq 2$ and $0 < l < m$.

The fault $f^{(k)}$ is represented by a constant input in the kth subsystem, where $0 < k < l$, which enters the subsystem from the same channel as the external source $u_E^{(k)}$.

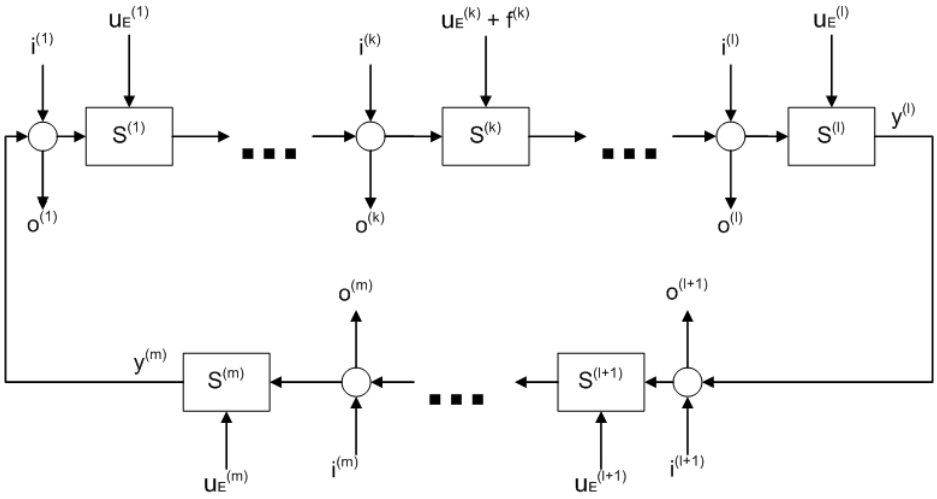

FIGURE 5.2: Diagram of a loop/cycle with fault in the process network.

For fault effect analysis, the Final Value Theorem (FVT) of the Laplace transform as in Eq (4.18) is applied.

To calculate the steady state value of the lth subsystem's output in the fault-free case $f^{(k)} = 0$, the FVT is applied to the previous transfer functions along with the superposition principle (see Fig 5.1). By assuming that the inputs are step functions with zero initial conditions, it yields:

$$
\begin{aligned}
y_{ss}^{(l)} = & \sum_{j=1}^{l} \left[\frac{\|S_2^{(j)}\|_0 \prod_{h=j+1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) u_E^{(j)} \\
& + \sum_{j=l+1}^{m} \left[\frac{\|S_2^{(j)}\|_0 \prod_{h=j+1}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) u_E^{(j)} \\
& + \sum_{j=1}^{l} \left[\frac{\prod_{h=j}^{m} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) i_{ss}^{(j)} + \sum_{j=l+1}^{m} \left[\frac{\prod_{h=j}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1 - \prod_{h=1}^{m} \|S_1^{(h)}\|_0} \right] \left(\frac{v^{(j)}}{V^{(j)}} \right) i_{ss}^{(j)}
\end{aligned}
\tag{5.4}
$$

where $V^{(j)}$ is the sum of the mass flow rate passing through the jth joining/splitting connection before the jth subsystem input, $y_{ss}^{(l)}$ is the steady state value of $y^{(l)}$ when there is no fault, $i_{ss}^{(j)}$ is the steady state value of $i^{(j)}$, and $\|\cdot\|_0$ is the steady state gain of the related transfer function. $u_E^{(j)}$ is assumed to be constant.

Since the addressed subsystem class is positive (see Appendices A.2 and 4.1), the steady state gains are also always positive.

Note that the terms of $\left(1 - \prod \|S_1^{(j)}\|_0\right)$ appear because of the loop (see Appendix

B.3). Meanwhile, the terms of $\left(\frac{v^{(j)}}{V^{(j)}}\right)$ come from the mass/energy conservation balance (see Appendix B.2).

When a step-like fault arises in the kth subsystem ($f^{(k)} \neq 0$), we obtain:

$$\begin{aligned} y_{fss}^{(l)} = & \left[\frac{\|S_2^{(k)}\|_0 \prod_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1-\prod_{j=1}^{m}\|S_1^{(j)}\|_0}\right]\left(\frac{v^{(k)}}{V^{(k)}}\right) f^{(k)} \\ & + \sum_{j=1}^{l}\left[\frac{\|S_2^{(j)}\|_0 \prod_{h=j+1}^{l} \|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right) u_E^{(j)} \\ & + \sum_{j=l+1}^{m}\left[\frac{\|S_2^{(j)}\|_0 \prod_{h=j+1}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right) u_E^{(j)} \\ & + \sum_{j=1}^{l}\left[\frac{\prod_{h=j}^{m} \|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right) i_{ss}^{(j)} + \sum_{j=l+1}^{m}\left[\frac{\prod_{h=j}^{m} \|S_1^{(h)}\|_0 \prod_{h=1}^{l} \|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right) i_{ss}^{(j)} \end{aligned} \tag{5.5}$$

where $y_{fss}^{(j)}$ is the steady state value of $y^{(j)}$ in the presence of a fault.

Now, by subtracting Eq (5.4) from Eq (5.5), we obtain the deviation of the faulty output related to the fault-free case:

$$y_{fss}^{(l)} - y_{ss}^{(l)} = \left[\frac{\|S_2^{(k)}\|_0 \prod_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1-\prod_{j=1}^{m}\|S_1^{(j)}\|_0}\right]\left(\frac{v^{(k)}}{V^{(k)}}\right) f^{(k)} \tag{5.6}$$

The difference between the faulty and fault-free outputs of the mth subsystem can be computed similarly:

$$y_{fss}^{(m)} - y_{ss}^{(m)} = \left[\frac{\|S_2^{(k)}\|_0 \prod_{j=k+1}^{m} \|S_1^{(j)}\|_0}{1-\prod_{j=1}^{m}\|S_1^{(j)}\|_0}\right]\left(\frac{v^{(k)}}{V^{(k)}}\right) f^{(k)} \tag{5.7}$$

Eqs (5.6) and (5.7) show that the fault influences all subsystems in the loop. However, the fault effect on the outputs of the subsystems can be calculated.

5.3 Fault isolation algorithm design in networks of linear heat exchange systems

Given a heat exchange network, consider that a fault event arises in one of the subsystem in the network. As a consequence of the branches and loops in the underlying graph ($\mathcal{G}$) of the process network, the effect of this fault could induce deviations in the outputs of all the subsystems. Moreover, due to the loops, the fault could

propagate back to the input of the faulty subsystem as well. The loops and network branches make the fault effect propagation barely traceable.

The presence of a fault in the network can simply be detected by comparing the measured outputs of the subsystems in the network with the predicted outputs based on a reliable model of the corresponding subsystem. However, due to fault effect propagation, the localization of the fault source in the network is a difficult task.

Recall that the subsystems represent the edges in the underlying graph $\mathcal{G}$ of the process network. We consider a directed path of subsystems $\left\{S^{(1)} \dots S^{(l)}\right\}$ where $l > 0$ such that this path may be part of at least one simple loop of $\mathcal{G}$.

Let a fault event happen in a subsystem in the network. Formulate the following fault diagnosis problems:

- Consider a path of l subsystems that can be part of a loop consisting of $m > l$ subsystems (see Fig 5.2). We must determine whether the fault occurred in the addressed path. Furthermore, we must determine which measurements are necessary to perform the isolation problem (sensor placement) in this path.
- If the fault has been isolated in one subsystem, an estimation algorithm must be designed that outputs the magnitude of the fault.

Note that the path can be part of more than one loop in the graph of the process network. As an example, in Fig 5.3, the path $\left\{S^{(A)}, S^{(F)}\right\}$ is in more than one simple loop. For an algorithm that finds all the simple loops in a graph, see, e.g., [154]. In this case, such a loop should be chosen that is more representative for the fault isolation process, e.g., from the perspective of the sensor placement.

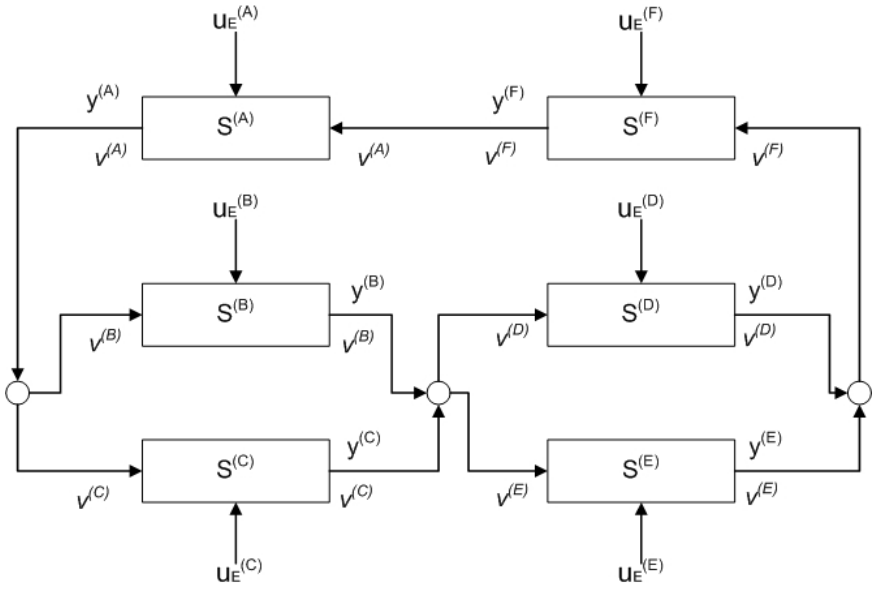

FIGURE 5.3: Example of a process network.

The formulated diagnosis problem can specify whether the fault happened in a group of subsystems (or one subsystem for $l = 1$), or in some other part of the network. However, the diagnosis process can be repeated for different groups (or subsystems). Thus, *by exhaustive search*, the fault in the network can be localized.

The fault isolation algorithm is designed by applying the fault effect analysis as presented in Section 5.2. For fault estimator design, we take the PI observer approach as presented in Appendix C.4.

To derive a fault isolation algorithm, we consider that two sensors are placed in a loop at two different locations (see Fig 5.2). They measure the output of the lth and mth subsystems.

The isolation logic is based on Eqs (5.6) and (5.7). By subtracting these equations, we obtain:

$$(y_{fss}^{(l)} - y_{ss}^{(l)}) - (y_{fss}^{(m)} - y_{ss}^{(m)}) = \left[\left(\frac{\|S_2^{(k)}\|_0 \prod_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod_{j=1}^{m} \|S_1^{(j)}\|_0} \right) \left(1 - \prod_{j=l+1}^{m} \|S_1^{(j)})\|_0 \right) \right] \left(\frac{v^{(k)}}{V^{(k)}} \right) f^{(k)} \tag{5.8}$$

We make the following assumptions on the steady state gains of the subsystems:

A6 Either every $\|S_1^{(j)}\|_0 \in (0,1)$ or every $\|S_1^{(j)}\|_0 > 1$;

A7 $\|S_2^{(j)}\|_0 > 0, \ \forall j$.

In the view of Eq (5.8), if these assumptions hold, then

$$\left(\frac{\|S_2^{(k)}\|_0 \prod_{j=k+1}^{l} \|S_1^{(j)}\|_0}{1 - \prod_{j=1}^{m} \|S_1^{(j)}\|_0} \right) \left(1 - \prod_{j=l+1}^{m} \|S_1^{(j)})\|_0 \right) > 0 \tag{5.9}$$

Note that A6 and A7 hold true for the subsystem's model introduced in Eq (4.1); if we apply the FVT to Eq (5.3), we obtain:

$$\begin{aligned} \|S_1^{(j)}\|_0 &= \lim_{s \to 0} S_1^{(j)}(s) = \left(\frac{v^{(j)}}{v^{(j)} + k_E^{(j)}} \right)^n \in (0,1) \\ \|S_2^{(j)}\|_0 &= \lim_{s \to 0} S_2^{(j)}(s) = \sum_{h=1}^{n} \left[\frac{(v^{(j)})^{(h-1)}}{(v^{(j)} + k_E^{(j)})^h} \right] k_E^{(j)} > 0 \end{aligned} \tag{5.10}$$

Furthermore, by assumptions A6 and A7, Eq (5.8) leads to:

$$\begin{aligned} (y_{fss}^{(l)} - y_{ss}^{(l)}) &> (y_{fss}^{(m)} - y_{ss}^{(m)}) \text{ for } f^{(k)} > 0 \text{ and} \\ (y_{fss}^{(l)} - y_{ss}^{(l)}) &< (y_{fss}^{(m)} - y_{ss}^{(m)}) \text{ for } f^{(k)} < 0 \\ \therefore \|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| &> 0 \text{ for } f^{(k)} \neq 0 \end{aligned} \tag{5.11}$$

where $\|\cdot\|$ is the absolute value of the related function.

Thus, Eq (5.11) shows that $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| > 0$ when a fault occurs in a subsystem between $S^{(1)}$ and $S^{(l)}$.

With the same assumptions, we can perform the same derivation to obtain $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| < 0$ when a fault occurs in a subsystem between $S^{(l+1)}$ and $S^{(m)}$.

For implementation of the fault isolation algorithm, the fault-free steady state values $y_{ss}^{(l)}$ and $y_{ss}^{(m)}$ have to be known a priori, or they have to be computed. In

view of the relation in Eq (5.4), to compute $y_{ss}^{(l)}$ and $y_{ss}^{(m)}$, the steady state value of the inputs $i_{ss}^{(j)}$ and $u_E^{(j)}$ have to be measured.

To conclude, the fault isolation can be performed according to the following Algorithm 1:

Algorithm 1 Fault isolation algorithm.

- Measure $y_{fss}^{(l)}$ and $y_{fss}^{(m)}$ in steady state.
- Compute $y_{ss}^{(l)}$ and $y_{ss}^{(m)}$.
- Isolate the fault:
  - If $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| = \|y_{fss}^{(m)} - y_{ss}^{(m)}\| = 0$, then no fault event occurred.
  - If $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| > 0$, then the fault occurred before l and after m.
  - If $\|y_{fss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fss}^{(m)} - y_{ss}^{(m)}\| < 0$, then the fault occurred before m and after l.

Remark: For future works, a possible way to generalize the proposed method and algorithm to Multi Input Multi Output (MIMO) systems is outlined in Appendix E.

5.4 Fault isolation in the presence of measurement noises

In a realistic environment, it has to be considered that the measurements on the subsystems are affected by signal noise. In the model shown in Eq (4.1), the noise inputs that influence the model's inputs and outputs are introduced as:

$$\begin{cases} \dot{\boldsymbol{x}}^{(j)} = A^{(j)}\boldsymbol{x}^{(j)} + B^{(j)}\left(\boldsymbol{u}_X^{(j)} + \boldsymbol{w}_{uX}^{(j)}\right) \\ y^{(j)} = C\boldsymbol{x}^{(j)} + w_y^{(j)} \end{cases} \tag{5.12}$$

where $\boldsymbol{w}_{uX}^{(j)} = [w_u^{(j)}\ w_E^{(j)}]^T$.

The following assumptions are considered for signal noise:

A8 $\|w_u^{(j)}(t)\|_\infty \leq w_{uM}^{(j)}, \|w_E^{(j)}(t)\|_\infty \leq w_{EM}^{(j)}, \|w_y^{(j)}(t)\|_\infty \leq w_{yM}^{(j)}$ where $\|\cdot\|_\infty$ is the infinity norm of the related function.

A9 $w_u^{(j)}(t), w_E^{(j)}(t), w_y^{(j)}(t)$ are locally integrable.

A10 The signal noise is unbiased with known variances: $w_E^{(j)}(t) \sim (0, R_u^{(j)}), w_y^{(j)}(t) \sim (0, Q_y^{(j)})$.

A11 $w_u^{(j)}(t), w_E^{(j)}(t), w_y^{(j)}(t)$ are mutually uncorrelated with both each other and system states.

The input-output model in Eq (5.3) in the presence of newly considered signal noise takes the form:

$$\begin{aligned} y_n^{(j)}(s) &= S_1^{(j)}(s)\left[u^{(j)}(s) + w_u^{(j)}(s)\right] + S_2^{(j)}(s)\left[u_E^{(j)}(s) + w_E^{(j)}(s)\right] + w_y^{(j)}(s) \\ y_n^{(j)}(s) &= y^{(j)}(s) + \left[S_1^{(j)}(s)w_u^{(j)}(s) + S_2^{(j)}(s)w_E^{(j)}(s) + w_y^{(j)}(s)\right] \\ y_n^{(j)}(s) &= y^{(j)}(s) + w^{(j)}(s). \end{aligned} \tag{5.13}$$

where $y_n^{(j)}$ is the subsystem's output in the presence of noise and $w^{(j)}$ is the noise's effect where

$$\|w^{(j)}(t)\|_\infty \leq w_M^{(j)}. \tag{5.14}$$

As the noise from each subsystem propagates inside the loop, the measurement at both points l and m in the network are affected. Thus, to compensate for noise in the fault isolation process, a suitable threshold needs to be specified.

By applying the FVT and superposition principle, the noise's effect on the measurements at sensors l and m can be computed, obtaining:

$$\begin{aligned} y_{fnss}^{(l)} &= y_{fss}^{(l)} + \sum_{j=1}^{l}\left[\frac{\prod_{h=j+1}^{l}\|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right)w^{(j)} + \sum_{j=l+1}^{m}\left[\frac{\prod_{h=j+1}^{m}\|S_1^{(h)}\|_0\prod_{h=1}^{l}\|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right)w^{(j)} \\ y_{fnss}^{(m)} &= y_{fss}^{(m)} + \sum_{j=1}^{m-1}\left[\frac{\prod_{h=j+1}^{m}\|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(j)}}{V^{(j)}}\right)w^{(j)} + \left[\frac{\prod_{h=1}^{m}\|S_1^{(h)}\|_0}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]\left(\frac{v^{(m)}}{V^{(m)}}\right)w^{(m)} \end{aligned} \tag{5.15}$$

where $y_{fnss}^{(j)}$ is the steady state value of $y^{(j)}$ in the presence of a fault and noise, $w^{(l)}$ is the noise's effect on the measurement at sensor l, and $w^{(m)}$ is the noise's effect on the measurement at sensor m.

With the addition of the noise's effect, Eq (5.8) becomes:

$$\begin{aligned} (y_{fnss}^{(l)} - y_{ss}^{(l)}) - (y_{fnss}^{(m)} - y_{ss}^{(m)}) = &\left[\left(\frac{\|S_2^{(k)}\|_0\prod_{j=k+1}^{l}\|S_1^{(j)}\|_0}{1-\prod_{j=1}^{m}\|S_1^{(j)}\|_0}\right)\left(1-\prod_{j=l+1}^{m}\|S_1^{(j)})\|_0\right)\right]\left(\frac{v^{(k)}}{V^{(k)}}\right)f^{(k)} \\ &+\sum_{j=1}^{l}\underbrace{\left[\frac{\prod_{h=j+1}^{l}\|S_1^{(h)}\|_0\left(1-\prod_{h=l+1}^{m}\|S_1^{(h)}\|_0\right)}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]}_{S_{wl}^{(j)}}\left(\frac{v^{(j)}}{V^{(j)}}\right)w^{(j)} \\ &+\sum_{j=l+1}^{m}\underbrace{\left[\frac{\prod_{h=j+1}^{m}\|S_1^{(h)}\|_0\left(\prod_{h=1}^{l}\|S_1^{(h)}\|_0-1\right)}{1-\prod_{h=1}^{m}\|S_1^{(h)}\|_0}\right]}_{S_{wm}^{(j)}}\left(\frac{v^{(j)}}{V^{(j)}}\right)w^{(j)} \end{aligned} \tag{5.16}$$

Now, for measurements at points l and m, the threshold value $th^{(lm)}$ is defined as:

$$th^{(lm)} = \sum_{j=1}^{l} \|S_{wl}^{(j)}\|_\infty w_M^{(j)} + \sum_{j=l+1}^{m} \|S_{wm}^{(j)}\|_\infty w_M^{(j)} \tag{5.17}$$

Then, by considering the measurement noise, Algorithm 1 is updated into Algorithm 2 as follows:

Algorithm 2 Fault isolation algorithm in the presence of measurement noise.

- Measure $y_{fnss}^{(l)}$ and $y_{fnss}^{(m)}$ in steady state.
- Compute $y_{ss}^{(l)}$, $y_{ss}^{(m)}$, and $th^{(lm)}$.
- Isolate the fault:
  - If $\|y_{fnss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fnss}^{(m)} - y_{ss}^{(m)}\| \in (-th^{(lm)}, th^{(lm)})$, then no fault event occurred or the fault is negligible in comparison to the threshold.
  - If $\|y_{fnss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fnss}^{(m)} - y_{ss}^{(m)}\| > th^{(lm)}$, then the fault occurred before l and after m.
  - If $\|y_{fnss}^{(l)} - y_{ss}^{(l)}\| - \|y_{fnss}^{(m)} - y_{ss}^{(m)}\| < -th^{(lm)}$, then the fault occurred before m and after l.

5.5 Observer design for additive fault estimation in networks of linear heat exchange systems

Assume that, by exhaustive search, the two sensors have been positioned such that the particular fault-affected subsystem has been located (see Section 5.3). After that, the fault estimation can be performed. A linear disturbance observer approach is proposed to determine the magnitude of the fault given by Eq (5.1).

During the estimation process, consider $k = l = 1$ and the fault has been isolated in subsystem $k = 1$.

The state space model given by Eq (4.1) with fault (5.1) can be rewritten as follows:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(k)} &= A^{(k)}\boldsymbol{x}^{(k)} + B^{(k)}\boldsymbol{u}_X^{(k)} + E^{(k)}f^{(k)} \\ y^{(k)} &= C\boldsymbol{x}^{(k)} \\ E^{(k)} &= \begin{bmatrix} k_E^{(k)} & k_E^{(k)} & \dots & k_E^{(k)} & \dots & k_E^{(k)} & k_E^{(k)} \end{bmatrix}^T \end{aligned} \tag{5.18}$$

where $E^{(k)}$ is a fault distribution column matrix and $f^{(k)}$ is assumed to be constant.

This new model in Eq (5.18) can be further transformed into an extended state space model by defining an extended state vector containing the fault as $\boldsymbol{z}^{(k)} =$

$[\boldsymbol{x^{(k)}}\ f^{(k)}]^T$:

$$\begin{aligned}\dot{\boldsymbol{z}}^{(k)} &= \begin{bmatrix} A^{(k)} & E^{(k)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x^{(k)}} \\ f^{(k)} \end{bmatrix} + \begin{bmatrix} B^{(k)} \\ 0 \end{bmatrix} \boldsymbol{u}_X^{(k)} = A_z^{(k)} \boldsymbol{z}^{(k)} + B_z^{(k)} \boldsymbol{u}_X^{(k)} \\ y^{(k)} &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x^{(k)}} \\ f^{(k)} \end{bmatrix} = C_z \boldsymbol{z}^{(k)} \end{aligned} \tag{5.19}$$

The state space model of a PI observer is:

$$\begin{aligned}\dot{\hat{\boldsymbol{x}}}^{(\boldsymbol{k})} &= A^{(k)} \hat{\boldsymbol{x}}^{(\boldsymbol{k})} + B^{(k)} \boldsymbol{u}_X^{(k)} + L_P^{(k)} (y^{(k)} - \hat{y}^{(k)}) + E^{(k)} \hat{f}^{(k)} \\ \dot{\hat{f}}^{(k)} &= L_I^{(k)} (y^{(k)} - \hat{y}^{(k)}) \\ \hat{y}^{(k)} &= C \hat{x}^{(k)} \end{aligned} \tag{5.20}$$

where $\hat{\boldsymbol{x}}^{(\boldsymbol{k})}$ is the estimated state vector, $\hat{f}^{(k)}$ is the estimated fault magnitude, $L_P^{(k)}$ is the observer's proportional gain, and $L_I^{(k)}$ is the observer's integral gain.

Hence, a PI observer can be designed for the extended state space model in Eq (5.19) to not only estimate the states but also the fault. By applying Eq (5.20) into Eq (5.19) with the assumption that $(A_z^{(k)}, C_z)$ is an observable pair, we obtain:

$$\begin{aligned}\dot{\hat{\boldsymbol{z}}}^{(\boldsymbol{k})} &= \left(\begin{bmatrix} A^{(k)} & E^{(k)} \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} L_P^{(k)} \\ L_I^{(k)} \end{bmatrix} \begin{bmatrix} C & 0 \end{bmatrix} \right) \hat{\boldsymbol{z}}^{(\boldsymbol{k})} + \begin{bmatrix} L_P^{(k)} \\ L_I^{(k)} \end{bmatrix} y^{(k)} + \begin{bmatrix} B^{(k)} \\ 0 \end{bmatrix} \boldsymbol{u}_X^{(k)} \\ &= (A_z^{(k)} - L_z^{(k)} C_z) \hat{\boldsymbol{z}}^{(\boldsymbol{k})} + L_z^{(k)} y^{(k)} + B_z^{(k)} \boldsymbol{u}_X^{(k)} \\ \hat{y}^{(k)} &= C_z \hat{\boldsymbol{z}}^{(\boldsymbol{k})} \end{aligned} \tag{5.21}$$

Thus, if $L_P^{(k)}$ and $L_I^{(k)}$ are chosen such that $(A_z^{(k)} - L_z^{(k)} C_z)$ is Hurwitz, then $\lim_{t \to \infty} (\boldsymbol{z}^{(k)} - \hat{\boldsymbol{z}}^{(k)}) = 0$.

In the presence of noise, the widely known linear quadratic estimator (LQE) procedure can be used to compute the observer's gain $L_z^{(k)}$.

5.6 Simulation results

To verify and validate the proposed fault isolation and fault estimation methods, two simulation case studies were examined in the MATLAB/Simulink environment. The first case study is based on Fig 5.4. Meanwhile, for the second case study, the investigated network comprises six subsystems as presented in Fig 5.3.

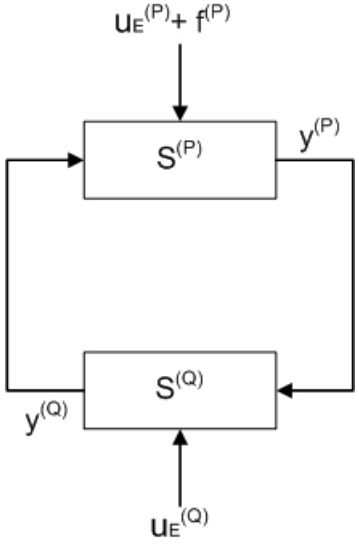

FIGURE 5.4: Diagram of two subsystems in a loop with a fault.

5.6.1 Case study 1

In this case study, two identical subsystems $S^{(P)}$ and $S^{(Q)}$ are connected as shown in Fig 5.4. The model of the subsystems is given by Eq (5.12) where each subsystem is considered to have five state variables ($n = 5$). The applied parameters and external inputs are shown in Table 5.1.

TABLE 5.1: Case study 1 parameters and external inputs.

j	P	Q
$v^{(j)}$	10	10
$k_E^{(j)}$	3	3
$u_E^{(j)}$	400	400
$R_u^{(j)}$	20	20
$Q_y^{(j)}$	20	20

It is considered that the measurements $y^{(P)}$ and $y^{(Q)}$ are influenced by noise as presented in Eq (5.12). To compensate for the noise that influences the system, a suitable threshold $th^{(PQ)}$ is computed by applying Eq (5.17). From the proposed fault isolation logic shown in Algorithm 2, as the loop is composed of two subsystems $S^{(P)}$ and $S^{(P)}$, the simulation results should show that $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is above $th^{(PQ)}$ when there is a fault in subsystem $S^{(P)}$, or $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is below $-th^{(PQ)}$ when there is a fault in subsystem $S^{(Q)}$.

In the first scenario, no fault is injected into the subsystems. The simulation result in Fig 5.5 shows $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\| \in (-th^{(PQ)}, th^{(PQ)})$, i.e., no fault is injected.

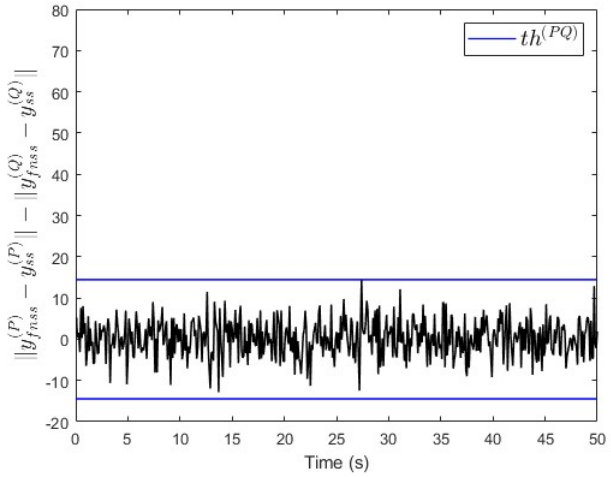

FIGURE 5.5: Case study 1 fault free case.

In the second scenario, a constant fault signal $f^{(P)} = 100$ is injected into subsystem $S^{(P)}$ at $t = 20$ s. Fig 5.6 shows that the value of $\|y^{(P)}_{fnss} - y^{(P)}_{ss}\| - \|y^{(Q)}_{fnss} - y^{(Q)}_{ss}\|$ is above the threshold after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(P)}$, which is correct.

To verify the proposed fault estimation approach, an observer is designed specifically for subsystem $S^{(P)}$ based on Eq (5.21). By using the *lqe* MATLAB function, the observer's gain is computed such that the measurement noise variances are taken into consideration. Fig 5.7 shows the fault and its estimated value. It is seen that the fault's magnitude is correctly estimated. Meanwhile, Fig 5.8 shows that the observer successfully estimates the states of subsystem $S^{(P)}$.

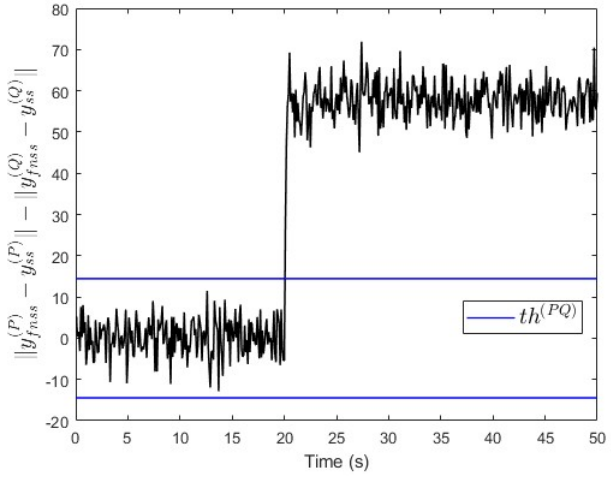

FIGURE 5.6: Case study 1 fault isolation in subsystem $S^{(P)}$.

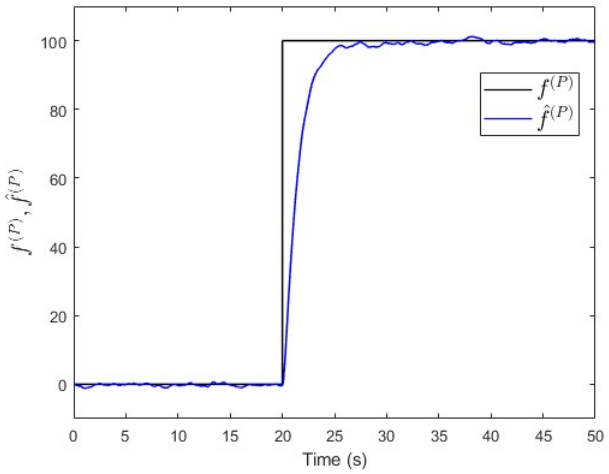

FIGURE 5.7: Case study 1 fault estimation in subsystem $S^{(P)}$.

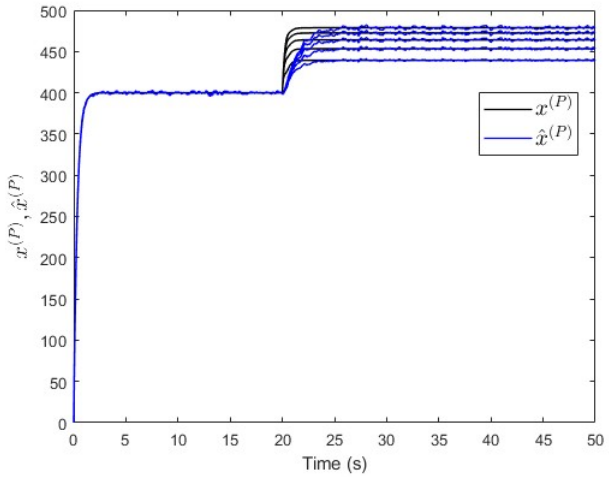

FIGURE 5.8: Case study 1 states estimation in subsystem $S^{(P)}$.

In the third scenario, a constant fault signal $f^{(Q)} = -50$ is injected into subsystem $S^{(Q)}$ at $t = 20$ s. Fig 5.9 shows that the value of $\|y^{(P)}_{fnss} - y^{(P)}_{ss}\| - \|y^{(Q)}_{fnss} - y^{(Q)}_{ss}\|$ is below $-th^{(PQ)}$ after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(Q)}$, which is correct.

Moreover, an observer is designed specifically for subsystem $S^{(Q)}$ based on Eq (5.21). The observer's gain is also computed using the same *lqe* MATLAB function such that the noise variances are taken into consideration. Fig 5.10 shows the fault and its estimated value. Here, the fault's magnitude is also correctly estimated. Meanwhile, Fig 5.11 shows that the observer successfully estimates the states of subsystem $S^{(Q)}$.

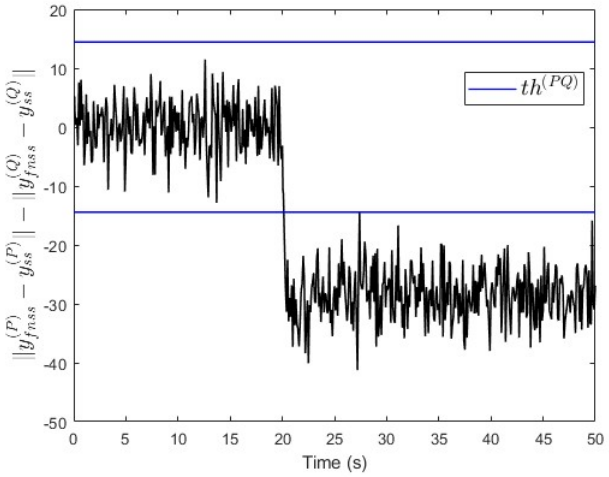

FIGURE 5.9: Case study 1 fault isolation in subsystem $S^{(Q)}$.

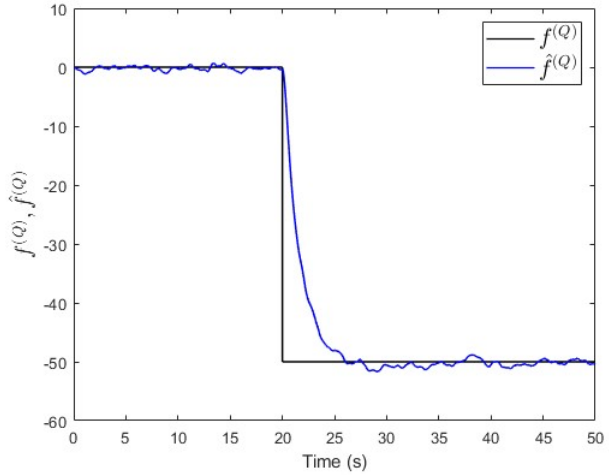

FIGURE 5.10: Case study 1 fault estimation in subsystem $S^{(Q)}$.

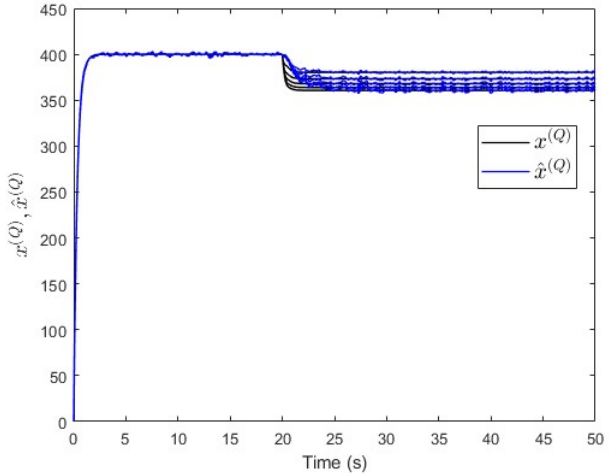

FIGURE 5.11: Case study 1 states estimation in subsystem $S^{(Q)}$.

5.6.2 Robustness analysis for case study 1

To analyse the robustness of our proposed approach, modified parameter values are considered in the subsystem's model as in Eq (4.1) only during the simulations. Here, we have two parameters to modify: the mass flow rate and the transfer coefficient. The mass flow rate $v^{(j)}$ can be easily measured. However, the transfer coefficient $k_E^{(j)}$ can only be estimated as its value changes depending on the physical conditions and circumstances. Thus, the robustness analysis is performed by checking the performance of fault isolation and estimation in the presence of transfer coefficient parameter uncertainty.

This robustness analysis for $k_E^{(j)}$ can be considered as a special type of analysis for modelling uncertainties that take into account the uncertainty of the most important parameter.

By using case study 1 for the sake of convenience but without loss of generality, the simulation is carried out by increasing the value of $k_E^{(j)}$ in Table 5.1 by 25% ($k_E^{(j)} = 3.75$ where $j = P, Q$). After this, a constant fault signal $f^{(Q)} = -50$ is injected into subsystem $S^{(Q)}$ at $t = 20$ s. Fig 5.12 shows that the value of $\|y_{fnss}^{(P)} - y_{ss}^{(P)}\| - \|y_{fnss}^{(Q)} - y_{ss}^{(Q)}\|$ is below $-th^{(PQ)}$ after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(Q)}$, which is correct. The real and estimated fault values are shown in Fig 5.13. There, the fault's magnitude is estimated with an error of less than 10%. This error is acceptable provided that the maximum tolerable value of the uncertainty in $k_E^{(j)}$ is 25%.

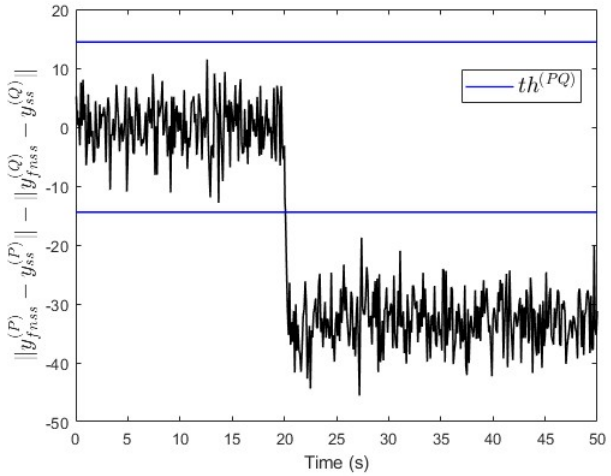

FIGURE 5.12: Case study 1 fault isolation in subsystem $S^{(Q)}$ with a parameter change.

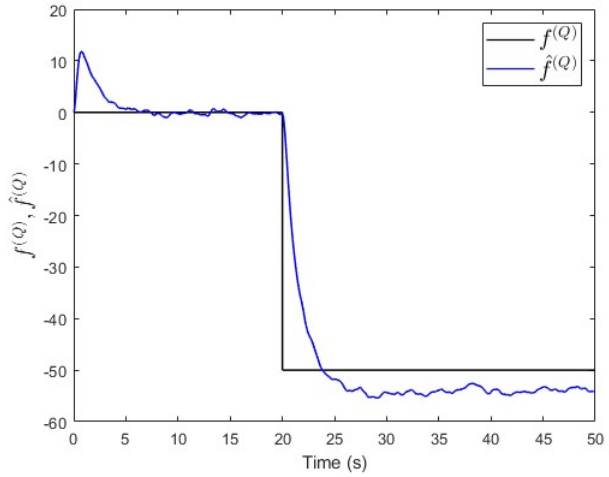

FIGURE 5.13: Case study 1 fault estimation in subsystem $S^{(Q)}$ with a parameter change.

5.6.3 Incipient fault analysis for case study 1

We also analysed the performance of our proposed approach in the case of an incipient fault (slowly developing fault). To do this, by using case study 1 again, a linearly increasing fault signal is assumed with an initial value of 0 and a slope value of 0.1 per unit of time. This fault is injected into subsystem $S^{(P)}$ at $t = 20$ s. Fig 5.14 shows that the value of $\|y^{(P)}_{fnss} - y^{(P)}_{ss}\| - \|y^{(Q)}_{fnss} - y^{(Q)}_{ss}\|$ is above the threshold $th^{(PQ)}$ some time after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(P)}$, which is correct. Meanwhile, Fig 5.15 shows that the observer successfully estimates the magnitude of the incipient fault.

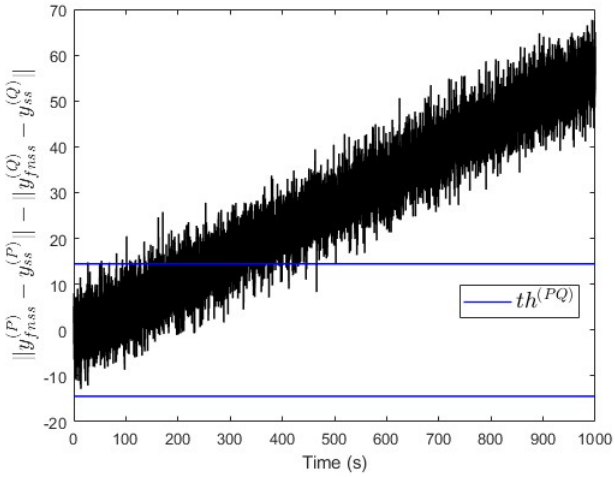

FIGURE 5.14: Case study 1 fault isolation in subsystem $S^{(P)}$ with an incipient fault.

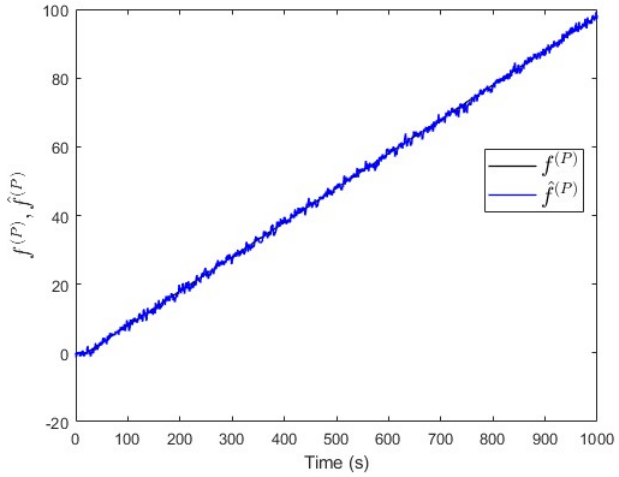

FIGURE 5.15: Case study 1 fault estimation in subsystem $S^{(P)}$ with an incipient fault.

5.6.4 Case study 2

In this case study, six subsystems $S^{(A)}$, $S^{(B)}$, $S^{(C)}$, $S^{(D)}$, $S^{(E)}$, and $S^{(F)}$ are connected as shown in Fig 5.3. The model of the subsystems is given by Eq (5.12) where each subsystem is considered to have five state variables ($n = 5$). The applied parameters and external inputs are shown in Table 5.2.

TABLE 5.2: Case study 2 parameters and external inputs.

j	A	B	C	D	E	F
$v^{(j)}$	20	4	16	8	12	20
$k_E^{(j)}$	3	3	3	3	3	3
$R_u^{(j)}$	20	20	20	20	20	20
$Q_y^{(j)}$	20	20	20	20	20	20

In this network, two sensors are placed to measure both $y^{(A)}$ and $y^{(F)}$ that are affected by noise as presented in Eq (5.12). To compensate for the noise that influences the system, a suitable threshold $th^{(AF)}$ is computed as presented in Eq (5.17). As both subsystem $S^{(A)}$ and $S^{(F)}$ are inside the loops of either $\left\{S^{(A)}, S^{(B)}, S^{(D)}, S^{(F)}\right\}$ or $\left\{S^{(A)}, S^{(B)}, S^{(E)}, S^{(F)}\right\}$, the simulation's results should show that: $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is above the threshold $th^{(AF)}$ when there is a fault in subsystem $S^{(A)}$, or $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is below $-th^{(AF)}$ when there is a fault in either subsystem $S^{(B)}$, $S^{(C)}$, $S^{(D)}$, $S^{(E)}$, or $S^{(F)}$ (see Algorithm 2).

In the first scenario, no fault is injected into the subsystems. The simulation results in Fig 5.16 show that $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\| \in (-th^{(AF)}, th^{(AF)})$, i.e., no fault is injected.

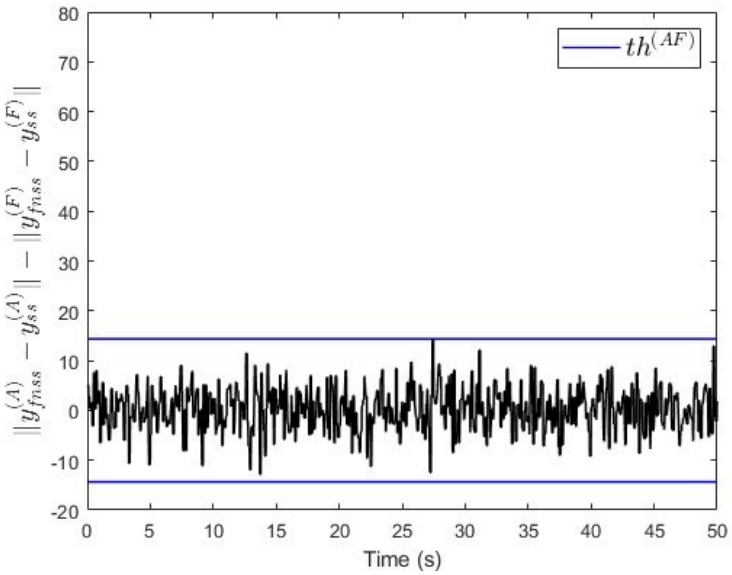

FIGURE 5.16: Case study 2 fault free case.

In the second scenario, a constant fault signal $f^{(A)} = 200$ is injected into subsystem $S^{(A)}$ at $t = 20$ s. Fig 5.17 shows that the value of $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is above the threshold after the fault event occurred. This indicates that a fault is occurring in subsystem $S^{(A)}$, which is correct.

To verify the proposed fault estimation approach, an observer is designed for subsystem $S^{(A)}$ based on Eq (5.21). By using the *lqe* MATLAB function, the observer's gain is computed such that the measurement noise variances are taken into consideration. Fig 5.18 shows the fault and its estimated value. It is seen that the fault's magnitude is correctly estimated. Meanwhile, Fig 5.19 shows that the observer successfully estimates the states of subsystem $S^{(A)}$.

In the third scenario, a constant fault signal $f^{(F)} = -150$ is injected into subsystem $S^{(F)}$ at $t = 20$ s. Fig 5.20 shows that the value of $\|y_{fnss}^{(A)} - y_{ss}^{(A)}\| - \|y_{fnss}^{(F)} - y_{ss}^{(F)}\|$ is below $-th^{(AF)}$ after the fault event occurred. This indicates that a fault is occurring either in subsystem $S^{(B)}$, $S^{(C)}$, $S^{(D)}$, $S^{(E)}$, or $S^{(F)}$, which is correct. However, to specifically locate the fault-affected subsystem ($S^{(F)}$), an exhaustive search must be done by shifting the position of the two sensors and repeating the fault isolation algorithm (see Section 5.3).

An observer is then designed specifically for subsystem $S^{(F)}$ based on the same Eq (5.21). The observer's gain is computed using the same *lqe* MATLAB function such that the noise variances are taken into consideration. Fig 5.21 shows the fault and its estimated value. Here, the fault's magnitude is also correctly estimated. Meanwhile, Fig 5.22 shows that the observer successfully estimates the states of subsystem $S^{(F)}$.

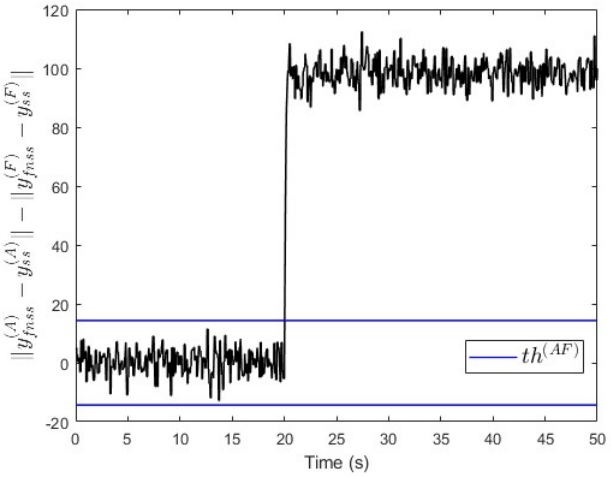

FIGURE 5.17: Case study 2 fault isolation in subsystem $S^{(A)}$.

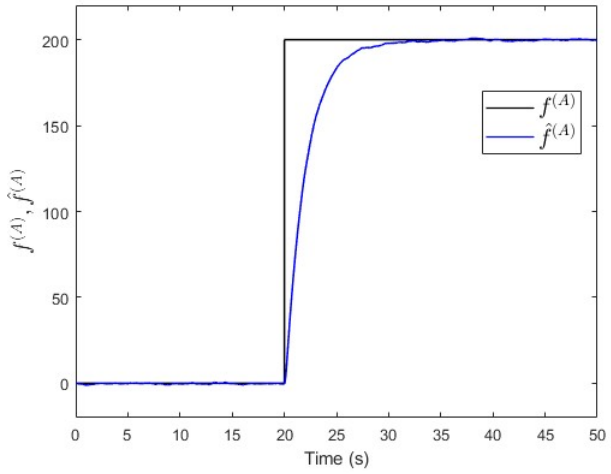

FIGURE 5.18: Case study 2 fault estimation in subsystem $S^{(A)}$.

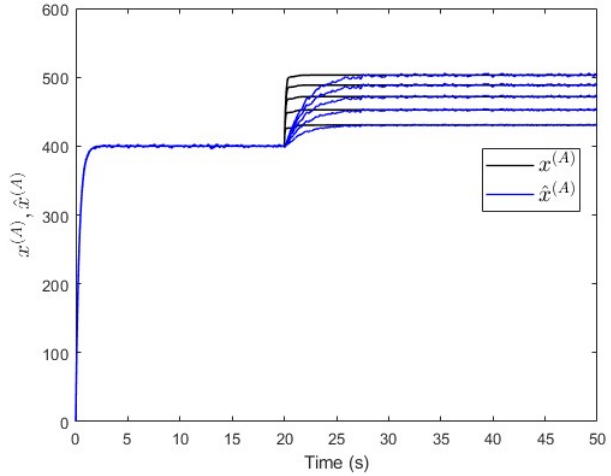

FIGURE 5.19: Case study 2 states estimation in subsystem $S^{(A)}$.

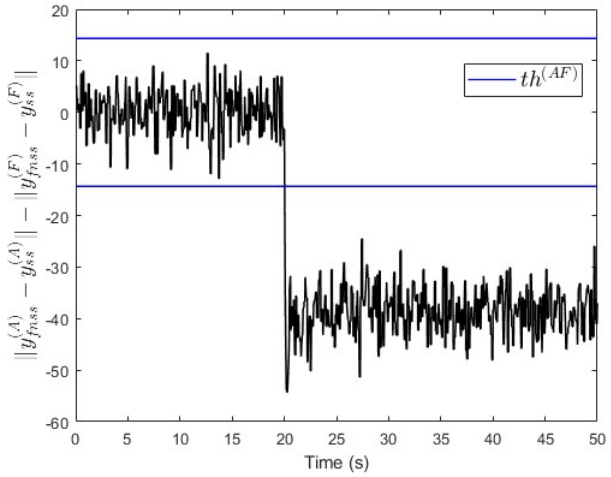

FIGURE 5.20: Case study 2 fault isolation in subsystem $S^{(F)}$.

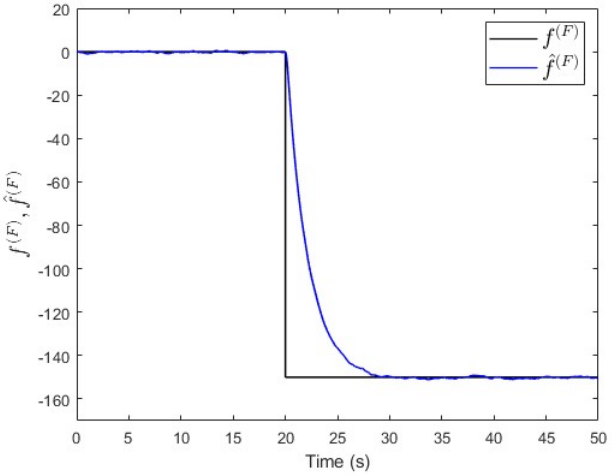

FIGURE 5.21: Case study 2 fault estimation in subsystem $S^{(F)}$.

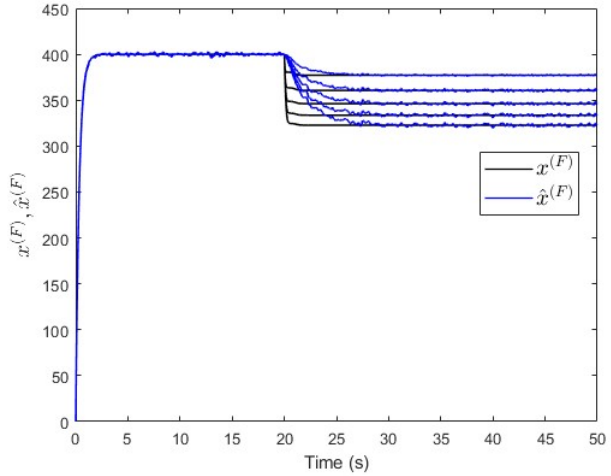

FIGURE 5.22: Case study 2 states estimation in subsystem $S^{(F)}$.

5.7 Summary and discussion

In this chapter, the case of networked linear heat exchange systems with only a single conserved extensive quantity but with a network structure containing loops is considered. It is assumed that the probability of multiple faults happening at the same time in the network is negligible.

For fault detection and isolation purposes, the network elements are described by 2ISO LTI state space models where the fault enters as an additive linear term into the second input of the equations representing the external environment. Using the models of the network elements, a general model of the network is constructed which includes static splitting and joining connections. This results in an LTI state space model for the overall networked system that is suitable for fault detection and isolation.

By analysing the effect of the fault in a subsystem that propagates to the sensors placed at different positions in the network, a fault isolation algorithm (Algorithm 1) was proposed. It uses two sensors that are installed at the output of the two subsystems in the loop inside the network to locate the fault. A steady state analysis and superposition principle were used to build the algorithm. An improved version of the algorithm (Algorithm 2) was also proposed to perform the fault localization in the presence of measurement noise.

Having completed the fault isolation process, a PI disturbance observer was then proposed to estimate the magnitude of the fault.

The proposed method shows good robustness against the uncertainty in transfer coefficient parameter k_E which is difficult to estimate in practice. In addition, it can also correctly estimate incipient faults modelled as ramp-like fault signals.

Even though the proposed method is built for networked heat exchange systems, it can be generalized to the networked LTI systems as long as the properties of the network elements are the same, which are: 1. The systems are positive. This positivity in LTI systems will always result in the same structure of the A matrix in the system's state space model $\dot{\boldsymbol{x}} = A\boldsymbol{x}$ where the off-diagonal elements are non-negative, i.e., a Metzler matrix [143]. Moreover, as the diagonal elements are negative, this implies that the systems are also stable [144, 145, 146] (see Appendix A.2), 2. It can be modelled as a 2ISO LTI system, 3. The fault enters the system through the system's second input, 4. The mass and/or energy conservation applies in the branching points, i.e., splitting and joining connections (see Appendix B.2). For the relation between additive and multiplicative fault modelling, we can refer to Eq (5.2).

The work here tries to contribute by giving an alternative using model based approach for fault diagnosis in a loop compared to other literature that uses data driven approaches [81, 82]. Moreover, the proposed method requires only 2 sensors for implementation.

Since the proposed method is built on the heat exchange network model, some further efforts can be made by trying to generalize it to MIMO systems. Thus, Appendix E is dedicated to outline this future work.

Chapter 6

Conclusions

6.1 Thesis points

The summary of the new scientific contributions is as follows:

Thesis I

The corresponding publications are [A1] and [A2].

In this thesis, a model based actuator and sensor fault diagnosis method for robot platoons that move in a leader-follower scheme is proposed. As a networked system with distributed control, the states of each robot are affected by the states of its preceding robot. Thus, to achieve reliable platoon control, network communication can be applied for the sake of information transmission. The sensor and actuator fault diagnosis method is studied separately because of their different goals. For actuator fault diagnosis, the goals are to detect, isolate, and estimate the fault. Meanwhile, for sensor fault diagnosis, the faulty sensor's detection and isolation are the goals. The considered sensors include a GPS-based sensor, a wheel-mounted velocity sensor, and a radar sensor for inter-vehicle distance measurement. No sensor fault is assumed in studying the actuator fault diagnosis and vice versa.

Sensor fault diagnosis related research results:

1. A model for robot platoons with network communication was proposed that can capture the effect of sensor fault propagation in the networked systems.

2. By using the networked system model as the basis, a bank of distributed Unknown Input Observers (UIO)s as residual generators was designed that isolate the faulty sensor in each robot.

3. A threshold computation was also proposed to prevent false alarms in the presence of measurement noise when there are no faulty sensors.

4. The extension of the UIO-based sensor fault isolation method for a more general class of networked LTI systems is also presented.

Actuator fault diagnosis related research results:

1. By exploiting the measurement redundancy, a filtering technique is proposed to eliminate the need for network communication so that the actuator fault and the communication fault can be decoupled from each other.

2. By using the filter, the model without network communication was derived that shows similar behaviour to the original model with network communication.

3. Based on the model without network communication, a PI observer was designed to estimate the magnitude of the actuator fault.

4. Two estimation filter gain design methods were examined. The pole placement design method results in a rapid fault estimation process but the measurement noise still influences the estimation result. Meanwhile, by using the Linear Quadratic Estimator (LQE), the measurement noise is eliminated but the estimation process needs a much longer time.

Thesis II

The corresponding publications are [A3] and [A4].

In this thesis, model based methods are proposed for fault detection, isolation, and estimation in heat exchange networks where the considered fault is the parameter change in the heat transfer coefficient that describes the effect of environmental temperature on the temperatures inside the heat exchange networks. In addition, the proposed model of the heat exchange networks includes splitting and joining connections that are common in real world applications.

The results of this research are:

1. By using SDG to investigate the structural observability in solving the sensor placement problem, it is found that

   - For splitting connections, it is enough to put sensors only at the end of each element after the split to detect, isolate, and estimate the fault that happens either before or after the split.
   - For joining connections, it is necessary to put sensors at the end of each element both before and after the joint to detect, isolate, and estimate the fault that happens either before or after the joint.
   - If the fault happens only at a certain location along the length of a specific tube, then putting a sensor only at the end of the considered tube is enough to detect and estimate the fault.

2. In the splitting connections, a bank of linear observers can detect and isolate whether the fault happens before or after the split.

3. As the parameter fault induces bilinear fault input terms in the model, a bank of nonlinear observers that is based on the parameter adaptation can estimate it.

4. By using sensitivity analysis, it is found that fault isolation can hardly be performed in the case of a single parameter fault in a certain location in a specific tube.

Thesis III

The corresponding publication is [A5].

In this thesis, a model based fault diagnosis method is proposed for the networked linear heat exchange systems that contain splitting and joining connections, and also loops. The considered fault enters into the subsystem as an additive linear input term.

The results of this research are:

1. By using steady state value analysis, a fault isolation algorithm requiring two sensors is proposed that can localize the fault in the loops inside the network.
2. After the fault has been isolated, a PI observer is proposed to estimate the magnitude of the fault.
3. The proposed method has good robustness against the heat transfer coefficient parameter uncertainty that is hard to estimate in practice.
4. The proposed method can also isolate and correctly estimate incipient faults (slowly developing faults).

6.2 Future works

In this subsection, possible future works for each thesis point are presented:

- For Thesis I, there is a possibility to explore other approaches to deal with the diagnosis of the communication faults in the network. In addition, the research results can further be used for designing a fault tolerant controller for such systems. Lastly, in this present study, actuator fault and sensor fault diagnosis are studied separately (i.e. no sensor fault is assumed during the actuator fault diagnosis study and vice versa). Thus, a further study to simultaneously integrate them can be considered.
- For Thesis II, it is shown that parameter fault at a certain location along the length of a tube in an element of heat exchange networks is hard to isolate. This leads to possible future work in finding other approaches to tackle this problem.
- For Thesis III, further studies can be carried out to generalize the proposed method to MIMO systems. The outline can be seen in Appendix E.

The author's publications

[A1] W. Kurniawan and L. Marton. "Actuator fault estimation in robot platoons". In: *Proceedings of the 11th International Conference on Systems and Control*. 2023, pp. 858–863.

[A2] W. Kurniawan and L. Marton. "Sensor Faults Isolation in Networked Control Systems: Application to Mobile Robot Platoons". In: *Sensors* 21.20 (2021), p. 6702.

[A3] W. Kurniawan, K. M. Hangos, and L. Márton. "Parameter fault diagnosis in heat exchange networks with distributed time delay". In: *IFAC-PapersOnLine* 55.18 (2022), pp. 39–44.

[A4] W. Kurniawan, K. M. Hangos, and L. Marton. "Parameter fault estimation in distributed heating/cooling systems". In: *Proceedings of the 7th International Conference on Sustainable Information Engineering and Technology*. 2022, pp. 111–118.

[A5] W. Kurniawan, K. M. Hangos, and L. Márton. "Fault Isolation and Estimation in Networks of Linear Process Systems". In: *Entropy* 25.6 (2023), p. 862.

Bibliography

[1] S. Sastry. *Nonlinear systems: analysis, stability, and control*. Vol. 10. Springer Science & Business Media, 2013.

[2] J. P. Hespanha. *Linear systems theory*. Princeton university press, 2018.

[3] S. L. Brunton and J. N. Kutz. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.

[4] L. Zhang et al. "Internet connected vehicle platoon system modeling and linear stability analysis". In: *Computer Communications* 174 (2021), pp. 92–100.

[5] Y. Zhu and F. Zhu. "Distributed adaptive longitudinal control for uncertain third-order vehicle platoon in a networked environment". In: *IEEE Transactions on Vehicular Technology* 67.10 (2018), pp. 9183–9197.

[6] C. Latrech et al. "Integrated longitudinal and lateral networked control system design for vehicle platooning". In: *sensors* 18.9 (2018), p. 3085.

[7] S. Stüdli, M. M. Seron, and R. H. Middleton. "From vehicular platoons to general networked systems: String stability and related concepts". In: *Annual Reviews in Control* 44 (2017), pp. 157–172.

[8] S. K. Singh. *Process control: concepts dynamics and applications*. PHI Learning Pvt. Ltd., 2010.

[9] M. Bakošová and J. Oravec. "Robust model predictive control for heat exchanger network". In: *Applied Thermal Engineering* 73.1 (2014), pp. 924–930.

[10] A. Vasičkaninová et al. "Neural network predictive control of a heat exchanger". In: *Applied Thermal Engineering* 31.13 (2011), pp. 2094–2100.

[11] A. H. González, D. Odloak, and J. L. Marchetti. "Predictive control applied to heat-exchanger networks". In: *Chemical Engineering and Processing: Process Intensification* 45.8 (2006), pp. 661–671.

[12] M. Blanke et al. *Diagnosis and fault-tolerant control*. Vol. 2. Springer, 2006.

[13] L. Manservigi et al. "A diagnostic approach for fault detection and identification in district heating networks". In: *Energy* 251 (2022), p. 123988.

[14] J. Granderson et al. "Characterization and survey of automated fault detection and diagnostic tools". In: *Report Number LBNL-2001075* (2017).

[15] S. Buffa et al. "Advanced control and fault detection strategies for district heating and cooling systems—A review". In: *Applied Sciences* 11.1 (2021), p. 455.

[16] S. Zhou, Z. O'Neill, and C. O'Neill. "A review of leakage detection methods for district heating networks". In: *Applied Thermal Engineering* 137 (2018), pp. 567–574.

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

[18] S. X. Ding. *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Science & Business Media, 2008.

[19] A. Varga. "Solving fault diagnosis problems". In: *Studies in Systems, Decision and Control, 1st ed.; Springer International Publishing: Berlin, Germany* 84 (2017), pp. 8–9.

[20] S. Simani, C. Fantuzzi, and R. J. Patton. "Model-based fault diagnosis techniques". In: *Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques*. Springer, 2003, pp. 19–60.

[21] K. Michail et al. "AI-based actuator/sensor fault detection with low computational cost for industrial applications". In: *IEEE Transactions on Control Systems Technology* 24.1 (2015), pp. 293–301.

[22] S. Nasiri, M. R. Khosravani, and K. Weinberg. "Fracture mechanics and mechanical fault detection by artificial intelligence methods: A review". In: *Engineering Failure Analysis* 81 (2017), pp. 270–293.

[23] S. Xiao et al. "Fault Detection and Isolation Methods in Subsea Observation Networks". In: *Sensors* 20.18 (2020), p. 5273.

[24] A. I. Pózna, A. Fodor, and K. M. Hangos. "Model-based fault detection and isolation of non-technical losses in electrical networks". In: *Mathematical and Computer Modelling of Dynamical Systems* 25.4 (2019), pp. 397–428.

[25] D. Li et al. "Recent advances in sensor fault diagnosis: A review". In: *Sensors and Actuators A: Physical* (2020), p. 111990.

[26] T. C. Yang. "Networked control system: a brief survey". In: *IEE Proceedings-Control Theory and Applications* 153.4 (2006), pp. 403–412.

[27] M. E. Newman, A.-L. E. Barabási, and D. J. Watts. *The structure and dynamics of networks.* Princeton university press, 2006.

[28] C. T. Abdallah and H. G. Tanner. "Complex networked control systems: Introduction to the special section". In: *IEEE Control Systems Magazine* 27.4 (2007), pp. 30–32.

[29] J. Baillieul and P. J. Antsaklis. "Control and communication challenges in networked real-time systems". In: *Proceedings of the IEEE* 95.1 (2007), pp. 9–28.

[30] R. A. Gupta and M.-Y. Chow. "Networked control system: Overview and research trends". In: *IEEE Transactions on Industrial Electronics* 57.7 (2009), pp. 2527–2535.

[31] L. Zhang, H. Gao, and O. Kaynak. "Network-induced constraints in networked control systems—A survey". In: *IEEE Transactions on Industrial Informatics* 9.1 (2012), pp. 403–416.

[32] X.-M. Zhang et al. "Networked control systems: A survey of trends and techniques". In: *IEEE/CAA Journal of Automatica Sinica* 7.1 (2019), pp. 1–17.

[33] H. Fang, H. Ye, and M. Zhong. "Fault diagnosis of networked control systems". In: *Annual reviews in control* 31.1 (2007), pp. 55–68.

[34] J. Song and X. He. "Model-based fault diagnosis of networked systems: A survey". In: *Asian Journal of Control* 24.2 (2022), pp. 526–536.

[35] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu. "A survey of recent results in networked control systems". In: *Proceedings of the IEEE* 95.1 (2007), pp. 138–162.

[36] F. Boem, R. M. Ferrari, and T. Parisini. "Distributed fault detection and isolation of continuous-time non-linear systems". In: *European Journal of Control* 17.5-6 (2011), pp. 603–620.

[37] Y. Peng et al. "Sensor fault detection and isolation for a wireless sensor network-based remote wind turbine condition monitoring system". In: *IEEE Transactions on Industry Applications* 54.2 (2017), pp. 1072–1079.

[38] P. M. Papadopoulos et al. "Robust fault detection, isolation, and accommodation of current sensors in grid side converters". In: *IEEE Transactions on Industry Applications* 53.3 (2016), pp. 2852–2861.

[39] Y. Huang et al. "Real-time detection of false data injection in smart grid networks: An adaptive CUSUM method and analysis". In: *IEEE Systems Journal* 10.2 (2014), pp. 532–543.

[40] K. Manandhar et al. "Detection of faults and attacks including false data injection attack in smart grid using Kalman filter". In: *IEEE transactions on control of network systems* 1.4 (2014), pp. 370–379.

[41] J. Tian et al. "Coordinated cyber-physical attacks considering DoS attacks in power systems". In: *International Journal of Robust and Nonlinear Control* 30.11 (2020), pp. 4345–4358.

[42] J. Shi et al. "Fault-tolerant formation control of non-linear multi-vehicle systems with application to quadrotors". In: *IET Control Theory & Applications* 11.17 (2017), pp. 3179–3190.

[43] J. Shi et al. "Distributed fault detection for formation of multi-agent systems". In: *2017 Chinese Automation Congress (CAC)*. IEEE. 2017, pp. 4134–4139.

[44] Y. Zhang et al. "Fault detection filter design for networked multi-rate systems with fading measurements and randomly occurring faults". In: *IET Control Theory & Applications* 10.5 (2016), pp. 573–581.

[45] C. Liu et al. "Minimum-variance unbiased unknown input and state estimation for multi-agent systems by distributed cooperative filters". In: *IEEE Access* 6 (2018), pp. 18128–18141.

[46] L. Ye, F. Zhu, and J. Zhang. "Sensor attack detection and isolation based on sliding mode observer for cyber-physical systems". In: *International Journal of Adaptive Control and Signal Processing* 34.4 (2020), pp. 469–483.

[47] Y. Gao et al. "Distributed soft fault detection for interval type-2 fuzzy-model-based stochastic systems with wireless sensor networks". In: *IEEE Transactions on Industrial Informatics* 15.1 (2018), pp. 334–347.

[48] X. Bu et al. "Non-fragile distributed fault estimation for a class of nonlinear time-varying systems over sensor networks: The finite-horizon case". In: *IEEE Transactions on Signal and Information Processing over Networks* 5.1 (2018), pp. 61–69.

[49] A. S. Khan et al. "Distributed fault detection and isolation in second order networked systems in a cyber–physical environment". In: *ISA transactions* 103 (2020), pp. 131–142.

[50] X. Zhang and F. Zhu. "Observer-based sensor attack diagnosis for cyber-physical systems via zonotope theory". In: *Asian Journal of Control* 23.5 (2021), pp. 2444–2458.

[51] D. Xu, Y. Wang, and L. Zhang. "Realization of actuator-fault estimation for distributed wireless networked control systems". In: *Asian Journal of Control* 19.4 (2017), pp. 1271–1283.

[52] P. N. Beni and M. Shahriari-kahkeshi. "Distributed actuator fault detection and estimation for linear multi agent systems". In: *2019 6th International Conference on Control, Instrumentation and Automation (ICCIA)*. IEEE. 2019, pp. 1–6.

[53] D. J. Chmielewski, T. Palmer, and V. Manousiouthakis. "On the theory of optimal sensor placement". In: *AIChE journal* 48.5 (2002), pp. 1001–1012.

[54] S. S. Dhillon and K. Chakrabarty. "Sensor placement for effective coverage and surveillance in distributed sensor networks". In: *2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003.* Vol. 3. IEEE. 2003, pp. 1609–1614.

[55] W. Ostachowicz, R. Soman, and P. Malinowski. "Optimization of sensor placement for structural health monitoring: A review". In: *Structural Health Monitoring* 18.3 (2019), pp. 963–988.

[56] M. Cantoni et al. "Control of large-scale irrigation networks". In: *Proceedings of the IEEE* 95.1 (2007), pp. 75–91.

[57] R. Mukherjee, U. M. Diwekar, and A. Vaseashta. "Optimal sensor placement with mitigation strategy for water network systems under uncertainty". In: *Computers & Chemical Engineering* 103 (2017), pp. 91–102.

[58] K. Worden and A. Burrows. "Optimal sensor placement for fault detection". In: *Engineering structures* 23.8 (2001), pp. 885–901.

[59] R. Rajamani et al. "A complete fault diagnostic system for automated vehicles operating in a platoon". In: *IEEE transactions on control systems technology* 9.4 (2001), pp. 553–564.

[60] A. Khalil et al. "Output-only fault detection and mitigation of networks of autonomous vehicles". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 2257–2264.

[61] A. Khalil and M. Al Janaideh. "On fault classification in connected autonomous vehicles using supervised machine learning". In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 1198–1204.

[62] G. Wang et al. "Two-level fault detection and isolation algorithm for vehicle platoon". In: *IEEE Access* 6 (2018), pp. 15106–15116.

[63] N. A. M. Subha and M. N. Mahyuddin. "Distributed adaptive cooperative control with fault compensation mechanism for heterogeneous multi-robot system". In: *IEEE Access* 9 (2021), pp. 128550–128563.

[64] A. Khalil et al. "Fault detection, localization, and mitigation of a network of connected autonomous vehicles using transmissibility identification". In: *2020 American Control Conference (ACC)*. IEEE. 2020, pp. 386–391.

[65] W. Wang et al. "Fault-tolerant platoon control of autonomous vehicles based on event-triggered control strategy". In: *IEEE Access* 8 (2020), pp. 25122–25134.

[66] X. Liu, X. Gao, and J. Han. "Robust unknown input observer based fault detection for high-order multi-agent systems with disturbances". In: *ISA Transactions* 61 (2016), pp. 15–28.

[67] I. Shames et al. "Distributed fault detection and isolation with imprecise network models". In: *2012 American Control Conference (ACC)*. IEEE. 2012, pp. 5906–5911.

[68] K. Zhang, G. Liu, and B. Jiang. "Robust unknown input observer-based fault estimation of leader–follower linear multi-agent systems". In: *Circuits, Systems, and Signal Processing* 36.2 (2017), pp. 525–542.

[69] T. Sorsa, H. N. Koivo, and H. Koivisto. "Neural networks in process fault diagnosis". In: *IEEE Transactions on systems, man, and cybernetics* 21.4 (1991), pp. 815–825.

[70] M. T. Amin, S. Imtiaz, and F. Khan. "Process system fault detection and diagnosis using a hybrid technique". In: *Chemical Engineering Science* 189 (2018), pp. 191–211.

[71] M. Alauddin et al. "A bibliometric review and analysis of data-driven fault detection and diagnosis methods for process systems". In: *Industrial & Engineering Chemistry Research* 57.32 (2018), pp. 10719–10735.

[72] N. Md Nor, C. R. Che Hassan, and M. A. Hussain. "A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems". In: *Reviews in Chemical Engineering* 36.4 (2020), pp. 513–553.

[73] R. Arunthavanathan et al. "An analysis of process fault diagnosis methods from safety perspectives". In: *Computers & Chemical Engineering* 145 (2021), p. 107197.

[74] D. Dragan. "Fault detection of an industrial heat-exchanger: A model-based approach". In: *Strojniški vestnik-Journal of Mechanical Engineering* 57.6 (2011), pp. 477–484.

[75] N. Khentout et al. "Fault monitoring and accommodation of the heat exchanger parameters of triga-mark II nuclear research reactor using model-based analytical redundancy". In: *Progress in Nuclear Energy* 109 (2018), pp. 97–112.

[76] H. O. Njoku et al. "Combined pinch and exergy evaluation for fault analysis in a steam power plant heat exchanger network". In: *Journal of Energy Resources Technology* 141.12 (2019), p. 122001.

[77] R. F. Garcia. "Improving heat exchanger supervision using neural networks and rule based techniques". In: *Expert Systems with Applications* 39.3 (2012), pp. 3012–3021.

[78] M Mohanraj, S Jayaraj, and C Muraleedharan. "Applications of artificial neural networks for thermal analysis of heat exchangers–a review". In: *International Journal of Thermal Sciences* 90 (2015), pp. 150–172.

[79] J. Huang et al. "An experimental study of clogging fault diagnosis in heat exchangers based on vibration signals". In: *IEEE access* 4 (2016), pp. 1800–1809.

[80] S. M. M. Alavi, R. Izadi-Zamanabadi, and M. J. Hayes. "On the Generation of a Robust Residual for Closed-loop Control systems that Exhibit Sensor Faults". In: *The IET Irish Signals and Systems Conference 2007*. 2007.

[81] S. Zhai, W. Wang, and H. Ye. "Fault diagnosis based on parameter estimation in closed-loop systems". In: *IET Control Theory & Applications* 9.7 (2015), pp. 1146–1153.

[82] B. Sun et al. "Fault identification for a closed-loop control system based on an improved deep neural network". In: *Sensors* 19.9 (2019), p. 2131.

[83] R. Olfati-Saber. "Flocking for multi-agent dynamic systems: Algorithms and theory". In: *IEEE Transactions on automatic control* 51.3 (2006), pp. 401–420.

[84] J. Qin et al. "Recent advances in consensus of multi-agent systems: A brief survey". In: *IEEE Transactions on Industrial Electronics* 64.6 (2016), pp. 4972–4983.

[85] M. Perc et al. "Statistical physics of human cooperation". In: *Physics Reports* 687 (2017), pp. 1–51.

[86] Y. Li and C. Tan. "A survey of the consensus for multi-agent systems". In: *Systems Science & Control Engineering* 7.1 (2019), pp. 468–482.

[87] L. Roveda, B. Spahiu, and W. Terkaj. "On the Proposal of a Unified Safety Framework for Industry 4.0 Multi-Robot Scenario." In: *SEBD*. 2019.

[88] N. Mohamed, J. Al-Jaroodi, and I. Jawhar. "A review of middleware for networked robots". In: *International Journal of Computer Science and Network Security* 9.5 (2009), pp. 139–148.

[89] A. Gonzalez-Ruiz, A. Ghaffarkhah, and Y. Mostofi. "A comprehensive overview and characterization of wireless channels for networked robotic and control systems". In: *Journal of Robotics* 2011 (2011).

[90] Á. Fehér and L. Márton. "Transient Behaviour Analysis and Design for Platoons with Communication Delay". In: *IFAC-PapersOnLine* 52.18 (2019), pp. 13–18.

[91] L. Xiao and F. Gao. "A comprehensive review of the development of adaptive cruise control systems". In: *Vehicle System Dynamics* 48.10 (2010), pp. 1167–1192.

[92] R. Rajamani. *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[93] J. Lunze. *Feedback control of large-scale systems*. Prentice Hall New York, 1992.

[94] C. Wu, Z. Yang, and Y. Liu. *Wireless Indoor Localization: A Crowdsourcing Approach*. Springer, 2018.

[95] L. Qin, X. He, and D. Zhou. "A survey of fault diagnosis for swarm systems". In: *Systems Science & Control Engineering: An Open Access Journal* 2.1 (2014), pp. 13–23.

[96] Y. Mo and B. Sinopoli. "False data injection attacks in control systems". In: *Preprints of the 1st workshop on Secure Control Systems*. Vol. 1. 2010.

[97] L. Hu et al. "State estimation under false data injection attacks: Security analysis and system protection". In: *Automatica* 87 (2018), pp. 176–183.

[98] M. Ahmed and A.-S. K. Pathan. "False data injection attack (FDIA): an overview and new metrics for fair evaluation of its countermeasure". In: *Complex Adaptive Systems Modeling* 8.1 (2020), pp. 1–14.

[99] S. Patel et al. "Distributed detection of malicious attacks on consensus algorithms with applications in power networks". In: *2020 7th International Conference on Control, Decision and Information Technologies (CoDIT)*. Vol. 1. IEEE. 2020, pp. 397–402.

[100] Z. Gao, C. Cecati, and S. X. Ding. "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches". In: *IEEE transactions on industrial electronics* 62.6 (2015), pp. 3757–3767.

[101] H. Alikhani and N. Meskin. "Event-triggered robust fault diagnosis and control of linear Roesser systems: A unified framework". In: *Automatica* 128 (2021), p. 109575.

[102] F. Xu. "Observer-based asymptotic active fault diagnosis: A two-layer optimization framework". In: *Automatica* 128 (2021), p. 109558.

[103] X.-G. Guo et al. "Distributed neuroadaptive fault-tolerant sliding-mode control for 2-D plane vehicular platoon systems with spacing constraints and unknown direction faults". In: *Automatica* 129 (2021), p. 109675.

[104] C. Keliris, M. M. Polycarpou, and T. Parisini. "A robust nonlinear observer-based approach for distributed fault detection of input–output interconnected systems". In: *Automatica* 53 (2015), pp. 408–415.

[105] J. Qin et al. "Adaptive sliding mode consensus tracking for second-order nonlinear multiagent systems with actuator faults". In: *IEEE Transactions on Cybernetics* 49.5 (2018), pp. 1605–1615.

[106] H. Jeong et al. "Fault detection and identification method using observer-based residuals". In: *Reliability Engineering & System Safety* 184 (2019), pp. 27–40.

[107] X. Liu et al. "Observer-based Actuator Fault Detection and Robust Tolerant Control for Vehicle Platoons". In: *International Journal of Control, Automation and Systems* (2023), pp. 1–12.

[108] Z. Zhou, M. Zhong, and Y. Wang. "Fault diagnosis observer and fault-tolerant control design for unmanned surface vehicles in network environments". In: *IEEE Access* 7 (2019), pp. 173694–173702.

[109] W. Wang et al. "Adaptive consensus of uncertain nonlinear systems with event triggered communication and intermittent actuator faults". In: *Automatica* 111 (2020), p. 108667.

[110] Z. Zhou et al. "Attack resilient control for vehicle platoon system with full states constraint under actuator faulty scenario". In: *Applied Mathematics and Computation* 419 (2022), p. 126874.

[111] Y. Lv et al. "Adaptive attack-free protocol for consensus tracking with pure relative output information". In: *Automatica* 117 (2020), p. 108998.

[112] A. Chakrabarty et al. "State and unknown input observers for nonlinear systems with bounded exogenous inputs". In: *IEEE Transactions on Automatic Control* 62.11 (2017), pp. 5497–5510.

[113] A. Chakrabarty et al. "State and unknown input observers for nonlinear systems with delayed measurements". In: *Automatica* 95 (2018), pp. 246–253.

[114] F. Xu et al. "A novel design of unknown input observers using set-theoretic methods for robust fault detection". In: *2016 American Control Conference (ACC)*. IEEE. 2016, pp. 5957–5961.

[115] L. Zuo, L. Yao, and Y. Kang. "UIO Based Sensor Fault Diagnosis and Compensation for Quadrotor UAV". In: *2020 Chinese Control And Decision Conference (CCDC)*. IEEE. 2020, pp. 4052–4057.

[116] A. F. Taha et al. "Unknown input observer design and analysis for networked control systems". In: *International Journal of Control* 88.5 (2015), pp. 920–934.

[117] I. Shames et al. "Distributed fault detection for interconnected second-order systems". In: *Automatica* 47.12 (2011), pp. 2757–2764.

[118] W. Chen and M. Saif. "Unknown input observer design for a class of nonlinear systems: an LMI approach". In: *2006 American Control Conference*. IEEE. 2006, pp. 834–838.

[119] A. Chakrabarty et al. "Distributed unknown input observers for interconnected nonlinear systems". In: *2016 American Control Conference (ACC)*. IEEE. 2016, pp. 101–106.

[120] L. Serrano et al. "A GPS velocity sensor: How accurate can it be?... A first look". In: *Proceedings of the 2004 National Technical Meeting of The Institute of Navigation*. 2004, pp. 875–885.

[121] M. Jibril, M. Tadesse, and E. Alemayehu. "State and disturbance estimation of a linear systems using proportional integral observer". In: *International Research Journal of Modernization in Engineering Technology and Science* 2.4 (2020).

[122] E. D. Sontag. "Mathematical control theory". In: *Texts in Applied Mathematics* (1998), p. 54.

[123] P. Zarchan. *Progress in astronautics and aeronautics: fundamentals of Kalman filtering: a practical approach*. Vol. 208. AIAA, 2005.

[124] I. T. Cameron and K. Hangos. *Process modelling and model analysis*. Elsevier, 2001.

[125] K. M. Hangos, J. Bokor, and G. Szederkényi. *Analysis and control of nonlinear process systems*. Vol. 13. Springer, 2004.

[126] D. Leitold, A. Vathy-Fogarassy, and J. Abonyi. *Network-Based Analysis of Dynamical Systems Methods for Controllability and Observability Analysis, and Optimal Sensor Placement*. Springer, 2020.

[127] C. Keliris, M. M. Polycarpou, and T. Parisini. "A robust nonlinear observer-based approach for distributed fault detection of input–output interconnected systems". In: *Automatica* 53 (2015), pp. 408–415.

[128] R. M. G. Ferrari, T. Parisini, and M. M. Polycarpou. "Distributed Fault Diagnosis With Overlapping Decompositions: An Adaptive Approximation Approach". In: *IEEE Transactions on Automatic Control* 54.4 (2009), pp. 794–799.

[129] X. Liu, D. Dong, and Y. Luo. "Fault diagnosis of train-ground wireless communication unit based on fuzzy neural network". In: *2009 4th IEEE Conference on Industrial Electronics and Applications*. IEEE. 2009, pp. 348–352.

[130] A. Sargolzaei et al. "A machine learning approach for fault detection in vehicular cyber-physical systems". In: *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE. 2016, pp. 636–640.

[131] S. Delrot et al. "Fouling detection in a heat exchanger by observer of Takagi–Sugeno type for systems with unknown polynomial inputs". In: *Engineering Applications of Artificial Intelligence* 25.8 (2012), pp. 1558–1566.

[132] X. Han et al. "Nonlinear Observer Based Fault Diagnosis for an Innovative Intensified Heat-Exchanger/Reactor". In: *Lecture notes in electrical engineering*. 2019, pp. 423–432.

[133] L. Manservigi et al. "A diagnostic approach for fault detection and identification in district heating networks". In: *Energy* 251 (2022), p. 123988.

[134] Q. Fan et al. "Two-Level Diagnosis of Heating Pipe Network Leakage Based on Deep Belief Network". In: *IEEE Access* 7 (2019), pp. 182983–182992.

[135] E. Weyer, G. Szederkényi, and K. M. Hangos. "Grey box fault detection of heat exchangers". In: *Control Engineering Practice* 8 (2000), pp. 121–131.

[136] W. Li, Y. Yan, and J. Bao. "Dissipativity-based distributed fault diagnosis for plantwide chemical processes". In: *Journal of Process Control* 96 (2020), pp. 37–48.

[137] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge University Press, 2012.

[138] K. J. Reinschke. *Multivariable control a graph-theoretic approach*. Springer, 1988.

[139] J.-M. Dion, C. Commault, and J. van der Woude. "Generic properties and control of linear structured systems". In: *IFAC Proceedings Volumes* 34.13 (2001), pp. 1–12.

[140] M. Pirani, A. Mitra, and S. Sundaram. "A survey of graph-theoretic approaches for analyzing the resilience of networked control systems". In: *arXiv preprint arXiv:2205.12498* (2022).

[141] B. Jiang and F. N. Chowdhury. "Parameter fault detection and estimation of a class of nonlinear systems using observers". In: *Journal of the Franklin Institute* 342.7 (2005), pp. 725–736.

[142] W. Levine. "The Control Handbook: Control System Fundamentals, Control System Applications, Control System Advanced Methods". In: *Electrical Engineering Handbook Series. Taylor & Francis Group* (2010).

[143] T. Kaczorek. *Positive 1D and 2D systems*. Springer Science & Business Media, 2012.

[144] R. Shorten, F. Wirth, and D. Leith. "A positive systems model of TCP-like congestion control: asymptotic results". In: *IEEE/ACM transactions on networking* 14.3 (2006), pp. 616–629.

[145] A. Hmamed et al. "Memoryless control to drive states of delayed continuous-time systems within the nonnegative orthant". In: *IFAC Proceedings Volumes* 41.2 (2008), pp. 3934–3939.

[146] A. Rantzer. "Scalable control of positive systems". In: *European Journal of Control* 24 (2015), pp. 72–80.

[147] J. Zhou and S Bennett. "Dynamic system fault diagnosis based on neural network modelling". In: *IFAC Proceedings Volumes* 30.18 (1997), pp. 55–60.

[148] H. H. Niemann and J. Stoustrup. "Robust fault detection in open loop vs. closed loop". In: *Proceedings of the 36th IEEE Conference on Decision and Control*. Vol. 5. IEEE. 1997, pp. 4496–4497.

[149] H. Niemann. "A setup for active fault diagnosis". In: *IEEE Transactions on Automatic Control* 51.9 (2006), pp. 1572–1578.

[150] A. E. Ashari, R. Nikoukhah, and S. L. Campbell. "Active Robust Fault Detection of Closed-Loop Systems: General Cost Cas". In: *IFAC Proceedings Volumes* 42.8 (2009), pp. 585–590.

[151] H. Niemann and N. K. Poulsen. "Active fault detection in MIMO systems". In: *2014 American Control Conference*. IEEE. 2014, pp. 1975–1980.

[152] H. Niemann and N. K. Poulsen. "Active fault isolation in MIMO systems". In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 8012–8017.

[153] M. Li et al. "A data-driven method for fault detection and isolation of the integrated energy-based district heating system". In: *IEEE Access* 8 (2020), pp. 23787–23801.

[154] D. B. Johnson. "Finding all the elementary circuits of a directed graph". In: *SIAM Journal on Computing* 4.1 (1975), pp. 77–84.

[155] J.-N. Juang and M. Q. Phan. *Identification and control of mechanical systems*. Cambridge University Press, 2001.

[156] A. Berman and R. J. Plemmons. *Nonnegative matrices in the mathematical sciences*. SIAM, 1994.

[157] C. Briat. "Sign properties of Metzler matrices with applications". In: *Linear Algebra and its Applications* 515 (2017), pp. 53–86.

[158] C. Cobelli and G. Romanin-Jacur. "Controllability, observability and structural identifiability of multi input and multi output biological compartmental systems". In: *IEEE Transactions on Biomedical Engineering* 2 (1976), pp. 93–100.

[159] L. Benvenuti and L. Farina. "Positive and compartmental systems". In: *IEEE Transactions on Automatic Control* 47.2 (2002), pp. 370–373.

[160] H. Smith. *An introduction to delay differential equations with applications to the life sciences*. Springer, 2011.

[161] B Krasznai, I Győri, and M. Pituk. "The modified chain method for a class of delay differential equations arising in neural networks". In: *Mathematical and computer modelling* 51.5-6 (2010), pp. 452–460.

[162] G. Lipták, M. Pituk, and K. M. Hangos. "Modelling and stability analysis of complex balanced kinetic systems with distributed time delays". In: *Journal of Process Control* 84 (2019), pp. 13–23.

[163] C.-T. Chen. *Linear system theory and design*. Saunders college publishing, 1984.

Appendix A

Linear time invariant model for fault diagnosis

Commonly, linear system models are obtained by linearising a nonlinear system model around its equilibrium points [124, 125]. Even though the instability may caused the system's states to move far away from the equilibrium points, a good controller hopefully can keep them in a small neighbourhood around the equilibriums where the linearisation still holds [3].

Consider a nonlinear system as follows:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \ , \ \boldsymbol{x}(\boldsymbol{t_0}) = \boldsymbol{x_0} \\ \boldsymbol{y} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) \end{aligned} \tag{A.1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the system's states, $\boldsymbol{x_0}$ is the states' initial conditions, $\boldsymbol{u} \in \mathbb{R}^r$ is the control input, and $\boldsymbol{y} \in \mathbb{R}^m$ is the measured system's output. $\boldsymbol{f}$ and $\boldsymbol{g}$ should be derivable.

By using Taylor series expansion, system in Eq (A.1) can be linearised for a small $\boldsymbol{\Delta x} = \boldsymbol{x} - \boldsymbol{x_{eq}}$ and $\boldsymbol{\Delta u} = \boldsymbol{u} - \boldsymbol{u_{eq}}$ around $(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})$ where $\boldsymbol{f}(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}}) = \boldsymbol{0}$. For small displacements around $(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})$, the higher order terms in the Taylor series expansion can be neglected because their values are small. For the lowest order terms, it can be represented by doing a partial derivative computed at $(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})$. This first order partial derivative gives the Jacobian matrix. Thus, the linearisation process can be written as:

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x_{eq}} + \boldsymbol{\Delta x}, \boldsymbol{u_{eq}} + \boldsymbol{\Delta u}) &= \boldsymbol{f}(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}}) + \underbrace{\left.\frac{\boldsymbol{df}}{\boldsymbol{dx}}\right|_{(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})}}_{A} \boldsymbol{\Delta x} + \underbrace{\left.\frac{\boldsymbol{df}}{\boldsymbol{du}}\right|_{(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})}}_{B} \boldsymbol{\Delta u} \\ \boldsymbol{g}(\boldsymbol{x_{eq}} + \boldsymbol{\Delta x}, \boldsymbol{u_{eq}} + \boldsymbol{\Delta u}) &= \boldsymbol{g}(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}}) + \underbrace{\left.\frac{\boldsymbol{dg}}{\boldsymbol{dx}}\right|_{(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})}}_{C} \boldsymbol{\Delta x} + \underbrace{\left.\frac{\boldsymbol{dg}}{\boldsymbol{du}}\right|_{(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})}}_{D} \boldsymbol{\Delta u} \end{aligned} \tag{A.2}$$

where A, B, C, and D are the Jacobian matrices with appropriate dimensions.

Then, by shifting $(\boldsymbol{x_{eq}}, \boldsymbol{u_{eq}})$ to the origin point and dropping Δ, the linearised model can be written in the form of LTI model as follows:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \ , \ \boldsymbol{x}(\boldsymbol{t_0}) = \boldsymbol{x_0} \\ \boldsymbol{y} &= C\boldsymbol{x} + D\boldsymbol{u} \end{aligned} \tag{A.3}$$

Eq (A.3) shows the general LTI system model in a state space form.

The following are two examples of particular LTI models: mechanical systems and lumped elements in energy systems.

A.1 Mechanical systems

In mechanical systems, the state vector usually contains the position states ($\boldsymbol{p}$) and the velocity states ($\boldsymbol{v}$) [155]. If the mechanical system has n Degrees of Freedom (DOF), then the dimension of the state vector is $2n$. Meanwhile, for the control inputs, generalized forces can be considered that affect directly the system's acceleration ($\dot{\boldsymbol{v}}$). The state matrix A also has a special structure due to the double integrator character of the dynamic model of mechanical systems. Thus, the model can be written as:

$$\begin{aligned} \boldsymbol{x} = \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{v} \end{bmatrix}, \begin{bmatrix} \dot{\boldsymbol{p}} \\ \dot{\boldsymbol{v}} \end{bmatrix} &= \begin{bmatrix} 0_{n\times n} & I_{n\times n} \\ A_1 & A_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{v} \end{bmatrix} + \begin{bmatrix} 0 \\ B \end{bmatrix} \boldsymbol{u} \\ \boldsymbol{y} &= C\boldsymbol{x} + D\boldsymbol{u} \end{aligned} \tag{A.4}$$

where 0 and I are the zero and identity matrices, respectively. A_1 and A_2 are matrices with appropriate dimensions.

As an example, consider an object in a two-dimensional (2D) space that moves in xy coordinate system where (x_0, y_0) is its initial condition and (v_x, v_y) is its velocity as shown in Fig A.1.

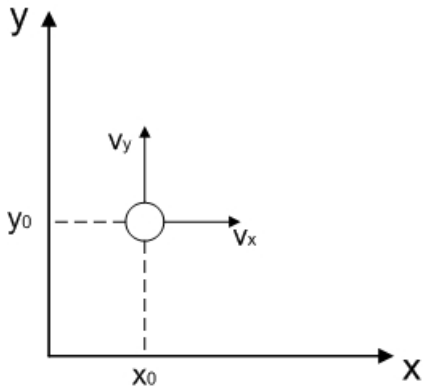

FIGURE A.1: A mechanical system example

Also, assume that the object's acceleration $\dot{\boldsymbol{v}}$ depends on its related position $\boldsymbol{p}$ by some stiffness coefficients, velocity $\boldsymbol{v}$ by some damping constants, and a control input u by some input gain constants. In addition, the measurement is done on the object's position. Then, similar to Eq (A.4), the model of the object's motion can be written as:

$$\begin{aligned} \begin{bmatrix} \dot{p_x} \\ \dot{p_y} \\ \dot{v_x} \\ \dot{v_y} \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ c_1 & 0 & c_2 & 0 \\ 0 & c_3 & 0 & c_4 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ c_5 \\ c_6 \end{bmatrix} u \\ \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix} \end{aligned} \tag{A.5}$$

where c_1 and c_3 are the stiffness coefficients, c_2 and c_4 are the damping constants, and c_5 and c_6 are the input gain constants.

A.2 Lumped elements in energy systems

Here, we are interested in energy systems that can be modelled as Lumped Parameter System (LPS). In this model, the system's state is the intensive variable (commonly temperature) which is assumed to be homogeneous for a single balance volume. Thus, it can be represented by a single state value in a function of time only. This leads to Ordinary Differential Equations (ODEs) representation in the system's model [124]. One example of such energy systems is distributed heating/cooling systems. Distributed heating/cooling systems transfer energy from one place to another using a heat transfer media, usually a fluid, transported in tubes and one tries to perfectly isolate the connecting tubes between the source and the consumer equipments. The distributed heating/cooling systems as process systems have three types of elements as operating units which are source elements, consumer elements, and connecting elements.

In the simplest case, one can construct the model of all of the above three system elements by spatially lumping the distributed engineering model derived from dynamic energy balance. Assume a tube as connecting element with a heat isolation wall containing incompressible liquid phase. The tube is well mixed in its cross section and has a spatially distributed temperature along its length. The state variables $\boldsymbol{x} \in \mathbb{R}^n$ will be the fluid temperatures x_i along the tube length (in the lumps) where $i = 1 \dots n$ represents the position of a flow element in the tube. The inlet temperature u is considered the system's input which acts only on the first state variable. The measured output of the system y is the outlet temperature which is the last state variable. A plug flow convection model is considered with flow-rate $v > 0$. Thus, the system's model can be written as:

$$\begin{bmatrix}\text{rate of change}\\ \text{of total energy}\end{bmatrix} = \begin{bmatrix}\text{flow of energy}\\ \text{into the system}\end{bmatrix} - \begin{bmatrix}\text{flow of energy}\\ \text{out of the system}\end{bmatrix}$$

$$\begin{aligned} \dot{x_1} &= -vx_1 + vu \\ \dot{x_i} &= -vx_i + vx_{i-1} \ , \ \text{for } i = 2, 3, ..., n \\ y &= x_n \end{aligned} \tag{A.6}$$

Eq (A.6) can be converted into an LTI state space model as follows:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\ \boldsymbol{y} &= C\boldsymbol{x} \end{aligned}$$

where

$$A = \begin{bmatrix} -v & 0 & ... & 0 \\ v & -v & ... & 0 \\ 0 & v & ... & 0 \\ 0 & ... & ... & 0 \\ 0 & ... & ... & -v \end{bmatrix}, \tag{A.7}$$

$$B = \begin{bmatrix} v & 0 & \dots & 0 \end{bmatrix}^T, \ C = \begin{bmatrix} 0 & 0 & 0 & ... & 1 \end{bmatrix}$$

By looking at Equation (A.7), we can derive some important points:

1. The model will always be stable because the eigenvalues are already stable shown by the negative sign of the diagonal entries of the A matrix [137].

2. The $A \in \mathbb{R}^{n \times n}$ matrix is a Metzler matrix by which all of its off-diagonal elements are non-negative. Thus, the matrix represents the time delayed differential equations and positive linear dynamical systems [156]. This is understandable because between the input and output states in the subsystem, a propagation inside the transport element will occur contributing to the increase in delayed processes. This also implies the stability of the model because this matrix is sign stable and Hurwitz [157].

3. It is jointly controllable and observable because it is derived from conservation balance equations (it is a compartmental system) [158, 159].

Appendix B

Networked systems

In networked systems where there are N connected subsystems, the states of each subsystem in the network are also affected by the states of its neighbouring subsystems [93] as follows:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(j)} &= A^{(j)}\boldsymbol{x}^{(j)} + B^{(j)}\boldsymbol{u}^{(j)} + I^{(j)}\boldsymbol{i}^{(j)} \\ \boldsymbol{y}^{(j)} &= C^{(j)}\boldsymbol{x}^{(j)} + D^{(j)}\boldsymbol{u}^{(j)} \end{aligned} \tag{B.1}$$

where $j = 1, 2, 3, ..., N$ represents the jth subsystem and $\boldsymbol{i}^{(j)}$ is the interconnection input which comes from the neighbouring subsystem's outputs. $I^{(j)}$ is an interconnection matrix with appropriate dimension.

By assuming static linear interconnections, the interconnection input $\boldsymbol{i}^{(j)}$ can be generally written as:

$$\boldsymbol{i}^{(j)} = L^{(jk)}\boldsymbol{y}^{(k)} \tag{B.2}$$

where $L^{(jk)}$ is called the adjacency matrix containing the relations between output measurements from the kth subsystem being connected as interconnection inputs to the jth subsystem.

Eqs (B.1) and (B.2) show the general model of networked system with static linear interconnection. However, through out this research, the connections in the networked system is expanded more details with distributed delay interconnections (using linear chain trick), joining and splitting connections, and loops.

B.1 Distributed delay interconnections

The interconnection between subsystems can be further generalized by allowing a delay kernel function g to describe interconnection with distributed time delay [160] as follows:

$$\begin{aligned} \boldsymbol{i}^{(j)} &= L^{(jk)}\left[\int_{-\tau}^{0} g^{(jk)}(s)\boldsymbol{y}^{(k)}(\boldsymbol{t}+\boldsymbol{s})\,ds\right] \\ &\int_{-\tau}^{0} g^{(jk)}(s)\,ds = 1 \end{aligned} \tag{B.3}$$

However, instead of trying to characterize the interconnection using a kernel function, a finite dimensional ODE realization through an LTI model can also be used to represent it. The basis of this approach is the linear chain trick method for approximating finite delays with a chain of linear reactions [161, 162]. Thus, from the perspective of linear chain trick, a linear interconnection with distributed time delay can equivalently be represented by a stable LTI system model. For example,

the model of lumped elements as in Eq (A.7) can represent the distributed delay interconnections in networked energy systems.

B.2 Splitting and joining connections

In a realistic network, it is common to have a branching phenomenon in the interconnection between subsystems. Here, the Kirchoff law applies to the flows of extensive variables entering and exiting this interconnection. Considering the ℓth junction, the sum of either the overall mass, component mass, or energy flows entering the junction is equal to the sum of flows exiting as shown in the following equation for the case of overall mass flows:

$$\sum_{\alpha \in IN^{(\ell)}} v^{(\alpha)} = \sum_{\beta \in OUT^{(\ell)}} v^{(\beta)} = V^{(\ell)} \tag{B.4}$$

where α runs over the branches by which the flow enters the interconnection. Meanwhile, β runs over those where the flows exits the interconnection. $IN^{(\ell)}$ is the input set of the interconnection junction point, $OUT^{(\ell)}$ is the output set of the interconnection junction point, and v is measured in $\frac{kg}{s}$.

Similar conservation equations apply for the energy flows $\dot{Q}$ in energy transport systems. However, to obtain the relations between the intensive temperature variable T among the subsystems in the networked energy systems, we substitute the algebraic equations representing the relationship between the intensive and extensive variables into the mass conservation as in Eq (B.4) [125]. The algebraic relationship between energy and temperature is in the following general form:

$$Q = Mc_P T \tag{B.5}$$

where Q is the thermal energy, M is the overall mass, and c_P is the specific heat in the balance volume.

Then, we can substitute relations in Eq (B.5) into the Kirchhoff law in Eq (B.4) while taking into account that the value of the intensive variable for all the outflows is the same. This way, one obtains the linear algebraic equations for the intensive temperature variable T at the ℓth junction in the following general form:

$$\sum_{\alpha \in IN^{(\ell)}} \frac{v^{(\alpha)}}{V^{(\ell)}} T^{(\alpha)} = T^{(\beta)} \ , \quad \sum_{\alpha \in IN^{(\ell)}} \frac{v^{(\alpha)}}{V^{(\ell)}} y^{(\alpha)} = u^{(\beta)} \ , \ \forall\, \beta \in OUT^{(\ell)} \tag{B.6}$$

B.3 Loops

For the sake of energy or material efficiency, it can be happen that a networked system contains loops. However, to ease the analysis of the networked system in the presence of loops, we transform the general LTI model from its state space form as in Eq (A.3) into transfer function in the Laplace domain s [3] as follows:

$$y(s) = S(s)u(s) \tag{B.7}$$

where $u(s)$ and $y(s)$ is the Laplace transform from the input $u(t)$ and $y(t)$ respectively, and $S(s)$ is the transfer function from input $u(s)$ to output $y(s)$.

By assuming zero initial conditions, the transfer function $S(s)$ can be obtained from the related state space matrices:

$$S(s) = C(sI - A)^{-1}B + D \tag{B.8}$$

Now, consider many interconnected subsystems forming a loop where $S^{(j)}(s)$ is the jth subsystem's transfer function and $j = 1, 2, ..., l-1, l, l+1, ..., m$ as shown in Fig B.1.

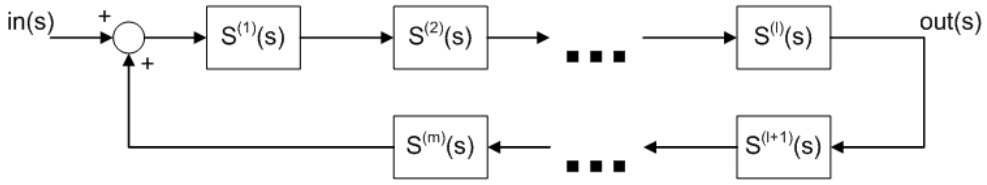

FIGURE B.1: Block diagram of a closed loop

By using block diagram analysis, the overall transfer function from the considered input $in(s)$ to the considered output $out(s)$ is:

$$\frac{out(s)}{in(s)} = \frac{\prod_{j=1}^{l} S^{(j)}(s)}{1 - \prod_{j=1}^{m} S^{(j)}(s)} \tag{B.9}$$

Appendix C

Linear observers

In the real world applications, not all of the system's states can be directly measured using the available sensors. Thus, it is important to investigate the observability of the system. A system is called observable if it is possible to construct or estimate the system's states from available measurements [3, 125].

C.1 State estimation

Consider a system as in Eq (A.3) where the states are $\boldsymbol{x} \in \mathbb{R}^n$. Define the observability matrix $\mathcal{O}$ as:

$$\mathcal{O} = \begin{bmatrix} C & CA & CA^2 & ... & CA^{n-1} \end{bmatrix}^T \tag{C.1}$$

If the rank of the matrix $\mathcal{O}$ is n, then the system is observable [125].

Now, assume that the pair (A, C) in Eq (A.3) is already observable. Then, a state observer can be constructed for such system. A state observer is an estimator that can construct or estimate the given real system's internal states from its available input and output measurements [3]. The model of the estimator is:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= A\hat{\boldsymbol{x}} + B\boldsymbol{u} + L(\boldsymbol{y} - \hat{\boldsymbol{y}}) \\ \hat{\boldsymbol{y}} &= C\hat{\boldsymbol{x}} + D\boldsymbol{u} \end{aligned} \tag{C.2}$$

where $\hat{\boldsymbol{x}}$ is the states estimation and L is the observer's gain.

System in Eq (C.2) is a linear observer.

From Eqs (A.3) and (C.2), the state estimation error $\boldsymbol{\epsilon} = \boldsymbol{x} - \hat{\boldsymbol{x}}$ can be obtained as follows:

$$\begin{aligned} \dot{\boldsymbol{\epsilon}} &= \dot{\boldsymbol{x}} - \dot{\hat{\boldsymbol{x}}} \\ &= (A\boldsymbol{x} + B\boldsymbol{u}) - ((A - LC)\hat{\boldsymbol{x}} + L\boldsymbol{y} + (B - LD)\boldsymbol{u}) \\ \dot{\boldsymbol{\epsilon}} &= (A - LC)\boldsymbol{\epsilon} \end{aligned} \tag{C.3}$$

Eq (C.3) shows that if we can choose L so that $A - LC$ is stable (or Hurwitz), then $\lim_{t\to\infty} \boldsymbol{\epsilon} = \boldsymbol{0}$. In other words, the observer successfully estimates the real system's states. There are two common methods to determine the observer's gain L which are pole placement design and Linear Quadratic Estimator (LQE).

Note that the system's observability is a duality problem with the system's controllability because they are mathematically similar.

C.2 Pole placement design

Pole placement design is originated from Full State Feedback (FSF) which is commonly used in control theory to place the poles (or eigenvalues) of a closed loop

system in desired locations. By placing the poles in the desired locations, we can modify the characteristics of the system's responses [3]. To apply this method, the system must be controllable (or observable in the case of state estimation).

From Eq (C.3), the poles can be determined from the characteristic equation of $A - LC$:

$$det[sI - (A - LC)] = 0 \tag{C.4}$$

where I is the identity matrix.

If the system is observable, Eq (C.4) shows that the values of L can be used to forcefully place the poles of $A - LC$ to some locations related to the desired characteristic responses of the state estimation.

C.3 Linear quadratic estimator design

Consider that system (A.3) is now affected by disturbance and measurement noise as follows:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} + \boldsymbol{d} \\ \boldsymbol{y} &= C\boldsymbol{x} + D\boldsymbol{u} + \boldsymbol{n} \end{aligned} \tag{C.5}$$

where $\boldsymbol{d}$ is the disturbance and $\boldsymbol{n}$ is the measurement noise.

LQE can optimally compensate the effect of disturbance and measurement noise in estimating the system's states. However, it must be assumed that both disturbance and measurement noise are zero mean Gaussian processes with known disturbance variance W_d and measurement noise variance W_n [3].

As observability and controllability are dual properties, LQE is also dual with Linear Quadratic Regulator (LQR) optimization.

LQE deals with how to choose a suitable observer gain L in estimating the system's states in the presence of disturbance and measurement noises. To do that, define a state estimation error covariance $\mathcal{Y}$ that satisfies:

$$\begin{aligned} &\mathcal{Y} \succ 0 \\ &\dot{\mathcal{Y}} = \mathcal{Y}A^T + A\mathcal{Y} - \mathcal{Y}C^T W_n^{-1} C\mathcal{Y} + W_d \end{aligned} \tag{C.6}$$

Therefore, by choosing $\mathcal{Y}$ to solve the following algebraic Riccati equation:

$$0 = \mathcal{Y}A^T + A\mathcal{Y} - \mathcal{Y}C^T W_n^{-1} C\mathcal{Y} + W_d \tag{C.7}$$

then, the error covariance $\mathcal{Y}$ will diminish in a steady state value ($\dot{\mathcal{Y}} = 0$).

The optimal gain L can be computed as:

$$L = \mathcal{Y}C^T W_n^{-1} \tag{C.8}$$

Remark: In this research, instead of manual computation, the author used the MATLAB commands *place* and *lqe* to compute the necessary observer's gain for pole placement and LQE methods, respectively.

C.4 Proportional-Integral observer

Consider an LTI system affected by unknown constant disturbance as follows:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} + E\boldsymbol{d} \\ \boldsymbol{y} &= C\boldsymbol{x} \end{aligned} \tag{C.9}$$

where $\boldsymbol{d}$ is the unknown constant disturbance and E is disturbance distribution matrix with appropriate dimension.

Assume that the states for the system as in Eq (C.9) are being estimated by the following linear observer:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= A\hat{\boldsymbol{x}} + B\boldsymbol{u} + L(\boldsymbol{y} - \hat{\boldsymbol{y}}) \\ \hat{\boldsymbol{y}} &= C\hat{\boldsymbol{x}} \end{aligned} \tag{C.10}$$

Then, the state estimation error dynamics will be:

$$\dot{\boldsymbol{\epsilon}} = \dot{\boldsymbol{x}} - \dot{\hat{\boldsymbol{x}}} = (A - LC)\boldsymbol{\epsilon} + E\boldsymbol{d} \tag{C.11}$$

As long as $A - LC$ is Hurwitz, Eq (C.11) shows that there will always be a constant steady state error if $\boldsymbol{d} \neq 0$.

Fortunately, a Proportional Integral (PI) observer has the ability to eliminate this steady state error [121]. Moreover, this PI observer can also be applied to estimate the disturbance by considering it as an additional state for the system in Eq (C.9) so that the model becomes an extended state space model as follows:

$$\begin{aligned} \boldsymbol{z} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{d} \end{bmatrix} \quad , \quad \dot{\boldsymbol{z}} &= \begin{bmatrix} A & E \\ 0 & 0 \end{bmatrix} \boldsymbol{z} + \begin{bmatrix} B \\ 0 \end{bmatrix} \boldsymbol{u} \\ \boldsymbol{y} &= \begin{bmatrix} C & 0 \end{bmatrix} \boldsymbol{z} \end{aligned} \tag{C.12}$$

Then, for system in Eq (C.12), a PI observer can be designed to estimate both the states and disturbance. Hence, a PI observer is sometimes called a disturbance observer. The model for the PI observer is:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= A\hat{\boldsymbol{x}} + B\boldsymbol{u} + E\hat{\boldsymbol{d}} + L_P(\boldsymbol{y} - \hat{\boldsymbol{y}}) \\ \hat{\boldsymbol{y}} &= C\hat{\boldsymbol{x}} \\ \dot{\hat{\boldsymbol{d}}} &= L_I(\boldsymbol{y} - \hat{\boldsymbol{y}}) \end{aligned} \tag{C.13}$$

where $\hat{\boldsymbol{d}}$ is the disturbance estimate, L_P is the observer's proportional gain matrix, and L_I is the observer's integral gain matrix.

By further derivation, Eq (C.13) can also be written as:

$$\begin{aligned} \dot{\hat{\boldsymbol{z}}} &= \left(\begin{bmatrix} A & E \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} L_P \\ L_I \end{bmatrix} \begin{bmatrix} C & 0 \end{bmatrix} \right) \hat{\boldsymbol{z}} + \begin{bmatrix} L_P \\ L_I \end{bmatrix} \boldsymbol{y} + \begin{bmatrix} B \\ 0 \end{bmatrix} \boldsymbol{u} \\ &= (A_z - L_z C_z)\, \hat{\boldsymbol{z}} + L_z \boldsymbol{y} + B_z \boldsymbol{u} \end{aligned} \tag{C.14}$$

The state estimation error dynamics for PI observer is:

$$\dot{\boldsymbol{\epsilon}}_{\boldsymbol{z}} = \dot{\boldsymbol{z}} - \dot{\hat{\boldsymbol{z}}} = (A_z - L_z C_z)\boldsymbol{\epsilon}_{\boldsymbol{z}} \tag{C.15}$$

Thus, if $L_z = [L_P \; L_I]^T$ is chosen such that $A_z - L_z C_z$ is Hurwitz, then $\lim_{t\to\infty} \boldsymbol{\epsilon}_{\boldsymbol{z}} = \boldsymbol{0}$. In other words, the PI observer will successfully estimate both the system's states

$\boldsymbol{x}$ and the unknown constant disturbance $\boldsymbol{d}$.

C.5 Unknown input observer

Here, consider again an LTI system with unknown disturbance as in Eq (C.9).

An Unknown Input Observer (UIO) is a special observer that produces a state estimation $\hat{\boldsymbol{x}}$ that converges to the real one if certain conditions hold despite of the presence of some unknown input $\boldsymbol{d}$ [17]. The state space model for the UIO is:

$$\begin{aligned} \dot{\boldsymbol{z}} &= F\boldsymbol{z} + TB\boldsymbol{u} + K\boldsymbol{y} \\ \hat{\boldsymbol{x}} &= \boldsymbol{z} + H\boldsymbol{y} \\ \boldsymbol{r} &= \boldsymbol{y} - C\hat{\boldsymbol{x}} = (I - CH)\boldsymbol{y} - C\boldsymbol{z} \end{aligned} \tag{C.16}$$

where I is the identity matrix, $\boldsymbol{z}$ is the UIO's internal states, $\hat{\boldsymbol{x}}$ is the states estimation, and $\boldsymbol{r}$ is the UIO's residual signal.

If the matrices F, T, K, and H with appropriate dimensions are chosen such that they satisfy:

$$\begin{aligned} HCE &= E \\ T &= I - HC \\ F &= A - HCA - K_1C \\ K_2 &= FH \\ K &= K_1 + K_2 \end{aligned} \tag{C.17}$$

where K_1 is chosen to stabilize F, then $\lim_{t\to\infty}(\boldsymbol{x} - \hat{\boldsymbol{x}}) = \boldsymbol{0}$. In this situation, $\boldsymbol{r} = \boldsymbol{0}$ also applies.

The necessary conditions so that Eq (C.17) can be satisfied are:

$$\begin{aligned} &1.\ rank(CE) = rank(E) \\ &2.\ (C, A - HCA)\ \textit{is a detectable pair} \end{aligned} \tag{C.18}$$

The first condition is to ensure that H matrix in Eq (C.17) exists. It also indicates that the number of available independent measurements limits the maximum size of the unknown input that can be decoupled. Meanwhile, the second condition is a milder condition related to observability. If $(C, A - HCA)$ is already observable, a pole placement design method can directly be used to determine the value of K_1. If it is not observable, an observable canonical decomposition procedure must be performed to do a matrix transformation before using pole placement to find the value of K_1. The details of this decomposition procedure can be seen in [163].

Appendix D

Parameter fault detection and estimation of a class of nonlinear systems using observers

This section presents the main results from [141]. See the related article for further details and mathematical proofs.

Consider a nonlinear system as follows:

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + \boldsymbol{g}(\boldsymbol{u}, \boldsymbol{y}) + B\theta \boldsymbol{f}(\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{x}) \\ \boldsymbol{y} &= C\boldsymbol{x} \end{aligned} \tag{D.1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the state, $\boldsymbol{u} \in \mathbb{R}^m$ is the input, and $\boldsymbol{y} \in \mathbb{R}^r$ is the output. The pair (A, C) is observable. $\boldsymbol{g}(\boldsymbol{u}, \boldsymbol{y})$ is a nonlinear term that depends on the available $\boldsymbol{u}$ and $\boldsymbol{y}$. $\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{x}) \in \mathbb{R}^r$ is a nonlinear vector function. $\theta \in \mathbb{R}$ is a parameter that changes when a fault occurs and it is bounded, i.e. $\|\theta\| \leq \theta_0$. $\theta = \theta_H$ when there is no fault where θ_H is a known scalar.

In addition, assume that:

1. A known positive constant L_0 exists such that for any norm bounded $\boldsymbol{x_1}, \boldsymbol{x_2} \in \mathbb{R}^n$, the following inequality holds (Lipschitz condition):

$$\|\boldsymbol{f}(\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{x_1}) - \boldsymbol{f}(\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{x_2})\| \leq L_0 \|\boldsymbol{x_1} - \boldsymbol{x_2}\| \tag{D.2}$$

2. For a given positive definite matrix $Q > 0 \in \mathbb{R}^{n \times n}$, a matrix $P = P^T > 0 \in \mathbb{R}^{n \times n}$ and a scalar R exist such that:

$$\begin{aligned} (A - KC)^T P + P(A - KC) &= -Q \\ PB &= C^T R \end{aligned} \tag{D.3}$$

Consider the following linear observer:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= A\hat{\boldsymbol{x}} + \boldsymbol{g}(\boldsymbol{u}, \boldsymbol{y}) + B\theta_H \boldsymbol{f}(\boldsymbol{u}, \boldsymbol{y}, \hat{\boldsymbol{x}}) + K(\boldsymbol{y} - \hat{\boldsymbol{y}}) \\ \hat{\boldsymbol{y}} &= C\hat{\boldsymbol{x}} \end{aligned} \tag{D.4}$$

Also, define the following errors:

$$\begin{aligned} \boldsymbol{e_x} &= \boldsymbol{x} - \hat{\boldsymbol{x}} \\ \boldsymbol{e_y} &= \boldsymbol{y} - \hat{\boldsymbol{y}} \\ e_\theta &= \theta_f - \hat{\theta} \end{aligned} \tag{D.5}$$

where θ_f is the changed parameter after the fault occurred and satisfies $\theta_f \neq \theta_H$, $\|\theta_f\| \leq \theta_0$. $\hat{\theta}$ is the estimate of θ_f.

Then, for system described by Eq (D.1) under the assumptions (D.2) and (D.3), the linear observer described by Eq (D.4) can be used to detect the fault as $\lim_{t\to\infty} \boldsymbol{e_x(t)} = \mathbf{0}$ and $\lim_{t\to\infty} \boldsymbol{e_y(t)} = \mathbf{0}$ when $\theta = \theta_H$ (fault free condition).

Now, consider the following nonlinear observer:

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= A\hat{\boldsymbol{x}} + \boldsymbol{g}(\boldsymbol{u},\boldsymbol{y}) + B\hat{\theta}\boldsymbol{f}(\boldsymbol{u},\boldsymbol{y},\hat{\boldsymbol{x}}) + K\boldsymbol{e_y} \\ \hat{\boldsymbol{y}} &= C\hat{\boldsymbol{x}} \\ \dot{\hat{\theta}} &= \Gamma \boldsymbol{f}^T(\boldsymbol{u},\boldsymbol{y},\hat{\boldsymbol{x}})R\boldsymbol{e_y} \end{aligned} \tag{D.6}$$

where $\Gamma > 0$ is a weighting scalar and R is given by Eq (D.3).

Then, under a persistent excitation, the nonlinear observer described by Eq (D.6) can be used to estimate $\hat{\theta}$ as $\lim_{t\to\infty} e_\theta(t) = 0$ within a bounded e_θ and can also ensure that $\lim_{t\to\infty} \boldsymbol{e_x(t)} = \mathbf{0}$.

Appendix E

A possible way to generalize networked fault isolation in MIMO systems

Consider a network of N interconnected subsystems where the state space model for each subsystem is:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(j)} &= A\boldsymbol{x}^{(j)} + B\boldsymbol{u}^{(j)} \\ \boldsymbol{y}^{(j)} &= C\boldsymbol{x}^{(j)} + D\boldsymbol{u}^{(j)} \end{aligned} \tag{E.1}$$

where $j = 1, 2, 3...N$ represents the jt subsystem, $\boldsymbol{x}^{(j)} \in \mathbb{R}^{n_j}$ is the subsystem's states, $\boldsymbol{u}^{(j)} \in \mathbb{R}^{m_j}$ is the subsystem's input, and $\boldsymbol{y}^{(j)} \in \mathbb{R}^{p_j}$ is the measured subsystem's output.

The input has the form of:

$$\boldsymbol{u}^{(j)} = \boldsymbol{w}^{(j)} + \boldsymbol{f}^{(j)} + \boldsymbol{i}^{(j)} \tag{E.2}$$

where $\boldsymbol{w}^{(j)}$ is the known subsystem's local input, $\boldsymbol{f}^{(j)}$ is the possible fault input in the subsystem ($\boldsymbol{f}^{(j)} = \boldsymbol{0}$ when there is no fault), and $\boldsymbol{i}^{(j)}$ is the subsystem's interconnection input.

Moreover, the interconnections among the subsystems can be described by the following interconnection matrix:

$$\begin{bmatrix} i^{(1)} \\ i^{(2)} \\ i^{(3)} \\ . \\ . \\ . \\ i^{(j)} \end{bmatrix} = \begin{bmatrix} 0 & L_2^{(1)} & L_3^{(1)} & . & . & . & L_j^{(1)} \\ L_1^{(2)} & 0 & L_3^{(2)} & . & . & . & L_j^{(2)} \\ L_1^{(3)} & L_2^{(3)} & 0 & . & . & . & L_j^{(3)} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ L_1^{(j)} & L_2^{(j)} & L_3^{(j)} & . & . & . & 0 \end{bmatrix} \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ . \\ . \\ . \\ y^{(j)} \end{bmatrix} \tag{E.3}$$

where $L_b^{(a)}$ are the interconnection matrices with appropriate dimensions that relate the interconnection input of the ath subsystem and the output of the bth subsystem.

Instead of state space representation, the relation between the subsystem's input and output can also be presented in the form of a transfer function matrix $G(s) = C(sI - A)^{-1}B + D$, where I is the identity matrix with appropriate dimension, so that $y^{(j)}(s) = G^{(j)}(s)u^{(j)}(s)$.

For the sake of convenience, the complex variable s will be neglected from here on (i.e., $G^{(j)}(s)$ will simply be written as $G^{(j)}$ and so will the others).

A chain of subsystems. Consider several subsystems in series connections. To derive its model, define:

$$\begin{aligned} u_{i/l}^{(j)} &= u^{(j)} - L_l^{(j)} y^{(l)} \\ i_{i/l}^{(j)} &= i^{(j)} - L_l^{(j)} y^{(l)} \end{aligned} \tag{E.4}$$

Example: Fig E.1 shows the chain of 3 interconnected subsystems.

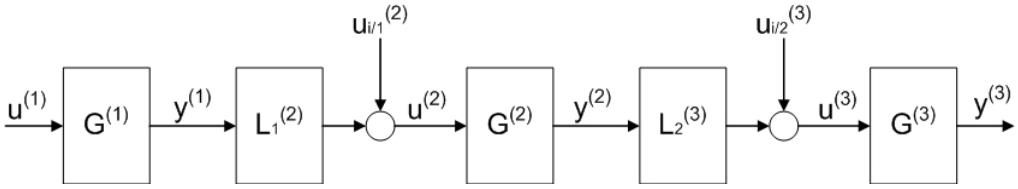

FIGURE E.1: An example of a chain of subsystems

By using the superposition principle, the output can be obtained as follows:

$$\begin{aligned} y^{(3)} &= G^{(3)}\left[(w^{(3)} + f^{(3)} + i_{i/2}^{(3)} + L_2^{(3)}G^{(2)}(w^{(2)} + f^{(2)} + i_{i/1}^{(2)} + L_1^{(2)}G^{(1)}(w^{(1)} + f^{(1)} + i^{(1)}))\right] \\ y^{(3)} &= G^{(3)}w^{(3)} + G^{(3)}L_2^{(3)}G^{(2)}w^{(2)} + G^{(3)}L_2^{(3)}G^{(2)}L_1^{(2)}G^{(1)}w^{(1)} \\ &\quad + G^{(3)}f^{(3)} + G^{(3)}L_2^{(3)}G^{(2)}f^{(2)} + G^{(3)}L_2^{(3)}G^{(2)}L_1^{(2)}G^{(1)}f^{(1)} \\ &\quad + G^{(3)}i_{i/2}^{(3)} + G^{(3)}L_2^{(3)}G^{(2)}i_{i/1}^{(2)} + G^{(3)}L_2^{(3)}G^{(2)}L_1^{(2)}G^{(1)}i^{(1)} \end{aligned} \tag{E.5}$$

Thus, based on Eq (E.5), the generalization for a chain of k interconnected subsystems is:

$$y^{(k)} = G^{(k)}\left(w^{(C)} + f^{(C)} + i^{(C)}\right) + \underbrace{G^{(k)}L_{k-1}^{(k)}\cdots L_2^{(3)}G^{(2)}L_1^{(2)}G^{(1)}}_{G^{(C)}} i^{(1)}$$

where

$$\begin{aligned} w^{(C)} &= w^{(3)} + L_{k-1}^{(k)}G^{(k-1)}w^{(k-1)} + \cdots + L_{k-1}^{(k)}G^{(k-1)}\cdots L_2^{(3)}G^{(2)}w^{(2)} \\ &\quad + L_{k-1}^{(k)}G^{(k-1)}\cdots L_1^{(2)}G^{(1)}w^{(1)} \\ f^{(C)} &= f^{(3)} + L_{k-1}^{(k)}G^{(k-1)}f^{(k-1)} + \cdots + L_{k-1}^{(k)}G^{(k-1)}\cdots L_2^{(3)}G^{(2)}f^{(2)} \\ &\quad + L_{k-1}^{(k)}G^{(k-1)}\cdots L_1^{(2)}G^{(1)}f^{(1)} \\ i^{(C)} &= i_{i/(k-1)}^{(k)} + L_{k-1}^{(k)}G^{(k-1)}i_{i/(k-2)}^{(k-1)} + \cdots + L_{k-1}^{(k)}G^{(k-1)}\cdots L_3^{(4)}G^{(3)}i_{i/2}^{(3)} \\ &\quad + L_{k-1}^{(k)}G^{(k-1)}\cdots L_2^{(3)}G^{(2)}i_{i/1}^{(2)} \end{aligned} \tag{E.6}$$

Loops in the network. Now, consider a loop as shown in Fig E.2 that is formed from a subsystem $G^{(1)}$ and a chain of interconnected subsystems $G^{(C)}$ where the last subsystem of the chain is the kth subsystem.

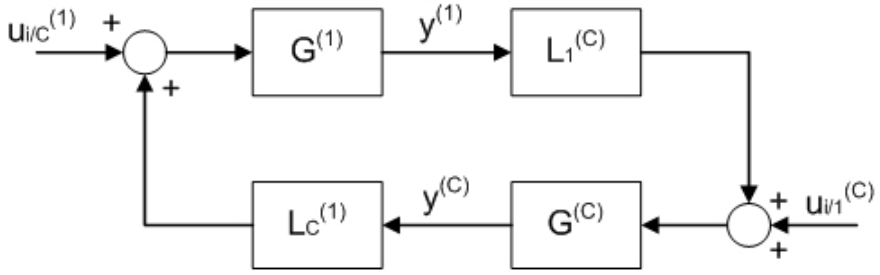

FIGURE E.2: A subsystem and a chain of subsystems forming a loop

Then, by using the same derivation as in Eq (E.5), we can write:

$$\begin{aligned}
y^{(1)} &= G^{(1)}w^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}w^{(C)} + G^{(1)}f^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}f^{(C)} \\
&\quad + G^{(1)}i_{i/C}^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}i_{i/1}^{(C)} + G^{(1)}L_C^{(1)}G^{(k)}L_1^{(C)}y^{(1)} \\
(I - G^{(1)}L_C^{(1)}G^{(k)}L_1^{(C)})y^{(1)} &= G^{(1)}f^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}f^{(C)} \\
&\quad + \underbrace{G^{(1)}w^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}w^{(C)} + G^{(1)}i_{i/C}^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}i_{i/1}^{(C)}}_{p^{(1)}} \\
y^{(C)} &= G^{(k)}L_1^{(C)}G^{(1)}w^{(1)} + G^{(k)}w^{(C)} + G^{(k)}L_1^{(C)}G^{(1)}f^{(1)} + G^{(k)}f^{(C)} \\
&\quad + G^{(k)}L_1^{(C)}G^{(1)}i_{i/C}^{(1)} + G^{(k)}i_{i/1}^{(C)} + G^{(k)}L_1^{(C)}G^{(1)}L_C^{(1)}y^{(C)} \\
(I - G^{(k)}L_1^{(C)}G^{(1)}L_C^{(1)})y^{(C)} &= G^{(k)}L_1^{(C)}G^{(1)}f^{(1)} + G^{(k)}f^{(C)} \\
&\quad + \underbrace{G^{(k)}L_1^{(C)}G^{(1)}w^{(1)} + G^{(k)}w^{(C)} + G^{(k)}L_1^{(C)}G^{(1)}i_{i/C}^{(1)} + G^{(k)}i_{i/1}^{(C)}}_{p^{(C)}}
\end{aligned} \tag{E.7}$$

Note that, $p^{(1)}$ and $p^{(C)}$ can be computed if the transfer functions and the input signals are known.

Weak fault isolation problem statement: given a loop as shown in Fig E.2, determine whether a fault occurred in $G^{(1)}$ or in $G^{(C)}$.

Assume that:

$$\begin{aligned}
&1.\ \text{There are no simultaneous faults in the loop.} \\
&2.\ \frac{\|G^{(k)}L_1^{(C)}x\|}{\|x\|} < 1 \text{ and } \frac{\|G^{(1)}L_C^{(1)}x\|}{\|x\|} < 1 \text{ for } \forall x \neq 0.
\end{aligned} \tag{E.8}$$

where $\|\cdot\|$ denotes the L_2 vector signal norm.

Now, define the residuals as follows:

$$\begin{aligned}
r^{(1)} &= (I - G^{(1)}L_C^{(1)}G^{(k)}L_1^{(C)})y^{(1)} - p^{(1)} \\
r^{(C)} &= (I - G^{(k)}L_1^{(C)}G^{(1)}L_C^{(1)})y^{(C)} - p^{(C)}
\end{aligned} \tag{E.9}$$

Then, by combining Eqs (E.7) and (E.9), we get:

$$\begin{aligned}
&\text{for } \|r^{(1)}\| - \|r^{(C)}\| = \|G^{(1)}f^{(1)} + G^{(1)}L_C^{(1)}G^{(k)}f^{(C)}\| - \|G^{(k)}L_1^{(C)}G^{(1)}f^{(1)} + G^{(k)}f^{(C)}\|, \\
&\text{if } f^{(1)} = f^{(C)} = 0, \text{ then } \|r^{(1)}\| - \|r^{(c)}\| = 0, \\
&\text{if } f^{(1)} \neq 0 \text{ and } f^{(C)} = 0, \text{ then } \|r^{(1)}\| - \|r^{(C)}\| = \|G^{(1)}f^{(1)}\| - \|G^{(k)}L_1^{(C)}G^{(1)}f^{(1)}\|, \\
&\text{if } f^{(1)} = 0 \text{ and } f^{(C)} \neq 0, \text{ then } \|r^{(1)}\| - \|r^{(C)}\| = \|G^{(1)}L_C^{(1)}G^{(k)}f^{(C)}\| - \|G^{(k)}f^{(C)}\|
\end{aligned} \tag{E.10}$$

Furthermore, by Assumptions (E.8), the following relations can be obtained:

$$\begin{aligned} &1.\ \|r^{(1)}\| - \|r^{(C)}\| > 0 \text{ when } f^{(1)} \neq 0 \text{ and } f^{(C)} = 0 \\ &2.\ \|r^{(1)}\| - \|r^{(C)}\| < 0 \text{ when } f^{(1)} = 0 \text{ and } f^{(C)} \neq 0 \end{aligned} \tag{E.11}$$

Thus, Eq (E.11) can be applied to solve the weak fault isolation problem.