Doctor of Philosophy

Real Implementation of Fault-Tolerant Control via Synchronous Methodology for Robot Manipulators

The Graduate School of the University of Ulsan

Department of Electrical, Electronic and Computer Engineering

LE, QUANG DAN

Real Implementation of Fault-Tolerant Control via Synchronous Methodology for Robot Manipulators

Supervisor: Professor KANG, HEE-JUN

A Dissertation

Submitted to

The Graduate School of the University of Ulsan

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

by

LE, QUANG DAN

Department of Electrical, Electronic and Computer Engineering

University of Ulsan

May 2022

Le Quang Dan 의 공학 박사 학위 논문을 인준함

심사위원장   김한실   (인)

심사위원   조강현   (인)

심사위원   서영수   (인)

심사위원   강현덕   (인)

심사위원   강희준   (인)

울산대학교 대학원
전기전자컴퓨터공학과
2022 년 8 월

Real Implementation of Fault-Tolerant Control
via Synchronous Methodology for Robot Manipulators

This certifies that the dissertation thesis of LE, QUANG DAN is approved.

Committee Chair: Prof. Kim Han-Sil

Committee Member: Prof. Sul Young Soo

Committee Member: Prof. Jo Kang-Hyun

Committee Member: Prof. Kang Hyun-Deok

Committee Member: Prof. Kang Hee-Jun

Department of Electrical, Electronic and Computer Engineering

University of Ulsan

May 2022

Abstract

During the past few decades, fault-tolerant control has been introduced to enhance safety and acceptable performance when faults occur in the system. In this dissertation, a real implementation of fault estimation (FE) and fault-tolerant control (FTC) for a robot manipulator based on synchronous technique is presented. First, the forward kinematics of the robot manipulator was investigated, and then, inverse kinematics problem was studied using both Pieper and numerical methods. In this study, Lagrangian method was used for the dynamic model of robot manipulator which plays an important role in FE and FTC. Moreover, the mathematical definition of faults has been showed. However, the exact dynamic model of robot manipulator is difficult to determine in a real system. Therefore, the least-square method was applied to identify the dynamic parameters of a real robot manipulator. After that, the point-to-point control and force control were implemented to verify the effectiveness of the hardware setup as well as the dynamic model of a robot manipulator.

To implement fault-tolerant control, the estimator was designed to estimate and identify the faults within one step. The extended state observer (ESO), which is easy to implement and fast to compute, was used to estimate the lumps of uncertainties, disturbances and faults. The active fault-tolerant control (AFTC) based on the ESO and super twisting sliding mode control was evaluated for the effectiveness of AFTC. By using the conventional controller in AFTC, the system responds slowly and the performance decreases when faults occur due to the picking phenomenal effect of the FE and slow response of AFTC. However, the AFTC has capable of handling the high magnitude of faults which common in real systems. Therefore, we proposed the AFTC with synchronous sliding mode control (AFTC-SSMC). Due to the ability to make the error at each joint simultaneously approach to zero, the proposed control can reduce the impact of picking phenomena and provide fast recovery compare to the conventional

controller. Moreover, due to the constraint inside the synchronous control, the system is able to deal with high magnitude faults. The next proposed AFTC has been developed with the ability to converge in finite-time of controller. In this proposed AFTC, the synchronous terminal sliding mode control (S-TSMC) with ESO has been combined with two novel synchronization position errors. The first proposed synchronization position error is based on the combination of novel synchronization error and coupling position error. The second proposed synchronization position error uses only the novel synchronization position error to reduce the disadvantage of conventional coupling position error in real system. All the proposed AFTCs are proved to be stable based on Lyapunov theory. To implement the proposed AFTC in a real robot manipulator system, a simulation model of the robot was created in Solidworks and then imported into Matlab Simulink to verify the proposed AFTC. After applying to the simulation system, the proposed AFTC was implemented on SAMSUNG 3-DOF FARA-AT2 robot manipulator using FPGA-Labview NI-PXI.

To my Parents

To my brother's family

Acknowledgements

First and foremost, I would like to express my great gratitude to my Advisor, Professor **Kang, Hee-Jun** *for his supports, encouragements, suggestions and comments during my study at the University of Ulsan.*

I would like to send my special thanks to Prof. Kim Han-Sil, Prof. Jo Kang-Hyun, Prof. Sul Yuong Soo, Prof. Kang Hyun-Deok for spending their time reading my dissertation and giving me the valuable advises in my dissertation defense which help me improve my dissertation quality.

A special note of thanks goes to my colleagues, particularly my members of the Intelligent Robotic System Laboratory and senior lab members: Dr. Le Tien Dung, Dr. Hoang Ngoc Bach, Dr. Tran Xuan Toa, Dr. Hoang Duy Tang, Dr. Le Phu Nguyen, Mr. Nguyen Van Cuong, Mr. Truong Thanh Nguyen for their help and encourage me whenever I need.

Many thanks are given to my friends in University of Ulsan.

I thank my parents, my brother's family, and my relatives' families for their constant support, encouragement, love and words of confidence. My parents are the ones who made it possible to start this journey, and the whole family has always been there for me.

May, 2022, Ulsan, Korea

Le Quang Dan

Table of Contents

List of figures

List of tables

Chapter 1: Introduction

1.1 General

Nowadays, in order to increase the profit, the quality of the products, safety and stability, the automation systems are designed to be near optimally and the theory of advantage control is applied. However, the more processes are automated, the greater the risk that the equipment in the systems will occur more faults. As a result, the quality of the product decreases because the performance of the system degrades and the system becomes unstable. This has a significant impact on profitability. Moreover, in highly automated industrial systems, it is difficult to perform warranty and maintenance immediately. They must be finished on a schedule. Therefore, the system must be able to remain performance and predictable when faults occur. Fault estimation and fault-tolerant control become an interesting topic in past few decays because they can predict and tolerate the occurrence of faults in systems.

With the advancement of technology, robot manipulators are increasingly used in industry and daily life. Robot manipulator become friendly with human and easier interaction with operator in both industry and daily life. The robot manipulators are not only stable and highly accurate in the production line, but also have the ability to share the workspace and cooperate with humans to complete complex tasks. The robot manipulator can work in all situations to maintain the safety, reliability and acceptable performance. These features are key to the success of the model robot manipulator. In next Section, the definition of fault, failure, malfunction, fault estimation and fault-tolerant control were presented. Previous work and the motivation for this research were described in Section 1.3. Finally, the contribution and outline of the work have been shown in Section 1.4.

1.2 Fault, Failure and Malfunction

1.2.1 Fault

A fault is an unpermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual, standard condition.

Remarks:

- A fault is a state within the system.

- The unpermitted deviation is the difference between the fault value and the violated threshold of a tolerance zone for its usual value.

- A fault is an abnormal condition that may cause a reduction in, or loss of, the capability of a functional unit to perform a required function.

- There exist many different types of faults, design fault, manufacturing fault, assembling fault, normal operation fault (wear), wrong operation's faults (overload), maintenance-fault, hardware-fault, software-fault, operator's fault (some of these faults are also called errors, especially if directly cause by humans).

- A fault in the system is independent of whether the system is in operation or not.

- A fault may not affect the correct functioning of a system (like a small rent in an axle).

- A fault may initiate a failure or a malfunction.

- Frequently, faults are difficult to detect, especially, if they are small or hidden.

- Faults may develop abruptly (stepwise) or incipiently (drift-wise).

1.2.2 Failure

A failure is a permanent interruption of a system's ability to perform a required function under specified operating conditions.

Remarks:

- A failure is the termination of the ability of a functional unit to perform a required function

- A failure is an event

-A failure results from one or more faults

- Different types of failures can be distinguished

    + Number of failures: single, multiple

    + Predictability

- Random failure (unpredictable statistically independent from operation time or other failures)

- Deterministic failure (predictable for certain conditions)

- Systematic failure or causal failure (dependent on known conditions)

- Usually, a failure arises after begin of operation or by increasingly stressing the system.

1.2.3 *Malfunction*

A malfunction is an intermittent irregularity in the fulfilment of a system's desired functional.

Remarks:

- A malfunction is a temporary interruption of a system's functional

- A malfunction is an event

- A malfunction results from one or more faults

- Usually, a malfunction arises after begin of the operation or by increasingly stressing the system.

The fault may develop abruptly, like a step-function, or incipiently, like a drift-like function. The corresponding feature of the system related to the fault is assumed to be proportional to the fault development. After exceeding the tolerant of normal values, the feature indicates a fault. After exceeding the tolerance of normal values, the feature indicates a fault. Dependent on its size, a failure or a malfunction of the system follow at time.
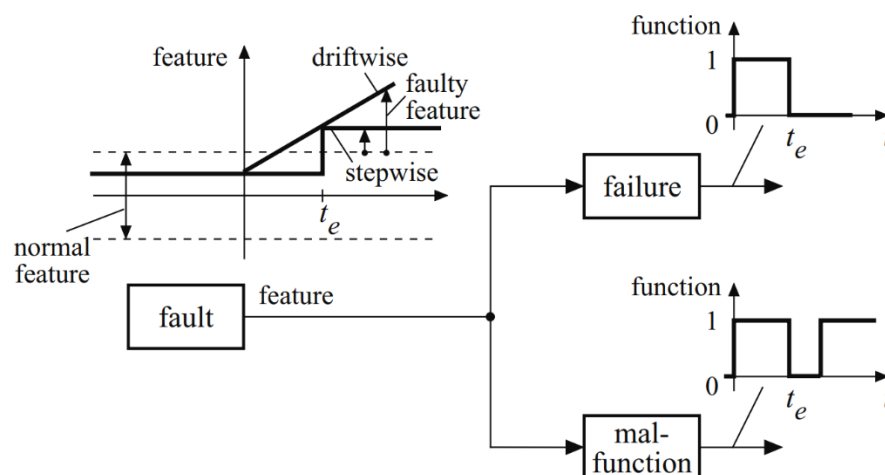
Figure 1.1. Development of the event "failure" or "malfunction" from a fault which causes a stepwise or drift-wise change of a feature.

In Figure 1.1, the development of failure and malfunction from a fault is presented. Therefore, early prediction of failures is very useful to maintain the continuous

operation of the system. In Table 1.1, the critical components, failure modes and the possible solutions for each mode are given.

Table 1.1. Critical components and failure modes

Component	Failure modes	Possible solutions
Joint sensor	-Frozen -Biased -Run-away -Spike	-Ignore sensor and find alternate data -Remove bias from results -Ignore sensor and find alternate data -Ignore spike value
Joint motor	-Frozen (constant output) -Zero (on output) -Run-away -Random (unknown/variant output)	-Halt motor, redistribute work load -Halt motor, redistribute work load -Counteract if have dual-motor -Halt motor, redistribute work load
Power supply	-Surge -Zero	-Surge protector -Back-up power supply
Control computer	-Software error: + invalid command + wrong value -Hardware error	-Ask operator for new command -Use alternate software modules -Back-up processors or redistribute to parallel processors

1.3 Fault estimation

In conventional system, fault diagnosis system is capable of performing the three tasks of detection, isolation, and identification of faults, which are defined as follows:

- Fault detection: To indicate the occurrence of a fault in a monitored system.

- Fault isolation: To determine the location of the fault and/or identifies which component, sensor, or actuator has become faulty.

- Fault identification: To estimate the magnitude of the fault.

4

Unlike traditional fault diagnosis, fault detection only shows the binary result. Nowadays, with the development of computers, the observer based on the dynamic model can detect the location and magnitude of faults in robot manipulators within one step, which is called Fault Estimation. In this thesis, fault estimation was used in the active fault-tolerant architecture.

1.4 Fault-tolerant

1.4.1 Hardware redundancy

This method is based on the hardware complement to fault tolerance. Hardware complementation for fault tolerance is widely used in fields such as aerospace, aircraft, and nuclear power systems. Hardware redundancy can consist of sensors, actuators, process parts, computers and communication systems. They are integrated into the system, in parallel. When faults are detected, hardware redundancy is automatically activated. This solution maintains high reliability, but increases cost and reduces payload due to the structure and weight of the systems change.

1.4.2 Dynamic redundancy

This method is based on the dynamic model of robot to detect and tolerate faults. With this technique, faults in sensors, actuators and controller can be tolerated. It can keep the acceptable performance. The operator can schedule maintenance and cost do not increase when compared to the hardware redundancy method. For example, in case, faults occur at actuators. Using the estimator based on the dynamic model, we can estimate the magnitude of faults and detect where faults occur. The information from estimator is compensated with main controller to tolerant faults. This scheme had been called active fault-tolerant control. Another scheme was called passive fault-tolerant control, which do not require the feedback from estimator to dealing with faults. This scheme based on the ability to handle uncertainties and disturbances from the main controller to tolerate faults. Both schemes have significant advantages. Passive fault-tolerant control response quickly when faults occur but has limit ability to deal with large faults and multiple faults. Active fault-tolerant, on the other hand, response slowly, but it has high ability to deal with high magnitude and multi faults. Depending on the characteristics of system, each scheme will be selected. In this thesis, the active fault-tolerant control was studied and developed.

1.5 Previous works and motivation

These the days, robot manipulators play the important role in the manufacturing industries and daily life. With the development of hardware and advantage technique control, robot manipulators are required to increase the safety for all situations. The safety plays a key role in the case of robots sharing the workspace with humans or cooperating with humans. Therefore, fault-tolerant control (FTC) was introduced to enhanced safety and acceptable performance when faults occur in the system. Generally, AFTC can be divided into two main types 1) active FTC (AFTC) [1–3] and 2) passive FTC (PFTC) [4–6]. Each FTC strategy has advantages and disadvantages defending on the characteristics of the system, the knowledge about that system and the type of faults.

In robot manipulator systems, PFTC has a simple architecture with one controller for both normal and fault conditions. PFTC quickly responds to faults due to the elimination of a fault diagnosis process. Some techniques have been classified into PFTC such as sliding mode control (SMC) [7, 8], adaptive control [9, 10] and so on. For instance, the SMC [8, 11] is widely known due to its robustness and the ability to deal with uncertainties and disturbances. In order to guarantee the stability of systems, the design of SMC required the upper bound knowledge of faults which is difficult to know in a real system. Therefore, PFTC with SMC has less flexible and limited to tolerate the faults capability in real systems. The second PFTC technique is adaptive control which does not need exact knowledge of faults, but the excessive adaptation rate is a problem of with this kind of controller. From these analyses above, it can be seen that PFTC had the advantage in the case of the systems with a good knowledge about the upper bound of uncertainties, disturbances and faults. In contrast, to guarantee the stability and acceptable performance of a system, AFTC used fault detection, and fault diagnosis (FDD) to compensate for the normal controller. In AFTC, the FDD process is a very important process which leads to increasing the ability of tolerating faults. Therefore, FDD [12–15] has been developed with several techniques. The advantage of AFTC strategy is it has the high ability to deal with the high magnitude faults and multiple faults. In addition, the upper bound of faults does not require to be exactly known in the same way as PFTC does. However, this strategy slowly responds to faults because the FDD process needs time to feedback the information of faults as well as the time delay problem in a real system. Hence, the performance of AFTC was affected.

In a real system, the degradation performance of AFTC can be obtained by the occurrence of the picking phenomenon due to the slow response of AFTC and high

magnitude of uncertainties and disturbances. In robot manipulator systems, the FDD has been widely replaced by the fault estimation process (FE) which uses the fault detection, fault isolation and fault diagnosis within one step. The structure controller has the combination with the estimator as known disturbance observer-base (DOB). The difference of DOB and AFTC with the estimator is in AFTC with an estimator, the output value was used to compare the threshold to detect fault and isolation faults. It warns the operator about the occurrence of faults to make the effective action. The AFTC with structure of DOB control (AFTC-DOB)[16–18] may have the effect in the case of bias faults. This kind of fault can be considered as uncertainties or disturbances so that AFTC-DOB can show the acceptable performance depending on the accuracy of the estimator. However, with the loss-effective fault or the combination of loss-effective and bias faults, using merely AFTC-DOB may not reduce the effect of this kind of fault. Hence, AFTC need a different way to handle the faults due to the occurrence of lost effect faults. Most of the research studies about FTC [1, 7, 19, 20] focus on the development of the ability to deal with the uncertainties, disturbances or handle the conventional problem of controller such as the chattering phenomenon or fast convergence problem of controller. However, they lack the picking phenomenon of AFTC strategy which the most impact degrades the performance of AFTC. Therefore, in this dissertation, the synchronization technique was proposed to suppress the picking phenomenon, fast response and dealing with high magnitude of faults when faults occur, especially with the occurrence of lost effect faults for AFTC strategy.

In the field of control, the synchronization control has been widely known with a close-loop chain mechanism such as dual drive gantry mechanism [21], parallel robot manipulator [22], cable-driver parallel manipulator [23], cooperation robot manipulator [24] and so on. These systems have the internal tensor force during the motion due to the close-loop mechanism. This kind of force can degrade the performance of systems. By using the synchronization technique, the controller makes the position errors at each joint simultaneously approach to zero and reduce the internal tensor force. Therefore, the accuracy of systems can be increased. Unlike a close-loop mechanism, a serial robot manipulator does not have the occurrence of internal tensor force. Due to the open-loop mechanism, this internal uncertainty can be compensated by the consideration of a dynamic model. Therefore, the synchronization technique does not show the advantage compared the model-based controller such as DOB in normal operation. However, the ability to make the position error simultaneously equal and approach to zero still

maintains in an open-loop mechanism with the synchronization technique. In case that faults occur, the internal constraints of the synchronization control to keep the error at each joint equal may suppress the effect of faults. The synchronization controller increases the ability to quickly respond to the controller before it gets the feedback information of faults from FDD or FE. Due to the internal constraints of position error at each joint of the synchronization technique, the proposed controller can reduce the picking effect and fast response before the FE gets the information of faults, increase the ability to deal with high magnitude faults and fine-time convergence to make the robot manipulator system acceptable performance.

1.6 Thesis contribution and Outline

The main contributions of this dissertation can be summarized as follows:

➢ Chapter 2: The fundamentals of robot manipulators and faults were presented, including forward kinematics, inverse kinematics, dynamic model and mathematics of faults. In particular, the inverse kinematics of robot manipulators was investigated using both geometric and numerical methods. In the numerical methods, Jacobian-base and Cyclic coordinate descent were theoretically represented and implemented in C-Shape. The inverse kinematics software can be used to compute inverse kinematics for any serial robot manipulators by filling the parameters of the D-H table.

➢ Chapter 3: Hardware development using the FARA-AT2 robot manipulator in torque control mode is presented. First, the hardware setup base on Labview with NI-PXI computer was shown. To verify the hardware setup, the point-to-point controller for 6-DOF FARA-AT2 robot manipulator was shown using Labview. Next, to apply the torque control mode for robot manipulator, the dynamics parameters of 3-DOF robot manipulator were shown by using the least squares method. Two types of force control and computed torque control were applied to the robot manipulator to verify the identification processing results. The results with computed torque control shows the effectiveness of dynamic model in improving the accuracy tracking trajectory of robot manipulators. Moreover, the admittance control without force sensor not only demonstrated the importance of the dynamic model in controlling a robot manipulator but also showed the potential of robot manipulator for human-robot interaction research.

➢ Chapter 4: In this chapter, the fault-tolerant control for robot manipulator based on the combination of the extended state observer and supper-twisting method is proposed. By using the extended state observer, we can easily monitor the faults and no need the knowledge about bound of uncertainties/disturbances and faults. The faults were

compensated in the supper-twisting controller. By using this combination, the accuracy of robot is increased and is shown acceptable performance when faults occur. The simulation and experimental results are shown for a 3-DOF robot manipulator to illustrate the effectiveness of the proposed fault-tolerant control.

➢ Chapter 5: In this chapter, an active fault-tolerant control for a robot manipulator based on synchronous sliding mode is proposed. As the synchronization errors approach zero, the joint errors tend to become equal and also approach zero. Therefore, the synchronization technique is inherently effective for a fault-tolerant controller. To demonstrate such a system, the following implementation is presented. First, an estimator was designed with an extended state observer to estimate uncertainties/disturbances along with faults/failures. The estimator signal was used for an online compensator in the controller. A fault-tolerant controller with a combination of synchronous sliding mode technique and estimator was proposed. The stability of the system was established using Lyapunov theory. Finally, fault-tolerant control was implemented in a three degree-of-freedom robot manipulator and compared to the conventional sliding mode control. This comparison shows the effectiveness of the proposed active fault-tolerant control with synchronous sliding mode technique. The contributions of this chapter are summarized as follows:

(1) Synchronization techniques are applied to fault-tolerant control for robot manipulators for the first time. Compared to active fault-tolerant control using conventional sliding mode control, the proposed system has achieved higher accuracy, robustness, and faster system reconfiguration when faults occur. These results confirm that synchronization techniques are very effective in fault-tolerant control.

(2) The stability of the proposed AFTC with the synchronous sliding mode technique is demonstrated using analysis via Lyapunov theory.

(3) Based on the extended state observer, the proposed controller can easily monitor faults without detection and isolation processes. This feature is helpful in maintenance systems as well as maintenance planning systems.

➢ Chapter 6: In this chapter, two finite-time active fault-tolerant controllers for a robot manipulator, which combine a synchronous terminal sliding mode control with an extended state observer, are proposed. First, an extended state observer is adopted to estimate the lumped uncertainties, disturbances, and faults. The estimation information is used to compensate the controller designed in the following step. We present an active

fault-tolerant control with finite-time synchronous terminal sliding mode control, largely based on a novel finite-time synchronization error and coupling position error. We also present an active fault-tolerant control that does not use a coupling position error. By using synchronization control, the position error at each joint can simultaneously approach toward zero and toward equality, which may reduce the picking phenomenon associated with the active fault-tolerant controller strategy. Finally, simulation and experimental results for a three degree-of-freedom robot manipulator verify the effectiveness of the two proposed active fault-tolerant controllers. The contributions of this chapter are summarized as follows:

(1) Two active fault-tolerant control algorithms for robot manipulators, based on novel finite-time synchronous terminal sliding mode controllers and an extended state observer, are proposed. The novel finite-time synchronization technique has the ability to make both the joint position error and the synchronization error simultaneously approach to zero. Due to these internal constraints of the synchronization control, the proposed controller can make the system quickly respond to the faults in a forward way before its feedback response after a fault estimation. Therefore, the proposed controller can reduce the occurrence of the picking phenomenon due to the slow feedback response of AFTC strategy.

(2) The novel synchronization error leads to better synchronization because it uses more information from other joints than the conventional synchronization error in [77],which has the information from only one neighbour joint.

(3) The novel coupling error can make the position error approach zero in a finite-time while the conventional coupling error in [76] cannot guarantee the finite-time convergence.

(4) Two proposed AFTCs can avoid the singularity in both the desired trajectory and the control action, while the control algorithms in [26–28] cannot guarantee avoiding the singularity. This ability allows increase in the workspace of the robot.

(5) Experimental results show the effectiveness of the proposed AFTC in reducing both the picking phenomenon and in handling faults of high magnitude.

Chapter 2: Kinematics, Dynamics of Robot Manipulators and Mathematical of Faults

2.1 Introduction

In this chapter, kinematics and dynamics of the FARA-AT2 robot manipulator were studied. In addition, the mathematics of faults was presented. In subsection 2.2, the kinematics of robot manipulator is considered, where the position and orientation of the tool are computed relative to the user's work station when given the joint angles of the robot manipulator. Next, in subsection 2.3, the motion of the robot is analysed based on the notions of linear and angular velocity. It turns out that the study of velocities leads to a matrix entity called the Jacobian of the manipulator. Subsection 2.4, the more difficult converse problem will be presented: Given the desired position and orientation of the tool relative to the station, how do we compute the set of joint angles which will achieve this desired result. The solution of this problem is of fundamental importance in order to transform the motion specifications, assigned to the end effector in the operational space, in to the corresponding joint space motions that allow execution of the desired motion. This problem is known as the inverse kinematics of the robot manipulators. In this subsection, three methods of inverse kinematics are presented: Pieper method, Jocobian-base method and cyclic coordinate descent (CCD) method. After considering the static positions and velocities, we address the Eq. of motion for the robot manipulator, the way in which motion of the manipulator arises from torques applied by the actuators for from external forces applied to the manipulator in subsection 2.5. Finally, the mathematical of faults was shown in subsection 2.6.

2.2 Forward Kinematics

In this subsection, some of definition of kinematics was defined then the forward kinematics of FARA-AT2 robot manipulator was presented.
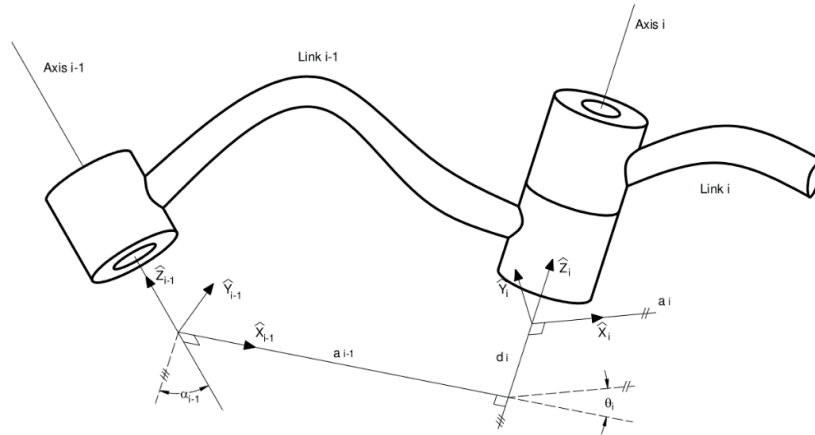
Figure 2.1. Link frames are attached so that frame {i} is attached rigidly to link i

As introduced in [27], typical robot manipulators consist of a series of rigid bodies (Links)connected by means of kinematic pairs or joints. The links are numbered starting from the robot base, which might be called link 0. The first moving body is link 1, and so on, the last rigid body is called link n.

Joint axis: are defined by lines in space. Joint axis i is defined by a line in space or a vector direction, about link i rotates relative to link $i-1$.

Link length: The distance is measured along the line that is mutually perpendicular to both axes.

Link twist: If we imagine a plane whose normal is the mutually perpendicular line just constructed, we can project the axes $i-1$ and i onto this plane and measure the angle form $i-1$ to axis i in the right-hand sense about a_{i-1}. This angle is defined as the twist of link $i-1$, α_{i-1}. Neighbouring links have a common joint axis between them.

Link offset: parameter of interconnection has to do with the distance along this common axis form one link to the next link. The link offset at joint i is called d_i.

Joint angle: the amount of rotation about this common axis between one link and its neighbour. In the case of a revolute joint, θ_i is called the Joint variable.

The definition of mechanisms by means of these quantities is a convention usually called the Denavit-Hartenberg notation. The following definition of the link parameters are valid

a_i = the distance from $\hat{Z}_i$ to $\hat{Z}_{i+1}$ measured along $\hat{X}_i$;

α_i = the angle $\hat{Z}_i$ to $\hat{Z}_{i+1}$ measured about $\hat{X}_i$;

d_i = the distance from $\hat{X}_{i-1}$ to $\hat{X}_i$ measured along $\hat{Z}_i$;

12

θ_i = the angle from $\hat{X}_{i-1}$ to $\hat{X}_i$ measured about $\hat{Z}_i$;

The Fara-AT2, which is used in this dissertation, is a robot with six degrees of freedom and all rotational joints. It is show in Figure 2.2 with link-frame assignments follow the procedure presented above. The link parameters corresponding to robot configuration shown in Figure 2.2 are shown in Table 2.1.

Table 2.1. Link Parameters of FARA-AT2 robot manipulator

	α_{i-1} (degree)	a_{i-1} (mm)	d_i (mm)	θ_i (degree)
1	0	0	0	θ_1
2	90	150	0	$\theta_2 + 90$
3	0	255	0	θ_3
4	90	96	300	θ_4
5	-90	0	0	θ_5
6	90	0	0	θ_6

Figure 2.2. Kinematic parameters and frame assignments for the FARA-AT2 robot manipulator.

The link transformation is generally obtained as follow

$$^{i-1}_iT = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ (2.1)

where the notation 's' is sine and 'c' refers to the cosine.

Using (2.1), we compute each of link transformations

$$
{}^0_1T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ; \quad
{}^1_2T = \begin{bmatrix} s\theta_2 & -c\theta_2 & 0 & a_1 \\ 0 & 0 & -1 & 0 \\ c\theta_2 & s\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ; \quad
{}^2_3T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & a_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ;
$$

$$
{}^3_4T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & a_3 \\ 0 & 0 & -1 & -d_4 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ; \quad
{}^4_5T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_5 & -c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ; \quad
{}^5_6T = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_6 & c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Forward kinematic solution is determined as

$$
{}^0_7T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T\,{}^6_7T \tag{2.2}
$$

Finally, we obtain the product of all six links transform

$$
{}^0_6T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.3}
$$

where

$$
r_{11} = s_1(c_4 s_6 + s_4 c_5 c_6) - c_1(c_6(s_{23} s_5 - c_{23} c_4 c_5) + c_{23} s_4 s_6) \tag{2.4}
$$

$$
r_{12} = s_1(c_4 c_6 - s_4 c_5 s_6) + c_1(s_6(s_{23} s_5 - c_{23} c_4 c_5) - c_{23} s_4 c_6) \tag{2.5}
$$

$$
r_{13} = c_1(s_{23} c_5 + c_{23} c_4 s_5) + s_1 s_4 s_5 \tag{2.6}
$$

$$
r_{21} = -s_1(c_6(s_{23} s_5 - c_{23} c_4 c_5) + c_{23} s_4 s_6) - c_1(c_4 s_6 + c_5 c_6 s_4) \tag{2.7}
$$

$$
r_{22} = s_1(s_6(s_{23} s_5 - c_{23} c_4 c_5) - c_{23} s_4 c_6) - c_1(c_4 c_6 - c_5 s_6 s_4) \tag{2.8}
$$

$$
r_{23} = s_1(s_{23} c_5 + c_{23} c_4 s_5) - c_1 s_4 s_5 \tag{2.9}
$$

$$
r_{31} = c_6(c_{23} s_5 + s_{23} c_4 c_5) - s_{23} s_4 s_6 \tag{2.10}
$$

$$
r_{32} = -s_6(c_{23} s_5 + s_{23} c_4 c_5) - s_{23} s_4 c_6 \tag{2.11}
$$

$$
r_{33} = s_{23} c_4 s_5 - c_{23} c_5 \tag{2.12}
$$

$$
p_x = c_1(a_1 + a_3 c_{23} + d_4 s_{23} + a_2 c_2) \tag{2.13}
$$

$$
p_y = s_1(a_1 + a_3 c_{23} + d_4 s_{23} + a_2 c_2) \tag{2.14}
$$

$$
p_z = a_3 s_{23} - d_4 c_{23} + a_2 s_2 \tag{2.15}
$$

Eq.(2.3) constitute the kinematics of the FaraAT2. They specify how to compute the position and orientation for frame $\{6\}$ relative to frame $\{0\}$ of the robot. These are the basic Eq.s for all kinematic analysis of this robot manipulator.

2.3 Jacobian

In this section, the robot Jacobian which is the tool to express the transition from a different kind of velocity inside the robotic structure was presented. There are three basic methods to compute the robot Jacobian includes: The simple propagation velocity from link to link; the geometry computation; and the numeric computation. The numeric computation of robot Jacobian was used in this dissertation.

For general n-degree of freedom manipulator

$$\dot{X}_E = J(\theta)\dot{\theta} \tag{2.16}$$

where $J(\theta) \in \Re^{6 \times n}$ is matrix Jacobian. $X_E = \left[p_x, p_y, p_z, \alpha, \beta, \lambda \right]^T \in \Re^{6 \times 1}$ is a vector of position and orientation of end-effector. $\theta = [\theta_1, \theta_2, ..., \theta_n]^T \in \Re^{n \times 1}$ is a vector of joint angle.

The FARA-AT2 included 6 revolute joints. Therefore, the robot Jacobian obtained as

$$J(\theta) = \begin{bmatrix} \hat{z}_1 \times (\vec{P} - \vec{0}_1) & \hat{z}_2 \times (\vec{P} - \vec{0}_2) & \hat{z}_3 \times (\vec{P} - \vec{0}_3) & \hat{z}_4 \times (\vec{P} - \vec{0}_4) & \hat{z}_5 \times (\vec{P} - \vec{0}_5) & \hat{z}_6 \times (\vec{P} - \vec{0}_6) \\ \hat{z}_1 & \hat{z}_2 & \hat{z}_3 & \hat{z}_4 & \hat{z}_5 & \hat{z}_6 \end{bmatrix} \tag{2.17}$$

where $\hat{z}_i = {}^o_i R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, $\vec{P} = X = \left[p_x, p_y, p_z \right]^T$, $\vec{O}_i = X_i = \left[p_{xi}, p_{yi}, p_{zi} \right]^T$, $p_{xi} = {}^0_i T(1,4)$, $p_{yi} = {}^0_i T(2,4)$,

$p_{zi} = {}^0_i T(3,4)$ ($i = 1, ..., 6$). Eq. (2.17) was rewritten as

$$J(\theta) = \begin{bmatrix} \hat{z}_1 \times (\vec{P} - \vec{0}_1) & \hat{z}_2 \times (\vec{P} - \vec{0}_2) & \hat{z}_3 \times (\vec{P} - \vec{0}_3) & \hat{z}_4 \times (\vec{P} - \vec{0}_4) & \hat{z}_5 \times (\vec{P} - \vec{0}_5) & \hat{z}_6 \times (\vec{P} - \vec{0}_6) \\ \hat{z}_1 & \hat{z}_2 & \hat{z}_3 & \hat{z}_4 & \hat{z}_5 & \hat{z}_6 \end{bmatrix}$$

$$= \begin{bmatrix} {}^0_1 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \left(\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - \begin{bmatrix} p_{x1} \\ p_{y1} \\ p_{z1} \end{bmatrix} \right) & \cdots & {}^0_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \left(\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - \begin{bmatrix} p_{x6} \\ p_{y6} \\ p_{z6} \end{bmatrix} \right) \\ {}^0_1 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & \cdots & {}^0_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} {}^0_1 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \left(\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - \begin{bmatrix} {}^0_1 T(1,4) \\ {}^0_1 T(2,4) \\ {}^0_1 T(3,4) \end{bmatrix} \right) & \cdots & {}^0_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \left(\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - \begin{bmatrix} {}^0_6 T(1,4) \\ {}^0_6 T(2,4) \\ {}^0_6 T(3,4) \end{bmatrix} \right) \\ {}^0_1 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & \cdots & {}^0_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{2.18}$$

2.4 Inverse Kinematics

The direct kinematic equation establishes the functional relationship between the joint variables and the position and orientation of the end-effector. The inverse kinematics

problem is to determine the joint variables corresponding to a given the position and orientation of the end-effector. Solving this problem is fundamental to transforming the motion specifications, assigned to the end-effector in the operational space, into the corresponding joint space that allow the desired motion to be executed. In this section, three inverse kinematics method for FARA-AT2 robot manipulator were presented: Pieper method, Jacobian-base method and cyclic coordinate descent (CCD) method.

2.4.1 Pieper method

The *Pieper's solution* could be applied to solve the inverse kinematic problem with closed-form solution. We can briefly explain the closed form solution of FataAT2 using the Pieper's solution as follows.

We wish to solve θ_i when ${}^0_6T$ is given. Assumed that the orientation of end-effector is X-Y-Z rotation. We have the ${}^0_6T$ given as

$$
{}^0_T T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^0_1T(\theta_1){}^1_2T(\theta_2){}^2_3T(\theta_3){}^3_4T(\theta_4){}^4_5T(\theta_5){}^5_6T(\theta_6)
\tag{2.19}
$$

and

$$
R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix}
\tag{2.20}
$$

where $X_E = \begin{bmatrix} p_x, p_y, p_z \end{bmatrix}^T$ and $\begin{bmatrix} \gamma, \beta, \alpha \end{bmatrix}^T$ is vector of orientation were given.

A restatement of Eq.(2.19) that puts the independence on θ_1 on the left-hand side of the Eq. is

$$
{}^0_1T^{-1}\,{}^0_6T = {}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T
\tag{2.21}
$$

where at the left-hand, we have

$$
{}^0_1T^{-1}\,{}^0_6T = \begin{bmatrix} c_1 & s_1 & 0 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} r_{11}c_1 + r_{21}s_1 & r_{12}c_1 + r_{22}s_1 & r_{13}c_1 + r_{23}s_1 & p_xc_1 + p_ys_1 \\ -r_{11}s_1 + r_{21}c_1 & -r_{12}s_1 + r_{22}c_1 & -r_{13}s_1 + r_{23}c_1 & -p_xs_1 + p_yc_1 \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{2.22}
$$

and at the right-hand, we have

$$
{}^1_6T = \begin{bmatrix}
-c_6(s_{23}s_5 - c_{23}c_4c_5) - s_6c_{23}s_4 & s_6(s_{23}s_5 - c_{23}c_4c_5) - c_{23}c_6s_4 & s_{23}c_5 + c_{23}c_4s_5 & a_1 + a_3c_{23} + d_4s_{23} + a_2c_2 \\
-c_4s_6 - c_5c_6s_4 & c_5s_4s_6 - c_4c_6 & -s_4s_5 & 0 \\
c_6(c_{23}s_5 + s_{23}c_4c_5) - s_6s_{23}s_4 & -s_6(c_{23}s_5 + s_{23}c_4c_5) - s_{23}c_6s_4 & s_{23}c_4s_5 - c_{23}c_5 & a_3s_{23} - d_4c_{23} + a_2s_2 \\
0 & 0 & 0 & 1
\end{bmatrix}
$$

$$(2.23)$$

Equating the (2,4) elements from both sides of Eq.(2.21), we have

$$-p_xs_1 + p_yc_1 = 0 \tag{2.24}$$

To solve this equation, we make the trigonometric substitutions

$$p_x = \rho\cos\phi;\ p_y = \rho\sin\phi \tag{2.25}$$

where $\rho = \sqrt{p_x^2 + p_y^2}$, $\phi = A\tan 2(p_y, p_x)$.

Substituting Eq. (2.25) into Eq. (2.24), we obtain

$$\sin(\phi - \theta_1) = 0 \tag{2.26}$$

The solution for θ_1 is

$$
\begin{aligned}
\theta_1 &= A\tan 2(p_y, p_x) \\
\theta_1 &= A\tan 2(p_y, p_x) - \pi
\end{aligned}
\tag{2.27}
$$

Now with θ_1 is known, we equate the (1,4) element and (3,4) element from the two sides of Eq.(2.21), we obtain

$$a_3c_{23} + d_4s_{23} + a_2c_2 = p_xc_1 + p_ys_1 - a_1 \tag{2.28}$$

$$a_3s_{23} - d_4c_{23} + a_2s_2 = p_z \tag{2.29}$$

Then, we square Eq. (2.28) and Eq. (2.29) plus together, we get

$$a_3s_3 + d_4c_3 = K \tag{2.30}$$

where

$$K = \frac{(p_xc_1 + p_ys_1 - a_1)^2 + p_z^2 - a_3^2 - d_4^2 - a_2^2}{2a_2} \tag{2.31}$$

To solve Eq. (2.30), we make the trigonometric substitutions

$$a_3 = \rho\cos\phi;\quad d_4 = \rho\sin\phi \tag{2.32}$$

where $\rho = \sqrt{a_3^2 + d_4^2};\quad \phi = A\tan 2(d_4, a_3)$.

Substituting Eq. (2.32) into Eq. (2.30), we have

$$\sin(\theta_3 - \phi) = \frac{K}{\rho} \tag{2.33}$$

So that, we have

$$\theta_3 = A\tan 2(d_4, a_3) + A\tan 2(K, \pm\sqrt{\rho^2 - K^2}) \tag{2.34}$$

17

To compute θ_2, Eq. (2.19) can be rewritten as

$$
{}^0_3T^{-1}\,{}^0_6T = {}^3_4T\,{}^4_5T\,{}^5_6T \tag{2.35}
$$

In Eq. (2.35), the left-hand obtain as

$$
{}^0_3T^{-1}\,{}^0_6T =
\begin{bmatrix}
c_{23}c_1 & c_{23}s_1 & s_{23} & -a_1c_{23}-a_2c_3 \\
-s_{23}c_1 & -s_{23}s_1 & c_{23} & a_1s_{23}+a_2s_3 \\
s_1 & -c_1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
r_{11} & r_{12} & r_{13} & p_x \\
r_{21} & r_{22} & r_{23} & p_y \\
r_{31} & r_{32} & r_{33} & p_z \\
0 & 0 & 0 & 1
\end{bmatrix} \tag{2.36}
$$

and the right-hand is

$$
{}^3_6T = {}^3_4T\,{}^4_5T\,{}^5_6T =
\begin{bmatrix}
c_4c_5c_6-s_4s_6 & -c_6s_4-c_4c_5s_6 & c_4s_5 & a_3 \\
c_6s_5 & -s_5s_6 & -c_5 & -d_4 \\
c_4s_6+c_5c_6s_4 & c_4c_6-c_5s_4s_6 & s_4s_5 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \tag{2.37}
$$

Equating the (1,4) elements and the (2,4) elements from the two sides of Eq.(2.35), we get

$$
c_{23}c_1p_x+c_{23}s_1p_x+s_{23}p_z-a_1c_{23}-a_2c_3 = a_3 \tag{2.38}
$$

$$
-s_{23}c_1p_x-s_{23}s_1p_x+c_{23}p_z+a_1s_{23}+a_2s_3 = -d_4 \tag{2.39}
$$

s_{23} and c_{23} solved as

$$
s_{23} = \frac{(a_3+a_2c_3)p_z+(p_xc_1+p_ys_1-a_1)(d_4+a_2s_3)}{p_z^2+(p_xc_1+p_ys_1-a_1)^2} \tag{2.40}
$$

$$
c_{23} = \frac{(p_xc_1+p_ys_1-a_1)(a_3+a_2c_3)-(d_4+a_2s_3)p_z}{p_z^2+(p_xc_1+p_ys_1-a_1)^2} \tag{2.41}
$$

We solve for $\theta_2+\theta_3$ as

$$
\theta_{23} = A\tan 2\left(\begin{matrix}(a_3+a_2c_3)p_z+(p_xc_1+p_ys_1-a_1)(d_4+a_2s_3),\\(p_xc_1+p_ys_1-a_1)(a_3+a_2c_3)-(d_4+a_2s_3)p_z\end{matrix}\right) \tag{2.42}
$$

then θ_2 is computed as

$$
\theta_2 = \theta_{23}-\theta_3 \tag{2.43}
$$

To compute θ_4, we consider Eq. (2.35). Equating the (1,3) elements and the (3,3) elements from the two sides of Eq.(2.35), we get

$$
r_{13}c_{23}c_1+r_{23}c_{23}s_1+r_{33}s_{23} = c_4s_5 \tag{2.44}
$$

$$
r_{13}s_1-r_{23}c_1 = s_4s_5 \tag{2.45}
$$

If $s_5 \neq 0$, we have

$$
c_4 = \frac{r_{13}c_{23}c_1+r_{23}c_{23}s_1+r_{33}s_{23}}{s_5} \tag{2.46}
$$

$$s_4 = \frac{r_{13}s_1 - r_{23}c_1}{s_5} \tag{2.47}$$

From Eq. (2.46) and (2.47), θ_4 was computed as

$$\theta_4 = A\tan 2(r_{13}s_1 - r_{23}c_1, r_{13}c_1c_{23} + r_{23}c_{23}s_1 + r_{33}s_{23}) \tag{2.48}$$

To compute θ_5, we consider Eq. (2.35). Equating the (2,3) from two sides of Eq. (2.35), we get

$$r_{13}s_{23}c_1 + r_{23}s_{23}s_1 - r_{33}c_{23} = c_5 \tag{2.49}$$

Then, θ_5 was computed as

$$\theta_5 = A\tan 2(\pm\sqrt{1 - (r_{13}s_{23}c_1 + r_{23}s_{23}s_1 - r_{33}c_{23})^2}, r_{13}s_{23}c_1 + r_{23}s_{23}s_1 - r_{33}c_{23}) \tag{2.50}$$

We consider Eq. (2.35). Equating the (2,1) and (2,2) from two sides of Eq. (2.35), we get

$$-r_{11}s_{23}c_1 - r_{21}s_{23}s_1 + r_{31}c_{23} = c_6s_5 \tag{2.51}$$

$$r_{12}s_{23}c_1 + r_{22}s_{23}s_1 - r_{32}c_{23} = s_5s_6 \tag{2.52}$$

If $s_5 \neq 0$, we have

$$\theta_6 = A\tan 2(r_{12}s_{23}c_1 + r_{22}s_{23}s_1 - r_{32}c_{23}, -r_{11}s_{23}c_1 - r_{21}s_{23}s_1 + r_{31}c_{23}) \tag{2.53}$$

In this method, we can be seen that in case $\theta_5 = 0$, the robot manipulator is in a singular configuration.

2.4.2 Jacobian-based method

With increasing computation power, numerical approaches are a common tool to solve the inverse kinematics problem. From the forward kinematics, we have

$$\vec{X}_0 = \vec{f}(\vec{\theta}_0) \text{ at } \theta_0 \tag{2.54}$$

where f is function of forward kinematics. Using linearization, we can have

$$\vec{f}(\vec{\theta}_d) \approx \vec{f}(\vec{\theta}_0) + J(\vec{\theta}_0)(\vec{\theta}_d - \vec{\theta}_0) \tag{2.55}$$

$$\vec{f}(\vec{\theta}_d) - \vec{f}(\vec{\theta}_0) \approx J(\vec{\theta}_0)(\vec{\theta}_d - \vec{\theta}_0) \tag{2.56}$$

$$\Leftrightarrow \vec{e} \approx J\Delta\vec{\theta} \tag{2.57}$$

where J is robot Jacobian which computed as section 2.3. $\vec{\theta}_d$ is vector of desired joint position. $\vec{\theta}_0$ is vector of present joint position. $\vec{e}$ is vector of the error of end-effector position and orientation in Cartesian space. The inverse kinematics problem became find $\Delta\vec{\theta}$ such that $\left\|\vec{e} - J\Delta\vec{\theta}\right\|^2$ is minimized.

Let consider $\left\|\vec{e} - J\Delta\vec{\theta}\right\|^2$

$$
\begin{aligned}
G &= \left\| \vec{e} - J\Delta\vec{\theta} \right\|^2 \\
&= \left(\vec{e} - J\Delta\theta \right)^T \left(\vec{e} - J\Delta\theta \right) \\
&= \vec{e}^T\vec{e} - 2\vec{e}^T J\Delta\vec{\theta} + \Delta\vec{\theta}^T J^T J\Delta\vec{\theta}
\end{aligned}
\tag{2.58}
$$

Derivative G with $\Delta\vec{\theta}$, we get

$$
\begin{aligned}
&\frac{\partial G}{\partial\Delta\vec{\theta}} = 0 \\
&\Leftrightarrow -2\vec{e}^T J + 2\Delta\vec{\theta}(J^T J) = 0 \\
&\Leftrightarrow \Delta\vec{\theta} = (J^T J)^{-1} J^T \vec{e} \\
&\Leftrightarrow \Delta\vec{\theta} = J^+\vec{e}
\end{aligned}
\tag{2.59}
$$

The algorithm of inverse kinematics based on Jacobian is show as

---
Algorithm 1. Jacobian-base inverse kinematics

Loop

      Calculate $\Delta\vec{\theta} \leftarrow J^+\vec{e}$

      Update joint angles: $\vec{\theta} \leftarrow \vec{\theta}_{previous} + \Delta\vec{\theta}$

      Calculate: $\vec{e} \leftarrow X_{desired} - \vec{f}(\vec{\theta})$

    Until $\left\| \vec{e} \right\| \le \varepsilon$ (ε is selected small enough)

To reduce the iterate and avoid singularity, the algorithm1 was modified as

---
Algorithm 2. Jacobian-base inverse kinematics

Loop

      Calculate $\Delta\vec{\theta} \leftarrow J^+\vec{e} + (I - J^+ J)\beta$ (β is selected)

      Update joint angles: $\vec{\theta} \leftarrow \vec{\theta}_{previous} + \alpha\Delta\vec{\theta}$ (α is selected)

      Calculate: $\vec{e} \leftarrow X_{desired} - \vec{f}(\vec{\theta})$

    Until $\left\| \vec{e} \right\| \le \varepsilon$ (ε is selected small enough)

2.4.3 Cyclic coordinate descent (CCD) method

      The use of CCD to solve the inverse kinematics problem was first proposed in [28]. The purpose of the CCD is to minimize the function that relates the current position of the robot's end effector and the desired position to reach under a certain precision, by moving the robot joints in succession. To do this, the algorithm has to find in each iteration which movement of the current joint minimizes this function. The main advantage is, that CCD is easier to implement. It starts at the end-effector which is usually the last shape of a model. The algorithm measures the difference between a joint's position and the end-effectors position. It then calculates either a rotation or quaternion

to reduce this difference to zero. It does this for each joint, iterating from the end-effector to the immobile joint at the root of the kinematic chain.
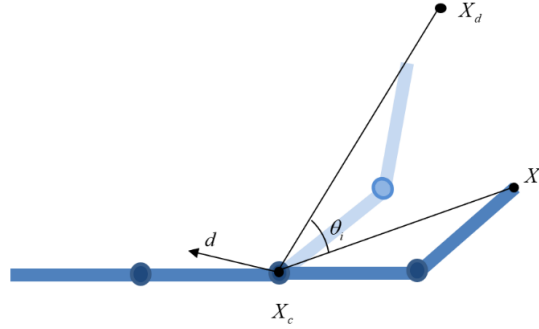
Figure 2.3. The CCD algorithm of a planar robot with rotation joint

For general n-link joint chain Fig. where X_d is the desired end-effector position, X_e is the current end-effector position, X_c is the position of the current joint to be moved, θ_i is value of rotation joint and $\vec{d}$ is direction of rotation. Vector $\overrightarrow{X_c X_e}$ and $\overrightarrow{X_c X_d}$ will be aligned, and the $\left\| \overrightarrow{X_e X_d} \right\|$ distance will be minimized.

$$\theta_i = \frac{X_e - X_c}{\|X_e - X_c\|} \cdot \frac{X_d - X_c}{\|X_d - X_c\|} \tag{2.60}$$

$$\vec{d}_i = \frac{X_e - X_c}{\|X_e - X_c\|} \times \frac{X_d - X_c}{\|X_d - X_c\|} \tag{2.61}$$

During the optimization process, the position and orientation can be weighted in different manners, so as to reach the goal requirement that includes both parameters.

Algorithm 3. CCD algorithm inverse kinematics

Loop

   For ($\theta_i = 1$, $\theta_i \le n$, $\theta_i = \theta_i + 1$)

      Calculate (θ_i, d_i)

      Solve the abruptness in movements, $\theta_i = k\theta_i$

      If vector ($\overrightarrow{X_c X_d}$, $\overrightarrow{X_c X_e}$) are parallel then manage the singularity end if

      If $\theta_c \ge \theta_{max}$ then $\theta_i = \theta_{max} - \theta_c$ end if

      Rotation joint i using (θ_i, d_i)

   End for

   Update distance $\left\| \overrightarrow{X_e X_d} \right\|$

  Until $\left\| \overrightarrow{X_e X_d} \right\| \le \varepsilon$ (ε is selected small enough)

To illustration the effective of inverse kinematics algorithms, the inverse kinematics of FARA-AT2 robot manipulator was programming on C#. In this software, three algorithms were used: CCD, Jacobian-base and the forward and backward reaching inverse kinematics (FABRIK) which based on the CCD idea. The test result was show in Figure 2.4 and Figure 2.5.
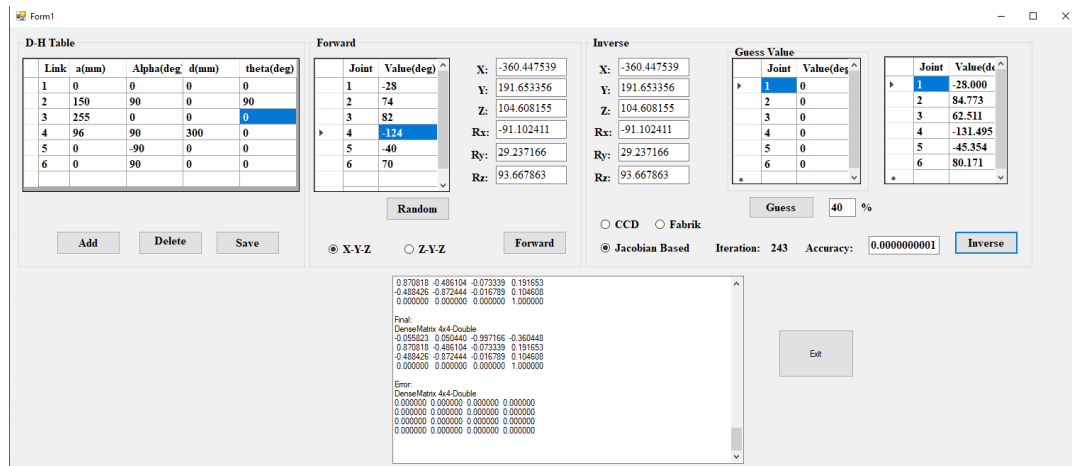
Figure 2.4.Inverse kinematics using Jacobian-base
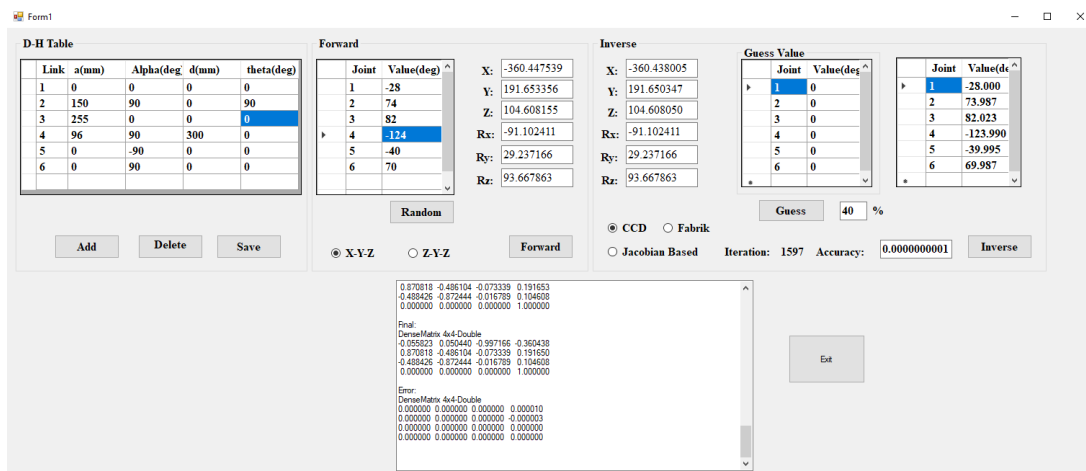
Figure 2.5. Inverse kinematics using CCD

2.5 Dynamics

Dynamics play an important role for model-base fault-tolerant control robot manipulator. The analysis of dynamic model could detect, identify and estimate the magnitude of the fault which occurred in robot manipulator system. Computation of the forces and torques required for the actuators. In this dissertation, the dynamic of robot manipulator was followed Lagrangian dynamic formulation.

The kinetic energy of the ith link, K_i can be expressed as

$$K_i = \frac{1}{2} m_i v_{C_i}^T v_{C_i} + \frac{1}{2} {}^i\omega_i^T {}^{C_i}I_i {}^i\omega_i \qquad (2.62)$$

where m_i is a point mass of ith link. v_{C_i} is linear velocity of the point mass. I_i is the moment of inertia and ω_i is angle velocity of the point mass.

The total kinetic energy of the manipulator is the sum of the kinematic energy in the individual links

$$K = \sum_{i=1}^{n} K_i \qquad (2.63)$$

The potential energy of the ith link, P_i, can be expressed as

$$P_i = -m_i \, {}^0g^T \, {}^0h_{C_i} + P_{ref_i} \qquad (2.64)$$

where h_{C_i} is the height of mass m_i in a gravitational field with constant g.

The total potential energy stored in the manipulator is the sum of the potential energy in the individual links

$$P = \sum_{i=1}^{n} P_i \qquad (2.65)$$

The Lagrangian dynamic formulation provides a means of deriving the equations of motion from a scalar function called the Lagrangian, which is defined as the difference between the kinetic and potential energy of a mechanical system. The Lagrangian of a manipulator is

$$L(\theta,\dot{\theta}) = K(\theta,\dot{\theta}) - P(\theta) \qquad (2.66)$$

The equations of motion for the manipulator are then given by

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = \tau \qquad (2.67)$$

where $\tau \in \Re^n$ is vector of actuator torques. In the case of a manipulator, this Eq. becomes

$$\frac{d}{dt}\frac{\partial K}{\partial \dot{\theta}} - \frac{\partial K}{\partial \theta} + \frac{\partial P}{\partial \theta} = \tau \qquad (2.68)$$

In the state-space, the dynamic of robot manipulator can be rewritten in the form

$$M(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) + G(\theta) = \tau \qquad (2.69)$$

where $M(\theta) \in \Re^{n \times n}$ is the mass matrix of the manipulator, $V(\theta,\dot{\theta}) \in \Re^{n \times 1}$ is a vector of centrifugal and Coriolis terms, and $G(\theta) \in \Re^{n \times 1}$ is a vector of gravity term.

In practice, the dynamic model of robots is not known exactly, so the system in (2.69) can be written as

$$(\mathbf{M}(\theta)+\Delta M(\theta))\ddot{\theta}+(\mathbf{V}(\theta,\dot{\theta})+\Delta V(\theta,\dot{\theta}))\dot{\theta}+(\mathbf{G}(\theta)+\Delta G(\theta))+\delta = \tau \tag{2.70}$$

where $\Delta M, \Delta C$ and ΔG are unknown dynamic uncertainties and δ is external disturbance. $\mathbf{M}(q), \mathbf{C}(q,\dot{q}),$ and $\mathbf{G}(q)$ are estimation values of $M(q), C(q,\dot{q}),$ and $G(q)$. Thus, (2.70) can be simply rewritten as

$$\mathbf{M}(q)\ddot{q}+\mathbf{C}(q,\dot{q})\dot{q}+\mathbf{G}(q)+\psi = \tau \tag{2.71}$$

where $\psi = \Delta M\ddot{q}+\Delta C\dot{q}+\Delta G+\delta$.

The detail of dynamic model was shown in Appendix A.

2.6 Mathematical of Faults

In a robot manipulator, the motions of robot were generated by joint actuator which consisted of an electronic motor, a motor driver and the gear. In practice, the occurrence of faults/failure in the electric motor, the gear, the bearing and the motor driver can degrade the performance of a robot manipulator. For instant, in the electric motor, the increased friction between the stator and the rotor or fault in driver motor may decrease the motor torque. It can be considered at the loss-effective fault. Generally, two kinds of faults commonly occur in a robot manipulator are bias and loss-effective fault which can be described as

$$\tau^t = (I-\rho(t))\tau+\mathbf{f}(t) \quad (t>t_f) \tag{2.72}$$

where $\tau^t \in \Re^n$ is the vector of torque at the output joint actuator. $\tau \in \Re^n$ is the vector of torque at the output controller. $\rho(\tau) = diag(\rho_i(t)) \in \Re^{n\times n}, \ 0 \le \rho_i(t) < 1, \ (i=1,2,...,n)$ denotes the loss-effective rate. $\mathbf{f}(t) \in \Re^n$ is the vector of the bias fault. t_f is the time of the occurrence of faults. $I \in \Re^{n\times n}$ is identity matrix.

Substituting Eq.(2.72) into Eq. (2.69), the dynamics model of an n-degree of freedom robot manipulator with actuator faults can be written as

$$\mathbf{M}(q)\ddot{q}+\mathbf{C}(q,\dot{q})\dot{q}+\mathbf{G}(q)+\psi = (I-\rho(t))\tau+\mathbf{f}(t) \tag{2.73}$$

Remark 1: In this chapter, only the loss-effective and bias fault is investigated. The loss-effective fault with $\rho_i(t)=1$ and lock-in-place fault are not considered because the robot does not have the redundancy actuator so that robot cannot tolerate those kinds of faults.

Chapter 3: Platform Development, Robot Dynamic Parameters Identification And Controller

3.1 Introduction

In this chapter, the development of the FARA-AT2 robot manipulator in torque control mode is presented. The hardware information is explained in detail in subsection 2. In subsection 3, the identification of 3-DOF robot manipulator was shown by using least-squares technique. Finally, several controllers were applied to the FARA-AT2 such as point-to-point controller, and force controller to verify the ability to control the hardware evolution. Point-to-point controller was applied to the 6-DOF FARA-AT2 with a user interface in Labview. Force controller was applied to 3-DOF robot manipulator where joint 4,5 and 6 of FARA-AT2 were blocked. The force controller used 3-DOF force sensor for impedance control and sensor-less force controller for admittance control.

3.2 Hardware setup

The hardware setup of FARA-AT2 robot manipulator was shown in Figure 3.1. The original controller was replayed by National Instruments computer (NI) where the controller development was established in Labview-FPGA. In Table 3.1, the specification of FARA-AT2 was shown.
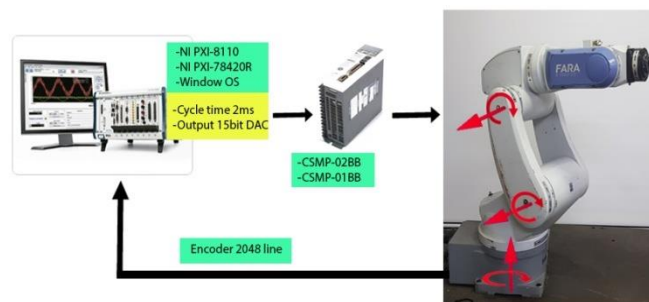
Figure 3.1. Hardware setup of FARA-AT2 robot manipulator.

Table 3.1. Specification FARA-AT2

Item		Specification
Type		Vertical Articulated
Degree of Freedom		6
Payload		3 Kgf
Repeatability		±0.04 mm
Length (max)		720 mm
Operating Range & Max Speed	Joint 1	±160º (240º/s)
	Joint 2	±150º -45 º (240º/s)
	Joint 3	±80º -200 º (240º/s)
	Joint 4	±200º (320º/s)
	Joint 5	±120º (300º/s)
	Joint 6	±360º (480º/s)
Allowable Moment	Joint 4	8.17 N.m
	Joint 5	8.8 N.m
	Joint 6	6.36 N.m
Motor Type		AC Servo Motor
Position Detection Method		Absolute Encoder
Installation Environment	Ambient Temperature	0º +40 ºC
	Ambient Humidity	20 80 %RH
	Vibration	<0.5 G
Air Pipe For User		ϕ6mm, 1ea
User Signal Line		14 pin
Weight	Main Body	34 kg
	Controller	< 25Kg
Power		220V 2.5KVA

From Figure 3.1, we can see that the servo motors which are used to control the robot joints are the CSMP series motor. The common specifications are shown in Table 3.1. Two servo motor CSMP-02BB are used in joint 1,2. One servo motor CSMP-01BB

is used in joint 3,4,5. The basis specification of those servo motor are shown in Table 3.2. In order to control those servo motor, the servo drive CSDJ series are used with the setting in torque control mode. The CSDJ driver wiring is shown in Figure 3.2. The torque command signal will be sent from the user computer to the servo drive through the channel 1 Pin 21,22. The joint angle is detected by a rotary encoder integrated inside each servo motor. The encoder is a 2048 line count incremental encoder. The output often coder is feedback to the computer through drive channel 1 pin 29 - 36. The torque output of the motor is measured through the analogue output channel 1 pin 23 - 28. The analogue voltage between the T-REF terminal (Channel 1 pin 21) and T-REF-SG (Channel 1 pin 22) determines the motor torque command input. Parameter SEt-05 (External torque command gain) of the servo drive CSDJ is the Analog Full Scale Torque Command Input parameter.

Table 3.2. CSMP Series motor basic specifications

	CSMP 01B	CSMP 02B
Driver	01BX1	02BX1
Rated Voltage(V)	220	
Rated Power (W)	100	200
Rated Torque (Nm)	0.319	0.637
Maximum Instant Torque (Nm)	0.95	1.911
Rated Revolving speed (RPM)	3000	
Maximum Revolving Speed (RPM)	5000	
Rotor Inertia (gf.cm.s2)	0.11	0.45
Power Rate (kw/s)	9.41	9.2
Mechanical Time Constant	0.9	1.1
Electrical Time Constant	2.9	5.0
Shaft Stiction Torque (Kgf.cm Max)	0.4	0.7
Space in Shaft Direction (mm)	0.04	
Revolving Direction	U-V-W	
Weight	0.8	1.8
Colour	Optional	
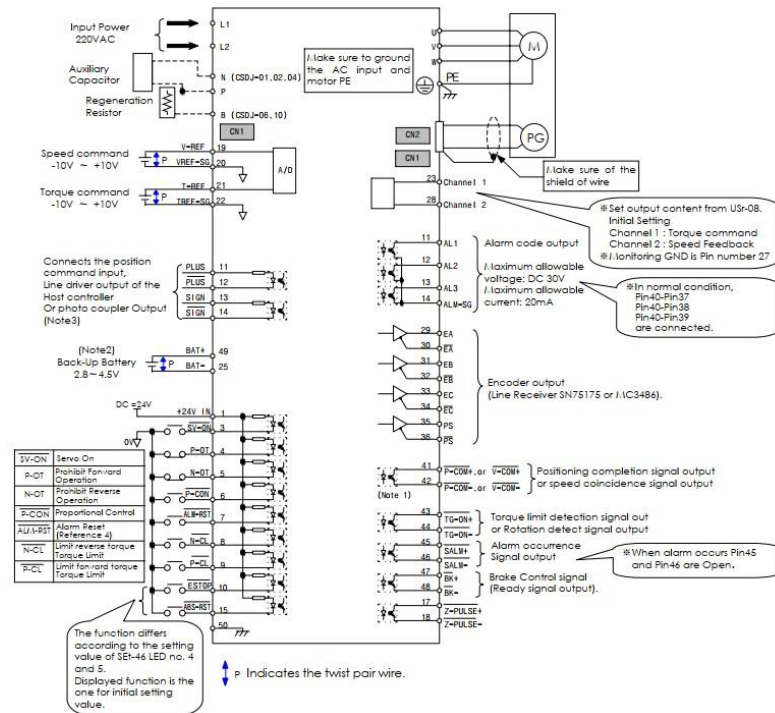
Figure 3.2. CSDJ driver wiring

The robot manipulator is using the National Embedded controller. This Embedded controller includes a PXI-1031 chassis with universal AC power supply, a PXI-8110embedded controller, PXI-7842R series multifunction RIO with virtex-5 LX50 FPGA. The entire control algorithm is written into two modules: Monitoring and control module and FPGA module (Encoder reading and AIO). In the FPGA module, the digital inputs are used to read the encoder signals. Several counters are programmed to compute the pulse numbers, then the joint positions and joint velocities (number of pulses per sampling time rate) are obtained. The digital inputs are connected to the encoder output signals in channel 1 pin 29-34 of the servo driver. With the onboard clock 40Mhz, the encoder readings are guaranteed to be obtained with high accuracy. The analogue output of PXI-7842R is used to send the torque command by connecting with the servo driver channel 1, pin 21,22. The torque feedbacks are obtained through the analogue input channel. In the monitoring and control module, the programming code is performed in PXI-8110 embedded controller. Some specifications of the PXI cards are listed in the next part. The overview of the Labview system is shown in Figure 3.7. The FPGA module is shown in Figure 3.8. The monitor control module is shown in Figure 3.9.

NI-PXI-7842R

 - R Series Multifunction RIO With Virtex-5 LX50 FPGA

- 8 analog inputs, independent sampling rates up to 200 kHz, 16-bit resolution, ±10 V

- 8 analog outputs, independent update rates up to 1 MHz, 16-bit resolution, ±10 V

- 96 digital lines configurable as inputs, outputs, counters, or custom logic at rates up to 40MHz

- Virtex-5 LX50 FPGA programmable with the Labview FPGA Module

- 3 DMA channels for high-speed data streaming

NI-PXI-8110

- 2.26 GHz Quad-Core PXI Embedded Controller

- 120 GB (or greater), 7200 rpm high-performance hard-drive standard

- 2 GB (1 x 2 GB DIMM), 800 MHz DDR2 RAM standard;4 GB maximum

- 10/100/1000BASE-TX (gigabit) Ethernet, Express Card/34 slot, and 4 Hi-Speed USB ports Integrated hard drive, GPIB, serial, and other peripheral I/O

- Windows OS and drivers already installedNI-PXI-1031

- 4-Slot 3U PXI Chassis with Universal AC

- Ability to accept both 3U PXI and Compact PCI modules

- HALT-tested design

- Auto/high temperature-controlled fan speed

- 400 W universal AC power supply

- Quiet acoustic noise emissions as low as 40 dBA
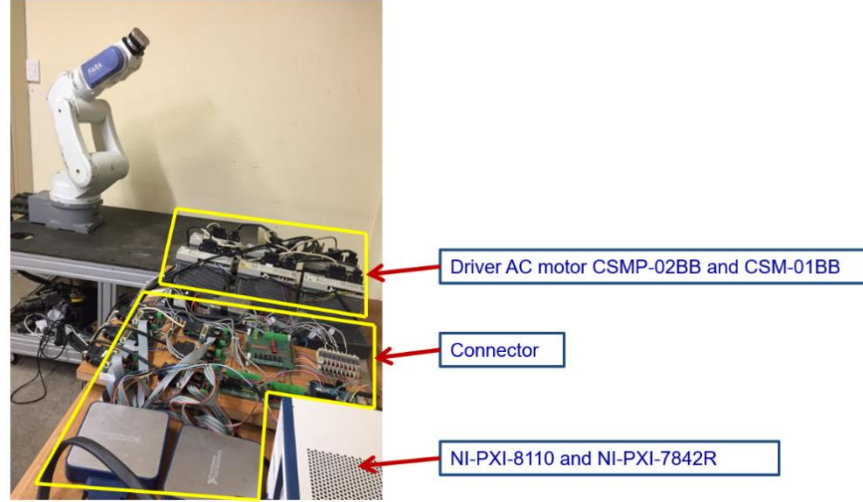
Figure 3.3. Final hardware setup

3.3 Dynamic parameter identification

An important property of the robot dynamic model is the linearity with respect to the dynamic parameters characterizing the manipulator links and rotors. Due to this property, the dynamic model Eq.(2.69) could be expressed as

$$\tau = Y(\theta, \dot{\theta}, \ddot{\theta})\pi \tag{3.1}$$

where $\pi \in \Re^{p\times 1}$ is a vector of constant parameters and $Y(\theta, \dot{\theta}, \ddot{\theta}) \in \Re^{n\times p}$ is a function of joint positions, velocities, and accelerations, usually called regressor. p is number of parameters. Due to the complicated of 6-DOF robot manipulator, in this dissertation, only 3-DOF robot manipulator was identified. In identification processing, it is necessary to ensure that excitation is sufficient to provide accurate and fast parameter estimation in the presence of disturbances, and the processing of the resulting data is simple and yields accurate results. Several approaches exist for parameterizing robot excitation trajectory such as finite sequences of joint acceleration or fifth-order polynomials interpolating between sets of joint position and velocities separated in time. Although these trajectories provide adequate excitation of the robot dynamics, the resulting measurement data are neither periodic nor band limited. Processing periodic, band limited measurements is more accurate, simplifying and improving the accuracy of the parameter estimates. In this dissertation, the trajectory $q_i(t)$ of each joint i is periodic, parameterized as a finite Fourier series as follows

$$q_i(t) = q_{i,0} + \sum_{k=1}^{N}(a_{i,k}\sin(k\omega_f t) + b_{i,k}\cos(k\omega_f t)) \tag{3.2}$$

where t is represent time, ω_f is the Fourier series frequency, $a_{i,k}, b_{i,k}, q_{i,0}$ are the coefficients of the sine and cosine function which are randomly selected for each joint trajectory.

The equation of the recursive least square method for updating the value of π are presented as follows

$$E_k = \tau_k - Y_k \hat{\pi}_{k-1} \tag{3.3}$$

$$\hat{\pi}_k = \hat{\pi}_{k-1} + P_{k-1} Y_k^T (I + Y_k P_{k-1} Y_k^T)^{-1} E_k \tag{3.4}$$

$$P_k = P_{k-1} - P_{k-1} Y_k^T (I + Y_k P_{k-1} Y_k^T)^{-1} Y_k^T P_{k-1} \tag{3.5}$$

where the matrix P_k is called the covariance matrix. The choice of the initial value of the covariance matrix plays an important key when applying the recursive least square method. If there is no primary knowledge about the system parameters, the choice of large starting value of the covariance matrix P(0) is recommended. However, if the starting values of the system parameters are similar to the exact value of the system parameters can be accepted, then enough to choose the small starting value of the covariance matrix *P(0)*.

The detail of $Y(\theta, \dot{\theta}, \ddot{\theta})$ and π were shown in Appendix B.

3.4 Controller

All the controller in this dissertation was established in Labview. The entire control algorithm is written into two modules: Monitoring and control module and FPGA module (Encoder reading and AIO). In the FPGA module, the digital inputs are used to read the encoder signals. Several counters are programmed to compute the pulse numbers, then the joint positions and joint velocities (number of pulses per sampling time rate) are obtained. The digital inputs are connected to the encoder output signals in channel 1 pin 29-34 of the servo driver. With the onboard clock 40Mhz, the encoder readings are guaranteed to be obtained with high accuracy.
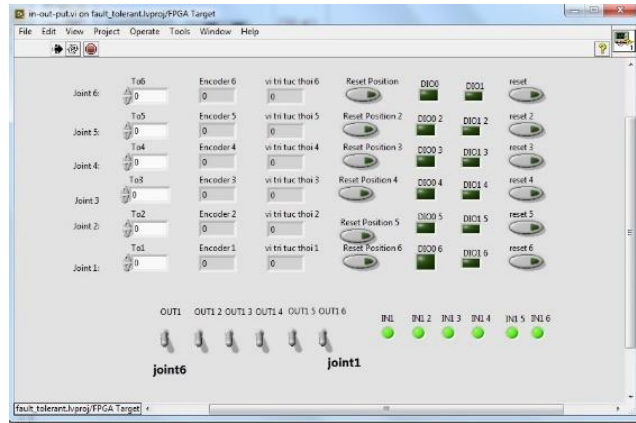
Figure 3.4. The interface of FPGA Model: Encoder reading and AIO
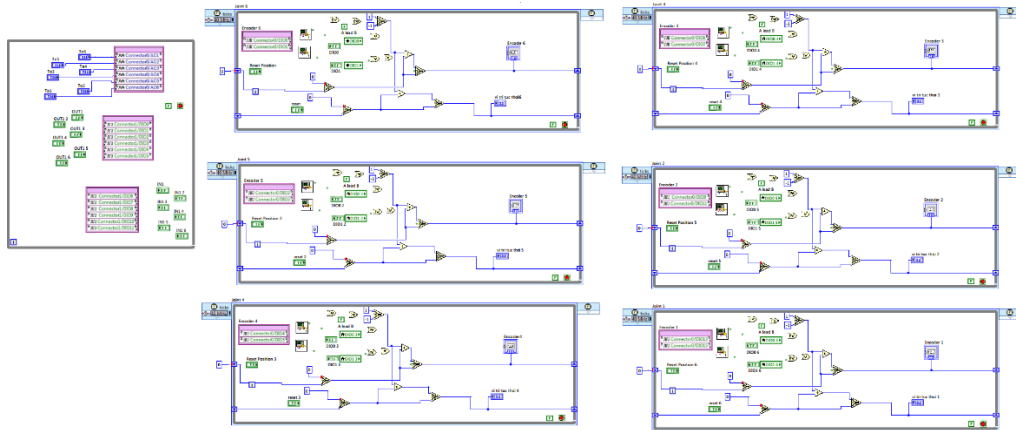
Figure 3.5. FPGA model code

3.4.1 Point-to-point controller

a) 6-DOF robot manipulator

In this subsection, to verify the hardware setup, the 6-DOF robot manipulator with PID control was designed with manual mode at each joint and monitoring the information of the system. In this programming, two modes control were designed:1) Tracking trajectory with desired trajectory in joint space, 2) Manually control for each joint. In Figure 3.6, the block diagram of position controller for 6-DOF robot manipulator. Then, in Figure 3.7, the interface of programming was shown. The PID parameters was selected as $K_p = diag(250, 280, 250, 300, 500, 250)$, $K_d = diag(70, 20, 20, 20, 20, 20)$ and

$K_i = diag(0.1, 0.1, 0.1, 0.1, 0.1, 0.1)$. The desired trajectory at each joint is $\theta_{1d} = \dfrac{\pi}{6} \sin\left(\dfrac{\pi t}{800}\right)$,

$\theta_{2d} = \dfrac{\pi}{3} \sin\left(\dfrac{\pi t}{800}\right)$, $\theta_{3d} = \dfrac{\pi}{2} \sin\left(\dfrac{\pi t}{800}\right)$, $\theta_{4d} = \dfrac{\pi}{5} \sin\left(\dfrac{\pi t}{800}\right)$, $\theta_{5d} = \dfrac{\pi}{6} \sin\left(\dfrac{\pi t}{800}\right)$, $\theta_{6d} = \dfrac{\pi}{3} \sin\left(\dfrac{\pi t}{800}\right)$ for

Mode 1. The tracking trajectory in mode 1 results was shown in Figure 3.9. Then, the
snapshoot of 6-DOF robot manipulator during the operation was presented in Figure 3.8.
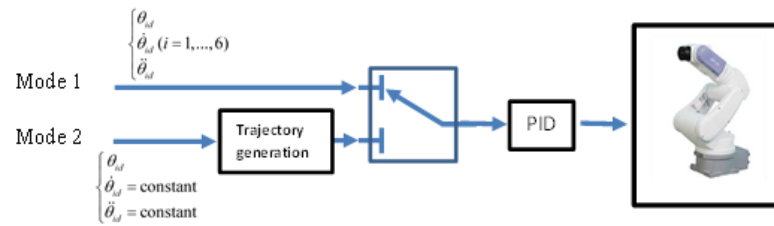
Figure 3.6. Block diagram of position control for 6-DOF robot manipulator
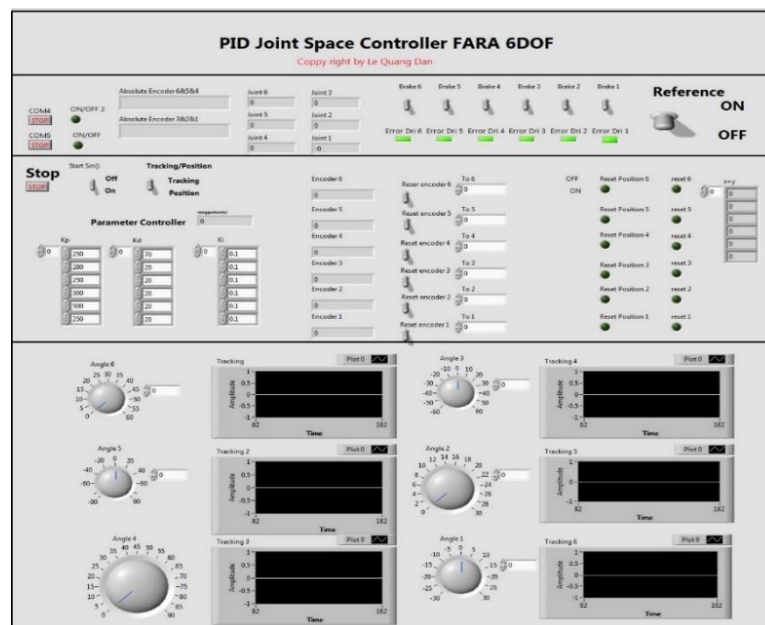
Figure 3.7. PID joint space controller for 6-DOF FARA-AT2

Figure 3.8. Snapshoot of 6-DOF robot manipulator moving

a) Joint1

b) Joint2

c) Joint3

d) Joint4

e) Joint5

f) Joint6
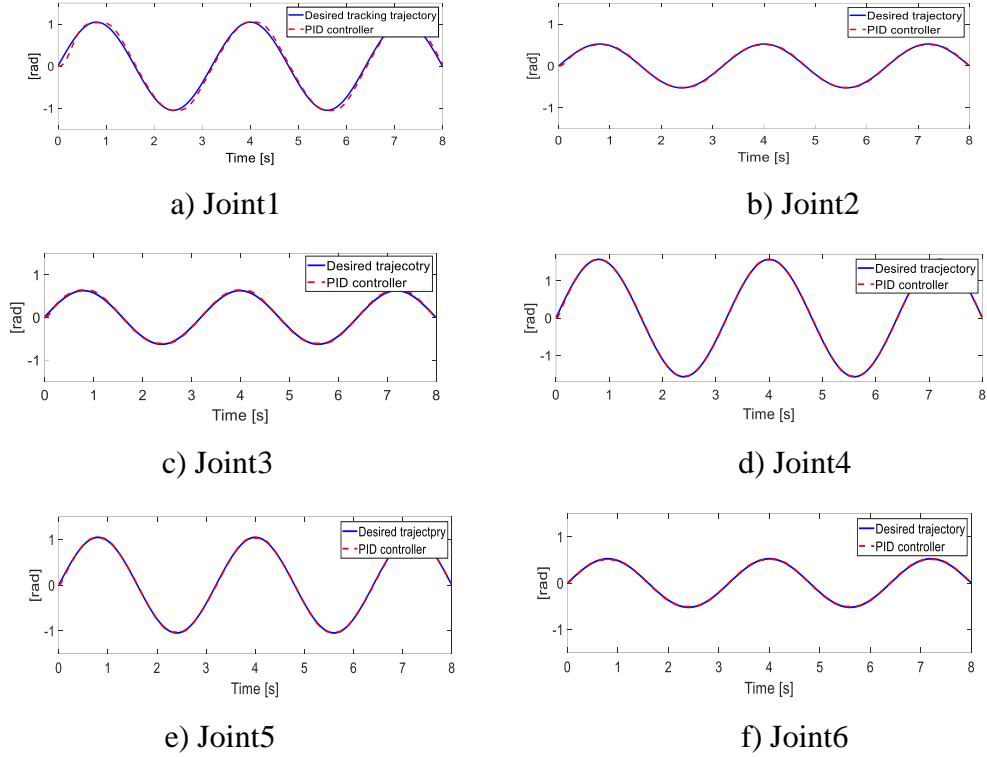
Figure 3.9. Tracking trajectory at each joint in model 1

b) Computed Torque control with user interface for 3-DOF robot manipulator

In this subsection, the Cartesian input of the user interface for 3-DOF robot manipulator with computed torque control was designed which used to verify the identification processing. The block diagram was shown in Figure 3.10. Invers kinematic block used the Piepe method. The block of trajectory generation utilized the cubic polynomial with 4 constraints showed as

$$\theta_i(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \tag{3.6}$$

where a_0, a_1, a_2 and a_3 were computed based on a desired velocity constant $\dot{\theta}_i = 0.002$ (rad/s). The computed torque controller given as

$$M(\theta)(\ddot{\theta}_d + K_p e + K_v \dot{e}) + V(\theta, \dot{\theta}) + G(\theta) + F(\theta, \dot{\theta}) = \tau \tag{3.7}$$

where $K_p = diag(130, 120, 140)$, $K_v = diag(20, 18, 18)$, $e = \theta_d - \theta$, $\dot{e} = \dot{\theta}_d - \dot{\theta}$.
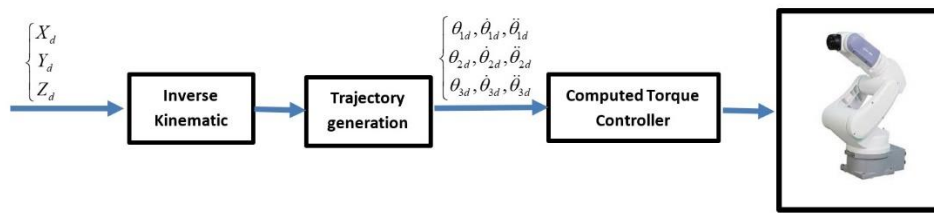
Figure 3.10. Block diagram of Computed Torque control with Cartesian input user interface for 3-DOF robot manipulator

In the Figure 3.11, 3 modes of Cartesian input were shown. In Figure 3.11-a, Auto step mode means that the distance of each axis was pre-defined with a contain. The second mode is Manual step which means that user can input any value of each axis. Final mode is Command mode. This mode can read G-code command from operator or from G-code file. This mode can work with 3D printer. The Computed Torque control code was shown in Figure 3.12.
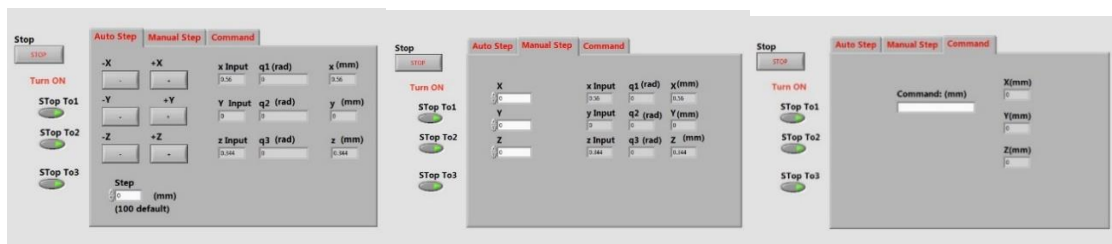
a)   Mode1: Auto step   b) Mode 2: Manual step   c) Mode 3: Send command

Figure 3.11. User interface of point-to-point control of 3-DOF robot manipulator
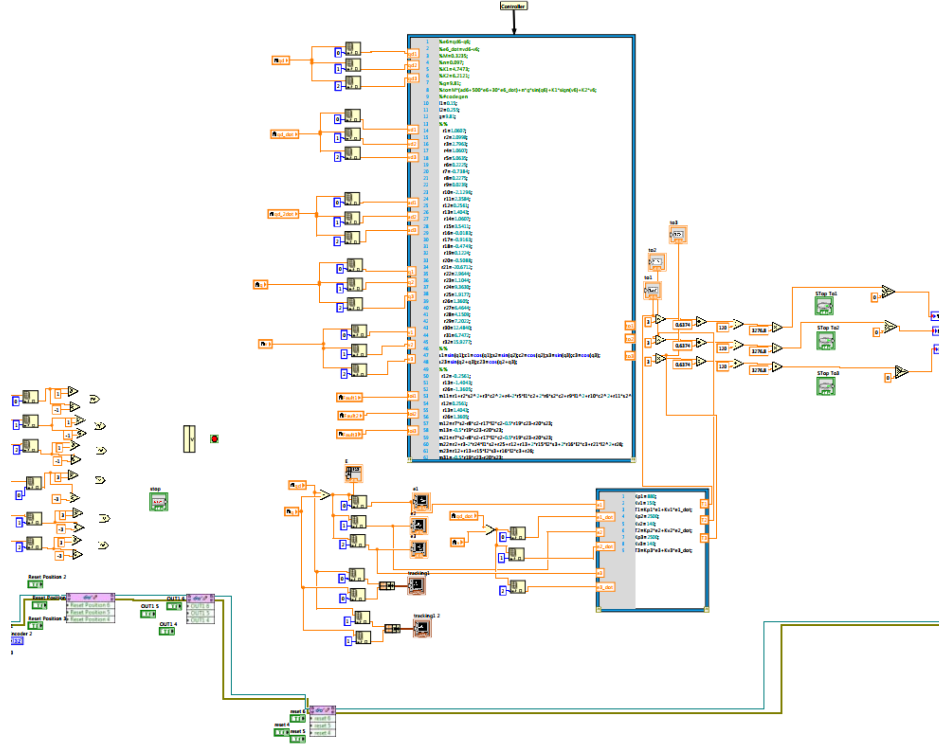
Figure 3.12. Computed Torque Control in Labview.

3.4.2 Force controller

a) Impedance control with force sensor at the end-effector

In this subsection, the impedance control with 3-DOF force sensor was implemented. The hardware setup was shown in Figure 3.13.The force sensor is 3-DOF 9317B Kistler which technical data showed in Table 3.3.The impedance control was given as

$$\tau = J_a^T \left(M_x(\theta)\alpha + V_x(\theta,\dot{\theta}) + G_x(\theta) - \tau_{env} \right) \tag{3.8}$$

where $M_x(\theta) = J_a^{-T}(\theta)M(\theta)J_a^{-1}(\theta)$, $V_x(\theta,\dot{\theta}) = J_a^{-T}(\theta)V(\theta,\dot{\theta})J_a^{-1}(\theta) - M_x(\theta)\dot{J}_a(\theta)J_a^{-1}(\theta)$,

$G_x(\theta) = J_a^{-T}(\theta)G(\theta)$ which J_a is Jacobian matrix. τ_{env} is forces from sensor, and

$$\alpha = \ddot{x}_d + M_m^{-1}\left(D_m(\dot{x}_d - \dot{x}) + K_m(x_d - x) + \tau_{env} \right) \tag{3.9}$$

$$M_m(\ddot{x} - \ddot{x}_d) + D_m(\dot{x} - \dot{x}_d) + K_m(x - x_d) = \tau_{env} \tag{3.10}$$

where $M_m = diag(1,1,1)$, $D_m = diag(0.2,0.2,0.2)$, $K_m = diag(2,2,2)$ are desired inertia, damping and stiffness, respectively.
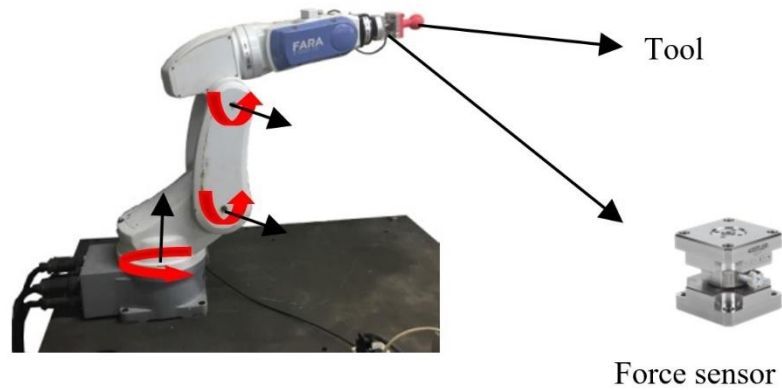
Figure 3.13. The 3-DOF robot manipulator with force sensor

Table 3.3. Technical Data of 9317B Kistler

Range (without moments if e.g four force links are mounted into one force place)	F_x, F_y	N	-1000 … 1000
Range (Example with point of force application $F_{x,y}$ 12mm above top plate)	F_x, F_y	N	-200 … 200
Range (Point of force application F_z centric)	F_z	N	-2000 … 2000
Overload		%	10
Calibrated range (Point of force application $F_{x,y}$ 7.5 mm below top plate surface)	F_x, F_y	N	0 … 600; 0 … 60
Calibrated range (Point of force application F_z centric)	F_z	N	0 … 2000; 0 … 200
Max moments $M_z = 0$; $F_z = 0$ $M_{x,y} = 0$; $F_z = 0$	$M_{x,y}$ M_z	N.m N.m	-5/5 -4/4
Threshold		N	<0.01
Sensitivity	F_x, F_y F_z	pC/N pC/N	≈-26 ≈-11

Nature frequency		$f_n(x)$	kHz	≈ 5
		$f_n(y)$	kHz	≈ 5
		$f_n(z)$	kHz	≈ 21
weight			g	85

Experiment 1:

In this experiment, the control of robot manipulator contacts with unknown environment was presented. The obstacle occurs during the tracking trajectory is setup as in Figure 3.14. The joint tracking trajectory with contact environment of controller is shown in Figure 3.15. In Figure 3.16, value of force at each axis was presented. The desired tracking trajectory given as

$$\begin{cases} x = 0 \\ y = 0.15\sin(\pi t/8) \\ z = 0.15\cos(\pi t/8) - 0.15 \end{cases} \tag{3.11}$$
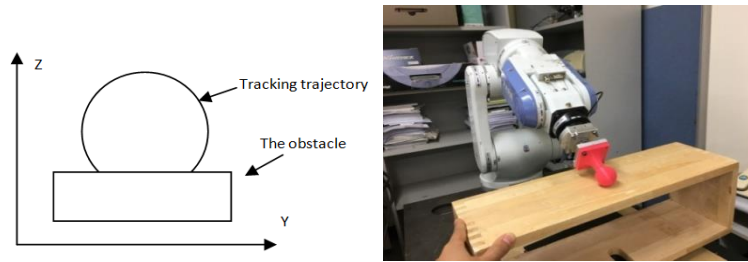
Figure 3.14. Experiment setup in contact with obstacle

a) Joint 1 *b) Joint 2*

c) Joint 3
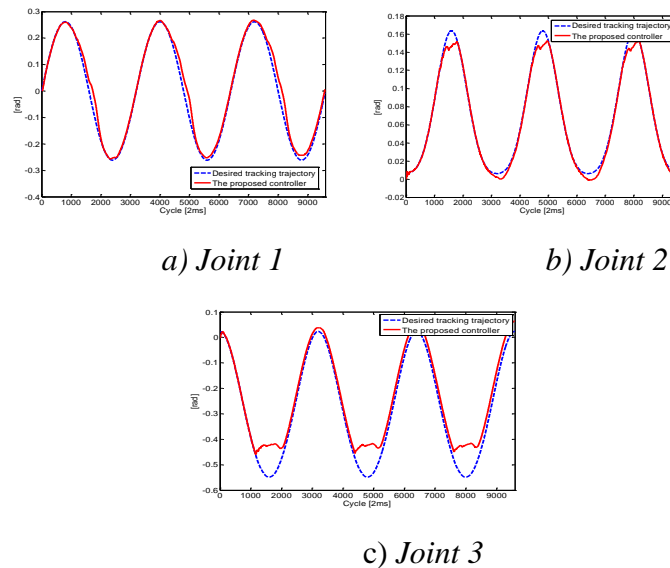
Figure 3.15. Tracking trajectory at joint space

a)X axis

b)Y axis

c)Z axis

Figure 3.16. Force measurement

Experiment 2:

In this experiment, human can directly contact with robot to push and pull at end-effector. This experiment to show that the robot can interact and share the workspace with human. The video of this demonstration can be see as https://www.youtube.com/watch?v=KV0D5S-DvTo and the snapshoot was shown in Figure 3.17.

Figure 3.17. Snapshoot human-robot interaction

b) Admittance control without force sensor

In this subsection, an admittance control without force sensor was present to show the effectiveness of dynamic model of robot in controller. To estimate the torque interaction, an adaptive third order sliding mode observer was used as

$$\begin{cases} \dot{\hat{x}}_1 = \kappa_1 |e_1|^{\frac{2}{3}} sgn(e_1) + \hat{x}_2 \\ \dot{\hat{x}}_2 = \kappa_2 |e_1|^{\frac{1}{3}} sgn(e_1) + f(x_1, \hat{x}_2, u) + \hat{x}_3 \\ \dot{\hat{x}}_3 = \kappa_3 sgn(e_1) \end{cases} \qquad (3.12)$$

Adaptive law defined as

$$\dot{L} = \begin{cases} \overline{L} |e_1| \, sign(|e_1| - \varepsilon) \, if \ L > \lambda \\ 0 \, if \ L \le \lambda \end{cases} \qquad (3.13)$$

where $\hat{x}_1$, $\hat{x}_2$ and $\hat{x}_3$ are estimated of $x_1 = \theta$, $x_2 = \dot{\theta}$, $x_3 = M^{-1}(\theta)(-\tau_{int})$ which τ_{int} is torque interaction. $e_1 = x_1 - \hat{x}_1$, $L = \kappa_3, \kappa_2 = 5.3 L^{\frac{2}{3}}$ and $\kappa_1 = 3.34 L^{\frac{1}{3}}$. Then, $\overline{L} = 1.5$, $\varepsilon = 0.002$, $\lambda_{max} = [8, 20, 22]$, $\lambda_{min} = [-8, -5, -10]$ were selected. In this experiment, the PD control was used for position control which $K_p = [1950, 2550, 2550]$ and $K_v = [100, 100, 100]$ were selected. The block diagram of admittance control without force sensor was shown in Figure 3.18. In Figure 3.19, torque at each joint was shown with constant boundary. The snapshoot of human-robot interaction was presented in Figure 3.20. The video had been shared at link https://www.youtube.com/watch?v=fNYOB5mzc_o .
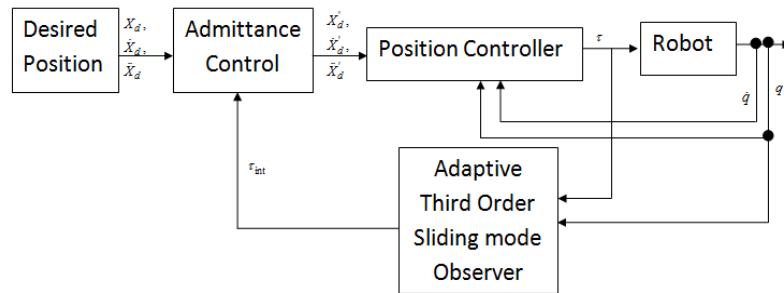
Figure 3.18. Block diagram of admittance control without force sensor
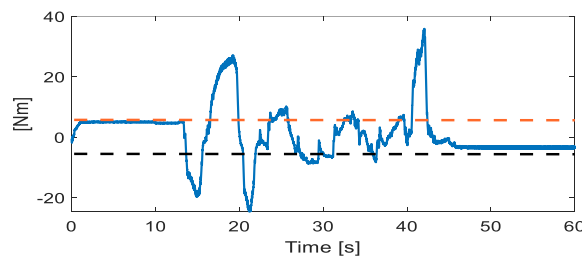
a)Joint 1
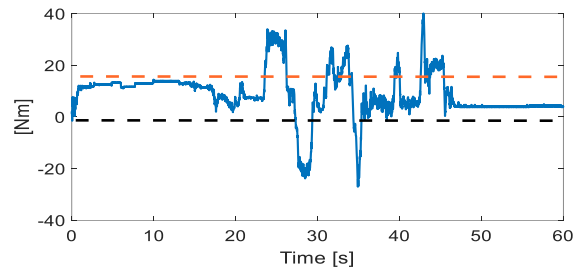
b)Joint 2
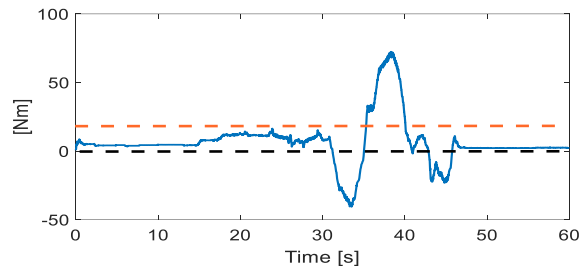
c)Joint 3

Figure 3.19. Torque estimated at each joint

Figure 3.20. Snapshoot of human push/pull robot manipulator

3.5 Conclusion

In this chapter, the controller for the hardware platform of robot manipulator was developed based on Labview. The position controller for 6-DOF in the joint space was established and verified using PID controller. In addition, the dynamic model of 3-DOF robot manipulator was presented, which is used for torque control of position controller and force controller. The identification processing increased the accuracy of position controller and was also useful for the force controller. With the complete parameters of the dynamic model of robot manipulator, the robot is able to apply fault -tolerant control based on model.

Chapter 4: Active Fault-tolerant Control Based on Supper Twisting Sliding Mode Controller

In this chapter, real implementation of an active fault-tolerant control for a robot manipulator based on the combination of an extended state observer and the super-twisting algorithm is proposed. This active fault-tolerant scheme uses an extended state observer to identify faults. Then, the fault information is used to compensate the uncertainties/disturbance and faults with the super twisting controller. Finally, the effectiveness of proposed control is verified by simulation and implementation for a 3-DOF robot manipulator. The results were illustrated that the proposed control can tolerate the relatively bigger faults due to the design of the observer and then show the better performances than the conventional super-twisting controller does.

4.1 Introduction

In these days, fault-tolerant control has paid a lot of attention to many researchers. There are wide industrial demands [29–31]. Generally, the fault-tolerant control can be categorized into two kinds 1) Passive fault-tolerant control (PFTC) [5, 6, 32] and 2) Active fault-tolerant control (AFTC) [33–35]. PFTC is based on the tolerant ability dealing with uncertainties/disturbances of control when faults occur. This method cannot handle the faults with high magnitude. Unlike the PFTC, the AFTC uses the fault estimation process to estimate faults, after that, by using the faults information, the control system was changed into the controller with a fault observer. Then the AFTC can cope with the fault with high magnitude occurred during operation of the system. In addition, the operator system can monitor and identify faults in the system.

In the field of robot manipulator control, to tolerate the faults, it can be solved by the hardware or the software. Due to the additional hardware, the price of robot has been increased and more space to install is necessary. Therefore, the algorithm based solution has been paid more attention to many researchers. Due to the ability of dealing with uncertainties/disturbances, sliding mode controller has been investigated as the PFTC[13, 36, 37]. This control technique is based on a discontinuous control action and sliding

surface. The most disadvantage of this technique is chattering behavior which comes from the high switching frequency of control signal. However, in fact, this controller is robust, easily implemented in practical system, easily stabilized with a proper choice of the sliding surface. To reduce the chattering in [38], Levant (1993) introduced the high order sliding mode concept. Then, the terminal sliding mode [39, 40], twisting controller [41, 42] and the supper-twisting controller [36, 43] are kind of second-order sliding mode algorithm. Among of them, the super-twisting algorithm is preferable over the conventional sliding mode due to the ability of reducing the chattering.

In AFTC scheme, the performance depends on the accuracy of both fault estimation and the quality of controller. By using the faults estimation process, the operator can easily monitor, identify faults and the system can deal with the high magnitude faults. It can guarantee that a robot manipulator has acceptable performance when faults occur. To estimate faults, several methods have been proposed such as parameter estimation [12], parity relations [44], and state observers [45]. Similarly to the sliding mode controller, high order sliding model observer (HSMO) has been paid a lot of attention to many researchers [35, 46, 47]. However, the most disadvantage of this technique is a proof of stability and the knowledge about bound of uncertainties/disturbances to choose the parameters of observer. To solve the proof of stability problem, Moreno J.A and co-authors had discussed about this issue [48–50]. Second issue of this method is how to choose the parameters of HSMO [51]. However, this technique requires the knowledge about bound of uncertainties/disturbances. Therefore, in [18, 28], the authors used the adaptive technique to solve this problem. As a practical substitute of HSMO, the extended state observer (ESO) is commonly used in real system. By using ESO [53], we no need to exactly know about the bound of uncertainties/disturbances. Therefore, in this chapter, the ESO is used to compensate faults and uncertainties/disturbances with the supper-twisting controller. By using this combination, the fault-tolerant control can reduce the chattering and achieve the acceptable performance when faults occur during operation the system.

In this chapter, the fault-tolerant control for robot manipulator based on the combination of the extended state observer and supper-twisting method is proposed. By using the ESO we can easily monitor the faults and no need the knowledge about bound of uncertainties/disturbances and faults. The faults were compensated in the supper-twisting controller. By using this combination, the accuracy of robot is increased and is

shown acceptable performance when faults occur. The simulation and experimental
results are shown for a 3-DOF robot manipulator to illustrate the effectiveness of the
proposed fault-tolerant control.

The rest of this chapter is as follows. Section 2, the dynamics of robot manipulator
is described. In section 3, the proposed fault-tolerant control algorithm is explained. The
simulation results of the proposed control algorithm are shown for a 3-DOF robot
manipulator in section 4. In section 5, experimental results for a3-DOF FARA-AT2 robot
manipulator are presented. Finally, some concluding remarks are given in section 6.

4.2 Dynamics model of robot manipulator

Dynamics of a n-degree of freedom robot manipulator was defined as

$$M(q)\ddot{q} + C(q,\dot{q}) + G(q) + F_f + \gamma(\text{t} - \text{T}_f)\boldsymbol{\xi} = \tau \tag{4.1}$$

where $\ddot{q}, \dot{q}, q \in \Re^n$ are the vectors of joint acceleration, velocity and position, respectively.
$M(q) \in \Re^{n \times n}, \mathbf{C}(q,\dot{q}) \in \Re^n$ and $G(q) \in \Re^n$ represent the inertia matrix, the centripetal and
Coriolis matrix, and the gravitation force, respectively. $F_f \in \Re^n$ is friction term. $\tau \in \Re^n$ is
the torque at joints. $\xi \in \Re^n$ is a vector composed of actuator faults and component faults.
$\gamma(\text{t} - \text{T}_f) \in \Re^n$ presents the time profile of the faults. T_f is the time of occurrence of the faults.

In practical, we do not exactly know about the dynamic model of robot so that
Eq.(4.1) can be written as

$$\begin{aligned}(M(q) + \Delta M(q))\ddot{q} + (C(q,\dot{q}) + \Delta C(q,\dot{q}))\dot{q} \\ + (G(q) + \Delta G(q)) + (F_f(\dot{q}) + \Delta F_f(\dot{q})) + \delta + \gamma(\text{t} - \text{T}_f)\xi = \tau\end{aligned} \tag{4.2}$$

where $\Delta M, \Delta C, \Delta G$ and ΔF are dynamic uncertainties and δ is extended disturbance. Eq.
(4.2) can be simply rewritten as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F_f(\dot{q}) + \psi = \tau \tag{4.3}$$

where $\psi = \Delta M + \Delta C + \Delta G + \Delta F + \delta + \gamma(\text{t} - \text{T}_f)\xi$ which includes uncertainties, extended
disturbances and actuator faults.

To estimate the faults and uncertainties/disturbances, the extended state observer
is used in this chapter. An active fault-tolerant control based on the combination of an
extended state observer and supper-twitting is proposed.

4.3 Proposed fault-tolerant control

4.3.1 Faults estimation based on an extended state observer

In this section, the faults estimation based on an extend(4.4)ed state observer is designed. In the position control, (4.3) can be rewritten as

$$\ddot{q} = M^{-1}(q)(\tau - H(q,\dot{q})) - M^{-1}(q)\psi \tag{4.5}$$

where $H(q,\dot{q}) = C(q,\dot{q}) + G(q) + F_f(\dot{q})$.

The dynamic model in Eq.(4.5) can be rewritten in state space as

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x_1, x_2, u) + \phi(x_1, x_2, t) \end{cases} \tag{4.6}$$

where $x_1 = q \in \Re^n$, $x_2 = \dot{q} \in \Re^n$, $u = \tau$, $f(x_1, x_2, u) = M^{-1}(q)(\tau - H(q,\dot{q}))$, $\phi = -M^{-1}(q)\psi$

An extended state observer [53] was given as

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + \dfrac{\alpha_1}{\varepsilon}(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_2 = f(x_1, \hat{x}_2, u) + \dfrac{\alpha_2}{\varepsilon}(x_1 - \hat{x}_1) + \hat{\phi} \\ \dot{\hat{\phi}} = \dfrac{\alpha_3}{\varepsilon}(x_1 - \hat{x}_1) \end{cases} \tag{4.7}$$

where $\hat{x}_1, \hat{x}_2$ and $\hat{\phi}$ are estimate of x_1, x_2 and ϕ, respectively. α_1, α_2 and α_3 are positive constant. $0 < \varepsilon < 1$, $|\dot{f}| \le L$.

Theorem 1: *Considering the system (4.6) with observer (4.7) and satisfy $0 < \varepsilon < 1$ and $|\dot{f}| \le L$ then $\hat{x}_1(t) \to x_1(t)$, $\hat{x}_2(t) \to x_2(t)$ and $\hat{\phi}(q,\dot{q},t) \to \phi(q,\dot{q},t)$ as $t \to \infty$.*

Proof: We define observer error as $\tilde{e} = [\tilde{e}_1, \tilde{e}_2, \tilde{e}_3]^T$ where $\tilde{e}_1 = (x_1 - \hat{x}_1)/\varepsilon^2$, $\tilde{e}_2 = (x_2 - \hat{x}_2)/\varepsilon$ and $\tilde{e}_3 = f - \hat{\phi}$.

We define the Lyapunov function as

$$V = \varepsilon\tilde{e}^T P\tilde{e} \tag{4.8}$$

where matrix P satisfying the Lyapunov Eq. as

$$A^T P + PA = -Q \tag{4.9}$$

in which A and Q are positive matrix.

The time derivative V in Eq.(4.8), we have

$$\begin{aligned} \dot{V} &= \varepsilon\dot{\tilde{e}}^T P\tilde{e} + \varepsilon\tilde{e}^T P\dot{\tilde{e}} \\ &= (A\tilde{e} + \varepsilon B\dot{f})^T P\tilde{e} + \tilde{e}^T P(A\tilde{e} + \varepsilon B\dot{f}) \\ &= \tilde{e}^T A^T P\tilde{e} + \varepsilon(B\dot{f})^T P\tilde{e} + \tilde{e}^T PA\tilde{e} + \varepsilon\tilde{e}^T PB\dot{f} \\ &= \tilde{e}^T(A^T P + PA)\tilde{e} + 2\varepsilon\tilde{e}^T PB\dot{f} \le -\tilde{e}^T Q\tilde{e} + 2\varepsilon\|PB\| \cdot \|\tilde{e}\| \cdot |\dot{f}| \end{aligned} \tag{4.10}$$

where B is positive matrix.

45

From Eq. (4.10), we have

$$\dot{V} \leq -\lambda_{\min}(Q)\|\tilde{e}\|^2 + 2\varepsilon L\|PB\|\|\tilde{e}\|$$

(4.11)

where $\lambda_{\min}(Q)$ is the minimum eigenvalue of Q. L is positive constant satisfy $|\dot{f}| \leq L$.

To guaranties stable of the system, $\dot{V} \leq 0$ we get observer error convergence conclusion as:

$$\|\tilde{e}\| \leq \frac{2\varepsilon L\|PB\|}{\lambda_{\min}(Q)}$$

(4.12)

4.3.2 The proposed fault-tolerant controller

In this section, the fault-tolerant controller based on the combination of an extended state observer and super-twitting is proposed. Sliding surface is designed as

$$s = \dot{e} + ce$$

(4.13)

where $e = q_d - q$ and $\dot{e} = \dot{q}_d - \dot{q} \in \Re^n$ are position error and velocity error. $c = diag(\text{c}) \in \Re^{n \times n}$ is constant.

The fault-tolerant controller is proposed as

$$\tau = \tau_0 + \tau_{smc} + \tau_{ob}$$

(4.14)

where $\tau_0 = M(q)(\ddot{q}_d + c\dot{e}) + H(q,\dot{q})$, $\tau_{smc} = M(\text{q})\left(K_1 |s|^{\frac{1}{2}} \text{sgn}(s) - \upsilon \right)$, $\dot{\upsilon} = -K_2 sign(\text{s})$, $\tau_{ob} = -M(q)\hat{\phi}$

which K_1 and $K_2 \in \Re^{n \times n}$ is diagonal positive matrices.

Theorem 2: *Consider the system (4.3) with the proposed controller (4.14), then the position error approach to zero at $t \to \infty$.*

Proof:

From Eq.(4.5) and Eq.(4.14) the system closed as

$$\begin{aligned}
\ddot{q} &= \ddot{q}_d + c\dot{e} + K_1 |s|^{\frac{1}{2}} sign(s) - \upsilon - \hat{\phi} - M^{-1}(q)\psi \\
&= \ddot{q}_d + c\dot{e} + K_1 |s|^{\frac{1}{2}} sign(s) - \upsilon - \hat{\phi} + \phi \\
&= \ddot{q}_d + c\dot{e} + K_1 |s|^{\frac{1}{2}} sign(s) - \upsilon
\end{aligned}$$

(4.15)

where $\dot{\upsilon} = -K_2 sign(\text{s})$

Differencing Eq.(4.13) with time given as

$$\dot{s} = \ddot{e} + c\dot{e}$$

(4.16)

Substituting Eq.(4.16) into Eq.(4.15) we have

$$\begin{cases} \dot{s} = -K_1 |s|^{\frac{1}{2}} sign(\text{s}) + \upsilon \\ \dot{\upsilon} = -K_2 sign(\text{s}) \end{cases}$$

(4.17)

Lyapunov function

$$V = \zeta^T P \zeta \tag{4.18}$$

where $\zeta = \left[|s|^{\frac{1}{2}} \, sign(s), \upsilon \right]^T$, P is symmetric and positive matrix and

$$\dot{\zeta} = \frac{1}{2} \frac{1}{|s|^{1/2}} \begin{bmatrix} -k_1 & 1 \\ -2k_2 & 0 \end{bmatrix} \begin{bmatrix} |s|^{\frac{1}{2}} \, sign(s) \\ \upsilon \end{bmatrix} = \frac{1}{2} \frac{1}{|s|^{1/2}} A\zeta \tag{4.19}$$

Differencing Eq.(4.18) with time given as

$$\dot{V} = \dot{\zeta}^T P \zeta + \zeta^T P \dot{\zeta} \tag{4.20}$$

Substituting Eq.(4.19) into Eq.(4.20) we have

$$
\begin{aligned}
\dot{V} &= \left(\frac{1}{2} \frac{1}{|s|^{1/2}} A\zeta \right)^T P\zeta + \zeta^T P \left(\frac{1}{2} \frac{1}{|s|^{1/2}} A\zeta \right) \\
&= \frac{1}{2} \frac{1}{|s|^{1/2}} \zeta^T \left(A^T P + AP \right) \zeta \\
&= -\frac{1}{2} \frac{1}{|s|^{1/2}} \zeta^T Q\zeta \leq 0
\end{aligned}
\tag{4.21}
$$

where Q is symmetric and positive-definite matrix

$$A^T P + AP = -Q \tag{4.22}$$

Therefore, the position error converges to zero at time infinity guaranties stable.
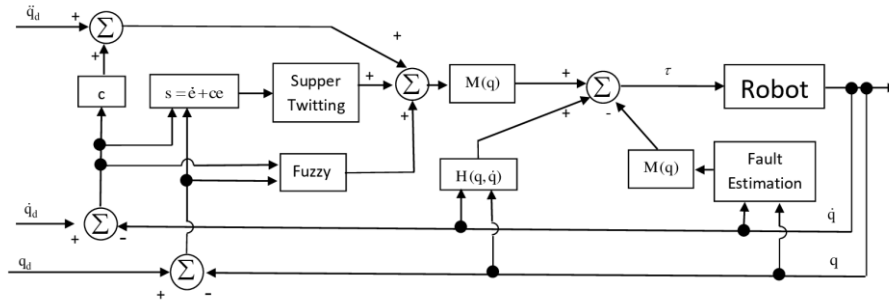
Figure 4.1. Block diagram of the proposed fault-tolerant control for robot manipulators

4.4 Simulation results

In this section, simulation results of a 3-DOF robot manipulator were shown. The robot manipulator was built on Solidwork and then imported to Mathlab SimMechainics shown in Figure 4.2. The proposed AFTC with super-twisting method are compared with conventional supper-twisting controller for the effectiveness. In this simulation, it is assumed that fault only occurs at joint 2 at 5th second.
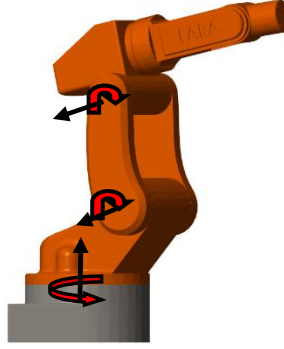
Figure 4.2.3-DOF robot manipulator in SimMechanics

The parameters of ESO are suitably chosen to be $\alpha_1 = 6, \alpha_2 = 25, \alpha_3 = 3$ and $\varepsilon = 0.01$. The parameters of proposed AFTC were suitably chosen as: $c = diag(1.5; 1.8; 2.0)$, $K_1 = diag(1.5; 1.7; 1.7)$ and $K_2 = diag(0.02; 0.07; 0.07)$. Then, the conventional Super-Twisting controller (STC) gives as

$$\tau = M(q)(\ddot{q}_d + c\dot{e}) + H(q, \dot{q}) + M(q)(K_1 |s|^{\frac{1}{2}} sign(s) - \upsilon)$$
$$\dot{\upsilon} = K_2 sign(s)$$
(4.23)

where K_1, K_2 and c are the positive diagonal matrix $\in \Re^{3 \times 3}$.

The parameters of the STC can be selected as: $c = diag(1.5; 1.8; 2.0)$, $K_1 = diag(1.5; 1.7; 1.7)$ and $K_2 = diag(0.02; 0.07; 0.07)$.

The desired tracking trajectory is given as

$$\begin{cases} q_1(t) = 0.5 \sin(t/2) \\ q_2(t) = 0.3 \sin(t) \\ q_3(t) = 0.2 \sin(t) \end{cases}$$
(4.24)

The faults function are assumed that occur at 5th second with attitude as

$$\begin{cases} \xi_1 = 0 \\ \xi_2 = -70 \cos(t-5) \\ \xi_3 = 0 \end{cases}$$
(4.25)

In Figure 4.3, the fault estimation results were shown. It can be seen that the estimator has suitable capability to estimate the fault. In Figure 4.4, the trajectory tracking errors are presented. Before fault occurs, the conventional super-twisting controller and the proposed AFTC have similar control performances. However, after fault occurs, the conventional STC has a large error profile because the ability to deal with the fault of high magnitude is limited. In Figure 4.4.b and c, the STC needs more time to recover the similar level of accuracy than the proposed AFTC does. From above results, it can be

seen that for small faults, the conventional STC (or PFTC with STC) has good performance as the proposed AFTC has. However, For the fault of high magnitude, the proposed AFTC show the better control performances that the conventional STC.

  c)   Joint 1   b) Joint 2   c)Joint 3
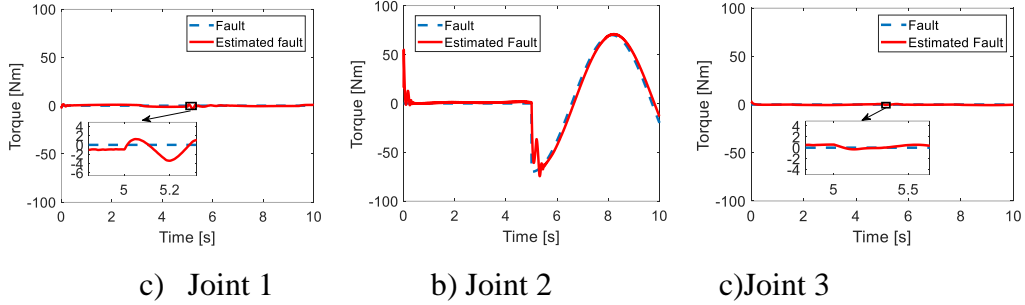
Figure 4.3.Fault estimation results

  a)   Joint 1   b) Joint 2   c)Joint 3
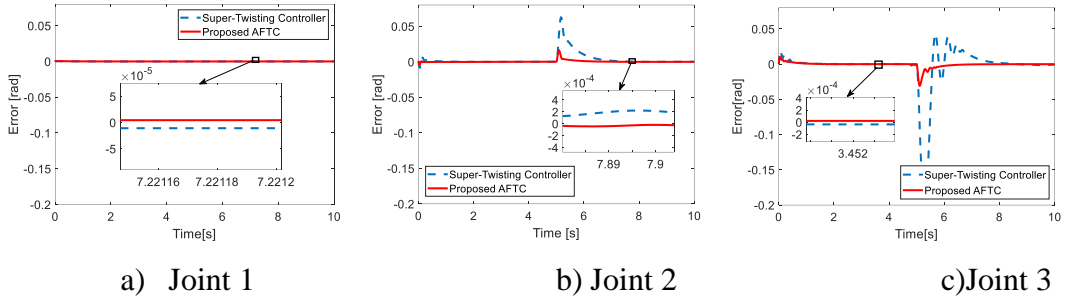
Figure 4.4. The trajectory tracking errors

4.5 Experimental results

4.5.1 Hardware setup

In this part, an experimental setup of hardware is presented. The first 3 joints of FARA-AT2 robot manipulator shown in Figure 4.5 is used for this experiment with the joints 4, 5 and 6 blocked. The robot has CSMP series motors to drive the actuated joints. The gear ratios at joints are 120:1, 120:1, 100:1, respectively. The encoder at each joint produce 8192 pulses per revolution in quadratic mode. The controller runs on Labview FPGA NI-PXI-8110 and NI-PXI-7842R PXI card. The frequency of control loop is set up at 500Hz. The controller NI-PXI-8110 runs window operating system. The desired tracking trajectory is given as

$$q_i(t) = \frac{\pi}{6}\sin(\pi t) \quad (i = 1, 2, 3) \tag{4.26}$$

Figure 4.5. Real-system 3-DOF robot manipulator

4.5.2 Experimental results

In this section, the proposed control algorithm will be implemented with conventional super-twisting control for the comparison as done in the simulation study.

The parameters of conventional super twisting control (STC) in(4.23) are chosen as $c = diag(4;5;5), K_1 = diag(1.60;2.70;2.60)$ and $K_2 = diag(2.5;2.5;2.5)$ .The parameters of an observer can be selected as $\alpha_1 = 0.2, \alpha_2 = 6, \alpha_3 = 0.3$ and $\varepsilon = 0.01$.The parameters of the proposed algorithm are suitably chosen as $c = diag(4;5;5)$, $K_1 = diag(1.60, 2.70, 1.60)$, $K_2 = diag(2.5, 2.5, 2.5)$.

The fault functions are assumed that occur at 10th second with attitude as

$$\begin{cases} \xi_1 = 8\sin(\frac{2\pi}{3}(t-10)) + 50(t-10) \\ \xi_2 = 30\sin(\frac{2\pi}{3}(t-10)) \\ \xi_3 = 40(t-10) \end{cases} \quad (4.27)$$
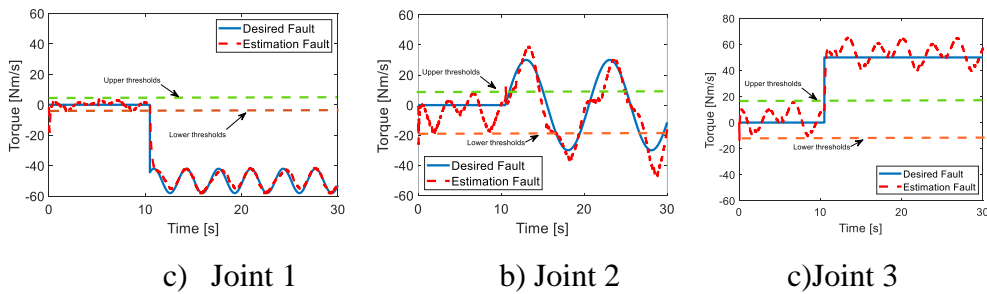
c)   Joint 1   b) Joint 2   c)Joint 3
Figure 4.6.The results of fault estimation

The fault estimation results were shown in Figure 4.6. The error at each joint from this experiment was presented in Figure 4.7. It can be seen that before and after faults occur, the tracking accuracy of the proposed AFTC is better than the conventional super-

twisting controller because the extended state observer works well to compensate the uncertainties/disturbances and faults with STC.

a)   Joint 1   b) Joint 2   c) Joint 3
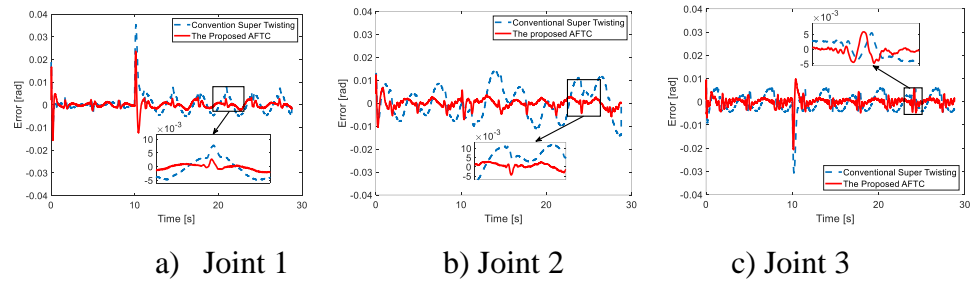
Figure 4.7. The error at joints

4.6 Conclusion

In this chapter, the real implementation of active fault-tolerant control based on the combination of an extended state observer and super-twisting controller for a robot manipulator was proposed. The proposed controller improves the ability to deal with the uncertainties/disturbances and the big faults. From the simulation study, it can be seen that the conventional super twisting controller has similar level of accuracy as the proposed AFTC does, when no faults. However, in the real experiment with no faults, the proposed AFTC has better control accuracy than the conventional STC does. It means that the experimental system has a certain amount of uncertainties/disturbances, even in case of no fault. Here, the observer is still effective in this stage. Then, in case of big faults, the effectiveness of the observer becomes more obvious. Therefore, the proposed AFTC has shown better control performances in case of large faults than the conventional STC. As demonstrated in real implementation, the proposed AFTC can be useful in real control system and be expected to have acceptable performances even when large faults occur.

Chapter 5: Implementation of Fault-tolerant Control Based on Synchronous Sliding Mode Control

In this chapter, an active fault-tolerant control for a robot manipulator based on synchronous sliding mode is proposed. As the synchronization errors approach zero, the joint errors tend to become equal and also approach zero. Therefore, the synchronization technique is inherently effective for a fault-tolerant controller. To demonstrate such a system, the following implementation is presented. First, an estimator was designed with an extended state observer to estimate uncertainties/disturbances along with faults/failures. The estimator signal was used for an online compensator in the controller. A fault-tolerant controller with a combination of synchronous sliding mode technique and estimator was proposed. The stability of the system was established using Lyapunov theory. Finally, fault-tolerant control was implemented in a three degree-of-freedom robot manipulator and compared to the conventional sliding mode control. This comparison shows the effectiveness of the proposed active fault-tolerant control with synchronous sliding mode technique.

5.1 Introduction

During the past two decades, fault detection and fault-tolerant control (FTC) have become attractive research subjects that can be used to improve system reliability and guarantee system stability in all situations. Implementation of fault-tolerant control in robot manipulators has encountered a number of challenges due to high nonlinearities, dynamic uncertainties, and external disturbances. In addition, the time delay inherent to mechanical systems also affects the performance of FTC. FTC strategies can be divided into two categories: passive FTC (PFTC) [5, 6] and active FTC (AFTC) [2, 34]. In PFTC, the control performances mainly depend on the robust capability dealing with uncertainties/disturbances of the controller such as sliding mode control [7] or adaptive control [40, 54]. In this strategy, the controller does not have a faults estimation process. The advantage of PFTC is the fast response when faults occur, because it does not need time to estimate faults. However, the ability to deal with high magnitude faults is limited.

Unlike PFTC, AFTC reconfigures the control system based on the estimation process. The fault information from the estimation process is used to compensate the conventional controller. The disadvantage of this strategy is slow response after faults occur, which leads to the occurrence of a picking phenomenon, because the controller needs the time to estimate the faults. Most studies in active fault-tolerant control [1, 3, 34]have focused on increasing the ability to deal with uncertainties/disturbances of the controller. Therefore, the performance degradation of the system with the AFTC strategy due to the slow response still remains an open problem. However, AFTC has better ability when dealing with high magnitude faults than PFTC. As above-mentioned, using PFTC or AFTC depends on the characteristics of the system and type of faults. In robot manipulator control, AFTC generally outperforms PFTC because AFTC includes a fault estimation (FE) step. The estimated faults can be compensated for by online controller reconfiguration. In this way, the stability and acceptable performance of the robot can be maintained.

In AFTC, the quality of the fault-tolerant control depends on the accuracy of both FE and the system reconfiguration after faults occur. In fault estimation processing, several techniques [12, 13, 44, 55–57] have been used. The parameter estimation method [12] is used to early detect faults applied for dynamic linear and nonlinear systems. The parity relations [44] used the parity Eq.s to be combined with the least-square method to estimate faults. The sliding mode observers [13] have been given lot of attention by researchers. However, this method is limited in real applications due to the disadvantages such as the chattering phenomenon, the requirement of the knowledge of the fault's bound to choose the observer parameters, and the stability issue. In addition, other techniques such as the Kalman Filter [55], zonotope [56], and nonlinear observer [57] were developed to estimate the faults as well as uncertainties/disturbances. After faults are estimated, they are compensated for by using various control strategies [19, 58, 59]. In this study, an extended state observer [60] was adopted for on-line observation of the dynamic uncertainties, disturbances, and faults. An extended state observer (ESO) is a simple technique for estimating faults in which simply adjusting the observer parameters leads to simple application in real systems, and the observer can detect and isolate faults without a fault diagnosis process. In addition, the upper bound of faults does not have to be exactly known in the design of ESO.

A synchronization technique based on cross-error was first introduced by Y. Koren[61] in the 1980s for a computerized numerical control (CNC) machine. In his idea, a CNC with independent axes control is extended to enable the control of each access to consider the effects of the other axes through cross-errors. L. Feng et al. [62] proposed cross-coupling control for mobile robots. He suggested that minimization of the most significant error leads to coordination of the motion of the two wheels. Lu Ren et al. [63] introduced synchronization errors, a new type of cross-coupling error, in controlling a parallel robot manipulator. Synchronization control has also been applied in multi-robot cooperation [24, 64]. Synchronization techniques combined with the sliding mode method have attracted the interest of many researchers. The position and velocity synchronization error have been used instead of position and velocity error, respectively, in sliding mode control structures [65]. Zhang et al. [66] also proposed a robust synchronous control based on a sliding mode variable structure for multi-motors. These studies [24, 64–66] have applied synchronization control techniques to a parallel robot, multi-robot cooperation, and multi-motors to improve trajectory tracking performance. This work is interested in addressing a slow response issue by using the synchronization control technique, which can make the position error at each joint equal. Therefore, the system can quickly respond to a fault due to the constraint of synchronization control before the controller has the feedback information from the fault estimation process. Considering the dynamic coupling effects between actuators and the upper bound of the uncertainties, the effectiveness of synchronization techniques might be somewhat limited to improvements of the trajectory tracking control. However, synchronization techniques become more effective in critical conditions, which lead to applying synchronization techniques to the fault-tolerant control. The contributions of this chapter are summarized as follows:

1) Synchronization techniques are applied to fault-tolerant control for robot manipulators for the first time. Compared to active fault-tolerant control using conventional sliding mode control, the proposed system has achieved higher accuracy, robustness, and faster system reconfiguration when faults occur. These results confirm that synchronization techniques are very effective in fault-tolerant control.

2) The stability of the proposed AFTC with the synchronous sliding mode technique is demonstrated using analysis via Lyapunov theory.

54

3) Based on the extended state observer, the proposed controller can easily monitor faults without detection and isolation processes. This feature is helpful in maintenance systems as well as maintenance planning systems.

4) The experimental results show that the proposed control can be easily applied to a real system with robust performance.

In this chapter, we present a fault-tolerant control for robot manipulators based on synchronous sliding mode control. Section 2 describes the robot dynamic models. Section3 explains the process of fault estimation with the extended state observer for robot manipulators. In Section 4, a fault-tolerant control based on synchronous sliding mode control is presented and its stability is demonstrated. In Section 5, a simulation study of the proposed fault-tolerant control for a 3-DOF manipulator was conducted to show the method's effectiveness. In Section 6, real implementation of the FTC for a 3-DOF FARA AT2 robot was carried out in two cases: a single fault and multiple faults. Finally, conclusions are discussed in Section 7.

5.2 Dynamics Model of Robot Manipulators

The dynamics of an n-degree of freedom robot manipulator was defined [27] as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F_f(\dot{q}) = \tau \tag{5.1}$$

where $\ddot{q}, \dot{q}, q \in \Re^n$ are the vectors of joint acceleration, velocity, and position, respectively. $M(q) \in \Re^{n \times n}, C(q,\dot{q}) \in \Re^n$, and $G(q) \in \Re^n$ represent the inertia matrix, the centripetal and Coriolis matrix, and the gravitation force, respectively. $F_f \in \Re^n$ is the friction term and $\tau \in \Re^n$ is the torque at the joints.

In practice, the dynamics model of a robot is not exactly known, so Eq. (5.1)can be written as

$$(\mathbf{M}(q) + \Delta M(q))\ddot{q} + (\mathbf{C}(q,\dot{q}) + \Delta C(q,\dot{q}))\dot{q} + (\mathbf{G}(q) + \Delta G(q)) + (\mathbf{F}_f(\dot{q}) + \Delta F_f(\dot{q})) + \delta = \tau \tag{5.2}$$

where $\Delta M, \Delta C, \Delta G$ and ΔF are unknown dynamic uncertainties and δ is the unknown external disturbances. $\mathbf{M}(q), \mathbf{C}(q,\dot{q}), \mathbf{G}(q)$ and $\mathbf{F}(\dot{q})$ are estimated of $M(q), C(q,\dot{q}), G(q)$ and $F(\dot{q})$. Thus, Eq. (5.2) can be simply rewritten as

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q,\dot{q})\dot{q} + \mathbf{G}(q) + \mathbf{F}_f(\dot{q}) + \mathbf{\psi} = \tau \tag{5.3}$$

where $\mathbf{\psi} = \Delta M\ddot{q} + \Delta C\dot{q} + \Delta G + \Delta F + \delta$.

Properties:

$$\mu_1 I_n \leq \mathbf{M}(q) \leq \mu_2 I_n$$

$$\frac{1}{\mu_1} I_n \leq \mathbf{M}^{-1}(q) \leq \frac{1}{\mu_2} I_n$$

In general, actuator faults can be divided into two types: bias faults and gain faults. In a robot manipulator, these are known as loss of effectiveness and lock-in-place faults. In practice, both kinds of actuator faults commonly occur. Actuator bias faults can be generally described as

$$\tau^b = \tau + \mathbf{f}(t) \tag{5.4}$$

where $\mathbf{f}(t) = [f_1, f_2, ... f_n]^T \in \mathfrak{R}^n{}_{(i=1,2,...,n)}$ denotes a bounded signal. τ^b is the torque at the joint when faults occur. Due to loss of effectiveness, actuator gain fault can be described as

$$\tau^g = (I - \boldsymbol{\rho}(t))\tau \tag{5.5}$$

where $\boldsymbol{\rho}(t) = diag(\rho_i(t)) \in \mathfrak{R}^{n \times n}, 0 \leq \rho_i(t) < 1 \ (i = 1, 2, ..., n)$, which is unknown, denotes the remaining control rate. $I \in \mathfrak{R}^{n \times n}$ is the identity matrix. Therefore, the total torque including two kinds of actuator faults can be comprehensively described as

$$\tau^t = (I - \boldsymbol{\rho}(t))\tau + \mathbf{f}(t) \quad (t > t_f) \tag{5.6}$$

where t_f is the time of occurrence of each fault.

Substituting Eq. (5.6) into Eq. (5.3), the dynamics model of an n-degree of freedom robot manipulator with actuator faults can be written as

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q, \dot{q})\dot{q} + \mathbf{G}(q) + \mathbf{F}_f(\dot{q}) + \boldsymbol{\psi} = (I - \boldsymbol{\rho}(t))\tau + \mathbf{f}(t) \tag{5.7}$$

Assumption1: There exist known positive constants $\bar{\rho}_i, \bar{\bar{\rho}}_i, \bar{\mathbf{f}}, \bar{\bar{\mathbf{f}}}$ such that $|\rho_i(t)| \leq \bar{\rho}_i$, $|\dot{\rho}_i(t)| \leq \bar{\bar{\rho}}_i, |\mathbf{f}| \leq \bar{\mathbf{f}}$, and $|\dot{\mathbf{f}}| \leq \bar{\bar{\mathbf{f}}}$.

5.3 Fault Estimation Using an Extended State Observer

In this section, an extended state observer of uncertainties or disturbances and faults or failures is presented.

The dynamic model of the robot manipulator of Eq. (5.7) can be rewritten in state space as

$$\ddot{q} = \mathbf{M}^{-1}(q)(\tau - \mathbf{H}(q, \dot{q})) - \mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q, \dot{q}, \ddot{q}, \tau, t) \tag{5.8}$$

where $\mathbf{H}(q, \dot{q}) = \mathbf{C}(q, \dot{q})q + \mathbf{G}(q) + \mathbf{F}_f(\dot{q})$. $\boldsymbol{\zeta}(q, \dot{q}, \ddot{q}, \tau, t) = \boldsymbol{\rho}(t)\tau + \boldsymbol{\psi}(q, \dot{q}, \ddot{q}, t) - \mathbf{f}(t)$ represents the uncertaintiesor disturbances and faults/failures.

In state space, the dynamic model of Eq. (8) becomes

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x_1, x_2, \tau) + \phi(x_1, x_2, \tau, t) \end{cases} \tag{5.9}$$

where $\quad x_1 = q \in \Re^n$, $x_2 = \dot{q} \in \Re^n$, $f(x_1, x_2, \tau) = \mathbf{M}^{-1}(q)(\tau - \mathbf{H}(q, \dot{q}))$, $\quad$ and

$\phi(x_1, x_2, \tau, t) = -\mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q, \dot{q}, \ddot{q}, \tau, t)$.

An extended state observer [60] is given as

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + \dfrac{\alpha_1}{\varepsilon}(x_1 - \hat{x}_1) \\[2mm] \dot{\hat{x}}_2 = \hat{f}(x_1, \hat{x}_2, \tau) + \dfrac{\alpha_2}{\varepsilon^2}(x_1 - \hat{x}_1) + \hat{\phi} \\[2mm] \dot{\hat{\phi}} = \dfrac{\alpha_3}{\varepsilon^3}(x_1 - \hat{x}_1) \end{cases} \tag{5.10}$$

where $\hat{x}_1, \hat{x}_2, \hat{f}$ and $\hat{\phi}$ are estimates of x_1, x_2, f and ϕ, respectively; α_1, α_2, and α_3 are positive constants; polynomial $s^3 + \alpha_1 s^2 + \alpha_2 s + \alpha_3$ is Hurwitz; and $0 < \varepsilon < 1$.

Theorem 1: *Considering the system (9) with observer (10) and satisfying $0 < \varepsilon < 1$ and $|\dot{\phi}| \le L$, then $\hat{x}_1(t) \to x_1(t)$, $\hat{x}_2(t) \to x_2(t)$ and $\hat{\phi}(q, \dot{q}, t) \to \phi(q, \dot{q}, t)$ as $t \to \infty$*

Proof: We define the observer error as $\tilde{e} = [\tilde{e}_1, \tilde{e}_2, \tilde{e}_3]^T$ where $\tilde{e}_1 = (x_1 - \hat{x}_1)/\varepsilon^2$, $\tilde{e}_2 = (x_2 - \hat{x}_2)/\varepsilon$ and $\tilde{e}_3 = \phi - \hat{\phi}$. From assumption 1and Eq. (9), it can be seen that there will exist a value L satisfying $|\dot{\phi}| \le L$. The value of L does not need to be exactly known. The function f is a known function of (x_1, x_2, τ), and we can take $\hat{f} = f$ as [60].

We define the Lyapunov function as

$$V = \varepsilon \tilde{e}^T P \tilde{e} \tag{5.11}$$

where the matrix P is unique, symmetric, and positive.

The time derivative of V in Eq. (5.11) is

$$\dot{V} = \varepsilon \dot{\tilde{e}}^T P \tilde{e} + \varepsilon \tilde{e}^T P \dot{\tilde{e}} \tag{5.12}$$

where

$$\varepsilon \dot{\tilde{e}} = \begin{bmatrix} \varepsilon \dot{\tilde{e}}_1 \\ \varepsilon \dot{\tilde{e}}_2 \\ \varepsilon \dot{\tilde{e}}_3 \end{bmatrix} = \begin{bmatrix} \varepsilon \dfrac{\dot{x}_1 - \dot{\hat{x}}_1}{\varepsilon} \\[2mm] \varepsilon \dfrac{\dot{x}_2 - \dot{\hat{x}}_2}{\varepsilon} \\[2mm] \varepsilon(\dot{\phi} - \dot{\hat{\phi}}) \end{bmatrix} = \begin{bmatrix} -\dfrac{\alpha_1}{\varepsilon^2}(x_1 - \hat{x}_1) + \dfrac{1}{\varepsilon}(x_2 - \hat{x}_2) \\[2mm] -\dfrac{\alpha_2}{\varepsilon^2}(x_1 - \hat{x}_1) + (\phi - \hat{\phi}) \\[2mm] \varepsilon \dot{\phi} - \dfrac{\alpha_3}{\varepsilon^2}(x_1 - \hat{x}_1) \end{bmatrix} \tag{5.13}$$

$$= \begin{bmatrix} -\alpha_1 \tilde{e}_1 + \tilde{e}_2 \\ -\alpha_2 \tilde{e}_1 + \tilde{e}_3 \\ -\alpha_3 \tilde{e}_1 + \varepsilon \dot{\phi} \end{bmatrix} = A\tilde{e} + \varepsilon B \dot{\phi}$$

where

$$A = \begin{bmatrix} -\alpha_1 & 1 & 0 \\ -\alpha_2 & 0 & 1 \\ -\alpha_3 & 0 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

It is easy to see that A is Hurwitz, so there is a symmetric positive definite matrix Q satisfying the Lyapunov Eq.

$$A^T P + PA = -Q \tag{5.14}$$

Substituting Eq.s (5.13) and (5.14) into Eq. (5.12), we have

$$\begin{aligned}
\dot{V} &= \varepsilon \dot{\tilde{e}}^T P \tilde{e} + \varepsilon \tilde{e}^T P \dot{\tilde{e}} \\
&= (A\tilde{e} + \varepsilon B\dot{\phi})^T P \tilde{e} + \tilde{e}^T P(A\tilde{e} + \varepsilon B\dot{\phi}) \\
&= \tilde{e}^T A^T P \tilde{e} + \varepsilon (B\dot{\phi})^T P \tilde{e} + \tilde{e}^T PA\tilde{e} + \varepsilon \tilde{e}^T PB\dot{\phi} \\
&= \tilde{e}^T (A^T P + PA)\tilde{e} + 2\varepsilon \tilde{e}^T PB\dot{\phi} \\
&\leq -\tilde{e}^T Q\tilde{e} + 2\varepsilon \|PB\| \cdot \|\tilde{e}\| \cdot |\dot{\phi}|
\end{aligned} \tag{5.15}$$

where L is the positive constant that satisfies $|\dot{\phi}| \leq L$. Therefore,

$$\dot{V} \leq -\lambda_{\min}(Q)\|\tilde{e}\|^2 + 2\varepsilon L \|PB\| \|\tilde{e}\| \tag{5.16}$$

To guaranty stability of the system, we impose $\dot{V} \leq 0$, so the observer error convergence is given as

$$\|\tilde{e}\| \leq \frac{2\varepsilon L \|PB\|}{\lambda_{\min}(Q)} \tag{5.17}$$

5.4 Fault-tolerant Control with Synchronous Sliding Mode Control

In this section, the fault-tolerant control based on synchronous sliding mode control is proposed. Some definitions are necessary to propose the fault-tolerant control law. Synchronization error [67] is given as

$$\begin{aligned}
\varepsilon_1 &= \psi_1 e_1 - \psi_2 e_2 \\
\varepsilon_2 &= \psi_2 e_2 - \psi_3 e_3 \\
&\vdots \\
\varepsilon_{n-1} &= \psi_{n-1} e_{n-1} - \psi_n e_n \\
\varepsilon_n &= \psi_n e_n - \psi_1 e_1
\end{aligned} \tag{5.18}$$

where $e_i (i=1,2,...,n)$ is the error at each joint and $\psi_i (i=1,2,...,n)$ is the corresponding positive gain. The synchronization control goal [67] is stated as $e_1 = e_2 = e_3 = ... = e_n$ and $e_i \to 0$ at $t \to \infty$. In a traditional controller, only the position error converges to zero, but in the synchronization control, the kinematic relationship among the errors as well as the position error converges to zero. For ease of practical implementation, $\psi_i = 1$ is chosen.

Then, Eq. (5.18) can be written as

$$\varepsilon_1 = e_1 - e_2$$
$$\varepsilon_2 = e_2 - e_3$$
$$\vdots$$
$$\varepsilon_n = \varepsilon_n - \varepsilon_1$$

$$(5.19)$$

Cross-coupling error [67] is given as

$$\xi_1 = \gamma_1 \varepsilon_1 - \gamma_n \varepsilon_n$$
$$\xi_2 = \gamma_2 \varepsilon_2 - \gamma_1 \varepsilon_1$$
$$\vdots$$
$$\xi_n = \gamma_n \varepsilon_n - \gamma_{n-1} \varepsilon_{n-1}$$

$$(5.20)$$

where $\gamma_i (i = 1, 2, ..., n)$ is the positive gain. In this chapter, $\gamma_i = 1$ was chosen. Then, Eq. (5.20) can be written as

$$\xi_1 = \varepsilon_1 - \varepsilon_n$$
$$\xi_2 = \varepsilon_2 - \varepsilon_1$$
$$\vdots$$
$$\xi_n = \varepsilon_n - \varepsilon_{n-1}$$

$$(5.21)$$

The coupling position error [67], which includes the position and synchronization errors, is defined as

$$E_i = \mu_i e_i + \eta_i \int \xi_i dt \qquad (5.22)$$

where η_i and μ_i are positive gains. The synchronous sliding surface [66] is defined as

$$S = \dot{E} + cE \qquad (5.23)$$

where $S = [S_1, S_2, ..., S_n]^T \in \Re^n$; $c \in \Re^{n \times n}$ is the diagonal positive matrix; and $E = [E_1, E_2, ..., E_n]^T \in \Re^n$ with E_i defined in Eq. (5.22). For ease of implementation, the synchronous sliding surface is rewritten as

$$s = \dot{e} + ce + \alpha\xi + \kappa \int \xi dt \qquad (5.24)$$

where $s = [S_1 / \mu_1, S_2 / \mu_2, ..., S_n / \mu_n]^T \in \Re^n$, $\alpha = diag(\eta_i / \mu_i) \in \Re^{n \times n}$ and $\kappa = diag(c_i \eta_i / \mu_i) \in \Re^{n \times n}$ $(i = 1, 2, ..., n)$ are positive gain matrices.

Here, the fault-tolerant control law is proposed as

$$\tau = \tau_0 + \tau_{SSMC} + \tau_{ob} \qquad (5.25)$$

where $\tau_0 = \mathbf{M}(q)\ddot{q}_d + \mathbf{H}(q, \dot{q})$, $\tau_{ssmc} = \mathbf{M}(q)(c\dot{e} + \alpha\dot{\xi}) + k_1 sgn(s) + k_2 s + k_3 \xi$, and $\tau_{ob} = -\mathbf{M}(q)\hat{\phi}$.

Theorem 2: *The system described in* (5.7)*, using controller specified in* (5.25) *guarantees that* $e \to 0$ *and* $\varepsilon \to 0$ *as time* $t \to \infty$ *under the condition* $k_3 > \kappa$.

Proof:

Following [22], the Lyapunov function can be selected as

$$V = \frac{1}{2}s^T s + \frac{1}{2}\sum_{i=1}^{n}\Lambda_\varepsilon \varepsilon_i^2 + \frac{1}{2}\sum_{i=1}^{n}\kappa_\varepsilon \Lambda_\varepsilon \left(\int \varepsilon_i - \varepsilon_{i-1}\right)^2 dt \geq 0 \tag{5.26}$$

The time derivative of V is therefore

$$\dot{V} = s^T \dot{s} + \sum_{i=1}^{n}\dot{\varepsilon}_i \Lambda_\varepsilon \varepsilon + \sum_{i=1}^{n}\kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i-1})\int (\varepsilon_i - \varepsilon_{i-1})dt$$

$$= A_1 + \sum_{i=1}^{n}\dot{\varepsilon}_i \Lambda_\varepsilon \varepsilon + \sum_{i=1}^{n}\kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i-1})\int (\varepsilon_i - \varepsilon_{i-1})dt \tag{5.27}$$

where $A_1 = s^T \dot{s}$. Λ_ε and κ_ε are positive gain.

Differentiation of Eq. (5.24) gives

$$\dot{s} = \ddot{e} + c\dot{e} + \alpha\dot{\xi} + \kappa\xi = \ddot{q}_d - \ddot{q} + c\dot{e} + \alpha\dot{\xi} + \kappa\xi \tag{5.28}$$

Substituting Eq. (5.8) into Eq. (5.28) yields

$$\dot{s} = \ddot{q}_d + c\dot{e} + \alpha\dot{\xi} + \kappa\xi - \mathbf{M}^{-1}(q)(\tau - \mathbf{H}(q,\dot{q})) - \mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q,\dot{q},\tau,t) \tag{5.29}$$

Substituting Eq. (5.25) into Eq. (5.29), we have

$$\dot{s} = \ddot{q}_d + c\dot{e} + \alpha\dot{\xi} + \kappa\xi - \mathbf{M}^{-1}(q)\{\mathbf{M}(q)\ddot{q}_d + \mathbf{H}(q,\dot{q}) + \mathbf{M}(q)(c\dot{e}+\alpha\dot{\xi}) + k_1 sgn(s) + k_2 s + k_3\xi$$

$$- \mathbf{M}(q)\hat{\phi} - \mathbf{H}(q,\dot{q})\} + \mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q,\dot{q},\ddot{q},\tau,t) \tag{5.30}$$

$$= -\mathbf{M}^{-1}(q)k_1 sgn(s) - \mathbf{M}^{-1}(q)k_2 s - (\kappa + \mathbf{M}^{-1}(q)k_3)\xi + \hat{\phi} + \mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q,\dot{q},\ddot{q},\tau,t)$$

It can be seen that when $\hat{\phi} \to \phi$ at $t \to \infty$ $\hat{\phi} = -\mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q,\dot{q},\tau,t)$ then Eq. (5.30) becomes

$$\dot{s} = \mathbf{M}^{-1}(q)(-k_1 sgn(s) - k_2 s - (k_3 - \kappa)\xi \tag{5.31}$$

From Properties, we have

$$\dot{s} \leq \frac{1}{\mu_2}I(-k_1 sgn(s) - k_2 s - (k_3 - \kappa)\xi) \tag{5.32}$$

and

$$A_1 = s^T \dot{s} \leq s^T \frac{1}{\mu_2}I(-k_1 sgn(s) - k_2 s - (k_3 - \kappa)\xi)$$

$$= -\frac{1}{\mu_2}Ik_1|s| - \frac{1}{\mu_2}Ik_2\|s\|^2 - s^T \frac{1}{\mu_2}I(k_3 - \kappa)\xi \tag{5.33}$$

$$= -\frac{1}{\mu_2}Ik_1|s| - \frac{1}{\mu_2}Ik_2\|s\|^2 - A_2$$

In Eq. (5.33), we have

$$A_2 = s^T \Lambda \xi ,$$

where $\Lambda = \frac{1}{\mu_2}I(k_3 - \kappa)$. Then,

$$A_2 = s^T \Lambda \xi = \sum_{i=1}^{n} s_i \Lambda_i \xi_i = \sum_{i=1}^{n} s_i \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i-1})$$

$$= s_1 \Lambda_\varepsilon (\varepsilon_1 - \varepsilon_n) + s_2 \Lambda_\varepsilon (\varepsilon_2 - \varepsilon_1) + s_3 \Lambda_\varepsilon (\varepsilon_3 - \varepsilon_2) + ... + s_n \Lambda_\varepsilon (\varepsilon_n - \varepsilon_1)$$

$$= (s_1 - s_2)\Lambda_\varepsilon \varepsilon_1 + (s_2 - s_3)\Lambda_\varepsilon \varepsilon_2 + (s_3 - s_3)\Lambda_\varepsilon \varepsilon_3 + ... + (s_n - s_1)\Lambda_\varepsilon \varepsilon_n$$

$$= \sum_{i=1}^{n} (s_i - s_{i+1})\Lambda_\varepsilon \varepsilon_i$$

(5.34)

where $\Lambda_\varepsilon = \Lambda_i = \Lambda_{i+1}$ $(i = 1, 2, .., n; n+1 = 1)$.

Using Eq.s (5.21) and (5.24), we have

$$s_i - s_{i+1} = \dot{e}_i + c_i e_i + \alpha_i(\varepsilon_i - \varepsilon_{i-1}) + \kappa_i \int (\varepsilon_i - \varepsilon_{i-1})dt - \dot{e}_{i+1} - c_{i+1}e_{i+1} - \alpha_{i+1}(\varepsilon_{i+1} - \varepsilon_i) - \kappa_{i+1} \int (\varepsilon_{i+1} - \varepsilon_i)dt$$

$$= (\dot{e}_i - \dot{e}_{i+1}) + (c_i + c_{i+1})(e_i - e_{i+1}) + (\alpha_i + \alpha_{i+1})(2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1}) + (\kappa_i + \kappa_{i+1})\int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt$$

$$= \dot{\varepsilon}_i + c_{i+1}^i \varepsilon_i + \alpha_{i+1}^i (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1}) + \kappa_{i+1}^i \int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt$$

(5.35)

where $c_{i+1}^i = c_i + c_{i+1}$, $\alpha_{i+1}^i = \alpha_i + \alpha_{i+1}$ and $\kappa_{i+1}^i = \kappa_i + \kappa_{i+1}$ $(i = 1, 2, ..., n$ and $n+1 = 1)$.

Substituting Eq. (5.35) into Eq. (5.34) gives

$$A_2 = \sum_{i=1}^{n} (s_i - s_{i+1})\Lambda_\varepsilon \varepsilon_i$$

$$= \sum_{i=1}^{n} \left(\begin{array}{l} \dot{\varepsilon}_i + c_{i+1}^i \varepsilon_i + \alpha_{i+1}^i (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1}) \\ + \kappa_{i+1}^i \int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt \end{array} \right) \Lambda_\varepsilon \varepsilon_i$$

(5.36)

$$= \sum_{i=1}^{n} \dot{\varepsilon}_i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} \varepsilon_i c_{i+1}^i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})\alpha_{i+1}^i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} \kappa_{i+1}^i \Lambda_\varepsilon \varepsilon_i \int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt$$

$$= \sum_{i=1}^{n} \dot{\varepsilon}_i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} \varepsilon_i c_{i+1}^i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})\alpha_\varepsilon \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} \kappa_\varepsilon \Lambda_\varepsilon \varepsilon_i \int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt$$

where $\alpha_\varepsilon = \alpha_i / 2 = \alpha_{i+1} / 2$, $\kappa_\varepsilon = \kappa_i / 2 = \kappa_{i+1} / 2$ $(i = 1, 2, ..., n$ and $n+1 = 1)$.

In Eq. (5.36), we can simplify the expression with

$$\sum_{i=1}^{n} (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})\alpha_\varepsilon \Lambda_\varepsilon \varepsilon_i = \alpha_\varepsilon \Lambda_\varepsilon (2\varepsilon_1 - \varepsilon_n - \varepsilon_2)\varepsilon_1 + \alpha_\varepsilon \Lambda_\varepsilon (2\varepsilon_2 - \varepsilon_1 - \varepsilon_3)\varepsilon_2 + ... + \alpha_\varepsilon \Lambda_\varepsilon (2\varepsilon_n - \varepsilon_{n-1} - \varepsilon_1)\varepsilon_n$$

$$= \alpha_\varepsilon \Lambda_\varepsilon (\varepsilon_1^2 + \varepsilon_2^2 - 2\varepsilon_1 \varepsilon_2) + \alpha_\varepsilon \Lambda_\varepsilon (\varepsilon_2^2 + \varepsilon_3^2 - 2\varepsilon_2 \varepsilon_3) + ... + \alpha_\varepsilon \Lambda_\varepsilon (\varepsilon_n^2 + \varepsilon_1^2 - 2\varepsilon_n \varepsilon_1)\varepsilon_n$$

$$= \sum_{i=1}^{n} \alpha_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i+1})^2$$

(5.37)

and

$$\sum_{i=1}^{n} \kappa_\varepsilon \Lambda_\varepsilon \varepsilon_i \int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt = \kappa_\varepsilon \Lambda_\varepsilon \varepsilon_1 \int (2\varepsilon_1 - \varepsilon_n - \varepsilon_2)dt + \kappa_\varepsilon \Lambda_\varepsilon \varepsilon_2 \int (2\varepsilon_2 - \varepsilon_1 - \varepsilon_3)dt$$

$$+ ... + \kappa_\varepsilon \Lambda_\varepsilon \varepsilon_n \int (2\varepsilon_n - \varepsilon_{n-1} - \varepsilon_1)dt$$

$$= \kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_1 - \varepsilon_n) \int (\varepsilon_1 - \varepsilon_n)dt + \kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_2 - \varepsilon_1) \int (\varepsilon_2 - \varepsilon_1)dt$$

(5.38)

$$+ ... + \kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_n - \varepsilon_1) \int (\varepsilon_n - \varepsilon_1)dt$$

$$= \sum_{i=1}^{n} \kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i-1}) \int (\varepsilon_i - \varepsilon_{i-1})dt$$

Substituting Eq.s (5.37) and (5.38) into Eq. (5.36) yields

$$A_2 = \sum_{i=1}^{n} \dot{\varepsilon}_i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} \varepsilon_i c_{i+1}^i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})\alpha_\varepsilon \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} \kappa_\varepsilon \Lambda_\varepsilon \varepsilon_i \int (2\varepsilon_i - \varepsilon_{i-1} - \varepsilon_{i+1})dt$$

$$= \sum_{i=1}^{n} \dot{\varepsilon}_i \Lambda_\varepsilon \varepsilon_i + \sum_{i=1}^{n} c_{i+1}^i \Lambda_\varepsilon \varepsilon_i^2 + \sum_{i=1}^{n} \alpha_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i+1})^2 + \sum_{i=1}^{n} \kappa_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i-1}) \int (\varepsilon_i - \varepsilon_{i-1})dt$$

$$(5.39)$$

Substituting Eq. (5.39) into Eq. (5.33) and then substituting Eq. (5.33)into Eq. (5.27) gives

$$\dot{V} \le -\frac{1}{\mu_2} Ik_1 |s| - \frac{1}{\mu_2} Ik_2 \|s\|^2 - \sum_{i=1}^{n} c_{i+1}^i \Lambda_\varepsilon \varepsilon_i^2 - \sum_{i=1}^{n} \alpha_\varepsilon \Lambda_\varepsilon (\varepsilon_i - \varepsilon_{i+1})^2 \le 0 \qquad (5.40)$$

Assume that $s_i, \varepsilon_i, \dot{s}_i$ and $\dot{\varepsilon}_i$ are bounded. Therefore, s_i and ε_i are uniformly continuous. Then, from Barbalat's lemma, it is concluded that $s_i \to 0$ and $\varepsilon_i \to 0$ as time $t \to \infty$.

Now, we prove $e_i = 0$ when $s_i \to 0$ and $\varepsilon_i \to 0$. From Eq.s (5.23) and (5.24), $s_i \to 0$ implies $E_i \to 0$. Combining all Eq.s in Eq. (5.22) with indices 1 to n, one obtains

$$e_1 + e_2 + ... + e_n = \sum_{i=1}^{n} E_i = 0 \qquad (5.41)$$

We also have $\varepsilon_i = 0$ when $t \to \infty$, which means $e_1 = e_2 = e_3 = ... = e_n$. Substituting this part into Eq. (5.41) we have

$$e_1 = e_2 = e_3 = ... = e_n = 0 \qquad (5.42)$$

Therefore, Theorem 2 is proven.

5.5 Simulation Results

In this section, a simulation of the proposed fault-tolerant control algorithm for the 3-DOF manipulator shown in Figure 5.1 is described.

Figure 5.1.3-DOF Robot Manipulator in MATLAB/Simulink.

For this trajectory tracking simulation, the desired trajectories at each joint are given as

$$\begin{cases} q_{1d} = 0.5\sin(t/2) \\ q_{2d} = 0.3\sin(t) \\ q_{3d} = 0.2\sin(t) \end{cases} \tag{5.43}$$

The friction at each joint is assumed to be

$$\begin{cases} F_{1f} = 0.2sgn(\dot{q}_1) + 0.3\dot{q}_1 \\ F_{2f} = 0.2sgn(\dot{q}_2) + 0.3\dot{q}_2 \\ F_{3f} = 0.2sgn(\dot{q}_3) + 0.3\dot{q}_3 \end{cases} \tag{5.44}$$

5.5.1 Simulation 1

In this part, the parameter selection of the extended state observer as seen in Eq. (5.10) is explained. The total torque function at each joint is assumed to be

$$\begin{cases} \tau_1^t = \tau_1 \\ \tau_2^t = (1-\rho_2(t))\tau_2 + f_2(t) \quad t > 5 \\ \tau_3^t = \tau_3 \end{cases} \tag{5.45}$$

where for loss of effectiveness, $\rho_2(t) = 0.4\sin(\pi t)$, and a fault function $f_2(t) = 60\sin(\pi t/2)$. For the extended state observer of Eq. (5.10), the four parameters such as $\alpha_1, \alpha_2, \alpha_3$ and ε should be suitably selected. The turning parameters in the set of $\alpha_1, \alpha_2, \alpha_3$ that should satisfy the Hurwitz condition are quite related to the qualities of the observer such as the accuracy and peaking values. From this consideration, the turning parameters were set as $\alpha_1 = 8, \alpha_2 = 28, \alpha_3 = 7$. For ε, Eq. (5.17) shows that the value of ε highly affects the behaviour of the observer convergence. Therefore, three different values of $\varepsilon = 0.1, 0.01$, and 0.001 were considered in this simulation. In Figure 5.2-b, it can be seen that the peaking value highly defends the selection of ε. The smaller value of ε causes faster convergence, but the high magnitude peaking value in the fault estimation. As a trade-off between the convergence and the peaking value, $\varepsilon = 0.01$ was selected.

a) Joint 1   b) Joint 2   c) Joint 3
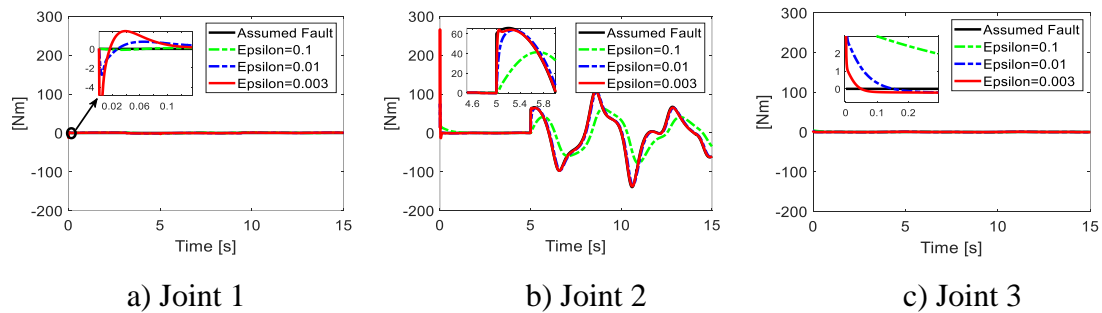
Figure 5.2. Fault estimation results with a single fault at Joint 2 with the different values of ε.

5.5.2 Simulation 2

For comparison, a passive fault-tolerant control with conventional sliding mode control (PFTC-SMC) and active fault-tolerant control with conventional sliding mode control (AFTC-SMC) were considered, in addition to the proposed active fault-tolerant control with synchronous sliding mode control (AFTC-SSMC).

The parameters of the observer were chosen as in Section 5.5.1. The AFTC-SMC is given as

$$\tau_{AFTC-SMC} = \tau_0 + \tau_{smc} + \tau_{ob} \tag{5.46}$$

where $\tau_0 = \mathbf{M}(q)(\ddot{q}_d + c\dot{e}) + \mathbf{H}(q,\dot{q})$, $\tau_{smc} = k_1 \operatorname{sgn}(s) + k_2 s$ and $\tau_{ob} = -\mathbf{M}(q)\hat{\phi}$, in which the sliding mode surface is selected as

$$s = \dot{e} + ce \tag{5.47}$$

The parameters in AFTC-SMC were suitably chosen as $c = diag(7;7;7), k_1 = diag(220;220;220)$, and $k_2 = diag(250;250;250)$. The PFTC-SMC is given as

$$\tau_{PFTC-SMC} = \mathbf{M}(q)(\ddot{q}_d + c\dot{e}) + \mathbf{H}(q,\dot{q}) + k_1 \operatorname{sgn}(s) + k_2 s \tag{5.48}$$

The parameters in PFTC-SMC were suitably chosen as $c = diag(7;7;7)$, $k_1 = diag(220;220;220)$ and $k_2 = diag(250;250;250)$. The parameters of AFTC-SSMC were suitably chosen as $c = diag(7;7;7)$, $\alpha = diag(20;20;20)$, $\kappa = diag(0.5;0.5;0.5)$, $k_1 = diag(220;220;220)$, $k_2 = diag(250;250;250)$ and $k_3 = diag(30,30,30)$.

To avoid chattering, the signum function in Eq.s (5.25), (5.46), and (5.48) are replaced with the saturation function,

$$sat(s) = \begin{cases} sgn(s) & if\ |s| \geq \lambda \\ \dfrac{s}{\lambda} & if\ |s| < \lambda \end{cases} \tag{5.49}$$

where $\lambda = 1.7$.

In this case, a failure occurs only at a single joint. The total torque function at each joint is assumed to be

$$\begin{cases} \tau_1^t = \tau_1 \\ \tau_2^t = (1-\rho_2(t))\tau_2 + f_2(t) & t > 5 \\ \tau_3^t = \tau_3 \end{cases} \tag{5.50}$$

where $\rho_2(t) = 0.4\sin(\pi t)$ and $f_2(t) = 60\sin(\pi t / 2)$.

In Figure 5.3-b, the single fault occurs at Joint 2 after five seconds. Figure 5.3 shows the fault estimation algorithm given in Eq. (5.10) seems to be working well even though its estimation error still exists on the order of 1/100 of the fault magnitude. In

Figure 5.4, without fault estimation, PFTC-SMC cannot guarantee the performance after the fault occurs. Unlike PFTC, the joint position errors of AFTC-SMC and proposed AFTC-SSMC look very different, but they were all on the order of 10^{-3}, so that the overall trajectory tracking performance of the two control algorithms can be said to be acceptable in this simulation. However, it can be seen that the proposed AFTC-SSMC has very high accuracy after a fault occurs. This fault-tolerant capability can be said to come from the synchronous SMC. Each joint error is coupled and constrained due to the synchronization errors and the newly defined synchronous sliding surface.

a) Joint 1	b) Joint 2	c) Joint 3
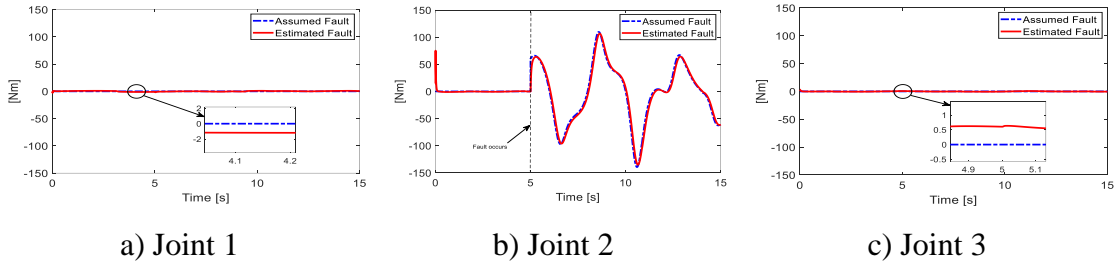
Figure 5.3.Fault estimation results with a single fault at Joint 2.

a) Joint 1	b) Joint 2	c) Joint 3
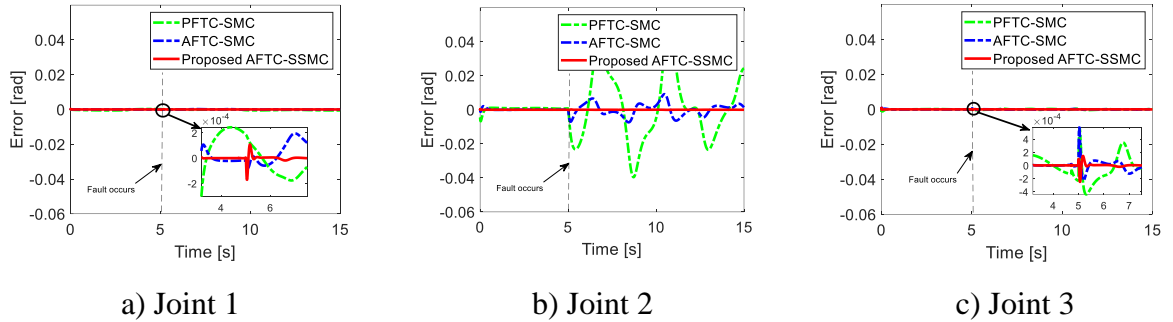
Figure 5.4.Tracking error at each joint in the simulation with a single fault at Joint 2.

5.5.3 Simulation 3

In this part, the effect of the actuator loss effectiveness factor to the proposed fault-tolerant control is discussed. The assumed fault is given as

$$\begin{cases} \tau_1^t = \tau_1 \\ \tau_2^t = (1-\rho_2(t))\tau_2 + f_2(t) & t > 5 \\ \tau_3^t = \tau_3 \end{cases} \tag{5.51}$$

where the fault function of $f_2(t) = 60\sin(\pi t/2)$ and $\rho_2(t)$ with the different values such as C1: $\rho_2(t) = 0.1$, C2: $\rho_2(t) = 0.5$, C3: $\rho_2(t) = 0.9$. The selected parameters of the observer are in Section 5.5.2

a) Joint 1   b) Joint 2   c) Joint 3
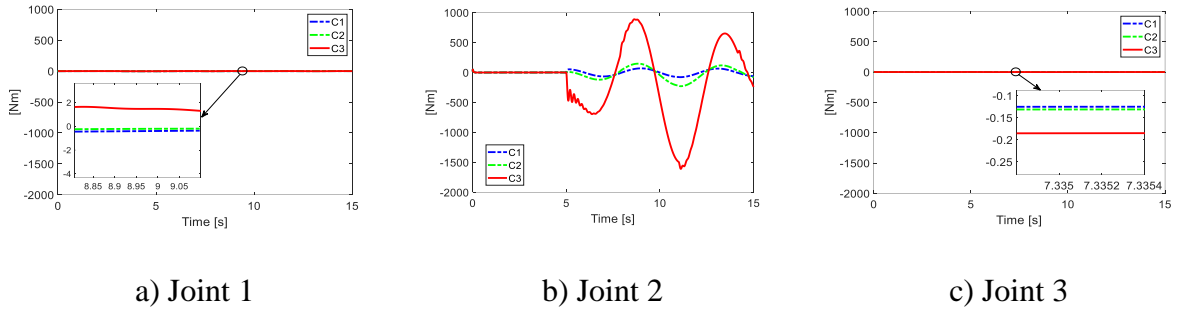
Figure 5.5.Estimated fault with the different values of $\rho_2(t)$.

a) Joint 1   b) Joint 2   c) Joint 3
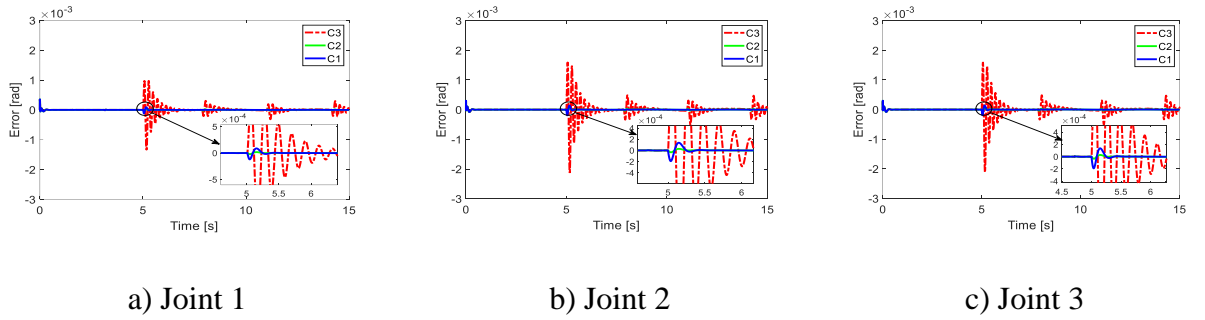
Figure 5.6.Tracking error at each joint with different values of $\rho_2(t)$.

Even though the total fault function has a large magnitude with $\rho_2(t)$ of 0.9 as seen in Figure 5.5-b, each joint error was still less than the value of 10^{-3} in Figure 5.6. Therefore, it can be said that the proposed fault-tolerant control can have the capability to show acceptable performances, even in the case of a high value of $\rho_2(t)$.

5.5.4 Simulation 4

To test the robustness of the fault-tolerant characteristics of the proposed algorithm, simulations with multiple faults were performed with the three control algorithms Eq.s (5.25), (5.46), and (5.48), for which the parameters were selected in Section 5.5.2. Multiple faults/failure functions were assumed to be

$$\begin{cases} \tau_1^t = \tau_1 + f_1(t) & t > 5 \\ \tau_2^t = (1 - \rho_2(t))\tau_2 + f_2(t) & t > 5 \\ \tau_3^t = (1 - \rho_3(t))\tau_3 & t > 5 \end{cases} \tag{5.52}$$

where $f_1(t) = 60$, $\rho_2(t) = 0.4\sin(\pi t)$, $f_2(t) = 60\sin(\pi t / 2)$, and $\rho_3(t) = 0.7\sin(\pi t)$.

Multiple faults simultaneously occurring at each joint seemed to burden the control system more than a single fault did. However, the proposed algorithm still resulted in smaller position-tracking errors than the conventional algorithm, demonstrating fault-tolerant characteristics, as seen in Figure 5.8.

66

a) Joint 1   b) Joint 2   c) Joint 3
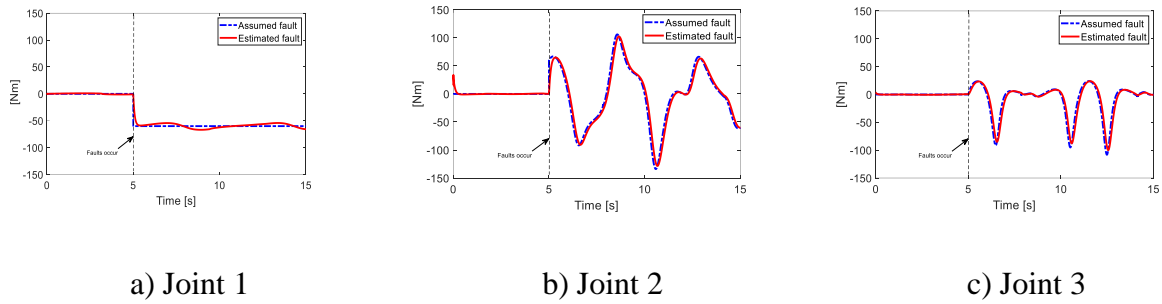
Figure 5.7.Fault estimation results in the simulation with multiple joint faults.

a) Joint 1   b) Joint 2   c) Joint 3
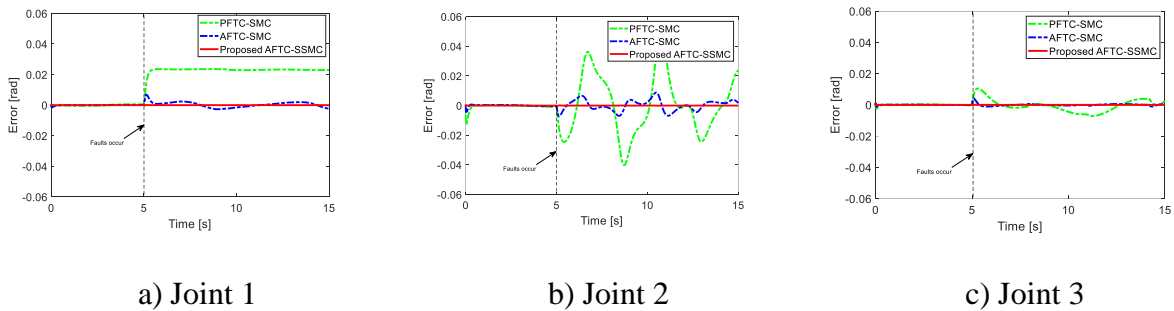
Figure 5.8.Tracking error at each joint in the simulation with multiple joint faults.

Remark 1: As mentioned in Section 1, the accuracy of fault estimation highly affects the performance of AFTCs, so Simulation 1 discussed the design of ESO. As a trade-off between the convergence and the peaking value of estimation results, the parameters of the observer were selected by using a trial–error technique, as shown in the Simulation 1 results. In Simulations 2,3, and 4, the outperformance of the proposed controller was presented by the tracking trajectory error results. In Simulation 3, the robustness of the proposed controller was described with a high magnitude of fault. In this case, PFTC-SMC and AFTC-SMC could not guarantee stability of the system. However, the proposed AFTC-SSMC could keep the system stable and showed acceptable performance results with the tracking trajectory error inside 10^{-3} rad. In Simulations 2 and 4, single and multiple faults were presented. Compared with the two fault-tolerant controllers without the consideration of the synchronization error, the proposed controller can reduce the picking phenomenon due to the simultaneous approach to zero at each joint of the synchronization technique. This characteristic of synchronization control can act to prevent the slow response of AFTCs.

5.6 Experimental Results

In this section, the real implementation of fault-tolerant control with AFTC-SMC and the proposed AFTC-SSMC is described, and the experimental results were compared and discussed in the case of both single and the multiple joint faults.
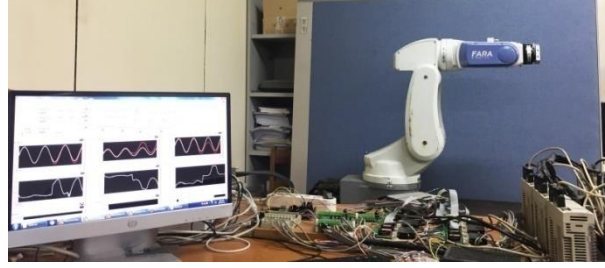
Figure 5.9.The experimental setup with FARA-AT2 Robot

5.6.1 Experimental Setup

The experimental setup is shown in Figure 5.9 with a 3-DOF FARA-AT2 robot manipulator. This robot manipulator has 6-DOF, but for these experiment joints 4, 5, and 6 were blocked. The 3-DOF FARA-AT2 robot had a CSMP series motor at each joint, and aCSMP-02BB driver was used for Joints 1 and 2 while the CSMP-01BB driver was used for Joint 3. The gear box at each joint is 120:1,120:1, 100:1 at Joints 1, 2, and 3, respectively. The encoder at each joint was a 2048 line count incremental encoder. The controller ran on Labview-FPGA NI-PXI-8110 and NI-PXI-7842R PXI cards with the frequency control set at 500Hz. NI-PXI-8110 was run on a Windows operating system.

The desired trajectory at each joint is given as

$$q_i(t) = \frac{\pi}{6}\sin(\frac{\pi t}{1600}) \quad (i = 1, 2, 3) \tag{5.53}$$

5.6.2 Experimental Design

To reduce the high-frequency chattering of fault estimation of $\hat{\phi}$ in Eq. (5.10) before it is used in the AFTC scheme, a simple low-pass filter was designed as

$$\hat{\phi}_k^{filted} = (1 - \upsilon)\hat{\phi}_{k-1}^{filted} + \upsilon\hat{\phi}_k \tag{5.54}$$

where $\hat{\phi}_k^{filted}$ and $\hat{\phi}_k$ are the output and input, respectively, of the low-pass filter at the kth step. The low-pass filter as seen in Figure 5.10 and Figure 5.13 allows the signal from the fault estimation (FE) to become smoother and is suitably applied to the AFTC schemes. However, the FE also increases the time delay of the feedback to the controller. To ensure that the smoothness and time delay are acceptable, $\upsilon = 0.05$ was selected.

The related parameters of the FE in Eq. (5.10)can be suitably selected as $\alpha_1 = 2, \alpha_2 = 3, \alpha_3 = 0.3$, and $\varepsilon = 0.01$.

The parameters of AFTC-SMC were suitably selected as $c = diag(3;3;3), k_1 = diag(50;110;90)$, and $k_2 = diag(50;100;90)$.

The parameters of the proposed AFTC-SSMC were suitably chosen as $c = diag(3;3;3)$, $\alpha = diag(3;3;3)$, $\kappa = diag(1.5;1.5;1.5)$, $k_1 = diag(50;110;90)$, $k_2 = diag(50;100;90)$, $k_3 = diag(30,30,30)$, and the *sign* function in the control law in Eq. (5.46) and the AFTC-SSMC was replaced by the saturation function

$$sat = \begin{cases} sgn(s) & if\ \ s > \lambda \\ s/\lambda & if\ \ s \le \lambda \end{cases},$$

where $\lambda = 1.6$.

5.6.3 Experimental Results

a) Single Fault

In this experiment, the fault occurred only at a single joint after 10 s. The desired fault function includes bias, and the fault is assumed to be

$$\begin{cases} \tau_1^t = \tau_1 \\ \tau_2^t = (1 - \rho_2(t))\tau_2 + f_2(t) & t > 5000 \\ \tau_3^t = \tau_3 \end{cases} \tag{5.55}$$

where $\rho_2(t) = 0.4\sin(\pi t / 2400)$ and $f_2(t) = -60\sin(\pi t / 2400)$.

The results are shown in Figure 5.10, Figure 5.11,and Figure 5.12. First, in Figure 5.10, the fault estimation is presented where the dashed aqua and pink lines are the upper and lower thresholds, respectively. The estimation results of the uncertainties and disturbance were within the thresholds for Joints 1 and 3, and for Joint 2 before the fault occurred. In contrast to the simulation results, the fault estimation results were not close to 0 before the fault occurred because in practice, relatively high value for the uncertainties and disturbances always exist. Therefore, using such thresholds is a good method for monitoring when faults occur in real applications. As shown in Figure 5.10, the extended state observer estimated the fault well. In Figure 5.11, the tracking error performance at each joint is shown. Before the fault occurred, the tracking trajectory of the proposed AFTC had no advantage over the AFTC with conventional sliding mode control. However, after the fault occurred, the advantages of the proposed AFTC were observed; the error at Joint 2 with AFTC-SSMC was smaller than that of AFTC-SMC, and the joint error was approximately 0.005rad when the fault occurred. This means that

the proposed controller has the ability to prevent tracking errors from getting larger due to a fault, and thus averting a critical condition (Figure 5.11-b); the Joint 2 error of the AFTC-SSMC was reduced when compared to that of the AFTC-SMC after faults. However, the Joints 1 and 3 errors of AFTC-SSMC were slightly increased compared with those of AFTC-SMC after faults (Figure 5.11-a, and c). This is probably due to the synchronization techniques, which tend to drive all joint errors to be comparable. The proposed synchronous control seems well-suited for fault-tolerant control. In addition, the synchronization joint errors theoretically approached zero, and as shown in Figure 5.12, they remained less than ~0.0025rad in this experiment.

a) Joint 1   b) Joint 2   c) Joint 3
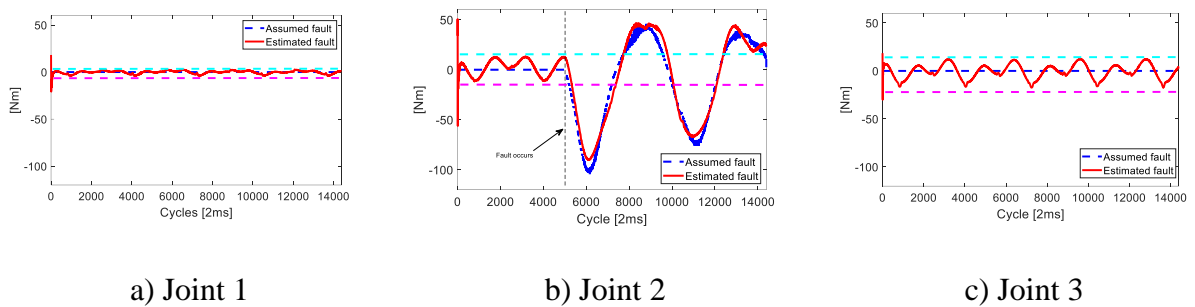
Figure 5.10.Fault estimation results with a single joint fault occurring at Joint 2. The aqua and pink dashed lines are the upper and lower thresholds, respectively.

a) Joint 1   b) Joint 2   c) Joint 3
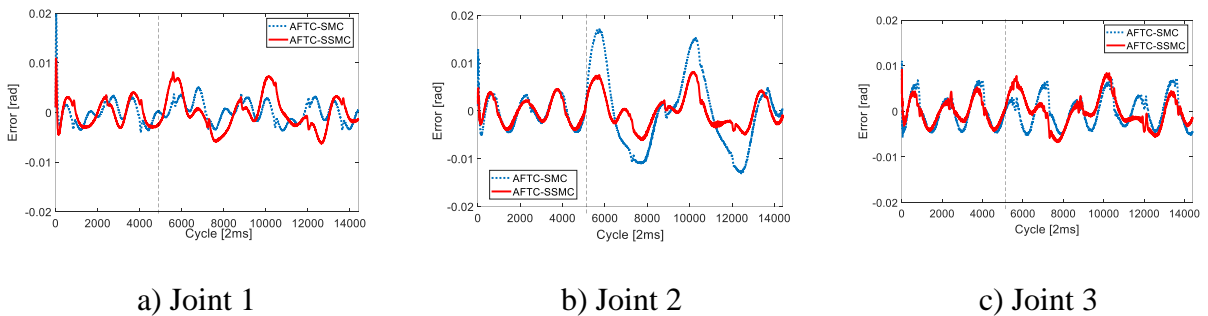
Figure 5.11.Tracking error at each joint with a single joint fault occurring at Joint 2.

| a) Joint 1 | b) Joint 2 | c) Joint 3 |
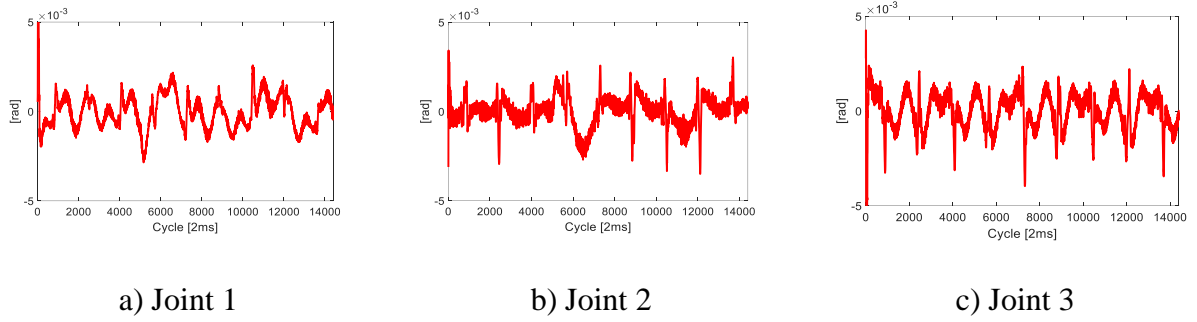
Figure 5.12.The synchronization error with a single joint fault occurring at Joint 2.

b) Multiple Faults

The effectiveness of the proposed fault-tolerant control in the case of multiple faults was also investigated. The desired faults occurred after 10 second are assumed to be

$$\begin{cases} \tau_1^t = \tau_1 + f_1(t) & t > 5000 \\ \tau_2^t = (1 - \rho_2(t))\tau_2 + f_2(t) & t > 5000 \\ \tau_3^t = (1 - \rho_3(t))\tau_3 & t > 5000 \end{cases} \qquad (5.56)$$

where $f_1(t) = 30$, $\rho_2(t) = 0.4\sin(\pi t / 2400)$, $f_2(t) = -60\sin(\pi t / 2400)$, and $\rho_3(t) = -0.7$.

The overall results of this experiment are similar to the previous results from the simulations and the single fault experiments. The fault estimation capability is still at the same level as that of the single fault case, as seen in Figure 5.13. The trajectory tracking performances of the two algorithms can be said to have behaviours similar to those seen in Figure 5.14. AFTC-SSMC showed the smaller trajectory tracking errors, especially after the faults occurred at 10 s, demonstrating the fault-tolerant characteristics, even in this multiple fault case. In Figure 5.14-a, and b, it can be seen that the picking phenomenon due to the slow response of the AFTC strategy was reduced by using the synchronization technique. And the synchronization joint errors were shown in Figure 5.15.

a) Joint 1   b) Joint 2   c) Joint3
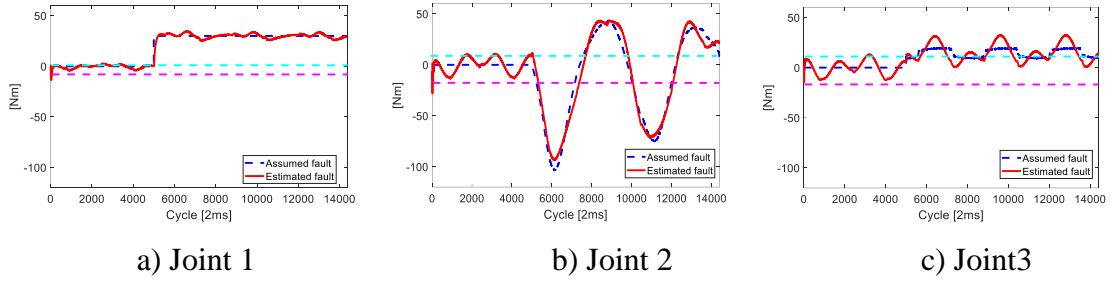
Figure 5.13.Fault estimation results with multiple faults. The aqua and pink dashed lines are upper and lower thresholds, respectively.

a) Joint 1   b) Joint 2   c) Joint 3
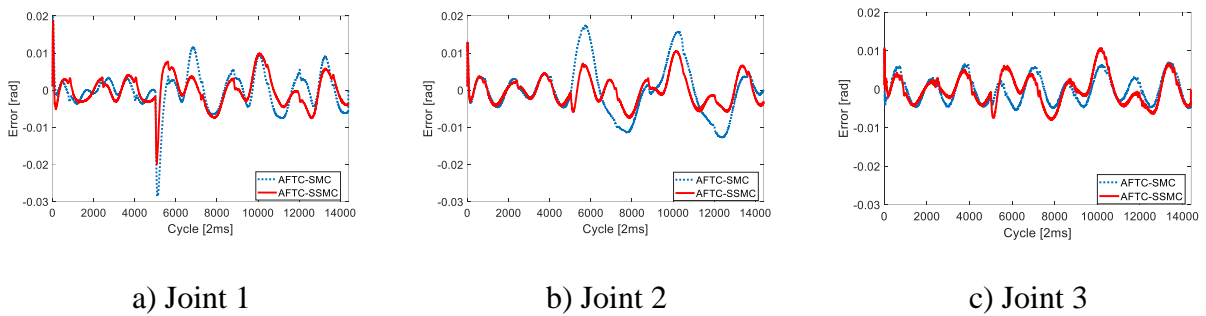
Figure 5.14.The error at joints with multiple faults.

a) Joint 1   b) Joint 2   c) Joint 3
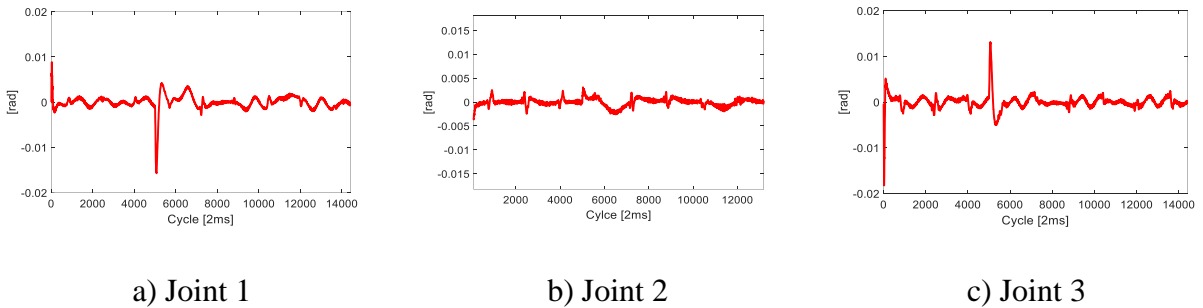
Figure 5.15.The synchronization error with multiple faults.

Remark 2: In this section, the experimental results were discussed. The resulting tracking trajectory errors were shown with single and multiple faults. In a real system, the uncertainties and disturbances have a higher magnitude than that of the simulation environment. Due to the limitation of selection parameters in a real system, the robot system with PFTC-SMC failed when multiple faults occurred. Therefore, in these experimental results, only the results of AFTC-SMC and the proposed AFTC-SSMC were presented. As mentioned in Section 1, the synchronization technique does not show effectiveness in an open-chain system such as a serial robot manipulator. Therefore, before a fault occurs, both controllers have a similar accuracy level. After a fault occurs,

72

due to the slow response of AFTCs, the controller without the consideration of the synchronization error showed a high picking value at the joint fault. The proposed controller with a constraint within the synchronization technique quickly responded to the occurrence of faults before the system had the feed-back information from fault estimation. Hence, AFTC-SSMC showed better performance than AFTC-SMC. However, in a real system, due to the limitation of adjustment controller parameters, the error at joints without faults were slightly increased when compared with those of AFTC-SMC by the synchronization technique. This characteristic may be a disadvantage of this proposed controller. However, the AFTC-SSMC still showed an acceptable performance at these actuators.

5.7 Conclusions

In this chapter, an active fault-tolerant control for a robot manipulator based on synchronous sliding mode was proposed. To verify its effectiveness, experimental implementation of the proposed control algorithm for a three degree-of-freedom FARA-AT2 robot were carried out and compared with the active fault-tolerant control with conventional sliding mode control in the both single and multiple fault cases. The results indicate that the active fault-tolerant control with synchronous sliding mode control has better fault-tolerant capability and results in better trajectory tracking performance when compared to the active fault-tolerant control with conventional sliding mode control algorithms. This fault-tolerant capability comes from synchronous sliding mode control, because each joint error is coupled and constrained due to the synchronization errors and newly defined synchronous sliding surface. Future work includes the optimal tuning of synchronization parameters by following some methods (e.g., the genetic algorithm, neural network technique, etc.) In addition, the synchronization technique will be applied to finite-time control such as terminal sliding mode control, non-singular terminal sliding mode control, and so on, in fault-tolerant control for a serial robot manipulator.

Chapter 6: Finite-Time Fault-tolerant Control Based on Synchronous Terminal Sliding Mode Control

In this chapter, two finite-time active fault-tolerant controllers for a robot manipulator, which combine a synchronous terminal sliding mode control with an extended state observer, are proposed. First, an extended state observer is adopted to estimate the lumped uncertainties, disturbances, and faults. The estimation information is used to compensate the controller designed in the following step. We present an active fault-tolerant control with finite-time synchronous terminal sliding mode control, largely based on a novel finite-time synchronization error and coupling position error. We also present an active fault-tolerant control that does not use a coupling position error. By using synchronization control, the position error at each joint can simultaneously approach toward zero and toward equality, which may reduce the picking phenomenon associated with the active fault-tolerant controller strategy. Finally, simulation and experimental results for a three degree-of-freedom robot manipulator verify the effectiveness of the two proposed active fault-tolerant controllers.

6.1 Introduction

In robot manipulator systems, the occurrence of faults or failures in the actuators or sensors may lead to degraded robot system performance, system breakdown, and economic loss. In response to the requirements of enhanced reliability and safety, fault-tolerant control (FTC) has attracted the attention of researchers over the past few decades. FTC strategies can be divided into two main types [68]: passive FTC (PFTC) [2,3], and active FTC (AFTC) [1, 19, 69].

In PFTC, the ability to tolerate abnormal operation in the presence of faults/failures in components mainly depends on the robustness of the controller, which can use sliding mode control (SMC) [7,8] or adaptive control [9, 58]. For example, in SMC, faults are considered to be external disturbances. To guarantee the stability of a system, knowledge of the bounded values of the faults is required in the SMC design. However, when the bounded value of a fault has a high magnitude, oscillation can occur

during normal operation of a system due to the problem of high gain control. Therefore, the ability of PFTC to deal with high magnitude faults is limited. Unlike PFTC, AFTC uses the fault information from the fault estimation process to compensate the conventional controller. Fault estimation (FE) is a powerful technique that includes fault detection, fault isolation, and fault identification within a step. The accuracy of the FE highly affects the performance of AFTC, so many estimation techniques have been developed to improve FE accuracy, such as sliding mode fault estimation [11,12], the extended state observer technique [6,13], and the learning observer technique [69]. The combination of FE and a conventional controller not only helps AFTC handle faults with a high magnitude, but also overcomes the high gain control issue associated with PFTC. Most work on developing AFTC focuses on improving the FE accuracy [14,15] or increasing performance of the controller [5,16]. Those AFTC approaches have led to acceptable performance, but the slow response issue still exists, causing the picking phenomenon after a fault occurs. This strongly affects the performance of AFTC. Therefore, in this chapter, using the concept of synchronization control, reduction of the picking phenomenon is addressed to increase the performance of AFTC.

Synchronization control is known as an effective controller for close-loop chain mechanism systems, such as parallel robot manipulators[73], cable-driver parallel manipulators[18,19], dual-drive gantry mechanisms [21], and cooperation robot manipulators[64]. Due to the constraints of the close-loop chain and the existence of position errors at the actuators during motion of the system, tensor internal forces may occur. This type of force is considered to be an internal uncertainty that can degrade the performance of a conventional controller. Using a synchronization controller, the position errors can simultaneously be equal and tend to zero, reducing the effect of the internal force. Therefore, the synchronization control technique can improve performance of a close-loop chain system. In an open-chain system, such as a serial robot manipulator, this kind of internal force may not exist, so the synchronization technique is not effective. However, the position errors at each joint still simultaneously approach zero when using synchronization control. In this chapter, we use the synchronization technique in a fault-tolerant controller to reduce the effect of the picking phenomenon associated with the AFTC strategy (AFTCs). When a fault occurs at an actuator, a controller using the synchronization technique can make the position error at each joint equal, so the controller can quickly respond to a fault before the controller has the feedback

information from the fault estimation process. Therefore, the picking phenomenon can be reduced.

In this chapter, two finite-time active fault-tolerant controls for robot manipulators are proposed. Both use a synchronization technique based on the synchronous terminal sliding mode control (S-TSMC) and the extended state observer. First, to estimate lumped uncertainties, disturbances, and faults, an extended state observer (ESO) [60] is adopted. An ESO is a simple technique for estimating faults in which simply adjusting the observer parameters leads to simple application in real systems. Next, an AFTC using the synchronization technique based on terminal sliding mode control (TSMC)[75] with a novel synchronization error and coupling position error (AFTC IS-TSMC) is proposed. The novel synchronization error can approach zero in a finite time compared to the conventional coupling position error in [76]. In addition, the novel synchronization error has an advantage over the conventional error [77], because it is more closely related to other joints. Compared with existing finite-time synchronization controls, such as [78], the novel synchronization error does not lead to a singularity when the desired trajectory crosses zero. This improves the range of the robot manipulator over that in[78]. Some authors (such as in [27,28]) use graphic theory and prescribe performance control in the synchronization control. However, the conventional prescribed performance term can become a singularity during operation, and graphic theory may not be suitable for a single robot manipulator controller. The second proposed AFTC based on the synchronization technique (AFTC S-TSMC), but without the coupling position error, is designed to improve the synchronization of the synchronous terminal sliding mode control. With the combination of ESO and the synchronization technique, the two proposed AFTC S-TSMCs can avoid the drawbacks of the PFTC strategy, and their fast response leads to a robot system that can deal with high-magnitude faults while reducing the picking phenomenon. Finally, both simulated and experimental results from the two proposed AFTCs verify the effectiveness of the two novel synchronous terminal sliding mode controllers. The contributions of this chapter are summarized as follows:

(1) Two active fault-tolerant control algorithms for robot manipulators, based on novel finite-time synchronous terminal sliding mode controllers and an extended state observer, are proposed. The novel finite-time synchronization technique has the ability to make both the joint position error and the synchronization error simultaneously approach to zero. Due to these internal constraints of the

synchronization control, the proposed controller can make the system quickly respond to the faults in a forward way before its feedback response after a fault estimation. Therefore, the proposed controller can reduce the occurrence of the picking phenomenon due to the slow feedback response of AFTC strategy.

(2) The novel synchronization error leads to better synchronization because it uses more information from other joints than the conventional synchronization error in [77],which has the information from only one neighbor joint.

(3) The novel coupling error can make the position error approach zero in a finite-time while the conventional coupling error in [76] cannot guarantee the finite-time convergence.

(4) Two proposed AFTCs can avoid the singularity in both the desired trajectory and the control action, while the control algorithms in [26–28] cannot guarantee avoiding the singularity. This ability allows increase in the workspace of the robot.

(5) Experimental results show the effectiveness of the proposed AFTC in reducing both the picking phenomenon and in handling faults of high magnitude.

The rest of this chapter is organized as follows. In Section 2, the dynamic model of a robot manipulator and associated faults are presented. Fault estimation based on the extended state observer is discussed in Section 3. In Section 4, a novel synchronization error, coupling position error, and the finite-time active fault-tolerant control using a synchronous fast terminal sliding mode control are proposed. Simulation results and discussion about the effect of synchronization parameters are given in Section 5. In Section 6, the experimental results are shown to verify the effectiveness of the proposed AFTC. Finally, conclusions are given in Section 7.

6.2 Dynamic Model of a Robot Manipulator and Faults

The dynamics of an n-degree-of-freedom (DOF) robot manipulator are defined as:

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+G(q)+F_f(\dot{q})=\tau \tag{6.1}$$

where $\ddot{q},\dot{q},q \in \Re^n$ are the vectors for joint acceleration, velocity, and position, respectively. $M(q) \in \Re^{n \times n}$, $C(q,\dot{q}) \in \Re^{n \times n}$, and $G(q) \in \Re^n$ represent the inertia matrix, the centripetal and Coriolis matrix, and the vector of gravitation force, respectively. $F_f(\dot{q}) \in \Re^n$ is the vector of friction term which includes a viscous friction and a dynamic friction, and $\tau \in \Re^n$ is the vector of torque at the joints.

In practice, the dynamic model of a robot is not known exactly, so Eq. (6.1) can be written as:

$$(\mathbf{M}(q) + \Delta M(q))\ddot{q} + (\mathbf{C}(q,\dot{q}) + \Delta C(q,\dot{q}))\dot{q} + (\mathbf{G}(q) + \Delta G(q)) + (\mathbf{F}_f(\dot{q}) + \Delta F_f(\dot{q})) + \delta = \tau \qquad (6.2)$$

where $\Delta M, \Delta C, \Delta G$, and ΔF are unknown dynamic uncertainties, and δ is an unknown external disturbance. $\mathbf{M}(q), \mathbf{C}(q,\dot{q}), \mathbf{G}(q)$ and $\mathbf{F}(\dot{q})$ are estimates of $M(q), C(q,\dot{q}), G(q)$ and $F(\dot{q})$.

Thus, Eq. (6.2) can be rewritten as:

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q,\dot{q})\dot{q} + \mathbf{G}(q) + \mathbf{F}_f(\dot{q}) + \psi = \tau \qquad (6.3)$$

where $\psi = \Delta M\ddot{q} + \Delta C\dot{q} + \Delta G + \Delta F + \delta$.

In general, actuator faults can be divided into two types: bias faults and gain faults. In a robot manipulator, these are known as loss-of-effectiveness and lock-in-place faults. In practice, both kinds of actuator faults commonly occur. Therefore, the total torque including both kinds of actuator faults can be comprehensively described as:

$$\tau^t = (I - \rho(t))\tau + \mathbf{f}(t) \quad (t > t_f) \qquad (6.4)$$

where $\mathbf{f}(t) = diag(f_i) \in \Re^n$ denotes a bounded signal. $\rho(t) = diag(\rho_i(t)) \in \Re^{n \times n}, 0 \le \rho_i(t) < 1$ $(i = 1, 2, ..., n)$, which is unknown, denotes the remaining control rate. $I \in \Re^{n \times n}$ is the identity matrix, and t_f is the time of occurrence of each fault.

Substituting Eq. (6.4) into Eq. (6.3), the dynamics model of an n-degree-of-freedom robot manipulator with actuator faults can be written as:

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q,\dot{q})\dot{q} + \mathbf{G}(q) + \mathbf{F}_f(\dot{q}) + \psi = (I - \rho(t))\tau + \mathbf{f}(t) \qquad (6.5)$$

6.3 Fault Estimation Using an Extended State Observer

In this section, an extended state observer of uncertainties/disturbances and faults/failures is presented.

The dynamic model of the robot manipulator of Eq. (6.5) can be rewritten in state space as:

$$\ddot{q} = \mathbf{M}^{-1}(q)(\tau - \mathbf{H}(q,\dot{q})) - \mathbf{M}^{-1}(q)\zeta \qquad (6.6)$$

where $\mathbf{H}(q,\dot{q}) = \mathbf{C}(q,\dot{q})q + \mathbf{G}(q) + \mathbf{F}_f(\dot{q})$. $\zeta(q,\dot{q},\ddot{q},\tau,t) = \rho(t)\tau + \psi - \mathbf{f}(t)$ represents uncertainties or disturbances and faults/failures.

In the state space, the dynamic model of Eq. (6.6) becomes:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x_1, x_2, \tau) + \phi(x_2, x_2, \tau, t) \end{cases} \qquad (6.7)$$

where $x_1 = q \in \Re^n$, $x_2 = \dot{q} \in \Re^n$, $f(x_1, x_2, \tau) = \mathbf{M}^{-1}(q)(\tau - \mathbf{H}(q,\dot{q}))$, and $\phi(x_1, x_2, \tau, t) = -\mathbf{M}^{-1}(q)\zeta$.

An extended state observer [60] is given as:

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + \dfrac{\alpha_1}{\varepsilon}(x_1 - \hat{x}_1) \\[2mm] \dot{\hat{x}}_2 = \hat{f}(x_1, \hat{x}_2, \tau) + \dfrac{\alpha_2}{\varepsilon^2}(x_1 - \hat{x}_1) + \hat{\phi} \\[2mm] \dot{\hat{\phi}} = \dfrac{\alpha_3}{\varepsilon^3}(x_1 - \hat{x}_1) \end{cases} \tag{6.8}$$

where $\hat{x}_1, \hat{x}_2, \hat{f}$, and $\hat{\phi}$ are estimates of x_1, x_2, f, and ϕ, respectively, α_1, α_2, and α_3 are positive constants, polynomial $s^3 + \alpha_1 s^2 + \alpha_2 s + \alpha_3$ is Hurwitz, and $0 < \varepsilon < 1$.

The stability of system Eq. (6.7) with observer Eq. (6.8) is shown in [6] with the conditions $0 < \varepsilon < 1$ and $|\dot{\phi}| \le L$. The observer error convergence is given as:

$$\|\tilde{e}\| \le \frac{2\varepsilon L \|PB\|}{\lambda_{\min}(Q)} \tag{6.9}$$

where $A = \begin{bmatrix} -\alpha_1 & 1 & 0 \\ -\alpha_2 & 0 & 1 \\ -\alpha_3 & 0 & 0 \end{bmatrix}$ and $B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, and there is a symmetric positive definite matrix Q

satisfying the Lyapunov Eq.:

$$A^T P + PA = -Q \tag{6.10}$$

6.4 Finite-Time Fault-Tolerant Control Using Synchronous Terminal Sliding Mode Control

In this section, two finite-time fault-tolerant controls based on a synchronous terminal sliding mode control are proposed.

Some definition will be useful in the rest of the chapter.

Definition 1. *We define* $\lceil x \rfloor^\Lambda = \left[|x_1|^{\lambda_1} sign(x_1), |x_2|^{\lambda_2} sign(x_2),..., |x_n|^{\lambda_n} sign(x_n) \right]^T \in \Re^n$, *where*

$\lambda_i (i = 1, 2, ..., n) > 0$ *and* $\Lambda = diag(\lambda_i)$. $x = [x_1, x_2, ..., x_n]^T \in \Re^n$ *and* $y = [y_1, y_2, ..., y_n]^T \in \Re^n$.

Definition 2. *We define* $x \cdot y = \left[x_1 y_1, x_2 y_2, ..., x_n y_n \right]^T \in \Re^n$.

Definition 3. *The time derivative of* $\lceil x \rfloor^\Lambda$ *is*

$$\frac{d}{dt} \lceil x \rfloor^\Lambda = \Lambda |x|^{\Lambda - 1} \cdot \dot{x} = \left[\lambda |x_1|^{\lambda - 1} \dot{x}_1, \lambda |x_2|^{\lambda - 1} \dot{x}_2, ..., \lambda |x_n|^{\lambda - 1} \dot{x}_n \right]^T,$$

and $x^{\Lambda - I} = \left[x_1^{\lambda_1 - 1}, x_2^{\lambda_2 - 1}, ..., x_n^{\lambda_n - 1} \right] \in \Re^n$, *where* $I = diag(1) \in \Re^{n \times n}$.

The novel synchronization error is defined as:

$$\varepsilon_1 = (1+\psi_1\psi_n)e_1 - \psi_1 e_2 - \psi_1 e_n$$
$$\varepsilon_2 = (1+\psi_2\psi_1)e_2 - \psi_2 e_3 - \psi_2 e_1$$
$$\vdots$$
$$\varepsilon_n = (1+\psi_n\psi_{n-1})e_n - \psi_n e_2 - \psi_n e_1$$

(6.11)

where $e_i (i=1,2,...,n)$ is the error at each joint, and $\psi_i (i=1,2,...,n)$ is the corresponding positive gain. In matrix form,

$$\varepsilon = Te$$

(6.12)

where $\varepsilon = [\varepsilon_1, \varepsilon_2,...,\varepsilon_n]^T \in \Re^n$, $e = [e_1, e_2,...,e_n]^T \in \Re^n$, $T \in \Re^{n\times n}$, and

$$T = \begin{bmatrix} (1+\psi_1\psi_n) & -\psi_1 & 0 & \cdots & -\psi_1 \\ -\psi_2 & (1+\psi_2\psi_1) & -\psi_2 & \cdots & 0 \\ 0 & -\psi_3 & (1+\psi_3\psi_2) & \cdots & 0 \\ \vdots & & & & \\ -\psi_n & -\psi_n & 0 & \cdots & (1+\psi_n\psi_{n-1}) \end{bmatrix}$$

(6.13)

6.4.1 The Proposed Active Fault-Tolerant Control with Integral Synchronous Terminal Sliding Mode Control (AFTC IS-TSMC)

The novel finite-time coupling position error is defined as:

$$E = \alpha e + \beta \int \lceil \varepsilon \rfloor^\Lambda dt$$

(6.14)

where $E = [E_1, E_2,...,E_n]^T \in \Re^n$, $\alpha = diag(\alpha_i) \in \Re^{n\times n}$, and $\beta = diag(\beta_i) \in \Re^{n\times n}$ are coupling parameters and positive $\Lambda = diag(\lambda_i)$ $0 < \lambda_i < 1$.

The synchronous terminal sliding surface is defined as:

$$S = \dot{E} + \Gamma \lceil E \rfloor^\Lambda$$

(6.15)

where $S = [S_1, S_2,...,S_n]^T \in \Re^n$, $\dot{E} = [\dot{E}_1, \dot{E}_2,...,\dot{E}_n]^T \in \Re^n$, $\Gamma = diag(\gamma_i) \in \Re^{n\times n}$, and $\gamma_i > 0$.

The proposed finite-time active fault-tolerant control is given as:

$$\tau = \tau_{eq} + \tau_0 + \tau_{ob}$$

(6.16)

where $\tau_{eq} = \mathbf{M}(q)\left(\ddot{q}_d + \alpha^{-1}\beta\Lambda|\varepsilon|^{\Lambda-1}\cdot\dot{\varepsilon} + \alpha^{-1}\Gamma\Lambda|E|^{\Lambda-1}\cdot\dot{E}\right) + \mathbf{H}(q,\dot{q})$, $\tau_0 = \mathbf{M}(q)K_1 sign(S)$,

$\tau_{ob} = -\mathbf{M}(q)\hat{\phi}$, where $K_1 = diag(k_{1i}) \in \Re^{n\times n}$.

Theorem 1. *The system described in Eq. (6.5), using the controller specified in Eq. (6.16) guarantees that $e \to 0$ as finite-time.*

Proof of Theorem 1. The Lyapunov function can be selected as:

$$V = \frac{1}{2}S^T S$$

(6.17)

The time derivative of V in Eq. (6.17) is:

$$\dot{V} = S^T \dot{S}$$

$$= S^T \left(\ddot{E} + \Gamma\Lambda |E|^{\Lambda-1} \cdot \dot{E} \right)$$

$$= S^T \left(\alpha(\ddot{q}_d - \ddot{q}) + \beta\Lambda |\varepsilon|^{\Lambda-1} \cdot \dot{\varepsilon} + \Gamma\Lambda |E|^{\Lambda-1} \cdot \dot{E} \right)$$

$$= S^T \left(\alpha \begin{pmatrix} \ddot{q}_d - \mathbf{M}^{-1}(q)(\tau - \mathbf{H}(q,\dot{q})) \\ + \mathbf{M}^{-1}(q)\zeta(q,\dot{q},\tau,t) \end{pmatrix} + \beta\lambda |\varepsilon|^{\Lambda-1} \cdot \dot{\varepsilon} \\ + \Gamma\Lambda |E|^{\Lambda-1} \cdot \dot{E} \right) \tag{6.18}$$

Substituting Eq. (6.16) into Eq. (6.18):

$$\dot{V} = -S^T \left(K_1 sign(S) \right)$$

$$\leq -\sigma_1 V^{\frac{1}{2}} < 0 \tag{6.19}$$

where $\sigma_1 = \lambda_{\min}(K_1)$. When $S = 0$ converges, then $E = 0$ and $\dot{E} = 0$, and we have:

$$\dot{e}_i = -\frac{\beta_i}{\alpha_i} \lceil \varepsilon_i \rfloor^\lambda$$

$$= -\frac{\beta_i}{\alpha_i} \left| (1+\psi_i\psi_{i-1})e_i - \psi_i e_{i-1} - \psi_i e_{i+1} \right|^\lambda sign \begin{pmatrix} (1+\psi_i\psi_{i-1})e_i \\ -\psi_i e_{i-1} - \psi_i e_{i+1} \end{pmatrix} \tag{6.20}$$

The system in Eq. (6.20) has equilibrium points at $e_i = 0$ $(i = 1, 2, ..., n; n+1 = 1)$.

According to the definition of terminal attractors [81], we have:

$$\left| \frac{\partial \dot{e}_i}{\partial e_j} \right| = \frac{\beta_i}{\alpha_i} \lambda \left| (1+\psi_i\psi_{i-1})e_i - \psi_i e_{i-1} - \psi_i e_{i+1} \right|^{\lambda-1} \left| \frac{\partial \varepsilon_i}{\partial e_j} \right| = \infty \tag{6.21}$$

where $j = (i-1, i, i+1)$ $(i = 1, 2, ..., n)$.

From Eq.(6.21), we have $e_i \to 0 (i = 1, 2, ..., n;)$ at a finite-time. Therefore, Theorem 1 is proven.
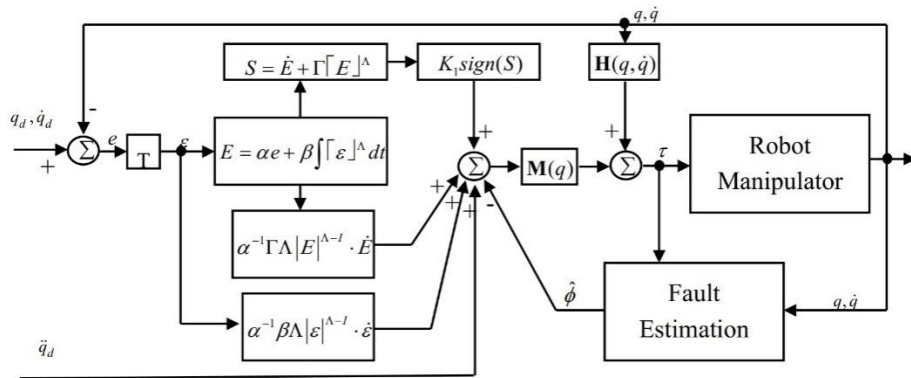
Figure 6.1. The block diagram of the proposed controller active fault-tolerant control with AFTC IS-TSMC.

In Figure 6.1, the block diagram of the proposed controller AFTC IS-TSMC is presented. The block T in the diagram is synchronization matrix in Eq. (6.13). The block of Fault Estimation uses the extended state observer in Eq. (6.8). The fault estimation results were used to compensate with a novel finite-time synchronous terminal sliding mode controller 1.

Remark 1. Time convergence from initial state to zero:

$$t = t_r + t_{s_i} + t_{e_i} \tag{6.22}$$

Time convergence $S \to 0$:

$$t_r \le \frac{2}{\sigma_1}\left(\ln\left(\sigma_1 V_0^{\frac{1}{2}} + \sigma_2 \right) - \ln \sigma_2 \right) \tag{6.23}$$

Time convergence $E \to 0$:

$$t_{s_i}(1) = \frac{1}{c_i(1-\lambda_i)}\left(\ln\left(\gamma_i + \sigma_2 E_i^{1-\lambda_i}(0) \right) - \ln \gamma_1 \right) \tag{6.24}$$

Time convergence $e_i \to 0$:

$$t_{e_i} = \frac{\alpha_i}{\beta_i(1-\lambda_i)}\left(ax_i(0) + bx_{i-1}(0) + cx_{i+1}(0) \right)^{1-\lambda_i} \tag{6.25}$$

where $a = 1 + \psi_i\psi_{i-1}, b = -\psi_i$ *and* $c = -\psi_i$.

6.4.2 The Proposed Active Fault-Tolerant Control with Synchronous Terminal Sliding Mode Control 2 (AFTC S-TSMC)

The novel synchronous terminal sliding surface:

$$\begin{aligned} S &= \dot{e} + \pi\lceil \varepsilon \rfloor^\Lambda \\ &= \dot{e} + \pi\lceil Te \rfloor^\Lambda \end{aligned} \tag{6.26}$$

where $\pi = diag(\pi_i) \in \Re^{n\times n}$ is the positive matrix gain.

The proposed active fault-tolerant control with synchronous terminal sliding mode (AFTC S-TSMC) is given as:

$$\tau = \tau_{eq} + \tau_0 + \tau_{ob} \tag{6.27}$$

where $\tau_{eq} = \mathbf{M}(q)(\ddot{q}_d + \pi\Lambda|\varepsilon|^{\Lambda-1}\cdot\dot{\varepsilon}) + \mathbf{H}(q,\dot{q})$, $\tau_o = \mathbf{M}(q)K_1 sign(S)$, $\tau_{ob} = -\mathbf{M}(q)\hat{\phi}$ and where

$K_1 = diag(k_{1i}) \in \Re^{n\times n}$.

Theorem 2. *The system described in Eq. (6.5), using the controller specified in Eq. (6.27) guarantees that* $e \to 0$ *as finite-time.*

Proof of Theorem 2. The Lyapunov function can be selected as:

$$V = \frac{1}{2}S^T S \tag{6.28}$$

The time derivative of V in Eq. (6.28) is:

$$\begin{aligned}
\dot{V} &= S^T \dot{S} \\
&= S^T \left(\ddot{e} + \Lambda\pi|\varepsilon|^{\Lambda-I} \cdot \dot{\varepsilon} \right) \\
&= S^T \left(\ddot{q}_d - \ddot{q} + \Lambda\pi|\varepsilon|^{\Lambda-I} \cdot \dot{\varepsilon} \right) \\
&= S^T \left(\begin{array}{c} \ddot{q}_d - \mathbf{M}^{-1}(q)\left(\tau - \mathbf{H}(q,\dot{q})\right) \\ +\mathbf{M}^{-1}(q)\boldsymbol{\zeta}(q,\dot{q},\tau,t) + \Lambda\pi|\varepsilon|^{\Lambda-I} \cdot \dot{\varepsilon} \end{array} \right)
\end{aligned} \tag{6.29}$$

Substituting Eq. (6.27) into Eq. (6.29) we have:

$$\dot{V} = -S^T K_1 sign(S) \leq -\sigma_1 V^{\frac{1}{2}} \leq 0 \tag{6.30}$$

where $\sigma_1 = \lambda_{\min}(K_1)$. When the sliding mode achieves, the system become as Eq. (6.20) and shown as above. Therefore, Theorem 2 is proven.

In Figure 6.2, the block diagram of the proposed controller AFTC S-TSMC was shown. The block T is the synchronization matrix in Eq. (6.13). The block of Fault Estimation uses the same extended state observer in Eq. (6.8) to estimate the lumped uncertainties, disturbances and faults. Compared with the first proposed controller, the second proposed controller has less computation, and the ability to converge in a finite-time.
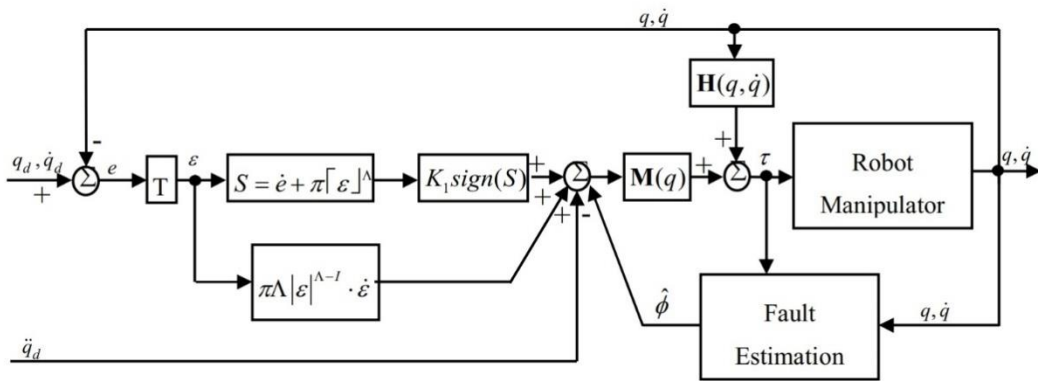
Figure 6.2. The block diagram of the proposed controller active fault-tolerant control with synchronous terminal sliding mode control 2 (AFTC S-TSMC).

Remark 2. A singularity may occur at $|\varepsilon|^{\Lambda-I}$ and $|E|^{\Lambda-I}$ in Eq. (6.16) and Eq. (6.27) as S approaches zero. By using saturation function $sat(x^{\Lambda-I}, u_s), x = |\varepsilon|$ and $|E|$ where $u_s > 0$, the singularity can be avoided and the system retains finite-time stability [75].

Remark 3. The time convergence can be shown as:

$$t = t_r + t_{e_i} \tag{6.31}$$

where t_r is as shown in Eq. (6.23), and t_{e_i} is as shown in Eq. (6.25).

6.5 Simulation Results

In this section, the simulation results for a conventional terminal sliding mode control combined with an extended state observer([75]+ESO) and the two proposed AFTCs on a 3-DOF robot manipulator are shown and discussed. The mechanical model of the 3-DOF robot manipulator was built on the SolidWorks (Dassault Systems, Waltham, Massachusetts, USA) with the geometry parameters from the catalogue of the SAMSUNG FARA-AT2 (Samsung, Namdong-gu, Incheon, Korea). Then, a robot manipulator model was exported to Matlab (MathWorks, Natick, Massachusetts, USA) simulation environment by Simmechanics toolbox. It can be seen in Figure 6.3, and the parameters of the robot manipulator are shown in Table 6.1.

Figure 6.3**.** A 3-degree-of-freedom (DOF) robot manipulator in Matlab/Simulink.

Table 6.1**.** Parameters of 3-DOF robot manipulator in Matlab simulation.

Links	Length (m)	Weight (kg)	Center of Mass (m)	Inertia (kg.m²)
Link 1	0.15	56.5	$[-98.3 \times 10^{-3} - 2.9 \times 10^{-8} - 85.4 \times 10^{-3}]$	[Ixx=0.39 Iyy=0.59 Izz=0.56]
Link 2	0.255	35.6	$[-5.5 \times 10^{-3} 0.001 \times 10^{-3} - 156.9 \times 10^{-3}]$	[Ixx=0.76 Iyy=0.44 Izz=0.39
Link 3	0.41	58.9	$[54.6 \times 10^{-3} - 0.01 \times 10^{-3} 80.5 \times 10^{-3}]$	[Ixx=0.22 Iyy=1.2 Izz=1.2]

For this trajectory tracking simulation, the desired trajectories at each joint are given as:

$$\begin{cases} q_{1d} = 0.5\cos(t/2) - 0.5 \\ q_{2d} = 0.3\cos(t) - 0.3 \\ q_{3d} = 0.2\cos(t) - 0.2 \end{cases} \tag{6.32}$$

where $\dot{q}_d$ and $\ddot{q}_d$ are the first order and second order derivatives of the desired position, respectively.

The friction at each joint was assumed to be:

$$\begin{cases} F_{1f} = 0.2sgn(\dot{q}_1) + 0.3\dot{q}_1 \\ F_{2f} = 0.2sgn(\dot{q}_2) + 0.3\dot{q}_2 \\ F_{3f} = 0.2sgn(\dot{q}_3) + 0.3\dot{q}_3 \end{cases} \tag{6.33}$$

The total torque function at each joint was assumed to be:

$$\begin{cases} \tau_1^t = \tau_1 \\ \tau_2^t = (1 - \rho_2(t))\tau_2 + f_2(t) \quad t > 5 \\ \tau_3^t = \tau_3 \end{cases} \tag{6.34}$$

where $\rho_2(t) = 0.4\sin(\pi t)$ and $f_2(t) = -80\sin(\pi t / 2)$.

The related parameters for the ESO were chosen to be $\alpha_1 = 8, \alpha_2 = 28, \alpha_3 = 7$, and $\varepsilon = 0.01$. The controller ([75]+ESO) is given as:

$$\tau_{[23]+ESO} = \tau_0 + \tau_{smc} + \tau_{ob} \tag{6.35}$$

where $\tau_{eq} = \mathbf{M}(q)(\ddot{q}_d + c\Lambda|e|^{\Lambda-1} \cdot \dot{e}) + \mathbf{H}(q,\dot{q})$, $\tau_o = \mathbf{M}(q)K_1 sign(S)$ and $\tau_{ob} = -\mathbf{M}(q)\hat{\phi}$, where $K_1 = diag(k_{1i}) \in \Re^{n \times n}$. The sliding mode surface was selected as:

$$s = \dot{e} + c\lceil e \rfloor^{\Lambda} \tag{6.36}$$

The parameters for the [75]+ESO were chosen as $c = diag(7;7;7), K_1 = diag(80;80;110)$, $u_s = 20$, and $\Lambda = diag(0.58;0.58;0.58)$ .The parameters for the AFTC IS-TSMC were chosen as $\psi_1 = \psi_2 = \psi_3 = 2$, $\alpha = diag(1;1;1)$, $\beta = diag(0.5;0.5;0.5)$, $\Lambda = diag(0.6;0.6;0.6)$, $u_s = 20$, $\Gamma = diag(7;7;7)$ and $K_1 = diag(80;80;110)$. The parameters for the AFTC S-TSMC were chosen as $\psi_1 = \psi_2 = \psi_3 = 2$, $\Lambda = diag(0.6;0.6;0.6)$, $c = diag(7;7;7)$, $u_s = 20$ and $K_1 = diag(80;80;110)$.

To avoid a singularity, the terms containing power $\Lambda - 1$ in Eq.s (6.16), (6.27)and (6.35) are replaced with the saturation function.

$$sat(u_f, u_s) = \begin{cases} u_s & if \ u_f \geq u_s \\ u_f & if \ u_f < u_s \end{cases} \tag{6.37}$$

where $u_s = 20$ is a positive constant, and $u_f = \Lambda|x|^{\Lambda-1} \cdot \dot{x}$ with $x = e, \varepsilon$ and E .

To avoid chattering, the signum function in Eq.s (6.16), (6.27) and (6.35) are replaced with the saturation function.

$$sat(s) = \begin{cases} sgn(s) & if \ |s| \geq \lambda \\ \dfrac{s}{\lambda} & if \ |s| < \lambda \end{cases} \tag{6.38}$$

where $\lambda = 1.7$.

Fault estimation using the extended state observer is presented in Figure 6.4. The estimation error with a high fault value at joint 2 is acceptable. The error trajectory

tracking results are shown in Figure 6.5. In general, all three controllers have an accuracy of within 10^{-3} rad, indicating that the AFTC strategy can tolerate faults, and shows acceptable performance. Now, the three controllers are discussed in more detail. It can be seen from Figure 6.5 that before five seconds, the errors associated with the three controllers are similar. As mentioned in Section 6.1, in the normal operation mode of the serial robot manipulator, there is no internal force during motion, so the synchronization technique has no effect in this case. After five seconds, it can be seen that the AFTC S-TSMC shows a smaller picking value than the other controller. In addition, the error characteristics seen with the AFTC S-TSMC are different from those seen with the other controller. This is because this controller has the ability to make the error at each joint simultaneously approach the zero of the synchronization control. However, in Figure 6.6**. Tracking error at each joint for AFTC IS-TSMC and AFTC S-TSMC.**, the synchronizations of AFTC IS-TSMC and AFTC S-TSMC are significantly different, causing the picking value of AFTC S-TSMC to be smaller than the other two controllers. In controller (6.16), AFTC IS-TSMC uses the coupling position error for sliding mode control. It can be seen that synchronization only occurs after the coupling position error approaches zero. Therefore, this method does not show the synchronization effect during fault compensation in fault-tolerant control. Unlike AFTC IS-TSMC, AFTC S-TSMC can achieve synchronization after reaching the sliding mode phase. Hence, the synchronization of the error position in AFTC S-TSMC is effective in fault-tolerant control. From these results and the above analysis, it can be said that the ability to reduce the picking value of AFTC S-TSMC is greater than AFTC IS-TSMC, due to the effective synchronization in AFTC S-TSMC.

a) Joint 1   b) Joint 2   c) Joint 3
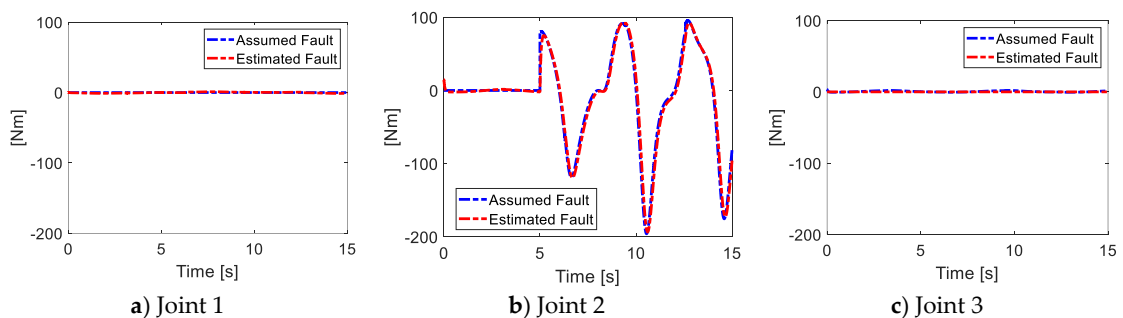
Figure 6.4**.** Fault estimation with a single fault at joint 2.

a) Joint 1   b) Joint 2   c) Joint 3

Figure 6.5. Tracking error at each joint with a single fault at joint 2.

a) **AFTC IS-TSMC**   b) **AFTC S-TSMC**

Figure 6.6. Tracking error at each joint for AFTC IS-TSMC and AFTC S-TSMC.
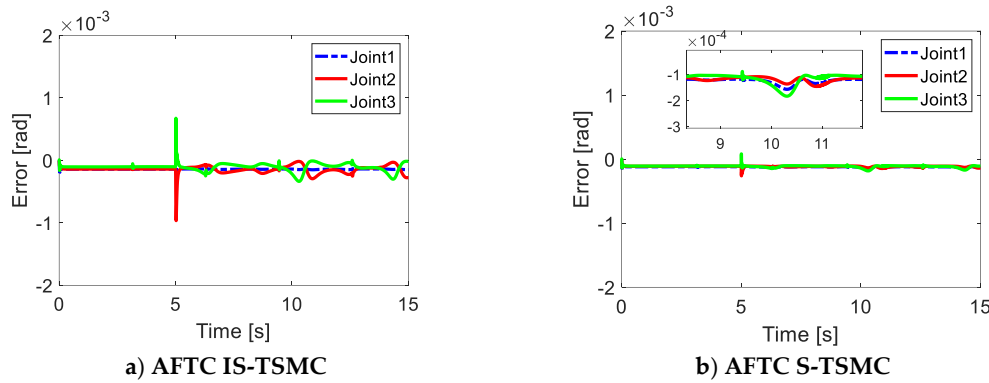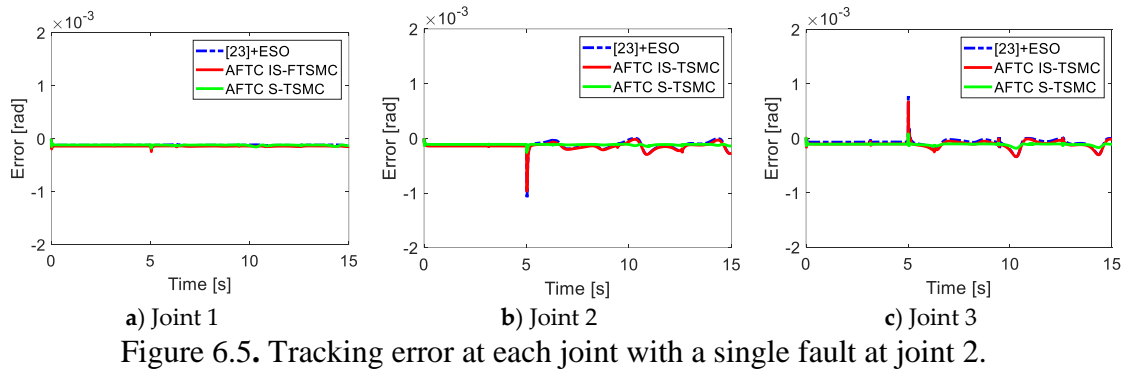
6.6 Experimental Results

In this section, implementations of the two proposed active fault-tolerant controls with a synchronous terminal sliding mode control and AFTC with a conventional terminal sliding mode control([23]+ESO) are described.

6.6.1 Experimental Setup

The experimental setup is shown in Figure 6.7 and uses a 3-DOF FARA-AT2 robot manipulator. This robot manipulator has 6-DOF, but for these experiments, joints 4, 5, and 6 were blocked. The 3-DOF FARA-AT2 robot had a CSMP series motor at each joint. The CSMP-02BB driver (Samsung, Namdong-gu, Incheon, Korea) was used for joints 1 and 2, while the CSMP-01BB driver was used for joint 3. The gear box at each joint was 120:1,120:1, and 100:1 at joints 1, 2, and 3, respectively. The encoder at each joint was a 2048 line count incremental encoder. The controller was run on Labview-FPGA, NI-PXI-8110 and NI-PXI-7842R PXI cards (National Instruments, Austin, Texas, USA) with the frequency control set at 500Hz. The NI-PXI-8110 was run on a Windows operating system.

Figure 6.7. 3-DOF FARA-AT2 robot manipulator.

The desired trajectory at each joint is given as:

$$q_{id}(t) = \frac{\pi}{6}\sin(\frac{\pi t}{8}) \quad (i = 1, 2, 3) \tag{6.39}$$

The related parameters were chosen to be $\alpha_1 = 8, \alpha_2 = 28, \alpha_3 = 7$ and $\varepsilon = 0.01$. The [75]+ESO is given as:

$$\tau_{[23]+ESO} = \tau_0 + \tau_{smc} + \tau_{ob} \tag{6.40}$$

where $\tau_{eq} = \mathbf{M}(q)(\ddot{q}_d + c\Lambda |e|^{\Lambda-I} \cdot \dot{e}) + \mathbf{H}(q, \dot{q})$, $\tau_o = \mathbf{M}(q)K_1 sign(S)$ and $\tau_{ob} = -\mathbf{M}(q)\hat{\phi}$, where $K_1 = diag(k_{1i}) \in \Re^{n \times n}$. The sliding mode surface was selected as:

$$s = \dot{e} + c\lceil e \rfloor^{\Lambda} \tag{6.41}$$

The parameters for the [75]+ESO were chosen as $c = diag(7;7;7)$, $K_1 = diag(80;80;110)$, $u_s = 10$ and $\Lambda = diag(0.5;0.5;0.5)$. The parameters for AFTC IS-TSMC were chosen as $\psi_1 = \psi_2 = \psi_3 = 2$, $\alpha = diag(1;1;1)$, $\beta = diag(0.5;0.5;0.5)$, $\Lambda = diag(0.5;0.5;0.5)$, $u_s = 3$, $\Gamma = diag(7;7;7)$ and $K_1 = diag(80;80;110)$. The parameters for AFTC S-TSMC were chosen as $\psi_1 = \psi_2 = \psi_3 = 2$, $\Lambda = diag(0.5;0.5;0.5)$, $c = diag(7;7;7)$, $u_s = 3$ and $K_1 = diag(80;80;110)$.

To avoid a singularity, the terms containing power $_{\Lambda-I}$ in Eq.s (6.16), (6.27) and (6.35) are replaced with the saturation function.

$$sat(u_f, u_s) = \begin{cases} u_s & if \ u_f \geq u_s \\ u_f & if \ u_f < u_s \end{cases} \tag{6.42}$$

where u_s is a positive constant, and $u_f = \Lambda |x|^{\Lambda-I} \cdot \dot{x}$ with $x = e, \varepsilon$ and E.

To avoid chattering, the signum functions in Eq.s (6.16), (6.27) and (6.35) are replaced with the saturation function.

$$sat(s) = \begin{cases} sgn(s) & if \ |s| \geq \lambda \\ \dfrac{s}{\lambda} & if \ |s| < \lambda \end{cases} \tag{6.43}$$

where $\lambda = 1.7$.

6.6.2 Experimental Results

The fault estimation results are shown in Figure 6.8. To reduce the high-frequency chattering and noises of fault estimation of $\hat{\phi}$ in Eq. (6.8) before it is used in the AFTC scheme, a simple low-pass filter was adopted as:

$$\hat{\phi}_k^{filted} = (1 - \upsilon)\hat{\phi}_{k-1}^{filted} + \upsilon\hat{\phi}_k \tag{6.44}$$

where $\hat{\phi}_k^{filted}$ and $\hat{\phi}_k$ are the output and input, respectively, of the low-pass filter at the kth step.

a) Joint 1   **b**) Joint 2   **c**) Joint 3
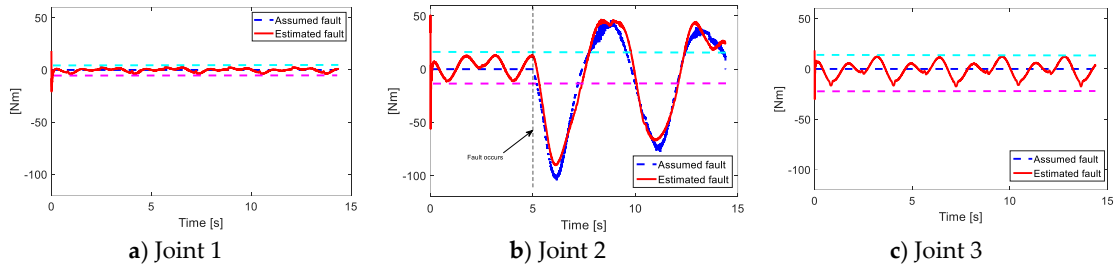
Figure 6.8. Fault estimation results with a single joint fault occurs at Joint 2. The aqua and pink dashed lines are the upper and lower thresholds, respectively.

The low-pass filter allowed the signal from the fault estimation (FE) to become smoother and was suitably applied to the AFTC schemes. However, the fault estimation also increased the time delay of the feedback to the controller. To ensure that the smoothness and time delay were acceptable, $\upsilon = 0.05$ was selected. Unlike in the simulation, the real system included large uncertainties, so upper and lower thresholds were used to detect faults. In Figure 6.9, the error tracking trajectory is presented. These results show that before five seconds, the error values at each joint were similar. However, after five seconds, due to the effects of synchronization control, the errors at each joint were different. Unlike in the simulation results, these experimental results show that AFTC IS-TSMC is more effective than AFTC S-TSMC. In a real system with large uncertainties and noise, the coupling position error in AFTC IS-TSMC with the integral term may affect how uncertainties are handled. In general, both proposed AFTCs can reduce the picking phenomenon, but the effect of each controller is different, and depends on its knowledge of the system.

a) Joint 1   **b**) Joint 2   **c**) Joint 3
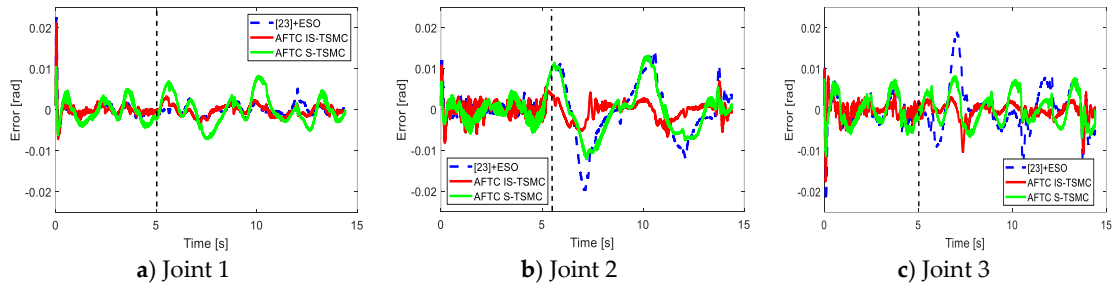
Figure 6.9. Tracking trajectory error results with a single joint fault occurs at Joint 2.

6.7 Conclusions

In this chapter, two finite-time active fault-tolerant controllers for a robot manipulator which combine a novel synchronous terminal sliding mode control with an extended state observer were proposed. The proposed controller can make the system quickly respond to the faults in a forward way before its feedback response after a fault estimation. This characteristic of two proposed controllers can reduce the occurrence of a picking phenomenon, due to the slow response of an active fault-tolerant control strategy. In addition, a novel synchronization error, coupling position error, and synchronous terminal sliding surface show better features such as better synchronization, avoiding the singularities when the trajectories cross zero, increasing the workspace of a robot, and finite-time convergence of the position errors. However, it should be noted that the effectiveness of the synchronization techniques depends on their knowledge of the system. The proposed active fault-tolerant control with synchronous terminal sliding mode control performs better in a well-known system, while the proposed active fault-tolerant control with integral synchronous terminal sliding mode control is more effective when the system has large uncertainties and noises. In the future work, the optimal tuning synchronization parameters will be studied with methods such as the genetic algorithm and neural network technique, to improve the effectiveness of synchronization technique in a fault-tolerant control.

Chapter 7: Conclusion and future works

7.1 Dissertation Conclusions

In this dissertation, a hardware platform development for robot manipulators was presented. The original controller was replaced with the new controller which established on Labview NI computer in torque control mode. The controllers such as Point-to-Point controller and Force controller were applied to the robot manipulator system to verify the hardware development capability and identification processing results of robot manipulators. From the above results, it can be seen that the robot system works well in torque mode. Especially, with the identification parameters of system, robot is able to interact with human by force controller with force sensor or without force sensor. This capability is an important feature in human-robot interaction controller. Finally, the main contribution of this dissertation is the proposed fault-tolerant control based on synchronous sliding mode control for robot manipulators. Due to the ability to make the error at each joint simultaneously approach to zero, the proposed control can reduce the impact of picking phenomena and provide fast recovery compare to the conventional controller. In addition, due to the constraint inside the synchronous control, the system is able to deal with high magnitude faults. The development of fault-tolerant control based on synchronization technique has special features such as new synchronization errors effecting a particular system, the ability to deal with high magnitude faults and the fine-time convergence. However, the accuracy of identification processing high effect to the ability to detect small magnitude of faults in system as well as the sensitivity in human-robot interaction controller.

7.2 Future works

Even though the hardware platform of the robot has been successfully setup and some control algorithms have been implemented, the following issues need further consideration:

- As mentioned earlier, the accuracy of dynamic parameters has a great impact on the detection of small magnitude fault in system. Therefore, the accuracy of the identification process should be improved.

- In this dissertation, the dynamic parameters of robot manipulator were determined only for 3-DOF. In the future, the identification for 6-DOF robot manipulator will bring more benefit to the real robot system.

- The selection of synchronous control parameters based on the experience of operator, so that the optimal technique for the proposed controller should be considered.

- The combination with intelligent control methods such as fuzzy or neural networks can increase the performance of the proposed controller.

- In this dissertation, the proposed controller is only applied to a single robot manipulator. However, it can also be applied to multi-robot manipulator system as well as multi-agents such as satellites, combination of mobile and manipulator system.

- The results of experiments with robot manipulator with force sensor and without force sensor show that the system can be developed to control human-robot interaction. Due to the ability to tolerate the fault, the system becomes more realistic and safer in collaboration robot control where humans and robots can share the workspace.

- Based on the development of low-level control and high-level control of 3-DOF, the high-level control should be applied for 6-DOF robot manipulator. It is able to develop a system capable of making artificial intelligence decisions for robot manipulator such as reinforcement learning.

Publications

Journal:

- Le Q.D.; Kang, H.-J. An Active Fault-Tolerant Control Based on Finite-Time Synchronous Fast Terminal Sliding Mode for Robot Manipulators. *Advances in Mechanical Engineering.* **2022 (Under Review)**.

- Le Q.D.; Kang, H.-J. Implementation of Sensor-less Contact Force Estimation in Collaborative Robot Based on Adaptive Third Order Sliding Mode Observer. *Systems Science and Control Engineering An Open Access Journal.* **2022 (Accepted)**.

- Le Q.D.; Kang, H.-J. Finite-Time Fault-Tolerant Control for a Robot Manipulator Based on Synchronous Terminal Sliding Mode Control. *Appl. Sci.* **2020**, *10*, 2998.

- Le Q.D.; Kang, H.-J. Implementation of Fault-Tolerant Control for a Robot Manipulator Based on Synchronous Sliding Mode Control. *Appl. Sci.* **2020**, *10*, 2534.

- Doan Q. V., Le, T. D., Le, Q. D., & Kang, H.-J. A neural network–based synchronized computed torque controller for three degree-of-freedom planar parallel manipulators with uncertainties compensation. *International Journal of Advanced Robotic Systems.* **2018**

- Le Q. D., & Kang, H. J. An Adaptive Controller with An Orthogonal Neural Network and A Third Order Sliding Mode Observer for Robot Manipulators. *International Journal of Mechanical Engineering and Robotics Research.* **2018,**Vol. 7, No. 2.

Chapter/Proceeding:

- Le, Q. D., & Kang, H. J. (2021, August). Sensor-Less Contact Force Estimation in Physical Human-Robot Interaction. In International Conference on Intelligent Computing (pp. 233-244). Springer, Cham.

- Le, Q. D., & Kang, H. J. (2021). Implementation of Fault-Tolerant Control for a Robot Manipulator Based on Synchronous Sliding Mode Control: An Advanced Study. New Approaches in Engineering Research Vol. 13, 63-82.

- Le, Q. D., & Kang, H. J. (2021). Study on Finite-time Fault-tolerant Control for a Robot Manipulator Based on Synchronous Terminal Sliding Mode Control. New Approaches in Engineering Research Vol. 13, 139-154.

- Le, Q. D., & Kang, H. J. (2019, August). Real Implementation of an Active Fault-tolerant Control Based on Super Twisting Technique for a Robot Manipulator. In International Conference on Intelligent Computing (pp. 294-305). Springer, Cham.

- Q. D. Le and H. Kang, "Real Implementation of Fault-Tolerant Sliding Mode Control for a Robot Manipulator," 2018 3rd International Conference on Control, Robotics and Cybernetics (CRC), Penang, Malaysia, 2018, pp. 48-52, doi: 10.1109/CRC.2018.00018.

- Le Q.D., Kang HJ., Le T.D. (2017) An Adaptive Position Synchronization Controller Using Orthogonal Neural Network for 3-DOF Planar Parallel Manipulators. In: Huang DS., Hussain A., Han K., Gromiha M. (eds) Intelligent Computing Methodologies. ICIC 2017. Lecture Notes in Computer Science, vol 10363. Springer, Cham.

- Le Q.D., Kang HJ., Le T.D. (2016) Adaptive Extended Computed Torque Control of 3 DOF Planar Parallel Manipulators Using Neural Network and Error Compensator. In: Huang DS., Han K., Hussain A. (eds) Intelligent Computing Methodologies. ICIC 2016. Lecture Notes in Computer Science, vol 9773. Springer, Cham.

Appendix A:

The kinetic energy of each link given as:

$$K_1 = \frac{1}{2}\left(m_1\left(P_{x1}^2 + P_{y1}^2\right) + I_{zz1}\right)\dot{\theta}_1^2$$

$$K_2 = \frac{1}{2}m_2 v_{c2}^2 + \frac{1}{2}\omega_2^T I_2 \omega_2$$

$$v_{c2} = \sqrt{x_{x2}^2 + y_{c2}^2 + z_{c2}^2} \quad \text{where} \quad \begin{cases} x_{c2} = -P_{x2}c_1s_2 - P_{y2}c_1c_2 + P_{z2}s_1 + l_1c_1 \\ y_{c2} = -P_{x2}s_1s_2 - P_{y2}s_1s_2 - P_{z2}c_1 + l_1s_1 \\ z_{c2} = P_{x2}c_2 - P_{y2}s_2 \end{cases}$$

where $P_{xi}, P_{yi}, P_{zi} \ (i = 1, 2, 3)$ is the position of the mass center of link i

$$\omega_2 = \begin{bmatrix} c_2\dot{\theta}_1 \\ -s_2\dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}, \quad I_2 = \begin{bmatrix} I_{xx2} & 0 & 0 \\ 0 & I_{yy2} & 0 \\ 0 & 0 & I_{zz2} \end{bmatrix}.$$

$$K_3 = \frac{1}{2}m_3 v_{c3}^2 + \frac{1}{2}\omega_3^T I_3 \omega_3$$

where $\omega_3 = \begin{bmatrix} s_{23}\dot{\theta}_1 \\ c_{23}\dot{\theta}_1 \\ \dot{\theta}_2 + \dot{\theta}_3 \end{bmatrix}, v_3 = \sqrt{x_{x3}^2 + y_{c3}^2 + z_{c3}^2}$ which $\begin{cases} x_{c3} = P_{x3}c_1c_{23} - P_{y3}c_1s_{23} + P_{z3}s_1 + c_1(l_1 - l_2s_2) \\ y_{c3} = P_{x3}s_1c_{23} - P_{y3}s_1s_{23} - P_{z3}c_1 + s_1(l_1 - l_2s_2) \\ z_{c3} = P_{x3}s_{23} + P_{y3}c_{23} + l_2c_2 \end{cases}$

The potential energy of each link given as:

$$P_1 = 0$$

$$P_2 = m_2 g\left(P_{x2}c_2 - P_{y2}s_2\right)$$

$$P_3 = m_3 g\left(P_{x3}s_{23} + P_{y3}c_{23} + l_2c_2\right)$$

where $v_{c2} = \sqrt{x_{x2}^2 + y_{c2}^2 + z_{c2}^2}$ which $\begin{cases} x_{c2} = -P_{x2}c_1s_2 - P_{y2}c_1c_2 + P_{z2}s_1 + l_1c_1 \\ y_{c2} = -P_{x2}s_1s_2 - P_{y2}s_1s_2 - P_{z2}c_1 + l_1s_1 \\ z_{c2} = P_{x2}c_2 - P_{y2}s_2 \end{cases}, \quad \omega_2 = \begin{bmatrix} c_2\dot{\theta}_1 \\ -s_2\dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$

$$I_3 = \begin{bmatrix} I_{xx3} & 0 & 0 \\ 0 & I_{yy3} & 0 \\ 0 & 0 & I_{zz3} \end{bmatrix}.$$

The Lagrangian of a manipulator is

$$L = K_1 + K_2 + K_3 - P_1 - P_2 - P_3$$

where

$$
L = \frac{1}{2}\left(m_1\left(P_{x1}^2 + P_{y1}^2\right) + I_{z1}\right)\dot{\theta}_1^2 + \frac{1}{2}m_2 P_{x2}^2\left(\dot{\theta}_2^2 + \dot{\theta}_1^2 s_2^2\right) + \frac{1}{2}m_2 P_{y2}^2\left(\dot{\theta}_2^2 + \dot{\theta}_1^2 c_2^2\right) + \frac{1}{2}m_2 P_{z2}^2\dot{\theta}_1^2
$$
$$
+ \frac{1}{2}m_2 P_{x2}\left(-2l_1\dot{\theta}_2^2 s_2\right) + \frac{1}{2}m_2 P_{y2}\left(-2l_1\dot{\theta}_1^2 c_2\right) + \frac{1}{2}m_2 P_{x2}P_{y2}\left(2\dot{\theta}_1^2 s_2 c_2\right)
$$
$$
+ \frac{1}{2}m_2 P_{y2}P_{z2}\left(2\dot{\theta}_1\dot{\theta}_2 s_2\right) + \frac{1}{2}m_2 P_{x2}P_{z2}\left(-2\dot{\theta}_1\dot{\theta}_2 c_2\right) + \frac{1}{2}m_2 l_1^2\dot{\theta}_1^2 + \frac{1}{2}I_{xx2}\dot{\theta}_1^2 c_2^2
$$
$$
+ \frac{1}{2}I_{yy2}\dot{\theta}_1^2 s_2^2 + \frac{1}{2}I_{zz2}\dot{\theta}_2^2 + \frac{1}{2}m_3 P_{x3}^2\left(\dot{\theta}_1^2 c_{23}^2 + \left(\dot{\theta}_2 + \dot{\theta}_3\right)^2\right)
$$
$$
+ \frac{1}{2}m_3 P_{y3}^2\left(\dot{\theta}_1^2 s_{23}^2 + \left(\dot{\theta}_2 + \dot{\theta}_3\right)^2\right) + \frac{1}{2}m_3 P_{z3}^2\dot{\theta}_1^2 + \frac{1}{2}m_3 P_{x3}\left(2l_2\dot{\theta}_2\left(\dot{\theta}_2 + \dot{\theta}_3\right)s_3 + 2\dot{\theta}_1^2 c_{23}\left(l_1 - l_2 s_2\right)\right)
$$
$$
+ \frac{1}{2}m_3 P_{y3}\left(2l_2\dot{\theta}_2\left(\dot{\theta}_2 + \dot{\theta}_3\right)c_3 - 2\dot{\theta}_1^2 s_{23}\left(l_1 - l_2 s_2\right)\right) + \frac{1}{2}m_3 P_{z3}\left(-2l_2\dot{\theta}_1\dot{\theta}_2 c_2\right) + \frac{1}{2}m_3 P_{x3}P_{y3}\left(-2\dot{\theta}_1^2 c_{23}s_{23}\right)
$$
$$
+ \frac{1}{2}m_3 P_{y3}P_{z3}\left(-\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)c_{23}\right) + \frac{1}{2}m_3 P_{x3}P_{z3}\left(-2\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)s_{23}\right) + \frac{1}{2}m_3\dot{\theta}_1^2\left(l_1 - l_2 s_2\right)^2
$$
$$
+ \frac{1}{2}I_{xx3}\dot{\theta}_1^2 s_{23}^2 + \frac{1}{2}I_{yy3}\dot{\theta}_1^2 c_{23}^2 + \frac{1}{2}I_{zz3}\left(\dot{\theta}_2 + \dot{\theta}_3\right)^2 - m_2 g P_{x2}c_2 + m_2 g P_{y2}s_2 - m_3 g P_{x3}s_{23} - m_3 g P_{y3}c_{23} - m_3 g l_2 c_2
$$

The Eq.s of motion for the manipulator are then given by

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = \tau$$

Joint 1:

$$
\frac{\partial L}{\partial \dot{\theta}_1} = \left(m_1\left(P_{x1}^2 + P_{y1}^2\right) + I_{zz1}\right)\dot{\theta}_1 + m_2 P_{x2}^2 s_2^2\dot{\theta}_1 + m_2 P_{y2}^2 c_2^2\dot{\theta}_1 + m_2 P_{z2}^2\dot{\theta}_1 - 2m_2 P_{y2}l_1 c_2\dot{\theta}_1 + 2m_2 P_{x2}P_{y2}c_2 s_2\dot{\theta}_1
$$
$$
+ m_2 P_{y2}P_{z2}s_2\dot{\theta}_2 - m_2 P_{x2}P_{z2}c_2\dot{\theta}_2 + m_2 l_1^2\dot{\theta}_1 + I_{xx2}c_2^2\dot{\theta}_1 + I_{yy2}s_2^2\dot{\theta}_1 + m_3 P_{x3}^2 c_{23}^2\dot{\theta}_1 + m_3 P_{y3}^2 s_{23}^2\dot{\theta}_1 + m_3 P_{z3}^2\dot{\theta}_1
$$
$$
+ 2m_3 P_{x3}c_{23}(l_1 - l_2 s_2)\dot{\theta}_1 - 2m_3 P_{y3}s_{23}(l_1 - l_2 s_2)\dot{\theta}_1 - m_3 P_{z3}l_2 c_2\dot{\theta}_2 - 2m_3 P_{x3}P_{y3}c_{23}s_{23}\dot{\theta}_1
$$
$$
- \frac{1}{2}m_3 P_{y3}P_{z3}c_{23}(\dot{\theta}_2 + \dot{\theta}_3) - m_3 P_{x3}P_{z3}s_{23}(\dot{\theta}_2 + \dot{\theta}_3) + m_3(l_1 - l_2 s_2)^2\dot{\theta}_1 + I_{xx3}s_{23}^2\dot{\theta}_1 + I_{yy3}c_{23}^2\dot{\theta}_1
$$

$$
\frac{d(\partial L)}{d(\partial\dot{\theta}_1)} = \left(m_1\left(P_{x1}^2 + P_{y1}^2\right) + I_{zz1}\right)\ddot{\theta}_1 + m_2 P_{x2}^2 s_2^2\ddot{\theta}_1 + 2m_2 P_{x2}^2 c_2 s_2\dot{\theta}_1\dot{\theta}_2 + m_2 P_{y2}^2 c_2^2\ddot{\theta}_1 - 2m_2 P_{y2}^2 c_2 s_2\dot{\theta}_1\dot{\theta}_2
$$
$$
+ m_2 P_{z2}^2\ddot{\theta}_1 - 2m_2 P_{y2}l_1 c_2\ddot{\theta}_1 + 2m_2 P_{y2}l_1 s_2\dot{\theta}_1\dot{\theta}_2 + 2m_2 P_{x2}P_{y2}s_2 c_2\ddot{\theta}_1 + 2m_2 P_{x2}P_{y2}c_2^2\dot{\theta}_1\dot{\theta}_2 - 2m_2 P_{x2}P_{y2}s_2^2\dot{\theta}_1\dot{\theta}_2
$$
$$
+ m_2 P_{y2}P_{z2}s_2\ddot{\theta}_2 + m_2 P_{y2}P_{z2}c_2\dot{\theta}_2^2 - m_2 P_{x2}P_{z2}c_2\ddot{\theta}_2 + m_2 P_{x2}P_{z2}s_2\dot{\theta}_2^2 + m_2 l_1^2\ddot{\theta}_1 + I_{xx2}c_2^2\ddot{\theta}_1 - 2I_{xx2}c_2 s_2\dot{\theta}_1\dot{\theta}_2
$$
$$
+ I_{yy2}s_2^2\ddot{\theta}_1 + 2I_{yy2}s_2 c_2\dot{\theta}_1\dot{\theta}_2 + m_3 P_{x3}^2 c_{23}^2\ddot{\theta}_1 - 2m_3 P_{x3}^2 c_{23}s_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) + m_3 P_{y3}^2 s_{23}^2\ddot{\theta}_1 + 2m_3 P_{y3}^2 s_{23}c_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)
$$

$$+ m_3 P_{z3}^2 \ddot{\theta}_1 + 2m_3 P_{x3}c_{23}\left(l_1 - l_2 s_2\right)\ddot{\theta}_1 - 2m_3 P_{x3}s_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1\left(\dot{\theta}_2 - \dot{\theta}_3\right) + 2m_3 P_{x3}l_2 c_2 c_3 \dot{\theta}_1 \dot{\theta}_2 - 2m_3 P_{y3}s_{23}\left(l_1 - l_2 s_2\right)\ddot{\theta}$$

$$- 2m_3 P_{y3}c_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1\left(\dot{\theta}_2 - \dot{\theta}_3\right) + 2m_3 P_{y3}l_2 c_2 s_{23}\dot{\theta}_1\dot{\theta}_2 - m_3 P_{z3}l_2 c_2 \ddot{\theta}_2 - m_3 P_{z3}l_2 s_2 \dot{\theta}_2^2 - 2m_3 P_{x3}P_{y3}c_{23}s_{23}\ddot{\theta}_1$$

$$+ 2m_3 P_{x3}P_{y3}s_{23}^2 \dot{\theta}_1\left(\dot{\theta}_2 - \dot{\theta}_3\right) - 2m_3 P_{x3}P_{y3}c_{23}^2 \dot{\theta}_1\left(\dot{\theta}_2 - \dot{\theta}_3\right) - \frac{1}{2}m_3 P_{y3}P_{z3}c_{23}\left(\ddot{\theta}_2 - \ddot{\theta}_3\right) + \frac{1}{2}m_3 P_{y3}P_{z3}s_{23}\left(\dot{\theta}_2 - \dot{\theta}_3\right)^2$$

$$- m_3 P_{x3}P_{z3}s_{23}\left(\ddot{\theta}_2 - \ddot{\theta}_3\right) - m_3 P_{x3}P_{z3}c_{23}\left(\dot{\theta}_2 - \dot{\theta}_3\right)^2 - m_3\left(l_1 - l_2 s_2\right)^2 \ddot{\theta} - 2m_3 l_2 c_{21}\left(l_1 - l_2 s_2\right)\dot{\theta}_1\dot{\theta}_2 + I_{xx3}s_{23}^2 \ddot{\theta}_1$$

$$+ 2I_{xx3}s_{23}c_{23}\dot{\theta}_1\left(\dot{\theta}_2 - \dot{\theta}_3\right) + I_{yy3}c_{23}^2 \ddot{\theta}_1 - 2I_{yy3}c_{23}s_{23}\dot{\theta}_1\left(\dot{\theta}_2 - \dot{\theta}_3\right)$$

$$\frac{\partial L}{\partial \theta_1} = 0$$

Friction term was added $F = k_1 sign(\dot{\theta}_i) + k_2 \dot{\theta}_i$ where k_1 and k_2 are constant.

Joint 2:

$$\frac{\partial L}{\partial \dot{\theta}_2} = m_2 P_{x2}^2 \dot{\theta}_2 + m_2 P_{y2}^2 \dot{\theta}_2 - 2m_2 P_{x2}l_1 s_2 \dot{\theta}_2 + m_2 P_{y2}P_{z2}s_2 \dot{\theta}_1 - m_2 P_{x2}P_{z2}c_2 \dot{\theta}_1 + I_{zz2}\dot{\theta}_2 + m_3 P_{x3}^2\left(\dot{\theta}_2 + \dot{\theta}_3\right) + 2m_3 P_{x3}l_2 s_3 \dot{\theta}_2$$

$$+ m_3 P_{x3}l_2 s_3 \dot{\theta}_3 + 2m_3 P_{y3}l_2 c_3 \dot{\theta}_2 + m_3 P_{y3}l_2 c_3 \dot{\theta}_3 - m_3 P_{z3}l_2 c_2 \dot{\theta}_1 - \frac{1}{2}m_3 P_{y3}P_{z3}c_{23}\dot{\theta}_1 - m_3 P_{x3}P_{z3}s_{23}\dot{\theta}_1 + m_3 l_2^2 \dot{\theta}_2 + I_{zz3}\left(\dot{\theta}_2 + \dot{\theta}_3\right)$$

$$\frac{d\left(\partial L\right)}{d\left(\partial \dot{\theta}_2\right)} = m_2 P_{x2}^2 \ddot{\theta}_2 + m_2 P_{y2}^2 \ddot{\theta}_2 - 2m_2 P_{x2}l_1 s_2 \ddot{\theta}_2 - 2m_2 P_{x2}l_1 c_2 \dot{\theta}_2^2 + m_2 P_{y2}s_2 \dot{\theta}_1 + m_2 P_{y2}P_{z2}c_2 \dot{\theta}_1 \dot{\theta}_2 - m_2 P_{x2}P_{z2}c_2 \ddot{\theta}_1$$

$$+ m_2 P_{x2}P_{z2}s_2 \dot{\theta}_1 \dot{\theta}_2 + I_{zz2}\ddot{\theta}_2 + m_3 P_{x3}^2\left(\ddot{\theta}_2 + \ddot{\theta}_3\right) + m_3 P_{y3}^3\left(\ddot{\theta}_2 + \ddot{\theta}_3\right) + 2m_3 P_{x3}l_2 s_3 \ddot{\theta}_2 + 2m_3 P_{x3}l_2 c_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 P_{x3}l_2 s_3 \ddot{\theta}_3$$

$$+ m_3 P_{x3}l_2 c_3 \dot{\theta}_3^2 + 2m_3 P_{y3}l_2 c_3 \ddot{\theta}_2 - 2m_3 P_{y3}l_2 s_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 P_{y3}l_2 c_3 \ddot{\theta}_3 - m_3 P_{y2}l_2 s_3 \dot{\theta}_3^2 - m_3 P_{z3}l_2 c_2 \ddot{\theta}_1 + m_3 P_{z3}l_2 s_2 \dot{\theta}_1 \dot{\theta}_2$$

$$- \frac{1}{2}m_3 P_{y3}P_{z3}c_{23}\ddot{\theta}_1 + \frac{1}{2}m_3 P_{y3}P_{z3}s_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) - m_3 P_{x3}P_{z3}s_{23}\ddot{\theta}_1 - m_3 P_{x3}P_{z3}c_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) + m_3 l_2^2 \ddot{\theta}_2 + I_{zz3}\left(\ddot{\theta}_2 + \ddot{\theta}_3\right)$$

$$\frac{\partial L}{\partial \theta_2} = m_2 P_{x2}^2 \dot{\theta}_1^2 s_2 c_2 - m_2 P_{y2}^2 c_2 s_2 \dot{\theta}_1^2 - m_2 P_{x2}l_1 c_2 \dot{\theta}_2^2 m_2 P_{y2}l_1 s_2 \dot{\theta}_1^2 + m_2 P_{x2}P_{y2}c_2^2 \dot{\theta}_1^2 + m_2 P_{x2}P_{y2}s_2^2 \dot{\theta}_1^2 + m_2 P_{x2}P_{z2}c_2 \dot{\theta}_1 \dot{\theta}_2$$

$$+ m_2 P_{x2}P_{z2}s_2 \dot{\theta}_1 \dot{\theta}_2 - I_{xx2}c_2 s_2 \dot{\theta}_1^2 + I_{yy2}c_2 s_2 \dot{\theta}_1^2 - m_3 P_{x3}^2 c_{23}s_{23}\dot{\theta}_1^2 + m_3 P_{y3}^2 c_{23}s_{23}\dot{\theta}_1^2 - m_3 P_{x3}s_{23}(l_1 - l_2 s_2)\dot{\theta}_1^2 - m_3 P_{x3}c_{23}l_2 c_2 \dot{\theta}_1^2$$

$$- m_3 P_{y3}c_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 + m_3 P_{y3}s_{23}l_2 c_2 \dot{\theta}_1^2 + m_3 P_{z3}l_2 s_2 \dot{\theta}_1 \dot{\theta}_2 + m_3 P_{x3}P_{y3}s_{23}^2 \dot{\theta}_1^2 - m_3 P_{x3}P_{y3}c_{23}^2 \dot{\theta}_1^2 + \frac{1}{2}m_3 P_{y3}P_{z3}s_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)$$

$$- m_3 P_{x3}P_{z3}c_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) - m_3 l_2 c_2\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 + I_{xx3}s_{23}c_{23}\dot{\theta}_1^2 - I_{yy3}c_{23}s_{23}\dot{\theta}_1^2 + m_2 P_{x2}gs_2 + m_3 P_{y2}gc_2 - m_3 P_{x3}gc_{23}$$

$$+ m_3 P_{y3}gs_{23} + m_3 gl_2 s_2$$

$$\tau_2 = \left(m_2 P_{y2}P_{z2}s_2 - m_2 P_{x2}P_{z2}c_2 - m_3 P_{z3}l_2 c_2 - \frac{1}{2}m_3 P_{y3}P_{z3}c_{23} - m_3 P_{x3}P_{z3}s_{23}\right)\ddot{\theta}_1$$

$$+ \left(m_2 P_{x2}^2 + m_2 P_{y2}^2 - 2m_2 P_{x2}l_1 s_2 + I_{zz2} + m_3 P_{x3}^2 + m_3 P_{y3}^2 + 2m_3 P_{x3}l_2 s_3 + 2m_3 P_{y3}l_2 c_3 + m_3 l_2^2 + I_{zz3}\right)\ddot{\theta}_2$$

$$+ \left(m_3 P_{x3}^2 + m_3 P_{y3}^2 + m_3 P_{x3}l_2 s_{23} + m_3 P_{y3}l_2 c_3 + I_{zz3}\right)\ddot{\theta}_3 - m_2 P_{x2}l_1 c_2 \dot{\theta}_2^2 + 2m_3 P_{x3}l_2 c_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 P_{x3}l_2 c_3 \dot{\theta}_3^2$$

$$- 2m_3 P_{y3}l_2 s_3 \dot{\theta}_2 \dot{\theta}_3 - m_3 P_{y3}l_2 s_3 \dot{\theta}_3^2 - m_2 P_{x2}^2 c_2 s_2 \dot{\theta}_1^2 + m_2 P_{y2}^2 c_2 s_2 \dot{\theta}_1^2 - m_3 P_{x3}l_1 s_2 \dot{\theta}_1^2 + m_2 P_{x2}P_{y2}c_2^2 \dot{\theta}_1^2$$

$$+ I_{xx2}s_2 c_2 \dot{\theta}_1^2 + I_{yy2}s_2 c_2 \dot{\theta}_1^2 + m_3 P_{x3}^2 c_{23}s_{23}\dot{\theta}_1^2 - m_3 P_{y3}^2 s_{23}c_{23}\dot{\theta}_1^2 + m_3 P_{x3}s_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 + m_3 P_{x3}l_2 c_2 c_{23}\dot{\theta}_1^2$$

$$+ m_3 P_{y3}c_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 - m_3 P_{y3}s_{23}l_2 c_2 \dot{\theta}_1^2 - m_3 P_{x3}P_{y3}\left(s_{23}^2 - c_{23}^2\right)\dot{\theta}_1^2 + m_3 l_2 c_2\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 - I_{xx3}s_{23}c_{23}\dot{\theta}_1^2$$

$$+ I_{yy3}c_{23}s_{23}\dot{\theta}_1^2 - m_2 P_{x2}gs_2 - m_2 P_{y2}gc_2 + m_3 P_{x3}gc_{23} - m_3 P_{y3}gs_{23} - m_3 gl_2 s_2 + k_3 sign\left(\dot{\theta}_2\right) + k_4 \dot{\theta}_2$$

Joint 3:

$$\frac{\partial L}{\partial \dot{\theta}_3} = m_3 P_{x3}^2 \left(\dot{\theta}_2 + \dot{\theta}_3 \right) + m_3 P_{y3}^2 \left(\dot{\theta}_2 + \dot{\theta}_3 \right) + m_3 P_{x3} l_2 s_3 \dot{\theta}_2 - \frac{1}{2} m_3 P_{y3} P_{z3} c_{23} \dot{\theta}_1 - m_3 P_{x3} P_{z3} s_{23} \dot{\theta}_1 + I_{zz3} \left(\dot{\theta}_2 + \dot{\theta}_3 \right)$$

$$\frac{d\left(\partial L \right)}{d\left(\partial \dot{\theta}_3 \right)} = m_3 P_{x3}^2 \left(\ddot{\theta}_2 + \ddot{\theta}_3 \right) + m_3 P_{y3}^2 \left(\ddot{\theta}_2 + \ddot{\theta}_3 \right) + m_3 P_{x3} l_2 s_3 \ddot{\theta}_2 + m_3 P_{x3} l_2 c_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 P_{y3} l_2 c_3 \ddot{\theta}_2 - m_3 P_{y3} l_2 s_3 \dot{\theta}_2 \dot{\theta}_3$$

$$- \frac{1}{2} m_3 P_{y3} P_{z3} c_{23} \ddot{\theta}_1 + \frac{1}{2} m_3 P_{y3} P_{z3} s_{23} \dot{\theta}_1 \left(\dot{\theta}_2 + \dot{\theta}_3 \right) - m_3 P_{x3} P_{z3} s_{23} \ddot{\theta}_1 - m_3 P_{x3} P_{z3} c_{23} \dot{\theta}_1 \left(\dot{\theta}_2 + \dot{\theta}_3 \right) + I_{zz3} \left(\ddot{\theta}_2 + \ddot{\theta}_3 \right)$$

$$\frac{\partial L}{\partial \theta_3} = -m_3 P_{x3}^2 c_{23} s_{23} \dot{\theta}_1^2 + m_3 P_{y3}^2 s_{23} c_{23} \dot{\theta}_1^2 + m_3 P_{x3} l_2 c_3 \dot{\theta}_2^2 - m_3 P_{x3} s_{23} \left(l_1 - l_2 s_2 \right) \dot{\theta}_1^2 - m_3 P_{y3} l_2 s_3 \dot{\theta}_2 \left(\dot{\theta}_2 + \dot{\theta}_3 \right)$$

$$- m_3 P_{y3} c_{23} \left(l_1 - l_2 s_2 \right) \dot{\theta}_1^2 - m_3 P_{x3} P_{y3} c_{23}^2 \dot{\theta}_1^2 + m_3 P_{x3} P_y s_{23}^2 \dot{\theta}_1^2 + m_3 P_{y3} P_{z3} s_{23} \dot{\theta}_1 \left(\dot{\theta}_2 + \dot{\theta}_3 \right) + I_{xx3} s_{23} c_{23} \dot{\theta}_1^2$$

$$- I_{yy3} c_{23} s_{23} \dot{\theta}_1^2 - m_3 P_{x3} g c_{23} + m_3 P_{y3} g s_{23}$$

$$\tau_3 = \left(-\frac{1}{2} m_3 P_{y3} P_{z3} c_{23} - m_3 P_{x3} P_{z3} s_{23} \right) \ddot{\theta}_1 + \left(m_3 P_{x3}^2 + m_3 P_{y3}^2 + m_3 P_{x3}^2 l_2 s_3 + m_3 P_{y3} l_2 c_3 + I_{zz3} \right) \ddot{\theta}_2$$

$$+ \left(m_3 P_{x3}^2 + m_3 P_{y3}^2 + I_{zz3} \right) \ddot{\theta}_3 - m_3 P_{y3} l_2 s_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 P_{x3}^2 c_{23} s_{23} \dot{\theta}_1^2 - m_3 P_{y3}^2 s_{23} c_{23} \dot{\theta}_1^2 - m_3 P_{x3} l_2 c_3 \dot{\theta}_2^2$$

$$+ m_3 P_{x3} s_{23} (l_1 - l_2 s_2) \dot{\theta}_1^2 + m_3 P_{y3} l_2 s_{23} \dot{\theta}_2^2 + m_3 P_{y3} l_2 s_3 \dot{\theta}_2 \dot{\theta}_3 + m_3 P_{y3} c_{23} \left(l_1 - l_2 s_2 \right) \dot{\theta}_1^2 + m_3 P_{x3} P_{y3} c_{23}^2 \dot{\theta}_1^2$$

$$- m_3 P_{x3} P_{y3} s_{23}^2 \dot{\theta}_1^2 - I_{xx3} s_{23} c_{23} \dot{\theta}_1^2 + I_{yy3} c_{23} s_{23} \dot{\theta}_1^2 + m_3 P_{x3} g c_{23} - m_3 P_{y3} g s_{23} + k_5 sign\left(\dot{\theta}_3 \right) + k_6 \dot{\theta}_3$$

Appendix B:

Let $\pi = [\pi_1, \ldots, \pi_{32}]^T$ where $\pi_1 = m_1\left(P_{x1}^2 + P_{y1}^2\right) + I_{zz1}$, $\pi_2 = m_2 P_{x2}^2$, $\pi_3 = m_2 P_{y2}^2$, $\pi_4 = m_2 P_{z2}^2$, $\pi_5 = m_2 P_{y2}$,

$\pi_6 = m_2 P_{x2} P_{y2}$, $\pi_7 = m_2 P_{y2} P_{z2}$, $\pi_8 = m_2 P_{x2} P_{z2}$, $\pi_9 = m_2$, $\pi_{10} = I_{xx2}$, $\pi_{11} = I_{yy2}$, $\pi_{12} = m_3 P_{x3}^2$,

$\pi_{13} = m_3 P_{y3}^2$, $\pi_{14} = m_3 P_{z3}^2$, $\pi_{15} = m_3 P_{x3}$, $\pi_{16} = m_3 P_{y3}$, $\pi_{17} = m_3 P_{z3}$, $\pi_{18} = m_3 P_{x3} P_{y3}$, $\pi_{19} = m P_{y3} P_{z3}$,

$\pi_{20} = m_3 P_{x3} P_{z3}$, $\pi_{21} = m_3$, $\pi_{22} = I_{xx3}$, $\pi_{23} = I_{yy3}$, $\pi_{25} = I_{zz2}$, $\pi_{26} = I_{zz3}$, $\pi_{27} = k_1$, $\pi_{28} = k_2$, $\pi_{29} = k_3$,

$\pi_{30} = k_4$, $\pi_{31} = k_5$, $\pi_{32} = k_6$.

Torque output at each joint can be rewritten as

$$
\begin{aligned}
\tau_1 = &\ \pi_1 \ddot{\theta}_1 + \pi_2\left(\ddot{\theta}_1 s_2^2 + 2s_2 c_2 \dot{\theta}_1 \dot{\theta}_2\right) + \pi_3\left(\ddot{\theta}_1 c_2^2 - 2c_2 s_2 \dot{\theta}_1 \dot{\theta}_2\right) \\
&+ \pi_4 \ddot{\theta}_1 + \pi_5\left(-2l_1 c_2 \ddot{\theta}_1 + 2l_1 s_2 \dot{\theta}_1 \dot{\theta}_2\right) + \pi_6\left(2\ddot{\theta}_1 c_2 s_2 + 2c_2^2 \dot{\theta}_1 \dot{\theta}_2 - 2s_2^2 \dot{\theta}_1 \dot{\theta}_2\right) \\
&+ \pi_7\left(s_2 \ddot{\theta}_2 + c_2 \dot{\theta}_2^2\right) + \pi_8\left(-\ddot{\theta}_2 c_2 + \dot{\theta}_2^2 s_2\right) + \pi_9 l_1^2 \ddot{\theta}_1 + \pi_{12}\left(\ddot{\theta}_1 c_{23}^2 - 2c_{23} s_{23} \dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)\right) \\
&+ \pi_{13}\left(\ddot{\theta}_1 s_{23}^2 + 2s_{23} c_{23} \dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)\right) + \pi_{14} \ddot{\theta}_1 + \pi_{15}\left(2c_{23}\left(l_1 - l_2 s_2\right) - 2s_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) + 2l_2 c_2 c_{23} \dot{\theta}_1 \dot{\theta}_2\right) \\
&+ \pi_{16}\left(-2s_{23}\left(l_1 - l_2 s_2\right) - 2c_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) + 2s_{23} l_2 c_2 \dot{\theta}_1 \dot{\theta}_2\right) + \pi_{17}\left(-l_2 c_2 \ddot{\theta}_2 - l_2 s_2 \dot{\theta}_2^2\right) \\
&+ \pi_{18}\left(-2c_{23} s_{23} + 2s_{23}^2 \dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right) - 2c_{23}^2 \dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)\right) + \pi_{19}\left(-\frac{1}{2}c_{23}\ddot{\theta}_2 - \frac{1}{2}c_{23}\ddot{\theta}_3 + \frac{1}{2}s_{23}\left(\dot{\theta}_2 + \dot{\theta}_3\right)^2\right) \\
&+ \pi_{20}\left(-s_{23}\ddot{\theta}_2 - s_{23}\ddot{\theta}_3 - c_{23}\left(\dot{\theta}_2 + \dot{\theta}_3\right)^2\right) + \pi_{21}\left(\ddot{\theta}_1\left(l_1 - l_2 s_2\right)^2 - 2l_2 c_2\left(l_1 - l_2 s_2\right)\dot{\theta}_1 \dot{\theta}_2\right) \\
&+ \pi_{22}\left(s_{23}^2 \ddot{\theta}_1 - 2c_2 s_2 \dot{\theta}_1 \dot{\theta}_2 + 2s_{23} c_{23} \dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)\right) + \pi_{23}\left(c_{23}^2 \ddot{\theta}_1 - 2\dot{\theta}_1 c_{23} s_{23}\left(\dot{\theta}_2 + \dot{\theta}_3\right)\right) + \pi_{10} c_2^2 \ddot{\theta}_1 \\
&+ \pi_{11} 2s_2 c_2 \dot{\theta}_1 \dot{\theta}_2 + \pi_{27}\, sign\left(\dot{\theta}_1\right) + \pi_{28}\dot{\theta}_1
\end{aligned}
$$

$$
\begin{aligned}
\tau_2 = &\ \pi_2\left(\ddot{\theta}_2 - \dot{\theta}_1^2 c_2 s_2\right) + \pi_3\left(\ddot{\theta}_2 + \dot{\theta}_1^2 c_2 s_2\right) + \pi_{24}\left(-2\ddot{\theta}_2 l_1 s_2 - l_1 c_2 \dot{\theta}_2^2 - g s_2\right) + \pi_5\left(-l_1 \dot{\theta}_1^2 s_2 - g c_2\right) \\
&+ \pi_6\left(\dot{\theta}_1^2 c_2^2 - \dot{\theta}_1^2 s_2^2\right) + \pi_7\left(\ddot{\theta}_1 s_2\right) + \pi_8\left(-c_2\right) + \pi_{10}\left(c_2 s_2 \dot{\theta}_1^2\right) + \pi_{11}\left(-s_2 c_2 \dot{\theta}_1^2\right) + \pi_{25}\ddot{\theta}_2 \\
&+ \pi_{12}\left(\ddot{\theta}_2 + \ddot{\theta}_3 + c_{23} s_{23} \dot{\theta}_1^2\right) + \pi_{13}\left(\ddot{\theta}_2 + \ddot{\theta}_3 - s_{23} c_{23} \dot{\theta}_1^2\right) + \pi_{18}\left(-\dot{\theta}_1^2\left(s_{23}^2 - c_{23}^2\right)\right) + \pi_{19}\left(-\frac{1}{2}c_{23}\ddot{\theta}_1\right) \\
&+ \pi_{20}\left(-s_{23}\ddot{\theta}_1\right) + \pi_{15}\left(2l_2 s_3 \ddot{\theta}_2 + l_2 s_3 \ddot{\theta}_3 + 2l_2 c_3 \dot{\theta}_2 \dot{\theta}_3 + l_2 c_3 \dot{\theta}_3^2 + s_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 + l_2 c_2 c_{23}\dot{\theta}_1^2 + g c_{23}\right) \\
&+ \pi_{16}\left(2l_2 c_3 \ddot{\theta}_2 + l_2 c_3 \ddot{\theta}_3 - 2l_2 s_3 \dot{\theta}_2 \dot{\theta}_3 - l_2 s_3 \dot{\theta}_3^2 + c_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 - l_2 c_2 s_{23}\dot{\theta}_1^2 - g s_{23}\right) + \pi_{17}\left(-l_2 c_2 \ddot{\theta}_1\right) \\
&+ \pi_{21}\left(l_2^2 + l_2 c_2\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 - g l_2 s_2\right) + \pi_{22}\left(-\dot{\theta}_1^2 s_{23} c_{23}\right) + \pi_{23}\left(c_{23} s_{23} \dot{\theta}_1^2\right) + \pi_{26}\left(\ddot{\theta}_2 + \ddot{\theta}_3\right) + \pi_{30}\, sign\left(\dot{\theta}_2\right) + \pi_{31}\dot{\theta}_2
\end{aligned}
$$

$$
\begin{aligned}
\tau_3 = &\ \pi_{12}\left(\ddot{\theta}_2 + \ddot{\theta}_3 + \dot{\theta}_1^2 c_{23} s_{23}\right) + \pi_{13}\left(\ddot{\theta}_2 + \ddot{\theta}_3 - \dot{\theta}_1^2 c_{23} s_{23}\right) + \pi_{15}\left(l_2 s_3 \ddot{\theta}_2 - l_2 c_3 \dot{\theta}_2^2 + s_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 + g c_{23}\right) \\
&+ \pi_{16}\left(l_2 c_3 \ddot{\theta}_2 + l_2 s_{23} \dot{\theta}_2^2 + c_{23}\left(l_1 - l_2 s_2\right)\dot{\theta}_1^2 - g s_{23}\right) + \pi_{18}\left(c_{23}^2 \dot{\theta}_1^2 - \dot{\theta}_1^2 s_{23}^2\right) + \pi_{19}\left(-\frac{1}{2}c_{23}\ddot{\theta}_1 - \frac{1}{2}s_{23}\dot{\theta}_1\left(\dot{\theta}_2 + \dot{\theta}_3\right)\right) \\
&+ \pi_{20}\left(-s_{23}\ddot{\theta}_1\right) + \pi_{26}\left(\ddot{\theta}_2 + \ddot{\theta}_3\right) + \pi_{23} c_{23} s_{23} \dot{\theta}_1^2 - \pi_{22} s_{23} c_{23} \dot{\theta}_1^2 + \pi_{31}\, sign\left(\dot{\theta}_3\right) + \pi_{32}\dot{\theta}_3
\end{aligned}
$$

Reference

1. Le QD, Kang H-J (2019) Real Implementation of an Active Fault-tolerant Control Based on Super Twisting Technique for a Robot Manipulator. In: Huang DS., Huang ZK., Hussain A. (eds) International Conference on Intelligent Computing. Springer, Cham, Nanchang, China, pp 294–305

2. Shen Q, Yue C, Goh CH, Wang D (2019) Active Fault-Tolerant Control System Design for Spacecraft Attitude Maneuvers with Actuator Saturation and Faults. IEEE Trans Ind Electron 66:3763–3772. https://doi.org/10.1109/TIE.2018.2854602

3. Zhang Q, Wang C, Su X, Xu D (2018) Observer-based terminal sliding mode control of non-affine nonlinear systems: Finite-time approach. J. Franklin Inst. 355:7985–8004

4. Wang R, Wang J (2013) Passive actuator fault-tolerant control for a class of overactuated nonlinear systems and applications to electric vehicles. IEEE Trans Veh Technol 62:972–985

5. Stefanovski JD (2018) Passive fault-tolerant perfect tracking with additive faults. Automatica 87:432–436

6. Benosman M, Lum K-Y (2010) Passive actuators' fault-tolerant control for affine nonlinear systems. IEEE Trans Control Syst Technol 18:152–163

7. Xu SS-D, Chen C-C, Wu Z-L (2015) Study of nonsingular fast terminal sliding-mode fault-tolerant control. IEEE Trans Ind Electron 62:3906–3913

8. Shen Q, Wang D, Zhu S, Poh EK (2015) Integral-Type Sliding Mode Fault-Tolerant Control for Attitude Stabilization of Spacecraft. IEEE Trans Control Syst Technol 23:1131–1138. https://doi.org/10.1109/TCST.2014.2354260

9. Shen Q, Jiang B, Cocquempot V, others (2013) Fuzzy logic system-based adaptive fault-tolerant control for near-space vehicle attitude dynamics with actuator faults. IEEE Trans Fuzzy Syst 21:289–300

10. Cao Y, Song YD (2018) Adaptive PID-like fault-tolerant control for robot manipulators with given performance specifications. Int J Control 7179:1–10.

https://doi.org/10.1080/00207179.2018.1468928

11. Xu SSD, Chen CC, Wu ZL (2015) Study of nonsingular fast terminal sliding-mode fault-tolerant control. IEEE Trans Ind Electron 62:3906–3913. https://doi.org/10.1109/TIE.2015.2399397

12. Jiang T, Khorasani K, Tafazoli S (2008) Parameter estimation-based fault detection, isolation and recovery for nonlinear satellite models. IEEE Trans Control Syst Technol 16:799–808

13. Edwards C, Spurgeon SK, Patton RJ (2000) Sliding mode observers for fault detection and isolation. Automatica 36:541–553

14. Han J, Zhang H, Wang Y, Zhang K (2018) Fault Estimation and Fault-Tolerant Control for Switched Fuzzy Stochastic Systems. IEEE Trans Fuzzy Syst 26:2993–3003. https://doi.org/10.1109/TFUZZ.2018.2799171

15. Bouibed K, Seddiki L, Guelton K, Akdag H (2014) Actuator and sensor fault detection and isolation of an actuated seat via nonlinear multi-observers. Syst Sci Control Eng 2:150–160. https://doi.org/10.1080/21642583.2014.888525

16. Lin X, Yao X (2021) Improved disturbance-observer-based fault-tolerant control for the linear system subject to unknown actuator faults and multiple disturbances. Int J Control 94:2730–2740

17. Zhang X, Zhu Z, Yi Y (2021) Anti-Disturbance Fault-Tolerant Sliding Mode Control for Systems with Unknown Faults and Disturbances. Electronics 10:1487

18. Han J, Zhang H, Wang Y, Liu Y (2015) Disturbance observer based fault estimation and dynamic output feedback fault-tolerant control for fuzzy systems with local nonlinear models. ISA Trans 59:114–124

19. Van M, Ge SS, Ren H (2017) Finite Time Fault-tolerant Control for Robot Manipulators Using Time Delay Estimation and Continuous Nonsingular Fast Terminal Sliding Mode Control. IEEE Trans Cybern 47:1681–1693. https://doi.org/10.1109/TCYB.2016.2555307

20. Yu X, Jiang J (2011) Hybrid Fault-Tolerant Flight Control System Design. IEEE Trans Control Syst Technol 20:1–16

21. Li C, Yao B, Wang Q (2018) Modeling and Synchronization Control of a Dual Drive Industrial Gantry Stage. IEEE/ASME Trans Mechatronics 23:2940–2951. https://doi.org/10.1109/TMECH.2018.2874876

22. Ren L, Mills JK, Sun D (2007) Experimental comparison of control approaches on trajectory tracking control of a 3-DOF parallel robot. IEEE Trans Control Syst

Technol 15:982–988. https://doi.org/10.1109/TCST.2006.890297

23. Shang W, Zhang B, Zhang B, et al (2019) Synchronization Control in the Cable Space for Cable-Driven Parallel Robots. IEEE Trans Ind Electron 66:4544–4554. https://doi.org/10.1109/TIE.2018.2864512

24. Cui R, Yan W (2012) Mutual synchronization of multiple robot manipulators with unknown dynamics. J Intell Robot Syst Theory Appl 68:105–119. https://doi.org/10.1007/s10846-012-9674-9

25. Le QD, Kang H-J (2020) Implementation of Fault-Tolerant Control for a Robot Manipulator Based on Synchronous Sliding Mode Control. Appl Sci 10:2534

26. Le QD, Kang H-J (2020) Finite-Time Fault-Tolerant Control for a Robot Manipulator Based on Synchronous Terminal Sliding Mode Control. Appl Sci 10:2998

27. Craig JJ (2004) Introduction to Robotics: Mechanics and Control 3rd. Prentice Hall 1:408

28. Pieper DL (1969) The kinematics of manipulators under computer control. Stanford University

29. Castaldi P, Mimmo N, Simani S (2014) Differential geometry based active fault-tolerant control for aircraft. Control Eng Pract 32:227–235

30. Badihi H, Zhang Y, Hong H (2015) Wind turbine fault diagnosis and fault-tolerant torque load control against actuator faults. IEEE Trans Control Syst Technol 23:1351–1372

31. Liu Z, Liu J, He W (2018) Robust adaptive fault-tolerant control for a linear cascaded ODE-beam system. Automatica 98:42–50

32. Niemann H, Stoustrup J (2005) Passive fault-tolerant control of a double inverted pendulum-a case study. Control Eng Pract 13:1047–1059

33. Gao Z, Ding SX, Ma Y (2007) Robust fault estimation approach and its application in vehicle lateral dynamic systems. Optim Control Appl Methods 28:143–156

34. Sadeghzadeh I, Mehta A, Chamseddine A, Zhang Y (2012) Active fault-tolerant control of a quadrotor uav based on gainscheduled pid control. In: Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference on. pp 1–4

35. Van M, Ge SS, Ren H (2017) Robust fault-tolerant control for a class of second-order nonlinear systems using an adaptive third-order sliding mode control. IEEE Trans Syst Man, Cybern Syst 47:221–228

36. Bahrami M, Naraghi M, Zareinejad M (2018) Adaptive super-twisting observer for fault reconstruction in electro-hydraulic systems. ISA Trans 76:235–245

37. Wen S, Chen MZQ, Zeng Z, et al (2017) Adaptive neural-fuzzy sliding-mode fault-tolerant control for uncertain nonlinear systems. IEEE Trans Syst Man, Cybern Syst 47:2268–2278

38. Levant A (1993) Sliding order and sliding accuracy in sliding mode control. Int J Control 58:1247–1263

39. Xu SS-D, Liu Y-K (2014) Study of Takagi-Sugeno fuzzy-based terminal-sliding mode fault-tolerant control. IET Control Theory Appl 8:667–674

40. Han Z, Zhang K, Yang T, Zhang M (2016) Spacecraft fault-tolerant control using adaptive non-singular fast terminal sliding mode. IET Control Theory Appl 10:1991–1999

41. Emel'Yanov S V, Korovin SK, Levantovskii L V (1986) Higher-order sliding modes in binary control systems. In: Soviet Physics Doklady. p 291

42. Mart\'\inez-Fuentes CA, Moreno JA, Fridman L (2018) Anti-Chattering Strategy using Twisting Controller. IFAC-ChaptersOnLine 51:384–389

43. Luo D, Xiong X, Jin S, Kamal S (2018) Adaptive gains of dual level to super-twisting algorithm for sliding mode design. IET Control THEORY Appl 12:2347–2356

44. Chan CW, Hua S, Hong-Yue Z (2006) Application of fully decoupled parity Eq. in fault detection and identification of DC motors. IEEE Trans Ind Electron 53:1277–1284

45. Fridman L, Shtessel Y, Edwards C, Yan X-G (2008) Higher-order sliding-mode observer for state estimation and input reconstruction in nonlinear systems. Int J Robust Nonlinear Control IFAC-Affiliated J 18:399–412

46. Dávila J, Salazar S (2017) Robust control of an uncertain UAV via high-order sliding mode compensation. IFAC-ChaptersOnLine 50:11553–11558

47. Kommuri SK, Lee S Bin, Veluvolu KC (2018) Robust Sensors-Fault-Tolerance With Sliding Mode Estimation and Control for PMSM Drives. IEEE/ASME Trans Mechatronics 23:17–28

48. Cruz-Zavala E, Moreno JA (2017) Homogeneous high order sliding mode design: a Lyapunov approach. Automatica 80:232–238

49. Cruz-Zavala E, Moreno JA (2018) Levant's Arbitrary Order Exact Differentiator: A Lyapunov Approach. IEEE Trans Automat Contr

50. Moreno JA, Osorio M (2012) Strict Lyapunov functions for the super-twisting algorithm. IEEE Trans Automat Contr 57:1035–1040

51. Kumari K, Chalanga A, Bandyopadhyay B (2016) Implementation of super-twisting control on higher order perturbed integrator system using higher order sliding mode observer. IFAC-ChaptersOnLine 49:873–878

52. Xiong X, Kamal S, Jin S (2018) Adaptive gains to super-twisting technique for sliding mode design. arXiv Prepr arXiv180507761

53. Liu J, Wang X (2012) Advanced Sliding Mode Control for Mechanical Systems: Design, Analysis and MATLAB Simulation. Springer Science & Business Media

54. Wang H, Bai W, Liu PX (2019) Finite-time adaptive fault-tolerant control for nonlinear systems with multiple faults. IEEE/CAA J Autom Sin 6:1417–1427

55. Fang H, Tian N, Wang Y, et al (2018) Nonlinear Bayesian estimation: from Kalman filtering to a broader horizon. IEEE/CAA J Autom Sin 5:401–417

56. Zhou M, Cao Z, Zhou M, et al (2020) Zonotoptic Fault Estimation for Discrete-Time LPV Systems with Bounded Parametric Uncertainty. IEEE Trans Intell Transp Syst 21:690–700. https://doi.org/10.1109/TITS.2019.2898853

57. Chen M (2017) Robust tracking control for self-balancing mobile robots using disturbance observer. IEEE/CAA J Autom Sin 4:458–465

58. Zhao Z, Yang Y, Zhang Y (2017) Fault-tolerant control using adaptive output integral-type sliding mode. J Franklin Inst 354:2648–2662

59. Lan J, Patton RJ (2017) Integrated Design of Fault-Tolerant Control for Nonlinear Systems Based on Fault Estimation and T--S Fuzzy Modeling. IEEE Trans Fuzzy Syst 25:1141–1154

60. Khalil HK, Praly L (2014) High-gain observers in nonlinear feedback control. Int J Robust Nonlinear Control 24:993–1015

61. Koren Y (1980) Cross-coupled biaxial computer control for manufacturing systems. J Dyn Syst Meas Control 102:265–272

62. Feng L, Koren Y, Borenstein J (1993) Cross-coupling motion controller for mobile robots. IEEE Control Syst 13:35–43

63. Ren L, Mills JK, Sun D (2007) Experimental comparison of control approaches on trajectory tracking control of a 3-DOF parallel robot. IEEE Trans Control Syst Technol 15:982–988

64. Sun D, Mills JK (2002) Adaptive synchronized control for coordination of multirobot assembly tasks. IEEE Trans Robot Autom 18:498–510

65. Gao G, Wen J, Liu X, Zhang Z (2013) Synchronous smooth sliding mode control for parallel mechanism based on coupling analysis. Int J Adv Robot Syst 10:173

66. Zhang C, Niu M, He J, et al (2017) Robust synchronous control of multi-motor integrated with artificial potential field and sliding mode variable structure. IEEE Access 5:197–207

67. Sun D (2016) Synchronization and control of multiagent systems. CRC Press

68. Jiang J, Yu X (2012) Fault-tolerant control systems: A comparative study between active and passive approaches. Annu Rev Control 36:60–72. https://doi.org/10.1016/j.arcontrol.2012.03.005

69. Jia Q, Chen W, Zhang Y, Li H (2015) Fault reconstruction and fault-tolerant control via learning observers in Takagi-Sugeno fuzzy descriptor systems with time delays. IEEE Trans Ind Electron 62:3885–3895. https://doi.org/10.1109/TIE.2015.2404784

70. Chu Z, Meng F, Zhu D, Luo C (2019) Fault reconstruction using a terminal sliding mode observer for a class of second-order MIMO uncertain nonlinear systems. ISA Trans. https://doi.org/10.1016/j.isatra.2019.07.024

71. Li B, Qin K, Xiao B, Yang Y (2019) Finite-time Extended State Observer based fault-tolerant output feedback control for attitude stabilization. ISA Trans 91:11–20. https://doi.org/10.1016/j.isatra.2019.01.039

72. Gao Z (2015) Fault estimation and fault-tolerant control for discrete-time dynamic systems. IEEE Trans Ind Electron 62:3874–3884

73. Doan QV, Le TD, Le QD, Kang H-J (2018) A neural network-based synchronized computed torque controller for three degree-of-freedom planar parallel manipulators with uncertainties compensation. Int J Adv Robot Syst 15:. https://doi.org/10.1177/1729881418767307

74. Jia H, Shang W, Xie F, et al (2019) Second-Order Sliding-Mode based Synchronization Control of Cable-Driven Parallel Robots. IEEE/ASME Trans Mechatronics PP:1. https://doi.org/10.1109/TMECH.2019.2960048

75. Feng Y, Yu X, Han F (2013) On nonsingular terminal sliding-mode control of nonlinear systems. Automatica 49:1715–1722. https://doi.org/10.1016/j.automatica.2013.01.051

76. Sun D (2003) Position synchronization of multiple motion axes with adaptive coupling control. Automatica 39:997–1005. https://doi.org/10.1016/S0005-1098(03)00037-2

77.   Sun D, Shao X, Feng G (2007) A model-free cross-coupled control for position synchronization of multi-axis motions: theory and experiments. IEEE Trans Control Syst Technol 15:306–314

78.   Zhao D, Li S, Gao F (2009) Finite time position synchronised control for parallel manipulators using fast terminal sliding mode. Int J Syst Sci 40:829–843. https://doi.org/10.1080/00207720902961022

79.   Hou H, Zhang Q (2016) Finite-time synchronization for second-order nonlinear multi-agent system via pinning exponent sliding mode control. ISA Trans 65:96–108. https://doi.org/10.1016/j.isatra.2016.07.004

80.   Guo Y, Song SM, Yan Z (2016) Finite-time coordination control for formation flying spacecraft without unwinding. Proc Inst Mech Eng Part G J Aerosp Eng 230:172–188. https://doi.org/10.1177/0954410015589978

81.   Zak M (1993) Introduction to terminal dynamics. Complex Syst 7:59