



Kalman Filter Based Fault Detection and Diagnosis

by

Wei Cui, M.Eng. (Electronics)

Supervisor: Fangpo He

Date of submission: October, 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 work 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 it does not contain any material previously published or written by another person except where due reference is made in the text.

Wei Cui

October, 2018

崔巍
11/12/2018

Acknowledgement

I would like to express my very great appreciation to Professor Fangpo He, my research supervisor, for her patient direction, warm encouragement and useful critiques of my thesis work. I would also like to thank ph.D. student Peng Zhang, for his advice and assistance in my research problems. My grateful thanks are also extended to Dr Andrew Lamma for his guidance in real time sampling data.

Also, I want to say thank you to Miss Abby Zhou for her selfless support and encouragement during the thesis project.

Finally, I would like to acknowledge the support provided by my family throughout my study in Australia.

Abstract

In many control systems, sensors and actuators are integral parts, which can be utilized to detect the change of the system and take the needful action to achieve the control purpose. However, sensors and actuators have their own service life and can be damaged by various factors, which results in financial loss and casualties. These phenomena, when sensors and actuators stop working, are called sensors and actuators faults.

In this thesis, a Kalman Filter based Fault Detection and Diagnosis (FDD) scheme is proposed to detect and isolate different actuator faults for a given three-input three-output plate structure resonant system. This system can be used in many areas, such as doing operation on an emergency vehicle. The mathematical model of the given resonant system is first modelled and obtained by using open loop transfer function method and then changed into a discrete-time State Space Representation (SSR) model for further design purpose.

By setting the initial estimated state and its corresponding estimation error covariance, the following estimated state can be calculated by using the real-time control signals and measured outputs using the technique of Kalman filter. To utilize the Kalman filter technique for FDD purpose in a resonant system, the estimated outputs can be calculated using the estimated state. Next, a set of corresponding output error residuals can be generated by comparing the difference between the estimated outputs and measured outputs. A normalization algorithm calculating the RMS value of the corresponding residuals is applied to determine a threshold value to identify the location of actuator faults occurred in the system.

The concept of Kalman filter technique is first tested in an artificial system. Simulation results indicate that the estimated states generated by the Kalman filter can quickly approach the actual state values and track the actual state values all the time. The given 3*3 plate structure system with one fault is tested via simulation in MATLAB SIMULINK, which validates the proposed Kalman filter based FDD design. In the FDD design, four identical Kalman filters are constructed to form a Kalman filter bank for no fault, as well as actuator 1, 2 and 3 fault cases respectively. The estimated output of each case is computed by using the Kalman filter bank estimated state. The residuals, which are produced using estimated outputs and measured outputs, are analysed for the single fault detection and diagnosis. Then multiple actuator fault cases are introduced to the plate structure system and tested via MATLAB SIMULINK. The

simulation results show that the corresponding multiple faults are detected successfully by analysing the output residuals using the proposed normalization algorithm.

After the proposed Kalman filter based FDD scheme is validated in simulation, it is then tested in a real-time experiment. Two discrete-time system models, a Kalman filter bank and a normalization algorithm are built to construct the experimental Kalman filter FDD scheme. One of the discrete-time system models is to produce the discrete-time measured outputs without fault, which is used for fault detection. The other is introduced to compute the measured outputs (possible with fault), which is used for fault diagnosis. The residuals computed in the Kalman filter bank for both single actuator fault cases and multiple actuator faults cases are generated. A set of residual data is recorded, and its RMS value is compared with the set threshold constant. The result verifies that all the actuator fault cases are isolated successfully.

List of Figures

Figure 2.1 Methods for Fault Detection in Control Systems [6]	4
Figure 2.2 A general structure of FDD.....	5
Figure 2.3 Illustrations of the concepts of hardware redundancy and analytical redundancy for FDD.....	6
Figure 2.4 Analytical redundancy.....	8
Figure 3.1 The MIMO mechanical structure	14
Figure 3.2 The MIMO close-loop system block diagram.....	14
Figure 3.3 Plate layout for open-loop system	16
Figure 3.4 MIMO control system block diagram.....	16
Figure 3.5 The arrangement of the physical system.....	18
Figure 3.6 Measured FRF curves for the open-loop real system.....	19
Figure 3.7 The block diagram of the plate structure system	25
Figure 4.1 The system model block diagram with noise	33
Figure 4.2 The discrete-time Kalman filter algorithm block diagram	42
Figure 4.3 The Kalman filter application system.....	43
Figure 4.4 The block diagram of the Kalman filter application system.....	44
Figure 4.5 The MATLAB Simulink for the Kalman filter application system.....	44
Figure 4.6 The compared result between the real system state and the estimated state	45
Figure 4.7 The system actuators position relationships	46
Figure 4.8 The classification of actuator faults	46
Figure 4.9 The simplified Kalman filter algorithm	47
Figure 4.10 The fault detection method block diagram.....	47
Figure 4.11 The fault diagnosis of the actuator fault 1 method block diagram	50
Figure 4.12 The fault diagnosis of actuator fault 2 method block diagram.....	52
Figure 4.13 The fault diagnosis of actuator fault 3 method block diagram.....	54
Figure 4.14 The Kalman filter bank to diagnose single fault.....	55
Figure 4.15 The block diagram of the Kalman filter bank for FDD	59
Figure 4.16 The fault detection simply using the threshold value	60
Figure 4.17 Non-zero initial condition example of the residual.....	61
Figure 4.18 The normalization algorithm block diagram for analysing the fault detection residual	62

Figure 4.19 The normalization algorithm block diagram for analysing the FDD residuals	63
Figure 4.20 The Kalman filter based FDD block diagram.....	64
Figure 5.1 The artificial system to validate the Kalman filter based FDD method.....	67
Figure 5.2 The Kalman filter bank simulation to derive four residuals	67
Figure 5.3 The structure of the system model.....	68
Figure 5.4 The structure of the Kalman filter bank	68
Figure 5.5 The simulation structure of the Kalman filter based FDD method	70
Figure 5.6 The structure of the normalization algorithm	71
Figure 5.7 The fault detection RMS value when system do not have fault	72
Figure 5.8 The fault detection RMS value when the system has fault.....	72
Figure 5.9 The four residuals and results without faults in simulation	73
Figure 5.10 The four residuals and results with actuator fault 1 in simulation.....	74
Figure 5.11 The four residuals and results with actuator fault 2 in simulation.....	74
Figure 5.12 The four residuals and results with actuator fault 3 in simulation.....	75
Figure 5.13 The four residuals and results with actuator fault 1 and 2 in simulation.....	76
Figure 5.14 The four residuals and results with actuator fault 2 and 3 in simulation.....	77
Figure 5.15 The four residuals and results with actuator fault 1 and 3 in simulation.....	78
Figure 5.16 The structure for the experiment	79
Figure 5.17 The four residuals and results without faults in experiment	80
Figure 5.18 The four residuals and results with actuator fault 1 in simulation.....	81
Figure 5.19 The four residuals and results with actuator fault 2 in simulation.....	82
Figure 5.20 The four residuals and results with actuator fault 3 in simulation.....	83
Figure 5.21 The four residuals and results with actuator fault 1 and 2 in simulation.....	84
Figure 5.22 The four residuals and results with actuator fault 2 and 3 in simulation	85
Figure 5.23 The four residuals and results with actuator fault 1 and 3 in simulation.....	86

Contents

Declaration.....	ii
Acknowledgement.....	iii
Abstract.....	iv
List of Figures.....	vi
Chapter 1 : Introduction.....	1
1.1 Thesis background.....	1
1.2 Research purpose and significance.....	1
1.3 The outline of the thesis.....	1
Chapter 2 : Literature Review.....	3
2.1 FDD development.....	3
2.2 Faults classification.....	5
2.2.1 Sensor faults.....	5
2.2.2 Actuator faults.....	6
2.2.3 System component faults.....	6
2.3 FDD methods.....	6
2.3.1 Hardware redundancy.....	7
2.3.2 Analytical redundancy.....	8
Chapter 3 : System Identification of Plate Structure.....	13
3.1 Plate structure system components.....	13
3.2 Theoretical open-loop transfer function for the plate structure.....	15
3.2.1 The general transfer function for the open-loop system.....	15
3.2.2 The plate structure transfer function for the open-loop system.....	15
3.3 The theoretical analysis of the system transfer function.....	17
3.4 Experiment of transfer function modelling.....	17
3.4.1 Open-loop transfer function modelling method.....	18

3.4.2 The experiment result parameters	18
3.5 Construction of continuous-time SSR.....	21
3.6 Construction of discrete-time SSR	27
Chapter 4 : Kalman Filter Based FDD Method	31
4.1 The Discrete-time Kalman Filter algorithm	31
4.2 The discrete-time Kalman filter conclusion	40
4.3 An application of the Kalman filter	43
4.4 The FDD method by using the Kalman filter	46
4.4.1 The classifications of system actuator faults	46
4.4.2 Fault detection method	47
4.4.3 Fault diagnosis method.....	48
4.4.4 The normalization algorithm.....	60
Chapter 5 : Kalman Filter Based FDD Method Validation.....	66
5.1 Conceptual Kalman filter based FDD method validation.	66
5.1.1 No fault system simulation result	73
5.1.2 Actuator fault 1 system simulation result.....	73
5.1.3 Actuator fault 2 system simulation result.....	74
5.1.4 Actuator fault 3 system simulation result.....	75
5.1.5 Actuator fault 1 and 2 system simulation result	75
5.1.6 Actuator fault 2 and 3 system simulation result	76
5.1.7 Actuator fault 1 and 3 system simulation result	77
5.2 The Kalman filter based FDD method via experiment.....	78
5.2.1 No fault system experiment result	79
5.2.2 Actuator fault 1 system simulation result.....	80
5.2.3 Actuator fault 2 system simulation result.....	81
5.2.4 Actuator fault 3 system simulation result.....	82
5.2.5 Actuator fault 1 and 2 system simulation result	83

5.2.6 Actuator fault 2 and 3 system simulation result	84
5.2.7 Actuator fault 1 and 3 system simulation result	85
Chapter 6 : Conclusion	87
Appendix A: MATLAB Code	88
Reference.....	89

Chapter 1 : Introduction

1.1 Thesis background

The thesis background is the Multi-Input Multi-Output (MIMO) vibration control system and equipment, in which the in-depth system research work is carried out. For the fault FDD of the vibration control system, a new solution is proposed by applying the Kalman filter.

In many vibration control systems, once a fault occurs, it is likely to cause catastrophic damage in the system due to a resonance effect. In order to improve the safety and reliability of the system, it is necessary to monitor the fault and determine the fault location in real time during the operation of the vibration system. This will allow for a certain period of time for hardware replacement and emergency maintenance. Therefore, FDD technology is especially important in vibration control.

Although FDD technology has achieved relatively fruitful research results through the decades of its development, the technology of FDD is still not mature. First of all, many research objects are nonlinear systems with diversity and complexity. Secondly, the system has some external noise and unknown input signals in addition to the known control inputs; these have made FDD research more difficult.

1.2 Research purpose and significance

Part of the vibration system state cannot be measured directly for a number of reasons. At this time, the state of the system can be predicted by a state estimator. The core task of a state estimate is to design filters. Among linear systems with Gaussian noise, the Kalman filter is undoubtedly the optimal estimator when compared to other filters. Upon knowing the estimated state of the system, it can be converted into an estimated measurement by the output equation of the vibration system. Comparing the estimated measurement with the actual measurement, it is possible to know whether the vibration system is in a normal state. The Kalman filter-based FDD can effectively detect and locate system faults in real time, which greatly helps the practical application of many vibration systems.

1.3 The outline of the thesis

In this project, a Kalman filter based FDD method is designed and demonstrated for a MIMO mechanical plate structure with three pairs of sensors and actuators.

Chapter 1: Introduction

In Chapter 2, an overall literature review is presented. FDD development and fault classifications are introduced in detail as well as different FDD approaches.

In Chapter 3, the features and characteristics of the MIMO mechanical plate structure, of this project, is introduced. Based on the theoretical analysis and physical experiments, the plant structure system is modelled in a discrete-time SSR.

In Chapter 4, the Kalman filter based FDD method is discussed as the core technology of the FDD approach adopted in this thesis. The structure of the discrete-time Kalman filter which forms the basic concept in the fault detection approach is introduced firstly. Then a simple example for understanding Kalman filter principle is discussed using MATLAB SIMULINK. Lastly, the method to apply the Kalman filter in FDD is introduced.

In Chapter 5, The Kalman filter based FDD method is verified in both simulation and experiment. The parameter values and simulation approach is discussed in MATLAB SIMULINK. The comparison and analysis is demonstrated through experiment by using dSPACE software.

In Chapter 6, meaningful conclusions are drawn based on the achievement of the thesis and future works are suggested to develop the Kalman filter based FDD method.

Chapter 2 : Literature Review

In this chapter, an overall literature related to FDD is reviewed. FDD development is firstly discussed to explain the FDD importance and its history. Next the fault classifications are introduced to illustrate the research object in this thesis. In the end, different FDD approaches are discussed to obtain the best choice for this project.

2.1 FDD development

Modern control systems have been introduced into different complex systems, such as robotic systems, radar systems, automotive systems, and so on, resulting with the control algorithms becoming more and more sophisticated. These control systems are based on different kinds of electronic components, which may cause a fault with the control systems. In high-level intelligent and automated systems, the fault process can be extremely rapid. For some safety-critical systems, the fault consequences can result in loss of life, economic loss and environmental damage. [1] To avoid the irreversible tragedy and improve the safety and reliability of the system, the technology of FDD is used to detect the reason, position and damage degree during the system processing.

Since about the 1970s, FDD has gained a rising worldwide consideration both in theory and application. Although FDD technique has developed over 40 years, the growing demand for safety, reliability, maintainability, and survivability in control systems has drawn abundant research achievements. [2] In 1996, Professor Frank (Duisburg University, German) assorted FDD into three basic categories, which are the knowledge and knowledge-based methods, analytical model-based methods and signal based methods.[3] Furthermore, each group includes more detailed plans, as shown in Figure 2.1. In 2003, Professor Venkata Subramanian (Purdue University, USA) classified FDD methods as quantitative model-based methods, qualitative model-based methods and process history.[4] As the development of artificial intelligence (AI) has increased at an exponential rate, the adoption of AI technology in monitoring, control, and diagnosis for FDD systems has also increased.[5]

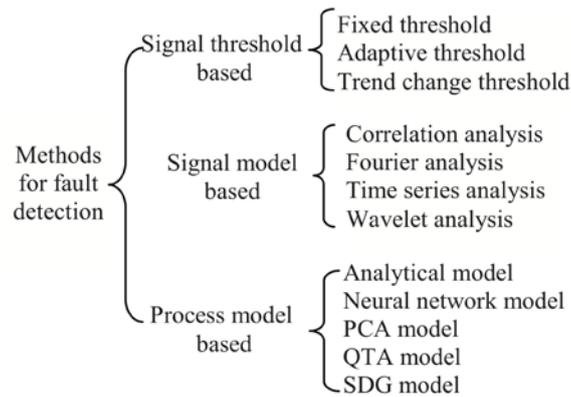


Figure 2.1 Methods for Fault Detection in Control Systems [6]

In general, the following functions should be distinguished in FDD systems: [7]

- (a) Monitoring: for tolerance purpose, the FDD system should check measurable variables in real time, and alarms should be generated all the time.
- (b) Automatic protection: the monitoring function automatically activates an appropriate countermeasure regarding a dangerous process state.
- (c) Supervision with fault diagnosis: based on measured variables, the FDD system can generate the changing feature and symptoms, detect the possible fault, and make the reasonable decision for counteractions.

The classic FDD systems based on supervision methods (a) and (b) have a tremendous advantage in simplicity and reliability. However, these two methods require a mass of information to generate changes then react to the fault. Besides, in some specific fields, FDD system with method (a) or (b) is not feasible. Therefore FDD system with method (c) is needed, especially for satisfying the following requirements.

- (1) Timely detection of small faults in germination or in an unexpected stage;
- (2) FDD in sensors, actuators, and process components;
- (3) FDD in closed-loop systems;
- (4) Process supervision in transient states.

Although FDD technology is one of the most famous research topics, it is still immature and can be further developed. Firstly, the linear and nonlinear systems have their complexity and diversity. Secondly, the control systems typically have interrupt noise and unknown input signals as well as known control signals. Besides, the introduction of the Internet in recent

years comes up with plenty of new issues, for example, packet loss, time delay, disorder, digitalisation, and so on. All of these bring further difficulties in FDD research. Therefore this thesis introduces a particular FDD approach and its application in detail.

2.2 Faults classification

Faults in dynamic systems mean the aberration of the system structure or the disparate system parameters forming the unusual situation. As the structure shown in Figure 2.2, the faults can be blocking of the actuator, loss of a sensor, or system component disconnection. Faults can change the interacting plant components or the plant and the controller interface. [8] Hence, faults are the unexpected elements that can improve system performance and cause degradation or a loss of system function.

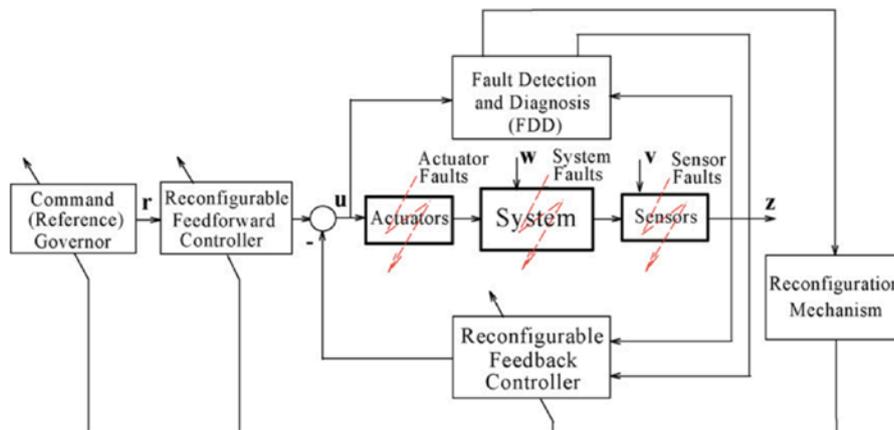


Figure 2.2 A general structure of FDD

Usually, the fault classification can be sensor faults, actuator faults, and plant component faults (system faults).

2.2.1 Sensor faults

Sensor fault, also named sensor data fault, is one of the major faults in FDD. The plant properties are not affected by the sensor fault, but the sensor readings have abundant errors. In the real project, three types of sensor faults often are observed: single-sample spikes sensor readings, longer duration noisy readings, and anomalous constant offset readings. [9] Numerous studies show that sensor fault frequently results in severe consequences for safety and plants operation. Therefore, it is essential to study sensor faults to guarantee the safety and the reliability of systems. [10]

2.2.2 Actuator faults

Actuator fault means the fault of moving or controlling components in the system. Such faults cannot affect the plant properties but can make the controller influence to be interrupted or modified. An actuator fault is a type of system input failure. It may occur in the systems due to abnormal operation or aging of material. Actuator faults can be represented by single or multiple faults that can change the system behaviour, leading to degradation or even instability. [11]

2.2.3 System component faults

System component faults are the other faults different from sensor faults and actuator faults. They can be caused by plant structure, system modelling or other reasons. These kinds of faults can change the system dynamical input/output properties. System component faults are the most sophisticated faults because the faults can occur anywhere in the system.

In this thesis, the FDD method only focuses on the actuator faults.

2.3 FDD methods

With decades of development, there are a great diversity of FDD methods. These methods are challenging problems in many disciplines such as aerospace engineering, nuclear engineering, chemical engineering, and automotive systems. [12] In general, FDD methods can be divided into two classes of redundancy; hardware redundancy and analytical redundancy.

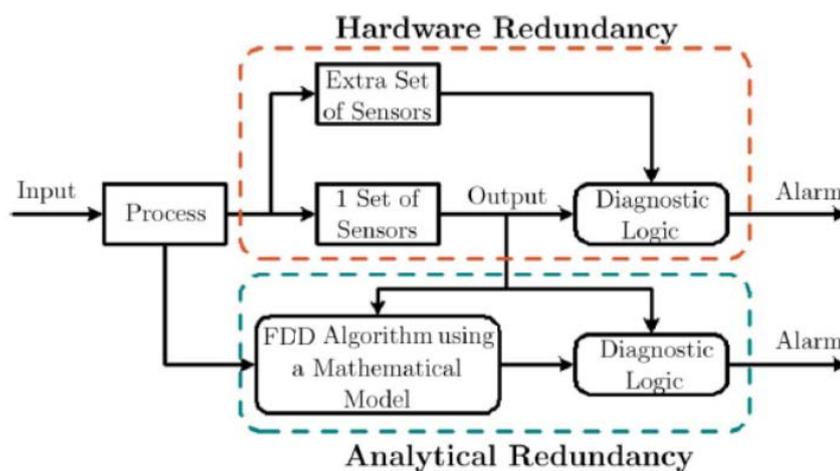


Figure 2.3 Illustrations of the concepts of hardware redundancy and analytical redundancy for FDD

2.3.1 Hardware redundancy

Hardware redundancy means to compare repetitive signals generated by various hardware, such as measurements of the same signal given by two or more physical copies of a hardware component. Hardware redundancy may be the only way to improve the dependability of a system.

Hardware redundancy has three classic types, which are passive, active, and hybrid. Without any information about action from the system, passive redundancy can achieve FDD by masking the faults. Active redundancy can achieve fault tolerance with the detected faults. After FDD, the actions are taken to remove the faulty component from the system. Active techniques must stop the system to replace the system fault component. Hybrid redundancy combines the two approaches that are mentioned above. Hybrid redundancy can fix system faults without system downtime.

1. Passive redundancy

Instead of detecting faults, the passive redundancy approach masks faults to achieve redundancy. This method only allows the correct values to perform as the system output regardless of the fault percentage. Passive redundancy techniques are normally used in applications with high-reliability. The high-reliability system does not accept any interruptions and it is impossible to be repaired. The high-reliability system examples include an air force combat control system, cardiac pacemaker, and deep-space electronics.

2. Active redundancy

Active redundancy method detects the faults in the system, then takes appropriate action regarding recovering the fault and makes the system return to the original state. This is usually applied in high availability systems, for example, transaction processing systems and time-shared computing systems. The high availability system allows the temporary fault to happen, which can be detected by active redundancy. However, these faults can be recovered quickly, and the system can return to a normal operating state.

3. Hybrid redundancy

Hybrid redundancy takes advantages from passive and active approaches and makes the better choice to tolerate the faults among the mentioned methods. It uses fault masking when the system has unpredicted faults that can bring more problems to the system. While the fault

tolerant method is also used when the unwanted faults can be fixed. Hybrid redundancy is normally used in systems, which have very high safety requirements. Examples of these kinds of systems are chemical process control systems, nuclear power plants, ordnance manufacturing systems, medical manufacturing systems, aerospace systems, traffic systems, and so on. [13]

Hardware redundancy also has its disadvantages: increasing weight, size, power consumption, cost, design time, fabrication and testing time. To achieve the best result, a few choices must be examined before incorporating hardware redundancy into a system.

2.3.2 Analytical redundancy

Analytical redundancy builds the mathematical model of the system and uses different algorithms to estimate the system state to achieve FDD. The analytical redundancy approach does not need additional hardware to perform its function, which can save money and time compared to the hardware redundancy. However, the target system is complicated by noise, an uncertain model, and unexpected disturbances, which makes the analytical redundancy approach more difficult to achieve. The systematic redundancy approach is harder to design compared to the hardware redundancy.

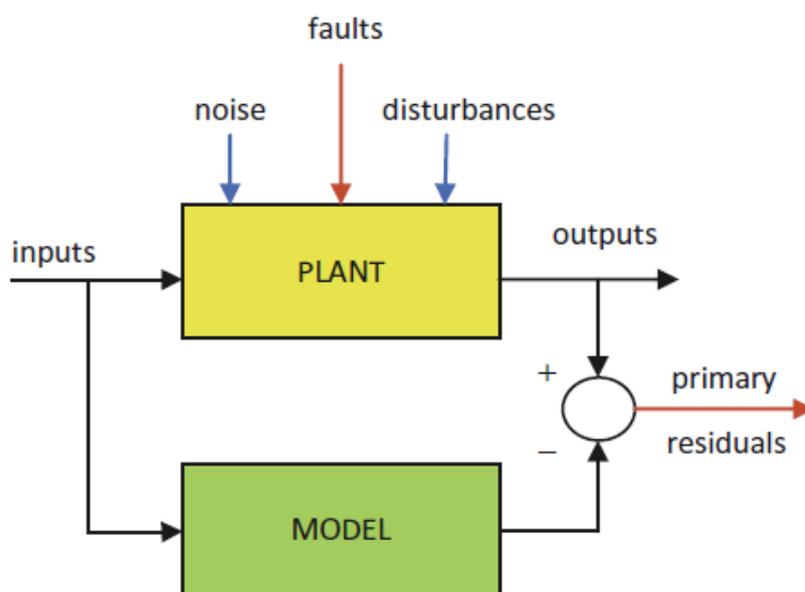


Figure 2.4 Analytical redundancy

There are many FDD methods of analytical redundancy, only several of the most popular methods are introduced in the following paragraph.

1. The parameter estimation

The parameter estimation means the processes use the sampling data to estimate the required parameters. The parameter estimation is a method, which provides tools for the efficient usage in sampling data to estimate the constants appearing in the mathematic model, and for helping to model the system. In a different situation, the modelled systems have their character, which can be algebraic, differential, integral equations, or their associated conditions. Depending on the system, the estimated parameter may or may not have a direct physical significance.

The parameter estimation can also be applied to the opposite problem. By knowing the initial condition, the boundary, and other parameters of the system, the parameter estimation can find the partial differential equations of the unknown systems. During the process, not all the constants of the system must need to be known. The constants that need to be estimated or used to find the system characters in parameter estimation are also called parameters. [14]

The parameter estimation can be divided into Rank Regression (or Least Squares), Maximum Likelihood Estimation and Bayesian Estimation Methods.

2. The Parity relation

Another popular approach of analytical redundancy is the Parity relation. This method is firstly used to detect sensory faults in military control systems. Later, academics used it to detect not only sensor faults but also actuator faults. A lot of experiments show that the Parity relation is only suitable for linear systems. In linear systems, the Parity relation can detect the unexpected fault in a concise period and report the exact fault location. This method can easily handle systems with noise, unknown disturbance, and uncertain model. The most suitable situation is the single output system where the observer-based method is impractical.

In 2007, academics have tried to apply the technique to the input-output model in nonlinear systems by using the TS fuzzy models.[15] In 2011, another group applied the Parity space approach to the fault detection and isolation based on nonlinear analytical redundancy. [16] These examples show the ability of the Parity relation in nonlinear systems.

3. The Beard-Jones filter

Beard and Jones first devised the Beard-Jones filter for fault diagnosis in real time systems. This filter is a state estimator like device, which can be only used in fault diagnosis. The core of the Beard-Jones fault detection filter is a Luenberger observer, which is designed in a unique

way that results in an “error” system. In the Luenberger observer, the faults model treats the fault as part of the input, and has a diagonal stable transfer function matrix. [17]

This approach does not rely on prior assumptions about the mode of component failure. The Beard-Jones filter is an individual observer that accentuates the effect of failure on the innovation (or prediction error) of the observer. The observer gain is chosen to detect the innovation vector in the output space, which can be used to identify the failed component. [18]

The Beard-Jones filter is also designed for the linear system at the beginning. However, more and more theoretical developments in the Beard-Jones filter tried to adjust the filter to nonlinear applications.

4. The Kalman filter

In 1960, the Hungarian born American mathematician Kalman presented a new filter approach: The Kalman filter, which can be quickly calculated by the computer.

(1) The basic Kalman filter (KF)

The KF, also known as a linear quadratic estimation, is an algorithm based on a description of SSR with signals and noise. This filter requires that the system state equation and the output equation are the linear equation. The system noise and the measurement noise in the system are Gaussian white noise and independent of each other. By knowing all statistic features, the KF filter is the minimum variance unbiased estimator for the dynamic system unknown state. [19]

The KF only needs the information of current measurement and previous estimation to calculate the current estimation by using the existing model. [20] This algorithm has many advantages.

(a) This algorithm brought the state variable into filter theory, along with solving filter problems of time-varying, multivariable, and non-stationary time series.

(b) The KF is a recursive algorithm that can be easily calculated by the computer.

(c) The KF does not need to store the historical data and is convenient for real-time processing. [21]

The linear KF is sufficiently mature with plentiful applications, for example, the Apollo Moon-landing project. The Kalman filter is an optimal filter in the linear system and can be easily

achieved in many systems. In this thesis, the system is modelled to a linear system that makes the Kalman filter the best choice.

(2) The extended Kalman filter (EKF)

In the next decades, academics tried to find a better way to apply the Kalman filter in the nonlinear system. One of the approaches is assuming all the changes are quasi-linear while ignoring high order terms. The first order Taylor series expansion is used to approximate the nonlinear systems in this method. [22] The EKF is an excellent estimator in nonlinear systems and has the outstanding robustness to model inaccuracy, parameter variations, measurement noise, and system uncertainties. [23] This filter makes a more natural algorithm, more convenient operation and at a cheaper cost. Therefore, it is widely used in industrial systems. However, it still has some theory limitations, which are discussed in the following:

- (a) When the systems have very high order items, the EKF method will increase the system error due to ignoring high order items with Taylor series expansion.
- (b) If the initial estimate of the system state is incorrect, or if the modelling system is wrong, the system will quickly diverge. [24]
- (c) When the system achieved the stable state, EKF will lose the ability to track saltation.

So EKF is designed as a new approach for nonlinear systems, but its shortages limit its application in more complicated and sensitive systems.

(3) The unscented Kalman filter (UKF)

To address the approximation issues of the EKF, a new FDD method, Julier and Uhlman came up with the unscented Kalman filter (UKF). This method is based on the unscented transformation, which can change the average value and covariance by sigma points. [25] The algorithm is not restricted to Gaussian distributions and based on the intuition that is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function or transformation. [26] Compared with the EKF, the UKF has a lot of advantages:

- (a) The UKF does not need to calculate the system's Jacobi matrix, which makes the process easier.
- (b) The UKF is approximating a probability distribution rather than approximating an arbitrary nonlinear function or transformation, which makes the system estimate accuracy higher than the 1st order in Taylor series.

Chapter 2: Literature Review

(c) The UKF can be used to solve the systems with Gaussian noise and has wider application.

(d) The UKF uses the certainty-sampling method, which makes the filter accuracy improved.

Overall, the UKF method is the better FDD method for nonlinear systems comparing with the EKF method. For further research of this thesis, the UKF method is another choice in the nonlinear system model.

Chapter 3 : System Identification of Plate Structure

To analyse and control a complex dynamic system, it is necessary to build a system model that shows the dynamic system as a mathematical model. The mathematical model is a simplified dynamic system, which contains all the essential information of the original system.

In this chapter, the method to build the existing dynamic system, the plate structure, into a mathematical model is discussed in detail. Firstly, the MIMO plate structure feature and characteristics are introduced in detail for analysing the dynamic model. Secondly, the theoretical method is introduced to analyse and build the transfer function between inputs and outputs. This theoretical method aims to get the general mathematic transfer function for describing the physical systems. The experiment is conducted to get the system parameters by using the ModalVIEW software. The system frequency response function (FRF) curve can be observed to calculate the system parameters, such as model shapes, natural frequency and damping ratios, for building the transfer function model. After that, the system transfer function is changed into a continuous-time SSR to apply the Kalman filter in FDD. The SSR can express the system relationships in detail, for example, position, velocity and acceleration. Finally, the continuous-time SSR is transformed to a discrete-time SSR by using the zero hold discretization method.

3.1 Plate structure system components

Figure 3.1 shows the MIMO mechanical plate structure that is used in FDD. This plate structure includes a base plate, a top plate, a disturbance transducer, three pairs of sensors on the top plate, and three pairs of actuators between top plate and base plate.

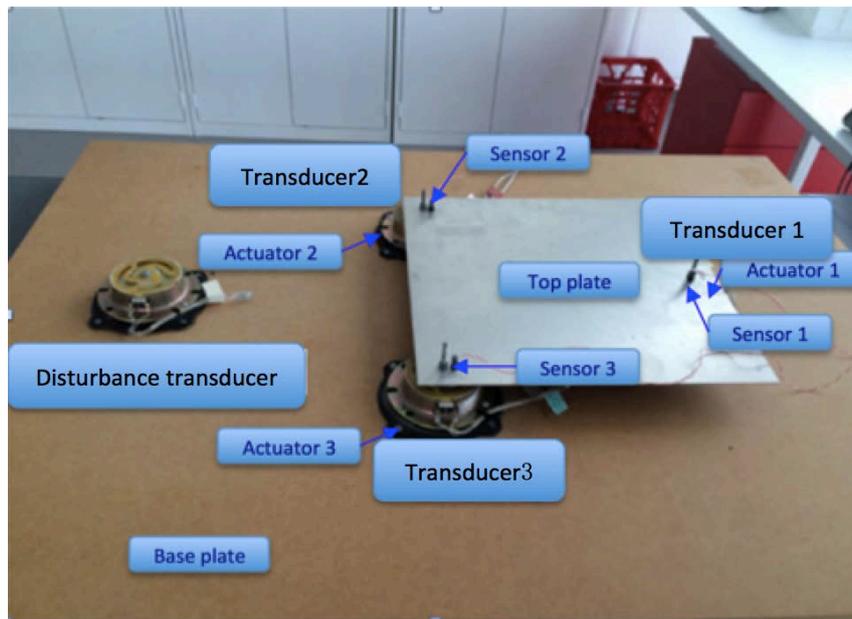


Figure 3.1 The MIMO mechanical structure

The whole system including sensors, actuators, and top plate is built on one side of the base plate. The three sensors, which are accelerometers, can generate the amplitude of vibration signals. The transducer 1, 2 and 3 are used as the three actuators that can operate the controlled signals to reduce the vibration. The top plate is the main controlled object, which mounts the three actuators by screws. The control aim is to keep the top plate motionless. At the other side of the base plate, the disturbance transducer is used to generate artificial unknown disturbance to the system.

As mentioned above, the system's function is to reduce the top plate vibration from the artificial disturbance. The system's working process is described in the block diagram of Figure 3.2.

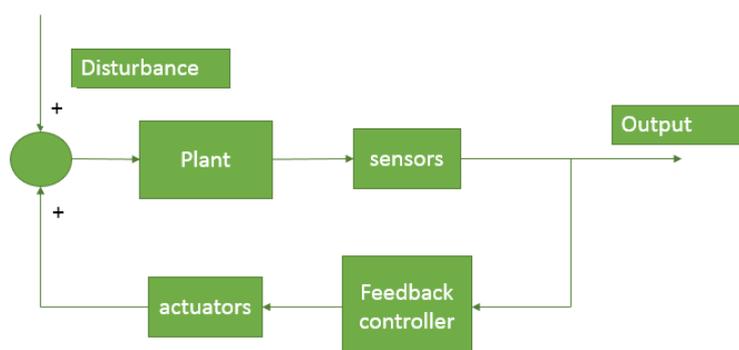


Figure 3.2 The MIMO close-loop system block diagram

In the block diagram, when the disturbance transducer starts to generate unknown vibration disturbance, the base plate will be influenced and transmit the vibration to three actuators by the connecting screws. Though the vibration transmission from the actuator, the top plate will no longer be stationary and start to vibrate at the same frequency of the disturbance signal. Then the sensors on the top plate sense the vibration amplitude and send the signals to the controller, which contains the control law to produce the controlled signals. Finally, the actuators use the controlled signals to cancel the vibration on the top plate.

3.2 Theoretical open-loop transfer function for the plate structure

3.2.1 The general transfer function for the open-loop system

For a general open-loop system, the transfer function can be expressed as the following equation.

$$Y(s) = G(s) \times U(s) \quad (3.1)$$

Where $Y(s)$ can either be a variable or a vector and is the output signal of the system, $U(s)$ can either be a variable or a vector and is the input signal of the system, and $G(s)$ can either be a scalar or a matrix and is the transfer matrix of the system.

3.2.2 The plate structure transfer function for the open-loop system

The plate structure system shown in Figure 3.3 is a three-input and three-output system. To get the transfer function of the plate structure, an input signal is introduced to each actuator separately. When one of actuator starts to vibrate by using the input signal, the three sensors can detect output signals at the same time. So the transfer function between single actuator and single sensor can be derived as following.

$$Y_n(s) = G_{nm}(s) \times U_m(s) \quad (3.2)$$

Where $Y_n(s)$ is the single sensor output, the $U_m(s)$ is the single actuator input, and the $G_{nm}(s)$ is the transfer function between the single actuator input and single sensor output. The number n ranges from 1 to 3 and represents the sensor 1, 2, and 3 respectively, and the number m ranges from 1 to 3 and represents the actuator 1, 2, and 3 respectively.

For example, if the transfer function of the plate structure system is:

$$Y_1(s) = G_{12}(s) \times U_2(s) \quad (3.3)$$

This represents the transfer function between input 1 and output 2.

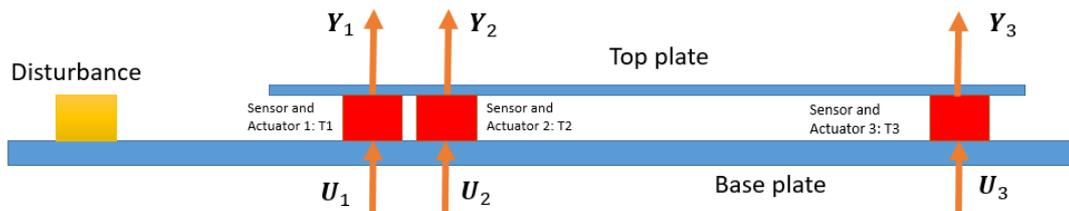


Figure 3.3 Plate layout for open-loop system

The plate layout related to the plate structure open-loop system modelling also describes the relationship between inputs and outputs. For the whole plate structure system, the system transfer function can be derived as:

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \\ Y_3(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) \end{bmatrix} \times \begin{bmatrix} U_1(s) \\ U_2(s) \\ U_3(s) \end{bmatrix} \quad (3.4)$$

So for the open-loop system, the transfer matrix $\mathbf{G}(s)$ for the plate structure is a 3*3 matrix, Which means that the measured output signal in each sensor is influenced by three actuators at the same time and the input signal in each actuator impacts the three measured output signals at the same time. The relationship between three sensor inputs and three actuator outputs is also represented in the block diagram below.

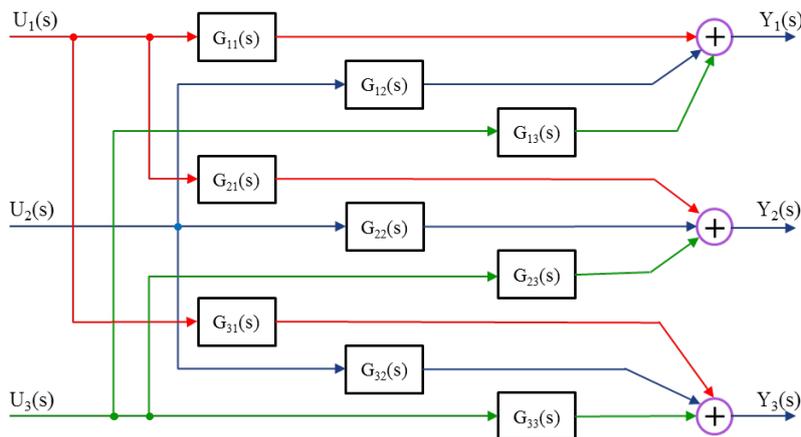


Figure 3.4 MIMO control system block diagram

The plate structure system has been clearly explained above and its transfer function is derived to model the system. To have a better understanding of the system transfer function, theoretical analysis is introduced to delineate the transfer matrix $\mathbf{G}(s)$.

3.3 The theoretical analysis of the system transfer matrix

For each transfer function between the single actuator input and single sensor output, the transfer function can be written as the following equation [27].

$$\mathbf{G}_{nm}(s) = \sum_{k=1}^{\infty} \frac{\varphi_n^k \varphi_m^k}{s^2 + 2\zeta_{nm}^k w_{nm}^k s + w_{nm}^{k2}} = \sum_{k=1}^{\infty} \frac{\varphi_{nm}^k}{s^2 + 2\zeta_{nm}^k w_{nm}^k s + w_{nm}^{k2}} \quad (3.5)$$

Where k is the number of the mode, φ_n^k and φ_m^k are mode shapes related to the actuator n and the sensor m respectively, φ_{nm}^k is integrated by φ_n^k and φ_m^k and is the mode shape between the sensor n and the actuator m , ζ_{nm}^k is the damping ratio between the sensor n and the actuator m , and w_{nm}^k is the nature frequency between the sensor n and the actuator m .

The representation of the transfer function between the single actuator input and single sensor output is critical for building the system. The parameters observed in the experiment are mode shape, damping ratio, and nature frequency. In the next section, the experiment's method and results are given to complete the system transfer function.

3.4 Experiment of transfer function modelling

This experiment aims to produce the system transfer function parameters, such as mode shape, damping ratio, and nature frequency. The physical experiment arrangement is shown in Figure 3.5.

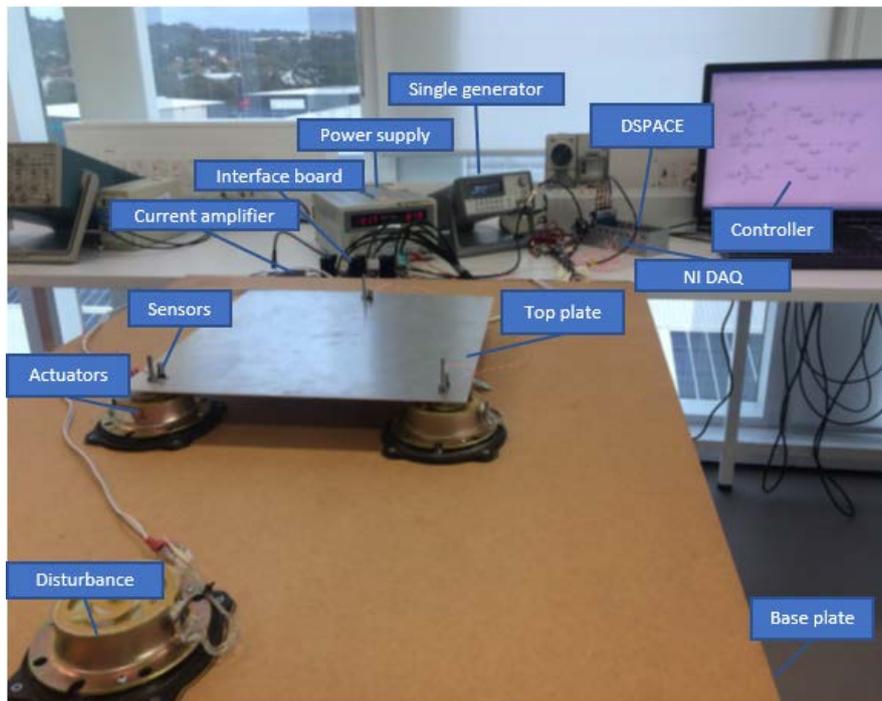


Figure 3.5 The arrangement of the physical system

3.4.1 Open-loop transfer function modelling method.

In the experiment, the signal generator produces a sinusoidal sweep signal, which has a frequency ranging from 20 Hz to 100 Hz, and sends it to each actuator respectively. The sinusoidal sweep signal is used as the input signal $U_1(s)$, $U_2(s)$, and $U_3(s)$ in the corresponding actuator 1, 2, and 3. When the sinusoidal is inserted to one of the actuators, the three sensors can detect the output signals $Y_1(s)$, $Y_2(s)$, and $Y_3(s)$ at the same time. The data acquisition system (NI DAQ) recodes the sinusoidal sweep signal, and three output signals, and sends them to ModalVIEW software for computing the frequency response function.

$$\mathbf{G}_{nm}(s) = \frac{\mathbf{Y}_n(s)}{\mathbf{U}_m(s)} \quad (3.6)$$

3.4.2 The experiment result parameters

As shown in Figure 3.6, the experiment shows each transfer function $\mathbf{G}_{nm}(s)$ is a complicated function that roughly has 11 modes. To simplify the transfer function $\mathbf{G}_{nm}(s)$, the first three modes are chosen to calculate their parameters, because they have the most significant impact on the system.

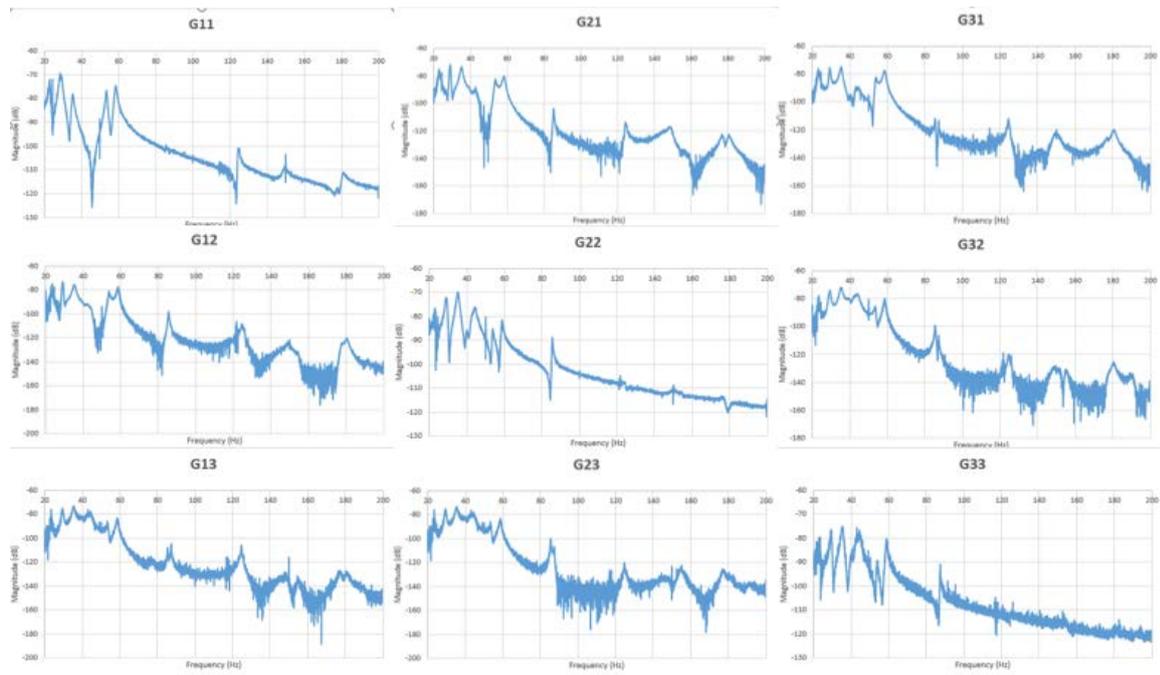


Figure 3.6 Measured FRF curves for the open-loop real system

The ModalVIEW software can provide a Multiple Degree of Freedom (MDOF) polynomial curve fitting method to compute the transfer function parameters in all the modes. The first three modes' parameters of each transfer function between single sensor and single actuator are shown in Table 1.

Table 1. The first three mode parameters of the transfer function model

$G_{11}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.15363842	0.01893171	145.907272
2	0.41430841	0.01653821	182.852866
3	0.19853058	0.01802016	222.785222
$G_{21}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.11828148	0.017293	146.334548
2	0.18811558	0.0102591	184.371104
3	0.34159332	0.01584925	221.993378
$G_{31}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.10027307	0.01737686	146.302265

Chapter 3: System Identification of Plate Structure

2	0.1920355	0.01125181	183.906556
3	0.27221838	0.01641266	221.994835
$G_{12}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.07290046	0.01208586	146.205623
2	0.17558329	0.0090579	181.912795
3	0.34610302	0.02018604	221.391273
$G_{22}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.05076275	0.01225517	145.738971
2	0.18305818	0.01036957	183.839301
3	0.61395187	0.01988154	221.280048
$G_{32}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.04862492	0.01229378	145.757852
2	0.15373242	0.01063791	183.606333
3	0.51509689	0.01994371	221.295643
$G_{13}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.04698852	0.0108145	145.234739
2	0.07202531	0.00769278	184.634413
3	0.18801199	0.01680347	222.570149
$G_{23}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.03451708	0.0108145	145.234739
2	0.10226404	0.00769278	184.634413
3	0.35679415	0.01680347	222.570149
$G_{33}(s)$			
Mode	Mode shape φ_{nm}^k	Damping ratio ζ_{nm}^k	Natural frequency (rad/s) w_{nm}^k
1	0.02300727	0.00939594	144.835311
2	0.07687854	0.00754164	184.843744
3	0.25701579	0.01651211	222.45161

By knowing all the parameters of the first three modes, the 3*3 plant structure transfer matrix can be derived by using theoretical analyse of the system transfer function. [28]

The system transfer function model is completed as following.

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \\ Y_3(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) \end{bmatrix} \times \begin{bmatrix} U_1(s) \\ U_2(s) \\ U_3(s) \end{bmatrix} \quad (3.7)$$

And

$$G_{nm}(s) = \sum_{k=1}^3 G_{nm}^k(s) \quad (3.8)$$

Where $G_{nm}^k(s)$ is the transfer function between the single actuator input and single sensor output at mode k , where k can be 1, 2, and 3 representing different modes.

3.5 Construction of continuous-time SSR

The SSR can describe many processes in the real world. It is a mathematical model of the system, which helps to control the process and observe information by using mathematical tools. For the system mathematical analysis and the FDD purpose, the system transfer function representation needs to transform to a continuous-time SSR firstly. Then the continuous-time SSR is transformed to a discrete-time SSR by using the zero-order hold discretization method.

As the system transfer function has been worked out before, the transfer function of the measurement output for each sensor can be produced individually.

For sensor 1:

$$Y_1(s) = [G_{11}^1(s) + G_{11}^2(s) + G_{11}^3(s)] \times U_1(s) + [G_{12}^1(s) + G_{12}^2(s) + G_{12}^3(s)] \times U_2(s) + [G_{13}^1(s) + G_{13}^2(s) + G_{13}^3(s)] \times U_3(s) \quad (3.9)$$

For sensor 2:

$$Y_2(s) = [G_{21}^1(s) + G_{21}^2(s) + G_{21}^3(s)] \times U_1(s) + [G_{22}^1(s) + G_{22}^2(s) + G_{22}^3(s)] \times U_2(s) + [G_{23}^1(s) + G_{23}^2(s) + G_{23}^3(s)] \times U_3(s) \quad (3.10)$$

For sensor 3:

$$Y_3(s) = [G_{31}^1(s) + G_{31}^2(s) + G_{31}^3(s)] \times U_1(s) + [G_{32}^1(s) + G_{32}^2(s) + G_{32}^3(s)] \times U_2(s) + [G_{33}^1(s) + G_{33}^2(s) + G_{33}^3(s)] \times U_3(s) \quad (3.11)$$

The transfer function between single actuator and single sensor is introduced before, which is the transfer function for all the modes.

$$\mathbf{G}_{nm}(s) = \sum_{k=1}^{\infty} \frac{\varphi_n^k \varphi_m^k}{s^2 + 2\zeta_{nm}^k w_{nm}^k s + w_{nm}^{k2}} \quad (3.12)$$

For the single mode transfer function between single actuator and single sensor, the transfer function can be written as:

$$\mathbf{G}_{nm}^k(s) = \frac{\varphi_n^k \varphi_m^k}{s^2 + 2\zeta_{nm}^k w_{nm}^k s + w_{nm}^{k2}} \quad (3.13)$$

The state space modelling method used in this thesis sets a common denominator for the entire modes. The denominator in each mode has different damping ratios and natural frequencies, which need to be uniform. For uniforming the damping ratio, the largest value of the damping ratios in one mode is selected to be the general damping ratio. For uniforming the natural frequency, the average value of the natural frequency values is selected to be the general frequency. Therefore two general equations can be defined to describe the general parameters of the common denominator.

$$\begin{aligned} \zeta_{nm}^k &= \zeta^k \\ w_{nm}^k &= w^k \end{aligned} \quad (3.14)$$

Where ζ^k is the largest value of the damping ratio at mode k and w^k is the average value of the nature frequency at mode k .

By using the common denominator, the transfer function between single actuator and single sensor at the single mode is changed to a simplified equation.

$$\mathbf{G}_{nm}^k(s) = \frac{\varphi_n^k \varphi_m^k}{s^2 + 2\zeta^k w^k s + w^{k2}} \quad (3.15)$$

Sub the simplified equation (3.15) to each sensor transfer function. The transfer function of the measurement output for each sensor is changed as follows:

For sensor 1:

$$\begin{aligned}
 \mathbf{Y}_1(s) &= \left[\frac{\varphi_1^1 \varphi_1^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_1^2 \varphi_1^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_1^3 \varphi_1^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_1(s) \\
 &\quad + \left[\frac{\varphi_1^1 \varphi_2^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_1^2 \varphi_2^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_1^3 \varphi_2^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_2(s) \\
 &\quad + \left[\frac{\varphi_1^1 \varphi_3^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_1^2 \varphi_3^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_1^3 \varphi_3^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_3(s) \\
 &= \frac{\varphi_1^1 (\varphi_1^1 \mathbf{U}_1(s) + \varphi_2^1 \mathbf{U}_2(s) + \varphi_3^1 \mathbf{U}_3(s))}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} \\
 &\quad + \frac{\varphi_1^2 (\varphi_1^2 \mathbf{U}_1(s) + \varphi_2^2 \mathbf{U}_2(s) + \varphi_3^2 \mathbf{U}_3(s))}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \\
 &\quad + \frac{\varphi_1^3 (\varphi_1^3 \mathbf{U}_1(s) + \varphi_2^3 \mathbf{U}_2(s) + \varphi_3^3 \mathbf{U}_3(s))}{s^2 + 2\zeta^3 w^3 s + w^{3^2}}
 \end{aligned} \tag{3.16}$$

For sensor 2:

$$\begin{aligned}
 \mathbf{Y}_2(s) &= \left[\frac{\varphi_2^1 \varphi_1^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_2^2 \varphi_1^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_2^3 \varphi_1^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_1(s) \\
 &\quad + \left[\frac{\varphi_2^1 \varphi_2^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_2^2 \varphi_2^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_2^3 \varphi_2^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_2(s) \\
 &\quad + \left[\frac{\varphi_2^1 \varphi_3^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_2^2 \varphi_3^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_2^3 \varphi_3^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_3(s)
 \end{aligned} \tag{3.17}$$

$$\begin{aligned}
 &= \frac{\varphi_2^1(\varphi_1^1 \mathbf{U}_1(s) + \varphi_2^1 \mathbf{U}_2(s) + \varphi_3^1 \mathbf{U}_3(s))}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} \\
 &\quad + \frac{\varphi_2^2(\varphi_1^2 \mathbf{U}_1(s) + \varphi_2^2 \mathbf{U}_2(s) + \varphi_3^2 \mathbf{U}_3(s))}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \\
 &\quad + \frac{\varphi_2^3(\varphi_1^3 \mathbf{U}_1(s) + \varphi_2^3 \mathbf{U}_2(s) + \varphi_3^3 \mathbf{U}_3(s))}{s^2 + 2\zeta^3 w^3 s + w^{3^2}}
 \end{aligned}$$

For sensor 3:

$$\begin{aligned}
 \mathbf{Y}_3(s) &= \left[\frac{\varphi_3^1 \varphi_1^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_3^2 \varphi_1^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_3^3 \varphi_1^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_1(s) \\
 &\quad + \left[\frac{\varphi_3^1 \varphi_2^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_3^2 \varphi_2^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_3^3 \varphi_2^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_2(s) \\
 &\quad + \left[\frac{\varphi_3^1 \varphi_3^1}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} + \frac{\varphi_3^2 \varphi_3^2}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \right. \\
 &\quad \left. + \frac{\varphi_3^3 \varphi_3^3}{s^2 + 2\zeta^3 w^3 s + w^{3^2}} \right] \times \mathbf{U}_3(s) \tag{3.18} \\
 &= \frac{\varphi_3^1(\varphi_1^1 \mathbf{U}_1(s) + \varphi_2^1 \mathbf{U}_2(s) + \varphi_3^1 \mathbf{U}_3(s))}{s^2 + 2\zeta^1 w^1 s + w^{1^2}} \\
 &\quad + \frac{\varphi_3^2(\varphi_1^2 \mathbf{U}_1(s) + \varphi_2^2 \mathbf{U}_2(s) + \varphi_3^2 \mathbf{U}_3(s))}{s^2 + 2\zeta^2 w^2 s + w^{2^2}} \\
 &\quad + \frac{\varphi_3^3(\varphi_1^3 \mathbf{U}_1(s) + \varphi_2^3 \mathbf{U}_2(s) + \varphi_3^3 \mathbf{U}_3(s))}{s^2 + 2\zeta^3 w^3 s + w^{3^2}}
 \end{aligned}$$

An overall block diagram for the plate structure system is drawn to compute the SSR from the three sensors' transfer functions of the measurement outputs.

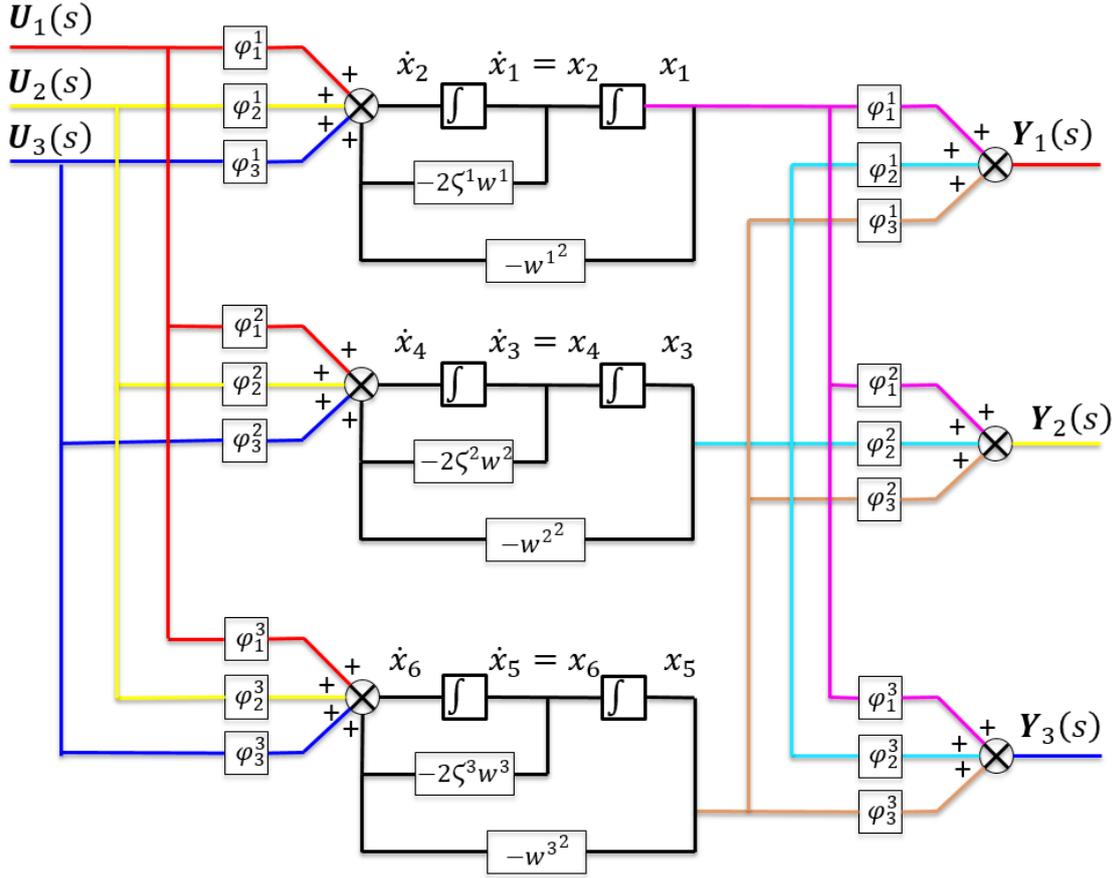


Figure 3.7 The block diagram of the plate structure system

The block diagram shows the relationship between the three inputs and the three outputs. From

the set state in both sides of the integration block, the equations of each state $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}$ can be

derived.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = [\varphi_1^1 \quad \varphi_2^1 \quad \varphi_3^1] \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - 2\zeta^1 w^1 x_2 - w^{1^2} x_1 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = [\varphi_1^2 \quad \varphi_2^2 \quad \varphi_3^2] \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - 2\zeta^2 w^2 x_4 - w^{2^2} x_3 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = [\varphi_1^3 \quad \varphi_2^3 \quad \varphi_3^3] \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - 2\zeta^3 w^3 x_6 - w^{3^2} x_5 \end{cases} \quad (3.19)$$

Chapter 3: System Identification of Plate Structure

For a general system in SSR, consider the process model is an n -th-order, linear, time invariant, continuous-time system with r inputs and m outputs.

$$\dot{\mathbf{X}}(t) = \mathbf{A}\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t) \quad (3.20)$$

$$\mathbf{Y}(t) = \mathbf{C}\mathbf{X}(t)$$

Where $\mathbf{X}(t)$ is $(n * 1)$ state vector, $\mathbf{Y}(t)$ is $(m * 1)$ output vector, and $\mathbf{U}(t)$ is $(r * 1)$ input vector. The state $\dot{\mathbf{X}}(t)$ is the input vector before the integrator of $\mathbf{X}(t)$. The \mathbf{A} matrix is $(n * n)$ system matrix, the \mathbf{B} matrix is $(n * r)$ input matrix, and the \mathbf{C} matrix is $(m * n)$ output matrix.

In the plate structure system, the state vector $\mathbf{X}(t) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}$, the output vector $\mathbf{Y}(t) = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$, the

system matrix $\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -w^{1^2} & -2\zeta^1 w^1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -w^{2^2} & -2\zeta^2 w^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -w^{3^2} & -2\zeta^3 w^3 \end{bmatrix}$, the input

matrix $\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ \varphi_1^1 & \varphi_2^1 & \varphi_3^1 \\ 0 & 0 & 0 \\ \varphi_1^2 & \varphi_2^2 & \varphi_3^2 \\ 0 & 0 & 0 \\ \varphi_1^3 & \varphi_2^3 & \varphi_3^3 \end{bmatrix}$, and the output matrix $\mathbf{C} = \begin{bmatrix} \varphi_1^1 & 0 & \varphi_1^2 & 0 & \varphi_1^3 & 0 \\ \varphi_2^1 & 0 & \varphi_2^2 & 0 & \varphi_2^3 & 0 \\ \varphi_3^1 & 0 & \varphi_3^2 & 0 & \varphi_3^3 & 0 \end{bmatrix}$.

Based on Table 1 parameters and the system transfer function analysis, the parameters in the system SSR are calculated in the following table.

Table 2. The system parameters of the SSR

Mode 1		Mode 2		Mode 3	
φ_1^1	0.3639	φ_1^2	0.7229	φ_1^3	0.4552
φ_2^1	0.2821	φ_2^2	0.4803	φ_2^3	0.8414
φ_3^1	0.2384	φ_3^2	0.4456	φ_3^3	0.758
ζ^1	0.02677	ζ^2	0.02533	ζ^3	0.02419
w^1	142.3048	w^2	182.434	w^3	218.9208

3.6 Construction of discrete-time SSR

To apply the discrete-time Kalman filter algorithm, the plate structure system is finally transformed to a discrete-time system SSR. There are several discretization methods, such as the zero order hold on the inputs, the linear interpolation of inputs (first order hold), the bilinear (Tustin) approximation, and the matched pole-zero method (for SISO system only). In this thesis, the discretization method, the zero order hold on the inputs, is explained in detail, and used to convert the continuous-time system to a discrete-time system.

The general form of the continuous-time system SSR was discussed before.

$$\begin{aligned}\dot{\mathbf{X}}(t) &= \mathbf{A}\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t) \\ \mathbf{Y}(t) &= \mathbf{C}\mathbf{X}(t)\end{aligned}\tag{3.21}$$

Where the first equation is called the state equation and the second equation is called the output equation.

To apply the zero order hold on the inputs method, the state equation is transformed into the following equation.

$$\dot{\mathbf{X}}(t) - \mathbf{A}\mathbf{X}(t) = \mathbf{B}\mathbf{U}(t)\tag{3.22}$$

Multiply both side of the equation (3.22) by e^{-At} at the same time:

$$e^{-At}[\dot{\mathbf{X}}(t) - \mathbf{A}\mathbf{X}(t)] = e^{-At}\mathbf{B}\mathbf{U}(t)\tag{3.23}$$

Since:

$$\begin{aligned}\frac{d}{dt}[e^{-At}\mathbf{X}(t)] &= \left[\frac{d}{dt}e^{-At}\right]\mathbf{X}(t) + e^{-At}\left[\frac{d}{dt}\mathbf{X}(t)\right] \\ &= e^{-At}(-\mathbf{A})\mathbf{X}(t) + e^{-At}\dot{\mathbf{X}}(t) = e^{-At}[\dot{\mathbf{X}}(t) - \mathbf{A}\mathbf{X}(t)]\end{aligned}\tag{3.24}$$

Sub the equation (3.24) to the equation (3.23):

$$\frac{d}{dt}[e^{-At}\mathbf{X}(t)] = e^{-At}\mathbf{B}\mathbf{U}(t)\tag{3.25}$$

Take the integration of both sides from t_0 to t :

$$\begin{aligned}\int_{t_0}^t \frac{d}{d\tau}[e^{-A\tau}\mathbf{X}(\tau)] d\tau &= \int_{t_0}^t e^{-A\tau}\mathbf{B}\mathbf{U}(\tau) d\tau \\ e^{-At}\mathbf{X}(t) - e^{-At_0}\mathbf{X}(t_0) &= \int_{t_0}^t e^{-A\tau}\mathbf{B}\mathbf{U}(\tau) d\tau\end{aligned}\tag{3.26}$$

Multiply both sides of the equation (3.26) by e^{At} at the same time:

$$e^{At} e^{-At} \mathbf{X}(t) = e^{At} e^{-At_0} \mathbf{X}(t_0) + e^{At} \int_{t_0}^t e^{-A\tau} \mathbf{B}\mathbf{U}(\tau) d\tau \quad (3.27)$$

$$\mathbf{X}(t) = e^{A(t-t_0)} \mathbf{X}(t_0) + \int_{t_0}^t e^{A(t-\tau)} \mathbf{B}\mathbf{U}(\tau) d\tau$$

Assume t_0 to be the initial time "0", which indicates the integration is from the initial time:

$$\mathbf{X}(t) = e^{At} \mathbf{X}(0) + \int_0^t e^{A(t-\tau)} \mathbf{B}\mathbf{U}(\tau) d\tau \quad (3.28)$$

Assume for the corresponding discrete-time system, the sampling time is a constant value T , and the discretization method is the holding effect (Zero-Order Hold on inputs). Therefore the input vector $\mathbf{U}(t)$ changes only at sampling time T . In another word, $\mathbf{U}(t) = \mathbf{U}(kT)$, when $kT \leq t \leq (k+1)T$.

Let $t = (k+1)T$ and sub it into the equation (3.28):

$$\mathbf{X}[(k+1)T] = e^{A(k+1)T} \mathbf{X}(0) + e^{A(k+1)T} \int_0^{(k+1)T} e^{-A\tau} \mathbf{B}\mathbf{U}(\tau) d\tau \quad (3.29)$$

Let $t = kT$ and sub it into the equation (3.28):

$$\mathbf{X}(kT) = e^{AkT} \mathbf{X}(0) + e^{AkT} \int_0^{kT} e^{-A\tau} \mathbf{B}\mathbf{U}(\tau) d\tau \quad (3.30)$$

Multiply both sides of the equation (3.30) by e^{AT} at the same time:

$$e^{AT} \mathbf{X}(kT) = e^{A(k+1)T} \mathbf{X}(0) + e^{A(k+1)T} \int_0^{kT} e^{-A\tau} \mathbf{B}\mathbf{U}(\tau) d\tau \quad (3.31)$$

Use the equation (3.29) to subtract the equation (3.31), the equation (3.32) can be derived.

$$\mathbf{X}[(k+1)T] = e^{AT} \mathbf{X}(kT) + e^{A(k+1)T} \int_{kT}^{(k+1)T} e^{-A\tau} \mathbf{B}\mathbf{U}(\tau) d\tau \quad (3.32)$$

As mentioned above, when $kT \leq t \leq (k+1)T$, the $\mathbf{U}(\tau)$ is constant and $\mathbf{U}(\tau) = \mathbf{U}(kT)$. The equation (3.32) can be modified as following.

$$\mathbf{X}[(k+1)T] = e^{AT} \mathbf{X}(kT) + e^{A(k+1)T} \int_{kT}^{(k+1)T} e^{-A\tau} d\tau \mathbf{B} \mathbf{U}(kT) \quad (3.33)$$

Let $\lambda = (k+1)T - \tau$, the following equations can be derived.

$$\begin{aligned} d\lambda &= -d\tau \\ \lambda &= T, \text{ when } \tau = kT \\ \lambda &= 0, \text{ when } \tau = (k+1)T \end{aligned} \quad (3.34)$$

Therefore, the integration term inside equation (3.33) can be derived as follows.

$$\int_{kT}^{(k+1)T} e^{-A\tau} d\tau = \int_T^0 e^{-A[(k+1)T-\tau]} (-d\lambda) = e^{-A(k+1)T} \int_0^T e^{-A\lambda} d\lambda \quad (3.35)$$

Sub the changed integration term equation (3.35) back to the equation (3.33).

$$\mathbf{X}[(k+1)T] = e^{AT} \mathbf{X}(kT) + \int_0^T e^{-A\lambda} d\lambda \mathbf{B} \mathbf{U}(kT) \quad (3.36)$$

From the equation (3.36), the discrete-time model in state space can be formed.

$$\mathbf{X}[(k+1)T] = \mathbf{G}(T) \mathbf{X}(kT) + \mathbf{H}(T) \mathbf{U}(kT) \quad (3.37)$$

Where $\mathbf{G}(T)$ is $(n \times n)$ system matrix and $\mathbf{H}(T)$ is $(n \times r)$ input matrix. Both of these matrices are functions of the sampling period T .

$$\mathbf{G}(T) = e^{AT} \text{ and } \mathbf{H}(T) = \int_0^T e^{-A\lambda} d\lambda \mathbf{B} \quad (3.38)$$

For the output equation, the continuous-time model is based on the current value of \mathbf{X} . Therefore the discrete-time output equation is directly discretised by replacing t with kT .

$$\mathbf{Y}(kT) = \mathbf{C} \mathbf{X}(kT) \quad (3.39)$$

Now, the continuous-time system has been discretised into a discrete-time system. The discrete-time system SSR is given below.

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{G} \mathbf{X}(k) + \mathbf{H} \mathbf{U}(k) \\ \mathbf{Y}(k) &= \mathbf{C} \mathbf{X}(k) \end{aligned} \quad (3.40)$$

Where $\mathbf{X}(k)$ is $(n \times 1)$ state vector, $\mathbf{Y}(k)$ is $(m \times 1)$ output vector, and $\mathbf{U}(k)$ is $(r \times 1)$ input vector. The \mathbf{G} matrix is $(n \times n)$ system matrix, the \mathbf{H} matrix is $(n \times r)$ input matrix, and the \mathbf{C} matrix is $(m \times n)$ output matrix. Since the matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , of the continuous-time system are time invariant, the matrices \mathbf{G} , \mathbf{H} , \mathbf{C} , of the discrete-time system are also time invariant.

According to the natural frequency of the previously measured system, the natural frequency is about 222 Hz at mode 3. So the minimum period of the system is:

$$T = \frac{1}{f} = 0.0045s \quad (3.41)$$

Therefore, the sampling period is taken as 0.001 seconds, which is about a quarter of the period. For the plate structure system, the discrete-time SSR is constructed with the system matrix **G**, the input matrix **H**, and the output matrix **C**.

$$\mathbf{G} = \begin{bmatrix} 0.9899 & 0.001 & 0 & 0 & 0 & 0 \\ -20.1057 & 0.9824 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9835 & 0.001 & 0 & 0 \\ 0 & 0 & -32.9454 & 0.9743 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.9762 & 0.001 \\ 0 & 0 & 0 & 0 & -47.2935 & 0.9658 \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} 0.0002 & 0.0001 & 0.0001 \\ 0.3613 & 0.2801 & 0.2367 \\ 0.0004 & 0.0002 & 0.0002 \\ 0.7156 & 0.4754 & 0.4411 \\ 0.0002 & 0.0004 & 0.0004 \\ 0.4492 & 0.8303 & 0.7480 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 0.3639 & 0 & 0.7229 & 0 & 0.4552 & 0 \\ 0.2821 & 0 & 0.4803 & 0 & 0.8414 & 0 \\ 0.2384 & 0 & 0.4456 & 0 & 0.7580 & 0 \end{bmatrix}$$

Chapter 4 : Kalman Filter Based FDD Method

In this chapter, the discrete-time Kalman filter is discussed in detail and summarized into a single algorithm. To verify the Kalman filter algorithm, a simple application successfully uses the Kalman filter to estimate its state. After that, a Kalman filter bank and a normalization algorithm are designed to achieve the FDD Method.

4.1 The Discrete-time Kalman Filter algorithm

The system using the basic Kalman filter must be a linear system. [29] Consider the process model is an n^{th} -order, linear, time invariant, continuous-time system with r inputs and m outputs.

$$\begin{aligned} X(k+1) &= \mathbf{G}X(k) + \mathbf{H}U(k) \\ Y(k) &= \mathbf{C}X(k) \end{aligned} \quad (4.1)$$

In reality, the discrete-time system model above is not an idealized model. Due to many factors such as material, connection, and environment, the system process may have unknown process noise $\boldsymbol{\omega}$. Also the system measurement cannot be exactly accurate, measurement noise \mathbf{v} is widely found in the real time system model.

$$\begin{aligned} X(k+1) &= \mathbf{G}X(k) + \mathbf{H}U(k) + \boldsymbol{\omega}(k) \\ Y(k) &= \mathbf{C}X(k) + \mathbf{v}(k) \end{aligned} \quad (4.2)$$

Where the system process noise $\boldsymbol{\omega}(k)$ and measurement noise $\mathbf{v}(k)$ are zero-mean Gaussian distribution white noise, which are uncorrelated from each other. The unknown system process noise $\boldsymbol{\omega}(k)$ is a $(n * 1)$ vector with a $(n * n)$ known covariance \mathbf{Q} and the unknown system measurement noise $\mathbf{v}(k)$ is a $(m * 1)$ vector with a $(m * m)$ known covariance \mathbf{R} . The mathematical representation is respectively showing below.

$$\begin{aligned}
 \boldsymbol{\omega}(k) &= \begin{bmatrix} \boldsymbol{\omega}_1(k) \\ \boldsymbol{\omega}_2(k) \\ \dots \\ \boldsymbol{\omega}_n(k) \end{bmatrix} \sim (0, \mathbf{Q}) \\
 \mathbf{v}(k) &= \begin{bmatrix} \mathbf{v}_1(k) \\ \mathbf{v}_2(k) \\ \dots \\ \mathbf{v}_m(k) \end{bmatrix} \sim (0, \mathbf{R}) \\
 E[\boldsymbol{\omega}(k) \boldsymbol{\omega}(j)^T] &= \begin{cases} \mathbf{Q}, & k = j \\ 0, & k \neq j \end{cases} \\
 E[\mathbf{v}(k) \mathbf{v}(j)^T] &= \begin{cases} \mathbf{R}, & k = j \\ 0, & k \neq j \end{cases} \\
 E[\boldsymbol{\omega}(k) \mathbf{v}(j)^T] &= 0
 \end{aligned} \tag{4.3}$$

In another word, the process noise covariance \mathbf{Q} and the measurement noise covariance $\mathbf{R}(k)$ are diagonal matrices.

$$\begin{aligned}
 \mathbf{Q} &= \\
 &\left[\begin{array}{cccccc} E[\boldsymbol{\omega}(1) \boldsymbol{\omega}(1)^T] & E[\boldsymbol{\omega}(1) \boldsymbol{\omega}(2)^T] & \dots & E[\boldsymbol{\omega}(1) \boldsymbol{\omega}(n-1)^T] & E[\boldsymbol{\omega}(1) \boldsymbol{\omega}(n)^T] \\ E[\boldsymbol{\omega}(2) \boldsymbol{\omega}(1)^T] & & & & E[\boldsymbol{\omega}(2) \boldsymbol{\omega}(n)^T] \\ \vdots & & \ddots & & \vdots \\ E[\boldsymbol{\omega}(n-1) \boldsymbol{\omega}(1)^T] & & & & E[\boldsymbol{\omega}(n-1) \boldsymbol{\omega}(n)^T] \\ E[\boldsymbol{\omega}(n) \boldsymbol{\omega}(1)^T] & E[\boldsymbol{\omega}(n) \boldsymbol{\omega}(2)^T] & \dots & E[\boldsymbol{\omega}(n) \boldsymbol{\omega}(n-1)^T] & E[\boldsymbol{\omega}(n) \boldsymbol{\omega}(n)^T] \end{array} \right] \\
 &= \begin{pmatrix} E[\boldsymbol{\omega}(1) \boldsymbol{\omega}(1)^T] & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & E[\boldsymbol{\omega}(n) \boldsymbol{\omega}(n)^T] \end{pmatrix} = \text{diag}(\mathbf{Q}_{11}, \dots, \mathbf{Q}_{nn}) \\
 \mathbf{R} &= \\
 &\left[\begin{array}{cccccc} E[\mathbf{v}(1) \mathbf{v}(1)^T] & E[\mathbf{v}(1) \mathbf{v}(2)^T] & \dots & E[\mathbf{v}(1) \mathbf{v}(n-1)^T] & E[\mathbf{v}(1) \mathbf{v}(n)^T] \\ E[\mathbf{v}(2) \mathbf{v}(1)^T] & & & & E[\mathbf{v}(2) \mathbf{v}(n)^T] \\ \vdots & & \ddots & & \vdots \\ E[\mathbf{v}(n-1) \mathbf{v}(1)^T] & & & & E[\mathbf{v}(n-1) \mathbf{v}(n)^T] \\ E[\mathbf{v}(n) \mathbf{v}(1)^T] & E[\mathbf{v}(n) \mathbf{v}(2)^T] & \dots & E[\mathbf{v}(n) \mathbf{v}(n-1)^T] & E[\mathbf{v}(n) \mathbf{v}(n)^T] \end{array} \right] \\
 &= \begin{pmatrix} E[\mathbf{v}(1) \mathbf{v}(1)^T] & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & E[\mathbf{v}(n) \mathbf{v}(n)^T] \end{pmatrix} = \text{diag}(\mathbf{R}_{11}, \dots, \mathbf{R}_{nn})
 \end{aligned}$$

Also, these two noises are uncorrelated from the state vectors.

$$\begin{aligned}
 E[\boldsymbol{\omega}(k) \mathbf{X}(j)^T] &= 0 \\
 E[\mathbf{v}(k) \mathbf{X}(j)^T] &= 0
 \end{aligned} \tag{4.4}$$

The system model representation can also be described as the following block diagram.

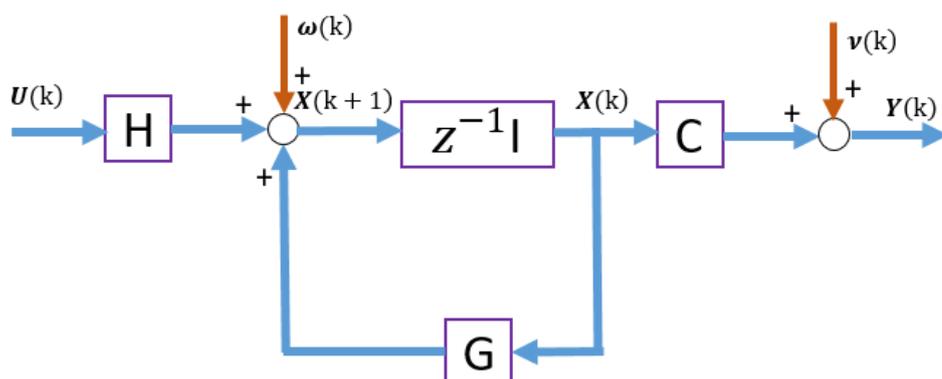


Figure 4.1 The system model block diagram with noise

In this block diagram, all the array signals are vectors, which include input vectors, state vectors, noise vectors and output vectors. All the arrowhead line directions represent the signal flow direction. The $z^{-1}I$ box is the n delay elements in parallel. The blocks G , H , C represent the respective matrices. $\omega(k)$ and $v(k)$ are the noises in the system, while they effect different parts of the system.

The Kalman filter algorithm is known as a linear quadratic estimate. Therefore, some definitions need to be declared before introduction of the Kalman filter algorithm.

The discrete-time system used in the Kalman filter is introduced before. The Kalman filter algorithm's purpose is to find the optimal estimate of the system state $X(k)$. The system state estimate is based on the system dynamics and the system output measurement with noise $Y(k)$. If the state estimate $X(k)$ is formed by using all the measurements up to and including time k , then this state estimate is called an posteriori estimate, which is defined as \hat{X}_k^+ . This state estimate can also be explained in another way, which uses the expectation value of $X(k)$ based on all of the measurements up to and including time k .

$$\hat{X}_k^+ = E[X(k)|Y(1), Y(2), \dots, Y(k)] = \text{a posteriori estimate} \quad (4.5)$$

If the state estimate of $X(k)$ is formed by using all the measurements up to but not including time k , then this state estimate is called an priori estimate, which is defined as \hat{X}_k^- . This state estimate can also be explained in another way, which uses the expected value of $X(k)$ based on all of the measurements up to, but not including, time k .

$$\hat{\mathbf{X}}_k^- = E[\mathbf{X}(k)|\mathbf{Y}(1), \mathbf{Y}(2), \dots, \mathbf{Y}(k-1)] = \text{a priori estimate} \quad (4.6)$$

Both of the state estimates $\hat{\mathbf{X}}_k^+$ and $\hat{\mathbf{X}}_k^-$ are Kalman filter estimates of the state $\mathbf{X}(k)$. However, the state estimate $\hat{\mathbf{X}}_k^-$ is computed before the current output measurement $\mathbf{Y}(k)$ is taken into account, and the state estimate $\hat{\mathbf{X}}_k^+$ is computed after the current output measurement $\mathbf{Y}(k)$ is taken into account. Therefore the posteriori estimate $\hat{\mathbf{X}}_k^+$ is expected to be more accurate than the priori estimate $\hat{\mathbf{X}}_k^-$.

Correspondingly, the term \mathbf{P}_k is used to donate the covariance of the state estimated error. \mathbf{P}_k^+ is the covariance of the state estimated error of $\hat{\mathbf{X}}_k^+$ and \mathbf{P}_k^- is the covariance of the state estimated error of $\hat{\mathbf{X}}_k^-$.

The covariance definition is that, for a vector $\mathbf{N} = (N_1, N_2, \dots, N_n)$, the covariance of the vector \mathbf{N} is:

$$\Sigma(\mathbf{N}) = \text{cov}(\mathbf{N}, \mathbf{N}) = E[(\mathbf{N} - \hat{\mathbf{N}})(\mathbf{N} - \hat{\mathbf{N}})^T] \quad (4.7)$$

Where $\hat{\mathbf{N}}$ is the estimated value of \mathbf{N} .

Hence, according to the definition of covariance, the covariance of the estimates are defined as following.

$$\begin{aligned} \mathbf{P}_k^+ &= E[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^+)(\mathbf{X}(k) - \hat{\mathbf{X}}_k^+)^T] \\ \mathbf{P}_k^- &= E[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)^T] \end{aligned} \quad (4.8)$$

As the Kalman filter required parameters are introduced, the Kalman filter algorithm can be operated under the conditioned of the information that was explained before

(1) The Kalman filter is based on the previous predicted state and the current measurement value. At time “0”, the Kalman filter doesn’t have any estimate, because the system’s initial condition is unknown. Thus, the initial step needs to set the initial estimate and only needs to be executed once. Assume the initial system state is $\mathbf{X}(0)$ and the initial system priori estimated state is $\hat{\mathbf{X}}_0^-$. Since there are no measurements for the estimated system state $\mathbf{X}(0)$, it is reasonable to set the posteriori estimate $\hat{\mathbf{X}}_0^+$ to be equal to priori estimate $\hat{\mathbf{X}}_0^-$, which is the expected value of initial system state $\mathbf{X}(0)$.

So the initial state estimate is the equation shown below:

$$\hat{\mathbf{X}}_0^+ = \hat{\mathbf{X}}_0^- = E[\mathbf{X}(0)] \quad (4.9)$$

The corresponding covariance \mathbf{P}_0^+ also assumes to be the given value based on the information of the system. For example, if the system's initial state estimate is totally the same as the system initial state, the covariance of the estimated error will be 0, $\mathbf{P}_0^+ = 0$. This is the ideal case and almost does not exist in the real world. If the system initial estimate is not the exact value (the estimate is not accurate), then \mathbf{P}_0^+ will be a symmetric matrix. Only the diagonal line parameters of the matrix have the corresponding covariance, which describe the difference from the actual situation and the estimated value. In general, \mathbf{P}_0^+ represents the uncertainty of the initial estimate $\mathbf{X}(0)$.

$$\mathbf{P}_0^+ = E[(\mathbf{X}(0) - \widehat{\mathbf{X}}_0^+)(\mathbf{X}(0) - \widehat{\mathbf{X}}_0^+)^T], \text{ which is given.} \quad (4.10)$$

(2) The measurement noise $\mathbf{v}(k)$ and the process noise $\boldsymbol{\omega}(k)$ are assumed to be unknown, the estimate of the next time can only derived from the equation (4.1). This means the noises in process and measurement are not taken into consideration in the posteriori estimation. Therefore, the state estimate $\widehat{\mathbf{X}}(k)$ can be calculated by the state function without noise.

$$\widehat{\mathbf{X}}(k) = \mathbf{G}\widehat{\mathbf{X}}(k-1) + \mathbf{H}\mathbf{U}(k-1) \quad (4.11)$$

In the Kalman filter algorithm, there are two estimates mentioned before, the state estimate $\widehat{\mathbf{X}}_k^-$ is computed before the current output measurement $\mathbf{Y}(k)$ is taken into account, and the state estimate $\widehat{\mathbf{X}}_k^+$ is computed after the current output measurement $\mathbf{Y}(k)$ is taken into account. The posteriori state estimate $\widehat{\mathbf{X}}_k^+$ is expected to be more accurate than the priori estimate $\widehat{\mathbf{X}}_k^-$. Therefore, the prior state estimate $\widehat{\mathbf{X}}_k^+$ can be computed from the previous time posterior state estimate $\widehat{\mathbf{X}}_{k-1}^+$.

$$\widehat{\mathbf{X}}_k^- = \mathbf{G}\widehat{\mathbf{X}}_{k-1}^+ + \mathbf{H}\mathbf{U}(k-1) \quad (4.12)$$

This is the general equation that shows how to obtain $\widehat{\mathbf{X}}_k^-$ from $\widehat{\mathbf{X}}_{k-1}^+$. The initial state estimate $\widehat{\mathbf{X}}_0^+$ is already given in step (1). Hence, from the equation (4.12), the priori estimate at time 1 can be derived.

$$\widehat{\mathbf{X}}_1^- = \mathbf{G}\widehat{\mathbf{X}}_0^+ + \mathbf{H}\mathbf{U}(0) \quad (4.13)$$

(3) The priori state estimate $\widehat{\mathbf{X}}_k^-$ is already computed in the equation (4.12). Accordingly, its corresponding covariance \mathbf{P}_k^- of state estimate error $\widehat{\mathbf{X}}_k^-$ can be derived from the equation (4.8). The real system equation (20) is also needed by going back a step. The previous step real system state equation is shown below.

$$\mathbf{X}(k) = \mathbf{G}\mathbf{X}(k-1) + \mathbf{H}\mathbf{U}(k-1) + \boldsymbol{\omega}(k-1) \quad (4.14)$$

Sub the equation (4.12) and the equation (4.14) into the equation (4.8).

$$\begin{aligned}
 \mathbf{P}_k^- &= E[(\mathbf{X}(k) - \widehat{\mathbf{X}}_k^-)(\mathbf{X}(k) - \widehat{\mathbf{X}}_k^-)^T] \\
 &= E[(\mathbf{G}\mathbf{X}(k-1) + \mathbf{H}\mathbf{U}(k-1) + \boldsymbol{\omega}(k-1) - \mathbf{G}\widehat{\mathbf{X}}_{k-1}^+ - \mathbf{H}\mathbf{U}(k-1))(\mathbf{G}\mathbf{X}(k-1) + \mathbf{H}\mathbf{U}(k-1) + \boldsymbol{\omega}(k-1) - \mathbf{G}\widehat{\mathbf{X}}_{k-1}^+ - \mathbf{H}\mathbf{U}(k-1))^T] \\
 &= E[(\mathbf{G}\mathbf{X}(k-1) + \boldsymbol{\omega}(k-1) - \mathbf{G}\widehat{\mathbf{X}}_{k-1}^+)(\mathbf{G}\mathbf{X}(k-1) + \boldsymbol{\omega}(k-1) - \mathbf{G}\widehat{\mathbf{X}}_{k-1}^+)^T] \\
 &= E[(\mathbf{G}(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+) + \boldsymbol{\omega}(k-1))(\mathbf{G}(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+) + \boldsymbol{\omega}(k-1))^T] \\
 &= \mathbf{G}E[(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)^T]\mathbf{G}^T \\
 &\quad - \mathbf{G}E[(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)\boldsymbol{\omega}(k-1)^T] \\
 &\quad - E[\boldsymbol{\omega}(k-1)(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)^T]\mathbf{G}^T \\
 &\quad + E[\boldsymbol{\omega}(k-1)\boldsymbol{\omega}(k-1)^T]
 \end{aligned} \tag{4.15}$$

The estimated error $(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)$ at time $(k-1)$ is independent of the process noise $\boldsymbol{\omega}(k-1)$. Therefore the covariance of estimated error and process noise is equal to 0.

$$E[\boldsymbol{\omega}(k-1)(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)^T] = E[(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)\boldsymbol{\omega}(k-1)^T] = 0 \tag{4.16}$$

Sub the equation (4.16) into the equation (4.15).

$$\begin{aligned}
 \mathbf{P}_k^- &= \mathbf{G}E[(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)(\mathbf{X}(k-1) - \widehat{\mathbf{X}}_{k-1}^+)^T]\mathbf{G}^T \\
 &\quad + E[\boldsymbol{\omega}(k-1)\boldsymbol{\omega}(k-1)^T]
 \end{aligned} \tag{4.17}$$

Recall the equation (4.3) and the equation (4.8), take a step back and sub them into the equation (4.17).

$$\mathbf{P}_k^- = \mathbf{G}\mathbf{P}_{k-1}^+\mathbf{G}^T + \mathbf{Q} \tag{4.18}$$

The equation (4.18) is the general equation of state estimated error covariance \mathbf{P}_k^- . At time 1, the state estimated error covariance is shown as follows.

$$\mathbf{P}_1^- = \mathbf{G}\mathbf{P}_0^+\mathbf{G}^T + \mathbf{Q} \tag{4.19}$$

(4) As mentioned before, the Kalman filter estimate is a quadratic estimate algorithm. The posteriori estimate $\widehat{\mathbf{X}}_k^+$ is calculated based on the measurement $\mathbf{Y}(k)$ on time K. Assume the

Kalman filter gain matrix is defined as K_k , the posteriori estimate $\hat{\mathbf{X}}_k^+$ can be calculated from the priori estimate $\hat{\mathbf{X}}_k^-$.

$$\hat{\mathbf{X}}_k^+ = \hat{\mathbf{X}}_k^- + K_k(\mathbf{Y}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-) \quad (4.20)$$

The term $(\mathbf{Y}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-)$ is the error between the measurement $\mathbf{Y}(k)$ and vector $\mathbf{C}\hat{\mathbf{X}}_k^-$, which is called the correction term.

To finish one sequence of the Kalman filter algorithm, the posteriori estimate is shown as follows at time 1.

$$\hat{\mathbf{X}}_1^+ = \hat{\mathbf{X}}_1^- + K_1(\mathbf{Y}(1) - \mathbf{C}\hat{\mathbf{X}}_1^-) \quad (4.21)$$

(5) The reason why the Kalman filter has the optimal estimate of states in the linear system is that the Kalman filter can estimate the state with the smallest estimate error covariance in total. Therefore, to calculate the Kalman filter gain K_k , the sum of the variances of the estimated errors at time k, which is cost function J_k , should be minimized.

$$\begin{aligned} J_k &= E \left[(\mathbf{X}(1) - \hat{\mathbf{X}}_1^+)^2 \right] + \dots + E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^+)^2 \right] \\ &= Tr(\mathbf{P}_k^+) \\ &= Tr \left(E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^+)(\mathbf{X}(k) - \hat{\mathbf{X}}_k^+)^T \right] \right) \end{aligned} \quad (4.22)$$

Sub the equation (4.20) into the equation (4.22).

$$\begin{aligned} J_k &= Tr \left(E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^- + K_k(\mathbf{Y}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-))(\mathbf{X}(k) - \hat{\mathbf{X}}_k^- + K_k(\mathbf{Y}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-))^T \right] \right) \end{aligned} \quad (4.23)$$

Sub the equation (4.1) into the equation (4.23).

$$\begin{aligned} J_k &= Tr \left(E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^- + K_k(\mathbf{C}\mathbf{X}(k) + \mathbf{v}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-))(\mathbf{X}(k) - \hat{\mathbf{X}}_k^- + K_k(\mathbf{C}\mathbf{X}(k) + \mathbf{v}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-))^T \right] \right) \\ &= Tr \left(E \left[((\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) + K_k\mathbf{C}(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) + K_k\mathbf{v}(k))((\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) + K_k\mathbf{C}(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) + K_k\mathbf{v}(k))^T \right] \right) \\ &= Tr \left(E \left[((\mathbf{I} - K_k\mathbf{C})(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) + K_k\mathbf{v}(k))((\mathbf{I} - K_k\mathbf{C})(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) + K_k\mathbf{v}(k))^T \right] \right) \end{aligned} \quad (4.24)$$

$$\begin{aligned}
 &= Tr \left((I - K_k \mathbf{C}) E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)^T \right] (I - K_k \mathbf{C})^T \right. \\
 &\quad + (I - K_k \mathbf{C}) E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) \mathbf{v}(k)^T \right] \\
 &\quad + E \left[\mathbf{v}(k) (\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)^T \right] (I - K_k \mathbf{C})^T \\
 &\quad \left. + K_k E \left[\mathbf{v}(k) \mathbf{v}(k)^T K_k^T \right] \right)
 \end{aligned}$$

The estimated error $(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)$ is independent of the measurement noise $\mathbf{v}(k)$.

$$E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) \mathbf{v}(k)^T \right] = E \left[\mathbf{v}(k) (\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)^T \right] = 0 \quad (4.25)$$

Sub the equation (4.25) into the equation (4.24)

$$\begin{aligned}
 J_k &= Tr \left((I - K_k \mathbf{C}) E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)(\mathbf{X}(k) - \hat{\mathbf{X}}_k^-)^T \right] (I - K_k \mathbf{C})^T \right. \\
 &\quad \left. + K_k E \left[\mathbf{v}(k) \mathbf{v}(k)^T K_k^T \right] \right)
 \end{aligned} \quad (4.26)$$

Sub the equation (4.3) and the equation (4.8) into the equation (4.26).

$$J_k = Tr \left((I - K_k \mathbf{C}) \mathbf{P}_k^- (I - K_k \mathbf{C})^T + K_k \mathbf{R} K_k^T \right) \quad (4.27)$$

The optimal Kalman filter gain K_k needs to be found to make the cost function J_k as small as possible. Here, the optimal Kalman filter gain K_k will change the cost function derivative equal to zero. The function to choose the optimal Kalman filter gain K_k is shown below.

$$\frac{\partial J_k}{\partial K_k} = \frac{\partial Tr \left((I - K_k \mathbf{C}) \mathbf{P}_k^- (I - K_k \mathbf{C})^T + K_k \mathbf{R} K_k^T \right)}{\partial K_k} = 0 \quad (4.28)$$

There is a matrix calculus that is used to simplify the equation (4.28), which is introduced as the following. [30]

$$\frac{\partial Tr(ABA^T)}{\partial A} = 2AB \quad \text{if } B = B^T \quad (4.29)$$

So the equation (4.28) can be transformed into the equation (4.30).

$$\begin{aligned}
 & \frac{\partial \text{Tr} \left((\mathbf{I} - K_k \mathbf{C}) \mathbf{P}_k^- (\mathbf{I} - K_k \mathbf{C})^T + K_k \mathbf{R} K_k^T \right)}{\partial K_k} \\
 &= \frac{\partial \text{Tr} \left((\mathbf{I} - K_k \mathbf{C}) \mathbf{P}_k^- (\mathbf{I} - K_k \mathbf{C})^T \right) + \partial \text{Tr} K_k \mathbf{R} K_k^T}{\partial K_k} \\
 &= \frac{\partial \text{Tr} \left((\mathbf{I} - K_k \mathbf{C}) \mathbf{P}_k^- (\mathbf{I} - K_k \mathbf{C})^T \right)}{\partial (\mathbf{I} - K_k \mathbf{C})} * \frac{\partial (\mathbf{I} - K_k \mathbf{C})}{\partial K_k} \\
 &+ \frac{\partial \text{Tr} K_k \mathbf{R} K_k^T}{\partial K_k} = 2(\mathbf{I} - K_k \mathbf{C}) \mathbf{P}_k^- (-\mathbf{C}^T) + 2K_k \mathbf{R} = 0 \\
 & \quad K_k \mathbf{R} = (\mathbf{I} - K_k \mathbf{C}) \mathbf{P}_k^- \mathbf{C}^T \\
 & \quad K_k (\mathbf{R} + \mathbf{C} \mathbf{P}_k^- \mathbf{C}^T) = \mathbf{P}_k^- \mathbf{C}^T \\
 & \quad K_k = \mathbf{P}_k^- \mathbf{C}^T (\mathbf{C} \mathbf{P}_k^- \mathbf{C}^T + \mathbf{R})^{-1}
 \end{aligned} \tag{4.30}$$

At time 1, the Kalman filter gain is shown as the following.

$$K_1 = \mathbf{P}_1^- \mathbf{C}^T (\mathbf{C} \mathbf{P}_1^- \mathbf{C}^T + \mathbf{R})^{-1} \tag{4.31}$$

(6) As the posteriori estimate $\hat{\mathbf{X}}_k^+$ is updated by the Kalman filter gain K_k , its corresponding estimate error covariance is derived from the posterior estimate $\hat{\mathbf{X}}_k^+$ function.

$$\begin{aligned}
 & \mathbf{P}_k^+ = E \left[(\mathbf{X}(k) - \hat{\mathbf{X}}_k^+) (\mathbf{X}(k) - \hat{\mathbf{X}}_k^+)^T \right] \\
 &= E \left[\left(\mathbf{X}(k) - \hat{\mathbf{X}}_k^- - K_k (\mathbf{Y}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) \right) \left(\mathbf{X}(k) - \hat{\mathbf{X}}_k^- - K_k (\mathbf{Y}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) \right)^T \right] \\
 &= E \left[\left((\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) - K_k (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) \right. \right. \\
 & \quad \left. \left. + K_k \mathbf{v}(k) \right) \left((\mathbf{X}(k) - \hat{\mathbf{X}}_k^-) - K_k (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) + K_k \mathbf{v}(k) \right)^T \right] \\
 &= E \left[\left((\mathbf{I} - K_k \mathbf{C}) (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) \right. \right. \\
 & \quad \left. \left. + K_k \mathbf{v}(k) \right) \left((\mathbf{I} - K_k \mathbf{C}) (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) + K_k \mathbf{v}(k) \right)^T \right] \\
 &= E \left[(\mathbf{I} - K_k \mathbf{C}) (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-)^T (\mathbf{I} - K_k \mathbf{C})^T \right. \\
 & \quad \left. + (\mathbf{I} - K_k \mathbf{C}) (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-) \mathbf{v}(k)^T K_k^T \right. \\
 & \quad \left. + K_k \mathbf{v}(k) (\mathbf{C} \mathbf{X}(k) - \mathbf{C} \hat{\mathbf{X}}_k^-)^T (\mathbf{I} - K_k \mathbf{C})^T + K_k \mathbf{v}(k) \mathbf{v}(k)^T K_k^T \right]
 \end{aligned} \tag{4.32}$$

Recall the equation (4.25) and sub it into the equation (4.32), the covariance of the estimate error can be simplified as below.

$$\begin{aligned}
 \mathbf{P}_k^+ &= (\mathbf{I} - K_k \mathbf{C}) E[(\mathbf{C}\mathbf{X}(k) - \mathbf{C}\widehat{\mathbf{X}}_k^-)(\mathbf{C}\mathbf{X}(k) - \mathbf{C}\widehat{\mathbf{X}}_k^-)^T] (\mathbf{I} - K_k \mathbf{C})^T \\
 &\quad + K_k E[\mathbf{v}(k)\mathbf{v}(k)^T] K_k^T \\
 &= (\mathbf{I} - K_k \mathbf{C}) \mathbf{P}_k^- (\mathbf{I} - K_k \mathbf{C})^T + K_k \mathbf{R} K_k^T
 \end{aligned} \tag{4.33}$$

At time 1, the Kalman filter gain is shown as the following.

$$\mathbf{P}_1^+ = (\mathbf{I} - K_1 \mathbf{C}) \mathbf{P}_1^- (\mathbf{I} - K_1 \mathbf{C})^T + K_1 \mathbf{R} K_1^T \tag{4.34}$$

As the examples at time 1 shown in each part, from the initial estimate $\widehat{\mathbf{X}}_0^+$ and its corresponding covariance of estimated error \mathbf{P}_0^+ , the posteriori estimate $\widehat{\mathbf{X}}_1^+$ and its corresponding covariance of estimated error \mathbf{P}_1^+ have been derived. These examples show the process of the Kalman filter, and can be used to predict the next time estimate.

4.2 The discrete-time Kalman filter conclusion

The previous steps show the procedure of the Kalman filter in several algorithms. However, only the five equations in the Kalman filter are needed to predict the state and correct the state. Therefore, a conclusion of the Kalman filter is drawn here by combining the above equations into a single algorithm.

1. The discrete-time system with the process and measurement noise is shown by the following equations.

$$\begin{aligned}
 \mathbf{X}(k+1) &= \mathbf{G}\mathbf{X}(k) + \mathbf{H}\mathbf{U}(k) + \boldsymbol{\omega}(k) \\
 \mathbf{Y}(k) &= \mathbf{C}\mathbf{X}(k) + \mathbf{v}(k) \\
 E[\boldsymbol{\omega}(k) \boldsymbol{\omega}(j)^T] &= \begin{cases} \mathbf{Q}, & k = j \\ 0, & k \neq j \end{cases} \\
 E[\mathbf{v}(k) \mathbf{v}(j)^T] &= \begin{cases} \mathbf{R}, & k = j \\ 0, & k \neq j \end{cases} \\
 E[\boldsymbol{\omega}(k) \mathbf{v}(j)^T] &= 0
 \end{aligned} \tag{4.35}$$

2. The initialization of the Kalman filter is artificially set from the basic information of the researched system.

$$\begin{aligned}
 \widehat{\mathbf{X}}_0^+ &= E[\mathbf{X}(0)] \\
 \mathbf{P}_0^+ &= E[(\mathbf{X}(0) - \widehat{\mathbf{X}}_0^+)(\mathbf{X}(0) - \widehat{\mathbf{X}}_0^+)^T]
 \end{aligned} \tag{4.36}$$

3. The Kalman filter algorithm is calculated using the following equations, which are used for each time step $k = 1, 2, 3 \dots$

$$\begin{aligned}
 \mathbf{P}_k^- &= \mathbf{G}\mathbf{P}_{k-1}^+\mathbf{G}^T + \mathbf{Q} \\
 \hat{\mathbf{X}}_k^- &= \mathbf{G}\hat{\mathbf{X}}_{k-1}^+ + \mathbf{H}\mathbf{U}(k-1) \\
 \hat{\mathbf{X}}_k^+ &= \hat{\mathbf{X}}_k^- + \mathbf{K}_k(\mathbf{Y}(k) - \mathbf{C}\hat{\mathbf{X}}_k^-) \\
 \mathbf{K}_k &= \mathbf{P}_k^-\mathbf{C}^T(\mathbf{C}\mathbf{P}_k^-\mathbf{C}^T + \mathbf{R})^{-1} \\
 \mathbf{P}_k^+ &= (\mathbf{I} - \mathbf{K}_k\mathbf{C})\mathbf{P}_k^-(\mathbf{I} - \mathbf{K}_k\mathbf{C})^T + \mathbf{K}_k\mathbf{R}\mathbf{K}_k^T
 \end{aligned} \tag{4.37}$$

The Kalman filter algorithm is also given in the following block diagram, which describes the equations relationship that were discussed before.

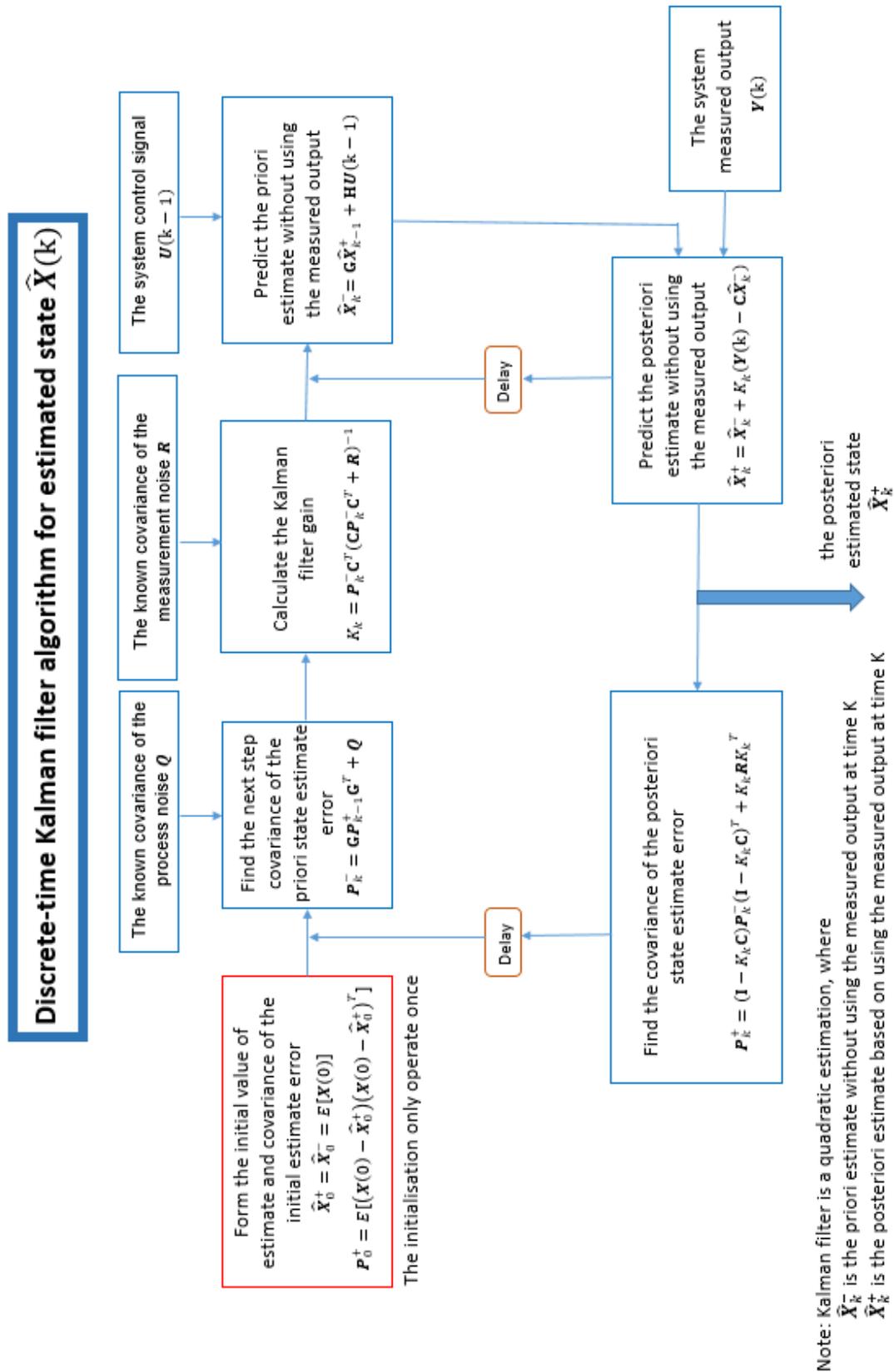


Figure 4.2 The discrete-time Kalman filter algorithm block diagram

4.3 An application of the Kalman filter

A simple application of the Kalman filter and its simulation is provided to verify the Kalman filter algorithm. As can be seen in Figure 4.3, a person is driving a car on a straight road. Only the position can be measured per 1 second in this case. By knowing the acceleration, the Kalman filter can be used to estimate its velocity.

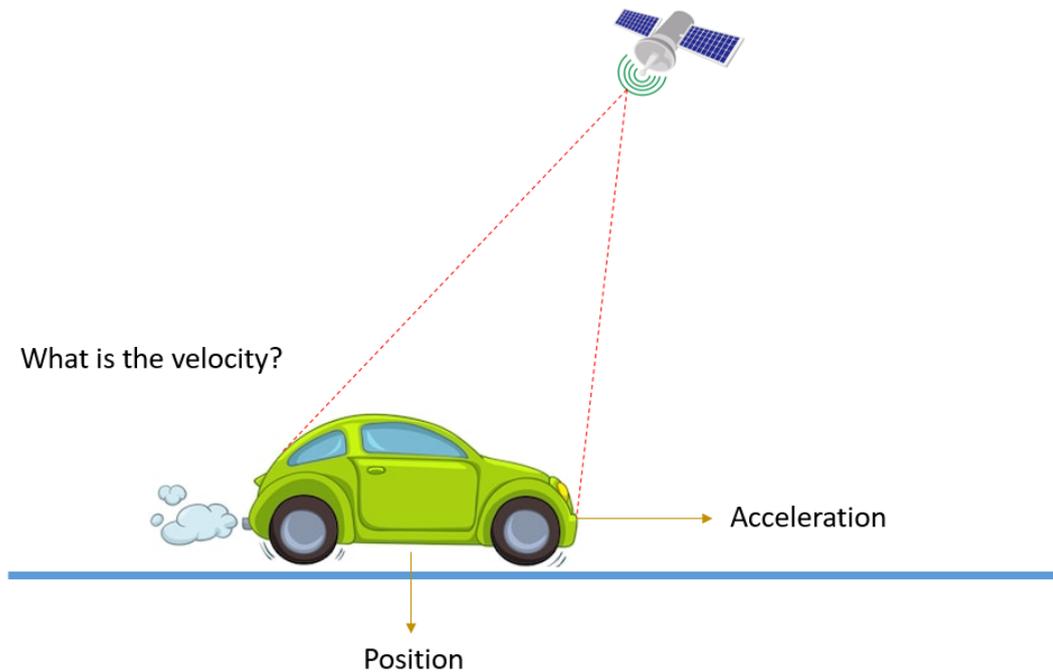


Figure 4.3 The Kalman filter application system

According to the Kinematic equations, the system can be derived as follows:

$$s_k = s_{k-1} + v_{k-1}t + \frac{1}{2}a_{k-1}t^2 \quad (4.38)$$

$$v_k = v_{k-1} + a_{k-1}t$$

Where s_k is the position at time k , v_k is the velocity at time k , a_{k-1} is acceleration at time $k - 1$, and t is time.

The system SSR can be obtained as:

$$\begin{bmatrix} s_k \\ v_k \end{bmatrix} = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_{k-1} \\ v_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{t^2}{2} \\ t \end{bmatrix} a_{k-1} \quad (4.39)$$

$$Y_k = [1 \quad 0] \begin{bmatrix} s_k \\ v_k \end{bmatrix}$$

Where Y_k is the measurement at time k .

If the system model is built in SSR block of MATLAB Simulink, it will only show the system input and system output. In this case, in order to compare the system state with the estimated state, the block diagram is drawn to simulate the system model in Figure 4.4.

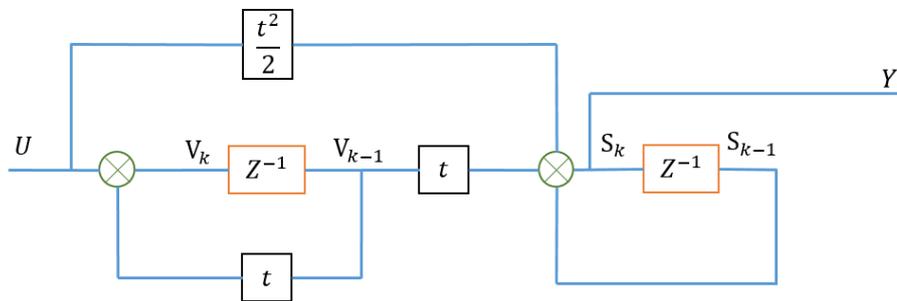


Figure 4.4 The block diagram of the Kalman filter application system

In the block diagram, the system state parameters are clearly shown. Therefore, each state parameter can be taken out as needed. According to the system block diagram, the system is built in the MATLAB Simulink in Figure 4.5.

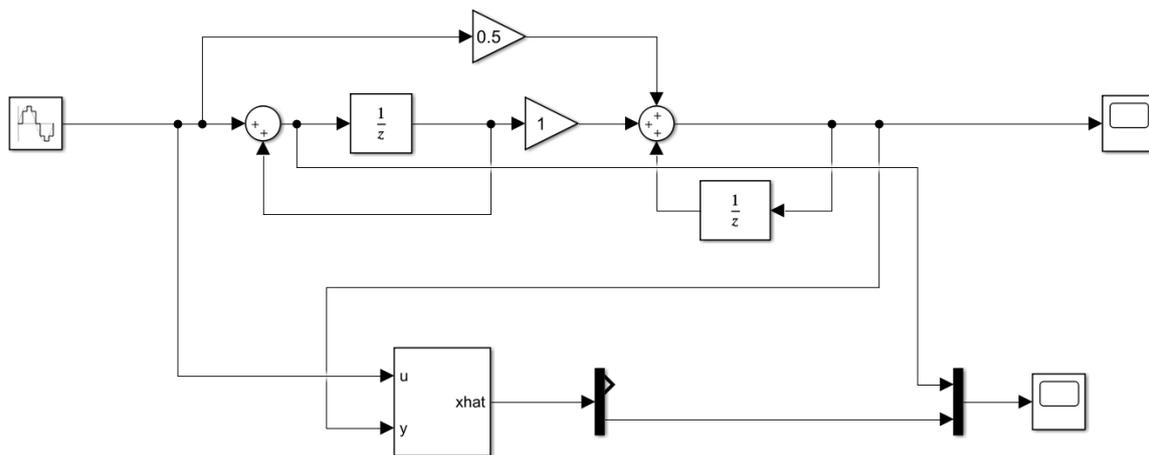


Figure 4.5 The MATLAB Simulink for the Kalman filter application system

The Kalman filter uses the input and output of the system to estimate the system state. The acceleration in this case is set to be a discrete-time sine wave with 100 Hz/s frequency and 1second sampling time. The system state velocity is taken to compare with the estimated state velocity in scope 2. For the initial state estimate of the system, the position and the velocity are assumed to be zero.

$$\begin{bmatrix} s_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Assume the covariance of the initial state estimated error to be:

$$P_0^+ = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

Assume the covariance of the process noise to be:

$$Q = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

Assume the covariance of the measurement noise to be:

$$R = [0]$$

The compared result is given in Figure 4.6.

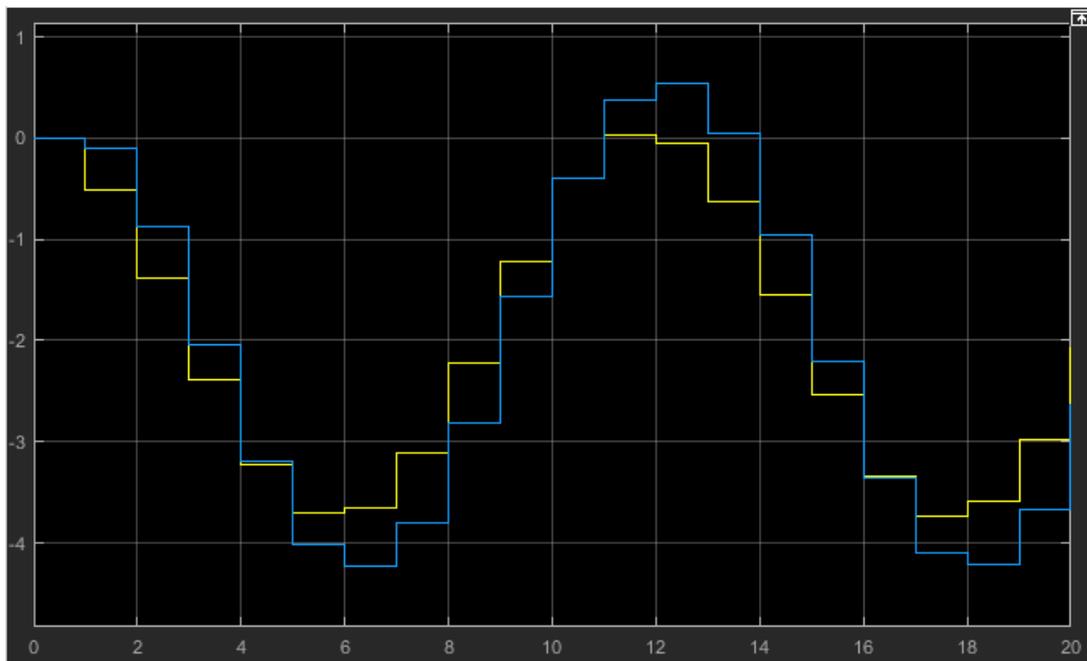


Figure 4.6 The compared result between the real system state and the estimated state

As shown in the figure, the yellow curve is the real system state while the blue curve is the estimated state. For the sine wave velocity, a certain difference will exist between the real system state and the estimated state when the velocity trend is changing. However, it is clearly shown that the estimated velocity is tracking the real velocity all the time. This application successfully indicates that the Kalman filter can produce the optimal estimate of the linear system.

4.4 The FDD method by using the Kalman filter

The system fault classifications are discussed before, which are sensor fault, actuator fault, and system component fault. In the plate structure system, the researched fault object is actuator faults. Once the actuator in the plate system is broken, the control signal cannot control the top plate to be stable. This could cause the control accident in the plate structure. Without the three control inputs into the system, the top plate will have strenuous vibration that will break the entire system.

4.4.1 The classifications of system actuator faults

As the plate structure system introduced before, there are three actuators in the system. Figure 4.7 shows their relationships in the system.

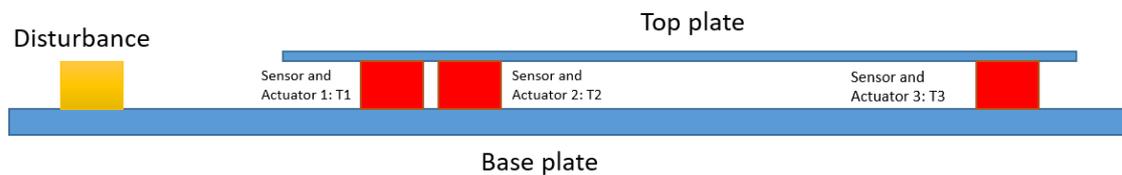


Figure 4.7 The system actuators position relationships

The classification of actuator faults can be defined as two kinds of faults. One of them is signal fault, which can be actuator 1 fault, actuator 2 fault, or actuator 3 fault. Another kind of fault is a double fault, which can be actuator 1, 2 fault, actuator 2, 3 fault, or actuator 1, 3 fault. The FDD purpose is to detect the random fault, and supply information to solve problems (to interrupt system or change the control law). If the three actuators have faults together, the whole system will stop working and will be uncontrollable. Therefore, detecting faults happening to the three actuators at the same time is unnecessary in the plate structure system. Figure 4.8 shows the classification of actuator faults.

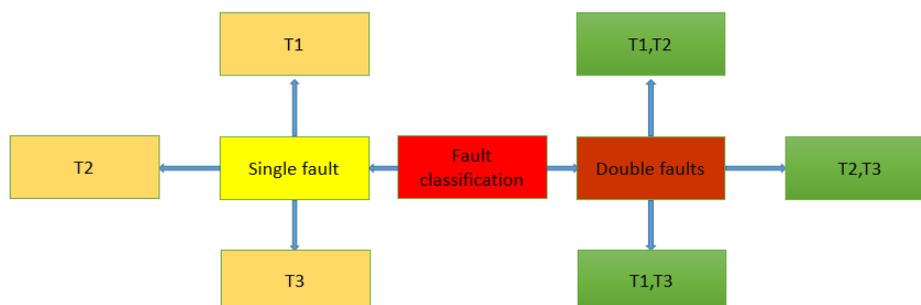


Figure 4.8 The classification of actuator faults

4.4.2 Fault detection method

Figure 4.9 gives a simplified Kalman filter algorithm block diagram. From the Kalman filter algorithm, it is known that the Kalman filter can compute the system's estimated state by using the real time system output measurement and system control input.

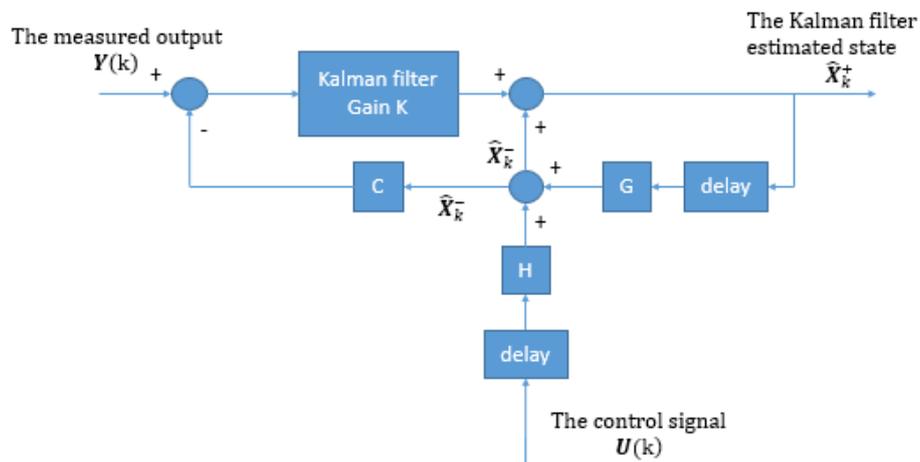


Figure 4.9 The simplified Kalman filter algorithm

When the actuator fault occurs in the system, the alterations in the system are the state and output measurement. According to the system measurement function without measurement noise, the system state estimate is transformed to the system output estimate.

$$\hat{Y}(k) = \mathbf{C}\hat{X}(k) \quad (4.40)$$

Where $\hat{Y}(k)$ is the estimated output, and the state estimate $\hat{X}(k)$ is the Kalman filter posteriori state estimate \hat{X}_k^+ .

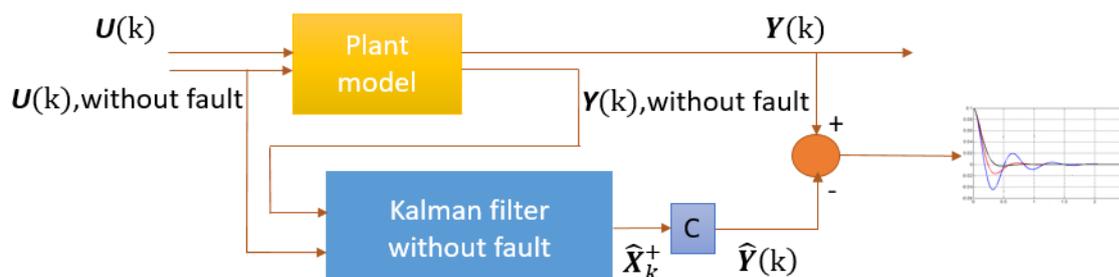


Figure 4.10 The fault detection method block diagram

The fault detection method is described in Figure 4.10. As shown in the figure, the Kalman filter used to detect faults is called the Kalman filter without fault. This Kalman filter uses the input signal and output signal without fault to produce a no-fault state estimate. A residual can be calculated by comparing the estimated output from the Kalman filter and measured output. From the Kalman filter algorithm design, the Kalman filter state estimate is the optimal estimate in linear system. Therefore, if the system does not have fault, the Kalman filter using the matrixes in the system will compute the optimal estimated output, which is almost the same as the measured output. In this situation, the residual is named no-fault residual, which will be a zero-mean noise with a very small value. If the system has fault, the residual is named fault residual that will be a very large zero-mean noise. Assume the residual is $E_{re}(k)$, its transfer function is:

$$E_{re}(k) = Y(k) - \hat{Y}(k) = Y(k) - C\hat{X}_k^+ \quad (4.41)$$

$$\begin{cases} \forall E_{re}(k) \leq T_{threshold}, \text{there is no fault in the system.} \\ \exists E_{re}(k) > T_{threshold}, \text{the system has unknown fault.} \end{cases}$$

Where $T_{threshold}$ is the threshold value, which is chosen between the upper limit of the no-fault residual and the upper limit of the fault residual.

In the block diagram, the Kalman filter without fault block is the Kalman filter using the system matrix G , the input matrix H , and the output matrix C in no-fault system. This is the essential point to get the almost equivalent estimated output and measured output.

This method can be used to detect any kind of fault in the system not only the actuator fault. If fault occurs to the system, the system model will change and influence the system output. However this method cannot be used to diagnose the fault, another fault diagnosis method is presented to complete the FDD function.

4.4.3 Fault diagnosis method

The fault diagnosis purpose is to find the accurate fault in this thesis. It is important to find the difference between different actuator faults. In the system modelling part, it is introduced that the control singles $U_1(s)$, $U_2(s)$, and $U_3(s)$ are executed by the actuators. The input matrix H and the control inputs $U_1(s)$, $U_2(s)$, and $U_3(s)$ construct the control impact to the system. Assume the control impact is $I_{con}(s)$, its transfer function of the plate structure is shown below.

$$\begin{aligned}
 I_{con}(s) = \mathbf{H}\mathbf{U}(s) &= \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} & \mathbf{H}_{13} \\ \mathbf{H}_{21} & \mathbf{H}_{22} & \mathbf{H}_{23} \\ \mathbf{H}_{31} & \mathbf{H}_{32} & \mathbf{H}_{33} \\ \mathbf{H}_{41} & \mathbf{H}_{42} & \mathbf{H}_{43} \\ \mathbf{H}_{51} & \mathbf{H}_{52} & \mathbf{H}_{53} \\ \mathbf{H}_{61} & \mathbf{H}_{62} & \mathbf{H}_{63} \end{bmatrix} \times \begin{bmatrix} \mathbf{U}_1(s) \\ \mathbf{U}_2(s) \\ \mathbf{U}_3(s) \end{bmatrix} \\
 &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\
 &+ (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\
 &+ (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s)
 \end{aligned} \tag{4.42}$$

This control impact transfer function is the essential part to apply the Kalman filter in fault diagnosis and shows the relationship of the input signal and input matrix in multiple input situations. All the fault situations discussed later will be based on the concept of control impact.

1. Single fault on actuator 1

The single fault situation is a very common situation in system fault. A way to detect single fault on actuator 1 is introduced here. When a fault occurs to actuator 1, the control signal $\mathbf{U}_1(s)$ cannot be executed to control the system. Therefore, the control signal $\mathbf{U}_1(s)$ is equal to 0 in the actuator 1 fault situation. The control impact of the whole system is changed from the no-fault situation to the actuator 1 fault situation.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} \\
 &+ (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\
 &+ (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s) \\
 &= (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\
 &+ (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s)
 \end{aligned} \tag{4.43}$$

From the plate structure state representation, the control impact of the actuator 1 fault situation influences the output of the system. To diagnose the fault, another Kalman filter called the Kalman filter with fault 1 is built to estimate the actuator 1 fault system output. The system matrix and the output matrix is the same comparing the actuator 1 system and the no-fault system. The input matrix needs to change for the actuator 1 system output estimation. The new input matrix should achieve the same control impact of the actuator 1 fault system regardless of the input signal. Therefore the new control impact is:

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s) \quad (4.44) \\
 &= (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s)
 \end{aligned}$$

From the control impact equation, a way of getting the same control impact is to set the sum of parameters in the first row of the no-fault system input matrix to be 0 and make other parameters in the first row of the no-fault system input matrix remain the same.

$$\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61} = 0 \quad (4.45)$$

In the input matrix of the Kalman filter with fault 1, the parameters in the first row are set to be 0 and the other parameters remain the same as the no-fault system input matrix.

$$\mathbf{H}_{f1} = \begin{bmatrix} 0 & \mathbf{H}_{12} & \mathbf{H}_{13} \\ 0 & \mathbf{H}_{22} & \mathbf{H}_{23} \\ 0 & \mathbf{H}_{32} & \mathbf{H}_{33} \\ 0 & \mathbf{H}_{42} & \mathbf{H}_{43} \\ 0 & \mathbf{H}_{52} & \mathbf{H}_{53} \\ 0 & \mathbf{H}_{62} & \mathbf{H}_{63} \end{bmatrix}$$

Where \mathbf{H}_{f1} is the input matrix of actuator fault 1 system.

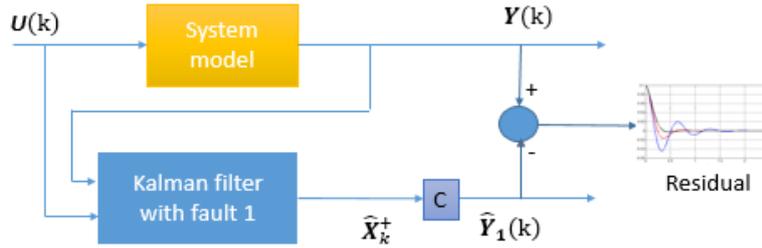


Figure 4.11 The fault diagnosis of the actuator fault 1 method block diagram

The fault diagnosis of the actuator fault 1 method is illustrated in Figure 4.11. A residual can be calculated by comparing the estimated output from the Kalman filter with fault 1 and the measured output. The residual is named the fault 1 residual, which will be a zero-mean noise with a very small value when the system has fault on actuator 1. If the system does not have fault on actuator 1, the residual is named no-fault 1 residual that will be a very large zero-mean noise. Assume the residual is $\mathbf{E}_{1re}(k)$, its transfer function is:

$$\mathbf{E}_{1re}(k) = \mathbf{Y}(k) - \hat{\mathbf{Y}}_1(k) \quad (4.46)$$

$$\begin{cases} \forall E_{1re}(k) \leq \mathbf{T}_{threshold}, \text{there is fault on actuator 1.} \\ \exists E_{1re}(k) > \mathbf{T}_{threshold}, \text{there is no fault on actuator 1} \end{cases}$$

Where $\mathbf{T}_{threshold}$ is the threshold value, which is chosen between the upper limit of the fault 1 residual and the upper limit of the no-fault 1 residual, and $\hat{\mathbf{Y}}_1(k)$ is the estimated output of the Kalman filter with fault 1.

2. Single fault on actuator 2

A way to detect single fault on actuator 2 is introduced here. When a fault occurs to actuator 2, the control signal $\mathbf{U}_2(s)$ cannot be executed to control the system. Therefore, the control signal $\mathbf{U}_2(s)$ is equal to 0 in the actuator 2 fault situation. The control impact of the whole system is changed from the no-fault situation to the actuator 2 fault situation.

$$\begin{aligned} I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\ &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} \\ &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s) \quad (4.47) \\ &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\ &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s) \end{aligned}$$

From the plate structure state representation, the control impact of the actuator 2 fault situation influences the output of the system. To diagnose the fault, another Kalman filter called Kalman filter with fault 2 is built to estimate the actuator 2 fault system output. The system matrix and the output matrix is the same comparing the actuator 2 system and the no-fault system. The input matrix needs to change for the actuator 2 system output estimation. The new input matrix should achieve the same control impact of actuator 2 fault system regardless of the input signal. Therefore the new control impact is:

$$\begin{aligned} I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\ &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\ &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s) \quad (4.48) \\ &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\ &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{U}_3(s) \end{aligned}$$

From the control impact equation, a way of getting the same control impact is to set the sum of parameters in the second row of the no-fault system input matrix to be 0 and make other parameters in the no-fault system input matrix remain the same.

$$\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62} = 0 \quad (4.49)$$

In the input matrix of the Kalman filter with fault 2, the parameters in the second row are set to be 0 and other parameters remain the same in the no-fault system input matrix.

$$\mathbf{H}_{f2} = \begin{bmatrix} \mathbf{H}_{11} & \mathbf{0} & \mathbf{H}_{13} \\ \mathbf{H}_{21} & \mathbf{0} & \mathbf{H}_{23} \\ \mathbf{H}_{31} & \mathbf{0} & \mathbf{H}_{33} \\ \mathbf{H}_{41} & \mathbf{0} & \mathbf{H}_{43} \\ \mathbf{H}_{51} & \mathbf{0} & \mathbf{H}_{53} \\ \mathbf{H}_{61} & \mathbf{0} & \mathbf{H}_{63} \end{bmatrix}$$

Where \mathbf{H}_{f2} is the input matrix of actuator fault 2 system.

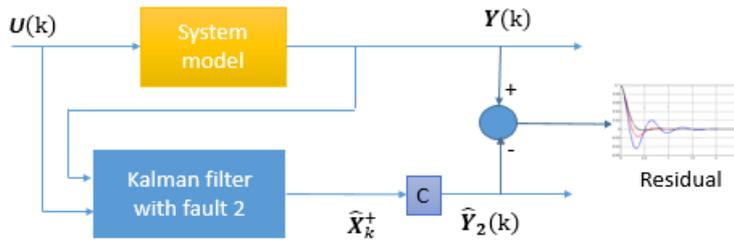


Figure 4.12 The fault diagnosis of actuator fault 2 method block diagram

The fault diagnosis of actuator fault 2 method is described in Figure 4.12. A residual can be calculated by comparing the estimated output from the Kalman filter with fault 2 and the measured output. The residual is named fault 2 residual, which will be a zero-mean noise with a very small value when the system has fault on actuator 2. If the system does not have fault on actuator 2, the residual is named no-fault 2 residual that will be a very large zero-mean noise. Assume the residual is $\mathbf{E}_{2re}(k)$, its transfer function is:

$$\begin{aligned} \mathbf{E}_{2re}(k) &= \mathbf{Y}(k) - \hat{\mathbf{Y}}_2(k) \\ \begin{cases} \forall \mathbf{E}_{2re}(k) \leq \mathbf{T}_{threshold}, \text{ there is fault on actuator 2.} \\ \exists \mathbf{E}_{2re}(k) > \mathbf{T}_{threshold}, \text{ there is no fault on actuator 2} \end{cases} \end{aligned} \quad (4.50)$$

Where $\mathbf{T}_{threshold}$ is the threshold value, which is chosen between the upper limit of the fault 2 residual and the upper limit of the no-fault 2 residual, and $\hat{\mathbf{Y}}_2(k)$ is the estimated output of the Kalman filter with fault 2.

3. Single fault on actuator 3

A way to detect single fault on actuator 3 is introduced here. When a fault occurs to actuator 3, the control signal $\mathbf{U}_3(s)$ cannot be executed to control the system. Therefore, the control signal

$U_3(s)$ is equal to 0 in the actuator 3 fault situation. The control impact of the whole system is changed from the no-fault situation to the actuator 3 fault situation.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times U_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} \\
 &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times U_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s)
 \end{aligned} \tag{4.51}$$

From the plate structure state representation, the control impact of the actuator 3 fault situation influences the output of the system. To diagnose the fault, another Kalman filter called Kalman filter with fault 3 is built to estimate the actuator 3 fault system output. The system matrix and the output matrix is the same comparing the actuator 3 system and the no-fault system. The input matrix needs to change for the actuator 3 system output estimation. The new input matrix should achieve the same control impact of actuator 3 fault system regardless of the input signal. Therefore the new control impact is:

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times U_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s) \\
 &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times U_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s)
 \end{aligned} \tag{4.52}$$

From the control impact equation, a way of getting the same control impact is to set the sum of parameters in the third row of the no-fault system input matrix to be 0 and make other parameters in the no-fault system input matrix remain the same.

$$\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63} = 0 \tag{4.53}$$

In the input matrix of the Kalman filter with fault 3, the parameters in the third row are set to be 0 and other parameters remain the same in the no-fault system input matrix.

$$\mathbf{H}_{f3} = \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} & \mathbf{0} \\ \mathbf{H}_{21} & \mathbf{H}_{22} & \mathbf{0} \\ \mathbf{H}_{31} & \mathbf{H}_{32} & \mathbf{0} \\ \mathbf{H}_{41} & \mathbf{H}_{42} & \mathbf{0} \\ \mathbf{H}_{51} & \mathbf{H}_{52} & \mathbf{0} \\ \mathbf{H}_{61} & \mathbf{H}_{62} & \mathbf{0} \end{bmatrix}$$

Where \mathbf{H}_{f3} is the input matrix of actuator fault 3 system.

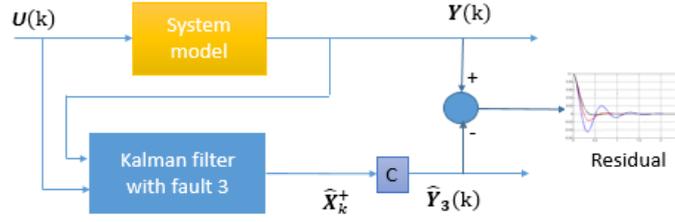


Figure 4.13 The fault diagnosis of actuator fault 3 method block diagram

The fault diagnosis of actuator fault 3 method is described in Figure 4.13. A residual can be calculated by comparing the estimated output from Kalman filter with fault 3 and measured output. The residual is named fault 3 residual, which will be a zero-mean noise with a very small value when the system has fault on actuator 3. If the system does not have fault on actuator 3, the residual is named no-fault 3 residual that will be a very large zero-mean noise. Assume the residual is $\mathbf{E}_{3re}(k)$, its transfer function is:

$$\mathbf{E}_{3re}(k) = \mathbf{Y}(k) - \hat{\mathbf{Y}}_3(k) \quad (4.54)$$

$$\begin{cases} \forall \mathbf{E}_{3re}(k) \leq \mathbf{T}_{threshold}, \text{there is fault on actuator 3.} \\ \exists \mathbf{E}_{3re}(k) > \mathbf{T}_{threshold}, \text{there is no fault on actuator 3} \end{cases}$$

Where $\mathbf{T}_{threshold}$ is the threshold value, which is chosen between the upper limit of the fault 3 residual and the upper limit of the no-fault 3 residual, and $\hat{\mathbf{Y}}_3(k)$ is the estimated output of the Kalman filter with fault 3.

Based on all the fault diagnosis methods mentioned before, the three Kalman filters are formed to achieve single fault diagnosis. Figure 4.14 shows the Kalman filter bank and the process of single fault diagnosis.

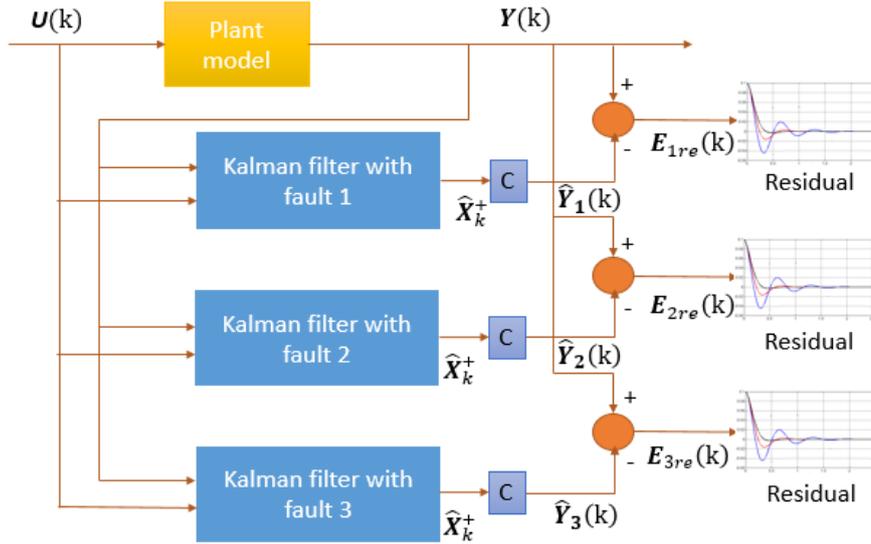


Figure 4.14 The Kalman filter bank to diagnose single fault

By using the Kalman filter bank above, the single actuator fault can be diagnosed in the plant structure system. In the fault classification, the actuator faults are classified into the single actuator fault and the double actuators fault. The following part will introduce the fault diagnosis for the double actuators fault situations.

4. Double faults on actuator 1 and 2

When a fault occurs to actuator 1 and 2, the control signal $U_1(s)$ and $U_2(s)$ cannot be executed to control the system. In another word, the control signal $U_1(s)$ and $U_2(s)$ are equal to 0 in the actuator 1 and 2 fault situation. The control impact in double faults on actuator 1 and 2 is shown as the following.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s) \\
 &= (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s)
 \end{aligned} \tag{4.55}$$

This transfer function illustrates the control impact on the real system. For the Kalman filter with fault 1 in figure 4.14, the control impact is computed as below.

$$\begin{aligned}
 I_{con}(s) &= \mathbf{0} \times \mathbf{0} + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s) \\
 &= (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s)
 \end{aligned} \tag{4.56}$$

For the Kalman filter with fault 2, the control impact is also represented.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} + \mathbf{0} \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s) \\
 &= (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_3(s)
 \end{aligned} \tag{4.57}$$

For the Kalman filter with fault 3, the control impact is also represented.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} + \mathbf{0} \times U_3(s) \\
 &= 0
 \end{aligned} \tag{4.58}$$

By comparing the control impacts, the first three control impacts are the same. According to the relationship between the control impacts and outputs, both of the estimated output $\hat{Y}_1(k)$ and $\hat{Y}_2(k)$ should be almost the same as the measured output $Y(k)$. However the estimated output $\hat{Y}_3(k)$ is different from the measured output $Y(k)$.

$$Y(k) \cong \hat{Y}_1(k) \cong \hat{Y}_2(k) \neq \hat{Y}_3(k) \tag{4.59}$$

Therefore, the following conclusion indicates the fault diagnosis result

$$\begin{cases} \forall E_{1re}(k) \cong \forall E_{2re}(k) \cong 0 \leq \mathbf{T}_{threshold}, \text{there is fault on actuator 1 and 2.} \\ \exists E_{3re}(k) > \mathbf{T}_{threshold}, \text{there is no fault on actuator 3} \end{cases} \tag{4.60}$$

5. Double faults on actuator 2 and 3

When a fault occurs to actuator 2 and 3, the control signal $U_2(s)$ and $U_3(s)$ cannot be executed to control the system. In another word, the control signal $U_2(s)$ and $U_3(s)$ is equal to 0 in the actuator 1 and 2 fault situation. The control impact in double faults on actuator 1 and 2 is shown as the following.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times U_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} \\
 &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times U_1(s)
 \end{aligned} \tag{4.61}$$

This transfer function illustrates the control impact on the real system. For the Kalman filter with fault 1, the control impact is computed as below.

$$\begin{aligned}
 I_{con}(s) &= \mathbf{0} \times U_1(s) + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} = \mathbf{0}
 \end{aligned} \tag{4.62}$$

For the Kalman filter with fault 2, the control impact is represented.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) + \mathbf{0} \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} \\
 &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s)
 \end{aligned} \tag{4.63}$$

For the Kalman filter with fault 3, the control impact is also represented.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s) \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{0} + \mathbf{0} \times \mathbf{0} \\
 &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{U}_1(s)
 \end{aligned} \tag{4.64}$$

By comparing the control impacts, these four control impacts are the same. According to the relationship between the control impacts and outputs, both of the estimated output $\hat{\mathbf{Y}}_2(k)$ and $\hat{\mathbf{Y}}_3(k)$ should be almost the same to the measured output $\mathbf{Y}(k)$. However the estimated output $\hat{\mathbf{Y}}_1(k)$ is different from the measured output $\mathbf{Y}(k)$.

$$\mathbf{Y}(k) \cong \hat{\mathbf{Y}}_2(k) \cong \hat{\mathbf{Y}}_3(k) \neq \hat{\mathbf{Y}}_1(k) \tag{4.65}$$

Therefore, the following conclusion indicates the fault diagnosis result

$$\begin{cases} \forall \mathbf{E}_{2re}(k) \cong \forall \mathbf{E}_{3re}(k) \cong 0 \leq \mathbf{T}_{threshold}, \text{there is fault on actuator 2 and 3.} \\ \exists \mathbf{E}_{1re}(k) > \mathbf{T}_{threshold}, \text{there is no fault on actuator 1} \end{cases} \tag{4.66}$$

6. Double faults on actuator 1 and 3

When a fault occurs to actuator 1 and 3, the control signal $\mathbf{U}_1(s)$ and $\mathbf{U}_3(s)$ cannot be executed to control the system. In another word, the control signal $\mathbf{U}_1(s)$ and $\mathbf{U}_3(s)$ are equal to 0 in the actuator 1 and 3 fault situation. The control impact in double faults on actuator 1 and 3 is shown as the following.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} \\
 &= (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times \mathbf{U}_2(s)
 \end{aligned} \tag{4.67}$$

This transfer function illustrates the control impact on the real system. For the Kalman filter with fault 1, the control impact is computed as below.

$$\begin{aligned}
 I_{con}(s) &= \mathbf{0} \times \mathbf{0} + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} \\
 &= (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times U_2(s)
 \end{aligned} \tag{4.68}$$

For the Kalman filter with fault 2, the control impact is represented.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} + \mathbf{0} \times U_2(s) \\
 &\quad + (\mathbf{H}_{13} + \mathbf{H}_{23} + \mathbf{H}_{33} + \mathbf{H}_{43} + \mathbf{H}_{53} + \mathbf{H}_{63}) \times \mathbf{0} = \mathbf{0}
 \end{aligned} \tag{4.69}$$

For the Kalman filter with fault 3, the control impact is also represented.

$$\begin{aligned}
 I_{con}(s) &= (\mathbf{H}_{11} + \mathbf{H}_{21} + \mathbf{H}_{31} + \mathbf{H}_{41} + \mathbf{H}_{51} + \mathbf{H}_{61}) \times \mathbf{0} \\
 &\quad + (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s) + \mathbf{0} \times \mathbf{0} \\
 &= (\mathbf{H}_{12} + \mathbf{H}_{22} + \mathbf{H}_{32} + \mathbf{H}_{42} + \mathbf{H}_{52} + \mathbf{H}_{62}) \times U_2(s)
 \end{aligned} \tag{4.70}$$

By comparing the control impacts, these four control impacts are the same. According to the relationship between the control impacts and outputs, both of the estimated output $\hat{Y}_1(k)$ and $\hat{Y}_3(k)$ should be almost the same to the measured output $Y(k)$. However the estimated output $\hat{Y}_2(k)$ is different from the measured output $Y(k)$.

$$Y(k) \cong \hat{Y}_1(k) \cong \hat{Y}_3(k) \neq \hat{Y}_2(k) \tag{4.71}$$

Therefore, the following conclusion indicates the fault diagnosis result

$$\begin{cases} \forall E_{1re}(k) \cong \forall E_{3re}(k) \cong 0 \leq \mathbf{T}_{threshold}, \text{there is fault on actuator 1 and 3.} \\ \exists E_{2re}(k) > \mathbf{T}_{threshold}, \text{there is no fault on actuator 2} \end{cases} \tag{4.72}$$

As discussed above, the three combinations of double actuators fault can be diagnosed by the existing Kalman filter bank. It is not necessary to build more Kalman filters for the double actuators fault. The Kalman filter bank for FDD is described in Figure 4.15.

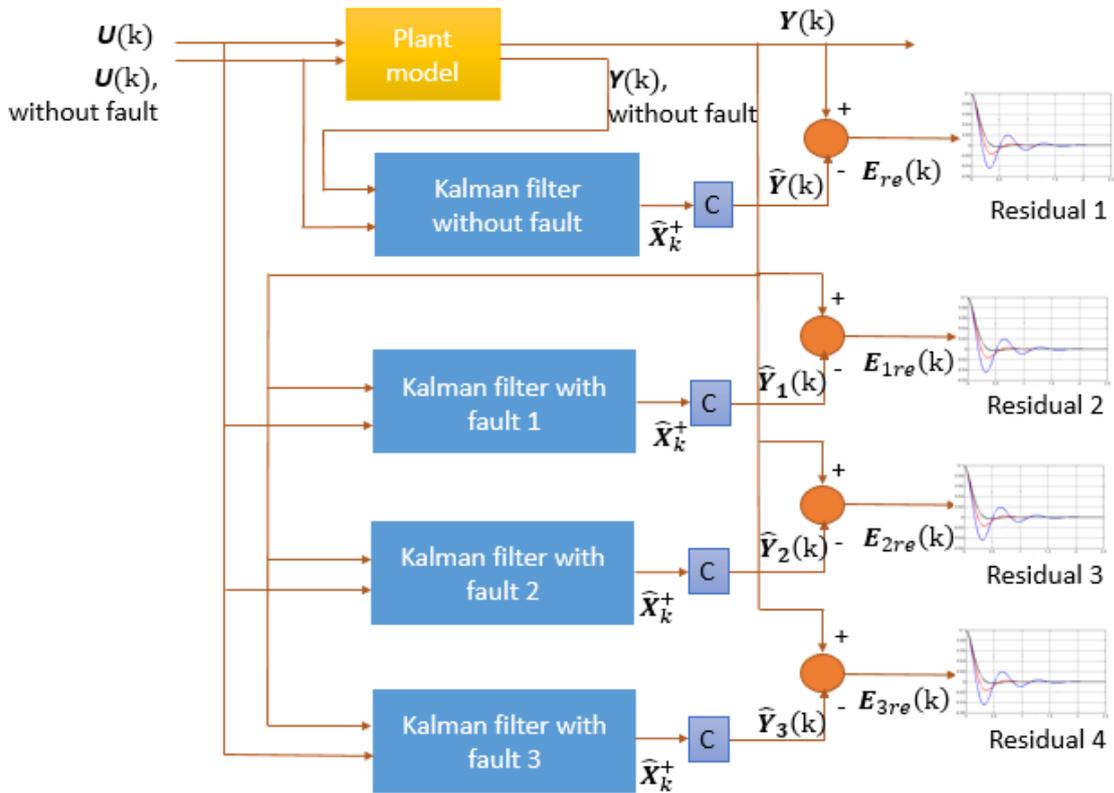


Figure 4.15 The block diagram of the Kalman filter bank for FDD

The system model and the Kalman filter bank supply the four residuals for FDD. By comparing the threshold values and the residuals, the result can be analysed to diagnosis fault. Taking the fault detection analyse as an example, the method for detecting system fault is as follows.

$$\begin{cases} \forall E_{re}(k) \leq \mathbf{T}_{threshold}, \text{there is no fault in the system.} \\ \exists E_{re}(k) > \mathbf{T}_{threshold}, \text{the system has unknown faults.} \end{cases} \quad (4.73)$$

However the residual $E_{re}(k)$ is a zero-mean noise. If the system has unknown faults, the residual $E_{re}(k)$ will not be larger than the threshold value all the time. If only the threshold value is used to analyse the residuals to detect faults, the detection result cannot properly reflect the real situation of the system. The fault detection, simply using the threshold value, is plotted in Figure 4.16.

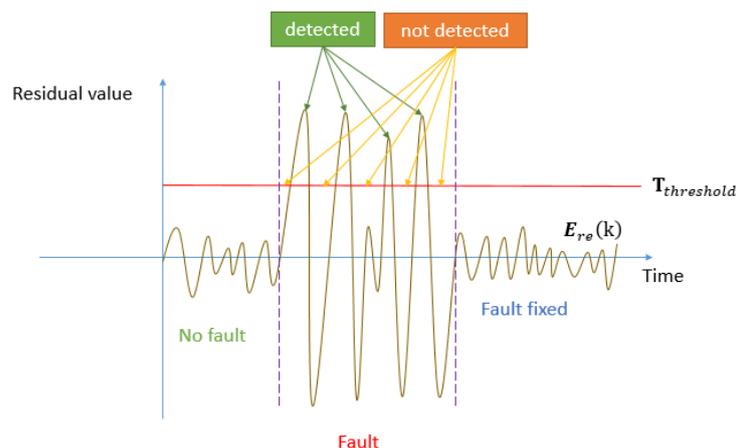


Figure 4.16 The fault detection simply using the threshold value

Based on the theoretical analysis before, the artificial residual is created in three different situations, which are no-fault situation, fault situation, fault fixed situation. In no-fault and fault fixed situations, the detection results are considered acceptable by comparing all the residual values and the threshold value. However, the fault detection method only detects the fault situation when the residual value is larger than the threshold value. In the case of a fault, the fault cannot be detected when the residual value is less than the threshold value. This kind of unstable detection can also occur during analysing the fault diagnosis residuals. Therefore, this thesis creates a new algorithm, the normalization algorithm to analyse the residuals to obtain the FDD results.

4.4.4 The normalization algorithm

In the theoretical analysis of the residual, it is generally considered that the initial condition of the residual is zero. But in reality, most of the initial condition of the residuals are not zero. Each different initial condition brings unnecessary interference to the fault detection.

So the first goal of the normalization algorithm is to eliminate the impact of the initial state on the residual analysis. The system previously described in this thesis was transformed to a discrete-time system. Figure 4.17 shows an artificial example where the initial condition of the residual is not zero.

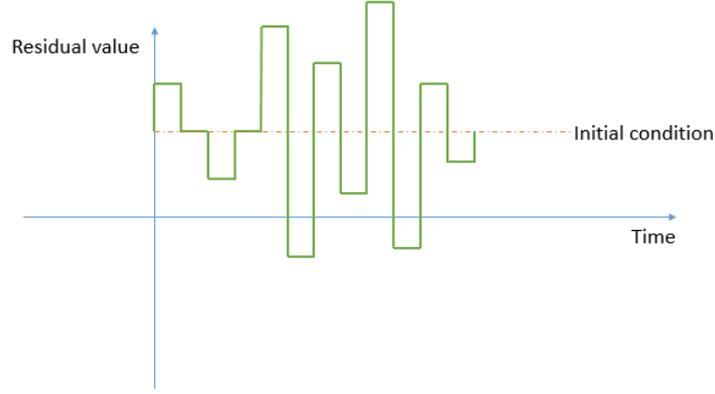


Figure 4.17 Non-zero initial condition example of the residual

One way to solve the above problem is to subtract the residual value of the time K from the residual value at time $K+1$ to obtain a new residual. (Take the fault detection as the example)

$$\mathbf{E}_{ne}(k+1) = \mathbf{E}_{re}(k+1) - \mathbf{E}_{re}(k) \quad (4.74)$$

Where $\mathbf{E}_{ne}(k+1)$ is the new zero mean residual with zero initial condition.

This method achieves the elimination of the initial state by calculating the difference between the residual $\mathbf{E}_{re}(k)$ at each moment and the previous moment. After eliminating the initial state of the residual $\mathbf{E}_{re}(k)$, at each moment, the normalization algorithm takes the previous N samples (including the samples at the current time) for the new residual with an initial state of zero. This means that the sampling must start at time N . The value of N needs to be greater than one period of the new residual to ensure the accuracy of the sample analysis. At the same time, the value of N is less than twice the new residual period to ensure the efficiency of fault detection.

$$\mathbf{S}_{sa} = [\mathbf{E}_{ne}(k), \mathbf{E}_{ne}(k-1), \mathbf{E}_{ne}(k-2), \dots, \mathbf{E}_{ne}(k-N)], T_{ne} \leq N \leq 2T_{ne} \quad (4.75)$$

Where \mathbf{S}_{sa} is the N samples of the residual $\mathbf{E}_{ne}(k)$, and T_{ne} is the period of the residual $\mathbf{E}_{ne}(k)$. After sampling, the normalization algorithm calculates the RMS value of the previous N samples at each time.

$$\text{RMS}_{sa} = \sqrt{\frac{1}{N}(\mathbf{E}_{ne}(k)^2 + \mathbf{E}_{ne}(k-1)^2 + \mathbf{E}_{ne}(k-2)^2 + \dots + \mathbf{E}_{ne}(k-N)^2)} \quad (4.76)$$

The normalization algorithm calculates the RMS value of the sample in the fault detection case and the non-fault situation will be very different. The calculated RMS value will not be a variable that is a zero mean. In the case of fault detection, the RMS value in the event of a fault would be a large variable and have a lower limit of not less than zero. At the same time, in the

case of non-fault, the RMS value will be a small value and have an upper limit of not less than zero. The upper RMS value of non-fault conditions will be less than the lower RMS value of fault conditions. Therefore, the fault detection based on the RMS value judgment is easier to implement.

$$\forall \text{RMS}_{no} \leq \forall \text{RMS}_{fa} \quad (4.77)$$

Where RMS_{no} is the RMS value of the non-fault condition and RMS_{fa} is the RMS value of the fault condition.

Based on the above analysis, a threshold value is created to determine if the system has fault. This threshold is an arbitrary value between the upper RMS value of non-fault condition and the lower RMS value of fault condition. The threshold value determines the response time and the accuracy of fault detection. How to value the threshold will be described in detail later in the system simulation.

$$\forall \text{RMS}_{no} \leq \mathbf{T}_{Rthreshold} \leq \forall \text{RMS}_{fa} \quad (4.78)$$

Where $\mathbf{T}_{Rthreshold}$ is the threshold value for analysing the RMS value.

The above method is a description of the working principle of the normalization algorithm. Figure 4.18 summarizes the normalization algorithm into a flow chart.

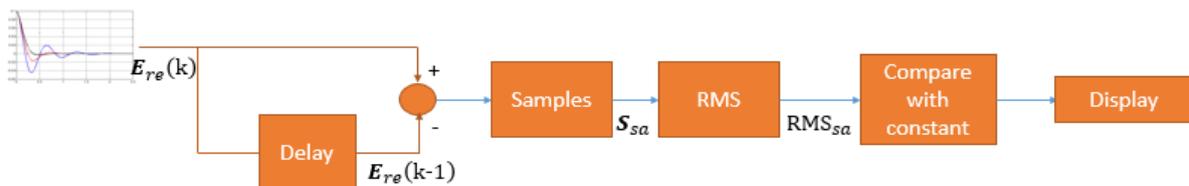


Figure 4.18 The normalization algorithm block diagram for analysing the fault detection residual

The above algorithm can also analyse and compare the three residuals of the fault diagnosis to obtain the detection result. Figure 4.19 shows the complete algorithm block diagram.

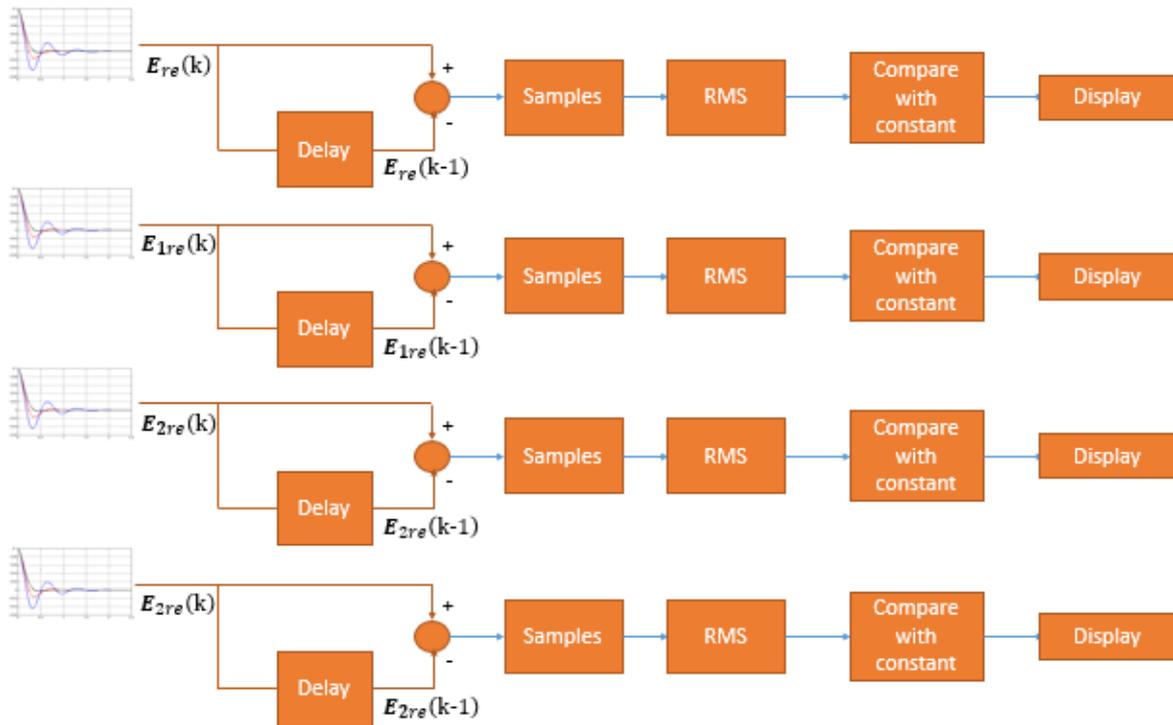


Figure 4.19 The normalization algorithm block diagram for analysing the FDD residuals
 Through the design of the Kalman filter bank and the normalization algorithm, the Kalman filter based FDD method can be successfully applied to the plant structure system. The overall Kalman filter based FDD method is shown in Figure 4.20.

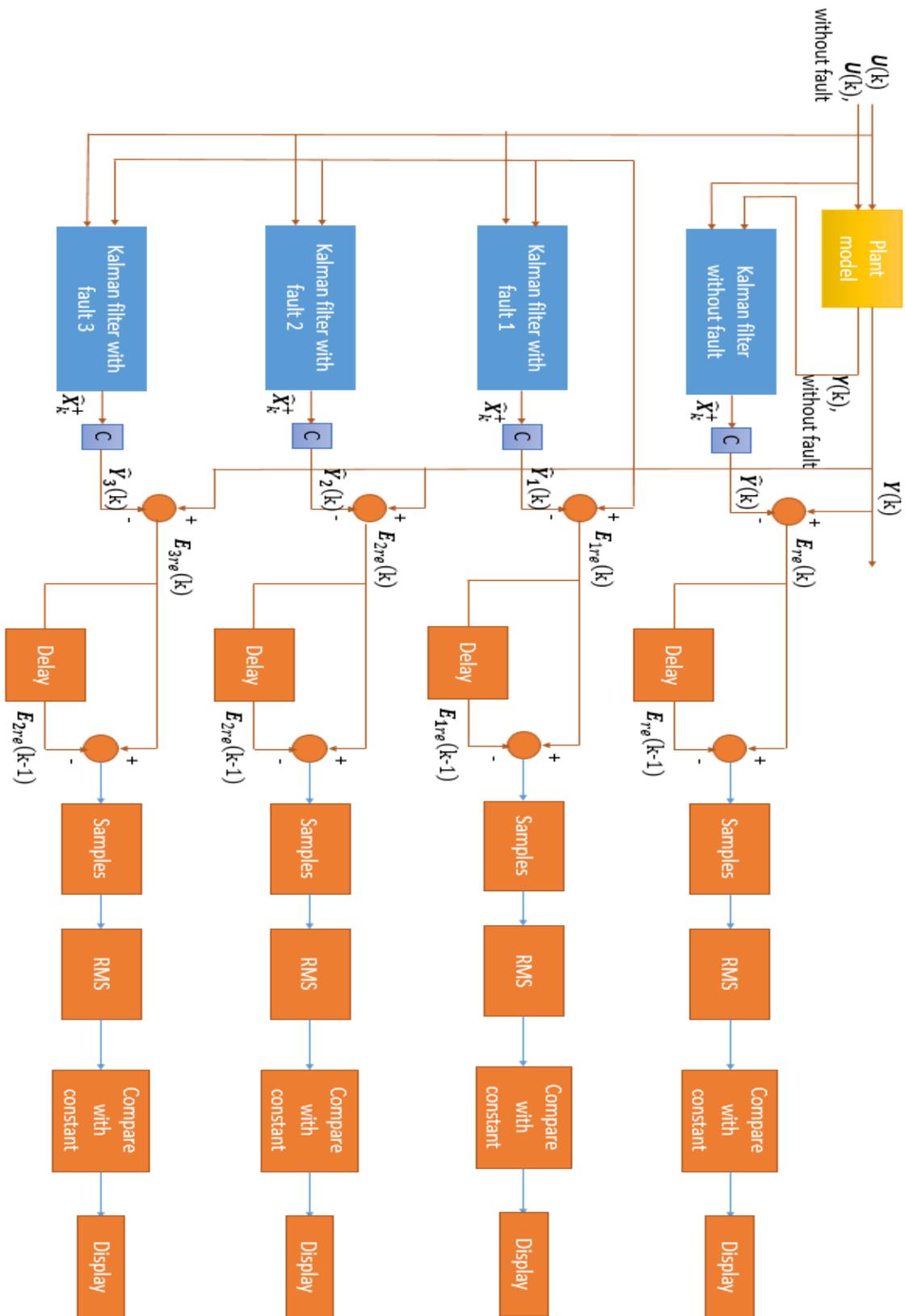


Figure 4.20 The Kalman filter based FDD block diagram

Chapter 4: Kalman Filter Based FDD method

In the following chapters, in order to verify the Kalman filter based FDD method, the corresponding system simulation and experiment are carried out.

Chapter 5 : Kalman Filter Based FDD Method Validation

Firstly, based on the knowledge of the introduction of the system identification and the Kalman filter based FDD method, an artificial plant structure system was created in MATLAB Simulink. In simulation, three gains are added between the controller and the system feedback to simulate the system's actuator failure. The plant structure system simulates the case of single fault and double faults by setting the values of three gains. The Kalman filter based FDD method is used to detect the artificially set fault. In the simulation, the relevant parameters are valued according to the actual situation of the system. Subsequently, the Kalman filter based FDD method was applied to the real vibration system. The two system models are linear discrete-time systems that are simplified according to the real plant structure system. The signals emitted the controller and emitted by the actuator are taken out in the real system and inputted into the two models. The experiment also shows the process of the Kalman filter based FDD, especially the observation of residuals.

5.1 Conceptual Kalman filter based FDD method validation.

MATLAB Simulink is a simulation and model-based design tool for dynamic systems and embedded systems. It can design, simulate, execute and test a variety of time-varying systems. In this part, a linear discrete-time system simplified by the Plant structure is created in Simulink. In order to make the system produce a vibration output, according to the frequency corresponding to the first three modes of the system at the time of modelling, three sine waves of the corresponding frequency are used as disturbance in the system. The system will generate vibration signals with the maximum observable amplitude at these three frequencies. A controller called PPF is applied to the system to reduce the impact of disturbance to the system. In fact, any controller that has control effect over the system can be used as a controller here, as long as it can generate the corresponding control signals. Therefore, the PPF controller is not analysed and introduced in detail in this article. The specific information can be found in the reference [28]. The three gain blocks represent actuators 1, 2, and 3. When the value of gain is 1, it is equivalent to the actuator not having fault. When the value of gain is 0, it is equivalent to the corresponding actuator fault. Because the sensor fault is not the researched object of this thesis, it is not deliberately drawn in the simulation. The signal from the artificial system can be the signal feedback from the sensor by default. The artificial system is shown in Figure 5.1.

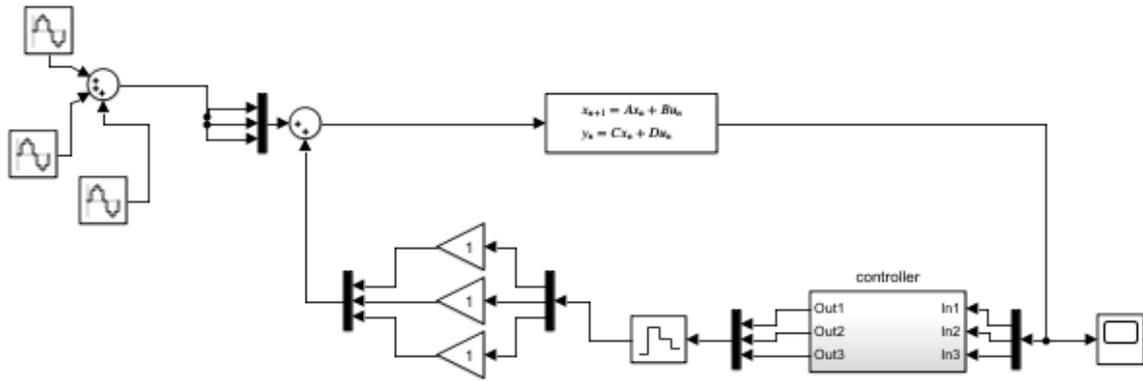


Figure 5.1 The artificial system to validate the Kalman filter based FDD method

As previously introduced, due to the presence of disturbances, the control signals given by the controller and the control signals after the actuator are taken from the artificial system. According to the original open-loop system, a system model containing two same artificial systems is introduced to provide the necessary information to the Kalman filter bank.

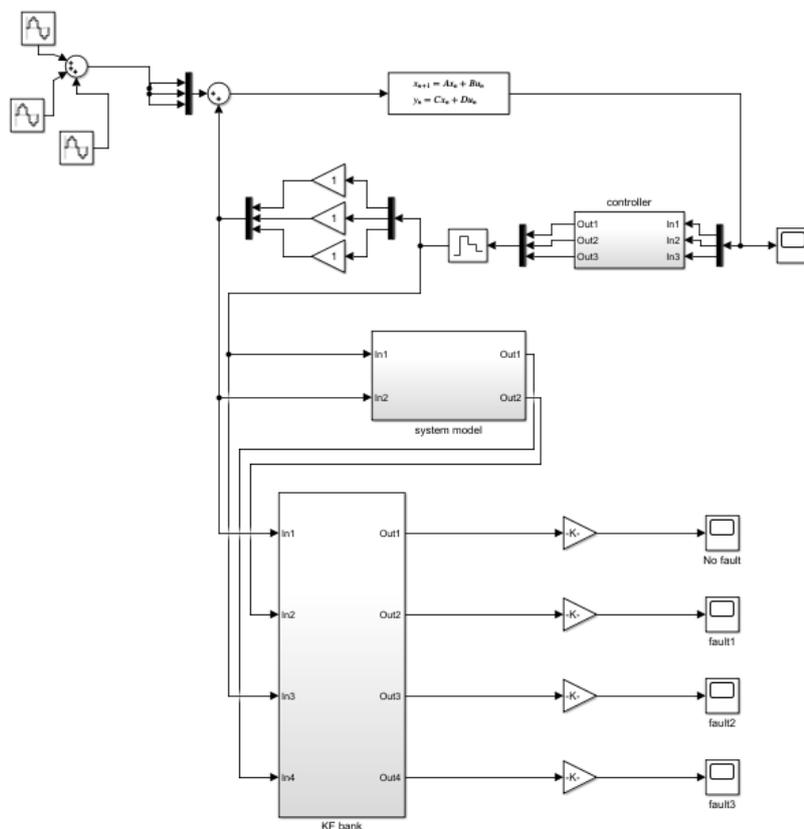


Figure 5.2 The Kalman filter bank simulation to derive four residuals

Figure 5.2 shows how the Kalman filter bank and system model are connected. At the output of the Kalman filter bank, four residuals are calculated in real time. The structure of the system model is shown in Figure 5.3 and the structure of the Kalman filter bank is shown in Figure 5.4.

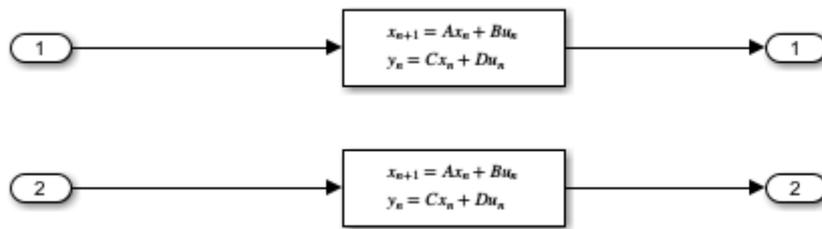


Figure 5.3 The structure of the system model

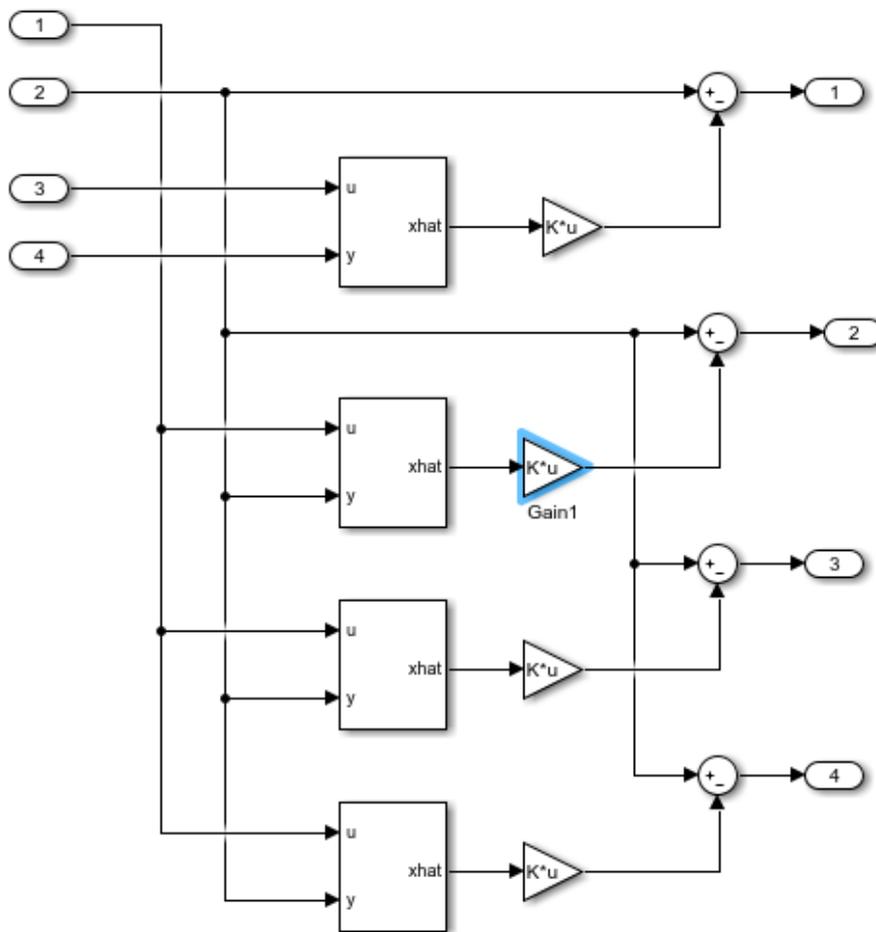


Figure 5.4 The structure of the Kalman filter bank

In each Kalman filter, the initial state estimate of the system, the covariance of the initial state estimate error, the covariance of the process noise, and the covariance of the measurement

noise must all be given. In the plant structure system, it is desirable that the top plate is always in a stable and stationary state, so the position and speed of the three measuring points of the system are all 0. Therefore, the initial state estimate of the system is set as follows.

$$\mathbf{X}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Assume all the initial estimated states have the error in 1. The covariance of the initial state estimate error is:

$$\mathbf{P}_0^+ = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The covariance of the process noise is shown as follows:

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

The covariance of the measurement noise is:

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

The analysis of the residuals by the normalization algorithm is also established in the system, the structure of which is shown in Figure 5.5.

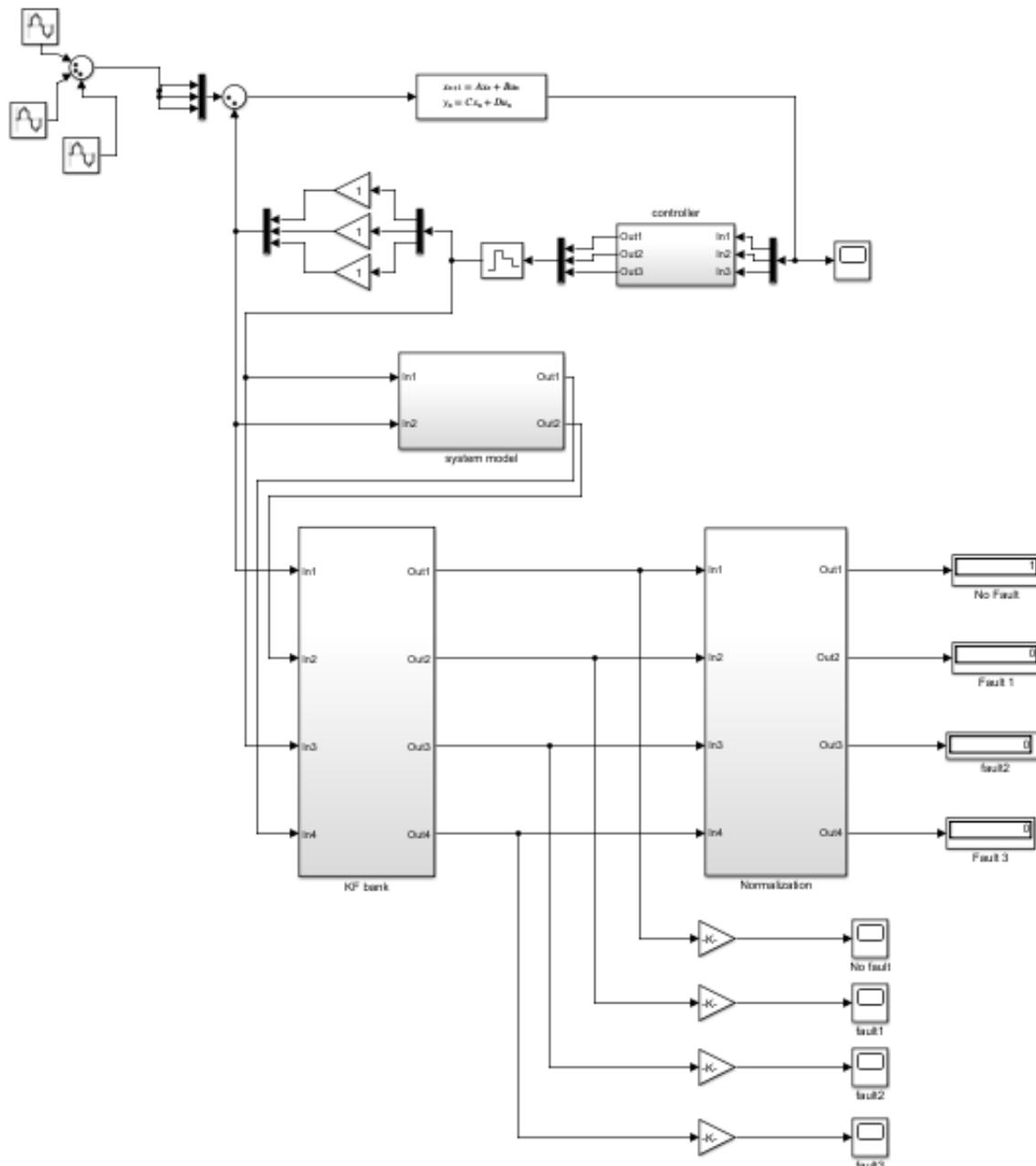


Figure 5.5 The simulation structure of the Kalman filter based FDD method

The four scopes in Figure 5.5 are used to observe the four residuals of the Kalman filter bank, while the four display blocks are used to display troubleshooting and results. The structure of the Normalization algorithm is shown in Figure 5.6, where the threshold values are explained in detail in the simulation.

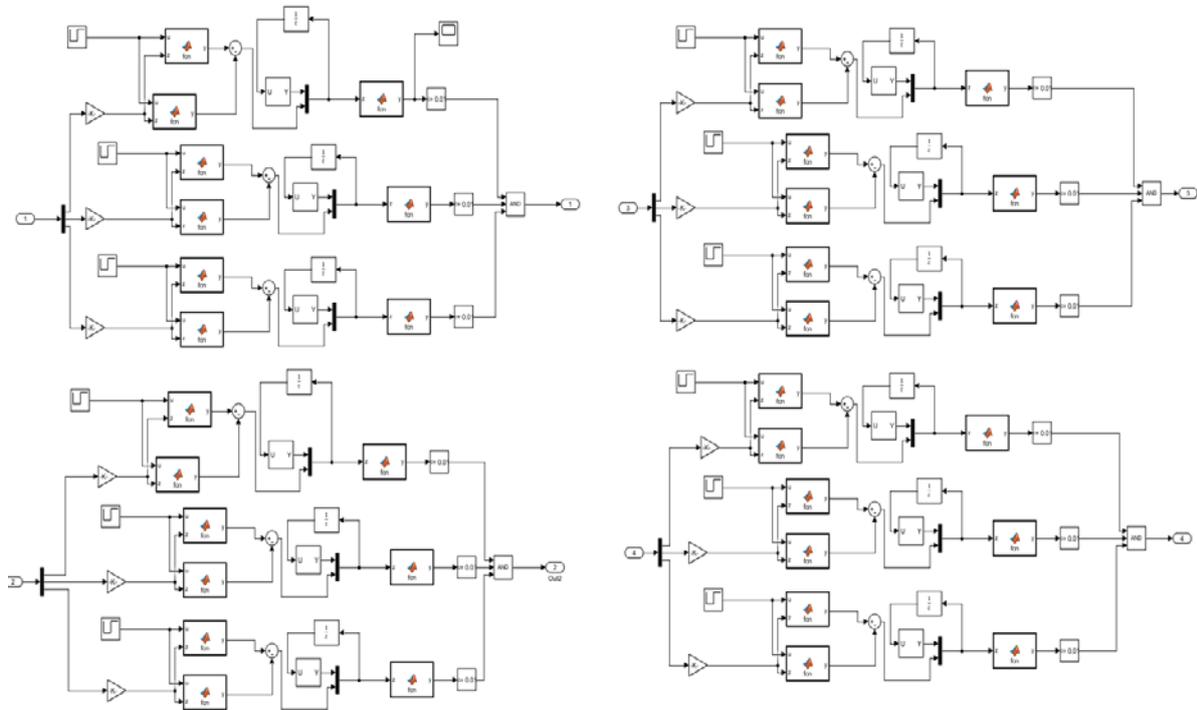


Figure 5.6 The structure of the normalization algorithm

In the normalization algorithm, each period has 4.5 samples according to the previous calculation and sampling time of the system, and the RMS value is calculated by taking the previous 7 samples at each time. Each residual of the Kalman filter bank has 3 signals. To analyse all of the signals, each signal uses the normalization algorithm. All the analysed signals have been amplified 10^{10} times.

Take the first signal of fault detection residual as an example. After introducing the complete simulation structure, the observations of the fault detection RMS values without any faults, and the fault detection RMS values with faults, are compared in the system. When the system does not have fault, the first signal fault detection RMS value is shown in Figure 5.7.

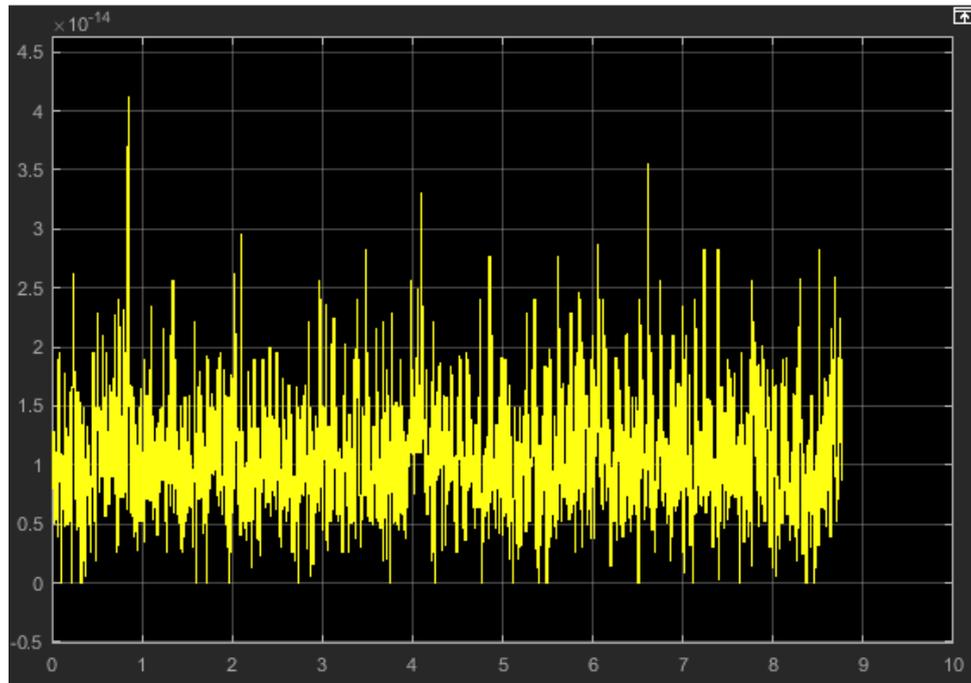


Figure 5.7 The fault detection RMS value when system do not have fault

When the system has fault, the first signal of fault detection RMS value is shown in Figure 5.8.

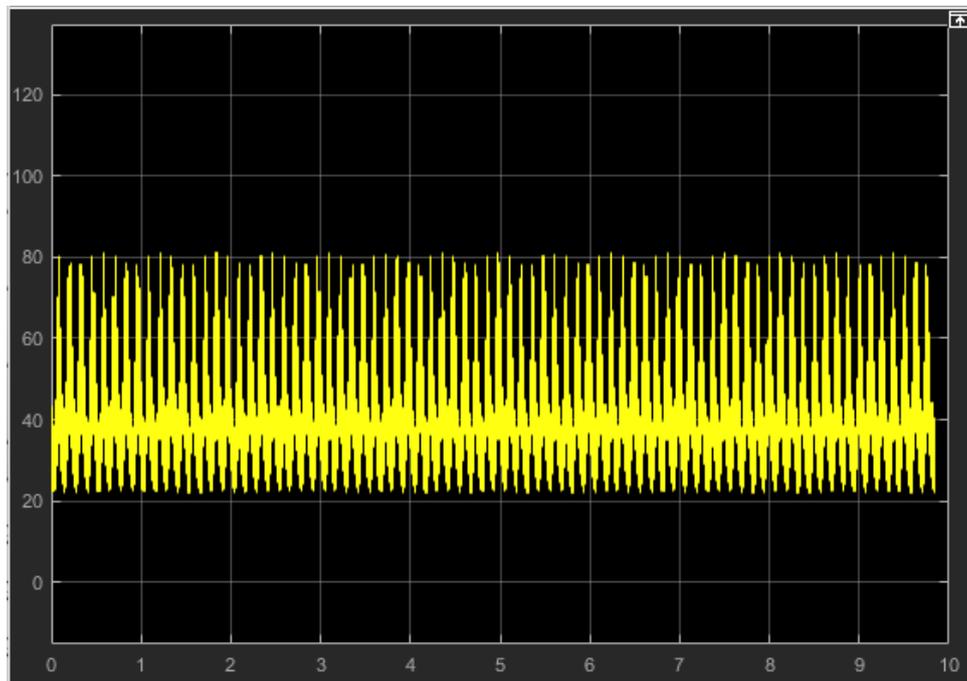


Figure 5.8 The fault detection RMS value when the system has fault

As shown in Figure 5.7 and Figure 5.8, when the system does not have fault, the first signal of fault detection RMS value has the largest value at 4.3×10^{-14} . When the system has fault, the first signal of fault detection RMS value has the lowest value at 22. Therefore, the threshold value is set to 0.01 to ensure the fault detection efficiency and accuracy for this case.

By observing all the RMS values in no-fault system and fault systems. All the threshold values are set to 0.01.

5.1.1 No fault system simulation result

All the structures and parameters of the fault detection and positioning system have been introduced. In the simulation, first observe the four residuals and test results without faults.

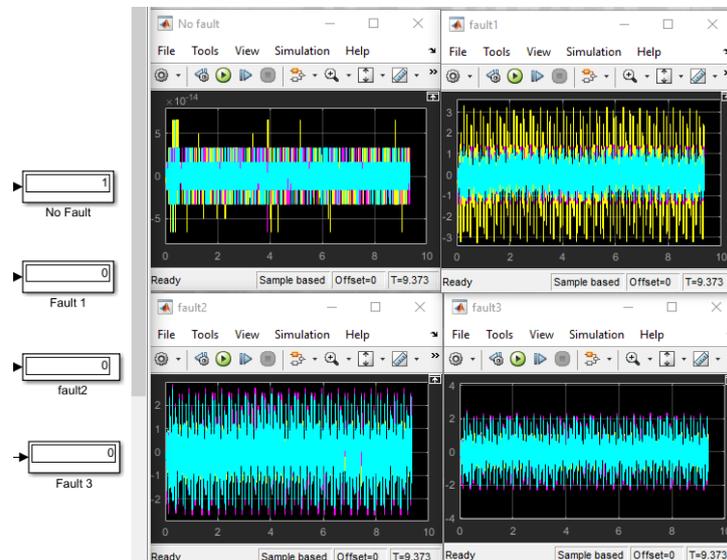


Figure 5.9 The four residuals and results without faults in simulation

According to the simulation, in the first no fault scope, the signals in residual are very small zero mean values, and their maximum values are around 5×10^{-14} . The other three scopes have maximum values of around 2 to 4. The maximum values in the first residual is significantly lower than that in the other three residuals, and the system is not faulty at the moment. In the display block, 1 means that the name of the display block is valid, and 0 means that the display block is invalid. In the display block, the same FDD result shows the system having no fault.

5.1.2 Actuator fault 1 system simulation result

By setting the gain in actuator fault 1, the simulation result can be observed in Figure 5.10.

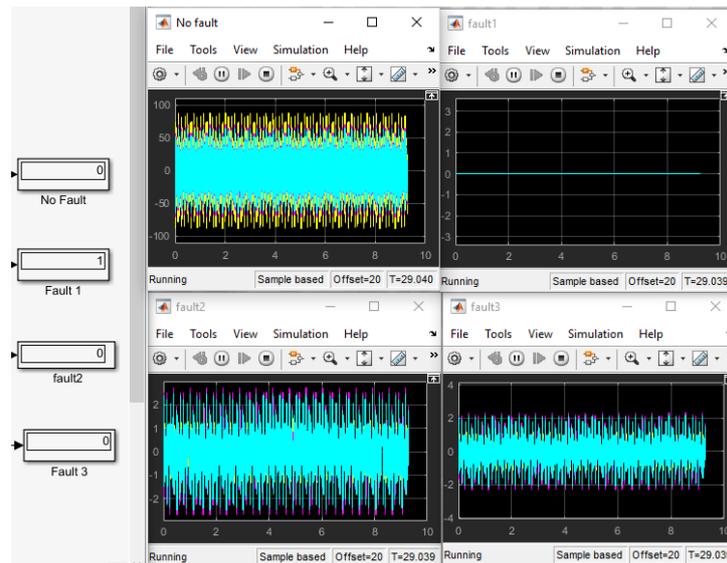


Figure 5.10 The four residuals and results with actuator fault 1 in simulation

According to the simulation, in the fault 1 scope, the three signals in residual are very small zero mean values, and their values are around 0. The other three scopes have maximum values of around 2 to 80. The signals in the second residual are significantly lower than that in the other three residuals, which means the system is having fault 1 at the moment. In the display block, the same FDD result shows the system having fault 1.

5.1.3 Actuator fault 2 system simulation result

By setting the gain in actuator fault 2, the simulation result can be observed in Figure 5.11.

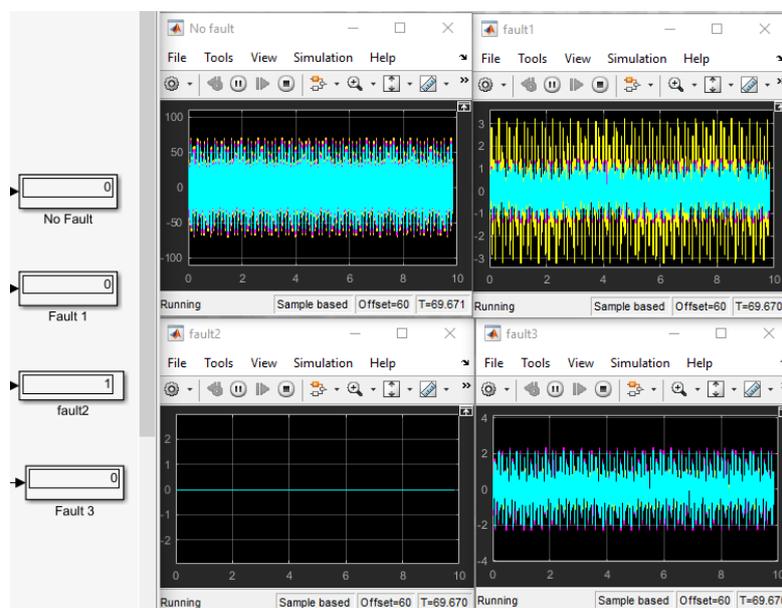


Figure 5.11 The four residuals and results with actuator fault 2 in simulation

According to the simulation, in the fault 2 scope, the three signals residual are very small zero mean values, and their values are around 0. The other three scopes have maximum values of around 2 to 75. The signals in the third residual are significantly lower than that in the other three residuals, which means the system is having fault 2 at the moment. In the display block, the same FDD result shows the system having fault 2.

5.1.4 Actuator fault 3 system simulation result

By setting the gain in actuator fault 3, the simulation result can be observed in Figure 5.12.

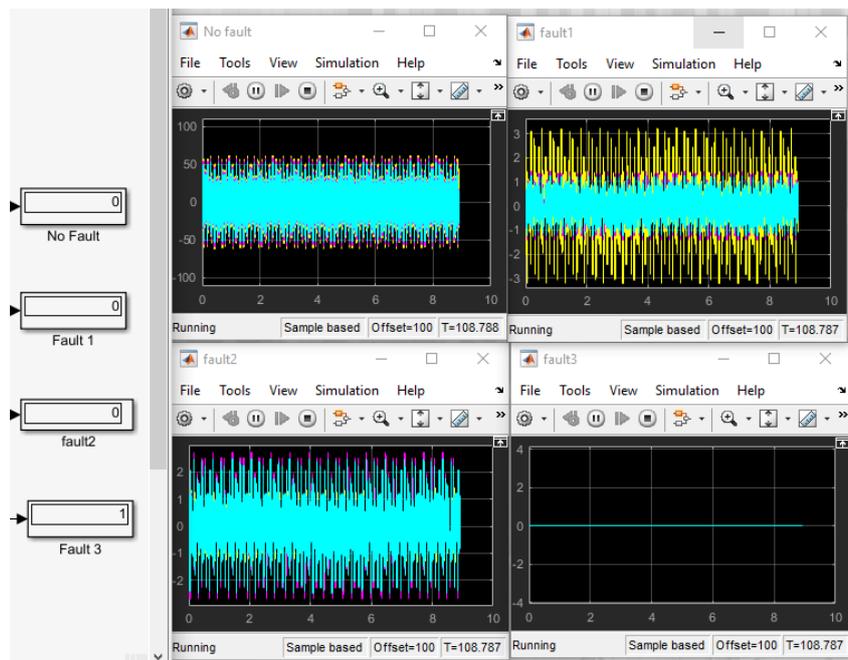


Figure 5.12 The four residuals and results with actuator fault 3 in simulation

According to the simulation, in the fault 3 scope, the signals in residual are very small zero mean values, and their values are around 0. The other three scopes have maximum values around 2 to 60. The fourth residual is significantly lower than the other three residuals, which means the system is having fault 3 at the moment. In the display block, the same FDD result shows the system having fault 3.

5.1.5 Actuator fault 1 and 2 system simulation result

By setting the gain in actuator fault 1 and 2, the simulation result can be observed in Figure 5.13.

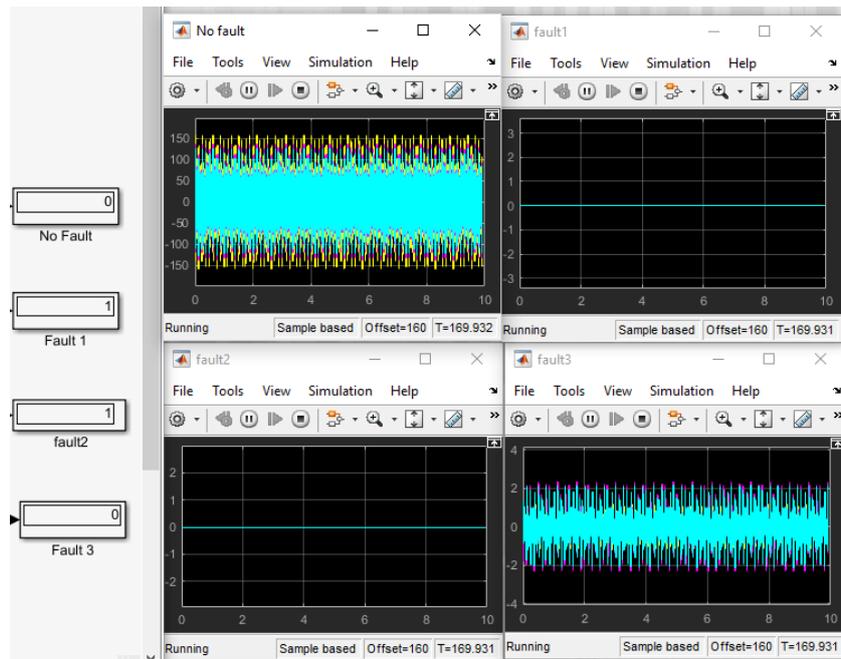


Figure 5.13 The four residuals and results with actuator fault 1 and 2 in simulation

According to the simulation, in the fault 1 and 2 scope, the signals in residuals are very small zero mean values, and all of the values are around 0. The other two scopes have maximum values of around 2 to 150. The second and third residuals are significantly lower than the other two residuals, which means the system is having fault 1 and 2 at the moment. In the display block, the same FDD result shows the system having fault 1 and 2.

The other interesting finding is that the maximum value of signal in no-fault block of the double faults situation is larger than that of the single fault situation.

5.1.6 Actuator fault 2 and 3 system simulation result

By setting the gain in actuator fault 2 and 3, the simulation result can be observed in Figure 5.14.

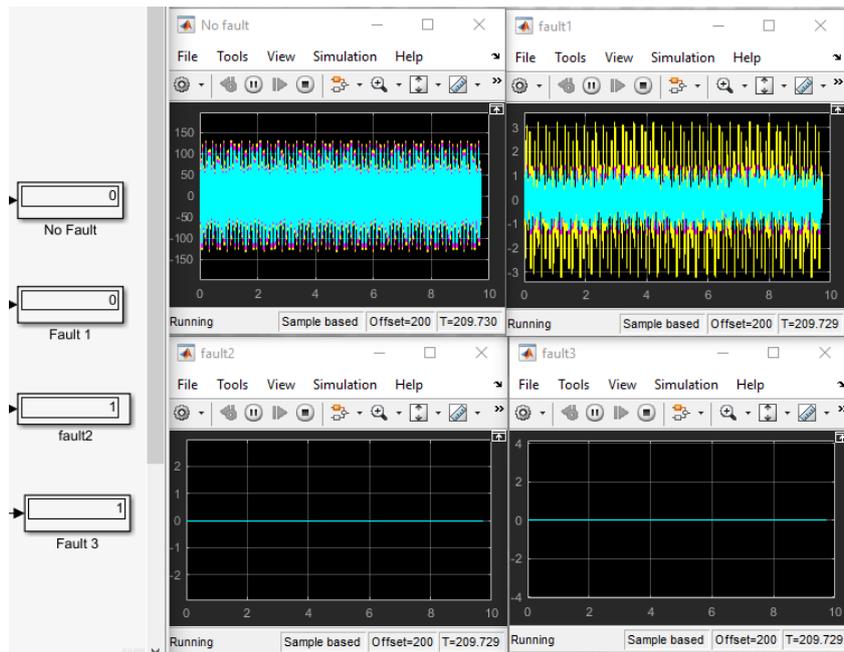


Figure 5.14 The four residuals and results with actuator fault 2 and 3 in simulation

According to the simulation, in the fault 2 and 3 scope, the signals in residuals are very small zero mean values, and all of the values are around 0. The other two scopes have maximum values around 3 to 140. The third and fourth residuals are significantly lower than the other two residuals, which means the system is having fault 2 and 3 at the moment. In the display block, the same FDD result shows the system having fault 2 and 3.

5.1.7 Actuator fault 1 and 3 system simulation result

By setting the gain in actuator fault 1 and 3, the simulation result can be observed in Figure 5.15.

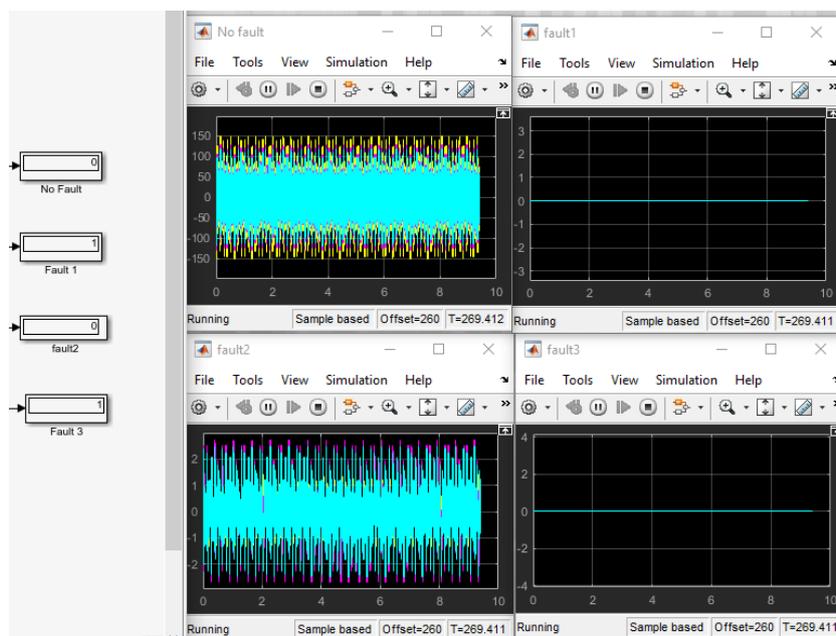


Figure 5.15 The four residuals and results with actuator fault 1 and 3 in simulation

According to the simulation, in the fault 1 and 3 scope, the signals in residuals are very small zero mean values, and all of the values are around 0. The other two scopes have maximum values around 3 to 140. The second and fourth residuals are significantly lower than the other three residuals, which means the system is having fault 1 and 3 at the moment. In the display block, the same FDD result shows the system having fault 1 and 3.

In conclusion, the simulations above validate the Kalman filter based FDD method can be applied to the plant structure system. The results via the different simulations confirm that the designed Kalman filter based FDD method is able to detect the real time faults in the plant structure successfully.

5.2 The Kalman filter based FDD method via experiment

After verifying the Kalman filter based FDD method in the simulation, this method was applied to the actual Plant structure system. The experiment used a software called dSPACE, which reads the signal generated by the sensor from the actual system, then the controller feeds the control signal back to the actuator of the system. At the same time, this software can observe various parameters in the real-time system. The controller used in the experiment is the same PPF control mentioned before. In the actual system, the experiment also takes two control signals from the controller behind the system and after the actuator. Therefore, when building the dSPACE model, three gains are added before the controller and before the external output signal. When any of the three gains is zero, the corresponding actuator of the system will not

work. As shown in Figure 5.16, the relevant model is set up in MATLAB and then modelled into dSPACE to control the real system.

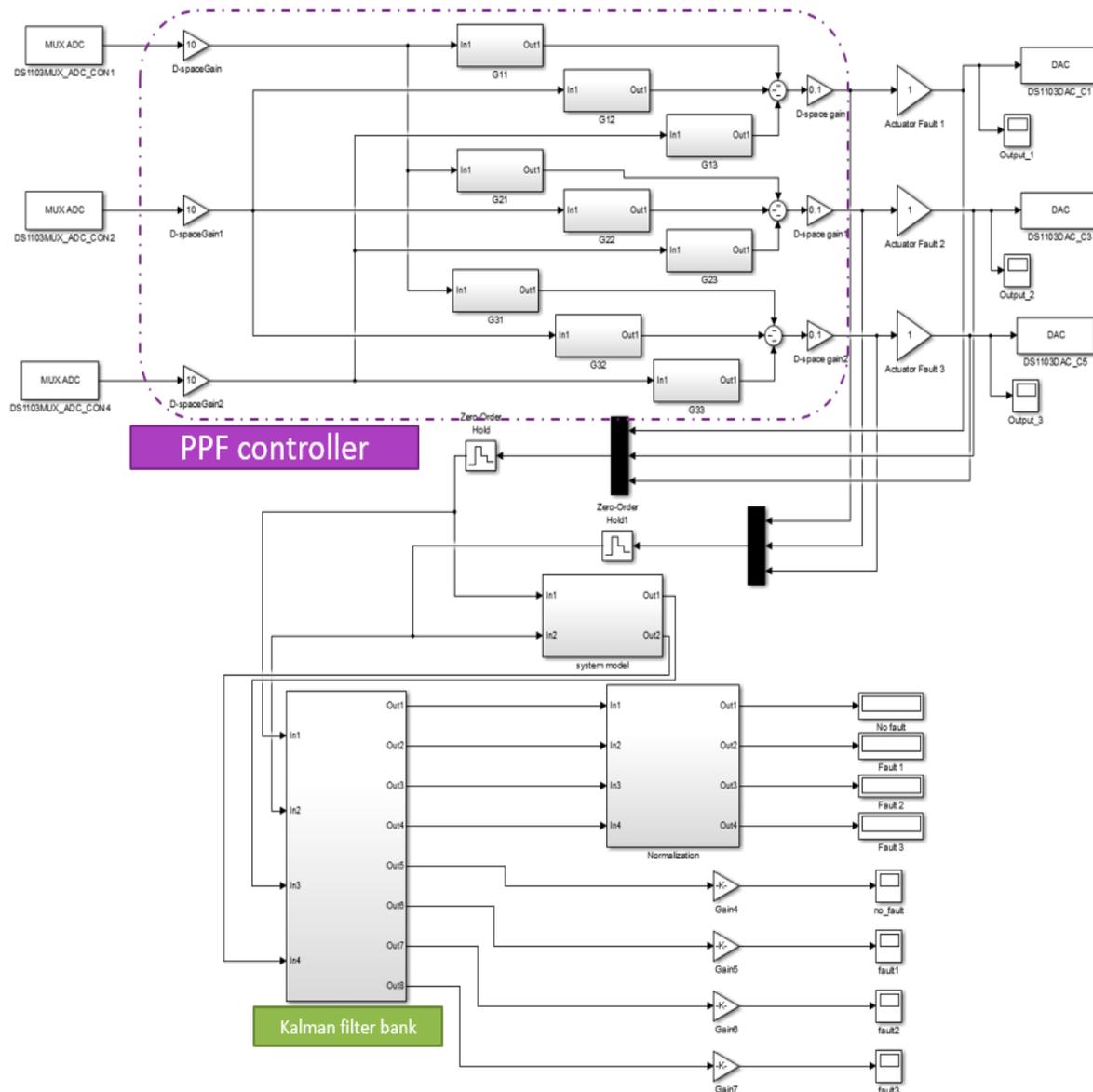


Figure 5.16 The structure for the experiment

In the simulation, no faults and six faulty conditions were tested. In order to test the practicality of the FDD method, the above seven cases are also applied in the test.

5.2.1 No fault system experiment result

First, the test is conducted without any fault in the system. In the dSPACE software, the FDD results of the four residuals obtained from the Kalman filter bank are shown in Figure 5.17.



Figure 5.17 The four residuals and results without faults in experiment

As can be seen in Figure 5.17, the value of the first residual is 0, and the largest value in the simulation is 5×10^{-14} , are almost equal. In theory, this residual should be a zero mean noise. The value in the experiment is zero because the scale of the coordinate axis is large, so this noise cannot be displayed. The other three residuals are very large noise, but not zero mean, which is consistent with theoretical analysis. This is because in the experiment, the initial condition of the system may not be 0. In the result of FDD, the system shows no fault. Although the results of the simulation and experiment seem to be a little different, in fact the results of the simulation and experiment are the same. The previous theoretical analysis about the residual is verified again.

5.2.2 Actuator fault 1 system simulation result

Artificial setting actuator fault 1, observe the results of FDD and four residuals in Figure 5.18.

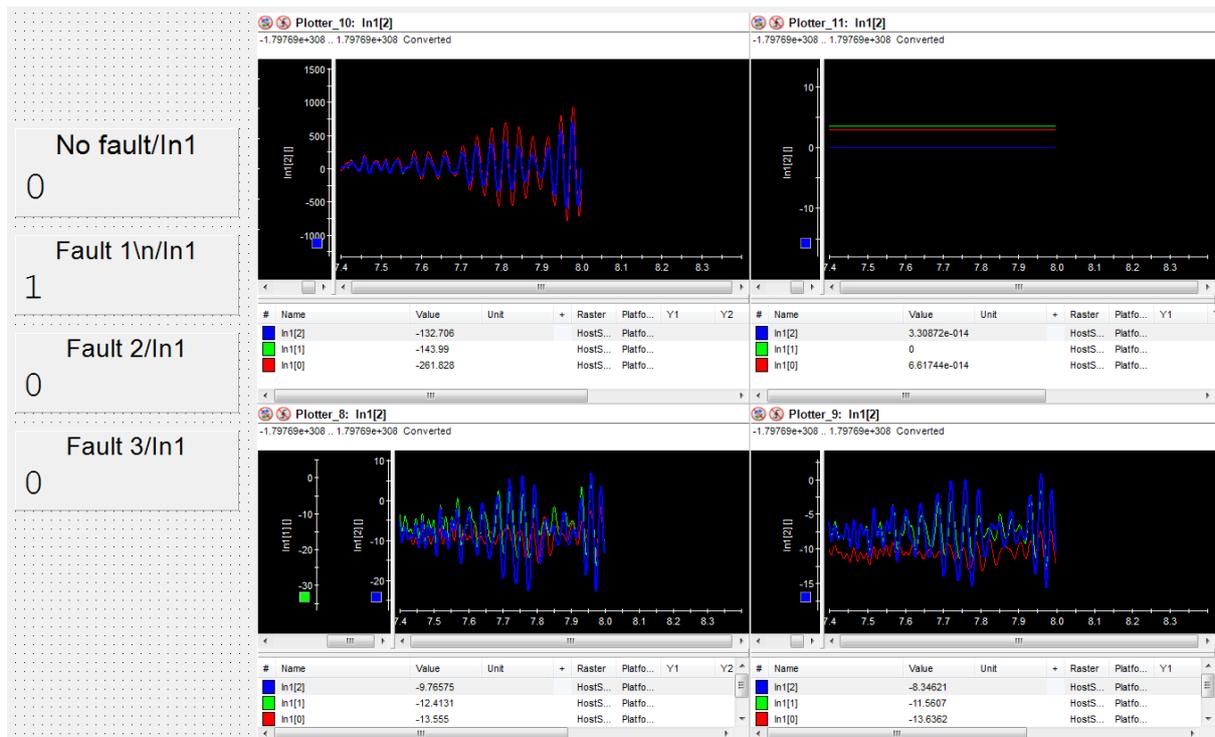


Figure 5.18 The four residuals and results with actuator fault 1 in simulation

Comparing Figure 5.18 with Figure 5.17, the three signals in the first residual changes from noises close to 0 to very large noises, while the signals in the second residual changes from very large noises to noises close to straight lines. As in the previous analysis, all results are the same compared to the simulation, except for the initial condition of the residual. In the fault detection and diagnostic results, the actuator 1 fault is displayed.

5.2.3 Actuator fault 2 system simulation result

Artificial setting actuator fault 2, observe the results of FDD and four residuals in Figure 5.19.

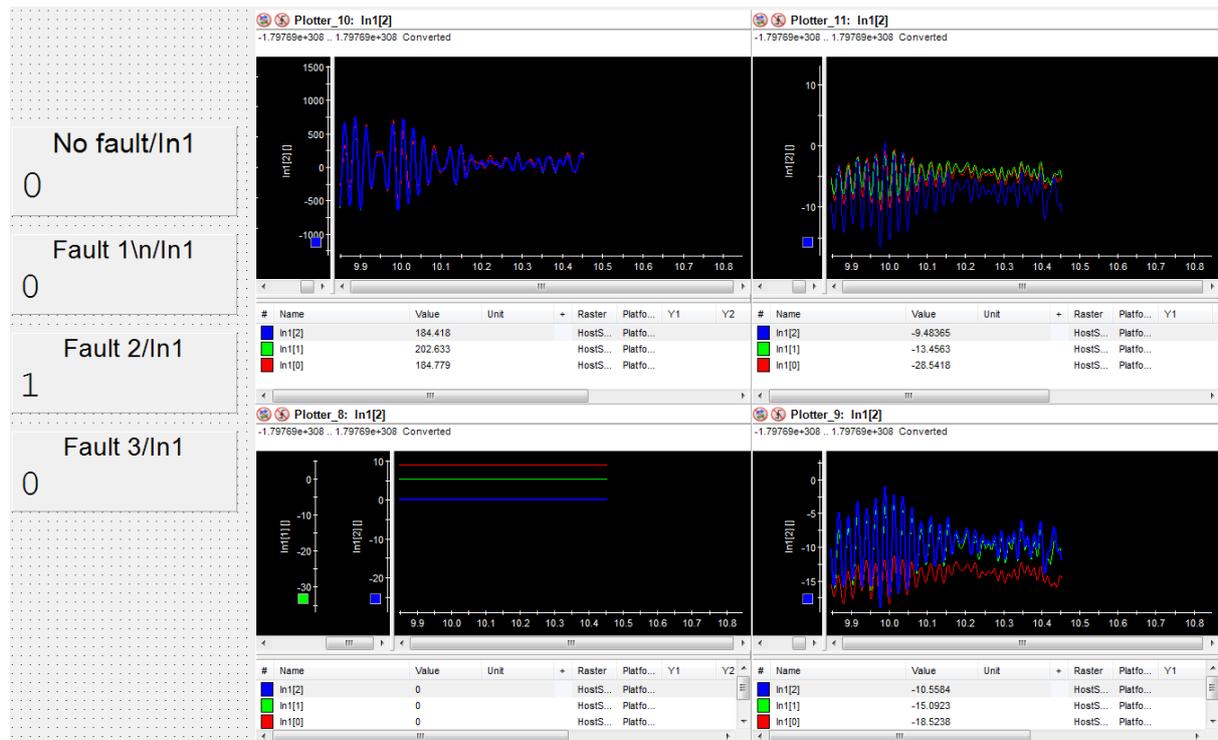


Figure 5.19 The four residuals and results with actuator fault 2 in simulation

Comparing Figure 5.19 with Figure 5.17, the three signals in the first residual change from noises close to 0 to very large noises, while the signals in the third residual change from very large noises to noises close to straight lines. As in the previous analysis, all results are the same compared to the simulation, except for the initial condition of the residual. In the fault detection and diagnostic results, the actuator 2 fault is displayed.

5.2.4 Actuator fault 3 system simulation result

Artificial setting actuator fault 3, observe the results of FDD and four residuals in Figure 5.20.

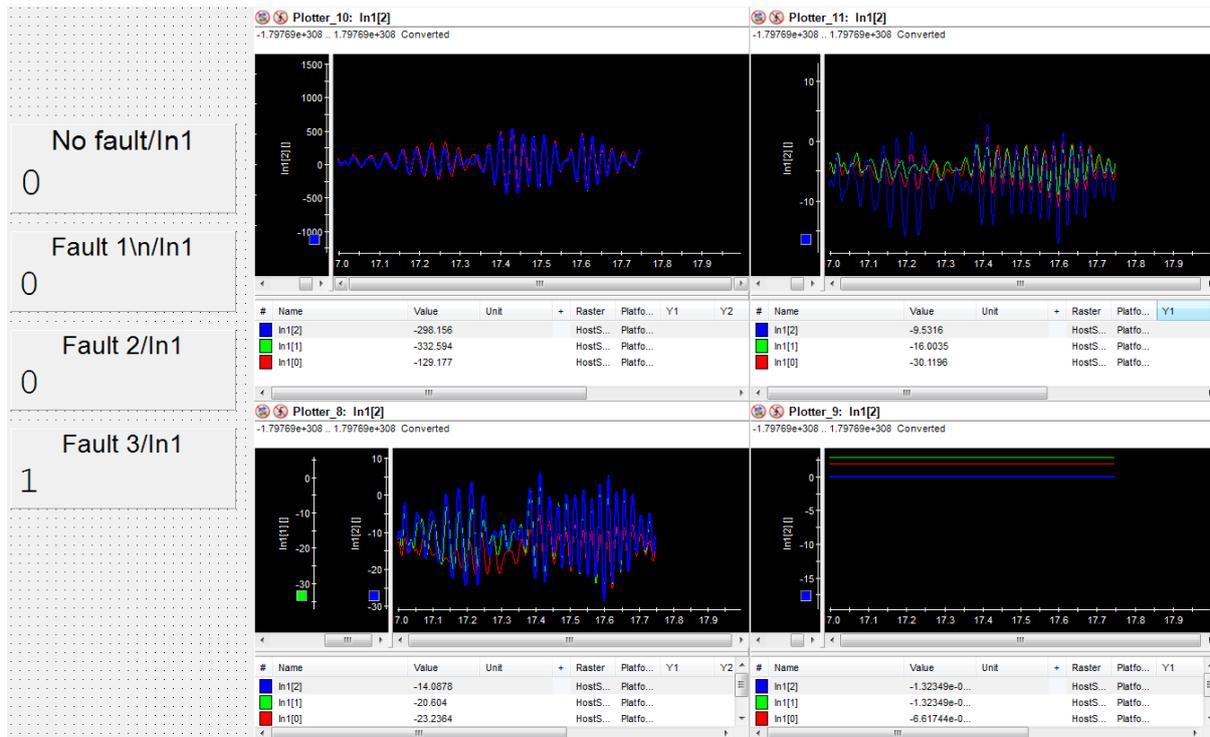


Figure 5.20 The four residuals and results with actuator fault 3 in simulation

Comparing Figure 5.20 with Figure 5.17, the three signals in the first residual change from noises close to 0 to very large noises, while the signals in the fourth residual change from very large noises to noises close to straight lines. As in the previous analysis, all results are the same compared to the simulation, except for the initial condition of the residual. In the fault detection and diagnostic results, the actuator 3 fault is displayed.

5.2.5 Actuator fault 1 and 2 system simulation result

Artificial setting actuator fault 1 and 2, observe the results of FDD and four residuals in Figure 5.21.

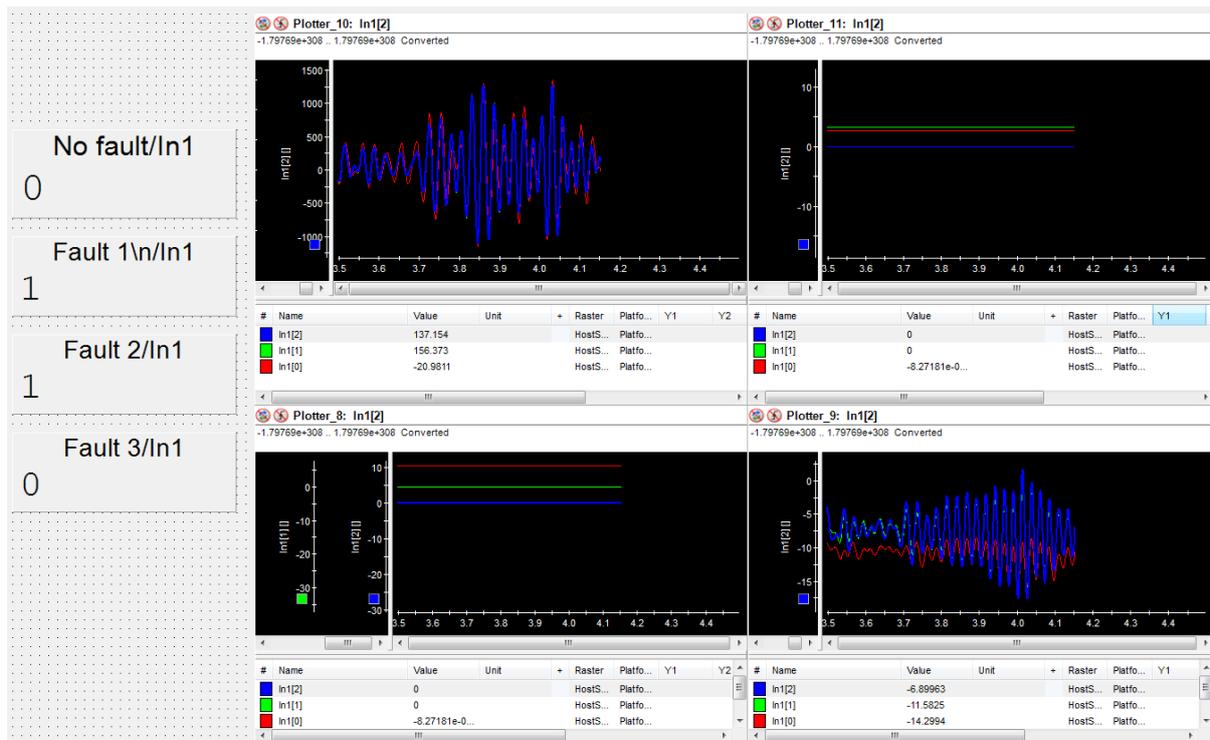


Figure 5.21 The four residuals and results with actuator fault 1 and 2 in simulation

Comparing Figure 5.21 with Figure 5.17, the three signals in the first residual changes from noises close to 0 to very large noises, while the second and third residuals change from very large noises to noises close to straight lines. As in the previous analysis, all results are the same compared to the simulation, except for the initial condition of the residual. In the fault detection and diagnostic results, the actuator 1 and 2 fault is displayed.

5.2.6 Actuator fault 2 and 3 system simulation result

Artificial setting actuator fault 2 and 3, observe the results of FDD and four residuals in Figure 5.22.



Figure 5.22 The four residuals and results with actuator fault 2 and 3 in simulation

Comparing Figure 5.22 with Figure 5.17, the three signals in first residual change from noises close to 0 to very large noises, while the third and fourth residuals change from very large noises to very small noises, in which the largest values are around 1.5×10^{-9} . However, the third and fourth residuals are still very close to straight lines. As in the previous analysis, all results are the same compared to the simulation, except for the initial condition of the residual. In the fault detection and diagnostic results, the actuator 2 and 3 fault is displayed.

5.2.7 Actuator fault 1 and 3 system simulation result

Artificial setting actuator fault 1 and 3, observe the results of FDD and four residuals in Figure 5.23.

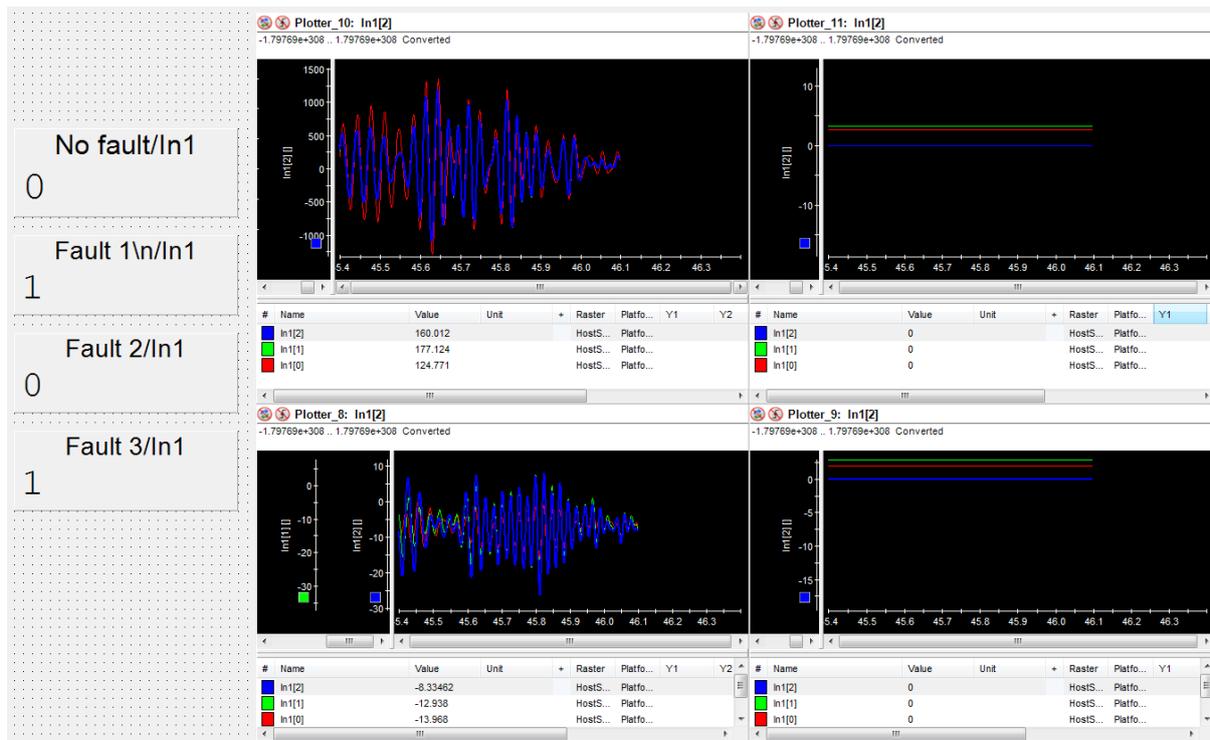


Figure 5.23 The four residuals and results with actuator fault 1 and 3 in simulation

Comparing Figure 5.23 with Figure 5.17, the three signals in the first residual changes from noises close to 0 to very large noises, while the second and fourth residuals change from very large noises to noises close to straight lines. As in the previous analysis, all results are the same compared to the simulation, except for the initial condition of the residual. In the fault detection and diagnostic results, the actuator 2 and 3 fault is displayed.

In summary, in all experiments the FDD results are exactly the same as the simulation, and only the residuals of the observations are slightly different. The reasons for the differences have also been discussed in the experiment. Therefore, the Kalman filter based FDD method has proven to be an effective method for detecting and diagnosing faults in simulations and experiments.

Chapter 6 : Conclusion

First of all, this thesis studies the MIMO plant structure system characteristics, and simplifies it into a linear discrete-time SSR. Then, the discrete-time Kalman filter is studied to produce the system state estimate. The system output estimate is obtained by using the system state estimate and the system output equation. By comparing the real system output with the estimated output, a series of residuals can be calculated to detect the similarity of the two outputs. Using the same method for other single fault cases, a Kalman filter bank is created, which included the Kalman filter of the normal system and the Kalman filter of three different single fault cases. In order to analyze the residuals, the thesis comes up with a new algorithm called normalization algorithm. And then, by observing the parameters of the simplified linear discrete-time system in the simulation, the previous theoretical variables are valued to obtain the correct FDD results. At the same time, the intermediate process signal, the residuals, are also observed in the simulation to verify the normalization algorithm. Finally, the Kalman filter based FDD method is used in the experiment. The experimental results are compared with the simulation results to further verify the feasibility of this method.

In the experiment, it is found that the existing experimental equipment has its limitations. In the existing experimental equipment, it is impossible to physically set the actuator fault and get relevant signals. However, this method also has its limitations. It requires that the theoretical model in the experiment must match the real system. Once the real system changes, the results of FDD will no longer be credible.

For the above two points, the future work can be completed in two steps. First, in the experiment, a new PCB board will be designed for the simulation between the actuator and the dSPACE. Accordingly, designing a switch on the new PCB board will simulate the actuator fault. Second, the Kalman filter based FDD method will be improved to make it more versatile, not only to perform correct FDD to required system changes (analyze a specific system and create a system library), but also detecting and diagnosing sensor faults as well as the actuator fault. After the improvement of the Kalman filter based FDD method, it will have a better performance to do FDD in other vibration control systems.

Appendix A: MATLAB Code

The MATLAB codes for each section are kept within the Advanced Control Research Group, Flinders University. The MATLAB codes can be provided upon request.

Reference

- [1] Chen, J. and Patton, R. 1999. Robust model-based fault diagnosis for dynamic systems. Boston: Kluwer. 1-20.
- [2] Zhang, Y. and Jiang, J., 2008. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual reviews in control*, 32(2), pp.229-252.
- [3] Frank, P.M., 1996. Analytical and qualitative model-based fault diagnosis—a survey and some new results. *European Journal of control*, 2(1), pp.6-28.
- [4] Venkatasubramanian, V., Rengaswamy, R., Kavuri, S.N. and Yin, K., 2003. A review of process fault detection and diagnosis: Part III: Process history based methods. *Computers & chemical engineering*, 27(3), pp.327-346.
- [5] Uraikul, V., Chan, C.W. and Tontiwachwuthikul, P., 2007. Artificial intelligence for monitoring and supervisory control of process systems. *Engineering Applications of Artificial Intelligence*, 20(2), pp.115-131.
- [6] Shui, A., Chen, W., Zhang, P., Hu, S. and Huang, X., 2009, June. Review of fault diagnosis in control systems. In *Control and Decision Conference, 2009. CCDC'09. Chinese* (pp. 5324-5329). IEEE.
- [7] Isermann, R., 2005. Model-based fault-detection and diagnosis—status and applications. *Annual Reviews in control*, 29(1), pp.71-85.
- [8] Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, M. and Schröder, J., 2006. *Diagnosis and fault-tolerant control* (Vol. 2). Berlin: springer., 8-18.
- [9] Sharma, A.B., Golubchik, L. and Govindan, R., 2010. Sensor faults: Detection methods and prevalence in real-world datasets. *ACM Transactions on Sensor Networks (TOSN)*, 6(3), p.23.
- [10] Jan, S.U., Lee, Y.D., Shin, J. and Koo, I., 2017. Sensor fault classification based on support vector machine and statistical time-domain features. *IEEE Access*, 5, pp.8682-8690.
- [11] Hrizi, O., Boussaid, B., Zouinkhi, A. and Abdelkrim, M.N., 2015. Robust unknown input observer-based fast adaptive fault estimation: Application to mobile robot. In *Handbook of Research on Advanced Intelligent Control Engineering and Automation* (pp. 427-456). IGI Global.

Reference

- [12] Hwang, I., Kim, S., Kim, Y. and Seah, C.E., 2010. A survey of fault detection, isolation, and reconfiguration methods. *IEEE transactions on control systems technology*, 18(3), pp.636-653.
- [13] Dubrova, E. 2013. *Fault-tolerant design*. New York: Springer. pp.55-65.
- [14] Beck, J. V., & Arnold, K. J. 1977. *Parameter estimation in engineering and science*. James Beck. pp.1-10.
- [15] Nguang, S. K., Zhang, P., & Ding, S. X. 2007. Parity relation based fault estimation for nonlinear systems: An LMI approach. *International Journal of Automation and Computing*, 4(2), 164-168.
- [16] Aitouche, A., Yang, Q. and Bouamama, B.O., 2011. Fault detection and isolation of PEM fuel cell system based on nonlinear analytical redundancy: An application via parity space approach. *The European Physical Journal-Applied Physics*, 54(2).
- [17] De Persis, C., & Isidori, A. (2001). A geometric approach to nonlinear fault detection and isolation. *IEEE transactions on automatic control*, 46(6), 853-865.
- [18] Massoumnia, M. A. 1986. A geometric approach to the synthesis of failure detection filters. *IEEE Transactions on automatic control*, 31(9), 839-846.
- [19] Bar-Shalom, Y., Li, X.R. and Kirubarajan, T., 2004. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 199-536.
- [20] Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1), pp.35-45.
- [21] Chui C K, Chen G. 2008. *Kalman Filtering with Real-time Application*, Fourth Edition, Berlin: Springer, 108-177
- [22] Ozbek, L. and Efe, M., 2004. An adaptive extended Kalman filter with application to compartment models. *Communications in Statistics-Simulation and Computation*, 33(1), pp.145-158.
- [23] Foo, G.H.B., Zhang, X. and Vilathgamuwa, D.M., 2013. A sensor fault detection and isolation method in interior permanent-magnet synchronous motor drives based on an extended Kalman filter. *IEEE Transactions on Industrial Electronics*, 60(8), pp.3485-3495.

Reference

- [24] Huang, G., Mourikis, A.I. and Roumeliotis, S.I., 2008, May. Analysis and improvement of the consistency of extended Kalman filter based SLAM. In ICRA (pp. 473-479).
- [25] Julier, S.J., Uhlmann, J.K. and Durrant-Whyte, H.F., 1995, June. A new approach for filtering nonlinear systems. In American Control Conference, Proceedings of the 1995 (Vol. 3, pp. 1628-1632). IEEE.
- [26] Julier, S., Uhlmann, J. and Durrant-Whyte, H.F., 2000. A new method for the nonlinear transformation of means and covariances in filters and estimators. IEEE Transactions on automatic control, 45(3), pp.477-482.
- [27] Zhang, P. and He, F. (2017). Experimental implementation of a multi-input multi-output active vibration control on vibration isolation systems. In: 24th International congress on sound and vibration. London: London Calling.
- [28] Zhang, P. and He, F. (2017). MIMO PPF Active Vibration Control of Asymmetrical Plate Structures. In: The 13th IEEE Conference on Industrial Electronics and Applications. Wuhan.
- [29] Meinhold, R.J. and Singpurwalla, N.D., 1983. Understanding the Kalman filter. The American Statistician, 37(2), pp.123-127.
- [30] Simon, D., 2006. Optimal state estimation: Kalman, H infinity, and nonlinear approaches. John Wiley & Sons.