



Flinders
UNIVERSITY

Fault Detection and Diagnosis for Vibration Isolation System Using Parity Relation Technique

by

Bo Sun, M.Eng. (Electronics)

Supervisor: Fangpo He

Date of submission: May, 2018

Submitted to the College of Science and Engineering in partial fulfilment of the requirements for the degree of master of engineering (electronics) at Flinders University – Adelaide Australia

Declaration

I certify that this thesis does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief, does not contain any material previously published or written by another person except where due reference is made in the text.

August, 2018

Acknowledgement

To begin with, I would like to express my gratitude to my supervisor, Associate Doctor-Fangpo He, who continued to encourage me to overcome various difficulties within the past one year and teach me how to self-adjust when feeling frustrated and how to study effectively and scientifically. I learned not only academic knowledge from her, but also her work spirit and her attitude to life.

In addition, the selfless supports from my family help me to pursuit my study all the time, especially my wife who takes the responsibility to raise my family and provides endless inspiration. Besides, the love from my daughter and other family members spur me to greater efforts.

Lastly, many staffs of the college supported and helped me a lot during my study period. The PHD student Michael Zhang made great contribution to my study. His generous support helped me success to get through the hardest period quickly at the beginning stage. Overall, I am grateful to all of them for their constant support.

Abstract

In modern industries, most systems are constructed by multiple components with different functionalities. Only when all components function as they are designed to be, the system can perform effectively. Once a fault exists in the system, it may cause productivity deterioration or even system failure. The goal of this project is to fulfil fault detection and diagnosis (FDD) for additive faults on a vibration isolation system. In this project, FDD processes based on the parity relation technique for a multiple-input multiple-output (MIMO) vibration control of a mechanical structure is designed and examined.

A top plate bonded with three feet on a base plate, which forms a three-input three-output system, is used in this project to demonstrate the design of the FDD process. The control goal is to keep the top plate vibration free while the base plate is shaken consistently by a disturbance signal. By introducing different fault cases into the system, the control performance of the previously designed positive position feedback may not provide the best control results. Therefore, identifying when and where the fault occurs becomes essential for the fault accommodation.

A system identification study is conducted to experimentally obtain the mathematical model of the plant (i.e., the top plate together with the sensors and actuators attached to the plate). The transfer function matrix of the plant is then acquired through curve fitting techniques. Based on the design criteria of parity relation, the transfer function matrix is used to produce the state-space representation (SSR) of the plant that is further converted into the discrete-time form.

FDD process is a significant feature in fault tolerant control architecture. The FDD processes designed in this thesis are based on the parity relation technique, which is a popular mode-based fault detection technique that uses the residual as a fault indicator. When a discrete system is fault free, the designed residual of the system should be stable and close to zero in value. However, when a fault is introduced into the system, the residual will alter to indicate that there exists error(s) in the system. The core part of the parity relation technique is to design a parity vector which can eliminate the effects of the unknown state $\mathbf{X}(t)$. The corresponding residuals can reflect whether the system works in a fault free condition or not. To detect fault, firstly one will need to define an order s of the parity space, which affects the performance of the FDD process. Secondly, the parity vector \mathbf{V}_s in this parity space will be designed based on the condition: $\mathbf{V}_s * \mathbf{H} = 0$, where \mathbf{H} is a deterministic

matrix associated with the state $\mathbf{X}(t)$. The designed residual $\mathbf{R}(t)$ is calculated based on the measurable system input $\mathbf{U}(t)$ and output $\mathbf{Y}(t)$, and is not affected by the state $\mathbf{X}(t)$.

Using the parity relation technique, FDD processes respectively for a SISO system case and a MIMO system case are designed and validated through MATLAB Simulink. Firstly, a random system is chosen and is injected with a pre-defined single fault. Simulation studies are carried out to show that the parity residual value corresponding to the fault indeed changes from zero to non-zero, which validates the design principle of the parity relation technique. Secondly, the designed parity relation method is implemented to the plant model. Simulation results show that the used parity relation technique is able to theoretically detect pre-defined faults for the plant.

Upon the validation of the used parity space technique, an actuator-and-sensor fault isolation is implemented through designing different parity vectors \mathbf{V}_s . A set of structured residuals are generated where each residual is sensitive to all faults apart from one corresponding fault. The corresponding fault will then be located through a designed fault-code table. To carry out the fault isolation examinations, studies on isolating individual sensor faults among the three sensors of the plant are first conducted, and the corresponding scenarios are implemented in MATLAB Simulink. Simulation results show that the set of designed residuals has an excellent performance in detecting single sensor faults in the real plate system. A similar procedure is arranged for the isolation of individual actuator faults among the three actuators of the plant. However, simulation results reveal that the effect of the actuator fault isolation is not as clear as that of the sensor fault isolation. Due to the existing physical properties of the given plant, the parity vectors designed respectively for Actuator 2 fault and Actuator 3 fault of the given plant appear to be very close to each other. This leads to the difficulties in clearly distinguishing the responses of these two residuals. It is anticipated that further modification of the designed parity vectors that can take into account the existing physical properties of the given plant may produce a more reliable FDD process applicable to all types of individual actuator faults effectively.

In conclusion, the studies carried out in this thesis have proven that in principle, the parity relation technique can be implemented on a MIMO vibration cancellation system successfully for additive faults' detection and diagnosis. Comparing with other methods, the implementation of the FDD processes using the parity relation technique is simple and flexible. Using the proposed FDD processes, the sensor or actuator fault occurrences of the given plant can be effectively detected. Furthermore, the individual fault among the three sensors of the given plant can be diagnosed

successively. Although the proposed method can only provide partial individual fault diagnoses among the three actuators of the system due to the physical properties of the given plant, the general concept of the design is verified via a specific actuator fault scenario. Future work includes the modification of the proposed FDD process design method to enable it to diagnose all individual actuator faults successfully, and to be physically implemented to the real laboratory experimental rig of the given plant.

List of Figures

Figure 1.1: Laboratory model used in this vibration cancellation project.....	3
Figure 1.2: Structure of fault tolerant control [5].....	4
Figure 2.1: The main schematic of model-based fault diagnosis.....	9
Figure 3.1: The hardware connection of the whole system	13
Figure 3.2: The block diagram of the laboratory model in state space representation	22
Figure 3.3: Step responses of continuous system and discrete system	25
Figure 4.1: Simulink model of a random SISO system for FDD validation	34
Figure 4.2: Residual response in fault-free SISO system.....	35
Figure 4.3: Residual response with sensor fault in SISO system.....	35
Figure 4.4: Simulink model of a MIMO system for FDD validation.....	36
Figure 4.5: Residual responses without sensor fault in MIMO.....	37
Figure 4.6: Residual responses with sensor 1 fault in MIMO	37
Figure 4.7: Residual responses with sensor 2 fault in MIMO	38
Figure 5.1: The design of FDD processes in Simulink.....	39
Figure 5.2: Sensor fault signal design.....	40
Figure 5.3: Inner structure of data memory block.....	40
Figure 5.4: Case 1 - The responses of residual R_1 for sensor fault	42
Figure 5.5: Case 1 - The responses of residual R_2 for sensor fault	43
Figure 5.6: Case 1 - The responses of residual R_3 for sensor fault	43
Figure 5.7: Case 1 - The responses of residual R_d for sensor fault	44
Figure 5.8: Case 2 - The responses of residual R_1 for sensor fault.....	45
Figure 5.9: Case 2 - The responses of residual R_2 for sensor fault.....	45
Figure 5.10: Case 2 - The responses of residual R_3 for sensor fault.....	46
Figure 5.11: Case 2 - The responses of residual R_d for sensor fault.....	46
Figure 5.12: Case 3 - The responses of residual R_1 for sensor fault	47
Figure 5.13: Case 3 - The responses of residual R_2 for sensor fault	48
Figure 5.14: Case 3 - The responses of residual R_3 for sensor fault	48
Figure 5.15: Case 3 - The responses of residual R_d for sensor fault.....	49
Figure 5.16: Case 4 - The responses of residual R_1 for sensor fault	50
Figure 5.17: Case 4 - The responses of residual R_2 for sensor fault	50
Figure 5.18: Case 4 - The responses of residual R_3 for sensor fault	51
Figure 5.19: Case 4 - The responses of residual R_d for sensor fault.....	51
Figure 5.20: Fault information when no fault exists in the system.	53
Figure 5.21: Fault information when a fault exists in sensor 1.....	53
Figure 5.22: Fault information when a fault exists in sensor 2.....	54
Figure 5.23: Fault information when a fault exists in sensor 3.....	54
Figure 5.24: Simulation model with actuator fault	55
Figure 5.25: Case 1 - The responses of residual R_1 for actuator fault	56
Figure 5.26: Case 1 - The responses of residual R_2 for actuator fault	57
Figure 5.27: Case 1 - The responses of residual R_3 for actuator fault	57
Figure 5.28: Case 1 - The responses of residual R_d for actuator fault.....	58
Figure 5.29: Case 2 - The responses of residual R_1 for actuator fault.....	59
Figure 5.30: Case 2 - The responses of residual R_2 for actuator fault.....	59
Figure 5.31: Case 2 - The responses of residual R_3 for actuator fault.....	60
Figure 5.32: Case 2 - The responses of residual R_d for actuator fault.....	60

Figure 5.33: Case 3 - The responses of residual R_1 for actuator fault	61
Figure 5.34: Case 3 - The responses of residual R_2 for actuator fault	62
Figure 5.35: Case 3 - The responses of residual R_3 for actuator fault	62
Figure 5.36: Case 3 - The responses of residual R_d for actuator fault	63

List of Tables

Table 3.1: The data of mode 1 for all transfer functions.....	23
Table 3.2: The data of mode 2 for all transfer functions.....	23
Table 3.3: The data of mode 3 for all transfer functions.....	23
Table 5.1: Designed parity vectors for sensor fault detection and isolation.....	41
Table 5.2: Designed parity vectors for actuator fault detection and isolation.....	55

Contents

Declaration.....	ii
Acknowledgement	iii
Abstract.....	iv
List of Figures	vii
List of Tables	ix
Chapter 1 : Introduction	1
1.1 Motivation.....	1
1.2 Vibration Control.....	1
1.3 Laboratory Model Specification.....	2
1.4 Methodology.....	3
1.5 Thesis Outline.....	4
Chapter 2 : Literature Review	6
2.1 Fault	6
2.2 Fault Detection History	7
2.3 Hardware Redundancy and Software Redundancy	7
2.4 Model-based Methods.....	8
2.4.1 Kalman Filter	9
2.4.2 BJ Filter	10
2.4.3 Parity Relation.....	10
Chapter 3 : System Identification.....	12
3.1 Laboratory Plate Structure Setup	12
3.2 Multivariable System Identification.....	13
3.3 State-space Representation Derivation	14
3.4 State-space Representation Calculation	22
Chapter 4 : Parity Relation Technique	26
4.1 Model structure	26
4.2 Fault detection	27
4.3 Design of parity space order s	29
4.4 Fault Isolation Design.....	30
4.4.1 Residual Sensitivity	30
4.4.2 Structured Residual Design	32
4.5 FDD Process Design and Validation	33

4.5.1 Example: SISO System FDD Design and Validation	34
4.5.2 Example: MIMO System FDD Design and Validation	36
Chapter 5 : Simulation	39
5.1 Simulation Setup with Sensor Fault	39
5.2 Sensor Fault Simulation Results	42
5.2.1 Simulation Results of Sensor Fault for Case 1: No Fault	42
5.2.2 Simulation Results of Sensor Fault for Case 2: Sensor 1 Fault	44
5.2.3 Simulation Results of Sensor Fault for Case 3: Sensor 2 Fault	46
5.2.4 Simulation Results of Sensor Fault for Case 4: Sensor 3 Fault	49
5.3 Simulation Results of Decision-making Block	51
5.3 Simulation Setup with Actuator Fault	55
5.4 Actuator Fault Simulation Results	55
5.4.1 Simulation Results of Actuator Fault for Case 1: No Fault	56
5.4.2 Simulation Results of Actuator Fault for Case 2: Actuator 1 Fault	58
5.4.3 Simulation Results of Actuator Fault for Case 3: Actuator 2 Fault	60
5.5 Simulation Summary	63
Chapter 6 : Conclusions and Future Works	64
6.1 Project Conclusion	64
6.2 Future Works	65
Appendix A: MATLAB Code for SISO System Validation	66
Appendix B: MATLAB Code for MIMO System Validation	67
Appendix C: MATLAB Code for Residual Generation	68
Appendix D: MATLAB Code for Fault Isolation Block	69
Bibliography	70

Chapter 1 : Introduction

The goal of this project is to implement fault detection and diagnosis (FDD) for additive faults on a vibration isolation system. In order to achieve the goal, the brief introduction of the study is concluded. This chapter briefly states the motivation of the study and provides a basic knowledge of the vibration isolation system. Furthermore, the methodology of fault detection and diagnosis is introduced. Finally, the outline of this thesis is summarized.

1.1 Motivation

Safety is the priority for numerous projects, especially for those systems with high-risk or relating to human's life, such as public transportation including aviation, railway and shipping, military equipment and aerospace industry. Sometimes, a tiny fault may lead to irretrievable consequences, such as enormous economic cost and severe casualties. It is well known that on July 4, 1996, the Ariane 5 rocket exploded after launching. The investigative result showed the reason of this aviation accident is that faulty Inertial Reference Unit produced incorrect measured data to the control system [1].

Therefore, the main motivation of the study about fault detection and diagnosis is to avoid the negative effects of faults and keep property and human safe. With the development of technique, fault detection (FD) methods also evolve to suit the requirements of modern industries, from simple hardware redundancies to complex analytical redundancies, from slow detection to fast detection and from few fields to various industries.

Consequently, FDD plays a crucial role in modern industrial field. This study aims to design a fault detection and diagnosis (FDD) scheme using parity relation technique on a vibration cancellation system which has been widely used in many industries and is introduced in next section.

1.2 Vibration Control

Vibration widely exists in nature environment, and all the objects are vibrating with specific frequencies. According to this character, people invented various musical instruments, whereas vibration also brings many demerits. In many early studies, a number of scientists interested on the resonance phenomenon as it resulted in negative effects on human development. One of the most well-known accident caused by resonance is the collapse of Tacoma Narrows Bridge in 1940 [2]. Besides, modern industries require high-precise control systems to produce accurate products. However, the extensive unwanted vibration always makes system difficult or unable to achieve its

Chapter 1: Introduction

goal and even results in system failures, therefore, a vibration-free platform is required. Hence, to overcome the influence of vibration, vibration control and vibration cancellation attract many engineers' interest. Furthermore, vibration isolation systems are widely used in many fields of engineering, particular in the field of earthquake engineering to lessen seismic impacts on buildings and other structures.

The topic of vibration control usually involves passive control, active control and hybrid control that means both passive control and active control are used together. Initially, passive control is to utilize the natural characters of the materials and some specific structures to absorb the energy of vibration and reduces the damage brought by external force to the system, such as vibration dampers and absorbers [3]. An obvious disadvantage of passive control is that it has low- frequency resonance problem and another problem is that it needs a system with large size and huge weight, which does not match the requirements of plenty of modern industries[3]. However, active control can provide a force that is equal but opposite to the force imposed by external vibration [4]. Active vibration control can cancel vibration at any frequency and has a relatively shorter setting time. Besides, it is easy to install and duplicate for any project, and the relatively smaller size and weight makes it use in some projects flexibly with constrains on space and weight.

This study concerns on vibration cancellation using active control, a laboratory model is designed to reduce the external vibration through three actuators. The details of this study are illustrated in next Section 1.3.

1.3 Laboratory Model Specification

A real plate structure is constructed to imitate a vibration cancellation platform in Flinders control laboratory. As shown in Figure 1.1, the laboratory model is comprised of a base plate, a top plate, three accelerometers (namely, $S1$, $S2$, $S3$ respectively), four transducers (namely, $T1$, $T2$, $T3$ and $T4$ respectively). The first three transducers act as actuators to keep the top plate vibration-free and the $T4$ generates external disturbance to vibrate the whole structure. On the base plate, the four transducers are positioned at predefined location and the top plate is fixed on top of the first three transducers by screws. On the top plate, three accelerometers are mounted at the same positions with $T1$, $T2$ and $T3$. After applying a variant periodic voltage to the fourth actuator, the whole structure including the top plate will vibrate.

The goal of this structure is to eliminate the vibration of the top plate and keep it vibration-free while the external disturbance persistently exists. However, once one sensor or actuator cannot work properly, the control performance will deteriorate, or even destabilizes the system.

Chapter 1: Introduction

Hence, when a fault occurs, in order to maintain the system works successfully, a fault detection and diagnosis (FDD) subsystem should find the fault and identify the type and location of the fault. Then, the current controller is replaced by a new controller which can eliminate the negative effect of the fault and keep the system working at an acceptable performance. The main scope of this study is to implement FDD processes using on this vibration cancellation system. The knowledge of FDD is introduced in the next section.

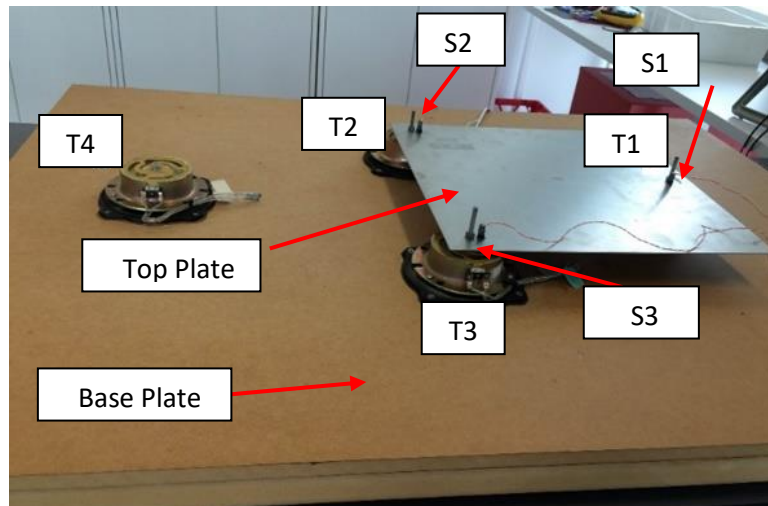


Figure 1.1: Laboratory model used in this vibration cancellation project

1.4 Methodology

In order to eliminate the negative effects of a fault, it is required that faults have to be detected as quickly as possible and the corresponding measures should be taken to correct the faults and maintain the system work successfully. This is known as fault tolerant control which usually contains two tasks [5]:

- Fault detection and diagnosis: The aim of the task is to detect whether a system has any fault and further identify the fault.
- Fault accommodation: Upon the information obtained in first step, an adjusted controller will replace the old one in this step so that the system cannot be affected by faults and work properly.

The structure of fault tolerant control of an open loop system is described in Figure1.2. On the supervision level, the block 'Diagnosis' and the block 'Fault Accommodation' stand for the two tasks of fault tolerant control respectively. The diagnostic block utilizes the mathematical model and the information of known input and output to detect a fault and identify the fault, and then sends the fault information to fault accommodation block. This block will modify the controller parameters or

choose pre-designed controller from controller library according to the fault information [5]. On the execution level, when the system is fault-free, the nominal controller can ensure the system meet the requirements of engineers and minimize the effect of disturbance at the same time. If a fault appears in the system, it will cause the system state differ from the fault-free state. Then, the diagnostic block will identify the fault and the fault accommodation block can provide a new controller to eliminate the impact of the fault on the system and maintain the system with satisfied performance [5].

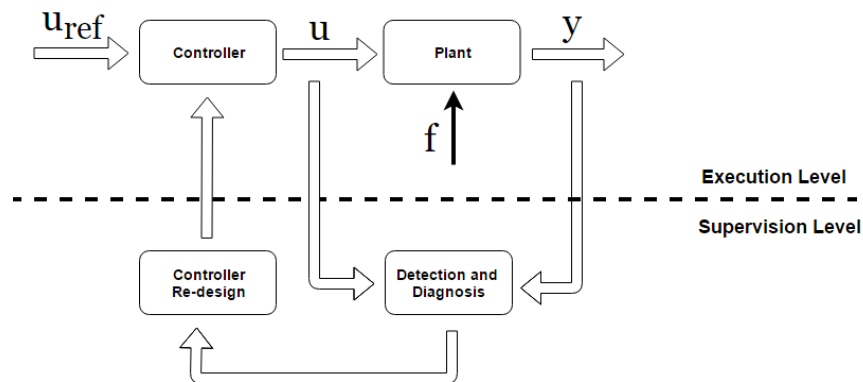


Figure 1.2: Structure of fault tolerant control [5].

1.5 Thesis Outline

In this thesis, the fault detection and diagnosis processes for a vibration cancellation system are realized via parity relation technique. The thesis outline is as follows:

In Chapter 2, an overall literature review about fault detection and diagnosis is given. The history of techniques used for fault detecting and diagnosis are summarized. The advantages and disadvantages of both hardware redundancy and software redundancy are illustrated. Furthermore, some important mode-based techniques, such as Kalman filter, BJ filter and parity relation are introduced and compared, including their merits and demerits, applications and so on.

In Chapter 3, the physical structure and vibration cancellation system setup are described and the aim of system identification is achieved, especially the state space representation which plays an important role in implementation of fault detection and isolation.

In Chapter 4, the design principles of FDD using parity relation method are presented. Firstly, fault detection method is illustrated, and then the selection of parity space order is analyzed. Lastly, validations of parity relation technique on SISO and MIMO systems are designed and evaluated separately via using MATLAB Simulink.

Chapter 1: Introduction

In Chapter 5, the simulation are proposed regarding to sensor fault and actuator fault. The simulation results are demonstrated and the performance of FDD using parity relation method is analysed and discussed.

In Chapter 6, the conclusion of this project is drawn. The problems encountered in this study are analysed and summarized. Besides, some advisable suggestions about future work are presented.

Chapter 2 : Literature Review

This chapter critically reviews contemporary literature concerning fault detection and diagnosis (FDD). Sometimes, it can be ignored that a minor fault occurred in one component. However, it may result in irreversible damage to the whole system if it accumulates or propagates in the system. This is why FDD becomes so considerable. Therefore, many researchers concentrate on developing FDD and figure out many methods or application in various fields. These methods can be basically summarized into two categories: hardware redundancy and software redundancy (also known as analytical redundancy). Firstly, the history of fault detection (FD) is introduced. Then, the features of hardware redundancy and software redundancy are outlined and compared with each other. Finally, this chapter will review several popular methods of analytical redundancy and their applications.

2.1 Fault

In modern society, the increasing enterprises require large-scale technological processes to produce highly-integrated products or high-precision products [5]. To achieve these purposes, these production lines must be equipped many different components and subsystems that are designed to fulfil certain tasks. Obviously, the availability and reliability of all the components and subsystems will determine the performance of the whole production line. This can be illustrated by numerous examples. Electricity plays a crucial role in people's daily life as well as in industrial production. The power grid has millions of components which transform and distribute electricity to match the requirements of users, whereas, a single fault may lead to major effects on wide areas, even result in damage to electric appliances. Similarly, in manufactory, many different machines and tools collaborate to produce qualified products, and strict regulations are implemented to prevent faults damaging the machines and humans. Overall, a fault can be defined as something unwanted that causes negative impacts on a system, so that the system cannot achieve its goal [6].

Generally, faults are classified as additive faults and multiplicative faults in a time-varying system. Additive faults are considered as an unknown input of the system and do not change the original system model. The influences of additive faults are added into the system through the system model. Usually, additive faults contain sensor faults, actuator faults and some component faults and these types of faults just alter the system responses. However, multiplicative faults will change the system parameters which multiply with the inputs or system state variables [7]. For instant, the structure of a system is changed or the mass of a physical plant alters, which are thought as multiplicative faults. Moreover, disturbances and model errors act on a system like faults and lead to similar effects. In common, disturbances can be represented as unknown inputs added on the

system, but model errors can change the system parameters and act on the system like multiplicative faults. For this study, only sensor fault and actuator fault are analysed due to the limitation of time and the requirements of this project.

2.2 Fault Detection History

The concept of fault detection and isolation was widely used in computer field, large-scale integrated circuits and system control fields to detect and deal with logic fault in the 1970s' [8]. Take the aviation as an example, some early studies in the area generally mentioned that in order to get the improved performance of airplane, such as dragging reduction and saving fuel, the more complex and accurate control technique including wings control and failure detection, were researched [9].

Meanwhile, many methods are not suitable for the rapidly developing control system. For instance, one of the traditional fault indication methods usually checks the limit of some important variables or analysis of particular signals, and it sets a threshold value regarding tolerances of the normal value [10]. If the value monitored exceeds the pre-set threshold, the system will indicate a fault happened. However, by using this method, some fault likes abrupt change or accumulated fault cannot be found immediately. Meanwhile, with the development of computer science and state space theory, the analytical redundancy method attracted more attention and became the main stream in 1980s' and developed with two branches: mode-based techniques and signal-based techniques (or data-based techniques)[11].

2.3 Hardware Redundancy and Software Redundancy

In engineering, hardware redundancy means to duplicate or triplicate some crucial components or functions to prevent a system failure and improve the performance [12]. This technique is mostly used in many safety-critical systems, such as emergency stop system, uninterruptible power supply and aircraft control system. Obviously, hardware redundancy has perfect reliability. Comparing to software redundancy (introduced later), hardware redundancy is naturally more stable, and it is the last-wall to protect the system and avoid further destruction in many life-relevant applications. Besides, it can be implemented and maintained easily and quickly, which is relatively suitable for some large projects or critically important subsystems.

Based on many practical experiences, engineers gradually found that due to the constraints of space, weight and cost, hardware redundancy could not achieve the requirements of investigators in small-scale projects. Since 1980s', with the emergence of advanced control system theory, the software redundancy (or analytical redundancy) has replaced the position of hardware redundancy in many

projects and become the main stream [13, 14]. Compared with hardware redundancy, analytical redundancy has been more widely used in different industries as it has many advantages:

- Cost-competition: It dramatically reduces the cost of projects.
- Flexibility: It is easy to duplicate and applied to other similar projects. Besides, evolution of current project is convenient to be implemented.
- Wide applications: It is the best choice when hardware redundancy unavailable or restrained.

Similarly, analytical redundancy also has some demerits. Initially, it is vulnerable to environmental noise or disturbance. Furthermore, it is based on the information of plants, so the modelling error or measure error should not be ignored [13]. For some complex system, robustness and sensitivity are hardly controlled, so there is a trade-off between the performance and stability. Hence, according to the system requirement and practical limits, different methods are suitable for different projects. In the next section, some methods will be involved and analysed.

2.4 Model-based Methods

Since 1980s', advanced control theory has brought powerful techniques for mathematical modelling and system identification, with progresses of computer technology. For mode-based technique, the base of this technique is to use an precise mathematical model to replace the physical plant [11]. Figure 2.1 depicts the main schematic of model-based fault diagnosis. The system consists of four parts: Actuator, Process, Sensor and Controller. The involved faults include actuator fault f_a , process fault f_c and sensor fault f_s . The diagnosis algorithm is to carry out a residual by using input u and output y and check the consistency of the residual of the real-time process against the prediction of a healthy system [13], and make the decision upon the diagnosis result. In the next three subsections, three main model-based methods are introduced and compared; especially the development of parity relation technique is highlighted.

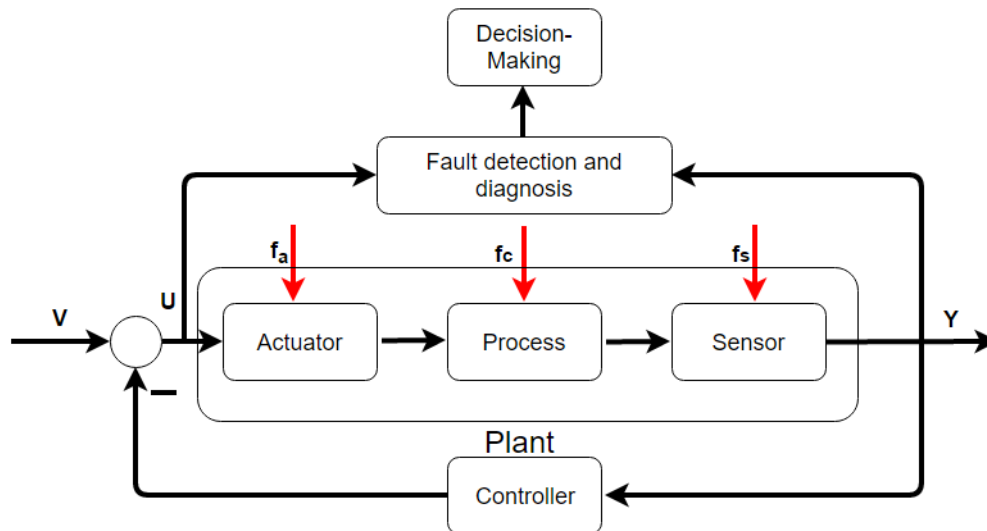


Figure 2.1: The main schematic of model-based fault diagnosis (adapted from [7]). V is the reference command, U is the input signal and Y is the measured output signal. f_a , f_c and f_s are actuator fault signal, component fault signal and sensor fault signal, respectively.

2.4.1 Kalman Filter

Kalman Filter, an excellent linear filter, has a numerous application in science such as navigation, vehicle control and signal process, and can be used to detect a variety of fault in linear system, especially for the sensor fault. Besides, Kalman filter has been developed into a number of kinds: Extended Kalman Filter, Unscented Kalman Filter and Hybrid Kalman Filter, which tremendously extend the applications of Kalman filter. For instance, NASA's Advanced Detection, Isolation, and Accommodation (ADIA) Program investigated how to use a bank of Kalman filter to deal with sensor failure [15]. In [16], Takahisa. K and Donald. L.S used a set of Kalman filters to detect and isolate aircraft gas turbine engine sensor and actuator fault and components faults. They also applied hybrid Kalman filter for aircraft engine in-flight diagnosis [17].

It is common that Kalman filters and its family members have played a crucial role in linear and non-linear estimation, and the aircraft and spacecraft control. However, some flaws should be pointed out in the practical utilities. Firstly, Kalman filter is based on the linear transfer function of the control system, and it requires the precise mathematic model of the plant to predict system states. This requirement strictly restrains Kalman filter's application, particularly for some systems that the transfer functions are not available. Secondly, a basic step of Kalman filter operation is to introduce a Gaussian noise variable to approximate the real trajectory. During updating, this can result in inevitable errors and worse performance [18].

2.4.2 BJ Filter

The BJ filter, also known as Beard-Jones Detection Filter, is one kind of diagnostic observer. Beard [19] and Jones [20] respectively described a scheme for system fault detection in 1970s. This detection filter originates from Luenberger observer and can detect and isolate the occurrence of faults in systems via imposing a directional constraint on the residual. In [21, 22] a geometric interpretation and a spectral approach of BJ filter are developed respectively. BJDF is extended to uncertain system in [23]. Besides, [24] presents two FD observer design methods to achieve multiple and simultaneous fault isolation and improve robustness to structural uncertainties. The residuals can fulfil disturbance decoupling and output direction to pre-defined fault. For Gaussian white noise affected system, [25] shown detailed analysis of noise response and proposed an optimal procedure using the error covariance.

2.4.3 Parity Relation

As one of the main implement of analytical redundancy, parity relation directly reflects the basic concept of analytical redundancy. The residuals are the difference between the real output of physical system and the prediction of system model. In general, the residuals are zero or a near zero pattern in the absence of fault and nonzero if a fault occurs. Through mathematic transformation the researchers can find out residual generator and finally carry out the fault detection and isolation properties [26].

The early studies show that the parity relation technique is closely related with aerospace industry, using transfer function and state space forms [26]. These ideas described by several author and their colleagues [8, 9, 27]. Gertler made a great contribution in fault detection and isolation with analytical redundancy. Firstly, Gertler in 1988 in [11] outlined some normal techniques to the problem of FDD, such as limit checking, installation of special sensors, physical redundancy, frequency analysis and Expert system approach, and presented the general structure of model-based methods. More importantly, he discussed three different ways of generating residual, input-output residuals, state-space residuals and identification-based methods. Finally, he stated the importance of isolation, sensitivity and robustness, and how to improve the sensitivity and robustness. After that, Gertler in [27] developed the parity equation-based FDD method, in which parity relation method was explained particularly with equations and some examples. This method can be applied to detect single additive fault with some robustness, however, for parametric fault or multiple fault, it is useless and the efficiency also needs to be improved, some research should be done to deal with complex situation. In 1997, Gertler shown the equal relation between parity relation and residual generator that parity relation can be used for describing any linear residual

Chapter 2: Literature Review

generator, and residual enhancement for fault isolation that includes three schemes: diagonal residuals, directional residuals and structured residuals, and successfully extended from single fault to multiple fault. In this paper, parity relation method is respectively used for detecting additive faults and parametric faults[26]. The latter utility is also known as “Chow-Willsky scheme”, which is relied on state-space system description and firstly revealed by Chow and Willsky in [28]. Similarly, [29] compared the residuals based on parity relation and residuals based on observer and stated that both of them are constructed in the same way.

Besides, parity relation FDD technique has been developed into many branches to improve detection performance or extend its applications. [30] derived the characteristics of all parity vectors, constructed a relationship between the dimension of the parity space and the order of the parity relations, and developed a Matlab toolbox which can prove the fact that the large order of parity relations may increase the reliability and robustness. [31] presented the methodology of parity relations FDD on linear parameter varying system and demonstrated an example based on a vehicle lateral dynamics model. [32] proposed a fault detection scheme using parity space technique for LDTV system, and shown the heavy computational burdens if the parity space order becomes large. Finally, through Krein space projection, the computation cost is reduced via introducing recursions.

Chapter 3 : System Identification

In this chapter, the method to model the physical system mathematically is described. Firstly, experimental setup of the laboratory structure is introduced. Then ModalVIEW, a modal analysis software, is used to model the real system via recording the frequency response function (FRF) of the plate as well as the parameters of the transfer functions, including natural frequencies, damping ratios, and mode shapes¹. Lastly, for the purpose of the FDD design, the transfer function matrix of the MIMO system is converted into state-space representation (SSR) systematically.

3.1 Laboratory Plate Structure Setup

The physical model used for this study has been introduced in Chapter 1. This section states the connections of every component in order to model the system mathematically. The hardware connection of the whole system is described in Figure 3.1[34].

The details of the main components in Figure 3.1 are listed as follows:

- NI DAQ module: The DAQ 9234 module is utilized to acquire vibration signals on the plate structure. Channel 0 is linked with input exciting signal and Channel 1 to Channel 3 are connected to 3 response signals of the three transducers respectively through accelerometers (model number: 353B17). The Software ModalVIEW uses all the signals from four channels to generate frequency response function (FRF) curve of the three transducers [35].
- Signal generator: It constantly provides an amplified sinusoidal signal to the disturbance transducer that result to vibration of the structure. To mimic a real external disturbance, the sine signal has fixed magnitude, but variable frequency that increases from 20 Hz to 60Hz within 10 seconds. When modelling the structure through ModalVIEW, it also provides the same signal to each transducer as input signal. The details are discussed in Section 3.3.

¹ Mode shape: In vibration analysis, vibration can be describes by combination of modes. A normal mode of a system is a pattern of motion. Modes are determined by the physical property of a structure. A mode of vibration can be described by a modal frequency and a mode shape. [33] R. D. Blevins and R. Plunkett, "Formulas for natural frequency and mode shape," *Journal of Applied Mechanics*, vol. 47, p. 461, 1980.

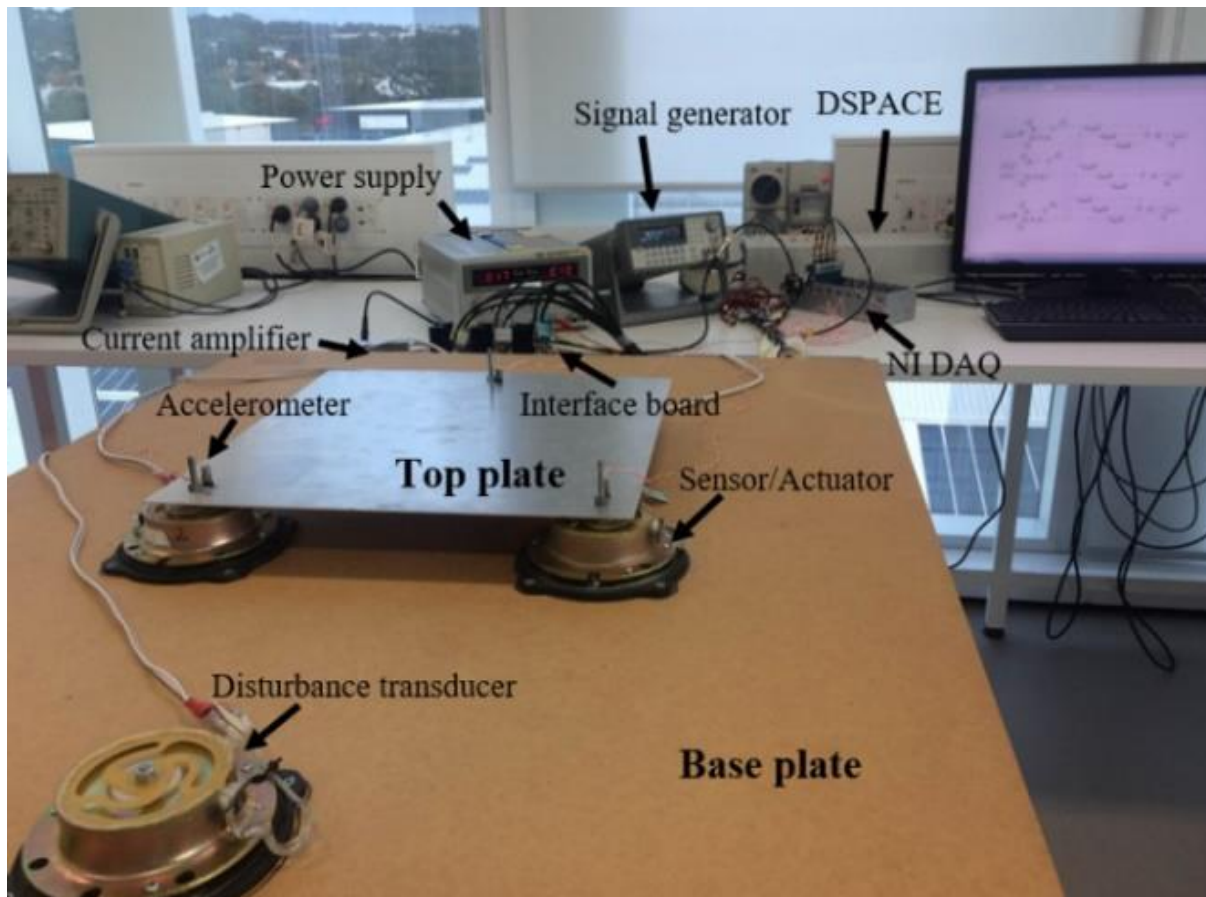


Figure 3.1: The hardware connection of the whole system

- Current amplifier: As the output signals from accelerometers are too small, the current amplifier filters and enlarges the accelerometer signals 100 times in order to avoid losing accuracy.
- Interface board: the interface board connects the transducers with the A/D modules and D/A modules of dSPACE together. The function of the interface board is to isolate the induced voltage of transducers before measuring and ensure they can be measure accurately.
- dSPACE: the abbreviation of 'Digital Signal Processing and Control Engineering'. It is a powerful tool used in engineering. It includes the software dSPACE and the hardware DS1103 PPC controller board used in this study. DS1103 PPC controller board contains a set of Analog-to-Digital (A/D) and Digital-to-Analog (D/A) channels, digital I/O ports, and a serial interface.

3.2 Multivariable System Identification

The mechanical structure used in this study can be represented by a multi-input multi-output (MIMO) system, actually a three-input three-output system. However, to control the system

precisely, the mathematical model of a physical system is required, which can monitor and predict the status of the physical system conveniently [36].

In [37], the author presented a numerical analysis method with the software ANSYS and MATLAB, and carried out the transfer function through ANSYS modal analysis. Nevertheless, the accuracy of the model obtained through this method is not good enough because of limitation on the accuracy of the software. This project uses the physical experiment method to figure out the mathematical model. As described in Chapter 1, the software ModalVIEW is used to drive the open-loop transfer function in this study, which can quickly obtain the accurate natural frequency, damping ratio and mode shape of each mode.

ModalVIEW generates the frequency response functions (FRF) of the system for each pair of input-output signals through physical experiment. Then, a multiple-degree-of-freedom polynomial curve fitting method is applied to each of the measured FRF curves to calculate the corresponding transfer function [35]. After that, the acquired transfer functions are combined to form the system transfer matrix $\mathbf{G}(s)$, which is shown in Equation (3.1). Reference [35] precisely stated the procedure how to implement through ModalVIEW and lay solid foundation for further research.

3.3 State-space Representation Derivation

The general input-output representation of a MIMO system in s-domain [34] is expressed as:

$$\mathbf{Y}(s) = \mathbf{G}(s) * \mathbf{U}(s) \quad (3.1)$$

Since the physical model used in this study is a three-input three-output system, one has:

$$\mathbf{Y}(s) = \begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} \quad \mathbf{G}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) \end{bmatrix}$$

$$\mathbf{U}(s) = \begin{bmatrix} u_1(s) \\ u_2(s) \\ u_3(s) \end{bmatrix}$$

where u_1, u_2, u_3 stand for three input signals and y_1, y_2, y_3 represent three output signals.

Therefore, Equation (3.1) becomes:

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) \end{bmatrix} * \begin{bmatrix} u_1(s) \\ u_2(s) \\ u_3(s) \end{bmatrix} \quad (3.2)$$

Chapter 4: Parity Relation Technique

Since the studied system is a linear system, each component of the transfer matrix $G(s)$ can be expressed by the sum of n second-order systems as follows:

$$G_{ij}(s) = \sum_{k=1}^n \left(\frac{\varphi_{ij}^k}{s^2 + 2\zeta_{ij}^k w_{ij}^k + w_{ij}^k{}^2} \right) \quad (3.3)$$

where i, j , and k represent the i^{th} output, the j^{th} input and the k^{th} mode respectively [34]. For a second-order system, ζ is the damping ratio, w is the natural frequency and φ_{ij}^k is the product of mode shape φ_i^k and mode shape φ_j^k [34]. The number n should be infinite in practice, however in this study, n is set to be three since only the first three modes are taken into consideration. The reason why the transfer function is truncated into only three modes is that in this experiment, the first three modes impose the dominant impacts on the system performance and the effects of high-frequency vibration can be ignored. In [35], the idea of simplification has been proved via the comparison of truncated model and complete model. Hence, every term of $G_{ij}(s)$ is a sum of three transfer functions of second-order systems which stand for the three modes respectively.

Take $G_{11}(s)$ as an example,

$$G_{11}(s) = \frac{\varphi_{11}^1}{s^2 + 2\zeta_{11}^1 w_{11}^1 s + w_{11}^1{}^2} + \frac{\varphi_{21}^2}{s^2 + 2\zeta_{21}^2 w_{21}^2 s + w_{21}^2{}^2} + \frac{\varphi_{31}^3}{s^2 + 2\zeta_{31}^3 w_{31}^3 s + w_{31}^3{}^2} \quad (3.4)$$

In order to implement FDD, the state-space representation of the mechanical structure is required. The transfer function matrix is re-combined according to the number of mode k . The details of derivation is explicitly explained as below. It is noted that only the first three modes are considered and analysed.

Firstly, rewrite the transfer function matrix of this system according to the mode number k . Derive the transfer function for output $y_1(s)$ regarding to the three modes.

$$y_1(s) = y_1^1(s) + y_1^2(s) + y_1^3(s) = \sum_{k=1}^3 [y_1^k(s)] \quad (3.5)$$

where k stands for the mode number, ranging from 1 to 3.

Secondly, as each $y_1^k(s)$ relates with input vector u_j , it can be written as follows:

$$y_1^k(s) = \sum_{j=1}^3 [y_{1j}^k(s)] = y_{11}^k(s) + y_{12}^k(s) + y_{13}^k(s) \quad (3.6)$$

where y_{ij}^k stands for the relation of the j^{th} input and 1^{st} output of mode k .

Hence, it is found that nine second-order equations are summed together to form one element y_i in output vector Y . According to Equation (3.6), it is obtained that:

Chapter 4: Parity Relation Technique

- when $k = 1$ which means only mode 1 is considered, the transfer function for output $y_1(s)$ of mode 1 is as follows:

$$\begin{aligned} y_1^1(s) &= y_{11}^1(s) + y_{12}^1(s) + y_{13}^1(s) \\ &= \frac{\varphi_{11}^1}{s^2 + 2\zeta_{11}^1 w_{11}^1 s + w_{11}^{1,2}} u_1(s) + \frac{\varphi_{12}^1}{s^2 + 2\zeta_{12}^1 w_{12}^1 s + w_{12}^{1,2}} u_2(s) + \frac{\varphi_{13}^1}{s^2 + 2\zeta_{13}^1 w_{13}^1 s + w_{13}^{1,2}} u_3(s) \end{aligned} \quad (3.7)$$

Similarly, the transfer function for output $y_2(s)$ of mode 1 is:

$$\begin{aligned} y_2^1(s) &= y_{21}^1(s) + y_{22}^1(s) + y_{23}^1(s) \\ &= \frac{\varphi_{21}^1}{s^2 + 2\zeta_{21}^1 w_{21}^1 s + w_{21}^{1,2}} u_1(s) + \frac{\varphi_{22}^1}{s^2 + 2\zeta_{22}^1 w_{22}^1 s + w_{22}^{1,2}} u_2(s) + \frac{\varphi_{23}^1}{s^2 + 2\zeta_{23}^1 w_{23}^1 s + w_{23}^{1,2}} u_3(s) \end{aligned} \quad (3.8)$$

The transfer function for output $y_3(s)$ of mode 1 is:

$$\begin{aligned} y_3^1(s) &= y_{31}^1(s) + y_{32}^1(s) + y_{33}^1(s) \\ &= \frac{\varphi_{31}^1}{s^2 + 2\zeta_{31}^1 w_{31}^1 s + w_{31}^{1,2}} u_1(s) + \frac{\varphi_{32}^1}{s^2 + 2\zeta_{32}^1 w_{32}^1 s + w_{32}^{1,2}} u_2(s) + \frac{\varphi_{33}^1}{s^2 + 2\zeta_{33}^1 w_{33}^1 s + w_{33}^{1,2}} u_3(s) \end{aligned} \quad (3.9)$$

- when $k = 2$, the transfer function for output $y_1(s)$ of mode 2 is:

$$\begin{aligned} y_1^2(s) &= y_{11}^2(s) + y_{12}^2(s) + y_{13}^2(s) \\ &= \frac{\varphi_{11}^2}{s^2 + 2\zeta_{11}^2 w_{11}^2 s + w_{11}^{2,2}} u_1(s) + \frac{\varphi_{12}^2}{s^2 + 2\zeta_{12}^2 w_{12}^2 s + w_{12}^{2,2}} u_2(s) + \frac{\varphi_{13}^2}{s^2 + 2\zeta_{13}^2 w_{13}^2 s + w_{13}^{2,2}} u_3(s) \end{aligned} \quad (3.10)$$

The transfer function for output $y_2(s)$ of mode 2 is:

$$\begin{aligned} y_2^2(s) &= y_{21}^2(s) + y_{22}^2(s) + y_{23}^2(s) \\ &= \frac{\varphi_{21}^2}{s^2 + 2\zeta_{21}^2 w_{21}^2 s + w_{21}^{2,2}} u_1(s) + \frac{\varphi_{22}^2}{s^2 + 2\zeta_{22}^2 w_{22}^2 s + w_{22}^{2,2}} u_2(s) + \frac{\varphi_{23}^2}{s^2 + 2\zeta_{23}^2 w_{23}^2 s + w_{23}^{2,2}} u_3(s) \end{aligned} \quad (3.11)$$

The transfer function for output $y_3(s)$ of mode 2 is:

$$\begin{aligned} y_3^2(s) &= y_{31}^2(s) + y_{32}^2(s) + y_{33}^2(s) \\ &= \frac{\varphi_{31}^2}{s^2 + 2\zeta_{31}^2 w_{31}^2 s + w_{31}^{2,2}} u_1(s) + \frac{\varphi_{32}^2}{s^2 + 2\zeta_{32}^2 w_{32}^2 s + w_{32}^{2,2}} u_2(s) + \frac{\varphi_{33}^2}{s^2 + 2\zeta_{33}^2 w_{33}^2 s + w_{33}^{2,2}} u_3(s) \end{aligned} \quad (3.12)$$

- when $k = 3$, the transfer function for output $y_1(s)$ of mode 3 is:

$$\begin{aligned} y_1^3(s) &= y_{11}^3(s) + y_{12}^3(s) + y_{13}^3(s) \\ &= \frac{\varphi_{11}^3}{s^2 + 2\zeta_{11}^3 w_{11}^3 s + w_{11}^{3,2}} u_1(s) + \frac{\varphi_{12}^3}{s^2 + 2\zeta_{12}^3 w_{12}^3 s + w_{12}^{3,2}} u_2(s) + \frac{\varphi_{13}^3}{s^2 + 2\zeta_{13}^3 w_{13}^3 s + w_{13}^{3,2}} u_3(s) \end{aligned} \quad (3.13)$$

The transfer function for output $y_2(s)$ of mode 3 is:

Chapter 4: Parity Relation Technique

$$\begin{aligned}
 y_2^3(s) &= y_{21}^3(s) + y_{22}^3(s) + y_{23}^3(s) \\
 &= \frac{\varphi_{21}^3}{s^2 + 2\zeta_{21}^3 w_{21}^3 s + w_{21}^3{}^2} u_1(s) + \frac{\varphi_{22}^3}{s^2 + 2\zeta_{22}^3 w_{22}^3 s + w_{22}^3{}^2} u_2(s) + \frac{\varphi_{23}^3}{s^2 + 2\zeta_{23}^3 w_{23}^3 s + w_{23}^3{}^2} u_3(s) \quad (3.14)
 \end{aligned}$$

The transfer function for output $y_3(s)$ of mode 3 is:

$$\begin{aligned}
 y_3^3(s) &= y_{31}^3(s) + y_{32}^3(s) + y_{33}^3(s) \\
 &= \frac{\varphi_{31}^3}{s^2 + 2\zeta_{31}^3 w_{31}^3 s + w_{31}^3{}^2} u_1(s) + \frac{\varphi_{32}^3}{s^2 + 2\zeta_{32}^3 w_{32}^3 s + w_{32}^3{}^2} u_2(s) + \frac{\varphi_{33}^3}{s^2 + 2\zeta_{33}^3 w_{33}^3 s + w_{33}^3{}^2} u_3(s) \quad (3.15)
 \end{aligned}$$

It is observed that all the first modes of the transfer functions for the output y_i have the much similar natural frequencies and slightly different damping ratios. Therefore, an average value method is chosen in order to reduce and merge the terms of Equation (3.7) – Equation (3.15). A scenario of parameters selection is as follows:

- For damping ratio ζ , select the largest value of nine sets of data in same mode.
- For natural frequency w , choose the average frequency in one mode.

Therefore, these nine different sets of ζ_{ij}^1 , ζ_{ij}^2 and ζ_{ij}^3 are merged into ζ_1 , ζ_2 and ζ_3 respectively, and these nine different sets ow_{ij}^1 , w_{ij}^2 and w_{ij}^3 are merged into w_1 , w_2 and w_3 .

After simplification, Equation (3.7) – Equation (3.9) become:

$$y_1^1(s) = y_{11}^1(s) + y_{12}^1(s) + y_{13}^1(s) = \frac{\varphi_{11}^1 u_1(s) + \varphi_{12}^1 u_2(s) + \varphi_{13}^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} \quad (3.16)$$

$$y_2^1(s) = y_{21}^1(s) + y_{22}^1(s) + y_{23}^1(s) = \frac{\varphi_{21}^1 u_1(s) + \varphi_{22}^1 u_2(s) + \varphi_{23}^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} \quad (3.17)$$

$$y_3^1(s) = y_{31}^1(s) + y_{32}^1(s) + y_{33}^1(s) = \frac{\varphi_{31}^1 u_1(s) + \varphi_{32}^1 u_2(s) + \varphi_{33}^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} \quad (3.18)$$

Equation (3.10) – Equation (3.12) become:

$$y_1^2(s) = y_{11}^2(s) + y_{12}^2(s) + y_{13}^2(s) = \frac{\varphi_{11}^2 u_1(s) + \varphi_{12}^2 u_2(s) + \varphi_{13}^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} \quad (3.19)$$

$$y_2^2(s) = y_{21}^2(s) + y_{22}^2(s) + y_{23}^2(s) = \frac{\varphi_{21}^2 u_1(s) + \varphi_{22}^2 u_2(s) + \varphi_{23}^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} \quad (3.20)$$

$$y_3^2(s) = y_{31}^2(s) + y_{32}^2(s) + y_{33}^2(s) = \frac{\varphi_{31}^2 u_1(s) + \varphi_{32}^2 u_2(s) + \varphi_{33}^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} \quad (3.21)$$

Equation (3.13) – Equation (3.15) become:

$$y_1^3(s) = y_{11}^3(s) + y_{12}^3(s) + y_{13}^3(s) = \frac{\varphi_{11}^3 u_1(s) + \varphi_{12}^3 u_2(s) + \varphi_{13}^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} \quad (3.22)$$

Chapter 4: Parity Relation Technique

$$y_3^3(s) = y_{21}^3(s) + y_{22}^3(s) + y_{23}^3(s) = \frac{\varphi_{21}^3 u_1(s) + \varphi_{22}^3 u_2(s) + \varphi_{23}^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} \quad (3.23)$$

$$y_3^3(s) = y_{31}^3(s) + y_{32}^3(s) + y_{33}^3(s) = \frac{\varphi_{31}^3 u_1(s) + \varphi_{32}^3 u_2(s) + \varphi_{33}^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} \quad (3.24)$$

After obtaining these nine transfer functions from mode 1 to mode 3, the mathematical model can be rewritten as follows:

$$\mathbf{Y} = \begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} = \begin{bmatrix} y_1^1(s) + y_1^2(s) + y_1^3(s) \\ y_2^1(s) + y_2^2(s) + y_2^3(s) \\ y_3^1(s) + y_3^2(s) + y_3^3(s) \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\varphi_{11}^1 u_1(s) + \varphi_{12}^1 u_2(s) + \varphi_{13}^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} + \frac{\varphi_{11}^2 u_1(s) + \varphi_{12}^2 u_2(s) + \varphi_{13}^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} + \frac{\varphi_{11}^3 u_1(s) + \varphi_{12}^3 u_2(s) + \varphi_{13}^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} \\ \frac{\varphi_{21}^1 u_1(s) + \varphi_{22}^1 u_2(s) + \varphi_{23}^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} + \frac{\varphi_{21}^2 u_1(s) + \varphi_{22}^2 u_2(s) + \varphi_{23}^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} + \frac{\varphi_{21}^3 u_1(s) + \varphi_{22}^3 u_2(s) + \varphi_{23}^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} \\ \frac{\varphi_{31}^1 u_1(s) + \varphi_{32}^1 u_2(s) + \varphi_{33}^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} + \frac{\varphi_{31}^2 u_1(s) + \varphi_{32}^2 u_2(s) + \varphi_{33}^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} + \frac{\varphi_{31}^3 u_1(s) + \varphi_{32}^3 u_2(s) + \varphi_{33}^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} \end{bmatrix} \quad (3.25)$$

In [35], it has been theoretically proven that the term φ_{ij}^k is the product of mode shape φ_j^k and φ_i^k , so a common term can be taken out in one mode. The mathematical model is modified as shown below:

$$\mathbf{Y} = \begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} =$$

$$\begin{bmatrix} \frac{\varphi_1^1(\varphi_1^1 u_1(s) + \varphi_2^1 u_2(s) + \varphi_3^1 u_3(s))}{s^2 + 2\zeta_1 w_1 s + w_1^2} + \frac{\varphi_1^2(\varphi_1^2 u_1(s) + \varphi_2^2 u_2(s) + \varphi_3^2 u_3(s))}{s^2 + 2\zeta_2 w_2 s + w_2^2} + \frac{\varphi_1^3(\varphi_1^3 u_1(s) + \varphi_2^3 u_2(s) + \varphi_3^3 u_3(s))}{s^2 + 2\zeta_3 w_3 s + w_3^2} \\ \frac{\varphi_2^1(\varphi_1^1 u_1(s) + \varphi_2^1 u_2(s) + \varphi_3^1 u_3(s))}{s^2 + 2\zeta_1 w_1 s + w_1^2} + \frac{\varphi_2^2(\varphi_1^2 u_1(s) + \varphi_2^2 u_2(s) + \varphi_3^2 u_3(s))}{s^2 + 2\zeta_2 w_2 s + w_2^2} + \frac{\varphi_2^3(\varphi_1^3 u_1(s) + \varphi_2^3 u_2(s) + \varphi_3^3 u_3(s))}{s^2 + 2\zeta_3 w_3 s + w_3^2} \\ \frac{\varphi_3^1(\varphi_1^1 u_1(s) + \varphi_2^1 u_2(s) + \varphi_3^1 u_3(s))}{s^2 + 2\zeta_1 w_1 s + w_1^2} + \frac{\varphi_3^2(\varphi_1^2 u_1(s) + \varphi_2^2 u_2(s) + \varphi_3^2 u_3(s))}{s^2 + 2\zeta_2 w_2 s + w_2^2} + \frac{\varphi_3^3(\varphi_1^3 u_1(s) + \varphi_2^3 u_2(s) + \varphi_3^3 u_3(s))}{s^2 + 2\zeta_3 w_3 s + w_3^2} \end{bmatrix} \quad (3.26)$$

For the purpose of simplification, variables m_1 , m_2 and m_3 are defined as follows:

$$m_1(s) = \frac{\varphi_1^1 u_1(s) + \varphi_2^1 u_2(s) + \varphi_3^1 u_3(s)}{s^2 + 2\zeta_1 w_1 s + w_1^2} = \frac{[\varphi_1^1 \quad \varphi_2^1 \quad \varphi_3^1] \begin{bmatrix} u_1(s) \\ u_2(s) \\ u_3(s) \end{bmatrix}}{s^2 + 2\zeta_1 w_1 s + w_1^2} \quad (3.27)$$

$$m_2(s) = \frac{\varphi_1^2 u_1(s) + \varphi_2^2 u_2(s) + \varphi_3^2 u_3(s)}{s^2 + 2\zeta_2 w_2 s + w_2^2} = \frac{[\varphi_1^2 \quad \varphi_2^2 \quad \varphi_3^2] \begin{bmatrix} u_1(s) \\ u_2(s) \\ u_3(s) \end{bmatrix}}{s^2 + 2\zeta_2 w_2 s + w_2^2} \quad (3.28)$$

$$m_3(s) = \frac{\varphi_1^3 u_1(s) + \varphi_2^3 u_2(s) + \varphi_3^3 u_3(s)}{s^2 + 2\zeta_3 w_3 s + w_3^2} = \frac{[\varphi_1^3 \quad \varphi_2^3 \quad \varphi_3^3] \begin{bmatrix} u_1(s) \\ u_2(s) \\ u_3(s) \end{bmatrix}}{s^2 + 2\zeta_3 w_3 s + w_3^2} \quad (3.29)$$

Chapter 4: Parity Relation Technique

Via substituting Equation (3.27) – Equation (3.29) into Equation (3.26), the relation of output Y and vector J can be deduced as follows:

$$Y = \begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} = \begin{bmatrix} \varphi_1^1 m_1 + \varphi_1^2 m_2 + \varphi_1^3 m_3 \\ \varphi_2^1 m_1 + \varphi_2^2 m_2 + \varphi_2^3 m_3 \\ \varphi_3^1 m_1 + \varphi_3^2 m_2 + \varphi_3^3 m_3 \end{bmatrix} = \begin{bmatrix} \varphi_1^1 & \varphi_1^2 & \varphi_1^3 \\ \varphi_2^1 & \varphi_2^2 & \varphi_2^3 \\ \varphi_3^1 & \varphi_3^2 & \varphi_3^3 \end{bmatrix} \begin{bmatrix} m_1(s) \\ m_2(s) \\ m_3(s) \end{bmatrix} \quad (3.30)$$

Equation (3.30) can be simplified as follows:

$$Y = \psi * M \quad (3.31)$$

$$\text{where } \psi = \begin{bmatrix} \varphi_1^1 & \varphi_1^2 & \varphi_1^3 \\ \varphi_2^1 & \varphi_2^2 & \varphi_2^3 \\ \varphi_3^1 & \varphi_3^2 & \varphi_3^3 \end{bmatrix} \quad M = \begin{bmatrix} m_1(s) \\ m_2(s) \\ m_3(s) \end{bmatrix}$$

In order to carry out the state space representation from the input-output form, some transformations should be done as follows:

Through Equation (3.27), one has:

$$s^2 m_1(s) + 2\zeta_1 w_1 s m_1(s) + w_1^2 m_1(s) = [\varphi_1^1 \quad \varphi_1^2 \quad \varphi_1^3] \begin{bmatrix} u_1(s) \\ u_2(s) \\ u_3(s) \end{bmatrix} \quad (3.32)$$

Then, Equation (3.32) can be derived through taking inverse Laplace transform on Equation (3.27):

$$\ddot{m}_1(t) + 2\zeta_1 w_1 \dot{m}_1(t) + w_1^2 m_1(t) = [\varphi_1^1 \quad \varphi_1^2 \quad \varphi_1^3] \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \quad (3.33)$$

Similarly, Equation (3.34) and Equation (3.35) are derived as follows:

$$\ddot{m}_2(t) + 2\zeta_2 w_2 \dot{m}_2(t) + w_2^2 m_2(t) = [\varphi_2^1 \quad \varphi_2^2 \quad \varphi_2^3] \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \quad (3.34)$$

$$\ddot{m}_3(t) + 2\zeta_3 w_3 \dot{m}_3(t) + w_3^2 m_3(t) = [\varphi_3^1 \quad \varphi_3^2 \quad \varphi_3^3] \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \quad (3.35)$$

Combining Equations (3.33) - (3.35) into Equation (3.36) yields:

Chapter 4: Parity Relation Technique

$$\begin{aligned}
 \begin{bmatrix} \ddot{m}_1(t) \\ \ddot{m}_2(t) \\ \ddot{m}_3(t) \end{bmatrix} + \begin{bmatrix} 2\zeta_1 w_1 & & \\ & 2\zeta_2 w_2 & \\ & & 2\zeta_3 w_3 \end{bmatrix} \begin{bmatrix} \dot{m}_1(t) \\ \dot{m}_2(t) \\ \dot{m}_3(t) \end{bmatrix} + \begin{bmatrix} w_1^2 & & \\ & w_2^2 & \\ & & w_3^2 \end{bmatrix} \begin{bmatrix} m_1(t) \\ m_2(t) \\ m_3(t) \end{bmatrix} \\
 = \begin{bmatrix} \varphi_1^1 & \varphi_2^1 & \varphi_3^1 \\ \varphi_1^2 & \varphi_2^2 & \varphi_3^2 \\ \varphi_1^3 & \varphi_2^3 & \varphi_3^3 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix}
 \end{aligned} \tag{3.36}$$

In order to implement parity relation technique, the state space form of a dynamic system should be in a Canonical form. Hence, a new definition is introduced as below:

$$\begin{cases} \dot{m}_1(t) = \dot{x}_1(t) = x_2(t) \\ \ddot{m}_1(t) = \ddot{x}_1(t) = \dot{x}_2(t) \end{cases} \tag{3.37}$$

Substituting Equation (3.37) into Equation (3.33) yields:

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = [\varphi_1^1 \quad \varphi_2^1 \quad \varphi_3^1] \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} - 2\zeta_1 w_1 x_2(t) - w_1^2 x_1(t) \end{cases} \tag{3.38}$$

The state space form of Equation (3.38) is derived as follows:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -w_1^2 & -2\zeta_1 w_1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \varphi_1^1 & \varphi_2^1 & \varphi_3^1 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \tag{3.39}$$

Through applying the same method, the other two equations in state space form can be obtained as follows:

$$\begin{bmatrix} \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -w_2^2 & -2\zeta_2 w_2 \end{bmatrix} \begin{bmatrix} x_3(t) \\ x_4(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \varphi_1^2 & \varphi_2^2 & \varphi_3^2 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \tag{3.40}$$

$$\begin{bmatrix} \dot{x}_5(t) \\ \dot{x}_6(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -w_3^2 & -2\zeta_3 w_3 \end{bmatrix} \begin{bmatrix} x_5(t) \\ x_6(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \varphi_1^3 & \varphi_2^3 & \varphi_3^3 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \tag{3.41}$$

Finally, through stacking Equations (3.39) - (3.41) together, the final state equation of the laboratory model is obtained as below:

Chapter 4: Parity Relation Technique

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \\ \dot{x}_5(t) \\ \dot{x}_6(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & & & & \\ -w_1^2 & -2\zeta_1 w_1 & & & & \\ & & 0 & 1 & & \\ & & -w_2^2 & -2\zeta_2 w_2 & & \\ & & & & 0 & 1 \\ & & & & -w_3^2 & -2\zeta_3 w_3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ \varphi_1^1 & \varphi_2^1 & \varphi_3^1 \\ 0 & 0 & 0 \\ \varphi_1^2 & \varphi_2^2 & \varphi_3^2 \\ 0 & 0 & 0 \\ \varphi_1^3 & \varphi_2^3 & \varphi_3^3 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} \quad (3.42)$$

In order to carry out the output equation of the laboratory model, taking inverse Laplace transform of Equation (3.31) can figure out the output equation of the laboratory model:

$$\begin{bmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \end{bmatrix} = \begin{bmatrix} \varphi_1^1 & \varphi_2^1 & \varphi_3^1 \\ \varphi_1^2 & \varphi_2^2 & \varphi_3^2 \\ \varphi_1^3 & \varphi_2^3 & \varphi_3^3 \end{bmatrix} \begin{bmatrix} m_1(t) \\ m_2(t) \\ m_3(t) \end{bmatrix} = \begin{bmatrix} \varphi_1^1 & 0 & \varphi_1^2 & 0 & \varphi_1^3 & 0 \\ \varphi_2^1 & 0 & \varphi_2^2 & 0 & \varphi_2^3 & 0 \\ \varphi_3^1 & 0 & \varphi_3^2 & 0 & \varphi_3^3 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix} \quad (3.43)$$

Finally, for the laboratory model, the system matrix \mathbf{A} is:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & & & & \\ -w_1^2 & -2\zeta_1 w_1 & & & & \\ & & 0 & 1 & & \\ & & -w_2^2 & -2\zeta_2 w_2 & & \\ & & & & 0 & 1 \\ & & & & -w_3^2 & -2\zeta_3 w_3 \end{bmatrix}$$

The input matrix \mathbf{B} is:

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ \varphi_1^1 & \varphi_2^1 & \varphi_3^1 \\ 0 & 0 & 0 \\ \varphi_1^2 & \varphi_2^2 & \varphi_3^2 \\ 0 & 0 & 0 \\ \varphi_1^3 & \varphi_2^3 & \varphi_3^3 \end{bmatrix}$$

The output matrix \mathbf{C} is:

$$\mathbf{C} = \begin{bmatrix} \varphi_1^1 & 0 & \varphi_1^2 & 0 & \varphi_1^3 & 0 \\ \varphi_2^1 & 0 & \varphi_2^2 & 0 & \varphi_2^3 & 0 \\ \varphi_3^1 & 0 & \varphi_3^2 & 0 & \varphi_3^3 & 0 \end{bmatrix}$$

The block diagram of the laboratory model is drawn, as shown in Figure 3.2. The relation of input signals and output signals is shown clearly in Figure 3.2, which is beneficial to understand the inner of the physical model. As shown in Figure 3.2, this is the open loop system model, in which the three inputs are zero. However, if a set of controllers are added, the inputs signals are connected with the output port of controllers, and then are sent to transducers $T1$, $T2$ and $T3$. The three outputs are measured by sensor $S1$, $S2$ and $S3$ that are shown in Figure 1.1.

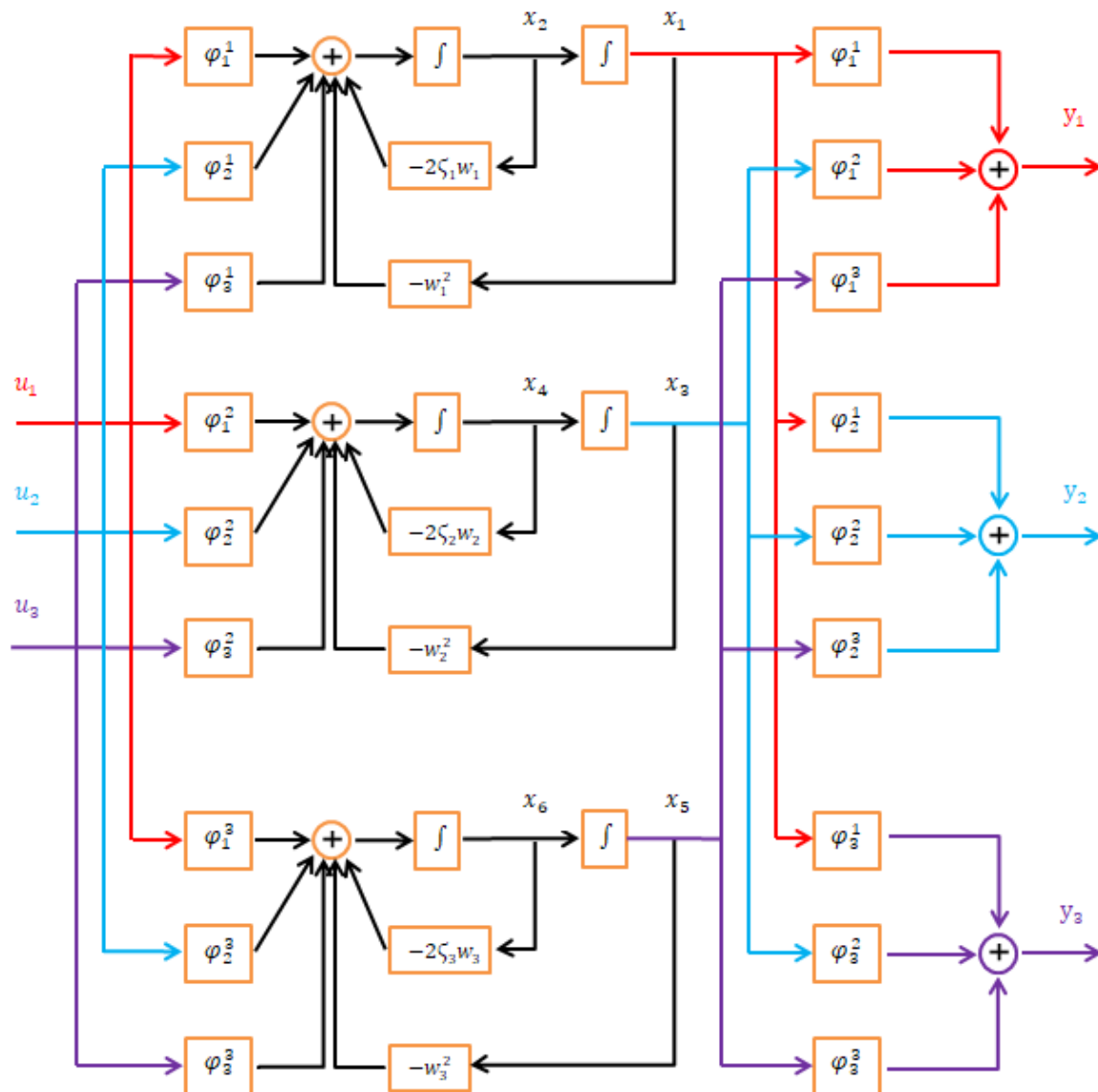


Figure 3.2: The block diagram of the laboratory model in state space representation

3.4 State-space Representation Calculation

This section calculates the values of state space matrixes A, B and C using the experiment data obtained by ModalVIEW.

After simplification in Section 3.3, the data of three modes obtained from ModalVIEW for each mode are listed in Table (3.1) - Table (3.3) respectively:

	Mode 1		
	Frequency (rad/s)	Damping ratio (%)	Mode shape
G_{11}	22.66	2.667	0.132387

Chapter 4: Parity Relation Technique

G_{12}	22.66	2.667	0.092623
G_{13}	22.66	2.667	0.07954
G_{21}	22.66	2.667	0.117467
G_{22}	22.66	2.667	0.079589
G_{23}	22.66	2.667	0.063085
G_{31}	22.66	2.667	0.098985
G_{32}	22.66	2.667	0.042448
G_{33}	22.66	2.667	0.056874

Table 3.1: The data of mode 1 for all transfer functions

	Mode 2		
	Frequency (rad/s)	Damping ratio (%)	Mode shape
G_{11}	29.05	2.533	0.522717
G_{12}	29.05	2.533	0.238071
G_{13}	29.05	2.533	0.094088
G_{21}	29.05	2.533	0.081823
G_{22}	29.05	2.533	0.230657
G_{23}	29.05	2.533	0.235156
G_{31}	29.05	2.533	0.082337
G_{32}	29.05	2.533	0.249072
G_{33}	29.05	2.533	0.198524

Table 3.2: The data of mode 2 for all transfer functions

	Mode 3		
	Frequency (rad/s)	Damping ratio (%)	Mode shape
G_{11}	34.86	2.419	0.20722
G_{12}	34.86	2.419	0.386224
G_{13}	34.86	2.419	0.336231
G_{21}	34.86	2.419	0.36751
G_{22}	34.86	2.419	0.70789
G_{23}	34.86	2.419	0.588276
G_{31}	34.86	2.419	0.321649
G_{32}	34.86	2.419	0.609829
G_{33}	34.86	2.419	0.574678

Table 3.3: The data of mode 3 for all transfer functions

Chapter 4: Parity Relation Technique

In MATLAB, the matrixes of the mathematical model are calculated and shown as below:

$$A = \begin{bmatrix} 0 & 1 & & & & \\ -20271 & -0.008 & & & & \\ & & 0 & 1 & & \\ & & -33316 & 9 & & \\ & & & & 0 & 1 \\ & & & & 47975 & 11 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0.3639 & 0.2821 & 0.2385 \\ 0 & 0 & 0 \\ 0.723 & 0.4803 & 0.4456 \\ 0 & 0 & 0 \\ 0.4552 & 0.8414 & 0.7581 \end{bmatrix}$$

$$C = \begin{bmatrix} 0.3639 & 0 & 0.7230 & 0 & 0.4452 & 0 \\ 0.2821 & 0 & 0.4803 & 0 & 0.8414 & 0 \\ 0.2385 & 0 & 0.4456 & 0 & 0.7581 & 0 \end{bmatrix}$$

Since parity relation technique is utilised in discrete system, these matrixes then are converted into discrete system forms in MATLAB using function 'c2d'. The sampling time is chosen as 0.01s. The new matrixes for discrete system are shown as below:

$$A_d = \begin{bmatrix} 0.1669 & 0.0067 & & & & \\ -135.6312 & -0.1161 & & & & \\ & & 0.2164 & 0.0051 & & \\ & & -168.7472 & -0.2631 & & \\ & & & & -0.5315 & 0.0035 \\ & & & & -169.2504 & -0.5689 \end{bmatrix}$$

$$B_d = \begin{bmatrix} 0 & 0 & 0 \\ 0.0024 & 0.0019 & 0.0016 \\ 0 & 0 & 0 \\ 0.0037 & 0.0024 & 0.0023 \\ 0 & 0 & 0 \\ 0.0016 & 0.0030 & 0.0027 \end{bmatrix}$$

$$C_d = \begin{bmatrix} 0.3639 & 0 & 0.7230 & 0 & 0.4452 & 0 \\ 0.2821 & 0 & 0.4803 & 0 & 0.8414 & 0 \\ 0.2385 & 0 & 0.4456 & 0 & 0.7581 & 0 \end{bmatrix}$$

Figure 3.3 depicts the step responses of continuous system and discrete system that are calculated according to the obtained data. From Figure 3.3, the two curves in each subplot have extremely similar shapes and the same trends, which proves that these two kinds of systems are almost the same and can be used for further research.

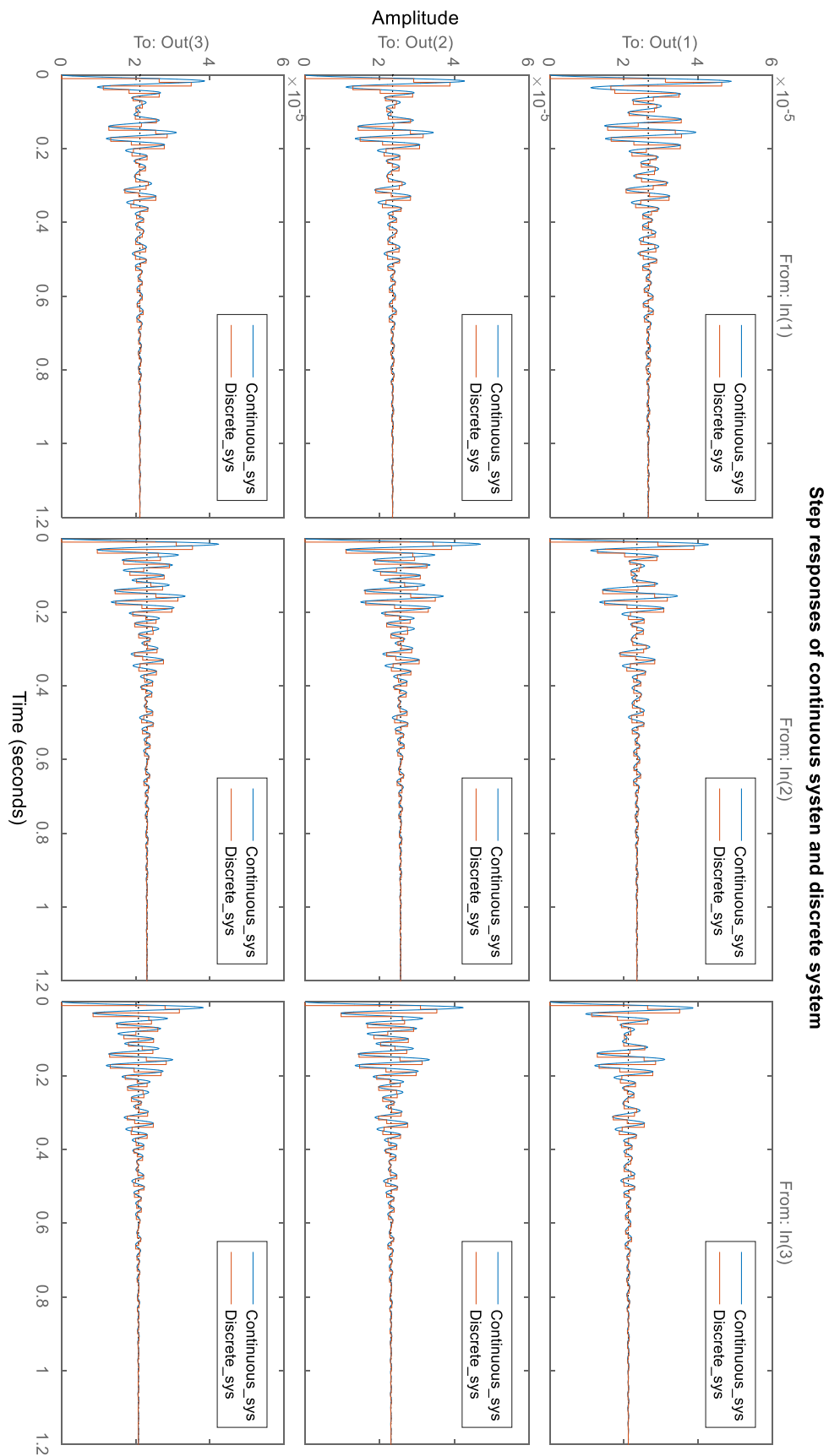


Figure 3.3: Step responses of continuous system and discrete system

Chapter 4 : Parity Relation Technique

The basic concept of fault detection and diagnosis and the main mode-based methods for implementing FDD have been introduced respectively in Chapter 1 and Chapter 2. Besides, the open loop discrete state space model is derived in Chapter 3 since the implementation of parity relation technique is based on the open loop model. Finally, the FDD using parity relation method is designed based on the assumption that all the signals of the given system are measurable and known.

This chapter focuses on the design of FDD processes using parity relation method, including two main parts:

- The theoretical research regarding to parity relation technique is illustrated, including the design of fault detection, parity space order and fault isolation.
- The validations for SISO and MIMO systems are programmed in MATLAB and the designed residuals are displayed and compared respectively.

Finally, the conclusion is drawn. For various simulations, the results are discussed and compared. Besides, the problem of this study is analysed and meanwhile some reasonable advices are given for further study.

4.1 Model structure

A general linear time-invariant discrete-time system is written in state-space representation (SSR) form:

$$x(k+1) = \mathbf{A}x(k) + \mathbf{B}u(k) \quad (4.1)$$

$$y(k) = \mathbf{C}x(k) + \mathbf{D}u(k) \quad (4.2)$$

where $x \in \mathbb{R}^n$ denotes the state vector, $u \in \mathbb{R}^r$ denotes the input vector, $y \in \mathbb{R}^m$ is the vector of output signal. \mathbf{A} is System Matrix with dimension of $n \times n$, \mathbf{B} is Input Matrix with dimension of $n \times r$, \mathbf{C} is Output Matrix with dimension of $m \times n$, and \mathbf{D} is Direct Transmission Matrix with dimension of $m \times r$. Note that Equation (4.1) and Equation (4.2) describe a system model without any fault.

As mentioned before, only the sensor faults and actuator faults are considered in this project. The SSR of the system with faults can be expressed as follows:

$$x(k+1) = \mathbf{A}x(k) + \mathbf{B}u(k) + \mathbf{F}f_a(k) \quad (4.3)$$

$$y(k) = \mathbf{C}x(k) + \mathbf{D}u(k) + f_s(k) \quad (4.4)$$

where $f_a \in \mathbb{R}^r$ and $f_s \in \mathbb{R}^m$ denotes the actuator fault vector and sensor fault vector respectively, and \mathbf{F} is Actuator Fault Matrix with dimension of $n \times r$ [7].

4.2 Fault detection

The core idea of parity relation technique is to check the parity (or consistency) of the measurements of a measurable system. In order to detect a fault, a parity equation is required to be generated firstly. According to [6], there are two main cases according to the dimension of Output Matrix \mathbf{C} for a system. It should be noted that in Case 1, only the sensor fault is considered as this case is only used in some particular linear systems and it is not suitable for the real laboratory model as well.

1. Case 1 $\text{rank}(\mathbf{C}) = n < m$

This means that the dimension of $y(k)$ is larger than the dimension of $x(k)$. Under this situation, matrix \mathbf{C} is a singular matrix, so there exists at least a vector \mathbf{V} that satisfies the condition:

$$\mathbf{V}\mathbf{C} = 0 \quad (4.5)$$

in which the vector \mathbf{V} is called parity vector.

Hence, based on previous analysis and Equations (4.1) and (4.2), the residual of the system can be defined as follows:

$$\mathbf{R}(k) = \mathbf{V}[y(k) - \mathbf{D}u(k)] = \mathbf{V}\mathbf{C}x(k) \quad (4.6)$$

Now, when the system is fault-free and noise-free, the residual $\mathbf{R}(k)$ should be zero. Then, when any sensor fault occurs, Equation (4.6) becomes

$$\begin{aligned} \mathbf{R}(k) &= \mathbf{V}[y(k) - \mathbf{D}u(k)] = \mathbf{V}\mathbf{C}x(k) + \mathbf{V}f(k) \\ &= v_1f_1(k) + v_2f_2(k) + \cdots + v_mf_m(k) \end{aligned} \quad (4.7)$$

where v_m is the m^{th} element of \mathbf{V} , and $f_m(k)$ is the m^{th} element of $f(k)$ which stands for the fault signal in the i^{th} sensor. Equation (4.7) states that the generated residual only contains information of faults and the unknown state $x(k)$ is cancelled by the parity vector \mathbf{V} [6]. Consequently, for Case 1, the fault detection process is implemented successfully through checking the residual signal.

2. Case 2 $\text{rank}(\mathbf{C}) = m < n$

As in Case 2, the parity vector \mathbf{V} does not exist because matrix \mathbf{C} is full rank. To overcome this problem, an extended parity relation technique called temporal redundancy relation is presented in [28]. The temporal redundancy is constructed by checking the inputs and the outputs data within a time window $[k - s, k]$ of the dynamic system.

For the fault-free system given in Equation (4.1) and Equation (4.2), the system variable $x(k - s)$ and output $y(k - s)$ for time $(k - s)$ can be expressed as follows

$$x(k - s + 1) = \mathbf{A}x(k - s) + \mathbf{B}u(k - s) \quad (4.8)$$

$$y(k - s) = \mathbf{C}x(k - s) + \mathbf{D}u(k - s) \quad (4.9)$$

Then, at the time $(k - s + 1)$, one has:

$$y(k - s + 1) = \mathbf{C}x(k - s + 1) + \mathbf{D}u(k - s + 1) \quad (4.9a)$$

Chapter 4: Parity Relation Technique

Substituting Equation (4.8) into Equation (4.9a) yields:

$$y(k - s + 1) = \mathbf{CA}x(k - s) + \mathbf{CB}u(k - s) + \mathbf{D}u(k - s + 1) \quad (4.10)$$

Similarly, the below equations can be yielded via recursion:

$$y(k - s + 2) = \mathbf{CA}^2x(k - s) + \mathbf{CAB}u(k - s) + \mathbf{CB}u(k - s + 1) + \mathbf{D}u(k - s + 1) \quad (4.11)$$

⋮

$$y(k) = \mathbf{CA}^s x(k - s) + \mathbf{CA}^{s-1} \mathbf{B}u(k - s) + \dots + \mathbf{CB}u(k + 1) + \mathbf{D}u(k) \quad (4.12)$$

Stacking these $s + 1$ equations together can yield the following relation:

$$\mathbf{Y}(k) = \mathbf{H}x(k - s) + \mathbf{Q}\mathbf{U}(k) \quad (4.13)$$

where

$$\mathbf{Y}(k) = \begin{pmatrix} y(k - s) \\ y(k - s + 1) \\ \vdots \\ y(k) \end{pmatrix} \in \mathbb{R}^{m(s+1) \times 1}, \quad \mathbf{U}(k) = \begin{pmatrix} u(k - s) \\ u(k - s + 1) \\ \vdots \\ u(k) \end{pmatrix} \in \mathbb{R}^{r(s+1) \times 1} \quad (4.14)$$

$$\mathbf{H} = \begin{pmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^s \end{pmatrix} \in \mathbb{R}^{m(s+1) \times n} \quad \mathbf{Q} = \begin{pmatrix} \mathbf{D} & 0 & 0 & 0 \\ \mathbf{CB} & \mathbf{D} & 0 & 0 \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{CA}^{s-1} \mathbf{B} & \mathbf{CA}^{s-2} \mathbf{B} & \dots & \mathbf{D} \end{pmatrix} \in \mathbb{R}^{m(s+1) \times r(s+1)} \quad (4.15)$$

It is noted that for the given system in Equation (4.13), $\mathbf{Y}(k)$ and $\mathbf{U}(k)$ are not scalar values. The vector $\mathbf{Y}(k)$ is a column vector with dimension $m(s + 1) \times 1$, and the vector $\mathbf{U}(k)$ is also a column vector with dimension $r(s + 1) \times 1$ [7].

Therefore, as shown in Equation (4.13), the relation of input vectors and output vectors is generated, which is so called parity equation. Note that if the variable s is given, $\mathbf{Y}(k)$ and $\mathbf{U}(k)$ are known, which include the temporal and past outputs and inputs respectively [7]. Besides, all the elements of matrix \mathbf{H} and matrix \mathbf{Q} are all known. Only the past state vector $x(k - s)$ is unknown in Equation (4.13).

To eliminate the term related to the past state vector in Equation (4.13), a non-zero vector \mathbf{V} can be designed so that it satisfies the condition

$$\mathbf{V}_s \mathbf{H} = 0 \quad (4.16)$$

The vector \mathbf{V}_s is called 'parity vector' with dimension $1 \times m(s + 1)$. According to Equation (4.16), the parity vector belongs to the left null space of matrix \mathbf{H} and plays an essential role in parity relation technique [38]. All parity vectors belong to a space \mathbf{P} defined by $\mathbf{P} = \{\mathbf{V} | \mathbf{V}\mathbf{H} = 0\}$ and this space is known as parity space with s^{th} order. The order s decides whether the parity vector can be found or not, and the design of s is illustrated in Section 4.3.

Chapter 4: Parity Relation Technique

According to [28], a residual can be constructed as follows

$$\mathbf{R}(k) = \mathbf{V}_s[\mathbf{Y}(k) - \mathbf{Q}\mathbf{U}(k)] \quad (4.17)$$

This algorithm is known as 'Chow - Willsky scheme', first proposed by Chow and Willsky in 1984 [39].

Substituting Equation (4.13) into Equation (4.17), the residual generator can be found as follows:

$$\mathbf{R}(k) = \mathbf{V}_s\mathbf{H}\mathbf{x}(k-s) = 0 \quad \forall \mathbf{x} \quad (4.18)$$

Now, taking sensor faults and actuator faults into account, Equation (4.17) becomes

$$\mathbf{R}(k) = \mathbf{V}_s[\mathbf{Y}(k) - \mathbf{Q}\mathbf{U}(k)] = \mathbf{V}_s[\mathbf{H}\mathbf{x}(k-s) + \mathbf{Q}_a\mathbf{F}_a(k) + \mathbf{F}_s(k)] \quad (4.19)$$

where

$$\mathbf{F}_s(k) = \begin{pmatrix} f_s(k-s) \\ f_s(k-s+1) \\ \vdots \\ f_s(k) \end{pmatrix} \in \mathbb{R}^{n(s+1) \times 1} \quad \mathbf{F}_a(k) = \begin{pmatrix} f_a(k-s) \\ f_a(k-s+1) \\ \vdots \\ f_a(k) \end{pmatrix} \in \mathbb{R}^{r(s+1) \times 1} \quad (4.20)$$

$$\mathbf{Q}_a = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \mathbf{C}\mathbf{F} & 0 & 0 & 0 \\ \mathbf{C}\mathbf{A}\mathbf{F} & \mathbf{C}\mathbf{F} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{C}\mathbf{A}^{s-1}\mathbf{F} & \mathbf{C}\mathbf{A}^{s-2}\mathbf{F} & \dots & 0 \end{pmatrix} \in \mathbb{R}^{m(s+1) \times r(s+1)} \quad (4.21)$$

It is noted that in Equation (4.19),

- \mathbf{F}_s is a vector of sensor faults
- \mathbf{F}_a is a vector of actuator faults

For fault detection using parity relation technique, it is easy to implement once the parity vector \mathbf{V}_s is found. When a system is fault free, the residual generated by parity relation is zero constantly, and when any fault exists in the system, the designed residual differs from zero, which indicates the occurrence of the fault. Therefore, it is easy to detect fault in control system using parity relation method. Nevertheless, to isolate the fault type and fault location requires more strict conditions, which are introduced in Section 4.4.

4.3 Design of parity space order s

In the last section, the core of parity relation technique is to find a suitable parity vector \mathbf{V} , so the below condition must hold:

$$\text{rank}(H) < \text{the row number of matrix } H = (s+1)m \quad (4.22)$$

Commonly, parity space order s is chosen as follows:

$$s \geq n \quad (4.23)$$

In Chen & Patton's book [6], the minimal order s_0 of parity space satisfies the condition:

$$\frac{\text{rank}(H_0)}{\text{rank}(C)} \leq s_0 \leq \text{rank}(H_0) - \text{rank}(C) + 1 \quad (4.24)$$

where H_0 is the observability matrix of pair (\mathbf{C}, \mathbf{A}) .

Chapter 4: Parity Relation Technique

Additionally, if a system is observable, and the output matrix C is full rank, Equation (4.24) becomes:

$$\frac{n}{m} \leq s_0 \leq n - m + 1 \quad (4.25)$$

Hence, for the laboratory model used in this study, the minimal order s_0 can be solved as follows:

1. Check the observability of the system through the observability matrix ϑ ,

$$\vartheta = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ CA^4 \\ CA^5 \end{bmatrix} \quad (4.26)$$

Through substituting the data obtained in Section 3.4, it could be found that the observability matrix ϑ does not contain any 6×6 singular matrixes, hence, the system is completely state-observable. This result can be checked via MATLAB with the function 'obsv(A,C)', where A and C are system matrix and output matrix of the laboratory system respectively.

2. Check the rank of matrix C in MATLAB,

$$\text{rank}(C) = 3 \quad (4.27)$$

According to Equation (4.25), the minimal order s_0 is obtained:

$$\frac{n}{m} = 2 \leq s_0 \leq n - m + 1 = 5 \quad (4.28)$$

Therefore, the minimal order s_0 of parity space is two. However, the parity order s is set to be six in the next simulations and experiments as the larger the order s is, the better performance of FDD processes is. Moreover, a larger s provides more freedom of designs.

4.4 Fault Isolation Design

In the previous section, the fault detection process has been designed and analysed. It is easy to understand that a single residual is not enough to isolate the fault in FDD design. In [26], Gertler introduced several methods to enhance the ability of fault isolation using parity relation technique, such as diagonal residuals, directional residuals and structured residuals. This study chooses to design a set of structured residuals for fault isolation, which is easy to be implemented.

4.4.1 Residual Sensitivity

Before designing a set of structured residuals, a concept of residual sensitivity (or insensitivity) is introduced firstly, which is the basis of fault isolation.

To distinguish different faults, a residual is required to be insensitive to a specific fault, but sensitive to other faults. So that, it can be predicted that a residual is close to zero when a fault that the residual is insensitive to appears in a system, however, other residuals deviate from zero.

Chapter 4: Parity Relation Technique

Initially, a new notation for V_s is introduced:

$$V_s = [V_0, V_1, \dots, V_i], \quad i = 0, 1, \dots, s, \quad V_i \in \mathbb{R}^m$$

If a residual is designed to be insensitive to a sensor fault, the element of V_i that multiplies the signal of this sensor is set to zero. This means that there are $(s + 1)$ zeros per sensor fault in the vector V_s . For example, if a residual is designed to be insensitive to faults in the i^{th} sensor, the modified parity vector should be as follows:

$$V_s = [\underbrace{V_{0,1}, \dots, V_{0,i-1}, \mathbf{0}, V_{0,i+1}, \dots, V_{0,m}}_{V_0}, \dots, \underbrace{V_{i,1}, \dots, V_{i,i-1}, \mathbf{0}, V_{i,i+1}, \dots, V_{i,m}}_{V_s}] \quad (4.29)$$

Example:

Given a one-input two-output discrete system, the system model is assumed as below:

$$\begin{aligned} x(k+1) &= \mathbf{A}x(k) + \mathbf{B}u(k) \\ y(k) &= \mathbf{C}x(k) + \mathbf{D}u(k) + f_s(k) \end{aligned} \quad (4.30)$$

where $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$ and f_s is a sensor fault. The dimensions of matrixes \mathbf{A} and \mathbf{C} are both 2×2 , and \mathbf{B} and \mathbf{D} have the dimensions 2×1 . The order of parity space s is set to two.

According to Equation (4.16) and Equation (4.29), if a residual is designed to be insensitive to the fault in sensor 1 that senses the value of output y_1 , the parity vector V_s should be as below:

$$V_s = [\mathbf{0}, V_{s0,2}, 0, V_{s1,2}, 0, V_{s2,2}] \quad (4.31)$$

Meanwhile, another condition should be satisfied as follows:

$$[V_{s0,2}, V_{s1,2}, V_{s2,2}]H_1 = 0$$

where $H_1 = [(H(:, 2)); H(:, 4)); H(:, 6)]$ and H is referred to Equation (4.15). Note that only the columns of matrix H that are related with sensor 2 are left.

Therefore, the designed residual $R(k)$ is only sensitive to sensor fault in sensor 2 as whenever a fault occurs in sensor 1 or not, the residual is constantly zero like below:

$$R(k) = V_s[Y(k) - QU(k)] = V_s[F_s(k)] = V_s[f_{s1}(k-2); 0; f_{s1}(k-1); 0; f_{s1}(k); 0]^T = 0 \quad (4.32)$$

Consequently, through designing different residuals with altered sensitivities, it is able to isolate a fault. For this example, when a fault occurs in sensor 1, the residual designed in Equation (4.32) is close to zero, however, the residual designed in Equation (4.19) for the purpose of fault detection becomes non-zero; similarly, when a fault appears in sensor 2, both of the two residuals become non-zero. Hence, it is available to isolate the sensor faults in the example, but for more complex system (e.g. more than two sensors), a set of residuals are required to be designed in order to isolate the fault. Additionally, this method can be used to design residual for actuator fault.

4.4.2 Structured Residual Design

The residual sensitivity for two pairs of has been discussed in Section 4.4.1, however, it is not enough to fulfil FDD processes when a system has more than two sensors or actuators. In [27, 40, 41], Gertler and his colleagues presented the enhancement of residual and introduced several schemes to isolate particular fault, such as diagonal residuals, directional residuals and structured residuals. Through comparing these methods, the method of this study to FDD is to design structured residuals as it is a strongly isolating method and it is reliable and easy to implement. Note that to illustrate the design principle, only the sensor faults are considered in this section and the actuator faults are dealt with in the same way, which is explained in Chapter 5.

It has been mentioned that in order to isolate a fault, a set of residuals are designed. For example, it is assumed that a system with three sensors and all of them may malfunction. Three residuals R_1 , R_2 and R_3 are designed, each one of which is only insensitive to one fault but sensitive to others. Note that:

- R_1 is sensitive to the fault in sensor 2 and sensor 3, which means that when sensor 1 malfunctions, R_1 is still zero, but when a fault occurs in other sensors, it deviates from zero.
- R_2 is sensitive to the fault in sensor 1 and sensor 3, which means when sensor 2 malfunctions, R_2 is still zero, but when a fault occurs in other sensors, it deviates from zero.
- R_3 is sensitive to the fault in sensor 1 and sensor 2, which means when sensor 3 malfunctions, R_3 is still zero, but when a fault occurs in other sensors, it deviates from zero.

After constructing these residuals, a fault code table is built in Table 4.1.

Table 4.1: Fault code table

	f_1	f_2	f_3
R_1	0	1	1
R_2	1	0	1
R_3	1	1	0

In Table 4.1, each row stands for a residual and each column represents a sensor fault. A '1' in row i and column j means that residual R_i is sensitive to fault f_j , and a '0' in row i and column j implies that fault f_j does not affect residual R_i . Obviously, the fault code table depicts how the residual is impacted by faults.

It is assumed that only one fault can occur in the system at one time, and the multiple-fault isolation is part of the future work. Therefore, no column has more than one '0' and all of them are unique to identify a fault.

When the three residual are calculated by the control system, the faulty sensor could be isolated using Table 4.1 according to the principle as follows:

- Sensor 1 malfunctions if and only if when R_2 and R_3 fire , and R_1 does not
- Sensor 2 malfunctions if and only if when R_1 and R_3 fire , and R_2 does not
- Sensor 3 malfunctions if and only if when R_1 and R_2 fire , and R_3 does not

4.5 FDD Process Design and Validation

So far, the design of FDD process theoretically has been finished, the further study is to validate this method via different system models. The validation includes two stages: the validation on SISO system and the validation on MIMO system.

Assumption:

1. Only sensor faults and actuator faults are considered.
2. Only a fault occurs in the system at one time.

The validations of FDD processes for SISO and MIMO systems are based on these assumptions later. Furthermore, a crucial question is that how to simulate fault signals of a system. Actually, in practice, transducer (sensor or actuator) faults are usually classified as several situations:

- 1) The value of a transducer is zero or negative value constantly. This malfunction usually means the transducer power is cut off or it has been completely broken. This type of fault is the easiest fault to be detected since it can be found easily through limit-checking method. For a digital system, some functions can limit the value of transducers and alert the system if the reading of a transducer exceeds its limitation. Hence, this type of fault is not considered in this study.
- 2) The value of a transducer fluctuates or changes dramatically. This phenomenon indicates that there is a bias between the transducer reading and real data, and a fault may occur. To mimic this type of fault, a step function is used with some modifications to simulate a real fault. The further research of FDD mainly focuses on this type of fault.
- 3) The value of a transducer is slightly higher or lower than the right value, and the difference increases gradually by a tiny amount. This kind of fault is usually impossible to detect in the early stage. Sometimes, if the system is robust and tolerant to this transducer, this fault will be ignored until the negative effects of the fault exceed the system's tolerance. For this type of fault, it makes the system deteriorate in a long term and it is actually hard to detect and simulate at this stage. Hence, this kind of fault is not considered in the next validations and simulations.

In the next two Sections 4.5.1 and 4.5.2, validations are implemented based on a random SISO system and a MIMO system respectively. Note that the residual signals designed for sensor faults and actuator faults have a similar principle, so that only sensor fault is considered at this chapter.

The actuator fault detection and isolation will be designed and explained in Chapter 5 based on the laboratory model.

4.5.1 Example: SISO System FDD Design and Validation

A single-input single-output system (SISO) model in state-space representation is given:

$$x(k + 1) = Ax(k) + Bu(k) + Ff_a(k) \quad (4.33)$$

$$y(k) = Cx(k) + Du(k) + f_s(k) \quad (4.34)$$

where $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ and f_s means the sensor fault and f_a represents the actuator fault in this model. A , B , C , D and F are constant matrixes with appropriate dimensions. The dimensions of matrixes A , B , C and D are 2×2 , 2×1 , 1×2 and 1×1 respectively. In this system, A , B , C and D are known matrixes and the order of parity space is set to be four. As actuator faults are added to the input signals, F is set to be the same with matrix B . The data used in this section is attached in Appendix A. As a SISO system, it is assumed that only one sensor and one actuator are used in the system. So that only the sensor fault (or actuator fault) detection is required to design, and fault isolation is not necessary.

Following the principle of Section 4.3, the codes for FD process are designed properly. Figure 4.1 shows the construction of SISO Simulink model. The input and output data are saved via the Block 'To Workspace' for FDD design in Editor later. The fault signal which is a unit step and occurs at 2 sec is injected into the system through a multiport switch. When the fault switch is set as '2', the system acts without sensor fault. However, when the switch is set as '1', a step input signal with proper magnitude which acts as the fault signal is added to the system output signal.

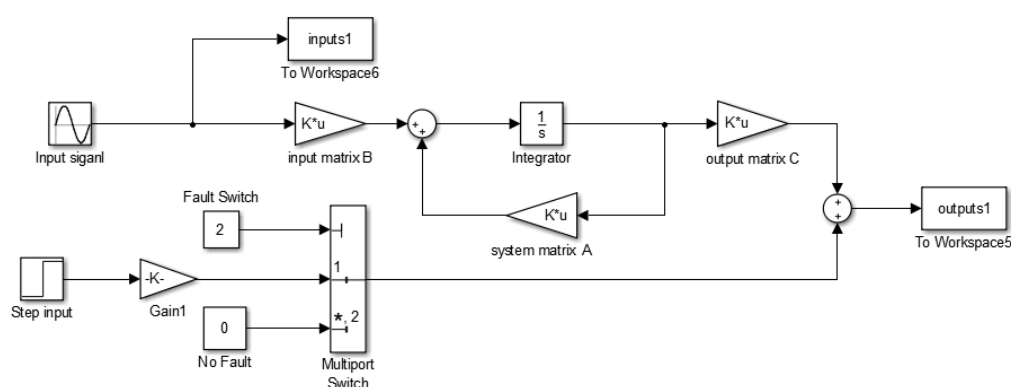


Figure 4.1: Simulink model of a random SISO system for FDD validation

The simulation result of fault detection design for SISO random system is depicted in Figure 4.2 and Figure 4.3. Figure 4.2 sketches the fault detection residual in the fault-free system, and Figure 4.3

Chapter 4: Parity Relation Technique

shows the residual when a sensor fault occurs. Obviously, once the fault signal is introduced into the system at 2 sec, the designed residual increases quickly.

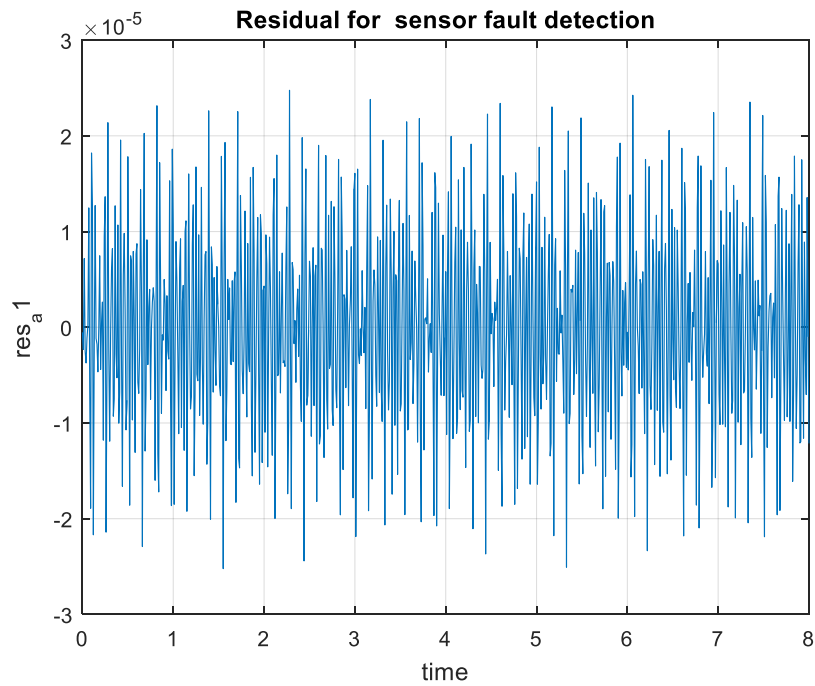


Figure 4.2: Residual response in fault-free SISO system

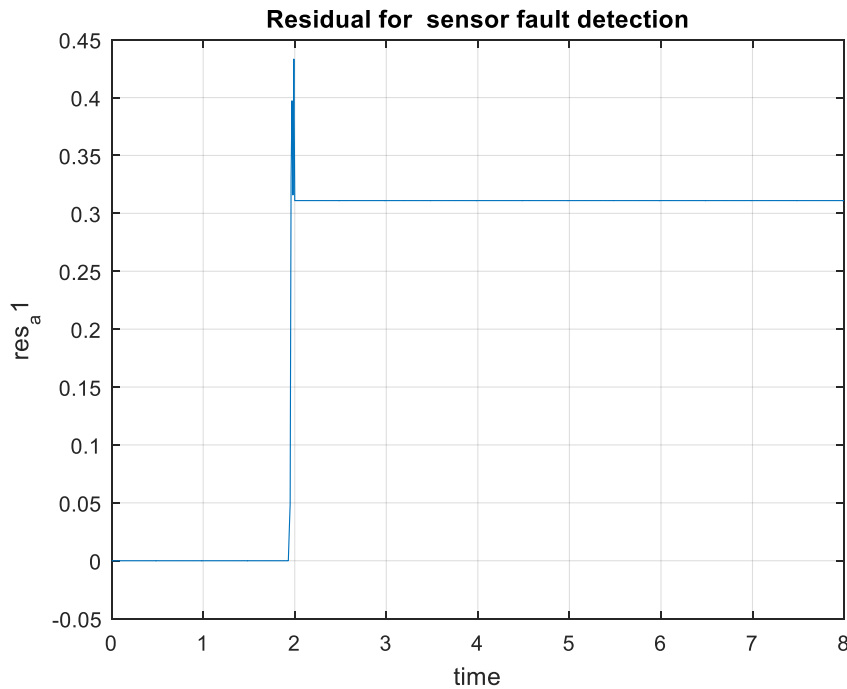


Figure 4.3: Residual response with sensor fault in SISO system

4.5.2 Example: MIMO System FDD Design and Validation

After the FDD processes for SISO case designed by parity relation technique is successfully validated through 'MATLAB Simulink', the application of parity relation technique is extended to a random MIMO system. The data used in this section is attached in Appendix B. The simulated fault signals has the same property with that used in SISO systems. The simulation model is shown in Figure 4.4.

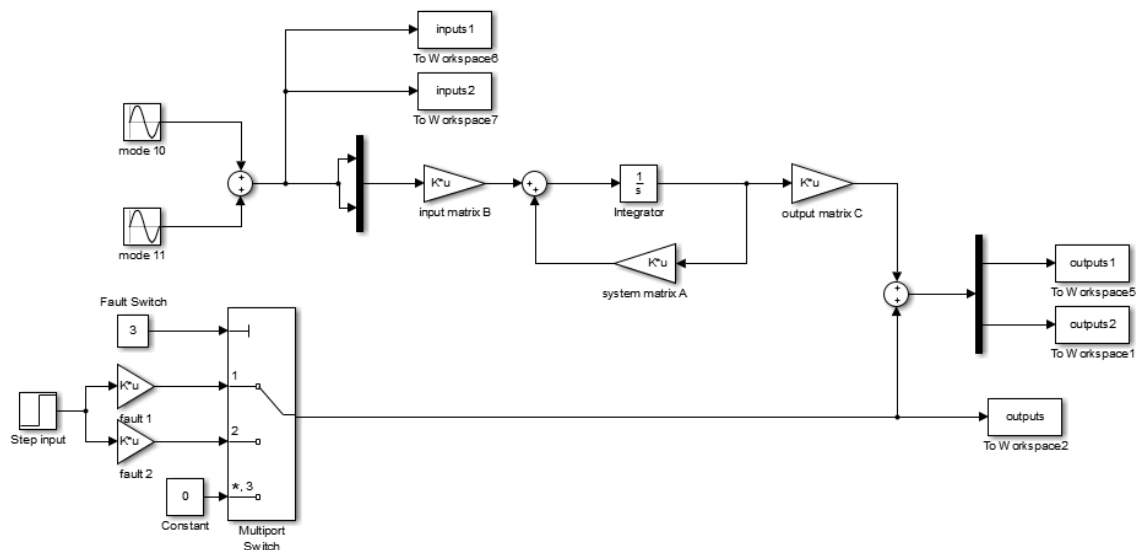


Figure 4.4: Simulink model of a MIMO system for FDD validation

The design procedure is similar with SISO system, whereas two sensors are assumed in this MIMO system, so that the residuals for fault isolation are required. The simulation results of fault detection design for MIMO random system is described in Figure 4.5, Figure 4.6 and Figure 4.7 for different cases. Figure 4.5 sketches different residuals' plots when no sensor fault appears in the MIMO system. It is seen clearly that when the system is fault free, all of three residuals are pretty small and close to zero. Figure 4.6 shows the change of these residuals when a fault occurs in sensor 1, in which the residual designed for sensor 2 and fault detection dramatically increase when the fault appears in sensor 1, but the residual generated for sensor 1 is still close to zero. According to the fault code table in Section 4.5, the change of this set of residuals indicates that sensor 1 is likely to have malfunction. Similar to Figure 4.6, Figure 4.7 displays the plots of residuals when sensor 2 is broken. Note that whenever a fault occurs in sensor 1 or sensor 2, the residual for fault detection always non-zero.

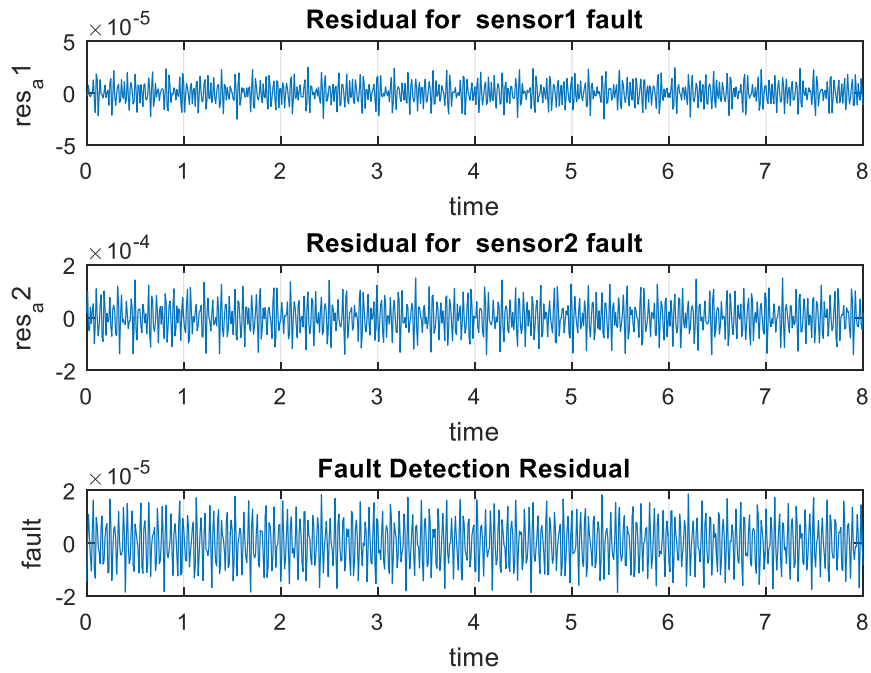


Figure 4.5: Residual responses without sensor fault in MIMO

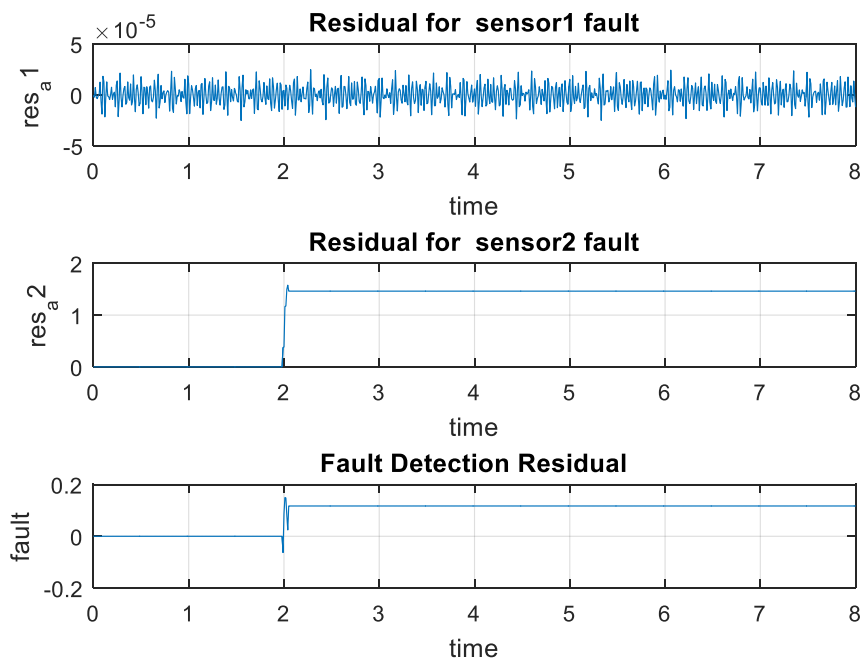


Figure 4.6: Residual responses with sensor 1 fault in MIMO

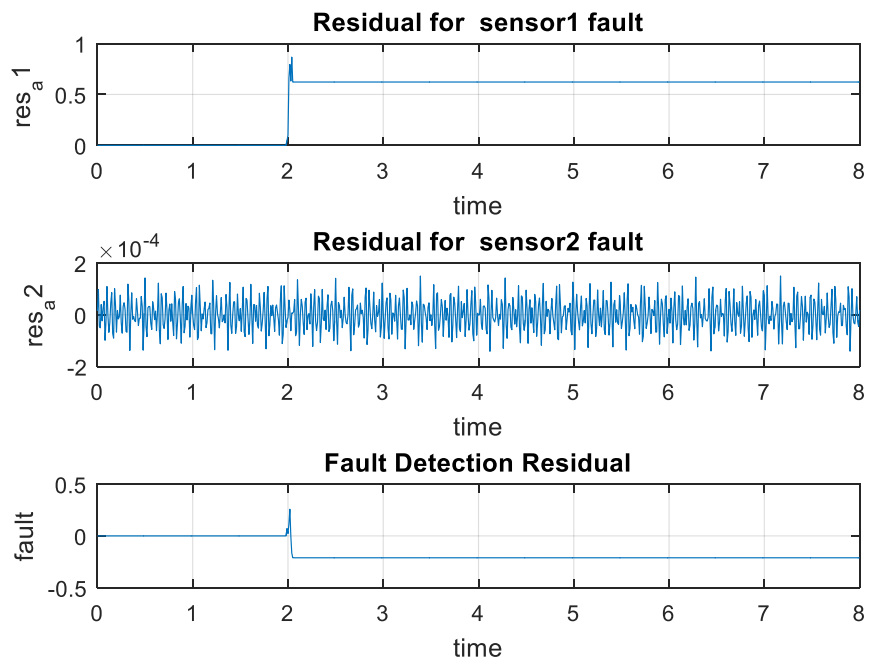


Figure 4.7: Residual responses with sensor 2 fault in MIMO

Chapter 5 : Simulation

In Chapter 4, the designs of FDD processes for various situations have been implemented and the validations of random SISO and MIMO systems have been checked as well. However, it is required to prove whether the parity relation technique can be used for FDD process design regarding to a vibration isolation system. Hence, a simulated implementation of single sensor fault and actuator fault detection and isolation on the plate structure is presented in this chapter. Besides, the discussions of the simulated results are stated and some useful suggestions are provided for further research.

5.1 Simulation Setup with Sensor Fault

In Chapter 4, the system identification of the laboratory plant has been derived successfully and all the parameters of the mathematical model have been obtained, so that the simulation model based on these data could be built in MATLAB Simulink, as shown in Figure 5.1.

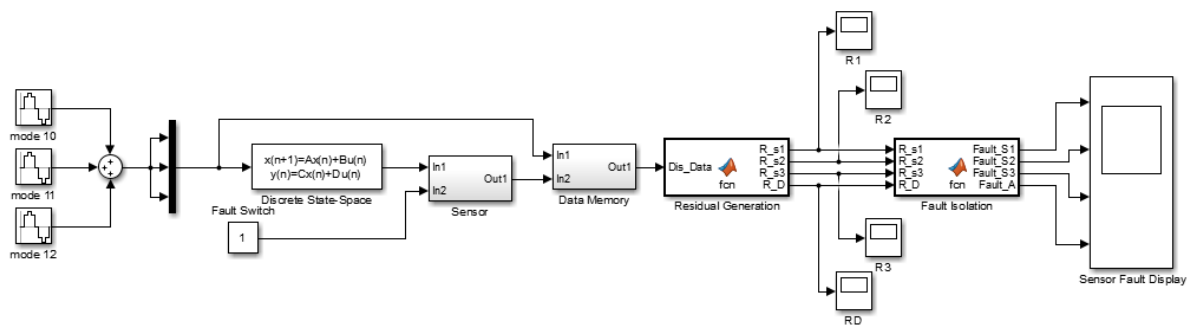


Figure 5.1: The design of FDD processes in Simulink

The simulation structure consists four main parts as follows:

1. Discrete system model of the laboratory plant

As shown in Figure 5.1, in order to mimic the real disturbance force that imposed on the laboratory plant the input signals of the model is the sum of three sine wave signal generators whose frequencies are the natural frequency of three modes, 141.3088 rad/s, 176.11768rad/s and 219.4088 rad/s respectively. (See Chapter 3).

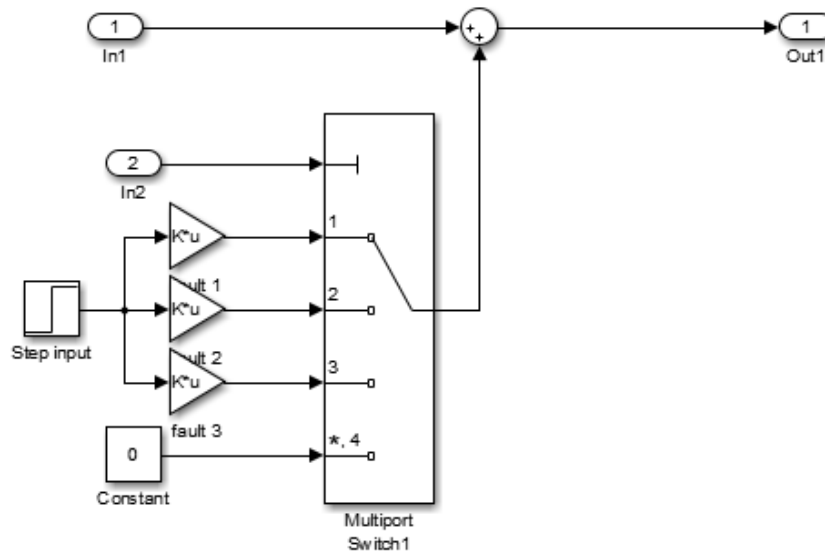


Figure 5.2: Sensor fault signal design

2. Data memory block

It has been mentioned that the parity space order s is set to 6, the same with the system order. Therefore, in order to use parity relation technique, the history data of $(s + 1)$ steps are demanded, including input signals and output signals. Figure 5.3 displays the inner structure of the data memory block. Input signal from port In1 and In2 are system input and output signals at time k , and output signal exported to port Out1 is all the organized input-output data from time $(k - s)$ to time (k) . These data are manipulated to produce the fault residuals.

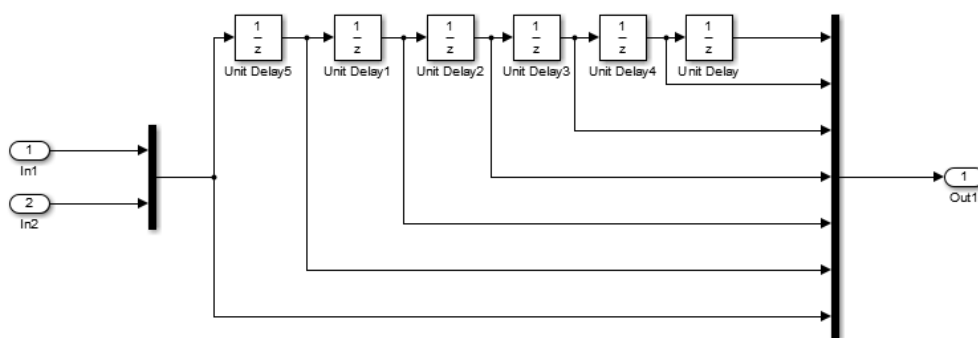


Figure 5.3: Inner structure of data memory block

3. Residual generation block

As what have been discussed in Chapter 5, a set of residuals are designed for the purposes of sensor fault detection and isolation respectively. A 'MATLAB function' block is placed in the simulation model, which can combine the script and Simulink together, so that

implementation of FDD using parity relation method is able to be run online instead of saving the system data first and then calculating residuals. This improvement can reduce the time consumed by FDD process and alert the system quickly.

In the Residual generation block, four parity vectors are designed corresponding to four residuals used to detect and isolate a fault. Particularly, V_{s_FD} is designed for fault detection, and V_{s_s1} , V_{s_s2} and V_{s_s3} are designed to be insensitive to sensor 1 fault, sensor 2 fault and sensor 3 fault respectively. The designed parity vectors are listed in Table 5.1.

Table 5.1: Designed parity vectors for sensor fault detection and isolation

$V_{s_FD}^T$	$V_{s_s1}^T$	$V_{s_s2}^T$	$V_{s_s3}^T$
0.0925	0.0000	-0.1154	-0.1192
-0.0462	-0.2445	0.0000	0.1470
0.0835	0.3720	0.1497	0.0000
-0.0548	0.0000	0.0791	0.0668
-0.1608	0.2348	0.0000	-0.2170
0.4226	-0.1014	-0.2487	0.0000
-0.0688	0.0000	0.3235	0.3346
-0.2620	-0.1626	0.0000	0.0184
0.7804	0.4814	0.0229	0.0000
-0.0013	0.0000	0.0098	0.0208
0.1407	0.1357	0.0000	-0.1279
0.1194	0.1045	-0.1109	0.0000
0.0623	0.0000	0.7820	0.7898
0.0943	0.5603	0.0000	-0.0331
0.0704	-0.3555	-0.0345	0.0000
0.1499	0.0000	0.0094	-0.0011
-0.0476	0.0485	0.0000	0.0048
-0.0375	0.0604	-0.0007	0.0000
-0.0628	0.0000	0.3744	0.3604
0.0705	0.0499	0.0000	0.1694
0.0746	0.0103	0.1601	0.0000

4. Fault isolation block

Despite of the fact that a residual generator is created to indicate the occurrences of faults, a residual evaluation and decision-maker block is necessary to isolate a fault directly. Usually, there are two main strategies to evaluate the residuals, one of which is norm-based residual evaluation and another one is statistic-based residual evaluation. This study uses the second method as the statistics of the experiment have been recorded.

5.2 Sensor Fault Simulation Results

Based on the simulation built in Section 5.1, the simulation results of sensor faults are illustrated and discussed in this section. The illustrated results are considered according to four cases, including fault-free, sensor 1 fault, sensor 2 fault and sensor 3 fault and are analysed in the next four sections respectively.

Additionally, for each fault case, a set of residuals are designed. Specifically, residual R_1 , R_2 and R_3 are not sensitive to fault 1, fault 2 and fault 3 individually and residual R_d is designed for the purpose of fault detection.

5.2.1 Simulation Results of Sensor Fault for Case 1: No Fault

This section displays the simulation results when no fault occurs in the system. Figure 5.4, Figure 5.5, Figure 5.6 and Figure 5.7 display the plots of the designed four residuals respectively. The implementation of FDD processes can be conducted through analysing these plots of residuals.

Clearly, it is seen that all of four residual signals are small and their values are less than 1×10^{-4} , which proves that there is no fault happened in the system.

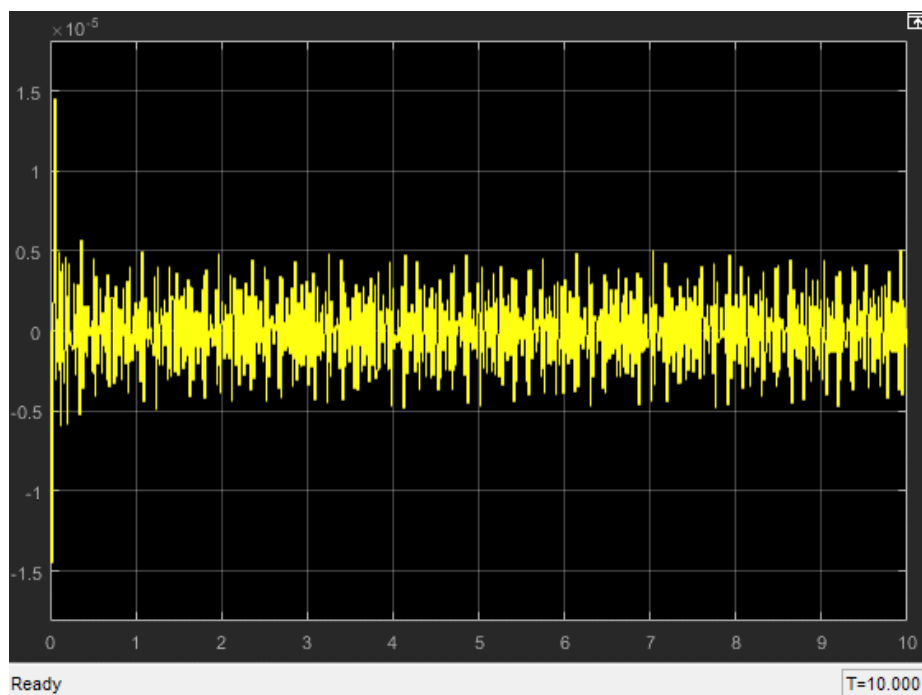


Figure 5.4: Case 1 - The responses of residual R_1 for sensor fault

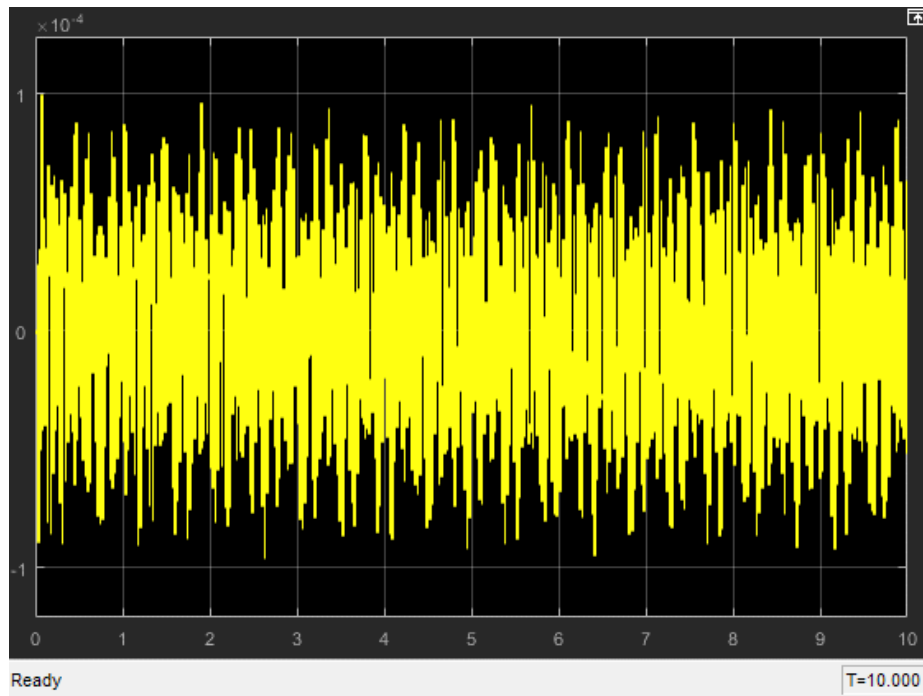


Figure 5.5: Case 1 - The responses of residual R_2 for sensor fault

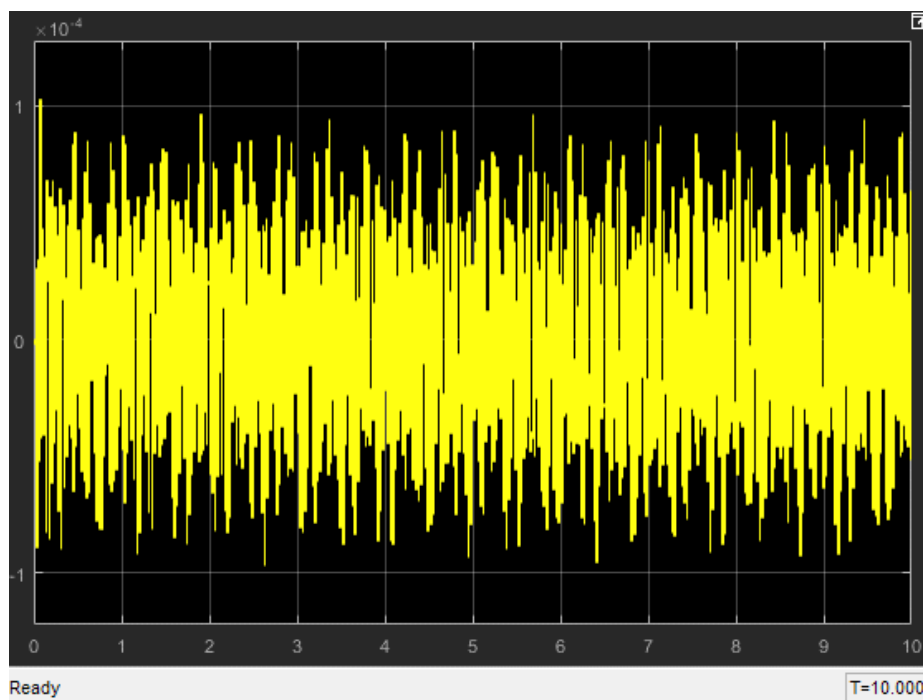


Figure 5.6: Case 1 - The responses of residual R_3 for sensor fault

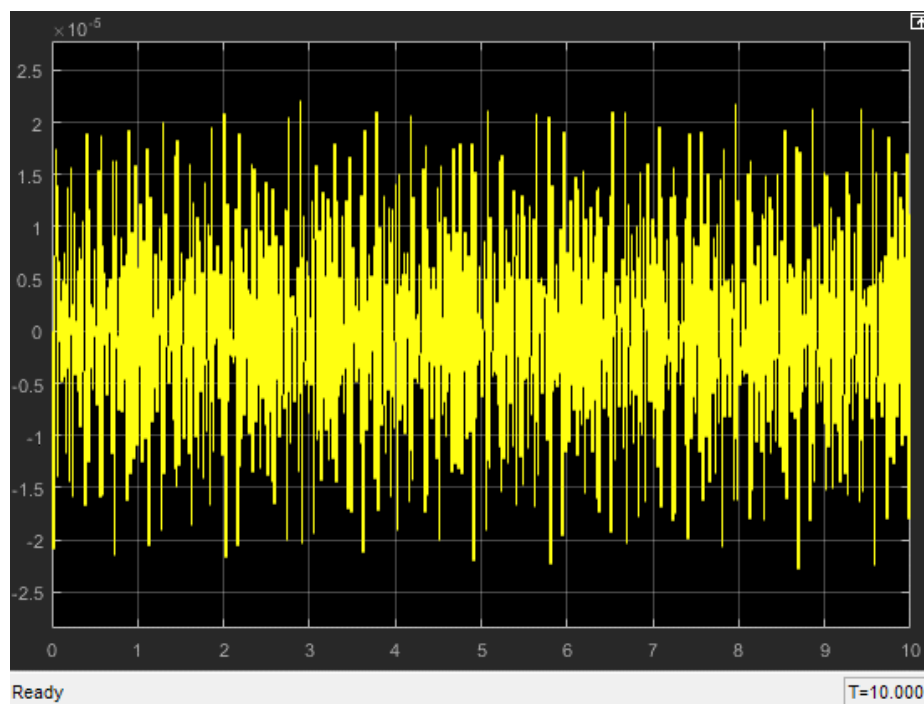


Figure 5.7: Case 1 - The responses of residual R_d for sensor fault

5.2.2 Simulation Results of Sensor Fault for Case 2: Sensor 1 Fault

In this section, a fault signal is introduced into sensor 1 of the analysed system at 4-second time with a magnitude of 0.1. Figure 5.8, Figure 5.9, Figure 5.10 and Figure 5.11 display the plots of the four residuals respectively when a fault appears in sensor 1.

It can be found that all of four residual signals are still minor before the fault appears in sensor 1. All the residual signals fluctuate at 4-second time. Additionally, the value of residual R_1 is still very small, about -0.5×10^{-5} unit, as shown in Figure 5.8. The value of residual R_2 and R_3 increase dramatically and are times larger than before. Meanwhile, the value of residual R_d also enlarges about 1×10^3 times, which states that a fault happened at 4-second time. This is due to the fact that residual R_1 is insensitive to sensor 1 fault and the other residuals are sensitive to sensor 1 fault, so that once a bias fault arises in sensor 1, residual R_2 and R_3 differ from zero and residual R_1 is still close to zero. Hence, it can deduce that sensor 1 may be broken.

In conclusion, through observing the change of the set of designed residuals, FDD process for sensor 1 can be implemented successfully.

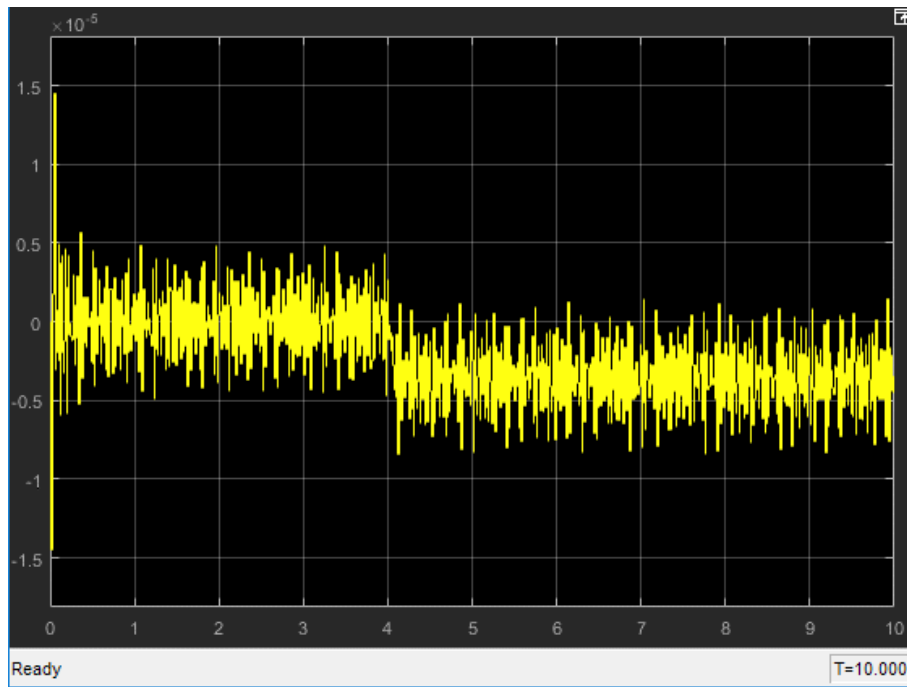


Figure 5.8: Case 2 - The responses of residual R_1 for sensor fault

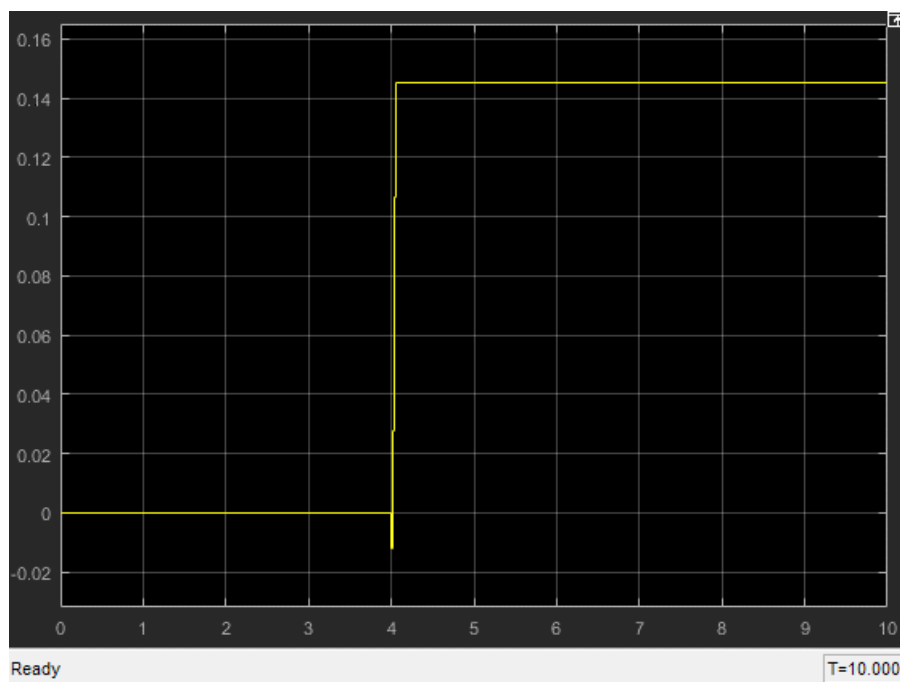


Figure 5.9: Case 2 - The responses of residual R_2 for sensor fault

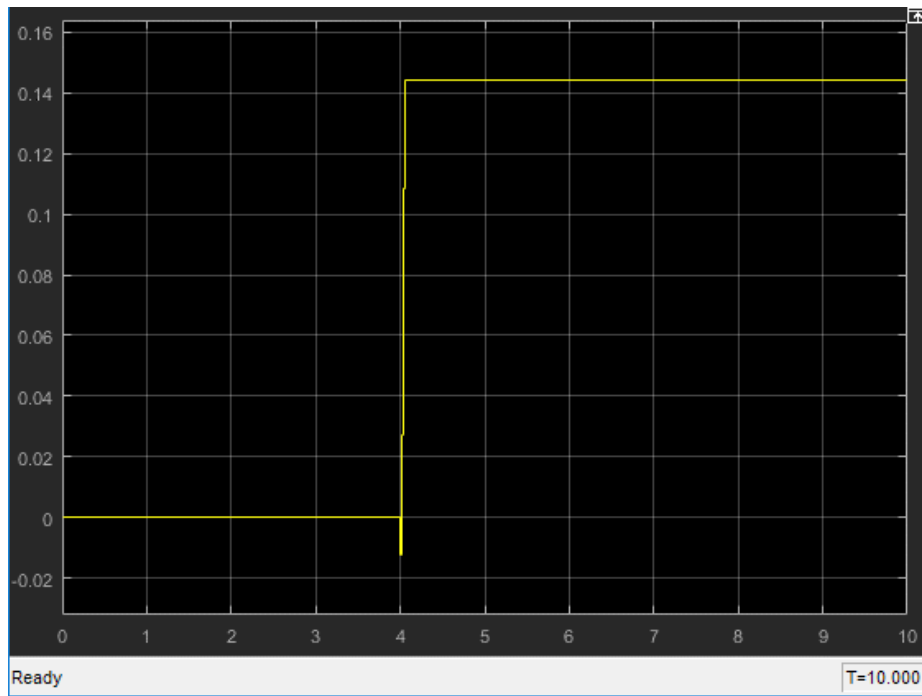


Figure 5.10: Case 2 - The responses of residual R_3 for sensor fault

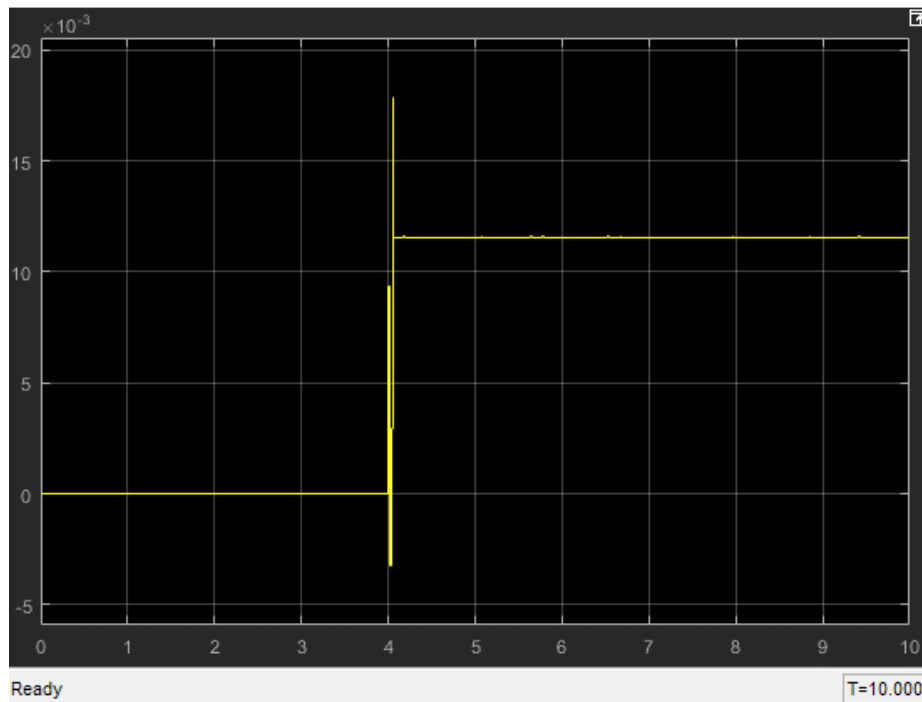


Figure 5.11: Case 2 - The responses of residual R_d for sensor fault

5.2.3 Simulation Results of Sensor Fault for Case 3: Sensor 2 Fault

A pre-defined fault signal is added into sensor 2 and the simulation results are analysed in this section. Figure 5.12, Figure 5.13, Figure 5.14 and Figure 5.15 display the curves of the four designed residuals respectively when a fault appears in sensor 2.

Chapter 5: Simulation

It has known that the fault occurred at 3-second time. The simulation results show that before the fault happened, all of three residuals are close to zero. Nevertheless, when sensor 2 has a malfunction, residual R_1 and residual R_3 are not close to zero anymore and have a rise, reaching 0.06 and 0.005 unit respectively, but residual R_2 does not have a distinct change. In addition, residual R_d is almost zero at the beginning, but there is a dramatic rise after the fault is injected.

Consequently, the simulated results illustrate that a fault happened in the system after running for 3 seconds and the location of the fault is most possibly in sensor 2. Hence, the fault in sensor 2 can be quickly detected and isolated.

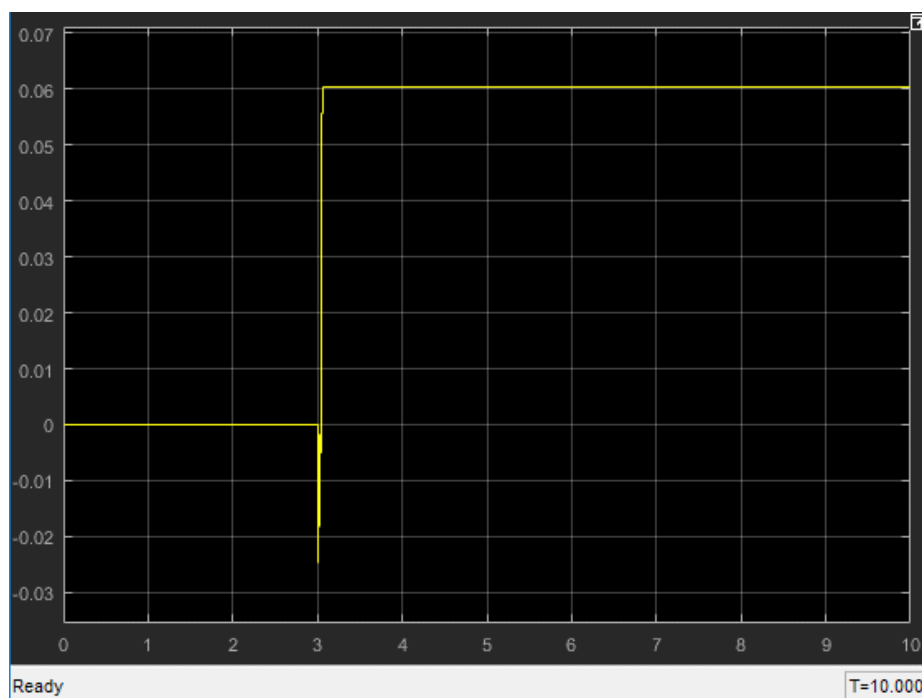


Figure 5.12: Case 3 - The responses of residual R_1 for sensor fault

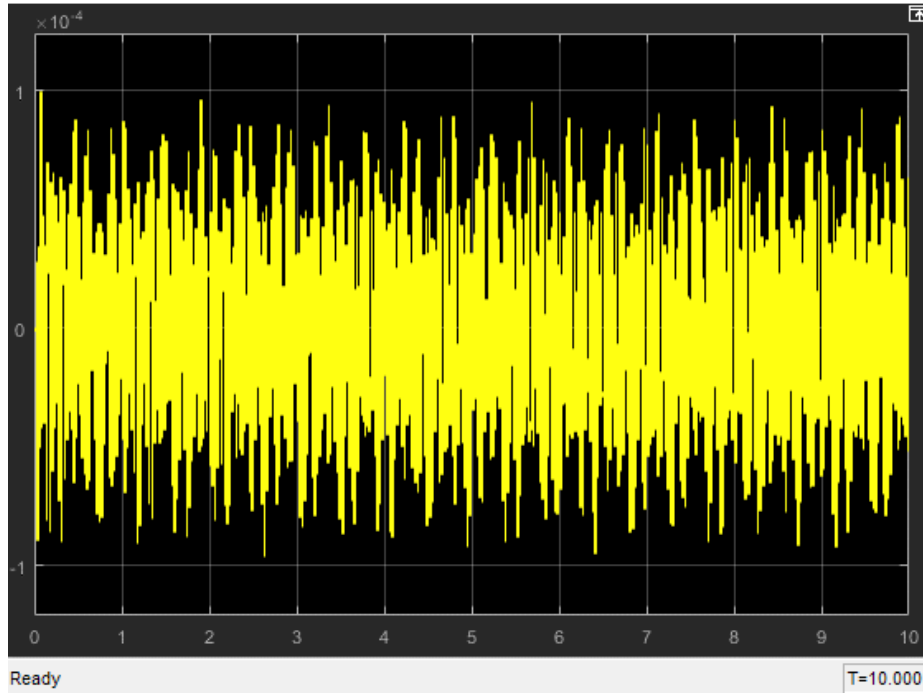


Figure 5.13: Case 3 - The responses of residual R_2 for sensor fault

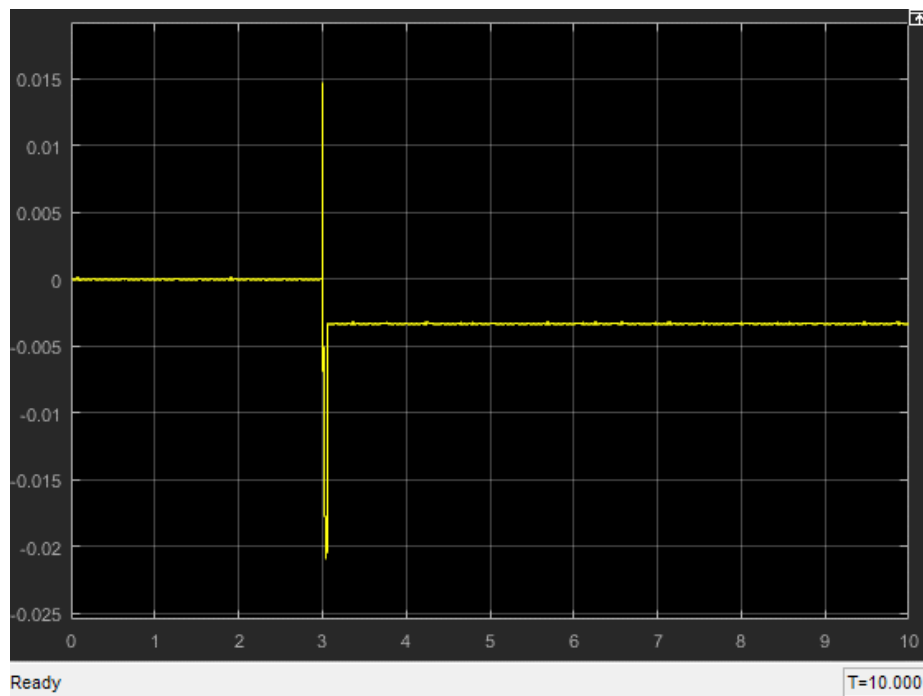


Figure 5.14: Case 3 - The responses of residual R_3 for sensor fault

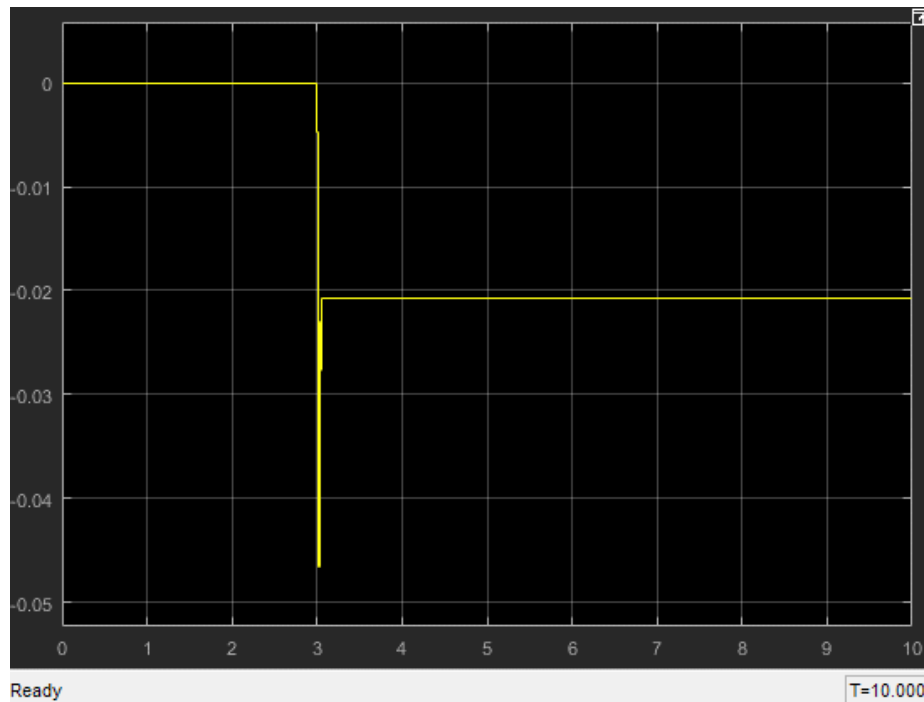


Figure 5.15: Case 3 - The responses of residual R_d for sensor fault

5.2.4 Simulation Results of Sensor Fault for Case 4: Sensor 3 Fault

A different fault is imposed on sensor 3 in this section, which happens at 3-second time, and lasts for 5 seconds then disappears. The simulation results are shown in Figure 5.16, Figure 5.17, Figure 5.18 and Figure 5.19.

As shown in Figure 5.16 and Figure 5.17, residual R_1 and residual R_2 are close to zero during the period from 0-second to 3-second. When the pre-defined fault exists, they increase or decrease to a non-zero value. Once, the fault is fixed at 8-second time, both of them become zero again. Besides, residual R_d reflects when a fault occurs in the system properly in Figure 5.19.

However, residual R_3 in Figure 5.18 is approximate to zero constantly whenever a fault happens or not. This result supports the design principle that residual R_3 is not insensitive to sensor fault 3.

To conclude, the responses of the three residuals reflect whether there is a fault in sensor 3, and the residuals return to zero to prove that the system is fault free when the fault disappears.

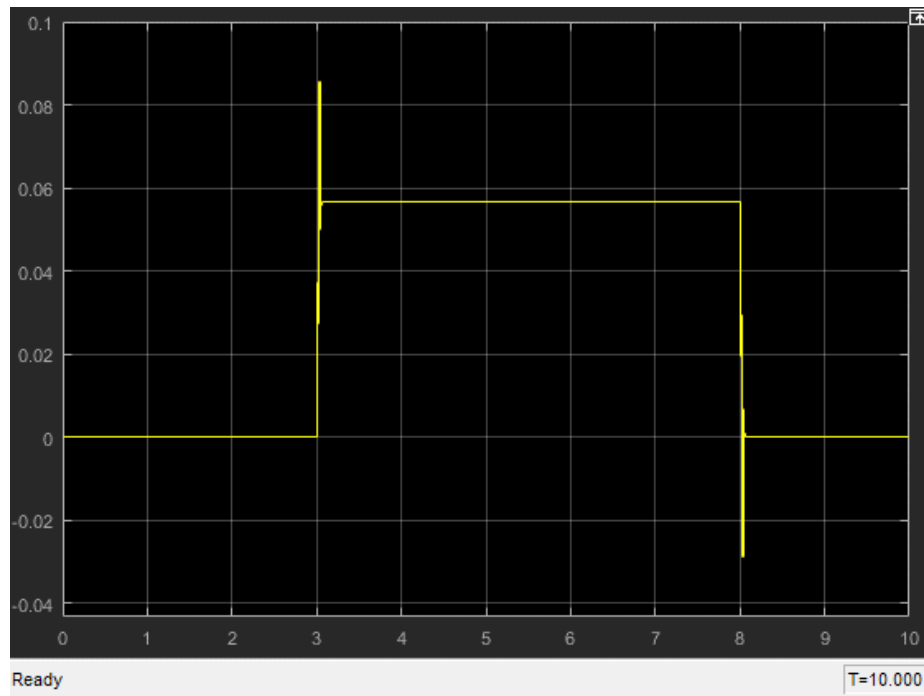


Figure 5.16: Case 4 - The responses of residual R_1 for sensor fault

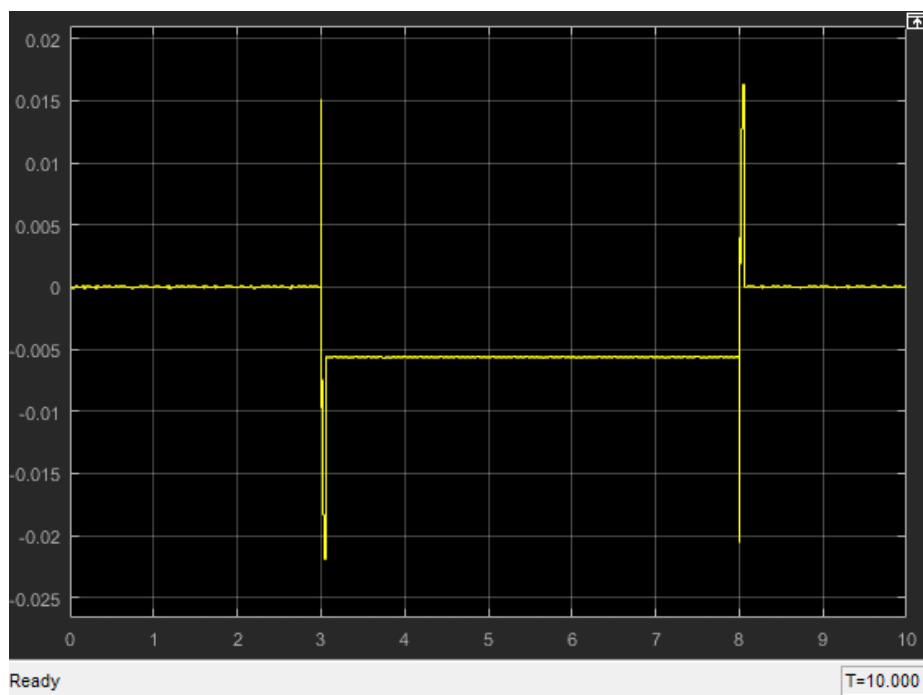


Figure 5.17: Case 4 - The responses of residual R_2 for sensor fault

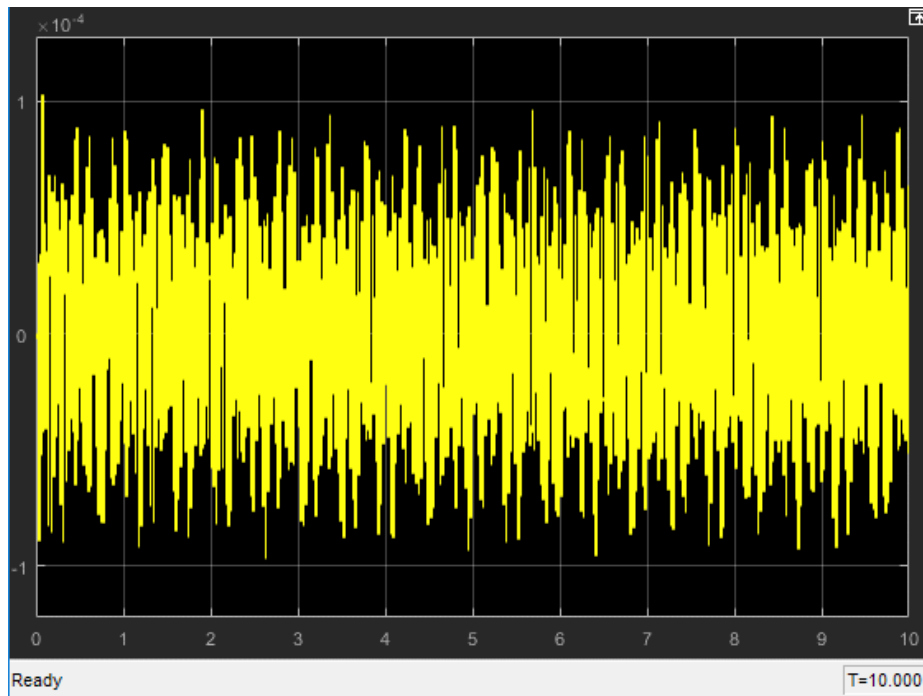


Figure 5.18: Case 4 - The responses of residual R_3 for sensor fault

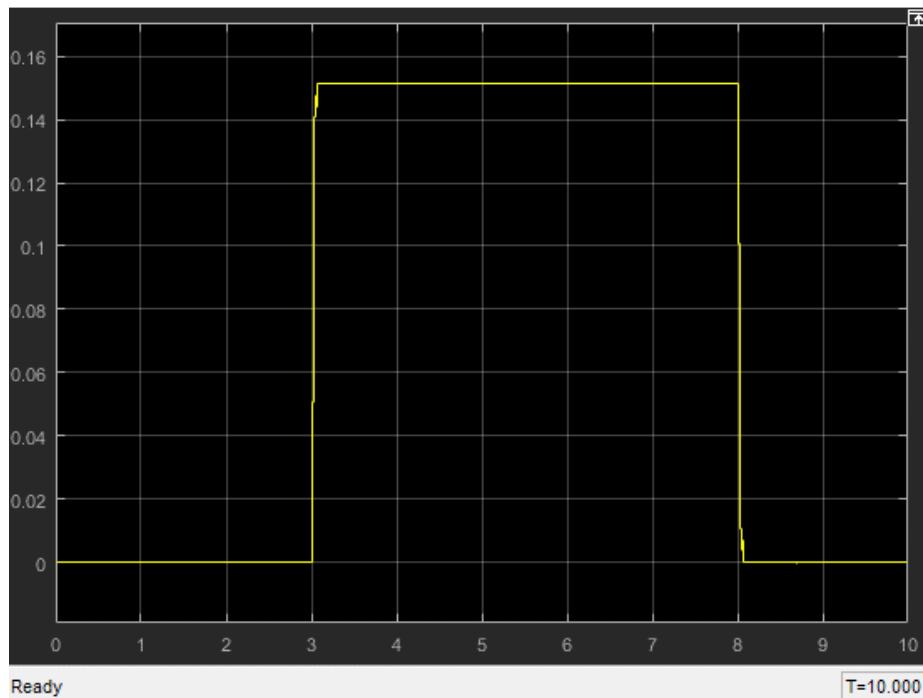


Figure 5.19: Case 4 - The responses of residual R_d for sensor fault

5.3 Simulation Results of Decision-making Block

The decision-making block evaluates the residuals and concludes the fault information, which is prepared for further study of fault accommodation. The simulation results are shown in Figure 5.20, Figure 5.21, Figure 5.22 and Figure 5.23 corresponding to different fault cases individually.

Chapter 5: Simulation

It can be seen that there are four signals in each graph, defined as 'indicator of sensor 1 fault', 'indicator of sensor 2 fault', 'indicator of sensor 3 fault' and 'indicator of any sensor fault'. The '1' value of these signals means a fault exists in corresponding location; the '0' value represents that no fault happens.

- Figure 5.20 displays the fault information exported by Decision-making block when no fault signal is added onto any sensor, and it can be observed that the values of four fault indicator signals are zero.
- As shown in Figure 5.21, the values of signal 'sensor 1 fault indicator' and signal 'any sensor fault indicator' are one and the other two signals remain zero when a fault signal is added on sensor 1.
- Figure 5.22 describes the fault information when a fault occurs in sensor 2, in which the values of signal 'sensor 2 fault indicator' and signal 'any sensor fault indicator' become one to indicate the occurrence of the fault and the other signals maintain at zero.
- Figure 5.23 displays the fault information when a fault signal is added on sensor 3. The signal 'sensor 2 fault indicator' and signal 'any sensor fault indicator' follows the shape of the injected fault signal in Section 5.2.4.

In conclusion, it has proven that the fault information produced by the decision-making block is correct because the results displayed by these indicators are the same with what has been discussed in Section 5.2 regarding to different fault cases.

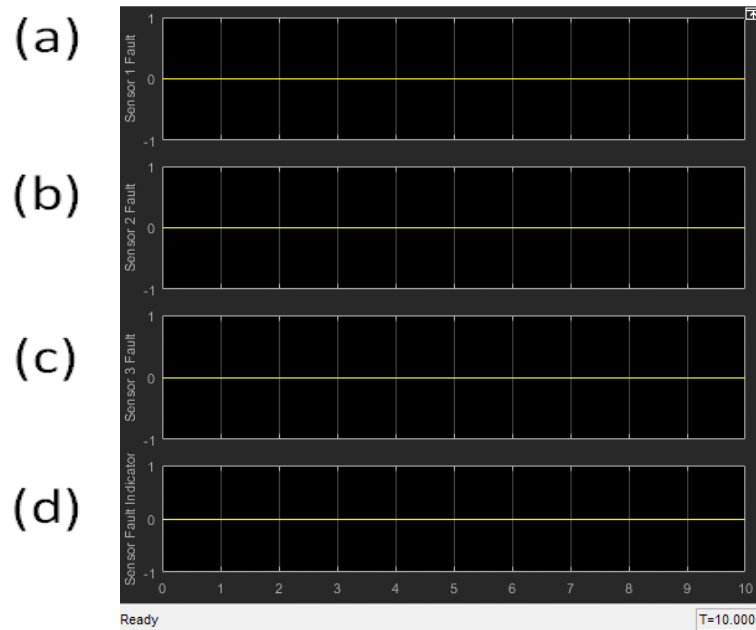


Figure 5.20: Fault information when no fault exists in the system. (a)-fault indicator for sensor 1: '0' means no fault and '1' means sensor 1 malfunctions. (b)-fault indicator for sensor 2: '0' means no fault and '1' means sensor 2 malfunctions. (c)-fault indicator for sensor 3: '0' means no fault and '1' means sensor 3 malfunctions. (d)-fault indicator for all sensors: '0' means no fault and '1' means one of the three sensor malfunctions.

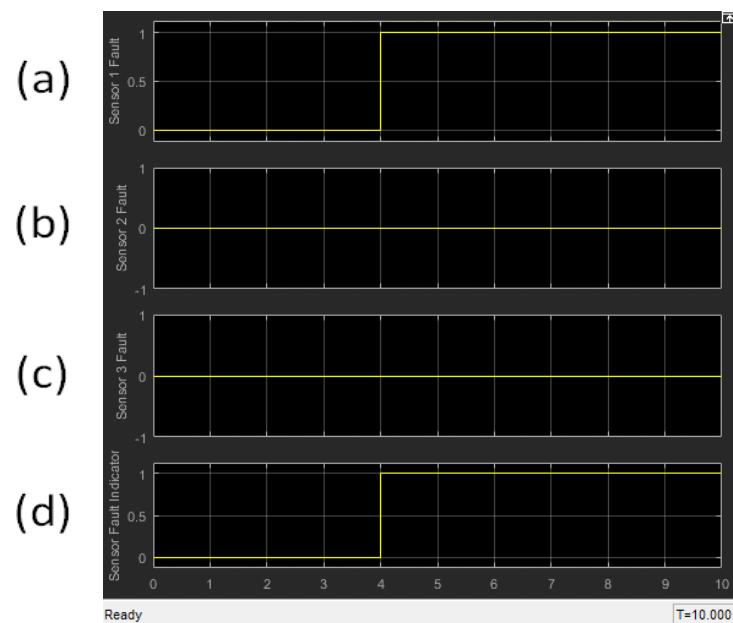


Figure 5.21: Fault information when a fault exists in sensor 1. (a)-fault indicator for sensor 1: '0' means no fault and '1' means sensor 1 malfunctions. (b)-fault indicator for sensor 2: '0' means no fault and '1' means sensor 2 malfunctions. (c)-fault indicator for sensor 3: '0' means no fault and '1' means sensor 3 malfunctions. (d)-fault indicator for all sensors: '0' means no fault and '1' means one of the three sensor malfunctions

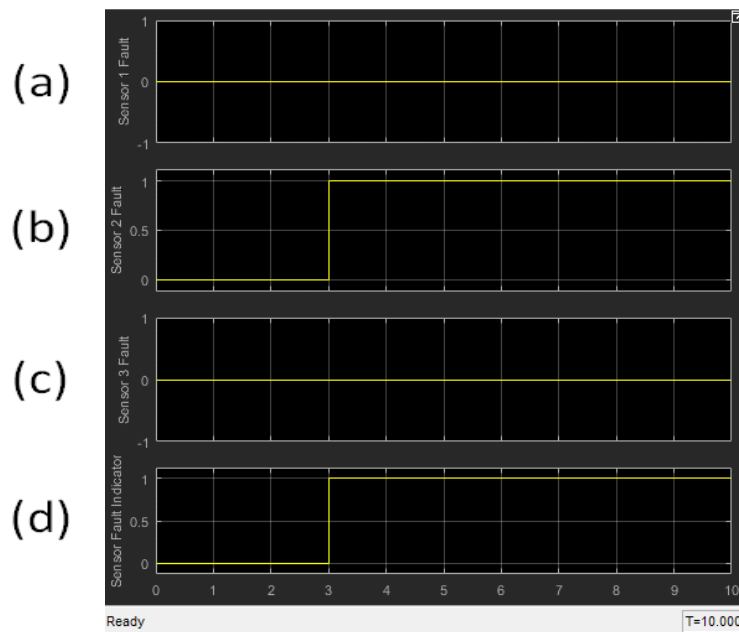


Figure 5.22: Fault information when a fault exists in sensor 2. (a)-fault indicator for sensor 1: '0' means no fault and '1' means sensor 1 malfunctions. (b)-fault indicator for sensor 2: '0' means no fault and '1' means sensor 2 malfunctions. (c)-fault indicator for sensor 3: '0' means no fault and '1' means sensor 3 malfunctions. (d)-fault indicator for all sensors: '0' means no fault and '1' means one of the three sensor malfunctions

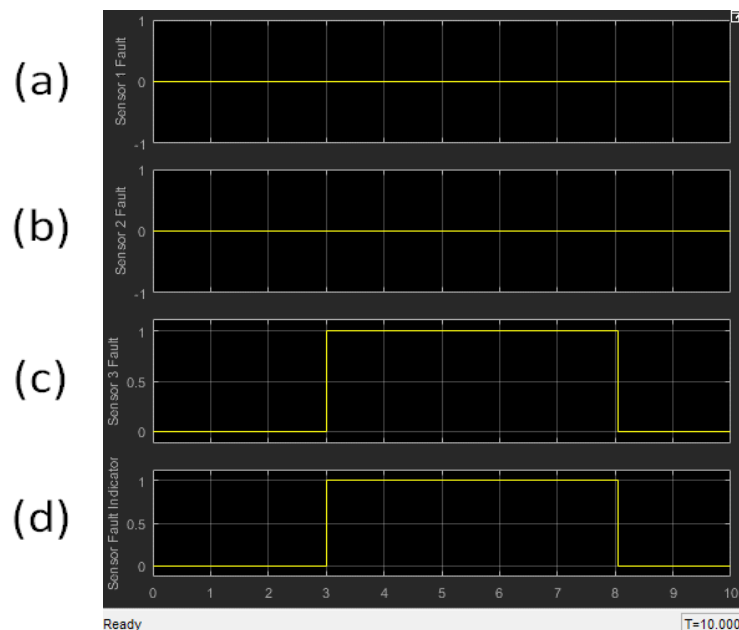


Figure 5.23: Fault information when a fault exists in sensor 3. (a)-fault indicator for sensor 1: '0' means no fault and '1' means sensor 1 malfunctions. (b)-fault indicator for sensor 2: '0' means no fault and '1' means sensor 2 malfunctions. (c)-fault indicator for sensor 3: '0' means no fault and '1' means sensor 3 malfunctions. (d)-fault indicator for all sensors: '0' means no fault and '1' means one of the three sensor malfunctions

5.3 Simulation Setup with Actuator Fault

Similar to the model built in Section 5.1, Figure 5.21 illustrates the simulation model for actuator fault.

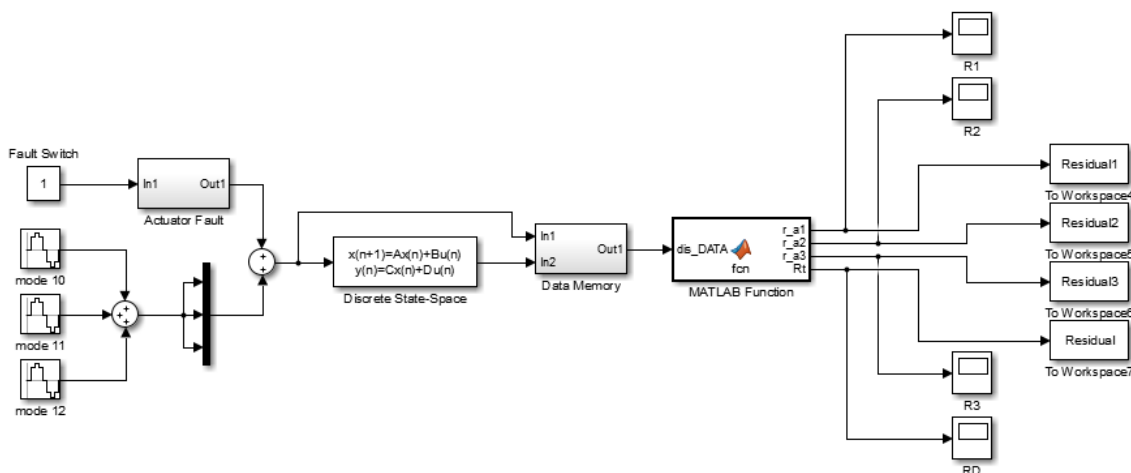


Figure 5.24: Simulation model with actuator fault

5.4 Actuator Fault Simulation Results

The simulation results of actuator fault are illustrated and discussed in this section. The simulated results are analysed for four different cases, including fault-free, actuator 1 fault, actuator 2 fault and actuator 3 fault, and each case is analysed in the next four sections respectively.

In each case, four residuals are designed to implement fault detection and isolation. Specifically, the designed residual R_1 , R_2 and R_3 are not sensitive to fault 1, fault 2 and fault 3 individually, and the residual R_d are created to detect fault.

Table 5.2: Designed parity vectors for actuator fault detection and isolation

$V_{s_FD}^T$	$V_{s_a1}^T$	$V_{s_a2}^T$	$V_{s_a3}^T$
-0.0802	-0.0590	-0.3118	-0.0574
-0.0658	-0.0892	0.1019	-0.1091
-0.2541	0.1661	0.1697	0.1729
0.1613	0.1538	-0.0222	0.2047
0.3523	-0.0860	-0.0249	-0.0863
-0.2862	-0.0808	0.0488	-0.0902
-0.0155	0.1819	0.5209	0.1405
-0.1043	0.1728	-0.3085	0.2110
0.7718	-0.3990	-0.1282	-0.3615
-0.0401	-0.2436	-0.1198	-0.2920
-0.0951	0.2601	0.2463	0.2487

-0.1990	-0.0099	-0.1626	-0.0114
-0.0357	-0.2265	-0.0240	-0.1993
-0.0580	-0.3127	0.1594	-0.3585
-0.1420	0.6030	-0.1536	0.5767
-0.0088	0.0987	0.1036	0.0992
-0.0002	0.0085	0.1414	0.0107
-0.0645	-0.1229	-0.2497	-0.1025
0.0327	0.1251	-0.2137	0.1368
0.0697	0.0011	-0.1803	-0.0007
0.0252	-0.1443	0.3935	-0.1236

5.4.1 Simulation Results of Actuator Fault for Case 1: No Fault

In this section, the simulation results when no fault occurs in the system are displayed and discussed. Figure 5.22, Figure 5.23, Figure 5.24 and Figure 5.25 show the curves of the designed four residuals respectively. The implementation of FDD processes can be conducted through comparing these residuals with the fault code Table 5.1 in Section 5.4 with replacing sensor fault by actuator fault.

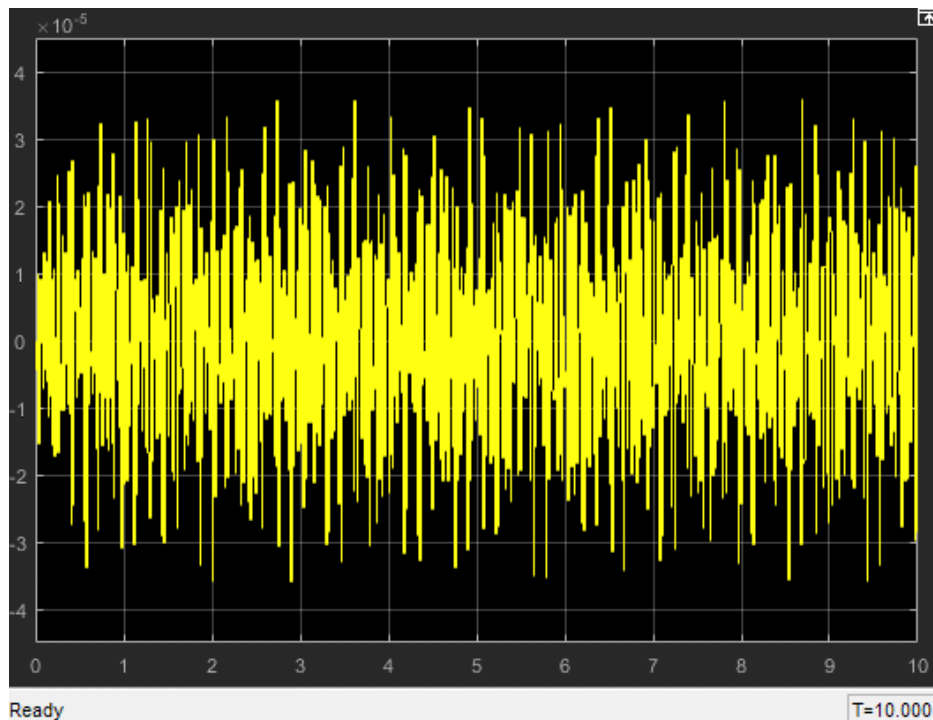


Figure 5.25: Case 1 - The responses of residual R_1 for actuator fault

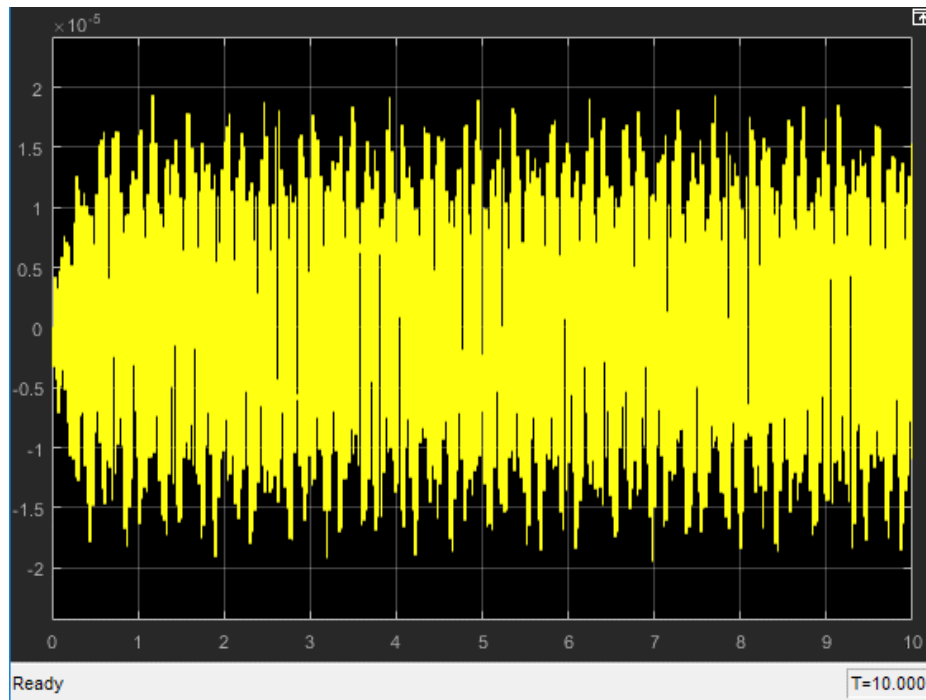


Figure 5.26: Case 1 - The responses of residual R_2 for actuator fault

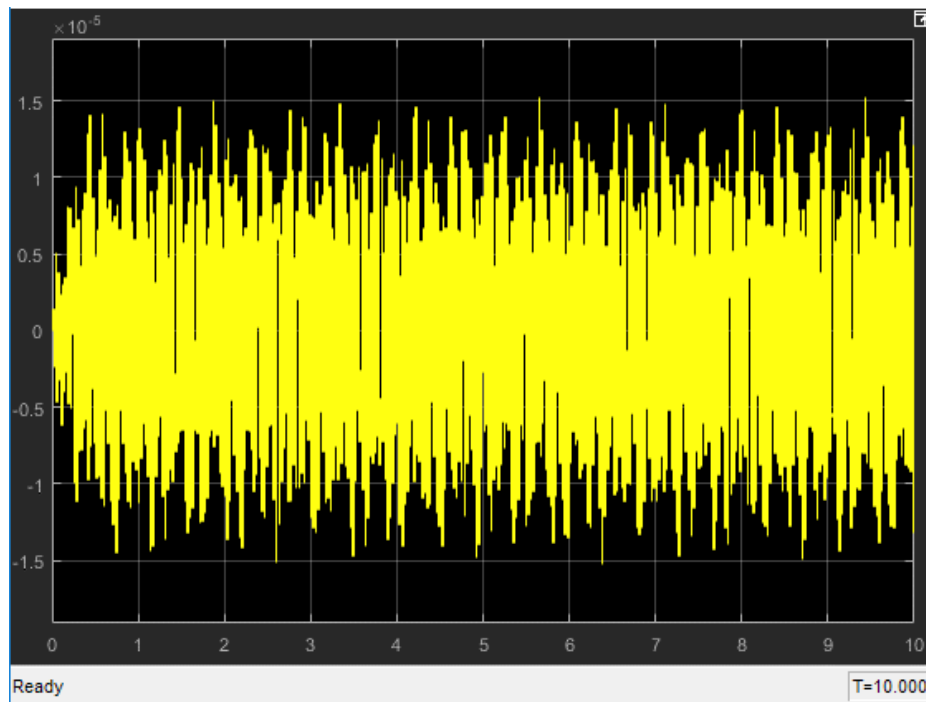


Figure 5.27: Case 1 - The responses of residual R_3 for actuator fault

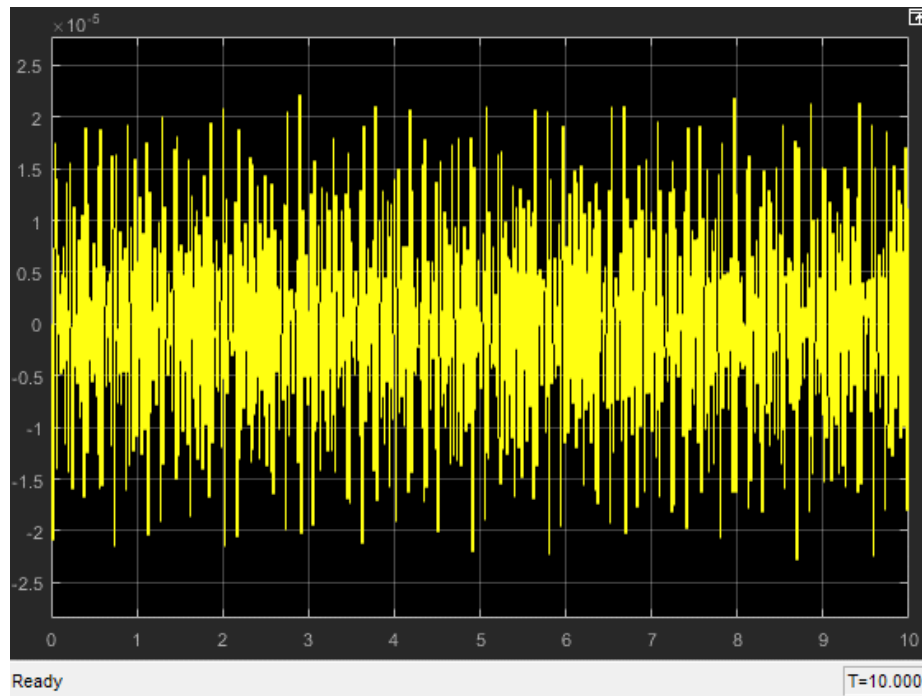


Figure 5.28: Case 1 - The responses of residual R_d for actuator fault

It is clearly seen that all of four residual signals are close to zero and their values are less than 1×10^{-4} , which prove that there is no fault happened during the simulation. Therefore, the designed fault detection process is available for vibration cancellation system.

5.4.2 Simulation Results of Actuator Fault for Case 2: Actuator 1 Fault

Similar to Section 5.2.2, a pre-defined fault signal is added into actuator 1 and the simulation results are analysed in this section. Figure 5.26, Figure 5.27, Figure 5.28 and Figure 5.29 display the curves of the four designed residuals respectively when a fault occurs in actuator 1.

It has known that the fault occurred at 2-second time. The simulation results show that before the fault happened, all of four residuals are close to zero. Nevertheless, when actuator 1 malfunctions, all the residuals have a sharp rise and then gradually drop to zero again within 1 second. Note that this phenomenon is different from the residual responses in Section 5.2.2 that maintain at a stable value. This is because that the fault signals in sensors are directly added onto the system output signals without other calculation, but the fault signals in actuators are considered as unknown input signals and are transformed through the system model. However, through comparing the magnitude of these residuals, it can be found that the magnitude of residual R_2 and residual R_3 are still much larger than that of residual R_1 when the fault is just introduced, but residual R_2 does not have a distinct difference. Therefore, the actuator fault isolation is still achieved but not a strong isolation at this stage. In addition, residual R_d is almost zero at the beginning, but there is a dramatic rise after the fault is injected. So the goal of fault detection is fulfilled.

Chapter 5: Simulation

Consequently, the simulated results illustrate that a fault happened in the system after running for 2 seconds and the location of the fault is most possibly in actuator 1. Hence, the designed residuals react at the injected fault, and the fault detection and isolation for actuator 1 is able to be implemented .

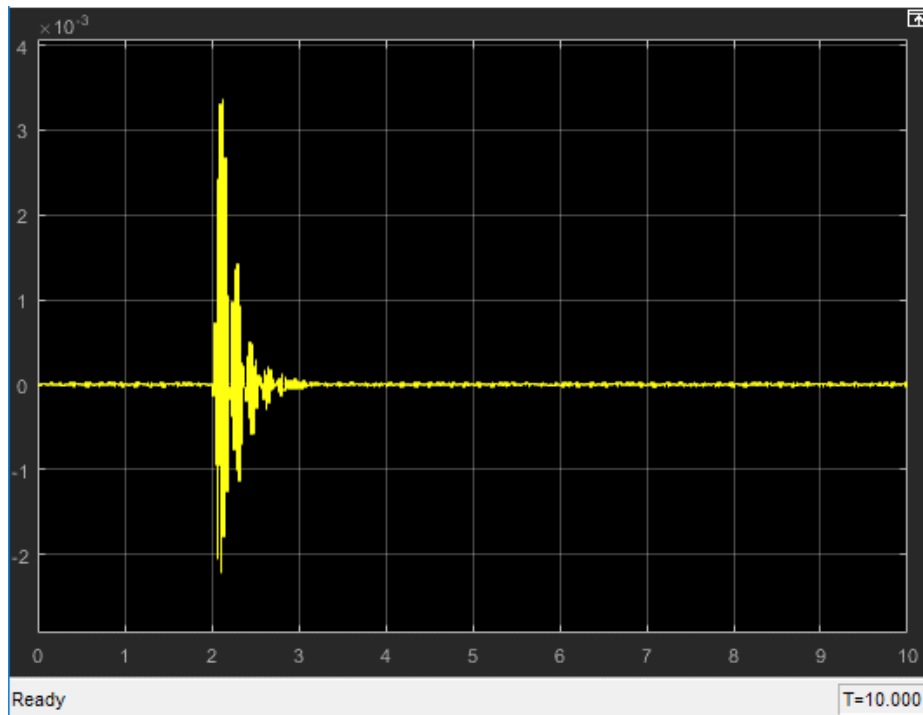


Figure 5.29: Case 2 - The responses of residual R_1 for actuator fault

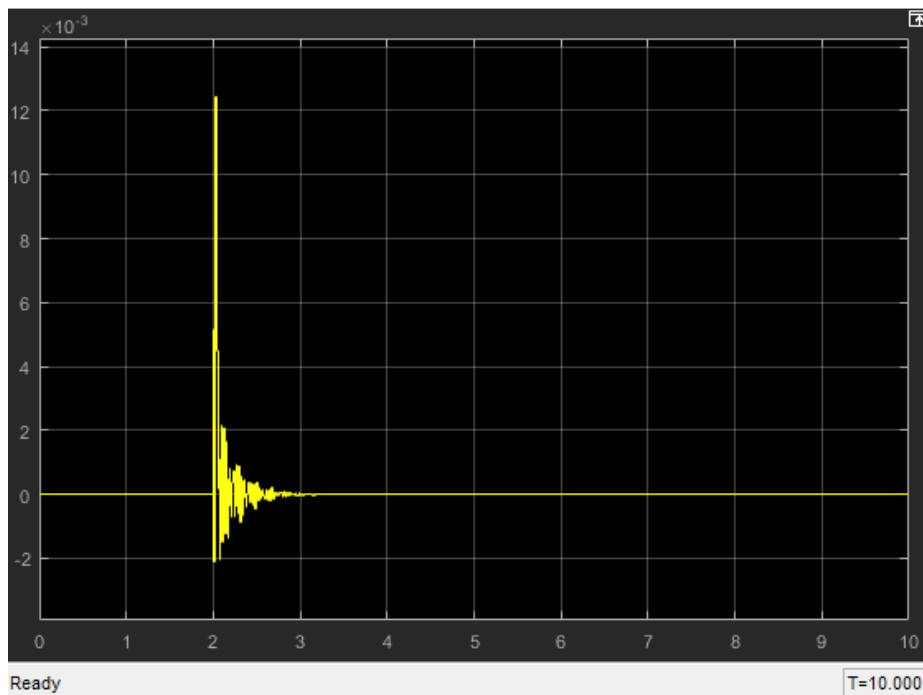


Figure 5.30: Case 2 - The responses of residual R_2 for actuator fault

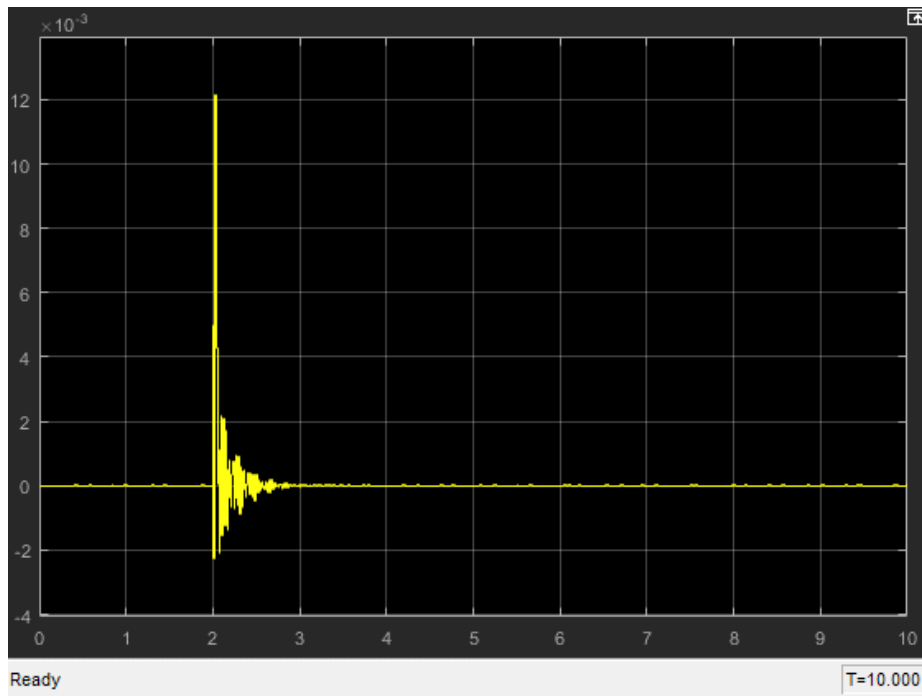


Figure 5.31: Case 2 - The responses of residual R_3 for actuator fault

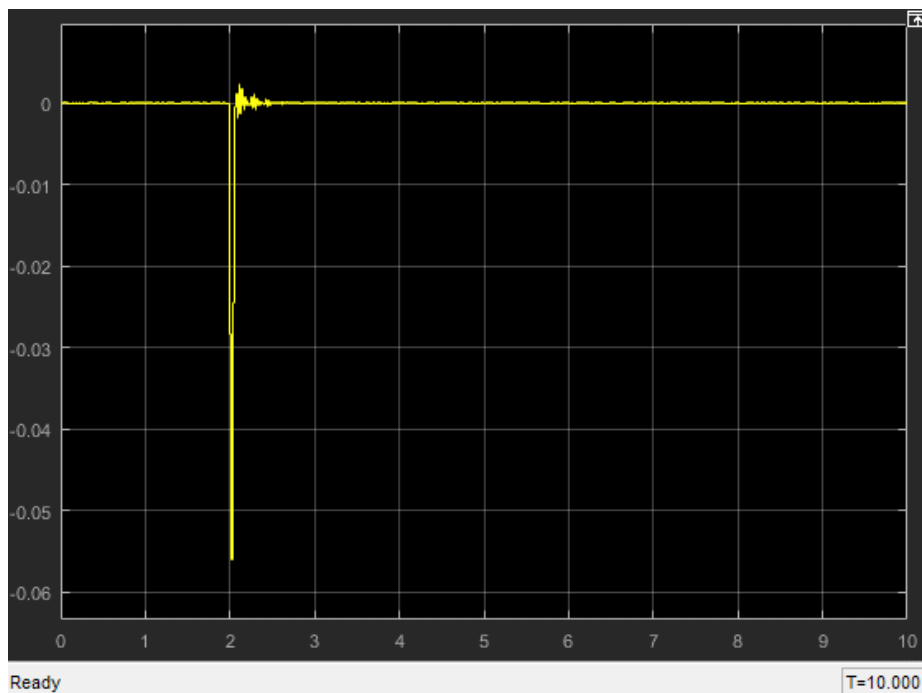


Figure 5.32: Case 2 - The responses of residual R_d for actuator fault

5.4.3 Simulation Results of Actuator Fault for Case 3: Actuator 2 Fault

The task of this section is to detect the injected fault and isolate it. Like to last Section 5.4.2, a predefined fault is added onto actuator 2 separately. The residual responses are shown in Figure 5.30, Figure 5.31, Figure 5.32 and Figure 5.33 respectively.

Chapter 5: Simulation

As shown in these figures, the responses of residual R_d indicates that a fault appears in the system and the magnitude of residual R_1 is also larger than that in Section 5.4.2 when a fault is introduced. Nevertheless, the responses of residual R_2 and R_3 have extremely shapes, so it is impossible to identify that the fault is in actuator 2 or actuator 3. Furthermore, the simulation results for Case 4 are almost same with Case 3, so that the plots of residuals are omitted.

To conclude, the isolation of actuator 2 fault and actuator 3 fail at this stage because the designed residual R_2 and R_3 have similar specifications. A possible reason is that the input matrix B contains two extremely similar columns corresponding to input signal 2 and input signal 3 and the effects on the output signals from input signal 2 and signal 3 are almost same.

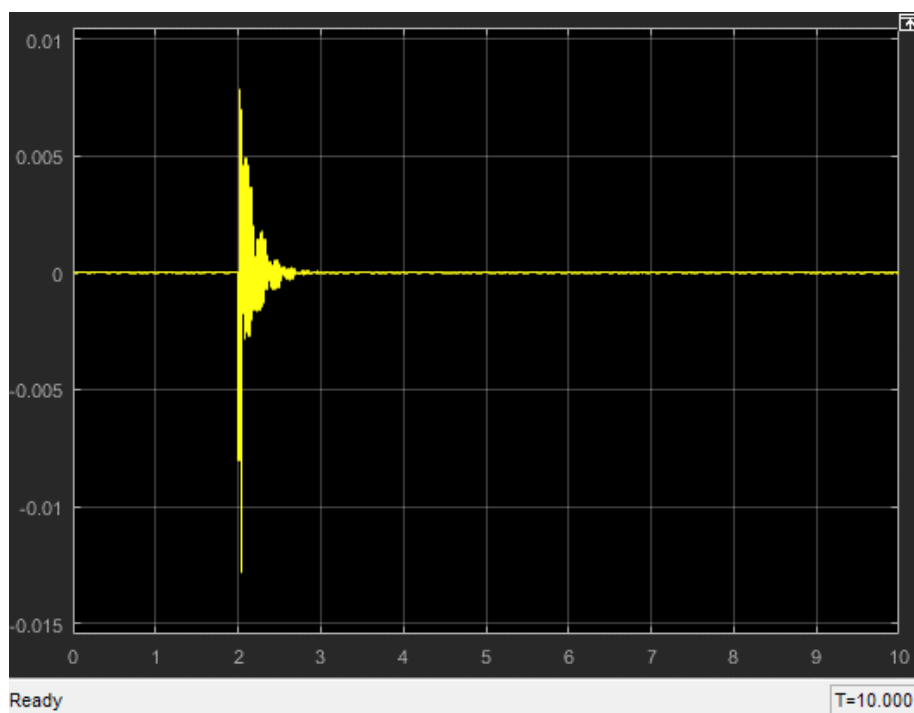


Figure 5.33: Case 3 - The responses of residual R_1 for actuator fault

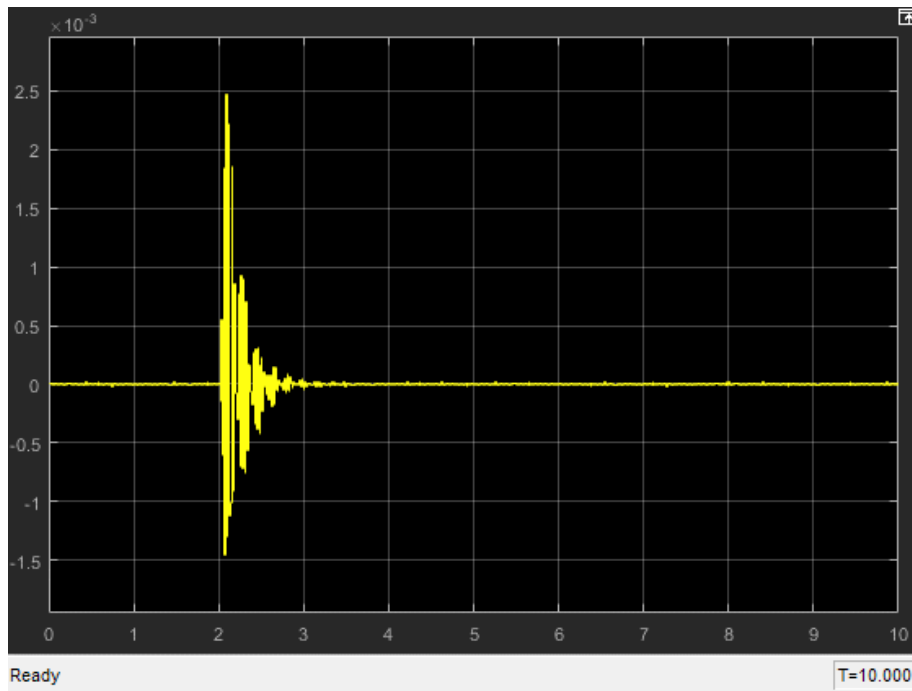


Figure 5.34: Case 3 - The responses of residual R_2 for actuator fault

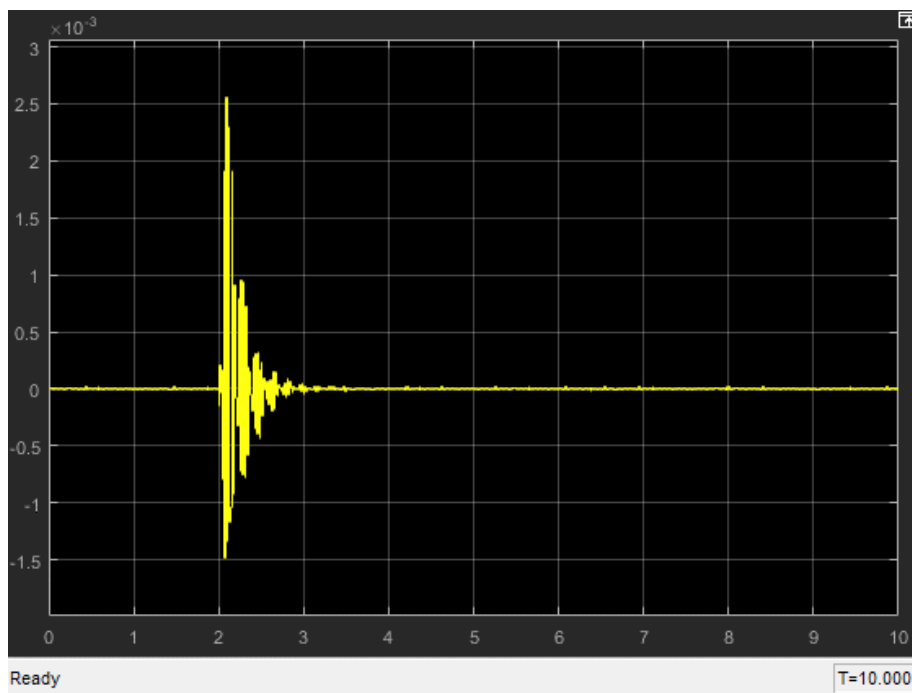


Figure 5.35: Case 3 - The responses of residual R_3 for actuator fault

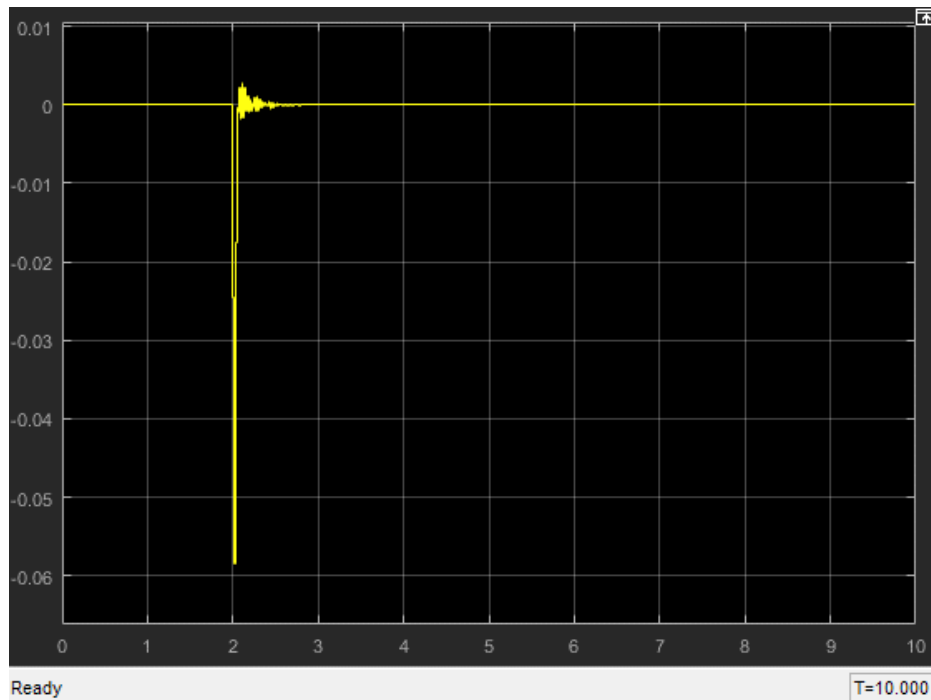


Figure 5.36: Case 3 - The responses of residual R_d for actuator fault

5.5 Simulation Summary

The FDD simulation of the laboratory structure is implemented in this chapter. According to simulation results in Section 5.2 and 5.4,

- For sensor fault, the FDD processes are completely successful. Wherever a fault occurs, the FDD processes can find the fault and identify its location quickly and reliably.
- For actuator fault, the goal of fault detection can be achieved properly, however, only the actuator 1 fault can be isolated and due to the similarity of columns in input matrix B, the fault in actuator 2 and actuator 3 could not be identified clearly.

Therefore, it is advised that further research could focus on how to distinguish the residuals designed for isolating actuator 2 and actuator 3, such as modelling the system in a different way or finding a suitable parity vector to identify the above residuals.

Chapter 6 : Conclusions and Future Works

6.1 Project Conclusion

The main contribution of this project is to implement FDD processes via using parity relation technique on a plate-structured vibration cancellation system. This project first addresses the principle of parity relation technique and then illustrates a procedure of fault detection and isolation process through using parity relation technique. After that, the designed FDD processes in terms of sensor faults are validated in SISO and MIMO systems. Finally, the FDD processes, including both sensor fault and actuator faults, are tested in MATLAB Simulink based on the real laboratory mathematical model. In term of sensor fault, the designed FDD processes work successfully. For actuator fault, fault detection process can be implemented properly, but fault isolation process cannot identify the location of an actuator fault in Actuator 2 or Actuator 3 due to the symmetric positions of two actuators.

The brief review of this project is draw as follows:

Initially, the laboratory model setup and hardware components are proposed. The laboratory structure is comprised of a base plate, a top plate supported by three transducers, one transducer providing external disturbance and three accelerometers placed on the top plate at the same location with transducers. The control goal is to maintain the top plate vibration-free while a disturbance signal is introduced to the base plate consistently and forces the top plate vibrating. So far, the laboratory model is clearly stated.

Furthermore, a physical experiment method is used to obtain the mathematical model by using model analysis software 'ModalVIEW'. Especially, the input-output transfer function form is manually converted into state-space representation upon the theoretical analysis method and experiment data, which is the preparation of FDD design.

After that, the theory of parity relation technique is illustrated. FDD is a significant aspect of fault tolerant control architectures. The core part of parity relation method is to build a constant equation through designing a parity vector which can eliminate the effects of unknown system variables. When the system is fault free, the designed residual of the system is stable and close to zero in value. However, when a fault is introduced into the system, the residual will alter to indicate there exists error(s) in the system. To detect fault, firstly need to define an order of the parity space, in which the designed parity vector V_s satisfies certain condition: $V_s * H = 0$, where H is a deterministic matrix associated with system variable $X(t)$. The designed residual is calculated based on the measurable system input and output, and is not affected by system variables $X(t)$.

Chapter 6: Conclusion

Additionally, fault isolation process is implemented through designing different parity vectors V_s and forming a set of structured residuals. Each residual is sensitive to all, but only insensitive to one fault. The fault will be located through a fault code table.

Based on the theoretical design, parity space technique for SISO system case and MIMO system case are designed and validated through MATLAB Simulink. The faulty system is created by injecting pre-defined single fault into the normal system. Simulation results of both SISO and MIMO systems show that the designed residual signals corresponding to a fault change from zero to non-zero when the fault is injected, which validates the FDD design using parity relation technique. After then, the parity relation method is extended to the real plate model in MATLAB. Simulation result shows that the parity relation technique is able to detect pre-defined faults for the real plate system. Additionally, fault isolation process can locate a fault through a fault code table. Simulation results show that parity relation method has an excellent performance in detecting single sensor fault of the studied vibration cancellation system.

6.2 Future Works

Although FDD processes for sensor fault have been proven and fulfilled on the laboratory model, some improvements should be made in future. It has been mentioned in Chapter 5 that the fault isolation for actuator fault failed in the laboratory model simulation. The reason may be complex and multiple, such as the mathematical model that is not enough accurate or symmetric positions of actuator 2 and actuator 3. The future research should focus on addressing fault isolation process for actuator fault.

Moreover, the experiments of the designed FDD processes are not completed due to time limitation. As an experimental validation may be affected easily by unknown noise for vibration cancellation system, the designed parity vector is required to be robust to noise and sensitive to a fault. The author mainly uses a directly mathematical method to generate the parity vector, which is easy to achieve and fast responses to dynamic systems. However, the performance is not as good as desired in practice. There are still a few mathematical methods to improve the performance of FDD processes using parity relation technique. [6] states that how to use singular value decomposition (SVD) to find a sub-optimal parity vector that has a strong robustness to noise.

Consequently, for future work, fault isolation for actuator fault could be researched and experiment methods should be addressed and tested on the real plate structure with an evolved parity vector.

Appendix A: MATLAB Code for SISO System Validation

The continuous system parameters used for SISO system validation are shown as follow:

$$\mathbf{A} = \begin{bmatrix} -1.6688 & 4.125 \\ -0.2926 & -3.8177 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 6 \\ 2.3077 \end{bmatrix}$$

$$\mathbf{C} = [0 \quad 1]$$

$$\mathbf{D} = [0]$$

The sampling time $T = 0.01s$ and the parity order $s = 4$.

The parity vector $V_s = [0.06 \quad -0.5495 \quad 0.8030 \quad -0.1699 \quad -0.144]$

The MATLAB code is kept within the advanced control research group, Flinders University.

The Code can be provided upon request.

Appendix B: MATLAB Code for MIMO System Validation

The continuous system parameters used for MIMO system validation are shown as follow:

$$\mathbf{A} = \begin{bmatrix} -1.6688 & 4.125 \\ -0.2926 & -3.8177 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 410.3077 \\ 55.6064 & 0 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The sampling time $T = 0.01s$ and the parity order $s = 4$.

The parity vector

$$V_{s1} = [0 \quad 0.0742 \quad 0 \quad -0.547 \quad 0 \quad 0.8006 \quad 0 \quad -0.175 \quad 0 \quad -0.1541]$$

$$V_{s2} = [-0.2692 \quad 0 \quad 0.7022 \quad 0 \quad -0.2831 \quad 0 \quad -0.4889 \quad 0 \quad 0.3395 \quad 0]$$

The MATLAB code is kept within the advanced control research group, Flinders University.

The Code can be provided upon request.

Appendix C: MATLAB Code for Residual Generation

The MATLAB code is kept within the advanced control research group, Flinders University.

The Code can be provided upon request.

Appendix D: MATLAB Code for Fault Isolation Block

The MATLAB code is kept within the advanced control research group, Flinders University.

The Code can be provided upon request.

Bibliography

- [1] M. Mahmoud, J. Jiang, and Y. Zhang, *Active fault tolerant control systems: stochastic analysis and synthesis*. Springer Science & Business Media, 2003.
- [2] K. Y. Bülak and R. H. Scanlan, "Resonance, Tacoma Narrows bridge failure, and undergraduate physics textbooks," *Am. J. Phys.*, vol. 59, p. 2, 1991.
- [3] S. S. Rao and F. F. Yap, *Mechanical vibrations*. Prentice Hall Upper Saddle River, 2011.
- [4] A. Preumont, *Vibration control of active structures*. Springer, 1997.
- [5] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, and J. Schröder, *Diagnosis and fault-tolerant control*. Springer, 2006.
- [6] J. Chen and R. J. Patton, *Robust model-based fault diagnosis for dynamic systems*. Springer Science & Business Media, vol.3, pp. 38-44,2012.
- [7] S. X. Ding, *Model-based fault diagnosis techniques: design schemes, algorithms, and tools*. Springer Science & Business Media, pp.3-11, 2008.
- [8] A. S. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, no. 6, pp. 601-611, 1976.
- [9] G. Meyer and L. S. Cicolani, "A formal structure for advanced automatic flight-control systems," 1975.
- [10] R. Isermann and P. Balle, "Trends in the application of model-based fault detection and diagnosis of technical processes," *Control engineering practice*, vol. 5, no. 5, pp. 709-719, 1997.
- [11] J. J. Gertler, "Survey of model-based failure detection and isolation in complex plants," *IEEE Control systems magazine*, vol. 8, no. 6, pp. 3-11, 1988.
- [12] A. Ray and M. Desai, "A Redundancy Management Procedure for Fault Detection and Isolation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 108, no. 3, pp. 248-254, 1986.
- [13] 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," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3757-3767, 2015.
- [14] R. Dorr, F. Kratz, J. Ragot, F. Loisy, and J.-L. Germain, "Detection, isolation, and identification of sensor faults in nuclear power plants," *IEEE Transactions on Control Systems Technology*, vol. 5, no. 1, pp. 42-60, 1997.
- [15] W. C. Merrill, J. C. DeLaat, and W. M. Bruton, "Advanced detection, isolation, and accommodation of sensor failures-Real-time evaluation," *Journal of Guidance, Control, and Dynamics*, vol. 11, no. 6, pp. 517-526, 1988.
- [16] T. Kobayashi and D. L. Simon, "Application of a bank of Kalman filters for aircraft engine fault diagnostics," in *ASME Turbo Expo 2003, collocated with the 2003 International Joint Power Generation Conference*, 2003, pp. 461-470: American Society of Mechanical Engineers.
- [17] T. Kobayashi and D. L. Simon, "Hybrid Kalman filter approach for aircraft engine in-flight diagnostics: Sensor fault detection case," *Journal of engineering for gas turbines and power*, vol. 129, no. 3, pp. 746-754, 2007.
- [18] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, 2000, pp. 153-158: IEEE.
- [19] R. V. Beard, "Failure accommodation in linear systems through self-reorganization," Massachusetts Institute of Technology, 1971.
- [20] H. L. Jones, "Failure detection in linear systems," Massachusetts Institute of Technology, 1973.
- [21] M.-A. Massoumnia, "A geometric approach to the synthesis of failure detection filters," *IEEE Transactions on automatic control*, vol. 31, no. 9, pp. 839-846, 1986.

Bibliography

- [22] J. White and J. Speyer, "Detection filter design: Spectral theory and algorithms," *IEEE Transactions on Automatic Control*, vol. 32, no. 7, pp. 593-603, 1987.
- [23] P. S. Min, "Robust application of Beard-Jones detection filter," in *Advances in Computing and Control*: Springer, 1989, pp. 162-173.
- [24] Y. Xiong and M. Saif, "Robust fault isolation observer design," in *American Control Conference, 1999. Proceedings of the 1999*, 1999, vol. 3, pp. 2077-2081: IEEE.
- [25] Y. Kim and J. Park, "Noise response of detection filters: relation between detection space and completion space," *IEE Proceedings-Control Theory and Applications*, vol. 150, no. 4, pp. 443-447, 2003.
- [26] J. Gertler, "Fault detection and isolation using parity relations," *Control engineering practice*, vol. 5, no. 5, pp. 653-661, 1997.
- [27] J. Gertler and D. Singer, "A new structural framework for parity equation-based failure detection and isolation," *Automatica*, vol. 26, no. 2, pp. 381-388, 1990.
- [28] E. Chow and A. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Transactions on automatic control*, vol. 29, no. 7, pp. 603-614, 1984.
- [29] M. Staroswiecki, V. Cocquempot, and J. P. Cassar, "Optimal design of FDI systems via parity space and observer based approaches," in *Industrial Electronics, Control and Instrumentation, 1991. Proceedings. IECON'91., 1991 International Conference on*, 1991, pp. 143-148: IEEE.
- [30] X. Ding, L. Guo, and T. Jeansch, "A characterization of parity space and its application to robust fault detection," *IEEE Transactions on Automatic Control*, vol. 44, no. 2, pp. 337-343, 1999.
- [31] S. Varrier, D. Koenig, and J. J. Martinez, "A parity space-based fault detection on lpv systems: Approach for vehicle lateral dynamics control system," *IFAC Proceedings Volumes*, vol. 45, no. 20, pp. 1191-1196, 2012.
- [32] M. Zhong, Y. Song, and S. X. Ding, "Parity space-based fault detection for linear discrete time-varying systems with unknown input," *Automatica*, vol. 59, pp. 120-126, 2015.
- [33] R. D. Blevins and R. Plunkett, "Formulas for natural frequency and mode shape," *Journal of Applied Mechanics*, vol. 47, p. 461, 1980.
- [34] P. Zhang and F. He, "EXPERIMENTAL IMPLEMENTATION OF MULTIPLE-INPUT MULTIPLE-OUTPUT ACTIVE VIBRATION CONTROL ON VI-BRATION ISOLATION SYSTEMS," 24th INTERNATIONAL CONGRESS ON SOUND AND VIBRATION, London, 2017.
- [35] Q. Chen, "Vibration Cancellation in Plate Structures," Master Thesis, Flinders University, 2016.
- [36] L. Ljung, "System identification," Springer, 1998, pp. 163-173.
- [37] S. Groothuis and R. Roesthuis, "Vibration Cancellation Using Synthetic Shunt Impedances," Internship Thesis, Flinders University, 2010.
- [38] H. A. A. Ab and A. S. Ari, "Data-Driven Design of Fault Diagnosis Systems," Springer, pp.113-12, 2014.
- [39] J. Gertler, *Fault Detection and Diagnosis*. Springer, pp.417-422, 2015.
- [40] J. Gertler, "Analytical redundancy methods in fault detection and isolation-survey and synthesis," *IFAC Proceedings Volumes*, vol. 24, no. 6, pp. 9-21, 1991.
- [41] J. J. Gertler and R. Monajemy, "Generating directional residuals with dynamic parity relations," *Automatica*, vol. 31, no. 4, pp. 627-635, 1995.